Vorlesung 401-0151-00L Lineare Algebra für D-ITET & RW

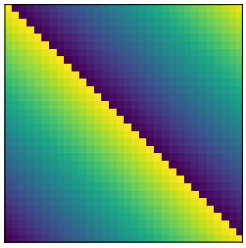
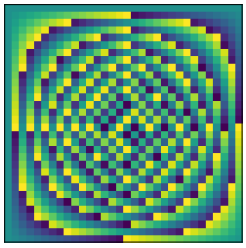
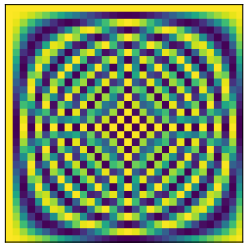
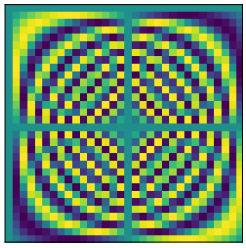
# Lineare Algebra

Dr. Vasile Gradinaru, SAM, ETH Zürich

HS 26

Generiert am 5. Oktober 2026

©Seminar für Angewandte Mathematik, ETH Zürich

Alle Rechte sind vorbehalten;
Vervielfältigungen jeglicher Art und Weise sind ohne die Zustimmung des Autors **nicht erlaubt**.
Es wird keine Verantwortung für eventuelle Fehler übernommen.
Die Verwendung dieser Notizen basiert auf eigener Verantwortung.
Unter Mitwirkung von: Filip Cvijanović, Thomas Herzog, Severin Nigg, Alexander Pietak
**Link** zur aktuellen Version dieses Skriptums




**Permanent in Bearbeitung**

Kleinere Änderungen in diesem Skriptum werden ständig vorgenommen, doch die Struktur kann als stabil betrachtet werden.
Generiert am 5. Oktober 2026

## Vorwort

Dieses Skript ist auf nachdrückliches Verlangen der Studierenden entstanden. Es widerspiegelt die Vorlesungen über Lineare Algebra, die ich seit Herbst 2014 den Erstsemestrigen aus den Studiengängen Elektrotechnik und Rechengestützte Wissenschaften (und bis 2020 auch Materialwissenschaften) der ETH Zürich unterrichte.
Bei der Entwicklung des Vorlesungsinhaltes habe ich mich am an der ETH meist verwendeten Lineare Algebra Buch von K. Nipp und D. Stoffer ‘‘Lineare Algebra’’, vdf Hochschulverlag, 5. Auflage (2002) orientiert. Gelegentlich habe ich das Skript ‘‘Lineare Algebra’’ von Martin Gutknecht einbezogen. Das Buch von Gilbert Strang ‘‘Introduction to Linear Algebra’’, 4. Auflage (2009) enthält sehr viele nützliche Beispiele. Aktuell gefällt mir das Buch von P. J. Olver und C. Shakiban ‘‘Applied Linear Algebra’’, 2nd ed. (2018), am meisten. Aus all diesen Quellen habe ich Beispiele übernommen. Ich verwende die Kürzel [NS15], [Gut07], [Str09] und [OS18] um die entsprechende Quelle kurz im Skript anzugeben.

# Inhaltsverzeichnis


**1 Lineare Gleichungssysteme** **6**

1.1 Motivation, Vektoren, Matrizen . . . . . 6

1.2 Gauss-Elimination . . . . . 16

1.2.1 Homogene Gleichungssysteme . . . . . 28

1.3 Operationen mit Matrizen . . . . . 29

1.4 Inverse einer Matrix . . . . . 37

1.5 Berechnung der Inverse . . . . . 40

1.6 LU-Zerlegung . . . . . 43

1.7 Orthogonale Matrizen . . . . . 54

1.8 QR-Zerlegung via orthogonale Transformationen . . . . . 62

**2 Lineare Räume** **72**

2.1 Raum und Unterraum . . . . . 72

2.2 Erzeugendensystem . . . . . 80

2.3 Lineare Unabhängigkeit und Basis . . . . . 90

2.4 Fundamentalsatz der linearen Algebra - Teil I . . . . . 95

2.5 Koordinaten und Basiswahl . . . . . 101

2.6 Basiswechsel und Koordinatentransformation . . . . . 103

**3 Lineare Abbildungen** **106**

3.1 Grundlagen . . . . . 106

3.2 Abbildungsmatrix . . . . . 108

3.3 Abbildungsmatrix bei Koordinatentransformationen . . . . . 114

3.4 Anwendung: Referenztransformationen . . . . . 118

3.4.1 Problemstellung . . . . . 118

3.4.2 Beispiel . . . . . 119

3.4.3 Erweiterte Matrix . . . . . 119

**4 Norm und Skalarprodukt in linearen Räumen 120**

4.1 Normierte lineare Räume . . . 120

4.2 Anwendung: Störungstheorem . . . 126

4.2.1 Einleitung . . . 126

4.2.2 Herleitung . . . 126

4.2.3 Anwendung . . . 128

4.2.4 Beispiel: Geradenschnittpunkt . . . 130

4.2.4.1 Gut Konditionierter Fall . . . 130

4.2.4.2 Schlecht Konditionierter Fall . . . 131

4.3 Skalarprodukt in linearen Räumen . . . 133

4.4 Gram-Schmidt-Algorithmus . . . 143

4.5 Projektoren . . . 149

4.6 Lineare Abbildungen auf Räume mit Skalarprodukt . . . 153

**5 Ausgleichsrechnung 157**

5.1 Ausgleichsrechnung (»Least Squares«) . . . 157

5.2 Lösung mit der Normalengleichung . . . 160

5.3 Lösung mit der QR-Zerlegung . . . 162

**6 Determinanten 164**

6.1 Definition und Eigenschaften . . . 164

6.2 Anwendungen . . . 170

**7 Eigenwertproblem 172**

7.1 Motivation . . . 172

7.2 Eigenwerte und Eigenvektoren . . . 179

7.3 Symmetrische Matrizen . . . 192

7.4 Symmetrisch positiv definite Matrizen . . . 198

7.4.1 Anwendungen . . . 200

7.5 Die Fourier Matrix . . . 205

7.6 Nicht-diagonalisierbare Matrizen . . . 213

7.6.1 Schur-Zerlegung . . . 214

7.6.2 Jordan-Form . . . 218

**8 Singulärwertzerlegung** **220**

8.1 Singulärwertzerlegung und Fundamentalsatz der linearen Algebra . . . 220

8.2 Anwendungen . . . 230

8.2.1 Ausgleichsrechnung . . . 230

8.2.2 Pseudoinverse . . . 230

8.2.3 Berechnung der Norm einer Matrix . . . 231

8.2.4 Numerischer Rang einer Matrix . . . 233

8.2.5 Speicherreduktion . . . 233

8.2.6 Datenkompression . . . 234

8.2.7 Trends in den Daten . . . 236

8.2.8 Totales lineares Ausgleichsproblem . . . 237

8.2.9 Lineare Ausgleichsrechnung mit linearen Nebenbedingungen . . . 238

8.2.10 Polarzerlegung . . . 239

**9 Python** **240**

9.1 Einleitung . . . 240

9.1.1 NumPy und SciPy . . . 241

9.1.2 Leistung . . . 241

9.1.3 Quellcode . . . 241

9.2 Lineare Gleichungssysteme . . . 242

9.2.1 Vektoren und Matrizen . . . 242

9.2.2 Lineare Gleichungssysteme . . . 243

9.2.3 Rang einer Matrix . . . 244

9.2.4 LU-Zerlegung . . . 244

9.2.5 QR-Zerlegung . . . 245

9.2.6 Weitere Zerlegungen . . . 246

9.2.7 Gauss-Elimination . . . 246

9.3 Lineare Räume . . . 259

9.3.1 Bild und Kern einer Matrix . . . 259

9.4 Normen und Skalarprodukte . . . 260

9.4.1 Vektor- und Matrixnormen . . . 260

9.4.2 Skalarprodukt . . . 260

9.5 Ausgleichsrechnung . . . 261

9.5.1 Pseudoinverse . . . 262

9.6 Determinante . . . 263

9.7 Eigenwertproblem . . . . . 264
9.7.1 Konditionszahl . . . . . 265
9.7.2 Zirkulante Matrix . . . . . 265
9.7.3 Fourier-Matrix . . . . . 265
9.7.4 Faltung . . . . . 266
9.8 Singulärwertzerlegung . . . . . 267
9.8.1 Trends in den Daten . . . . . 267

**Sachverzeichnis** **269**

Symbole . . . . . 269
Definitionen . . . . . 270
Beispiele . . . . . 273
Bemerkungen und Theoreme . . . . . 275
Pseudocode . . . . . 278
Literaturverzeichnis . . . . . 279

# Kapitel 1

# Lineare Gleichungssysteme

## Inhalt


**1.1 Motivation, Vektoren, Matrizen . . . . . 6**
**1.2 Gauss-Elimination . . . . . 16**
1.2.1 Homogene Gleichungssysteme . . . . . 28
**1.3 Operationen mit Matrizen . . . . . 29**
**1.4 Inverse einer Matrix . . . . . 37**
**1.5 Berechnung der Inverse . . . . . 40**
**1.6 LU-Zerlegung . . . . . 43**
**1.7 Orthogonale Matrizen . . . . . 54**
**1.8 QR-Zerlegung via orthogonale Transformationen . . . . . 62**


## 1.1 Motivation, Vektoren, Matrizen

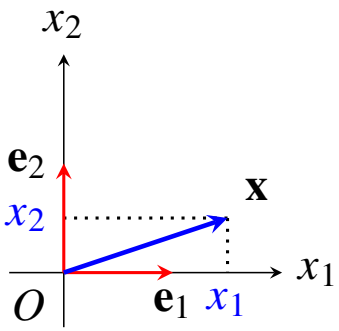
Das Lösen linearer Gleichungssysteme ist der Grundbaustein der linearen Algebra. Wenn wir über lineare Gleichungssysteme reden, definieren wir nebenbei auch Vektoren und Matrizen. Dabei sehen wir, warum das Matrix-Vektor-Produkt auf eine spezielle Art und Weise definiert ist. Das wichtige Konzept der linearen Kombination wird auch ersichtlich.

Vektoren und Matrizen sind den Meisten wahrscheinlich schon aus der Schule bekannt. Sie wurden oft in einem geometrischen Sinne verwendet. Beispielsweise dienen Vektoren, um Punkte in einem zwei- oder dreidimensionalen Raum zu beschreiben, und Matrizen, um diese ‘‘Punkte’’ zu manipulieren. Die Verwendung von Vektoren und Matrizen ist aber nicht nur auf die Geometrie des zwei- oder dreidimensionalen Raumes beschränkt, sondern hat auch viele Anwendungen in der Analysis, der Informationsverarbeitung und -übermittlung, der künstliche Intelligenz und vielen weiteren Bereichen der Wissenschaft und Technik. All dies wäre ohne der lineare Algebra und den ihr zugrunde liegenden Vektoren und Matrizen undenkbar.

Im folgenden Abschnitt werden Vektoren und Matrizen eingeführt, beginnend mit zwei- und dreidimensionalen Beispielen. In dieser Vorlesung sehen und schreiben wir Vektoren normalerweise nur als Spalten. Falls wir sie als Zeilen benötigen, dann müssen wir sie ‘‘umlegen’’, d. h. transponieren, was eine neue Notation braucht.

### 2 Dimensionen

Betrachten wir zuerst ein 2-dimensionales Koordinatensystem mit den Vektoren $\mathbf{e}_1$, $\mathbf{e}_2$ und $\mathbf{x}$.




Der hier definierte Vektor $\mathbf{x}$ hat die Koordinaten $x_1, x_2 \in \mathbb{R}$:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \in \mathbb{R}^2 .$$

Die beiden Vektoren $\mathbf{e}_1, \mathbf{e}_2$ sind spezielle Vektoren, denn sie liegen auf den beiden Achsen des Koordinatensystems (wir werden sie später als zwei Elemente der kanonischen Basis, auch Standardbasis genannt, ansehen):

$$\mathbf{e}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \qquad \mathbf{e}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

Wir sehen hier eine wichtige Eigenschaft von Vektoren, wenn wir den Zusammenhang zwischen $\mathbf{e}_1, \mathbf{e}_2$ und $\mathbf{x}$ genauer anschauen: der Vektor $\mathbf{x}$ kann aus den beiden Vektoren $\mathbf{e}_1, \mathbf{e}_2$ gebildet werden:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} x_1 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ x_2 \end{bmatrix} = x_1 \underbrace{\begin{bmatrix} 1 \\ 0 \end{bmatrix}}_{\mathbf{e}_1} + x_2 \underbrace{\begin{bmatrix} 0 \\ 1 \end{bmatrix}}_{\mathbf{e}_2} = x_1\mathbf{e}_1 + x_2\mathbf{e}_2$$

hier sind $x_1, x_2$ *Skalare* in $\mathbb{R}$ und $\mathbf{e}_1, \mathbf{e}_2$ *Vektoren* in $\mathbb{R}^2$.

Dies werden wir *Linearkombination* von $\mathbf{e}_1$ und $\mathbf{e}_2$ nennen: $\mathbf{x}$ ist die Summe der zwei gestreckten (oder gestauchten) Vektoren $x_1\mathbf{e}_1$ und $x_2\mathbf{e}_2$. Wir sagen, dass sich $\mathbf{x}$ als eine Linearkombination von $\mathbf{e}_1$ und $\mathbf{e}_2$ schreiben lässt oder dass $\mathbf{x}$, $\mathbf{e}_1$ und $\mathbf{e}_2$ linear abhängig sind.

Dieses Beispiel kann auf zwei einfache Arten erweitert werden. Die erste Änderung besteht darin, die in die Rechnung einbezogenen Zahlen zu ändern.

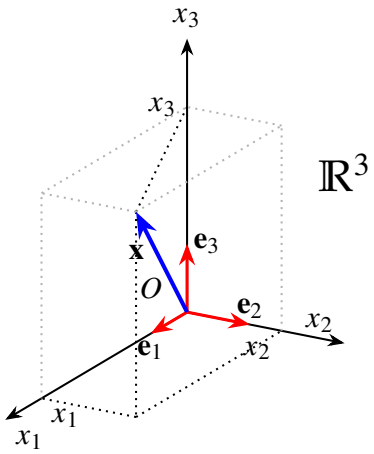
Wir können überall $\mathbb{R}$, d. h. die Menge der reellen Zahlen, mit $\mathbb{C}$, d. h. mit der Menge der komplexen Zahlen, ersetzen. Dann sind die Vektoren $\mathbf{x}$, $\mathbf{e}_1$ und $\mathbf{e}_2$ nicht in $\mathbb{R}^2$, sondern in $\mathbb{C}^2$. Auch die *Skalare* sind in diesem Fall nicht mehr in $\mathbb{R}$, sondern in $\mathbb{C}$.

Die zweite Erweiterung besteht darin, die Dimension des Koordinatensystems zu ändern: Statt wie hier nur zwei Dimensionen zu verwenden, werden wir drei und sogar beliebig viele ($n \in \mathbb{N}$) Dimensionen betrachten.

## 3 Dimensionen

Wir arbeiten nun in einem reellen 3-dimensionalen Koordinatensystem: alle Vektoren sind in $\mathbb{R}^3$.

Seien die vier Vektoren $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$ und $\mathbf{x}$ in diesem Koordinatensystem gegeben.




Die Koordinaten des Vektors $\mathbf{x}$ sind $x_1, x_2, x_3$:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \in \mathbb{R}^3 .$$

Die Vektoren $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$ liegen, wie in dem 2-dimensionalen Beispiel, auf den Koordinatenachsen. Da nun in drei Dimensionen gearbeitet wird, sind es auch drei Vektoren:

$$\mathbf{e}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \qquad \mathbf{e}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \qquad \mathbf{e}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

Hier kann dasselbe wie beim 2-dimensionalen Beispiel gemacht werden. Wir können den Vektor $\mathbf{x}$ durch die Skalare $x_1, x_2, x_3$ und die Vektoren $\mathbf{e}_1, \mathbf{e}_2$ und $\mathbf{e}_3$ darstellen:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = x_1 \underbrace{\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}}_{\mathbf{e}_1} + x_2 \underbrace{\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}}_{\mathbf{e}_2} + x_3 \underbrace{\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}}_{\mathbf{e}_3} .$$

Die Vektoren $\mathbf{e}_1$, $\mathbf{e}_2$ und $\mathbf{e}_3$ sind spezielle Vektoren. Sie bilden die sogenannte Standardbasis (auch kanonische Basis) in $\mathbb{R}^3$. Was genau eine Basis ist und wieso diese hier Standardbasis genannt wird, wird in einem späteren Kapitel besprochen. Wir halten jedoch fest, dass die Vektoren $\mathbf{e}_1$, $\mathbf{e}_2$ aus diesem Beispiel aus den gleichnamigen Vektoren des vorigen Beispiels entstehen, indem man diese mit $0$ erweitert.

Wie in $2$ Dimensionen kann auch hier mit den komplexen Zahlen aus $\mathbb{C}$ anstelle der reellen Zahlen aus $\mathbb{R}$ gerechnet werden.

Eine Summe skalierter Vektoren (d. h. jeweils ein Produkt eines Skalars und eines Vektors) nennen wir *Linearkombination von Vektoren*. Sie ist einer der wichtigsten Begriffe der linearen Algebra.

## Linearkombination

**Definition 1.1.0.1. Linearkombination**

Die Linearkombination von Vektoren $\mathbf{v}_1, \ldots, \mathbf{v}_n$ ist:

$$c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \ldots + c_n\mathbf{v}_n ,$$

wobei die $c_1, c_2, \ldots, c_n$ Skalare in $\mathbb{R}$ oder $\mathbb{C}$ sind.

In den vorherigen Beispielen haben wir gesehen: Der Vektor $\mathbf{x}$ wurde als Summe der Multiplikationen von $x_1, x_2, x_3$ mit Vektoren aus der Standardbasis $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$ dargestellt, demzufolge war das eine Linearkombination.

## $n$ Dimensionen

Wir erweitern die vorherigen Überlegungen auf eine beliebige Dimension $n \in \mathbb{N}$. Auch wenn wir solche Vektoren nicht mehr graphisch darstellen können, sind Vektoren mit $n$ Einträgen für grosse $n$ sehr natürlich. Um zum Beispiel ein diskretes Signal (wie z. B. Temperatur- oder Druckmessungen) zu speichern, braucht man $n$ Plätze. Wenn diese Plätze beliebige reelle Zahlen enthalten sollen, dann werden wir nicht mit $\mathbb{R}^2$ oder $\mathbb{R}^3$ auskommen und wir brauchen $\mathbb{R}^n$.

Erweitern wir also das Koordinatensystem des vorherigen Beispiels, als die Dimension $n = 3$ war, auf ein $n$-dimensionales Koordinatensystem. In diesem Koordinatensystem besitzt ein beliebiges aber festes $\mathbf{x} \in \mathbb{R}^n$ die Koordinaten $x_1, x_2, \ldots, x_n$:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} .$$

Wie auch in den vorherigen Beispielen kann dieser Vektor als Linearkombination der Vektoren $\mathbf{e}_1, \cdots, \mathbf{e}_n$,

welche die Standardbasis des $\mathbb{R}^n$ bilden, mit den Skalaren $x_1, x_2, \ldots, x_n$ dargestellt werden:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = x_1\mathbf{e}_1 + x_2\mathbf{e}_2 + \ldots + x_n\mathbf{e}_n \,.$$

**Definition 1.1.0.2. Standardbasis**

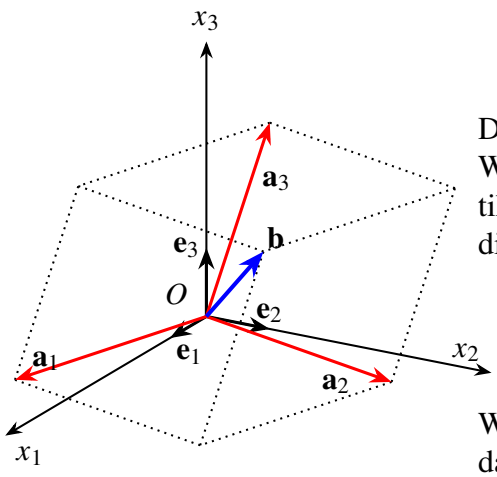
Die *Standardbasis* (oder auch die *kanonische Basis*) in diesem $n$-dimensionalen Koordinatensystem besteht aus folgenden $n$ Vektoren:

$$\mathbf{e}_1 = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \mathbf{e}_2 = \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}, \quad \ldots, \quad \mathbf{e}_j = \begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix}, \quad \ldots, \quad \mathbf{e}_n = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}.$$

## Anwendung der Linearkombination: Die Superposition von Feldern

Eigentlich kennen wir die Linearkombination bereits aus der Physik; dort ist sie als *Superposition, bzw. Überlagerung, von Kräften und Feldern* bekannt.

Nehmen wir an, wir haben drei Kräfte $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$, die auf ein Objekt wirken. Wir messen die resultierende Gesamtkraft $\mathbf{b}$:




$$\mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \in \mathbb{R}^3 \,.$$

Diese gemessene Kraft ist die Überlagerung der Kräfte $\mathbf{a}_1$, $\mathbf{a}_2$, $\mathbf{a}_3$. Was in der Physik Überlagerung genannt wird, heisst in der Mathematik Linearkombination; d. h. der Vektor $\mathbf{b}$ ist die Linearkombination dieser Kräfte:

$$\mathbf{b} = x_1\mathbf{a}_1 + x_2\mathbf{a}_2 + x_3\mathbf{a}_3 \,.$$

Wir wollen im Folgenden die drei Skalare $x_1$, $x_2$, $x_3$ bestimmen, so dass wir die vorgegebene Gesamtkraft $\mathbf{b}$ realisieren. Dazu müssen wir jede der vorgegebenen Kräfte $\mathbf{a}_1$, $\mathbf{a}_2$, $\mathbf{a}_3$ mit den entsprechenden Faktoren $x_1$, $x_2$, $x_3$ verstärken oder abschwächen.

**Beispiel 1.1.0.3 (Eine Linearkombination von Vektoren liest man als lineares Gleichungssystem)** [1]
Nehmen wir an, dass die Kräfte $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$ so gegeben sind:

$$\mathbf{a}_1 = \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix}, \ \mathbf{a}_2 = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}, \ \mathbf{a}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

Sei $\mathbf{b}$ eine beliebige Linearkombination dieser Vektoren, also $\mathbf{b} = x_1\mathbf{a}_1 + x_2\mathbf{a}_2 + x_3\mathbf{a}_3$.
Daraus ergibt sich:

$$\begin{aligned} \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} &= x_1 \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix} + x_2 \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} + x_3 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} x_1 \\ -x_1 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ x_2 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ x_3 \end{bmatrix}. \end{aligned}$$

Diese Vektoren können nun komponentenweise zusammenaddiert werden und man erhält ein Gleichungssystem mit $x_1$, $x_2$ und $x_3$ als Unbekannten:

$$\begin{aligned} x_1 \qquad\quad &= b_1 \\ -x_1 + x_2 \qquad &= b_2 . \\ x_2 + x_3 \ &= b_3 \end{aligned}$$

Die Überlagerung von Kräften ist eine Linearkombination der Kraftvektoren mit den Skalaren $x_1, \ldots, x_3$. Diese Linearkombination kann man als ein Gleichungssystem mit den Skalaren als Unbekannten lesen; die Koeffizienten der Unbekannten sind spaltenweise die Koordinaten der Kraftvektoren.

⌟

## Im Allgemeinen

Wir ersetzten nun das vorherige Zahlenbeispiel mit dem allgemeinen Fall in 3 Dimensionen. Seien drei allgemeine Vektoren $\mathbf{a}_1$, $\mathbf{a}_2$, $\mathbf{a}_3$ gegeben:

$$\mathbf{a}_1 = \begin{bmatrix} a_{11} \\ a_{21} \\ a_{31} \end{bmatrix}, \ \mathbf{a}_2 = \begin{bmatrix} a_{12} \\ a_{22} \\ a_{32} \end{bmatrix}, \ \mathbf{a}_3 = \begin{bmatrix} a_{13} \\ a_{23} \\ a_{33} \end{bmatrix}.$$

Aus diesen Vektoren bilden wir durch Linearkombination einen Vektor $\mathbf{b}$, dessen Einträge $b_1$, $b_2$ und $b_3$ gegeben sind:

$$\mathbf{b} = x_1\mathbf{a}_1 + x_2\mathbf{a}_2 + x_3\mathbf{a}_3 , \ \text{mit } x_1, \ x_2, \ x_3 \ \in \mathbb{R} .$$

Wird diese Linearkombination ausgerechnet, so erhält man folgendes Gleichungssystem:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 &= b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 &= b_2 . \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 &= b_3 \end{cases}$$

Von nun an schreiben wir lineare Gleichungssysteme kompakter: In einer neuen Notation werden die Einträge der Vektoren $\mathbf{a}_1$, $\mathbf{a}_2$, $\mathbf{a}_3$ nebeneinander, und von eckigen Klammern eingeschlossen, geschrieben. Dieses

[1] Beispiel inspiriert durch Str09, Kapitel 1.3, Seiten 22-27.

neu entstehende Konstrukt wird Matrix genannt. Matrizen werden mit fetten Grossbuchstaben notiert (zum Beispiel $\mathbf{A}$, $\mathbf{B}$ oder $\mathbf{C}$). Die Skalare $x_1,\ x_2,\ x_3$ werden in einem einzelnen Vektor zusammengefasst, dasselbe gilt für $b_1,\ b_2,\ b_3$.

In diesem Fall notieren wir die entstehende Matrix mit $\mathbf{A}$, den ersten Vektor mit den Einträgen $x_1,\ x_2,\ x_3$ mit $\mathbf{x}$ und den zweiten Vektor mit den Einträgen $b_1,\ b_2,\ b_3$ mit $\mathbf{b}$. So sieht nun das lineare Gleichungssystem von oben etwas kompakter aus:

$$\underbrace{\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}}_{\mathbf{A}\text{ Matrix} \overset{\text{def}}{=} \text{Spalten (Vektoren) } \mathbf{a}_1,\ \mathbf{a}_2,\ \mathbf{a}_3 \text{ zusammen}} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}.$$

Oder noch kompakter:

$$\mathbf{A}\mathbf{x} = \mathbf{b}\,.$$

Das obige *lineare Gleichungssystem (LGS)* ist nun eine einzige Gleichung, jedoch ist die rechte Seite und die Unbekannte jeweils ein Vektor und der Faktor vor der Unbekannten ist eine Matrix.

**Definition 1.1.0.4. Lineares Gleichungssystem (LGS)**

Ein lineares Gleichungssystem (LGS) wird kurz geschrieben als:

$$\mathbf{A}\,\mathbf{x} = \mathbf{b}\,,$$

wobei $\mathbf{A}$ die Koeffizientenmatrix, $\mathbf{x}$ die Unbekannte und $\mathbf{b}$ die rechte Seite ist.

**Bemerkung 1.1.0.5 (Matrix-Vektor-Multiplikation)**
$\mathbf{A}\,\mathbf{x}$ sieht wie eine Multiplikation aus. In der Tat ist es eine Matrix-Vektor-Multiplikation. Viele wissen wahrscheinlich noch aus der Schulzeit, wie diese Multiplikation berechnet wird. Sie haben sich wohl gefragt, wieso sie auf eine auf den ersten Blick willkürlich erscheinende Art und Weise definiert ist. Wir haben nun eine Begründung dafür gesehen. Denn die Matrix-Vektor-Multiplikation ist in dieser neuen Notation für lineare Gleichungssysteme enthalten und kann auch aus ihr abgelesen werden. Ausserdem ist das eine typische Erscheinung in jeder physikalischen Theorie, in der das Superpositionsprinzip gilt.
Das System gibt einem direkt die Definitionen von $b_1,\ b_2,\ b_3$:

$$\begin{aligned} b_1 &= a_{11}x_1 + a_{12}x_2 + a_{13}x_3 \\ b_2 &= a_{21}x_1 + a_{22}x_2 + a_{23}x_3 \\ b_3 &= a_{31}x_1 + a_{32}x_2 + a_{33}x_3\,. \end{aligned}$$

Diese sind die Einträge des Vektors $\mathbf{b}$, welcher das Resultat der Matrix-Vektor-Multiplikation $\mathbf{A}\mathbf{x}$ ist. Diese Gleichungen definieren also die Multiplikation. Daraus kann folgendes Schema abgeleitet werden:

$$\mathbf{A}\,\mathbf{x} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 \end{bmatrix}.$$

Achten Sie dabei darauf, wie das Farbschema ist: Jede **Zeile** in der Matrix hat eine eigene Farbe bekommen, die dann in der Berechnung des Ergebnisses wiederzusehen ist. Dieses Schema ist nützlich, wenn wir solch eine Multiplikation von Hand berechnen müssen. Sonst ist das, was bei der Definition der Matrix-Vektor-Multiplikation gesehen wurde, viel wichtiger:

**Definition 1.1.0.6. Matrix-Vektor-Multiplikation**

Das Ergebnis **b** der Multiplikation einer Matrix **A** mit einem Vektor **x** (in dieser Reihenfolge) ist die **Linearkombination der Spalten der Matrix A mit den Koeffizienten aus dem Vektor x**. Diese Operation ist nur dann möglich, wenn die Matrix **A** ebenso viele Spalten besitzt, wie der Vektor **x** Einträge hat.

Kurzgefasst schreiben wir:

$$\mathbf{A}\mathbf{x} = x_1\mathbf{a}_1 + x_2\mathbf{a}_2 + x_3\mathbf{a}_3 \,.$$

⌟

**Beispiel 1.1.0.7 (Elimination und Inverse)** [2]

Im folgenden Beispiel wird das LGS aus 1.1.0.3 nach der Unbekannten in **x** aufgelöst. Viele haben wahrscheinlich schon in der Schulzeit Gleichungssysteme gelöst. Um den Übergang zu der Matrix-Vektor-Darstellung zu vereinfachen, sind im folgenden Beispiel beide Darstellungen gegeben.

Um ein lineares Gleichungssystem in der traditionellen Darstellung zu lösen, wird in jedem Schritt versucht, durch die Addition der Gleichungen, die Koeffizienten der Unbekannten null werden zu lassen. Eine Addition alleine reicht im Allgemeinen dazu nicht. In der Regel muss man die Gleichungen zuerst mit einem passenden Skalar multiplizieren und dann die Gleichungen addieren. So wird z. B. die Unbekannte $x_1$ zunächst aus der zweiten und dann aus der dritten Gleichung eliminiert. Das wird so lange gemacht, bis nur noch eine Unbekannte pro Gleichung vorhanden ist. Bei der neuen Notation wird das Gleiche gemacht. Anstatt dass man Gleichungen addiert, werden einfach Zeilen der Matrix **A** aufeinander addiert. Hier sind nun die Schritte geschrieben: links mit ganzen Gleichungen, rechts mit Matrixnotation:

$$\begin{cases} x_1 & & & = b_1 & |+1 \\ -x_1 & +x_2 & & = b_2 & \\ & x_2 & +x_3 & = b_3 & \end{cases} \qquad \begin{matrix} \\ \\ +1| \\ \\ \end{matrix}\begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}\mathbf{x} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

$$\begin{cases} x_1 & & & = b_1 & \\ & +x_2 & & = b_1 + b_2 & |-1 \\ & x_2 & +x_3 & = b_3 & \end{cases} \qquad \begin{matrix} \\ \\ -1| \\ \\ \end{matrix}\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}\mathbf{x} = \begin{bmatrix} b_1 \\ b_1 + b_2 \\ b_3 \end{bmatrix} \,.$$

$$\begin{cases} x_1 & & & = b_1 & \\ & x_2 & & = b_1 + b_2 & \\ & & x_3 & = -b_1 - b_2 + b_3 & \end{cases} \qquad \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\mathbf{x} = \begin{bmatrix} b_1 \\ b_1 + b_2 \\ -b_1 - b_2 + b_3 \end{bmatrix}$$

Durch dieses wiederholte Addieren von Gleichungen hat man in diesem Fall eindeutige Lösungen für $x_1$, $x_2$, $x_3$ erhalten:

$$\begin{cases} x_1 = & b_1 & & \\ x_2 = & b_1 & +b_2 & \\ x_3 = & -b_1 & -b_2 & +b_3 \end{cases} \,.$$

Wir sehen hier auch ein lineares Gleichungssystem mit unbekanntem **b** und rechter Seite **x**, es kann also auch in der Matrix Notation geschrieben werden:

$$\mathbf{x} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ -1 & -1 & 1 \end{bmatrix}}_{\mathbf{A}^{-1}}\begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \,.$$

[2] Beispiel inspiriert durch Str09, Kapitel 1.3, Seiten 22-27.

Diese neue Matrix $\mathbf{A}^{-1}$ wird Inverse der Matrix $\mathbf{A}$ genannt:

$$\mathbf{A}\,\mathbf{x} = \mathbf{b} \iff \mathbf{x} = \mathbf{A}^{-1}\mathbf{b}\,.$$

Was genau eine Inverse ist und wie die Multiplikation von Matrizen funktioniert, wird zu einem späteren Zeitpunkt besprochen. Jedoch sehen wir bereits in diesem Beispiel, dass diese Begriffe sehr natürlich sind.

Wir sind nun in der Lage $\mathbf{x}$ für beliebige $\mathbf{b}$ zu bestimmen.
Sei das gemessene $\mathbf{b}$:

$$\mathbf{b} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}.$$

Dann kann das dazugehörige $\mathbf{x}$ durch eine einfache Matrix-Vektor-Multiplikation bestimmt werden.

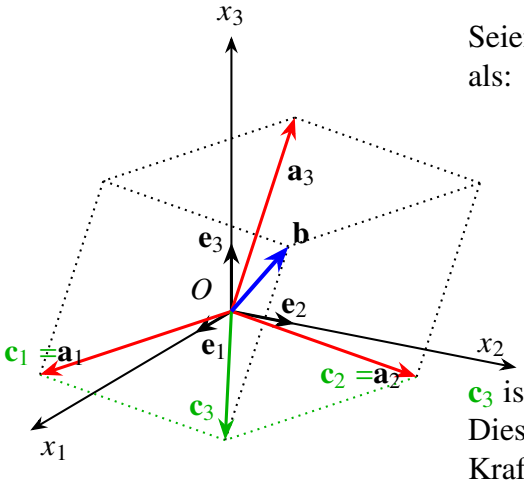
$$\mathbf{x} = \mathbf{A}^{-1}\mathbf{b} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ -1 & -1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ -1 \end{bmatrix}.$$

Folgende zwei Fragen werden an dieser Stelle kurz ohne Begründung beantwortet. Wieso die Antworten stimmen, wird erst später klar werden.

- Gibt es immer eine Lösung?
  *Für dieses $\mathbf{A}$, ja, denn wir können immer mit diesem $\mathbf{A}^{-1}$ multiplizieren.*
- Gibt es für jede Matrix $\mathbf{A}$ eine Inverse $\mathbf{A}^{-1}$?
  *Nein!*

⌟

**Beispiel 1.1.0.8 (Linear abhängige Vektoren führen zu besonderen linearen Gleichungssystemen)** [3]
Im folgenden Beispiel werden wir ein LGS besprechen, welches nicht immer eine Lösung hat. Im Gegenteil, wir werden sehen, dass es entweder keine oder unendlich viele Lösungen besitzt, abhängig von der rechten Seite des Systems.




Seien drei Kräfte $\mathbf{c}_1$, $\mathbf{c}_2$, $\mathbf{c}_3$, welche auf ein Objekt wirken, definiert als:

$$\mathbf{c}_1 = \mathbf{a}_1$$
$$\mathbf{c}_2 = \mathbf{a}_2$$
$$\mathbf{c}_3 = \mathbf{a}_1 + \mathbf{a}_2\,.$$

$\mathbf{c}_3$ ist also eine Linearkombination von $\mathbf{a}_1$ und $\mathbf{a}_2$.
Diese drei Kräfte wirken auf ein Objekt und führen zu einer resultieren Kraft $\mathbf{b}$.

Geometrisch betrachtet bedeutet dies, dass $\mathbf{c}_3$ in der Ebene liegt, welche von $\mathbf{c}_1$ und $\mathbf{c}_2$ aufgespannt wird.

[3] Beispiel inspiriert durch Str09, Kapitel 1.3, Seiten 22-27.

Daraus folgt: Wenn $\mathbf{b}$ nicht auch in dieser Ebene liegt, kann er nicht als Linearkombination von $\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3$ dargestellt werden. Falls $\mathbf{b}$ aber in dieser Ebene liegt, so kann es auf unendlich viele Arten als Linearkombination von $\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3$ dargestellt werden, z. B. $\mathbf{b} = x_1\mathbf{c}_1 + x_2\mathbf{c}_2 = (x_1 - x_3)\mathbf{c}_1 + (x_2 - x_3)\mathbf{c}_2 + x_3\mathbf{c}_3$ mit beliebigem $x_3 \in \mathbb{R}$.

Diese geometrische Intuition zeigen wir nun auch noch algebraisch. Wir machen ähnliche Eliminationsschritte wie im vorherigen Beispiel, denn nur die dritte Spalte ist anders:

$$\begin{array}{l} \\ +1| \\ \\ \end{array}\begin{bmatrix} 1 & 0 & 1 \\ -1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}\mathbf{x} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

$$\begin{array}{l} \\ -1| \\ \\ \end{array}\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix}\mathbf{x} = \begin{bmatrix} b_1 \\ b_1 + b_2 \\ b_3 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}\mathbf{x} = \begin{bmatrix} b_1 \\ b_1 + b_2 \\ -b_1 - b_2 + b_3 \end{bmatrix}.$$

Wenn wir uns die letzte Gleichung $0 = -b_1 - b_2 + b_3$ genauer ansehen, so stellen wir fest:

- falls $-b_1 - b_2 + b_3 \neq 0$, dann gibt es keine Lösung;
- falls $-b_1 - b_2 + b_3 = 0$, dann gibt es unendlich viele Lösungen, denn die letzte Gleichung ist für beliebiges $x_3$ erfüllt.

Die letzte Gleichung $-b_1 - b_2 + b_3 = 0$ wird *Kompatibilitätsbedingung (KB)* genannt, da sie entscheidet, ob das LGS Lösungen haben wird oder nicht. In unserem Fall beschreibt die Kompatibilitätsbedingung eine 2-dimensionale Ebene in einem 3-dimensionalen Raum.

Wenn diese Kompatibilitätsbedingung erfüllt ist, so können wir die letzte Gleichung in dem LGS weglassen (es wäre die Gleichung $0 = 0$):

$$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_1 + b_2 \end{bmatrix}.$$

Wir haben nun die Gleichungen: $x_2 + x_3 = b_1 + b_2$ und $x_1 + x_3 = b_1$. Der Wert von $x_3$ ist frei wählbar und wird als *freie Variable* bezeichnet. Die Werte von $x_1$ und $x_2$ lassen sich einfach ausrechnen als

$$\begin{aligned} x_2 + x_3 = b_1 + b_2 \implies x_2 &= b_1 + b_2 - x_3 \\ x_1 &= b_1 - x_3\,. \end{aligned}$$

$x_1, x_2$ sind beide von $x_3$ abhängig, $x_3$ ist jedoch beliebig und deswegen hat das System unendlich viele Lösungen:

$$x_3\text{beliebig} \implies \infty \text{ viele Lösungen}$$

Wir bemerken auch, dass diese unendliche Menge der Lösungen sich in diesem Fall durch einen einzigen reellen Parameter ($x_3$) beschreiben lässt.

**Bemerkung 1.1.0.9 (Elimination entspricht Linearkombinationen von Gleichungen/Zeilen)**
**Die Eliminationsschritte führen nur Linearkombinationen von Gleichungen durch**: Wir multiplizieren eine Gleichung mit einer Konstanten und addieren das Ergebnis zu einer anderen Gleichung. Wenn wir mit einer (oder mehreren) erfüllten Kompatibilitätsbedingung(en) $0 = 0$ enden, bedeutet dies, dass eine (oder mehrere) Gleichung(en) eine Linearkombination von den anderen (nicht zur Kompatibilitätsbedingung gehörenden) Gleichungen ist (sind). Die zu erfüllten Kompatibilitätsbedingungen gehörenden Gleichungen entscheiden über die Existenz der Lösungen. Auf die zu erfüllten Kompatibilitätsbedingungen gehörenden Gleichungen kann man von Anfang an verzichten; sie bringen keine Information zur Lösung. Man kann aber nicht darauf verzichten, wenn die entstandene rechte Seite der Kompatibilitätsbedingung nicht $0$ ist, denn das bedeutet, dass es eben keine Lösung gibt.
Wenn die ursprüngliche rechte Seite nur Nullen enthält, dann sind Linearkombinationen davon auch nur Nullen. Also sind die Gleichungen, die den Kompatibilitätsbedingungen entsprechen, Linearkombinationen der restlichen Gleichungen.
Wenn wir die Koeffizienten der Unbekannten in jeder Gleichung in je einem Vektor schreiben, erhalten wir also, dass die Vektoren, die zu Gleichungen aus den Kompatibilitätsbedingungen gehörenden Linearkombinationen der restlichen Vektoren sind. **Die den Kompatibilitätsbedingungen entsprechenden Zeilen der Koeffizientenmatrix sind also Linearkombinationen der anderen Zeilen.** Das ist eine Bemerkung, die wir später brauchen werden. Zurzeit fokussieren wir uns aber hauptsächlich auf die Spalten der Matrix.
In diesem Beispiel starteten wir mit einer Matrix, in der der letzte Spaltenvektor eine Linearkombination der ersten zwei Spaltenvektoren ist. Die Elimination zeigt uns, dass auch die letzte Zeile eine Linearkombination der restlichen Zeilen ist. ⌟

**Beispiel 1.1.0.10 (Kompatibilitätsbedingungen)**
Schauen wir uns dieses Konzept noch mit Zahlen für die rechte Seite an.

Sei $\mathbf{Ax} = \mathbf{b}$ ein lineares Gleichungssystem mit der Matrix $\mathbf{A}$ aus dem Beispiel 1.1.0.8.

Sei $\mathbf{b}$ gegeben als $\mathbf{b} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$.
Die Kompatibilitätsbedingung $-b_1 - b_2 + b_3$ für dieses $\mathbf{b}$ ist

$$-1 - 1 + 1 = -1 \neq 0\,.$$

Die Kompatibilitätsbedingung ist offensichtlich nicht erfüllt und das LGS hat somit keine Lösungen.

Sei ein anderer Vektor $\mathbf{b}'$ gegeben als $\mathbf{b}' = \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix}$.
Für diesen neuen Vektor sieht die Kompatibilitätsbedingen folgendermassen aus:

$$-1 - 1 + 2 = 0\,.$$

Sie ist also erfüllt und deswegen gibt es unendlich viele $\mathbf{x}$, welche, wenn mit $\mathbf{A}$ multipliziert, dieses $\mathbf{b}'$ ergeben.

Diese Lösungen lassen sich schreiben als

$$\begin{aligned} x_1 &= 1 - x_3 \\ x_2 &= 2 - x_3 \\ x_3 &= \text{beliebig}\,. \end{aligned}$$

⌟

⌟

## 1.2 Gauss-Elimination

Die Gauss-Elimination ist ein Algorithmus, welcher die Lösung(en) linearer Gleichungssysteme bestimmt. Es ist nichts anderes als das, was wir im vorigen Abschnitt gesehen haben, nur systematisch und allgemein durchgeführt. Wir schauen uns zuerst ein etwas kompliziertes aber kleines LGS $\mathbf{A}\,\mathbf{x} = \mathbf{b}$ an:

$$\underbrace{\begin{bmatrix} 1 & 1 & 0 \\ 2 & 1 & -1 \\ 3 & -1 & -1 \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}}_{\mathbf{x}} = \underbrace{\begin{bmatrix} 4 \\ 1 \\ -3 \end{bmatrix}}_{\mathbf{b}}.$$

Falls die Matrix $\mathbf{A}$ nur Einträge auf der Hauptdiagonale hätte (anders gesagt: alle Einträge, die nicht auf der Hauptdiagonale sind, haben den Wert Null), so wäre dieses lineare Gleichungssystem einfach zu lösen. Das Gleiche gilt auch, wenn alle Einträge unterhalb der Hauptdiagonale Null sind, da wir dann $x_3$ einfach bestimmen können und dann in die zweite Zeile einsetzen können, um $x_2$ zu bestimmen. Diese beiden bestimmten Werte werden dann in die erste Zeile eingesetzt, um $x_1$ zu erhalten. Dieses Prozedere wird *Rücksubstitution* genannt, da Unbekannte auf einer Zeile durch ihre, auf der folgenden Zeile, berechneten Werte substituiert werden.

Das Ziel der Gauss-Elimination ist es, durch das Kombinieren von Gleichungen, ein äquivalentes lineares Gleichungssystem zu kreieren, welches nur Nullen unterhalb der Hauptdiagonale hat. Das Ziel wird erreicht, indem eine Zeile der Matrix mit einem Faktor multipliziert wird und dann zu einer anderen Zeile addiert wird.

Bei der Berechnung wird noch eine weitere Matrix mitgeführt, die sogenannte Protokollmatrix. Diese hat dieselben Dimensionen, wie die Systemmatrix und besteht am Anfang nur aus Einsen auf der Hauptdiagonale und sonst Nullen (die *Identitätsmatrix*). In ihr wird das Negative der Faktoren, mit welchen die Zeilen multipliziert werden, eingetragen. Diese Zahlen werden an den Stellen der Protokollmatrix eingefügt, die den Stellen in der Systemmatrix entsprechen, an denen gerade Nullen entstehen. Wir sehen dies gleich an einem Beispiel. Die Bedeutung und Verwendung der Protokollmatrix werden später besprochen.

Zu Beginn werden die Einträge 2 und 3 der ersten Spalte auf null gesetzt, indem zu ihnen eine, durch Multiplikation, modifizierte ersten Zeile addiert wird.

Die erste Zeile hat an der ersten Position eine 1; sie wird nun mit dem Faktor $-2$ multipliziert und zur zweiten Zeile addiert, damit an der Stelle 2 der Systemmatrix eine 0 entsteht. Darum schreiben wir an der zweiten Stelle (der ersten Spalte) der Protokollmatrix das Negative des Faktors, also 2. An der dritten Stelle (der ersten Spalte) der Protokollmatrix kommt analog dazu eine 3.

| Protokollmatrix | LGS |
|---|---|
| $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ | $\begin{matrix} -2\vert \\ \searrow \end{matrix}\begin{bmatrix} \textcircled{1} & 1 & 0 \\ 2 & 1 & -1 \\ 3 & -1 & -1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 4 \\ 1 \\ -3 \end{bmatrix}$ |
| $\begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ | $\begin{matrix} -3\vert \\ \searrow \end{matrix}\begin{bmatrix} \textcircled{1} & 1 & 0 \\ 0 & -1 & -1 \\ 3 & -1 & -1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 4 \\ -7 \\ -3 \end{bmatrix}$ |
| $\begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 3 & 0 & 1 \end{bmatrix}$ | $\begin{bmatrix} 1 & 1 & 0 \\ 0 & -1 & -1 \\ 0 & -4 & -1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 4 \\ -7 \\ -15 \end{bmatrix}.$ |

Wir sehen im Rosa-Teil des Systems, dass die letzten zwei Gleichungen kein $x_1$ mehr beinhalten. Wir erhalten somit ein kleineres LGS, für welches wir auf die gleiche Weise fortfahren.

Protokollmatrix LGS

$$\begin{bmatrix}1&0&0\\2&1&0\\3&0&1\end{bmatrix} \qquad \begin{matrix}\\-4|\\\searrow\end{matrix}\begin{bmatrix}1&1&0\\0&-1&-1\\0&-4&-1\end{bmatrix}\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix}=\begin{bmatrix}4\\-7\\-15\end{bmatrix}$$

$$\begin{bmatrix}1&0&0\\2&1&0\\3&4&1\end{bmatrix} \qquad \begin{bmatrix}1&1&0\\0&-1&-1\\0&0&3\end{bmatrix}\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix}=\begin{bmatrix}4\\-7\\13\end{bmatrix}.$$

Die Matrix **A** hat nun nur noch Nullen unterhalb der Hauptdiagonale. Sie hat also die gewünschte Form. Dieses System können wir mithilfe der Rücksubstitution lösen.

Zunächst lesen wir ab:

$$3x_3 = 13 \implies x_3 = \frac{13}{3}$$

Damit können wir in die zweite Zeile einsetzen:

$$-1x_2 + (-1x_3) = -7 \implies x_2 = \frac{8}{3}$$

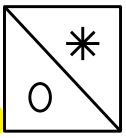
Schliesslich:

$$1x_1 + 1x_2 = 4 \implies x_1 = \frac{4}{3}$$

Es ergibt sich also: $x_3 = \frac{13}{3}$, $x_2 = \frac{8}{3}$, $x_1 = \frac{4}{3}$.
Wir beobachten auch: Die entstandene Protokollmatrix hat nur 1 auf der Hauptdiagonale und nur Nullen oberhalb der Hauptdiagonale. Sie ist eine Matrix speziellen Typs: eine *untere Dreiecksmatrix*.

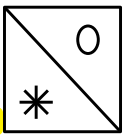
Die Matrix, die man nach der Gauss-Elimination erhält, hat nur Nullen unterhalb der Hauptdiagonale und wird deswegen *obere Dreiecksmatrix* genannt. In diesem Paragraphen werden wir kurz die verschiedenen Matrixformen besprechen, denen wir immer wieder begegnen werden. In dieser Vorlesung werden wir hauptsächlich mit vier Typen von Matrizen arbeiten.

**Definition 1.2.0.1. Obere Dreiecksmatrix**


Fig. 1

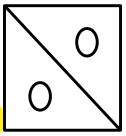
Die obere Dreiecksmatrix hat unterhalb der Hauptdiagonale nur Nullen und beliebige Zahlen oberhalb. Sie wird oft mit **U** für das Englische *''upper''* oder **R** für das Deutsche *''rechts''* notiert.

**Definition 1.2.0.2. Untere Dreiecksmatrix**


Fig. 2

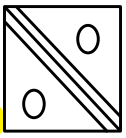
Die untere Dreiecksmatrix hat oberhalb der Hauptdiagonale nur Nullen und beliebige Zahlen unterhalb. Sie wird oft mit **L** für das Englische *''lower''*, oder das Deutsche *''links''* notiert.

**Definition 1.2.0.3. Diagonalmatrix**


Fig. 3

Die Diagonalmatrix hat nur auf der Hauptdiagonale beliebige Zahlen und sonst überall Nullen. Sie wird oft mit **D** notiert. Eine wichtige Diagonalmatrix ist die Identitätsmatrix $\mathbf{I} = \begin{bmatrix} 1 & & 0 \\ & \ddots & \\ 0 & & 1 \end{bmatrix}$, bei welcher alle Diagonaleinträge Eins sind.

**Definition 1.2.0.4. Tridiagonalmatrix**


Fig. 4

Die Tridiagonalmatrix hat eine Hauptdiagonale und zwei direkt anliegende Nebendiagonalen mit beliebigen Zahlen und sonst überall Nullen.

**Bemerkung 1.2.0.5 (Elimination verwendet nur lineare Kombinationen und produziert spezielle Matrizen)** Wir sehen also, dass die Gauss-Elimination **A** in eine obere Dreiecksmatrix **U** umwandelt und dass die Protokollmatrix von einer Diagonalmatrix zu einer unteren Dreiecksmatrix **L** wird. **Dabei werden nur lineare Kombinationen der Zeilen der Matrix gebildet.**

⌟

Die Zerlegung einer beliebigen Matrix **A** in eine untere Dreiecksmatrix **L** und eine Obere **U**, sodass $\mathbf{A} = \mathbf{L}\,\mathbf{U}$ wird LU-Zerlegung (oder auch LR-Zerlegung) genannt. Dies wird zu einem späteren Zeitpunkt im Skript noch genauer besprochen. Hier wird nur verraten, dass die Protokollmatrix genau ein solches **L** ist.

## Gauss-Elimination einer allgemeinen Matrix

Nun schauen wir uns den obigen Algorithmus, die Gauss-Elimination, für eine beliebige $m \times n$ Matrix an. Wir haben also ein LGS mit $m$ Gleichungen und $n$ Unbekannten.

Die Gauss-Elimination ist ein Algorithmus, der bei jeder Durchführung das LGS um eins verkleinert (wir sehen gleich wie) und dann rekursiv wieder auf dieses kleinere LGS angewendet wird. Auf einer abstrakten Ebene sieht die Gauss-Elimination mit einer Protokollmatrix so aus:

1. Wir gehen durch jede Spalte $j$, $j \in \{1, 2, \ldots, n\}$ unserer Matrix **A** und tun Folgendes:

   I Wir schauen das Element $a_{jj}$ an, welches das $j$-te Element entlang der Hauptdiagonalen ist.

   II Falls $a_{jj} = 0$ ist, führen wir folgende Fallunterscheidung durch:

   - Sind alle anderen Elemente $a_{ij}$ $(j < i \leq m)$, die sich in derselben Spalte unterhalb des Elements $a_{jj}$ befinden, auch $0$, dann markieren wir die aktuelle Spalte $j$ als *freie Variable* und gehen weiter zu Schritt 2.
   - Gibt es ein $i > j$ sodass $a_{ij} \neq 0$, dann vertauschen wir die $j$-te Zeile (*Pivotzeile*) mit dieser $i$-ten Zeile. An der Stelle $jj$ steht nun ein $a_{jj} \neq 0$, das wir nun *Pivot* nennen.

   III Wir gehen nun durch die darunterliegenden Zeilen $i$ und addieren zu dieser Zeile $-\frac{a_{ij}}{a_{jj}}$ mal die $j$-te Zeile. Nun sind alle $a_{ij}$, $(j < i \leq m)$ Einträge $0$.

2. Wir haben nun also das LGS um eins verkleinert. Wir können nun wieder zu Schritt 1 gehen und dasselbe für die nächste Spalte tun (und nur die Zeilen ab $j+1$ berücksichtigen).

Dieser Algorithmus, obwohl er hier vielleicht kompliziert scheinen mag, ist in der Ausführung sehr einfach und ist nach wenigen Durchführungen schon sehr intuitiv. Wenn Zeilenvertauschungen nötig sind, kann man im Prinzip alle nötigen Vertauschungen bereits am Anfang machen (wie das genau geht, sehen wir später), so dass wir mitten im Algorithmus keine Vertauschungen mehr brauchen. An der Stelle $ij$ der Protokollmatrix kommt jeweils $+\frac{a_{ij}}{a_{jj}}$.

Im Folgenden werden die ersten Schritte der Gauss-Elimination ohne Zeilenvertauschungen an einer beliebigen $m \times n$ Matrix aufgeführt:

$$\begin{bmatrix} 1 & 0 & \cdots & 0 \\ -l_{21} & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & 1 \\ 0 & \cdots & 0 & \end{bmatrix} \quad l_{21}=-\frac{a_{21}}{a_{11}} \qquad \overset{a_{11}=\text{Pivot}\neq 0}{\underset{-\frac{a_{21}}{a_{11}}}{\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ a_{31} & a_{32} & \cdots & a_{3n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}}} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_m \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & \cdots & 0 \\ -l_{21} & \ddots & \ddots & \vdots \\ -l_{31} & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & 1 \\ 0 & \cdots & 0 & \end{bmatrix} \quad l_{31}=-\frac{a_{31}}{a_{11}} \qquad \overset{a_{11}=\text{Pivot}\neq 0}{\underset{-\frac{a_{31}}{a_{11}}}{\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ 0 & a_{22}^{(1)} & \cdots & a_{2n}^{(1)} \\ a_{31} & a_{32} & \cdots & a_{3n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}}} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2^{(1)} \\ b_3 \\ \vdots \\ b_m \end{bmatrix}$$

$$\cdots$$

$$\begin{bmatrix} 1 & 0 & \cdots & 0 \\ -l_{21} & \ddots & \ddots & \vdots \\ -l_{31} & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & 1 \\ -l_{m1} & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ 0 & a_{22}^{(1)} & \cdots & a_{2n}^{(1)} \\ 0 & a_{32}^{(1)} & \cdots & a_{3n}^{(1)} \\ \vdots & \vdots & & \vdots \\ 0 & a_{m2}^{(1)} & \cdots & a_{mn}^{(1)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2^{(1)} \\ b_3^{(1)} \\ \vdots \\ b_m^{(1)} \end{bmatrix}.$$

Alle Einträge der ersten Spalte unterhalb des Pivots $= a_{11}$ sind nun Null und es ist ein kleineres LGS (rosa markiert) entstanden, für welches wir auf die gleiche Art und Weise fortfahren.

Nachdem die Gauss-Elimination für den Rest der Matrix durchgeführt wurde, hat das erhaltene LGS mit der $m \times n$-Matrix $\mathbf{U}$ die folgende Gestalt:

$$\begin{bmatrix} * & * & \cdots & * & * & * & \cdots & * & * & \cdots & * \\ 0 & 0 & \cdots & 0 & * & * & \cdots & * & * & \cdots & * \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & * & * & \cdots & * \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & \vdots & & \vdots \\ r & & & & & & & * & * & \cdots & * \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_r \\ c_{r+1} \\ \vdots \\ c_m \end{bmatrix} \} \text{ Kompatibilitätsbedingungen.}$$

Diese Matrix hat $r$ *Pivotspalten* mit den rosa markierten Pivoten und $n-r$ grün markierte Spalten mit *freien Variablen*. Die orange Linie folgt den Pivoten und liegt auf $r$ Stufen von Nullen. Man sagt, dass die Matrix nun in einer *Zeilenstufenform* ist. Diese hat mehr Nullen als eine obere Dreiecksmatrix, da die Diagonale eventuell unterbrochen ist und die Matrix Stufen aufweist (orange Linie). Diese Stufen werden von den freien Variablen hervorgerufen. Die rote Gerade liegt auf der niedrigsten Stufe (nach der $r$-ten Zeile) und gibt den sogenannten *Rang* $r$ der Matrix an. Es gibt $r$ *Stufen* und $r$ *Pivote*, d. h. $r$ Pivotzeilen, $n-r$ *freie Variablen*, also $n-r$ zu freie Variablen entsprechende Spalten, und $m-r$ *Nullzeilen*, also $m-r$ *Kompatibilitätsbedingungen*.

**Definition 1.2.0.6. Rang einer Matrix**

Der Rang $r$ einer Matrix $\mathbf{A}$, notiert als $\mathrm{Rang}(\mathbf{A})$, ist die Anzahl der Pivote der Matrix nach der Gauss-Elimination.

Das heisst, eine Systemmatrix mit $n$ Spalten, $m$ Zeilen und Rang $r$, liefert nach der Elimination $r$ Spalten mit Pivoten, $n-r$ Spalten ohne Pivote (entsprechend den freien Variablen) und $m-r$ Kompatibilitätsbedingungen. Das System hat also $r$ relevante Gleichungen, sofern die Kompatibilitätsbedingungen erfüllt sind:

$$\begin{aligned} 0 &= c_{r+1} \\ &\vdots \\ 0 &= c_m \,. \end{aligned}$$

Eine Spalte, die einer freien Variablen entspricht, hat kein Pivot. Daher gibt es zu dieser Spalte kein Element, das in die Protokollmatrix $\mathbf{L}$ zu schreiben ist. In dem Fall muss man die oben genannten Indizes entsprechend genauer schreiben. Notieren wir die (im Bild oben mit rosa eingekreisten) Pivote $u_{1\,n_1}, u_{2\,n_2}, \ldots, u_{r\,n_r}$. Der $j$-te Eliminationsschritt mit dem Pivot $u_{j\,n_j}$ ist

$$\begin{aligned} u_{ji} &= a_{ji}^{(j-1)} \text{ für } i = n_j+1, \ldots n \text{ d. h., Elemente in der Zeile nach dem Pivot} \\ c_j &= b_j^{(j-1)} \text{ die Rechte Seite} \\ l_{kj} &= \frac{a_{k\,n_j}^{(j-1)}}{u_{j\,n_j}} \text{ für } k = j+1, \ldots n \text{ Multiplikator für Zeile } k \text{ und Pivot } u_{j\,n_j} \\ a_{kj}^{(j)} &= a_{ki}^{(j-1)} - l_{kj} u_{ji} \text{ für } i = n_j+1, \ldots n \text{ und } k = j+1, \ldots n \\ b_k^{(j)} &= b_k^{(j-1)} - l_{kj} c_j \text{ für } k = j+1, \ldots n \,. \end{aligned}$$

Besondere Achtung verdienen die Plätze in der Protokollmatrix $\mathbf{L}$, wo die Multiplikatoren (mit gewechseltem Vorzeichen) geschrieben werden: an der Stelle $kj$, wenn wir die $k$-te Zeile und die $j$ Pivotzeile (d. h. mit dem Pivot $u_{j\,n_j}$). Das ist wichtig, wenn es freie Variablen gibt, denn dann gilt im Allgemeinen $j \neq n_j$. Es entsteht also eine $m \times m$ Protokollmatrix $\mathbf{L}$ mit der Gestalt:

$$\left[\begin{array}{cc|cc} 1 & 0 \cdots 0 & 0 \cdots 0 & 0 \\ l_{21} & \ddots & \vdots & \vdots \\ \vdots & \ddots\ 0 & \vdots & \vdots \\ l_{r\,1} & l_{r\,2} \cdots 1 & 0 \cdots 0 & 0 \\ \hline l_{r+1\,1} & l_{r+1\,2} \cdots l_{r+1\,r} & 1 \cdots 0 & 0 \\ \vdots & \vdots & \ddots & \vdots \\ l_{m\,1} & l_{m\,2} \cdots l_{m\,r} & 0 \cdots 1 & 0 \\ & & 0 \cdots 0 & 1 \end{array}\right] \begin{matrix} \\ \\ \\ r \\ \\ \\ \\ \\ \end{matrix},$$

**L** hat genau so viele Zeilen wie die ursprüngliche Matrix **A**, denn die Elimination behandelt alle $m$ Zeilen/-Gleichungen. Der obere linke Block ist $r \times r$: hier sind $r$ Spalten, da jede dieser Spalten zu einem Pivot in $A$ entspricht, und es sind hier $r$ Zeilen, da die letzten $m - r$ Zeilen von **L** den $m - r$ Kompatibilitätsbedingungen entsprechen.

Das eventuelle Vertauschen der Zeilen in **A** muss auch in **L** gemacht werden, jedoch bleiben die Elemente auf der Hauptdiagonale 1. Eine alternative, gleichwertige Vorstellung ist die Folgende: Am Beginn werden alle Elemente der Protokollmatrix auf null gesetzt, bei Bedarf die Zeilen vertauscht und erst am Ende die Elemente der Hauptdiagonale auf 1 gesetzt.

Es folgen zunächst Bemerkungen und Beispiele zur Gauss-Elimination. Detaillierte Erklärungen finden sich im Abschnitt 1.6 zu der LU-Zerlegung. Auch die Bedeutung der Matrix **L** wird dort eingeführt. Aktuell protokolliert die Matrix **L** nur die Rechenoperationen.

**Bemerkung 1.2.0.7 (Speicherplatz)** Die wesentlichen Elemente der Matrix **L** können wir im Speicherplatz der Matrix **U** behalten, denn an den entsprechenden Stellen in **U** stehen nur unwesentliche 0. ⌟

**Bemerkung 1.2.0.8 (Elimination berechnet lineare Kombinationen der Zeilen)**
Die Elimination bildet nur lineare Kombinationen der Zeilen. Die Zeilen, die den Kompatibilitätsbedingungen entsprechen, sind lineare Kombinationen der anderen $r$ Zeilen. ⌟

**Bemerkung 1.2.0.9 (Elimination und lineare Kombinationen der Spalten)**
Die Spalten in **U**, die keine Pivoten haben, sind lineare Kombinationen der jeweils vorigen $r$ Pivotspalten.

Das kann man jetzt schnell sehen: Sei $u_{i_f}$ eine Spalte aus **U**, die einer freien Variablen entspricht. Wenn wir die Matrix mit den Pivotspalten $a_{i_p}$ mit $i_p < i_f$ und die Spalte $a_{i_f}$ betrachten, so ist es offensichtlich, dass eine Elimination mit Spalten von unten nach oben für die letzte Spalte diese ganz mit Nullen auffüllt. Das bedeutet: die Spalte $a_{i_f}$ lässt sich als eine Linearkombination der Spalten $a_{i_p}$ mit $i_p < i_f$ schrieben. ⌟

**Bemerkung 1.2.0.10 (Kompatibilitätsbedingungen für mindestens eine Lösung)**
Damit das LGS mit dieser Matrix mindestens eine Lösung hat, müssen alle Kompatibilitätsbedingungen erfüllt sein, also

$$c_{r+1} = c_{r+2} = \ldots = c_m = 0\,.$$

⌟

**Bemerkung 1.2.0.11 (Der Rang ist gleich gross oder kleiner als die Matrixdimensionen)**
Es ist klar, dass der Rang $r$ kleiner gleich $m$ und auch kleiner gleich $n$ sein muss. Also gilt, dass

$$r \leq m,\ r \leq n \implies r \leq \min\{m, n\}\,.$$

⌟

**Bemerkung 1.2.0.12 (Die Pivotspalten sind die wesentlichen Spalten)**
Die zu den freien Variablen gehörenden Spalten von **U** sind lineare Kombinationen von den $r$ Pivotspalten von **U**. Dasselbe gilt für die Spalten der ursprünglichen Matrix **A**. ⌟

**Bemerkung 1.2.0.13 (Die Pivotzeilen sind die wesentlichen Zeilen)**
Die letzten $m - r$ Zeilen der ursprünglichen Matrix sind lineare Kombinationen der ersten $r$ Zeilen, falls wir keine Zeilenvertauschungen gemacht haben. Sonst müssen wir von Anfang an alle nötigen Vertauschungen machen, so dass diese Bemerkung mit derselben Formulierung immer noch gültig bleibt. ⌟

**Bemerkung 1.2.0.14 (Der Rang ist charakteristisch für die ganze Matrix)**
Die letzten vier Bemerkungen begründen, dass $r$ eine Grösse ist, die charakteristisch für die ganze Matrix ist, nicht nur für Zeilen oder Spalten. ⌟

## Beispiele der Gauss-Elimination

Im Folgenden sind nun einige Beispiele der Gauss-Elimination gegeben.

**Beispiel 1.2.0.15 (Zwei Pivote und eine einzige Gleichung als Kompatibilitätsbedingung)** [4]
Gegeben sei ein lineares Gleichungssystem mit einer $3 \times 3$ Matrix, das gelöst werden soll:

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \begin{bmatrix} 1 & 2 & -1 \\ 2 & -1 & 3 \\ -3 & -1 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \begin{matrix} \\ |-2 \\ |3 \end{matrix}$$

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -3 & 0 & 1 \end{bmatrix} \qquad \begin{bmatrix} 1 & 2 & -1 \\ 0 & -5 & 5 \\ 0 & 5 & -5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ -2b_1 + b_2 \\ 3b_1 + b_3 \end{bmatrix} \begin{matrix} \\ \\ |1 \end{matrix}$$

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -3 & -1 & 1 \end{bmatrix} \qquad \begin{bmatrix} 1 & 2 & -1 \\ 0 & -5 & 5 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ -2b_1 + b_2 \\ b_1 + b_2 + b_3 \end{bmatrix} .$$

Der Rang dieser Matrix ist zwei, die beiden Pivote sind $1$ und $-5$. Da der Rang zwei ist, hat diese Matrix eine freie Variable und eine Kompatibilitätsbedingung: Die freie Variable ist $x_3$ und die Kompatibilitätsbedingung ist die Gleichung $b_1 + b_2 + b_3 = 0$.

Die letzte Zeile der ursprünglichen Systemmatrix ist eine Linearkombination der ersten zwei Zeilen (mit Pivoten).

Die letzte Spalte der ursprünglichen Systemmatrix ist eine Linearkombination der ersten zwei Spalten (mit Pivoten). ⌟

**Beispiel 1.2.0.16 (Zwei Pivote und eine Gleichung als Kompatibilitätsbedingung)** [5]
Nun wollen wir sehen, wie die Protokollmatrix $\mathbf{L}$ entsteht, wenn es freie Variablen gibt. Wir betrachten hier das lineare Gleichungssystem:

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \begin{bmatrix} 2 & -1 & 2 \\ 4 & -2 & 2 \\ 8 & -4 & 6 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \begin{matrix} \\ |-2 \\ |-4 \end{matrix}$$

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 4 & 0 & 1 \end{bmatrix} \qquad \begin{bmatrix} 2 & -1 & 2 \\ 0 & 0 & -2 \\ 0 & 0 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ -2b_1 + b_2 \\ -4b_1 + b_3 \end{bmatrix} \begin{matrix} \\ \\ |-1 \end{matrix}$$

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 4 & 1 & 1 \end{bmatrix} \qquad \begin{bmatrix} 2 & -1 & 2 \\ 0 & 0 & -2 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ -2b_1 + b_2 \\ -2b_1 - b_2 + b_3 \end{bmatrix} .$$

Der Rang dieser Matrix ist zwei, die beiden Pivote sind $2$ und $-2$. Da der Rang zwei ist, hat diese Matrix eine freie Variable und eine Kompatibilitätsbedingung: Die freie Variable ist $x_2$ und die Kompatibilitätsbedingung ist die Gleichung $-2b_1 - b_2 + b_3 = 0$.
Die zweite Variable ist frei, da die zweite Spalte kein Pivot hat. In die Protokollmatrix kommt an der Stelle $(3,2)$ die Zahl, die vom nächsten Pivot abhängig ist: $l_{32} = a_{33}^{(1)} / a_{23}^{(1)} = 1$.

Die letzte Zeile der ursprünglichen Systemmatrix ist eine Linearkombination der ersten zwei Zeilen (mit Pivoten).

[4] Beispiel B, Seite 8 aus NS15.
[5] Beispiel B, Seite 8 aus NS15.

Die zweite Spalte der ursprünglichen Systemmatrix ist $-0.5$ mal die erste Spalte. ⌟

**Beispiel 1.2.0.17 (Kompatibilitätsbedingungen mit Parameter)** [6]
Sei ein lineares Gleichungssystem mit einer $m \times n$ Matrix, wobei $m = n = 5$, gegeben.

In diesem Beispiel wird auch eine neue Notation eingeführt. Die Unbekannte $\mathbf{x}$ wird weggelassen, da sie für die Eliminationsschritte nicht gebraucht wird. Anstatt die Matrix $\mathbf{A}$ und den Vektor $\mathbf{b}$ separat zu schreiben, fasst man sie zu einer Matrix zusammen und trennt sie mit einem Strich. Man spricht von einer erweiterten Matrix. Die Elimination in der ersten Spalte

$$\overbrace{\begin{bmatrix} 2 & -1 & 3 & -1 & 1 \\ 2 & -1 & 3 & 0 & -1 \\ -4 & 2 & -4 & 5 & -5 \\ 0 & 0 & -2 & 2 & -7 \\ -2 & 1 & -1 & 0 & 4 \end{bmatrix}}^{\mathbf{A}} \overbrace{\left|\begin{matrix} -2 \\ -3 \\ 3 \\ -5+s \\ 5 \end{matrix}\right.}^{\mathbf{b}} \quad \begin{matrix} |-1 \\ \\ \\ \\ \end{matrix} \quad \begin{matrix} |2 \\ \\ \\ \\ \end{matrix} \quad \begin{matrix} |1 \\ \\ \\ \\ \end{matrix}$$

ergibt die Protokollmatrix

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ -2 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

und die Systemmatrix samt rechter Seite verändert sich zu:

$$\left[\begin{array}{ccccc|c} 2 & -1 & 3 & -1 & 1 & -2 \\ 0 & 0 & 0 & 1 & -2 & -1 \\ 0 & 0 & 2 & 3 & -3 & -1 \\ 0 & 0 & -2 & 2 & -7 & -5+s \\ 0 & 0 & 2 & -1 & 5 & 3 \end{array}\right] \quad \updownarrow$$

.

Die zweite Spalte dieser Matrix enthält kein Pivot, also wird $x_2$ eine freie Variable und wir gehen zur dritten Spalte. Da an der Stelle $(2,3)$ eine Null steht, müssen wir eine Zeilenvertauschung machen; zum Beispiel vertauschen wir die zweite Zeile mit der Dritten, so dass eine $2$ an der Stelle $(2,3)$ erscheint und als Pivot dienen kann. Das Vertauschen der zweiten mit der dritten Zeile muss man auch in der Protokollmatrix $\mathbf{L}$ machen (nur die Hauptdiagonale bleibt gleich):

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -2 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

.

$$\left[\begin{array}{ccccc|c} 2 & -1 & 3 & -1 & 1 & -2 \\ 0 & 0 & 2 & 3 & -3 & -1 \\ 0 & 0 & 0 & 1 & -2 & -1 \\ 0 & 0 & -2 & 2 & -7 & -5+s \\ 0 & 0 & 2 & -1 & 5 & 3 \end{array}\right] \quad \begin{matrix} |1 \\ \\ \\ \\ \end{matrix} \quad \begin{matrix} |-1 \\ \\ \\ \\ \end{matrix}$$

[6] Beispiel C, Seite 9 aus NS15.

Die Elimination in der dritten Spalte liefert:

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -2 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 \\ -1 & 1 & 0 & 0 & 1 \end{bmatrix}$$

und

$$\left[\begin{array}{ccccc|c} 2 & -1 & 3 & -1 & 1 & -2 \\ 0 & 0 & 2 & 3 & -3 & -1 \\ 0 & 0 & 0 & 1 & -2 & -1 \\ 0 & 0 & 0 & 5 & -10 & -6+s \\ 0 & 0 & 0 & -4 & 8 & 4 \end{array}\right] \quad |-5 \quad |4 \quad ,$$

was zu

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -2 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & -1 & 5 & 1 & 0 \\ -1 & 1 & -4 & 0 & 1 \end{bmatrix}$$

und

$$\mathbf{U} = \left[\begin{array}{ccccc|c} 2 & -1 & 3 & -1 & 1 & -2 \\ 0 & 0 & 2 & 3 & -3 & -1 \\ 0 & 0 & 0 & 1 & -2 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1+s \\ 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right]$$

führt.

Der Rang $r$ dieser Matrix ist drei, die Pivote sind $2,\ 2,\ 1$. Die beiden freien Variablen sind $x_2$ und $x_5$.

Die 4-te und 5-te Zeile sind die beiden Kompatibilitätsbedingungen, von welchen die Letztere immer erfüllt ist und die Andere $0 = -1 + s$ lautet. Falls $s \neq 1$ hat das System keine Lösungen, falls $s = 1$ gibt es unendlich viele Lösungen mit den frei wählbaren Variablen $x_2$ und $x_5$.

Wenn $s = 1$ ist und die freie Variable $x_5$ in die dritte Gleichung eingesetzt wird so erhält man

$$\begin{aligned} x_4 - 2x_5 &= -1 \\ x_4 &= -1 + 2x_5 \,. \end{aligned}$$

$x_4$ wird durch die freie Variable $x_5$ definiert. Sie kann nun, im Sinne der Rücksubstitution, in die zweite Gleichung einsetzten werden, um $x_3$ zu erhalten:

$$\begin{aligned} 2x_3 + 3x_4 - 3x_5 &= -1 \\ 2x_3 + 3(-1 + 2x_5) - 3x_5 &= -1 \\ x_3 &= 1 - \frac{3}{2}x_5 \,. \end{aligned}$$

Nun ist auch $x_3$ in Abhängigkeit von $x_5$ definiert und kann nun in die erste Gleichung eingesetzt werden:

$$\begin{aligned} 2x_1 - x_2 + 3x_3 - x_4 + x_5 &= -2 \\ 2x_1 - x_2 + 3(1 - \frac{3}{2}x_5) - (-1 + 2x_5) + x_5 &= -2 \\ x_1 &= -3 + \frac{1}{2}x_2 + \frac{11}{4}x_5 \,. \end{aligned}$$

Dies gibt einem zum Schluss folgende Werte für die Unbekannten:

$$x_1 = -3 + \frac{1}{2}x_2 + \frac{11}{4}x_5$$
$$x_2 \text{ ist frei}$$
$$x_3 = 1 - \frac{3}{2}x_5$$
$$x_4 = -1 + 2x_5$$
$$x_5 \text{ ist frei.}$$

Aus der **U** Matrix lesen wir:

1. die beiden letzten Zeilen der ursprünglichen Systemmatrix **A** sind jeweils eine Linearkombination der ersten drei Zeilen (mit Pivoten).
2. die zweite Spalte der ursprünglichen Systemmatrix **A** ist ein Vielfaches der ersten Spalte, die 5-te Spalte ist eine Linearkombination der Spalten 1, 3 und 4 (die vorigen Spalten, die Pivote haben).

In der Matrix **L** entsprechen die letzten $m - r = 5 - 3 = 2$ Spalten den Kompatibilitätsbedingungen. ⌟

**Beispiel 1.2.0.18 (Freie Variablen und Kompatibilitätsbedingungen)** [7]
In diesem Beispiel, einem linearen Gleichungssystem mit einer $7 \times 9$ Matrix, werden wir die Gauss-Elimination nicht vollständig ausführen, sondern nur die resultierende Matrix angeben und an ihr die wichtigen Eigenschaften besprechen.

$$\begin{bmatrix} 1 & 0 & 5 & 0 & 4 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 5 & 0 & 4 & 3 & 0 & 2 \\ 1 & 5 & 9 & 3 & 6 & 1 & 0 & 1 & 0 \\ 1 & 10 & 13 & 11 & 8 & 6 & 3 & 1 & 2 \\ 0 & 5 & 4 & 18 & 2 & 18 & 12 & 2 & 7 \\ 1 & 10 & 13 & 21 & 8 & 24 & 16 & 5 & 8 \\ 0 & 5 & 4 & 13 & 2 & 24 & 17 & 6 & 7 \end{bmatrix}$$

Gauss-Elimination…

$$\begin{bmatrix} 1 & 0 & 5 & 0 & 4 & 0 & 0 & 1 & 0 \\ 0 & 5 & 4 & 3 & 2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & 4 & 3 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 5 & 3 & 2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

.

Diese Matrix hat demzufolge Rang $r = 5$, mit Pivoten in den Spalten 1, 2, 4, 6, 7. Die Matrix hat $m - r = 2$ Kompatibilitätsbedingungen und $n - r = 4$ freie Variablen, die den Spalten 3,5,8,9 entsprechen.
Auch hier gilt, falls die beiden Kompatibilitätsbedingungen erfüllt sind, so gibt es unendlich viele Lösungen mit den vier freien Parametern $x_3$, $x_5$, $x_8$, $x_9$. Wenn die Kompatibilitätsbedingungen nicht erfüllt sind, so wird es keine Lösung geben.

Aus der **U** Matrix lesen wir:

[7] Beispiel 1.4, Seite 1-11 aus Gut07.

1. die beiden letzten Zeilen der ursprünglichen Systemmatrix sind jeweils eine Linearkombination der ersten fünf Zeilen (mit Pivoten);
2. die 3-te Spalte der ursprünglichen Systemmatrix ist eine Linearkombination der ersten zwei Spalten, die 5-te Spalte ist eine Linearkombination der Spalten 1, 2, 4 (mit Pivoten), die letzten zwei Spalten sind jeweils eine Linearkombination der Spalten (mit Pivoten): 1,2,4,6,7.

Da die Berechnungen für diese Matrix etwas komplizierter sind, geben wir hier die Ergebnisse der einzelnen Schritte und eine Berechnungsstrategie, die mehr Zeilenvertauschungen mit sich bringt, als es eigentlich notwendig wären.
Die Eliminationen in der ersten Spalte ergeben:

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \text{ und } \begin{bmatrix} 1 & 0 & 5 & 0 & 4 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 5 & 0 & 24 & 16 & 8 & 6 \\ 0 & 5 & 4 & 3 & 2 & 1 & 0 & 0 & 0 \\ 0 & 10 & 8 & 11 & 4 & 6 & 3 & 0 & 2 \\ 0 & 5 & 4 & 18 & 2 & 18 & 12 & 2 & 7 \\ 0 & 10 & 8 & 21 & 4 & 24 & 16 & 4 & 8 \\ 0 & 5 & 4 & 13 & 2 & 24 & 17 & 6 & 7 \end{bmatrix}.$$

An der Stelle $(2,2)$ steht nun eine Null, so dass wir die zweite und dritte Zeile miteinander vertauschen, um ein Pivot (in diesem Fall 5) zu erhalten:

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \text{ und } \begin{bmatrix} 1 & 0 & 5 & 0 & 4 & 0 & 0 & 1 & 0 \\ 0 & 5 & 4 & 3 & 2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & 24 & 16 & 8 & 6 \\ 0 & 10 & 8 & 11 & 4 & 6 & 3 & 0 & 2 \\ 0 & 5 & 4 & 18 & 2 & 18 & 12 & 2 & 7 \\ 0 & 10 & 8 & 21 & 4 & 24 & 16 & 4 & 8 \\ 0 & 5 & 4 & 13 & 2 & 24 & 17 & 6 & 7 \end{bmatrix}.$$

Die Elimination in der zweiten Spalte ergibt:

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 2 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 2 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \text{ und } \begin{bmatrix} 1 & 0 & 5 & 0 & 4 & 0 & 0 & 1 & 0 \\ 0 & 5 & 4 & 3 & 2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & 24 & 16 & 8 & 6 \\ 0 & 0 & 0 & 5 & 0 & 4 & 3 & 0 & 2 \\ 0 & 0 & 0 & 15 & 0 & 17 & 12 & 2 & 7 \\ 0 & 0 & 0 & 15 & 0 & 22 & 16 & 4 & 8 \\ 0 & 0 & 0 & 10 & 0 & 23 & 17 & 6 & 7 \end{bmatrix}.$$

Die dritte Spalte enthält nun kein Pivot, also ist $x_3$ eine freie Variable und das nächste Pivot wird in der vierten Spalte gesucht. Man kann 5 aus der Position $(3,4)$ verwenden, was aber zu grossen Zahlen führt. Das kann man vermeiden, wenn man die dritte und vierte Zeile miteinander vertauscht:

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 2 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \text{ und } \begin{bmatrix} 1 & 0 & 5 & 0 & 4 & 0 & 0 & 1 & 0 \\ 0 & 5 & 4 & 3 & 2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & 4 & 3 & 0 & 2 \\ 0 & 0 & 0 & 5 & 0 & 24 & 16 & 8 & 6 \\ 0 & 0 & 0 & 15 & 0 & 17 & 12 & 2 & 7 \\ 0 & 0 & 0 & 15 & 0 & 22 & 16 & 4 & 8 \\ 0 & 0 & 0 & 10 & 0 & 23 & 17 & 6 & 7 \end{bmatrix}.$$

Elimination in der vierten Spalte führt zu

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 3 & 0 & 1 & 0 & 0 \\ 1 & 2 & 3 & 0 & 0 & 1 & 0 \\ 0 & 1 & 2 & 0 & 0 & 0 & 1 \end{bmatrix} \text{ und } \begin{bmatrix} 1 & 0 & 5 & 0 & 4 & 0 & 0 & 1 & 0 \\ 0 & 5 & 4 & 3 & 2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & 4 & 3 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 20 & 13 & 8 & 4 \\ 0 & 0 & 0 & 0 & 0 & 5 & 3 & 2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 10 & 7 & 4 & 2 \\ 0 & 0 & 0 & 0 & 0 & 15 & 11 & 6 & 3 \end{bmatrix}.$$

Die fünfte Spalte enthält nun kein Pivot, also ist $x_5$ eine freie Variable und das nächste Pivot wird in der sechsten Spalte gesucht. Man kann 20 aus der Position $(4,6)$ verwenden, was aber zu grossen Zahlen führt. Das kann man vermeiden, wenn man die vierte und die fünfte Zeile vertauscht:

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 3 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 2 & 3 & 0 & 0 & 1 & 0 \\ 0 & 1 & 2 & 0 & 0 & 0 & 1 \end{bmatrix} \text{ und } \begin{bmatrix} 1 & 0 & 5 & 0 & 4 & 0 & 0 & 1 & 0 \\ 0 & 5 & 4 & 3 & 2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & 4 & 3 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 5 & 3 & 2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 20 & 13 & 8 & 4 \\ 0 & 0 & 0 & 0 & 0 & 10 & 7 & 4 & 2 \\ 0 & 0 & 0 & 0 & 0 & 15 & 11 & 6 & 3 \end{bmatrix}.$$

Elimination in der sechsten Spalte liefert:

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 3 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 4 & 1 & 0 & 0 \\ 1 & 2 & 3 & 2 & 0 & 1 & 0 \\ 0 & 1 & 2 & 3 & 0 & 0 & 1 \end{bmatrix} \text{ und } \begin{bmatrix} 1 & 0 & 5 & 0 & 4 & 0 & 0 & 1 & 0 \\ 0 & 5 & 4 & 3 & 2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & 4 & 3 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 5 & 3 & 2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \end{bmatrix}.$$

Elimination in der siebten Spalte liefert:

$$\mathbf{L} = \left[\begin{array}{ccccc|cc} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 3 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 4 & 1 & 0 & 0 \\ \hline 1 & 2 & 3 & 2 & 1 & 1 & 0 \\ 0 & 1 & 2 & 3 & 2 & 0 & 1 \end{array}\right] \text{ und } \mathbf{U} = \begin{bmatrix} 1 & 0 & 5 & 0 & 4 & 0 & 0 & 1 & 0 \\ 0 & 5 & 4 & 3 & 2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & 4 & 3 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 5 & 3 & 2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

**Bemerkung 1.2.0.19 (Die Kompatibilitätsbedingungen entscheiden über die Existenz von Lösungen)**
Die Anzahl der Kompatibilitätsbedingungen (KB) eines LGS ist der massgebende Faktor, ob es Lösungen hat oder nicht.

1. Es gibt genau dann keine KB, wenn es mindestens eine Lösung für jede rechte Seite gibt.
   Dies ist der Fall, wenn $r = m \leq n$.
2. Es gibt genau dann KB $c_{r+1} = c_{r+2} = \ldots = c_m = 0$, wenn $r < m$. Für jede gegebene rechte Seite müssen wir diese überprüfen, um zu wissen, ob es überhaupt Lösungen gibt.

⌟

**Bemerkung 1.2.0.20 (Ohne KB bestimmen die freien Variablen die Menge der Lösungen)**
Die Anzahl der Lösungen eines linearen Gleichungssystems ohne KB ist abhängig von der Anzahl an freien Variablen:

1. Das LGS hat keine KB, aber es hat freie Variablen; dann hat es unendlich viele Lösungen für beliebige rechte Seiten.
   Dies ist der Fall, wenn $r = m < n$.
2. Ein LGS hat keine KB und keine freien Variablen genau dann, wenn es *genau eine* Lösung für jede rechte Seite hat.
   Dies ist der Fall, wenn $r = m = n$.

⌟

**Satz 1.2.0.21.**

*Für ein lineares Gleichungssystem mit Matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ und $\mathrm{Rang}(\mathbf{A}) = r$ gilt:*
*Falls es eine Lösung gibt, ist diese eindeutig dann und nur dann, wenn $r = n$.*

## 1.2.1 Homogene Gleichungssysteme

**Definition 1.2.1.1. Homogenes LGS**

Ein lineares Gleichungssystem heisst *homogen*, falls die rechte Seite Null ist:

$$\mathbf{A}\,\mathbf{x} = 0\,.$$

Ein homogenes Gleichungssystem hat immer $x = 0$ als Lösung, die sogenannte *triviale* Lösung.

**Bemerkung 1.2.1.2 (Nicht-triviale Lösungen eines homogenen Systems)**
Ein homogenes System hat genau dann nicht triviale Lösungen ($\mathbf{x} \neq \mathbf{0}$) wenn $\mathrm{Rang}(\mathbf{A}) = r < n$, da es in diesem Fall freie Variablen hat. ⌟

**Bemerkung 1.2.1.3 (Äquivalenzschema quadratischer Matrizen)** Jede quadratische Matrix ($m = n$) kann in eine der zwei folgenden Kategorien eingeteilt werden:

| $\mathbf{A} \in \mathbb{R}^{n\times n}$ heisst *regulär (voller Rang)* | $\mathbf{A} \in \mathbb{R}^{n\times n}$ heisst *singulär (nicht voller Rang)* |
|---|---|
| $\mathrm{Rang}(\mathbf{A}) = r = n$ | $\mathrm{Rang}(\mathbf{A}) = r < n$ |
| $\Updownarrow$ | $\Updownarrow$ |
| Für jedes $\mathbf{b}$ gibt es mindestens eine Lösung | Für gewisse $\mathbf{b}$ gibt es keine Lösung |
| $\Updownarrow$ | $\Updownarrow$ |
| Für jedes $\mathbf{b}$ gibt es genau eine Lösung | Für kein $\mathbf{b}$ gibt es eine eindeutige Lösung |
| $\Updownarrow$ | $\Updownarrow$ |
| $\mathbf{Ax} = \mathbf{0}$ hat nur $\mathbf{x} = \mathbf{0}$ als Lösung | $\mathbf{Ax} = \mathbf{0}$ hat nicht triviale Lösungen |

⌟

# 1.3 Operationen mit Matrizen

## Transponierte Matrizen

Wir können eine Matrix $\mathbf{A}$ auf verschiedene Arten notieren.

Wir können sie komplett aufschreiben:

$$\mathbf{A} = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix}.$$

Wir können sie mithilfe ihrer Spaltenvektoren $\mathbf{a}_1, \ldots, \mathbf{a}_n$ aufschreiben:

$$\mathbf{A} = [\mathbf{a}_1 \ \cdots \ \mathbf{a}_n],$$

oder auch nur mithilfe ihrer Koeffizienten $a_{ij}$:

$$\mathbf{A} = \left[a_{ij}\right]_{\substack{i=1,2,\cdots,m \text{ (Zeilen)} \\ j=1,2,\cdots,n \text{ (Spalten)}}}.$$

**Definition 1.3.0.1. Transponierte Matrix**

Für eine beliebige reelle Matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ mit Einträgen:

$$\mathbf{A} = \left[a_{ij}\right]_{\substack{i=1,2,\ldots,m \\ j=1,2,\ldots,n}},$$

definieren wir die *transponierte Matrix* $\mathbf{A}^\intercal \in \mathbb{R}^{n\times m}$, folgendermassen:

$$\mathbf{A}^\intercal = \left[a_{ji}\right]_{\substack{i=1,2,\ldots,m \\ j=1,2,\ldots,n}}.$$

Für $\mathbf{A} = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \end{bmatrix}$

ist die transponierte Matrix:

$\mathbf{A}^\intercal = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$.

Für komplexe Matrizen gibt noch eine zusätzliche Definition.

Hier ist kurz als Erinnerung die Definition der Konjugierten einer komplexen Zahl gegeben. Die Konjugierte der komplexen Zahl $z = x + \imath y$ ist notiert als $\overline{z}$ und definiert als $\overline{z} = x - \imath y$; hier ist $\imath^2 = -1$.

**Definition 1.3.0.2. Hermite transponierte Matrix**

Für eine beliebige komplexe Matrix $\mathbf{A} \in \mathbb{C}^{m\times n}$ mit Einträgen

$$\mathbf{A} = \left[a_{kj}\right]_{\substack{k=1,2,\ldots,m\\ j=1,2,\ldots,n}}$$

definieren wir die *Hermite transponierte Matrix* $\mathbf{A}^{\mathsf{H}} \in \mathbb{C}^{n\times m}$ als

$$\mathbf{A}^{\mathsf{H}} = \left[\overline{a}_{jk}\right]_{\substack{k=1,2,\ldots,m\\ j=1,2,\ldots,n}},$$

wobei $\overline{a}_{jk}$ die komplex Konjugierte von $a_{jk}$ ist.

Ein Beispiel der Hermite Transponierung einer Matrix $\mathbf{A}$ ist:

$$\mathbf{A} = \begin{bmatrix} 1+\imath & 1+3\imath & 1+5\imath \\ 1+2\imath & 1+4\imath & 1+6\imath \end{bmatrix} \Longrightarrow \mathbf{A}^{\mathsf{H}} = \begin{bmatrix} 1-\imath & 1-2\imath \\ 1-3\imath & 1-4\imath \\ 1-5\imath & 1-6\imath \end{bmatrix}.$$

Mithilfe der Transponierung und der Hermite Transponierung können wir Matrizen mit speziellen Eigenschaften definieren.

**Definition 1.3.0.3. Symmetriearten von Matrizen**

**Symmetrisch:**
Eine Matrix $\mathbf{A}$ heisst symmetrisch, falls $\mathbf{A}^{\mathsf{T}} = \mathbf{A}$.

**Antisymmetrisch:**
Eine Matrix $\mathbf{A}$ heisst antisymmetrisch, falls $\mathbf{A}^{\mathsf{T}} = -\mathbf{A}$.

**Hermite symmetrisch:**
Eine Matrix $\mathbf{A}$ heisst Hermite symmetrisch, falls $\mathbf{A}^{\mathsf{H}} = \mathbf{A}$.

Zwei Beispiele von symmetrischen Matrizen sind:

$$\begin{bmatrix} 1 & 2 \\ 2 & 5 \end{bmatrix}, \quad \begin{bmatrix} 2 & 3 & 1 \\ 3 & 2 & 6 \\ 1 & 6 & 2 \end{bmatrix},$$

von antisymmetrischen Matrizen sind:

$$\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, \quad \begin{bmatrix} 0 & a & 5 \\ -a & 0 & 4 \\ -5 & -4 & 0 \end{bmatrix},$$

und von Hermite symmetrischen Matrizen:

$$\begin{bmatrix} 1 & 1+\imath \\ 1-\imath & 2 \end{bmatrix}, \quad \begin{bmatrix} 1 & 0 & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & 1 \end{bmatrix}, \quad \begin{bmatrix} 0 & \cdots & \cdots & 0 \\ \vdots & & & \vdots \\ \vdots & & & \vdots \\ 0 & \cdots & \cdots & 0 \end{bmatrix}.$$

## Addition und Multiplikation mit Skalaren

Ein Vektor $\mathbf{x}$ in $\mathbb{R}^n$ oder $\mathbb{C}^n$ kann als eine Matrix in $\mathbb{R}^{n\times 1}$ oder $\mathbb{C}^{n\times 1}$ gesehen werden.

Die Addition von Matrizen passender Dimensionen und die Multiplikation von Matrizen mit einem Skalar sind, genau wie bei Vektoren, elementweise definiert.

**Definition 1.3.0.4. Transponierte Vektoren**

Sei $\mathbf{x}$ ein Vektor oder auch eine Matrix, so lässt sich die Transponierung $\mathbf{x}^\mathsf{T}$ und Hermite Transponierung $\mathbf{x}^\mathsf{H}$ für diesen Vektor definieren als:

$$\mathbf{x}^\mathsf{T} = [x_1\, x_2\, \cdots\, x_n]$$
$$\mathbf{x}^\mathsf{H} = [\overline{x}_1\, \overline{x}_2\, \cdots\, \overline{x}_n]\,.$$

Dies wird oft verwendet, wenn ein Vektor auf einer einzelnen Zeile geschrieben werden muss.

Die Addition von Vektoren, welche bereits bei der Linearkombination eingeführt wurde, ist definiert als die Addition der einzelnen Komponenten beider Vektoren, welche an gleicher Position sind.

$$\mathbf{x}^{(1)} + \mathbf{x}^{(2)} = \begin{bmatrix} x_1^{(1)} \\ x_2^{(1)} \\ \vdots \\ x_n^{(1)} \end{bmatrix} + \begin{bmatrix} x_1^{(2)} \\ x_2^{(2)} \\ \vdots \\ x_n^{(2)} \end{bmatrix} = \begin{bmatrix} x_1^{(1)} + x_1^{(2)} \\ x_2^{(1)} + x_2^{(2)} \\ \vdots \\ x_n^{(1)} + x_n^{(2)} \end{bmatrix}.$$

Auch haben wir die Multiplikation mit einem Skalar bei der Linearkombination gesehen. Bei der Multiplikation mit einem Skalar wird jede Komponente des Vektors $\mathbf{x}$ mit dem Skalar $\lambda$, mit $\lambda \in \mathbb{R}$ oder $\mathbb{C}$, multipliziert.

$$\lambda\mathbf{x} = \lambda \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} \lambda x_1 \\ \lambda x_2 \\ \vdots \\ \lambda x_n \end{bmatrix}.$$

Aus diesen beiden Definitionen für den Vektor lässt sich einfach eine Definition für die Matrix ableiten.

**Definition 1.3.0.5.**

Seien $\mathbf{A}$, $\mathbf{B}$ Matrizen mit $\mathbf{A}$, $\mathbf{B} \in \mathbb{R}^{m\times n}$ und $\beta$ ein Skalar in $\mathbb{R}$ oder $\mathbb{C}$. So ist die Addition und die Multiplikation mit einem Skalar definiert als:

**Addition:**
Sei $\mathbf{C} = \mathbf{A} + \mathbf{B}$, so werden alle Einträge an gleicher Position addiert und man erhält $c_{ij} = a_{ij} + b_{ij}$.

**Multiplikation mit Skalare:**
Sei $\mathbf{D} = \beta\mathbf{A}$, so werden alle Einträge von $\mathbf{A}$ mit $\beta$ multipliziert $d_{ij} = \beta a_{ij}$.

**Satz 1.3.0.6. Eigenschaften der Matrixaddition**

*Im Folgenden sind einige Eigenschaften der Matrixaddition aufgelistet:*

1. $\mathbf{A}+\mathbf{B}=\mathbf{B}+\mathbf{A}$ *(kommutativ)*,
2. $\mathbf{0}+\mathbf{A}=\mathbf{A}+\mathbf{0}=\mathbf{A}$ *($\mathbf{0}$ ist das neutrale Element)*,
3. $\mathbf{A}+(-1)\mathbf{A}=\mathbf{0}$ ,
4. $(\mathbf{A}+\mathbf{B})+\mathbf{C}=\mathbf{A}+(\mathbf{B}+\mathbf{C})$ *(assoziativ)*,
5. $(\mathbf{A}+\mathbf{B})^{\mathsf{T}}=\mathbf{A}^{\mathsf{T}}+\mathbf{B}^{\mathsf{T}}$, $(\mathbf{A}+\mathbf{B})^{\mathsf{H}}=\mathbf{A}^{\mathsf{H}}+\mathbf{B}^{\mathsf{H}}$.

## Matrixmultiplikation

Die Matrixmultiplikation wird über die Matrix-Vektor-Multiplikation definiert. Die Definition dieser Multiplikation sahen wir bereits in Bemerkung 1.1.0.5. Dort wurde gesagt: Für eine beliebige Matrix $\mathbf{A}\in\mathbb{R}^{m\times n}$ und beliebigen Vektor $\mathbf{x}\in\mathbb{R}^n$ ist die Matrix-Vektor-Multiplikation definiert als die Linearkombination der Spalten von $\mathbf{A}$ mit den Koeffizienten von $\mathbf{x}$:

$$\mathbf{b}=\mathbf{A}\mathbf{x}$$
$$\mathbf{b}=x_1\mathbf{a}_1+x_2\mathbf{a}_2+\cdots+x_n\mathbf{a}_n$$
$$b_i=x_1a_{i1}+x_2a_{i2}+\cdots+x_na_{in}=\sum_{j=1}^{n}x_j\,a_{ij}\ \text{für } i=1,\ldots,m\,.$$

Da ein Vektor $\mathbf{x}$ in $\mathbb{R}^n$ oder $\mathbb{C}^n$ als eine Matrix in $\mathbb{R}^{n\times 1}$ oder $\mathbb{C}^{n\times 1}$ geschrieben werden kann, sollte die Matrixmultiplikation konsistent mit dieser Definition sein. Als Motivation und Intuition für die Einführung dieser Matrixmultiplikation fragen wir uns: Was sollte man tun, um viele LGS, die alle dieselbe Matrix $\mathbf{A}$ haben, effizient zu lösen?

$$\mathbf{A}\mathbf{x}^{(1)}=\mathbf{b}^{(1)}\,,\qquad \mathbf{A}\mathbf{x}^{(2)}=\mathbf{b}^{(2)}\,.$$

Dies kann man vereinfacht schreiben, indem man die Matrix $\mathbf{A}$ nur einmal notiert und die Vektoren $\mathbf{x}^{(1)}$, $\mathbf{x}^{(2)}$ und $\mathbf{b}^{(1)}$, $\mathbf{b}^{(2)}$ als Spalten zweier neuen Matrizen ansieht.

$$\mathbf{A}\begin{bmatrix}\mathbf{x}^{(1)} & \mathbf{x}^{(2)}\end{bmatrix}=\begin{bmatrix}\mathbf{b}^{(1)} & \mathbf{b}^{(2)}\end{bmatrix}$$

Diese Notation hilft, da die Gauss-Elimination nur einmal für beide rechte Seiten $\mathbf{b}$ gleichzeitig ausgeführt wird. Dieses Vorgehen ist also effizienter. Die Gauss-Elimination ist nur von den Einträgen in der Matrix $\mathbf{A}$ abhängig. Die Faktoren, mit denen Zeilen multipliziert werden, sind unabhängig von $\mathbf{b}$ und $\mathbf{x}$. Welche Zeilen man addiert oder vertauscht werden, ist unabhängig von $\mathbf{b}$ und $\mathbf{x}$. Deswegen sind auch die durchgeführten Operationen, welche an $\mathbf{b}$ vorgenommen werden, unabhängig von $\mathbf{b}$ und $\mathbf{x}$.

Dies kann für $p$ rechte Seiten verallgemeinert werden:

$$\mathbf{A}\begin{bmatrix}\mathbf{x}^{(1)}\,\mathbf{x}^{(2)}\,\cdots\,\mathbf{x}^{(p)}\end{bmatrix}=\begin{bmatrix}\mathbf{b}^{(1)}\,\mathbf{b}^{(2)}\,\cdots\,\mathbf{b}^{(p)}\end{bmatrix}$$

$$\mathbf{A}\mathbf{X}=\mathbf{B}\,.$$

Wir erhalten also eine natürliche Definition der Matrixmultiplikation aus der Definition der Matrix-Vektor-Multiplikation und den linearen Gleichungssystemen. Bevor wir die formale Definition schreiben, beachten wir noch die Dimensionen der beteiligten Matrizen :

$$\underbrace{\mathbf{A}}_{m\times n}\ \underbrace{\begin{bmatrix}\mathbf{x}^{(1)}\,\mathbf{x}^{(2)}\,\cdots\,\mathbf{x}^{(p)}\end{bmatrix}}_{\mathbf{X}\ n\times p}=\underbrace{\begin{bmatrix}\mathbf{b}^{(1)}\,\mathbf{b}^{(2)}\,\cdots\,\mathbf{b}^{(p)}\end{bmatrix}}_{\mathbf{B}\ m\times p}\,.$$

Die Dimension von $\mathbf{X}$ ist $n \times p$, da $\mathbf{A}$ eine Matrix mit $n$ Spalten (Anzahl der Unbekannten) ist und $p$ Systeme gleichzeitig gelöst werden. Die Dimension von $\mathbf{B}$ ist $m \times p$, da die Matrix $\mathbf{A}$ $m$ Zeilen hat und $p$ Systeme gleichzeitig gelöst werden.
Bei einer Matrixmultiplikation setzt sich die Dimension der resultierenden Matrix aus den Dimensionen der ihr zum Ursprung liegenden Matrizen zusammen, da sie ihre Zeilenanzahl aus der linken und ihre Spaltenanzahl aus der rechten Matrix der Multiplikation erbt. Genau wie bei der Matrix-Vektor-Multiplikation müssen die Dimensionen der Matrizen bei der Multiplikation passen.

Aus diesem intuitiven Blick ergibt sich nun eine allgemeine Definition:

**Definition 1.3.0.7. Matrixmultiplikation**

Seien $\mathbf{A}$, $\mathbf{X}$ Matrizen mit passenden Dimensionen $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{X} \in \mathbb{R}^{n\times p}$. So ist die Multiplikation der Matrix $\mathbf{A}$ mit der Matrix $\mathbf{X}$ in genau dieser Reihenfolge definiert als die Zusammenstellung der Ergebnisse der Multiplikation der Matrix $\mathbf{A}$ mit jedem Spaltenvektor von $\mathbf{X}$:

$$\mathbf{A}\left[\mathbf{x}^{(1)}\, \mathbf{x}^{(2)}\, \cdots\, \mathbf{x}^{(p)}\right] = \left[\mathbf{b}^{(1)}\, \mathbf{b}^{(2)}\, \cdots\, \mathbf{b}^{(p)}\right]$$

$$\begin{aligned} \mathbf{A}\mathbf{x}^{(1)} &= \mathbf{b}^{(1)} \\ \mathbf{A}\mathbf{x}^{(2)} &= \mathbf{b}^{(2)} \\ &\vdots \\ \mathbf{A}\mathbf{x}^{(p)} &= \mathbf{b}^{(p)} \end{aligned}$$

$$\mathbf{A}\mathbf{X} = \mathbf{B}\,.$$

Weiter ergibt sich eine Rechenregel:

$$\overset{m\times n}{\begin{bmatrix} \cdots\cdots\cdots\cdots \\ a_{i1}\quad a_{i2}\cdots\cdots a_{in} \\ \cdots\cdots\cdots\cdots \end{bmatrix}} \overset{n\times p}{\begin{bmatrix} \vdots & & \vdots \\ x_{1j} & & \\ x_{2j} & & \\ \vdots & & \\ x_{nj} & & \end{bmatrix}} = \overset{m\times p}{\begin{bmatrix} \vdots \\ \cdots b_{ij} \cdots \\ \vdots \end{bmatrix}}.$$

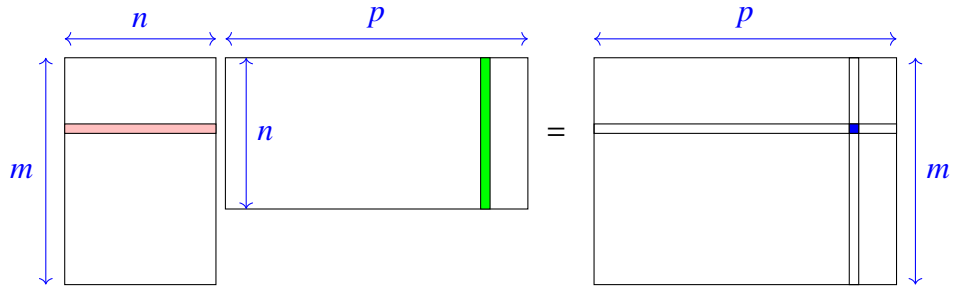
Die $i$-te Gleichung im $j$-ten LGS ist somit:

$$a_{i1}x_{1j} + a_{i2}x_{2j} + \cdots + a_{in}x_{nj} = b_{ij}$$

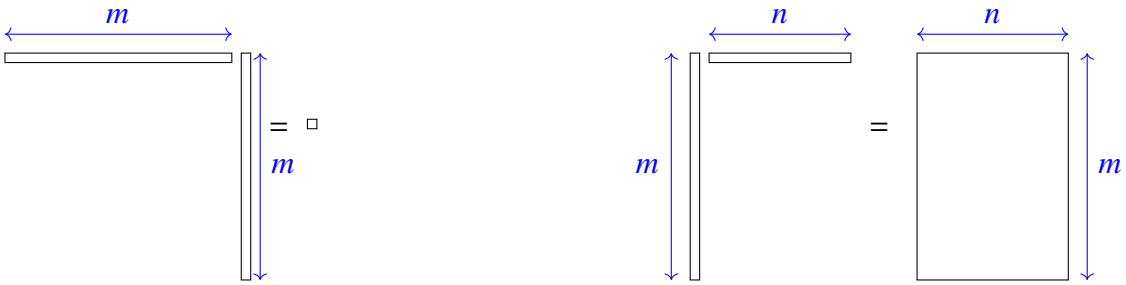
$$\sum_{k=1}^{n} a_{ik}x_{kj} = b_{ij}\,.$$

Die letzte Summe definiert das $ij$-te Element in $\mathbf{B}$ der Matrixmultiplikation $\mathbf{AX} = \mathbf{B}$.

Man kann diese Multiplikation auch so visualisieren:



Es gibt zwei wichtige spezielle Fälle: ein Fall entspricht dem (Euklidischen) Skalarprodukt, wenn das Ergebnis ein Skalar ist, ein Fall entspricht dem äusseren Produkt zweier Vektoren, wenn das Ergebnis eine Matrix vom Rang 1 ist:




**Bemerkung 1.3.0.8 (Für die Matrixmultiplikation müssen die Dimensionen passen!)**
Eine $m \times n$, $(m \neq n)$ Matrix kann nicht mit einer anderen $m \times n$ Matrix multipliziert werden. ⌟

**Bemerkung 1.3.0.9 (Der Rang des Ergebnisses der Matrixmultiplikation ist maximal der kleinste Rang der Faktoren)**
Jede Spalte im Ergebnis $\mathbf{B}$ der Matrixmultiplikation $\mathbf{AX} = \mathbf{B}$ ist eine Linearkombination der Spalten von $\mathbf{A}$. Die Koeffizienten in dieser Linearkombination sind die Einträge in den entsprechenden Spalten von $\mathbf{X}$.

Jede Zeile im Ergebnis $\mathbf{B}$ der Matrixmultiplikation $\mathbf{AX} = \mathbf{B}$ ist eine Linearkombination der Zeilen von $\mathbf{X}$. Die Koeffizienten in dieser Linearkombination sind die Einträge in den entsprechenden Zeilen von $\mathbf{A}$.

Es kann aber passieren, dass solche Kombinationen auch Null ergeben, wodurch der Rang des Ergebnisses sinken kann. Damit ist der Rang des Ergebnisses maximal der kleinste Rang von $\mathbf{A}$ und $\mathbf{X}$:

$$\operatorname{Rang}(\mathbf{AX}) \leq \min\{\operatorname{Rang}\mathbf{A}, \operatorname{Rang}\mathbf{X}\}\,.$$

⌟

**Bemerkung 1.3.0.10 (Multiplikation mit einer Diagonalmatrix)** Es ist wichtig zu erkennen, worin der Unterschied im Ergebnis der Multiplikation mit einer Diagonalmatrix von links bzw. von rechts auf eine andere Matrix liegt:

- Multiplikation mit einer Diagonalmatrix *von links* ➤ Skalierung der Zeilen

$$\begin{bmatrix} d_1 & 0 & & 0 \\ 0 & d_2 & & 0 \\ & & \ddots & \\ 0 & 0 & & d_n \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & & a_{2m} \\ \vdots & & & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{bmatrix} = \begin{bmatrix} d_1a_{11} & d_1a_{12} & \dots & d_1a_{1m} \\ d_2a_{21} & d_2a_{22} & \dots & d_2a_{2m} \\ \vdots & & & \vdots \\ d_na_{n1} & d_na_{n2} & \dots & d_na_{nm} \end{bmatrix} = \begin{bmatrix} d_1(\mathbf{A})_{1,:} \\ \vdots \\ d_n(\mathbf{A})_{n,:} \end{bmatrix}.$$

- Multiplikation mit einer Diagonalmatrix *von rechts* ➤ Skalierung der Spalten

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & & a_{2m} \\ \vdots & & & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{bmatrix} \begin{bmatrix} d_1 & 0 & & 0 \\ 0 & d_2 & & 0 \\ & & \ddots & \\ 0 & 0 & & d_m \end{bmatrix} = \begin{bmatrix} d_1a_{11} & d_2a_{12} & \dots & d_ma_{1m} \\ d_1a_{21} & d_2a_{22} & \dots & d_ma_{2m} \\ \vdots & & & \vdots \\ d_1a_{n1} & d_2a_{n2} & \dots & d_ma_{nm} \end{bmatrix}$$
$$= \begin{bmatrix} d_1(\mathbf{A})_{:,1} & \dots & d_m(\mathbf{A})_{:,m} \end{bmatrix}.$$

⌟

**Bemerkung 1.3.0.11 (Die Struktur einer Matrix ist wichtig)** Die vorhandene Struktur in einer Matrix wirkt sich entscheidend auf das Ergebnis aus. Die unter gezeigte Matrix $\mathbf{E}$ multipliziert von links auf eine Matrix $\mathbf{A}$ bewirkt folgendes: Die erste Zeile von $\mathbf{A}$ wird mit $-2$ multipliziert und zu der zweiten Zeile addiert:

$$\mathbf{EA} = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ -2a_{11}+a_{21} & -2a_{12}+a_{22} & -2a_{13}+a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}.$$

Wenn wir $\mathbf{A}$ von rechts mit $\mathbf{E}$ multiplizieren, dann erhalten wir:

$$\mathbf{AE} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} a_{11}-2a_{12} & a_{12} & a_{13} \\ a_{21}-2a_{22} & a_{22} & a_{23} \\ a_{31}-2a_{32} & a_{32} & a_{33} \end{bmatrix}.$$

Wenn wir $\mathbf{A}$ von rechts mit $\mathbf{E}^T$ multiplizieren, dann erhalten wir:

$$\mathbf{AE}^T = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} 1 & -2 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} a_{11} & -2a_{11}+a_{12} & a_{13} \\ a_{21} & -2a_{21}+a_{22} & a_{23} \\ a_{31} & -2a_{31}+a_{32} & a_{33} \end{bmatrix}$$

d.h. die erste Spalte wird mit $-2$ multipliziert und zur zweiten Spalte addiert. ⌟

**Bemerkung 1.3.0.12 (Die Matrixmultiplikation ist nicht kommutativ)**
Für zwei $n \times n$ Matrizen $\mathbf{A}$, $\mathbf{B}$ gilt im Allgemeinen $\mathbf{AB} \neq \mathbf{BA}$. Die Matrixmultiplikation ist also **nicht kommutativ**. Dies impliziert, dass sich die Multiplikation von Matrizen nicht wie die Multiplikation von Zahlen verhält. ⌟

**Bemerkung 1.3.0.13 (Die Identitätsmatrix ist das neutrale Element für die Matrixmultiplikation)**
$\mathbf{AI} = \mathbf{A} = \mathbf{IA}$, wobei $\mathbf{I}$ die Identitätsmatrix ist; sie ist das neutrale Element der Matrixmultiplikation. Zur Erinnerung: Die Identitätsmatrix ist die Matrix, die nur Einsen auf der Hauptdiagonale und sonst überall Nullen hat. $\mathbf{I} = \begin{bmatrix} 1 & & 0 \\ & \ddots & \\ 0 & & 1 \end{bmatrix}$. ⌟

**Definition 1.3.0.14. Potenzen von Matrizen**

Eine Matrix $\mathbf{A}$ zu einer natürlichen Potenz $k$ ist folgendermassen rekursiv definiert:
$\mathbf{A}^0 = \mathbf{I}$, $\mathbf{A}^1 = \mathbf{A}^0\mathbf{A}^1$, $\ldots$, $\mathbf{A}^{k+1} = \mathbf{A}^k\mathbf{A}$ für $1 \leq k$, $k \in \mathbb{N}$.

**Bemerkung 1.3.0.15 (Blick auf die Multiplikation von Matrizen: Zeilenweise & Spaltenweise)** Wir haben das Produkt zweier Matrizen $\mathbf{A}, \mathbf{B}$ als die Zusammensetzung von Spalten definiert: die Ergebnisse der Multiplikation der ganzen Matrix $\mathbf{A}$ mit jeder Spalte der Matrix $\mathbf{B}$. Wenn wir aber die Rechenformel anschauen, ist es klar, dass das Ergebnis $\mathbf{AB}$ auch als eine Zusammensetzung von Zeilen gesehen werden kann: die Ergebnisse der Multiplikation jeder Zeile der Matrix $\mathbf{A}$ mit der ganzen Matix $\mathbf{B}$: $\mathbf{A} \in \mathbb{R}^{m,n}$, $\mathbf{B} \in \mathbb{R}^{n,p}$:

$$\mathbf{AB} = \begin{bmatrix} \mathbf{A}[\mathbf{B}]_{:,1} & \ldots & \mathbf{A}[\mathbf{B}]_{:,p} \end{bmatrix} \quad , \quad \mathbf{AB} = \begin{bmatrix} [\mathbf{A}]_{1,:}\mathbf{B} \\ \vdots \\ [\mathbf{A}]_{m,:}\mathbf{B} \end{bmatrix}. \tag{1.3.0.16}$$

$\downarrow$ Matrix gebaut aus Spalten — $\downarrow$ Matrix gebaut aus Zeilen

⌟

Wie auch für die Addition lassen sich für die Multiplikation gewisse Eigenschaften beweisen.

**Satz 1.3.0.17. Eigenschaften der Matrixmultiplikation**

*Alle folgenden Sätze gelten unter der Annahme, dass die Dimensionen der einzelnen Matrizen passen. Wir sind also in der Lage, die Matrizen miteinander zu multiplizieren:*

1. $(\mathbf{A}\cdot\mathbf{B})\cdot\mathbf{C} = \mathbf{A}\cdot(\mathbf{B}\cdot\mathbf{C})$ *(assoziativ)*
2. $(\mathbf{A}+\mathbf{B})\cdot\mathbf{C} = \mathbf{A}\cdot\mathbf{C}+\mathbf{B}\cdot\mathbf{C}$ *(''·'' distributiv bezüglich ''+'')*
   $\mathbf{A}\cdot(\mathbf{B}+\mathbf{C}) = \mathbf{A}\cdot\mathbf{B}+\mathbf{A}\cdot\mathbf{C}$
3. $(\mathbf{A}\cdot\mathbf{B})^{\mathsf{T}} = \mathbf{B}^{\mathsf{T}}\cdot\mathbf{A}^{\mathsf{T}}$
   $(\mathbf{A}\cdot\mathbf{B})^{\mathsf{H}} = \mathbf{B}^{\mathsf{H}}\cdot\mathbf{A}^{\mathsf{H}}$
4. $(\mathbf{A}\cdot\mathbf{B}\cdot\mathbf{C})^{\mathsf{T}} = \mathbf{C}^{\mathsf{T}}\cdot\mathbf{B}^{\mathsf{T}}\cdot\mathbf{A}^{\mathsf{T}}$

**Bemerkung 1.3.0.18 (Blockweises Produkt)** Das Produkt zweier Matrizen kann auch blockweise berechnet werden. Dabei muss man darauf achten, dass die Multiplikation von Matrizen bzw. Blöcken nicht kommutativ ist.
Seien die Dimensionen $M, N, K \in \mathbb{N}$ und die Blocklängen $1 \le n < N$ ($n' := N-n$), $1 \le m < M$ ($m' := M-m$), $1 \le k < K$ ($k' := K-k$); nehmen wir die Matrizen

$$\begin{matrix}\mathbf{A}_{11}\in\mathbb{R}^{m,n} & \mathbf{A}_{12}\in\mathbb{R}^{m,n'}\\ \mathbf{A}_{21}\in\mathbb{R}^{m',n} & \mathbf{A}_{22}\in\mathbb{R}^{m',n'}\end{matrix}\quad,\quad\begin{matrix}\mathbf{B}_{11}\in\mathbb{R}^{n,k} & \mathbf{B}_{12}\in\mathbb{R}^{n,k'}\\ \mathbf{B}_{21}\in\mathbb{R}^{n',k} & \mathbf{B}_{22}\in\mathbb{R}^{n',k'}\end{matrix},$$

die wir als Blöcke verwenden, um grössere Matrizen zu formen:

$$\mathbf{A}=\begin{bmatrix}\mathbf{A}_{11}&\mathbf{A}_{12}\\ \mathbf{A}_{21}&\mathbf{A}_{22}\end{bmatrix}\in\mathbb{R}^{M,N}\quad,\quad\mathbf{B}=\begin{bmatrix}\mathbf{B}_{11}&\mathbf{B}_{12}\\ \mathbf{B}_{21}&\mathbf{B}_{22}\end{bmatrix}\in\mathbb{R}^{N,K}.$$

Das Produkt $\mathbf{AB}$ kann man mit der selben Formel wie das Produkt einfacher $2\times 2$-Matrizen ausrechnen:

$$\begin{bmatrix}\mathbf{A}_{11}&\mathbf{A}_{12}\\ \mathbf{A}_{21}&\mathbf{A}_{22}\end{bmatrix}\begin{bmatrix}\mathbf{B}_{11}&\mathbf{B}_{12}\\ \mathbf{B}_{21}&\mathbf{B}_{22}\end{bmatrix}=\begin{bmatrix}\mathbf{A}_{11}\mathbf{B}_{11}+\mathbf{A}_{12}\mathbf{B}_{21}&\mathbf{A}_{11}\mathbf{B}_{12}+\mathbf{A}_{12}\mathbf{B}_{22}\\ \mathbf{A}_{21}\mathbf{B}_{11}+\mathbf{A}_{22}\mathbf{B}_{21}&\mathbf{A}_{21}\mathbf{B}_{12}+\mathbf{A}_{22}\mathbf{B}_{22}\end{bmatrix}. \tag{1.3.0.19}$$

⌟

## 1.4 Inverse einer Matrix

Die Inverse einer Matrix ist nur für quadratische Matrizen definiert.

**Definition 1.4.0.1. Inverse einer Matrix**

Die $n \times n$ Matrix $\mathbf{X}$ heisst *Inverse* der $n \times n$ Matrix $\mathbf{A}$, falls $\mathbf{AX} = \mathbf{I}$.
Wenn die $n \times n$ Matrix $\mathbf{A}$ eine Inverse hat, dann nennen wir sie *invertierbar*.

Jede Matrix kann als die Aneinanderreihung ihrer Spaltenvektoren geschrieben werden:

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 \cdot \cdots \cdot \mathbf{x}_2 \end{bmatrix}.$$

Wenn dies die Inverse von $\mathbf{A}$ ist, dann ist

$$\mathbf{A}\begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 \cdot \cdots \cdot \mathbf{x}_n \end{bmatrix} = \begin{bmatrix} 1 & \cdot & 0 \\ 0 & & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{e}_1 & \mathbf{e}_2 \cdot \cdots \cdot \mathbf{e}_n \end{bmatrix},$$

wobei die $\mathbf{e}_i$ die Standardbasisvektoren von $\mathbb{R}^n$ sind.

Im Kapitel über Matrixmultiplikation definierten wir dies als das gleichzeitige Lösen von mehreren Gleichungssystemen:

$$\mathbf{AX} = \mathbf{I} \iff \mathbf{Ax}_1 = \mathbf{e}_1,\ \mathbf{Ax}_2 = \mathbf{e}_2, \ldots,\ \mathbf{Ax}_n = \mathbf{e}_n,$$

d. h. dass die $i$-te Spalte von $\mathbf{X}$ die Lösung des LGS $\mathbf{Ax}_i = \mathbf{e}_i$ für jedes $i = 1, 2, \ldots, n$ ist.

**Bemerkung 1.4.0.2 (Wenn eine Matrix eine Inverse hat, dann ist sie regulär (hat vollen Rang))**
Sei nun ein $\mathbf{b} \in \mathbb{R}^n$ beliebig. Wir betrachten das LGS $\mathbf{Ax} = \mathbf{b}$.

Wir nehmen an, dass $\mathbf{A}$ eine Inverse $\mathbf{X}$ besitzt. Mit diesem $\mathbf{X}$ und $\mathbf{b}$ definieren wir nun

$$\mathbf{x} = \mathbf{Xb} = b_1\mathbf{x}_1 + b_2\mathbf{x}_2 + \ldots + b_n\mathbf{x}_n.$$

Wir rechnen:

$$\mathbf{Ax} = \mathbf{A}(\mathbf{Xb}) = (\mathbf{AX})\mathbf{b} = \mathbf{Ib} = \mathbf{b}.$$

Somit gilt, dass $\mathbf{x} = \mathbf{Xb}$ die Lösung des linearen Gleichungssystems $\mathbf{Ax} = \mathbf{b}$, $\mathbf{b}$ beliebig, ist. Das $\mathbf{x}$ ist demzufolge eindeutig für jedes $\mathbf{b}$ und dies bedeutet, nach dem Äquivalenzschema quadratischer Matrizen, dass $\mathrm{Rang}(\mathbf{A}) = n$ ist. Die Matrix ist also regulär.

Dies bedeutet, dass wenn eine Matrix $\mathbf{A}$ eine Inverse besitzt, muss sie auch vollen Rang haben.

⌟

**Bemerkung 1.4.0.3 (Eine reguläre Matrix hat eine Inverse)**
In der vorherigen Bemerkung wurde gezeigt, dass eine Matrix, die eine Inverse besitzt, regulär ist. Das Gleiche geht aber auch umgekehrt: Eine Matrix, die regulär ist, besitzt auch eine Inverse.

Sei $\mathbf{A}$ eine Matrix mit der Eigenschaft, dass $\mathrm{Rang}(\mathbf{A}) = n$ ist. Sie ist also regulär. Für diese Matrix können die folgenden linearen Gleichungssysteme betrachtet werden.

$$\mathbf{Ay}_1 = \mathbf{e}_1,\ \mathbf{Ay}_2 = \mathbf{e}_2,\ \ldots,\ \mathbf{Ay}_n = \mathbf{e}_n$$

Das Äquivalenzschema quadratischer Matrizen sagt, dass jedes dieser Systeme eine eindeutige Lösung hat. Alle $\mathbf{y}_i$ sind also eindeutig und können als Spalten in eine Matrix $\mathbf{Y}$ geschrieben werden:

$$\mathbf{Y} = \begin{bmatrix} \mathbf{y}_1 \cdot \cdots \cdot \mathbf{y}_n \end{bmatrix}.$$

Für dieses $\mathbf{Y}$ gilt, dass $\mathbf{AY} = \mathbf{I}$ ist. $\mathbf{Y}$ ist also die Inverse von $\mathbf{A}$.

Eine Matrix welche regulär ist, hat also auch eine Inverse, welche eindeutig ist.

⌟

Aus diesen beiden Bemerkungen kann ein Theorem über die Existenz und Eindeutigkeit der Inverse formuliert werden.

**Satz 1.4.0.4. Existenz und Eindeutigkeit der Inverse**

*Die Matrix $\mathbf{A}$ hat eine Inverse genau dann, wenn $Rang(\mathbf{A}) = n$, die Matrix also regulär ist. Ausserdem ist die Inverse eindeutig.*

**Bemerkung 1.4.0.5.**
Die Begriffe *regulär*, *invertierbar*, *voller Rang* werden also als Synonyme verwendet (für quadratische Matrizen). ⌟

Notation der Inverse:
Die Inverse von $\mathbf{A}$ wird mit $\mathbf{A}^{-1}$ notiert, also: $\mathbf{A}\mathbf{A}^{-1} = \mathbf{I}$.

**Satz 1.4.0.6. Eigenschaften der Inverse**

*Seien $\mathbf{A}$, $\mathbf{B}$ invertierbare Matrizen, dann haben sie folgende Eigenschaften:*

1. *Wenn gilt, dass $\mathbf{A}\mathbf{A}^{-1} = \mathbf{I}$, dann gilt auch, dass $\mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$;*
2. *$\mathbf{A}^{-1}$ ist invertierbar und $(\mathbf{A}^{-1})^{-1} = \mathbf{A}$;*
3. *$\mathbf{I}$ ist invertierbar und $\mathbf{I}^{-1} = \mathbf{I}$;*
4. *$\mathbf{AB}$ ist invertierbar und $(\mathbf{AB})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1}$;*
5. *$\mathbf{A}^\intercal$ ist invertierbar und $(\mathbf{A}^\intercal)^{-1} = (\mathbf{A}^{-1})^\intercal$.*

*Beweis.* Im Folgenden werden nun Beweise für die Aussagen in Satz 1.4.0.6 gegeben:

**1) Wenn gilt, dass $\mathbf{A}\mathbf{A}^{-1} = \mathbf{I}$, dann gilt auch, dass $\mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$:**
Wir notieren $\mathbf{D} = \mathbf{A}^{-1}\mathbf{A}$ und zeigen: $\mathbf{D} = \mathbf{I}$.

$$\mathbf{AD} = \mathbf{A}\underbrace{(\mathbf{A}^{-1}\mathbf{A})}_{\mathbf{D}} = (\mathbf{A}\mathbf{A}^{-1})\mathbf{A} = \mathbf{I}\mathbf{A} = \mathbf{A}\,.$$

Somit gilt: $\mathbf{AD} = \mathbf{A}$. Dies ist genau dann der Fall, wenn

$$\begin{aligned} \mathbf{A}\mathbf{d}_1 &= \mathbf{a}_1 \\ &\dots \\ \mathbf{A}\mathbf{d}_n &= \mathbf{a}_n\,. \end{aligned}$$

Die $\mathbf{a}_1, \dots, \mathbf{a}_n$ sind die Spalten von $\mathbf{A}$ und es gilt $\mathbf{A}\mathbf{e}_1 = \mathbf{a}_1, \dots, \mathbf{A}\mathbf{e}_n = \mathbf{a}_n$.
Des Weiteren ist $\mathbf{A}$ invertierbar, hat also Rang $n$ und somit haben alle diese linearen Gleichungssysteme eindeutige Lösungen und es muss gelten, dass

$$\begin{aligned} &\mathbf{d}_1 = \mathbf{e}_1,\ \mathbf{d}_2 = \mathbf{e}_2,\ \dots,\ \mathbf{d}_n = \mathbf{e}_n \\ \Longleftrightarrow\ &\mathbf{D} = \mathbf{I}\,. \end{aligned}$$

Wenn $\mathbf{D} = \mathbf{I}$ und $\mathbf{D}$ definiert wurde als $\mathbf{A}^{-1}\,\mathbf{A}$, dann gilt:

$$\mathbf{A}^{-1}\mathbf{A} = \mathbf{I}\,.$$

**2) $\mathbf{A}^{-1}$ ist invertierbar und $(\mathbf{A}^{-1})^{-1} = \mathbf{A}$:**
Folgt direkt aus **1)**: $\mathbf{A}^{-1}\mathbf{A} = \mathbf{I} \iff \mathbf{A}^{-1}(\mathbf{A}^{-1})^{-1} = \mathbf{I}$.

**3) $\mathbf{I}$ ist invertierbar und $\mathbf{I}^{-1} = \mathbf{I}$:**
Dies ist trivial, da $\mathbf{I}\,\mathbf{I} = \mathbf{I}$.

**4) $\mathbf{AB}$ ist invertierbar und $(\mathbf{AB})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1}$:**
Dies können wir einfach durch Ausrechnen beweisen:

$$\begin{aligned}(\mathbf{AB})(\mathbf{B}^{-1}\mathbf{A}^{-1}) &= \mathbf{A}(\mathbf{B}\mathbf{B}^{-1})\mathbf{A}^{-1}\\ &= (\mathbf{A}\mathbf{I})\mathbf{A}^{-1}\\ &= \mathbf{A}\mathbf{A}^{-1}\\ &= \mathbf{I}\,.\end{aligned}$$

Also ist $(\mathbf{B}^{-1}\mathbf{A}^{-1})$ die Inverse von $(\mathbf{AB})$.

**5) $\mathbf{A}^{\mathsf{T}}$ ist invertierbar und $(\mathbf{A}^{\mathsf{T}})^{-1} = (\mathbf{A}^{-1})^{\mathsf{T}}$:**
Dies werden wir durch mathematische Umformungen zeigen, insbesondere werden wir **2)** ausnützen, $\mathbf{I} = \mathbf{A}^{-1}\mathbf{A}$

$$\begin{aligned}\mathbf{I} = \mathbf{I}^{\mathsf{T}} &= (\mathbf{A}^{-1}\mathbf{A})^{\mathsf{T}}\\ &= \mathbf{A}^{\mathsf{T}}(\mathbf{A}^{-1})^{\mathsf{T}}\,.\end{aligned}$$

Nach der Definition der Inverse bedeutet dies, dass $\mathbf{A}^{\mathsf{T}}(\mathbf{A}^{-1})^{\mathsf{T}} = \mathbf{I}$. Daraus folgt, dass $(\mathbf{A}^{\mathsf{T}})^{-1} = (\mathbf{A}^{-1})^{\mathsf{T}}$.

□

**Satz 1.4.0.7. Kriteria für Existenz der Inverse**

*Für eine $n \times n$ Matrix $\mathbf{A}$ sind die folgenden Aussagen äquivalent:*

1. $\mathbf{A}$ *invertierbar.*
2. $\mathit{Rang}(\mathbf{A}) = n$.
3. $\mathbf{Ax} = \mathbf{b}$ *ist lösbar für alle* $\mathbf{b}$.
4. $\mathbf{Ax} = \mathbf{0}$ *hat nur die triviale Lösung* $\mathbf{x} = \mathbf{0}$.

## 1.5 Berechnung der Inverse

Für die Berechnung der Inverse wird das sogenannte *Gauss-Jordan* Verfahren benutzt.

Gegeben sei eine invertierbare $n \times n$-Matrix $\mathbf{A}$. Das Ziel ist es, eine Matrix $\mathbf{X}$ zu finden, so dass $\mathbf{AX} = \mathbf{I}$. Dies entspricht dem gleichzeitigen Lösen von $n$ linearen Gleichungssystemen mit $\mathbf{A}$ als gemeinsame Systemmatrix und den Elementen $\mathbf{e}_j$ der kanonischen Basis als rechte Seite:

$$\left[\mathbf{A} \mid \mathbf{I}\right] \rightsquigarrow \left[\mathbf{I} \mid \mathbf{A}^{-1}\right].$$

Wir werden aber nicht nur die Gauss-Elimination und dann separat eine Rücksubstitution ausführen, sondern wir werden die Rücksubstitution anschliessend direkt in diesem System ausführen. Wenn wir dies machen, so erhalten wir zum Schluss die Identitätsmatrix auf der linken Seite und auf der rechten Seite die gesuchte Inverse.

Dies lässt sich am besten an einem Beispiel sehen.

**Beispiel 1.5.0.1 (Eine invertierbare $3 \times 3$-Matrix)** [8]
Sei die $3 \times 3$ Matrix $\mathbf{A}$ gegeben als:

$$\mathbf{A} = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}$$

Für diese Matrix soll die Inverse $\mathbf{A}^{-1}$ mittels des Gauss-Jordan-Verfahren berechnet werden.

$$\left[\begin{array}{ccc|ccc} 2 & -1 & 0 & 1 & 0 & 0 \\ -1 & 2 & -1 & 0 & 1 & 0 \\ 0 & -1 & 2 & 0 & 0 & 1 \end{array}\right] \begin{array}{l} \\ \\ \\ \end{array} \left|\tfrac{1}{2}\right.$$

$$\left[\begin{array}{ccc|ccc} 2 & -1 & 0 & 1 & 0 & 0 \\ 0 & \frac{3}{2} & -1 & \frac{1}{2} & 1 & 0 \\ 0 & -1 & 2 & 0 & 0 & 1 \end{array}\right] \begin{array}{l} \\ \\ \\ \end{array} \left|\tfrac{2}{3}\right.$$

$$\left[\begin{array}{ccc|ccc} 2 & -1 & 0 & 1 & 0 & 0 \\ 0 & \frac{3}{2} & -1 & \frac{1}{2} & 1 & 0 \\ 0 & 0 & \frac{4}{3} & \frac{1}{3} & \frac{2}{3} & 1 \end{array}\right].$$

An diesem Punkt ist die Gauss-Elimination abgeschlossen und wir werden mittels Rücksubstitution die Einträge von $\mathbf{A}^{-1}$ ermitteln.

$$\left[\begin{array}{ccc|ccc} 2 & -1 & 0 & 1 & 0 & 0 \\ 0 & \frac{3}{2} & -1 & \frac{1}{2} & 1 & 0 \\ 0 & 0 & \frac{4}{3} & \frac{1}{3} & \frac{2}{3} & 1 \end{array}\right] \begin{array}{l} \\ \\ \\ \end{array} \left|\tfrac{3}{4}\right.$$

[8] Beispiel aus Kapitel 2.5, Seite 84 aus Str09.

$$\left[\begin{array}{ccc|ccc} 2 & -1 & 0 & 1 & 0 & 0 \\ 0 & \frac{3}{2} & 0 & \frac{3}{4} & \frac{3}{2} & \frac{3}{4} \\ 0 & 0 & \frac{4}{3} & \frac{1}{3} & \frac{2}{3} & 1 \end{array}\right] \begin{array}{l} \\ \\ \left|\frac{2}{3}\right. \end{array}$$

$$\left[\begin{array}{ccc|ccc} 2 & 0 & 0 & \frac{3}{2} & 1 & \frac{1}{2} \\ 0 & \frac{3}{2} & 0 & \frac{3}{4} & \frac{3}{2} & \frac{3}{4} \\ 0 & 0 & \frac{4}{3} & \frac{1}{3} & \frac{2}{3} & 1 \end{array}\right] \begin{array}{l} \left|\cdot\frac{1}{2}\right. \\ \left|\cdot\frac{2}{3}\right. \\ \left|\cdot\frac{3}{4}\right. \end{array}$$

$$\left[\begin{array}{ccc|ccc} 1 & 0 & 0 & \frac{3}{4} & \frac{1}{2} & \frac{1}{4} \\ 0 & 1 & 0 & \frac{1}{2} & 1 & \frac{1}{2} \\ 0 & 0 & 1 & \frac{1}{4} & \frac{1}{2} & \frac{3}{4} \end{array}\right] .$$

Die mittels des Gauss-Jordan Verfahren berechnete Inverse von $\mathbf{A}$ ist also

$$\mathbf{A}^{-1} = \frac{1}{4}\begin{bmatrix} 3 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 3 \end{bmatrix}.$$

⌟

**Bemerkung 1.5.0.2.**
Die Pivot-Elemente der oberen Gauss-Elimination sind $2$, $\frac{3}{2}$, $\frac{4}{3}$. Das Produkt dieser Pivote ist $2 \cdot \frac{3}{2} \cdot \frac{4}{3} = 4$ also genau der Nenner des Bruchs, der vor der Inversen steht. Dies ist kein Zufall.

⌟

**Definition 1.5.0.3. Determinante einer Matrix (Berechnung)**

Das Produkt der Diagonalelemente (mal $(-1)^n$, wobei $n$ die Anzahl der Zeilenvertauschungen ist) nach der Gauss-Elimination heisst *Determinante* der quadratischen Matrix $\mathbf{A}$.

Diese Definition mag den einen oder anderen Kenner vielleicht erstaunen, denn das ist keine der üblichen (äquivalenten) Definitionen der Determinante. Das ist die Art, wie man Determinanten für grosse Matrizen praktisch berechnet. Determinanten werden in einem späteren Kapitel genauer besprochen, einschliesslich der Äquivalenz einiger alternativer Definitionen.

**Bemerkung 1.5.0.4 (Eine Matrix ist regulär (invertierbar, voller Rang) genau dann, wenn ihre Determinante nicht Null ist)**
$\det\mathbf{A} \neq 0$ genau dann, wenn $\mathbf{A}$ invertierbar ist, also wenn $\mathrm{Rang}(\mathbf{A}) = n$.

⌟

Die Inverse einer Matrix kann verwendet werden, um lineare Gleichungssysteme zu lösen. Gegeben sei eine quadratische $n \times n$ Matrix $\mathbf{A}$, die invertierbar (regulär) ist. Mit dieser Matrix ist ein LGS $\mathbf{Ax} = \mathbf{b}$ mit beliebigem $\mathbf{b}$ definiert.

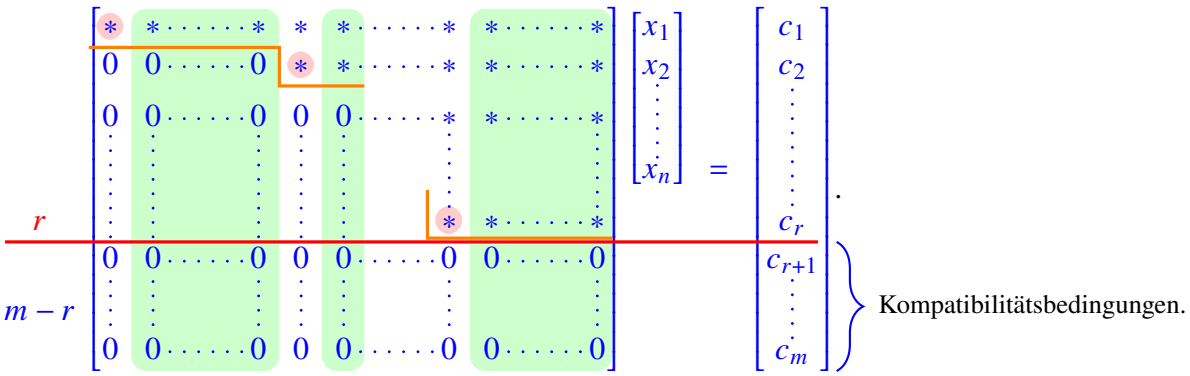
$$\mathbf{Ax} = \mathbf{b} \overset{\mathbf{A}^{-1}}{\Longrightarrow} \mathbf{A}^{-1}(\mathbf{Ax}) = \mathbf{A}^{-1}\mathbf{b}$$

$$(\mathbf{A}^{-1}\mathbf{A})\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$$
$$\mathbf{I}\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$$
$$\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}\,.$$

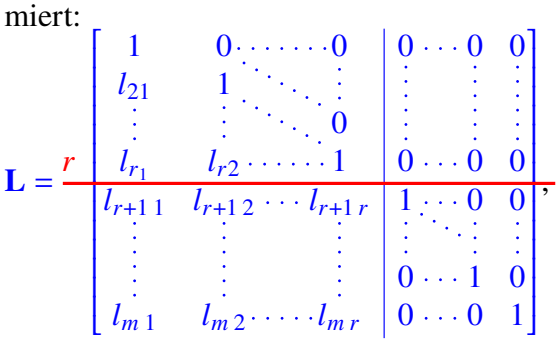
Die Unbekannte $\mathbf{x}$ kann also auch in einem Schritt durch die Multiplikation von $\mathbf{A}^{-1}$ mit $\mathbf{b}$ berechnet werden. Allerdings ist die Berechnung der Inversen einer $n \times n$-Matrix teurer als das Lösen eines LGS. Darum wird die Inverse in der Realität praktisch nie berechnet, sondern es werden oft nur LGS gelöst.

## 1.6 LU-Zerlegung

Bei der Gauss-Elimination wird das lineare Gleichungssystem $\mathbf{Ax} = \mathbf{b}$ zu $\mathbf{Ux} = \mathbf{c}$ transformiert. Aus der $m \times n$ Matrix $\mathbf{A}$ wird die $m \times n$ Matrix $\mathbf{U}$, welche eine obere Zeilenstufenform hat:




Die Protokollmatrix wird dabei von einer Diagonalmatrix in eine $m \times m$ untere Dreiecksmatrix $\mathbf{L}$ transformiert:




Wir werden bald beweisen, was wir nun als eine einfache Bemerkung angeben:

**Bemerkung 1.6.0.1 (Die Gauss-Elimination liefert die LU-Zerlegung)**
Die Gauss-Elimination einer Matrix $\mathbf{A}$, bei welcher keine Zeilen vertauscht werden, die liefert zwei Matrizen $\mathbf{L}$, $\mathbf{U}$ mit der Eigenschaft, dass

$$\mathbf{A} = \mathbf{LU}\,.$$

Falls bei der Gauss-Elimination auch Zeilen vertauscht werden, so bekommt man noch eine dritte Matrix $\mathbf{P}$, die Permutationsmatrix, und es gilt

$$\mathbf{PA} = \mathbf{LU}\,.$$

⌟

**Bemerkung 1.6.0.2 (Verschiedene Implementierungen können verschiedene Matrizen produzieren)**
Einerseits gibt es in `NumPy` die Funktion `lu`, die die Matrizen $\mathbf{P}, \mathbf{L}, \mathbf{U}$ liefert, so dass $\mathbf{A} = \mathbf{PLU}$. Andererseits gibt eine gleichnamige Funktion in `Matlab` die Matrizen $\mathbf{P}, \mathbf{L}, \mathbf{U}$ zurück, die $\mathbf{PA} = \mathbf{LU}$ erfüllen. Diese $\mathbf{P}$ ist die Inverse der oben erwähnten Matrix $\mathbf{P}$. Da $\mathbf{P}$ eine Permutationsmatrix ist, genügt eine Transponierung, um zwischen den beiden Konventionen zu wechseln. (Der Grund dafür wird in einem späteren Kapitel behandelt.)

Ausserdem können verschiedene Implementierungen oder verschiedene mitgeführte Parameter zu unterschiedlichen Permutationsmatrizen führen. Somit ist $\mathbf{P}$ selbst bei der gleichen Konvention nicht eindeutig.
⌟

Die Matrix $\mathbf{A}$ wurde also in eine untere Dreiecksmatrix $\mathbf{L}$ und eine Matrix mit oberer Zeilenstufenform $\mathbf{U}$ zerlegt; dies wird *LU-Zerlegung* oder auch *LR-Zerlegung* einer Matrix genannt.

**Bemerkung 1.6.0.3 (LGS löst man mit der LU-Zerlegung, nicht mit der Inversen)**
Wo müssen wir bei der Gauss-Elimination am meisten Rechenarbeit leisten?
Wenn ein LGS $\mathbf{Ax} = \mathbf{b}$ gelöst werden muss, werden die folgenden Schritte gemacht:

$$\mathbf{Ax} = \mathbf{b} \xrightarrow{\mathbf{P}|} \mathbf{PAx} = \mathbf{Pb} \implies \mathbf{L} \underbrace{\mathbf{Ux}}_{\mathbf{y}} = \mathbf{Pb}\,.$$

Der erste Schritt, in dem die Permutationsmatrix mit beiden Seiten multipliziert wird, ist billig (die Speichereinträge werden natürlich nicht umherkopiert, nur die Indizes/Zeiger werden umgeschrieben). Der zweite Schritt, in dem die Gauss-Elimination angewendet wird, um $\mathbf{A}$ in die beiden Matrizen $\mathbf{L}$ und $\mathbf{U}$ zu zerlegen, ist teuer; $\mathcal{O}(n^3)$ Operationen für $n \times n$ Matrizen. Nach dieser Zerlegung sind alle weiteren Berechnungen billig, da $\mathbf{Ux}$ durch $\mathbf{y}$ ersetzt wird und wir dann nur noch zwei einfache LGS haben.

$\mathbf{Ly} = \mathbf{Pb}$ kann in $\mathcal{O}(n^2)$ Operationen via Substitution gelöst werden, da $\mathbf{L}$ eine untere Dreiecksmatrix ist. Dies ist also für grosse Matrizen viel billiger als die Gauss-Elimination.

Um $\mathbf{x}$ zu berechnen, muss das LGS $\mathbf{Ux} = \mathbf{y}$ ausgerechnet werden. Dies wird auch wieder in $\mathcal{O}(n^2)$ Operationen via Substitution gelöst, ist also auch relativ billig.

Dieses Verfahren für das Lösen von linearen Gleichungssystemen ist billiger als das Verfahren über die Berechnung der Inverse. Wichtig anzumerken ist, dass selbst, wenn bereits die Inverse berechnet ist, kein Vorteil gegenüber der LU-Zerlegung besteht.

Nehmen wir an wir haben bereits sowohl eine Inverse $\mathbf{A}^{-1}$ als auch eine LU-Zerlegung $\mathbf{A} = \mathbf{LU}$. Um das Gleichungssystem $\mathbf{Ax} = \mathbf{b}$ zu lösen haben wir als zwei Möglichkeiten:

1. Wir führen, wie zuvor beschrieben, eine Vorwärtsubstitution und eine Rückwärtssubsitution durch. Das kostet $\mathcal{O}(n^2)$ Operationen.
2. Wir multiplizieren mit der Inversen $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$. Das kostet ebenfalls $\mathcal{O}(n^2)$ Operationen.

Der höhere Rechenaufwand bei der Berechnung der Inversen amortisiert sich selbst dann nicht, wenn viele (z.B. $\mathcal{O}(n)$) Gleichungssysteme gelöst werden müssen.[9]

⌟

[9] Diese Aussage gilt selbst dann, wenn die exakte Anzahl an durchgeführter Operationen und nicht nur die asymptotischen Kosten betrachtet werden.

Die beiden Aussagen $\mathbf{A} = \mathbf{L}\,\mathbf{U}$ und $\mathbf{P}\,\mathbf{A} = \mathbf{L}\,\mathbf{U}$ müssen wir noch beweisen.

Wir beginnen mit einem einfachen Beispiel[10]. Wir wollen einen Schritt in der Gauss-Elimination mittels einer Matrix-Vektor-Multiplikation realisieren. Der Schritt

$$\mathbf{b} = \begin{bmatrix} 2 \\ 8 \\ 10 \end{bmatrix} \begin{matrix} \\ |-2 \\ \swarrow \end{matrix} \implies \mathbf{c} = \begin{bmatrix} 2 \\ 4 \\ 10 \end{bmatrix}$$

soll nun mittels einer Matrix gemacht werden. Wir suchen also eine Matrix $\mathbf{E}$, welche die erste Zeile des Vektors mit $-2$ multipliziert und dann zur Zweiten addiert. Solch eine Matrix zu finden ist relativ einfach und man kommt auf eine Matrix welche der Identität entspricht, ausser im Eintrag der ersten Spalte und zweiten Zeile, dieser ist $-2$:

$$\mathbf{E} = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \implies \mathbf{E}\mathbf{b} = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ -2b_1 + b_2 \\ b_3 \end{bmatrix}.$$

Wenn wir diese `Eliminationsmatrix` nicht mit einem Vektor, sondern mit einer anderen Matrix $\mathbf{A}$ multiplizieren, so entspricht dies der Multiplikation von $\mathbf{E}$ mit jeder Spalte von $\mathbf{A}$. Der erste Eintrag dieser Spaltenvektoren wird mit $-2$ multipliziert und zum zweiten addiert:

$$\mathbf{E}\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ -2a_{11} + a_{21} & -2a_{12} + a_{22} & -2a_{13} + a23 \\ a_{31} & a_{32} & a_{33} \end{bmatrix}.$$

Wir sehen also: Die Matrix $\mathbf{E}$ multipliziert die erste Zeile von $\mathbf{A}$ mit $-2$ und addiert sie zu der zweiten Zeile.

**Definition 1.6.0.4. Eliminationsmatrix $\mathbf{E}_{ij}$**

$\mathbf{E}_{ij}$ ist die Matrix, welche man aus der Identitätsmatrix $\mathbf{I}$ mit $l_{ij}$ an der Stelle $ij$ erhält.

$\mathbf{E}_{ij}\mathbf{A}$ realisiert den Schritt in der Gauss-Elimination, in welchem die $j$-te Gleichung mit $l_{ij}$ multipliziert wird und zur $i$-ten Gleichung hinzuaddiert wird.

Bevor wir mit dem tatsächlichen Beweis beginnen, geben wir noch zwei Rechenregeln:

**Bemerkung 1.6.0.5 (Operationen mit Eliminationsmatrizen)**
**1) Die Inverse von $\mathbf{E}_{ij}$:**
Die Inverse $\mathbf{E}_{ij}^{-1}$ von $\mathbf{E}_{ij}$ ist identisch mit $\mathbf{E}_{ij}$, ausser dass der Eintrag $l_{ij}$ zu $-l_{ij}$ wird:

$$\mathbf{E}_{ij} = \begin{bmatrix} 1 & 0 & \cdots & \cdots & 0 \\ 0 & 1 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & l_{ij} & 0 & 1 & 0 \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & 1 \end{bmatrix} \implies \mathbf{E}_{ij}^{-1} = \begin{bmatrix} 1 & 0 & \cdots & \cdots & 0 \\ 0 & 1 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & -l_{ij} & 0 & 1 & 0 \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & 1 \end{bmatrix}$$

Dies können wir einfach durch Ausrechnen beweisen. Bei der Multiplikation $\mathbf{E}_{ij}\mathbf{E}_{ij}^{-1}$ ergeben sich offensichtlich als Einträge Einsen und Nullen, ausser wenn die $j$-te Spalte aus $\mathbf{E}_{ij}^{-1}$ mit der $i$-ten Zeile aus $\mathbf{E}_{ij}$ multipliziert wird. In diesem Fall erhält man aber $l_{ij} - l_{ij}$ also $0$ und demzufolge ist das Ergebnis die

[10] Beispiel aus Kapitel 2.3, Seite 57 aus Str09.

Identitätsmatrix $\mathbf{E}_{ij}\mathbf{E}_{ij}^{-1} = \mathbf{I}$.

**2) Multiplikation von $\mathbf{E}_{ij}$ mit $\mathbf{E}_{kr}$:**
Die Multiplikation zweier solcher Matrizen resultiert in einer Identitätsmatrix in der die Einträge $ij$ und $kr$ definiert sind durch $l_{ij}$ und $l_{kr}$ aus $\mathbf{E}_{ij}$ und $\mathbf{E}_{kr}$:

$$\mathbf{E}_{ij}\mathbf{E}_{kr} = \begin{bmatrix} 1 & 0 & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & & \vdots \\ \vdots & \ddots & l_{ij} & \ddots & \vdots \\ \vdots & & \ddots & l_{kr} & 0 \\ 0 & \cdots & \cdots & 0 & 1 \end{bmatrix}.$$

⌟

*Beweis.* Wir sind nun in der Lage zu beweisen, dass die Gauss-Elimination (ohne Permutationen) die LU-Zerlegung liefert.

Wir schreiben die Gauss-Elimination (ohne Permutationen) nur mit solchen Matrixmultiplikationen: wir haben eine Kette von Multiplikationen mit $\mathbf{E}_{ij}$ Matrizen, wobei jede einen Schritt der Gauss-Elimination darstellt:

$$\mathbf{E}_{n-1\,n-1} \ldots \mathbf{E}_{31}\mathbf{E}_{21}\mathbf{A} = \mathbf{U}\,.$$

Diese Kette von Matrizen auf der linken Seite kann, mittels wiederholtem multiplizieren der Inversen, auf die rechte Seite gebracht werden:

$$\begin{aligned}
&\mathbf{E}_{n-1\,n-1}\,\mathbf{E}_{n-1\,n-2}\,\ldots\,\mathbf{E}_{31}\,\mathbf{E}_{21}\,\mathbf{A} = \mathbf{U} \\
\xrightarrow{\mathbf{E}_{n-1\,n-1}^{-1}\cdot}\;&\mathbf{E}_{n-1\,n-2}\,\ldots\,\mathbf{E}_{31}\,\mathbf{E}_{21}\,\mathbf{A} = \mathbf{E}_{n-1\,n-1}^{-1}\,\mathbf{U} \\
\xrightarrow{\mathbf{E}_{n-1\,n-2}^{-1}\cdot}\;&\mathbf{E}_{n-1\,n-3}\,\ldots\,\mathbf{E}_{31}\,\mathbf{E}_{21}\,\mathbf{A} = \mathbf{E}_{n-1\,n-2}^{-1}\,\mathbf{E}_{n-1\,n-1}^{-1}\,\mathbf{U} \\
&\ldots \\
\xrightarrow{\mathbf{E}_{31}^{-1}\cdot}\;&\mathbf{E}_{21}\,\mathbf{A} = \mathbf{E}_{31}^{-1}\,\ldots\,\mathbf{E}_{n-1\,n-2}^{-1}\,\mathbf{E}_{n-1\,n-1}^{-1}\,\mathbf{U} \\
\xrightarrow{\mathbf{E}_{21}^{-1}\cdot}\;&\mathbf{A} = \mathbf{E}_{21}^{-1}\,\mathbf{E}_{31}^{-1}\,\ldots\,\mathbf{E}_{n-1\,n-2}^{-1}\,\mathbf{E}_{n-1\,n-1}^{-1}\,\mathbf{U}\,.
\end{aligned}$$

Die zweite Rechenregel aus 1.6.0.5 fasst die Matrixkette auf der rechten Seite in eine einzelne untere Dreiecksmatrix $\mathbf{L}$ zusammen. Diese Matrix entspricht genau der Protokollmatrix der Gauss-Elimination:

$$\mathbf{E}_{21}^{-1}\,\mathbf{E}_{31}^{-1}\,\ldots\,\mathbf{E}_{n-1\,n-2}^{-1}\,\mathbf{E}_{n-1\,n-1}^{-1} = \mathbf{L}\,.$$

$\mathbf{A}$ wurde als das Produkt der unteren Dreiecksmatrix $\mathbf{L}$ mit der oberen Dreiecksmatrix $\mathbf{U}$ geschrieben:

$$\mathbf{A} = \mathbf{L}\,\mathbf{U}\,.$$

Die LU-Zerlegung entspricht also der Gauss-Elimination mit der Verwendung einer Protokollmatrix.

□

**Bemerkung 1.6.0.6 (Die LU-Zerlegung (ohne Permutationen) ist eindeutig!)**
Die Matrix $\mathbf{L}$ ist aus Multiplikationen von Eliminationsmatrizen entstanden. Nach demselben Prinzip kann man einfach zeigen, dass die Multiplikation von unteren Dreiecksmatrizen mit Einsen auf der Diagonale ($\mathbf{L}_1$ und $\mathbf{L}_2$) auch eine untere Dreiecksmatrix ist. Die Inverse einer unteren Dreiecksmatrix $\mathbf{L}$ ist auch eine untere Dreiecksmatrix mit Einsen auf der Diagonale. Analoges gilt auch für obere Dreiecksmatrizen. Diese Erkenntnisse kann man kombinieren, um zu beweisen, dass die LR-Zerlegung eindeutig ist, d. h., wenn $\mathbf{A} = \mathbf{L}_1\mathbf{R}_1 = \mathbf{L}_2\mathbf{R}_2$, dann $\mathbf{L}_1 = \mathbf{L}_2$ und $\mathbf{R}_1 = \mathbf{R}_2$.

⌟

Die LU-Zerlegung mit Permutationen besprechen wir nicht rigoros, sondern wir veranschaulichen sie nur an einem Beispiel.

**Beispiel 1.6.0.7 (LU-Zerlegung mit Permutationen.)**

Seien eine Matrix $\mathbf{A}$ und ein Vektor $\mathbf{b}$ gegeben als

$$\mathbf{A} = \begin{bmatrix} 0 & 2 & 1 \\ 4 & 6 & 1 \\ 2 & 3 & 4 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 7 \\ 19 \\ 20 \end{bmatrix}.$$

Mit dieser Matrix und diesem Vektor wollen wir das LGS $\mathbf{Ax} = \mathbf{b}$ lösen.

Wenn wir mit der Gauss-Elimination beginnen, so bemerken wir, dass $a_{11}$ Null ist, und dementsprechend kein Pivot sein kann. Deswegen vertauschen wir die erste Zeile mit einer Anderen, zum Beispiel der Dritten.

$$\begin{bmatrix} 0 & 2 & 1 \\ 4 & 6 & 1 \\ 2 & 3 & 4 \end{bmatrix}.$$

Wir realisieren einen Zeilentausch mit einer *Permutationsmatrix* $\mathbf{P}$ , die mit $\mathbf{A}$ multipliziert wird. Die *Permutationsmatrix* entsteht aus der Identitätsmatrix: Man vertauscht in ihr die Zeilen, die in $\mathbf{A}$ vertauscht werden sollen. Hier sind ein paar Beispiele von $3 \times 3$ Permutationsmatrizen:

$$\mathbf{P}_{13} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \qquad \mathbf{P}_{12} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \mathbf{P}_{23} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

Aus ihrer Definition sieht man, dass die Inverse einer Permutationsmatrix ihre Transponierte ist:

$$\mathbf{P}_{ij}^{-1} = \mathbf{P}_{ij}^{\mathsf{T}},$$

und dass für diese einfache Permutation (Vertauschung von genau zwei Zeilen aus der Identitätsmatrix) auch $\mathbf{P}_{ij}^{\mathsf{T}} = \mathbf{P}_{ij}$ gilt, so dass $\mathbf{P}_{ij}^{-1} = \mathbf{P}_{ij}$.

Die Wirkung einer Permutationsmatrix hängt davon ab, von welcher Seite sie mit einer anderen Matrix multipliziert wird:

$\mathbf{P}_{13}\,\mathbf{B}$ vertauscht die *Zeilen* $1,\ 3$ in der Matrix $\mathbf{B}$,

$\mathbf{B}\,\mathbf{P}_{13}$ vertauscht die *Spalten* $1,\ 3$ in der Matrix $\mathbf{B}$.

Dies wird später in diesem Beispiel noch verwendet.

Wenn wir die Matrix $\mathbf{A}$ mit $\mathbf{P}_{13}$ von links multiplizieren, dann wird die erste und dritte Zeile vertauscht und wir realisieren somit den ersten Schritt der Gauss-Elimination:

$$\mathbf{P}_{13}\,\mathbf{A} = \begin{bmatrix} 2 & 3 & 4 \\ 4 & 6 & 1 \\ 0 & 2 & 1 \end{bmatrix}.$$

Die Gauss-Elimination verwendet $\mathbf{E}_{ij}$-Matrizen, um Zeilen zu addieren und Permutationsmatrizen $\mathbf{P}_{ij}$ um sie zu vertauschen. Die Multiplikation mit der Matrix $\mathbf{E}_{21}$ realisiert eine $0$ an der Stelle $(2,1)$:

$$\mathbf{P}_{13}\,\mathbf{A} = \begin{bmatrix} 2 & 3 & 4 \\ 4 & 6 & 1 \\ 0 & 2 & 1 \end{bmatrix} \begin{matrix} \\ \\ \end{matrix} \quad \begin{matrix} \\ \\ |-2 \end{matrix} \qquad \mathbf{E}_{21} = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

was in diesem Fall auch eine $0$ an der Stelle $(2,2)$ zur Folge hat, so dass wir danach die Permutation $\mathbf{P}_{23}$ brauchen, die die Zeilen $2$ und $3$ vertauscht:

$$\mathbf{E}_{21}\,\mathbf{P}_{13}\,\mathbf{A} = \begin{bmatrix} 2 & 3 & 4 \\ 0 & 0 & -7 \\ 0 & 2 & 1 \end{bmatrix} \quad \begin{matrix} \\ \updownarrow \\ \end{matrix} \qquad \mathbf{P}_{23} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

Alle Schritte der Gauss-Elimination zusammen ergeben also:

$$\mathbf{P}_{23}\,\mathbf{E}_{21}\,\mathbf{P}_{13}\,\mathbf{A} = \begin{bmatrix} 2 & 3 & 4 \\ 0 & 2 & 1 \\ 0 & 0 & -7 \end{bmatrix} = \mathbf{U}\,.$$

Wir haben nun also die gewünschte Matrix $\mathbf{U}$ und alle Schritte der Gauss-Elimination, in Form von Permutationsmatrizen $\mathbf{P}_{ij}$ und $\mathbf{E}_{ij}$ Matrizen, erhalten. Das Ziel ist aber auch alle Permutationsmatrizen in die Matrix $\mathbf{P}$ und alle $\mathbf{E}_{ij}$ in die Matrix $\mathbf{L}$ zusammenzufassen. Dies ist aber nur möglich, wenn alle Permutationsmatrizen nebeneinander und alle $\mathbf{E}_{ij}$-Matrizen nebeneinanderstehen. Da die Matrixmultiplikation nicht kommutativ ist, können ihre Positionen aber nicht einfach vertauscht werden.
Die Lösung dieses Problems steckt im Resultat der Multiplikation von $\mathbf{P}_{23}$ mit $\mathbf{E}_{21}$.

$$\mathbf{P}_{23}\,\mathbf{E}_{21} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ -2 & 1 & 0 \end{bmatrix}$$

Diese Matrix hat also ihren Teil aus der Identitätsmatrix verloren. Dieser kann aber wieder zurückgewonnen werden, indem die $2$-te und $3$-te Spalte vertauscht werden. Diese Vertauschung kann durch die Multiplikation mit $\mathbf{P}_{23}$ von rechts erreicht werden:

$$\mathbf{P}_{23}\,\mathbf{E}_{21} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ -2 & 1 & 0 \end{bmatrix} \quad \begin{matrix} \\ \\ \curvearrowright \end{matrix} \implies \mathbf{P}_{23}\,\mathbf{E}_{21}\,\mathbf{P}_{23} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -2 & 0 & 1 \end{bmatrix} = \mathbf{E}_{31} \implies \mathbf{P}_{23}\,\mathbf{E}_{21} = \mathbf{E}_{31}\,\mathbf{P}_{23}^{\mathsf{T}} = \mathbf{E}_{31}\,\mathbf{P}_{23}\,.$$

Die Position der E-Matrix und der Permutationsmatrix wurde vertauscht: Statt dem Produkt $\mathbf{P}_{23}\,\mathbf{E}_{21}$ hat man nun $\mathbf{E}_{31}\,\mathbf{P}_{23}$. Bei einer grösseren Matrix, welche mehr Permutationen und E-Matrizen hat, kann das Gleiche

wieder gemacht werden. Wir erhalten eine Gleichung, in der alle $\mathbf{P}_{ij}$ und alle $\mathbf{E}_{ij}$ nebeneinander stehen. So können sie in $\mathbf{P}$ und $\mathbf{L}$ zusammengefasst werden. In unserem Beispiel geht das so:

$$\mathbf{E}_{31}\underbrace{\mathbf{P}_{23}\mathbf{P}_{13}}_{\mathbf{P}}\mathbf{A}=\mathbf{U}$$
$$\mathbf{P}\mathbf{A}=\underbrace{\mathbf{E}_{31}^{-1}}_{\mathbf{L}}\mathbf{U}$$
$$\mathbf{P}\mathbf{A}=\mathbf{L}\mathbf{U}\,.$$

Unter der Berücksichtigung von allfälligen Permutationen in der Matrix $\mathbf{A}$, wurde sie in das Produkt der Matrix $\mathbf{L}$ und $\mathbf{U}$ zerlegt.

⌟

Was hier an einer $3\times 3$ Matrix gezeigt wurde, funktioniert auch für beliebige $m\times n$ Matrizen, wobei der Fall $m\neq n$ mehr Aufmerksamkeit bei den Vertauschungen erfordert.

Die Beispiele 1.2.0.17 und 1.2.0.18 enthielten solche Vertauschungen. Es lohnt sich, diese aus der Perspektive der Matrixmultiplikation anzuschauen. Notieren wir mit $\mathbf{E}_{ij}(a)$ die Eliminationsmatrix, die $a$ an der Stelle $(ij)$ hat ($i>j$).

Im Beispiel 1.2.0.17 kommt genau eine Vertauschung vor:

$$\begin{aligned}\mathbf{P}_{23}\mathbf{E}_{51}(-1)\mathbf{E}_{31}(-2)\mathbf{E}_{21}(1)&=\mathbf{E}_{51}(-1)\mathbf{P}_{23}\mathbf{E}_{31}(-2)\mathbf{E}_{21}(1)=\\ \mathbf{E}_{51}(-1)\mathbf{E}_{21}(-2)\mathbf{P}_{23}\mathbf{E}_{21}(1)&=\mathbf{E}_{51}(-1)\mathbf{E}_{21}(-2)\mathbf{E}_{31}(1)\mathbf{P}_{23}\,,\end{aligned}$$

was genau der Vertauschung der entsprechenden Zeilen in der Kontrollmatrix (ohne den Diagonalelementen) entspricht. Die Diagonalelemente werden bei der Vertauschung der Eliminationsmatrix mit der Permutationsmatrix zur eigenen Stelle wieder zurückgelegt.

Im Beispiel 1.2.0.18 kommen drei Vertauschungen vor. Zuerst:

$$\begin{aligned}\mathbf{P}_{23}\mathbf{E}_{61}(1)\mathbf{E}_{41}(1)\mathbf{E}_{21}(1)&=\\ \mathbf{E}_{61}(1)\mathbf{P}_{23}\mathbf{E}_{41}(1)\mathbf{E}_{21}(1)&=\\ \mathbf{E}_{61}(1)\mathbf{E}_{41}(1)\mathbf{P}_{23}\mathbf{E}_{21}(1)&=\\ \mathbf{E}_{61}(1)\mathbf{E}_{41}(1)\mathbf{E}_{31}(1)\mathbf{P}_{23}\,,&\end{aligned}$$

was genau der Vertauschung der zweiten und dritten Zeilen in der Kontrollmatrix (ohne den Diagonalelementen) entspricht. Die zweite Vertauschung ist

$$\begin{aligned}&\mathbf{P}_{34}\mathbf{E}_{72}(1)\mathbf{E}_{62}(2)\mathbf{E}_{52}(1)\mathbf{E}_{32}(2)\mathbf{E}_{61}(1)\mathbf{E}_{41}(1)\mathbf{E}_{31}(1)\mathbf{P}_{23}\\ \Longrightarrow\ &\mathbf{E}_{72}(1)\mathbf{E}_{62}(2)\mathbf{E}_{52}(1)\mathbf{E}_{42}(2)\mathbf{E}_{61}(1)\mathbf{E}_{31}(1)\mathbf{E}_{41}(1)\mathbf{P}_{34}\mathbf{P}_{23}\end{aligned}$$

was genau der Vertauschung der dritten und vierten Zeilen in der Kontrollmatrix (ohne den Diagonalelementen) entspricht. Die dritte Vertauschung ist

$$\begin{aligned}&\mathbf{P}_{45}\mathbf{E}_{73}(2)\mathbf{E}_{63}(3)\mathbf{E}_{53}(3)\mathbf{E}_{43}(1)\mathbf{E}_{72}(1)\mathbf{E}_{62}(2)\mathbf{E}_{52}(1)\mathbf{E}_{42}(2)\mathbf{E}_{61}(1)\mathbf{E}_{31}(1)\mathbf{E}_{41}(1)\mathbf{P}_{34}\mathbf{P}_{23}\,,\\ \Longrightarrow\ &\mathbf{E}_{73}(2)\mathbf{E}_{63}(3)\mathbf{E}_{43}(3)\mathbf{E}_{53}(1)\mathbf{E}_{72}(1)\mathbf{E}_{62}(2)\mathbf{E}_{42}(1)\mathbf{E}_{52}(2)\mathbf{E}_{61}(1)\mathbf{E}_{31}(1)\mathbf{E}_{51}(1)\mathbf{P}_{45}\mathbf{P}_{34}\mathbf{P}_{23}\,,\end{aligned}$$

so dass eine Vertauschung der vierten und fünften Zeilen in der Kontrollmatrix (ohne den Diagonalelementen) entsteht. Letztendlich haben wir

$$\mathbf{PA}=\mathbf{LU}$$

mit

$$\mathbf{P} = \mathbf{P}_{45}\mathbf{P}_{34}\mathbf{P}_{23}\,.$$

Die Permutationsmatrix ist in diesem Fall also:

$$= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Die Faktoren sind:

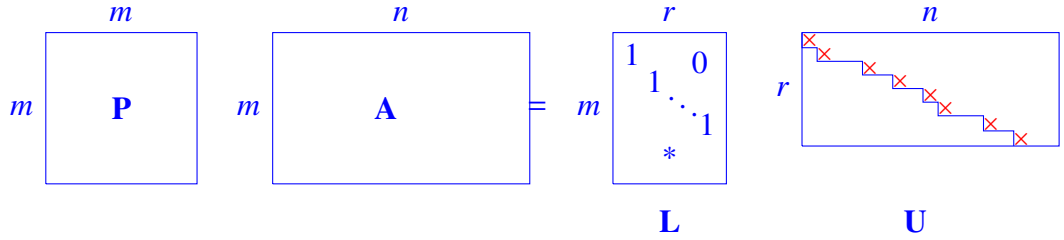
$$\mathbf{L} = \left[\begin{array}{ccccc|cc} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 3 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 4 & 1 & 0 & 0 \\ \hline 1 & 2 & 3 & 2 & 1 & 1 & 0 \\ 0 & 1 & 2 & 3 & 2 & 0 & 1 \end{array}\right]$$

und

$$\mathbf{U} = \begin{bmatrix} 1 & 0 & 5 & 0 & 4 & 0 & 0 & 1 & 0 \\ 0 & 5 & 4 & 3 & 2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & 4 & 3 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 5 & 3 & 2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Da die letzten $m - r$ Spalten von $\mathbf{L}$ den letzten $m - r$ Nullzeilen (Kompatibilitätsbedingungen) in $\mathbf{U}$ entsprechen, kann man die LU-Zerlegung sparsamer schreiben, indem man auf diese Spalten und Zeilen verzichtet.

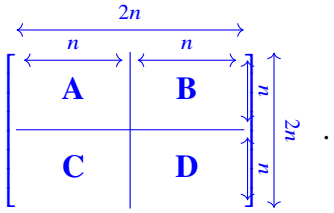
**Bemerkung 1.6.0.8 (Allgemeine sparsame Form der LU-Zerlegung)**
Die Gauss-Elimination einer beliebigen $m \times n$ Matrix lässt sich auch als eine LU-Zerlegung schreiben.
Eine $m \times n$ Matrix mit Rang $= r < m$ lässt sich folgendermassen aufteilen:




⌟

**Bemerkung 1.6.0.9 (Blockweise Gauss-Elimination)**
Bis jetzt haben wir die Gauss-Elimination immer elementweise gemacht, wir können sie aber auch blockweise durchführen.

Sei eine $2n \times 2n$ Matrix gegeben. Diese Matrix wird in vier Untermatrizen, sogenannte Blöcke, der Dimensionen $n \times n$ aufgeteilt:




Die Gauss-Elimination hat das Ziel die Einträge unterhalb der Hauptdiagonale Null werden zu lassen. Wir können hier dasselbe tun, ausser dass wir nicht die Einträge, sondern die Blöcke verwenden. Der Eintrag unten links, also der Block $\mathbf{C}$, soll Null werden. Um dies zu erreichen multiplizieren wir die erste Zeile mit $-\mathbf{C}\,\mathbf{A}^{-1}$ und addieren es zur zweiten Zeile hinzu (natürlich nur, wenn die kleinere Matrix $\mathbf{A}$ invertierbar ist):

$$\left[\begin{array}{c|c} \mathbf{I} & \mathbf{0} \\ \hline -\mathbf{C}\mathbf{A}^{-1} & \mathbf{I} \end{array}\right]\left[\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array}\right].$$

Diese Multiplikation können wir auch blockweise ausführen.

$$\left[\begin{array}{c|c} \mathbf{I} & \mathbf{0} \\ \hline -\mathbf{C}\mathbf{A}^{-1} & \mathbf{I} \end{array}\right]\left[\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array}\right] = \left[\begin{array}{c|c} \mathbf{I}\mathbf{A}+\mathbf{0}\mathbf{C} & \mathbf{I}\mathbf{B}+\mathbf{0}\mathbf{D} \\ \hline -\mathbf{C}\mathbf{A}^{-1}\mathbf{A}+\mathbf{I}\mathbf{C} & -\mathbf{C}\mathbf{A}^{-1}\mathbf{B}+\mathbf{I}\mathbf{D} \end{array}\right].$$

Dies können wir nun weiter vereinfachen:

$$\left[\begin{array}{c|c} \mathbf{I}\mathbf{A}+\mathbf{0}\mathbf{C} & \mathbf{I}\mathbf{B}+\mathbf{0}\mathbf{D} \\ \hline -\mathbf{C}\mathbf{A}^{-1}\mathbf{A}+\mathbf{I}\mathbf{C} & -\mathbf{C}\mathbf{A}^{-1}\mathbf{B}+\mathbf{I}\mathbf{D} \end{array}\right] = \left[\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{0} & \mathbf{D}-\mathbf{C}\mathbf{A}^{-1}\mathbf{B} \end{array}\right].$$

Das gewünschte Resultat wurde erreicht, alle Einträge unterhalb der Hauptdiagonale sind Null. Der Block unten rechts mit dem Eintrag $\mathbf{D}-\mathbf{C}\mathbf{A}^{-1}\mathbf{B}$ wird *Schur Komplement* genannt. Das ist ein wichtiger Begriff und wird in anderen Vorlesungen vorkommen (z. B. Numerik), wird in dieser Vorlesung aber nicht weiter gebraucht.

⌟

**Bemerkung 1.6.0.10 (Sherman-Morrison-Woodbury Formel)** Nehmen wir an, wir können das $n \times n$ lineare Gleichungsystem $\mathbf{A}\mathbf{x} = \mathbf{b}$ leicht lösen; die Aufgabe ist nun $\widetilde{\mathbf{A}}\widetilde{\mathbf{x}} = \mathbf{b}$ mit

$$\widetilde{\mathbf{A}} = \mathbf{A} + \mathbf{u}\mathbf{v}^{\mathsf{H}}$$

mit beliebigen $\mathbf{u}, \mathbf{b}$ der Länge $n$. Mit anderen Worten: kann man die Information von $\mathbf{A}^{-1}$ wiederverwenden, um effizient eine Rang-1-Modifitierte Matrix $\widetilde{\mathbf{A}}$ zu invertieren? Die Sherman-Morrison-Woodbury Formel gibt die Lösung. Schauen wir uns, wie wir diese Formel einfach mit Block-Elimination erhalten können. Zuerst schreiben wir die Gleichung $\widetilde{\mathbf{A}}\widetilde{\mathbf{x}} = \mathbf{b}$ in Blockform um, indem wir eine zusätzliche Unbekannte $\xi \in \mathbb{C}$ einführen. Ein Gauss-Eliminationschritt für die erste Spalte in

$$\begin{bmatrix} -1 & \mathbf{v}^{\mathsf{H}} \\ \mathbf{u} & \mathbf{A} \end{bmatrix}\begin{bmatrix} \xi \\ \widetilde{\mathbf{x}} \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{b} \end{bmatrix} \tag{1.6.0.11}$$

liefert

$$\begin{bmatrix} -1 & \mathbf{v}^H \\ 0 & \mathbf{A}+\mathbf{u}\mathbf{v}^H \end{bmatrix} \begin{bmatrix} \xi \\ \widetilde{\mathbf{x}} \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{b} \end{bmatrix} . \tag{1.6.0.12}$$

Die zweite Blockgleichung ist genau $\widetilde{\mathbf{A}}\widetilde{\mathbf{x}} = \mathbf{b}$, so reicht es das Blocksystem zu lösen. Vertauschen wir nun die Blockzeilen:

$$\begin{bmatrix} \mathbf{A} & \mathbf{u} \\ \mathbf{v}^H & -1 \end{bmatrix} \begin{bmatrix} \widetilde{\mathbf{x}} \\ \xi \end{bmatrix} = \begin{bmatrix} \mathbf{b} \\ 0 \end{bmatrix} . \tag{1.6.0.13}$$

Ein Gauss-Eliminationschritt für die erste Spalte liefert die Gleichung mit dem Schur-Komplement für $\xi$:

$$(1+\mathbf{v}^H\mathbf{A}^{-1}\mathbf{u})\xi = \mathbf{v}^H\mathbf{A}^{-1}\mathbf{b} \tag{1.6.0.14}$$

und die Rücksubstitution von $\xi$ liefert das Gleichungsystem für $\widetilde{\mathbf{x}}$ mit der unmodifizierter Matrix $\mathbf{A}$:

$$\mathbf{A}\widetilde{\mathbf{x}} = \mathbf{b} - \frac{\mathbf{u}\mathbf{v}^H\mathbf{A}^{-1}}{1+\mathbf{v}^H\mathbf{A}^{-1}\mathbf{u}}\mathbf{b} . \tag{1.6.0.15}$$

Für Algorithmen (wo die Inverse niemals berechenet wird) reicht dieses lineare Gleichungssystem zu lösen. Die daraus resultierende *Sherman-Morrison-Woodbury Formel* für die Inverse der Rang-1-Modifizerte Matrix ist:

$$(\mathbf{A}+\mathbf{u}\mathbf{v}^H)^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\mathbf{u}\mathbf{v}^H\mathbf{A}^{-1}}{1+\mathbf{v}^H\mathbf{A}^{-1}\mathbf{u}} . \tag{1.6.0.16}$$

⌟

Die natürliche Erweiterung zu Rang-$k$-Modifikationen ist:

**Lemma 1.6.0.17. Sherman-Morrison-Woodbury Formel**

*Für reguläre $\mathbf{A} \in \mathbb{C}^{n,n}$, und $\mathbf{U}, \mathbf{V} \in \mathbb{C}^{n,k}$, $n, k \in \mathbb{N}$, $k \le n$, gilt*

$$(\mathbf{A}+\mathbf{U}\mathbf{V}^H)^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{U}(\mathbf{I}+\mathbf{V}^H\mathbf{A}^{-1}\mathbf{U})^{-1}\mathbf{V}^H\mathbf{A}^{-1} ,$$

***wenn*** $\mathbf{I}+\mathbf{V}^H\mathbf{A}^{-1}\mathbf{U}$ *ist regulär.*

was auch durch direkte Multiplikation leicht zu beweisen ist.

## Weitere Zerlegungen

[11]

### LDU-Zerlegung

Die LU-Zerlegung zerlegt eine Matrix $\mathbf{A}$ in die Matrizen $\mathbf{L}$ und $\mathbf{U}$. Die Matrix $\mathbf{L}$ ist eine untere Dreiecksmatrix welche nur Einsen auf der Hauptdiagonale hat, die Matrix $\mathbf{U}$ ist eine obere Dreiecksmatrix mit beliebigen Werten auf der Diagonale:

$$\mathbf{A} = \mathbf{L}\,\mathbf{U} = \begin{bmatrix} 1 & \cdot & 0 \\ * & \cdot & 1 \end{bmatrix} \begin{bmatrix} \diagdown & * \\ 0 & \diagdown \end{bmatrix} .$$

[11] Die Berechnung dieser Zerlegungen mit Python wird kurz im Abschnitt 9.2.6 aufgegriffen.

Die Matrix $\mathbf{U}$ hat nicht nur Einsen auf der Diagonale, deswegen wird eine neue Zerlegung definiert, die *LDU-Zerlegung*. In dieser Zerlegung wird eine Diagonalmatrix $\mathbf{D}$ eingeführt, welche die Diagonaleinträge von $\mathbf{U}$ auf der Diagonale hat. Des Weiteren wird die Matrix $\widetilde{\mathbf{U}}$ definiert mit der Eigenschaft, dass wenn sie mit $\mathbf{D}$ multipliziert wird, $\mathbf{U}$ ergibt. sie ist also die Lösung von $\mathbf{D}\,\widetilde{\mathbf{U}} = \mathbf{U}$. Somit hat man:

$$\mathbf{A} = \mathbf{L}\,\mathbf{U} = \mathbf{L}\,\mathbf{D}\,\widetilde{\mathbf{U}}\,.$$

Die Einträge von $\mathbf{D}$ sind die Diagonaleinträge $d_1,\ d_2, \ldots,\ d_n$ von $\mathbf{U}$. Die Einträge von $\widetilde{\mathbf{U}}$ entsprechen denen von $\mathbf{U}$, wobei die $i$-te Zeile aber durch das dazugehörige $d_i$ geteilt wurde:

$$\mathbf{D} = \begin{bmatrix} d_1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & d_n \end{bmatrix} \quad \widetilde{\mathbf{U}} = \begin{bmatrix} 1 & \frac{u_{12}}{d_1} & \cdots & \frac{u_{1n}}{d_1} \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \frac{u_{n-1\,n}}{d_{n-1}} \\ 0 & \cdots & 0 & 1 \end{bmatrix}.$$

Dies funktioniert aber nur unter der Annahme, dass alle Diagonalelemente $d_1, \ldots, d_n$ von $\mathbf{U}$ nicht Null sind:

$$d_1 \neq 0,\ d_2 \neq 0,\ \ldots,\ d_n \neq 0\,,$$

da wir nur in diesem Fall in der Lage sind die Matrix $\widetilde{\mathbf{U}}$ zu definieren, weil wir durch die Diagonaleinträge $d_i$ teilen. Dies ist genau dann der Fall, wenn die Matrix $\mathbf{A}$ vollen Rang hat, $\mathrm{Rang}(\mathbf{A}) = n$.

### LDU-Zerlegung symmetrischer Matrizen

Die transponierte Matrix $\mathbf{A}^\mathsf{T}$ von $\mathbf{A}$ lässt sich also schreiben als:

$$\mathbf{A}^\mathsf{T} = \widetilde{\mathbf{U}}^\mathsf{T}\,\mathbf{D}\,\mathbf{L}^\mathsf{T}\,.$$

Wenn wir auch noch die Annahme treffen, dass die Matrix $\mathbf{A}$ symmetrisch ist, dass also gilt $\mathbf{A} = \mathbf{A}^\mathsf{T}$ so haben wir $\mathbf{L}^\mathsf{T} = \widetilde{\mathbf{U}}$ und $\widetilde{\mathbf{U}}^\mathsf{T} = \mathbf{L}$, denn wir haben gesehen, dass die LU-Zerlegung einer regulären Matrix eindeutig ist.
In diesem Fall liefert die Gauss-Elimination die folgende Zerlegung einer symmetrischen regulären Matrix:

$$\mathbf{A} = \mathbf{L}\,\mathbf{D}\,\mathbf{L}^\mathsf{T}\,.$$

### Cholesky-Zerlegung

Wenn nun auch noch gilt, dass alle (Pivote in der Gauss-Elimination) $d_1, .., d_n > 0$, so kann für die symmetrische Matrix $\mathbf{A}$ die sogenannte *Cholesky-Zerlegung* definiert werden.

Für diese Zerlegung wird $\mathbf{D}$ in zwei identische Matrizen zerteilt $\mathbf{D} = \sqrt{\mathbf{D}}\,\sqrt{\mathbf{D}}$ mit

$$\sqrt{\mathbf{D}} = \begin{bmatrix} \sqrt{d_1} & & 0 \\ & \ddots & \\ 0 & & \sqrt{d_n} \end{bmatrix}.$$

Dies wird nun mit der LDU-Zerlegung von symmetrischen Matrix kombiniert um eine weitere Zerlegung zu erhalten:

$$\mathbf{A} = \mathbf{L}\,\mathbf{D}\,\mathbf{L}^\mathsf{T} = \underbrace{\mathbf{L}\,\sqrt{\mathbf{D}}}_{\mathbf{R}^\mathsf{T}}\,\underbrace{\sqrt{\mathbf{D}}\,\mathbf{L}^\mathsf{T}}_{\mathbf{R}} = \mathbf{R}^\mathsf{T}\,\mathbf{R}$$
$$\mathbf{A} = \mathbf{R}^\mathsf{T}\,\mathbf{R},\,.$$

Das $\mathbf{R}$ ist eine obere Dreiecksmatrix. Diese Zerlegung von $\mathbf{A}$ in $\mathbf{R}^\mathsf{T}\,\mathbf{R}$ ist die *Cholesky-Zerlegung*. Die Cholesky-Zerlegung hat wichtige Anwendungen in der Statistik, in der Physik und in der Numerik. Man kann die Symmetrie der Matrix $\mathbf{A}$ ausnutzen, um die Cholesky-Zerlegung mit nur einem Drittel der Operationen, die man für die standardmässige Gauss-Elimination braucht, zu realisieren.

# 1.7 Orthogonale Matrizen

## Vereinfachte Einführung von Norm und Skalarprodukt

Aus der Schule kennen viele wahrscheinlich bereits die Definition der Länge eines Vektors in $\mathbb{R}^3$. Was vermutlich damals mit $|\mathbf{x}|$ notiert war, wird nun mit $\|\mathbf{x}\|$ notiert und euklidische Norm genannt:

$$\mathbb{R}^3 \ni \mathbf{x}, \quad \|\mathbf{x}\| = \sqrt{x_1^2 + x_2^2 + x_3^2}.$$

Dies können wir einfach für Vektoren der Dimension $n$ erweitern:

$$\mathbb{R}^n \ni \mathbf{x}, \quad \|\mathbf{x}\| = \sqrt{x_1^2 + x_2^2 + \ldots + x_n^2}$$
$$\mathbb{C}^n \ni \mathbf{x}, \quad \|\mathbf{x}\| = \sqrt{|x_1|^2 + |x_2|^2 + \ldots + |x_n|^2}.$$

Auch ist vielen wahrscheinlich schon das euklidische Skalarprodukt zweier Vektoren in $\mathbb{R}^3$ bekannt. Auch hier wird eine neue Notation eingeführt: Das Skalarprodukt zwischen zwei Vektoren wird mit $\langle \cdot, \cdot \rangle$ notiert:

$$\mathbb{R}^3 \ni \mathbf{x},\ \mathbf{y}, \quad \langle \mathbf{x}, \mathbf{y} \rangle = x_1 y_1 + x_2 y_2 + x_3 y_3 = \mathbf{x}^\mathsf{T} \mathbf{y}.$$

Dies kann auch einfach für $n$-dimensionale Vektoren erweitert werden:

$$\mathbb{R}^n \ni \mathbf{x},\ \mathbf{y}, \quad \langle \mathbf{x}, \mathbf{y} \rangle = x_1 y_1 + x_2 y_2 + \ldots + x_n y_n = \mathbf{x}^\mathsf{T} \mathbf{y}$$
$$\mathbb{C}^n \ni \mathbf{x},\ \mathbf{y}, \quad \langle \mathbf{x}, \mathbf{y} \rangle = \overline{x}_1 y_1 + \overline{x}_2 y_2 + \ldots + \overline{x}_n y_n = \mathbf{x}^\mathsf{H} \mathbf{y}.$$

Mithilfe des Skalarprodukts ist man in der Lage den *Winkel zwischen zwei Vektoren* $\mathbf{x},\ \mathbf{y}$, notiert mit $\widehat{(\mathbf{x}, \mathbf{y})}$, zu definieren:

$$\widehat{(\mathbf{x}, \mathbf{y})} = \arccos \frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\|\mathbf{x}\| \|\mathbf{y}\|}.$$

Warum $\frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\|\mathbf{x}\| \|\mathbf{y}\|} \in [-1, 1]$ liegt (damit diese Definition überhaupt erlaubt ist), ist eine nette mathematische Aufgabe, die wir später ansprechen werden. Die Neugierigen können nach der Cauchy-Bouniakovsky-Schwarz-Ungleichung suchen. Aus dieser Definition des Winkels kann auch der *Cosinus Satz* abgeleitet werden:

$$\langle \mathbf{x}, \mathbf{y} \rangle = \|\mathbf{x}\| \|\mathbf{y}\| \cos \widehat{(\mathbf{x}, \mathbf{y})}.$$

Die allgemeine Definition und die Eigenschaften der Norm und des Skalarprodukts befinden sich im Kapitel 4.

## Orthogonalität

**Definition 1.7.0.1. Orthogonale Vektoren**

Zwei Vektoren $\mathbf{x},\ \mathbf{y} \in \mathbb{R}^n$ (keiner davon darf Null sein), heissen orthogonal, notiert als $\mathbf{x} \perp \mathbf{y}$, falls

$$\langle \mathbf{x}, \mathbf{y} \rangle = 0.$$

Wenn wir mehrere paarweise orthogonale Vektoren als die Spalten einer Matrix zusammenfassen und wir zusätzlich verlangen, dass jede dieser Spalten Norm 1 haben muss, bekommen wir eine wichtige und besondere Art von Matrix.

**Definition 1.7.0.2. Orthogonale Matrix**

Eine reelle $n \times n$ Matrix $\mathbf{A}$ heisst orthogonal, wenn

$$\mathbf{A}^\mathsf{T}\mathbf{A} = \mathbf{I}\,.$$

Eine komplexe $n \times n$ Matrix $\mathbf{A}$ heisst unitär, wenn

$$\mathbf{A}^\mathsf{H}\mathbf{A} = \mathbf{I}\,.$$

In anderen Worten, alle Spaltenvektoren einer orthogonalen/unitären Matrix sind paarweise orthogonal und haben Länge (Norm) 1.

**Beispiel 1.7.0.3 (Orthogonale Matrix)**
In diesem Beispiel wollen wir den Zusammenhang zwischen der Orthogonalität von Vektoren und der von Matrizen noch etwas veranschaulichen.

Sei eine orthogonale Matrix $\mathbf{A}$ mit den Spaltenvektoren $\mathbf{u},\ \mathbf{v},\ \mathbf{w}$ gegeben. Die Matrix $\mathbf{A}$ und ihre transponierte $\mathbf{A}^\mathsf{T}$ sind:

$$\mathbf{A} = \begin{bmatrix} | & | & | \\ \mathbf{u} & \mathbf{v} & \mathbf{w} \\ | & | & | \end{bmatrix}, \qquad \mathbf{A}^\mathsf{T} = \begin{bmatrix} \text{---} & \mathbf{u}^\mathsf{T} & \text{---} \\ \text{---} & \mathbf{v}^\mathsf{T} & \text{---} \\ \text{---} & \mathbf{w}^\mathsf{T} & \text{---} \end{bmatrix}.$$

Um die Orthogonalität dieser Matrix zu überprüfen, werden wir $\mathbf{A}^\mathsf{T}\mathbf{A}$ berechnen und gleich der Identitätsmatrix setzten:

$$\begin{bmatrix} \mathbf{u}^\mathsf{T}\mathbf{u} & \mathbf{u}^\mathsf{T}\mathbf{v} & \mathbf{u}^\mathsf{T}\mathbf{w} \\ \mathbf{v}^\mathsf{T}\mathbf{u} & \mathbf{v}^\mathsf{T}\mathbf{v} & \mathbf{v}^\mathsf{T}\mathbf{w} \\ \mathbf{w}^\mathsf{T}\mathbf{u} & \mathbf{w}^\mathsf{T}\mathbf{v} & \mathbf{w}^\mathsf{T}\mathbf{w} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Damit diese Gleichung stimmt, müssen die Vektoren $\mathbf{u},\ \mathbf{v},\ \mathbf{w}$ die folgenden Bedingungen erfüllen:

$$\begin{cases} \mathbf{u}^\mathsf{T}\mathbf{v} = 0,\ \mathbf{u}^\mathsf{T}\mathbf{w} = 0,\ \mathbf{v}^\mathsf{T}\mathbf{w} = 0 \\ \mathbf{u}^\mathsf{T}\mathbf{u} = 1,\ \mathbf{v}^\mathsf{T}\mathbf{v} = 1,\ \mathbf{w}^\mathsf{T}\mathbf{w} = 1 \end{cases} \iff \begin{cases} \mathbf{u} \perp \mathbf{v},\ \mathbf{u} \perp \mathbf{w},\ \mathbf{v} \perp \mathbf{w} \\ \|\mathbf{u}\| = \|\mathbf{v}\| = \|\mathbf{w}\| = 1 \end{cases}.$$

Damit die Matrix orthogonal ist, müssen also alle Spaltenvektoren von $\mathbf{A}$ orthogonal aufeinander stehen und Länge 1 haben.

⌟

**Definition 1.7.0.4. Matrix-Abbildung**

Gegeben sei eine Matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$, definieren wir eine dazu gehörige Abbildung $\mathscr{A}$:

$$\begin{aligned} \mathscr{A} : \mathbb{R}^n &\to \mathbb{R}^n \\ \mathbf{x} &\mapsto \mathbf{A}\mathbf{x} \end{aligned},$$

d. h. $\mathscr{A}(\mathbf{x})$ ist definiert als $\mathscr{A}(\mathbf{x}) = \mathbf{A}\mathbf{x} \in \mathbb{R}^n$ für $\mathbf{x} \in \mathbb{R}^n$.

**Bemerkung 1.7.0.5 (Matrix-Abbildung)**
Die Matrix und die dazugehörige Abbildung werden oft synonym verwendet, z. B., wenn wir sagen, dass eine Matrix Vektoren transformiert, meinen wir natürlich die dazu gehörige Abbildung; es geht um die Matrix-Vektor-Multiplikation. ⌟

**Satz 1.7.0.6. Erhaltungssatz**

*Orthogonale Matrizen verändern Längen und Winkel nicht.*

*Beweis.* Sei $\mathbf{A}$ eine beliebige orthogonale Matrix, also $\mathbf{A}^\top \mathbf{A} = \mathbf{I}$. Den Beweis, dass die Matrix Längen und Winkel nicht ändert, werden wir in drei Schritten führen.

1. Wir zeigen, dass diese Matrix das Skalarprodukt zwischen zwei Vektoren nicht ändert.
2. Wir erkennen, dass sich die Längen auch nicht ändern.
3. Wir verwenden die ersten beiden Schritte, um zu zeigen, dass sich die Winkel nicht ändern.

1. **Skalarprodukt:**
   Für den Beweis, dass das Skalarprodukt vor und nach der Abbildung durch eine orthogonale Matrix gleich bleibt, werden wir verwenden die Definition des Skalarproduktes von $\langle \mathbf{Ax}, \mathbf{Ay} \rangle$ und dann durch Umformungen zeigen, dass es gleich $\langle \mathbf{x}, \mathbf{y} \rangle$ ist:

$$\begin{aligned} \langle \mathbf{Ax}, \mathbf{Ay} \rangle &= (\mathbf{Ax})^\top (\mathbf{Ay}) \\ &= (\mathbf{x}^\top \mathbf{A}^\top)(\mathbf{Ay}) \\ &= \mathbf{x}^\top (\mathbf{A}^\top \mathbf{A}) \mathbf{y} \\ &= \mathbf{x}^\top \mathbf{I} \mathbf{y} \\ &= \mathbf{x}^\top \mathbf{y} \\ &= \langle \mathbf{x}, \mathbf{y} \rangle . \end{aligned}$$

   Somit gilt: $\langle \mathbf{Ax}, \mathbf{Ay} \rangle = \langle \mathbf{x}, \mathbf{y} \rangle$, d. h. die beliebige orthogonale Matrix $\mathbf{A}$ ändert das Skalarprodukt zwischen den beliebigen Vektoren $\mathbf{x}$, $\mathbf{y}$ nicht.

2. **Länge:**
   Um zu beweisen, dass die Längen nicht verändert werden, verwenden wir, dass die Länge eines Vektors quadriert gleich dem Skalarprodukt dieses Vektors mit sich selbst ist, $\|\mathbf{x}\|^2 = \langle \mathbf{x}, \mathbf{x} \rangle$:

$$\begin{aligned} \|\mathbf{Ax}\|^2 &= \langle \mathbf{Ax}, \mathbf{Ax} \rangle \\ &= \langle \mathbf{x}, \mathbf{x} \rangle \\ &= \|\mathbf{x}\|^2 . \end{aligned}$$

   Somit ist $\|\mathbf{Ax}\| = \|\mathbf{x}\|$.

3. **Winkel:**
   Nach der Definition des Winkels zwischen zwei Vektoren folgt:

$$\begin{aligned} \widehat{(\mathbf{Ax}, \mathbf{Ay})} &= \arccos \frac{\langle \mathbf{Ax}, \mathbf{Ay} \rangle}{\|\mathbf{Ax}\| \|\mathbf{Ay}\|} \\ &= \arccos \frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\|\mathbf{x}\| \|\mathbf{y}\|} \\ &= \widehat{(\mathbf{x}, \mathbf{y})} . \end{aligned}$$

   Somit gilt: $\widehat{(\mathbf{Ax}, \mathbf{Ay})} = \widehat{(\mathbf{x}, \mathbf{y})}$, d. h. der Winkel zwischen zwei beliebigen Vektoren ändert sich nicht unter der Wirkung einer orthogonalen Matrix.

□

Diese Eigenschaft orthogonaler Matrizen (Abbildungen) wird in der Technik sehr oft verwendet und ist von sehr grosser Bedeutung. Der nächste Satz lässt sich sehr einfach beweisen.

**Satz 1.7.0.7. Operationen mit orthogonalen Matrizen**

*Wenn* $\mathbf{A}$ *und* $\mathbf{B}$ *zwei orthogonale Matrizen mit passenden Dimensionen sind, so gilt folgendes:*

1. $\mathbf{A}$ *ist invertierbar und* $\mathbf{A}^{-1} = \mathbf{A}^{\mathsf{T}}$.
2. $\mathbf{A}^{-1}$ *ist orthogonal.*
3. $\mathbf{A}\,\mathbf{B}$ *ist orthogonal.*
4. $\mathbf{I}$ *ist orthogonal.*

Es gibt drei Arten von orthogonalen Matrizen:

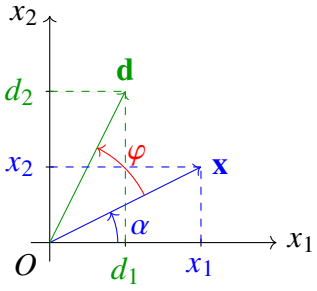
1. Permutationsmatrizen
2. Rotationsmatrizen
3. Spiegelungsmatrizen

Die Rotationsmatrizen und Spiegelungsmatrizen sind spannender als Permutationsmatrizen und von grosser Wichtigkeit; sie werden in den folgenden beiden Abschnitten besprochen. Aber zuerst kommt eine kleine Bemerkung zu den Permutationsmatrizen, die wir bereits bei der LU-Zerlegung gesehen haben.

**Beispiel 1.7.0.8 (Permutationsmatrizen sind orthogonal)**
Permutationsmatrizen sind orthogonal, da $\mathbf{P}\,\mathbf{P}^{\mathsf{T}} = \mathbf{I}$. ⌟

## Drehmatrix

**Beispiel 1.7.0.9 (Rotationsmatrix in 2D)**
Seien $\mathbf{x}, \mathbf{d} \in \mathbb{R}^2$ mit $\phi = \widehat{(\mathbf{x}, \mathbf{d})}$.
Wir wollen nun eine Matrix $\mathbf{D}(\phi)$ finden, welche $\mathbf{x}$ auf $\mathbf{d}$ abbildet, also die Drehung von $\mathbf{x}$ um den Winkel $\phi$ auf $\mathbf{d}$ realisiert.




Sei $r = \|\mathbf{x}\|$.

Gesucht ist die Matrix $\mathbf{D}(\phi)$, für welche gilt, dass $\mathbf{d} = \mathbf{D}(\phi)\mathbf{x}$. Um diese Matrix zu finden stellen wir die Komponenten $d_1$ und $d_2$ von $\mathbf{d}$ durch $r$, $\alpha$, $\phi$ dar und formen sie mittels trigonometrischen Identitäten um:

$$d_1 = r\cos(\alpha+\phi) = \underbrace{r\cos\alpha}_{x_1}\cos\phi - \underbrace{r\sin\alpha}_{x_2}\sin\phi$$
$$d_2 = r\sin(\alpha+\phi) = \underbrace{r\sin\alpha}_{x_2}\cos\phi + \underbrace{r\cos\alpha}_{x_1}\sin\phi\,.$$

Die Gleichung $\mathbf{d} = \mathbf{D}(\phi)\mathbf{x}$ ist also:

$$\mathbf{d} = \mathbf{D}(\phi)\mathbf{x} = \begin{bmatrix} x_1\cos\phi - x_2\sin\phi \\ x_1\sin\phi + x_2\cos\phi \end{bmatrix}.$$

Aus dieser neuen Gleichung können wir die Definition der Drehmatrix $\mathbf{D}(\phi)$ leicht ablesen:

$$\mathbf{D}(\phi) = \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix}.$$

Wenn wir den Vektor wieder zurückdrehen wollen, so müssen wir mit $-\phi$ arbeiten. Diese Drehung in die andere Richtung ist genau das Gleiche wie $\mathbf{D}(\phi)^\mathsf{T}$:

$$\mathbf{D}(\phi)^\mathsf{T} = \begin{bmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{bmatrix} = \mathbf{D}(-\phi)\,.$$

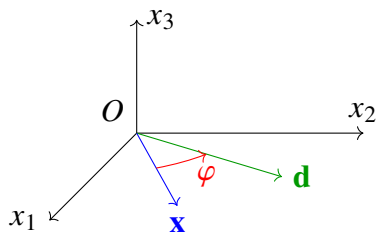
Wenn ein Vektor mit der Matrix $\mathbf{D}(\phi)$ multipliziert wird, rotiert sich der Vektor um den Winkel $\phi$ in eine Richtung. Die Matrix $\mathbf{D}(\phi)^\mathsf{T}$ rotiert diesen Vektor um den gleichen Winkel in die andere Richtung, so dass der Vektor wieder in seine ursprüngliche Richtung zeigt. Mathematisch heisst das: $\mathbf{D}(\phi)^\mathsf{T}\,\mathbf{D}(\phi) = \mathbf{I}$, d. h. die Matrix $\mathbf{D}(\phi)$ ist orthogonal. ⌟

Bald werden wir vor allem Drehungen gegen die trigonometrische Richtung verwenden, weshalb wir Folgendes definieren:

**Definition 1.7.0.10. Givens-Rotation**

Die Givens-Rotation ist die Matrix $\mathbf{G}(\phi)$ für welche gilt, dass $\mathbf{G}(\phi) = \mathbf{D}(-\phi)$.

**Beispiel 1.7.0.11 (Rotationsmatrix in 3D)**
In einem dreidimensionalen Raum wollen wir nun einen Vektor der zweidimensionalen Unterebene, welche durch die erste und zweite Koordinatenachse definiert ist, um den Winkel $\phi$ rotieren.




Für diese Rotation wollen wir nun die dazugehörige Drehmatrix $\mathbf{D}(\phi)$ finden. Der Vektor, der auf diese Weise gedreht wird, wird seine Komponente in der $x_3$-Richtung nicht ändern. Darum schreiben wir in der dritten Zeile und Spalte die entsprechenden Teile der Identitätsmatrix. An den restlichen Stellen brauchen wir die Einträge der Drehmatrix, die wir in zwei Dimensionen beschrieben haben. Die Matrix ist also:

$$\mathbf{D}(\phi) = \begin{bmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

⌟

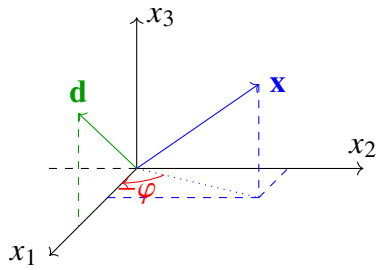
Was im Beispiel 1.7.0.11 für drei Dimensionen gemacht wurde, kann auf $n$ Dimensionen verallgemeinert werden. Die Matrix, die eine Rotation in der $i$-$j$-Ebene eines $n$ dimensionalen Raumes ausführt, gleicht überall der Identitätsmatrix, ausser in den $i$-ten, $j$-ten Spalten und Zeilen, in denen die Rotationsmatrix für

den zweidimensionalen Raum eingefügt wird:

$$\mathbf{D}_{ij}(\phi) = \begin{bmatrix} 1 & & & & & & & & & & & & 0 \\ & \ddots & & & & & & & & & & & \\ & & 1 & & & & & & & & & & \\ & & & \cos\phi & & & & & & -\sin\phi & & & \leftarrow i\text{-te Zeile} \\ & & & & 1 & & & & & & & & \\ & & & & & \ddots & & & & & & & \\ & & & & & & 1 & & & & & & \\ & & & \sin\phi & & & & & & \cos\phi & & & \leftarrow j\text{-te Zeile} \\ & & & & & & & & & & \ddots & & \\ & & & & & & & & & & & 1 & \\ 0 & & & & & & & & & & & & 1 \end{bmatrix} \begin{matrix} \\ \\ \\ \uparrow i\text{-te Spalte} \\ \\ \\ \\ \uparrow j\text{-te Spalte} \\ \\ \\ \\ \end{matrix}.$$

Diese Matrix realisiert die Rotation in der $x_i$–$x_j$-Ebene.

Die Drehung in der $i$–$j$-Ebene, welche die $j$ Komponente des gedrehten Vektors Null werden lässt, ist:




$$\mathbf{D}_{ij}(-\phi) \begin{bmatrix} x_1 \\ \vdots \\ x_i \\ \vdots \\ x_j \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} x_1 \\ \vdots \\ r \\ \vdots \\ 0 \\ \vdots \\ x_n \end{bmatrix}.$$

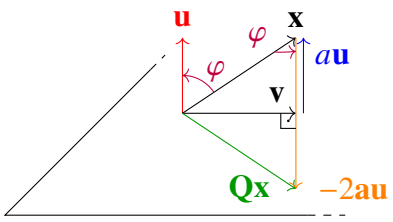
Hier ist $r = \sqrt{x_i^2 + x_j^2}$.
Solch eine Drehung wird von einer Givens Rotation mit Winkel $\phi$ realisiert, so dass:

$$\cos\phi = \frac{x_i}{r}$$
$$\sin\phi = \frac{x_j}{r}.$$

## Spiegelung mittels Householder-Matrizen

**Beispiel 1.7.0.12 (Spiegelung)**
Wir definieren hier die sogenannte Householder-Matrix **Q**, welche einen Vektor an einer Ebene spiegelt.

Gegeben sei eine Ebene mit dem Normalenvektor $\mathbf{u}$ der Länge 1: $\|\mathbf{u}\| = 1$.




Sei nun ein beliebiger, aber fester Vektor $\mathbf{x}$ gegeben, der nicht in dieser Ebene liegt. Der gespiegelte Vektor $\mathbf{Qx}$ entspricht dem Vektor $\mathbf{x} - 2a\mathbf{u}$, also müssen wir nur noch den Wert von $a$ finden. Eine Eigenschaft von $a\mathbf{u}$ ist, dass es die Projektion von $\mathbf{x}$ auf $\mathbf{u}$ ist, wobei $a$ hierbei die Länge dieser Projektion ist. Projektoren und ihre Anwendungen werden wir später noch besprechen.

Um den Skalar $a$ zu bestimmen, berechnen wir das Skalarprodukt zwischen $\mathbf{x}$ und $\mathbf{u}$ mithilfe einer Identität des euklischen Skalarprodukts ($\langle \mathbf{x}, \mathbf{y} \rangle = \|\mathbf{x}\|\|\mathbf{y}\| \cos\phi$). Wir sehen in der Skizze: $\cos\phi = \frac{a}{\|x\|}$, wir haben also:

$$\begin{aligned}
\mathbf{u}^\mathsf{T}\mathbf{x} &= \langle \mathbf{u}, \mathbf{x} \rangle \\
&= \|\mathbf{u}\|\|\mathbf{x}\| \cos\phi \\
&= 1 \cdot \|\mathbf{x}\| \frac{a}{\|\mathbf{x}\|} \\
&= a\,.
\end{aligned}$$

Somit ist der Skalar $a = \mathbf{u}^\mathsf{T}\mathbf{x}$. Mit ihm berechnen wir nun $\mathbf{Qx}$ und somit auch die Spiegelungsmatrix $\mathbf{Q}$:

$$\begin{aligned}
\mathbf{Qx} &= \mathbf{x} + (-2a\mathbf{u}) \\
&= \mathbf{x} - 2a\mathbf{u} \\
&= \mathbf{x} - 2\mathbf{u}a \\
&= \mathbf{x} - 2\mathbf{u}(\mathbf{u}^\mathsf{T}\mathbf{x}) \\
&= \mathbf{x} - 2(\mathbf{u}\mathbf{u}^\mathsf{T})\mathbf{x} \\
&= (\mathbf{I} - 2\mathbf{u}\mathbf{u}^\mathsf{T})\mathbf{x}\,.
\end{aligned}$$

⌟

**Definition 1.7.0.13. Householder-Matrix**

$\mathbf{Q} = \mathbf{I} - 2\mathbf{u}\mathbf{u}^\mathsf{T}$ ist die Householder-Matrix, welche einen Vektor an einer Ebene mit Normalenvektor $\mathbf{u}$ spiegelt. Der Normalenvektor $\mathbf{u}$ muss Norm 1 haben, falls dies nicht der Fall ist, muss anstelle von $\mathbf{u}$ mit $\frac{\mathbf{u}}{\|\mathbf{u}\|}$ gerechnet werden.

**Bemerkung 1.7.0.14 (Jede Householder-Matrix ist orthogonal)**
Die Householder-Matrix $\mathbf{Q}$ ist orthogonal. ⌟

*Beweis.* Die Matrix $\mathbf{Q}$ ist genau dann orthogonal, wenn $\mathbf{Q}^\mathsf{T}\mathbf{Q} = \mathbf{I}$. Das zeigen wir durch einfache mathematische Umformungen:

$$\begin{aligned}
\mathbf{Q}^\mathsf{T}\mathbf{Q} &= (\mathbf{I} - 2\mathbf{u}\mathbf{u}^\mathsf{T})^\mathsf{T}(\mathbf{I} - 2\mathbf{u}\mathbf{u}^\mathsf{T}) \\
&= (\mathbf{I}^\mathsf{T} - 2(\mathbf{u}\mathbf{u}^\mathsf{T})^\mathsf{T})(\mathbf{I} - 2\mathbf{u}\mathbf{u}^\mathsf{T}) \\
&= (\mathbf{I} - 2\mathbf{u}\mathbf{u}^\mathsf{T})(\mathbf{I} - 2\mathbf{u}\mathbf{u}^\mathsf{T}) \\
&= \mathbf{I} - 2\mathbf{u}\mathbf{u}^\mathsf{T} - 2\mathbf{u}\mathbf{u}^\mathsf{T} + 4(\mathbf{u}\mathbf{u}^\mathsf{T})(\mathbf{u}\mathbf{u}^\mathsf{T}) \\
&= \mathbf{I} - 4\mathbf{u}\mathbf{u}^\mathsf{T} + 4\mathbf{u}\underbrace{(\mathbf{u}^\mathsf{T}\mathbf{u})}_{\|\mathbf{u}\|^2=1}\mathbf{u}^\mathsf{T} \\
&= \mathbf{I} - 4\mathbf{u}\mathbf{u}^\mathsf{T} + 4\mathbf{u}\mathbf{u}^\mathsf{T} \\
&= \mathbf{I}\,.
\end{aligned}$$

□

**Beispiel 1.7.0.15 (Householder-Matrix)**
In diesem Beispiel soll die Householder-Matrix für ein gegebenes $\mathbf{u}$ berechnet werden.

Sei der Normalenvektor $\mathbf{u}$ einer Ebene gegeben als

$$\mathbf{u} = \begin{bmatrix} -\frac{2}{3} \\ \frac{1}{3} \\ -\frac{2}{3} \end{bmatrix}.$$

Als Erstes müssen wir die Norm des Normalenvektors berechnen, um zu überprüfen, ob sie $1$ ist:

$$\|\mathbf{u}\|^2 = \mathbf{u}^\mathsf{T}\mathbf{u} = \left(-\frac{2}{3}\right)^2 + \left(\frac{1}{3}\right)^2 + \left(-\frac{2}{3}\right)^2 = 1\,.$$

Der Vektor hat also Länge $1$ und er muss nicht geändert werden, falls $\mathbf{u}$ nicht Norm $1$ hätte, so müssten wir mit $\frac{\mathbf{u}}{\|\mathbf{u}\|}$ weiterarbeiten.

Die Householder-Matrix $\mathbf{Q}$ ist definiert als $\mathbf{I} - 2\mathbf{u}\mathbf{u}^\mathsf{T}$. Der erste Schritt besteht also darin, $\mathbf{u}\mathbf{u}^\mathsf{T}$ zu berechnen:

$$\mathbf{u}\mathbf{u}^\mathsf{T} = \begin{bmatrix} -\frac{2}{3} \\ \frac{1}{3} \\ -\frac{2}{3} \end{bmatrix} \begin{bmatrix} -\frac{2}{3} & \frac{1}{3} & -\frac{2}{3} \end{bmatrix} = \frac{1}{9}\begin{bmatrix} 4 & -2 & 4 \\ -2 & 1 & -2 \\ 4 & -2 & 4 \end{bmatrix},$$

und damit:

$$\begin{aligned} \mathbf{Q} &= \mathbf{I} - 2\mathbf{u}\mathbf{u}^\mathsf{T} \\ &= \frac{1}{9}\begin{bmatrix} 9 & 0 & 0 \\ 0 & 9 & 0 \\ 0 & 0 & 9 \end{bmatrix} - \frac{2}{9}\begin{bmatrix} 4 & -2 & 4 \\ -2 & 1 & -2 \\ 4 & -2 & 4 \end{bmatrix} \\ &= \frac{1}{9}\begin{bmatrix} 1 & 4 & -8 \\ 4 & 7 & 4 \\ -8 & 4 & 1 \end{bmatrix}. \end{aligned}$$

Die Householder-Matrix für die Spiegelung an der Ebene mit dem Normalenvektor $\mathbf{u}$ ist also:

$$\mathbf{Q} = \frac{1}{9}\begin{bmatrix} 1 & 4 & -8 \\ 4 & 7 & 4 \\ -8 & 4 & 1 \end{bmatrix}.$$

⌟

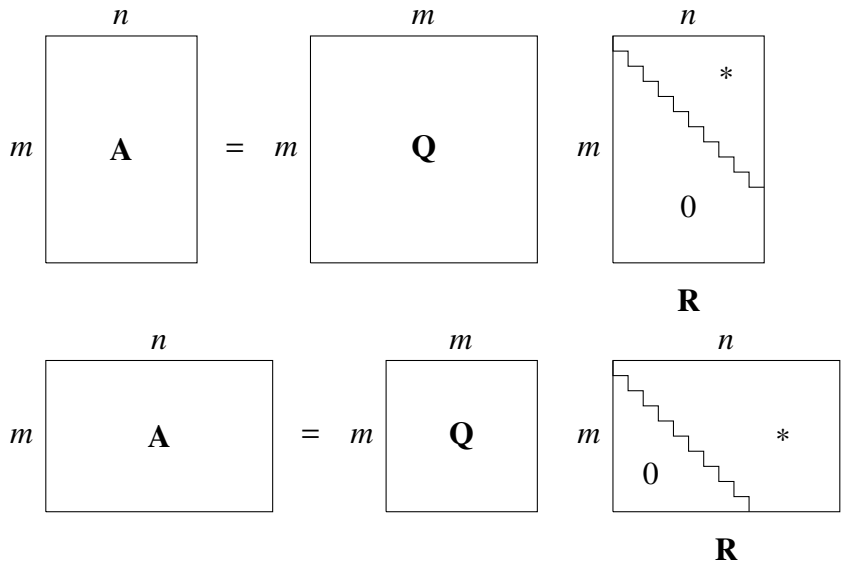
## 1.8 QR-Zerlegung via orthogonale Transformationen

Wir haben die Gauss-Elimination besprochen: Eine Matrix $\mathbf{A}$ wird durch elementare Transformationen in die obere Zeilenstufenform gebracht. Wir sahen auch, dass die Gauss-Elimination in der Sprache der Matrizen als LU-Zerlegung gesehen werden kann. Die Matrix $\mathbf{A}$ wird in die Matrizen $\mathbf{L}$ und $\mathbf{U}$ zerlegt, so dass $\mathbf{A} = \mathbf{L}\,\mathbf{U}$. Die LU-Zerlegung wurde durch die Multiplikation mit $\mathbf{E}_{ij}$-Matrizen realisiert. Diese $\mathbf{E}_{ij}$-Matrizen sind aber nicht orthogonal und deswegen können sich die Rundungsfehler für gewisse $\mathbf{A}$ akkumulieren. Im schlimmsten Fall kann dies zur Folge haben, dass die Gauss-Elimination falsche Resultate liefert.

Wenn wir aber $\mathbf{A}$ via orthogonale Transformationen in die obere Zeilenstufenform bringen, laufen wir nicht in die Gefahr, dass sich die Rundungsfehler sammeln, da orthogonale Transformationen Längen und Winkel nicht ändern. Ein solcher Algorithmus ist also deutlich stabiler und robuster gegen die Akkumulation von Rundungsfehlern.

**Definition 1.8.0.1. QR-Zerlegung**

Die QR-Zerlegung ist die Zerlegung einer Matrix $\mathbf{A}$ in eine orthogonale Matrix $\mathbf{Q}$ und eine obere Dreiecksmatrix $\mathbf{R}$, so dass $\mathbf{A} = \mathbf{Q}\,\mathbf{R}$.




Wir können die QR-Zerlegung wie die LU-Zerlegung verwenden, um lineare Gleichungssysteme stabil (d. h. ohne grosse Rundungsfehler) zu lösen:

$$\mathbf{A}\,\mathbf{x} = \mathbf{b} \iff \mathbf{Q}\,\mathbf{R}\,\mathbf{x} = \mathbf{b} \overset{\mathbf{Q}^T\cdot}{\iff} \mathbf{R}\mathbf{x} = \mathbf{Q}^T\mathbf{b}\,.$$

Wir können dann via Rücksubstitution die Unbekannte $\mathbf{x}$ auf einfache Art und Weise berechnen.

Der Vorteil dieser Methode gegenüber der LU-Zerlegung ist, dass sich die Rundungsfehler nicht akkumulieren. Sie hat aber auch einen Nachteil, denn sie kostet im schlimmsten Fall ungefähr dreimal so viel wie die LU-Zerlegung.

Um die QR-Zerlegung zu realisieren, verfolgen wir dasselbe Ziel wie bei der LU-Zerlegung: Nullen sollen unterhalb der Hauptdiagonale von $\mathbf{A}$ erzeugt werden, sodass diese zu einer oberen Dreiecksmatrix wird. Dies erreichen wir mit Matrixmultiplikationen; im Falle der LU-Zerlegung waren das die $\mathbf{E}_{ij}$-Matrizen, bei der QR-Zerlegung werden dies orthogonale Matrizen sein. Wenn wir nun mit diesen orthogonalen Matrizen arbeiten, werden sich die Rundungsfehler nicht akkumulieren, denn die Winkel zwischen den Spalten des Ergebnisses $\mathbf{R}$ sind dieselben wie die Winkel zwischen den Spalten von $\mathbf{A}$.

Wir sahen die drei Klassen von orthogonalen Matrizen: Permutationsmatrizen, Givens-Rotationen und Householder-Matrizen. Die Givens-Rotationen und Householder-Matrizen können beide verwendet werden, um die QR-Zerlegung auszuführen. Für beide Arten besprechen wir jetzt, wie die QR-Zerlegung durchgeführt werden kann.

## QR-Zerlegung mit Givens-Rotationen

Gegeben sei eine Matrix $\mathbf{A}$, für welche wir die QR-Zerlegung mittels Givens-Rotationen erhalten wollen. Wir sehen die Einträge der Matrix $\mathbf{A}$ als

$$\mathbf{A} = \begin{bmatrix} y_1 & * \cdots \cdots * \\ y_2 & * \cdots \cdots * \\ y_3 & * \cdots \cdots * \end{bmatrix}.$$

In einem ersten Schritt soll der Eintrag $y_2$ durch eine Drehung zu null werden. In einem zweiten Schritt wird dann das gleiche für $y_3$ gemacht.

Damit der Eintrag $y_2$ Null wird, und wir nur die ersten beiden Zeilen ändern, muss eine Drehung in der $1$–$2$-Ebene mit dem Winkel $\phi_{12}$ ausgeführt werden. Für diese Drehung verwenden wir die Givens-Rotation:

$$\mathbf{G}_{12}(\phi_{12}) = \begin{bmatrix} \cos\phi_{12} & \sin\phi_{12} & 0 \\ -\sin\phi_{12} & \cos\phi_{12} & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Wenn diese Matrix von links auf $\mathbf{A}$ multipliziert wird, so wird der Eintrag $y_2$ Null. Sie realisiert also den gleichen Schritt wie $\mathbf{E}_{12}$ in der Gauss-Elimination und verändert die unbeteiligten Zeilen nicht:

$$\mathbf{G}_{12}\,\mathbf{A} = \begin{bmatrix} *^{(1)} & *^{(1)} \cdots \cdots *^{(1)} \\ 0 & *^{(1)} \cdots \cdots *^{(1)} \\ y_3 & * \quad \cdots \cdots * \end{bmatrix}.$$

Die Einträge der ersten beiden Zeilen haben sich geändert, die in der Dritten aber nicht. Auch wurde der Eintrag $y_2$ wie gewünscht zu $0$.

Dasselbe machen wir nun für den Eintrag $y_3$. Eine neue Drehmatrix $\mathbf{G}_{13}(\phi_{13})$ in der $1$–$3$-Ebene, mit Winkel $\phi_{13}$, wird verwendet, um den Eintrag $y_3$ Null werden zu lassen:

$$\mathbf{G}_{13}(\phi_{13}) = \begin{bmatrix} \cos\phi_{13} & 0 & \sin\phi_{13} \\ 0 & 1 & 0 \\ -\sin\phi_{13} & 0 & \cos\phi_{13} \end{bmatrix}.$$

In diesem Fall ändert die Matrix $\mathbf{G}_{13}$ nur die Einträge der ersten und dritten Zeile, aber nicht Einträge in der zweiten.
Wenn wir nun die beiden Matrizen, $\mathbf{G}_{12}$ und $\mathbf{G}_{13}$, nacheinander mit $\mathbf{A}$ multiplizieren so werden beide Einträge $y_2,\ y_3$ zu Null abgebildet:

$$\mathbf{G}_{13}\,\mathbf{G}_{12}\,\mathbf{A} = \begin{bmatrix} *^{(2)} & *^{(2)} \cdots \cdots *^{(2)} \\ 0 & *^{(1)} \cdots \cdots *^{(1)} \\ 0 & *^{(2)} \cdots \cdots *^{(2)} \end{bmatrix}.$$

Beide Matrizen $\mathbf{G}_{13}$ und $\mathbf{G}_{12}$ sind orthogonal und wir können sie deswegen durch Multiplikation in eine einzige orthogonale Matrix $\mathbf{Q}_1$ zusammenfassen. Die Matrizen $\mathbf{G}_{ij}$ spielen dieselbe Rolle wie die Matrizen $\mathbf{E}_{ij}$ mit dem Unterschied, dass sie orthogonal sind.
Die Matrix $\mathbf{Q}_1$ erzeugt Nullen unterhalb des ersten Elements auf der Hauptdiagonale:

$$\mathbf{Q}_1\,\mathbf{A} = \begin{bmatrix} * & z_1 & * \cdots \cdots * \\ 0 & z_2 & * \cdots \cdots * \\ 0 & z_3 & * \cdots \cdots * \end{bmatrix}$$

Das Problem wurde nun reduziert auf eine kleinere Matrix, für die genau gleich fortgefahren wird. Der nächste Schritt würde darin bestehen, den Eintrag $z_3$ durch eine Drehung Null werden zu lassen. Für dies definieren wir eine neue Givens-Rotation $\mathbf{G}_{23}$ die dann wieder von links mit $\mathbf{Q}_1\,\mathbf{A}$ multipliziert wird. $\mathbf{G}_{23}$ wird hierbei die erste Zeile unverändert lassen.

Für eine grosse Matrix wird mit einer orthogonalen Matrix $\mathbf{Q}_1 = \mathbf{Q}_{1m}\,\mathbf{Q}_{1(m-1)}\ldots\mathbf{Q}_{13}\mathbf{Q}_{12}$ gearbeitet, welche das Produkt von den entsprechenden Drehungen in den $i-j$-Ebenen, mit $j = 2, 3, \ldots, m-1, m$ ist und Nullen in der ersten Spalte erzeugt:

$$\mathbf{Q}_1\,\mathbf{A} = \begin{bmatrix} * & * \cdots\cdots * \\ 0 & \\ \vdots & \quad \widetilde{\mathbf{A}} \\ 0 & \end{bmatrix}.$$

Nun würde man für die Matrix $\widetilde{\mathbf{A}}$ analog vorgehen, wie für die ursprüngliche Matrix $\mathbf{A}$. Man definiert Drehmatrizen, welche die gewünschten Einträge in der ersten Spalte zu Null werden lassen. Diese Drehmatrizen lassen die erste Zeile in der Matrix $\mathbf{A}$ unverändert. Anschliessend erhält man wieder eine kleinere Matrix, auf die das selbe Verfahren angewandt wird. Alle Drehungen, die in dieser neuen Matrix ausgeführt werden, ändern die Einträge in den vorherigen Spalten und Zeilen nicht mehr. Wenn dieses Prozedere für die ganze Matrix ausgeführt wurde, so erhält man eine Reihe von orthogonalen Matrizen $\mathbf{Q}_i$, die alle von links mit der Matrix $\mathbf{A}$ multipliziert werden, um die obere Dreiecksmatrix $\mathbf{R}$ zu erhalten:

$$\mathbf{Q}_{n-1}\ \ldots\ \mathbf{Q}_2\,\mathbf{Q}_1\,\mathbf{A} = \mathbf{R}\,.$$

Durch Multiplikation von links mit $(\mathbf{Q}_{n-1})^\mathsf{T},\ (\mathbf{Q}_{n-2})^\mathsf{T}\ldots\ (\mathbf{Q}_1)^\mathsf{T}$ können wir all diese Matrizen auf die rechte Seite bringen:

$$\mathbf{A} = (\mathbf{Q}_1)^\mathsf{T}\,(\mathbf{Q}_2)^\mathsf{T}\ldots(\mathbf{Q}_{n-1})^\mathsf{T}\,\mathbf{R}\,.$$

Die Matrizen $(\mathbf{Q}_1)^\mathsf{T}$ bis $(\mathbf{Q}_{n-1})^\mathsf{T}$ sind alle orthogonal und wir können sie demzufolge in eine einzelne orthogonale Matrix $\mathbf{Q}$ zusammenfassen:

$$\mathbf{A} = \underbrace{(\mathbf{Q}_1)^\mathsf{T}\,(\mathbf{Q}_2)^\mathsf{T}\ldots\ (\mathbf{Q}_{n-1})^\mathsf{T}}_{\text{orthogonal}}\,\mathbf{R} = \mathbf{Q}\,\mathbf{R}$$

$$\mathbf{A} = \mathbf{Q}\,\mathbf{R}\,,$$

und wir erhalten die gewünschte QR-Zerlegung von $\mathbf{A}$.

**Beispiel 1.8.0.2 (Nullen via Givens-Rotationen)**
Das folgende einfache Beispiel zeigt, wie in einem Vektor (oder auch einer Spalte einer Matrix) Einträge durch Drehungen zu null transformiert werden können.

Sei $\mathbf{a}$ gegeben:

$$\mathbf{a} = \begin{bmatrix} 4 \\ -3 \\ 1 \end{bmatrix} \rightsquigarrow \begin{bmatrix} * \\ 0 \\ 0 \end{bmatrix}.$$

Zuerst transformieren wir den zweiten Eintrag $-3$ zu null. Wir brauchen also die Rotationsmatrix $\mathbf{G}_{12}$ mit dem Winkel $\phi_{12}$. Im Kapitel 1.7 über Givens-Rotationen sahen wir folgende Eigenschaften für diesen Winkel. $\cos\phi_{12} = \frac{x_1}{r}$ und $\sin\phi_{12} = \frac{x_2}{r}$ wobei $r = \sqrt{x_1^2 + x_2^2}$.

Wir berechnen also die Werte für $r$, $\cos\phi_{12}$ und $\sin\phi_{12}$ und aus ihnen bilden wir die Matrix $\mathbf{G}_{12}$:

$$r = \sqrt{4^2 + (-3)^2} = 5$$
$$\cos\phi_{12} = \frac{4}{5}, \quad \sin\phi_{12} = -\frac{3}{5}$$

$$\mathbf{G}_{12} = \begin{bmatrix} \frac{4}{5} & -\frac{3}{5} & 0 \\ \frac{3}{5} & \frac{4}{5} & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Multiplizieren wir diese Matrix mit dem Vektor $\mathbf{a}$, so erhalten wir:

$$\mathbf{G}_{12}\,\mathbf{a} = \begin{bmatrix} \frac{4}{5} & -\frac{3}{5} & 0 \\ \frac{3}{5} & \frac{4}{5} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4 \\ -3 \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 0 \\ 1 \end{bmatrix}.$$

Nun muss noch der dritte Eintrag von $\mathbf{a}$ null werden. Wir verwenden eine neue Rotationsmatrix $\mathbf{G}_{13}$ in der 1−3-Ebene, mit dem Winkel $\phi_{13}$. Wie auch bei der vorherigen Drehmatrix, können wir die Werte $\cos\phi_{13}$, $\sin\phi_{13}$ und demzufolge die Matrix $\mathbf{G}_{13}$ einfach berechnen:

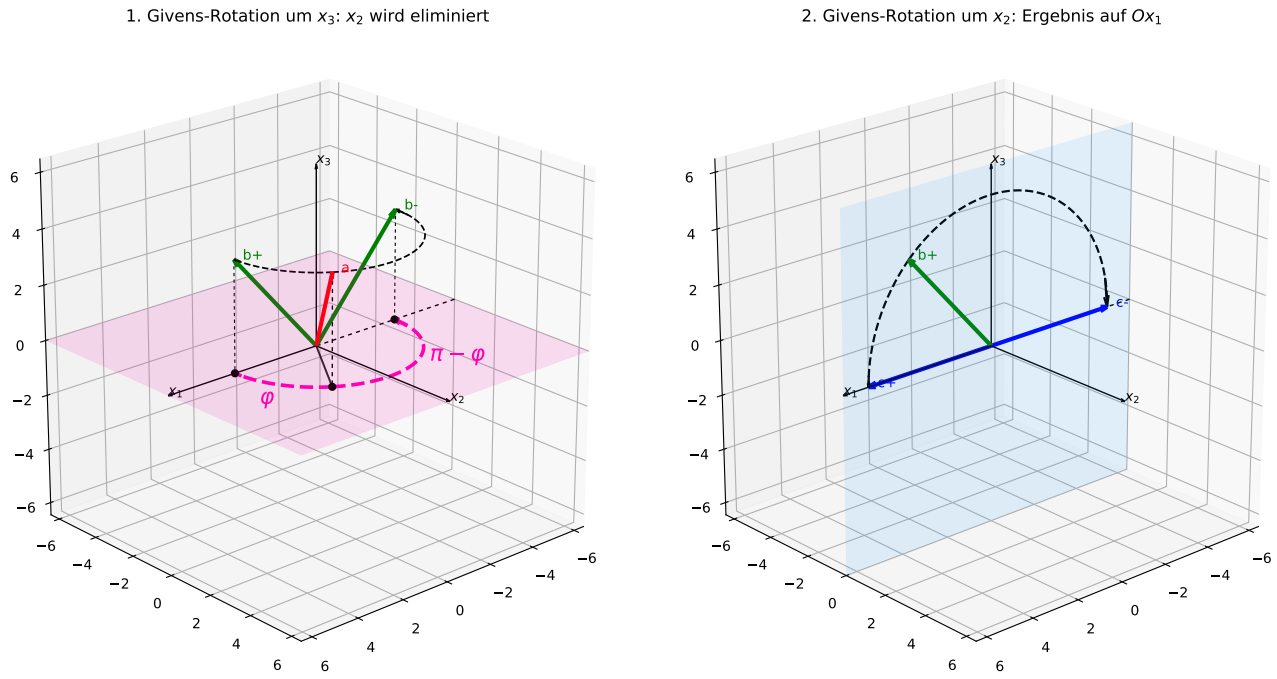
$$r = \sqrt{5^2 + 1^2} = \sqrt{26}$$
$$\cos\phi_{13} = \frac{5}{\sqrt{26}}, \quad \sin\phi_{13} = \frac{1}{\sqrt{26}}$$

$$\mathbf{G}_{13} = \begin{bmatrix} \frac{5}{\sqrt{26}} & 0 & \frac{1}{\sqrt{26}} \\ 0 & 1 & 0 \\ -\frac{1}{\sqrt{26}} & 0 & \frac{5}{\sqrt{26}} \end{bmatrix}.$$

Im letzten Schritt müssen wir diese Matrix mit $\mathbf{G}_{12}\mathbf{a}$ multiplizieren:

$$\mathbf{G}_{13}\,\mathbf{G}_{12}\,\mathbf{a} = \begin{bmatrix} \frac{5}{\sqrt{26}} & 0 & \frac{1}{\sqrt{26}} \\ 0 & 1 & 0 \\ -\frac{1}{\sqrt{26}} & 0 & \frac{5}{\sqrt{26}} \end{bmatrix} \begin{bmatrix} 5 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \sqrt{26} \\ 0 \\ 0 \end{bmatrix}.$$

Wir haben nun also den Vektor $\mathbf{a}$ durch rein orthogonale Transformationen in den Vektor $\begin{bmatrix} \sqrt{26} \\ 0 \\ 0 \end{bmatrix}$ umgewandelt.

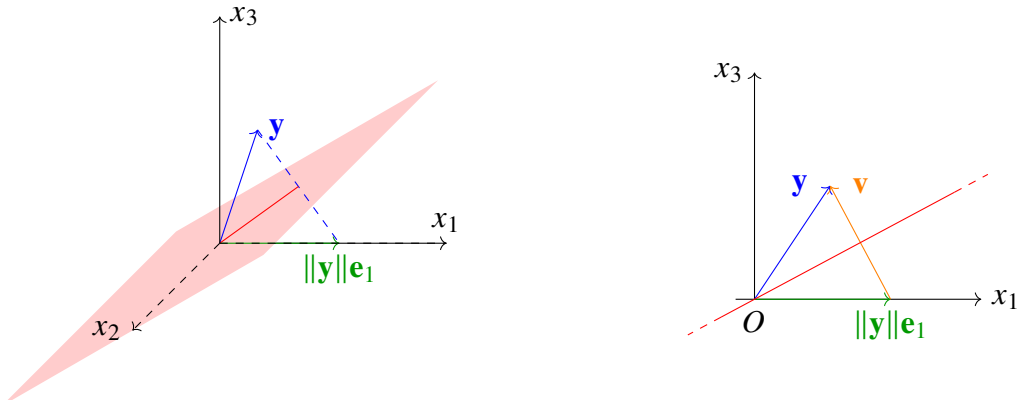



$\mathbf{a} = [2, 3, 4]^T$ wird zuerst in der $x_1Ox_2$-Ebene gedreht: $\mathbf{a}$ landet in der Ebene $x_2 = 0$.
Man entscheidet sich für das numerisch stabilere Zwischenergebnis (einer der zwei grünen Pfeile).
Eine zweite Givens-Drehung rotiert in der $x_1Ox_3$-Ebene zum Endergebnis auf der $Ox_1$-Achse.
Auch hier entscheidet man sich für das numerisch stabilere Ergebnis.

**Bemerkung 1.8.0.3 (QR-Zerlegung via Givens-Rotationen ist günstig für dünnbesetzte Matrizen)**
Diese Methode (mit Givens-Rotationen) ist günstig, falls $\mathbf{A}$ bereits viele Nulleinträge hat, da diese Methode in der Lage ist, gezielt an wenigen Stellen Nullen zu erzeugen. Matrizen, welche viele Nullen besitzen, werden dünnbesetzte Matrizen genannt und sind sehr wichtig in den Anwendungen.

## QR-Zerlegung mit Householder-Spiegelungen

Allgemeine Software arbeitet nicht mit Givens-Rotationen, sondern mit Householder-Spiegelungen, da bei dieser Methode alle Nullen einer Spalte in einem Schritt erzeugt werden und somit keine spezielle Form der Matrix berücksichtigt werden muss.




Die Matrix, welche diese Spiegelung durchführt, ist die Householder-Matrix $\mathbf{Q}$ mit $\mathbf{Q} = \mathbf{I} - 2\mathbf{u}\mathbf{u}^\top$, wobei $\mathbf{u}$ der Normalenvektor der Ebene ist und Länge Eins hat. Um dieses $\mathbf{u}$, und demzufolge die Matrix $\mathbf{Q}$, zu finden, nutzen wir die Tatsache aus, dass wir wissen, wie der gespiegelte Vektor $\mathbf{Q}\mathbf{y}$ aussieht. In diesem Beispiel sollen die $y_2$, $y_3$ Komponenten null werden; da sich die Länge von $\mathbf{y}$ nicht ändern wird, kann $\mathbf{Q}\mathbf{y}$ geschrieben werden als $\|\mathbf{y}\|\mathbf{e}_1$. Den Vektor $\mathbf{u}$ definieren wir einfach als die normierte Differenz zwischen

dem ursprünglichen und dem gespiegelten $\mathbf{y}$.

**Beispiel 1.8.0.4 (Nullen mit Householder-Spiegelungen)**
Sei der Vektor $\mathbf{y}$ gegeben und es soll die Householder-Matrix $\mathbf{Q}$ berechnet werden, welche die zweite und dritte Komponente Null werden lässt. Die Einträge von $\mathbf{y}$ sind:

$$\mathbf{y} = \begin{bmatrix} 2 \\ 2 \\ 1 \end{bmatrix}.$$

Die Householder-Matrix $\mathbf{Q}$ wird folgende Transformation an diesem Vektor ausführen:

$$\mathbf{y} = \begin{bmatrix} 2 \\ 2 \\ 1 \end{bmatrix} \xrightarrow{\mathbf{Q}} \|\mathbf{y}\|\mathbf{e}_1 = 3 \cdot \mathbf{e}_1 = \begin{bmatrix} 3 \\ 0 \\ 0 \end{bmatrix}.$$

Mit dieser Information können wir den Vektor $\mathbf{v}$, der $\mathbf{y}$ mit seinem Spiegelbild $\mathbf{Qy}$ verbindet, berechnen. Diesen Vektor $\mathbf{v}$ benutzen wir dann, um $\mathbf{u}$ zu bestimmen, denn $\mathbf{u}$ ist der normierte Vektor von $\mathbf{v}$:

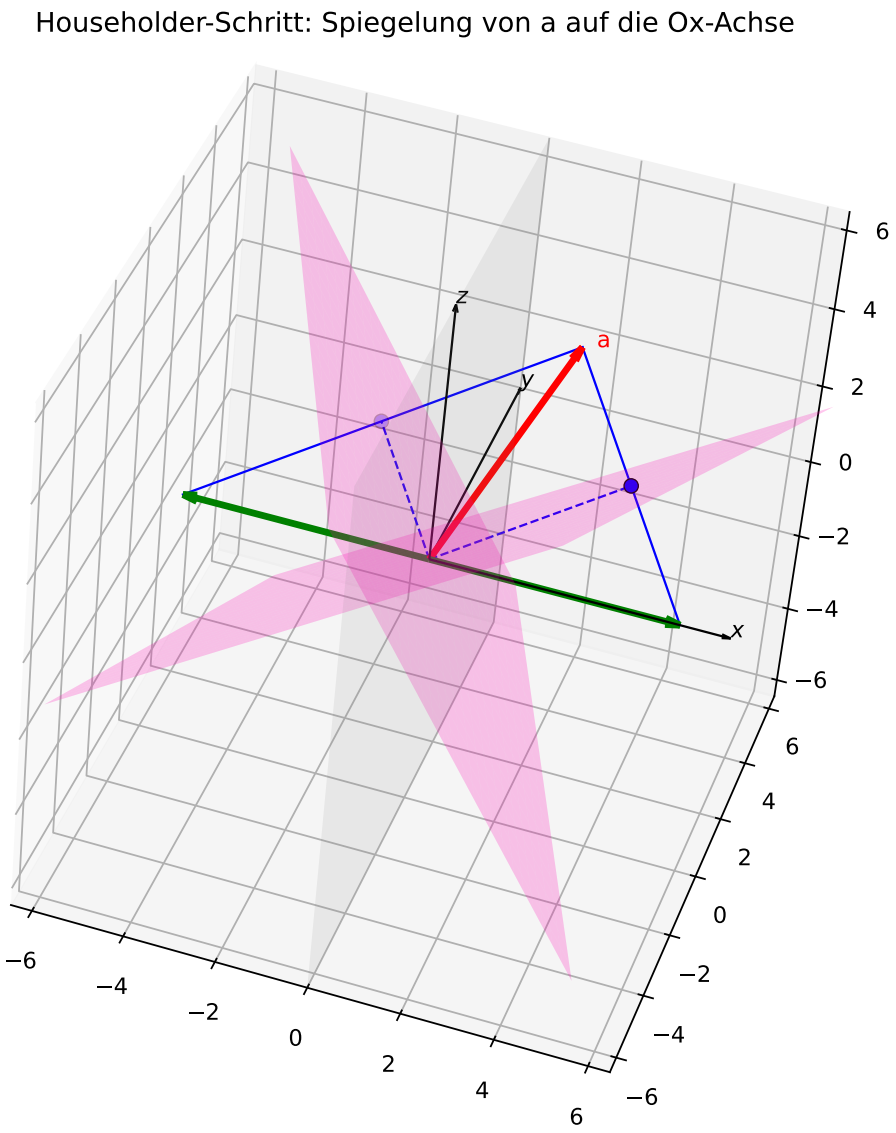
$$\begin{aligned} \mathbf{v} &= \mathbf{y} - \|\mathbf{y}\|\mathbf{e}_1 \\ &= \begin{bmatrix} 2 \\ 2 \\ 1 \end{bmatrix} - \begin{bmatrix} 3 \\ 0 \\ 0 \end{bmatrix} \\ \mathbf{v} &= \begin{bmatrix} -1 \\ 2 \\ 1 \end{bmatrix}. \end{aligned}$$

Aus diesem Vektor $\mathbf{v}$ können wir den Vektor $\mathbf{u}$ berechnen:

$$\mathbf{u} = \frac{\mathbf{v}}{\|\mathbf{v}\|} = \frac{1}{\sqrt{6}} \begin{bmatrix} -1 \\ 2 \\ 1 \end{bmatrix}.$$

Nachdem $\mathbf{u}$ nun bekannt ist, setzen wir ihn in die Definition der Householder-Matrix ein, um die gesuchte Matrix $\mathbf{Q}$ zu berechnen:

$$\begin{aligned} \mathbf{Q} &= \mathbf{I} - 2\mathbf{u}\mathbf{u}^\top \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \frac{2}{6} \begin{bmatrix} -1 \\ 2 \\ 1 \end{bmatrix} \begin{bmatrix} -1 & 2 & 1 \end{bmatrix} \\ &= \frac{1}{3} \begin{bmatrix} 2 & 2 & 1 \\ 2 & -1 & -2 \\ 1 & -2 & 2 \end{bmatrix}. \end{aligned}$$




Spiegelung von $\mathbf{a} = [2, 3, 4]^T$ um die Winkelhalbierenden (rosa Ebenen) von $\mathbf{a}$ und $\mathbf{e}_2$; es gibt zwei mögliche Ergebnisse auf $Ox_2$: Man verwendet die stabilere Version (bezüglich Rundungsfehler).

**Im Allgemeinen**

Was wir in diesem Beispiel für einen Vektor gemacht haben, lässt auf eine allgemeinere Matrix $\mathbf{A}$ übertragen. Sei diese Matrix $\mathbf{A}$ gegeben als

$$\mathbf{A} = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{bmatrix}.$$

Für diese Matrix definieren wir eine Householder-Matrix $\mathbf{Q}_1$, welche die gewünschten Nullen in der ersten Spalte erzeugt, d. h. den zweiten, dritten und vierten Eintrag der ersten Spalte auf Null setzt:

$$\mathbf{Q}_1 \mathbf{A} = \begin{bmatrix} * & * & * & * \\ 0 & * & * & * \\ 0 & * & * & * \\ 0 & * & * & * \end{bmatrix}.$$

Als Nächstes definieren wir eine andere Householder-Matrix $\mathbf{Q}_2$, welche auf die kleinere (grün markierte) Matrix wirkt, um in ihr die Nullen in der ersten Spalte zu realisieren. Diese neue Spiegelungsmatrix spiegelt nun nicht mehr in einem $\mathbb{R}^{m\times n}$, sondern in einem $\mathbb{R}^{m-1\times n-1}$, Raum und darf die Einträge der ersten Zeile und Spalte von $\mathbf{Q}_1\,\mathbf{A}$ nicht mehr ändern. Dieses $\mathbf{Q}_2$ muss also von spezieller Form sein. Die tatsächliche Spiegelung ist eine Matrix der Dimension $(m-1)\times(m-1)$, damit diese aber mit der Matrix $\mathbf{A}\in\mathbb{R}^{m\times n}$ multipliziert werden kann, muss sie mit Teilen der Identitätsmatrix aufgefüllt werden, bis die Dimensionen passen. Die Matrix $\mathbf{Q}_2$ wird also diese Form haben:

$$\mathbf{Q}_2 = \begin{bmatrix} 1 & 0\cdots\cdots 0 \\ 0 & \\ \vdots & \text{Spiegelungs Matrix} \\ \vdots & \\ 0 & \end{bmatrix}.$$

Wenn wir diese Matrix mit $\mathbf{Q}_1\,\mathbf{A}$ multiplizieren, erzeugen wir die Nullen in der zweiten Spalte:

$$\mathbf{Q}_2\,\mathbf{Q}_1\,\mathbf{A} = \begin{bmatrix} * & * & * & * \\ 0 & * & * & * \\ 0 & 0 & * & * \\ 0 & 0 & * & * \end{bmatrix}.$$

Dies führen wir nun so weiter, bis wir die obere Dreiecksmatrix $\mathbf{R}$ erhalten. Hierbei werden die Dimensionen der benötigten Spiegelungsmatrizen $\mathbf{Q}_i$ immer kleiner und müssen deswegen mit immer mehr Spalten und Zeilen der Identitätsmatrix aufgefüllt werden. Bei der Matrix $\mathbf{Q}_3$ zum Beispiel sind die ersten beiden Spalten und Zeilen identisch zu der Identitätsmatrix und die tatsächliche Spiegelung wird von einer $(m-2)\times(m-2)$ Matrix ausgeführt.

Wenn wir die obere Dreiecksmatrix erreichen, die QR-Zerlegung also abgeschlossen ist, werden wir eine Reihe an orthogonalen Householder-Matrizen $\mathbf{Q}_i$ haben, die mit der Matrix $\mathbf{A}$ multipliziert werden. Genau gleich wie bei der QR-Zerlegung mit Givens-Rotationen bringen wir diese Matrizen auf die andere Seite und fassen sie in einer einzelnen Matrix $\mathbf{Q}$ zusammen.

$$\mathbf{Q}_{n-1}\,\ldots\,\mathbf{Q}_2\,\mathbf{Q}_1\mathbf{A} = \mathbf{R} \implies \mathbf{A} = \mathbf{Q}\mathbf{R}.$$

**Beispiel 1.8.0.5 (QR-Zerlegung mittels Householder-Spiegelungen)** [12]
Sei die Matrix $\mathbf{A}$ gegeben als

$$\mathbf{A} = \begin{bmatrix} 1 & 1 \\ 2 & 0 \\ 2 & 0 \end{bmatrix}.$$

Der erste Schritt besteht darin, die Householder-Matrix $\mathbf{Q}_1$ zu finden. Um dies zu tun, betrachten wir das Produkt von $\mathbf{Q}_1$ und $\mathbf{A}$ spaltenweise:

$$\mathbf{Q}_1\,\mathbf{A} = \begin{bmatrix} \mathbf{Q}^1 \overset{\mathbf{a}_1}{\begin{bmatrix} 1 \\ 2 \\ 2 \end{bmatrix}} & \mathbf{Q}^1 \overset{\mathbf{a}_2}{\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}} \end{bmatrix}.$$

[12] Eine Methode um QR-Zerlegung mittels Householder-Spiegelungen mit Python durchzuführen wird in Abschnitt 9.2.5 anhand dieses Beispiels veranschaulicht.

Die erste Spalte in der Matrix $\mathbf{A}$ hat die euklidische Norm $\|\mathbf{a}\| = 3$. Das Resultat der ersten Multiplikation, $\mathbf{Q}_1\mathbf{a}_1$, ist das erste Zwischenziel:

$$\|\mathbf{a}_1\|\,\mathbf{e}_1 = \begin{bmatrix}3\\0\\0\end{bmatrix}.$$

Damit können wir nun $\mathbf{v}$ und anschliessend $\mathbf{u}$ berechnen:

$$\mathbf{v} = \mathbf{a}_1 - 3\mathbf{e}_1 = \begin{bmatrix}1\\2\\2\end{bmatrix} - \begin{bmatrix}3\\0\\0\end{bmatrix} = \begin{bmatrix}-2\\2\\2\end{bmatrix}$$

$$\|\mathbf{v}\| = \sqrt{12}\,.$$

Aus $\mathbf{v}$ und seiner Länge lässt sich der Vektor $\mathbf{u}$ bestimmen:

$$\mathbf{u} = \frac{\mathbf{v}}{\|\mathbf{v}\|} = \frac{1}{\sqrt{12}}\begin{bmatrix}-2\\2\\2\end{bmatrix}.$$

Da $\mathbf{u}$ nun bestimmt ist, berechnen wir die Householder-Matrix $\mathbf{Q}_1$:

$$\mathbf{Q}_1 = \mathbf{I} - \frac{2}{12}\begin{bmatrix}-2\\2\\2\end{bmatrix}\begin{bmatrix}-2 & 2 & 2\end{bmatrix} = \frac{1}{3}\begin{bmatrix}1 & 2 & 2\\2 & 1 & -2\\2 & -2 & 1\end{bmatrix}.$$

Mit dieser Matrix $\mathbf{Q}_1$ realisieren wir die Nullen in der ersten Spalte:

$$\mathbf{Q}_1\,\mathbf{A} = \begin{bmatrix}3 & \frac{1}{3}\\0 & \frac{2}{3}\\0 & \frac{2}{3}\end{bmatrix}.$$

In einem nächsten Schritt bestimmen wir $\mathbf{Q}_2$, welche auf die grün markierte Matrix wirkt, um den zweiten Eintrag Null werden zu lassen:

$$\mathbf{Q}_2 = \begin{bmatrix}1 & 0 & 0\\0 & \multicolumn{2}{|c}{\multirow{2}{*}{Spiegelungs-Matrix}}\\0 & \multicolumn{2}{|c}{}\end{bmatrix}.$$

Notieren wir den unteren Teil von $\mathbf{Q}_1\mathbf{a}_2$ mit $\mathbf{b}$:

$$\mathbf{b} = \frac{2}{3}\begin{bmatrix}1\\1\end{bmatrix}.$$

Dieser Vektor zeigt in die Richtung der Diagonale/Winkelhalbierenden des ersten Quadranten. Darum würde eine Drehung um den Winkel $-\frac{\pi}{4}$ reichen, um ihn in die Richtung von $\mathbf{e}_1$ zu bringen. Diese Drehung und ein Orientierungswechsel (Tauschen/Permutation der Koordinaten) ergibt auch die Spiegelung an der Winkelhalbierenden des ersten $\frac{\pi}{2}$-Winkels:

$$\begin{bmatrix}0 & 1\\1 & 0\end{bmatrix}\frac{1}{\sqrt{2}}\begin{bmatrix}1 & -1\\1 & 1\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}.$$

Als Übung und Beispiel für die Rechenführung, führen wir die Rechenschritte dennoch nach dem Algorithmus durch. Die euklidische Norm des zu drehenden Vektors ist $\|\boldsymbol{b}\| = \frac{2}{3}\sqrt{2}$. Damit berechnen wir $\mathbf{v}$:

$$\mathbf{v} = \mathbf{b} - \frac{2}{3}\sqrt{2}\mathbf{e}_1 = \frac{2}{3}\begin{bmatrix}1\\1\end{bmatrix} - \frac{2}{3}\sqrt{2}\begin{bmatrix}1\\0\end{bmatrix} = \frac{2}{3}\begin{bmatrix}1-\sqrt{2}\\1\end{bmatrix}$$

$$\|\mathbf{v}\| = \frac{2}{3}\sqrt{4-2\sqrt{2}}\,.$$

Damit wir weniger Wurzeln schreiben müssen, setzen wir die Ausdrücke direkt in die Formel für die Householder-Matrix ein:

$$\mathbf{I} - 2\mathbf{u}\mathbf{u}^\top = \mathbf{I} - \frac{2}{\|\mathbf{v}\|^2}\mathbf{v}\mathbf{v}^\top = \mathbf{I} - \frac{2}{4-2\sqrt{2}}\begin{bmatrix}1-\sqrt{2}\\ 1\end{bmatrix}\begin{bmatrix}1-\sqrt{2} & 1\end{bmatrix} = \mathbf{I} - \frac{1}{2-\sqrt{2}}\begin{bmatrix}3-2\sqrt{2} & 1-\sqrt{2}\\ 1-\sqrt{2} & 1\end{bmatrix},$$

das heisst:

$$\mathbf{I} - 2\mathbf{u}\mathbf{u}^\top = \frac{1}{2-\sqrt{2}}\begin{bmatrix}\sqrt{2}-1 & \sqrt{2}-1\\ \sqrt{2}-1 & 1-\sqrt{2}\end{bmatrix} = \frac{\sqrt{2}-1}{2-\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}.$$

Somit haben wir die Matrix berechnet, mit der wir nun $\mathbf{Q}_2$ und die Nullen in der zweiten Spalte realisieren können:

$$\mathbf{R} = \mathbf{Q}_2\,\mathbf{Q}_1\,\mathbf{A} = \begin{bmatrix}3 & \frac{1}{3}\\ 0 & \frac{2}{3}\sqrt{2}\\ 0 & 0\end{bmatrix},$$

wobei

$$\mathbf{Q}_2 = \begin{bmatrix}1 & 0 & 0\\ 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\\ 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}}\end{bmatrix}.$$

Die Matrix $\mathbf{Q}$ aus der QR-Zerlegung von $\mathbf{A}$ ist dann $\mathbf{Q} = \mathbf{Q}_1^\top\,\mathbf{Q}_2^\top$, damit $\mathbf{A} = \mathbf{Q}\,\mathbf{R}$. Wir sehen, wenn wir nur ein LGS lösen wollen, müssen wir die Matrizen $\mathbf{Q}, \mathbf{Q}_1, \mathbf{Q}_2$ gar nicht speichern, sondern lassen die Drehungen/Spiegelungen direkt auf die rechte Seite wirken, so dass wir einem oberen Dreieckssystem erhalten. Dies ist viel effizienter, als die Matrizen der Zwischenschritte zu speichern.

⌟

**Satz 1.8.0.6. QR-Zerlegung einer allgemeinen Matrix**

*Für jedes $\mathbf{A} \in \mathbb{R}^{m\times n}$ mit $m \geq n$ und $Rang(\mathbf{A}) = n$ gibt es eine eindeutige orthogonale Matrix $\mathbf{Q} \in \mathbb{R}^{m\times m}$ so dass*

$$\mathbf{A} = \mathbf{Q}\begin{bmatrix}\mathbf{R}\\ \mathbf{0}\end{bmatrix}$$

*mit $\mathbf{R}$ obere Dreiecksmatrix mit allen Diagonalelementen $\geq 0$.*

*Falls alle Diagonalelemente $> 0$ dann ist $\mathbf{R}$ die Cholesky-Zerlegung von $\mathbf{A}^\top\mathbf{A}$.*

$$\mathbf{A}^\top\mathbf{A} = \begin{bmatrix}\mathbf{R}^\top & \mathbf{0}\end{bmatrix}\underbrace{\mathbf{Q}^\top\mathbf{Q}}_{\mathbf{I}}\begin{bmatrix}\mathbf{R}\\ \mathbf{0}\end{bmatrix} = \mathbf{R}^\top\mathbf{R} + \mathbf{0}\,\mathbf{0} = \mathbf{R}^\top\mathbf{R}.$$

# Kapitel 2

# Lineare Räume / Vektorräume

## Inhalt


**2.1 Raum und Unterraum . . . . . 72**
**2.2 Erzeugendensystem . . . . . 80**
**2.3 Lineare Unabhängigkeit und Basis . . . . . 90**
**2.4 Fundamentalsatz der linearen Algebra - Teil I . . . . . 95**
**2.5 Koordinaten und Basiswahl . . . . . 101**
**2.6 Basiswechsel und Koordinatentransformation . . . . . 103**


## 2.1 Raum und Unterraum

Wenn man an einen Vektorraum denkt, dann kommt einem zuerst der Vektorraum $\mathbb{R}^n$ oder einer seiner Unterräume in den Sinn. Das ist aber nur ein gängiges Beispiel eines linearen Raumes. Das Konzept eines Vektorraumes ist dadurch entstanden, dass die Eigenschaften des uns inzwischen vertrauten euklidischen Raumes abstrahiert wurden.

Ein Vektorraum ist eine algebraische Struktur, für die gewisse Eigenschaften bezüglich der Addition und Multiplikation gelten müssen.

**Definition 2.1.0.1. Vektorraum**

Ein reeller *Vektorraum / linearer Raum* $V$ ist eine Menge mit zwei Operationen:

**Addition von Elementen aus** $V$ $(+)$:

$$a, b \in V : a + b \in V,$$

**Multiplikation mit Skalaren** $(\cdot)$:

$$\alpha \in \mathbb{R}, a \in V : \alpha \cdot a \in V.$$

Zusätzlich müssen folgende Eigenschaften gelten:
**Eigenschaften bezüglich der Addition von Elementen aus** $V$ $(+)$:

1. Kommutativitätsgesetz: $a + b = b + a$ für alle $a, b \in V$.
2. Assoziativitätsgesetz: $(a + b) + c = a + (b + c)$ für alle $a, b, c \in V$.
3. Es gibt $0 \in V$, sodass $a + 0 = a$ für alle $a \in V$ (dieses $0 \in V$ heisst Nullvektor).
4. Für jedes $a \in V$ gibt es ein $-a \in V$, so dass $a + (-a) = 0$.

**Eigenschaften bezüglich der Multiplikation mit Skalaren** $(\cdot)$:

1. Kompatibilität mit der Multiplikation mit Skalaren:
   $\alpha(\beta a) = (\alpha\beta)a$ für alle $\alpha, \beta \in \mathbb{R}, a \in V$.
2. Die Addition der Skalare ist distributiv gegen die Multiplikation mit Elementen aus $V$:
   $(\alpha + \beta)a = \alpha a + \beta a$ für alle $\alpha, \beta \in \mathbb{R}, a \in V$;
   Die Addition in $V$ ist distributiv gegen die Multiplikation mit Skalaren:
   $\alpha(a + b) = \alpha a + \alpha b$ für alle $\alpha \in \mathbb{R}, a, b \in V$.
3. Neutralelement für die Multiplikation mit Skalaren: $1 \cdot a = a$ für alle $a \in V$.

Ein komplexer linearer Raum ist analog definiert, wobei die Skalare komplexe Zahlen sind.

**Beispiel 2.1.0.2 (Die Räume $\mathbb{R}^n$ und $\mathbb{C}^n$)**

Zwei einfache Beispiele, die wir sofort geben können, sind die Räume $\mathbb{R}^n$ und $\mathbb{C}^n$:

$$V = \mathbb{R}^n = \{\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}, x_1, x_2, \ldots, x_n \in \mathbb{R}\} \text{ reeller linearer Raum.}$$

$$V = \mathbb{C}^n = \{\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}, x_1, x_2, \ldots, x_n \in \mathbb{C}\} \text{ komplexer linearer Raum.}$$

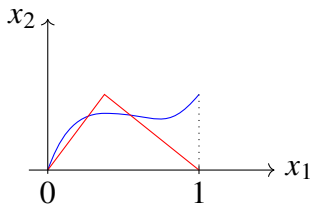
⌟

**Beispiel 2.1.0.3 (Raum der stetig differenzierbaren Funktionen)**
In diesem Beispiel werden wir einen, vielleicht etwas erstaunlichen, linearen Raum betrachten: Den Raum der $s$-mal stetig differenzierbaren Funktionen auf dem Intervall $[a, b]$. Dieser Raum wird in der Analysis oft verwendet und wir definieren ihn folgendermassen:

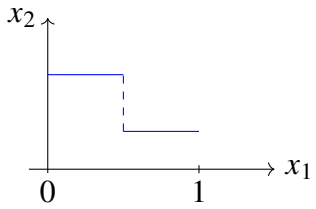
$$V = C^s[a, b] = \{f : [a, b] \to \mathbb{R},\ f \text{ ist } s\text{-mal stetig differenzierbar}\}.$$

Zur Erinnerung: Die Funktion $f$ heisst $s$-mal stetig differenzierbar, wenn die Ableitungen $f^{(k)}$ für $k = 0, 1, \ldots, s$ existieren und $f^{(s)}$ stetig ist.

Zum Beispiel ist $C^0[0, 1]$ die Menge aller stetiger Funktionen auf dem Intervall $[0, 1]$. Die Funktionen, deren Graphen unten geplottet sind, liegen beide in dieser Menge, da sie beide stetig im Intervall $[0, 1]$ sind:




Die folgende Funktion liegt allerdings nicht in $C^0[0,1]$, da sie eine Sprungstelle an der Stelle $\frac{1}{2}$ hat:




Ein anderes Beispiel in dieser Kategorie ist $C^\infty[a,b]$, d. h. die Menge aller Funktionen die beliebig oft stetig differenzierbar sind. Zu dieser Menge gehören zum Beispiel die trigonometrischen Funktionen sin und cos, sowie die Polynome. Die Definition des Raumes ist:

$$C^\infty[a,b] = \{f : [a,b] \to \mathbb{R},\ f \text{ ist } \infty\text{-mal stetig differenzierbar}\}\,.$$

Diese Mengen sind ineinander verschachtelt:

$$C^\infty \subset \ldots \subset C^{s+1} \subset C^s \subset \ldots \subset C^1 \subset C^0\,.$$

Mit dieser $C$-Notation können wir also verschiedene Mengen, deren Elemente Funktionen sind, definieren. Diese Funktionen haben auf den ersten Blick nichts mit Vektoren zu tun, wir werden aber gleich sehen, dass wir diese Funktionen analog zu Vektoren behandeln können.

Damit wir diesen Zusammenhang sehen können, werden wir die Addition und die Multiplikation mit Skalaren von Elementen aus einem beliebigen, aber festen $V = C^s$ definieren. Beginnen wir also mit einer Definition für die Addition von Elementen.

**''+'' von Funktionen:**

Seien $f,\ g \in V$, also $f,\ g : [a,b] \to \mathbb{R}$ für welche die Ableitungen $f^{(k)},\ g^{(k)}$ existieren für $k = 0,\ 1, \ldots,\ s$ und $f^{(s)},\ g^{(s)}$ stetig sind. So definieren wir die Addition zwischen diesen beiden Funktionen als:

$$f + g : [a,b] \to \mathbb{R},\ (f+g)(t) = f(t) + g(t) \quad \text{für alle } t \in [a,b]\,.$$

**''·'' mit Skalaren:**

Sei $\alpha \in \mathbb{R}$ und $f \in V$. So definieren wir die Multiplikation des Skalars $\alpha$ mit der Funktion $f$ als:

$$\alpha \cdot f : [a,b] \to \mathbb{R},\ (\alpha f)(t) = \alpha f(t) \quad \text{für alle } t \in [a,b]\,.$$

Es ist offensichtlich, dass diese Addition und Multiplikation die Eigenschaften für den linearen Raum erfüllen und deswegen $V$ ein linearer Raum ist. Deswegen sind auch die Mengen der $s$-mal stetig differenzierbaren Funktionen $C^s$ lineare Räume.

Wir sehen also: Wenn alle Funktionen einer gegebenen Menge denselben Definitionsbereich und Wertebereich haben, dann ergibt es Sinn, eine solche punktweise Addition und Skalarmultiplikation der Elemente der Menge zu definieren. Wenn das Ergebnis in derselben Menge bleibt, ist es nicht schwer zu sehen, ob die nötigen Eigenschaften für einen linearen Raum erfüllt werden. ⌟

**Beispiel 2.1.0.4 (Räume von Polynomen)**
Die Menge der Polynome werden wir in dieser Vorlesung oft verwenden:

$$\mathcal{P}_n = \{p \text{ Polynome vom Grad } \leq n-1\},$$

wobei wir ein Polynom $p$ in der Variable $t$ meist als eine spezielle Funktion in $t$ betrachten:

$$p \in \mathcal{P}_n : \; p(t) = a_0 + a_1 t + a_2 t^2 + \ldots + a_{n-1} t^{n-1} \quad \text{mit } a_0,\, a_1, \ldots,\, a_{n-1} \in \mathbb{R}\,.$$

Etwas was einem unnatürlich erscheinen mag ist, dass wir in der Menge $\mathcal{P}_n$ nur Polynome vom Grad kleiner gleich $n-1$ und nicht $n$ haben. Der Grund dahinter ist, dass wir insgesamt $n$ Koeffizienten und auch $n$ Potenzen $(t^0, t^1, t^2, \ldots, t^{n-1})$ von $t$ haben. In dieser Vorlesung wollen wir diese Notation behalten, damit der Index $n$ in $\mathcal{P}_n$ uns an diese $n$ Koeffizienten erinnert. Das ist als Warnung zu verstehen, in anderen Vorlesungen oder Texten, kann $\mathcal{P}_n$ Polynome bis Grad $n$ enthalten: dort sind eben $n+1$ Koeffizienten und Potenzen von $t$.

Natürlich ist $\mathcal{P}_n$ auch ein linearer Raum bezüglich der Addition und der Multiplikation mit einem Skalar. Die einfacheren Polynome:

$$p_k \in \mathcal{P}_n \subset \mathcal{P} \text{ mit } p_k(t) = t^k, k \leq n-1$$

heissen *Monome* und spielen eine ähnliche Rolle, wie die Vektoren $\mathbf{e}_k$ in $\mathbb{R}^n$.

Der Satz von Taylor aus Analysis zeigt, wie man jede Funktion aus $C^n$ mittels eines Polynoms aus $\mathcal{P}_k$ approximiert und gibt sogar den Fehler an. Am Ende der Analysis I Vorlesung sieht man, dass sich sogar jede Funktion aus $C^0$ auch mit Polynomen approximieren lässt. Glattere Funktionen lassen sich besser mit Polynomen approximieren. Man arbeitet oft lieber in einem Raum von Polynomen $\mathcal{P}_k$ statt in den etwas unbequemeren Räumen $C^n$. ⌟

**Beispiel 2.1.0.5 (Raum der quadratisch integrierbaren Funktionen $L^2$)**
Der Raum der quadratisch integrierbaren Funktionen $L^2$ ist sehr wichtig in den Naturwissenschaften:

$$L^2[0,1] = \{f : [0,1] \to \mathbb{R}, \int_0^1 |f(t)|^2 dt \quad \text{existiert}\}\,.$$

Der Sinn des Integrals und allgemein die Eigenschaften dieses Raumes werden Sie erst später im Studium sehen. Für den Moment kennen Sie nur das Riemannsche Integral und es ist klar, dass die Elemente von $L^2$ viel weniger Glattheit als die Elemente von $C^0$ haben müssen. Die Rolle von $\mathbf{e}_k \in \mathbb{R}^n$ mit $k \leq n$ spielen hier $\cos(2n\pi\, t)$ und $\sin(2n\pi\, t)$ (Fourier lässt Grüssen). Natürlich müssen wir hier nicht auf dem Intervall $[0,1]$ bleiben, sondern auch $\mathbb{R}$, $\mathbb{R}^n$, $[a,b] \subset \mathbb{R}$, $[\mathbf{a},\mathbf{b}] \subset \mathbb{R}^n$ sind möglich. Wenn die Funktionen Werte in $\mathbb{C}$ annehmen, dann spielt die Menge der trigonometrischen Polynome $\exp(2\pi i n\, t)$ eine wesentliche Rolle. Die Namen *ebene Wellen*, *Schwingungen*, *Fourier Basis* kommen in verschiedenen Gebieten der Naturwissenschaften für diese speziellen Elemente von $L^2$ vor. Die trigonometrischen Polynome liegen auch in $C^\infty$.

In Analysis wird bewiesen, dass Funktionen aus $L^2$ sich mit (Summen von) trigonometrischen Polynomen approximieren lassen; je glatter die Funktion ist, desto weniger Terme (in der Summe) ergeben eine gleich

genaue Approximation. Einer der wichtigsten Algorithmen des zwanzigsten Jahrhunderts (FFT) hilft, um solche approximierende trigonometrische Polynome sehr effizient zu berechnen. Darum wird in der Praxis oft in linearen Räumen, die aus trigonometrischen Polynomen bestehen, gearbeitet. $L^2$ ist der richtige lineare Raum, um auch mit sehr unglatten Funktionen zu arbeiten: Wavelets und Finite Elemente sind zwei bedeutende Technologien, die zunächst in $L^2$ Platz haben.

Beispielweise, ist die in Beispiel 2.1.0.3 vorgestellte, unstetige Funktion ein Element des $L^2$-Raumes jedoch keines $C^s$-Raumes. ⌟

**Beispiel 2.1.0.6. Menge der konvergenten Folgen**
Die Menge der konvergenten Folgen in $\mathbb{R}$

$$\ell = \{(x_n) \text{ konvergent}\}$$

ist ein linearer Raum. ⌟

**Beispiel 2.1.0.7. Lösungen einer linearen DGL**
Die Menge der Lösungen des homogenen linearen Differentialgleichungssystem

$$\mathbf{y}' = \mathbf{A}\mathbf{y}$$

ist ein linearer Raum. ⌟

Diese Beispiele sollen einem die Wichtigkeit und den Grund hinter der Abstraktion in linearen Räumen aufzeigen. Denn wenn wir eine Eigenschaft, ein Theorem, für die linearen Räume beweisen, dann gilt dieses Theorem auch für all die obigen Beispiele, denn es handelt sich bei allen um lineare Räume.
Das Ganze funktioniert aber auch in die andere Richtung. Die Sätze, die wir für den Raum $\mathbb{R}^n$ oder $\mathbb{C}^n$ bewiesen haben, die nur auf der internen Addition und der Skalarmultiplikation basieren, lassen sich auch auf lineare Räume und demzufolge für all die obigen Beispiele erweitern.

**Bemerkung 2.1.0.8 (Rechenregeln in linearen Räumen)**
Es folgen nun einige weitere Rechenregeln die in linearen Räumen gültig sind.
Sei $V$ ein reeller, linearer Raum, $a,\ b \in V$ und $\alpha \in \mathbb{R}$, so gilt:

(1) Aus $a + b = a$ folgt, dass $b = 0$.

(2) $\alpha \cdot 0 = 0 \in V$.

(3) $0 \cdot a = 0 \in V$.

⌟

Bei den Räumen $C^s$ haben wir gesehen, dass diese ineinander verschachtelt sind, sie bilden also Teilmengen voneinander. Beim Raum $\mathbb{R}^3$ haben wir gesehen, dass eine Ebene eine Untermenge ist. Es ist daher sinnvoll eine allgemeine Definition für den sogenannten Unterraum zu geben.

**Definition 2.1.0.9. Unterraum**

Sei $V$ ein linearer Raum mit $U \subseteq V$ und $U \neq \emptyset$. Dann heisst $U$ *Unterraum* von $V$, falls gilt:
1. Wenn $x,\ y \in U$ dann auch $x + y \in U$.
2. Wenn $\alpha \in \mathbb{R},\ x \in U$ dann auch $\alpha x \in U$.

**Bemerkung 2.1.0.10.**
Wenn wir $\alpha = 0$ wählen und $x \in U$ betrachten, wobei $U$ ein Unterraum von $V$ ist, so muss laut der obigen Definition und der Eigenschaft (2) gelten, dass

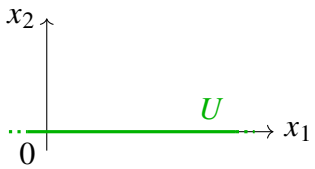
$$0 \cdot x = 0 \in U\,.$$

Ein Unterraum eines linearen Raumes $V$ muss also immer das Element $0$ enthalten.

⌟

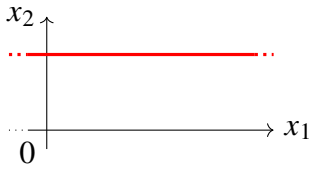
**Beispiel 2.1.0.11 (Beispiele und Gegenbeispiele von Unterräumen)**
In diesem Beispiel betrachten wir Untermengen von $\mathbb{R}^2$.

Sei $V = \mathbb{R}^2$. Hierfür definieren wir einen Unterraum $U = \{\begin{bmatrix} x_1 \\ 0 \end{bmatrix} \in \mathbb{R}^2 \mid x_1 \in \mathbb{R}\} \subset V$. Wir haben also folgende Situation:




Wir können leicht überprüfen, dass dies ein Unterraum von $\mathbb{R}^2$ ist und deswegen können wir auch sagen, dass $\mathbb{R}^1$ ein Unterraum von $\mathbb{R}^2$ ist.

Wir können aber auch eine andere Menge $\tilde{U} = \{\begin{bmatrix} x_1 \\ 5 \end{bmatrix} \in \mathbb{R}^2 \mid x_1 \in \mathbb{R}\} \subset V$ angeben. Wenn wir dies darstellen, erhalten wir:




Wir sehen, dass dies kein Unterraum von $\mathbb{R}^2$ ist, da $0$ nicht Teil von $\tilde{U}$ ist.

Mit der gleichen Methode wie in $U$, wo wir immer eine Komponente auf $0$ setzen, um einen Unterraum zu bilden, können wir auch zeigen, dass

$$\mathbb{R}^1 \subset \mathbb{R}^2 \subset \ldots \subset \mathbb{R}^n .$$

⌟

**Beispiel 2.1.0.12 (Verschachtelung einiger Räume von Funktionen)**
Auf ähnliche Art und Weise können wir Aussagen über die Verschachtelung der anderen linearen Räume, die wir gesehen haben, machen:

$$\mathcal{P}_1 \subset \mathcal{P}_2 \subset \ldots \subset \mathcal{P}_n \subset \mathcal{P}_{n+1} \subset \ldots \subset \mathcal{P} \subset C^\infty \subset \ldots \subset C^{s-1} \subset C^s \subset \ldots \subset C^1 \subset C^0 \subset L^2 .$$

Hier haben wir stillschweigend die Polynome als Funktionen auf der gewünschten Definitionsmenge $[a, b]$ verstanden. ⌟

**Beispiel 2.1.0.13 (Ein Unterraum in $\mathbb{R}^4$)**
In diesem Beispiel wollen wir beweisen, dass $U$ definiert als:

$$U = \{\mathbf{x} \in \mathbb{R}^4;\ \mathbf{x} = \begin{bmatrix} x_3 - x_4 \\ x_3 + x_4 \\ x_3 \\ x_4 \end{bmatrix} x_3,\ x_4 \in \mathbb{R}\}$$

einen Unterraum von $\mathbb{R}^4$, mit der Addition aus $\mathbb{R}^4$ und der Multiplikation mit Skalaren aus $\mathbb{R}$, bildet.

Wir beweisen zuerst: Wenn wir ein Element aus $U$ mit einem Skalar $\alpha$ multiplizieren, erhalten wir ein Element aus $U$. In einem zweiten Schritt werden wir zeigen, dass auch die Summe von zwei Elementen aus $U$ ebenfalls ein Element von $U$ ist.

Sei $\alpha \in \mathbb{R}$ beliebig und $\mathbf{x} \in U$ beliebig, wir berechnen nun die Multiplikation mit einem Skalar:

$$\alpha\mathbf{x} = \alpha \begin{bmatrix} x_3 - x_4 \\ x_3 + x_4 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} \alpha x_3 - \alpha x_4 \\ \alpha x_3 + \alpha x_4 \\ \alpha x_3 \\ \alpha x_4 \end{bmatrix}.$$

Wir substituieren nun $\alpha x_3$ mit $t_3 \in \mathbb{R}$ und $\alpha x_4$ mit $t_4 \in \mathbb{R}$ und erhalten:

$$\begin{bmatrix} t_3 - t_4 \\ t_3 + t_4 \\ t_3 \\ t_4 \end{bmatrix},$$

dieser Vektor ist in $U$, da sowohl $t_3$ als auch $t_4$ in $\mathbb{R}$ sind. Also haben wir $\alpha\mathbf{x} \in U$.

Nun müssen wir noch die Eigenschaft der Addition beweisen.
Seien $\mathbf{x}, \mathbf{y} \in U$ beide beliebig. Um zu beweisen, dass die Summe dieser beider Elemente immer noch in $U$ ist, berechnen wir diese:

$$\mathbf{x} + \mathbf{y} = \begin{bmatrix} x_3 - x_4 \\ x_3 + x_4 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} y_3 - y_4 \\ y_3 + y_4 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} (x_3 + y_3) - (x_4 + y_4) \\ (x_3 + y_3) + (x_4 + y_4) \\ x_3 + y_3 \\ x_4 + y_4 \end{bmatrix}.$$

Auch hier machen wir wieder eine Substitution und ersetzen $x_3 + y_3$ mit $t_3 \in \mathbb{R}$ und $x_4 + y_4$ mit $t_4 \in \mathbb{R}$. Mit dieser Substitution erhalten wir:

$$\begin{bmatrix} t_3 - t_4 \\ t_3 + t_4 \\ t_3 \\ t_4 \end{bmatrix}.$$

Also gilt nun auch wieder, dass dies in $U$ ist und wir haben, dass $\mathbf{x} + \mathbf{y} \in U$.

Da $U$ eine Teilmenge von $\mathbb{R}^4$ ist und beide definierenden Eigenschaften eines Unterraums erfüllt, ist es ein Unterraum von $\mathbb{R}^4$.

⌟

In den folgenden beiden Beispielen sehen wir zwei Teilmengen von $\mathbb{R}^2$, die keinen Unterraum bilden.

**Beispiel 2.1.0.14 (Gegenbeispiel von Unterraum)**
Sei $X$ eine Teilmenge von $\mathbb{R}^2$ und folgendermassen definiert:

$$X = \{\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \in \mathbb{R}^2,\ x_1 \geq 0,\ x_2 \geq 0\}.$$

Diese Menge entspricht dem ersten Quadranten eines 2-dimensionalen Koordinatensystems.

Um zu zeigen, dass dies kein Unterraum von $\mathbb{R}^2$ ist, suchen wir einfach ein Gegenbeispiel für eine der Eigenschaften. Z. B.:

$$\mathbf{x} = \begin{bmatrix} 1 \\ 1 \end{bmatrix},\ \alpha = -1 \implies \alpha\mathbf{x} = \begin{bmatrix} -1 \\ -1 \end{bmatrix} \notin X.$$

Dies zeigt uns also, dass $X$ kein Unterraum von $\mathbb{R}^2$ ist.

⌟

**Beispiel 2.1.0.15 (Gegenbeispiel von Unterraum)**
Wir versuchen nun also das Problem der Menge im vorherigen Beispiel zu umgehen, indem wir $X$ neu definieren als:

$$X = \{\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \in \mathbb{R}^2,\ x_1 x_2 \geq 0\}\,.$$

Aber auch hier sind wir wieder in der Lage ein Gegenbeispiel zu finden:

$$\mathbf{x} = \begin{bmatrix} -4 \\ -2 \end{bmatrix},\ \mathbf{y} = \begin{bmatrix} 3 \\ 3 \end{bmatrix} \implies \mathbf{x} + \mathbf{y} = \begin{bmatrix} -1 \\ 1 \end{bmatrix} \notin X\,.$$

Also ist auch dieses $X$ kein Unterraum von $\mathbb{R}^2$.

⌟

Um zu beweisen, dass eine Menge kein Unterraum ist, muss ein einziges Gegenbeispiel gefunden werden. Um aber zu beweisen, dass eine Menge ein Unterraum ist, müssen wir die beiden Eigenschaften des Unterraums für beliebige Elemente in der Menge beweisen.

**Bemerkung 2.1.0.16 (Charakterisierungen von Unterräumen)**
Folgende drei Aussagen sind äquivalent:

1. $U$ ist ein Unterraum eines linearen Raumes $V$.
2. Für beliebige $u,\ v \in U$ und $\alpha$ Skalar gilt, dass:

$$\alpha u + v \in U\,.$$

3. Jede beliebige Linearkombination von Elementen aus $U$ ist in $U$.

⌟

## 2.2 Erzeugendensystem

Bei der Definition 1.1 im ersten Kapitel haben wir bereits den Begriff der linearen Kombinationen für Vektoren in $\mathbb{R}^n$ verwendet. Mit einem leichten Missbrauch der Sprache haben wir in der Bemerkung 1.1.0.9 sogar über Linearkombinationen von Gleichungen geredet. Mit den Operationen aus der Definition eines linearen Raumes kann man lineare Kombinationen von Elementen aus dem linearen Raum definieren.

**Definition 2.2.0.1. Linearkombination**

Eine *Linearkombination* der Elemente $v_1, v_2, \ldots, v_n$ eines linearen Raumes $V$ ist

$$x_1 v_1 + x_2 v_2 + \ldots + x_n v_n \in V,$$

mit den Skalaren $x_1, x_2, \ldots, x_n$.
Falls für ein Element $b \in V$ gilt

$$b = x_1 v_1 + x_2 v_2 + \ldots + x_n v_n \in V,$$

mit geeignet gewählten Skalaren $x_1, x_2, \ldots, x_n$, dann sagt man: *$b$ lässt sich als Linearkombination von $v_1, v_2, \ldots, v_n$ darstellen.*

Man kann sich natürlich fragen, welche Menge sich durch gegebene $v_1, v_2, \ldots, v_n$ aus dem linearen Raum $V$ darstellen lässt.

**Definition 2.2.0.2. Span**

Wir bezeichnen die Menge aller Linearkombinationen der Elemente $\{v_1, v_2, \ldots, v_n\} \subset V$ mit

$$U = \mathrm{Span}\{v_1, v_2, \ldots, v_n\} = \{x_1 v_1 + x_2 v_2 + \ldots + x_n v_n \in V, \text{ mit } x_1, \ldots, x_n \text{ Skalare}\}.$$

Wir sagen, dass $U$ *aufgespannt* oder auch *erzeugt* wird von $v_1, v_2, \ldots, v_n \in V$.

**Beispiel 2.2.0.3 (Bild einer Matrix / Bild der Matrixabbildung ist ein linearer Unterraum)**
Sei $\mathbf{A}$ eine $m \times n$ Matrix und $\mathbf{x} \in \mathbb{R}^n$. Als wir die Matrix-Vektor-Multiplikation besprochen haben, haben wir gesehen, dass das Ergebnis eine Linearkombination der Spaltenvektoren von $\mathbf{A}$ ist:

$$\mathbf{A}\mathbf{x} = x_1 \mathbf{a}_1 + x_2 \mathbf{a}_2 + \ldots + x_n \mathbf{a}_n \in \mathbb{R}^m.$$

Diese Multiplikation können wir auch als eine Abbildung $\mathscr{A}$ von $\mathbb{R}^n$ nach $\mathbb{R}^m$ betrachten:

$$\begin{array}{rccc} \mathscr{A}: & \mathbb{R}^n & \to & \mathbb{R}^m \\ & \mathbf{x} & \mapsto & \mathbf{A}\mathbf{x} \end{array}.$$

Von dieser soeben definierten Abbildung betrachten wir das Bild:

**Definition 2.2.0.4. Bild**

Die Menge aller Bildelemente einer Abbildung heisst *Bild der Abbildung*.
Da die eben definierte Abbildung $\mathscr{A}$ von der Matrix $\mathbf{A}$ eindeutig definiert ist, nennen wir das Bild von $\mathscr{A}$ auch das *Bild der Matrix* $\mathbf{A}$.

$$\begin{aligned} \mathrm{Bild}\,\mathscr{A} &= \{\mathbf{y} \in \mathbb{R}^m : \mathbf{y} = \mathbf{A}\mathbf{x}, \mathbf{x} \in \mathbb{R}^n\} \\ &= \{\mathbf{y} \in \mathbb{R}^m : \mathbf{y} = x_1 \mathbf{a}_1 + \ldots + x_n \mathbf{a}_n\}, \end{aligned}$$

oder auch

$$\mathrm{Bild}\,\mathscr{A} = \mathrm{Span}\{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n\}.$$

Anders gesagt: Das Bild der Abbildung $\mathscr{A}$, auch das *Bild der Matrix* $\mathbf{A}$ genannt, wird von den Spalten $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n$ der Matrix $\mathbf{A}$ aufgespannt / erzeugt.

Schauen wir uns ein Zahlenbeispiel dazu an. Sei die Matrix $\mathbf{A}$:

$$\mathbf{A} = \begin{bmatrix} 1 & 0 \\ 4 & 3 \\ 2 & 3 \end{bmatrix}.$$

Wir haben:

$$\text{Bild}\,\mathscr{A} = \{\mathbf{Ax} \in \mathbb{R}^3, \mathbf{x} \in \mathbb{R}^2\} \quad = \{x_1 \begin{bmatrix} 1 \\ 4 \\ 2 \end{bmatrix} + x_2 \begin{bmatrix} 0 \\ 3 \\ 3 \end{bmatrix},\ x_1, x_2 \in \mathbb{R}\} = \text{Span}\{\begin{bmatrix} 1 \\ 4 \\ 2 \end{bmatrix},\ \begin{bmatrix} 0 \\ 3 \\ 3 \end{bmatrix}\}.$$

Das ist ein Unterraum von $\mathbb{R}^3$. Geometrisch betrachtet ist das eine 2-dimensionale Ebene in einem 3-dimensionalen Raum.

⌟

**Beispiel 2.2.0.5 (Span von Monomen)**

Seien $V$ und $U$ lineare Räume:

$$V = C^0[-1,1] = \{f : [-1,1] \to \mathbb{R},\ f \text{ stetig}\},$$
$$U = \mathscr{P}_n[-1,1] = \{p : [-1,1] \to \mathbb{R},\ p(t) = a_0 + a_1 t + \ldots + a_{n-1}t^{n-1} \quad \text{mit } a_0, a_1, \ldots, a_{n-1} \in \mathbb{R}\}.$$

Wir können direkt sehen, dass $U$ ein Unterraum von $V$ ist: Polynome sind stetige Funktionen. Die Summe zweier Polynome vom Grad höchstens $n-1$ und deren Produkt mit einem Skalar sind Polynome vom Grad höchstens $n-1$.

Des Weiteren lässt sich jedes Element in $U$ als eine Linearkombination der Monome $p_0, p_1, \ldots, p_{n-1} \in U$ schreiben. Zur Erinnerung: Die Monome $p_k$ sind:

$$p_0(t) = 1,\ p_1(t) = t,\ p_2(t) = t^2, \ldots,\ p_{n-1}(t) = t^{n-1}.$$

Sei $p \in U$ beliebig: $p : [-1,1] \to \mathbb{R}$ mit

$$p(t) = a_0 + a_1 t + a_2 t^2 + \ldots + a_{n-1}t^{n-1}.$$

Dies können wir aber auch mittels der oben definierten Monome aufschreiben:

$$p = a_0 p_0 + a_1 p_1 + a_2 p_2 + \ldots + a_{n-1} p_{n-1}.$$

Wir erhalten so eine andere Betrachtungsweise der Polynome: Anstatt die Polynome als Funktionen mit dem Parameter $t$ zu betrachten, können wir diese als Elemente eines Vektorraumes auffassen. Insbesondere verschwindet in dieser neuen Betrachtungsweise die Abhängigkeit von dem Parameter $t$. Stattdessen entsteht eine Abhängigkeit von den Monomen. Wir können also sagen, dass $U$ von den Monomen $p_0, p_1, \ldots, p_{n-1}$ aufgespannt wird:

$$U = \text{Span}\{p_0, p_1, \ldots, p_{n-1}\}.$$

⌟

**Beispiel 2.2.0.6 (Raum $\mathscr{P}_5$ wird auch von Legendre-Polynomen erzeugt)**
In diesem Beispiel schauen wir uns den linearen Raum $\mathscr{P}_5$ der Polynome vom Grad höchstens 4 an.

$$\mathscr{P}_5 = \text{Span}\{p_0, p_1, p_2, p_3, p_4\}$$

Bevor wir aber weitergehen müssen wir uns folgende Bemerkung anschauen.

**Bemerkung 2.2.0.7 (Die ersten Legendre-Polynome)**
Wir definieren die ersten Legendre-Polynome, $q_0,\ q_1,\ q_2,\ q_3,\ q_4$ in Abhängigkeit der Monome:

$$\begin{aligned}
q_0: &\quad q_0 = p_0\\
q_1: &\quad q_1 = p_1\\
q_2: &\quad q_2(t) = \tfrac{3}{2}t^2 - \tfrac{1}{2} = \tfrac{3}{2}p_2(t) - \tfrac{1}{2}p_0(t)\\
&\quad q_2 = \tfrac{3}{2}p_2 - \tfrac{1}{2}p_0\\
q_3: &\quad q_3(t) = \tfrac{5}{2}t^3 - \tfrac{3}{2}t = \tfrac{5}{2}p_3(t) - \tfrac{3}{2}p_1(t)\\
&\quad q_3 = \tfrac{5}{2}p_3 - \tfrac{3}{2}p_1\\
q_4: &\quad q_4(t) = \tfrac{35}{8}t^4 - \tfrac{15}{4}t^2 + \tfrac{3}{8} = \tfrac{35}{8}p_4(t) - \tfrac{15}{4}p_2(t) + \tfrac{3}{8}p_0(t)\\
&\quad q_4 = \tfrac{35}{8}p_4 - \tfrac{15}{4}p_2 + \tfrac{3}{8}p_0\,.
\end{aligned}$$

⌟

Eine Eigenschaft dieser Legendre-Polynome $q_0,\ q_1,\ q_2,\ q_3$ und $q_4$ im Zusammenhang mit $\mathcal{P}_5$ ist, dass

$$\mathcal{P}_5 = \operatorname{Span}\{q_0,\ q_1,\ q_2,\ q_3,\ q_4\}\,.$$

Diese Aussage müssen wir aber auch beweisen. Deswegen definieren wir $B$ als

$$B = \operatorname{Span}\{q_0,\ q_1,\ q_2,\ q_3,\ q_4\}$$

und zeigen, dass $\mathcal{P}_5 = B$.
Diese Gleichheit zeigen wir aber nicht direkt. Wir zeigen, dass die Aussagen $\mathcal{P}_5 \subseteq B$ und $\mathcal{P}_5 \supseteq B$ beide gelten. Diese beiden Aussagen können aber nur gleichzeitig korrekt sein, wenn die beiden Mengen gleich sind. Deswegen reicht der Beweis dieser beiden Aussagen aus, um $\mathcal{P}_5 = B$ zu zeigen.

**Aussage $\mathcal{P}_5 \supseteq B$:**

Wenn wir für ein beliebiges $q \in B$ zeigen können, dass es auch in $\mathcal{P}_5$ ist, so muss $B$ in $\mathcal{P}_5$ liegen.
Sei $q \in B = \operatorname{Span}\{q_0, \ldots,\ q_4\}$ beliebig. Dies bedeutet, dass $q$ eine Linearkombination von $q_0, \ldots,\ q_4$ ist:

$$q = c_0q_0 + c_1q_1 + c_2q_2 + c_3q_3 + c_4q_4\,.$$

Hier ersetzen wir nun $q_0, \ldots, q_4$ durch ihre Definition als Linearkombination der Monome $p_0, \ldots,\ p_4$ welche wir in der Bemerkung 2.2.0.7 gesehen haben:

$$\begin{aligned}
q &= c_0p_0 + c_1p_1 + c_2(\frac{3}{2}p_2 - \frac{1}{2}p_0) + c_3(\frac{5}{2}p_3 - \frac{3}{2}p_1) + c_4(\frac{35}{8}p_4 - \frac{15}{4}p_2 + \frac{3}{8}p_0)\\
&= (c_0 - \frac{1}{2}c_2 + \frac{3}{8}c_4)p_0 + (c_1 - \frac{3}{2}c_3)p_1 + (\frac{3}{2}c_2 - \frac{15}{4}c_4)p_2 + \frac{5}{2}c_3p_3 + \frac{35}{8}c_4p_4\,.
\end{aligned}$$

Wir sehen also, dass $q$ eine Linearkombination von $p_0, \ldots,\ p_4$ ist und deswegen auch in $\mathcal{P}_5$ sein muss. Also haben wir nun bewiesen, dass ein beliebiges Element von $B$ auch in $\mathcal{P}_5$ ist. Darum:

$$B \subseteq \mathcal{P}_5\,.$$

**Aussage $\mathcal{P}_5 \subseteq B$:**

Diesen Beweis werden wir analog zum Vorherigen führen, wir nehmen ein beliebiges Element $p$ aus $\mathcal{P}_5$ und zeigen, dass dieses auch in $B$ liegt.
Sei $p \in \mathcal{P}_5$ beliebig. Dieses $p$ lässt sich als Linearkombination von $p_0, \ldots,\ p_4$ schreiben:

$$p = a_0p_0 + a_1p_1 + a_2p_2 + a_3p_3 + a_4p_4\,.$$

Unser Ziel ist nun, dieses $p$ als eine Linearkombination von $q_0, \dots, q_4$ zu schreiben. Dies können wir erreichen, indem wir die Monome $p_0, \dots, p_4$ durch die $q_0, \dots, q_4$ beschreiben und dann gleich wie beim Beweis der vorherigen Aussage weitergehen.

Wie können wir aber $p_k$ durch eine Linearkombination von $q_0, \dots, q_4$ darstellen? Für dies schauen wir uns die Definition dieser Legendre-Polynome nochmals an:

$$\begin{aligned} q_0 &= p_0 \\ q_1 &= p_1 \\ q_2 &= \frac{3}{2}p_2 - \frac{1}{2}p_0 \\ q_3 &= \frac{5}{2}p_3 - \frac{3}{2}p_1 \\ q_4 &= \frac{35}{8}p_4 - \frac{15}{4}p_2 + \frac{3}{8}p_0 \,. \end{aligned}$$

Dies entspricht einem LGS mit den Unbekannten $p_0, \dots, p_4$. Wenn wir dieses lösen, so erhalten wir die gewünschte Darstellung von $p_k$ als Linearkombination von $q_0, \dots q_4$:

$$\begin{aligned} p_0 &= q_0 \\ p_1 &= q_1 \\ p_2 &= \frac{1}{3}(q_0 + 2q_2) \\ p_3 &= \frac{1}{5}(3q_1 + 2q_3) \\ p_4 &= \frac{1}{35}(7q_0 + 20q_2 + 8q_4) \,. \end{aligned}$$

Dies können wir nun in die obige Definition von $p$ einsetzen, die $q_0, .., q_4$ herausfaktorisieren und erhalten, dass $p \in B$. Da $p$ beliebig ist, haben wir:

$$\mathscr{P}_5 \subseteq B \,.$$

Somit haben wir beide Aussagen bewiesen und es gilt, dass $\mathscr{P}_5 = B$ und demzufolge auch, dass

$$\mathscr{P}_5 = \mathrm{Span}\{q_0, q_1, q_2, q_3, q_4\} \,.$$

⌟

**Definition 2.2.0.8. Erzeugendensystem**

Sei $V$ ein linearer Raum. Die Menge $\{v_1, v_2, \dots, v_n\} \subset V$ heisst *Erzeugendensystem* in $V$, falls sich jedes $b \in V$ als eine Linearkombination von $v_1, \dots, v_n$ schreiben lässt.

**Definition 2.2.0.9. Kern einer Matrix**

Sei $\mathbf{A}$ eine $m \times n$ Matrix. Wir nennen die Menge der Vektoren $\mathbf{x} \in \mathbb{R}^n$, die von $\mathbf{A}$ zu null abgebildet werden, den *Kern der Matrix* $\mathbf{A}$:

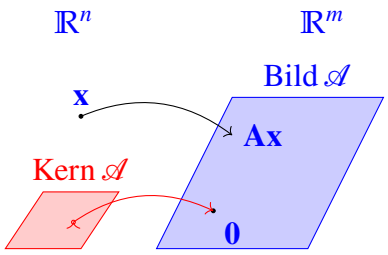
$$\mathrm{Kern}\,\mathbf{A} = \{\mathbf{x} \in \mathbb{R}^n \text{ so dass } \mathbf{A}\mathbf{x} = \mathbf{0}\} \,.$$

Das entspricht der Menge der Lösungen des homogenen linearen Gleichungssystems $\mathbf{A}\mathbf{x} = \mathbf{0}$.

**Beispiel 2.2.0.10 (Kern einer Matrix / Kern der Matrixabbildung ist ein linearer Unterraum)**
Wir können wieder die entsprechende Abbildung $\mathscr{A}$ definieren:

$$\begin{array}{rccc} \mathscr{A}: & \mathbb{R}^n & \to & \mathbb{R}^m \quad \mathscr{A}(\mathbf{x}) = \mathbf{A}\mathbf{x} \\ & \mathbf{x} & \mapsto & \mathbf{A}\mathbf{x} \end{array}$$

Diese Abbildung können wir auch visuell betrachten: Wir haben in der Figur unten den Raum $\mathbb{R}^n$, den Raum $\mathbb{R}^m$ und die Abbildung $\mathscr{A}$ schematisch dargestellt. Die Elemente aus $\mathbb{R}^n$ werden auf eine Teilmenge von $\mathbb{R}^m$ abgebildet, das Bild von $\mathscr{A}$. Im Raum $\mathbb{R}^n$ gibt es eine spezielle Teilmenge, den Kern von $\mathscr{A}$, welche auf $\mathbf{0}$ abgebildet wird.




**Bemerkung 2.2.0.11 (Der Kern ist ein Unterraum)**
Kern $\mathbf{A}$ ist ein Unterraum von $\mathbb{R}^n$.

⌟

Dies lässt sich relativ einfach beweisen.
*Beweis.* Um dies zu beweisen, prüfen wir einfach die beiden Bedingungen für den Unterraum.

Seien $\mathbf{x}_1, \mathbf{x}_2 \in \operatorname{Kern}\mathbf{A}$ beliebig gewählt, es gilt also $\mathbf{A}\mathbf{x}_1 = \mathbf{0}$ und $\mathbf{A}\mathbf{x}_2 = \mathbf{0}$. Wir müssen nun überprüfen, dass $\mathbf{x}_1 + \mathbf{x}_2$ auch im Kern von $\mathbf{A}$ ist:

$$\mathbf{A}(\mathbf{x}_1 + \mathbf{x}_2) = \mathbf{A}\mathbf{x}_1 + \mathbf{A}\mathbf{x}_2 = \mathbf{0} + \mathbf{0} = \mathbf{0},$$

also haben wir, dass $\mathbf{x}_1 + \mathbf{x}_2 \in \operatorname{Kern}\mathbf{A}$.

Sei nun $\alpha$ ein Skalar. Wir zeigen: Wenn wir $\mathbf{x}_1$ mit diesem Skalar multiplizieren, ist dies immer noch Teil vom Kern:

$$\mathbf{A}(\alpha\mathbf{x}_1) = \alpha(\mathbf{A}\mathbf{x}_1) = \alpha\mathbf{0} = \mathbf{0},$$

wir haben also auch, dass $\alpha\mathbf{x}_1 \in \operatorname{Kern}\mathbf{A}$.

Mit dem Beweis dieser beiden Aussagen erhalten wir, dass Kern $\mathbf{A}$ ein Unterraum von $\mathbb{R}^n$ ist.

□

Das heisst, sobald wir eine Matrix haben, haben wir auch sofort zwei Unterräume definiert, das Bild der Matrix und den Kern der Matrix.

Dies werden wir nun auch noch an einem Zahlenbeispiel sehen[1]. Sei eine Matrix $\mathbf{A}$ gegeben als:

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 2 & 3 \\ 2 & 2 & 8 & 10 \\ 3 & 3 & 10 & 13 \end{bmatrix}.$$

Von dieser Matrix werden wir nun das Bild und den Kern bestimmen. Betrachten wir das lineare Gleichungssystem $\mathbf{A}\mathbf{x} = \mathbf{b}$. Die Gauss-Elimination liefert:

$$\left[\begin{array}{cccc|l} 1 & 1 & 2 & 3 & b_1 \\ 0 & 0 & 4 & 4 & b_2 - 2b_1 \\ 0 & 0 & 0 & 0 & b_3 - b_2 - b_1 \end{array}\right].$$

[1] Dieses Zahlenbeispiel wird im Abschnitt 9.3 mit Python gelöst.

Wir haben also die Pivotelemente 1 und 4 und die freien Variablen $x_2$ und $x_4$. Die Kompatibilitätsbedingung dieses Systems ist:

$$b_3 - b_2 - b_1 = 0\,.$$

**Beispiel 2.2.0.12 (Berechnung vom Kern)**
Der Kern von $\mathbf{A}$ ist die Menge aller Vektoren $\mathbf{x}$ für welche $\mathbf{Ax} = \mathbf{0}$.

Es ist wichtig zu bemerken, dass die triviale Lösung $\mathbf{0}$ des homogenen LGS immer ein Element des Kerns ist. Somit ist das System immer lösbar und die Prüfung der Kompatibilitätsbedingung kann entfallen.

Wir können per Rücksubstitution die Lösungen des homogenen LGS erhalten:

$$\mathbf{Ax} = \mathbf{0} \implies \mathbf{x} = \begin{bmatrix} -x_2 - x_4 \\ x_2 \\ -x_4 \\ x_4 \end{bmatrix}.$$

Mit diesem $\mathbf{x}$ sind wir nun in der Lage den Kern von $\mathbf{A}$ als Menge zu schreiben:

$$\operatorname{Kern}\mathbf{A} = \{\mathbf{x} = \begin{bmatrix} -x_2 - x_4 \\ x_2 \\ -x_4 \\ x_4 \end{bmatrix} \in \mathbb{R}^4 \text{ mit } x_2,\ x_4 \in \mathbb{R}\} \subset \mathbb{R}^4\,.$$

Dies ist ein nicht trivialer Unterraum von $\mathbb{R}^4$.

Wir können nun auch spezielle Werte für $x_2$ und $x_4$ wählen, die uns helfen werden die obige Menge anders darzustellen:

$$x_2 = 1,\ x_4 = 0 \quad \implies \quad \text{spezielle Lösung } \begin{bmatrix} -1 \\ 1 \\ 0 \\ 0 \end{bmatrix}.$$

$$x_2 = 0,\ x_4 = 1 \quad \implies \quad \text{spezielle Lösung } \begin{bmatrix} -1 \\ 0 \\ -1 \\ 1 \end{bmatrix}.$$

Wir können nun die allgemeine Lösung $\mathbf{x}$ in diese beiden Vektoren aufteilen:

$$\mathbf{x} = \begin{bmatrix} -x_2 - x_4 \\ x_2 \\ -x_4 \\ x_4 \end{bmatrix} = \begin{bmatrix} -x_2 \\ x_2 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} -x_4 \\ 0 \\ -x_4 \\ x_4 \end{bmatrix} = x_2 \begin{bmatrix} -1 \\ 1 \\ 0 \\ 0 \end{bmatrix} + x_4 \begin{bmatrix} -1 \\ 0 \\ -1 \\ 1 \end{bmatrix}.$$

Also ist der Kern von $\mathbf{A}$ von diesen beiden Vektoren aufgespannt. Wir können das auch schreiben als:

$$\operatorname{Kern}\mathbf{A} = \operatorname{Span}\{\underbrace{\begin{bmatrix} -1 \\ 1 \\ 0 \\ 0 \end{bmatrix}}_{\mathbf{v}_1}, \underbrace{\begin{bmatrix} -1 \\ 0 \\ -1 \\ 1 \end{bmatrix}}_{\mathbf{v}_2}\} = \operatorname{Span}\{\mathbf{v}_1,\ \mathbf{v}_2\}\,.$$

**Beispiel 2.2.0.13 (Berechnung vom Bild)**
Nun wollen wir das Bild von $\mathbf{A}$ finden. Ein Vektor $\mathbf{b}$ ist im Bild von $\mathbf{A}$ genau dann, wenn $\mathbf{b} = \mathbf{A}\mathbf{x}$ Lösungen hat. Dies ist aber nur genau dann der Fall, wenn die Kompatibilitätsbedingung erfüllt ist:

$$b_3 - b_2 - b_1 = 0$$

oder auch

$$b_3 = b_1 + b_2 \,.$$

Jeder Vektor, der diese Kompatibilitätsbedingung erfüllt, liegt in $\operatorname{Bild}\mathbf{A}$. Das Bild von $\mathbf{A}$ können wir dann schreiben als:

$$\operatorname{Bild}\mathbf{A} = \{\mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ b_1 + b_2 \end{bmatrix} \text{ mit } b_1,\ b_2 \in \mathbb{R}\} \,.$$

Das ist ein Unterraum von $\mathbb{R}^3$. Analog zur Vorgehensweise beim Kern können wir diesen Vektor in zwei Vektoren aufteilen, indem wir die Linearität verwenden. Anschaulich gesprochen ergeben sich die Vektoren, indem wir abwechselnd $b_1$ und $b_2$ auf $0$ beziehungsweise $1$ setzen:

$$\begin{bmatrix} b_1 \\ b_2 \\ b_1 + b_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ 0 \\ b_1 \end{bmatrix} + \begin{bmatrix} 0 \\ b_2 \\ b_2 \end{bmatrix} = b_1 \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} + b_2 \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} .$$

Auch hier können wir nun sagen, dass das Bild von $\mathbf{A}$ von diesen beiden Vektoren aufgespannt wird:

$$\operatorname{Bild}\mathbf{A} = \operatorname{Span}\{ \underbrace{\begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}}_{\mathbf{w}_1} , \underbrace{\begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}}_{\mathbf{w}_2} \} = \operatorname{Span}\{\mathbf{w}_1,\ \mathbf{w}_2\} \,.$$

**Bemerkung 2.2.0.14 (Allgemeine Lösung eines LGS)**
Für dieses lineare Gleichungssystem $\mathbf{A}\mathbf{x} = \mathbf{b}$ können wir die beiden freien Variablen $x_2$ und $x_4$ auf $0$ setzen und erhalten dann die sogenannte *partikuläre Lösung* $\mathbf{x}^p$:

$$\mathbf{x}^p = \begin{bmatrix} 2b_1 - \frac{1}{2}b_2 \\ 0 \\ \frac{1}{4}b_2 - \frac{1}{2}b_1 \\ 0 \end{bmatrix} .$$

Mit dieser partikulären Lösung und der Lösung des homogenen Gleichungssystems $\mathbf{A}\mathbf{x} = \mathbf{0}$ (die wir vorher gefunden haben) können wir die *allgemeine Lösung* von $\mathbf{A}\mathbf{x} = \mathbf{b}$ als die Summe der beiden schreiben:

$$\mathbf{x} = \mathbf{x}^p + \underbrace{\alpha\mathbf{v}_1 + \beta\mathbf{v}_2}_{\text{allgemeine Lösung des homogenen Systems } \mathbf{A}\mathbf{x} = \mathbf{0}} .$$

Die hier an einem Zahlenbeispiel illustrierte Vorgangsweise, gilt auch für beliebige Matrizen $\mathbf{A}$. Der Sinn dieses Zahlenbeispiels ist es einem die Mechanismen zu zeigen, so dass man dann in der Lage ist, diese auf allgemeine Probleme anzuwenden.

Diese Aussage, dass $\mathbf{x}$ genau dann das lineare Gleichungssystem $\mathbf{Ax} = \mathbf{b}$ löst, wenn sich $\mathbf{x}$ als Summe der partikulären Lösung und einer Lösung des homogenen Gleichungssystems $\mathbf{Ax} = 0$ schreiben lässt, müssen wir aber noch beweisen. Wir betrachten dabei weiterhin das vorhergehende Zahlenbeispiel. Zur Erinnerung: Wir haben mit $\mathbf{v}_1$ und $\mathbf{v}_2$ jene Vektoren notiert, welche den Kern von $\mathbf{A}$ aufspannen.

*Beweis.* Wir werden hier beide Richtungen der Aussage einzeln beweisen.

**Richtung $\Leftarrow$:**

Wir beginnen mit der Aussage, dass $\mathbf{x} = \mathbf{x}^p + \alpha\mathbf{v}_1 + \beta\mathbf{v}_2$ und beweisen, dass dann $\mathbf{x}$ die Lösung von $\mathbf{Ax} = \mathbf{b}$ ist.

$$\begin{aligned}\mathbf{Ax} &= \mathbf{Ax}^p + \alpha\mathbf{Av}_1 + \beta\mathbf{Av}_2\\ &= \mathbf{b} + \mathbf{0} + \mathbf{0}\end{aligned}$$

$$\mathbf{Ax} = \mathbf{b}\,.$$

Und somit ist diese Richtung bewiesen.

**Richtung $\Rightarrow$:**

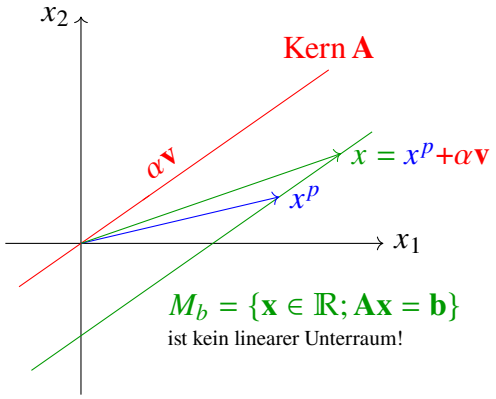
Wir nehmen einen Vektor $\mathbf{x}$ welcher eine Lösung von $\mathbf{Ax} = \mathbf{b}$ ist, gleichzeitig wissen wir aber auch, dass $\mathbf{Ax}^p = \mathbf{b}$ eine Lösung ist. Wenn wir diese beiden Gleichungen voneinander subtrahieren erhalten wir:

$$\mathbf{A}(\mathbf{x} - \mathbf{x}^p) = \mathbf{0}\,.$$

Das bedeutet, dass $\mathbf{x} - \mathbf{x}^p$ im Kern von $\mathbf{A}$ liegt, und so durch ein Linearkombination $\alpha\mathbf{v}_1 + \beta\mathbf{v}_2$ darstellbar sein muss. Durch weiteres Umformen erhalten wir dann das gesuchte Resultat:

$$\mathbf{x} - \mathbf{x}^p = \alpha\mathbf{v}_1 + \beta\mathbf{v}_2 \quad \Longleftrightarrow \quad \mathbf{x} = \mathbf{x}^p + \alpha\mathbf{v}_1 + \beta\mathbf{v}_2\,.$$

Dies ist die Algebra dahinter. Wir können das Ganze aber auch geometrisch in zwei Dimensionen betrachten. In zwei Dimensionen ist der Kern von $\mathbf{A}$ eine Gerade, die durch den Nullpunkt gehen muss (rot eingezeichnet). Jeder Punkt auf dieser Geraden ist durch $\alpha\mathbf{v}$ beschrieben. (Wir betrachten hier, anders als im Beweis, einen Kern, der von einem einzigen Vektor $\mathbf{v}$ aufgespannt wird) Auch haben wir einen Vektor $\mathbf{x}^p$.
Die Bemerkung sagt nun, dass die Menge $M_b$, welche alle Lösungen für ein fixes $\mathbf{b}$ enthält, auf einer (grünen) Gerade liegt, die parallel zum Kern ist und durch den Punkt $\mathbf{x}^p$ geht. Wichtig ist, dass dieses $M_b$ im Allgemeinen kein Unterraum ist. Dies kann man bei dem dargestellten Beispiel sofort erkennen, da der Ursprung nicht Teil der Gerade ist. Es ist aber ein verschobener linearer Unterraum, da es um $\mathbf{x}^p$ parallel zum Kern verschoben ist.




□

⌟

**Beispiel 2.2.0.15 (Erzeugende Systeme)**

1) Wir beginnen mit dem einfachen Beispiel von $\mathbb{R}^n$: Wir haben bereits gesehen, dass dieser Raum von den Vektoren $\mathbf{e}_1, \ldots, \mathbf{e}_n$ erzeugt wird:

$$\mathbf{e}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \mathbf{e}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \ldots, \mathbf{e}_n = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}.$$

2) Sei $\mathbf{A}$ eine $n \times n$ Matrix mit Rang $n$, dann hat das lineare Gleichungssystem $\mathbf{Ax} = \mathbf{b}$ eine eindeutige Lösung für jedes $\mathbf{b} \in \mathbb{R}^n$. Dies bedeutet, dass $\{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n\}$, die Spalten der Matrix $\mathbf{A}$, ein Erzeugendensystem in $\mathbb{R}^n$ sind.

3) Die Monome $\{p_0, p_1, \ldots, p_{n-1}\}$ erzeugen den Raum $\mathcal{P}_n$.

4) Wir betrachten den Raum $\mathbb{R}^3$ mit den vier Vektoren $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3, \mathbf{a}_4$ definiert als:

$$\mathbf{a}_1 = \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix}, \mathbf{a}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \mathbf{a}_3 = \begin{bmatrix} 2 \\ 0 \\ 0 \end{bmatrix}, \mathbf{a}_4 = \begin{bmatrix} 2 \\ 1 \\ 0 \end{bmatrix}.$$

Wir sehen, dass $\{\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3, \mathbf{a}_4\}$ kein erzeugendes System für $\mathbb{R}^3$ bildet, sondern nur für

$$V = \operatorname{Span}\{\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}\},$$

also eigentlich für den Raum $\mathbb{R}^2$ welcher ein Unterraum von $\mathbb{R}^3$ ist; auch würden die Vektoren $\mathbf{a}_2$ und $\mathbf{a}_3$ ausreichen um $V$ zu erzeugen.

5) Für den Raum aller Polynome $P$ gibt es keine endliche Menge von Polynomen, die $\mathcal{P}$ erzeugen.

Es gibt also einen fundamentalen Unterschied zwischen $\mathcal{P}$ und $\mathcal{P}_n$ oder $\mathbb{R}^n$, da wir $\mathcal{P}_n$ und $\mathbb{R}^n$ aus endlich vielen Elementen erzeugen können, während dies bei $\mathcal{P}$ nicht möglich ist.

⌟

**Definition 2.2.0.16. Endlichdimensionale lineare Räume**

Ein linearer Raum $V$ heisst *endlichdimensional*, wenn es eine endliche Menge $\{v_1, \ldots, v_n\} \subset V$ gibt, die erzeugend für $V$ ist.

**Beispiel 2.2.0.17 (Endlichdimensionale lineare Räume)**
Die Räume $\mathbb{R}^2$ und $\mathcal{P}_n$ sind endlichdimensional.
Die Räume $\mathcal{P}$, $C^k$ oder $L^2$ sind nicht endlichdimensional.

⌟

**Beispiel 2.2.0.18 (Nützliche endlichdimensionale Unterräume)**

Nehmen wir an, wir wollen eine Funktion aus $C^4$ im Computer darstellen. Wir werden schnell bemerken, dass dies nicht einfach ist, denn $C^4$ ist ein unendlichdimensionaler Raum und ein Computer ist eine endliche Maschine. Um dieses Problem zu umgehen, können wir die Funktion approximieren, zum Beispiel mit der Taylor-Approximation um den Punkt $0$:

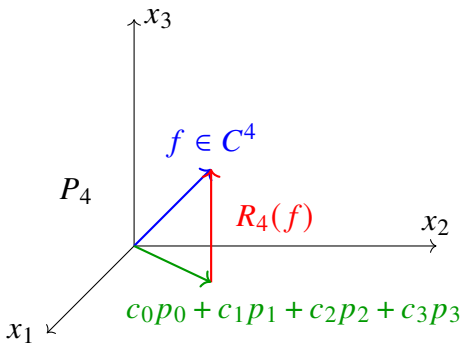
$$f(t) = f(0)t^0 + \frac{f'(0)}{1!}t + \frac{f''(0)}{2!}t^2 + \frac{f^{(3)}(0)}{3!}t^3 + R_4(f)\,.$$

Wir sehen nun also, dass wir die Funktion mit den Monomen approximieren können:

$$\underbrace{f}_{\in C^4} = \underbrace{c_0p_0 + c_1p_1 + c_2p_2 + c_3p_3}_{\in \mathcal{P}_4} + \underbrace{R_4(f)}_{\notin \mathcal{P}_4}\,.$$

Diese Approximation hat es uns also ermöglicht, eine Funktion aus dem unendlichdimensionalen Raum $C^4$ durch Elemente aus dem endlichdimensionalen Raum $\mathcal{P}_4$ zu ersetzen. Wenn wir dies in einem Computer speichern wollen, so müssen wir einfach nur die Koeffizienten $c_k$ speichern.

Das lässt sich auch graphisch veranschaulichen.




Die Funktion $f$ (hier dargestellt als Vektor) wird von dem Polynom $c_0p_0 + c_1p_1 + c_2p_2 + c_3p_3$ approximiert, der Fehler wird mit dem roten Vektor $R_4(f)$ dargestellt.

In diesem Bild sieht es so aus, als ob $R_4$ senkrecht auf dem grünen Vektor steht, die Approximation also gut ist. In Wirklichkeit ist dies aber nicht immer der Fall, wenn wir die Monome als Basis verwenden. Es gibt deutlich bessere Darstellungsmöglichkeiten als mit Monomen!

⌟

## 2.3 Lineare Unabhängigkeit und Basis

Im Beispiel 2.2.0.12 haben wir ein Erzeugendensystem von Bild $\mathbf{A}$ gesehen, welches aus vier Vektoren (die Spalten der Matrix $\mathbf{A}$) besteht. Wir haben dann aber auch gesehen, dass wir eigentlich nur zwei dieser vier Vektoren brauchen, um Bild $\mathbf{A}$ zu erzeugen. Wir können uns also folgende Frage stellen:
Kann man ein erzeugendes System $\{v_1, \ldots, v_n\}$ in eine möglichst kleine Menge $\{v_{j_1}, v_{j_2}, \ldots, v_{j_r}\}$ reduzieren, so dass

$$\mathrm{Span}\{v_1, .., v_n\} = \mathrm{Span}\{v_{j_1}, \ldots, v_{j_r}\}$$

gilt?

Bevor wir eine Antwort auf diese Frage geben, schauen wir zuerst den folgenden Trick an.

**Bemerkung 2.3.0.1 (Lineare Abhängigkeit)**
Betrachten wir eine Linearkombination von $\{v_1, \ldots, v_n\}$ in $V$, die gleich null ist (eine solche Linearkombination muss existieren, da $\mathbf{0}$ Teil jedes linearen Raumes ist):

$$x_1 v_1 + x_2 v_2 + \ldots + x_n v_n = \mathbf{0}\,.$$

Wenn wir nun die Annahme treffen, dass das $x_n \neq 0$ ist, so sind wir in der Lage durch dieses $x_n$ zu teilen und einen Ausdruck für $v_n$ zu erhalten:

$$v_n = -\frac{x_1}{x_n} v_1 - \frac{x_2}{x_n} v_2 - \ldots - \frac{x_{n-1}}{x_n} v_{n-1}\,.$$

Das heisst, dass $v_n$ eine Linearkombination der $v_1, \ldots, v_{n-1}$ ist. Deswegen können wir sagen, dass $\{v_1, .., v_{n-1}\}$ bereits ein erzeugendes System ist, und wir $v_n$ weglassen können. Diese Berechnung motiviert die wichtige Definition der linearen Unabhängigkeit.

⌟

**Definition 2.3.0.2. Lineare Unabhängigkeit**

Die Elemente $v_1, v_2, \ldots, v_n$ eines linearen Raumes $V$ sind *linear unabhängig*, falls

$$x_1 v_1 + x_2 v_2 + \ldots + x_n v_n = \mathbf{0} \quad \Longrightarrow \quad x_1 = x_2 = \ldots = x_n = 0\,.$$

Sonst heissen $v_1, v_2, \ldots, v_n$ *linear abhängig*.

**Beispiel 2.3.0.3 (Wichtige Mengen linear unabhängiger Vektoren)**

1) Die Vektoren der Standardbasis $\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_n \in \mathbb{R}^n$ sind linear unabhängig. Das zeigen wir, indem wir die Definition der linearen Unabhängigkeit überprüfen:

$$x_1 \mathbf{e}_1 + x_2 \mathbf{e}_2 + \ldots + x_n \mathbf{e}_n = \mathbf{0}$$

$$x_1 \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + x_2 \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + \ldots + x_n \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} = \mathbf{0}$$

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \mathbf{0}\,.$$

Dies ist genau dann der Fall, wenn $x_1 = x_2 = x_3 = \ldots = x_n = 0$. Deswegen sind die Vektoren $\mathbf{e}_1, .., \mathbf{e}_n$ linear unabhängig.

2) Wann sind beliebige Vektoren $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_k \in \mathbb{R}^n$ linear unabhängig?

Wir müssen die Definition der linearen Unabhängigkeit überprüfen:

$$x_1\mathbf{a}_1 + x_2\mathbf{a}_2 + \ldots + x_k\mathbf{a}_k = \mathbf{0} \quad \Longrightarrow \quad \mathbf{x} = \mathbf{0}\,.$$

Solch eine Linearkombination haben wir aber schon oft gesehen, denn sie entspricht der Matrix-Vektor-Multiplikation. Wir können dies also umschreiben:

$$\mathbf{A}\mathbf{x} = \mathbf{0} \quad \Longrightarrow \quad \mathbf{x} = \mathbf{0}\,.$$

Das heisst, dass die Vektoren $\mathbf{a}_1, \ldots, \mathbf{a}_k$ genau dann linear unabhängig sind, wenn das homogene Gleichungssystem $\mathbf{A}\mathbf{x} = \mathbf{0}$ nur die triviale Lösung ($\mathbf{x} = \mathbf{0}$) hat, d. h. es keine freien Variablen gibt. Dies ist genau dann der Fall, wenn die $n \times k$ Matrix $\mathbf{A}$ Rang $k \leq n$ hat.

Das bedeutet auch, dass mehr als $n$ Vektoren in $\mathbb{R}^n$ nicht linear unabhängig sein können.

3) Die Monome $p_0, p_1, \ldots, p_{n-1}$ sind linear unabhängig.

*Beweis.* Um dies zu beweisen, werden wir wieder die Definition der linearen Unabhängigkeit überprüfen und dabei einen wichtigen Trick anwenden, der uns auch später noch helfen wird.

Seien $x_1, x_2, \ldots, x_n \in \mathbb{R}$ Skalare. Die Bedingung $x_0p_0 + x_1p_1 + \ldots + x_{n-1}p_{n-1} = \mathbf{0}$ bedeutet

$$\begin{aligned} x_1p_0(t) + x_2p_1(t) + \ldots + x_np_{n-1}(t) &= 0 \quad \text{für alle } t \in \mathbb{R} \\ x_1 1 + x_2 t + \ldots + x_n t^{n-1} &= 0 \quad \text{für alle } t \in \mathbb{R}\,. \end{aligned}$$

Da dies für alle $t \in \mathbb{R}$ gelten muss, gilt dies auch für $t = 0$ und in dieser Summe verschwinden dann alle Terme ausser dem ersten und es bleibt:

$$x_1 = 0\,.$$

Wir haben also gezeigt, dass der erste Koeffizienten $x_1 = 0$ ist. Somit können wir die Linearkombination umschreiben:

$$\begin{aligned} x_2p_1 + x_3p_2 + \ldots + x_np_{n-1} &= 0 \quad \text{für alle } t \in \mathbb{R} \\ x_2 t + x_3 t^2 + \ldots + x_n t^{n-1} &= 0 \quad \text{für alle } t \in \mathbb{R} \\ t(x_2 + x_3 t + \ldots + x_n t^{n-2}) &= 0 \quad \text{für alle } t \in \mathbb{R}\,. \end{aligned}$$

Von hier aus können wir nun entweder algebraisch oder analytisch argumentieren und sagen, dass

$$x_2 + x_3 t + \ldots + x_n t^{n-2} = 0 \quad \text{für alle } t \in \mathbb{R}$$

gelten muss.

Die algebraische Argumentation sagt, dass $tz = 0$ für alle $t$ impliziert, dass $z$ Null sein muss.

Die analytische Argumentation ist: Falls $tz(t) = 0$ für alle $t \in \mathbb{R}$ und $z(t)$ stetig ist (da es ein Polynom ist), dann muss $z(t) = 0$ für alle $t \in \mathbb{R}$ sein.

Wir können nun also wieder $t = 0$ setzen und daraus schliessen, dass $x_2 = 0$ ist. Wir gehen dann genau gleich weiter und erhalten zum Schluss, dass

$$x_1 = x_2 = \ldots = x_n = 0\,.$$

Dies sagt uns, dass die Monome $p_0, \ldots, p_{n-1}$ linear unabhängig sind.

□

4) Die Funktionen $f(t) = 1$, $g(t) = \cos^2(t)$ und $h(t) = \cos(2t)$ sind linear abhängig, denn:

$$\cos(2t) = 2\cos^2(t) - 1 \quad \Longleftrightarrow \quad g = \frac{1}{2}f + \frac{1}{2}h\,.$$

$g$ ist also eine Linearkombination von $f$ und $h$ und deswegen sind diese drei Funktionen nicht linear unabhängig.

⌟

Das zweite Teilbeispiel über die lineare Unabhängigkeit von Vektoren ist so wichtig, dass wir dessen Resultat als ein eigenes Theorem aufschreiben.

**Satz 2.3.0.4. Linear unabhängige Spalten einer Matrix**

*Die Spaltenvektoren $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_k \in \mathbb{R}^n$ einer $n \times k$ Matrix $\mathbf{A}$ sind linear unabhängig genau dann , wenn der Rang der Matrix $k$ ist. Damit diese Bedingung erfüllt werden kann, muss $k \leq n$ gelten.*

Wir geben nun auch noch die wichtige Definition der Basis.

**Definition 2.3.0.5. Basis**

Ein linear unabhängiges Erzeugendensystem eines linearen Raumes $V$ heisst *Basis* in $V$.

**Beispiel 2.3.0.6 (Basen)**

1) Wir haben bereits im ersten Kapitel dieses Skripts die kanonische Basis / Standardbasis in $\mathbb{R}^n$ genannt:

$$\{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_n\}\,.$$

2) Die Monome $p_0, p_1, \ldots, p_{n-1}$ bilden eine Basis in $\mathcal{P}_n$.

3) Seien die drei Polynome $q_1, q_2, q_3 \in \mathcal{P}$ gegeben[2] als:

$$\begin{aligned} q_1(t) &= 1 + t^2 \\ q_2(t) &= 1 - t^2 \\ q_3(t) &= 1 + t^2 + t^4\,. \end{aligned}$$

Wir werden sehen, dass diese Polynome linear unabhängig sind und eine Basis im Raum der geraden Polynome vom Grad höchstens vier bilden.

Um die lineare Unabhängigkeit zu beweisen, beschreiben wir zuerst die $q_1, q_2, q_3$ mithilfe der Monome $p_0, p_2$ und $p_4$:

$$\begin{aligned} q_1 &= p_0 + p_2 \\ q_2 &= p_0 - p_2 \\ q_3 &= p_0 + p_2 + p_4\,. \end{aligned}$$

Wir überprüfen nun die Definition der linearen Unabhängigkeit. Seien $x_1, x_2, x_3 \in \mathbb{R}$ so gewählt, dass

$$x_1 q_1 + x_2 q_2 + x_3 q_3 = 0\,.$$

Wir ersetzen hier nun die $q_1, q_2, q_3$ durch ihre Beschreibungen mittels Monomen und formen dies dann um:

$$\begin{aligned} x_1(p_0 + p_2) + x_2(p_0 - p_2) + x_3(p_0 + p_2 + p_4) &= 0 \\ (x_1 + x_2 + x_3)p_0 + (x_1 - x_2 + x_3)p_2 + x_3 p_4 &= 0\,. \end{aligned}$$

[2] Beispiel 4.16 Seite 4-9 in Gut07.

Die Monome sind linear unabhängig, also müssen ihre Vorfaktoren alle Null sein. Wir erhalten ein lineares Gleichungssystem:

$$\begin{cases} x_1 + x_2 + x_3 &= 0 \\ x_1 - x_2 + x_3 &= 0 \\ x_3 &= 0 \,. \end{cases}$$

Die Systemmatrix ist

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -1 & 1 \\ 0 & 0 & 1 \end{bmatrix}.$$

Da diese Matrix vollen Rang hat, hat das homogene LGS, $\mathbf{Ax} = \mathbf{0}$ nur die triviale Lösung. Damit sind die Skalare $x_1,\ x_2$ und $x_3$ alle Null und somit sind $q_1,\ q_2$ und $q_3$ linear unabhängig.

Der Raum, der von $q_1,\ q_2$ und $q_3$ gebildet wird, wird auch von $p_0,\ p_2,\ p_4$ aufgespannt. Dies entspricht dem Raum der geraden Polynome vom Grad höchstens vier und wir notieren ihn mit $G_5$.

$$\mathrm{Span}\{q_1,\ q_2,\ q_3\} = \mathrm{Span}\{p_0,\ p_2,\ p_4\} = G_5 \,.$$

Die $\{p_0,\ p_2,\ p_4\}$ bilden hierbei eine Basis in diesem Raum und deswegen ist auch $\{q_1,\ q_2,\ q_3\}$ eine Basis von $G_5$. Ein linearer Raum kann also auch mehrere Basen haben.

⌟

**Definition 2.3.0.7. Dimension eines linearen Raumes**

Die Dimension eines linearen Raumes $V$ entspricht der Anzahl an Elementen in einer Basis von $V$.

Die Dimension eines linearen Raumes ist jedoch nur dann wohldefiniert, wenn alle Basen eines linearen Raumes gleich viele Elemente haben. Deswegen geben wir folgendes Theorem:

**Satz 2.3.0.8. Grösse der Basen**

*Verschiedene Basen für den linearen Raum $V$ bestehen aus gleich vielen Vektoren, deren Anzahl die Dimension des Raumes $V$ ist.*

*Beweis.* Seien $\{v_1, \ldots,\ v_\ell \in V\}$ und $\{w_1, \ldots,\ w_k \in V\}$ zwei Basen in $V$. Wir verwenden hier die Eigenschaften: $V = \mathrm{Span}\{v_1, \ldots,\ v_\ell\}$ und $w_1, \ldots,\ w_k \in V$ linear unabhängige Elemente. Es reicht nun aus, zu zeigen, dass $k \leq \ell$, da wir, um die andere Richtung zu zeigen, die beiden Mengen vertauschen können.

Für jedes $j = 1,\ 2, \ldots,\ k$ betrachten wir das Element $w_j \in V$. Da $V$ von $v_1, \ldots,\ v_\ell$ aufgespannt wird, können wir $w_j$ als Linearkombination von $v_1, \ldots,\ v_\ell$ schreiben:

$$w_j = \alpha_{1j} v_1 + \alpha_{2j} v_2 + \ldots + \alpha_{\ell j} v_\ell \,,$$

mit den $\alpha_{1j}, \ldots,\ \alpha_{\ell j}$ Skalaren aus $\mathbb{R}$. Wenn wir dies für jedes $w_j$ mit $j = 1, \ldots,\ k$ machen, erhalten wir eine $\ell \times k$ Matrix $\mathbf{A}$ mit den Einträgen $\alpha_{ij}$:

$$\mathbf{A} = \left[\alpha_{ij}\right]_{\substack{j=1,\,2,\,\ldots k, \\ i=1,\,2,\,\ldots,\,\ell}} .$$

Wir nutzen nun aus, dass die $w_1, \ldots,\ w_k$ linear unabhängig sind, also gilt, dass

$$x_1 w_1 + x_2 w_2 + \ldots + x_k w_k = 0 \quad \Longrightarrow \quad x_1 = x_2 = \ldots = x_k = 0 \,.$$

Wir schreiben den ersten Teil auch als Summe und dann ersetzen wir die $w_j$ durch ihre Linearkombination von $v_1, \ldots, v_\ell$:

$$\sum_{j=1}^{k} x_j w_j = \sum_{j=1}^{k} x_j \sum_{i=1}^{\ell} \alpha_{ij} v_i \,.$$

Wir vertauschen hier die beiden Summen miteinander und erhalten also

$$x_1 w_1 + x_2 w_2 + \ldots + x_k w_k = \sum_{i=1}^{\ell} \left( \sum_{j=1}^{k} \alpha_{ij} x_j \right) v_i \,.$$

Nun betrachten wir das homogene Gleichungssystem $\mathbf{Ax} = \mathbf{0}$; d. h. für alle $i = 1, 2, \ldots, \ell$ gilt:

$$\sum_{j=1}^{k} \alpha_{ij} x_j = 0 \,.$$

Wenn $\mathbf{x} \in \mathbb{R}^k$ eine Lösung dieses homogenen Gleichungssystems ist, dann ist jeder Koeffizient von $v_i$ in der oberen Darstellung von $\sum_{j=1}^{k} x_j w_j$ null, so dass

$$\sum_{j=1}^{k} x_j w_j = 0$$

gelten muss. Da aber alle $w_j$ linear unabhängig sind, müssen alle $x_j$ Null sein. Das bedeutet, dass das homogene System $\mathbf{Ax} = \mathbf{0}$ nur die triviale Lösung hat und dass die Matrix $\mathbf{A}$ Rang $k \leq \ell$ haben muss. Dies ist genau das, was wir zeigen wollten.

□

**Beispiel 2.3.0.9 (Dimensionen)**
Die Dimension des Raumes $\mathbb{R}^n$, welche als $\dim \mathbb{R}^n$ geschrieben wird, ist $n$.
Da $\dim \mathscr{P}_n = n$ sehen wir noch einen Grund, wieso die Notation vom Raum $\mathscr{P}_n$ sinnvoll ist.

Die Räume $\mathscr{P}$, $C^k$, $L^2$ sind nicht endlichdimensional.

⌟

Aus dem letzten Beweis ist auch der folgende Satz klar:

**Satz 2.3.0.10. Grösse von erzeugenden Systemen in einem endlichdimensionalen Raum**

*Sei $V$ ein linearer Raum mit Dimension $n$, dann:*

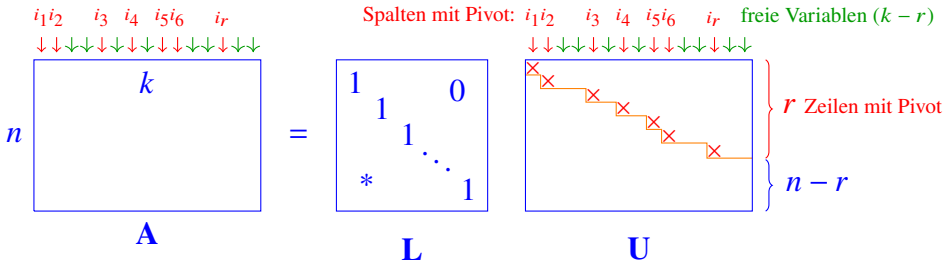
1) *sind mehr als $n$ Elemente von $V$ linear abhängig.*
2) *sind weniger als $n$ Elemente von $V$ nicht erzeugend.*
3) *sind $n$ Elemente von $V$ linear unabhängig genau dann, wenn sie auch erzeugend sind.*

## 2.4 Fundamentalsatz der linearen Algebra - Teil I

Wenn wir $k$ Vektoren $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_k \in \mathbb{R}^n$ haben, wie wählen wir daraus die maximale Anzahl linear unabhängiger Vektoren?

Diese Frage, auch wenn sie auf den ersten Blick einfach erscheinen mag, ist sehr wichtig. Zum Beispiel bei der Datenanalyse, wo man viele Messungen hat und dann herausfinden möchte, welche dieser Messungen einem etwas Neues sagt, und welche linear abhängig von anderen Messungen sind, man also ignorieren kann. Mathematisch formuliert: Wir haben eine Menge von Vektoren und wollen die maximale Anzahl linear unabhängiger Vektoren finden, denn der Aufspann dieser Vektoren ist der gleiche wie der ganzen Menge.

Um diese Frage zu beantworten, verwenden wir die LU-Zerlegung. Einfachheitshalber nehmen wir an, dass keine Zeilenvertauschungen nötig sind. Die Gauss-Elimination gibt uns die Faktoren $\mathbf{L}$ (untere Dreiecksmatrix mit Einsen auf der Hauptdiagonale) und $\mathbf{U}$ (obere Dreiecksmatrix) einer beliebigen Matrix $\mathbf{A}$, so dass $\mathbf{A} = \mathbf{LU}$ gilt:




Die obere Dreiecksmatrix $\mathbf{U}$ ist die Zeilenstufenform der ursprünglichen Matrix $\mathbf{A}$. Wir notieren die Indizes der Spalten, die ein Pivot haben, mit $i_1, i_2, \ldots, i_r$ (im Bild rot markiert). Wir zeigen nun, dass die entsprechenden Spalten $i_1, i_2, \ldots, i_r$ aus $\mathbf{A}$ linear unabhängig sind. Die LU-Zerlegung erlaubt uns also, linear unabhängige Vektoren aus einer Menge von Vektoren auszuwählen.

**Satz 2.4.0.1. Basis im Bild $\mathbf{A}$**

*Sei $\mathbf{A} = \mathbf{LU}$ die LU-Zerlegung der Matrix $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_k] \in \mathbb{R}^{n \times k}$. Wir notieren die Indizes der Spalten in $\mathbf{U}$, die ein Pivot haben, mit $i_1, i_2, \ldots, i_r$. Dann sind die dazugehörigen Spalten $\mathbf{a}_{i_1}, \mathbf{a}_{i_2}, \ldots, \mathbf{a}_{i_r}$ aus $\mathbf{A}$ linear unabhängig. Alle anderen Spalten von $\mathbf{A}$ lassen sich als lineare Kombinationen von $\mathbf{a}_{i_1}, \mathbf{a}_{i_2}, \ldots, \mathbf{a}_{i_r}$ schreiben. Somit bilden $\mathbf{a}_{i_1}, \mathbf{a}_{i_2}, \ldots, \mathbf{a}_{i_r}$ eine Basis im Bild $\mathbf{A}$.*

*Beweis.* Um die lineare Unabhängigkeit zu überprüfen, schauen wir uns zuerst den Zusammenhang zwischen den entsprechenden Spaltenvektoren aus den Matrizen $\mathbf{A}$ und $\mathbf{U}$, d. h. zwischen den $\mathbf{a}_{i_j}$ und den $\mathbf{u}_{i_j}$ für $j = 1, 2, \ldots, r$. Wir erinnern uns nun an die Begründung der Definition der Multiplikation der Matrizen 1.3: $\mathbf{LU} = \mathbf{A}$ ist eine Sammlung von linearen Gleichungssystemen mit der gleichen Systemmatrix $\mathbf{L}$, mit den Spalten von $\mathbf{A}$ als verschiedene rechte Seiten und mit den Spalten von $\mathbf{U}$ als entsprechende Lösungen. Für unsere ausgewählten Spalten mit den Indizes $i_1, i_2, \ldots, i_r$ bedeutet das:

$$\begin{aligned} \mathbf{a}_{i_1} &= \mathbf{L}\mathbf{u}_{i_1} \\ \mathbf{a}_{i_2} &= \mathbf{L}\mathbf{u}_{i_2} \\ &\ldots \\ \mathbf{a}_{i_r} &= \mathbf{L}\mathbf{u}_{i_r} . \end{aligned}$$

Wir betrachten eine Linearkombination davon und formen diese dann um:

$$\sum_{j=1}^{r} x_j \mathbf{a}_{i_j} = \sum_{j=1}^{r} x_j \mathbf{L}\mathbf{u}_{i_j} = \mathbf{L} \sum_{j=1}^{r} x_j \mathbf{u}_{i_j} .$$

Die Vektoren $\mathbf{a}_{i_j}$ sind linear unabhängig genau dann, wenn aus

$$\sum_{j=1}^{r} x_j \mathbf{a}_{i_j} = 0$$

folgt, dass alle $x_j$ Null sind. Da alle obigen Summen äquivalent sind, und die Matrix $\mathbf{L}$ invertierbar ist, können wir dies umschreiben zu

$$\sum_{j=1}^{r} x_j \mathbf{u}_{i_j} = 0\,.$$

Die Vektoren $\mathbf{u}_{i_1}, \mathbf{u}_{i_2}, \ldots, \mathbf{u}_{i_r}$ sind aber linear unabhängig. In der Tat, das letzte lineare Gleichungssystem mit Unbekannten $x_1, x_2, \ldots, x_r$ hat eine $r \times r$ obere Dreiecksmatrix mit den Spalten $\mathbf{u}_{i_1}, \mathbf{u}_{i_2}, \ldots, \mathbf{u}_{i_r}$ und lässt sich also einfach via Rückwärtssubstitution lösen:

$$x_r = x_{r-1} = \ldots = x_1 = 0\,.$$

Somit haben wir bewiesen, dass die Vektoren $\mathbf{a}_{i_1}, \mathbf{a}_{i_2}, \ldots, \mathbf{a}_{i_r}$ linear unabhängig sind.

Eine ähnliche Argumentation funktioniert auch für die restlichen Spalten von $\mathbf{A}$. Wir erinnern uns an die Bemerkung 1.2.0.12: Die zu den freien Variablen gehörenden Spalten von $\mathbf{U}$ sind lineare Kombinationen der $r$ Pivotspalten in $\mathbf{U}$. In der Tat, sei $u_{i_f}$ eine Spalte aus $\mathbf{U}$, die einer freien Variablen entspricht; betrachten wir die Matrix mit den Pivotspalten $a_{i_p}$ mit $i_p < i_f$ und die Spalte $a_{i_f}$, so ist der Rang dieser Matrix mit 1 weniger als der Anzahl seiner Spalten, laut Satz 2.3.0.4, sind somit diese Spalten linear abhängig. Das bedeutet: die Spalte $a_{i_f}$ ist eine Linearkombination der Spalten $a_{i_p}$ mit $i_p < i_f$. Das überträgt sich auf die Spalten von $A$, denn jede $a_j = \mathbf{L}u_j$ mit der $\mathbf{L}$, die untere Dreiecksmatrix mit 1-er auf der Diagonale ist. Darum lassen sich die zu freien Variablen gehörenden Spalten in $\mathbf{A}$ als lineare Kombinationen von den Spalten $\mathbf{a}_{i_1}, \mathbf{a}_{i_2}, \ldots, \mathbf{a}_{i_r}$ schreiben, die somit das Bild von $\mathbf{A}$ aufspannen:

$$\mathrm{Span}\{\mathbf{a}_{i_1}, \mathbf{a}_{i_2}, \ldots, \mathbf{a}_{i_r}\} = \mathrm{Bild}\,\mathbf{A}\,.$$

Da diese linear unabhängig sind, bilden sie eine Basis in $\mathrm{Bild}\,\mathbf{A}$.

□

**Bemerkung 2.4.0.2** ($\mathrm{Bild}\,\mathbf{A} \neq \mathrm{Bild}\,\mathbf{U}$)
Etwas Wichtiges, das man nicht vergessen sollte, ist: Obwohl beide Mengen von entsprechenden Spalten $\{\mathbf{a}_{i_1}, \mathbf{a}_{i_2}, \ldots, \mathbf{a}_{i_r}\}$ und $\{\mathbf{u}_{i_1}, \mathbf{u}_{i_2}, \ldots, \mathbf{u}_{i_r}\}$ linear unabhängig sind, und beide Mengen Basen in je einem linearen Unterraum von $\mathbb{R}^n$ sind, sind diese beiden Unterräume nicht zwingend gleich! Es gilt also:

$$\mathrm{Bild}\,\mathbf{A} \neq \mathrm{Bild}\,\mathbf{U}\,.$$

⌟

**Bemerkung 2.4.0.3** ($\dim \mathrm{Bild}\,\mathbf{A} = \dim \mathrm{Bild}\,\mathbf{U} = \mathrm{Rang}\,\mathbf{A}$)
Andererseits haben die beiden Räume dieselbe Dimension:

$$\dim \mathrm{Bild}\,\mathbf{A} = \dim \mathrm{Bild}\,\mathbf{U} = r = \mathrm{Rang}\mathbf{A}\,.$$

⌟

**Satz 2.4.0.4. Dimension von $\mathrm{Kern}\,\mathbf{A}$**

*Sei die $n \times k$ Matrix $\mathbf{A}$ von Rang $r$. Dann:*

$$\dim(\mathrm{Kern}\,\mathbf{A}) = k - r\,.$$

*Beweis.* Sei $\mathbf{A}$ eine $n \times k$ Matrix mit Rang $r$. Wir verwenden hier die $k-r$ freien Variablen $x_{f_1}, \ldots, x_{f_{k-r}}$ von $\mathbf{A}$. Mit diesen freien Variablen bilden wir gleich wie in der Bemerkung 2.2.0.14 verschiedene partikuläre Lösungen des homogenen Systems $\mathbf{Ax} = \mathbf{0}$. Diese partikulären Lösungen spannen also den Kern von $\mathbf{A}$ auf. Wir zeigen nun, dass diese $k-r$ partikulären Lösungen linear unabhängig sind, sie also eine Basis bilden und somit haben wir $\dim(\operatorname{Kern}\mathbf{A}) = k-r$.

Beginnen wir also mit den freien Variablen. Jedes Element im $\operatorname{Kern}\mathbf{A}$ ist eine Lösung des homogenen Systems $\mathbf{Ax} = \mathbf{0}$. Da die Matrix $\mathbf{A}$ Rang $r$ hat, gibt es in der Gauss-Elimination $r$ Pivotspalten und es bleiben $k-r$ freie Variablen, die wir mit den Indizes $f_1, f_2, \ldots, f_{k-r}$ nummerieren:

$$x_{f_1}, x_{f_2}, \ldots, x_{f_{k-r}}\,.$$

Aus diesen freien Variablen können wir, gleich wie in der Bemerkung 2.2.0.14, partikuläre Lösungen erhalten, indem wir eine dieser Variablen auf $1$ setzen und alle anderen auf $0$:

$$\begin{aligned}
x_{f_1} = 1,\, x_{f_2} = 0,\, x_{f_3} = 0, \ldots, x_{f_{k-r-1}} = 0,\, x_{f_{k-r}} = 0 &\implies \mathbf{x}^1 \in \mathbb{R}^k \quad \text{Lösung von } \mathbf{Ax} = \mathbf{0}\,,\\
x_{f_1} = 0,\, x_{f_2} = 1,\, x_{f_3} = 0, \ldots, x_{f_{k-r-1}} = 0,\, x_{f_{k-r}} = 0 &\implies \mathbf{x}^2 \in \mathbb{R}^k \quad \text{Lösung von } \mathbf{Ax} = \mathbf{0}\,,\\
&\vdots\\
x_{f_1} = 0,\, x_{f_2} = 0,\, x_{f_3} = 0, \ldots, x_{f_{k-r-1}} = 1,\, x_{f_{k-r}} = 0 &\implies \mathbf{x}^{k-r} \in \mathbb{R}^k \quad \text{Lösung von } \mathbf{Ax} = \mathbf{0}\,.
\end{aligned}$$

Die Lösung des homogenen Systems $\mathbf{Ax} = \mathbf{0}$ lässt sich eindeutig schreiben, sobald wir Werte für die freien Variablen gewählt haben. Somit spannen die Vektoren $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^{k-r}$ den Kern von $\mathbf{A}$ auf. Wir haben jetzt diese partikulären Lösungen mit einem oberen Index notiert, weil wir den Platz für den unteren Index gleich brauchen werden, wenn wir die Komponenten dieser Vektoren schreiben wollen.

Um zu beweisen, dass $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^{k-r}$ eine Basis bilden, müssen wir noch zeigen, dass sie linear unabhängig sind.

Seien $\alpha_1, \alpha_2, \ldots, \alpha_{k-r}$ Skalare, so dass

$$\alpha_1\mathbf{x}^1 + \alpha_2\mathbf{x}^2 + \ldots + \alpha_{k-r}\mathbf{x}^{k-r} = \mathbf{0}\,.$$

Wir zeigen jetzt, dass $\alpha_1 = \alpha_2 = \ldots = \alpha_{k-r} = 0$. Wir betrachten die Komponente $f_i$ mit $i = 1, 2, \ldots, k-r$ der Linearkombination. Diese Komponente ist:

$$\alpha_1 x^1_{f_i} + \alpha_2 x^2_{f_i} + \ldots + \alpha_{k-r} x^{k-r}_{f_i} = 0\,.$$

Jede davon resultiert in einem Koeffizienten $\alpha_i = 0$:

$$\begin{aligned}
i = 1 \quad &: \quad \alpha_1 1 + \alpha_2 0 + \ldots + \alpha_{k-r} 0 \quad &\implies \alpha_1 = 0\\
i = 2 \quad &: \quad \alpha_1 0 + \alpha_2 1 + \ldots + \alpha_{k-r} 0 \quad &\implies \alpha_2 = 0\\
&\vdots\\
i = k-r \quad &: \quad \alpha_1 0 + \alpha_2 0 + \ldots + \alpha_{k-r} 1 \quad &\implies \alpha_{k-r} = 0\,.
\end{aligned}$$

Wir haben also gezeigt, dass die partikulären Lösungen $\mathbf{x}^1, \ldots, \mathbf{x}^{k-r}$ linear unabhängig sind und den $\operatorname{Kern}\mathbf{A}$ aufspannen, d. h. eine Basis bilden. Nach der Definition 2.3 ist dann die Dimension des Kerns von $\mathbf{A}$ gleich der Grösse der Basis, also $k-r$.

□

Der Rang der Matrix $\mathbf{A}$ gibt also die Anzahl der linear unabhängigen Spalten von $\mathbf{A}$ an. Darüber hinaus legt er auch die Anzahl der abhängigen Spalten fest. In Anbetracht der Bemerkung 1.2.0.12 entscheidet der Rang

$r$ von $\mathbf{A}$ auch über die Anzahl der linear abhängigen Gleichungen des entsprechenden homogenen Systems: Anzahl Zeilen minus Rang. Wenn wir die Matrix $\mathbf{A}$ transponieren, sind die Rollen von $n$ und $k$ vertauscht und die Spalten von $\mathbf{A}^\mathsf{T}$ sind die Zeilen von $\mathbf{A}$. Darum ist das folgende Ergebnis zwar nicht erstaunlich, aber beachtlich.

**Satz 2.4.0.5. Dimension von $\operatorname{Kern}\mathbf{A}^\mathsf{T}$**

*Sei die $n \times k$ Matrix $\mathbf{A}$ von Rang $r$. Dann hat die $k \times n$ Matrix $\mathbf{A}^\mathsf{T}$ auch Rang $r$ und:*

$$\dim\left(\operatorname{Kern}\mathbf{A}^\mathsf{T}\right) = n - r\,, \quad \dim\left(\operatorname{Bild}\mathbf{A}^\mathsf{T}\right) = r\,.$$

*Beweis.* Aus der Perspektive der letzten zwei Sätze lohnt sich ein Blick auf die Bemerkung 1.2.0.12. Dort sahen wir: Wenn man die Gleichungen des homogenen Systems mit der Matrix $\mathbf{A}$ in der Reihenfolge hat, die eine Gauss-Elimination ohne Permutationen erlaubt, dann sind die letzten $n-r$ Gleichungen lineare Kombinationen der ersten $r$ Gleichungen. In anderen Worten: die letzten $n-r$ Zeilen lassen sich als Linearkombination der ersten $r$ Zeilen schreiben. Die Zeilen der Matrix $\mathbf{A}^\mathsf{T}$ sind aber die Spalten der Matrix $\mathbf{A}$. D. h., die letzten $n-r$ Spaltenvektoren der Matrix $\mathbf{A}^\mathsf{T}$ lassen sich als lineare Kombinationen der ersten $r$ Spaltenvektoren von $\mathbf{A}^\mathsf{T}$ schreiben. Diese spannen also das $\operatorname{Bild}\mathbf{A}^\mathsf{T}$ auf. Die Gauss-Elimination für die Matrix $\mathbf{A}^\mathsf{T}$ liefert also $n-r$ freie Variablen, $r$ Pivoten und somit $k-r$ Kompatibilitätsbedingungen. Wie im vorigen Beweis bedeutet dies:

$$\dim(\operatorname{Kern}\mathbf{A}^\mathsf{T}) = n - r\,, \quad \dim(\operatorname{Bild}\mathbf{A}^\mathsf{T}) = r\,.$$

□

Den Fundamentalsatz der linearen Algebra kann man mit einem Bild visualisieren. Für eine gegebene Matrix $\mathbf{A} \in \mathbb{R}^{n\times k}$ betrachten wir zwei Abbildungen zwischen den Räumen $\mathbb{R}^k$ und $\mathbb{R}^n$: die Abbildung $\mathscr{A}$ von $\mathbb{R}^k$ nach $\mathbb{R}^n$ und die Abbildung $\mathscr{A}^\mathsf{T}$ von $\mathbb{R}^n$ nach $\mathbb{R}^k$:

$$\begin{array}{rccl} \mathscr{A}: & \mathbb{R}^k & \to & \mathbb{R}^n \quad \mathscr{A}(x) = \mathbf{A}\mathbf{x} \\ & \mathbf{x} & \mapsto & \mathbf{A}\mathbf{x} \end{array}, \qquad \begin{array}{rccl} \mathscr{A}^\mathsf{T}: & \mathbb{R}^n & \to & \mathbb{R}^k \quad \mathscr{A}^\mathsf{T}(x) = \mathbf{A}^\mathsf{T}\mathbf{x} \\ & \mathbf{x} & \mapsto & \mathbf{A}^\mathsf{T}\mathbf{x} \end{array}.$$

Von diesen beiden Abbildungen können wir Kern und Bild betrachten. Die Dimensionen dieser insgesamt vier Unterräume haben wir gerade vorher berechnet:

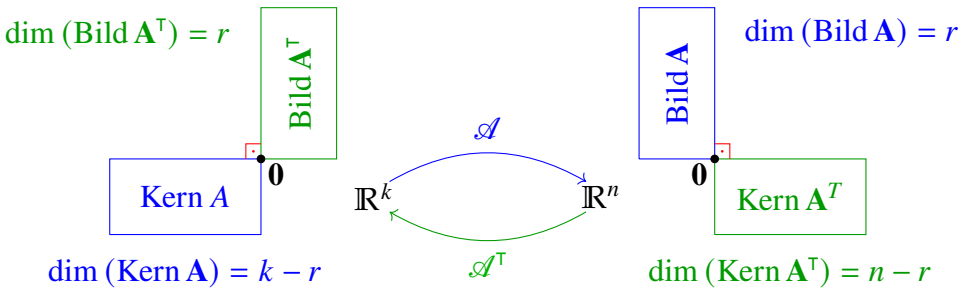
$$\begin{aligned} \dim(\operatorname{Kern}\mathbf{A}) &= k - r, \ \dim(\operatorname{Bild}\mathbf{A}) = r, \\ \dim(\operatorname{Kern}\mathbf{A}^\mathsf{T}) &= n - r, \ \dim(\operatorname{Bild}\mathbf{A}^\mathsf{T}) = r. \end{aligned}$$

Wir sehen, dass die Summe der Dimensionen von den beiden Unterräumen von $\mathbb{R}^k$ genau $k$, die Dimension des Raumes, ergeben. Das Gleiche gilt natürlich auch für $\mathbb{R}^n$.

Eine zusätzliche wichtige Eigenschaft ist, dass die entsprechenden Unterräume orthogonal aufeinander stehen.

**Definition 2.4.0.6. Orthogonalität von Unterräumen**

Seien $U$ und $V$ Unterräume von $\mathbb{R}^k$. Wir sagen, dass $U$ orthogonal auf $V$ steht, notiert mit $U \perp V$, falls beliebige Vektoren $\mathbf{u}$ aus $U$ und $\mathbf{v}$ aus $V$ orthogonal sind ($\mathbf{u} \perp \mathbf{v}$) d. h. mit dem Skalarprodukt in $\mathbb{R}^k$ gilt: $\langle \mathbf{u}, \mathbf{v}\rangle = \mathbf{u}^\mathsf{T}\mathbf{v} = 0$.




Was wir hier gesehen haben, ist der erste Teil des Fundamentalsatzes der linearen Algebra. Wir fassen diese Ergebnisse noch in einem Theorem zusammen.

**Satz 2.4.0.7. Fundamentalsatz der linearen Algebra - Teil I**

*Sei* $\mathbf{A}$ *eine* $n \times k$ *Matrix vom Rang* $r$. *Dann hat die* $k \times n$ *Matrix* $\mathbf{A}^\mathsf{T}$ *auch Rang* $r$ *und:*

1) *die Dimensionen der Unterräume sind:*

$$\dim(\text{Bild}\,\mathbf{A}) = r,\ \dim(\text{Kern}\,\mathbf{A}) = k - r,\ \dim(\text{Bild}\,\mathbf{A}^\mathsf{T}) = r,\ \dim(\text{Kern}\,\mathbf{A}^\mathsf{T}) = n - r,$$

2) *es gilt der Dimensionssatz:*

$$\dim(\text{Kern}\,\mathbf{A}) + \dim(\text{Bild}\,\mathbf{A}^\mathsf{T}) = k,\ \dim(\text{Kern}\,\mathbf{A}^\mathsf{T}) + \dim(\text{Bild}\,\mathbf{A}) = n,$$

3) *die Unterräume stehen orthogonal aufeinander:*

$$\text{Kern}\,\mathbf{A} \perp \text{Bild}\,\mathbf{A}^\mathsf{T},\ \text{Bild}\,\mathbf{A} \perp \text{Kern}\,\mathbf{A}^\mathsf{T}.$$

*Beweis.*

Die ersten zwei Aussagen des Fundamentalsatzes der linearen Algebra haben wir bereits bewiesen, die dritte aber noch nicht. Dies werden wir nun tun.

Wir müssen zeigen, dass Kern $\mathbf{A}$ orthogonal auf Bild $\mathbf{A}^\mathsf{T}$ steht. Hierfür nehmen wir ein beliebiges $\mathbf{x} \in$ Kern $\mathbf{A}$ und ein beliebiges $\mathbf{y} \in$ Bild $\mathbf{A}^\mathsf{T}$ und zeigen, dass diese orthogonal aufeinander stehen.

Da $\mathbf{x}$ aus dem Kern von $\mathbf{A}$ und $\mathbf{y}$ aus dem Bild von $\mathbf{A}^\mathsf{T}$ kommt, gilt:

$$\mathbf{A}\mathbf{x} = \mathbf{0}$$
$$\mathbf{y} = \mathbf{A}^\mathsf{T}\mathbf{b}\,.$$

Wenn wir die Orthogonalität zwischen $\mathbf{x}$ und $\mathbf{y}$ zeigen wollen, berechnen wir das Skalarprodukt zwischen ihnen und überprüfen, ob es Null ist:

$$\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^\mathsf{T}\mathbf{y} = \mathbf{x}^\mathsf{T}(\mathbf{A}^\mathsf{T}\mathbf{b}) = (\mathbf{x}^\mathsf{T}\mathbf{A}^\mathsf{T})\mathbf{b} = (\mathbf{A}\mathbf{x})^\mathsf{T}\mathbf{b} = \mathbf{0}^\mathsf{T}\mathbf{b} = 0\,.$$

Da sowohl $\mathbf{x}$ als auch $\mathbf{y}$ beliebig sind, haben wir gezeigt, dass Kern $\mathbf{A}$ orthogonal auf Bild $\mathbf{A}^\mathsf{T}$ steht.

Der Beweis von Bild $\mathbf{A} \perp$ Kern $\mathbf{A}^\mathsf{T}$ funktioniert analog zu diesem hier und wir werden ihn deswegen nicht explizit machen.

□

**Bemerkung 2.4.0.8 (Orthogonale Zerlegung nach dem Fundamentalsatz der linearen Algebra - I)**
Sei $\mathbf{A}$ eine $n \times k$ Matrix vom Rang $k$.
Jedes $\mathbf{x} \in \mathbb{R}^k$ kann eindeutig in zwei orthogonale Komponenten zerlegt werden, eine in $\operatorname{Kern} \mathbf{A}$ und eine in $\operatorname{Bild} \mathbf{A}^\top$:

$$\mathbf{x} = \mathbf{u} + \mathbf{v}, \quad \mathbf{u} \in \operatorname{Kern} \mathbf{A}, \quad \mathbf{v} \in \operatorname{Bild} \mathbf{A}^\top, \quad \text{mit } \mathbf{u} \perp \mathbf{v}.$$

Jedes $\mathbf{y} \in \mathbb{R}^n$ kann eindeutig in zwei orthogonale Komponenten zerlegt werden, eine in $\operatorname{Kern} \mathbf{A}^\top$ und eine in $\operatorname{Bild} \mathbf{A}$:

$$\mathbf{y} = \mathbf{z} + \mathbf{w}, \quad \mathbf{z} \in \operatorname{Kern} \mathbf{A}^\top, \quad \mathbf{w} \in \operatorname{Bild} \mathbf{A}, \quad \text{mit } \mathbf{z} \perp \mathbf{w}.$$

⌟

Der Name *Fundamentalsatz der linearen Algebra* wurde 1993 mit dem Essay von Gilbert Strang *The Fundamental Theorem of Linear Algebra* aus *The American Mathematical Monthly*, Vol. 100, No. 9. (Nov., 1993), pp. 848-855 weit verbreitet[3]. Den zweiten Teil des Fundamentalsatzes werden wir erst nach der Besprechung der Singulärwertzerlegung sehen. Schon jetzt verraten wir aber: Die hier gesehenen Basen der vier Unterräume $\operatorname{Kern} \mathbf{A}$, $\operatorname{Bild} \mathbf{A}$, $\operatorname{Kern} \mathbf{A}^\top$ und $\operatorname{Bild} \mathbf{A}^\top$ sind nicht bequem oder gar ungeeignet für numerische Berechnungen. Die Singulärwertzerlegung wird uns orthogonale Basen in diesen Räumen geben, die von grosser Bedeutung in den Anwendungen sind.

[3]Der Begriff *Fundamentalsatz der linearen Algebra* muss auch zuvor verwendet worden sein, da ich ihn bereits im Herbst 1992 in meinem ersten Semester des Mathematik-Studiums von meinem damaligen Lineare Algebra Professor Dorian Spulber hörte.

## 2.5 Koordinaten und Basiswahl

**Satz 2.5.0.1. Eindeutige Darstellung in gegebener Basis**

*Sei in einem endlichdimensionalen linearen Raum $V$ mit Dimension $n$ eine Basis*

$$v_1, v_2, \ldots, v_n .$$

*Jedes Element $x \in V$ lässt sich als eine Linearkombination dieser Basiselemente eindeutig darstellen.*

*Beweis.* Seien zwei lineare Kombinationen von $x$ durch Terme der Elemente der Basis dargestellt:

$$x = x_1 v_1 + x_2 v_2 + \ldots + x_n v_n$$
$$x = y_1 v_1 + y_2 v_2 + \ldots + y_n v_n .$$

Wenn wir diese beiden Gleichungen voneinander subtrahieren, erhalten wir:

$$0 = (x_1 - y_1) v_1 + (x_2 - y_2) v_2 + \ldots + (x_n - y_n) v_n .$$

Da aber die $v_1, \ldots, v_n$ eine Basis bilden, sind sie linear unabhängig. So müssen alle Koeffizienten $x_i - y_i$ für $i = 1, \ldots, n$ Null sein und wir haben

$$x_1 = y_1$$
$$x_2 = y_2$$
$$\ldots$$
$$x_n = y_n .$$

Die beiden Linearkombinationen von $x$ sind also gleich und wir haben die Eindeutigkeit bewiesen. □

Diese Eindeutigkeit ermöglicht es uns, jeden linearen Raum der Dimension $n$ mit dem Raum $\mathbb{R}^n$ bzw. $\mathbb{C}^n$ zu identifizieren:

Sei $V$ ein linearer Raum der Dimension $n$ und mit einer Basis $\mathscr{B} = \{v_1, \ldots, v_n\}$. Wir können jedes Element aus $V$ eindeutig als eine Linearkombination von Elementen aus der Basis $\mathscr{B}$ darstellen. Wenn wir die Koeffizienten der Linearkombination als einen Vektor von $\mathbb{R}^n$ auffassen, haben wir eine eindeutige Repräsentation als Vektor in $\mathbb{R}^n$.

**Definition 2.5.0.2. Koordinatenabbildung und Koordinaten**

Sei $V$ ein linearer Raum von Dimension $n$ und mit einer Basis $\mathscr{B} = \{v_1, \ldots, v_n\}$.

Die Abbildung $\mathscr{K}_{\mathscr{B}}$

$$\begin{aligned} \mathscr{K}_{\mathscr{B}} : V &\to \mathbb{R}^n \\ x = x_1 v_1 + \ldots + x_n v_n &\mapsto \mathscr{K}_{\mathscr{B}} := \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \end{aligned}$$

nennen wir Koordinatenabbildung in der Basis $\mathscr{B}$.

Die Komponenten $x_1, \ldots, x_n$ des bei der Koordinatenabbildung eines Elementes $x$ entstehenden Vektors nennen wir Koordinaten von $x$ in der Basis $\mathscr{B}$.

**Beispiel 2.5.0.3 (Koordinaten in der Standardbasis)**
Sei $V = \mathbb{R}^3$ mit der Standardbasis $\mathscr{B} = \{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$. In diesem linearen Raum können wir die Koordinaten

sehr einfach ablesen, denn es sind einfach die Komponenten des Vektors:

$$\mathbf{x} \in V \quad \Longrightarrow \quad \mathbf{x} = x_1\mathbf{e}_1 + x_2\mathbf{e}_2 + x_3\mathbf{e}_3 = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}.$$

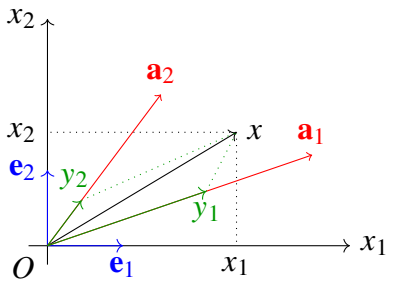
Die Koordinaten sind also $x_1,\ x_2,\ x_3$ und die Koordinatenabbildung bildet jedes Element auf sich selbst ab.

Wir können aber auch eine andere Basis $\mathscr{B}' = \{\mathbf{a}_1,\ \mathbf{a}_2,\ \mathbf{a}_3\}$ in $\mathbb{R}^3$ wählen. Ein Element $\mathbf{x}$ wird nun andere Koordinaten haben:

$$\mathbf{x} = y_1\mathbf{a}_1 + y_2\mathbf{a}_2 + y_3\mathbf{a}_3\,.$$

Die $y_1,\ y_2,\ y_3$ sind nun die Koordinaten von $\mathbf{x}$ in der neuen Basis.

Dies können wir in zwei Dimensionen darstellen. Einmal haben wir die Standardbasis $\mathscr{B} = \{\mathbf{e}_1,\ \mathbf{e}_2\}$ und ein anderes Mal die Basis $\mathscr{B}' = \{\mathbf{a}_1,\ \mathbf{a}_2\}$. In der Standardbasis $\mathscr{B}$ hat ein Vektor $\mathbf{x}$ die Koordinaten $x_1,\ x_2$ und in der Basis $\mathscr{B}'$ die Koordinaten $y_1,\ y_2$.




⌟

Die Koordinatenabbildung ist aber auch umkehrbar, denn wir können auch vom Koordinatenvektor in $\mathbb{R}^n$ zum Element in $V$ gehen.

$$\begin{array}{ccc} V & \xrightarrow{\Bbbk_{\mathscr{B}}} & \mathbb{R}^n \\ x & \longmapsto & \Bbbk_{\mathscr{B}}(x) = \mathbf{x} \end{array} \qquad \begin{array}{ccc} V & \xleftarrow{\Bbbk_{\mathscr{B}}^{-1}} & \mathbb{R}^n \\ x & \longleftarrow & \mathbf{x} \end{array} \qquad x = \Bbbk_{\mathscr{B}}^{-1}(\mathbf{x}) = x_1v_1 + \ldots + x_nv_n$$

Dies funktioniert, da die Koordinaten eindeutig sind, sobald wir eine Basis gewählt haben. In diesem Sinne können wir $V$ mit $\mathbb{R}^n$ identifizieren.

**Beispiel 2.5.0.4 (Koordinaten in der Basis von Monomen)**
Sei $\mathscr{P}_n$ der Raum der Polynome vom Grad höchstens $n-1$ und sei als Basis $\mathscr{B}$ die Monome $p_0, \ldots,\ p_{n-1}$ gewählt.

Dann können wir von einem $p \in \mathscr{P}_4$, definiert als

$$p(t) = 12 + 13t + 9t^2 + 4t^3 \tag{2.5.0.5}$$

, die Koordinaten und somit den Koordinatenvektor in der Basis der Monome $\mathscr{B}$ sofort schreiben als:

$$\mathbf{p} = \begin{bmatrix} 12 \\ 13 \\ 9 \\ 4 \end{bmatrix}.$$

⌟

## 2.6 Basiswechsel und Koordinatentransformation

Wie realisieren wir einen Wechsel von einer Basis in eine andere?

Sei $V$ ein linearer Raum mit zwei Basen, einer alten Basis $\mathscr{B}$ und einer neuen Basis $\tilde{\mathscr{B}}$:

$$\mathscr{B} = \{b_1, b_2, \ldots, b_n\}$$
$$\tilde{\mathscr{B}} = \{\tilde{b}_1, \tilde{b}_2, \ldots, \tilde{b}_n\}\,.$$

Sei $v$ ein Element von $V$. In der alten Basis lässt es sich schreiben als

$$v = x_1 b_1 + x_2 b_2 + \ldots + x_n b_n \quad \text{mit } x_1, \ldots, x_n \in \mathbb{R}\,.$$

$v$ hat also die Koordinaten $x_1, \ldots, x_n$ in der Basis $\mathscr{B}$. Wir wollen nun die Koordinaten von $v$ in der neuen Basis $\tilde{\mathscr{B}}$ finden und $v$ darstellen als:

$$v = \tilde{x}_1 \tilde{b}_1 + \tilde{x}_2 \tilde{b}_2 + \ldots + \tilde{x}_n \tilde{b}_n\,.$$

Jedes $\tilde{b}_j$ ist in $V$ und wir können es deswegen als eine Linearkombination von Elementen aus der Basis $\mathscr{B}$ darstellen:

$$\tilde{b}_j = c_{1j} b_1 + c_{2j} b_2 + \ldots + c_{nj} b_n\,.$$

Alle diese Koeffizienten $c_{ij}$ können wir in eine quadratische $n \times n$ Matrix $\mathbf{C}$ zusammenfassen:

$$\mathbf{C} = \left[c_{ij}\right]_{i,j=1,2,\ldots n}\,.$$

Hier ist zu beachten, dass die Koeffizienten der Darstellung jedes Vektors aus der neuen Basis in der alten Basis als **Spalten** verwendet werden. Somit können wir ein Element in der neuen Basis in Abhängigkeit der alten Basis umschreiben:

$$\sum_{j=1}^{n} \tilde{x}_j \tilde{b}_j = \sum_{j=1}^{n} \tilde{x}_j \sum_{i=1}^{n} c_{ij} b_i = \sum_{i=1}^{n} \left( \sum_{j=1}^{n} c_{ij} \tilde{x}_j \right) b_i = \sum_{i=1}^{n} (\mathbf{C}\tilde{\mathbf{x}})_i b_i\,,$$

wobei $(\mathbf{C}\tilde{\mathbf{x}})_i$ die $i$-te Komponente der Vektor-Matrix-Multiplikation zwischen $\mathbf{C}$ und $\mathbf{x}$ ist.

$v$ hat aber auch eine Darstellung in der alten Basis

$$v = \sum_{i=1}^{n} x_i b_i\,.$$

Wir kombinieren diese beiden Darstellungen und da die Koordinatendarstellung eindeutig ist, erhalten wir:

$$(\mathbf{C}\tilde{\mathbf{x}})_i = x_i \quad \text{für } i = 1, \ldots, n\,.$$

Somit realisiert die Matrix $\mathbf{C}$ die direkte Transformation der Koordinaten bei einem Basiswechsel:

$$\mathbf{C}\tilde{\mathbf{x}} = \mathbf{x}$$
$$\tilde{\mathbf{x}} = \mathbf{C}^{-1}\mathbf{x}\,.$$

In der Realität würden wir nicht die Inverse berechnen, sondern nur das lineare Gleichungssystem $\mathbf{C}\tilde{\mathbf{x}} = \mathbf{x}$ lösen, was deutlich günstiger ist. Wir sehen also, dass der Koordinatenwechsel sehr einfach ist, sobald wir die Matrix $\mathbf{C}$ gefunden haben.

**Beispiel 2.6.0.1 (Basiswechsel)**
Sei $G_5$ der Raum der geraden Polynome vom Grad höchstens 4. In diesem linearen Raum haben wir im Beispiel 2.3.0.6 zwei Basen gesehen:

$$\mathscr{B} = \{p_0,\ p_2,\ p_4\}$$
$$\tilde{\mathscr{B}} = \{q_1,\ q_2,\ q_3\}\,.$$

In diesem Beispiel waren $p_0, p_2, p_4$ die ersten Monome geraden Grades und die $q_1,\ q_2,\ q_3$ waren:

$$q_1(t) = 1 + t^2$$
$$q_2(t) = 1 - t^2$$
$$q_3(t) = 1 + t^2 + t^4\,.$$

Wir wählen nun ein beliebiges $g \in G_5$ und wir schreiben es als Linearkombination von Elementen aus der Basis $\mathscr{B}$ mit den Koeffizienten $x_1,\ x_2,\ x_3$ schreiben:

$$g = x_1 p_0 + x_2 p_2 + x_3 p_4\,.$$

Wir identifizieren also $g$ mit einem Koordinatenvektor $\mathbf{x}$ in der Basis $\mathscr{B}$. Dieser Koordinatenvektor $\mathbf{x}$ hat die Koordinaten $x_1,\ x_2,\ x_3$:

$$g \mapsto \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \mathbf{x}\,.$$

Wir möchten nun aber das Element $g$ in der neuen Basis darstellen, das heisst, wir suchen die Koordinaten $\tilde{x}_1,\ \tilde{x}_2,\ \tilde{x}_3$ so dass $g = \tilde{x}_1 q_1 + \tilde{x}_2 q_2 + \tilde{x}_3 q_3$.

Dies machen wir in drei Schritten: Zuerst schreiben wir die Elemente der neuen Basis als Linearkombination von Elementen der alten Basis, dann führen wir die Koeffizienten dieser Linearkombinationen in einer Matrix $\mathbf{C}$ zusammen und zum Schluss lösen wir das lineare Gleichungssystem $\mathbf{C}\tilde{\mathbf{x}} = \mathbf{x}$.

**1)** Wir schreiben jedes Element aus der neuen Basis $\tilde{\mathscr{B}}$ als eine Linearkombination von Elementen aus der alten Basis $\mathscr{B}$:

$$q_1 = 1p_0 + 1p_2 + 0p_4 \implies \text{Spalte} \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}.$$
$$q_2 = 1p_0 - 1p_2 + 0p_4 \implies \text{Spalte} \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix}.$$
$$q_3 = 1p_0 + 1p_2 + 1p_4 \implies \text{Spalte} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}.$$

**2)** Wir nehmen die Vektoren, die wir im Schritt 1) erhalten haben und schreiben diese **spaltenweise** in einer Matrix:

$$\mathbf{C} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -1 & 1 \\ 0 & 0 & 1 \end{bmatrix}.$$

**3)** Wir benutzen diese Matrix um das lineare Gleichungssystem $\mathbf{C}\tilde{\mathbf{x}} = \mathbf{x}$ nach $\tilde{\mathbf{x}}$ zu lösen:

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & -1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \tilde{x}_1 \\ \tilde{x}_2 \\ \tilde{x}_3 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad \Longrightarrow \quad \begin{cases} \tilde{x}_1 &= \frac{1}{2}x_1 + \frac{1}{2}x_2 - x_3 \\ \tilde{x}_2 &= \frac{1}{2}x_1 - \frac{1}{2}x_2 \\ \tilde{x}_3 &= x_3 \end{cases} .$$

⌟

# Kapitel 3

# Lineare Abbildungen

## Inhalt


**3.1 Grundlagen . . . . . 106**
**3.2 Abbildungsmatrix . . . . . 108**
**3.3 Abbildungsmatrix bei Koordinatentransformationen . . . . . 114**
**3.4 Anwendung: Referenztransformationen . . . . . 118**
3.4.1 Problemstellung . . . . . 118
3.4.2 Beispiel . . . . . 119
3.4.3 Erweiterte Matrix . . . . . 119


## 3.1 Grundlagen

**Definition 3.1.0.1. Lineare Abbildung**

Seien $X$, $Y$ lineare Räume und sei eine Funktion $\mathcal{F} : X \to Y$ gegeben.
Diese Funktion heisst *linear* falls:

1. $\mathcal{F}(x_1 + x_2) = \mathcal{F}(x_1) + \mathcal{F}(x_2)$ für alle $x_1, x_2 \in X$.
2. $\mathcal{F}(\alpha x) = \alpha \mathcal{F}(x)$ für alle $x \in X$ und $\alpha$ ein Skalar.

Alternative Namen sind: lineare Abbildung, lineare Funktion oder linearer Operator.
Falls $Y = \mathbb{R}$ oder $Y = \mathbb{C}$, dann bezeichnet man $\mathcal{F}$ als ein Funktional.
Wenn eine Abbildung linear ist, dann werden die Klammern bei der Notation oft weggelassen $\mathcal{F}(x) = \mathcal{F}x$.

**Beispiel 3.1.0.2 (Matrixabbildung)**
Die Abbildung $\mathcal{A}$ definiert als

$\mathcal{A} : \mathbb{R}^n \to \mathbb{R}^m$ mit $\mathcal{A}(\mathbf{x}) = \mathbf{Ax}$,

mit $\mathbf{A}$ eine $m \times n$ Matrix, ist eine lineare Abbildung, die sogenannte Matrixabbildung.

⌟

**Beispiel 3.1.0.3 (Koordinatenabbildung)** Sei $V$ ein linearer Raum der Dimension $n$ mit einer Basis $\mathcal{B} = \{v_1, \ldots, v_n\}$.

Die Koordinatenabbildung 2.5

$$\begin{array}{rcl} \mathcal{K}_{\mathcal{B}} : V & \to & \mathbb{R}^n \\ x = x_1 v_1 + \ldots + x_n v_n & \mapsto & \mathcal{K}_{\mathcal{B}} := \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \end{array}$$

ist linear. ⌟

**Beispiel 3.1.0.4 (Ableitungsoperator)**
Die Ableitung ist eine lineare Abbildung. Wir notieren den Ableitungsoperator $\mathcal{D}$:

$$\begin{array}{rcl} \mathcal{D} : C^1[a,b] & \to & C^0[a,b] \\ f & \mapsto & \mathcal{D}f \end{array}$$

$\mathcal{D}f$ ist eine Funktion definiert als

$\mathcal{D}f : [a,b] \to \mathbb{R}$ mit $(\mathcal{D}f)(t) = f'(t)$. ⌟

**Beispiel 3.1.0.5 (Differentialoperator)**
Das vorherige Beispiel 3.1.0.4 können wir verallgemeinern.

Für gegebene Skalare $c_0, c_1, \ldots, c_m$ ist der allgemeine Differentialoperator $\mathcal{L}$ definiert als

$$\begin{array}{rcl} \mathcal{L} : C^m[a,b] & \to & C^0[a,b] \\ f & \mapsto & \mathcal{L}f \end{array}$$

$\mathcal{L}f$ ist eine Funktion definiert als

$\mathcal{L}f : [a,b] \to \mathbb{R}$ mit $(\mathcal{L}f)(t) = c_0 f(t) + c_1 f'(t) + \ldots + c_m f^{(m)}(t)$.

$\mathcal{L}$ ist eine lineare Abbildung.

⌟

## 3.2 Abbildungsmatrix

Wenn wir eine Basis in einem Raum der Dimension $n$ festlegen, können wir den Raum mit $\mathbb{R}^n$ oder $\mathbb{C}^n$ identifizieren. Daraus folgt, dass lineare Abbildungen in diesen Räumen auch eine Repräsentation im Raum der Vektoren, in Form von Matrizen, haben.

**Definition 3.2.0.1. Abbildungsmatrix**

Sei $X$ ein linearer Raum mit Basis $\mathcal{B} = \{b_1, b_2, \ldots, b_n\}$,
sei $Y$ ein linearer Raum mit Basis $\mathcal{C} = \{c_1, c_2, \ldots, c_m\}$

und sei $\mathcal{F}$ eine lineare Abbildung von $X$ nach $Y$

$$\begin{array}{rccc} \mathcal{F} : & X & \to & Y \\ & b_j & \mapsto & \mathcal{F} b_j \end{array}$$

Die Bilder der Basiselemente aus $X$ sind Elemente in $Y$. Wir können sie also als Linearkombination von Elementen aus der Basis $\mathcal{C}$ schreiben:

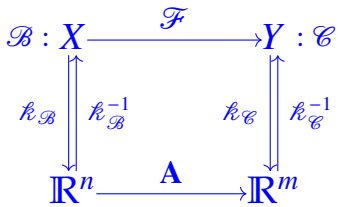
$$\mathcal{F} b_j = \sum_{i=1}^{m} a_{ij}\, c_i \,.$$

Wir erhalten somit eine $m \times n$-Matrix $\mathbf{A}$, mit den Einträgen $a_{ij}$.

Diese Matrix $\mathbf{A}$ nennen wir die *Abbildungsmatrix* von $\mathcal{F}$ zu den Basen $\mathcal{B}$, $\mathcal{C}$.

Im Folgenden werden wir den Zusammenhang zwischen der Abbildung $\mathcal{F}$ und der Abbildungsmatrix $\mathbf{A}$ besprechen und sehen, dass wir die Beiden in gewissem Sinne identifizieren können.

Wir stellen die Abbildung in einem Diagramm schematisch dar. Zum einen gibt es die Koordinatendarstellungen der Vektorräume $X$, $Y$ in $\mathbb{R}^n$ und $\mathbb{R}^m$. Wir notieren diese linearen Abbildungen mit $\kappa_{\mathcal{B}}$ und $\kappa_{\mathcal{C}}$. Zum anderen haben wir die lineare Abbildung $\mathcal{F}$ von $X$ nach $Y$ und die lineare Abbildung von $\mathbb{R}^n$ nach $\mathbb{R}^m$, die durch die Abbildungsmatrix $\mathbf{A}$ definiert ist:




Um die Verbindung zwischen $\mathcal{F}$ und $\mathbf{A}$ zu sehen, nehmen wir ein beliebiges Element $x$ aus dem Raum $X$.

Da $x \in X$ gilt, können wir $x$ als Linearkombination der Basiselemente aus $\mathcal{B}$ darstellen:

$$x = x_1 b_1 + x_2 b_2 + \ldots + x_n b_n \,.$$

Aus den $x_1, \ldots, x_n$ können wir den Koordinatenvektor $\mathbf{x}$ von $x$ bestimmen. Diese Abbildung vom Element $x$ zu dem Vektor $\mathbf{x}$ wird durch die Koordinatenabbildung $\kappa_{\mathcal{B}}$ realisiert:

$$\kappa_{\mathcal{B}} x = \mathbf{x} \in \mathbb{R}^n$$

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}.$$

Im Folgenden sehen wir, was passiert, wenn $\mathscr{F}$ auf $x$ angewandt wird:

$$\begin{aligned}
\mathscr{F}(x) &= \mathscr{F}(x_1 b_1 + x_2 b_2 + \ldots + x_n b_n) \\
(\mathscr{F} \text{ linear}) \quad &= x_1 \mathscr{F}(b_1) + x_2 \mathscr{F}(b_2) + \ldots + x_n \mathscr{F}(b_n) \\
&= \sum_{j=1}^{n} x_j \, \mathscr{F} b_j \\
(\text{Def. von } \mathscr{F} b_j \text{ aus } 3.2.0.1) \quad &= \sum_{j=1}^{n} x_j \sum_{i=1}^{m} a_{ij} \, c_i \\
&= \sum_{i=1}^{m} \underbrace{\left( \sum_{j=1}^{n} a_{ij} \, x_j \right)}_{y_i} c_i \\
&= \sum_{i=1}^{m} y_i \, c_i \, .
\end{aligned}$$

$\mathscr{F}(x)$ ist ein Element in $Y$ und aus der Definition, die hier gegeben ist können wir auch direkt die Koordinaten des dazugehörigen Koordinatenvektors $\mathbf{y}$ ablesen:

$$\mathscr{F}(x) = \sum_{i=1}^{m} y_i \, c_i \in Y \implies \mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix} \in \mathbb{R}^m \, .$$

Die einzelnen Komponenten dieses Vektors sind gegeben durch $\sum_{j=1}^{n} a_{ij} \, x_j$ für alle $i = 1, \ldots, m$. Wenn wir dies mit der Definition der Abbildungsmatrix aus 3.2.0.1 und dem Vektor $\mathbf{x}$ kombinieren so kommen wir zum Schluss, dass

$$\mathbf{y} = \mathbf{A}\mathbf{x} \, .$$

Somit ist $\mathscr{F}(x) = y$ das gleiche wie $\mathbf{A}\mathbf{x} = \mathbf{y}$; dies ist genau dann der Fall, wenn

$$\mathscr{F} = \hbar_{\mathscr{C}}^{-1} \, \mathbf{A} \, \hbar_{\mathscr{B}} \, .$$

In diesem Sinne lässt sich jede lineare Abbildung in einem endlichdimensionalen Raum durch eine Matrix darstellen.

**Definition 3.2.0.2. Kommutatives Diagramm**

Ein Diagramm heisst *kommutativ*, falls jede Abbildung aus dem Diagramm auf verschiedene Wege im Diagramm dargestellt werden kann.

**Beispiel 3.2.0.3 (Abbildungsmatrix der Multiplikation)**
In diesem Beispiel wollen wir die Abbildungsmatrix einer linearen Abbildung $\mathscr{M}$ berechnen.

Sei eine lineare Abbildung $\mathscr{M}$ gegeben als
$\mathscr{M} : \mathscr{P}_4 \to \mathscr{P}_5$
$f \mapsto \mathscr{M} f$'

mit $(\mathscr{M}f)(t) = tf(t)$.

Diese Abbildung ist offensichtlich linear, da

$$\mathscr{M}(f_1 + f_2) = tf_1(t) + tf_2(t) = \mathscr{M}f_1 + \mathscr{M}f_2$$
$$\mathscr{M}(cf) = ctf(t) = c\mathscr{M}f\,.$$

$\mathscr{M}$ agiert von $\mathscr{P}_4$ nach $\mathscr{P}_5$. In beiden Räumen sind die Monome als die Basis gewählt. Die jetzt berücksichtigte Basis von $\mathscr{P}_4$ ist also $\{p_0,\ p_1,\ p_2,\ p_3\}$ und die von $\mathscr{P}_5$ ist $\{p_0,\ p_1,\ p_2,\ p_3,\ p_4\}$.

Um die Abbildungsmatrix $\mathbf{M}$ zu erhalten, müssen wir die Abbildungen aller Basiselemente aus $\mathscr{P}_4$ berechnen und von diesen Abbildungen dann die Koordinatenvektoren zu der Basis $\mathscr{P}_5$, bestimmen:

$$\mathscr{M}(p_0)(t) = t1 \quad = p_1(t) \implies \mathscr{M}p_0 = 0p_0 + 1p_1 + 0p_2 + 0p_3 + 0p_4 \implies \begin{bmatrix}0\\1\\0\\0\\0\end{bmatrix}$$

$$\mathscr{M}(p_1)(t) = tt \quad = p_2(t) \implies \mathscr{M}p_1 = 0p_0 + 0p_1 + 1p_2 + 0p_3 + 0p_4 \implies \begin{bmatrix}0\\0\\1\\0\\0\end{bmatrix}$$

$$\mathscr{M}(p_2)(t) = tt^2 \quad = p_3(t) \implies \mathscr{M}p_2 = 0p_0 + 0p_1 + 0p_2 + 1p_3 + 0p_4 \implies \begin{bmatrix}0\\0\\0\\1\\0\end{bmatrix}$$

$$\mathscr{M}(p_3)(t) = tt^3 \quad = p_4(t) \implies \mathscr{M}p_3 = 0p_0 + 0p_1 + 0p_2 + 0p_3 + 1p_4 \implies \begin{bmatrix}0\\0\\0\\0\\1\end{bmatrix}.$$

Nachdem wir alle Koordinatenvektoren der Basiselemente aus $\mathscr{P}_4$ nach ihrer Abbildung durch $\mathscr{M}$ berechnet haben, werden wir diese Vektoren als Spalten zu einer Matrix zusammenfügen, diese Matrix entspricht dann der gesuchten Abbildungsmatrix $\mathbf{M}$:

$$\mathbf{M} = \begin{bmatrix}0&0&0&0\\1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix} \in \mathbb{R}^{5\times 4}\,.$$

Das ist also die Abbildungsmatrix der linearen Abbildung $\mathscr{M}$ zu den Basen $\{p_0, \ldots,\ p_3\}$ und $\{p_0, \ldots,\ p_4\}$.

Mit dieser Matrix können wir die Abbildung beliebiger Polynome aus $\mathscr{P}_4$ berechnen.
Sei ein Polynom $q \in \mathscr{P}_4$ gegeben durch $q(t) = 15 + \frac{1}{2}t + 4t^2 + \sqrt{3}t^3$. Das Bild von $q$ durch $\mathscr{M}$ können wir nun einfach mit der Matrix $\mathbf{M}$ berechnen.

Der Koordinatenvektor $\mathbf{a}$ von $q$ ist

$$\mathbf{a} = \begin{bmatrix} 15 \\ \frac{1}{2} \\ 4 \\ \sqrt{3} \end{bmatrix}.$$

$(\mathscr{M}q)(t)$ bekommt man so:

$$\mathbf{Ma} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 15 \\ \frac{1}{2} \\ 4 \\ \sqrt{3} \end{bmatrix} = \begin{bmatrix} 0 \\ 15 \\ \frac{1}{2} \\ 4 \\ \sqrt{3} \end{bmatrix}$$

$$\Longrightarrow \quad (\mathscr{M}q)(t) = 15t + \frac{1}{2}t^2 + 4t^3 + \sqrt{3}t^4 .$$

⌟

## Isomorphismen und Automorphismen

**Definition 3.2.0.4. Isomorphismen und Automorphismen**

Eine bijektive lineare Abbildung $\mathscr{F} : X \to Y$ heisst *Isomorphismus*.
In diesem Fall sagt man, dass die linearen Räume $X$, $Y$ *isomorph* zueinander sind. Falls $X = Y$, dann heisst $\mathscr{F}$ *Automorphismus*.

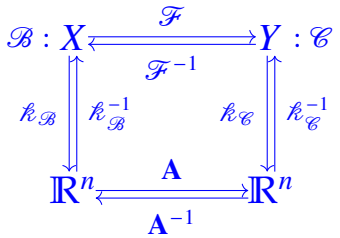
**Bemerkung 3.2.0.5 (Inverse eines Isomorphismus ist auch Isomorphismus)**
Wenn $\mathscr{F} : X \to Y$ ein Isomorphismus ist, dann existiert eine Inverse $\mathscr{F}^{-1} : Y \to X$, die auch eine lineare Abbildung und ein Isomorphismus ist. ⌟

**Bemerkung 3.2.0.6 (Die Koordinatenabbildung ist ein Isomorphismus)**
Wenn $X$ ein linearer Raum mit Dimension $n$ und Basis $\mathscr{B} = \{b_1, \ldots, b_n\}$ ist, dann ist die Koordinatenabbildung $\mathscr{k}_{\mathscr{B}} : X \to \mathbb{R}^n$ ein Isomorphismus.

⌟

## Inverse

Gegeben seien zwei endlichdimensionale lineare Räume $X$ und $Y$ mit Basen $\mathscr{B}$, $\mathscr{C}$. Zwischen diesen Räumen sei ein Isomorphismus $\mathscr{F}$ gegeben, und somit gibt es nach 3.2.0.5 eine Inverse $\mathscr{F}^{-1}$.
Auch ist jede lineare Abbildung zwischen endlichdimensionalen Räumen durch eine Abbildungsmatrix repräsentierbar, also gibt es eine Matrix $\mathbf{A}$ für $\mathscr{F}$ und eine Matrix $\mathbf{A}^{-1}$ für $\mathscr{F}^{-1}$. Hier ist das Diagramm aller dieser Abbildungen:

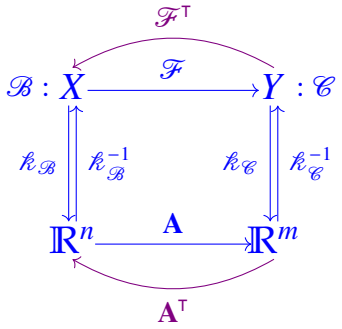


Wenn $\mathcal{F}$ durch $\mathbf{A}$ und die $\mathcal{F}^{-1}$ durch $\mathbf{A}^{-1}$ realisiert werden, dann muss gelten, dass $\mathbf{A}^{-1}$ die Inverse von $\mathbf{A}$ ist.

Das heisst, falls $\mathcal{F}$ ein Isomorphismus ist, dann ist die dazugehörige Abbildungsmatrix invertierbar.

## Die transponierte Abbildung

Im Folgenden seien $X$ und $Y$ zwei lineare Räume mit Basen $\mathcal{B}$ und $\mathcal{C}$. Wir notieren die Dimensionen von $X$ und $Y$ entsprechend mit $n$ und $m$. Zwischen diesen Räumen sei eine lineare Abbildung $\mathcal{F}$ definiert:



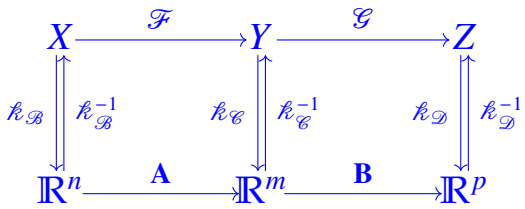

Wenn die Dimensionen von $X$ und $Y$ nicht übereinstimmen, hat $\mathbf{A}$ keine Inverse und demzufolge ist $\mathcal{F}$ nicht bijektiv. Wir können aber immer eine Matrix $\mathbf{A}^\mathsf{T}$ definieren, welche Elemente von $\mathbb{R}^m$ nach $\mathbb{R}^n$ transformiert. Für diese Matrix $\mathbf{A}^\mathsf{T}$ können wir dann auch die lineare Abbildung $\mathcal{F}^\mathsf{T}$ definieren. Diese Funktion definieren wir über die Koordinatenabbildung aus der Matrix $\mathbf{A}^\mathsf{T}$.

$$\begin{aligned} \mathcal{F}^\mathsf{T} : Y &\to X \\ y &\mapsto \mathcal{F}^\mathsf{T} y \end{aligned}$$

mit $\mathcal{F}^\mathsf{T}$ gegeben als $\mathcal{F}^\mathsf{T} = \Phi_\mathcal{B}^{-1} \mathbf{A}^\mathsf{T} \Phi_\mathcal{C}$.

## Verknüpfung von Funktionen

Was geschieht, wenn mehrere lineare Abbildungen miteinander verknüpft werden?
Seien drei lineare Räume $X$, $Y$ und $Z$ gegeben und zwischen diesen Räumen seien zwei lineare Abbildungen $\mathcal{F} : X \to Y$ und $\mathcal{G} : Y \to Z$:




Da dieses Diagramm kommutativ ist, können wir die Verknüpfung $\mathcal{G}\mathcal{F}$ wie folgt ausdrücken:

$$\mathcal{G}\mathcal{F}\, x = \Phi_\mathcal{D}^{-1} \mathbf{B}\mathbf{A} \Phi_\mathcal{B}\, x\,.$$

## Kern und Bild

**Definition 3.2.0.7. Kern und Bild**

Sei $\mathcal{F} : X \to Y$ eine Abbildung, dann können folgende zwei Mengen definiert werden:

**Kern von $\mathcal{F}$:**
$\text{Kern}(\mathcal{F}) = \{x \in X,\ \mathcal{F}x = 0\}$.
Wenn $\mathcal{F}$ eine lineare Abbildung ist, dann ist $\text{Kern}(\mathcal{F})$ ein linearer Unterraum von $X$.

**Bild von $\mathcal{F}$:**
$\text{Bild}(\mathcal{F}) = \{\mathcal{F}x \in Y, x \in X\}$.
Wenn $\mathcal{F}$ eine lineare Abbildung ist, dann ist $\text{Bild}(\mathcal{F})$ ein linearer Unterraum von $Y$.

**Satz 3.2.0.8. Kern einer linearen Injektion**

*Wenn $\mathcal{F} : X \to Y$ eine lineare Abbildung ist, dann gilt:*
*$\mathcal{F}$ ist injektiv genau dann, wenn $\text{Kern}(\mathcal{F}) = \{0\}$.*

**Satz 3.2.0.9. Dimensionssatz**

*Wenn $\mathcal{F} : X \to Y$ eine lineare Abbildung zwischen den beiden endlichdimensionalen Räumen $X$ und $Y$ ist, dann gilt:*

$$\dim(\text{Kern}(\mathcal{F})) + \dim(\text{Bild}(\mathcal{F})) = \dim(X)\,.$$

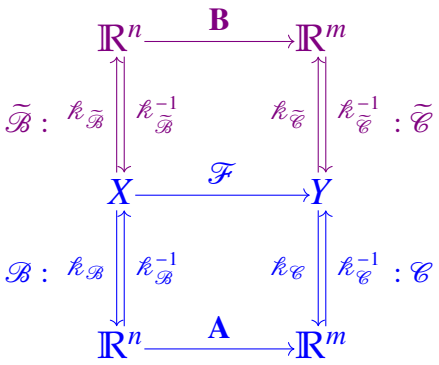
**Definition 3.2.0.10. Rang einer linearen Abbildung**

Der Rang einer linearen Abbildung $\mathcal{F}$ zwischen zwei endlichdimensionalen Räumen ist

$$\text{Rang}(\mathcal{F}) = \text{Rang}(\mathbf{F}) = \dim(\text{Bild}(\mathbf{F}))\,.$$

## 3.3 Abbildungsmatrix bei Koordinatentransformationen

Sei $X$ ein linearer Raum mit den beiden Basen $\mathscr{B}$ und $\widetilde{\mathscr{B}}$, sei $Y$ ein anderer linearer Raum auch mit zwei Basen $\mathscr{C}$ und $\widetilde{\mathscr{C}}$ und sei $\mathscr{F} : X \to Y$ eine lineare Abbildung. Das Diagramm dazu ist:




Es gibt also eine neue Matrix $\mathbf{B}$, welche auch die Abbildung $\mathscr{F}$ darstellt, und zwar mit den neuen Basen $\widetilde{\mathscr{B}}$, $\widetilde{\mathscr{C}}$. Da dies ein kommutatives Diagramm ist, können wir die Matrix $\mathbf{B}$ mit den internen Basistransformationen und $\mathbf{A}$ ausrechnen:

$$\mathbf{B} = \underbrace{\kappa_{\widetilde{\mathscr{C}}}\kappa_{\mathscr{C}}^{-1}}_{\mathbf{S}^{-1}}\,\mathbf{A}\,\underbrace{\kappa_{\mathscr{B}}\kappa_{\widetilde{\mathscr{B}}}^{-1}}_{\mathbf{T}} = \mathbf{S}^{-1}\mathbf{A}\,\mathbf{T}$$
$$\mathbf{B} = \mathbf{S}^{-1}\mathbf{A}\,\mathbf{T} \iff \mathbf{A} = \mathbf{S}\,\mathbf{B}\,\mathbf{T}^{-1}\,.$$

$\mathbf{T}$ realisiert den Wechsel von der Basis $\widetilde{\mathscr{B}}$ in die Basis $\mathscr{B}$ im linearen Raum $X$. D. h. $\mathbf{T}\widetilde{\mathbf{x}} = \mathbf{x}$ für den Koordinatenvektor $\mathbf{x}$ von $x \in X$ in der alten Basis $\mathscr{B}$ und den Koordinatenvektor $\widetilde{\mathbf{x}}$ von $x$ in der neuen Basis $\widetilde{\mathscr{B}}$. $\mathbf{S}^{-1}$ realisiert den Wechsel von der alten Basis $\mathscr{C}$ in die neue Basis $\widetilde{\mathscr{C}}$ im linearen Raum $Y$.

Den Rang der neuen Matrix $\mathbf{B}$ können wir mit der Definition 3.2.0.10 bestimmen:

$$\mathrm{Rang}(\mathbf{A}) = \mathrm{Rang}(\mathscr{F}) = \mathrm{Rang}(\mathbf{B})\,.$$

Beide Matrizen haben also denselben Rang, den Rang der linearen Abbildung $\mathscr{F}$.

**Beispiel 3.3.0.1 (Abbildungsmatrix bei Basiswechsel)**
Seien $G_3$ und $U_2$ zwei lineare Räume. $G_3$ der Raum der geraden Polynome vom Grad höchstens vier mit der Basis $\mathscr{B} = \{p_0,\ p_2,\ p_4\}$ und $U_2$ der Raum der ungeraden Polynome vom Grad höchstens 3 mit der Basis $\mathscr{C} = \{p_1,\ p_3\}$:

$$G_3 = \mathrm{Span}\,\mathscr{B} = \mathrm{Span}\{p_0,\ p_2,\ p_4\}\,,$$
$$U_2 = \mathrm{Span}\,\mathscr{C} = \mathrm{Span}\{p_1,\ p_3\}\,.$$

Zwischen diesen beiden Räumen ist eine lineare Abbildung in Form des Ableitungsoperators $\mathscr{D}$ definiert:

$$\begin{array}{rccc}\mathscr{D} : & G_3 & \to & U_2 \\ & f & \mapsto & \mathscr{D}f\end{array}\,,$$

mit der Funktion $\mathscr{D}f$ definiert als $(\mathscr{D}f)(t) = f'(t)$.

Für $\mathscr{D}$ wollen wir nun die Abbildungsmatrix $\mathbf{D}$ berechnen. Die Koordinaten aller Basiselemente von $G_3$ nach der Abbildung durch $\mathscr{D}$ ergeben diese Koordinatenvektoren:

$$\begin{aligned}
(\mathscr{D}p_0)(t) = 0 \quad &= 0 \qquad \Longrightarrow \mathscr{D}p_0 = 0p_1 + 0p_3 \Longrightarrow \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\
(\mathscr{D}p_2)(t) = 2t \quad &= 2p_1(t) \Longrightarrow \mathscr{D}p_2 = 2p_1 + 0p_3 \Longrightarrow \begin{bmatrix} 2 \\ 0 \end{bmatrix} \\
(\mathscr{D}p_4)(t) = 4t^3 \quad &= 4p_3(t) \Longrightarrow \mathscr{D}p_4 = 0p_1 + 4p_3 \Longrightarrow \begin{bmatrix} 0 \\ 4 \end{bmatrix}.
\end{aligned}$$

Die Abbildungsmatrix $\mathbf{D}$ ist die Matrix, welche die obigen Koordinatenvektoren als Spalten hat:

$$\mathbf{D} = \begin{bmatrix} 0 & 2 & 0 \\ 0 & 0 & 4 \end{bmatrix}.$$

Für die Matrix $\mathbf{D}$ bestimmen wir den Kern und das Bild und deren Dimensionen:

$$\begin{aligned}
\mathscr{K}_{\mathscr{B}}^{-1}(\mathrm{Kern}(\mathbf{D})) = \mathrm{Span}\{p_0\} \quad &\Longrightarrow \dim(\mathrm{Kern}(\mathbf{D})) = 1\,, \\
\mathscr{K}_{\mathscr{C}}^{-1}(\mathrm{Bild}(\mathbf{D})) = \mathrm{Span}\{p_1,\, p_3\} \quad &\Longrightarrow \dim(\mathrm{Bild}(\mathbf{D})) = 2\,.
\end{aligned}$$

Wenn diese beiden Dimensionen addiert werden, so müssten wir laut Satz 3.2.0.9 die Dimension des Raumes $G_3$ erhalten. Dies ist tatsächlich der Fall, denn

$$\dim(\mathrm{Kern}(\mathbf{D})) + \dim(\mathrm{Bild}(\mathbf{D})) = 3 = \dim(G_3)\,.$$

Für die beiden linearen Räume $G_3$ und $U_2$ können wir aber auch neue Basen definieren: $\tilde{\mathscr{B}} = \{r_1,\, r_2,\, r_3\}$ mit

$$\begin{aligned}
r_1(t) &= 1 + t^2 \\
r_2(t) &= 1 - t^2 \\
r_3(t) &= 1 + t^2 + t^4
\end{aligned}$$

und $\tilde{\mathscr{C}} = \{q_1,\, q_2\}$ mit

$$\begin{aligned}
q_1(t) &= t \\
q_2(t) &= 3t + 2t^3\,.
\end{aligned}$$

Für diese beiden neuen Basen können wir die Transformationsmatrizen $\mathbf{T}$ und $\mathbf{S}$ (jeweils für $G_3$ und $U_2$) bestimmen. Um die Transformationsmatrizen zu erhalten, schreiben wir jedes Element der neuen Basis als eine Linearkombination von Elementen der alten Basis. Davon nehmen wir die Koordinatenabbildung und fügen sie in eine Matrix zusammen.

Die Transformationsmatrix $\mathbf{T}$ für den Basiswechsel im Raum $G_3$ berechnen wir so:

$$\begin{aligned}
r_1 = 1p_0 + 1p_2 + 0p_4 \quad &\Longrightarrow \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} \\
r_2 = 1p_0 - 1p_2 + 0p_4 \quad &\Longrightarrow \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix} \\
r_3 = 1p_0 + 1p_2 + 1p_4 \quad &\Longrightarrow \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix},
\end{aligned}$$

$$\mathbf{T} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -1 & 1 \\ 0 & 0 & 1 \end{bmatrix}.$$

Die Transformationsmatrix $\mathbf{S}$ für den Basiswechsel im Raum $U_2$ berechnen wir so:

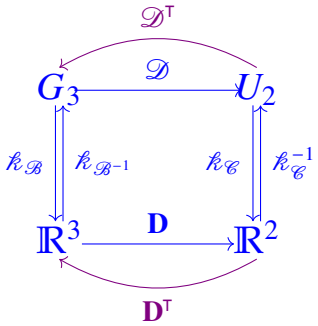
$$q_1 = 1p_1 + 0p_3 \quad \Longrightarrow \quad \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$
$$q_2 = 3p_1 + 2p_3 \quad \Longrightarrow \quad \begin{bmatrix} 3 \\ 2 \end{bmatrix}.$$

$$\mathbf{S} = \begin{bmatrix} 1 & 3 \\ 0 & 2 \end{bmatrix} \Longrightarrow \mathbf{S}^{-1} = \begin{bmatrix} 1 & -\frac{3}{2} \\ 0 & \frac{1}{2} \end{bmatrix}.$$

Mit diesen beiden Matrizen können wir die Abbildungsmatrix $\mathbf{B}$ von $\mathscr{D}$ in den neuen Basen $\widetilde{\mathscr{B}}$, $\widetilde{\mathscr{C}}$ berechnen:

$$\begin{aligned} \mathbf{B} &= \mathbf{S}^{-1}\,\mathbf{D}\mathbf{T} \\ &= \begin{bmatrix} 1 & -\frac{3}{2} \\ 0 & \frac{1}{2} \end{bmatrix} \begin{bmatrix} 0 & 2 & 0 \\ 0 & 0 & 4 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & -1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 2 & -2 & -4 \\ 0 & 0 & 2 \end{bmatrix}. \end{aligned}$$

In 3.2 haben wir die Transponierte einer Abbildung definiert. Was ist die Transponierte von $\mathscr{D}$?




Die Matrix $\mathbf{D}^\mathsf{T}$ zu bestimmen ist einfach, da wir die Matrix $\mathbf{D}$ bereits berechnet haben:

$$\mathbf{D}^\mathsf{T} = \begin{bmatrix} 0 & 0 \\ 2 & 0 \\ 0 & 4 \end{bmatrix},$$

also

$$\mathscr{D}^\mathsf{T} g = \Phi_{\mathscr{B}}^{-1} \mathbf{D}^\mathsf{T} \Phi_{\mathscr{C}} g\,.$$

Sei $g \in U_2$, also $g = g_1 p_1 + g_2 p_3$ mit den Koordinaten $g_2,\ g_2$ in der Basis $\mathscr{B}$:

$$\mathbf{g} = \begin{bmatrix} g_1 \\ g_2 \end{bmatrix} \Longrightarrow$$

$$\begin{aligned} \mathscr{D}^{\mathsf{T}} g = \not{k}_{\mathscr{B}}^{-1} \mathbf{D}\,\mathbf{g} = \not{k}_{\mathscr{B}}^{-1} \begin{bmatrix} 0 & 0 \\ 2 & 0 \\ 0 & 4 \end{bmatrix} \begin{bmatrix} g_1 \\ g_2 \end{bmatrix} = \not{k}_{\mathscr{B}}^{-1} \begin{bmatrix} 0 \\ 2g_1 \\ 4g_2 \end{bmatrix} \\ = 0p_0 + (2g_1)p_2 + (4g_2)p_4 \\ = 2g_1\ p_2 + 4g_2\ p_4\,. \end{aligned}$$

Dies gibt uns eine direkte andere Art, $\mathscr{D}^{\mathsf{T}}$ zu beschreiben:

$$\begin{array}{rccc} \mathscr{D}^{\mathsf{T}} : & U_2 & \to & G_3 \\ & g & \mapsto & \mathscr{D}^{\mathsf{T}} g \end{array},$$

mit $\mathscr{D}^{\mathsf{T}} g$ definiert als $\mathscr{D}^{\mathsf{T}} g = 2g_1\ p_2 + 4g_2\ p_4$, wobei $g_1,\ g_2$ die Koordinaten von $g$ in $U_2$ sind.

Genau wie für $\mathbf{D}$ betrachten wir auch für $\mathscr{D}^{\mathsf{T}}$ den Kern und das Bild und deren Dimensionen:

$$\begin{aligned} &\mathrm{Kern}(\mathscr{D}^{\mathsf{T}}) = \{0\} && \Longrightarrow\ \dim(\mathrm{Kern}(\mathscr{D}^{\mathsf{T}})) = 0 \\ &\mathrm{Bild}(\mathscr{D}^{\mathsf{T}}) = \mathrm{Span}\{p_2,\ p_4\} && \Longrightarrow\ \dim(\mathrm{Bild}(\mathscr{D}^{\mathsf{T}})) = 2\,. \end{aligned}$$

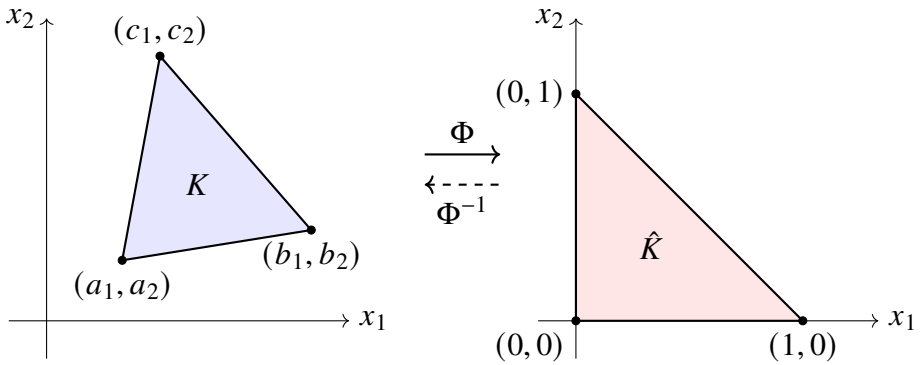
Auch hier können wir den Dimensionssatz 3.2.0.9 anwenden:

$$\dim(\mathrm{Kern}(\mathscr{D}^{\mathsf{T}})) + \dim(\mathrm{Bild}(\mathscr{D}^{\mathsf{T}})) = 2 = \dim(U_2)\,.$$

## 3.4 Anwendung: Referenztransformationen

### 3.4.1 Problemstellung

Gegeben sei ein allgemeines Dreieck $K$ in $\mathbb{R}^2$ mit den Eckpunkten $\mathbf{a} = [a_1\, a_2]^\mathsf{T}$, $\mathbf{b} = [b_1\, b_2]^\mathsf{T}$ und $\mathbf{c} = [c_1\, c_2]^\mathsf{T}$. Gesucht ist eine Transformation $\Phi : K \to \hat{K}$, welche dieses auf das Referenzdreieck $\hat{K}$ abbildet.




Man beachte, dass es sich im Allgemeinen um eine affine Transformation handelt.[1] Das heisst, dass sich die Transformation aus einer linearen Transformation und einer Verschiebung zusammensetzt. Demnach ist $\Phi$ von der folgenden Form:

$$\Phi(\mathbf{x}) = \mathbf{M}\mathbf{x} + \mathbf{t}$$

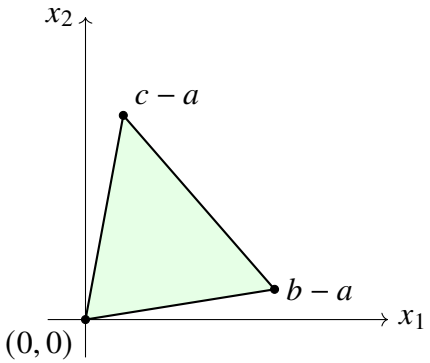
wobei $\mathbf{M} \in \mathbb{R}^{2\times 2}$ eine reguläre Matrix und $\mathbf{t} \in \mathbb{R}^2$ ein Verschiebungsvektor ist.

Da es wesentlich einfacher ist, die inverse Transformation $F := \Phi^{-1} : \hat{K} \to K \quad \Phi^{-1}(\mathbf{y}) = \mathbf{J}\mathbf{y} + \mathbf{p}$ aufzustellen, werden wir uns zunächst mit dieser auseinandersetzen.

Betrachten wir zunächst, welche Bedingungen $F$ erfüllen muss:

1. $F([0\,0]^\mathsf{T}) = [a_1\, a_2]^\mathsf{T}$
2. $F([1\,0]^\mathsf{T}) = [b_1\, b_2]^\mathsf{T}$
3. $F([0\,1]^\mathsf{T}) = [c_1\, c_2]^\mathsf{T}$

Aus Bedingung (1.) folgt bereits unmittelbar, dass $\mathbf{p} = \mathbf{a}$ gelten muss. Abschliessend müssen die Einheitsvektoren $e_1$ und $e_2$ auf die Punkte $\mathbf{b}$ und $\mathbf{c}$, relativ zu $\mathbf{a}$, abgebildet werden.




Dies kann durch die folgende Abbildungsmatrix realisiert werden.

[1] Insbesondere reicht eine lineare Transformation nicht aus, da diese fordern würde, dass $\Phi([0\,0]^\mathsf{T}) = [0\,0]^\mathsf{T}$.

$$\mathbf{J} = \begin{bmatrix} b_1 - a_1 & c_1 - a_1 \\ b_2 - a_2 & c_2 - a_2 \end{bmatrix}$$

Um die ursprünglich gesuchte Abbildung zu erhalten, muss die Inverse von $F$ gefunden werden. Eine solche existiert genau dann, wenn die Punkte $\mathbf{a}$, $\mathbf{b}$ und $\mathbf{c}$ nicht kollinear sind. Also wenn das Dreieck nicht zu einer Linie entartet ist. Falls die Abbildung $\Phi$ also existiert, ist sie durch den folgenden Ausdruck gegeben:

$$\Phi(\mathbf{x}) = \mathbf{J}^{-1}(\mathbf{x} - \mathbf{a})$$

### 3.4.2 Beispiel

Wir betrachten ein Dreieck $K$ mit den Eckpunkten $\mathbf{a} = [1\,1]^\mathsf{T}$, $\mathbf{b} = [3\,2]^\mathsf{T}$ und $\mathbf{c} = [2\,3]^\mathsf{T}$.

Zunächst bestimmen wir die Abbildungsmatrix $\mathbf{J}$ und den Verschiebungsvektor $\mathbf{p}$ für die Transformation $F : \hat{K} \to K$:

$$\mathbf{p} = \mathbf{a} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$\mathbf{J} = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

Die affine Abbildung vom Referenzdreieck auf das allgemeine Dreieck lautet somit:

$$F(\mathbf{y}) = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \mathbf{y} + \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

Um die Rücktransformation $\Phi : K \to \hat{K}$ aufzustellen, berechnen wir die Inverse von $\mathbf{J}$:

$$\mathbf{J}^{-1} = \frac{1}{3} \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}$$

Die gesuchte Referenz-Transformation $\Phi$ ergibt sich dann zu:

$$\Phi(\mathbf{x}) = \frac{1}{3} \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \left( \mathbf{x} - \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right)$$

### 3.4.3 Erweiterte Matrix

Wir haben im vorhergehenden Abschnitt bemerkt, dass für eine solche Referenztransformation eine affine Abbildung notwendig ist, jedoch sind lineare Abbildungen in der Praxis deutlich angenehmer. Das Ziel dieses Abschnitts ist zu zeigen, wie affine Abbildungen als Abbildungsmatrizen linearer Abbildungen dargestellt werden können.

Wir bemerken, dass die beiden folgenden Darstellungen äquivalent sind.

$$\begin{bmatrix} \mathbf{y} \\ 1 \end{bmatrix} = \left[\begin{array}{ccc|c} & \mathbf{A} & & \mathbf{b} \\ 0 & \cdots & 0 & 1 \end{array}\right] \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix}$$

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{b}$$

Ein wesentlicher Vorteil dieser Matrixdarstellung ist, dass diese rechnergestützte Verfahren vereinfachen können. Beispielsweise kann die Verknüpfung von affinen Abbildungen durch eine Matrix-Multiplikation berechnet werden.

# Kapitel 4

# Norm und Skalarprodukt in linearen Räumen

## Inhalt


**4.1 Normierte lineare Räume . . . 120**
**4.2 Anwendung: Störungstheorem . . . 126**
4.2.1 Einleitung . . . 126
4.2.2 Herleitung . . . 126
4.2.3 Anwendung . . . 128
4.2.4 Beispiel: Geradenschnittpunkt . . . 130
**4.3 Skalarprodukt in linearen Räumen . . . 133**
**4.4 Gram-Schmidt-Algorithmus . . . 143**
**4.5 Projektoren . . . 149**
**4.6 Lineare Abbildungen auf Räume mit Skalarprodukt . . . 153**


## 4.1 Normierte lineare Räume

**Definition 4.1.0.1. Norm**

Sei $V$ ein linearer Raum.
Die Funktion $\|\cdot\| : V \to [0, \infty[$ heisst Norm auf $V$, falls sie folgende Eigenschaften erfüllt:

(N1) Aus $\|v\| = 0$ folgt $v = 0$.

(N2) Sei $\alpha$ ein Skalar und $v \in V$ beliebig. Dann gilt: $\|\alpha v\| = |\alpha|\|v\|$.

(N3) Für beliebige $v,\ w \in V$ gilt: $\|v + w\| \leq \|v\| + \|w\|$ (Dreiecksungleichung).

**Definition 4.1.0.2. Normierter linearer Raum**

Ein linearer Raum $V$, welcher eine Norm $\|\cdot\|$ besitzt, heisst normierter linearer Raum.

**Beispiel 4.1.0.3 (Euklidische Norm)**
Im Kapitel 1.7 über orthogonale Matrizen sahen wir die Definition der euklidischen Norm auf den Räumen $\mathbb{R}^d$ und $\mathbb{C}^d$:

$$\|\mathbf{x}\|_2 \stackrel{\text{def}}{=} \sqrt{x_1^2 + x_2^2 + \ldots + x_d^2} = \sqrt{\mathbf{x}^\mathsf{T}\mathbf{x}}$$

und

$$\|\mathbf{x}\|_2 \stackrel{\text{def}}{=} \sqrt{\overline{x_1}x_1 + \overline{x_2}x_2 + \ldots + \overline{x_d}x_d} = \sqrt{\mathbf{x}^\mathsf{H}\mathbf{x}}.$$

Als wir die euklidische Norm für $\mathbb{R}^d$ im Kapitel über orthogonale Matrizen eingeführt haben, sahen wir diese Definition als die natürliche Erweiterung der Norm auf $\mathbb{R}^3$. Da wir nun eine klare Definition der Norm haben, müssen wir beweisen, dass es sich tatsächlich um eine Norm handelt. ⌟

Die Beweise von (N1) und (N2) sind trivial, weshalb wir sie nicht besprechen werden. Der Beweis von (N3) hingegen ist nicht trivial und wir werden ihn deswegen auch durchführen.

Wir müssen zeigen, dass $\|\mathbf{v}+\mathbf{w}\| \leq \|\mathbf{v}\| + \|\mathbf{w}\|$ für beliebige $\mathbf{v}$ und $\mathbf{w}$ gilt. Um das Rechnen mit Wurzeln zu vermeiden, beweisen wir die identische Aussage $\|\mathbf{v}+\mathbf{w}\|^2 \leq (\|\mathbf{v}\| + \|\mathbf{w}\|)^2$, oder auch:

$$\begin{aligned}(\mathbf{v}+\mathbf{w})^\mathsf{T}(\mathbf{v}+\mathbf{w}) &\leq \|\mathbf{v}\|^2 + 2\|\mathbf{v}\|\|\mathbf{w}\| + \|\mathbf{w}\|^2\\ \mathbf{v}^\mathsf{T}\mathbf{v} + \mathbf{v}^\mathsf{T}\mathbf{w} + \mathbf{w}^\mathsf{T}\mathbf{v} + \mathbf{w}^\mathsf{T}\mathbf{w} &\leq \|\mathbf{v}\|^2 + 2\|\mathbf{v}\|\|\mathbf{w}\| + \|\mathbf{w}\|^2\\ \mathbf{v}^\mathsf{T}\mathbf{w} + \mathbf{w}^\mathsf{T}\mathbf{v} &\leq 2\|\mathbf{v}\|\|\mathbf{w}\|\,,\end{aligned}$$

d. h., es reicht zu zeigen:

**Satz 4.1.0.4. Cauchy-Bunjakowski-Schwarz-Ungleichung**

$$\mathbf{v}^\mathsf{T}\mathbf{w} \leq \|\mathbf{v}\|\|\mathbf{w}\|\,.$$

Diese Ungleichung ist die berühmteste Ungleichung in der Mathematik, die *Cauchy-Schwarz-Ungleichung* (komplett eigentlich Cauchy-Bunjakowski[1]-Schwarz). Sie hat extrem wichtige Anwendungen, so basiert zum Beispiel praktisch die ganze Physik auf der Richtigkeit dieser Ungleichung.

*Beweis.* (Beweis der Cauchy-Bunjakowski-Schwarz-Ungleichung)
Damit wir nicht mit Wurzel arbeiten müssen, quadrieren wir diese Ungleichung und schreiben die Ausdrücke komplett mit Summen:

$$\left(\sum_{j=1}^{d} v_j\, w_j\right)^2 \leq \left(\sum_{j=1}^{d} v_j^2\right)\left(\sum_{j=1}^{d} w_j^2\right).$$

Der Beweis dieser Ungleichung wird mit mathematischer Induktion durchgeführt.
Für $d = 1$ ist die Ungleichung trivial erfüllt. Schreiben Sie diese Ungleichung für $d = 2$ auf, multiplizieren Sie die Faktoren aus und kürzen Sie, was möglich ist! Sie werden dann gleich die Beweisidee und die Bedingung für das Gleichheitszeichen erkennen.
Wir nehmen an, die Ungleichung ist für $d-1$ erfüllt und zeigen, dass sie auch für $d$ erfüllt ist. Wie im Fall $d = 2$ rechnen wir (mit Voraussicht) aus:

$$\begin{aligned}\left(\sum_{j=1}^{d-1} v_j w_j + v_d w_d\right)^2 &\leq \left(\sum_{j=1}^{d-1} v_j^2 + v_d^2\right)\left(\sum_{j=1}^{d-1} w_j^2 + w_d^2\right)\\ 2v_d w_d\left(\sum_{j=1}^{d-1} v_j w_j\right) + \left(\sum_{j=1}^{d-1} v_j w_j\right)^2 &\leq w_d^2\left(\sum_{j=1}^{d-1} v_j^2\right) + v_d^2\left(\sum_{j=1}^{d-1} w_j^2\right) + \left(\sum_{j=1}^{d-1} v_j^2\right)\left(\sum_{j=1}^{d-1} w_j^2\right).\end{aligned}$$

Die Induktionsannahme (für $d-1$) sichert die Ungleichung für die jeweils letzten Terme:

$$\left(\sum_{j=1}^{d-1} v_j\, w_j\right)^2 \leq \left(\sum_{j=1}^{d-1} v_j^2\right)\left(\sum_{j=1}^{d-1} w_j^2\right),$$

[1] Schreibweise und Übersicht in Wikipedia

d. h., es reicht zu zeigen:

$$\sum_{j=1}^{d-1}\left(2v_d w_d v_j w_j\right) \leq \sum_{j=1}^{d-1} w_d^2 v_j^2 + \sum_{j=1}^{d-1} v_d^2 w_j^2 .$$

Wenn wir alles auf die rechte Seite bringen, sehen wir, dass die letzte Ungleichheit äquivalent mit

$$0 \leq \sum_{j=1}^{d-1}\left(w_d v_j - v_d w_j\right)^2$$

ist. Diese ist offensichtlich erfüllt. Somit ist auch die Cauchy-Schwarz-Ungleichung für $d$ Terme erfüllt. Die mathematische Induktion sichert also, dass die Ungleichung für allgemeines $d \in \mathbb{N}$ gilt.

□

## Andere Normen

Die euklidische Norm ist nicht die einzige Norm, die wir in $\mathbb{R}^d$ definieren können. Es gibt auch andere wichtige Normen, so zum Beispiel die $p$-Norm und die $\infty$-Norm.

**$p$-Norm:**

Für beliebiges aber festes $p \geq 1$ definieren wir die Norm $\|\cdot\|_p$:

$$\|x\|_p = \left(|x_1|^p + |x_2|^p + \ldots + |x_d|^p\right)^{\frac{1}{p}} .$$

**$\infty$-Norm:**

Eine andere wichtige Norm auf $\mathbb{R}^d$ ist

$$\|x\|_\infty = \max\{|x_1|,\ |x_2|, \ldots, |x_d|\} .$$

Die Bedeutung der Norm eines Vektors ist die Länge. Da wir verschiedene Normen haben, haben wir also auch verschiedene Längen eines Vektors und demzufolge verschiedene Geometrie-Arten. Es ist interessant zu schauen, wie ein Ball , d. h. die Menge $\{\mathbf{x} \in \mathbb{R}^d, \text{ mit } \|\mathbf{x}\| \leq 1\}$ in diesen verschiedenen Normen aussieht:

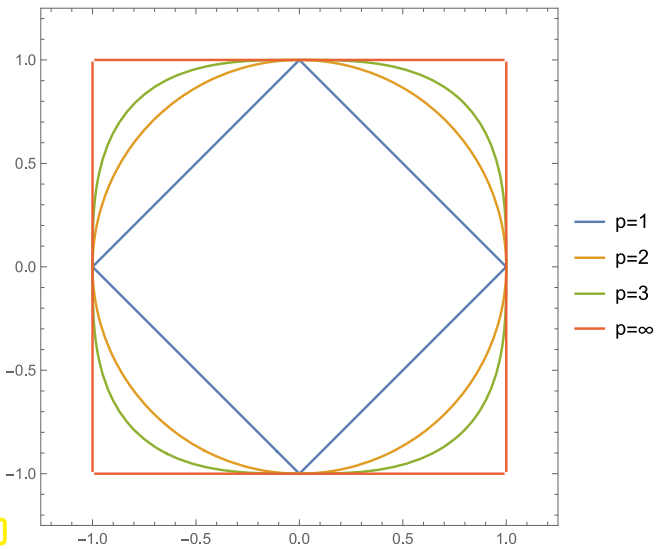



Fig. 5

Normen können auch für unendlichdimensionale Räume definiert werden, so zum Beispiel für den Raum der stetigen Funktionen auf $[0,1]$.

**Beispiel 4.1.0.5 (sup-Norm)**
Sei der lineare Raum der stetigen Funktionen $V = \{f : [0,1] \to \mathbb{R}, f \text{ stetig}\}$. Die Supremumsnorm (kurz: sup-Norm), teils auch Maximumsnorm (kurz: max-Norm) oder Cauchy-Norm genannt, ist

$$\|f\|_0 = \sup_{x\in[0,1]} |f(x)| .$$

Man kann einfach beweisen (Analysis I), dass dies eine Norm im linearen Raum der stetigen Funktionen ist. Diese Norm wird in der Analysis verwendet, um die gleichmässige Konvergenz von Folgen von Funktionen zu beschreiben. Eine Funktionenfolge $f_n$ konvergiert gleichmässig zur Funktion $f$, genau dann, wenn $\|f_n - f\|_0$ gegen null konvergiert.

⌟

**Satz 4.1.0.6. Äquivalenz der Normen**

*Alle Normen in $\mathbb{R}^d$ sind äquivalent. In anderen Worten:*
*Seien $\|\cdot\|$ und $|||\cdot|||$ Normen in $\mathbb{R}^d$, dann gibt es eine Konstante $C \geq 1$, abhängig von der Dimension $d$, so dass*

$$\frac{1}{C}\|\mathbf{v}\| \leq |||\mathbf{v}||| \leq C\|\mathbf{v}\| \text{ für alle } \mathbf{v} \in \mathbb{R}^d .$$

Diese Ungleichung bedeutet für die Konvergenz: Eine Folge konvergiert in der $\|\cdot\|$ Norm dann und nur dann, wenn sie in der $|||\cdot|||$ Norm konvergiert. In diesem Sinne sind die Normen äquivalent. Da die Konstante $C$ sehr gross mit grossem $d$ wird, ist diese Äquivalenz für grosse Raumdimensionen $d$ mit entsprechender Vorsicht zu beachten.

**Beispiel 4.1.0.7 ($L^2$-Norm)**
Im Raum der quadratisch integrierbaren Funktionen $L^2$

$$L^2(0,1) = \{f : (0,1) \to \mathbb{R}, \int_0^1 |f(t)|^2 dt < \infty\}$$

haben wir die sehr wichtige Norm:

$$\|f\|_2 = \sqrt{\int_0^1 |f(t)|^2 dt} .$$

Diese Norm ist die natürliche Erweiterung der euklidischen Norm der Vektoren auf Funktionen. Die Summe über die Komponenten wird durch ein Integral über die Funktion ersetzt.

Das Skalarprodukt von $\mathbb{R}^d$ kann, wie die Norm, auch auf den Raum $L^2(0,1)$ erweitert werden. Das Skalarprodukt in $\mathbb{R}^d$ ist

$$\langle \mathbf{x}, \mathbf{y} \rangle = \sum_{j=1}^{d} \overline{x_j} y_j .$$

Für den Raum $L^2(0,1)$ definieren wir nun analog zur Definition der Norm $\|f\|_2$ das Skalarprodukt $\langle\cdot,\cdot\rangle$. Für dies ersetzen wir die Summen über das Produkt der Komponenten der Vektoren durch ein Integral über das Produkt der beiden Funktionen:

$$\langle f, g\rangle = \int_0^1 \overline{f(t)} g(t) dt$$

Im nächsten Abschnitt sehen wir die allgemeine Definition eines Skalarprodukts. ⌟

## Matrixnormen

Wenn wir in $\mathbb{R}^n$ eine Vektornorm haben, dann können wir eine assoziierte Matrixnorm definieren. Das entspricht genau der Norm eines linearen Operators, der der Matrix entspricht.

**Definition 4.1.0.8. Matrixnorm aus der Vektornorm**

Die Vektornorm $\|\cdot\|$ in $\mathbb{R}^n$ induziert im linearen Raum der $m \times n$-Matrizen eine Norm:

$$\|\mathbf{A}\| = \max_{\|x\|=1} \|\mathbf{A}\mathbf{x}\| .$$

Wir transformieren alle Vektoren der Länge eins mit der Matrix $\mathbf{A}$. Die Länge des längsten Resultats gibt die assoziierte Norm der Matrix. Das ist eine intuitive (und zur verwendeten Vektornorm relative) ''Grösse'' der Matrix, die angibt, wie viel die Matrix einen Vektor vergrössern könnte.

**Beispiel 4.1.0.9 (Matrixnormen, die von einer Vektornorm induziert sind)**
Hier sind zwei Beispiele von Matrixnormen, die von einer Vektornorm induziert sind:

$$\|\mathbf{A}\|_2 = \max_{\|x\|_2=1} \|\mathbf{A}\mathbf{x}\|_2 , \qquad \|\mathbf{A}\|_1 = \max_{\|x\|_1=1} \|\mathbf{A}\mathbf{x}\|_1 .$$

⌟

**Bemerkung 4.1.0.10 (Nicht alle Normen von Matrizen kommen aus einer Vektornorm)** [2]
Nicht alle Normen von Matrizen kommen aus einer Vektornorm. So zum Beispiel die *Frobenius-Norm* einer Matrix:

$$\|\mathbf{A}\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} |a_{ij}|^2} .$$

Das ist die euklidische Norm des Vektors, der entsteht, wenn man alle Einträge der Matrix in einen Vektor schreibt. Diese Matrixnorm kommt nicht aus einer Vektornorm, sondern ist von den einzelnen Einträgen der Matrix abhängig.

Hier sind die wichtigsten Normen, die von einer Vektornorm induziert worden sind:

**Die Zeilensummen-Norm $\|\cdot\|_\infty$:**

$$\|\mathbf{A}\|_\infty = \max_{\|x\|_\infty=1} \|\mathbf{A}\mathbf{x}\|_\infty = \max_{i=1,\ldots,m} \sum_{j=1}^{n} |a_{ij}| .$$

Für eine Matrix $\mathbf{A}$ würden wir dies folgendermassen berechnen:

$$\mathbf{A} = \begin{bmatrix} 1 & -2 & -3 \\ 2 & 3 & -1 \end{bmatrix} \implies \|\mathbf{A}\|_\infty = \max\{|1|+|-2|+|-3|,\ |2|+|3|+|-1|\} = 6 .$$

[2] Die Berechnung von Vektor- und Matixnormen mit Python wird in Abschnitt 9.4.1 behandelt.

**Die Spaltensummen-Norm $\|\cdot\|_1$:**

$$\|\mathbf{A}\|_1 = \max_{\|x\|_1=1} \|\mathbf{A}\mathbf{x}\|_1 = \max_{j=1,\dots,n} \sum_{i=1}^{m} |a_{ij}| .$$

Für eine Matrix $\mathbf{A}$ würden wir dies folgendermassen berechnen:

$$\mathbf{A} = \begin{bmatrix} 1 & -2 & -3 \\ 2 & 3 & -1 \end{bmatrix} \implies \|\mathbf{A}\|_1 = \max\{|1|+|2|,\ |-2|+|3|,\ |-3|+|1|\} = 5 .$$

**Die Spektral-Norm $\|\cdot\|_2$:**

$$\|\mathbf{A}\|_2 = \max_{\|x\|_2=1} \|\mathbf{A}\mathbf{x}\|_2 = \sqrt{\lambda_{\max}(\mathbf{A}^{\mathsf{H}}\mathbf{A})} ,$$

wobei $\lambda_{\max}$ der grösste Eigenwert von $\mathbf{A}^{\mathsf{H}}\mathbf{A}$ ist. Was ein Eigenwert ist und wie wir diesen berechnen können, werden wir im Kapitel 7.2 sehen. ⌟

## 4.2 Anwendung: Störungstheorem

### 4.2.1 Einleitung

Wenn wir in der Praxis ein Gleichungssystem $\mathbf{Ax} = \mathbf{b}$ lösen möchten, muss uns bewusst sein, dass dieses System oftmals fehlerbehaftet ist. Das ist dann der Fall, wenn $\mathbf{A}$ und $\mathbf{b}$ beispielsweise Ergebnisse von Messungen oder anderen numerischen Berechnungen sind. Also arbeiten wir eigentlich mit einem gestörten System $(\mathbf{A} + \Delta\mathbf{A})\tilde{\mathbf{x}} = (\mathbf{b} + \Delta\mathbf{b})$.

Nun stellt sich die für Anwendungen wesentliche Frage: *In welchem Zusammenhang stehen die gestörte Lösung $\tilde{\mathbf{x}}$ und die Lösung $\mathbf{x}$*. Insbesondere, darf die gestörte Lösung $\tilde{\mathbf{x}}$ überhaupt herangezogen werden, um praktische Fragen zu beantworten, bei denen die Lösung $\mathbf{x}$ benötigt wird?

### 4.2.2 Herleitung

Um diese Fragen zu beantworten, benötigen wir das folgende Theorem:

**Satz 4.2.2.1. Störungstheorem**

*Sei $\mathbf{B} \in \mathbb{R}^{n\times n}$, sodass $\|\mathbf{B}\| < 1$ wobei $\|\cdot\|$ eine aus einer Vektornorm kommende Norm ist.*
*Dann gilt:*

1. $(\mathbf{I} + \mathbf{B})$ *ist regulär*
2. $\left\|(\mathbf{I}+\mathbf{B})^{-1}\right\| \leq \frac{1}{1-\|\mathbf{B}\|}$

*Beweis.*

$$\begin{aligned} \|\mathbf{x}\| = \|(\mathbf{I}+\mathbf{B})\mathbf{x} - \mathbf{Bx}\| &\leq \|(\mathbf{I}+\mathbf{B})\mathbf{x}\| + \|\mathbf{Bx}\| && \text{(Dreiecksungleichung)} \\ \Rightarrow \|\mathbf{x}\| - \|\mathbf{Bx}\| &\leq \|(\mathbf{I}+\mathbf{B})\mathbf{x}\| && (4.2.2.2) \end{aligned}$$

Aus der Definition der Norm folgt:

$$\begin{aligned} \|\mathbf{B}\| &= \sup_{\mathbf{x}\neq 0} \frac{\|\mathbf{Bx}\|}{\|\mathbf{x}\|} \\ \Longrightarrow \|\mathbf{B}\| &\geq \frac{\|\mathbf{Bx}\|}{\|\mathbf{x}\|} && (\forall x \neq 0, \text{ def. sup}) \\ \Longrightarrow \|\mathbf{Bx}\| &\leq \|\mathbf{B}\|\|\mathbf{x}\| \\ \Longrightarrow -\|\mathbf{Bx}\| &\geq -\|\mathbf{B}\|\|\mathbf{x}\| \\ \Longrightarrow \|x\| - \|\mathbf{Bx}\| &\geq \|x\| - \|\mathbf{B}\|\|\mathbf{x}\| = \|\mathbf{x}\|(1 - \|\mathbf{B}\|) && (4.2.2.3) \end{aligned}$$

Aus (4.2.2.2) und (4.2.2.3) folgt mittels Transitivität:

$$\|(\mathbf{I}+\mathbf{B})\mathbf{x}\| \geq \|\mathbf{x}\|(1 - \|\mathbf{B}\|) \tag{4.2.2.4}$$

Definitionsgemäss gilt auch:

$$x \in \operatorname{Kern}(\mathbf{I}+\mathbf{B}) \iff (\mathbf{I}+\mathbf{B})\mathbf{x} = 0 \tag{4.2.2.5}$$

Aus (4.2.2.4) und (4.2.2.5) folgt also wegen $1 - \|\mathbf{B}\| > 0$:

$$x \in \operatorname{Kern}(\mathbf{I}+\mathbf{B}) \iff \|\mathbf{x}\|(1 - \|\mathbf{B}\|) = 0 \iff \|\mathbf{x}\| = 0 \iff \mathbf{x} = 0$$

Somit folgt:

$$\begin{aligned} &\operatorname{Kern}(\mathbf{I}+\mathbf{B}) = \{0\} \\ \iff &\mathbf{I}+\mathbf{B} \text{ injektiv} && (3.2.0.8) \\ \implies &\mathbf{I}+\mathbf{B} \text{ bijektiv} && (3.2.0.9) \\ \implies &\mathbf{I}+\mathbf{B} \text{ regulär} \end{aligned}$$

Somit ist die erste Aussage des Satzes bewiesen.

Sei $\mathbf{y} := (\mathbf{I}+\mathbf{B})^{-1}\mathbf{x} \iff \mathbf{x} = (\mathbf{I}+\mathbf{B})\mathbf{y}$.

$$\begin{aligned} \left\|(\mathbf{I}+\mathbf{B})^{-1}\right\| &= \sup_{\mathbf{x}\neq 0} \frac{\left\|(\mathbf{I}+\mathbf{B})^{-1}\mathbf{x}\right\|}{\|\mathbf{x}\|} \\ &= \sup_{\mathbf{y}\neq 0} \frac{\|\mathbf{y}\|}{\|(\mathbf{I}+\mathbf{B})\mathbf{y}\|} \qquad (\mathbf{x}\neq 0 \iff y \neq 0 \text{ da } (\mathbf{I}+\mathbf{B}) \text{ regulär}) \qquad (4.2.2.6) \end{aligned}$$

Aus (4.2.2.4) folgt weiters durch Substitution von $\mathbf{x}$ mit $\mathbf{y}$:

$$\begin{aligned} &\|(\mathbf{I}+\mathbf{B})\mathbf{y}\| \geq \|\mathbf{y}\|(1-\|\mathbf{B}\|) \\ \implies &1 \geq \frac{\|\mathbf{y}\|}{\|(\mathbf{I}+\mathbf{B})\mathbf{y}\|}(1-\|\mathbf{B}\|) \\ \implies &\frac{\|\mathbf{y}\|}{\|(\mathbf{I}+\mathbf{B})\mathbf{y}\|} \leq \frac{1}{1-\|\mathbf{B}\|} \qquad (4.2.2.7) \end{aligned}$$

Durch Verwendung von 4.2.2.7 in 4.2.2.6 gemeinsam mit der Definition von sup schliessen wir:

$$\left\|(\mathbf{I}+\mathbf{B})^{-1}\right\| \leq \frac{1}{1-\|\mathbf{B}\|}$$

Damit ist auch die zweite Aussage des Satzes und somit dieser selbst bewiesen. □

**Korollar 4.2.2.8.**

*Sei $\mathbf{A} \in \mathbb{R}^{n\times n}$ eine invertierbare Matrix und $\mathbf{B} \in \mathbb{R}^{n\times n}$, sodass $\left\|\mathbf{A}^{-1}\mathbf{B}\right\| < 1$ wobei $\|\cdot\|$ eine aus einer Vektornorm kommende Norm ist.*
*Dann gilt:*

1. $(\mathbf{A}+\mathbf{B})$ *ist regulär*
2. $\left\|(\mathbf{A}+\mathbf{B})^{-1}\right\| \leq \frac{\|\mathbf{A}^{-1}\|}{1-\|\mathbf{A}^{-1}\mathbf{B}\|}$

*Beweis.* Um auf das Störungstheorem 4.2.2.1 zu reduzieren, schreiben wir:

$$(\mathbf{A}+\mathbf{B}) = \mathbf{A}(\mathbf{I}+\mathbf{A}^{-1}\mathbf{B})$$

Daraus folgt unmittelbar:

$$(\mathbf{A}+\mathbf{B})^{-1} = (\mathbf{I}+\mathbf{A}^{-1}\mathbf{B})^{-1}\mathbf{A}^{-1}$$

Aus der Annahme $\left\|\mathbf{A}^{-1}\mathbf{B}\right\| < 1$ folgt mit dem Störungstheorem 4.2.2.1, dass $(\mathbf{I}+\mathbf{A}^{-1}\mathbf{B})$ invertierbar ist und die folgende Abschätzung gilt:

$$\left\|(\mathbf{I}+\mathbf{A}^{-1}\mathbf{B})^{-1}\right\| \leq \frac{1}{1-\left\|\mathbf{A}^{-1}\mathbf{B}\right\|}$$

Durch Anwendung von Eigenschaften der Norm gilt also auch:

$$\left\|(\mathbf{A}+\mathbf{B})^{-1}\right\| \leq \frac{\left\|\mathbf{A}^{-1}\right\|}{1-\left\|\mathbf{A}^{-1}\mathbf{B}\right\|}$$

Damit ist dieses Korollar bewiesen. □

### 4.2.3 Anwendung

Nun können wir uns der eingangs gestellte Frage widmen. Zur Erinnerung: Wir betrachten ein gestörtes System $(\mathbf{A}+\Delta\mathbf{A})\tilde{\mathbf{x}} = (\mathbf{b}+\Delta\mathbf{b})$ und möchten wissen, welche Schlüsse sich daraus für das System $\mathbf{Ax} = \mathbf{b}$ ableiten lassen. Insbesondere, möchten wir eine Abschätzung für den relativen Fehler der Lösung erhalten.

Durch Umformungen erhalten wir den folgenden Ausdruck:

$$\begin{aligned}
&(\mathbf{A}+\Delta\mathbf{A})\tilde{\mathbf{x}} = (\mathbf{b}+\Delta\mathbf{b}) \\
\Longrightarrow\ &(\mathbf{A}+\Delta\mathbf{A})\tilde{\mathbf{x}} = (\mathbf{Ax}+\Delta\mathbf{b}) \\
\Longrightarrow\ &(\mathbf{A}+\Delta\mathbf{A})\tilde{\mathbf{x}} - (\mathbf{A}+\Delta\mathbf{A})\mathbf{x} = (\mathbf{Ax}+\Delta\mathbf{b}) - (\mathbf{A}+\Delta\mathbf{A})\mathbf{x} \\
\Longrightarrow\ &(\mathbf{A}+\Delta\mathbf{A})(\tilde{\mathbf{x}}-\mathbf{x}) = \Delta\mathbf{b} - \Delta\mathbf{Ax}
\end{aligned}$$

Wenn wir zusätzlich $\tilde{\mathbf{x}}-\mathbf{x}$ als $\Delta\mathbf{x}$ notieren können wir folgendes bereits aus den Definitionen schliessen.

$$\Delta\mathbf{x} = (\mathbf{A}+\Delta\mathbf{A})^{-1}(\Delta\mathbf{b}-\Delta\mathbf{Ax})$$

Insbesondere, gilt also folgende Abschätzung hinsichtlich der Norm:

$$\begin{aligned}
\|\Delta\mathbf{x}\| &= \left\|(\mathbf{A}+\Delta\mathbf{A})^{-1}\right\|\|(\Delta\mathbf{b}-\Delta\mathbf{Ax})\| \\
&\leq \left\|(\mathbf{A}+\Delta\mathbf{A})^{-1}\right\|(\|\Delta\mathbf{b}\|+\|\Delta\mathbf{Ax}\|) \qquad \text{(Dreiecksungleichung)} \\
&\leq \left\|(\mathbf{A}+\Delta\mathbf{A})^{-1}\right\|(\|\Delta\mathbf{b}\|+\|\Delta\mathbf{A}\|\|\mathbf{x}\|)
\end{aligned}$$

Unter der Annahme, dass $\|\Delta\mathbf{A}\| < \frac{1}{\|\mathbf{A}^{-1}\|}$ können wir das Korollar des Störungstheorems anwenden, welches uns folgende Abschätzung liefert:

$$\left\|(\mathbf{A}+\Delta\mathbf{A})^{-1}\right\| \leq \frac{\left\|\mathbf{A}^{-1}\right\|}{1-\left\|\mathbf{A}^{-1}\Delta\mathbf{A}\right\|}$$

Eingesetzt die vorherige Abschätzung erhalten wir:

$$\begin{aligned}\|\Delta\mathbf{x}\| &\leq \frac{\left\|\mathbf{A}^{-1}\right\|}{1-\left\|\mathbf{A}^{-1}\Delta\mathbf{A}\right\|}(\|\Delta\mathbf{b}\|+\|\Delta\mathbf{A}\|\|\mathbf{x}\|)\\ &= \frac{\left\|\mathbf{A}^{-1}\right\|}{1-\left\|\mathbf{A}^{-1}\Delta\mathbf{A}\right\|}(\frac{\|\Delta\mathbf{b}\|}{\|\mathbf{x}\|}+\|\Delta\mathbf{A}\|)\|\mathbf{x}\|\end{aligned}$$

Mit einer weiteren Abschätzung

$$\begin{aligned}&\left\|\mathbf{A}^{-1}\Delta\mathbf{A}\right\| \leq \left\|\mathbf{A}^{-1}\right\|\|\Delta\mathbf{A}\|\\ \Longrightarrow\ &-\left\|\mathbf{A}^{-1}\Delta\mathbf{A}\right\| \geq -\left\|\mathbf{A}^{-1}\right\|\|\Delta\mathbf{A}\|\\ \Longrightarrow\ &1-\left\|\mathbf{A}^{-1}\Delta\mathbf{A}\right\| \geq 1-\left\|\mathbf{A}^{-1}\right\|\|\Delta\mathbf{A}\|\\ \Longrightarrow\ &\frac{1}{1-\left\|\mathbf{A}^{-1}\Delta\mathbf{A}\right\|} \leq \frac{1}{1-\left\|\mathbf{A}^{-1}\right\|\|\Delta\mathbf{A}\|}\\ \Longrightarrow\ &\frac{\left\|\mathbf{A}^{-1}\right\|}{1-\left\|\mathbf{A}^{-1}\Delta\mathbf{A}\right\|} \leq \frac{\left\|\mathbf{A}^{-1}\right\|}{1-\left\|\mathbf{A}^{-1}\right\|\|\Delta\mathbf{A}\|}\end{aligned}$$

erhalten wir:

$$\begin{aligned}\|\Delta\mathbf{x}\| &\leq \frac{\left\|\mathbf{A}^{-1}\right\|}{1-\left\|\mathbf{A}^{-1}\right\|\|\Delta\mathbf{A}\|}\left(\frac{\|\Delta\mathbf{b}\|}{\|\mathbf{x}\|}+\|\Delta\mathbf{A}\|\right)\|\mathbf{x}\|\\ &= \frac{\left\|\mathbf{A}^{-1}\right\|\|\mathbf{A}\|}{1-\left\|\mathbf{A}^{-1}\right\|\|\Delta\mathbf{A}\|}\left(\frac{\|\Delta\mathbf{b}\|}{\|\mathbf{x}\|}+\|\Delta\mathbf{A}\|\right)\frac{\|\mathbf{x}\|}{\|\mathbf{A}\|} \qquad \left(\cdot\frac{\|\mathbf{A}\|}{\|\mathbf{A}\|}\right)\\ &= \frac{\left\|\mathbf{A}^{-1}\right\|\|\mathbf{A}\|}{1-\left\|\mathbf{A}^{-1}\right\|\|\Delta\mathbf{A}\|}\left(\frac{\|\Delta\mathbf{b}\|}{\|\mathbf{A}\|\|\mathbf{x}\|}+\frac{\|\Delta\mathbf{A}\|}{\|\mathbf{A}\|}\right)\|\mathbf{x}\|\end{aligned}$$

Unter Verwendung der aus der Definition der Norm folgenden Identität, welche bereits zuvor genauer betrachtet wurde.

$$\begin{aligned}&\|\mathbf{A}\mathbf{x}\| \leq \|\mathbf{A}\|\|\mathbf{x}\|\\ \Longrightarrow\ &\frac{1}{\|\mathbf{A}\mathbf{x}\|} \geq \frac{1}{\|\mathbf{A}\|\|\mathbf{x}\|}\end{aligned}$$

erhalten wir:

$$\begin{aligned}\|\Delta\mathbf{x}\| &\leq \frac{\left\|\mathbf{A}^{-1}\right\|\|\mathbf{A}\|}{1-\left\|\mathbf{A}^{-1}\right\|\|\Delta\mathbf{A}\|}\left(\frac{\|\Delta\mathbf{b}\|}{\|\mathbf{A}\mathbf{x}\|}+\frac{\|\Delta\mathbf{A}\|}{\|\mathbf{A}\|}\right)\|\mathbf{x}\|\\ &\leq \frac{\left\|\mathbf{A}^{-1}\right\|\|\mathbf{A}\|}{1-\left\|\mathbf{A}^{-1}\right\|\|\Delta\mathbf{A}\|}\left(\frac{\|\Delta\mathbf{b}\|}{\|\mathbf{b}\|}+\frac{\|\Delta\mathbf{A}\|}{\|\mathbf{A}\|}\right)\|\mathbf{x}\|\end{aligned}$$

$$\leq \frac{\left\|\mathbf{A}^{-1}\right\|\|\mathbf{A}\|}{1-\left\|\mathbf{A}^{-1}\right\|\|\mathbf{A}\|\frac{\|\Delta\mathbf{A}\|}{\|\mathbf{A}\|}}\left(\frac{\|\Delta\mathbf{b}\|}{\|\mathbf{b}\|}+\frac{\|\Delta\mathbf{A}\|}{\|\mathbf{A}\|}\right)\|\mathbf{x}\|$$

Somit haben wir also die gesuchte Abschätzung für den relativen Fehler der Lösung gefunden.

$$\frac{\|\Delta\mathbf{x}\|}{\|\mathbf{x}\|} \leq \frac{\left\|\mathbf{A}^{-1}\right\|\|\mathbf{A}\|}{1-\left\|\mathbf{A}^{-1}\right\|\|\mathbf{A}\|\frac{\|\Delta\mathbf{A}\|}{\|\mathbf{A}\|}}\left(\frac{\|\Delta\mathbf{b}\|}{\|\mathbf{b}\|}+\frac{\|\Delta\mathbf{A}\|}{\|\mathbf{A}\|}\right) \tag{4.2.3.1}$$

Bemerke, dass die Ausdrücke $\frac{\|\Delta\mathbf{A}\|}{\|\mathbf{A}\|}$ und $\frac{\|\Delta\mathbf{b}\|}{\|\mathbf{b}\|}$ die relativen Fehler in den (Mess-)Daten angeben.

Anhand der Abschätzung für den relativen Fehler der Lösung ist ebenfalls klar, dass der Ausdruck $\left\|\mathbf{A}^{-1}\right\|\|\mathbf{A}\|$ von entscheidender Bedeutung für die Qualität der Lösung ist. Dieser Ausdruck heisst die Konditionszahl $k(\mathbf{A})$ der Matrix $\mathbf{A}$. Bemerkenswert ist, dass die Konditionszahl bereits durch die Natur des Systems bestimmt ist und insbesondere unabhängig von den Fehlern in den (Mess-)Daten ist.

Es gibt also Probleme, die gut konditioniert sind und Probleme die schlecht konditioniert sind.

### 4.2.4 Beispiel: Geradenschnittpunkt

Um das Konzept der Konditionszahl und das von gut bzw. schlecht konditionierten Problemen zu veranschaulichen betrachten wir das Problem, den Schnittpunkt zweier Geraden im zwei-dimensionalen Raum zu finden.

Der Einfachheit halber werden wir in den folgenden Beispielen nur $\mathbf{b}$ stören. In diesem Fall reduziert sich die Abschätzung für den relativen Fehler der Lösung auf:

$$\frac{\|\Delta\mathbf{x}\|}{\|\mathbf{x}\|} \leq k(\mathbf{A})\frac{\|\Delta\mathbf{b}\|}{\|\mathbf{b}\|}$$

Zudem werden wir die Zeilensummen-Norm $\|\cdot\|_\infty$ verwenden.

#### 4.2.4.1 Gut Konditionierter Fall

Der bestmöglich konditionierte Fall ist, dass die zwei Geraden orthogonal aufeinander stehen.

Betrachten wir also die Geraden $x_1 = 1$ und $x_2 = 1$. Daraus ergibt sich ein Gleichungssystem mit:

$$\mathbf{A} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad \mathbf{b} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \qquad \Longrightarrow \ \mathbf{x} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

Somit gilt:

$$\begin{aligned} \mathbf{A}^{-1} &= \mathbf{A} = \mathbf{I} \\ k(\mathbf{A}) &= \left\|\mathbf{A}^{-1}\right\|_\infty \|\mathbf{A}\|_\infty = 1 \cdot 1 = 1 \end{aligned}$$

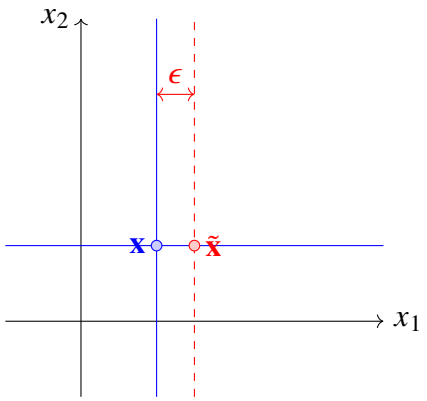
Wir betrachten nun stattdessen ein gestörtes System mit:

$$\tilde{\mathbf{A}} = \mathbf{A} \qquad \tilde{\mathbf{b}} = \mathbf{b} + \begin{bmatrix} \epsilon \\ 0 \end{bmatrix}$$

Dann muss gelten für das definierte System gelten:

$$\begin{aligned} & \frac{\|\Delta \mathbf{x}\|_\infty}{\|\mathbf{x}\|_\infty} \leq k(\mathbf{A}) \frac{\|\Delta \mathbf{b}\|_\infty}{\|\mathbf{b}\|_\infty} \\ \Longrightarrow\ & \|\Delta \mathbf{x}\|_\infty \leq \|\Delta \mathbf{b}\|_\infty \qquad (k(\mathbf{A}) = 1,\ \|\mathbf{b}\|_\infty = 1 \text{ und } \|\mathbf{x}\|_\infty = 1) \\ \Longrightarrow\ & \|\Delta \mathbf{x}\|_\infty \leq \epsilon \end{aligned}$$

In diesem Fall ist also der Fehler in der Lösung nie grösser als der Fehler in den Daten. Wir haben es also hier tatsächlich mit einem sehr glücklichen Fall zu tun.




#### 4.2.4.2 Schlecht Konditionierter Fall

Einen schlecht konditionierten Fall erhalten wir, wenn die beiden Geraden beinahe parallel sind.

Betrachten wir also die Geraden $x_2 = 1$ und $\delta x_1 + x_2 = 1$. Daraus ergibt sich ein Gleichungssystem mit:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ \delta & 1 \end{bmatrix} \qquad \mathbf{b} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \qquad \Longrightarrow\ \mathbf{x} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Somit gilt:

$$\begin{aligned} \mathbf{A}^{-1} &= \begin{bmatrix} -\frac{1}{\delta} & \frac{1}{\delta} \\ 1 & 0 \end{bmatrix} \\ \|\mathbf{A}\|_\infty &= 1 + \delta \\ \left\|\mathbf{A}^{-1}\right\|_\infty &= \frac{2}{\delta} \\ k(\mathbf{A}) &= \left\|\mathbf{A}^{-1}\right\|_\infty \|\mathbf{A}\|_\infty = \frac{2(1+\delta)}{\delta} = \frac{2}{\delta} + 2 \end{aligned}$$

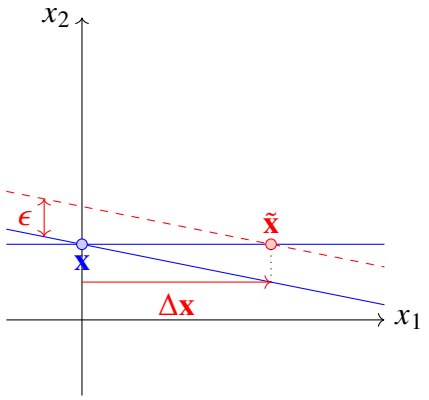
Betrachten wir nun erneut ein gestörtes System mit:

$$\tilde{\mathbf{A}} = \mathbf{A} \qquad \tilde{\mathbf{b}} = \mathbf{b} + \begin{bmatrix} 0 \\ \epsilon \end{bmatrix}$$

Dann muss für das definierte System gelten:

$$\begin{aligned} & \frac{\|\Delta\mathbf{x}\|_\infty}{\|\mathbf{x}\|_\infty} \leq k(\mathbf{A}) \frac{\|\Delta\mathbf{b}\|_\infty}{\|\mathbf{b}\|_\infty} \\ \Longrightarrow\ & \|\Delta\mathbf{x}\|_\infty \leq \left(\frac{2}{\delta} + 2\right) \|\Delta\mathbf{b}\|_\infty \qquad \left(k(\mathbf{A}) = \frac{2}{\delta} + 2,\ \|\mathbf{b}\|_\infty = 1 \text{ und } \|\mathbf{x}\|_\infty = 1\right) \\ \Longrightarrow\ & \|\Delta\mathbf{x}\|_\infty \leq \left(\frac{2}{\delta} + 2\right) \epsilon \end{aligned} \tag{4.2.4.1}$$

Wir befinden uns also in einer völlig anderen Situation als zuvor. Falls $\delta \ll 1$, wird der Fehler im Ergebnis unbeherrschbar.



## 4.3 Skalarprodukt in linearen Räumen

**Definition 4.3.0.1. Skalarprodukt**

Sei $V$ ein linearer Raum.
Ein *Skalarprodukt* $\langle\cdot,\cdot\rangle$ ist eine Funktion von zwei Variablen in diesem Raum $V$

$$\langle\cdot,\cdot\rangle : V \times V \to \mathbb{R}/\mathbb{C}$$

welche die folgenden drei Eigenschaften erfüllt:

(S1) Die Funktion ist *linear im zweiten Argument* [3]:

$$\langle x, ay + bz\rangle = a\langle x, y\rangle + b\langle x, z\rangle$$

für alle $x,\ y,\ z \in V$ und $a,\ b$ Skalare;

(S2) Die Funktion ist *symmetrisch*:

$$\langle x, y\rangle = \langle y, x\rangle \text{ über } \mathbb{R}$$
$$\langle x, y\rangle = \overline{\langle y, x\rangle} \text{ über } \mathbb{C}\,;$$

(S3) Die Funktion ist *positiv definit*:

$$\langle x, x\rangle \geq 0 \text{ für alle } x \in V$$
$$\langle x, x\rangle = 0 \implies x = 0\,;$$

falls nur der erste Bedingung erfüllt ist,
so sprechen wir von *positiv semi-definit*.

**Beispiel 4.3.0.2 (Das euklidische und das $L^2$-Skalarprodukt )**
Bis jetzt haben wir bereits zwei Skalarprodukte gesehen, das euklidische Skalarprodukt und das Skalarprodukt im Raum der quadratisch integrierbaren Funktionen $L^2(0,1)$:

$$\langle \mathbf{x}, \mathbf{y}\rangle = \sum_{j=1}^{d} x_j y_j, \qquad \langle f, g\rangle = \int_0^1 \overline{f(t)} g(t) dt\ .$$

⌟

**Bemerkung 4.3.0.3 (Vom Skalarprodukt zur Norm)**
Wir sehen, dass $\|\mathbf{x}\|_2 = \sqrt{\langle \mathbf{x}, \mathbf{x}\rangle}$ und $\|f\|_2 = \sqrt{\langle f, f\rangle}$. Wir können diese zwei Normen also auch mit dem Skalarprodukt beschreiben. ⌟

**Definition 4.3.0.4. Norm aus einem Skalarprodukt**

Man sagt, dass *eine Norm* $\|\cdot\|$ *von einem Skalarprodukt* $\langle\cdot,\cdot\rangle$ *induziert wird*, falls

$$\|x\| = \sqrt{\langle x, x\rangle}$$

für alle Elemente $x$ eines linearen Raumes $X$.

[3] Manchmal wird stattdessen die Linearität im ersten Argument gefordert. Diese alternative Konvention wird in der Mathematik verwendet, während sich diese Vorlesung nach jener Konvention richtet, welche unter anderem in der Physik bevorzugt wird.

**Definition 4.3.0.5. Orthogonalität von Elementen eines linearen Raumes mit Skalarprodukt**

Sei $x,\ y \in V$, wobei $V$ ein linearer Raum mit einem Skalarprodukt $\langle\cdot,\cdot\rangle$ ist.
Wir sagen, dass $x$ und $y$ orthogonal ($x \perp y$) sind, falls $\langle x, y\rangle = 0$.

**Beispiel 4.3.0.6 (A-Skalarprodukt)**
In diesem Beispiel führen wir das **A**-*Skalarprodukt* für den Raum $\mathbb{R}^n$ ein, welches viele wichtige Anwendungen hat.

Sei $V = \mathbb{R}^n$ ein linearer Raum und $\mathbf{Q} \in \mathbb{R}^{n\times n}$ eine orthogonale Matrix und $\mathbf{\Lambda}$ eine Diagonalmatrix mit den Einträgen $\lambda_1,\ \lambda_2, \ldots, \lambda_n > 0$.

$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix}.$$

Mit diesen beiden Matrizen können wir eine neue Matrix $\mathbf{A}$,

$$\mathbf{A} = \mathbf{Q}^\mathsf{T}\,\mathbf{\Lambda}\,\mathbf{Q}\,,$$

definieren. Diese neue Matrix $\mathbf{A}$ ist offensichtlich symmetrisch, da $\mathbf{\Lambda} = \mathbf{\Lambda}^\mathsf{T}$ gilt

$$\begin{aligned}\mathbf{A}^\mathsf{T} &= \mathbf{Q}^\mathsf{T}\,\mathbf{\Lambda}\,(\mathbf{Q}^\mathsf{T})^\mathsf{T} \\ &= \mathbf{Q}^\mathsf{T}\,\mathbf{\Lambda}\,\mathbf{Q} \\ &= \mathbf{A}\,.\end{aligned}$$

Mit dieser Matrix definieren wir das $\mathbf{A}$-Skalarprodukt $\langle\cdot,\cdot\rangle_\mathbf{A}$ zwischen zwei beliebigen Vektoren $\mathbf{u}$ und $\mathbf{v}$ in $\mathbb{R}^n$ als

$$\langle \mathbf{u}, \mathbf{v}\rangle_\mathbf{A} \overset{\text{def}}{=} \mathbf{u}^\mathsf{T}\mathbf{A}\mathbf{v} \in \mathbb{R}\,.$$

Wir müssen nun noch beweisen, dass $\langle\cdot,\cdot\rangle_\mathbf{A} :\ \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ tatsächlich ein Skalarprodukt in $\mathbb{R}^n$ ist.

*Beweis.* Wir müssen zeigen, dass $\langle\cdot,\cdot\rangle_\mathbf{A}$ die drei Bedingungen (S1), (S2) und (S3) erfüllt.

**(S1):**
Dass (S1) erfüllt wird, kommt aus der Linearität der Matrix-Vektor-Multiplikation:

$$\begin{aligned}\langle \mathbf{u}, a\mathbf{v} + b\mathbf{w}\rangle_\mathbf{A} &= \mathbf{u}^\mathsf{T}\,\mathbf{A}(a\mathbf{v} + b\mathbf{w}) \\ &= a\mathbf{u}^\mathsf{T}\mathbf{A}\mathbf{v} + b\mathbf{u}^\mathsf{T}\mathbf{A}\mathbf{w} \\ &= a\langle \mathbf{u}, \mathbf{v}\rangle_\mathbf{A} + b\langle \mathbf{u}, \mathbf{w}\rangle_\mathbf{A}\,.\end{aligned}$$

**(S2):**
Hier benutzen wir, dass die Transponierte eines Skalars gleich dem Skalar selber ist, $a = a^\mathsf{T} \in \mathbb{R}$. Da das Skalarprodukt in $\mathbb{R}$ liegt, gilt also $\langle\cdot,\cdot\rangle = \langle\cdot,\cdot\rangle^\mathsf{T}$. Auch benutzen wir, dass $\mathbf{A}$ symmetrisch ist ($\mathbf{A}^\mathsf{T} = \mathbf{A}$):

$$\begin{aligned}\langle \mathbf{u}, \mathbf{v}\rangle_\mathbf{A} &= \langle \mathbf{u}, \mathbf{v}\rangle_\mathbf{A}^\mathsf{T} \\ &= (\mathbf{u}^\mathsf{T}\mathbf{A}\mathbf{v})^\mathsf{T} \\ &= \mathbf{v}^\mathsf{T}\mathbf{A}^\mathsf{T}(\mathbf{u}^\mathsf{T})^\mathsf{T} \\ &= \mathbf{v}^\mathsf{T}\mathbf{A}\mathbf{u} \\ &= \langle \mathbf{v}, \mathbf{u}\rangle_\mathbf{A}\end{aligned}$$

Somit ist die Eigenschaft (S2) auch erfüllt und wir müssen nur noch zeigen, dass auch (S3) erfüllt ist.

**(S3):**
Sei $\mathbf{u}$ beliebig und notieren wir $\mathbf{v} = \mathbf{Q}\mathbf{u}$, so dass:

$$\begin{aligned}\langle \mathbf{u}, \mathbf{u} \rangle_{\mathbf{A}} &= \mathbf{u}^{\mathsf{T}} \mathbf{A} \mathbf{u} \\ &= \mathbf{u}^{\mathsf{T}} \mathbf{Q}^{\mathsf{T}} \mathbf{\Lambda} \underbrace{\mathbf{Q}\mathbf{u}}_{\mathbf{v}} \\ &= \mathbf{v}^{\mathsf{T}} \mathbf{\Lambda} \mathbf{v} \,.\end{aligned}$$

Wir haben nun also einen neuen Ausdruck für $\langle \mathbf{u}, \mathbf{u} \rangle$ erhalten, in der die Matrix $\mathbf{\Lambda}$ vorkommt. Da diese eine Diagonalmatrix ist, können wir diese Matrixmultiplikation als die einfache Summe

$$\mathbf{v}^{\mathsf{T}} \mathbf{\Lambda} \mathbf{v} = \sum_{j=1}^{n} \lambda_j |v_j|^2$$

schreiben. Diese Summe ist immer grösser gleich null für alle $\mathbf{u}$, da $\lambda_j > 0$ und $|v_j|^2 \geq 0$. Wir müssen noch zeigen, dass die Summe nur null ist, wenn $\mathbf{u}$ null ist. Da alle $\lambda_j$ strikt grösser null sind, kann diese Summe nur dann Null sein, wenn alle $v_j$ null sind, d. h. wenn $\mathbf{v} = \mathbf{0}$ und demzufolge $\mathbf{u} = \mathbf{Q}^{\mathsf{T}}\mathbf{v} = \mathbf{0}$.

Wir haben also gezeigt, dass die Funktion $\langle \cdot, \cdot \rangle_{\mathbf{A}}$ alle drei Bedingungen eines Skalarproduktes erfüllt; deswegen ist sie ein Skalarprodukt.

□

Mit diesem $\mathbf{A}$-Skalarprodukt können wir auch die *induzierte* $\mathbf{A}$*-Norm definieren*:

$$\|\mathbf{u}\|_{\mathbf{A}}^2 = \langle \mathbf{u}, \mathbf{u} \rangle_{\mathbf{A}} \,.$$

⌟

**Beispiel 4.3.0.7 (Semi-positiv definierte bilineare Funktion)**
Was geschieht, wenn $\mathbf{\Lambda}$ eine Null auf der Diagonale hat? Z. B. nehmen wir an:

$$\mathbf{\Lambda} = \begin{bmatrix} 0 & 0 & \cdots & \cdots & 0 \\ 0 & \lambda_2 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & \\ 0 & \cdots & \cdots & 0 & \lambda_n \end{bmatrix}.$$

Dann bleibt alles, was bisher gezeigt wurde gleich, ausser den Bedingungen für $\langle \mathbf{u}, \mathbf{u} \rangle_{\mathbf{A}} = 0$.

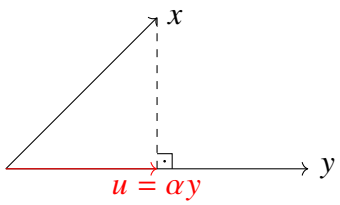
$$\begin{aligned}\langle \mathbf{u}, \mathbf{u} \rangle_{\mathbf{A}} = 0 &\overset{(\lambda_1 = 0)}{\iff} \sum_{j=2}^{n} \lambda_j |v_j|^2 = 0 \\ &\overset{(\lambda_j > 0, j = 2, \ldots, n)}{\iff} v_2, \ldots, v_n = 0 \\ &\iff \mathbf{u} = \mathbf{Q}^{\mathsf{T}} \begin{bmatrix} v_1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \text{ mit } v_1 \text{ beliebig.}\end{aligned}$$

Wenn $\lambda_1$ null ist, dann kann der Wert von $v_1$ beliebig sein und dies bedeutet, dass $\langle \cdot, \cdot \rangle_{\mathbf{A}}$ nur noch semi-positiv definit ist und deswegen kein Skalarprodukt mehr ist.

⌟

## Orthogonale Projektionen

**Bemerkung 4.3.0.8 (Berechnung der orthogonalen Projektion in einem linearen Raum mit Skalarprodukt)**
Wie berechnet man eine orthogonale Projektion in einem linearen Raum $V$ mit Skalarprodukt $\langle \cdot, \cdot \rangle$? Es ist nichts anderes als das, was wir in $\mathbb{R}^n$ gesehen haben.




Die Projektion von $x$ auf $y$, mit $y \neq 0$, ist $u = \alpha y$.
Für die orthogonale Projektion $u$ gilt $x - u \perp y$, d. h. $\langle y, x - u \rangle = 0$. Das Skalarprodukt ist linear im zweiten Argument und deswegen ist dies dasselbe wie $\langle y, x \rangle - \langle y, u \rangle = 0$. Mit diesem Ausdruck können wir nun weiterrechnen:

$$\begin{aligned} \langle y, x \rangle &= \langle y, u \rangle \\ &= \langle y, \alpha y \rangle \\ &= \alpha \langle y, y \rangle . \end{aligned}$$

Da $y \neq 0$ ist und wegen der positiven Definitheit des Skalarprodukts gilt, dass $\langle y, y \rangle \neq 0$. Deswegen können wir durch $\langle y, y \rangle$ teilen und erhalten einen Ausdruck für $\alpha$:

$$\alpha = \frac{\langle y, x \rangle}{\langle y, y \rangle} .$$

Somit können wir die orthogonale Projektion von $x$ auf $y$ definieren.

⌟

**Definition 4.3.0.9. Orthogonale Projektion auf einen Vektor**

Sei $V$ ein linearer Raum mit einem Skalarprodukt $\langle \cdot, \cdot \rangle$. So ist die orthogonale Projektion $P_y x$ von $x \in V$ auf $y \in V$, für $y \neq 0$, definiert als

$$P_y x = \frac{\langle y, x \rangle}{\langle y, y \rangle} y .$$

Diese Funktion $P_y : V \to V$ ist eine lineare Abbildung.

**Bemerkung 4.3.0.10 (Orthogonale Projektion in $\mathbb{R}^n$)**
Als spezielle Anwendung der orthogonalen Projektion $P_{\mathbf{y}}\mathbf{x}$ betrachten wir diese im linearen Raum $\mathbb{R}^n$. Es

ist also die Projektion eines Vektors $\mathbf{x}$ auf einen Vektor $\mathbf{y}$:

$$\begin{aligned} P_{\mathbf{y}}\mathbf{x} &= \frac{\langle \mathbf{y}, \mathbf{x}\rangle}{\langle \mathbf{y}, \mathbf{y}\rangle}\mathbf{y} \\ &= \frac{\mathbf{y}^{\mathsf{H}}\mathbf{x}}{\|\mathbf{y}\|^2}\mathbf{y} \\ &= \mathbf{y}\frac{\mathbf{y}^{\mathsf{H}}\mathbf{x}}{\|\mathbf{y}\|\|\mathbf{y}\|} \\ &= \underbrace{\frac{\mathbf{y}}{\|\mathbf{y}\|}\left(\frac{\mathbf{y}}{\|\mathbf{y}\|}\right)^{\mathsf{H}}}_{\mathbf{P}_{\mathbf{y}}}\mathbf{x} \\ &= \mathbf{P}_{\mathbf{y}}\mathbf{x}\,. \end{aligned}$$

Wir erhalten eine Matrix $\mathbf{P}_{\mathbf{y}}$, welche die gewünschte Projektion realisiert. Das ist die gleiche Matrix, die wir bei der Konstruktion der Householder-Matrix verwendet haben.

Wir sehen auch:

$$\boxed{\text{Bild}\,(\mathbf{P}_{\mathbf{y}}) = \text{Span}\{\mathbf{y}\}\,.}$$

⌟

## Schwarz'sche Ungleichung und Winkel

**Satz 4.3.0.11. Schwarz'sche Ungleichung**

*Sei $V$ ein linearer Raum mit einem Skalarprodukt $\langle\cdot,\cdot\rangle$, dann gilt folgende Ungleichung für alle Elemente $x,\ y \in V$:*

$$\boxed{|\langle x, y\rangle| \le \sqrt{\langle x, x\rangle}\sqrt{\langle y, y\rangle}\,.}$$

*Dies kann alternativ auch mit der aus dem Skalarprodukt kommenden Norm $\|\cdot\|$ geschrieben werden:*

$$|\langle x, y\rangle| \le \|x\|\|y\|\,.$$

*Beweis.* Für $x = 0$ ist dies trivial wahr, deswegen treffen wir die Annahme, dass $x \neq 0$ und $y \neq 0$. Wir führen ein neues Element $z = \langle x, y\rangle x - \langle x, x\rangle y \in V$ ein. Mit diesem $z$ machen wir nun die folgende Berechnung, wobei wir die Eigenschaften des Skalarprodukts verwenden:

$$\begin{aligned} 0 \le \langle z, z\rangle &= \langle \langle x, y\rangle x - \langle x, x\rangle y,\ \langle x, y\rangle x - \langle x, x\rangle y\rangle && \text{(Def. } z\text{)} \\ &= \langle x, y\rangle\langle \langle x, y\rangle x - \langle x, x\rangle y, x\rangle - \langle x, x\rangle\langle \langle x, y\rangle x - \langle x, x\rangle y, y\rangle && \text{(S1)} \\ &= \langle x, y\rangle\overline{\langle x, y\rangle}\langle x, x\rangle - \langle x, y\rangle\overline{\langle x, x\rangle}\langle y, x\rangle - \langle x, x\rangle\overline{\langle x, y\rangle}\langle x, y\rangle + \langle x, x\rangle\overline{\langle x, x\rangle}\langle y, y\rangle && \text{(S1, S2)} \\ &= \|x\|^4\|y\|^2 - \|x\|^2\langle x, y\rangle\langle y, x\rangle \end{aligned}$$

Bis jetzt haben wir gezeigt, dass $0 \le \|x\|^4\|y\|^2 - \|x\|^2\langle x, y\rangle\langle y, x\rangle$. Wir bringen nun diese Ungleichung auf in die gewünschte Form, da $x \neq 0$ und also $\|x\| \neq 0$ ist:

$$\begin{aligned} 0 &\le \|x\|^4\|y\|^2 - \|x\|^2\langle x, y\rangle\langle y, x\rangle \\ \|x\|^2\langle x, y\rangle\overline{\langle x, y\rangle} &\le \|x\|^4\|y\|^2 \\ |\langle x, y\rangle|^2 &\le \|x\|^2\|y\|^2 \\ |\langle x, y\rangle| &\le \|x\|\|y\|\,. \end{aligned}$$

Wir haben also die Richtigkeit der sehr wichtigen Ungleichung $|\langle x, y\rangle| \leq \|x\|\|y\|$ gezeigt. Diese Ungleichung ist demzufolge auch gegeben, sobald ein Skalarprodukt definiert ist.

□

Erst nachdem wir diese Ungleichung bewiesen haben, sind wir auch berechtigt, den Winkel zwischen zwei Elementen $x,\ y$ aus einem linearen Raum $V$ mit dem Skalarprodukt $\langle\cdot,\cdot\rangle$ zu definieren.

**Definition 4.3.0.12. Winkel in einem linearen Raum**

Der Winkel $\widehat{\cdot,\cdot}$ zwischen zwei Elementen $x$ und $y$ eines reellen linearen Raumes $V$ mit Skalarprodukt $\langle\cdot,\cdot\rangle$ ist definiert als

$$\widehat{x,y} = \arccos \frac{\langle x, y\rangle}{\|x\|\|y\|}\,.$$

Erst nachdem wir die Schwarz'sche Ungleichung bewiesen haben, kann gesagt werden, dass $\frac{\langle x,y\rangle}{\|x\|\|y\|}$ immer im Intervall $[-1, 1]$ liegt und es deswegen möglich ist, den arccos dieses Bruches zu berechnen. Ausserdem erhalten wir sofort den Satz des Pythagoras in linearen Räumen mit Skalarprodukt:

**Satz 4.3.0.13. Pythagoras**

*Sei $V$ ein linearer Raum mit Skalarprodukt $\langle\cdot,\cdot\rangle$ und einer aus diesem Skalarprodukt kommenden Norm $\|\cdot\|$.*
*Für zwei Elemente $x$ und $y$, welche orthogonal aufeinander stehen ($x \perp y$, d. h. $\langle x, y\rangle = 0$), gilt*

$$\|x + y\|^2 = \|x - y\|^2 = \|x\|^2 + \|y\|^2\,.$$

## Einheitsvektoren und orthonormale Basen

**Definition 4.3.0.14. Einheitsvektor**

Sei $v \in V$, wobei $V$ ein linearer Raum mit der Norm $\|\cdot\|$ ist. Falls $\|v\| = 1$, dann heisst $v$ *Einheitsvektor*.

**Satz 4.3.0.15. Orthogonale Vektoren sind linear unabhängig**

*Seien $e_1,\ e_2, \ldots, e_n$ Einheitsvektoren in einem linearen Raum $V$ mit einer vom Skalarprodukt $\langle\cdot,\cdot\rangle$ induzierten Norm $\|\cdot\|$.*
*Falls die Einheitsvektoren $e_1,\ e_2, \ldots, e_n$ paarweise orthogonal sind, so sind sie auch linear unabhängig.*

*Beweis.* Um zu zeigen, dass die Vektoren $e_1,\ e_2, \ldots, e_n$ linear unabhängig sind, überprüfen wir die Definition der linearen Unabhängigkeit:

$$\lambda_1 e_1 + \lambda_2 e_2 + \ldots + \lambda_n e_n = 0 \implies \lambda_1 = \lambda_2 = \ldots = \lambda_n = 0\,.$$

Um dies zu beweisen, berechnen wir das Skalarprodukt mit $e_j$ für beliebiges $j$:

$$\begin{aligned} \lambda_1 e_1 + \lambda_2 e_2 + \ldots + \lambda_n e_n = 0 \quad &|\langle\cdot, e_j\rangle \\ \langle \lambda_1 e_1 + \lambda_2 e_2 + \ldots + \lambda_n e_n, e_j\rangle = 0 \quad & \\ \overline{\lambda_1}\underbrace{\langle e_1, e_j\rangle}_{0} + \overline{\lambda_2}\underbrace{\langle e_2, e_j\rangle}_{0} + \ldots + \overline{\lambda_j}\underbrace{\langle e_j, e_j\rangle}_{\|e_j\|^2} + \ldots + \overline{\lambda_n}\underbrace{\langle e_n, e_j\rangle}_{0} = 0\,. & \end{aligned}$$

Die Orthogonalität der Vektoren $e_1, e_2, \ldots, e_n$ bewirkt, dass in dieser Summe nur ein Term verbleibt:

$$\overline{\lambda_j}\langle e_j, e_j\rangle = 0\,.$$

Da $e_j$ ein Einheitsvektor ist, ist $\langle e_j, e_j\rangle = \|e_j\|^2 = 1 \neq 0$. Es muss also gelten, dass $\lambda_j = 0$. Wenn wir das Gleiche für $j = 1, 2, \ldots, n$ tun, so erhalten wir, dass alle Koeffizienten $\lambda_1 = \lambda_2 = \ldots = \lambda_n = 0$.
Somit sind $e_1, e_2, \ldots, e_n$ linear unabhängig.

□

Wir haben im vorigen Beweis gesehen, dass es lediglich entscheidend ist, dass keiner dieser Vektoren der Nullvektor ist, und nicht zwingend, dass alle die Norm 1 besitzen. Dagegen ist die Art, wie wir die Orthogonalität verwendet haben, sehr wichtig und wird im Folgenden noch häufiger eine Rolle spielen.

Das folgende Theorem ist eine direkte Konsequenz des Theorems 4.3.0.15.

**Satz 4.3.0.16. Orthonormale Basis**

*$n$ paarweise orthogonale Einheitsvektoren in einem linearen Raum der Dimension $n$ bilden eine orthonormale Basis dieses Raumes.*

**Bemerkung 4.3.0.17 (Orthogonale Matrix aus orthonormaler Basis)**
Seien in $\mathbb{R}^n$ die Vektoren $\mathbf{v}_1, \ldots, \mathbf{v}_n$, welche eine orthonormale Basis bilden. Damit können wir eine orthogonale Matrix $\mathbf{V}$ definieren:

$$\mathbf{V} = \left[\mathbf{v}_1 \cdot \cdots \cdot \mathbf{v}_n\,.\right]$$

. ⌟

**Bemerkung 4.3.0.18 (Berechnung der Koordinaten in einer orthonormalen Basis)**
Sei $\{e_1, e_2, \ldots, e_n\}$ eine orthonormale Basis im linearen Raum $V$. Wenn wir ein beliebiges $v \in V$ nehmen, so können wir es als eine eindeutige Linearkombination der Basiselemente $e_1, e_2, \ldots, e_n$ darstellen:

$$v = \lambda_1 e_1 + \lambda_2 e_2 + \ldots + \lambda_n e_n\,.$$

Wenn wir die Werte von $\lambda_1, \lambda_2, \ldots, \lambda_n$ bestimmen wollen, so wenden wir den gleichen Trick wie beim Beweis der linearen Unabhängigkeit orthogonaler Vektoren an: Wir bilden das Skalarprodukt mit einem beliebigen Element $e_j$:

$$\begin{aligned}
\langle e_j, \cdot\rangle| \qquad & v = \lambda_1 e_1 + \lambda_2 e_2 + \ldots + \lambda_n e_n \\
& \langle e_j, v\rangle = \lambda_1 \underbrace{\langle e_j, e_1\rangle}_{0} + \lambda_2 \underbrace{\langle e_j, e_2\rangle}_{0} + \ldots + \lambda_j \underbrace{\langle e_j, e_j\rangle}_{1} + \ldots + \lambda_n \underbrace{\langle e_j, e_n\rangle}_{0} \\
& \langle e_j, v\rangle = \lambda_j\,.
\end{aligned}$$

Wenn wir dies für alle $e_j$ mit $j = 1, 2, \ldots, n$ machen, erhalten wir die Werte aller $\lambda_j$. Wir können also $v$ als

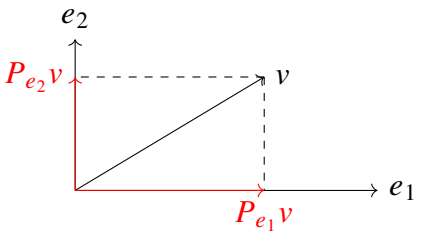
$$v = \langle e_1, v\rangle e_1 + \langle e_2, v\rangle e_2 + \ldots + \langle e_n, v\rangle e_n$$

schreiben. ⌟

**Bemerkung 4.3.0.19 (Projektoren auf die Elemente einer orthonormalen Basis)**
Aus der Definition 4.3.0.9 der orthogonalen Projektionen eines beliebigen Elements $x$ auf einen Einheitsvektor $y$ in einem linearen Raum geht hervor, dass $v$ die Summe der orthogonalen Projektionen von sich auf die Basiselemente $e_1, e_2, \ldots, e_n$ ist:

$$P_{e_j} v = \frac{\langle e_j, v\rangle}{\langle e_j, e_j\rangle} e_j \quad \xrightarrow{\|e_j\|^2 = 1} \quad P_{e_j} v = \langle e_j, v\rangle e_j$$

$$v = P_{e_1} v + P_{e_2} v + \ldots + P_{e_n} v \,.$$




Das ist die allgemeine Form. Wir können aber auch eine interessante Eigenschaft dieser Projektoren im Raum $V = \mathbb{R}^n/\mathbb{C}^n$ mit der orthonormalen Basis $\{\mathbf{b}_1, \ldots, \mathbf{b}_n\}$ zeigen.
Sei ein Vektor $\mathbf{v}$ in $\mathbb{R}^n$ oder $\mathbb{C}^n$ beliebig, diesen Vektor können wir als die Summe der Projektionen auf die Basiselemente schreiben:

$$\mathbf{v} = \textstyle\sum_{j=1}^{n} \langle \mathbf{b}_j, \mathbf{v} \rangle \mathbf{b}_j \,.$$

Wenn wir die konkrete Definition des Skalarprodukts in $\mathbb{R}^n/\mathbb{C}^n$ verwenden, können wir diese Summe weiter umformen:

$$\mathbf{v} = \sum_{j=1}^{n} \langle \mathbf{b}_j, \mathbf{v} \rangle \mathbf{b}_j = \sum_{j=1}^{n} (\mathbf{b}_j^{\mathsf{H}} \mathbf{v}) \mathbf{b}_j = \sum_{j=1}^{n} \mathbf{b}_j (\mathbf{b}_j^{\mathsf{H}} \mathbf{v}) = \sum_{j=1}^{n} \underbrace{(\mathbf{b}_j \mathbf{b}_j^{\mathsf{H}})}_{\mathbf{P}_{b_j}} \mathbf{v}$$

$$\mathbf{v} = \textstyle\sum_{j=1}^{n} \mathbf{P}_{b_j} \mathbf{v} \,.$$

Wir sind nun an einem interessanten Punkt angelangt, denn aus der obigen Gleichung können wir ablesen, dass

$$\mathbf{I} = \sum_{j=1}^{n} \mathbf{P}_{b_j} \,.$$

Also ist die Summe der Projektionsmatrizen auf die orthonormalen Basiselemente genau die Identitätsmatrix. Dies mag einem, wenn man sich die Basiselemente ansieht, nicht offensichtlich erscheinen, ist aber in der geometrischen Sicht trivial.

⌟

## Satz von Parseval und Folgerungen

Sei $V$ ein linearer Raum der Dimension $n$, der ein Skalarprodukt $\langle \cdot, \cdot \rangle$ und eine orthonormale Basis $\{b_1, \ldots, b_n\}$ besitzt. Jedes Element von $V$ schreiben wir als die Summe seiner Projektionen auf die Basiselemente. Daraus bilden wir die Koordinatendarstellung in der orthonormalen Basis.
Wir nehmen nun zwei beliebige Elemente $x$ und $y$ aus $V$. Dann haben wir:

$$x \in V \implies x = \sum_{i=1}^{n} \underbrace{\langle b_i, x \rangle}_{x_i} b_i = \sum_{i=1}^{n} x_i b_i \;\text{ mit } x_i = \langle b_i, x \rangle \,,$$

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \quad \text{Koordinatendarstellung in Basis } \{b_1, .., b_n\} \,,$$

$$y \in V \implies y = \sum_{i=1}^{n} \underbrace{\langle b_i, y\rangle}_{y_i} b_i = \sum_{i=1}^{n} y_i b_i \text{ mit } y_i = \langle b_i, y\rangle ,$$

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \quad \text{Koordinatendarstellung in Basis } \{b_1, \ldots, b_n\} .$$

Mit diesen Voraussetzungen und Notationen haben wir also:

**Satz 4.3.0.20. Parseval**

*Das Skalarprodukt in $V$ lässt sich über das euklidischen Skalarprodukt in $\mathbb{R}^n$ berechnen:*

$$\langle x, y\rangle = \mathbf{x}^{\mathsf{H}}\mathbf{y} = \langle \mathbf{x}, \mathbf{y}\rangle .$$

**Bemerkung 4.3.0.21 (Projektion auf einen Unterraum)**
Was geschieht, wenn ein Element eines $n$-dimensionalen Raumes mit orthonormaler Basis nicht auf alle Basiselemente projiziert wird, sondern nur auf $k < n$?
Wir erhalten eine Projektion auf einen Unterraum.

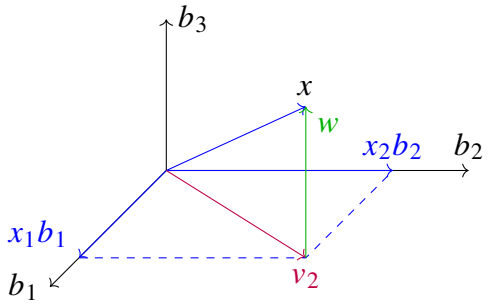
Sei $V$ ein linearer Raum und seien $b_1, b_2, \ldots, b_k \in V$ paarweise orthogonale Elemente mit Norm 1.

Aus diesem Raum können wir ein beliebiges Element $x$ nehmen und die Projektionen dieses Elementes auf die Elemente $b_1, \ldots, b_k$ bestimmen. Wir erhalten die Koordinaten:

$$x_1 = \langle b_1, x\rangle, \; x_2 = \langle b_2, x\rangle, \ldots, \; x_k = \langle b_k, x\rangle$$

und wir können ein weiteres Element $w$ als die Differenz zwischen $x$ und der Summe dieser Projektionen definieren:

$$w = x - (x_1 b_1 + x_2 b_2 + \ldots + x_k b_k) .$$




Eine Eigenschaft von $w$ ist, dass es orthogonal auf allen $b_1, \ldots, b_k$ steht. Dies können wir einfach zeigen:

$$\begin{aligned} \langle b_j, w\rangle &= \langle b_j, x - (x_1 b_1 + \ldots + x_k b_k)\rangle \\ &= \langle b_j, x\rangle - \sum_{i=1}^{k} x_i \underbrace{\langle b_j, b_i\rangle}_{0 \text{ für } i \neq j} \\ &= \langle b_j, x\rangle - x_j = 0 . \end{aligned}$$

Auf diese Weise bestimmen wir ein Element $v_k$, welches die Projektion von $x$ auf den Unterraum $\mathrm{Span}\{b_1, \ldots, b_k\}$ ist:

$$\begin{aligned} v_k &= x_1 b_1 + \ldots + x_k b_k \\ &= P^{\perp}_{\mathrm{Span}\{b_1,\ldots,b_k\}} x \,. \end{aligned}$$

Das Element $w$ ist hierbei der Vektor, welcher diese Projektion von $x$ auf den Unterraum realisiert. ⌟

## 4.4 Gram-Schmidt-Algorithmus

**Satz 4.4.0.1. Gram-Schmidt**

*Wenn $V$ ein linearer Raum mit Basis $\{v_1, v_2, \dots, v_n\}$ und Skalarprodukt $\langle \cdot, \cdot \rangle$ ist, dann gibt es eine orthonormale Basis $\{q_1, q_2, \dots, q_n\}$ für $V$, so dass $\mathrm{Span}\{q_1, q_2, \dots, q_j\} = \mathrm{Span}\{v_1, v_2, \dots, v_j\}$ für $j = 1, 2, \dots, n$ ist.*

Dieser Satz macht die folgenden zwei wichtigen Aussagen:

1. Ein endlichdimensionaler Raum besitzt immer eine orthonormale Basis.
2. Diese orthogonalen Vektoren bilden verschachtelte Räume.

   In der Tat ist $\mathrm{Span}\{v_1\}$ offensichtlich ein Unterraum von $\mathrm{Span}\{v_1, v_2\}$, der ein Unterraum von $\mathrm{Span}\{v_1, v_2, v_3\}$ ist, usw. Die Gleichheit von $\mathrm{Span}\{v_1, \dots v_j\}$ und $\mathrm{Span}\{q_1, \dots, q_j\}$ ergibt, dass $\mathrm{Span}\{q_1\}$ ein Unterraum von $\mathrm{Span}\{q_1, q_2\}$ ist, der ein Unterraum von $\mathrm{Span}\{q_1, q_2, q_3\}$ ist, usw.

Genauso wichtig wie das Theorem selber ist sein Beweis, denn es ist ein konstruktiver Beweis, der einen sehr wichtigen Algorithmus liefert, mit dem wir diese orthogonalen Vektoren berechnen können.

*Beweis.* Gegeben sei eine Basis $\{v_1, \dots, v_n\}$ eines linearen Raumes $V$, dann suchen wir eine orthonormale Basis $\{q_1, .., q_n\}$ im Raum $V$ mit der Eigenschaft, dass $\mathrm{Span}\{v_1, \dots, v_j\} = \mathrm{Span}\{q_1, .., q_j\}$.

Diese $q_1, \dots, q_n$ bilden wir einzeln nacheinander, beginnend mit $q_1$.
Die Bedingungen für $q_1$ sind, dass es Norm eins haben muss und den gleichen Unterraum wie $v_1$ aufspannen muss. Deswegen ist $q_1$ einfach der normierte Vektor $v_1$:

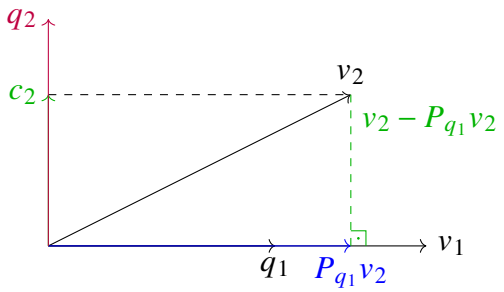
$$q_1 = \frac{1}{\|v_1\|} v_1 .$$

Daraus haben wir auch:

$$v_1 = \|v_1\| q_1 = \langle q_1, v_1 \rangle q_1 .$$

In all diesen Berechnungen stellen wir $v_j$ immer als lineare Kombination von $\{q_1, \dots, q_j\}$ dar. Dies werden wir zu einem späteren Zeitpunkt benutzen.

Nun müssen wir ein $q_2$ bilden, welches orthogonal auf $q_1$ steht. Projektionen kreieren solche orthogonale Elemente.



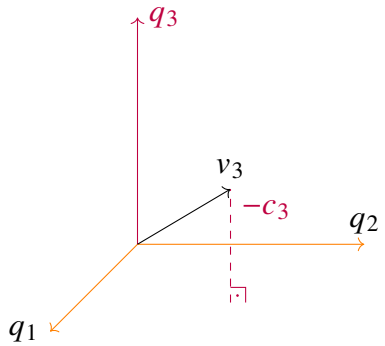

In dieser Skizze betrachten wir das Ganze im Raum der Vektoren, wir sind jedoch nicht auf ihn limitiert.

Wir betrachten nun $v_1$ und $v_2$. Das Element $v_1$ ist bereits auf $q_1$ projiziert. Da $v_1$ und $v_2$ als Basiselemente linear unabhängig sind, liegt $v_2$ nicht in $\mathrm{Span}\{v_1\} = \mathrm{Span}\{q_1\}$. Wenn wir nun $v_2$ auf $\mathrm{Span}\{q_1\}$ projizieren, erhalten wir ein Element $c_2$, welches $v_2$ mit seiner Projektion verbindet und orthogonal auf $q_1$ steht. Wenn wir $c_2$ normieren, entspricht es dem gesuchten $q_2$. Ebenfalls gilt, dass $\mathrm{Span}\{v_1, v_2\} = \mathrm{Span}\{q_1, q_2\}$. Schreiben wir diese Schritte mathematisch aus:

$$
\begin{aligned}
q_1: \quad & q_1 = \tfrac{1}{\|v_1\|} v_1 \\
& \Longrightarrow \; v_1 = \|v_1\| q_1 = \langle q_1, v_1 \rangle q_1 \,.
\end{aligned}
$$

$$
\begin{aligned}
q_2: \quad & c_2 = v_2 - P^{\perp}_{q_1} v_2 = v_2 - \langle q_1, v_2 \rangle q_1 \\
& q_2 = \tfrac{c_2}{\|c_2\|} \\
& \Longrightarrow \; c_2 = \|c_2\| q_2 = \langle q_2, v_2 \rangle q_2 \\
& \Longrightarrow \; v_2 = \langle q_1, v_2 \rangle q_1 + \langle q_2, v_2 \rangle q_2 \,.
\end{aligned}
$$

Das Gleiche machen wir nun für $q_3$.




Wir nehmen nun $v_3$ dazu. Da $v_1$, $v_2$ und $v_3$ als Basiselemente linear unabhängig sind, liegt $v_3$ nicht in $\mathrm{Span}\{v_1, v_2\} = \mathrm{Span}\{q_1, q_2\}$. Wenn wir nun $v_3$ auf $\mathrm{Span}\{q_1, q_2\}$ projizieren, erhalten wir ein Element $c_3$, welches $v_3$ mit seiner Projektion verbindet und orthogonal auf $q_1$ und $q_2$ steht. Wenn wir $c_3$ normieren, entspricht es dem gesuchten $q_3$. Ebenfalls gilt, dass $\mathrm{Span}\{v_1, v_2, v_3\} = \mathrm{Span}\{q_1, q_2, q_3\}$. Ergänzen wir die oberen mathematischen Schritte mit diesen Überlegungen:

$$
\begin{aligned}
q_1: \quad & q_1 = \tfrac{1}{\|v_1\|} v_1 \\
& \Longrightarrow \; v_1 = \|v_1\| q_1 = \langle q_1, v_1 \rangle q_1 \,.
\end{aligned}
$$

$$
\begin{aligned}
q_2: \quad & c_2 = v_2 - P^{\perp}_{q_1} v_2 = v_2 - \langle q_1, v_2 \rangle q_1 \\
& q_2 = \tfrac{c_2}{\|c_2\|} \\
& \Longrightarrow \; c_2 = \|c_2\| q_2 = \langle q_2, v_2 \rangle q_2 \\
& \Longrightarrow \; v_2 = \langle q_1, v_2 \rangle q_1 + \langle q_2, v_2 \rangle q_2 \,.
\end{aligned}
$$

$$
\begin{aligned}
q_3: \quad & c_3 = v_3 - P^{\perp}_{\mathrm{Span}\{q_1, q_2\}} v_3 = v_3 - \langle q_1, v_3 \rangle q_1 - \langle q_2, v_3 \rangle q_2 \\
& q_3 = \tfrac{c_3}{\|c_3\|} \\
& \Longrightarrow \; c_3 = \|c_3\| q_3 = \langle q_3, v_3 \rangle q_3 \\
& \Longrightarrow \; v_3 = \langle q_+, v_3 \rangle q_1 + \langle q_2, v_3 \rangle q_2 + \langle q_3, v_3 \rangle q_3 \,.
\end{aligned}
$$

Das Gleiche werden wir nun für alle weiteren $q_j$ und $v_j$ machen. In jedem Schritt projizieren wir $v_k$ auf $\text{Span}\{q_1, .., q_{k-1}\}$ um $c_k$ und dann $q_k$ zu erhalten. Dass alle $c_k$ immer orthogonal auf den vorherigen Elementen stehen, haben wir in 4.3.0.21 gezeigt. Dieser $k$-te Schritt sieht dann folgendermassen aus:

$$\begin{aligned} c_k &= v_k - P^{\perp}_{\text{Span}\{q_1,..,q_{k-1}\}} v_k = v_k - \langle q_1, v_k\rangle q_1 - \langle q_2, v_k\rangle q_2 - \ldots - \langle q_{k-1}, v_k\rangle q_{k-1} \\ q_k &= \frac{c_k}{\|c_k\|} \\ &\Longrightarrow \quad c_k = \|c_k\| q_k = \langle q_k, v_k\rangle q_k \\ &\Longrightarrow \quad v_k = \langle q_1, v_k\rangle q_1 + \langle q_2, v_k\rangle + \ldots + \langle q_k, v_k\rangle q_k \,. \end{aligned}$$

Nach $n$ solchen Schritten erhalten wir die gewünschten $q_1, q_2, \ldots, q_n$ und mit ihnen die orthonormale Basis in $V$.

□

**Beispiel 4.4.0.2 (Legendre-Polynome aus der Gram-Schmidt Orthogonalisierung)**
In diesem Beispiel[4] werden wir den Gram-Schmidt Algorithmus verwenden, um eine orthonormale Basis im Raum $\mathcal{P}_3[-1,1]$ der Polynome zu finden.

Sei $V = \mathcal{P}_3[-1,1]$ der lineare Raum der Polynome von Grad kleiner als 3. Die Basis sei definiert als die Monome $\{p_0, p_1, p_2\}$, wobei $p_j(t) = t^j$.

Zudem sei das Skalarprodukt $\langle \cdot, \cdot \rangle$ in diesem Raum:

$$\langle p, q\rangle = \int_{-1}^{1} p(t)q(t)dt \,.$$

Eine Frage, die wir uns stellen können, ist, ob $\{p_0, p_1, p_2\}$ eine orthonormale Basis in $\mathcal{P}_3$ bilden?

Damit $\{p_0, p_1, p_2\}$ eine orthonormale Basis bilden, müssen sie alle orthogonal aufeinander stehen. Zwei Elemente stehen orthogonal aufeinander genau dann, wenn das Skalarprodukt zwischen ihnen null ist. Wenn wir aber das Skalarprodukt zwischen $p_0$ und $p_2$ berechnen, sehen wir, dass dieses nicht null ist:

$$\langle p_0, p_2\rangle = \int_{-1}^{1} 1t^2 dt = \frac{2}{3} \neq 0 \implies p_0 \not\perp p_2 \,.$$

Das heisst, dass $p_0, p_1, p_2$ keine orthonormale Basis in $\mathcal{P}_3$ bilden. Wenn wir aber nun solch eine orthogo-

[4] Auch im Beispiel 3 auf Seite 97 in NS15.

nale Basis in $\mathscr{P}_3$ haben wollen, so liefert uns der Gram-Schmidt Algorithmus eine:

$$
\begin{aligned}
q_1: \quad & q_1 = \tfrac{1}{\|p_0\|} p_0 \\
& \|p_0\|^2 = \langle p_0, p_0\rangle = \int_{-1}^{1} 1\,dt = 2 \implies \|p_0\| = \sqrt{2} \\
& q_1 = \tfrac{1}{\sqrt{2}} p_0 \,.
\end{aligned}
$$

$$
\begin{aligned}
q_2: \quad & c_2 = p_1 - \langle q_1, p_1\rangle q_1 = p_1 - \tfrac{1}{\sqrt{2}} \underbrace{\langle p_0, p_1\rangle}_{0} q_1 = p_1 \\
& \|c_2\|^2 = \langle c_2, c_2\rangle = \int_{-1}^{1} t^2 dt = \tfrac{2}{3} \implies \|c_2\| = \sqrt{\tfrac{2}{3}} \\
& q_2 = \tfrac{c_2}{\|c_2\|} = \sqrt{\tfrac{3}{2}} p_1 \,.
\end{aligned}
$$

$$
\begin{aligned}
q_3: \quad & c_3 = p_2 - \langle q_1, p_2\rangle q_1 - \langle q_2, p_2\rangle q_2 \\
& \langle q_1, p_2\rangle = \int_{-1}^{1} \tfrac{1}{\sqrt{2}} t^2 dt = \tfrac{\sqrt{2}}{3} \\
& \langle q_2, p_2\rangle = \int_{-1}^{1} t^2 \sqrt{\tfrac{3}{2}} t\,dt = 0 \\
& c_3 = p_2 - \tfrac{\sqrt{2}}{3} q_1 = p_2 - \tfrac{1}{3} p_0 \\
& \|c_3\|^2 = \langle c_3, c_3\rangle = \int_{-1}^{1} (t^2 - \tfrac{1}{3})^2 dt = \tfrac{8}{45} \implies \|c_3\| = \tfrac{3\sqrt{5}}{2\sqrt{2}} \\
& q_3 = \tfrac{c_3}{\|c_3\|} = \tfrac{3\sqrt{5}}{2\sqrt{2}} p_2 - \tfrac{\sqrt{5}}{2\sqrt{2}} p_0 \,.
\end{aligned}
$$

Gram-Schmidt liefert $q_1$, $q_2$ und $q_3$, welche eine orthonormale Basis im Raum $\mathscr{P}_3[-1,1]$ bezüglich des gegebenen Skalarprodukts bilden. Wenn wir den Algorithmus mit $p_4, p_5, \ldots, p_n$ weiterführen, erhalten wir zusätzliche Polynome $q_4, q_5, \ldots, q_n$, welche, zusammen mit den vorigen, jeweils eine orthonormale Basis im Raum $\mathscr{P}_4[-1,1]$, $\mathscr{P}_5[-1,1]$, $\ldots$, $\mathscr{P}_n[-1,1]$ bezüglich des gegebenen Skalarprodukts bilden.

Diese Polynome, die wir mit diesem Skalarprodukt erhalten haben, heissen *Legendre-Polynome*. Ein anderes Skalarprodukt liefert als Ergebnis vom Gram-Schmidt Algorithmus auch andere Polynome. In der Physik und in der angewandten Mathematik spielen solche Basen von orthogonalen Polynomen bezüglich speziell gewählten Skalarprodukten eine sehr wichtige Rolle, so dass es viele spezielle Namen dafür gibt (Z. B.: Legendre, Hermite, Laguerre). ⌟

## QR-Zerlegung mittels Gram-Schmidt

**Bemerkung 4.4.0.3 (Gram-Schmidt in $\mathbb{R}^n$ gibt eine QR-Zerlegung)**
Was passiert, wenn wir Gram-Schmidt im Raum $\mathbb{R}^n/\mathbb{C}^n$ anwenden?

Sei $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\}$ eine Basis von $\mathbb{R}^n$ und sei die Matrix $\mathbf{A}$ definiert als

$$
\mathbf{A} = \begin{bmatrix} | & & | \\ \mathbf{v}_1 & \cdots & \mathbf{v}_n \\ | & & | \end{bmatrix}.
$$

Als wir den Gram-Schmidt Algorithmus besprochen haben, führten wir auch immer $\mathbf{v}_k$ als lineare Kombination von $\mathbf{q}_1, \ldots, \mathbf{q}_k$ mit. Das werden wir nun ausnützen, um eine QR-Zerlegung der Matrix $\mathbf{A}$ zu bilden. Solch ein $\mathbf{v}_k$ haben wir dort geschrieben (jetzt mit dem euklidischen Skalarprodukt) als:

$$
\mathbf{v}_k = (\mathbf{q}_1^{\mathsf{H}} \mathbf{v}_k)\mathbf{q}_1 + (\mathbf{q}_2^{\mathsf{H}} \mathbf{v}_k)\mathbf{q}_2 + \ldots + (\mathbf{q}_k^{\mathsf{H}} \mathbf{v}_k)\mathbf{q}_k \,.
$$

Diese lineare Kombination können wir als eine Matrix-Vektor-Multiplikation ansehen:

$$\overset{n\times 1}{\begin{bmatrix} | \\ \mathbf{v}_k \\ | \end{bmatrix}} = \overset{n\times k}{\begin{bmatrix} | & | & & | \\ \mathbf{q}_1 & \mathbf{q}_2 & \cdots & \mathbf{q}_k \\ | & | & & | \end{bmatrix}} \overset{k\times 1}{\begin{bmatrix} \mathbf{q}_1^{\mathsf{H}}\mathbf{v}_k \\ \vdots \\ \mathbf{q}_k^{\mathsf{H}}\mathbf{v}_k \end{bmatrix}}.$$

Wenn wir aber nicht nur $\mathbf{v}_k$, sondern alle $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k$ berechnen wollen, so können wir das obige System zu einer Matrix-Matrix-Multiplikation erweitern:

$$\underbrace{\overset{n\times k}{\begin{bmatrix} | & | & & | \\ \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_k \\ | & | & & | \end{bmatrix}}}_{\mathbf{A}_k} = \underbrace{\overset{n\times k}{\begin{bmatrix} | & | & & | \\ \mathbf{q}_1 & \mathbf{q}_2 & \cdots & \mathbf{q}_k \\ | & | & & | \end{bmatrix}}}_{\mathbf{Q}_k} \underbrace{\overset{k\times k}{\begin{bmatrix} \mathbf{q}_1^{\mathsf{H}}\mathbf{v}_1 & \mathbf{q}_1^{\mathsf{H}}\mathbf{v}_2 & \cdots & \mathbf{q}_1^{\mathsf{H}}\mathbf{v}_k \\ 0 & \mathbf{q}_2^{\mathsf{H}}\mathbf{v}_2 & \cdots & \mathbf{q}_2^{\mathsf{H}}\mathbf{v}_k \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \mathbf{q}_k^{\mathsf{H}}\mathbf{v}_k \end{bmatrix}}}_{\mathbf{R}_k}.$$

Die Matrix $\mathbf{Q}_k$ hat orthonormale Spalten, ist aber keine orthogonale Matrix, da sie nicht quadratisch ist. Die Matrix $\mathbf{R}_k$ ist eine obere Dreiecksmatrix der Dimension $k \times k$.
Wenn wir dies nicht nur für $k$ Basisvektoren, sondern für alle $n$ Basisvektoren machen, so wird $\mathbf{A}_n$ eine $n \times n$ Matrix sein, deren Spalten der Basis $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ entspricht. $\mathbf{Q}_n$ ist eine orthogonale Matrix, da alle Spalten orthonormal sind und sie Dimension $n \times n$ hat. $\mathbf{R}_n$ ist eine obere Dreiecksmatrix der Dimension $n \times n$:

$$\underbrace{\overset{n\times n}{\begin{bmatrix} | & & | \\ \mathbf{v}_1 & \cdots & \mathbf{v}_n \\ | & & | \end{bmatrix}}}_{\mathbf{A}_n} = \underbrace{\overset{n\times n}{\begin{bmatrix} | & & | \\ \mathbf{q}_1 & \cdots & \mathbf{q}_n \\ | & & | \end{bmatrix}}}_{\mathbf{Q}_n} \underbrace{\overset{n\times n}{\begin{bmatrix} * & \cdots & * \\ & \ddots & \vdots \\ 0 & & * \end{bmatrix}}}_{\mathbf{R}_n}.$$

Das ist nun $\mathbf{A} = \mathbf{Q}\,\mathbf{R}$, wobei $\mathbf{Q}$ eine orthogonale Matrix und $\mathbf{R}$ eine obere Dreiecksmatrix ist. Dies entspricht demzufolge einer QR-Zerlegung von $\mathbf{A}$. Ein Nebenprodukt des Gram-Schmidt Verfahrens ist also eine QR-Zerlegung der Basisvektoren.

⌟

**Bemerkung 4.4.0.4 (QR-Zerlegung mit orthogonalen Transformationen oder mit Gram-Schmidt?)**
Bis jetzt haben wir die QR-Zerlegung mithilfe von Givens-Rotationen oder Householder-Spiegelungen gesehen. Dort haben wir in jedem Schritt eine orthogonale Matrix verwendet, um ein Teil der oberen Dreiecksmatrix zu bauen. Da wir nur mit orthogonalen Transformationen arbeiteten, war der Prozess stabil. Das heisst, dass sich die Rundungsfehler nicht sammelten. Allerdings ist diese Art der QR-Zerlegung auch sehr teuer! Diese Methoden lieferten eine orthonormale Basis (die Spalten von $\mathbf{Q}$), aber erst ganz am Schluss, wenn der ganze Algorithmus durchgeführt wurde. Falls wir früher aufhören, dann haben wir nichts Aussagekräftiges, wir haben keinen Teil einer orthonormalen Basis, sondern nur einige orthogonale Matrizen.

Hier ist die Situation eine ganz andere: Gram-Schmidt bildet im $k$-ten Schritt $\mathbf{q}_1, \mathbf{q}_2, .., \mathbf{q}_k$ orthonormale Vektoren, die eine orthonormale Basis im Unterraum $\mathrm{Span}\{\mathbf{v}_1, \mathbf{k}_2, .., \mathbf{v}_k\}$ bilden. Wir können also früher aufhören und wir haben eine orthonormale Basis in einem Unterraum! Des Weiteren ist es günstiger in der Berechnung als die anderen Methoden. Gram-Schmidt hat aber auch einen grossen Nachteil, die Stabilität der Berechnung geht verloren. Das spielt keine Rolle bei analytischen Berechnungen, wie z. B. bei einzelnen Polynomen oder auch sehr kleinen Matrizen. Sobald wir aber grössere Matrizen haben, z. B. ab circa $n \geq 10$,

können sich die Rundungsfehler so sammeln, dass die entstandenen Vektoren gar nicht mehr orthogonal sind und die Ergebnisse dramatisch falsch sind. Darum wird Gram-Schmidt in der Praxis nie in dieser Form verwendet, sondern in leicht veränderter Form, die versucht, Orthogonalität trotz Rundungsfehler so gut wie möglich zu erhalten. Diese modifizierten Gram-Schmidt Algorithmen können mit Erfolg verwendet werden, wobei mehr Orthogonalität nur mit mehr Rechenaufwand erreicht werden kann.

⌟

**Bemerkung 4.4.0.5 (Vorzeitiges Ende in Gram-Schmidt-Algorithmus)**
Was passiert bei Gram-Schmidt, wenn $\|c_k\| = 0$?
In diesem Fall bricht der ganze Algorithmus zusammen. Dies mag auf den ersten Blick schlecht erscheinen, ist es aber nicht:

$$\|c_k\| = 0 \implies c_k = 0$$
$$c_k = 0 \iff v_k = \langle q_1, v_k\rangle q_1 + \ldots + \langle q_{k-1}, v_k\rangle q_{k-1}\,.$$

Dies bedeutet, dass $v_k$ die lineare Kombination von $q_1, \ldots, q_{k-1}$ ist. Da diese Elemente aber den genau gleichen Raum wie $v_1, \ldots, v_{k-1}$ aufspannen, bedeutet dies, dass $v_k$ eine lineare Kombination von $v_1, \ldots, v_{k-1}$ ist, also sind $v_1, \ldots, v_k$ linear abhängig.

Gram-Schmidt liefert also auch noch einen Algorithmus, mit dem überprüft werden kann, ob Elemente eines linearen Raumes linear unabhängig sind oder nicht. Des Weiteren ist dieser Algorithmus in der Lage, die Elemente zu identifizieren, die linear abhängig sind. ⌟

## Modifizierter Gram-Schmidt Algorithmus

Die fehlende Stabilität des Gram-Schmidt Algorithmus ist eine grosse Schwierigkeit. Deswegen zeigen wir nun eine modifizierte Version des Gram-Schmidt Algorithmus, welche dieses Problem zwar nicht komplett löst, aber die Stabilität etwas verbessert.

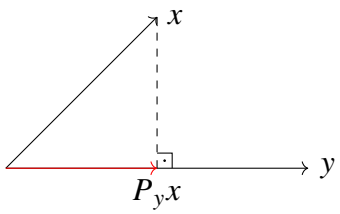
**Pseudo-code 4.4.0.6: Modifizierter Gram-Schmidt Algorithmus**

```
1 for j = 1,2,...,n:
2 |   v = v_j
3 |   for i = 1,2,...,j-1:
4 |   |   r_ij = ⟨q_i, v⟩
5 |   |   v = v - r_ij q_i
6 |
7 |   q_i = v/‖v‖
```

In Zeile 5 sieht man die Veränderung: Statt dass wir zuerst alle Projektionen $\langle q_i, v\rangle$ ausrechnen und erst am Ende von $v$ abziehen, berechnen wir jede Projektion einzeln und ziehen sie direkt von $v$ ab, so dass wir bei der nächsten Projektion $\langle q_{i+1}, v\rangle$ ein modifiziertes $v$ haben. Diese einfache Modifikation lässt die Stabilitätsprobleme nicht komplett verschwinden, aber ermöglicht auch bei grösseren Vektoren eine bessere Verwendung.

## 4.5 Projektoren

Im Abschnitt **Orthogonale Projektoren** des Kapitels 4.3 haben wir den orthogonalen Projektor $P_y^\perp x$, welcher $x$ orthogonal auf $y$ projiziert, besprochen.




Solch einen orthogonalen Projektor $P_y^\perp$ haben wir als eine Funktion definiert

$$P_y^\perp : V \to V \text{ mit } P_y^\perp x = \frac{\langle y, x\rangle}{\langle y, y\rangle} y\,.$$

Wenn der lineare Raum $\mathbb{R}^n$ oder $\mathbb{C}^n$ ist, so können wir eine Matrix $\mathbf{P}_y$ definieren, welche diese orthogonale Projektion durchführt:

$$\mathbf{P}_\mathbf{y}^\perp = \frac{\mathbf{y}}{\|\mathbf{y}\|}\frac{\mathbf{y}^\mathsf{H}}{\|\mathbf{y}\|}\,.$$

In diesem Kapitel werden wir den Projektor auf eine allgemeinere Art besprechen und ihn klar definieren.

**Definition 4.5.0.1. Projektor**

Sei $V$ ein linearer Raum. Eine lineare Abbildung $\mathscr{P} : V \to V$ heisst *Projektor* falls

$$\mathscr{P}^2 = \mathscr{P}\,.$$

Falls $V = \mathbb{R}^n/\mathbb{C}^n$ dann wird $\mathscr{P}$ mit einer Matrix $\mathbf{P}$ identifiziert, für welche

$$\mathbf{P} : \mathbb{C}^n \to \mathbb{C}^n \quad \text{und} \quad \mathbf{P}^2 = \mathbf{P}$$

gilt. Falls der lineare Raum $V$ auch ein Skalarprodukt $\langle\cdot,\cdot\rangle$ hat, dann macht es Sinn, über orthogonale Projektoren zu reden.

**Definition 4.5.0.2. Orthogonale und schiefe Projektoren**

Der Projektor $\mathscr{P} : V \to V$ im linearen Raum $V$ mit dem Skalarprodukt $\langle\cdot,\cdot\rangle$ heisst *orthogonaler Projektor*, falls

$$x - \mathscr{P}x \perp \text{Bild}\,\mathscr{P} \text{ für alle } x \in V\,,$$

sonst heisst er *schiefer Projektor*.

Man kann einen orthogonalen Projektor auch anders charakterisieren.

**Satz 4.5.0.3. Orthogonale und schiefe Projektoren**

*Der Projektor $\mathscr{P} : V \to V$ im linearen Raum $V$ mit dem Skalarprodukt $\langle \cdot, \cdot \rangle$ ist ein orthogonaler Projektor genau dann, wenn*

$$\langle x, \mathscr{P}y \rangle = \langle \mathscr{P}x, y \rangle \text{ für alle } x, y \in V\,,$$

*d. h. wenn der Projektor $\mathscr{P}$ ein selbstadjungierter Operator ist.*

*Beweis.* $\mathscr{P}$ ist orthogonal bedeutet, dass für beliebige $x, y \in V$ gilt:

$$\langle x - \mathscr{P}x, \mathscr{P}y \rangle = 0 = \langle \mathscr{P}x, y - \mathscr{P}y \rangle\,.$$

Wenn wir die Linearität des Skalarprodukts verwenden, sehen wir die äquivalente Bedingung:

$$\langle x, \mathscr{P}y \rangle = \langle \mathscr{P}x, \mathscr{P}y \rangle = \langle \mathscr{P}x, y \rangle\,.$$

Auch umgekehrt, falls der Projektor $\mathscr{P}$ selbstadjungiert ist, rechnen wir z. B.:

$$\langle x - \mathscr{P}x, \mathscr{P}y \rangle = \langle x - \mathscr{P}x, \mathscr{P}\mathscr{P}y \rangle = \langle \mathscr{P}(x - \mathscr{P}x), \mathscr{P}y \rangle = \langle \mathscr{P}x - \mathscr{P}\mathscr{P}x, \mathscr{P}y \rangle = \langle \mathscr{P}x - \mathscr{P}x, \mathscr{P}y \rangle = 0\,,$$

für alle $x, y \in V$.

□

Im Fall $V = \mathbb{C}^n$ haben wir also, dass der Projektor $\mathbf{P}$ orthogonal ist, dann und nur dann, wenn $\mathbf{P}^H = \mathbf{P}$. In anderen Worten: Die Projektionsmatrix $\mathbf{P}$ ist ein orthogonaler Projektor genau dann, wenn sie Hermitesymmetrisch ist. Im Fall $V = \mathbb{R}^n$: Die Projektionsmatrix $\mathbf{P}$ ist ein orthogonaler Projektor genau dann, wenn sie symmetrisch ist.

**Satz 4.5.0.4. Orthogonale und schiefe Projektoren**

*Die Matrix $\mathbf{P}$ ist ein orthogonaler Projektor genau dann, wenn*

$$\mathbf{P}^2 = \mathbf{P} \quad \textit{und} \quad \mathbf{P}^H = \mathbf{P}\,.$$

*Falls nur die Bedingung $\mathbf{P}^2 = \mathbf{P}$ erfüllt ist, so ist $\mathbf{P}$ ein schiefer Projektor.*

**Beispiel 4.5.0.5 (Schiefer Projektor)**
Im Folgenden sehen wir ein Beispiel eines schiefen Projektors.

Seien $\mathbf{u},\ \mathbf{v} \in \mathbb{C}^n$ zwei Vektoren, die nicht orthogonal aufeinander stehen, d.h mit $\mathbf{v}^H\mathbf{u} \neq 0$. Wir definieren die Matrix $\mathbf{P}$:

$$\mathbf{P} = \frac{\mathbf{u}\mathbf{v}^H}{\mathbf{v}^H\mathbf{u}}\,.$$

Um zu zeigen, dass die Matrix ein schiefer Projektor ist, müssen wir zeigen, dass sie ein Projektor, aber kein orthogonaler Projektor ist. Sie ist ein Projektor sobald $\mathbf{P}^2 = \mathbf{P}$ gilt. Rechnen wir also:

$$\mathbf{P}^2 = \frac{\mathbf{u}\mathbf{v}^H\mathbf{u}\mathbf{v}^H}{(\mathbf{v}^H\mathbf{u})(\mathbf{v}^H\mathbf{u})} = \frac{\mathbf{u}(\mathbf{v}^H\mathbf{u})\mathbf{v}^H}{(\mathbf{v}^H\mathbf{u})(\mathbf{v}^H\mathbf{u})} = \frac{\mathbf{u}\mathbf{v}^H}{\mathbf{v}^H\mathbf{u}} = \mathbf{P}\,.$$

Wir müssen also nur noch zeigen, dass sie kein orthogonaler Projektor ist. Es wäre ein orthogonaler Projektor, wenn $\mathbf{P} = \mathbf{P}^H$ gelten würde. Dies ist aber nicht erfüllt, denn:

$$\mathbf{P}^H = \frac{\mathbf{v}\mathbf{u}^H}{\mathbf{u}^H\mathbf{v}} \neq \mathbf{P}\,.$$

Also ist diese Matrix ein schiefer Projektor. Um herauszufinden, wie die Projektion aussieht, werden wir das Bild und den Kern dieser Matrix bestimmen.
Wir zeigen zuerst, dass für diese Matrix

$$\text{Bild}\,(\mathbf{P}) = \text{Span}\{\mathbf{u}\}$$

ist. Wir schauen uns ein beliebiges Element im Bild an: $\mathbf{Px} = \frac{1}{\mathbf{v}^H\mathbf{u}}\mathbf{u}(\mathbf{v}^H\mathbf{x})$. Das $\mathbf{u}$ wird von links mit dem Skalar $\frac{1}{\mathbf{v}^H\mathbf{u}}$ und von rechts mit dem Skalar $\mathbf{v}^H\mathbf{x}$ multipliziert. Das Resultat ist also immer ein Vielfaches von $\mathbf{u}$ und deswegen ist das Bild $\text{Span}\{\mathbf{u}\}$.

Wir zeigen nun, dass für diese Matrix $\mathbf{P}$

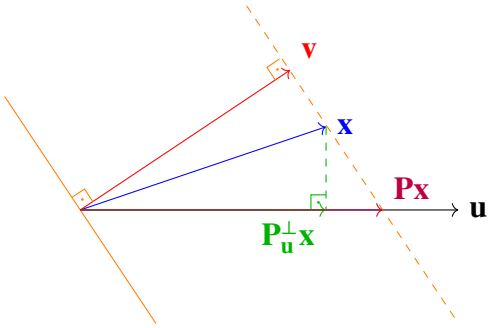
$$\text{Kern}\,(\mathbf{P}) = \text{Span}\{\mathbf{v}\}^\perp$$

ist, d. h. der Kern besteht aus allen Vektoren, welche orthogonal auf $\mathbf{v}$ stehen. Sei $\mathbf{x}$ beliebig im Kern der Projektion:

$$\begin{aligned}
\mathbf{Px} &= \mathbf{0}\\
\frac{1}{\mathbf{v}^H\mathbf{u}}\mathbf{u}\mathbf{v}^H\mathbf{x} &= \mathbf{0}\\
\mathbf{u}\mathbf{v}^H\mathbf{x} &= \mathbf{0}\\
(\mathbf{u}\mathbf{v}^H\mathbf{x})^H &= \mathbf{0}^H\\
\underbrace{\mathbf{x}^H\mathbf{v}}_{\alpha\in\mathbb{C}}\ \mathbf{u}^H &= \mathbf{0}\,,
\end{aligned}$$

d. h. $\alpha u_1 = 0,\ \alpha u_2 = 0, \ldots,\ \alpha u_n = 0$. Der Vektor $\mathbf{u}$ ist nicht Null (sonst ist die Bedingung $\mathbf{v}^H\mathbf{u} \neq 0$ verletzt), also muss $\alpha = 0$ und demzufolge $\mathbf{x}^H\mathbf{v} = 0$ sein. Das bedeutet: $\mathbf{x} \perp \mathbf{v}$ für jedes $\mathbf{x}$ im Kern der Projektion. Demzufolge ist der Kern von $\mathbf{P}$ die Menge aller Vektoren, die orthogonal auf $\mathbf{v}$ stehen.

Die folgende Skizze zeigt, wieso dies ein schiefer Projektor genannt wird:




⌟

**Beispiel 4.5.0.6 (Projektoren auf Unterräume)**
Wir hatten oft die orthogonale Projektion auf einen Vektor, d. h. auf einen 1-dimensionalen Raum, gesehen. Im Folgenden besprechen wir orthogonale Projektionen, die auf mehrdimensionale Unterräume projizieren.

Sei $\mathbf{U} \in \mathbb{C}^{n\times k}$ mit der Eigenschaft, dass $\mathbf{U}^H\mathbf{U} = \mathbf{I}$. Die Spalten von $\mathbf{U}$ sind also paarweise orthogonal und haben alle die euklidische Norm 1. Sie bilden also eine orthonormale Basis im Bild $\mathbf{U}$.

Mit diesem $\mathbf{U}$ definieren wir einen orthogonalen Projektor $\mathbf{P}$, welcher einen Vektor $\mathbf{x}$ auf den Unterraum, der von den Spaltenvektoren von $\mathbf{U}$ aufgespannt wird, projiziert:

$$\mathbf{P} = \mathbf{U}\mathbf{U}^H\,.$$

Das ist ein orthogonaler Projektor, da beide Bedingungen des orthogonalen Projektors erfüllt sind:

$$\mathbf{P}^2 = \mathbf{U}\,\mathbf{U}^{\mathsf{H}}\,\mathbf{U}\,\mathbf{U}^{\mathsf{H}} = \mathbf{U}\,\mathbf{U}^{\mathsf{H}} = \mathbf{P}\,,$$
$$\mathbf{P}^{\mathsf{H}} = (\mathbf{U}^{\mathsf{H}})^{\mathsf{H}}\,\mathbf{U}^{\mathsf{H}} = \mathbf{U}\mathbf{I}\mathbf{U}^{\mathsf{H}} = \mathbf{P}\,.$$

Das Bild dieses Projektors $\mathrm{Bild}\,(\mathbf{P})$ ist einfach $\mathrm{Bild}\,(\mathbf{U})$, das Bild von $\mathbf{U}$, denn $\mathbf{U}\mathbf{U}^{\mathsf{H}}\mathbf{x}$ ist eine lineare Kombination der Spalten aus $\mathbf{U}$.

Wenn $\mathbf{U}$ zum Beispiel eine $n \times 2$ Matrix

$$\mathbf{U} = \begin{bmatrix} | & | \\ \mathbf{q}_1 & \mathbf{q}_2 \\ | & | \end{bmatrix}$$

mit den Vektoren $\mathbf{q}_1,\ \mathbf{q}_2$ orthogonal aufeinander und euklidischer Norm 1 (d. h. $\mathbf{U}^{\mathsf{H}}\mathbf{U} = \mathbf{I} \in \mathbb{C}^{2\times 2}$) ist, dann haben wir den orthogonalen Projektor $\mathbf{P}$, welcher auf den Unterraum $\mathrm{Span}\{\mathbf{q}_1,\ \mathbf{q}_2\}$ projiziert:

$$\mathbf{P} = \mathbf{U}\mathbf{U}^{\mathsf{H}} = \begin{bmatrix} | & | \\ \mathbf{q}_1 & \mathbf{q}_2 \\ | & | \end{bmatrix} \begin{bmatrix} \text{—} & \mathbf{q}_1^{\mathsf{H}} & \text{—} \\ \text{—} & \mathbf{q}_2^{\mathsf{H}} & \text{—} \end{bmatrix}$$
$$= \mathbf{q}_1\mathbf{q}_1^{\mathsf{H}} + \mathbf{q}_2\mathbf{q}_2^{\mathsf{H}}\,.$$

Dieser Projektor ist die Summe von Rang eins Matrizen der Form $\mathbf{q}_i\mathbf{q}_i^{\mathsf{H}}$. Alle diese Matrizen repräsentieren die Projektionen auf die einzelnen Koordinatenachsen $\mathbf{q}_i$ und wenn wir diese einzelnen Projektionen aufsummieren, so erhalten wir die Projektion auf den Unterraum. Genau dies haben wir im Beweis des Gram-Schmidt Algorithmus gemacht!

⌟

**Definition 4.5.0.7. Orthogonaler Projektor auf einen Unterraum**

Sei $\mathbf{U}$ eine $n \times k$ Matrix mit paarweise orthogonalen Spaltenvektoren $q_1, \ldots,\ q_k$ welche Norm 1 haben. Der orthogonale Projektor $\mathbf{P}$, der auf den Unterraum $\mathrm{Span}\{q_1, .., q_k\}$ projiziert, ist

$$\mathbf{P} = \mathbf{U}\,\mathbf{U}^{\mathsf{H}}\,.$$

**Beispiel 4.5.0.8 (Schiefer Projektor auf einem Unterraum)**
Im Folgenden erweitern wir auch das Beispiel des schiefen Projektors.

Seien zwei Matrizen $\mathbf{U},\ \mathbf{V}$, so dass $\mathbf{V}^{\mathsf{H}}\mathbf{U} = \mathbf{I}$. Bei solchen Matrizen sagen wir, dass die Spalten von $\mathbf{U}$ und $\mathbf{V}$ *biorthogonal* sind.

Für diese beiden Matrizen definieren wir den schiefen Projektor

$$\mathbf{P} = \mathbf{U}\mathbf{V}^{\mathsf{H}}\,.$$

Dieser Projektor projiziert auf den Unterraum der Spaltenvektoren von $\mathbf{U}$. Das Bild und der Kern dieser Projektion sind:

$$\mathrm{Bild}\,(\mathbf{P}) = \mathrm{Bild}\,(\mathbf{U})\,,$$
$$\mathrm{Kern}\,\mathbf{P} = \mathrm{Kern}\,(\mathbf{V}^{\mathsf{H}}) = (\mathrm{Bild}\,(\mathbf{V}))^{\perp}\,.$$

Dass Kern $(\mathbf{U}^H) = (\text{Bild}\,(\mathbf{U}))^\perp$ ist, folgt direkt aus dem Fundamentalsatz der linearen Algebra.

⌟

**Definition 4.5.0.9. Schiefer Projektor auf einen Unterraum**

Wenn $\mathbf{U}$ und $\mathbf{V}$ zwei Matrizen mit der Eigenschaft $\mathbf{V}^H\mathbf{U} = \mathbf{I}$ sind, dann kann ein schiefer Projektor $\mathbf{P}$, welcher auf den Unterraum der Spaltenvektoren von $\mathbf{U}$ projiziert, definiert werden als

$$\mathbf{P} = \mathbf{U}\mathbf{V}^H .$$

## 4.6 Lineare Abbildungen auf Räume mit Skalarprodukt

**Definition 4.6.0.1. Funktional**

Sei $V$ ein linearer Raum über $\mathbb{R}/\mathbb{C}$. Eine lineare Abbildung $\varphi : V \to \mathbb{R}/\mathbb{C}$ heisst lineares Funktional (auf $V$).

**Beispiel 4.6.0.2.** Sei $V$ ein linearer Raum über $\mathbb{R}/\mathbb{C}$ mit dem Skalarprodukt $< \cdot, \cdot >$ und sei $w \in V$ gegeben. Wir definieren die Abbildung:

$$\varphi(v) =< w, v > \text{ , für alle } v \in V .$$

$\varphi$ ist offensichtlich ein lineares Funktional. ⌟

**Beispiel 4.6.0.3.** Sei $V = C[0,1]$ mit dem Standard-$L^2$-Skalarprodukt und $\varphi : C[0,1] \to \mathbb{R}$ mit $\varphi(f) = f(\frac{1}{2})$ für jedes $f \in C[0,1]$. Dieses $\varphi$ ist ein lineares Funktional. ⌟

**Satz 4.6.0.4. Riesz Darstellungssatz**

*Sei $V$ ein linearer Raum von Dimension $n \in \mathbb{N}$ über $\mathbb{R}/\mathbb{C}$ mit dem Skalarprodukt $\langle \cdot, \cdot \rangle$ und sei $\varphi : V \to \mathbb{R}/\mathbb{C}$ ein lineares Funktional.*

*(a) Es gibt ein eindeutiges $w \in V$, so dass*

$$\varphi(v) = \langle w, v \rangle \text{ , } \textit{für alle } v \in V .$$

*(b) Sei $b_1, \ldots, b_n$ eine orthonormale Basis in $V$. Dann:*

$$w = \overline{\varphi(b_1)}b_1 + \overline{\varphi(b_2)}b_2 + \ldots + \overline{\varphi(b_n)}b_n .$$

*Beweis.* Da $V$ Dimension $n$ hat, gibt es (dank Gram-Schmidt) eine orthonormale Basis $b_1, \ldots, b_n$ in $V$. Somit lässt sich jedes beliebige $v \in V$ schreiben

$$v = \sum_{i-1}^{n} \langle b_i, v \rangle b_i$$

und die Linearität von $\varphi$ und die Anti-Linearität vom Skalarprodukt im ersten Argument liefern:

$$\begin{aligned} \varphi(v) &= \varphi\left(\sum_{i=1}^{n} \langle b_i, v \rangle b_i\right) = \sum_{i=1}^{n} \langle b_i, v \rangle \varphi(b_i) \\ &= \sum_{i=1}^{n} \left\langle \overline{\varphi(b_i)} b_i, v \right\rangle = \left\langle \sum_{i=1}^{n} \overline{\varphi(b_i)} b_i, v \right\rangle \\ &= \langle w, v \rangle . \end{aligned}$$

Somit haben wir bewiesen: es gibt $w = \overline{\varphi(b_1)}b_1 + \overline{\varphi(b_2)}b_2 + \ldots + \overline{\varphi(b_n)}b_n$ so dass $\varphi(v) = \langle w, v\rangle$ für alle $v \in V$.
Sei nun $y \in V$ so dass $\varphi(v) = \langle y, v\rangle$ für alle $v \in V$. Durch Subtraktion und Linearität erhalten wir, dass $\langle y - w, v\rangle = 0$ für alle $v \in V$. Wählen wir jetzt $v = y - w$, so erhalten wir $\langle y - w, y - w\rangle = 0$ und die Definition vom Skalarprodukt ergibt $y - w = 0$, also $y = w$. Somit ist $w$ eindeutig, es hängt nicht von der Wahl der orthnormalen Basis ab.

□

**Beispiel 4.6.0.5.** Sei $\varphi : \mathbb{R}^3 \to \mathbb{R}$ mit $\varphi(\mathbf{v}) = v_2$. Wir wollen $\varphi$ mit dem Satz von Riesz als ein Skalarprodukt darstellen, d. h. $\mathbf{w} \in \mathbb{R}^3$ finden, so dass $\varphi(\mathbf{v}) = \langle \mathbf{w}, \mathbf{v}\rangle$ für alle $\mathbf{v} \in \mathbb{R}^3$. Wenn wir die Standardbasis verwenden, dann:

$$\mathbf{w} = \varphi(\mathbf{e}_1)\mathbf{e}_1 + \varphi(\mathbf{e}_2)\mathbf{e}_2 + \varphi(\mathbf{e}_3)\mathbf{e}_3 = 0\mathbf{e}_1 + 1\mathbf{e}_2 + 0\mathbf{e}_3 = \mathbf{e}_2\,.$$

Wir können überprüfen, dass eine andere Wahl der orthonormalen Basis dasselbe $\mathbf{w}$ ergibt. Sei die orthonormale Basis

$$\mathbf{u}_1 = \frac{1}{3}\begin{bmatrix}-2\\-1\\2\end{bmatrix},\ \mathbf{u}_2 = \frac{1}{3}\begin{bmatrix}1\\2\\2\end{bmatrix},\ \mathbf{u}_3 = \frac{1}{3}\begin{bmatrix}-2\\2\\-1\end{bmatrix}.$$

Berechnen wir nun $\mathbf{w}$ aus dem Darstellungssatz von Riesz:

$$\begin{aligned}\mathbf{w} &= \varphi(\mathbf{u}_1)\mathbf{u}_1 + \varphi(\mathbf{u}_2)\mathbf{u}_2 + \varphi(\mathbf{u}_3)\mathbf{u}_3 = -\frac{1}{3}\mathbf{u}_1 + 2\frac{1}{3}\mathbf{u}_2 + 2\frac{1}{3}\mathbf{u}_3\\ &= -\frac{1}{3}\frac{1}{3}\begin{bmatrix}-2\\-1\\2\end{bmatrix} + 2\frac{1}{3}\frac{1}{3}\begin{bmatrix}1\\2\\2\end{bmatrix} + 2\frac{1}{3}\frac{1}{3}\begin{bmatrix}-2\\2\\-1\end{bmatrix}\\ &= \frac{1}{9}\begin{bmatrix}2+2-4\\1+4+4\\-2+4-2\end{bmatrix} = \mathbf{e}_2\,.\end{aligned}$$

⌟

**Beispiel 4.6.0.6.** Sei $V = \mathcal{P}_3[0,1]$ mit dem Standard-$L^2$-Skalarprodukt und $\varphi : \mathcal{P}_3[0,1] \to \mathbb{R}$ mit $\varphi(p) = p(\frac{1}{2})$. Wenn wir die orthonormale Basis der entsprechenden Lagrange-Polynome betrachten

$$u_1(t) = 1\,,\quad u_2(t) = \sqrt{3}(2t-1)\,,\quad u_3(t) = \sqrt{5}(6t^2 - 6t + 1)\,,$$

so erhalten wir $\varphi(u_1) = 1$, $\varphi(u_2) = u_2(\frac{1}{2}) = 0$, $\varphi(u_3) = u_3(\frac{1}{2}) = -\frac{\sqrt{5}}{2}$ und
$w(t) = 1 + 0 - \frac{5}{2}(6t^2 - 6t + 1) = -15t^2 + 15t - \frac{3}{2}$. So ist $\varphi(p) = \langle w, p\rangle$ für jedes $p\ in \mathcal{P}_3[0,1]$. ⌟

**Bemerkung 4.6.0.7 (Dualer Raum)** Die Menge linearen Funktionalen auf $V$ bildet einen linearen Raum, der duale Raum $V'$. Der Darstellungssatz von Riesz gibt einen kanonischen Isomorphismus zwischen $V'$ und $V$, falls $V$ endlichdimensional ist.
Wenn $V$ unendlichdimensional ist, dann braucht man mehr Voraussetzungen damit der Darstellungssatz von Riesz weiterhin erfüllt ist: die Cauchy-Folgen in der vom Skalaprodukt induzierten Norm müssen in $V$ konvergent sein (d. h. $V$ muss ein Hilbert-Raum sein) und $V'$ muss die Menge der stetig linearen Funktionale sein, d. h. $\varphi$ muss nicht nur linear , sondern auch stetig sein.
Der Darstellungssatz von Riesz hat sehr wichtige Anwendungen, z.B. in der Signalverarbeitung und Quantenphysik. ⌟

**Bemerkung 4.6.0.8 (Abbildungsmatrix und Skalarprodukt)** Seien $V$ und $W$ zwei endlichdimensionale lineare Räume mit den jeweiligen Skalarpodukte $\langle\cdot,\cdot\rangle_V$ und $\langle\cdot,\cdot\rangle_W$. Weiter seien $v_1, \ldots, v_n$ und $w_1, \ldots, w_m$

orthonormale Basen in $V$ und $W$ und $F : V \to W$ eine lineare Abbildung. Die Matrix, die die Abblidung $F$ in diesen Basen darstellt, lässt sich aufgrund der Orthogonalität leicht berechnen, denn:

$$Fv_j = \sum_{i=1}^{m} \langle w_i, Fv_j \rangle_W w_i \Rightarrow a_{ij} = \langle w_i, Fv_j \rangle_W$$

also

$$\mathbf{A} = \begin{bmatrix} \langle w_1, Fv_1 \rangle_W \dots \langle w_1, Fv_n \rangle_W \\ \langle w_2, Fv_1 \rangle_W \dots \langle w_2, Fv_n \rangle_W \\ \dots\dots\dots \\ \langle w_m, Fv_1 \rangle_W \dots \langle w_m, Fv_n \rangle_W \end{bmatrix}$$

. ⌟

**Definition 4.6.0.9. Adjungierte Abblidung**

Seien $V$ und $W$ zwei lineare Räume mit den jeweiligen Skalarpodukte $\langle \cdot, \cdot \rangle_V$ und $\langle \cdot, \cdot \rangle_W$. Eine Abbildung $F^* : W \to V$ heisst die adjungierte Abbildung einer linearen Abbildung $F : V \to W$, wenn

$$\langle Fv, w \rangle_W = \langle v, F^*(w) \rangle_V \,, \text{ für alle } v \in V \,, w \in W \,.$$

**Bemerkung 4.6.0.10.** Wenn die adjungierte Abbildung existiert, dann ist sie eindeutig und linear.
Schauen wir uns zuerst die Eindeutigkeit an. Seien $F^*$ und $G$ zwei solche adjungierte von $F$, dann muss $\langle v, F^*(w) \rangle_V = \langle v, G(w) \rangle_V$, für alle $v \in V$ und $w \in W$. Die linearität vom Skalarprodukt im zweiten Argument ergibt $\langle v, F^*(w) - G(w) \rangle_V = 0$ für alle $v \in V$ und $w \in W$. Wählen wir $v = F^*(w) - G(w)$, erhalten wir $F^*(w) - G(w) = 0$ für alle $w \in W$, d. h. $G = F^*$.
Jetzt untersuchen wir die Linearität. Seien beliebige $u, w \in W$, $v \in V$, $\alpha, \beta$ Skalare. Dann

$$\begin{aligned} \langle v, F^*(\alpha u + \beta w) \rangle_V &= \langle Fv, \alpha u + \beta w \rangle_W = \alpha \langle Fv, u \rangle_W + \beta \langle Fv, w \rangle_W = \\ &= \alpha \langle v, F^* u \rangle_W + \beta \langle v, F^* w \rangle_W = \langle v, \alpha F^*(u) + \beta F^*(w) \rangle_V \,, \end{aligned}$$

für alle $v \in V$, was zu

$$F^*(\alpha u + \beta w) = \alpha F^*(u) + \beta F^*(w)$$

mit beliebige $u, w \in W$ und $\alpha, \beta$ Skalare führt, d. h. $F^*$ ist linear. ⌟

**Satz 4.6.0.11. Adjungierte Abbildung für endlich dimensionale Räume**

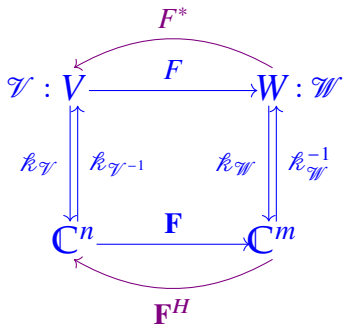
*Seien $V$ und $W$ zwei endlichdimensionale lineare Räume mit den jeweiligen Skalarpodukte $\langle \cdot, \cdot \rangle_V$ und $\langle \cdot, \cdot \rangle_W$ und sei eine lineare Abbildung $F : V \to W$ mit der $n \times m$ Darstellungsmatrix $\mathbf{F}$ in orthonormalen Basen in $V$ und $W$. Dann:*

*(a) $F$ hat eine adjungierte Abbildung $F^*$ und*
*(b) $F^*$ wird von $\mathbf{F}^H$ dargestellt.*

**Bemerkung 4.6.0.12.** Dieser Satz erklärt die gängige Notation $\mathbf{F}^*$ für die hermitesche Transponierte einer Matrix, die wir $\mathbf{F}^H$ notiert haben. ⌟

*Beweis.* Seien $\mathscr{V} = \{v_1, \dots, v_n\}$ und $\mathscr{W} = \{w_1, \dots, w_m\}$ orthonormale Basen in $V$ und $W$. Dann ist $\mathbf{F} = \left[\langle w_i, Fv_j \rangle_W\right]_{i=1,\dots,m,\, j=1,\dots,n}$ und $\mathbf{F}^H = \left[\langle Fv_j, w_i \rangle_W\right]_{j=1,\dots,n,\, i=1,\dots,m}$. Wir definieren nun $F^* : W \to V$. Für $w \in W$ notieren wir mit $\mathbf{w}$ den Koordinatenvektor in der Basis $w_1, \dots, w_m$ und bilden $\mathbf{v} = \mathbf{F}^H \mathbf{w}$, der als Koordinatenvektor des Ergebnisses $F^* w$ dient. Mit der bekannten Notation der Koordinatenabbildungen $\mathscr{k}_{\mathscr{V}}$ und $\mathscr{k}_{\mathscr{W}}$ können wir die adjungierte Abbildung auch so schreiben: $F^H := \mathscr{k}_{\mathscr{V}}^{-1} \mathbf{F}^H \mathscr{k}_{\mathscr{W}}$. Sei auch $u \in V$

beliebig mit $\mathbf{u} = \mathcal{k}_{\mathscr{V}} u$. Dann gilt $\langle F^* w, u\rangle_V = \mathbf{v}^H \mathbf{u} = (\mathbf{F}^H \mathbf{w})^H \mathbf{u} = \mathbf{w}^H \mathbf{F}\mathbf{u} = \langle w, Fu\rangle_V$ also ist $F^*$ die adjungierte Abbildung von $F$.




□

**Beispiel 4.6.0.13 (Eine Abbildung mit Adjungierte.)** Sei $V = \mathbb{C}^n$, $W = \mathbb{C}^m$ mit dem jeweiligen Euklidischen Skalaprodukt, $\mathbf{A} \in \mathbb{C}^{m\times n}$ und die entsprechende lineare Abbildung $T_{\mathbf{A}} : V \to W$, $T_{\mathbf{A}}(\mathbf{x}) = \mathbf{A}\mathbf{x}$. Dann ist $\langle T_{\mathbf{A}}\mathbf{x}, \mathbf{y}\rangle = \langle \mathbf{A}\mathbf{x}, \mathbf{y}\rangle = \mathbf{x}^H \mathbf{A}^H \mathbf{y} = \langle \mathbf{x}, \mathbf{A}^H \mathbf{y}\rangle$. Somit ist die adjungierte Abbildung von $T_{\mathbf{A}}$ die Abbildung $T^*_{\mathbf{A}} = T_{\mathbf{A}^H}$, die von der Matrix $\mathbf{A}^H$ dargestellt wird. ⌟

**Beispiel 4.6.0.14 (Eine Abbildung auf $\mathscr{P}$ mit Adjungierte)** Sei $V = W = \mathscr{P}[0,1]$ mit dem Standard-$L^2$-Skalarprodukt. Sei ein festes $p \in \mathscr{P}[0,1]$. Mit diesem $p$ definieren wir die lienare Abbildung $F : \mathscr{P}[0,1] \to \mathscr{P}[0,1]$ mit $F(q) = pq$ für jedes $q \in \mathscr{P}[0,1]$. Dann:

$$\langle Fq, g\rangle = \int_0^1 \overline{p(t)q(t)}g(t)dt = \int_0^1 \overline{q(t)p(t)}g(t)dt = \langle q, F^* g\rangle\,,$$

wobei $F^* g = \overline{p}g$. Somit hat dies Abbildung $F$ eine adjungierte $F^*$. ⌟

**Bemerkung 4.6.0.15.** Die adjungierte Abbildung existiert nicht immer!] Achtung: im Fall unendlicher Dimension nicht jede lineare Abbildung hat eine adjungierte Abbildung! ⌟

**Beispiel 4.6.0.16 (Eine Abbildung auf $\mathscr{P}$ ohne Adjungierte)** Die Abbildung $F : \mathscr{P} \to \mathscr{P}$, $F(p) = p'$ (Ableitung) hat keine Adjungierte.
Der Beweis basiert auf Widerspruch: *reductio ab absurdum*: nehmen wir mal an, diese $F$ hat eine Adjungierte $F^*$. Sei $q = F^* p_0$ das Bild vom ersten Monom $p_0(t) = 1$ durch $F^*$. Für ein beliebiges Element $p \in \mathscr{P}$ gilt:

$$\langle p, q\rangle = \langle p, F^* p_0\rangle = \langle Fp, p_0\rangle = \int_0^1 p'(t)dt = p(1) - p(0)\,.$$

Für $p(t) = t^2(1-t)^2 q(t)$ gibt das $\langle p, q\rangle = 0$, also

$$0 = \langle p, q\rangle = \int_0^1 \overline{p(t)}q(t)dt = \int_0^1 t^2(1-t)^2|q(t)|^2 dt = \|r\|\,.$$

mit $r(t) = t(1-t)q(t)$. Da $\|\cdot\|$ eine Norm ist, erhalten wir $t(1-t)q(t) = 0$ für alle $t \in [0,1]$, d. h. $q = 0$. Wenn wir diesen $q$ in der ersten Gleichung einsetzen, erhalten wir

$$0 = \langle p, q\rangle = p(1) - p(0)\,,$$

für jedes $p \in \mathscr{P}$, was offensichtlich falsch ist. Somit muss die Annahme, dass $F$ eine Adjungierte hat, falsch sein; das Gegenteil ist dann wahr: diese $F$ hat keine Adjungierte. ⌟

# Kapitel 5

# Ausgleichsrechnung

## Inhalt


5.1 Ausgleichsrechnung (»Least Squares«) . . . . . 157
5.2 Lösung mit der Normalengleichung . . . . . 160
5.3 Lösung mit der QR-Zerlegung . . . . . 162


## 5.1 Ausgleichsrechnung (»Least Squares«)

Die Ausgleichsrechnung ist ein wichtiges Werkzeug für Ingenieure, um verschiedene komplexe Systeme durch geschicktes Approximieren einfach darzustellen und zu lösen. Die Lösung einer Ausgleichsrechnung liefert uns die bestmögliche Approximation in einem gewissen Sinn (normalerweise bezüglich der euklidischen Norm) für unser Problem. Hier besprechen wir die *lineare Ausgleichsrechnung* im Sinne der *kleinsten Quadrate*. Um die Theorie zu verdeutlichen, betrachten wir ein Beispiel:

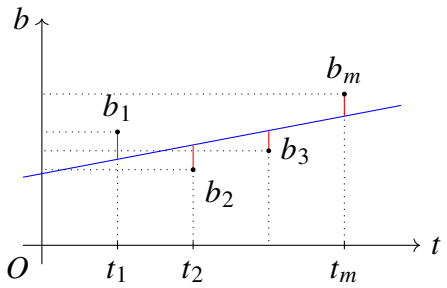
**Beispiel 5.1.0.1 (Lineares Modell)**
Wir betrachten ein lineares Modell in den Parametern $\mathbf{d} \in \mathbb{R}^s, c \in \mathbb{R}$.

$$\mathbb{R} \ni b = \mathbf{d}^\mathsf{T}\mathbf{t} + c \qquad \mathbf{t} \in \mathbb{R}^s$$

Für die Variable $\mathbf{t}$ werden Messpunkte $\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_m$ eingesetzt. Die Werte von $b$ zu diesen Messpunkten werden gemessen: $b_1, b_2, \ldots, b_m$. Das Ziel ist, aus diesen Daten, die ursprünglich unbekannten Parameter $\mathbf{d} \in \mathbb{R}^s, c \in \mathbb{R}$ zu finden.

Nehmen wir nun einfachheitshalber an, dass die Messpunkte nur eindimensional sind ($s = 1$). Nach unserem Modell gilt $b = dt + c$. Da $\mathbf{d} \in \mathbb{R}^1$ ist, kann man die vektorielle Schreibweise zunächst ignorieren. Es gelten für die Messpunkte:

$$\begin{aligned} dt_1 + c &= b_1 \\ dt_2 + c &= b_2 \\ &\vdots \\ dt_m + c &= b_m \,. \end{aligned}$$




Diese Gleichungen gehen davon aus, dass keine Messwertfehler vorliegen. In der Praxis liegen aber Messfehler vor, deswegen schreiben wir die Gleichungen in die Form $dt + c - b = r$ um:

$$\begin{aligned} dt_1 + c - b_1 &= r_1 \\ dt_2 + c - b_2 &= r_2 \\ &\vdots \\ dt_m + c - b_m &= r_m \,. \end{aligned}$$

Den neu vorliegenden Rest $r_i$ nennen wir *Residuum*. Der Residuenvektor fasst alle Residuen in einem Vektor zusammen:

$$\mathbf{r} = \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_m \end{bmatrix}.$$

Was wir nun nach der Definition der Ausgleichsrechnung machen möchten, ist die euklidische Norm des Residuums $\|\mathbf{r}\|_2^2$ zu minimieren. Das machen wir, um die Lösung möglichst genau zu approximieren.

Für die Optimierung führen wir die Parameter $\tilde{c} \in \mathbb{R}$ und $\tilde{\mathbf{d}} \in \mathbb{R}^s$ ein. Die optimalen Werte $c, \mathbf{d}$ dieser Parameter sind das Ergebnis dieser Ausgleichsrechnung.

Wir schreiben:

$$\min_{\tilde{\mathbf{d}} \in \mathbb{R}^s, \tilde{c} \in \mathbb{R}} \{|r_1|^2 + \ldots + |r_m|^2\} = \min_{\tilde{\mathbf{d}} \in \mathbb{R}^s, \tilde{c} \in \mathbb{R}} \sum_{i=1}^{m} |\tilde{\mathbf{d}}^\mathsf{T} \mathbf{t}_i + \tilde{c} - b_i|^2. = \min_{\tilde{c} \in \mathbb{R}, \tilde{\mathbf{d}} \in \mathbb{R}^s} \sum_{i=1}^{m} |1 \cdot \tilde{c} + \mathbf{t}_i^\mathsf{T} \tilde{\mathbf{d}} - b_i|^2.$$

Da die gesuchten Parameter $\tilde{c}, \tilde{\mathbf{d}}$ nur linear vorkommen, sprechen wir von der *linearen* Ausgleichsrechnung.

Die Gleichungen der Messungen können wir auch in der Form $\mathbf{A}\mathbf{x} - \mathbf{b} = \mathbf{r}$ schreiben, wobei

$$\mathbf{x} = \begin{bmatrix} \tilde{c} \\ \tilde{\mathbf{d}} \end{bmatrix} \in \mathbb{R}^n \,,$$

$$\mathbf{A} = \begin{bmatrix} 1 & \mathbf{t}_1^\mathsf{T} \\ 1 & \mathbf{t}_2^\mathsf{T} \\ \vdots & \vdots \\ 1 & \mathbf{t}_m^\mathsf{T} \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}.$$

Die Matrix $\mathbf{A}$ hat $m$ Zeilen (z. B. Messungen) und $n = 1 + s$ Spalten.

Die zu minimierende Grösse lautet nun $\min_{\mathbf{x}\in\mathbb{R}^n} \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2$.

Im Allgemeinen:

**Definition 5.1.0.2. Lineare Ausgleichsrechnung als lineares Gleichungssystem**

Sei eine $m \times n$ Matrix $\mathbf{A}$ und der Vektor $\mathbf{b}$ mit $m$ Einträgen. Gesucht ist ein $\hat{\mathbf{x}}$, so dass

$$\|\mathbf{A}\hat{\mathbf{x}}-\mathbf{b}\|_2 = \min_{\mathbf{x}\in\mathbb{R}^n} \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2 .$$

**Beispiel 5.1.0.3 (Lineares Modell - konkret)**
Seien die Messungen[1]:

$$\begin{aligned} t_1 &= 0, \quad b_1 = 6 \\ t_2 &= 1, \quad b_2 = 0 \\ t_3 &= 2, \quad b_3 = 0 . \end{aligned}$$

Demnach sieht unsere Ausgleichsrechnung in Matrixform folgendermassen aus:

$$\begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} c \\ d \end{bmatrix} = \begin{bmatrix} 6 \\ 0 \\ 0 \end{bmatrix}.$$

Dieses Gleichungssystem hat keine Lösung. Wenn das mathematische Modell stimmen soll, so müssen (Mess-) Fehler vorliegen. Sind Messungen gegeben, so schlägt die Ausgleichsrechnung (auch Methode der kleinsten Quadrate genannt) Modellparameter $c$ und $d$ vor, so dass die euklidische Norm vom Residuenvektor $\|\mathbf{r}\|_2^2$ minimal ist.

⌟

Die Ausgleichsrechnung ist auch eine sehr nützliche und einfache Methode, um komplizierte Modelle zu vereinfachen.

**Beispiel 5.1.0.4 (Modell im Unterraum)**
Sei $f \in V$, wobei $V$ ein $\infty$-dimensionaler linearer Raum (z. B. $L^2$) ist. Sei nun $V_n \subset V$ ein endlichdimensionaler Unterraum. Wir wollen eine Funktion $f_n \in V_n$ finden, die $f \in V$ im gewissen Sinne approximiert.

Sei $b_1, \ldots, b_n$ eine Basis in $V_n$. Das Ziel ist also:

$$f \approx f_n = \sum_{j=1}^{n} x_j b_j .$$

Ist das System z. B. zeitabhängig, so wählen wir eine zeitabhängige Basis:

$$f(t) \approx f_n(t) = \sum_{j=1}^{n} x_j b_j(t) .$$

Verwenden wir nur die Werte/Messungen $y_i = f(t_i)$, so bekommen wir sehr einfach eine Approximation: Für die gegebenen $t_i, y_i$, $i = 1, 2, \ldots, m$, finden wir die Koeffizienten $x_1, .., x_n$, sodass die Summe der Quadrate der punktweisen Approximationsfehler

$$\sum_{i=1}^{m} |f_n(t_i) - y_i|^2 ,$$

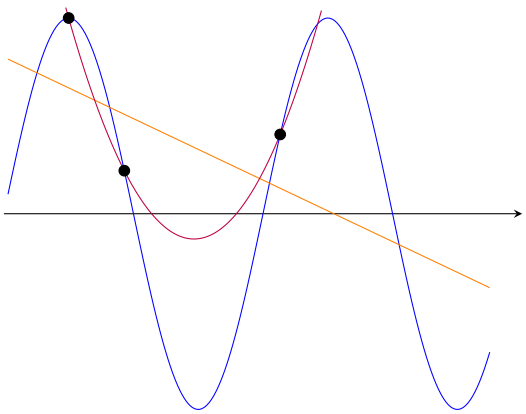
[1] Beispiel 1 im Kapitel 4.3, Seite 218 in Str09.

d. h. die euklidische Norm des Residuums (zum Quadrat), minimal ist.

Nehmen wir beispielsweise an, dass $V = L^2(0,1) = \{f : [0,1] \to \mathbb{R}, \int_0^1 |f(t)|^2 dt < \infty\}$ und wählen die Basis der Monome $b_i(t) = t^{i-1}$, so dass wir $V$ durch eine Polynommenge $\mathscr{P}$ beschreiben können ($V_n = \mathscr{P}_n$). Dieser Ansatz produziert dann das »beste« Polynom $p_n$ vom Grad maximal $n-1$, so dass die euklidische Norm des Residuenvektor

$$\sum_{i=1}^{m} |p_n(t_i) - y_i|^2$$

minimal ist.



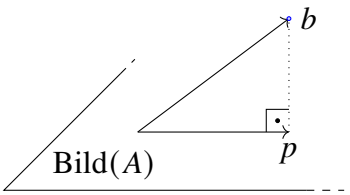
## 5.2 Lösung mit der Normalengleichung

Kehren wir nun zur Aufgabe der linearen Ausgleichsrechnung 5.1 zurück. Gegeben sind $\mathbf{A}$ und $\mathbf{b}$. Wenn $\mathbf{b}$ nicht im Bild $\mathbf{A}$ liegt, hat das lineare Gleichungssystem $\mathbf{Ax} = \mathbf{b}$ keine Lösung.

Nach dem Fundamentalsatz der linearen Algebra können wir $\mathbf{b} \in \mathbb{R}^m$ eindeutig in zwei orthogonale Komponenten zerlegen:

$$\mathbf{b} = \mathbf{p} + \mathbf{c}\,, \quad \text{mit } \mathbf{p} \in \operatorname{Bild}\mathbf{A}\,, \ \mathbf{c} \in \operatorname{Kern}\mathbf{A}^{\mathsf{T}}\,, \quad \text{mit } \mathbf{p} \perp \mathbf{c}\,.$$

Die Orthogonalität bewirkt, dass $\mathbf{p}$ die (im Sinne der euklidischen Norm aus $\mathbb{R}^m$) bestmögliche Approximation aus Bild $\mathbf{A}$ an $\mathbf{b}$ ist.




Da $\mathbf{p} \in \operatorname{Bild}\mathbf{A}$, gibt es eine Lösung $\widehat{\mathbf{x}} \in \mathbb{R}^n$ für das System $\mathbf{A}\widehat{\mathbf{x}} = \mathbf{p}$. Diese Lösung ist somit die Lösung des linearen Ausgleichsproblems, denn

$$\mathbf{b} - \mathbf{A}\widehat{\mathbf{x}} = \mathbf{b} - \mathbf{p} = \mathbf{c} \perp \operatorname{Bild}\mathbf{A} \Longrightarrow \min_{\mathbf{y} \in \operatorname{Bild}\mathbf{A}} \|\mathbf{b} - \mathbf{y}\|_2^2 \geq \|\mathbf{c}\|_2^2\,.$$

Aus dem Fundamentalsatz der linearen Algebra folgt auch:

$$\mathbf{c} \perp \operatorname{Bild} \mathbf{A} \iff \mathbf{c} \in \operatorname{Kern} \mathbf{A}^\mathsf{T} .$$

Es gilt also:

$$\mathbf{A}^\mathsf{T}\mathbf{b} = \mathbf{A}^\mathsf{T}(\mathbf{p}+\mathbf{c}) = \mathbf{A}^\mathsf{T}\mathbf{p} + \mathbf{A}^\mathsf{T}\mathbf{c} = \mathbf{A}^\mathsf{T}\mathbf{p} .$$

Wenn wir nun die Ersetzung $\mathbf{A}\widehat{\mathbf{x}} = \mathbf{p}$ durchführen, erhalten wir die Normalengleichung.

$$\mathbf{A}^\mathsf{T}\mathbf{A}\widehat{\mathbf{x}} = \mathbf{A}^\mathsf{T}\mathbf{b} \tag{5.2.0.1}$$

Das ursprüngliche lineare Gleichungssystem ist überbestimmt, aber hier erhalten wir ein modifiziertes lineares Gleichungssystem, das mindestens eine Lösung hat, denn die neue rechte Seite $\mathbf{A}^\mathsf{T}\mathbf{b}$ liegt im $\operatorname{Bild} \mathbf{A}^\mathsf{T}$.

**Definition 5.2.0.2. Normalengleichung**

Die Gleichung

$$\mathbf{A}^\mathsf{T}\mathbf{A}\mathbf{x} = \mathbf{A}^\mathsf{T}\mathbf{b}$$

heisst *Normalengleichung*.

**Satz 5.2.0.3. Lösung der Ausgleichsrechnung mit der Normalengleichung**

*Sei eine Matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ gegeben. Falls der Rang dieser Matrix gleich $n$ ist, so ist $\mathbf{A}^\mathsf{T}\mathbf{A}$ invertierbar. In diesem Fall hat die Normalengleichung also genau eine Lösung, die die Lösung des linearen Ausgleichsproblems ist.*

*Beweis.* $\mathbf{A}^\mathsf{T}\mathbf{A} \in \mathbb{R}^{n\times n}$. Nach den Bemerkungen 1.2.1.2 und 1.2.1.3 ist sie invertierbar dann und nur dann, wenn das homogene System $\mathbf{A}^\mathsf{T}\mathbf{A}\mathbf{x} = \mathbf{0}$ nur die triviale Lösung $\mathbf{x} = \mathbf{0}$ hat. Rechnen wir dies nach:

$$\begin{aligned} &\mathbf{A}^\mathsf{T}\mathbf{A}\mathbf{x} = \mathbf{0} \\ &\Rightarrow \mathbf{x}^\mathsf{T}\mathbf{A}^\mathsf{T}\mathbf{A}\mathbf{x} = 0 \\ &\Leftrightarrow (\mathbf{A}\mathbf{x})^\mathsf{T}(\mathbf{A}\mathbf{x}) = 0 \\ &\Leftrightarrow \|\mathbf{A}\mathbf{x}\|_2^2 = 0 \\ &\Rightarrow \mathbf{A}\mathbf{x} = \mathbf{0} . \end{aligned}$$

Da $\mathbf{A}$ Rang $n$ hat, gilt $\mathbf{x} = \mathbf{0}$.

□

Wenn die Matrix $\mathbf{A}$ Rang $n$ hat, reduziert sich die lineare Ausgleichsrechnung auf eine Multiplikation mit $\mathbf{A}^\mathsf{T}$ und die Lösung eines linearen Gleichungssystems mit einer symmetrischen $n \times n$ Matrix.
Falls die Matrix $\mathbf{A}$ nicht Rang $n$ hat, dann hat die Normalengleichung unendlich viele Lösungen. In diesem Fall muss man eine Wahl für die freien Variablen in der Gauss-Elimination treffen.

Diese Methode ist einfach, birgt aber grosse Schwierigkeiten in der Praxis. Die Bedingung, dass der Rang der Matrix $\mathbf{A}$ genau $n$ ist, kann im Allgemeinen **nicht** garantiert werden. Schlimmer noch: Manche Spalten oder Zeilen der Matrix $\mathbf{A}$ könnten aufgrund von Rundungsfehlern rein mathematisch zwar linear unabhängig sein, aber numerisch und praktisch nicht, was zu sehr grossen Fehlern im Ergebnis führen kann. Darum verwenden professionelle Löser nie die Normalengleichung, sondern die Singulärwertzerlegung, die wir später auch diskutieren werden.

Im Falle grosser Matrizen ist mit wesentlichen Schwierigkeiten zu rechnen, auch wenn die Matrix $\mathbf{A}$ Rang $n$ hat. Ohne genauer auf die Details einzugehen, ist zu erwarten, dass die Rundungsfehler sich stark negativ auf

die Qualität des Ergebnisses der Gauss-Elimination mit der Matrix $\mathbf{A}^\top\mathbf{A}$ auswirken, viel schlimmer, als wenn man direkt die Matrix $\mathbf{A}$ verwenden würde. Ausserdem lässt sich eine eventuelle Struktur (viele Nullen) in der Matrix $\mathbf{A}$ im Lösungsweg über die Normalengleichung kaum ausnutzen.

## 5.3 Lösung mit der QR-Zerlegung

Wir schauen uns eine alternative Lösungsmethode an, um numerische Fehler für grosse Matrizen zu vermeiden. Dazu bedienen wir uns wieder bei der QR-Zerlegung:

$$\mathbf{A} = \mathbf{Q}\,\mathbf{R} = \mathbf{Q}\begin{bmatrix}\widehat{\mathbf{R}}\\ 0\end{bmatrix}.$$

Wenn die $m \times n$ Matrix $\mathbf{A}$ Rang $n$ hat, dann hat die $m \times n$ Matrix $\mathbf{R}$ auch Rang $n$, so dass die ersten $n$ Zeilen eine $n \times n$ invertierbare obere Dreiecksmatrix $\widehat{\mathbf{R}}$ sind. $\mathbf{Q}$ ist eine $m \times m$ orthogonale (bzw. unitäre) Matrix: $\mathbf{Q}^\top\mathbf{Q} = \mathbf{I}$, bzw. $\mathbf{Q}^H\mathbf{Q} = \mathbf{I}$. Da orthogonale/unitäre Matrizen die Längen nicht verändern, haben wir:

$$\begin{aligned}\min_{\mathbf{x}\in\mathbb{R}^n} \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2 &= \min_{\mathbf{x}\in\mathbb{R}^n} \|\mathbf{Q}\mathbf{R}\mathbf{x}-\mathbf{Q}\mathbf{Q}^H\mathbf{b}\|_2^2 = \min_{\mathbf{x}\in\mathbb{R}^n} \|\mathbf{Q}(\mathbf{R}\mathbf{x}-\mathbf{Q}^H\mathbf{b})\|_2^2\\ &= \min_{\mathbf{x}\in\mathbb{R}^n} \|\mathbf{R}\mathbf{x}-\mathbf{Q}^H\mathbf{b}\|_2^2.\end{aligned}$$

Definieren wir nun einen neuen Vektor $\widehat{\mathbf{b}} = \mathbf{Q}^H\mathbf{b}$. Wir fassen die ersten $n$ Komponenten von $\widehat{\mathbf{b}}$ im Vektor $\mathbf{c}$ zusammen und können dann die Block-Matrix-Schreibweise verwenden:

$$\min_{\mathbf{x}\in\mathbb{R}^n} \left\| \begin{bmatrix}\widehat{\mathbf{R}}\\ 0\end{bmatrix}\begin{bmatrix}x_1\\ \vdots\\ x_n\end{bmatrix} - \begin{bmatrix}\widehat{\mathbf{b}}_1\\ \vdots\\ \widehat{\mathbf{b}}_n\\ \widehat{\mathbf{b}}_{n+1}\\ \vdots\\ \widehat{\mathbf{b}}_m\end{bmatrix}\right\|_2^2 = \min_{\mathbf{x}\in\mathbb{R}^n} \left\| \begin{bmatrix}\widehat{\mathbf{R}}\mathbf{x}\\ 0\end{bmatrix} - \begin{bmatrix}\mathbf{c}\\ \widehat{\mathbf{b}}_{n+1}\\ \vdots\\ \widehat{\mathbf{b}}_m\end{bmatrix}\right\|_2^2$$

$$= \min_{\mathbf{x}\in\mathbb{R}^n} \left\| \begin{bmatrix}\widehat{\mathbf{R}}\mathbf{x}-\mathbf{c}\\ \widehat{\mathbf{b}}_{n+1}\\ \vdots\\ \widehat{\mathbf{b}}_m\end{bmatrix}\right\|_2^2 = \min_{\mathbf{x}\in\mathbb{R}^n} (\|\widehat{\mathbf{R}}\mathbf{x}-\mathbf{c}\|_2^2 + |\widehat{b}_{n+1}|^2 + \ldots + |\widehat{b}_m|^2)$$

$$= 0 + |\widehat{b}_{n+1}|^2 + \ldots + |\widehat{b}_m|^2$$

falls $\widehat{\mathbf{R}}\mathbf{x} = \mathbf{c}$ ist. Wir sehen, dass in der letzten Gleichung die Terme ab $\widehat{b}_{n+1}$ unabhängig von $\mathbf{x}$ sind. Dieses $\mathbf{x}$ ist somit die Lösung im Sinne der kleinsten Quadrate, wobei der Fehler gleich der Norm des Residuums ist.
Schauen wir uns abschliessend ein kleines Zahlenbeispiel an.

Diese Methode hat gegenüber dem Weg über die Normalengleichung den Vorteil der Stabilität. Hat die Matrix $\mathbf{A}$ sehr viele Nullen, dann kann die QR-Zerlegung mit Givens-Rotationen eventuell auch effizient durchgeführt werden, so dass auch die Rechenzeit moderat sein kann. Das ist ein Vorteil gegenüber der Methode mit der Singulärwertzerlegung, die wir später kennenlernen werden. Jedoch macht diese Methode auch die sehr starke Voraussetzung (die in der Praxis nicht gegeben ist), dass die $m \times n$ Matrix $\mathbf{A}$ Rang $n$ hat. Falls die Matrix $\mathbf{A}$ nicht Rang $n$ hat, dann hat die letzte Gleichung unendlich viele Lösungen. Auch in diesem Fall müsste man eine Wahl für die freien Variablen treffen.

**Beispiel 5.3.0.1 (Lineare Ausgleichsrechnung - konkret)** [2]
Sei für $\mathbf{Ax} = \mathbf{b}$ die Gleichung

$$\begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 6 \\ 0 \\ 0 \end{bmatrix}$$

gegeben. Die Normalengleichung ist in diesem Fall

$$\begin{bmatrix} 3 & 3 \\ 3 & 5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 6 \\ 0 \end{bmatrix},$$

mit der Lösung

$$\begin{aligned} x_1 &= 5 \\ x_2 &= -3\,. \end{aligned}$$

Schauen wir uns alternativ die Berechnung mittels QR-Zerlegung an:

$$\mathbf{A} = \mathbf{QR} = \begin{bmatrix} -\frac{1}{\sqrt{3}} & \frac{1}{\sqrt{2}} & ? \\ -\frac{1}{\sqrt{3}} & 0 & ? \\ -\frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{2}} & ? \end{bmatrix} \begin{bmatrix} -\frac{3}{\sqrt{3}} & -\frac{3}{\sqrt{3}} \\ 0 & -\frac{2}{\sqrt{2}} \\ 0 & 0 \end{bmatrix}.$$

Durch Umformen mit $\widehat{\mathbf{R}}\mathbf{x} = \mathbf{Q}^{\mathsf{T}}\mathbf{b}$ erhalten wir

$$\begin{bmatrix} -\frac{3}{\sqrt{3}} & -\frac{3}{\sqrt{3}} \\ 0 & -\frac{2}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} -\frac{6}{\sqrt{3}} \\ \frac{6}{\sqrt{2}} \end{bmatrix}.$$

Mit dieser Methode erhalten wir schliesslich die Lösung

$$\begin{aligned} x_2 &= -3 \\ x_1 &= 5\,, \end{aligned}$$

welche mit den Lösungen der Methode der kleinsten Quadrate übereinstimmt.

⌟

[2] Dieses Beispiel wird in Abschnitt 9.5 mit Python gelöst.

# Kapitel 6

# Determinanten

## Inhalt


6.1 Definition und Eigenschaften . . . . . 164
6.2 Anwendungen . . . . . 170


## 6.1 Definition und Eigenschaften

**Definition 6.1.0.1. Multilineare Abbildung**

Seien $V$ und $W$ zwei lineare Räume.
$f : V \to W$ ist eine lineare Abbildung falls $f(\lambda x + \mu y) = \lambda f(x) + \mu f(y)$.
$g : V \times V \to W$ ist eine bilineare Abbildung, falls $g$ linear in jedem der beiden Argumente ist.
$h : V \times V \times \ldots \times V \to W$ ist eine multilineare Abbildung, falls $h$ linear in jedem Argument ist.

Wir werden gleich sehen, dass die Determinante eine multilineare Abbildung mit $V = \mathbb{R}^n$ und $W = \mathbb{R}$ ist.

$$\det : \underbrace{\mathbb{R}^n \times \mathbb{R}^n \times \ldots \times \mathbb{R}^n}_{n} \to \mathbb{R}\,.$$

Da wir genau $n$ mal $\mathbb{R}^n$ haben, können wir dies auch als eine Funktion quadratischer $n \times n$ Matrizen darstellen:

$$\det : \mathbb{R}^{n\times n} \to \mathbb{R}\,, \qquad \det : \mathbb{C}^{n\times n} \to \mathbb{C}\,.$$

Als wir die Berechnung der Inversen einer Matrix diskutierten, haben wir auch eine (zugegeben erstaunliche) Definition der Determinante 1.5 gesehen. Damit sollte betont werden, dass die praktische Berechnung der Determinante (für $n > 3$) über die Gauss-Elimination geht, und nicht über andere Formeln, die oft in Büchern zuerst vorkommen! Gleich geben wir eine alternative Definition, die relevant in den Anwendungen ist. Zuvor aber noch zwei Bemerkungen, die aus der ersten Definition sofort ersichtlich sind und die wir oft verwenden. Die erste ist eine Umschreibung der Bemerkung 1.5.0.4, die zweite ist die konkrete Formel für $2 \times 2$ Matrizen.

**Bemerkung 6.1.0.2 (Determinante erschien bereits in der Gauss-Elimination)**
**1:** $\det \mathbf{A} = 0 \iff \mathbf{A}$ singulär.

**2:** Die LU-Zerlegung einer $2 \times 2$ Matrix ist

$$\mathbf{A} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \mathbf{L} \cdot \begin{bmatrix} a & b \\ 0 & d - \frac{c}{a}b \end{bmatrix} \text{ für } a \neq 0\,.$$

Das Produkt der Elemente auf der Diagonale der oberen Dreiecksmatrix $\mathbf{R}$ aus der LU-Zerlegung/Gauss-Elimination ist die Determinante von $\mathbf{A}$

$$a(d - \frac{c}{a}b) = ad - cb =: \det \mathbf{A}\,.$$

Die Inverse $\mathbf{A}^{-1}$ der Matrix $\mathbf{A}$ lautet:

$$\mathbf{A}^{-1} = \frac{1}{ad - cb}\begin{bmatrix} d & -b \\ -c & a \end{bmatrix}.$$

Der Kehrwert der Determinante steht also als Faktor in der Inverse. Das haben wir bereits mehrmals gesehen.

⌟

**Definition 6.1.0.3. Determinante**

Die Determinante ist eine Funktion

$$\det : \underbrace{\mathbb{R}^n \times \mathbb{R}^n \times \ldots \times \mathbb{R}^n}_{n} \to \mathbb{R}\,.$$

mit folgenden Eigenschaften:

(D1) $\det \mathbf{I}_n = 1$,

(D2) det wechselt das Vorzeichen, wenn zwei Zeilen oder Spalten vertauscht werden (*Antisymmetrie*).

(D3) det ist linear in jeder Zeile und Spalte:

$$\det \begin{bmatrix} ta & tb \\ c & d \end{bmatrix} = t \cdot \det \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$
$$\det \begin{bmatrix} a + \tilde{a} & b + \tilde{b} \\ c & d \end{bmatrix} = \det \begin{bmatrix} a & b \\ c & d \end{bmatrix} + \det \begin{bmatrix} \tilde{a} & \tilde{b} \\ c & d \end{bmatrix}.$$

**Bemerkung 6.1.0.4 (Die ersten drei Eigenschaften legen die Determinante fest)**
Man kann beweisen, dass die drei Eigenschaften (D1), (D2) und (D3) die Determinante exakt festlegen. Es gibt also keine andere Funktion, welche diese Eigenschaften erfüllt und nicht die Determinante ist.

⌟

**Bemerkung 6.1.0.5 (Die Determinante ist eine multilineare, antisymmetrische Form)** Die Determinante ist eine multilineare antisymmetrische Form. ⌟

**Bemerkung 6.1.0.6 (Weitere wichtige Eigenschaften der Determinante)**
Wir sehen nun weitere sehr nützliche Eigenschaften, die direkt aus (D1), (D2) und (D3) folgen und die zu Rechenregeln führen.

(4) Falls zwei Spalten oder Zeilen einer Matrix identisch sind, so ist die Determinante der Matrix 0.

Das kommt direkt aus der Antisymmetrie: Ein Tausch der zwei gleichen Zeilen muss einen Vorzeichenwechsel der Determinante mit sich bringen, aber die Matrix ist dieselbe. Das geht nur, wenn die Determinante 0 ist.

(5) Linearkombinationen von Zeilen der Matrix ändern die Determinante dieser Matrix nicht.

$$\det \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \det \begin{bmatrix} a & b \\ c - ta & d - tb \end{bmatrix}.$$

Diese Eigenschaft folgt direkt aus (D3) und (4):

$$\det\begin{bmatrix} a & b \\ c-ta & d-tb \end{bmatrix} \overset{(D3)}{=} \det\begin{bmatrix} a & b \\ c & d \end{bmatrix} - t\cdot \underbrace{\det\begin{bmatrix} a & b \\ a & b \end{bmatrix}}_{\overset{(4)}{=}0}.$$

Somit gilt für die Gauss-Elimination, welche nur Linearkombinationen der Zeilen der Matrix macht, dass sie die Determinante nicht ändert. Also haben wir:

$$\mathbf{A} = \mathbf{L}\,\mathbf{R} \implies \det\mathbf{A} = \det\mathbf{R}.$$

Falls wir auch noch Permutationen haben, so gibt (D2) noch einen eventuellen Vorzeichenwechsel, der für alle nötigen Vertauschungen steht:

$$\mathbf{P}\,\mathbf{A} = \mathbf{L}\,\mathbf{R} \implies \det\mathbf{A} = \det\mathbf{P}\cdot\det\mathbf{R}.$$

Dabei wurde verwendet, dass $\det\mathbf{P} \in \{-1, 1\}$ und somit $\det\mathbf{P} = \frac{1}{\det\mathbf{P}}$ gilt.

(6) Wenn eine Matrix eine Nullzeile oder eine Nullspalte hat, so ist ihre Determinante $0$.

Das ist klar, denn für eine solche Matrix kann man den Vorfaktor $t = 0$ in (D3) setzen.

(7) Falls $\mathbf{A}$ eine Dreiecksmatrix ist, so ist die Determinante von $\mathbf{A}$ das Produkt der Diagonaleinträge:

$$\det\begin{bmatrix} a_{11} & * & \cdots & * \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & * \\ 0 & \cdots & 0 & a_{nn} \end{bmatrix} \overset{(D3)}{=} \det\begin{bmatrix} a_{11} & 0 & \cdots & 0 \\ 0 & a_{22} & * & * \\ \vdots & \ddots & \ddots & * \\ 0 & \cdots & 0 & a_{nn} \end{bmatrix} + \det\begin{bmatrix} 0 & * & \cdots & * \\ 0 & a_{22} & \ddots & \vdots \\ \vdots & \ddots & \ddots & * \\ 0 & \cdots & 0 & a_{nn} \end{bmatrix}$$

$$\overset{(6)}{=} \det\begin{bmatrix} a_{11} & 0 & \cdots & 0 \\ 0 & a_{22} & * & * \\ \vdots & \ddots & \ddots & * \\ 0 & \cdots & 0 & a_{nn} \end{bmatrix}$$

$$= \ldots$$

$$= \det\begin{bmatrix} a_{11} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & a_{nn} \end{bmatrix}$$

$$= a_{11}a_{22}\ldots a_{nn}.$$

Die Diagonalmatrix ergibt sich, wenn die Eigenschaften (D3) und (6), wie für die erste Zeile demonstriert, auf alle $n$ Zeilen angewandt werden.

Bei der letzten Gleichheit wurde jedes $a_{jj}$ als $t$ in (D3) verwendet und dann (D1) ausnutzt.

(8) Falls $\mathbf{A}$ singulär ist, entsteht bei der Gauss-Elimination eine Nullzeile und die Determinante ist, wie beim Punkt (6) bemerkt, $0$.

(9) $\det\mathbf{A}\,\mathbf{B} = \det\mathbf{A}\cdot\det\mathbf{B}$.

Der Beweis dafür ist weniger elegant, man muss ja rechnen!

(10) $\det \mathbf{A}^{-1} = \dfrac{1}{\det \mathbf{A}}$.

Das kommt aus $1 = \det \mathbf{I} = \det(\mathbf{A}\mathbf{A}^{-1}) = \det \mathbf{A} \cdot \det(\mathbf{A}^{-1})$.

(11) $\det \mathbf{A}^\top = \det \mathbf{A}$.

Das kommt aus $\mathbf{A}^\top = (\mathbf{L}\mathbf{R})^\top = \mathbf{R}^\top \mathbf{L}^\top$ und den vorherigen Rechenregeln.

(12) $\det \mathbf{A} = a_{11} \cdot \det \mathbf{A}_{11} - a_{12} \cdot \det \mathbf{A}_{12} + \ldots + (-1)^{n-1} a_{1n} \cdot \det \mathbf{A}_{1n}$.
Diese Formel nennt man die Entwicklung nach der ersten Zeile und kommt in den Büchern als Definition der Determinante vor. Die Matrix $\mathbf{A}_{1j}$, $j = 1, \ldots, n$, ist die $(n-1) \times (n-1)$-Matrix, die aus der $n \times n$-Matrix entsteht, wenn man die 1-ste Zeile und die $j$-te Spalte wegnimmt. Unter der Verwendung der Eigenschaft (D2) kann man diese Formel für jede Zeile/Spalte umschreiben.

⌟

**Bemerkung 6.1.0.7 (Berechnung der Determinante)**
Für eine grosse Matrix berechnet man die Determinante numerisch mit der Gauss-Elimination.
Für $n = 2$ verwenden wir die Formel aus der Bemerkung 6.1.0.2:

$$\det \begin{bmatrix} a & b \\ c & d \end{bmatrix} = ad - bc\,.$$

Für $n = 3$ kann man bereits die Gauss-Elimination verwenden. Dabei muss man auf die Divisionen aufpassen. Diese kann man vermeiden, wenn man die Formel (12) dafür verwendet.
(12) kann sehr praktisch sein, wenn wir von Hand die Determinante einer Matrix berechnen wollen, die viel Struktur, z. B. Nullen, hat, die wir ausnutzen können. ⌟

**Beispiel 6.1.0.8 (Berechnung der Determinante - Rechenführung)** [1] Wir wollen nun diese Techniken anhand der Berechnung von zwei Determinanten illustrieren. Auf dem Computer verwendet man dazu die Gauss-Elimination. Von Hand muss man die Regeln und eine saubere Rechenführung beherrschen. Die Matrizen, deren Determinanten wir berechnen wollen, sind:

$$\mathbf{A} = \begin{bmatrix} 2 & 1 & 3 \\ 3 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 4 & 3 & -1 & -2 \\ 2 & 1 & -1 & 0 \\ -2 & -2 & 0 & 2 \\ 5 & 1 & 0 & 2 \end{bmatrix}.$$

Starten wir mit der Berechnung von $\det \mathbf{A}$ via Gauss-Elimination. Damit wir unbequeme Brüche in der Handberechnung vermeiden, vertauschen wir die erste und die letzte Zeile, danach sind die Eliminationsschritte einfach:

$$\det \begin{bmatrix} 2 & 1 & 3 \\ 3 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix} = -\det \begin{bmatrix} 1 & 1 & 2 \\ 3 & 2 & 1 \\ 2 & 1 & 3 \end{bmatrix} = -\det \begin{bmatrix} 1 & 1 & 2 \\ 0 & -1 & -5 \\ 0 & -1 & -1 \end{bmatrix} = -\det \begin{bmatrix} 1 & 1 & 2 \\ 0 & -1 & -5 \\ 0 & 0 & 4 \end{bmatrix} = 4\,.$$

Die Formel (12) würde gemeinsam mit der Formel für $2 \times 2$-Matrizen direkt das Ergebnis liefern:

$$\det \mathbf{A} = 2 \det \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} - 1 \det \begin{bmatrix} 3 & 1 \\ 1 & 2 \end{bmatrix} + 3 \det \begin{bmatrix} 3 & 2 \\ 1 & 1 \end{bmatrix} = 2(4-1) - 1(6-1) + 3(3-2) = 4\,.$$

Noch einfacher ist es, wenn wir (D3) und (12) kombinieren. Wir bilden lineare Kombinationen von Spalten oder von Zeilen, sodass wir Nullen in einer Zeile oder Spalte erzeugen. Dabei wählen wir die Spalten so,

[1] Dieses Beispiel wird im Abschnitt 9.5.1 mit Python gelöst.

dass keine komplizierten Rechnungen auftreten. Z. B. hat die untere Zeile von $\mathbf{A}$ zweimal die 1 und eine 2. Dann:

$$\det \mathbf{A} = \det \begin{bmatrix} 2 & -1 & -1 \\ 3 & -1 & -5 \\ 1 & 0 & 0 \end{bmatrix} = -\det \begin{bmatrix} 1 & 0 & 0 \\ 3 & -1 & -5 \\ 2 & -1 & -1 \end{bmatrix} = (-1)\det \begin{bmatrix} -1 & -5 \\ -1 & -1 \end{bmatrix} = (-1)(1-5) = 4\,.$$

Für die Matrix $\mathbf{B}$ beginnen wir mit der Strategie, die die Formel (12) verwendet:

$$\det \mathbf{B} = -\det \begin{bmatrix} -1 & 3 & 4 & -2 \\ -1 & 1 & 2 & 0 \\ 0 & -2 & -2 & 2 \\ 0 & 1 & 5 & 2 \end{bmatrix} = \det \begin{bmatrix} 1 & 2 & 0 \\ -2 & -2 & 2 \\ 1 & 5 & 2 \end{bmatrix} - \det \begin{bmatrix} 3 & 4 & -2 \\ -2 & -2 & 2 \\ 1 & 5 & 2 \end{bmatrix}.$$

Dann kann genau wie bei $\mathbf{A}$ weitergerechnet werden. Man sieht aber, dass die Strategie mit linearen Kombinationen hier die Rechnung einfacher macht. Darum machen wir von Anfang an Nullen in der dritten Zeile:

$$\det \mathbf{B} = \begin{bmatrix} 2 & 1 & -1 & -2 \\ 2 & 1 & -1 & 0 \\ 0 & 0 & 0 & 2 \\ 7 & 3 & 0 & 2 \end{bmatrix} = -\det \begin{bmatrix} 0 & 0 & 0 & 2 \\ 2 & 1 & -1 & 0 \\ 2 & 1 & -1 & -2 \\ 7 & 3 & 0 & 2 \end{bmatrix} = \det \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 1 & -1 & 2 \\ -2 & 1 & -1 & 2 \\ 2 & 3 & 0 & 7 \end{bmatrix} = 2\det \begin{bmatrix} 1 & -1 & 2 \\ 1 & -1 & 2 \\ 3 & 0 & 7 \end{bmatrix} = 0\,,$$

da die ersten zwei Zeilen gleich sind und die Regel (4) gleich den Wert 0 liefert. Man hätte dies auch etwas früher merken können:

$$\det \mathbf{B} = \begin{bmatrix} 2 & 1 & -1 & -2 \\ 2 & 1 & -1 & 0 \\ 0 & 0 & 0 & 2 \\ 7 & 3 & 0 & 2 \end{bmatrix} = \det \begin{bmatrix} 2 & 1 & -1 & -2 \\ 2 & 1 & -1 & 0 \\ 2 & 1 & -1 & 0 \\ 7 & 3 & 0 & 2 \end{bmatrix} = 0\,,$$

da die zweite und dritte Zeile identisch sind.
Die Matrix $\mathbf{B}$ hat Determinante 0, sie ist also singulär: Ihre Spalten (und ihre Zeilen) sind linear abhängige Vektoren. ⌟

**Beispiel 6.1.0.9 (Berechnung einer Determinante)** Wir wollen nun die Determinante von

$$\mathbf{A} = \begin{bmatrix} 2-a & 2 & 2 \\ 2 & 2-a & 2 \\ 2 & 2 & 2-a \end{bmatrix}$$

berechnen:

$$\det \mathbf{A} = \begin{bmatrix} -a & 0 & a \\ 0 & -a & a \\ 2 & 2 & 2-a \end{bmatrix} = \det \begin{bmatrix} -a & 0 & 0 \\ 0 & -a & a \\ 2 & 2 & 4-a \end{bmatrix} = \det \begin{bmatrix} -a & 0 & 0 \\ 0 & -a & 0 \\ 2 & 2 & 6-a \end{bmatrix} = a^2(6-a)\,.$$

⌟

**Beispiel 6.1.0.10 (Finite Differenz für die zweite Ableitung.)**
Sei die $n \times n$-Matrix[2]:

$$\mathbf{D}_n = \begin{bmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{bmatrix}.$$

[2]Beispiel 6 vom Kapitel 5.2, Seite 261 in Str09.

Wir entwickeln nach der ersten Zeile (Formel (12)) und erhalten

$$\begin{aligned}\det \mathbf{D}_n &= 2\det \mathbf{D}_{n-1} - (-1)\det \begin{bmatrix} -1 & -1 & & & & \\ & 2 & -1 & & & \\ & -1 & 2 & \ddots & & \\ & & \ddots & \ddots & -1 & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{bmatrix} \\ &= 2\det \mathbf{D}_{n-1} - (-1)(-1)\det \begin{bmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & \ddots & & \\ & & \ddots & \ddots & -1 & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{bmatrix} \\ &= 2\det \mathbf{D}_{n-1} - \det \mathbf{D}_{n-2}.\end{aligned}$$

Da wir $\det \mathbf{D}_2 = 3$ und $\det \mathbf{D}_3 = 4$, sehen wir, dass die obere Rekursion zu $\det \mathbf{D}_n = n+1$ führt, was wir schnell mit mathematischer Induktion beweisen können. ⌟

**Beispiel 6.1.0.11 (Veränderte Ableitung)** Sei die $n \times n$-Matrix[3]:

$$\mathbf{B}_n = \begin{bmatrix} 1 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & \ddots & & \\ & & \ddots & \ddots & -1 & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{bmatrix}.$$

Wir entwickeln nach der ersten Zeile (Formel (12)) und erhalten
$\det \mathbf{B}_n = \det \mathbf{D}_{n-1} - \det \mathbf{D}_{n-2} = n - (n-1) = 1$. ⌟

**Bemerkung 6.1.0.12 (Eine sehr lange Formel für die Determinante)**
Die Idee vom Beweis von Formel (12) kann man so erweitern, dass man die Determinante einer $n \times n$-Matrix als eine sehr lange ($n! = 1 \cdot 2 \cdot 3 \cdots n$) Summe von Determinanten von Permutationsmatrizen mal einfache Produkte der entsprechenden $n$ Einträge der Matrix:

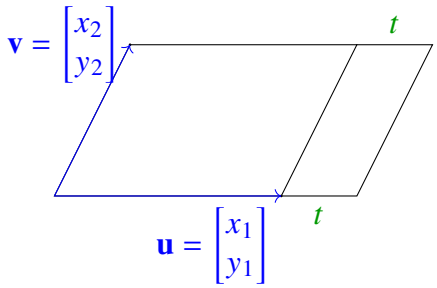
$$\det \mathbf{A} = \sum_{\substack{\text{alle } n! \text{ Permutationen} \\ \mathbf{P}=[\mathbf{e}_{\sigma(1)}\mathbf{e}_{\sigma(2)}\ldots\mathbf{e}_{\sigma(n)}]}} (\det \mathbf{P}) a_{1\sigma(1)} a_{2\sigma(2)} \cdots a_{n\sigma(n)} .$$

Diese Formel wird gelegentlich für theoretische Zwecke gebraucht, jedoch nicht für praktische Berechnungen, denn die Summe enthält $n!$ Terme, ist also viel zu teuer! Vergleichen Sie das Wachstum von $n!$ gegenüber $n^3$, was die Anzahl der Operationen in der Gauss-Elimination ist! ⌟

[3] Beispiel 7 vom Kapitel 5.2, Seite 261 in Str09.

## 6.2 Anwendungen

### Flächeninhalt und Volumen



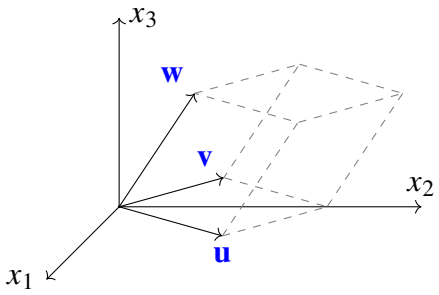

In zwei Dimensionen betrachten wir ein Parallelogramm mit einer Ecke im Ursprung der Koordinatenachsen, dessen Seiten durch die Richtungsvektoren $\mathbf{u} = \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$ und $\mathbf{v} = \begin{bmatrix} x_2 \\ y_2 \end{bmatrix}$ gegeben sind. Dieses Parallelogramm kann auch noch durch eine Skalierung mit $t$ in der Richtung einer diesen Vektoren vergrössert oder verkleinert werden. Diese beiden Vektoren $\mathbf{u}$ und $\mathbf{v}$ können wir als Spalten oder als Zeilen in einer Matrix $\mathbf{M}$ zusammenfassen:

$$\mathbf{M} = \begin{bmatrix} \text{---} & \mathbf{u}^\mathsf{T} & \text{---} \\ \text{---} & \mathbf{v}^\mathsf{T} & \text{---} \end{bmatrix}.$$

Für das Parallelogramm kann man noch eine Orientierung definieren: Positiv, wenn der erste Vektor über den zweiten Vektor durch eine Rotation (in die trigonometrische Richtung) mit einem Winkel in $[0, \pi]$ gebracht werden kann, und negativ sonst. Die so orientierte Fläche des Parallelogramms erfüllt offensichtlich (D1), (D2) und (D3). Da es aber nur eine einzige Funktion mit diesen Eigenschaften gibt, muss der Wert der orientierten Fläche gleich mit dem Wert der Determinanten sein. Für die Berechnung der Fläche selbst gilt also

$$\text{Fläche } A = |\det \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \end{bmatrix}|.$$

Dies lässt sich auch für den Volumeninhalt eines Parallelotops in beliebigen Dimensionen erweitern. Dort hat man es statt mit einer $2 \times 2$-Matrix, mit einer $3 \times 3$-, bzw. einer $n \times n$-Matrix zu tun. Im Fall $n = 3$ notieren wir die Zeilenvektoren $\mathbf{u}$, $\mathbf{v}$ und $\mathbf{w}$:




$$\text{Volumen } V = |\det \begin{bmatrix} u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 \\ w_1 & w_2 & w_3 \end{bmatrix}|.$$

Die Erweiterung für $n$ Vektoren (die Seiten des Parallelotops) in $n$ Dimensionen erfolgt analog.

Die Determinante und somit das Volumen des von den $n$ Vektoren aufgespannten geometrischen Objekts ist genau dann null, wenn die $n$ Vektoren linear abhängig sind, d. h. sobald sich einer von ihnen als Linearkombination der Anderen schreiben lässt, d. h. sobald einer im von den Anderen aufgespannten Unterraum

liegt.

Da die Determinante einer Matrix und die ihrer Transponierten gleich sind, kann man die Vektoren alternativ auch spaltenweise statt zeilenweise in die Matrix einfügen.

## Beziehung zum Kreuzprodukt

Die von zwei Vektoren aufgespannte Fläche und diese Orientierung haben Sie in der Schule als Kreuzprodukt der zwei Vektoren in drei Dimensionen kennengelernt. Seien $\mathbf{u}$, $\mathbf{v}$ zwei Vektoren in $\mathbb{R}^3$. Mit diesen Vektoren kann man das Kreuzprodukt bilden: Das Ergebnis ist einen Vektor mit der oben definierten Orientierung. Der Betrag des Ergebnisses ist die Fläche des aufgespannten Parallelogramms. Da der Betrag des Kreuzprodukts die drei Eigenschaften der Determinante erfüllt, muss es einer Determinante entsprechen. Dazu gab es noch eine Rechenformel. Diese kann man auch formal als die Rechenregel (12) sehen:

$$\mathbf{u} \times \mathbf{v} = \det \begin{bmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \mathbf{e}_3 \\ u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 \end{bmatrix} = (u_2v_3 - v_2u_3)\mathbf{e}_1 - (u_1v_3 - v_1u_3)\mathbf{e}_2 + (u_1v_2 - v_1u_2)\mathbf{e}_3 \,.$$

Dies ist aber nur eine formale Schreibweise, da die Einträge in der ersten Zeile die Elemente der Standardbasis in $\mathbb{R}^3$ sind. Allerdings lässt sich eine solche Formel für $n-1$ Vektoren in $n$ Dimensionen formal verallgemeinern.

Diese Schreibweise passt zur Definition des Volumens einer Schachtel mit Vektoren $\mathbf{u}$, $\mathbf{v}$ und $\mathbf{w}$, in der Schule vermutlich definiert als $|(\mathbf{u} \times \mathbf{v})\mathbf{w}|$:

$$|(\mathbf{u} \times \mathbf{v})\mathbf{w}| = |\det \begin{bmatrix} u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 \\ w_1 & w_2 & w_3 \end{bmatrix}| \,.$$

In Analysis II und in der Physik muss man oft Determinanten ausrechnen, z. B. bei einem Variablenwechsel in einem mehrdimensionalen Integral. Im Kapitel über die Eigenwerte benutzten wir die Determinante auch. In der Praxis gibt es auch weitere Anwendungen der Determinante. Beispielsweise trifft man in der Quantenmechanik auf das Pauli-Prinzip: Der Zustand von Elektronen muss durch eine Funktion darstellbar sein, die die Eigenschaften (D1), (D2) und (D3) erfüllt, also muss dies eine Determinante sein. Im Fachjargon: In der Ein-Elektronen-Näherung der Quantenmechanik hat das Pauli-Prinzip zur Konsequenz, dass sich die Gesamtwellenfunktion für Fermionen als Slater-Determinante der Ein-Elektronen-Funktionen (Orbitale) darstellen lässt.

# Kapitel 7

# Eigenwerte

## Inhalt


**7.1 Motivation . . . . . 172**
**7.2 Eigenwerte und Eigenvektoren . . . . . 179**
**7.3 Symmetrische Matrizen . . . . . 192**
**7.4 Symmetrisch positiv definite Matrizen . . . . . 198**
7.4.1 Anwendungen . . . . . 200
**7.5 Die Fourier Matrix . . . . . 205**
**7.6 Nicht-diagonalisierbare Matrizen . . . . . 213**
7.6.1 Schur-Zerlegung . . . . . 214
7.6.2 Jordan-Form . . . . . 218


## 7.1 Motivation

Wenn eine lineare Abbildung $\mathscr{A} : V \to V$ auf einen Vektor $x \neq 0$ angewendet wird, erhält man einen neuen Vektor $y$. Wenn das Ergebnis $y$ mit dem ursprünglichen Vektor $x$ kollinear ist, d. h. $y = \lambda x$ mit einem Skalar $\lambda$, dann haben wir einen besonderen Fall. Wenn man (für eine gegebene lineare Abbildung $\mathscr{A}$) nach solchen speziellen Vektoren $x$ sucht, handelt es sich um das Standard-Eigenwertproblem. Den Skalierungsfaktor $\lambda$ bezeichnen wir als Eigenwert der Abbildung.

In diesem Kapitel werden wir ausschliesslich quadratische Matrizen in der Rolle der linearen Abbildung $\mathscr{A} : V \to V$ sehen.

**Beispiel 7.1.0.1 (Einige lineare Abbildungen in 2 Dimensionen)**
Wir wählen als Basis jeweils die Standardbasis in $\mathbb{R}^2$ oder $\mathbb{C}^2$ und stellen eine lineare Abbildung $\mathscr{A} : \mathbb{R}^2 \to \mathbb{R}^2$ durch eine Matrix $\mathbf{A}$ in diesen Basen dar. Die folgenden Diagramme zeigen die Bilder einiger geometrischer Objekte[1] durch die lineare Abbildung, die durch die jeweils angegebene Matrix realisiert wird. Bestimmte Geraden aus $\mathbb{R}^2$ könnten für die entsprechenden Abbildungen invariant sein. Die Vektoren, die diese Geraden erzeugen werden jeweils mit Pfeilen in entsprechenden Farben angezeigt.

[1] Die Katze links wurde mithilfe von Gemini 2.5 pro als eine Menge von Vektoren in $\mathbb{R}^2$ beschrieben.

Abbildung mit Matrix: $\begin{bmatrix} -1 & 0 \\ 0 & 2 \end{bmatrix}$

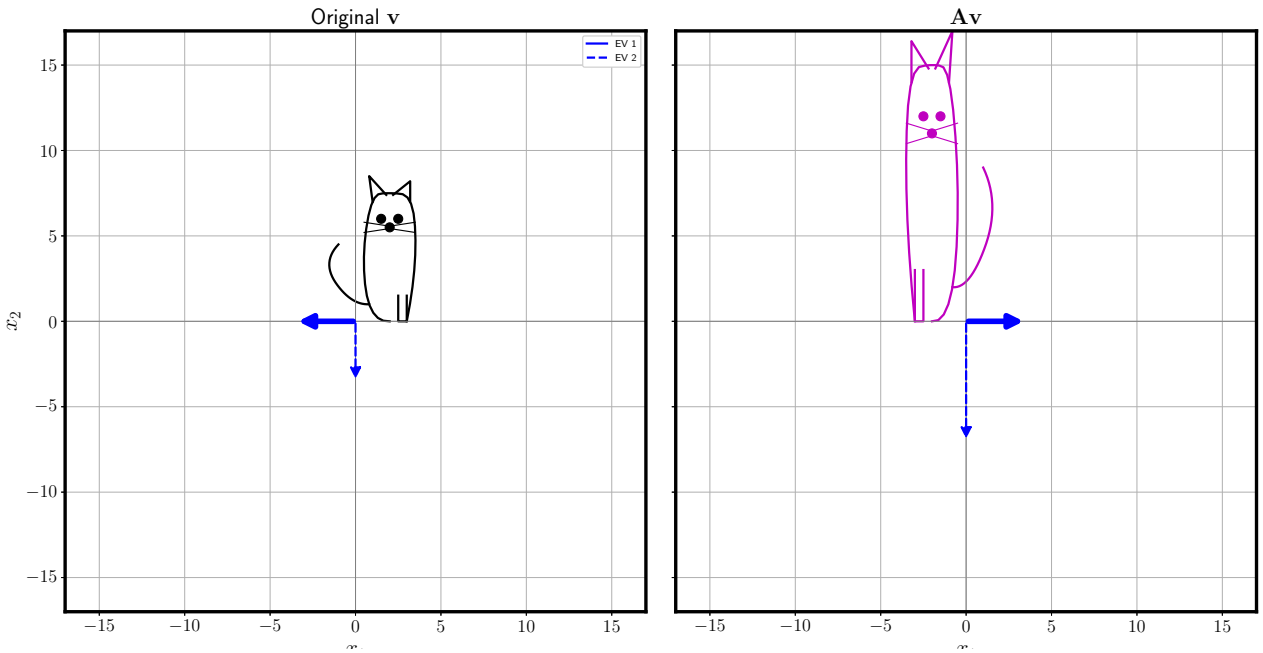



Abbildung mit Matrix: $\begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix}$

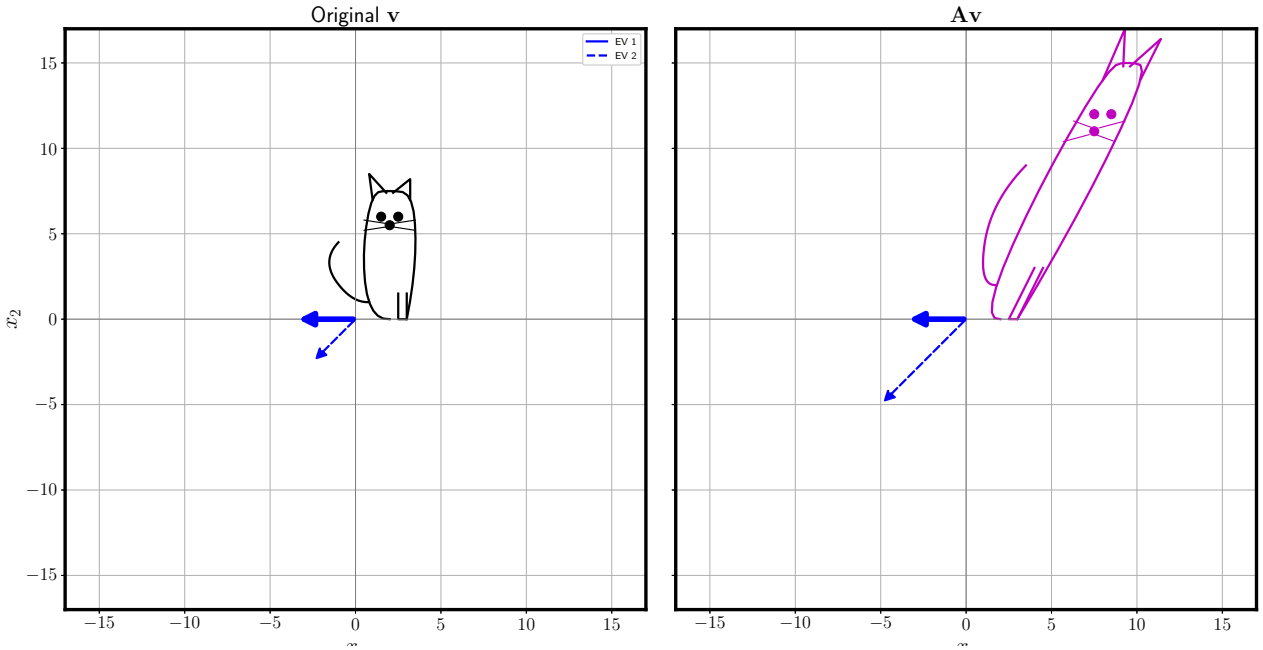


Abbildung mit Matrix: $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$

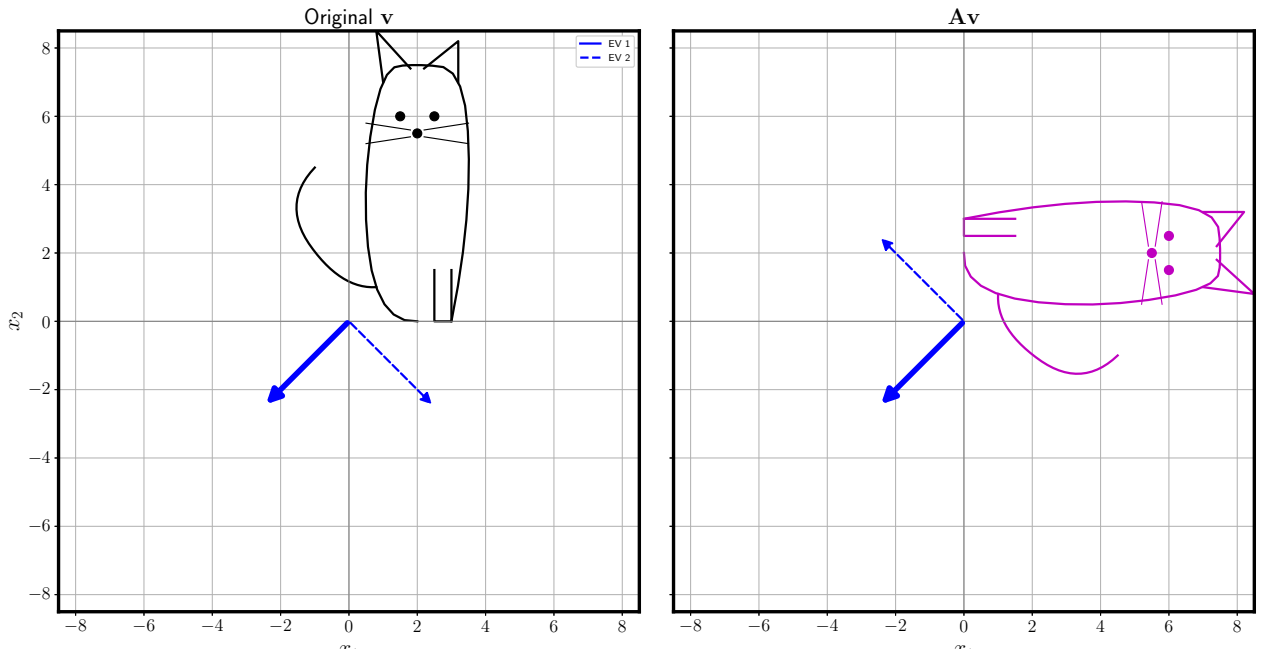
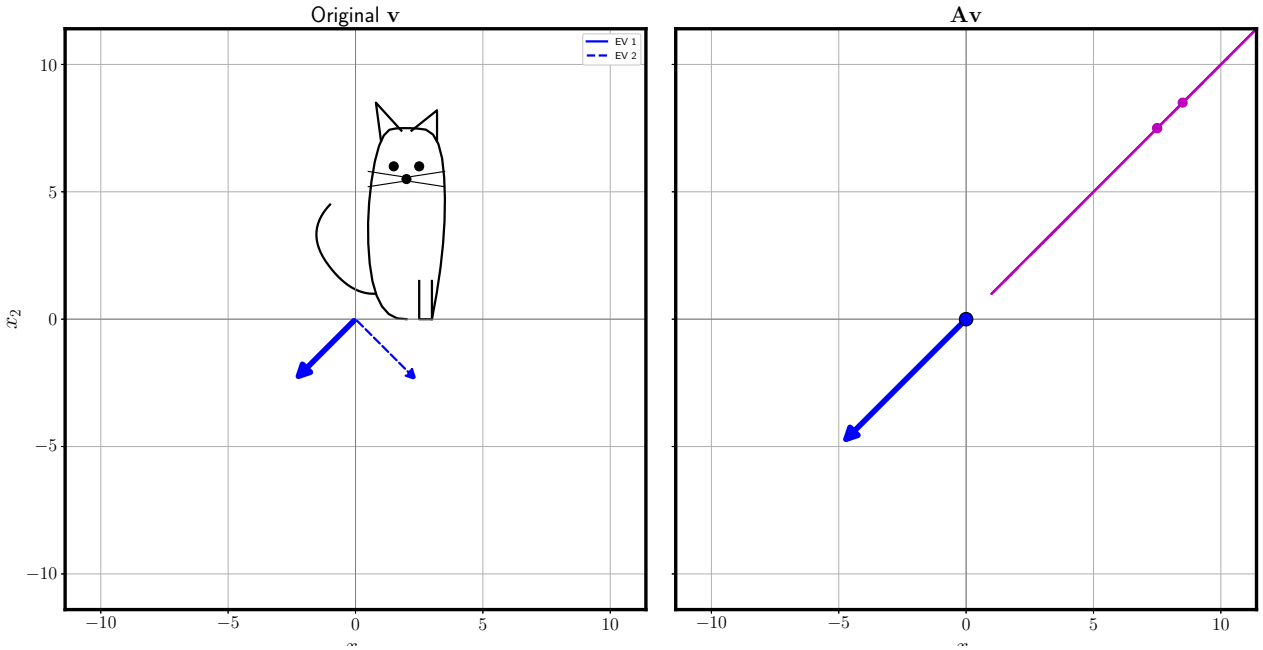
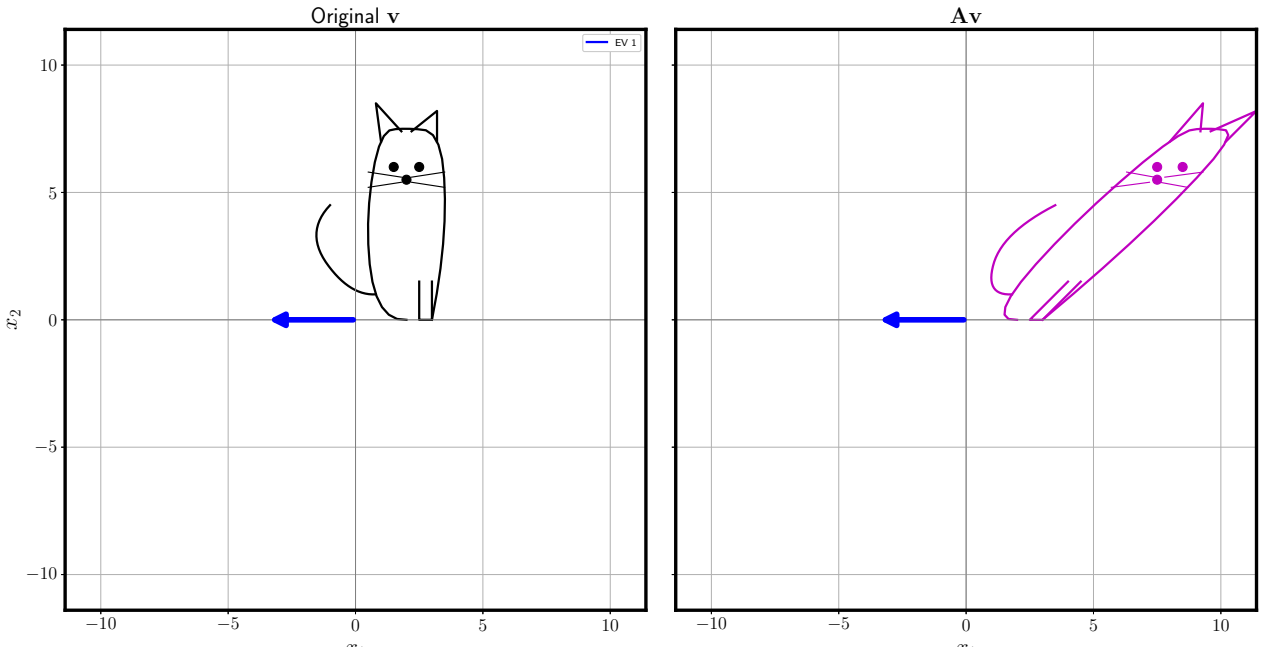



Abbildung mit Matrix: $\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$



Manchmal gibt es nur eine invariante Gerade.

Abbildung mit Matrix: $\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$




Manchmal gibt es zwar zwei invariante Geraden, die sich aber in $\mathbb{C}^2 \setminus \mathbb{R}^2$ befinden, d. h. die Abbildung **muss** als $\mathscr{A} : \mathbb{C}^2 \to \mathbb{C}^2$ gesehen werden; in diesem Fall zeichnen wir Vektoren $\mathbf{u}, \mathbf{v}$, die aus dem Realteil und aus dem Imaginärteil jeder komplexwertigen Richtung entstehen; die Ebene, die von $\mathbf{u}, \mathbf{v}$ in $\mathbb{R}^4$ aufgespannt ist, ist invariant, was wir nicht darstellen können; stattdessen bilden wir die Ellipse $\mathbf{u}\cos(t) + \mathbf{v}\sin(t)$, für $t \in [0, 2\pi]$ ab; die Orientierung, d. h. die Richtungen der Achsen dieser Ellipsen erscheinen invariant.

Abbildung mit Matrix: $\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}$

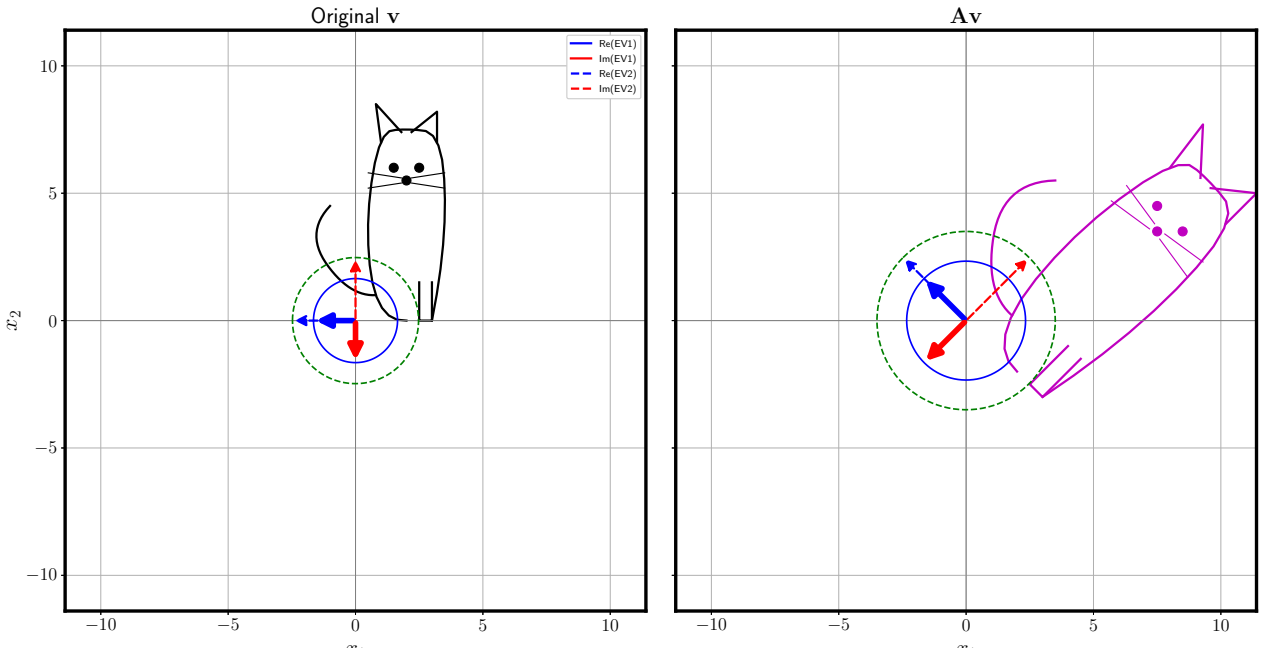


Abbildung mit Matrix: $\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$

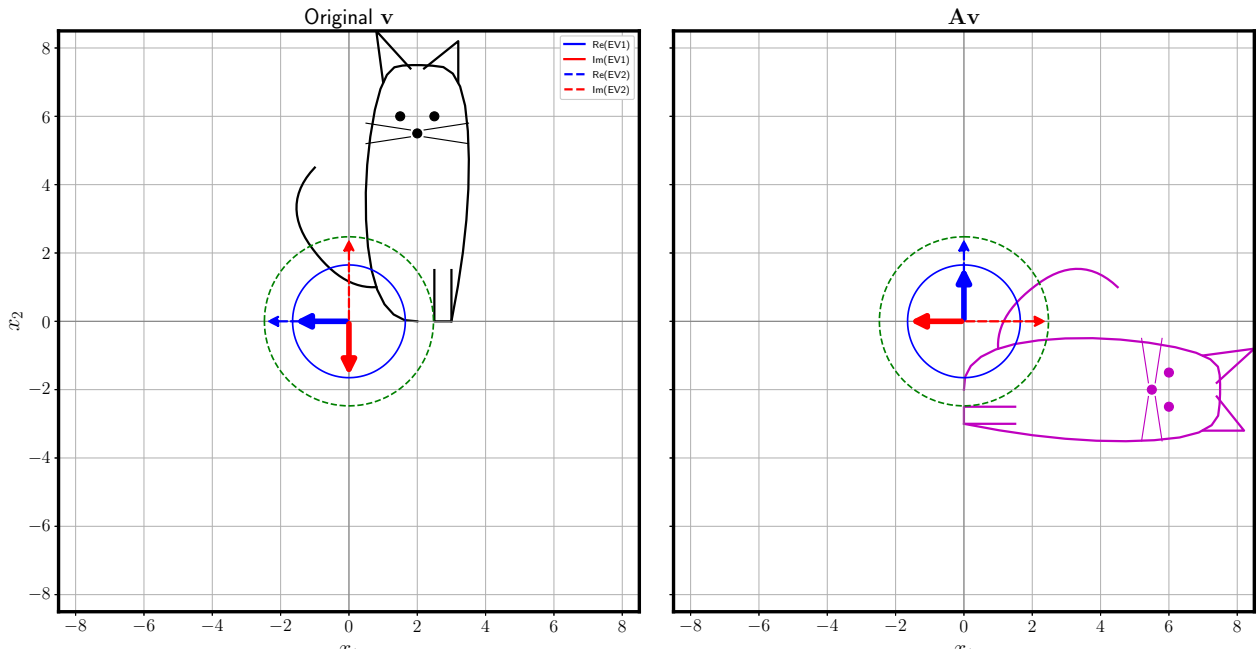
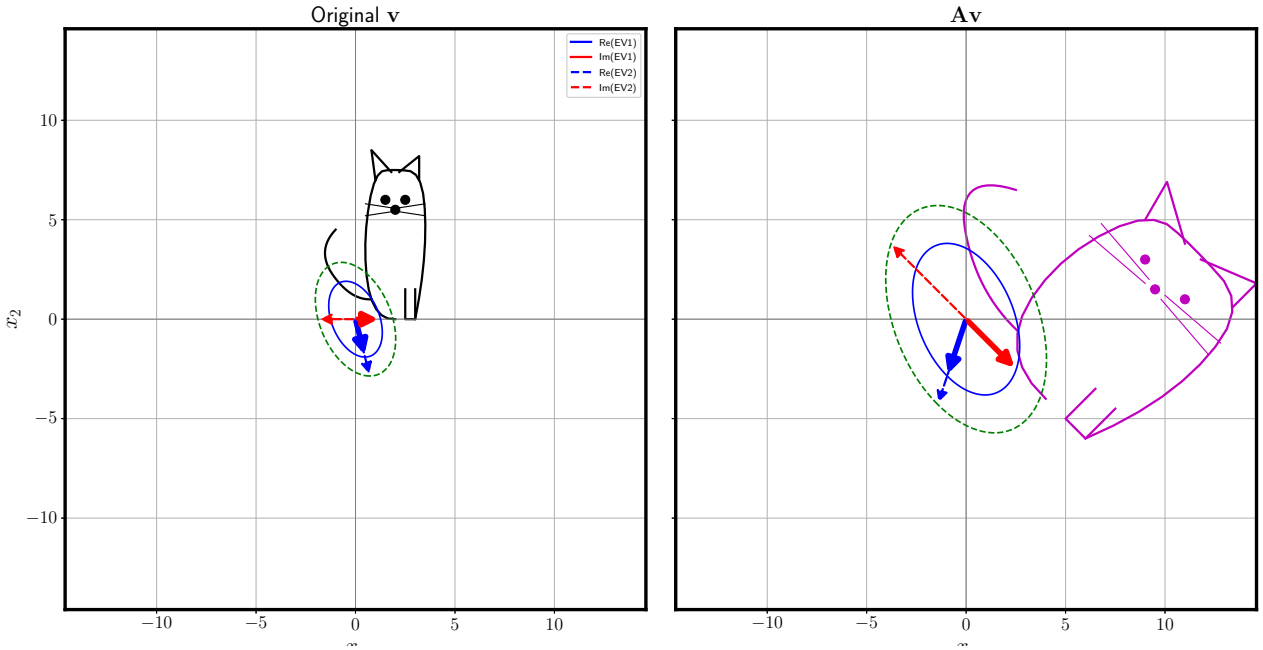



Abbildung mit Matrix: $\begin{bmatrix} 2 & 1 \\ -2 & 1 \end{bmatrix}$




**Beispiel 7.1.0.2 (Potenz einer Matrix)** Wie können wir direkt die Potenz $\mathbf{A}^k$ einer Matrix $\mathbf{A}$ berechnen, oder sie für grosses $k$ abschätzen?
Für eine Diagonalmatrix $\mathbf{D} = \operatorname{diag}\{\lambda_1, \ldots, \lambda_n\}$ ist das einfach $\mathbf{D}^k = \operatorname{diag}\{\lambda_1^k, \ldots, \lambda_n^k\}$, aber für eine allgemeine Matrix geht das nicht mehr so ‘‘punktweise’’. Wenn es eine invertierbare Matrix $\mathbf{S}$ gibt, so dass $\mathbf{A} = \mathbf{S}\mathbf{D}\mathbf{S}^{-1}$, dann ist das aber auch einfach:

$$\mathbf{A}^k = \left(\mathbf{S}\mathbf{D}\mathbf{S}^{-1}\right)^k = \mathbf{S}\mathbf{D}^k\mathbf{S}^{-1}\,.$$

Nun betrachten wir zwei Anwendungen, die in Physik I und Analysis I vorkommen.

**Beispiel 7.1.0.3 (System entkoppelter linearer Differenzialgleichungen 1. Ordnung)**

$$\begin{aligned}\dot{u}_1(t) &= a_1 u_1(t) \Rightarrow u_1(t) = e^{a_1 t} u_1(0)\\ \dot{u}_2(t) &= a_2 u_2(t) \Rightarrow u_2(t) = e^{a_2 t} u_2(0)\,.\\ &\vdots\end{aligned}$$

Wir schreiben die Lösungen zusammen als Vektor:

$$\mathbf{u}(t) = \begin{bmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_n(t) \end{bmatrix} \in \mathbb{R}^n\,.$$

Mit der *Diagonalmatrix* $\mathbf{D}$ lässt sich das oben genannte System linearer Differenzialgleichungen in der Form

$$\dot{\mathbf{u}}(t) = \mathbf{D}\mathbf{u}(t), \qquad \mathbf{D} = \begin{bmatrix} a_1 & & 0 \\ & a_2 & \\ & \ddots & \\ 0 & & a_n \end{bmatrix}$$

schreiben. Die Lösung des Systems lautet demnach:

$$\mathbf{u}(t) = \begin{bmatrix} e^{a_1 t} u_1(0) \\ \vdots \\ e^{a_n t} u_n(0) \end{bmatrix} = \begin{bmatrix} e^{a_1 t} & & 0 \\ & \ddots & \\ 0 & & e^{a_n t} \end{bmatrix} \begin{bmatrix} u_1(0) \\ \vdots \\ u_n(0) \end{bmatrix} = \mathrm{e}^{Dt}\mathbf{u}(0)\,.$$

**Definition 7.1.0.4. Exponentialmatrix $\mathrm{e}^{\mathbf{D}t}$**

Wir definieren das *Exponential einer Diagonalmatrix* $\mathbf{D}$, notiert $e^{Dt}$ als

$$\begin{bmatrix} e^{a_1 t} & & & 0 \\ & e^{a_2 t} & & \\ & & \ddots & \\ 0 & & & e^{a_n t} \end{bmatrix}$$

(alternativ: $\operatorname{diag}(e^{a_1 t}, e^{a_2 t}, \ldots, e^{a_n t})$). Das ist eine Diagonalmatrix.

Im Allgemeinen haben wir mit Systemen gekoppelter Differenzialgleichungen der Form $\dot{\mathbf{u}}(t) = \mathbf{A}\mathbf{u}(t)$ mit gegebener (und hier konstanter) Matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ zu tun. Schauen wir uns das folgende Zahlenbeispiel an:

**Beispiel 7.1.0.5 (System gekoppelter linearer Differenzialgleichungen 1. Ordnung)**
Für

$$\mathbf{A} = \begin{bmatrix} -4 & 1 \\ 1 & -4 \end{bmatrix}$$

können wir die Gleichungen als

$$\dot{u}_1(t) = -4u_1(t) + u_2(t)$$
$$\dot{u}_2(t) = u_1(t) - 4u_2(t)$$

lesen. ⌟

**Definition 7.1.0.6. Diagonalisierung**

Nehmen wir an, es gibt eine Matrix $\mathbf{S}$, so dass

$$\mathbf{A} = \mathbf{S}\mathbf{D}\mathbf{S}^{-1},$$

wobei $\mathbf{D}$ eine Diagonalmatrix ist. Ist diese *Diagonalisierung* möglich, so nennen wir die Matrix $\mathbf{A}$ *diagonalisierbar*.

Unter dieser Annahme rechnen wir:

$$\begin{aligned} \dot{\mathbf{u}} = \mathbf{A}\mathbf{u} &\Leftrightarrow \dot{\mathbf{u}} = \mathbf{S}\mathbf{D}\mathbf{S}^{-1}\mathbf{u} \\ &\Leftrightarrow \mathbf{S}^{-1}\dot{\mathbf{u}} = \mathbf{D}\mathbf{S}^{-1}\mathbf{u} \qquad |\mathbf{v} = \mathbf{S}^{-1}\mathbf{u};\ \ \mathbf{u} = \mathbf{S}\mathbf{v} \\ &\Leftrightarrow \dot{\mathbf{v}} = \mathbf{D}\mathbf{v} \qquad \text{mit } \mathbf{D} \text{ Diagonalmatrix} \end{aligned}$$

$$\mathbf{v}(t) = \mathrm{e}^{\mathbf{D}t}\mathbf{v}(0)\,.$$

Aus diesen Erkenntnissen sehen wir, dass eine Diagonalisierung unter Umständen nützlich für das Lösen von Differenzialgleichungen sein kann. Von $\mathbf{v}$ kommen wir auf $\mathbf{u}$ zurück:

$$\mathbf{u}(t) = \mathbf{S}\mathrm{e}^{\mathbf{D}t}\mathbf{v}(0) = \mathbf{S}\mathrm{e}^{\mathbf{D}t}\mathbf{S}^{-1}\mathbf{u}(0)\,.$$

Wir sehen, dass die Propagation von $\mathbf{u}(0)$ mit einer Matrix realisiert wird, die die Form der Diagonalisierung verwendet. Dies ist eine wichtige Erkenntnis, die wir zur Definition des Exponentials einer diagonalisierbaren Matrix verwenden können:

$$\mathrm{e}^{\mathbf{A}t} = \mathbf{S}\mathrm{e}^{\mathbf{D}t}\mathbf{S}^{-1}\,.$$

Zudem kennen wir aus der Analysis die folgende Schreibweise:

$$\mathrm{e}^{\mathbf{A}t} = \sum_{k=0}^{\infty} \frac{1}{k!}(\mathbf{A}t)^k\,.$$

Man kann zeigen, dass die Definitionen äquivalent zueinander sind. Da dieser Beweis zu den einfacheren Beweisen zählt, wird er dem Leser als Übung überlassen.

**Definition 7.1.0.7. Ähnlichkeit**

Man sagt, dass $\mathbf{A}$ und $\mathbf{B}$ *ähnlich* sind, falls eine invertierbare Matrix $\mathbf{S}$ existiert, so dass

$$\mathbf{A} = \mathbf{S}\mathbf{B}\mathbf{S}^{-1}\,.$$

**Bemerkung 7.1.0.8 (Diagonalisierbare Matrix)**
Eine Matrix $\mathbf{A}$ ist diagonalisierbar, dann und nur dann, wenn $\mathbf{A}$ ähnlich zu einer Diagonalmatrix ist. ⌟

**Bemerkung 7.1.0.9 (Diagonalisierung, Eigenwerte, Eigenvektoren)** Wenn eine Matrix diagonalisierbar ist, dann sind die Elemente auf der Diagonale von $\mathbf{D}$ Eigenwerte und die Spalten von $\mathbf{S}$ sind die entsprechenden Eigenvektoren. ⌟

## 7.2 Eigenwerte und Eigenvektoren

Gegeben sei eine quadratische $n \times n$-Matrix $\mathbf{A}$, sodass $\mathbf{y} = \mathbf{A}\mathbf{x}$. Für welche $\mathbf{x}$ ist das Ergebnis $\mathbf{y} = \mathbf{A}\mathbf{x}$ kollinear zu $\mathbf{x}$ ? D. h., für welche $\lambda \in \mathbb{C}$, gibt es ein $\mathbf{x} \neq \mathbf{0}$, so dass

$$\mathbf{A}\mathbf{x} = \lambda \mathbf{x}\,, \tag{7.2.0.1}$$

wobei $\lambda$ ein Eigenwert und $\mathbf{x}$ ein Eigenvektor von $\mathbf{A}$ ist.

**Definition 7.2.0.2. Eigenwertproblem**

$\lambda \in \mathbb{C}$ heisst *Eigenwert* der Matrix $\mathbf{A}$, falls es ein $\mathbf{x} \neq \mathbf{0}$ gibt, sodass $\mathbf{A}\mathbf{x} = \lambda \mathbf{x}$.
$\mathbf{x}$ heisst dann *Eigenvektor von* $\mathbf{A}$ *zum Eigenwert* $\lambda$.

**Satz 7.2.0.3. Diagonalisierung, Eigenwerte, Eigenvektoren**

*Wenn eine Matrix diagonalisierbar ist, dann sind die Elemente auf der Diagonale von* $\mathbf{D}$ *Eigenwerte und die Spalten von* $\mathbf{S}$ *sind die entsprechenden Eigenvektoren.*

**Satz 7.2.0.4. $n$ linear unabhängige Eigenvektoren diagonalisieren eine $n \times n$-Matrix**

*Sei* $\mathbf{A}$ *eine* $n \times n$*-Matrix mit* $n$ *linear unabhängigen Eigenvektoren, und seien diese Eigenvektoren die Spalten einer Matrix* $\mathbf{S}$

$$\mathbf{S} = \begin{bmatrix} \mathbf{s}_1 & \mathbf{s}_2 \cdots\cdots \mathbf{s}_n \end{bmatrix}.$$

*Dann ist*

$$\mathbf{S}^{-1}\mathbf{A}\mathbf{S} = \begin{bmatrix} \lambda_1 & 0 & \cdots & \cdots & 0 \\ 0 & \lambda_2 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 0 & \lambda_n \end{bmatrix} = \mathbf{D}\,,$$

*mit* $\lambda_1, \lambda_2, \ldots, \lambda_n$ *Eigenwerte von* $\mathbf{A}$.

*Beweis.*

$$\begin{aligned} \mathbf{A}\mathbf{S} &= \mathbf{A}\begin{bmatrix} \mathbf{s}_1 & \mathbf{s}_2 \cdots\cdots \mathbf{s}_n \end{bmatrix} = \begin{bmatrix} \mathbf{A}\mathbf{s}_1 & \mathbf{A}\mathbf{s}_2 \cdots\cdots \mathbf{A}\mathbf{s}_n \end{bmatrix} \\ &= \begin{bmatrix} \lambda_1\mathbf{s}_1 & \lambda_2\mathbf{s}_2 \cdots\cdots \lambda_n\mathbf{s}_n \end{bmatrix} \\ &= \mathbf{S}\mathbf{D} \end{aligned}$$

Somit $\mathbf{A}\mathbf{S} = \mathbf{S}\mathbf{D} \Rightarrow \mathbf{S}^{-1}\mathbf{A}\mathbf{S} = \mathbf{D}$.

□

**Satz 7.2.0.5. Verschiedene Eigenwerte haben linear unabhängige Eigenvektoren**

*Sei* $\mathbf{A}$ *eine* $k \times k$*-Matrix und* $\lambda_1, \lambda_2, \ldots, \lambda_k$ *paarweise verschiedene Eigenwerte von* $\mathbf{A}$. *Dann sind die entsprechenden Eigenvektoren* $\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_k$ *linear unabhängig.*

Dieser Satz wird im Folgenden basierend auf dem Prinzip der mathematischen Induktion bewiesen:
*Beweis.* Sei $P(k)$ die mathematische Aussage, dass $k$ Eigenvektoren linear unabhängig sind, falls die entsprechenden Eigenwerte paarweise verschieden sind.

**Induktionsanfang:** Für $k = 1$ stimmt die Aussage offensichtlich (es ist ja nur ein Vektor!).

**Induktionsschritt:** Nehmen wir an, die Aussage ist wahr für $k - 1$ und wir beweisen, dass sie auch für $k$ stimmt.

Seien also $\lambda_1, \lambda_2, \ldots, \lambda_k$ Eigenwerte von $\mathbf{A}$, die sich alle unterscheiden. Wir betrachten eine Linearkombination der entsprechenden Eigenvektoren $\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_k$:

$$\alpha_1\mathbf{s}_1 + \alpha_2\mathbf{s}_2 + \ldots + \alpha_k\mathbf{s}_k = \mathbf{0}\,.$$

Um die lineare Unabhängigkeit zu beweisen, müssen wir zeigen, dass daraus zwingend $\alpha_1 = \alpha_2 = \ldots = \alpha_k = 0$ folgt.

Wenn wir die vorherige Gleichung mit $\lambda_k$ multiplizieren, so erhalten wir

$$\alpha_1\lambda_k\mathbf{s}_1 + \alpha_2\lambda_k\mathbf{s}_2 + \ldots + \alpha_k\lambda_k\mathbf{s}_k = \mathbf{0}\,.$$

Wenn wir die gleiche Gleichung aber mit $\mathbf{A}$ multiplizieren, so erhalten wir

$$\alpha_1\mathbf{A}\mathbf{s}_1 + \alpha_2\mathbf{A}\mathbf{s}_2 + \ldots + \alpha_k\mathbf{A}\mathbf{s}_k = \mathbf{0}\,.$$

Da $\lambda_1, \lambda_2, \ldots, \lambda_k$ Eigenwerte von $\mathbf{A}$ sind, gilt

$$\alpha_1\lambda_1\mathbf{s}_1 + \alpha_2\lambda_2\mathbf{s}_2 + \ldots + \alpha_k\lambda_k\mathbf{s}_k = \mathbf{0}\,.$$

Wenn wir die zwei Gleichungen voneinander subtrahieren, verschwindet der letzte Term:

$$\alpha_1(\lambda_k - \lambda_1)\mathbf{s}_1 + \alpha_2(\lambda_k - \lambda_2)\mathbf{s}_2 + \ldots + \alpha_{k-1}(\lambda_k - \lambda_{k-1})\mathbf{s}_{k-1} = \mathbf{0}\,.$$

Die Induktionsannahme sichert, dass $\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_{k-1}$ linear unabhängig sind. Darum sind alle $\alpha_1(\lambda_k - \lambda_1) = \alpha_2(\lambda_k - \lambda_2) = \ldots = \alpha_{k-1}(\lambda_k - \lambda_{k-1}) = 0$. Da alle Eigenwerte verschieden sind, müssen alle $\alpha_1 = \ldots = \alpha_{k-1} = 0$. Wenn man diese in die ursprüngliche Gleichung einsetzt, erhält man $\alpha_k\mathbf{s}_k = \mathbf{0}$. Da der Eigenvektor definitionsgemäss nicht der Nullvektor ist, erhalten wir, dass auch $\alpha_k = 0$ gilt. $\alpha_k = 0$ sein. Wir folgern daraus, dass die entsprechenden Eigenvektoren linear unabhängig sind.
Nachdem $P(1)$ sowie $P(k-1) \implies P(k)$ bewiesen sind, schliessen wir mithilfe des Induktionsprinzips, dass die Aussage für alle $k \in \mathbb{N} \setminus \{0\}$ gilt.

□

**Bemerkung 7.2.0.6 ($n$ unterschiedliche Eigenwerte)** Wenn eine $n \times n$-Matrix $n$ unterschiedliche Eigenwerte hat, dann ist sie diagonalisierbar. ⌟

**Bemerkung 7.2.0.7 (Betrag der Eigenwerte)**
Wenn wir eine von einer Vektornorm induzierten Matrixnorm auf die Gleichung 7.2.0.1:

$$\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$$

anwenden, erhalten wir:

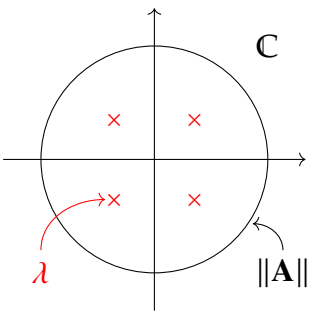
$$|\lambda| \cdot \|\mathbf{x}\| = \|\lambda\mathbf{x}\| = \|\mathbf{A}\mathbf{x}\| \leq \|\mathbf{A}\| \cdot \|\mathbf{x}\|\,,$$

wobei $\mathbf{x} \neq 0$ und somit $\|\mathbf{x}\| \neq 0$ gilt. Nach der Definition der Norm einer Matrix 4.1 schliessen wir

$$|\lambda| \leq \|\mathbf{A}\|$$

ab.

⌟



**Definition 7.2.0.8. Gerschgorin-Kreis**

Sei $\mathbf{A}$ eine $n \times n$-Matrix. Für jedes $1 \leq i \leq n$ definieren wir den Gerschgorin-Kreis

$$D_i = \{|z - a_{ii}| \leq r_i \,, z \in \mathbb{C}\} \,, \qquad \text{wobei } r_i = |a_{i1}| + \ldots + |a_{ii-1}| + |a_{ii+1}| + \ldots + |a_{in}| \,.$$

Das Gerschgorin-Gebiet ist die Vereinigung der Gerschgorin-Kreise einer Matrix:

$$D_{\mathbf{A}} = \bigcup_{i=1}^{n} D_i \subset \mathbb{C} \,.$$

In Worten: Der $i$-te Gerschgorin-Kreis hat sein Zentrum im $i$-ten Diagonaleintrag $a_{ii}$ und Radius $r_i$, der gleich der Summe der Beträge der $i$-ten Zeile *ohne* den Diagonaleintrag $a_{ii}$ ist. Das Gerschgorin-Gebiet ist nützlich, um die Lage der Eigenwerte einer Matrix zu bestimmen, ohne die Eigenwerte zu berechnen:

**Satz 7.2.0.9. Satz von Gerschgorin**

*Alle Eigenwerte einer Matrix $\mathbf{A}$ liegen im Gerschgorin-Gebiet $D_{\mathbf{A}} \subset \mathbb{C}$.*

*Beweis.* Sei $\lambda$ ein Eigenwert und $\mathbf{v}$ der entsprechende Eigenvektor von $\mathbf{A}$. Sei $\mathbf{u} = \mathbf{v}/\|\mathbf{v}\|_\infty$ der bezüglich der $\infty$-Norm normierte Eigenvektor. D. h. $\mathbf{u}$ zeigt in dieselbe Richtung wie $\mathbf{v}$ aber hat $\|\mathbf{u}\|_\infty = \max\{|u_1|,...,|u_n|\} = 1$. Sei $u_i$ ein Eintrag in $\mathbf{u}$ so dass $|u_i| = 1$. $\mathbf{u}$ ist auch ein Eigenvektor zum Eigenwert $\lambda$. Schreiben wir die $i$-te Gleichung aus $\mathbf{Au} = \lambda\mathbf{u}$ aus:

$$a_{i1}u_1 + \ldots + a_{in}u_n = \lambda u_i$$

oder auch

$$a_{i1}u_1 + \ldots + a_{ii-1}u_{i-1} + a_{ii+1}u_{i+1} + \ldots + a_{in}u_n = (\lambda - a_{ii})u_i \,.$$

Somit gilt:

$$|\lambda - a_{ii}| = |\lambda - a_{ii}| \cdot 1 = |\lambda - a_{ii}||u_i| = |(\lambda - a_{ii})u_i| = |a_{i1}u_1 + \ldots + a_{ii-1}u_{i-1} + a_{ii+1}u_{i+1} + \ldots + a_{in}u_n| \,.$$

Die Dreiecksungleichung und $|u_j| \leq 1$ ergeben also weiter:

$$|\lambda - a_{ii}| \leq |a_{i1}||u_1| + \ldots + |a_{ii-1}||u_{i-1}| + |a_{ii+1}||u_{i+1}| + \ldots + |a_{in}||u_n| \leq r_i \,.$$

Somit liegt $\lambda$ im $i$-ten Gerschgorin-Kreis: $\lambda \in D_i \subset D_{\mathbf{A}}$

□

**Satz 7.2.0.10. Lösungen für $\mathbf{Ax} = \mathbf{0}$**

*Es gibt für $\mathbf{Ax} = \mathbf{0}$ nicht-triviale Lösungen ($\mathbf{x} \neq \mathbf{0}$) dann und nur dann, wenn $0$ ein Eigenwert der Matrix $\mathbf{A}$ ist.*

Das heisst, wenn wir $0$ als Eigenwert von $\mathbf{A}$ ausschliessen können, dann wissen wir, dass die Matrix $\mathbf{A}$ regulär ist (d. h. $\mathbf{A}$ hat vollen Rang), ohne dass wir die Gauss-Elimination durchführen müssen. Wenn $0$ nicht im Gerschgorin-Gebiet $D_{\mathbf{A}}$ der Matrix $\mathbf{A}$ liegt, dann ist sie regulär! Das ist für Anwendungen so wichtig, dass es einen eigenen Namen verdient:

**Definition 7.2.0.11. Diagonaldominante Matrix**

Eine $n \times n$-Matrix heisst strikt diagonaldominant, wenn

$$|a_{i\,i}| > |a_{i\,1}| + \ldots + |a_{i\,i-1}| + |a_{i\,i+1}| + \ldots + |a_{i\,n}| \qquad \text{für alle } i = 1, \ldots, n.$$

**Satz 7.2.0.12. Strikt diagonaldominante Matrizen sind regulär**

*Eine strikt diagonaldominante Matrix ist regulär.*

*Beweis.* Wenn $\mathbf{A}$ strikt diagonaldominant ist, dann ist der Radius jedes Gerschgorin-Kreises kleiner als der Betrag seines Zentrums: $r_i < |a_{ii}|$; so liegt $0$ ausserhalb jedes Gerschgorin-Kreises. □

**Beispiel 7.2.0.13 (Eigenwerte und Eigenvektoren einer strikt diagonaldominanten Matrix)**

$$\mathbf{A} = \begin{bmatrix} 4 & 1 & 0 & 0 & 0 \\ 1 & 4 & 1 & 0 & 0 \\ 0 & 1 & 4 & 1 & 0 \\ 0 & 0 & 1 & 4 & 1 \\ 0 & 0 & 0 & 1 & 4 \end{bmatrix}$$

ist strikt diagonaldominant, also regulär. Die (numerisch berechneten)[2] Eigenwerte von $\mathbf{A}$ sind $2.26794919$, $3$, $4$, $5.73205081$, $5$. ⌟

**Beispiel 7.2.0.14 (Eine Matrix mit rein imaginären Eigenwerten)**
Sei

$$\mathbf{Q} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \in \mathbb{R}^{2\times 2}\,.$$

Diese $\mathbf{Q}$ erfüllt $\mathbf{Q}^2 = -\mathbf{I}$. Das kann man ausnutzen, um die Eigenwerte zu berechnen:

$$\begin{aligned} \mathbf{Q}\mathbf{x} = \lambda\mathbf{x} \;&\Rightarrow\; \mathbf{Q}^2\mathbf{x} = \lambda\mathbf{Q}\mathbf{x} = \lambda\lambda\mathbf{x} \\ &\Rightarrow\; -\mathbf{I}\mathbf{x} = \lambda^2\mathbf{x} \\ &\Rightarrow \lambda^2 = -1 \qquad\qquad \Rightarrow\; \lambda_1 = i, \lambda_2 = -i\,. \end{aligned}$$

Hierbei wurde verwendet, dass $\mathbf{x}$ ein Eigenvektor ist und somit definitionsgemäss nicht der Nullvektor ist. ⌟

**Bemerkung 7.2.0.15 (Reelle Matrizen können komplexe Eigenwerte haben)**
Eine reelle Matrix ($\mathbf{A} \in \mathbb{R}^{n\times n}$) kann komplexe Eigenwerte (d. h. in $\mathbb{C} \setminus \mathbb{R}$) haben. Diese kommen aber immer paarweise vor:
Wenn $\lambda \in \mathbb{C}$ Eigenwert einer reellen Matrix ($\mathbf{A} \in \mathbb{R}^{n\times n}$) ist, dann ist sein komplex konjugierter Wert $\overline{\lambda} \in \mathbb{C}$ auch ein Eigenwert (von $\mathbf{A}$).

⌟

*Beweis.*

$$\begin{aligned} \mathbf{A}\mathbf{x} = \lambda\mathbf{x} \;&\Rightarrow\; \overline{\mathbf{A}\mathbf{x}} = \overline{\lambda\mathbf{x}} \\ &\Leftrightarrow \mathbf{A}\overline{\mathbf{x}} = \overline{\lambda}\overline{\mathbf{x}} \\ &\Rightarrow \overline{\lambda}\ \text{EW}\,. \end{aligned}$$

□

**Bemerkung 7.2.0.16 (Berechnung der Eigenvektoren)**
Wenn ein Eigenwert $\lambda$ bekannt ist, reduziert sich die Frage nach den dazugehörigen Eigenvektoren auf das Lösen eines homogenen linearen Gleichungssystems:

$$\begin{aligned} \mathbf{A}\mathbf{x} = \lambda\mathbf{x} &\Leftrightarrow \mathbf{A}\mathbf{x} - \lambda\mathbf{x} = 0 \\ &\Leftrightarrow \mathbf{A}\mathbf{x} - \lambda\mathbf{I}\mathbf{x} = 0 \\ &\Leftrightarrow (\mathbf{A} - \lambda\mathbf{I})\mathbf{x} = 0\,. \end{aligned}$$

[2] Die Berechnung von Eigenwerten mit Python wird in Abschnitt 9.7 behandelt.

⌟

Die Bemerkung 1.2.1.2 und die Definition 2.2 sagen uns:
Dieses homogene lineare Gleichungssystem hat dann und nur dann nicht-triviale Lösungen (also Eigenvektoren) $\mathbf{x}$, wenn $\mathrm{Kern}(\mathbf{A}-\lambda\mathbf{I}) \neq \{\mathbf{0}\}$, d. h. wenn $\mathbf{A}-\lambda\mathbf{I}$ singulär ist, d. h. wenn $\det(\mathbf{A}-\lambda\mathbf{I}) = 0$.

Dank des Fundamentalsatzes der linearen Algebra wissen wir, dass die Lösungsmenge dieses homogenen Gleichungssystems einen **linearen Raum** $\mathrm{Kern}(\mathbf{A}-\lambda\mathbf{I})$ bildet. Dieser Raum enthält alle Eigenvektoren zum Eigenwert $\lambda$ sowie den Nullvektor und hat die Dimension $n-r$, wobei $r = \mathrm{Rang}(\mathbf{A}-\lambda\mathbf{I})$.

**Definition 7.2.0.17. Charakteristische Gleichung**

Um die Eigenwerte einer Matrix $\mathbf{A}$ auszurechnen, bedienen wir uns der Gleichung

$$\det(\mathbf{A}-\lambda\mathbf{I}) = 0\,.$$

Diese Gleichung heisst die *charakteristische Gleichung* der Matrix $\mathbf{A}$.

**Bemerkung 7.2.0.18 (Spur & Determinante)**
Mit der Bemerkung 6.1.0.12 können wir die charakteristische Gleichung so schreiben:

$$(-1)^n\lambda^n + (-1)^{n-1}(\mathrm{Spur}\,\mathbf{A})\lambda^{n-1} + \ldots + \det(\mathbf{A}) = 0\,,$$

wobei die *Spur der Matrix* $\mathbf{A}$ die Summe der Diagonalelemente ist:

$$\mathrm{Spur}\,\mathbf{A} = a_{11} + a_{22} + \ldots + a_{nn}\,.$$

Das heisst, die charakteristische Gleichung ist das Problem der Nullstellen eines Polynoms, das *charakteristische Polynom*. Der *Fundamentalsatz der Algebra*[3] sagt unter anderem, dass jedes Polynom vom Grad $n$ auch $n$ Nullstellen (mehrfache Nullstellen auch gezählt) in $\mathbb{C}$ hat. Die Nullstellen sind nicht unbedingt alle reell. Der Fundamentalsatz der Algebra sichert also:

**Satz 7.2.0.19. Existenz und Anzahl der Eigenwerte einer Matrix**

*Jede $n\times n$-Matrix hat mindestens einen Eigenwert und kann maximal $n$ verschiedene Eigenwerte haben.*

Man kann also das charakteristische Polynom auch als Produkt der Polynome vom Grad 1 schreiben, so dass man die Eigenwerte/Nullstellen einschliesslich deren Multiplizität sichtbar macht:

$$(-1)^n(\lambda-\lambda_1)(\lambda-\lambda_2)\cdots(\lambda-\lambda_n) = 0\,.$$

Wenn man die Koeffizienten der Monome in den zwei Darstellungen des charakteristischen Polynoms vergleicht (dank der Eindeutigkeit der Darstellung in einer Basis), sieht man sehr nützliche Beziehungen zwischen den Elementen der Matrix $\mathbf{A}$ und deren Eigenwerte:

$$\begin{aligned}\mathrm{Spur}\,\mathbf{A} = a_{11} + a_{22} + \ldots + a_{nn} &= \lambda_1 + \lambda_2 + \ldots + \lambda_n\\ \det\mathbf{A} &= \lambda_1\lambda_2\ldots\lambda_n\,.\end{aligned}$$

Das Berechnen der Nullstellen des charakteristischen Polynoms ist jedoch im allgemeinen Fall sehr problematisch. Wenn wir von Hand rechnen, haben wir bei Polynomen mit Grad $n \geq 5$ schon Probleme: Nach dem Satz von Abel-Ruffini existiert dann keine geschlossene Formel für die Lösungen. Ab dieser Grenze **muss** man also im Allgemeinen **iterative Algorithmen** verwenden, um die Eigenwerte und Eigenvektoren numerisch zu **approximieren**. Mit der Ausnahme von einigen glücklichen Fällen, in denen man die Nullstellen

[3] Nicht zu verwechseln mit dem Fundamentalsatz der *linearen* Algebra. Den Fundamentalsatz der Algebra beweisen wir nicht in dieser Vorlesung.

des charakteristischen Polynoms analytisch ausrechnen kann, muss man numerische Methoden verwenden. Für grosse Matrizen, die keine spezielle Struktur haben, die man clever ausnutzen kann, ist die Berechnung der Eigenwerte sehr **teuer** und die entsprechenden Eigenvektoren eventuell auch numerisch instabil. Weitere Details gibt es in einer Vorlesung über Numerische Methoden.

⌟

**Beispiel 7.2.0.20 (Eigenwerte und Eigenvektoren einer singulären Matrix)**
Die Matrix[4]

$$\mathbf{A} = \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix}$$

ist singulär und hat $\operatorname{Rang} \mathbf{A} = 1$, also $\operatorname{Kern} \mathbf{A} \neq \{\mathbf{0}\}$. Wir stellen die Matrix zur Berechnung der charakteristischen Gleichung auf:

$$\mathbf{A} - \lambda \mathbf{I} = \begin{bmatrix} 1-\lambda & 2 \\ 2 & 4-\lambda \end{bmatrix}.$$

Nun berechnen wir die Determinante der Matrix $\mathbf{A} - \lambda \mathbf{I}$:

$$\det(\mathbf{A} - \lambda \mathbf{I}) = (1-\lambda)(4-\lambda) - 2 \cdot 2 = \lambda^2 - 5\lambda = \lambda(\lambda - 5)\,.$$

Wir sehen, dass unsere charakteristische Gleichung durch

$$\lambda^2 - 5\lambda = 0$$

gegeben ist. Demzufolge sind die gesuchten Eigenwerte

$$\lambda_1 = 0\,, \quad \lambda_2 = 5\,.$$

Alternativ dazu kann man die Koeffizienten des charakteristischen Polynoms direkt mithilfe von Bemerkung 7.2.0.18 bestimmen.

$$n = 2 \qquad \operatorname{Spur} \mathbf{A} = 1 + 4 = 5 \qquad \det \mathbf{A} = (1 \cdot 4) - (2 \cdot 2) = 0$$

Wir erhalten also ebenfalls die charakteristische Gleichung

$$\lambda^2 - 5\lambda = 0\,.$$

Nun berechnen wir die Eigenvektoren zu den Eigenwerten. Wie wir wissen, lassen sich die zu $\lambda_1 = 0$ gehörigen Eigenvektoren mit $\mathbf{Ax} = \mathbf{0}$ leicht berechnen.

$$\mathbf{Ax} = \mathbf{0} \qquad \Leftrightarrow \qquad \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

Demzufolge erhalten wir durch Gauss-Elimination

$$\mathbf{x}_1 = \alpha \begin{bmatrix} -2 \\ 1 \end{bmatrix}, \qquad \alpha \in \mathbb{R}\,.$$

Das bedeutet, dass $[-2, 1]^\mathsf{T}$ ein Eigenvektor zum Eigenwert $\lambda_1 = 0$ ist. Wir sagen *ein* Eigenvektor (und nicht *der* Eigenvektor), weil es unendlich viele Eigenvektoren zu $\lambda_1$ gibt.

[4] Beispiel 4 vom Kapitel 6.1, Seite 287 in Str09.

**Definition 7.2.0.21. Eigenraum**

Der *Eigenraum zum Eigenwert* $\lambda$ ist der lineare Raum, der von den Eigenvektoren zum Eigenwert $\lambda$ aufgespannt wird:

$$E_\lambda = \operatorname{span}\{\mathbf{x} \neq \mathbf{0}, \text{ mit } \mathbf{A}\mathbf{x} = \lambda\mathbf{x}\}\,.$$

Der Eigenraum zu $\lambda_1$ ist dementsprechend

$$E_0 = \operatorname{span}\{\begin{bmatrix}-2\\1\end{bmatrix}\} = \operatorname{Kern}(\mathbf{A})$$

und hat Dimension 1. Nun berechnen wir noch den Eigenvektor zu $\lambda_2 = 5$:

$$(\mathbf{A} - 5\mathbf{I})\mathbf{x} = \mathbf{0} \qquad \Longleftrightarrow \qquad \begin{bmatrix}-4 & 2\\ 2 & -1\end{bmatrix}\begin{bmatrix}x_1\\x_2\end{bmatrix} = \begin{bmatrix}0\\0\end{bmatrix}.$$

Durch Gauss-Elimination erhalten wir:

$$\mathbf{x} = \alpha\begin{bmatrix}1\\2\end{bmatrix}.$$

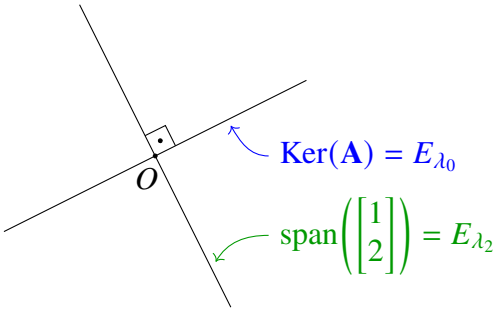
Der Eigenraum zu $\lambda_2$ lautet dann:

$$E_5 = \operatorname{span}\{\begin{bmatrix}1\\2\end{bmatrix}\}\,.$$

**Bemerkung 7.2.0.22 (Orthogonalität mancher Eigenvektoren)**
Wir bemerken:

$$\left\langle\begin{bmatrix}-2\\1\end{bmatrix}, \begin{bmatrix}1\\2\end{bmatrix}\right\rangle = [-2\ 1]\cdot\begin{bmatrix}1\\2\end{bmatrix} = -2+2 = 0.$$

Diese Eigenvektoren sind orthogonal zueinander.




**Beispiel 7.2.0.23 (Gauss-Elimination zur Lösung des Eigenwertproblems)**
Gegeben sei die Matrix[5]

$$\mathbf{A} = \begin{bmatrix}2 & 1 & 1\\ 1 & 2 & 1\\ 1 & 1 & 2\end{bmatrix}.$$

[5] Beispiel 2 vom Kapitel 7.1, Seite 148 in NS15.

Wir berechnen nun $\det(\mathbf{A}-\lambda\mathbf{I})$ indem wir Zeilen vertauschen und die Gauss-Elimination durchführen:

$$\begin{aligned}\det(\mathbf{A}-\lambda\mathbf{I}) &= \det\begin{bmatrix}2-\lambda & 1 & 1\\ 1 & 2-\lambda & 1\\ 1 & 1 & 2-\lambda\end{bmatrix} = -\det\begin{bmatrix}1 & 1 & 2-\lambda\\ 1 & 2-\lambda & 1\\ 2-\lambda & 1 & 1\end{bmatrix}\\ &= -\det\begin{bmatrix}1 & 1 & 2-\lambda\\ 0 & 1-\lambda & -1+\lambda\\ 0 & \lambda-1 & 1-(2-\lambda)^2\end{bmatrix} = -\det\begin{bmatrix}1 & 1 & 2-\lambda\\ 0 & 1-\lambda & -1+\lambda\\ 0 & 0 & -4+5\lambda-\lambda^2\end{bmatrix}\\ &= -1\cdot(1-\lambda)(-4+5\lambda-\lambda^2) = -1\cdot(\lambda-1)^2(\lambda-4)\,.\end{aligned}$$

Somit lautet die charakteristische Gleichung:

$$\det(\mathbf{A}-\lambda\mathbf{I}) = 0 \qquad \Leftrightarrow \qquad (\lambda-1)^2(\lambda-4) = 0\,.$$

Wir erhalten also die Lösungen/Eigenwerte

$$\lambda_1 = 1 \text{ und } \lambda_2 = 4\,,$$

wobei $\lambda_1$ die sogenannte *algebraische Multiplizität* 2 und $\lambda_2$ die *algebraische Multiplizität* 1 hat.

**Definition 7.2.0.24. Algebraische Multiplizität (AM)**

Die *algebraische Multiplizität* (auch *algebraische Vielfachheit* genannt) zeigt uns, wie oft $\lambda$ unter den Nullstellen der charakteristischen Gleichung vorkommt.

Wir berechnen nun die Eigenvektoren. Für $\lambda_1 = 1$ schreiben wir das entstehende lineare Gleichungssystem:

$$\begin{bmatrix}1 & 1 & 1\\ 1 & 1 & 1\\ 1 & 1 & 1\end{bmatrix}\mathbf{x} = \begin{bmatrix}0\\0\\0\end{bmatrix}$$

$$\begin{bmatrix}1 & 1 & 1\\ 0 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\mathbf{x} = \begin{bmatrix}0\\0\\0\end{bmatrix}$$

$$\mathbf{x} = \alpha\begin{bmatrix}-1\\0\\1\end{bmatrix} + \beta\begin{bmatrix}-1\\1\\0\end{bmatrix} = \alpha\mathbf{u}+\beta\mathbf{v}\,.$$

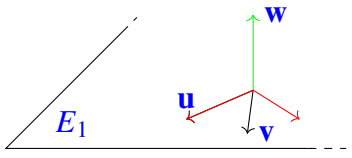
Der Eigenraum zu $\lambda_1$ wird von zwei Vektoren aufgespannt:

$$E_1 = \operatorname{span}\{\mathbf{u},\mathbf{v}\} = \operatorname{span}\{\begin{bmatrix}-1\\0\\1\end{bmatrix}, \begin{bmatrix}-1\\1\\0\end{bmatrix}\}\,.$$

Man sagt, der Eigenwert $\lambda_1$ hat *geometrische Multiplizität (GM)* 2. Wir bemerken jedoch, dass die gefundenen Vektoren $\mathbf{u}$ und $\mathbf{v}$ nicht senkrecht aufeinander stehen, da

$$\langle \mathbf{u},\mathbf{v}\rangle = 1+0+0 = 1 \neq 0 \qquad \Rightarrow \qquad \mathbf{u} \not\perp \mathbf{v}\,.$$

Mit dem Gram-Schmidt-Algorithmus kann man aber eine orthonormale Basis von $E_1 = \operatorname{span}\{\mathbf{u},\mathbf{v}\}$ finden.



Für $\lambda_2 = 4$ haben wir das lineare Gleichungssystem:

$$\begin{bmatrix} 1 & 1 & -2 \\ 0 & -3 & 3 \\ 0 & 0 & 0 \end{bmatrix} \mathbf{x} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\mathbf{x} = \alpha \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \alpha \mathbf{w} .$$

Wir stellen folgende Orthogonalitäten fest:

$$\langle \mathbf{w}, \mathbf{u} \rangle = -1 + 0 + 1 = 0 \quad \Leftrightarrow \quad \mathbf{w} \perp \mathbf{u} ,$$
$$\langle \mathbf{w}, \mathbf{v} \rangle = -1 + 1 + 0 = 0 \quad \Leftrightarrow \quad \mathbf{w} \perp \mathbf{v} ,$$

d. h., der Eigenraum $E_2$ steht orthogonal auf dem Eigenraum $E_1$.
In diesem Beispiel hat der Eigenwert $\lambda_1$ die *geometrische Multiplizität* 2. Der Eigenwert $\lambda_2$ hat die *geometrische Multiplizität* 1.

⌟

**Definition 7.2.0.25. Geometrische Multiplizität (GM)**

Die *geometrische Multiplizität* (auch *geometrische Vielfachheit* genannt) eines Eigenwertes ist die Dimension vom Eigenraum.

**Bemerkung 7.2.0.26 (Wenn alle Eigenwerte AM 1 haben, dann ist die Matrix diagonalisierbar)**
Wenn alle Eigenwerte einer Matrix algebraische Multiplizität 1 haben, sind sie alle verschieden, also hat die $n \times n$-Matrix $n$ linear unabhängige Eigenvektoren und somit ist sie diagonalisierbar.

⌟

Damit eine Matrix nicht diagonalisierbar ist, muss sie also so sein, dass es unmöglich ist, $n$ linear unabhängige Eigenvektoren zu haben. Zumindest muss sie Eigenwerte mit algebraischer Multiplizität grösser 1 haben.

**Satz 7.2.0.27. Ähnliche Matrizen haben gleiche Eigenwerte**

*Ähnliche Matrizen haben das gleiche charakteristische Polynom. Sie haben also die gleichen Eigenwerte mit den gleichen algebraischen Multiplizitäten.*

*Beweis.* Seien $\mathbf{A}$ und $\mathbf{B}$ zwei ähnliche Matrizen. Nach der Definition 7.1 gibt es eine invertierbare Matrix $\mathbf{S}$ so, dass

$$\mathbf{A} = \mathbf{S}^{-1}\mathbf{B}\mathbf{S} .$$

Das charakteristische Polynom von $\mathbf{A}$ ist

$$\begin{aligned} \det(\mathbf{A} - \lambda\mathbf{I}) &= \det(\mathbf{S}^{-1}\mathbf{B}\mathbf{S} - \lambda\mathbf{S}^{-1}\mathbf{I}\mathbf{S}) = \det(\mathbf{S}^{-1}(\mathbf{B} - \lambda\mathbf{I})\mathbf{S}) \\ &= \det(\mathbf{S}^{-1}) \det(\mathbf{B} - \lambda\mathbf{I}) \det \mathbf{S} = (\det \mathbf{S})^{-1} \det(\mathbf{B} - \lambda\mathbf{I}) \det \mathbf{S} = \\ &= \det(\mathbf{B} - \lambda\mathbf{I}) , \end{aligned}$$

d. h. gleich dem charakteristischen Polynom von $\mathbf{B}$. Dabei haben wir verschiedene Rechenregeln der Determinanten verwendet.

□

**Satz 7.2.0.28. Multiplizitätsregel**

*Für die verschiedenen Multiplizitäten eines Eigenwerts gilt:*

$$1 \leq GM \leq AM\,. \tag{7.2.0.29}$$

*Beweis.* Sei $\lambda$ ein Eigenwert der $n\times n$-Matrix $\mathbf{A}$. Für $\mathbf{M} = \mathbf{A} - \lambda\mathbf{I}$ gilt also $\det\mathbf{M} = 0$ und somit $\operatorname{Rang}\mathbf{M} = r \leq n-1$. Sei $g$ die geometrische Multiplizität von $\lambda$. Der Fundamentalsatz der linearen Algebra liefert $g = \dim\operatorname{Kern}\mathbf{M} = n - r \geq 1$. Wählen wir $\{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_g\}$ als Basis des $\operatorname{Kern}\mathbf{M}$ und $\{\mathbf{s}_{g+1}, \ldots, \mathbf{s}_n\}$ als Basis des $\operatorname{Bild}\mathbf{M}^T$. Nach dem Fundamentalsatz der linearen Algebra ist $\{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_g, \mathbf{s}_{g+1}, \ldots, \mathbf{s}_n\}$ eine Basis des $\mathbb{R}^n$.
Wir schreiben alle diese $n$ Basisvektoren in einer Matrix $\mathbf{S} = \begin{bmatrix}\mathbf{s}_1 & \mathbf{s}_2 \cdots\cdots \mathbf{s}_g & \mathbf{s}_{g+1} \cdots\cdots \mathbf{s}_n\end{bmatrix}$ zusammen. Da es sich um eine Basis handelt, ist die Matrix $\mathbf{S}$ regulär. Notieren wir ihre Inverse mit $\mathbf{V} = \mathbf{S}^{-1}$ und die Zeilen davon seien $\mathbf{v}_1^\mathsf{T}, \ldots, \mathbf{v}_n^\mathsf{T}$. Da $\mathbf{VS} = \mathbf{I}$ ist, gilt $\mathbf{v}_i^\mathsf{T}\mathbf{s}_j = 0$ für $i \neq j$ und $1$ sonst.
Aus der Definition von $\mathbf{M}$ sehen wir

$$\mathbf{A}\mathbf{s}_j = \lambda\mathbf{s}_j\,, \text{ für } j = 1, 2, \ldots, g\,.$$

Sei $\mathbf{B}$ die zu $\mathbf{A}$ ähnliche Matrix

$$\mathbf{B} = \mathbf{S}^{-1}\mathbf{A}\mathbf{S} = \mathbf{V}\begin{bmatrix}\lambda\mathbf{s}_1 & \lambda\mathbf{s}_2 \cdots\cdots \lambda\mathbf{s}_g & \mathbf{A}\mathbf{s}_{g+1} \cdots\cdots \mathbf{A}\mathbf{s}_n\end{bmatrix}.$$

Die Zeilen von $\mathbf{V} = \mathbf{S}^{-1}$ treffen also die ersten $g$ Spalten von $\mathbf{S}$ und wir erhalten die folgende Blockstruktur in $\mathbf{B}$:

$$\mathbf{B} = \left[\begin{array}{ccc|c} \lambda & & 0 & * \\ & \ddots & & \vdots \\ 0 & & \lambda & * \\ \hline \multicolumn{3}{c|}{\mathbf{0}} & \mathbf{C} \end{array}\right] \begin{matrix} \\ \\ g \\ \\ n-g \end{matrix} \quad \begin{matrix} g & n-g \end{matrix} = \left[\begin{array}{c|c} \lambda\mathbf{I} & * \\ \hline \mathbf{0} & \mathbf{C} \end{array}\right].$$

Der linke obere Block in $\mathbf{B}$ ist die $g\times g$-Identitätsmatrix multipliziert mit $\lambda$. Da $\mathbf{B}$ und $\mathbf{A}$ ähnlich sind, haben sie dasselbe charakteristische Polynom und dieselben Eigenwerte mit derselben Multiplizität.
Die spezielle Blockstruktur von $\mathbf{B}$ erlaubt uns $g$ Eigenwerte von $\mathbf{B}$ einfach auszurechnen. Wir entwickeln $\det(\mathbf{B} - z\mathbf{I})$ nach der ersten Spalte ($g$-mal nacheinander, immer in einer kleineren Matrix) und erhalten die charakteristische Gleichung (in der Variable $z$, damit wir es nicht mit dem festen Eigenwert $\lambda$ verwechseln)

$$0 = \det(\mathbf{B} - z\mathbf{I}) = (\lambda - z)^g \det(\mathbf{C} - z\mathbf{I})\,.$$

Das heisst, der Eigenwert $\lambda$ hat mindestens die algebraische Multiplizität $g$.

□

Es folgt automatisch, dass wenn ein Eigenwert die algebraische Multiplizität $1$ hat, die geometrische Multiplizität auch $1$ ist.

**Beispiel 7.2.0.30 (Rechenführung beim Eigenwertproblem)** [6]
Gegeben sei die Matrix[7]:

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 0 & 1 \\ 0 & -1 & 0 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & -1 \end{bmatrix}.$$

[6] Dieses Beispiel wird in Abschnitt 7.2.0.30 mit Python gelöst.
[7] Beispiel 3 vom Kapitel 7.2, Seite 151 in NS15.

Die charakteristische Gleichung $\det(\mathbf{A} - \lambda\mathbf{I}) = 0$ ist also

$$\det \begin{bmatrix} 1-\lambda & 2 & 0 & 1 \\ 0 & -1-\lambda & 0 & 1 \\ 0 & 1 & 1-\lambda & 1 \\ 0 & 0 & 0 & -1-\lambda \end{bmatrix} = 0\,.$$

Bei der Berechnung von Determinanten haben wir Rechenführungstechniken gelernt: Mittels Linearkombinationen von Spalten brachten wir die Determinante zu einer einfachen Form, sodass wir mit einer Entwicklung nach einer Zeile/Spalte das Problem vereinfachten. Dabei haben wir Einfachheit halber versucht, Divisionen zu vermeiden. Hier ist besonders wichtig, Divisionen mit Polynomen in $\lambda$ zu vermeiden, damit wir keine unnötigen Falldiskussionen führen müssen (ausserdem wissen wir, dass das Ergebnis ein Polynom vom Grad $n$ in $\lambda$) ist.
Wenn wir nachher auch noch die Eigenvektoren finden müssen, bevorzuge ich eine dem Problem angepasste Gauss-Elimination durchzuführen. In diesem Beispiel sehen wir, dass in der ersten Spalte nur das erste Element nicht-null ist, d. h. wir können die Gauss-Elimination bereits mit der zweiten Spalte beginnen. Damit wir die Division mit $-1-\lambda$ vermeiden, vertauschen wir die zweite und dritte Zeile. Das bringt einen Vorzeichenwechsel in der Determinante mit sich, da wir aber nicht die Determinante, sondern die Gleichung mit rechter Seite $0$ lösen wollen, spielt dies keine Rolle:

$$\det \begin{bmatrix} 1-\lambda & 2 & 0 & 1 \\ 0 & 1 & 1-\lambda & 1 \\ 0 & -1-\lambda & 0 & 1 \\ 0 & 0 & 0 & -1-\lambda \end{bmatrix} = 0\,.$$

Wir multiplizieren die zweite Zeile mit $1+\lambda$ und addieren zur dritten Zeile:

$$\det \begin{bmatrix} 1-\lambda & 2 & 0 & 1 \\ 0 & 1 & 1-\lambda & 1 \\ 0 & 0 & (1-\lambda)(1+\lambda) & 1+(1+\lambda) \\ 0 & 0 & 0 & -1-\lambda \end{bmatrix} = 0\,.$$

Die Gauss-Elimination hat uns auf die Form einer oberen Dreiecksmatrix geführt. Daraus lesen wir die charakteristische Gleichung in bereits faktorisierter Form:

$$(\lambda-1)^2(\lambda+1)^2 = 0\,.$$

Also sind die Lösungen

$$\lambda_1 = 1 \text{ (AM 2)} \qquad \lambda_2 = -1 \text{ (AM 2)}\,.$$

Wir sehen, dass beide Eigenwerte algebraische Multiplizität $2$ haben.

Nun wollen wir auch die dazugehörigen Eigenvektoren berechnen. Wir sehen hier den Vorteil der obigen Rechenführung: Die Gauss-Elimination für das Lösen der entsprechenden homogenen linearen Systeme steht bereits, wir müssen nur noch $\lambda$ einsetzen.
Für $\lambda = 1$ haben wir das homogene lineare Gleichungssystem

$$\begin{bmatrix} 0 & 2 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 3 \\ 0 & 0 & 0 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = 0\,.$$

mit freien Variablen $x_1$ und $x_3$. Aus der letzten Gleichung sehen wir $x_4 = 0$ und aus der zweiten Gleichung $x_2 = 0$. Somit ist der Eigenraum zu $\lambda = 1$:

$$E_{\lambda=1} = \operatorname{Span}\left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \right\}.$$

Die geometrische Multiplizität vom Eigenwert $\lambda = 1$ ist also 2.
Für $\lambda = -1$ haben wir das homogene lineare Gleichungssystem

$$\begin{bmatrix} 2 & 2 & 0 & 1 \\ 0 & 1 & 2 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = 0 .$$

Hier passiert etwas Besonderes: Die letzte Gleichung ist eine (erfüllte) Kompatibilitätsbedingung und die dritte Gleichung legt den Wert von $x_4 = 0$ fest! Es bleibt nur eine freie Variable $x_3$ übrig. Aus der zweiten Gleichung sehen wir $x_2 = -2x_3$ und aus der ersten $x_1 = -x_2 = 2x_3$. Somit ist das der Eigenraum zu $\lambda = -1$:

$$E_{\lambda=-1} = \text{Span}\left\{ \begin{bmatrix} 2 \\ -2 \\ 1 \\ 0 \end{bmatrix} \right\} .$$

Die geometrische Multiplizität vom Eigenwert $\lambda = -1$ ist 1, also weniger als die algebraische Multiplizität.

Es gibt also keine 4 linear unabhängigen Eigenvektoren von $\mathbf{A}$. Diese Matrix ist nicht diagonalisierbar!

⌟

**Bemerkung 7.2.0.31 (Nicht diagonalisierbare Matrizen)**
Wann ist $\mathbf{A}$ nicht diagonalisierbar?

$$\begin{aligned} \text{EW } \mathbf{A}: &\quad \lambda_1, \lambda_2, \ldots, \lambda_j, \ldots \\ \text{AM}: &\quad a_1, a_2, \ldots, a_j, \ldots \\ \text{GM}: &\quad g_1, g_2, \ldots, g_j, \ldots \end{aligned}$$

Für die Multiplizitäten gilt immer noch

$$\begin{aligned} a_1 + a_2 + \ldots + a_j + \ldots = n \\ 1 \le g_j \le a_j \forall j . \end{aligned}$$

Wir schreiben

$$\begin{aligned} g_j = \dim \ \text{Kern}(\mathbf{A} - \lambda_j \mathbf{I}) \qquad & \text{Eigenraum zu EW} \lambda_j \\ \Rightarrow \mathbf{u}_1^j, \mathbf{u}_2^j, \ldots, \mathbf{u}_{g_j}^j \ & \text{Basis des } \text{Kern}(\mathbf{A} - \lambda_j \mathbf{I}) . \end{aligned}$$

Mit dem Gram-Schmidt-Verfahren erhalten wir schliesslich eine orthonormale Basis

$$\mathbf{s}_1^j, \mathbf{s}_2^j, \ldots, \mathbf{s}_{g_j}^j$$

im Eigenraum zum Eigenwert $\lambda_j$.
Schreiben wir diese Eigenvektoren zusammen, so erhalten wir

$$\mathbf{S} = \left[ \mathbf{s}_1^1 \ \ \mathbf{s}_2^1 \cdot \cdot \cdot \cdot \cdot \mathbf{s}_{g_1}^1 \ \ \mathbf{s}_1^2 \ \ \mathbf{s}_2^2 \ \ \mathbf{s}_{g_2}^2 \cdot \cdot \cdot \cdot \right].$$

In $\mathbf{S}$ haben wir $n$ linear unabhängige Spalten nur dann, wenn $g_j = a_j$ für alle $j$. Somit gilt, falls es einen Eigenwert $\lambda_j$ gibt, sodass

$$g_j < a_j$$

dann haben wir nicht $n$ linear unabhängige Eigenvektoren. Es folgt, dass $\mathbf{A}$ nicht diagonalisierbar ist!

⌟

Das vorige Beispiel 7.2.0.30 zeigt uns, dass **nicht jede Matrix diagonalisierbar ist**. Die Bemerkung 7.2.0.31 zeigt uns, wann genau das passiert.

**Satz 7.2.0.32. Zusammenhang zwischen GM und AM entscheidet, ob diagonalisierbar oder nicht**

*Wenn die Matrix* $\mathbf{A}$ *einen Eigenwert besitzt, dessen geometrische Multiplizität strikt kleiner als die algebraische Multiplizität ist, dann ist die Matrix nicht diagonalisierbar.*
*Wenn für alle Eigenwerte die geometrische Multiplizität gleich der algebraischen Multiplizität ist, dann ist die Matrix diagonalisierbar.*

## 7.3 Symmetrische Matrizen

In den Anwendungen treten oft symmetrische (oder Hermite-symmetrische) Matrizen auf. Die Symmetrie bringt besonders nützliche Eigenschaften und Vereinfachungen mit sich. Zur Erinnerung: Eine symmetrische Matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ (bzw. eine Hermite-symmetrische Matrix $\mathbf{A} \in \mathbb{C}^{n\times n}$) erfüllt:

$$\mathbf{A}^T = \mathbf{A} \qquad (\mathbf{A}^H = \mathbf{A})\,.$$

Mit dem jeweiligen Euklidischen Skalarprodukt haben wir:

- $\langle\cdot,\cdot\rangle$ in $\mathbb{R}^n$:
$$\langle \mathbf{x}, \mathbf{A}\mathbf{y}\rangle = \mathbf{x}^T\mathbf{A}\mathbf{y} = \mathbf{x}^T\mathbf{A}^T\mathbf{y} = \langle \mathbf{A}\mathbf{x}, \mathbf{y}\rangle\,,$$
- $\langle\cdot,\cdot\rangle$ in $\mathbb{C}^n$:
$$\langle \mathbf{x}, \mathbf{A}\mathbf{y}\rangle = \mathbf{x}^H\mathbf{A}\mathbf{y} = \mathbf{x}^H\mathbf{A}^H\mathbf{y} = \langle \mathbf{A}\mathbf{x}, \mathbf{y}\rangle\,,$$

d. h. (Hermite-) symmetrische Matrizen sind selbstadjungiert.

**Satz 7.3.0.1. Symmetrische Matrizen haben reelle Eigenwerte und orthogonale Eigenvektoren**

*Sei $\mathbf{A} \in \mathbb{R}^{n\times n}$ (bzw. $\mathbf{A} \in \mathbb{C}^{n\times n}$) eine symmetrische (bzw. Hermite-symmetrische) Matrix. Dann sind alle Eigenwerte von $\mathbf{A}$ reell und Eigenvektoren zu verschiedenen Eigenwerten von $\mathbf{A}$ sind orthogonal.*

*Beweis.* Seien $\lambda$, $\mu$ Eigenwerte von $\mathbf{A}$ mit Eigenvektoren $\mathbf{u}$,$\mathbf{v}$:

$$\mathbf{A}\mathbf{u} = \lambda\mathbf{u}\,, \quad \mathbf{A}\mathbf{v} = \mu\mathbf{v}\,.$$

Aus der ersten Gleichung folgt

$$\lambda\mathbf{v}^H\mathbf{u} = \mathbf{v}^H\mathbf{A}\mathbf{u}\,.$$

Da $\mathbf{A}^H = \mathbf{A}$ gilt, kann man schreiben:

$$\begin{aligned}\mathbf{v}^H\mathbf{A}^H\mathbf{u} &= (\mathbf{A}\mathbf{v})^H\mathbf{u}\\ &= (\mu\mathbf{v})^H\mathbf{u} = \overline{\mu}\mathbf{v}^H\mathbf{u}\,.\end{aligned}$$

Somit haben wir gezeigt, dass $\lambda\mathbf{v}^H\mathbf{u} = \overline{\mu}\mathbf{v}^H\mathbf{u}$. Wählt man zunächst $\mu = \lambda$ sowie $\mathbf{v} = \mathbf{u}$, erhalten wir:

$$\lambda\mathbf{u}^H\mathbf{u} = \overline{\lambda}\mathbf{u}^H\mathbf{u}\,.$$

$\mathbf{u}^H\mathbf{u}$ lässt sich als $\|\mathbf{u}\|_2^2$ schreiben. Da für den Eigenvektor $\mathbf{u}$ definitionsgemäss $\mathbf{u} \neq \mathbf{0}$ gilt, muss auch $\|\mathbf{u}\|_2^2 \neq 0$ gelten. Wir schliessen für $\lambda$:

$$\lambda = \overline{\lambda} \quad \Rightarrow \quad \lambda \in \mathbb{R}\,.$$

Somit sind alle Eigenwerte von $\mathbf{A}$ reell. Übrig bleibt nur noch, die Orthogonalität zu zeigen. Aus der soeben abgeleiteten Bedingung folgt

$$\lambda\mathbf{v}^H\mathbf{u} = \mu\mathbf{v}^H\mathbf{u} \quad \Rightarrow \quad (\lambda-\mu)\mathbf{v}^H\mathbf{u} = 0\,.$$

Wählt wir nun $\lambda \neq \mu$, dann erhalten wir

$$\mathbf{v}^H\mathbf{u} = 0 \quad \mathbf{v} \perp \mathbf{u}\,.$$

□

**Bemerkung 7.3.0.2 (Matrizen mit besonderer Struktur haben spezielle Eigenwerte)**
Struktur zahlt sich aus: Je nachdem, wie die transponierte Matrix mit der ursprünglichen Matrix zusammenhängt, können wir ablesen, in welcher Menge sich die Eigenwerte befinden:

$$\begin{aligned}\mathbf{A}^\mathsf{T} = \mathbf{A} \quad &\Rightarrow \quad \lambda \in \mathbb{R}\\ \mathbf{A}^\mathsf{T} = -\mathbf{A} \quad &\Rightarrow \quad \imath\lambda \in \mathbb{R}\\ \mathbf{A}^\mathsf{T}\mathbf{A} = \mathbf{I} \quad &\Rightarrow \quad |\lambda| = 1\,.\end{aligned}$$

Für Eigenvektoren zu verschiedenen Eigenwerten dieser Art von Matrizen gilt immer noch, dass sie orthogonal aufeinander stehen.

⌟

**Definition 7.3.0.3. Normale Matrix**

Die Matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ (bzw. $\mathbb{C}^{n\times n}$) heisst *normal*, wenn $\mathbf{A}^\mathsf{T}\mathbf{A} = \mathbf{A}\mathbf{A}^\mathsf{T}$ (bzw. $\mathbf{A}^\mathsf{H}\mathbf{A} = \mathbf{A}\mathbf{A}^\mathsf{H}$) erfüllt ist.

**Bemerkung 7.3.0.4 (Symmetrische Matrizen sind normal)**
Jede symmetrische Matrix ist normal.

⌟

Wir werden später (nach der Schurzerlegung) folgendes stärkere Theorem 7.6.1.7 beweisen:

**Satz 7.3.0.5.**
**Normale Matrizen sind die durch unitäre Transformationen diagonalisierbaren Matrizen**

*$\mathbf{A}$ hat $n$ orthogonale Eigenvektoren dann und nur dann, wenn $\mathbf{A}$ normal ist.*

Damit gilt automatisch das wohl bekannteste Theorem aus der linearen Algebra:

**Satz 7.3.0.6. Spektralsatz**

*Jede symmetrische Matrix lässt sich durch orthogonale Transformationen diagonalisieren.*

Mit anderen Worten: Sei $\mathbf{A}$ eine $n \times n$ reelle symmetrische (komplexe Hermite-symmetrische) Matrix. Dann gibt es eine orthogonale (unitäre) Matrix $\mathbf{U}$ und eine diagonale Matrix $\mathbf{D} \in \mathbb{R}^{n\times n}$, so dass

$$\mathbf{A} = \mathbf{U}\,\mathbf{D}\,\mathbf{U}^\mathsf{H}.$$

Der Spektralsatz ist eines der wichtigsten Theoreme in dieser Vorlesung, er hat unzählige Anwendungen in allen Disziplinen, aber vor allem in der Physik.

In den Beispielen 7.2.0.13, 7.2.0.20. 7.2.0.23 hatten wir reelle symmetrische Matrizen, die also durch orthogonale Transformationen diagonalisierbar sind: es gibt jeweils orthogonale Basen bestehend nur aus Eigenvektoren. Im Beispiel 7.2.0.30 haben wir keine diagonalisierbare Matrix, also sicher auch keine normale Matrix. Im Beispiel 7.2.0.14 haben wir eine schiefsymmetrische Matrix, also eine normale Matrix, wie auch die Matrizen aus der Bemerkung 7.3.0.2: sie sind jeweils diagonalisierbar mit unitären Transformationen. Auch wenn diese Matrizen nur reelle Einträge haben, sind die Eigenwerte effektiv komplexe Zahlen und die Eigenvektoren liegen in $\mathbb{C}^n$ und nicht in $\mathbb{R}^n$. Hier ist noch ein Beispiel:

**Beispiel 7.3.0.7.** Sei

$$\mathbf{A} = \begin{bmatrix} 5 & 4 \\ -4 & 5 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{i}{\sqrt{2}} & -\frac{i}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 5+4i & 0 \\ 0 & 5-4i \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{i}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{i}{\sqrt{2}} \end{bmatrix} \in \mathbb{R}^2.$$

Diese reelle Matrix $\mathbf{A}$ ist weder symmetrisch noch schiefsymmetrisch, aber sie ist normal: $\mathbf{A}^H\mathbf{A} = \mathbf{A}^T\mathbf{A} = 41\mathbf{I} = \mathbf{A}\mathbf{A}^T = \mathbf{A}\mathbf{A}^H$. Eine Basis von Eigenvektoren ist $\mathbf{v}_1 = [1, i]^T$, $\mathbf{v}_2 = [1, -i]^T$. Orthogonalität dieser Vektoren gilt im Sinne des komplexen Skalarprodukts: $\langle v_1, v_2\rangle = v_1^H v_2 = 1 + i^2 = 0$ und nicht im Sinne des reellen Skalarprodukts: $v_1^T v_2 = 1 - i^2 = 2 \neq 0$. Die Matrix $\mathbf{A}$ hat echt komplexe Eigenwerte und Eigenvektoren in $\mathbb{C}^n$ und nicht in $\mathbb{R}^n$. Es ist klar, aber doch interessant zu schauen, was beim Transponieren passiert: $\mathbf{A}^T = (\mathbf{U}^H\mathbf{D}\mathbf{U})^T = \mathbf{U}^T\mathbf{D}\bar{\mathbf{U}} \neq \mathbf{A}$ und $\mathbf{A}^H = (\mathbf{U}^H\mathbf{D}\mathbf{U})^H = \mathbf{U}^H\bar{\mathbf{D}}\mathbf{U} \neq \mathbf{A}$. ⌟

**Beispiel 7.3.0.8.** Die Matrix $\mathbf{A}$ aus dem Beispiel 7.2.0.23 ist symmetrisch, also durch orthogonale Transformationen diagonalisierbar. Bei den Berechnungen haben wir den Eigenwert 1 mit der algebraischen (und

geometrischen) Multiplizität 2 und den Eigenwert 4 mit der algebraischen (und geometrischen) Multiplizität 1 gefunden. Die dort berechneten linear unabhängigen Eigenvektoren zum Eigenwert 1 sind nicht orthogonal: $\mathbf{u} = [-1, 0, 1]$ und $\mathbf{v} = [-1, 1, 0]$. Das ist kein Widerspruch, denn wir können aus $\mathbf{u}, \mathbf{v}$ (z. B. mit Gram-Schmidt) eine orthonormale Basis für den Eigenraum zum Eigenwert 1 finden. Der Eigenvektor $w = [1, 1, 1]^T$ zum Eigenwert 4 steht orthogonal auf $\mathbf{u}$ und $\mathbf{v}$, also auf dem ganzen Eigenraum zum Eigenwert 1, so dass es eine orthogonale Basis in $\mathbb{R}^3$ gibt, die aus Eigenvektoren der Matrix $\mathbf{A}$ besteht.

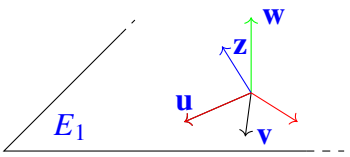
Jetzt verändern wir dieses Beispiel, sodass eine neue Matrix entsteht, die diagonalisierbar ist, aber nicht mit unitären Transformationen, d. h., die auch nicht normal ist. Wir behalten den Raum zum doppelten Eigenwert 1, d. h. die Vektoren $\mathbf{u}, \mathbf{v}$ als Eigenvektoren zum Eigenwert 1, und zum Eigenwert 4 wählen wir einen Vektor $\mathbf{z}$, der nicht orthogonal darauf ist, z. B. $\mathbf{z} = [1, 1, 0]^T$. Wir erhalten die Matrizen

$$\mathbf{S} = [\mathbf{u}\,\mathbf{v}\,\mathbf{z}] = \begin{bmatrix} -1 & -1 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

und

$$\mathbf{B} = \mathbf{S}\,\mathrm{diag}\{1,1,4\}\,\mathbf{S}^{-1} = \begin{bmatrix} -1 & -1 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 4 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \\ -\frac{1}{2} & \frac{1}{2} & -\frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} = \begin{bmatrix} 5/2 & 3/2 & 3/2 \\ 3/2 & 5/2 & 3/2 \\ 0 & 0 & 1 \end{bmatrix}.$$

Die so erhaltene Matrix $\mathbf{B}$ ist zwar diagonalisierbar, jedoch gibt es keine orthogonale Basis von Eigenvektoren von $\mathbf{B}$, weder in $\mathbb{R}^3$ noch in $\mathbb{C}^3$. Sie ist auch nicht normal. Die Matrizen $\mathbf{A}$ und $\mathbf{B}$ haben dieselben Eigenwerte mit denselben Multiplizitäten und einen gemeinsamen Eigenraum. Der Eigenraum zum Eigenwert 4 der Matrix $\mathbf{B}$ steht nicht orthogonal auf dem Eigenraum zum Eigenwert 1:




**Bemerkung 7.3.0.9 (Normale Matrizen sind Summen von unitären/orthogonalen Projektoren)**
Sei $\mathbf{A}$ eine normale/symmetrische Matrix. Nach dem Spektralsatz lässt sie sich durch unitäre/orthogonale Transformationen diagonalisieren:

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix} \mathbf{u}_1 & \mathbf{u}_2 \cdots\cdots \mathbf{u}_n \end{bmatrix} \begin{bmatrix} \lambda_1 & & 0 \\ & \lambda_2 & \\ & & \ddots \\ 0 & & \lambda_n \end{bmatrix} \begin{bmatrix} \mathbf{u}_1^{\mathsf{H}} \\ \mathbf{u}_2^{\mathsf{H}} \\ \vdots \\ \mathbf{u}_n^{\mathsf{H}} \end{bmatrix} \\ &= \begin{bmatrix} \lambda_1\mathbf{u}_1 & \lambda_2\mathbf{u}_2 \cdots\cdots \lambda_n\mathbf{u}_n \end{bmatrix} \begin{bmatrix} \mathbf{u}_1^{\mathsf{H}} \\ \mathbf{u}_2^{\mathsf{H}} \\ \vdots \\ \mathbf{u}_n^{\mathsf{H}} \end{bmatrix} \\ &= \lambda_1\mathbf{u}_1\mathbf{u}_1^{\mathsf{H}} + \lambda_2\mathbf{u}_2\mathbf{u}_2^{\mathsf{H}} + \ldots + \lambda_n\mathbf{u}_n\mathbf{u}_n^{\mathsf{H}} \\ &= \text{Summe von Rang-1-Matrizen.} \end{aligned}$$

Wir notieren $\mathbf{u}_j\mathbf{u}_j^{\mathsf{H}} = \mathbf{P}_j$. Offensichtlich gilt $\mathbf{P}_j = \mathbf{P}_j^{\mathsf{H}}$. Ausserdem gilt:

$$\mathbf{P}_j^2 = \mathbf{P}_j\mathbf{P}_j = \mathbf{u}_j\mathbf{u}_j^{\mathsf{H}}\mathbf{u}_j\mathbf{u}_j^{\mathsf{H}} = \mathbf{u}_j\|\mathbf{u}_j\|_2^2\mathbf{u}_j^{\mathsf{H}} = \mathbf{u}_j\mathbf{u}_j^{\mathsf{H}} = \mathbf{P}_j\,.$$

Damit ist $\mathbf{P}_j$ der orthogonale Projektor auf $\mathbf{u}_j$. D. h., $\mathbf{A}$ lässt sich schreiben als die Summe der orthogonalen Projektoren auf die Elemente der Orthonormalbasis, die aus den Eigenvektoren besteht:

$$\mathbf{A} = \sum_{j=1}^{n} \lambda_j \mathbf{P}_j$$

und

$$\sum_{j=1}^{n} \mathbf{P}_j = \mathbf{U}\mathbf{U}^{\mathsf{H}} = \mathbf{I}\,.$$

⌟

**Bemerkung 7.3.0.10 (Diagonalisierbare Matrizen sind Summen von (schiefen) Projektoren)**
Was gilt, wenn die Matrix $\mathbf{A}$ nur diagonalisierbar (also nicht durch orthogonale Transformationen) ist? In diesem Fall haben wir nur:

$$\mathbf{A} = \mathbf{S}\mathbf{\Lambda}\mathbf{S}^{-1} = \mathbf{S}\mathbf{\Lambda}\mathbf{V} = \begin{bmatrix} | & & | \\ \mathbf{s}_1 & \cdots & \mathbf{s}_n \\ | & & | \end{bmatrix} \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix} \begin{bmatrix} — & \mathbf{v}_1^{\mathsf{T}} & — \\ — & \mathbf{v}_2^{\mathsf{T}} & — \\ & \vdots & \\ — & \mathbf{v}_n^{\mathsf{T}} & — \end{bmatrix},$$

wobei $\mathbf{v}_1^{\mathsf{T}}, \ldots, \mathbf{v}_n^{\mathsf{T}}$ die Zeilen der Matrix $\mathbf{V} = \mathbf{S}^{-1}$ sind. Wichtig zu beachten ist, dass die Zeilenvektoren von $\mathbf{S}$ nicht zwingend denen von $\mathbf{V} = \mathbf{S}^{-1}$ gleichen müssen (das wäre nur so wenn $\mathbf{S}$ orthogonal wäre, also nicht in diesem Fall). Für $\mathbf{A}$ können wir nun schreiben:

$$\mathbf{A} = \sum_{j=1}^{n} \lambda_j \mathbf{s}_j \mathbf{v}_j^{\mathsf{T}} = \sum_{j=1}^{n} \lambda_j \mathbf{P}_j$$

wobei $\mathbf{P}_j$ ein *schiefer* Projektor ist (siehe Beispiel 4.5.0.5). ⌟

**Bemerkung 7.3.0.11 (Optimierung mit einer Diagonalmatrix)** Sei die Diagonalmatrix $\mathbf{D} = \operatorname{diag}\{\lambda_1, \lambda_2, \ldots, \lambda_n\}$ mit monoton fallend angeordneten Eigenwerte $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$ und die Funktion $f : \mathbb{R}^n \to \mathbb{R}^n$ mit

$$f(\mathbf{y}) = \mathbf{y}^T \mathbf{D}\mathbf{y} = \lambda_1 y_1^2 + \lambda_2 y_2^2 + \ldots + \lambda_n y_n^2\,.$$

Für $\|\mathbf{y}\| = 1$ gilt offensichtlich

$$\lambda_1 = \lambda_1 \|\mathbf{y}\|^2 = \lambda_1 y_1^2 + \lambda_1 y_2^2 + \ldots + \lambda_1 y_n^2 \geq f(\mathbf{y}) \geq \lambda_n \|\mathbf{y}\|^2 = \lambda_n\,,$$

so dass:

$$\lambda_1 = \max_{\|\mathbf{y}\|=1} f(\mathbf{y}) \text{ erreicht für } \mathbf{y} = \mathbf{e}_1\,,$$
$$\lambda_n = \min_{\|\mathbf{y}\|=1} f(\mathbf{y}) \text{ erreicht für } \mathbf{y} = \mathbf{e}_n\,.$$

Das kann man zu einer symmetrischen Matrix $\mathbf{A}$ erweitern:

$$g(\mathbf{x}) = \mathbf{x}^T \mathbf{A}\mathbf{x} = \mathbf{x}^T \mathbf{Q}\mathbf{D}\mathbf{Q}^T \mathbf{x} = \mathbf{y}^T \mathbf{D}\mathbf{y}\,,$$

mit $\mathbf{y} = \mathbf{Q}^T\mathbf{x}$ und $\mathbf{Q}$ orthogonal. Somit gilt:

$$\lambda_1 = \max_{\|\mathbf{x}\|=1} g(\mathbf{x}) \text{ erreicht für } \mathbf{x} = \mathbf{Q}\mathbf{e}_1 = \mathbf{q}_1\,,$$
$$\lambda_n = \min_{\|\mathbf{x}\|=1} g(\mathbf{x}) \text{ erreicht für } \mathbf{x} = \mathbf{Q}\mathbf{e}_n = \mathbf{q}_n\,.$$

Einen allgemeinen Vektor $\mathbf{v} \neq 0$ kann man zu einem normierten $\mathbf{x} = \mathbf{v}/\|\mathbf{v}\|$ umschreiben, so dass:

$$\lambda_1 = \max_{\mathbf{v}\neq 0} \frac{\mathbf{x}^T\mathbf{A}\mathbf{x}}{\|\mathbf{x}\|^2} \text{ erreicht für } \mathbf{x} = \mathbf{q}_1 ,$$
$$\lambda_n = \min_{\mathbf{v}\neq 0} \frac{\mathbf{x}^T\mathbf{A}\mathbf{x}}{\|\mathbf{x}\|^2} \text{ erreicht für } \mathbf{x} = \mathbf{q}_n .$$

Die Grösse $\frac{\mathbf{x}^T\mathbf{A}\mathbf{x}}{\|\mathbf{x}\|^2}$ heisst Rayleigh-Quotient und ist sehr wichtig für numerische Algorithmen für Eigenwertprobleme. ⌟

**Satz 7.3.0.12. Euklidische Norm und Eigenwerte**

1. *Für jede quadratische Matrix $\mathbf{A} \in \mathbb{C}^{n\times n}$ gilt:*
$$\|\mathbf{A}\|_2 = \sqrt{\max\{d;\, d \text{ Eigenwert von } \mathbf{A}^H\mathbf{A}\}} .$$
*Für $\mathbf{A} \in \mathbb{R}^{n\times n}$ gilt eine analoge Aussage mit $\mathbf{A}^T\mathbf{A}$ statt $\mathbf{A}^H\mathbf{A}$.*
2. *Für jede orthogonale/unitäre Matrix $\mathbf{U}$ gilt:*
$$\|\mathbf{U}\|_2 = 1 .$$
3. *Ist $\mathbf{A}$ eine symmetrische Matrix, dann gilt:*
$$\|\mathbf{A}\|_2 = \max\{|\lambda|;\, \lambda \text{ Eigenwert von } \mathbf{A}\} .$$

*Beweis.*

1. Da die Matrix $\mathbf{A}^H\mathbf{A}$ Hermite-symmetrisch ist, lässt sie sich mit einer unitären Matrix $\mathbf{U}$ diagonalisieren: $\mathbf{A}^H\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{U}^H$. So rechnen wir
$$\|\mathbf{A}\mathbf{U}\mathbf{x}\|_2^2 = \langle \mathbf{A}\mathbf{U}\mathbf{x}, \mathbf{A}\mathbf{U}\mathbf{x}\rangle = \mathbf{x}^H\mathbf{U}^H\mathbf{A}^H\mathbf{A}\mathbf{U}\mathbf{x} = \mathbf{x}^H\mathbf{D}\mathbf{x} = \sum_{j=1}^{n} d_j|x_j|^2 .$$
Unitäre/orthogonale Transformationen verändern Längen nicht: $\|\mathbf{x}\|_2 = \|\mathbf{U}\mathbf{x}\|_2$. Nach der Definition der Matrixnorm 4.1 haben wir:
$$\|\mathbf{A}\|_2^2 = \sup_{\|\mathbf{y}\|_2=1} \|\mathbf{A}\mathbf{y}\|_2^2 = \sup_{\|\mathbf{U}\mathbf{x}\|_2=1} \|\mathbf{A}\mathbf{U}\mathbf{x}\|_2^2 = \sup_{\|\mathbf{x}\|_2=1} \mathbf{x}^H\mathbf{D}\mathbf{x} = \sup_{\|\mathbf{x}\|_2=1} \sum_{j=1}^{n} d_j|x_j|^2 = \max_{j=1,\ldots,n} d_j .$$
Daraus sieht man auch, dass $\max_{j=1,\ldots,n} d_j \geq 0$.

2. Für eine unitäre/orthogonale Matrix $\mathbf{U}$ mit Eigenwert $\lambda$ und entsprechendem Eigenvektor $\mathbf{x}$ haben wir:
$$\mathbf{U}\mathbf{x} = \lambda\mathbf{x} \Rightarrow |\lambda|\|\mathbf{x}\|_2 = \|\mathbf{U}\mathbf{x}\|_2 = \|\mathbf{x}\|_2 \Rightarrow |\lambda| = 1 .$$

3. Sei nun $\mathbf{A}$ eine (Hermite-) symmetrische Matrix. Nach dem Spektralsatz ist sie diagonalisierbar durch orthogonale / unitäre Transformationen: $\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^H$ mit reellen Eigenwerten $\lambda_1, \ldots \lambda_n$. Dann ist $\mathbf{A}^H\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^H\mathbf{U}\mathbf{\Lambda}\mathbf{U}^H = \mathbf{U}\mathbf{\Lambda}^2\mathbf{U}^H$. Nach dem ersten Punkt in diesem Satz ist also
$$\|\mathbf{A}\|_2^2 = \max\{|d|;\, d \text{ Eigenwert von } \mathbf{A}^H\mathbf{A}\} = \max\{\lambda^2;\, \lambda^2 \text{ Eigenwert von } \mathbf{A}\} .$$

□

Analog kann man ein Resultat für die inverse Matrix erhalten:

**Satz 7.3.0.13. Die Eigenwerte der inversen Matrix**

*Für eine invertierbare Matrix* $\mathbf{A}$ *gilt:*

1. $$\left\|\mathbf{A}^{-1}\right\|_2 = \frac{1}{\inf_{\|\mathbf{y}\|_2=1}\|\mathbf{A}\mathbf{y}\|_2} = \frac{1}{\sqrt{\min\{d,\, d \text{ Eigenwert von } \mathbf{A}^{\mathrm{H}}\mathbf{A}\}}}\,.$$

   *Damit ist auch der kleinste Eigenwert von* $\mathbf{A}^{\mathrm{H}}\mathbf{A}$ *strikt positiv.*

2. *Für eine invertierbare symmetrische Matrix* $\mathbf{A}$ *gilt*

   $$\left\|\mathbf{A}^{-1}\right\|_2 = \frac{1}{\min\{|\lambda|,\, \lambda \text{ Eigenwert von } \mathbf{A}\}}\,.$$

   *Damit ist auch der betragskleinste Eigenwert von* $\mathbf{A}$ *nicht Null.*

**Definition 7.3.0.14. Konditionszahl**

Die Konditionszahl einer invertierbaren Matrix $\mathbf{A}$ ist

$$k(\mathbf{A}) = \|\mathbf{A}\|_2\left\|\mathbf{A}^{-1}\right\|_2 = \sqrt{\frac{d_{\max}}{d_{\min}}}\,,$$

wobei $d_{\max}$ der grösste und $d_{\min}$ der kleinste Eigenwert von $\mathbf{A}^H\mathbf{A}$ ist.

**Bemerkung 7.3.0.15 (Kondition einer symmetrischen Matrix)**
Falls $\mathbf{A}$ invertierbar und symmetrisch ist, lautet ihre Konditionszahl

$$k(\mathbf{A}) = \|\mathbf{A}\|_2\left\|\mathbf{A}^{-1}\right\|_2 = \frac{|\lambda_{\max}|}{|\lambda_{\min}|}\,,$$

wobei $|\lambda_{\max}|$ der betragsgrösste und $|\lambda_{\min}|$ der betragskleinste Eigenwert von $\mathbf{A}$ ist.

⌟

Man kann beweisen, dass die Konditionszahl der Matrix der Verstärkungsfaktor des relativen Fehlers in den Daten eines linearen Gleichungssystems mit der Matrix $\mathbf{A}$ ist. Matrizen mit kleiner Kondition (nahe 1) sind gut konditioniert (keine grosse Verstärkung). Matrizen mit grosser Kondition sind schlecht konditioniert (grosse Verstärkung).
Die Konditionszahl der Normalengleichung ist das *Quadrat* der Konditionszahl der Matrix bei der Ausgleichsrechnung. Darum ist die Methode mit der Normalengleichung in den numerischen Berechnungen zu vermeiden. Besser ist die Anwendung der QR-Zerlegung, welche orthogonale Transformationen verwendet, die bestmöglich konditioniert sind.

## 7.4 Symmetrisch positiv definite Matrizen

**Bemerkung 7.4.0.1 (Das Produkt der Eigenwerte ist das Produkt der Diagonalelemente von U)**
In der Bemerkung 7.2.0.18 haben wir gesehen, dass das Produkt der Eigenwerte die Determinante ist:

$$\det \mathbf{A} = \lambda_1 \lambda_2 \cdots \lambda_n .$$

Weiter erinnern wir uns an die erste Definition der Determinante 1.5: Das Produkt der Diagonalelemente nach der Gauss-Elimination ist die Determinante. Notieren wir die Diagonalelemente $u_{11}, u_{22}, \ldots, u_{nn}$ der Matrix $\mathbf{U}$ in der LU-Zerlegung (Gauss-Elimination). Dann gilt:

$$\lambda_1 \lambda_2 \ldots \lambda_n = u_{11} u_{22} \ldots u_{nn} = \det \mathbf{A} .$$

⌟

Mit dieser Bemerkung und ein wenig Analysis kann man noch folgendes beweisen:

**Bemerkung 7.4.0.2 (Anzahl positiver Eigenwerte symmetrischer, invertierbarer Matrizen)**
Für eine symmetrische und invertierbare Matrix ist die Anzahl strikt positiver Eigenwerte gleich der Anzahl strikt positiver Pivotelemente.

⌟

**Definition 7.4.0.3. Symmetrisch positiv definite Matrizen**

Eine symmetrische Matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$

1. heisst *symmetrisch positiv definit (s.p.d.)*, falls

$$\mathbf{x}^T \mathbf{A} \mathbf{x} \geq 0 \text{ für alle } \mathbf{x} \in \mathbb{R}^n$$
$$\mathbf{x}^T \mathbf{A} \mathbf{x} = 0 \implies \mathbf{x} = \mathbf{0}$$

   gilt.

2. heisst *symmetrisch positiv semi-definit*, falls nur

$$\mathbf{x}^T \mathbf{A} \mathbf{x} \geq 0 \text{ für alle } \mathbf{x} \in \mathbb{R}^n$$

   erfüllt ist.

**Bemerkung 7.4.0.4 (Skalarprodukt entsprechend einer s.p.d. Matrix)**
Eine symmetrisch positiv definite Matrix $\mathbf{A}$ definiert, wie im Beispiel 4.3.0.6 bereits gesehen, ein Skalarprodukt auf $\mathbb{R}^n$:

$$\langle \mathbf{x}, \mathbf{y} \rangle_{\mathbf{A}} = \mathbf{x}^T \mathbf{A} \mathbf{y} .$$

⌟

**Satz 7.4.0.5. Eigenwerte einer s.p.d. Matrix**

*Die symmetrische Matrix $\mathbf{A}$ ist positiv definit dann und nur dann, wenn alle ihre Eigenwerte strikt positiv sind.*

*Beweis.* Sei $\mathbf{A}$ eine symmetrische Matrix, dann lässt sich $\mathbf{A}$ mit orthogonalen Transformationen diagonalisieren: $\mathbf{A} = \mathbf{S} \mathbf{\Lambda} \mathbf{S}^T$, mit $\mathbf{S}$ orthogonal und den Eigenwerten $\lambda_1, \ldots, \lambda_n \in \mathbb{R}$. Es folgt:

$$\mathbf{x}^T \mathbf{A} \mathbf{x} = \mathbf{x}^T (\mathbf{S} \mathbf{\Lambda} \mathbf{S}^T) \mathbf{x} = (\mathbf{x}^T \mathbf{S}) \mathbf{\Lambda} (\mathbf{S}^T \mathbf{x}) \quad | \mathbf{y} = \mathbf{S}^T \mathbf{x}$$
$$= \mathbf{y}^T \mathbf{\Lambda} \mathbf{y} = \sum_{j=1}^{n} \lambda_j |y_j|^2 .$$

Wir nehmen zuerst an, dass alle Eigenwerte von $\mathbf{A}$ strikt positiv sind. Wir wollen zeigen, dass sie positiv definit ist. Offensichtlich:

$$\mathbf{x}^T\mathbf{A}\mathbf{x} = \sum_{j=1}^{n} \lambda_j |y_j|^2 \geq 0 \quad \text{für alle } \mathbf{x}$$

$$\mathbf{x}^T\mathbf{A}\mathbf{x} = 0 \;\Leftrightarrow\; \sum_{j=1}^{n} \lambda_j |y_j|^2 = 0 \;\Rightarrow\; |y_j| = 0 \text{ für } j = 1, 2, \ldots, n\,.$$

Da alle Eigenwerte *strikt* positiv sind, ist die letzte Summe nur $0$, wenn alle $y_j = 0$ sind:

$$\begin{aligned} &\mathbf{y} = \mathbf{0} \\ \Leftrightarrow\; &\mathbf{x} = \mathbf{S}\mathbf{y} = \mathbf{S}\mathbf{0} = \mathbf{0}\,. \end{aligned}$$

Nun nehmen wir an, dass $\mathbf{A}$ positiv definit ist und wollen zeigen, dass alle ihre Eigenwerte strikt positiv sind. Nehmen wir einen Vektor $\mathbf{y}$, definiert als:

$$\mathbf{y} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \mathbf{S}^T\mathbf{x}\,.$$

Mit dem entsprechenden

$$\mathbf{x} = \mathbf{S}\begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \neq 0\,,$$

bleibt aus der Summe nur ein Term:

$$0 \leq \mathbf{x}^T\mathbf{A}\mathbf{x} = \mathbf{y}^T\mathbf{\Lambda}\mathbf{y} = \lambda_1\,.$$

Wäre $\lambda_1 = 0$, dann hätten wir ein $\mathbf{x} \neq \mathbf{0}$ gefunden, für welches $\mathbf{x}^T\mathbf{A}\mathbf{x} = 0$, was der Annahme widerspricht, dass $\mathbf{A}$ positiv definit ist. Somit muss $\lambda_1 > 0$ sein.
Dieses Argument kann man für alle anderen Elemente der Standardbasis in der Rolle von $\mathbf{y}$ wiederholen, so dass wir $\lambda_j > 0$ für $j = 1, 2, \ldots, n$ erhalten.

□

**Bemerkung 7.4.0.6 (S.p.d.-Test via Pivotelemente.)**
Wenn die Gauss-Elimination einer symmetrischen $n \times n$-Matrix $n$ (strikt) positive Pivotelemente hat, dann ist die Matrix positiv definit.

In der Tat, da $\mathbf{A}$ symmetrisch ist gibt es eine Cholesky-Zerlegung

$$\mathbf{A} = \mathbf{L}\mathbf{D}\mathbf{L}^T\,,$$

wobei die Elemente der Diagonalmatrix $\mathbf{D}$ die Pivotelemente der Gauss-Elimination sind. Da all diese strikt positiv sind, braucht man in der Gauss-Elimination auch keine Zeilenvertauschungen, also kommt in der LU-Zerlegung auch keine Permutation vor. Somit

$$\begin{aligned} \mathbf{x}^T\mathbf{A}\mathbf{x} &= \mathbf{x}^T\mathbf{L}\mathbf{D}\mathbf{L}^T\mathbf{x} \qquad |\text{notiere } \mathbf{y} = \mathbf{L}^T\mathbf{x} \\ &= \sum_{k=1}^{n} d_k y_k^2 > 0\,, \end{aligned}$$

für $\mathbf{x} \neq 0$.

Die Bedingung, dass die Matrix symmetrisch ist, ist notwendig: Es gibt nicht-symmetrische Matrizen, die nur strikt positive Pivotelemente haben, aber nicht positiv definit sind.

⌟

### 7.4.1 Anwendungen

Nun wollen wir einige Anwendungen (bzw. Eigenschaften) der symmetrischen positiven Definitheit diskutieren.

1. Gegeben sei nun eine symmetrische Matrix (einfachheitshalber nur $2 \times 2$):

$$\mathbf{A} = \begin{bmatrix} a & b \\ b & c \end{bmatrix}.$$

Das Quadrat der von dem $\mathbf{A}$-Skalarprodukt $\langle \mathbf{x}, \mathbf{x} \rangle_{\mathbf{A}}$ induzierten $\mathbf{A}$-Norm ist:

$$\mathbf{x}^\mathsf{T}\mathbf{A}\mathbf{x} = ax_1^2 + 2bx_1x_2 + cx_2^2 = f(x_1, x_2)\,.$$

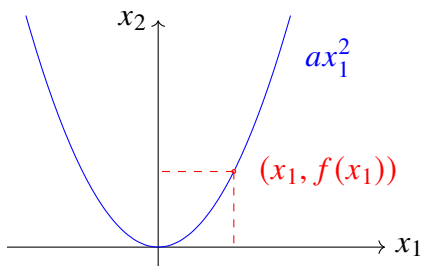
Diese Art von Funktion heisst *quadratische Form*. Ist nun die Matrix $\mathbf{A}$ s.p.d., so gilt für die quadratische Form des Skalarprodukts:

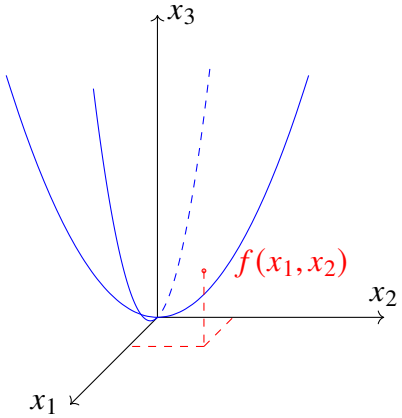
$$\begin{aligned} \mathbf{A} \text{ s.p.d.} \quad \Longleftrightarrow \quad & f(x_1, x_2) \geq 0 \ \ \forall\, x_1, x_2 \in \mathbb{R} \\ & = 0, \ \ x_1 = x_2 = 0\,. \end{aligned}$$

Das bedeutet, dass die quadratische Form $f$ ein Minimum in $x_1 = 0,\ x_2 = 0$ hat. Stellen wir dies grafisch dar, so sehen wir, dass die quadratische Form

$$f : \mathbb{R}^n \times \mathbb{R}^n \longrightarrow \mathbb{R}, \quad f(x) = \mathbf{x}^\mathsf{T}\mathbf{A}\mathbf{x}$$

mit einer symmetrisch positiven Matrix $\mathbf{A}$ die natürliche Verallgemeinerung der Parabel von einer Dimension auf $n$ Dimensionen ist. Sie hat ein Minimum bei $\mathbf{x} = \mathbf{0}$.




2. Sei $g : \mathbb{R}^n \longrightarrow \mathbb{R}$ differenzierbar. Für $n = 1$ wissen wir, dass an einem lokalen Minimum die erste Ableitung null und die zweite Ableitung positiv ist. Für $n \geq 2$ ergibt sich aus der Bedingung zur ersten Ableitung

$$Dg(\mathbf{x})^\mathsf{T} = \left[ \frac{\partial g}{\partial x_1}(\mathbf{x}) \cdots \frac{\partial g}{\partial x_n}(\mathbf{x}) \right]^\mathsf{T} = \mathbf{0}$$

ein lineares Gleichungssystem. Das entspricht dem linearen Term in der Taylor-Reihe für $g$. Das entspricht einer Approximation durch eine Tangentialebene an dem Graph von $g$, z. B. die Approximation um $\mathbf{0}$ ist $g(\mathbf{0}) + Dg(\mathbf{0})^\mathsf{T}\mathbf{x}$. Die zweite Ableitung von $g$ ist eine $n \times n$-Matrix

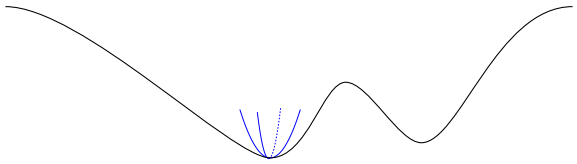
$$D^2 g(\mathbf{x}) = \left[ \frac{\partial^2 g}{\partial x_i \partial x_j}(\mathbf{x}) \right]_{i,j=1,\ldots,n},$$

die den quadratischen Term in der Taylor-Reihe um $\mathbf{0}$ für $g$ bestimmt:

$$g(\mathbf{x}) \approx g(\mathbf{0}) + Dg(\mathbf{0})^\mathsf{T}\mathbf{x} + \frac{1}{2}\mathbf{x}^\mathsf{T} D^2 g(\mathbf{0})\mathbf{x}.$$

Diese quadratische Form hat ein Minimum in $\mathbf{0}$ dann und nur dann, wenn die symmetrische Matrix $D^2 g(\mathbf{0})$ positiv definit ist.
Entsprechend müssen die folgenden Bedingungen gelten, so dass $\mathbf{x}$ ein lokales Minimum ist:

$$Dg(\mathbf{x}) = \mathbf{0}$$
$$D^2 g(\mathbf{x}) \text{ s.p.d.}$$



Um dieses fundamentale Prinzip zu festigen, behandeln wir ein Zahlenbeispiel:

**Beispiel 7.4.1.1 (Positive quadratische Form)**
Sei die quadratische Form $f(x_1, x_2) = x_1^2 + 8x_1x_2 + cx_2^2$. Für welches $c$ ist $f(x_1, x_2) > 0$ für alle $x_1, x_2 \neq 0$? Die entsprechende Matrix $\mathbf{A}$ sieht so aus:

$$\mathbf{A} = \begin{bmatrix} 1 & 4 \\ 4 & c \end{bmatrix}.$$

Zuerst berechnen wir die Eigenwerte der Matrix und bestimmen $c$ so, dass $\lambda_1 > 0$, $\lambda_2 > 0$. Als Trick können wir aber auch die Vorzeichen der Pivotelemente anschauen, denn wir dürfen nicht vergessen, dass eine symmetrische Matrix positiv definit ist, solange alle (hier $n = 2$) Pivotelemente strikt positiv sind. Mithilfe der LU-Zerlegung schliessen wir auf:

$$\mathbf{A} = \mathbf{LU} \implies U = \begin{bmatrix} 1 & 4 \\ 0 & c - 16 \end{bmatrix}.$$

Wir sehen nun, dass $c > 16$ sein muss, damit die Matrix positiv definit ist.
Was passiert, wenn $c = 16$? Die Matrix lautet dann:

$$\mathbf{A} = \begin{bmatrix} 1 & 4 \\ 4 & 16 \end{bmatrix}$$

Unsere Eigenwerte lauten dann $\lambda_1 = 0$, $\lambda_2 = 17$. Die Matrix $\mathbf{A}$ ist in diesem Fall nur symmetrisch positiv semi-definit.

⌟

**Bemerkung 7.4.1.2 (Quadratische Form als Summe von Quadraten für symmetrische $\mathbf{A}$)**
Wenn $\mathbf{A}$ symmetrisch ist, dann können wir die Cholesky-Zerlegung

$$\mathbf{A} = \mathbf{LDL}^\mathsf{T}$$

verwenden. Die quadratische Form können wir dann umschreiben zu:

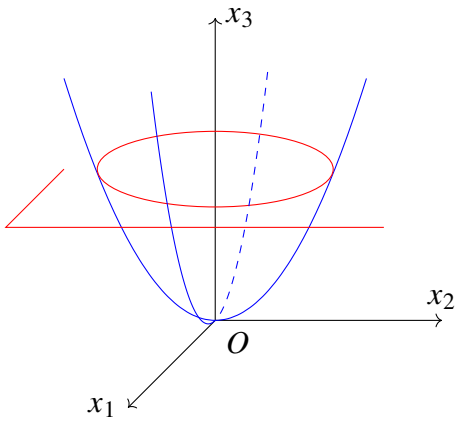
$$\begin{aligned} f(\mathbf{x}) = \mathbf{x}^\mathsf{T}\mathbf{A}\mathbf{x} = \mathbf{x}^\mathsf{T}\mathbf{L}\mathbf{D}\mathbf{L}^\mathsf{T}\mathbf{x} \quad & |\mathbf{y} = \mathbf{L}^\mathsf{T}\mathbf{x} \\ &= \sum_{k=1}^{n} d_k y_k^2 \\ &= \sum_{k=1}^{n} d_k (\mathbf{L}^\mathsf{T}\mathbf{x})_k^2 \,. \end{aligned}$$

Stützen wir uns auf den Spektralsatz, so können wir alternativ auch folgende Berechnung durchführen:

$$f(\mathbf{x}) = \mathbf{x}^\mathsf{T}\mathbf{A}\mathbf{x} = \mathbf{x}^\mathsf{T}\mathbf{S}\mathbf{\Lambda}\mathbf{S}^\mathsf{T}\mathbf{x} = \sum_{k=1}^{n} \lambda_k (\mathbf{S}^\mathsf{T}\mathbf{x})_k^2 \,.$$

Beide Formeln enthalten die quadratische Form als Summe von Quadraten mit positiven Vorfaktoren. Die zwei Formeln sind aber unterschiedlich: Die Pivotelemente sind nicht die Eigenwerte! Ausserdem ist $\mathbf{L}$ eine untere Dreiecksmatrix mit 1er auf der Hauptdiagonale, während $\mathbf{S}$ eine orthogonale Matrix ist.
Wenn nur die Vorzeichen der Eigenwerte einer quadratischen Form bestimmt werden sollen, ist die Cholesky-Zerlegung der effizienteste Lösungsansatz. Sollen aber die Eigenwerte (z. B. die Achsen der Ellipse) bestimmt werden, dann muss man die Diagonalisierung durchführen. ⌟

3. Der quadratischen Form können wir uns auch bedienen, um beispielsweise ein elliptisches Paraboloid in mehreren Dimensionen mit einer Hyperebene zu schneiden und die dadurch entstehende Ellipse darzustellen.




Als Beispiel betrachten wir das elliptische Paraboloid in der Dimension $n = 2$:

$$f(x_1, x_2) = ax_1^2 + 2bx_1x_2 + cx_2^2 = \mathbf{x}^\mathsf{T}\mathbf{A}\mathbf{x}\,,$$

Schneiden wir mit einer Ebene $x_3 = 1$, d. h. $\mathbf{x}^\mathsf{T}\mathbf{A}\mathbf{x} = 1$, so erhalten wir die Ellipse:

$$(\mathbf{x}^\mathsf{T}\mathbf{S})\mathbf{D}(\mathbf{S}^\mathsf{T}\mathbf{x}) = 1\,.$$

Man kann die orthogonale Transformation anwenden $\mathbf{y} = \mathbf{S}^\mathsf{T}\mathbf{x}$, um die Ellipsengleichung besser zu sehen.
Schauen wir uns ein Beispiel an:

**Beispiel 7.4.1.3 (Ellipse)**
Seien $a = 5, b = 8, c = 5$ gegeben[8]. Unsere Matrix sieht dann folgendermassen aus:

$$\mathbf{A} = \begin{bmatrix} 5 & 4 \\ 4 & 5 \end{bmatrix}.$$

Die Eigenwerte lauten dementsprechend $\lambda_1 = 9,\ \lambda_2 = 1$ mit ihren jeweiligen Eigenvektoren

$$\mathbf{v}_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \qquad \mathbf{v}_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

Da $\mathbf{A}$ symmetrisch ist, lässt sich diese leicht diagonalisieren:

$$\mathbf{A} = \mathbf{S}\mathbf{D}\mathbf{S}^\top = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} 9 & 0 \\ 0 & 1 \end{bmatrix}\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}.$$

Wir berechnen noch die neue Variable $\mathbf{y}$:

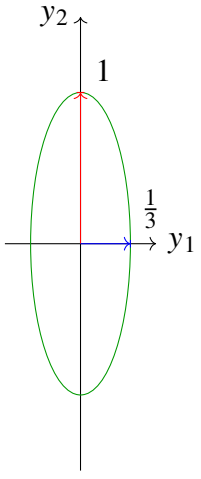
$$\mathbf{y} = \mathbf{S}^\top\mathbf{x} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}}(x_1 + x_2) \\ \frac{1}{\sqrt{2}}(x_1 - x_2) \end{bmatrix}.$$

Kehren wir zu unserer ursprünglichen Gleichung zurück, so erhalten wir:

$$1 = 5x_1^2 + 8x_1x_2 + 5x_2^2 = 9y_1^2 + y_2^2$$
$$\Rightarrow \quad \frac{y_1^2}{\left(\frac{1}{3}\right)^2} + \frac{y_2^2}{1} = 1\,.$$

Wir haben somit eine Ellipse im Koordinatensystem $(y_1, y_2)$. Diese Koordinatenachsen schneiden die Ellipse in den Punkten

$$\mathbf{y}_1 = \begin{bmatrix} \frac{1}{3} \\ 0 \end{bmatrix},\ \mathbf{y}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$



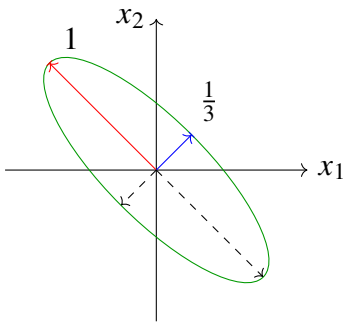

Möchten wir die Ellipse im Koordinatensystem $(x_1, x_2)$ darstellen, müssen wir dies umrechnen:

$$1 = 9y_1^2 + 1y_2^2 = 9(\frac{x_1 + x_2}{\sqrt{2}})^2 + (\frac{x_1 - x_2}{\sqrt{2}})^2$$

$$\mathbf{x} = \mathbf{S}\mathbf{y}$$

$$\Rightarrow \quad \mathbf{x}_1 = \mathbf{S}\mathbf{y}_1 = \frac{1}{3\sqrt{2}}\begin{bmatrix} 1 \\ 1 \end{bmatrix} \qquad \Rightarrow \quad \mathbf{x}_2 = \mathbf{S}\mathbf{y}_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ -1 \end{bmatrix}.$$

[8] Beispiel 2 in Kapitel 6.5, Seite 346 in Str09.




⌟

**Satz 7.4.1.4. Ellipse**

*Sei die Matrix* $\mathbf{A}$ *diagonalisierbar als* $\mathbf{A} = \mathbf{S}\mathbf{\Lambda}\mathbf{S}^\mathsf{T}$ *und s.p.d.*
*Dann beschreibt* $\mathbf{x}^\mathsf{T}\mathbf{A}\mathbf{x} = 1$ *eine Ellipse. In ausgeschriebener Form nimmt die Ellipse die Gestalt*

$$1 = [x_1 \;\; x_2]\mathbf{A}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \lambda_1 y_1^2 + \lambda_2 y_2^2$$

*an.*
*Die Längen der Halbachsen dieser Ellipse sind dementsprechend* $\frac{1}{\sqrt{\lambda_1}}, \frac{1}{\sqrt{\lambda_2}}$.

**Bemerkung 7.4.1.5 (Geometrie in $n$ Dimensionen)**
Ist die Matrix $\mathbf{A}$ in $\mathbb{R}^{n\times n}$ symmetrisch positiv definit, so beschreibt der Schnitt der quadratischen Form mit einer Hyperebene ein Ellipsoid.
Die Eigenschaften der Eigenwerte der Matrix, die mit der quadratischen Form assoziiert sind, bestimmen die Art der dargestellten Flächen und Kurven. In Dimension $n = 3$ gibt es entsprechend ein *Ellipsoid*, wenn alle Eigenwerte positiv sind, ein *Hyperboloid*, wenn nicht alle Eigenwerte dasselbe Vorzeichen haben. Gute Erklärungen solcher Anwendungen samt Bilder befinden sich auf Wikipedia bei Hauptachsentransformation und bei Quadrik.
Partielle Differentialgleichungen werden auch mit Hilfe von Eigenwerten einer Matrix klassifiziert. Elliptische Gleichungen (alle Eigenwerte haben dasselbe Vorzeichen) werden für Modelle im Äquilibrium benutzt. Hyperbolische Gleichungen (alle Eigenwerte haben dasselbe Zeichen und nur einer hat das andere Vorzeichen) werden als Modell für Wellenpropagation verwendet. Parabolische Gleichungen (alle Eigenwerte haben dasselbe Vorzeichen und ein Eigenwert ist Null) werden für die Modellierung von abklingenden Zuständen, wie z. B. bei der Wärmeleitungsgleichung, benutzt.

⌟

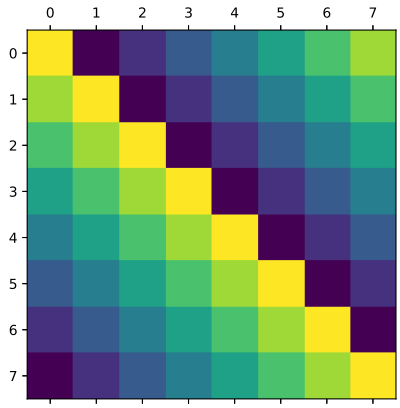
## 7.5 Die Fourier Matrix

**Definition 7.5.0.1. Zirkulante Matrix**

Eine Matrix mit folgender Struktur:

$$\mathbf{C} = \begin{bmatrix} c_0 & c_{N-1} & c_{N-2} & \cdots & c_3 & c_2 & c_1 \\ c_1 & c_0 & & & & c_3 & c_2 \\ c_2 & & \ddots & & & & c_3 \\ \vdots & & & & & & \vdots \\ c_{N-3} & & & & & & c_{N-2} \\ c_{N-2} & c_{N-3} & & & & c_0 & c_{N-1} \\ c_{N-1} & c_{N-2} & c_{N-3} & \cdots & c_2 & c_1 & c_0 \end{bmatrix}$$

heisst zirkulante Matrix (oder einfach nur ein Zirkulant).




Die erste Spalte $\mathbf{c} = [c_0\, c_1\, \ldots\, c_{n-1}]^T \in \mathbb{R}^n$ definiert eine zirkulante Matrix eindeutig. Für $\mathbf{c} = e_2 = [0\, 1\, \ldots\, 0]^T$ haben wir eine spezielle zirkulante Matrix, die Shift-Matrix mit $n$ Spalten:

$$\mathbf{S}_n = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \\ 1 & 0 & & & 0 \\ 0 & 1 & \ddots & & \vdots \\ \vdots & & \ddots & 0 & 0 \\ 0 & \cdots & 0 & 1 & 0 \end{bmatrix}.$$

Die Shift-Matrix $\mathbf{S}_n$ verschiebt die Einträge periodisch vorwärts und $\mathbf{S}_n^T$ verschiebt sie zurück:

$$\mathbf{S}_n \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{n-2} \\ x_{n-1} \end{bmatrix} = \begin{bmatrix} x_{n-1} \\ x_0 \\ x_1 \\ \vdots \\ x_{n-2} \end{bmatrix}, \qquad \mathbf{S}_n^T \begin{bmatrix} x_{n-1} \\ x_0 \\ x_1 \\ \vdots \\ x_{n-2} \end{bmatrix} = \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{n-2} \\ x_{n-1} \end{bmatrix},$$

d. h. $\mathbf{S}_n$ ist orthogonal, also diagonalisierbar mit unitären Transformationen. Gauss-Elimination liefert die

charakteristische Gleichung $\lambda^n = 1$, d. h. die $n$ Eigenwerte von $\mathbf{S}_n$ sind

$$\lambda_k = e^{2\pi i \frac{-k}{n}}\,,\ k = 0, 1, 2, \ldots, n-1\,.$$

Notieren wir $w_n = e^{2\pi i/n}$ und $\omega_n = e^{-2\pi i/n} = \overline{w_n}$, sodass $w_n^k = \omega_n^{-k} = \overline{\omega_n^k}$. Wir notieren die $n$ Fourier-Vektoren:

$$\mathbf{v}_k = \begin{bmatrix} w_n^{0\cdot k} \\ w_n^{1\cdot k} \\ \vdots \\ w_n^{(n-1)\cdot k} \end{bmatrix} = \begin{bmatrix} \lambda_k^0 \\ \lambda_k^{-1} \\ \vdots \\ \lambda_k^{-(n-1)} \end{bmatrix},\ k = 0, 1, 2, \ldots, n-1\,.$$

Für $j = 0, 1, \ldots, n-1$ schreiben wir $w_n^{(j-1)\cdot k} = w_n^{j\cdot k} w_n^{-k} = w_n^{j\cdot k}\lambda_k$, so dass der $k$-te Fourier-Vektor $\mathbf{v}_k$ ein Eigenvektor der Shift-Matrix $\mathbf{S}_n$ zum Eigenwert $\lambda_k$ ist. Somit bilden die Fourier-Vektoren $\mathbf{v}_0, \mathbf{v}_1, \ldots, \mathbf{v}_{n-1}$ eine orthogonale Basis in $\mathbb{C}^n$, die Fourier-Basis. Wenn wir diese Vektoren auch normieren $\mathbf{u}_k = \mathbf{v}_k/\sqrt{n}$, erhalten wir die unitäre Basis, die die Shift-Matrix diagonalisiert. Sammeln wir die Fourier-Vektoren als Spalten einer Matrix

$$\mathbf{V} = [\mathbf{v}_0\, \mathbf{v}_1\, \ldots\, \mathbf{v}_{n-1}]$$

und

$$\mathbf{U} = [\mathbf{u}_0\, \mathbf{u}_1\, \ldots\, \mathbf{u}_{n-1}]\,,$$

so haben wir:

$$\mathbf{V}^{\mathsf{H}}\mathbf{V} = n\mathbf{I}_n\,,\quad \mathbf{U} = \frac{1}{\sqrt{n}}\mathbf{V}\,,\quad \mathbf{U}^{\mathsf{H}}\mathbf{U} = \mathbf{I}_n$$

und die Diagonalisierung der Shift-Matrix $\mathbf{S}_n$ ist

$$\mathbf{S}_n = \mathbf{U}\mathbf{D}\mathbf{U}^{\mathsf{H}}\,,$$

wobei $\mathbf{D}$ die Diagonalmatrix mit den Eigenwerten $\lambda_k = w_n^{-k} = \omega_n^k$ ist.

Man kann für jede zirkulante Matrix $\mathbf{C}$ ein Polynom $p(z) = c_0 z^0 + c_1 z^1 + c_2 z^2 + \ldots + c_{n-1} z^{n-1}$ finden, sodass $\mathbf{C} = p(\mathbf{S}_n)$ gilt. Denn $\mathbf{S}_n^m$ für $m \in \{0, 1, \ldots, n-1\}$ entspricht einer, um $m$ Zeilen periodisch vorwärts verschobenen Einheitsmatrix.

$$\begin{aligned}
\mathbf{C} &= c_0\mathbf{S}_n^0 + c_1\mathbf{S}_n^1 + \cdots + c_{n-1}\mathbf{S}_n^{n-1} \\
&= c_0\begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 \end{bmatrix} + c_1\begin{bmatrix} 0 & \cdots & 0 & 1 \\ 1 & \ddots & & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 1 & 0 \end{bmatrix} + \cdots + c_{n-1}\begin{bmatrix} 0 & 1 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & & \ddots & 1 \\ 1 & 0 & \cdots & 0 \end{bmatrix} \\
&= \begin{bmatrix} c_0 & c_{n-1} & \cdots & c_1 \\ c_1 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & c_{n-1} \\ c_{n-1} & \cdots & c_1 & c_0 \end{bmatrix}
\end{aligned}$$

Die Diagonalisierung von $\mathbf{S}_n$ liefert also die Diagonalisierung jeder beliebiger zirkulanten Matrix:

$$\mathbf{C} = p(\mathbf{S}_n) = \mathbf{U}p(\mathbf{D})\mathbf{U}^{\mathsf{H}},$$

Somit haben wir bereits den folgenden Satz bewiesen.

**Satz 7.5.0.2. Gleichzeitige Diagonalisierung aller zirkulanten Matrizen**

*Die Elemente der Fourier-Basis sind die Eigenvektoren der Shift-Matrix $\mathbf{S}_n$ und sie diagonalisieren (sind Eigenvektoren von) jede(r) zirkulante(n) Matrix $\mathbf{C}$; die entsprechenden Eigenwerte sind in der Diagonalmatrix $p(\mathbf{D})$, die die Einträge $p(\lambda_k)$ hat.*

Die Fourier-Vektoren $\mathbf{v}_k$ haben wir in die Spalten der Matrix $\mathbf{V}$ geschrieben. Für die Standardbasisvektoren $\mathbf{e}_k$ haben wir:

$$\mathbf{v}_k = \mathbf{V}\mathbf{e}_{k+1}, \; k = 0, 1, \ldots, n-1$$

und ein allgemeines $\mathbf{y} \in \mathbb{C}^n$ lässt sich in der Standardbasis mit den Koeffizienten $y_k$ und in der Fourier-Basis mit den Koeffizienten $\hat{y}_k$ schreiben:

$$\mathbf{y} = \sum_{k=0}^{n-1} y_k \mathbf{e}_{k+1} = \sum_{k=0}^{n-1} \hat{y}_k \mathbf{v}_k = \sum_{k=0}^{n-1} \hat{y}_k \mathbf{V}\mathbf{e}_{k+1} = \mathbf{V}\hat{\mathbf{y}}.$$

Somit realisiert die Matrix $\mathbf{V}$ die Transformation/Koordinatenwechsel von der Fourier-Basis zu der Standardbasis:

$$\mathbf{y} = \mathbf{V}\hat{\mathbf{y}}, \tag{7.5.0.3}$$

und die Transformation/Koordinatenwechsel von der Standardbasis zur Fourier-Basis ist

$$\hat{\mathbf{y}} = \mathbf{V}^{-1}\mathbf{y} = \frac{1}{n}\mathbf{V}^{\mathsf{H}}\mathbf{y}. \tag{7.5.0.4}$$

Das begründet die Notation und die Definition als Fourier-Matrix:

**Definition 7.5.0.5. Fourier Matrix**

*a*

$$\mathbf{F} = \mathbf{V}^{\mathsf{H}} = \begin{bmatrix} \omega^{0\cdot 0} & \omega^{1\cdot 0} & \cdots & \omega^{(n-1)\cdot 0} \\ \vdots & \vdots & & \vdots \\ \omega^{0\cdot k} & \omega^{1\cdot k} & \cdots & \omega^{(n-1)\cdot k} \\ \vdots & \vdots & & \vdots \\ \omega^{0\cdot(n-1)} & \omega^{1\cdot(n-1)} & \cdots & \omega^{(n-1)\cdot(n-1)} \end{bmatrix}.$$

Das ist die Fourier-Matrix in der Konvention ‘‘negative Exponenten’’, denn $\omega_n = e^{-2\pi i/n} = \overline{w_n}$. Diese Konvention wird von den meisten Implementierungen verwendet (z. B. in `scipy.fft.fft`).

**Bemerkung 7.5.0.6 (Eigenschaften der Fourier-Matrix)** Einige wichtige Eigenschaften der Fourier-Matrix sind:

- $\frac{1}{\sqrt{n}}\mathbf{F}$ ist unitär
- $\mathbf{F}$ ist symmetrisch ($\mathbf{F}^T = \mathbf{F}$) aber nicht Hermite-symmetrisch ($\mathbf{F}^{\mathsf{H}} \neq \mathbf{F}$)
- $\overline{\mathbf{F}} = \mathbf{V}$
- $\mathbf{F} = \sqrt{n}\mathbf{U}^{\mathsf{H}}$

- $\mathbf{F}^{-1} = \mathbf{U}/\sqrt{n}$

⌟

**Definition 7.5.0.7. Diskrete Fourier Transformation**

Die Abbildung

$$\mathscr{F}_n : \mathbb{C}^n \to \mathbb{C}^n \,, \; \mathscr{F}_n(\mathbf{y}) = \mathbf{F}\mathbf{y}$$

heisst die diskrete Fourier Transformation (DFT).

**Bemerkung 7.5.0.8 (Eigenschaften der Einheitswurzel)**
Wir haben eine Periodizität in den Potenzen: $\omega_n^{k+n} = \omega_n^k, \omega_n^{t+kn} = \omega_n^t, \omega_n^n = 1, \omega_{2m}^m = -1$, für alle $k, n, m \in \mathbb{N}$, $t \in \mathbb{R}$. Ausserdem gilt:

$$\sum_{k=0}^{n-1} \omega_n^{kj} = \begin{cases} n\,, & j = 0 \\ \frac{1-\omega_n^n}{1-\omega_n} = 0\,, & j \neq 0 \end{cases}.$$

⌟

Verwenden wir die letzte Eigenschaft um die Einträge der Matrix $\mathbf{A} = \mathbf{F}^2 = \mathbf{F}\mathbf{F}$ zu berechnen:

$$a_{\ell j} = \sum_{k=0}^{n-1} \omega_n^{\ell k} \omega_n^{kj} = \sum_{k=0}^{n-1} \omega_n^{k(\ell+j)} = \begin{cases} n\,, & \ell + j = mn \\ 0\,, & \ell + j \neq mn \end{cases}.$$

Zusammen mit dem Normierungsfaktor erhalten wir:

$$\mathbf{U}^{\mathrm{H}}\mathbf{U}^{\mathrm{H}}\mathbf{y} = \frac{1}{n}\mathbf{F}\mathbf{F}\begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{n-2} \\ y_{n-1} \end{bmatrix} = \begin{bmatrix} y_0 \\ y_{n-1} \\ \vdots \\ y_2 \\ y_1 \end{bmatrix},$$

d. h. die zweifache Anwendung der normierten diskreten Fourier Transformation kehrt die Reihenfolge der Einträge in einem Vektor bis auf das erste Element um (Zeitumkehrung modulo $n$: $y_0$ wird zu $y_{n-0} = y_{-0}$, $y_1$ wird zu $y_{n-1}$,..., $y_{n-1}$ wird zu $y_{n-n+1}$, ).

Das können wir auch ein bisschen eleganter beweisen, wenn wir die Permutationsmatrix für die diskrete Zeitumkehrung

$$\mathbf{P}_n = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & 0 & \cdots & 1 & 0 \\ \vdots & \vdots & \vdots & \iddots & \vdots & \vdots \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 \end{bmatrix}.$$

notieren. Die Orthogonalität $\mathbf{V}^{\mathrm{H}}\mathbf{V} = n\mathbf{I}$ bedeutet $\mathbf{F}\overline{\mathbf{V}} = n\mathbf{I}$; da $\overline{\mathbf{F}} = \mathbf{V}$, gibt das $\mathbf{F}\overline{\mathbf{F}} = n\mathbf{I}$. Da $\overline{\mathbf{F}}_{jk} = \omega^{-jk} = \omega^{j(-k)} = \omega^{j(n-k)}$ ist, haben wir $\overline{\mathbf{F}} = \mathbf{F}\mathbf{P}$, denn die Multiplikation von Rechts mit der Permutationsmatrix vertauscht die Spalten von $\mathbf{F}$. Somit schreiben wir die Orthogonalität als $\mathbf{F}\mathbf{F}\mathbf{P} = n\mathbf{I}$. Da $\mathbf{P}$ eine Spiegelung ist, haben wir $\mathbf{P}^{-1} = \mathbf{P}$, also $\mathbf{F}^2 = n\mathbf{P}$.

So erhalten wir

**Bemerkung 7.5.0.9 (Eigenwerte der Fourier-Matrix)**

$$\left(\mathbf{U}^{\mathsf{H}}\right)^4 = \frac{1}{n^2}\mathbf{F}^4 = \mathbf{I}_n\,.$$

Die Eigenwerte der normierten diskrete Fourier Transformation sind $1, -1, i, -i$, die Eigenvektoren sind nicht eindeutig, die entsprechende Eigenräume haben verschiedene Dimensionen. ⌟

**Bemerkung 7.5.0.10 (Inverse DFT)** Die inverse Fourier Transformation kann man mit Tricks so schrieben, dass man nur die direkte Transformation implementieren muss:

$$\mathbf{F}^{-1}\begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{n-2} \\ x_{n-1} \end{bmatrix} = \frac{1}{n}\mathbf{F}\begin{bmatrix} x_0 \\ x_{n-1} \\ x_{n-2} \\ \vdots \\ x_1 \end{bmatrix}.$$

$$\mathbf{F}^{-1}\mathbf{x} = \frac{1}{n}\overline{\mathbf{F}\overline{\mathbf{x}}}\,,$$

$$\mathbf{F}^{-1}\mathbf{x} = \frac{1}{n}\operatorname{swap}(\mathbf{F}\operatorname{swap}(\mathbf{x}))\,,$$

wobei: $\texttt{swap}(a+bi) = b+ai$. ⌟

**Bemerkung 7.5.0.11 (FFT)** Die Fourier Transformation kann in $n\log(n)$ Operationen berechnet werden, was deutlich schneller als die erwarteten $n^2$ Operationen ist. Zusammen mit der Stabilität macht diese Geschwindigkeit die Fourier Transformation zu einer der wichtigsten Algorithmen unserer Zeit. Für $n = 2m$ kann man das leicht sehen, in dem wir die Koeffizienten mit geraden Indizes von denen mit ungeraden Indizes in der Summation trennen:

$$\begin{aligned}\hat{y}_k &= \sum_{j=0}^{n-1} y_j e^{-\frac{2\pi i}{n}jk} = \\ &= \sum_{j=0}^{m-1} y_{2j} e^{-\frac{2\pi i}{2m}2jk} + \sum_{j=0}^{m-1} y_{2j+1} e^{-\frac{2\pi i}{2m}(2j+1)k} \\ &= \sum_{j=0}^{m-1} y_{2j} e^{-\frac{2\pi i}{m}jk} + e^{-\frac{2\pi i}{2m}k}\sum_{j=0}^{m-1} y_{2j+1} e^{-\frac{2\pi i}{m}jk}\,.\end{aligned}$$

Wenn $n$ eine 2er-Potenz ist, kann man diese Rechnung rekursiv so weiterführen, dass ein perfekter ‘‘divide-and-conquer’’ Algorithmus entsteht, der nur $\frac{n}{2}\log_2(n) + n\log_2(n)$ komplexe Operationen (Additionen oder Multiplikationen) braucht. Für allgemeine $n$ haben Cooley und Tucker in 1965 auch einen schnellen Algorithmus, der $C \cdot n\log_2(n)$ Operationen braucht, dafür gefunden, was zum grossen Erfolg der FFT geführt hat. Der optimale Wert von $C$ ist im Allgemeinen unbekannt, Ergebnisse von 2007 schätzen ihn auf $34/9$, wenn es sich um reelle Operationen handelt. Man sollte FFT nicht selber implementieren, sondern die vorhandenen Bibliotheken (z. B. FFTW in C++) verwenden. Wenn auch die Laufzeit des Algorithmus auf modernen Computern von anderen Faktoren als der Anzahl der arithmetischen Operationen dominiert ist, bleibt der wesentliche Vorteil einer asymptotischen Laufzeit von $n\log_2(n)$ der FFT gegenüber $n^2$ der Standard Multiplikation mit der Fourier Matrix erhalten. ⌟

Seien $g(x) = \sum_{k=-\infty}^{\infty} a_k x^k$ und $h(x) = \sum_{k=-\infty}^{\infty} b_k x^k$. Das Produkt dieser beiden Funktionen kann man auch als Potenzreihe schreiben:

$$f(x) = g(x)h(x) = \sum_{k=-\infty}^{\infty} c_k x^k = \sum_{k=-\infty}^{\infty}\left(\sum_{n=-\infty}^{\infty} a_n b_{k-n}\right)x^k\,.$$

Wenn wir nur endliche Summen haben, z. B. wenn $g$ und $h$ Polynome vom Grad maximal $N-1$ sind, dann sind auch alle Summen im Ergebnis endlich:

$$f(x) = g(x)h(x) = \sum_{k=0}^{2N-2} c_k x^k = \sum_{k=0}^{2N-2} \left( \sum_{n=0}^{k} a_n b_{k-n} \right) x^k \,.$$

Das begründet die folgende Definition:

**Definition 7.5.0.12. Faltung**

Die Faltung zweier unendlicher Folgen $a = (a_k)_{k\in\mathbb{Z}}$ und $b = (b_k)_{k\in\mathbb{Z}}$ ist die unendliche Folge $c = (c_k)_{k\in\mathbb{Z}}$ mit $c_k = \sum_{n=-\infty}^{\infty} a_n b_{k-n}$. Man schreibt $c = a * b$.

Die Faltung von zweier Vektoren $\mathbf{a} = [a_0, \ldots, a_{N-1}]^\mathsf{T}$ und $\mathbf{b} = [b_0, \ldots, b_{N-1}]^\mathsf{T}$ erhält man aus dieser Definition, wenn man die Einträge von $\mathbf{a}$ und $\mathbf{b}$ als Folgeglieder auffasst und diese beidseitig mit Nullen zu unendlichen Folgen fortsetzt.

**Beispiel 7.5.0.13 (Multiplikation als Faltung von Ziffern)** [9] Ein einfaches und anschauliches Beispiel[10] der Faltung liefert die Multiplikation zweier ganzer Zahlen. Betrachten wir als Beispiel $421 \cdot 503$.

Um den Zusammenhang zwischen einer Multiplikation und einer Faltung zu zeigen, führen wir diese elementare Rechnung explizit durch:

$$\begin{aligned} 421 \cdot 503 &= \left(1 \cdot 10^0 + 2 \cdot 10^1 + 4 \cdot 10^2\right)\left(3 \cdot 10^0 + 0 \cdot 10^1 + 5 \cdot 10^2\right) \\ &= 3 \cdot 10^0 + 6 \cdot 10^1 + 17 \cdot 10^2 + 10 \cdot 10^3 + 20 \cdot 10^4 \\ &= 211763 \end{aligned}$$

Man kann diese Rechnung aber auch als Auswertung eines Produkts zweier Polynome an der Stelle 10 betrachten. Also als $p(10)$ mit

$$\begin{aligned} p : \mathbb{N} \to \mathbb{N} \quad p(x) &= \left(1 \cdot x^0 + 2 \cdot x^1 + 4 \cdot x^2\right)\left(3 \cdot x^0 + 0 \cdot x^1 + 5 \cdot x^2\right) \\ &= 3 \cdot x^0 + 6 \cdot x^1 + 17 \cdot x^2 + 10 \cdot x^3 + 20 \cdot x^4 \,. \end{aligned}$$

Definitionsgemäss entspricht das der Faltung von $[1,2,4]^\mathsf{T}$ mit $[3,0,5]^\mathsf{T}$.

$$[1,2,4]^\mathsf{T} * [3,0,5]^\mathsf{T} = [3,6,17,10,20]^\mathsf{T}$$

Der Ergebnisvektor enthält die Koeffizienten des Polynoms $p$ und entspricht zudem (bis auf einen Übertrag) bereits dem Ergebnis der Multiplikation. ⌟

Ein spezieller und wichtiger Fall entsteht, wenn $x^N = 1$, d. h. wenn $x = \omega_N$, denn dann ist $x^{N+j} = x^j$. In diesem Fall erhalten wir die zyklische Faltung.

**Definition 7.5.0.14. Zyklische Faltung von periodischen Folgen**

Die **zyklische Faltung** der periodischen Folgen $a = (a_k)_{k\in\mathbb{Z}}$ und $b = (b_k)_{k\in\mathbb{Z}}$ mit Periode $N$ ist gegeben durch:

$$c = a \circledast b \,, \text{ mit } c_k = \sum_{n=0}^{N-1} a_n b_{k-n} = \sum_{n=0}^{N-1} a_{k-n} b_n \,.$$

[9] Dieses Beispiel wird in Abschnitt 9.7.4 mit Python gelöst.

[10] Aus dem Buch «Wissenschaftliches Rechnen» von G. Strang veröffentlicht 2010 auf Seite 410

Analog wird auch die zyklische Faltung von Vektoren definiert.

**Definition 7.5.0.15. Zyklische Faltung von Vektoren**

Die **zyklische Faltung** der Vektoren $\mathbf{a} = [a_0, \ldots, a_{N-1}]^\mathsf{T}$ und $\mathbf{b} = [b_0, \ldots, b_{N-1}]^\mathsf{T}$ ist gegeben durch:

$$\mathbf{c} = \mathbf{a} \circledast \mathbf{b}\,, \text{ mit } c_k = \sum_{n=0}^{N-1} a_n b_{((k-n) \bmod N)} = \sum_{n=0}^{N-1} a_{((k-n) \bmod N)} b_n\,.$$

**Bemerkung 7.5.0.16 (Zyklische Faltung von Vektoren)** Die zyklische Faltung von Vektoren entspricht der Faltung von periodischen Folgen, wenn man die Einträge von $\mathbf{a}$ und $\mathbf{b}$ als Folgeglieder auffasst und diese periodisch zu unendlichen Folgen fortsetzt. ⌟

**Beispiel 7.5.0.17 (Zyklische Faltung)** Für das vorige Beispiel 7.5.0.13 kann man mit $x^3 = 1$ auch eine zyklische Faltung berechnen:

$$\begin{aligned}\left(1 \cdot x^0 + 2 \cdot x^1 + 4 \cdot x^2\right)\left(3 \cdot x^0 + 0 \cdot x^1 + 5 \cdot x^2\right) &= \\ (1 \cdot 3 + 2 \cdot 5 \cdot x^3 + 4 \cdot 0 \cdot x^3)x^0 + & \\ (1 \cdot 0 + 2 \cdot 3 + 4 \cdot 5 \cdot x^3)x^1 + & \\ (1 \cdot 5 + 2 \cdot 0 + 4 \cdot 3)x^2 &= \\ 13 \cdot x^0 + 26 \cdot x^1 + 17 \cdot x^2\,, &\end{aligned}$$

d. h. die zyklische Faltung in diesem Fall ist $[1, 2, 4]^\mathsf{T} \circledast [3, 0, 5]^\mathsf{T} = [13, 26, 17]^\mathsf{T}$. ⌟

**Bemerkung 7.5.0.18 (zyklische Faltung via Matrixmultiplikation)** Die zyklische Faltung von $\mathbf{a} = [a_0, a_1, \ldots, a_{N-1}]^T$ und $\mathbf{b} = [b_0, b_1, \ldots, b_{N-1}]^T$ lässt sich einfach mit zirkulanten Matrizen schreiben. Sei $\mathbf{A}$ der Zirkulant, der durch die ersten Spalte $\mathbf{a} = [a_0, a_1, \ldots, a_{N-1}]$ definiert ist. Dann ist die zyklische Faltung $\mathbf{c} = \mathbf{a} \circledast \mathbf{b}$ durch die Multiplikation mit dem Zirkulanten gegeben:

$$\mathbf{c} = \mathbf{A}\mathbf{b}\,.$$

⌟

**Bemerkung 7.5.0.19 (Faltungstheorem)** Da die Fourier-Basis jede zirkulante Matrix diagonalisiert, bedeutet die letzte Bemerkung, dass sich die Faltung $\mathbf{c} = \mathbf{a} \circledast \mathbf{b}$ als die folgende Matrix-Multiplikation schreiben lässt:

$$\mathbf{c} = \mathbf{A}\mathbf{b} = \mathbf{U}p(\mathbf{D})\mathbf{U}^\mathsf{H}\mathbf{b} = \frac{1}{\sqrt{N}}\mathbf{V}p(\mathbf{D})\frac{1}{\sqrt{N}}\mathbf{V}^\mathsf{H}\mathbf{b} = \mathbf{F}_N^{-1}p(\mathbf{D})\mathbf{F}_N\mathbf{b}\,,$$

wobei, wie oben, $p(x) = a_0 + a_1 x + \ldots + a_{N-1}x^{N-1}$, sodass $p(\mathbf{S}) = \mathbf{A}$ und $\mathbf{D}$ die Diagonalmatrix mit den Einträgen $\lambda_k = \omega^k$ ist. Das ergibt dann:

$$\mathbf{F}_N\mathbf{c} = \operatorname{diag}(p(\omega^0), p(\omega^1), \ldots, p(\omega^{N-1}))\mathbf{F}_N\mathbf{b} = \operatorname{diag}(\mathbf{V}_N^\mathsf{H}\mathbf{a})\mathbf{F}_N\mathbf{b} = \operatorname{diag}(\mathbf{F}_N\mathbf{a})\mathbf{F}_N\mathbf{b}\,. \tag{7.5.0.20}$$

d. h. die Fouriertransformierte der Faltung $\mathbf{F}_N\mathbf{c}$ ist die punktweise Multiplikation der Fouriertransformierten $\mathbf{F}_N\mathbf{a}$ und $\mathbf{F}_N\mathbf{b}$. Das ist ein sehr wichtiges Ergebnis für die Signalverarbeitung und trägt den Namen «Faltungstheorem». ⌟

Wir sahen, dass die Fourier-Vektoren die Eigenvektoren allen zirkulanten Matrizen sind, d. h. alle zirkulanten Matrizen sind gleichzeitig diagonalisierbar, d. h. mit der selben Matrix $\mathbf{S} = \mathbf{V}$. Es stellt sich natürlich die interessante Frage, ob es auch andere Klassen von Matrizen gibt, die gleichzeitig diagonalisierbar sind.

**Satz 7.5.0.21. Gleichzeitig diagonalisierbare Matrizen**

*Seien* $\mathbf{A}$ *und* $\mathbf{B}$ *diagonalisierbare* $n \times n$ *Matrizen. Dann:*
*(a) Wenn* $\mathbf{A}$ *und* $\mathbf{B}$ *gleichzeitig diagonalisierbar sind, dann kommutieren sie:* $\mathbf{AB} = \mathbf{BA}$.
*(b) Wenn* $\mathbf{AB} = \mathbf{BA}$, *dann sind* $\mathbf{A}$ *und* $\mathbf{B}$ *gleichzeitig diagonalisierbar.*

*Beweis.*

(a) Seien $\mathbf{A}$ und $\mathbf{B}$ gleichzeitig diagonalisierbar, d. h. es gibt $\mathbf{S}$ invertierbar so dass:

$$\mathbf{A} = \mathbf{S}\mathbf{D}_1\mathbf{S}^{-1}\,, \quad \mathbf{B} = \mathbf{S}\mathbf{D}_2\mathbf{S}^{-1}\,.$$

Dann:

$$\mathbf{AB} = \mathbf{S}\mathbf{D}_1\mathbf{S}^{-1}\mathbf{S}\mathbf{D}_2\mathbf{S}^{-1} = \mathbf{S}\mathbf{D}_1\mathbf{D}_2\mathbf{S}^{-1} = \mathbf{S}\mathbf{D}_2\mathbf{D}_1\mathbf{S}^{-1} = \mathbf{S}\mathbf{D}_2\mathbf{S}^{-1}\mathbf{S}\mathbf{D}_1\mathbf{S}^{-1} = \mathbf{BA}\,.$$

(b) Falls alle Eigenwerte von $\mathbf{A}$ und $\mathbf{B}$ die algebraische Multiplizität 1 haben, dann ist der Beweis einfach. Sei $\lambda$ ein Eigenwert und $\mathbf{v}$ ein entsprechender Eigenvektor von $\mathbf{A}$:

$$(\mathbf{A} - \lambda\mathbf{I})\mathbf{v} = 0\,.$$

Wir multiplizieren von links mit $\mathbf{B}$ und verwenden die Kommutativität:

$$(\mathbf{AB} - \lambda\mathbf{B})\mathbf{v} = 0\,,$$

so ist $\mathbf{Bv}$ auch ein Eigenvektor von $\mathbf{A}$ zum Eigenwert $\lambda$. Wenn alle Eigenwerte von $\mathbf{A}$ Multiplizität 1 haben, dann hat $\mathrm{Ker}(\mathbf{A} - \lambda\mathbf{I})$ die Dimension 1. Also gibt es ein Skalar $\mu$, so dass $\mathbf{Bv} = \mu\mathbf{v}$, d. h. $\mathbf{v}$ ist auch ein Eigenvektor von $\mathbf{B}$ (zum Eigenwert $\mu$). Wenn alle Eigenwerte von $\mathbf{A}$ Multiplizität 1 haben, dann gibt es $n$ linear unabhängige Eigenvektoren von $\mathbf{A}$, die auch Eigenvektoren von $\mathbf{B}$ sind, also sind $\mathbf{A}$ und $\mathbf{B}$ gleichzeitig diagonalisierbar.
Der allgemeinere Fall diagonalisierbarer kommutierender Matrizen ist etwas komplizierter. Analog zum vorigen Fall gilt, dass jeder Eigenraum von $\mathbf{A}$ invariant für $\mathbf{B}$ ist, so dass mit einer nicht einfachen Argumentation eine Basis von Eigenvektoren von $\mathbf{A}$ und $\mathbf{B}$ gebaut werden kann.

□

**Bemerkung 7.5.0.22 (Die Bedingung, dass beide Matrizen diagonalisierbar sind, ist notwendig.)** Die Matrizen

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 3 & 1 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

kommutieren. $\mathbf{B}$ ist aber gar nicht diagonalisierbar, denn der Eigenwert 3 hat die algebraische Multiplizität 2 und geometrische Multiplizität 1. Somit wird der entsprechende Eigenraum von $\mathbf{e}_1$ alleine aufgespannt. Der Eigenraum zum Eigenwert 1 von $\mathbf{B}$ ist $\mathbf{e}_3$.
$\mathbf{e}_2$ ist ein Eigenvektor von $\mathbf{A}$, aber $\mathbf{Be}_2 = [1\,3\,0]^T$ ist kein skalares Vielfaches von $\mathbf{e}_2$, also kann $\mathbf{Be}_2$ kein Eigenvektor von $\mathbf{B}$ sein. Diese zwei Matrizen sind nicht gleichzeitig diagonalisierbar, obwohl sie kommutieren.

⌟

## 7.6 Nicht-diagonalisierbare Matrizen

In den vorangegangenen Kapiteln haben wir uns angeschaut, wie wir Matrizen diagonalisieren, Eigenwerte und Eigenvektoren berechnen und sogar Ellipsen darstellen. Dies funktionierte alles gut, solange die Matrizen diagonalisierbar waren. Nun möchten wir uns anschauen, was wir tun können, falls eine Matrix nicht diagonalisierbar ist. Dazu beginnen wir mit einem Beispiel:

**Beispiel 7.6.0.1 (Die ideale Dämpfung eines harmonischen Oszillators braucht eine nicht-diagonalisierbare Matrix)**

$$x : [0,T] \longrightarrow \mathbb{R}, \quad \dot{x} = \frac{\mathrm{d}x}{\mathrm{d}t}, \quad \ddot{x} = \frac{\mathrm{d}^2x}{\mathrm{d}t^2}$$

Die Gleichung für einen gedämpften harmonischen Oszillator ist:

$$\ddot{x} = -x - 2a\dot{x}\,.$$

Der Dämpfungsfaktor $2a$ wird in der Physik üblicherweise mit $\gamma$ bezeichnet. Wir möchten nun die Konstante $a$ so bestimmen, dass die Oszillation schnell abklingt. Führen wir nun die Substitution $y = \dot{x}$ durch, so erhalten wir:

$$\begin{aligned} y : [0,T] \longrightarrow \mathbb{R}, \quad & \dot{y} = \ddot{x} \\ \Rightarrow \quad & \dot{x} = y \\ & \dot{y} = -x - 2ay\,. \end{aligned}$$

Wir können nun das System auch in Matrixschreibweise darstellen:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & -2a \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}.$$

Wir bemerken, dass die Systemmatrix $\mathbf{A}$ zeitunabhängig ist.
Die Eigenwerte und die entsprechenden Eigenvektoren sind schnell berechnet:

$$\lambda_\pm = -a \pm \sqrt{a^2 - 1}, \quad \mathbf{v}_\pm = \begin{bmatrix} 1 \\ \lambda_\pm \end{bmatrix}.$$

Für die Eigenwerte können wir nun folgende Fälle feststellen:

- $a > 1$: reelle Eigenwerte, die Lösung $x$ nimmt ohne Schwingung mit der Zeit exponentiell ab. Dieser Fall entspricht der überkritischen Dämpfung also dem sogenannten Kriechfall.
- $0 \leq a < 1$: komplex-konjugierte Eigenwerte der Form

$$-a \pm i\sqrt{1 - a^2} = -a \pm iz(a)\,.$$

Als Lösung erhalten wir dann:

$$x(t) = \alpha_1 e^{-at} e^{-iz(a)t} + \alpha_2 e^{-at} e^{iz(a)t}\,.$$

Die Dämpfungen werden durch den Realteil, die Schwingungen durch den Imaginärteil bestimmt. Wir können daher sagen, dass für $a$ möglichst nahe bei $1$ die Oszillation am schnellsten abklingt.
Wenden wir uns nun der Berechnung der Eigenvektoren zu, so stellen wir folgendes fest:

$$\begin{aligned} & a = 1 \;\Rightarrow\; \lambda = -1 \\ & \begin{bmatrix} 0 & -1 \\ -1 & -2 \end{bmatrix} \mathbf{x} = \mathbf{0} \;\Rightarrow\; \text{Eigenvektor} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}. \end{aligned}$$

Der Eigenwert für $a = 1$ ist $\lambda = -1$, die algebraische Multiplizität ist 2 und die geometrische Multiplizität ist 1. Die Lösung der Differentialgleichung ist in diesem Fall

$$x(t) = \alpha_1 e^{-t} + \alpha_2 t e^{-t} .$$

Für diese Aufgabe ist dies genau der Fall, wenn die Matrix nicht diagonalisierbar ist. Der physikalisch interessantere Fall!

Welche Techniken können wir auf nicht-diagonalisierbare Matrizen anwenden?

⌟

Um trotzdem mit nicht-diagonalisierbaren Matrizen rechnen zu können, bedienen wir uns verschiedener Zerlegungen. Wir werden im Rahmen dieser Vorlesung zwei hervorheben.

### 7.6.1 Schur-Zerlegung

**Satz 7.6.1.1. Schur-Zerlegung**

1. *Die »komplexe« Schur-Zerlegung.*
   *Sei eine Matrix $\mathbf{A} \in \mathbb{C}^{n\times n}$ gegeben. Es existiert eine unitäre Matrix $\mathbf{U} \in \mathbb{C}^{n\times n}$, sodass*

   $$\mathbf{A} = \mathbf{U}\mathbf{T}\mathbf{U}^{\mathsf{H}}$$

   *erfüllt ist (mit $\mathbf{T}$ als obere Dreiecksmatrix). So gilt auch*

   $$\mathbf{U}^{\mathsf{H}}\mathbf{A}\mathbf{U} = \mathbf{T} .$$

   *Die Einträge auf der Hauptdiagonale von $\mathbf{T}$ sind die Eigenwerte von $\mathbf{A}$.*
2. *Sei $\mathbf{A} \in \mathbb{R}^{n\times n}$ beliebig. Es gibt eine orthogonale Matrix $\mathbf{U} \in \mathbb{R}^{n\times n}$, sodass*

   $$\mathbf{U}^{\mathsf{T}}\mathbf{A}\mathbf{U} = \begin{bmatrix} \mathbf{R}_{11} & \mathbf{R}_{12} & \cdots & \mathbf{R}_{1m} \\ 0 & \mathbf{R}_{22} & \cdots & \mathbf{R}_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{R}_{mm} \end{bmatrix} \in \mathbb{R}^{n\times n}$$

   *eine obere Blockdreiecksmatrix ist, mit $\mathbf{R}_{ii}$ als eine $1\times 1$ bzw. $2\times 2$-Matrix.*
   *Wenn $\mathbf{R}_{jj} = \lambda_j$ ist, dann ist $\lambda_j$ ein Eigenwert von $\mathbf{A}$. Wenn $\mathbf{R}_{jj}$ eine $2\times 2$-Matrix ist, dann sind ihre Eigenwerte $\lambda_j$, $\overline{\lambda_j}$ Eigenwerte von $\mathbf{A}$.*

Wir schauen uns noch einige kurze Zahlenbeispiele an:

**Beispiel 7.6.1.2 (Obere Dreiecksmatrix)**
Sei $\mathbf{T} \in \mathbb{R}^{3\times 3}$ gegeben als:

$$\mathbf{T} = \begin{bmatrix} 5 & 1 & 2 \\ 0 & 5 & 4 \\ 0 & 0 & 7 \end{bmatrix} .$$

Die Eigenwerte lauten $\lambda_1 = 5$, $\lambda_2 = 7$, die Eigenvektoren sind dementsprechend $\mathbf{v}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$, $\mathbf{v}_2 = \begin{bmatrix} 2 \\ 2 \\ 1 \end{bmatrix}$. Damit ist für den Eigenwert $\lambda_1 = 5$ die geometrische Multiplizität (1) strikt kleiner als die algebraische Multiplizität (2) und $\mathbf{T}$ gemäss 7.2.0.32 nicht diagonalisierbar.

⌟

**Beispiel 7.6.1.3 (Schur-Form)**
Sei $\mathbf{T} \in \mathbb{R}^{3\times 3}$ gegeben als:

$$\mathbf{T} = \left[\begin{array}{cc|c} 0 & 1 & 0 \\ -1 & 0 & 0 \\ \hline 0 & 0 & 7 \end{array}\right] = \begin{bmatrix} \mathbf{R}_{11} & \mathbf{R}_{12} \\ \mathbf{0} & \mathbf{R}_{22} \end{bmatrix}$$

Durch die Blockgruppierungen $\mathbf{R}_{11} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$, $\mathbf{R}_{12} = \mathbf{0}$, $\mathbf{R}_{22} = 7$ haben wir die ursprüngliche Matrix in Teilrechnungen unterteilt. Aus $\mathbf{R}_{11}, \mathbf{R}_{22}$ erhalten wir die Eigenwerte $\lambda_1 = 7, \lambda_2 = i, \lambda_3 = -i$.

⌟

**Bemerkung 7.6.1.4 (Weiteres zur Schur-Zerlegung)**

1. Die Schur-Zerlegung ist nicht eindeutig.
2. Die Schur-Zerlegung kann numerisch stabil berechnet werden. Diese Zerlegung wird in der Praxis verwendet, um Eigenwerte und Eigenvektoren numerisch zu berechnen.
3. Für die Determinantenberechnung gilt:

$$\det\left(\begin{bmatrix} t_{11} & & * \\ & t_{22} & \\ & & \ddots \\ 0 & & t_{nn} \end{bmatrix} - \lambda \mathbf{I}\right) = \det\begin{bmatrix} t_{11}-\lambda & & * \\ & t_{22}-\lambda & \\ & & \ddots \\ 0 & & t_{nn}-\lambda \end{bmatrix} = (t_{11}-\lambda)(t_{22}-\lambda)\dots(t_{nn}-\lambda)\,.$$

⌟

*Beweis. (Schur-Zerlegung)*
Wir verwenden mathematische Induktion.
**Induktionsverankerung:** Wir verankern die Aussage, indem wir sie für $n = 1$ beweisen. Dass die Aussage für $n = 1$ gilt, ist klar ersichtlich.

**Induktionsschritt:** Nun nehmen wir an, dass die Aussage auch für die Dimension $n-1$ stimmt. Sei nun eine $n \times n$ Matrix $\mathbf{A}$ beliebig und sei $\mathbf{q}_1$ ein normierter Eigenvektor von $\mathbf{A}$ zum Eigenwert $t_{11}$:

$$\mathbf{A}\mathbf{q}_1 = t_{11}\mathbf{q}_1\,,$$

mit $\|\mathbf{q}_1\|_2 = 1$. Unser Ziel ist es nun, zu zeigen, dass wir eine unitäre Matrix $\mathbf{U}$ finden können, so dass:

$$\mathbf{A}\mathbf{U} = \mathbf{U}\mathbf{T}.$$

Dazu vervollständigen wir $\mathbf{q}_1$ zu einer Orthonormalbasis in $\mathbb{C}^n$:

$$\mathbf{Q}_1 = \begin{bmatrix} | & | & & | \\ \mathbf{q}_1 & \mathbf{q}_2 & \cdots & \mathbf{q}_n \\ | & | & & | \end{bmatrix}, \quad \mathbf{Q}_1^{\mathsf{H}}\mathbf{Q}_1 = \mathbf{I}\,.$$

Somit können wir schreiben:

$$\mathbf{Q}_1^{\mathsf{H}}\mathbf{A}\mathbf{Q}_1 = \begin{bmatrix} — & \mathbf{q}_1^{\mathsf{H}} & — \\ & \vdots & \\ — & \mathbf{q}_n^{\mathsf{H}} & — \end{bmatrix}\begin{bmatrix} | & | & & | \\ \mathbf{A}\mathbf{q}_1 & \mathbf{A}\mathbf{q}_2 & \cdots & \mathbf{A}\mathbf{q}_n \\ | & | & & | \end{bmatrix} = \left[\begin{array}{c|c} \mathbf{q}_1^{\mathsf{H}}\mathbf{A}\mathbf{q}_1 & \mathbf{q}_1^{\mathsf{H}}\mathbf{A}\mathbf{q}_2 \cdots \mathbf{q}_1^{\mathsf{H}}\mathbf{A}\mathbf{q}_n \\ \hline \begin{matrix} 0 \\ \vdots \\ 0 \end{matrix} & \mathbf{A}_2 \end{array}\right]_{n-1}^{1}\,.$$

Wieso ist die erste Spalte so einfach? Jede Zeile von $\mathbf{Q}^H$ trifft $\mathbf{A}\mathbf{q}_1 = t_{11}\mathbf{q}_1$, da $\mathbf{q}_1$ Eigenvektor von $\mathbf{A}$ zum Eigenwert $t_{11}$ ist. Da die Basis orthonormal ist, verschwinden alle Einträge in der ersten Spalte, ausser dem ersten Eintrag, der genau der Eigenwert $t_{11}$ ist:

$$\begin{aligned}
\mathbf{q}_1^H(\mathbf{A}\mathbf{q}_1) &= \mathbf{q}_1^H(t_{11}\mathbf{q}_1) = t_{11}\mathbf{q}_1^H\mathbf{q}_1 = t_{11}\cdot 1 = t_{11}\\
\mathbf{q}_2^H(\mathbf{A}\mathbf{q}_1) &= \mathbf{q}_2^H(t_{11}\mathbf{q}_1) = t_{11}\mathbf{q}_2^H\mathbf{q}_1 = t_{11}\cdot 0 = 0\\
&\vdots\\
\mathbf{q}_n^H(\mathbf{A}\mathbf{q}_1) &= \mathbf{q}_n^H(t_{11}\mathbf{q}_1) = t_{11}\mathbf{q}_n^H\mathbf{q}_1 = t_{11}\cdot 0 = 0\,.
\end{aligned}$$

Die anderen Elemente sind nicht so einfach darzustellen, denn sie haben keinen Eigenvektor mehr rechts von $\mathbf{A}$. Etwas ganz besonderes passiert aber im Fall $\mathbf{A}^H = \mathbf{A}$:

$$(\mathbf{q}_1^H\mathbf{A}\mathbf{q}_2)^H = \mathbf{q}_2^H\mathbf{A}^H\mathbf{q}_1 = \mathbf{q}_2^H\mathbf{A}\mathbf{q}_1 = \mathbf{q}_2^H t_{11}\mathbf{q}_1 = 0\,.$$

Das bedeutet, im Falle einer Hermite-symmetrischen Matrix erhalten wir auch eine erste Zeile, die überall Nullen enthält, ausser im ersten Element.

Unsere Induktionsannahme lautet:

$$\mathbf{A}_2 = \widehat{\mathbf{U}}\widehat{\mathbf{T}}\widehat{\mathbf{U}}^H\,.$$

Wir schreiben:

$$\begin{aligned}
\mathbf{Q}_1^H\mathbf{A}\mathbf{Q}_1 &= \left[\begin{array}{c|c} t_{11} & * \\ \hline 0 & \\ \vdots & \widehat{\mathbf{U}}\widehat{\mathbf{T}}\widehat{\mathbf{U}}^H \\ 0 & \end{array}\right]\\
&= \left[\begin{array}{c|c} 1 & 0\cdots\cdots 0 \\ \hline 0 & \\ \vdots & \widehat{\mathbf{U}} \\ 0 & \end{array}\right]\left[\begin{array}{c|c} t_{11} & * \\ \hline 0 & \\ \vdots & \widehat{\mathbf{T}} \\ 0 & \end{array}\right]\left[\begin{array}{c|c} 1 & 0\cdots\cdots 0 \\ \hline 0 & \\ \vdots & \widehat{\mathbf{U}}^H \\ 0 & \end{array}\right]
\end{aligned}$$

und somit:

$$\begin{aligned}
\mathbf{A} &= \mathbf{Q}_1\begin{bmatrix} 1 & \mathbf{0} \\ \mathbf{0} & \widehat{\mathbf{U}} \end{bmatrix}\begin{bmatrix} t_{11} & * \\ \mathbf{0} & \widehat{\mathbf{T}} \end{bmatrix}\begin{bmatrix} 1 & \mathbf{0} \\ \mathbf{0} & \widehat{\mathbf{U}}^H \end{bmatrix}\mathbf{Q}_1^H\\
&= \mathbf{U}\mathbf{T}\mathbf{U}^H\,.
\end{aligned}$$

□

**Bemerkung 7.6.1.5 (Schur-Zerlegung einer Hermite-symmetrischen Matrix)**
Falls die Matrix Hermite-symmetrisch ($\mathbf{A}^H = \mathbf{A}$) ist, dann ist die Matrix $\mathbf{T}$ aus ihrer Schur-Zerlegung eine Diagonalmatrix.

⌟

**Satz 7.6.1.6. Normale obere Dreiecksmatrizen sind Diagonalmatrizen**

*Ist die Matrix $\mathbf{T}$ normal und eine obere Dreiecksmatrix, so ist diese eine Diagonalmatrix.*

*Beweis.* Sei die obere Dreiecksmatrix

$$\mathbf{T} = \begin{bmatrix} t_{11} & t_{12} & \cdots & t_{1n} \\ 0 & t_{22} & \cdots & t_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & t_{nn} \end{bmatrix}, \quad \mathbf{T}^H = \begin{bmatrix} \overline{t_{11}} & 0 & \cdots & 0 \\ \overline{t_{12}} & \overline{t_{22}} & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ \overline{t_{1n}} & \overline{t_{2n}} & \cdots & \overline{t_{nn}} \end{bmatrix}.$$

$\mathbf{T}$ ist normal, wenn $\mathbf{T}^{\mathsf{H}}\mathbf{T} = \mathbf{T}\mathbf{T}^{\mathsf{H}}$. Wir berechnen:

$$\mathbf{T}^{\mathsf{H}}\mathbf{T} = \begin{bmatrix} |t_{11}|^2 & \overline{t_{11}}t_{12} & \cdots & \overline{t_{11}}t_{1n} \\ \overline{t_{12}}t_{11} & |t_{12}|^2 + |t_{22}|^2 & \cdots & \overline{t_{12}}t_{1n} + \overline{t_{22}}t_{2n} \\ \vdots & \vdots & & \vdots \end{bmatrix}$$

$$\mathbf{T}\mathbf{T}^{\mathsf{H}} = \begin{bmatrix} \sum_{j=1}^{n} |t_{1j}|^2 & 0 & \cdots & 0 \\ 0 & \sum_{j=2}^{n} |t_{2j}|^2 & & \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & |t_{nn}|^2 \end{bmatrix}.$$

Wir vergleichen die Einträge der zwei Ergebnisse. Dabei sehen wir, dass viele Terme in den Summen Null sind. Das $(1,1)$-Element im Ergebnis ergibt:

$$|t_{11}|^2 = \sum_{j=1}^{n} |t_{1j}|^2$$

$$|t_{11}|^2 = |t_{11}|^2 + \sum_{j=2}^{n} |t_{1j}|^2 \quad \Leftrightarrow \quad 0 = \sum_{j=2}^{n} |t_{1j}|^2 \quad \Leftrightarrow \quad t_{1j} = 0 \text{ für } j = 2, 3, \ldots, n\,,$$

d. h. die erste Zeile (und die erste Spalte) des Ergebnisses enthält nur Nullen ausser im ersten Eintrag. Das $(2,2)$-Element im Ergebnis ergibt:

$$|t_{22}|^2 = \sum_{j=2}^{n} |t_{2j}|^2$$

$$|t_{22}|^2 = |t_{22}|^2 + \sum_{j=3}^{n} |t_{2j}|^2 \quad \Leftrightarrow \quad 0 = \sum_{j=3}^{n} |t_{2j}|^2 \quad \Leftrightarrow \quad t_{2j} = 0 (j = 3, 4, \ldots, n)\,,$$

d. h. die zweite Zeile (und die zweite Spalte) des Ergebnisses enthält nur Nullen ausser im zweiten Eintrag. Diese Prozedur wiederholt man für jedes Diagonalelement im Ergebnis und erhält damit, dass alle $t_{ij} = 0$ mit $j > i$ sind. Die Matrix $\mathbf{T}$ ist somit eine Diagonalmatrix.

□

**Satz 7.6.1.7.**
**Normale Matrizen sind die durch unitäre Transformationen diagonalisierbaren Matrizen**

*$\mathbf{A}$ hat $n$ orthogonale Eigenvektoren, dann und nur dann, wenn $\mathbf{A}$ normal ist.*

*Beweis.* Sei die Schur-Zerlegung

$$\mathbf{A} = \mathbf{Q}\mathbf{T}\mathbf{Q}^{\mathsf{H}}$$

mit $\mathbf{Q}$ unitär und $\mathbf{T}$ eine obere Dreiecksmatrix. Das ergibt

$$\begin{aligned}\mathbf{A}^H\mathbf{A} = \mathbf{A}\mathbf{A}^H &\Leftrightarrow \mathbf{Q}\mathbf{T}^H\mathbf{Q}^H \cdot \mathbf{Q}\mathbf{T}\mathbf{Q}^H = \mathbf{Q}\mathbf{T}\mathbf{Q}^H \cdot \mathbf{Q}\mathbf{T}^H\mathbf{Q}^H \\ &\Leftrightarrow \mathbf{Q}\mathbf{T}^H\mathbf{T}\mathbf{Q}^H = \mathbf{Q}\mathbf{T}\mathbf{T}^H\mathbf{Q}^H \\ &\Leftrightarrow \mathbf{T}^H\mathbf{T} = \mathbf{T}\mathbf{T}^H .\end{aligned}$$

Somit ist $\mathbf{A}$ normal, dann und nur dann, wenn $\mathbf{T}$ normal ist. Wir haben gerade gesehen, dass normale obere Dreiecksmatrizen Diagonalmatrizen sind. Somit ist $\mathbf{A}$ mit unitären Transformationen diagonalisierbar, dann und nur dann, wenn die Matrix $\mathbf{A}$ normal ist. Die Diagonalisierung ist in diesem Fall die Schur-Zerlegung. □

### 7.6.2 Jordan-Form

Die Jordan-Form ist nur als Theoriewerkzeug gedacht und sollte **nie** für numerische Berechnungen verwendet werden!

**Definition 7.6.2.1. Jordanblock**

Seien $n \in \mathbb{N}$ und $\lambda \in \mathbb{R}/\mathbb{C}$. Ein *Jordanblock* ist eine $n \times n$ *bidiagonale Matrix*

$$\mathbf{J}_{\lambda,n} = \begin{bmatrix} \lambda & 1 & & 0 \\ & \lambda & \ddots & \\ & & \ddots & 1 \\ 0 & & & \lambda \end{bmatrix} .$$

**Definition 7.6.2.2. Jordan-Matrix**

Eine *Jordan-Matrix* ist eine blockdiagonale Matrix mit Jordanblöcken auf der Hauptdiagonale:

$$\mathbf{J} = \operatorname{diag}(\mathbf{J}_{\lambda_1,n_1}, \mathbf{J}_{\lambda_2,n_2}, \ldots, \mathbf{J}_{\lambda_k,n_k}) = \begin{bmatrix} \mathbf{J}_{\lambda_1,n_1} & & & 0 \\ & \mathbf{J}_{\lambda_2,n_2} & \ddots & \\ & & \ddots & \\ 0 & & & \mathbf{J}_{\lambda_k,n_k} \end{bmatrix} .$$

**Satz 7.6.2.3. Eigenschaften einer Jordan-Matrix**

*$\mathbf{J}_{\lambda,n}$ hat einen einzigen Eigenwert $\lambda$ mit algebraischer Multiplizität $n$ und geometrischer Multiplizität 1. Der Eigenvektor ist das erste Element der Standardbasis ($\mathbf{e}_1$).*
*Die Jordan-Matrix mit den Blöcken $\mathbf{J}_{\lambda_1,n_1}, \mathbf{J}_{\lambda_2,n_2}, \ldots, \mathbf{J}_{\lambda_k,n_k}$ hat die Eigenwerte $\lambda_1, \lambda_2, \ldots, \lambda_k$ mit algebraischen Multiplizitäten $n_1, n_2, \ldots, n_k$ und Eigenvektoren $\mathbf{e}_1, \mathbf{e}_{n_1+1}, \ldots, \mathbf{e}_{n_1+n_2+\ldots+n_k+1}$.*

Im Folgenden sollen einige Beispiele angeschaut werden:

**Beispiel 7.6.2.4 (Jordan-Matrix)**

$$\left[\begin{array}{cccc|cc} -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 \end{array}\right] \quad \Rightarrow \quad \text{EW } = -1 \quad (\text{AM } 6,\ \text{GM } 2) \text{ EV } \mathbf{e}_1, \mathbf{e}_5 .$$

⌟

**Beispiel 7.6.2.5 (Jordan-Matrix)**

$$\left[\begin{array}{cc|cc|cc} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 2 \end{array}\right] \quad \Rightarrow \quad \text{EW } 0,1,2 \text{ (jeweils AM 2, GM 1) EV } \mathbf{e}_1, \mathbf{e}_3, \mathbf{e}_5 \,.$$

⌟

**Satz 7.6.2.6. Jordanzerlegung**

*Sei* $\mathbf{A}$ *eine* $n \times n$*-Matrix. Es gibt eine invertierbare Matrix* $\mathbf{S}$*, so dass*

$$\mathbf{S}^{-1}\mathbf{A}\mathbf{S} = \mathbf{J} = diag(\mathbf{J}_{\lambda_1,n_1}, \ldots, \mathbf{J}_{\lambda_k,n_k}) \tag{7.6.2.7}$$

*(mit Eigenwerten* $\lambda_1, \ldots, \lambda_k$*). Diese Jordanzerlegung ist eindeutig bis auf eine Permutation der Jordanblöcke.*

Eine kurze analytische Anwendung der Jordanzerlegung ist im folgenden Beispiel gegeben.

**Beispiel 7.6.2.8 (Lösung einer Differentialgleichung mit einer oberen Dreiecksmatrix)**
Gegeben sei das Differentialgleichungssystem:

$$\dot{\mathbf{u}} = \mathbf{J}\mathbf{u}$$

mit der Matrix $\mathbf{J}$:

$$\mathbf{J} = \begin{bmatrix} 5 & 1 & 0 \\ 0 & 5 & 1 \\ 0 & 0 & 5 \end{bmatrix}.$$

Wir lösen das System durch Rücksubstitution:

$$\begin{aligned} \dot{u}_3 = 5u_3 \quad &\Rightarrow \quad u_3(t) = e^{5t}u_3(0) \\ \dot{u}_2 = 5u_2 + e^{5t}u_3(0) \quad &\Rightarrow \quad u_2(t) = (u_2(0) + tu_3(0))e^{5t} \\ \dot{u}_1 = 5u_1 + (u_2(0) + tu_3(0))e^{5t} \quad &\Rightarrow \quad u_1(t) = (u_1(0) + tu_2(0) + \frac{t^2}{2}u_3(0))e^{5t} \,. \end{aligned}$$

⌟

# Kapitel 8

# Singulärwertzerlegung

## Inhalt


**8.1 Singulärwertzerlegung und Fundamentalsatz der linearen Algebra . . . . . 220**
**8.2 Anwendungen . . . . . 230**
8.2.1 Ausgleichsrechnung . . . . . 230
8.2.2 Pseudoinverse . . . . . 230
8.2.3 Berechnung der Norm einer Matrix . . . . . 231
8.2.4 Numerischer Rang einer Matrix . . . . . 233
8.2.5 Speicherreduktion . . . . . 233
8.2.6 Datenkompression . . . . . 234
8.2.7 Trends in den Daten . . . . . 236
8.2.8 Totales lineares Ausgleichsproblem . . . . . 237
8.2.9 Lineare Ausgleichsrechnung mit linearen Nebenbedingungen . . . . . 238
8.2.10 Polarzerlegung . . . . . 239


## 8.1 Singulärwertzerlegung und Fundamentalsatz der linearen Algebra

In diesem Kapitel kommen wir zum wichtigsten Theorem der Vorlesung, das auch die meisten aktuellen Anwendungen in der Wissenschaft und Technik hat. Das ist die Singulärwertzerlegung (Englisch: singular value decomposition, SVD), die auch den Fundamentalsatz der linearen Algebra ergänzt.

Im vorigen Kapitel haben wir gesehen, dass diagonalisierbare Matrizen sehr praktisch sind und wichtige Anwendungen haben. Im Falle der nicht-diagonalisierbaren Matrizen stehen die Schur-Zerlegung für praktische und die Jordan-Zerlegung für theoretische Zwecke zur Verfügung. Jedoch ist die Singulärwertzerlegung bequemer und liefert noch zahlreiche wichtige, moderne Anwendungen. Zudem funktioniert sie auch für nicht-quadratische Matrizen.

Wir erinnern uns: Für eine normale Matrix $\mathbf{A} \in \mathbb{R}^{n\times n}/\mathbb{C}^{n\times n}$, haben wir die Diagonalisierung (oder auch Spektralzerlegung) mit unitären Transformationen gesehen:

$$\begin{aligned}\mathbf{A} = \mathbf{S}\mathbf{\Lambda}\mathbf{S}^{\mathsf{H}} &= \lambda_1\mathbf{P}_1 + \lambda_2\mathbf{P}_2 + \ldots + \lambda_n\mathbf{P}_n \\ &= \lambda_1\mathbf{s}_1\mathbf{s}_1^{\mathsf{H}} + \lambda_2\mathbf{s}_2\mathbf{s}_2^{\mathsf{H}} + \ldots + \lambda_n\mathbf{s}_n\mathbf{s}_n^{\mathsf{H}}.\end{aligned}$$

Die Elemente der Diagonalmatrix $\mathbf{\Lambda}$ sind die Eigenwerte von $\mathbf{A}$ und die orthonormalen Vektoren aus den Spalten von $\mathbf{S}$ sind die entsprechenden Eigenvektoren. Bisher haben wir nur den Fall quadratischer Matrizen

betrachtet und ausserdem haben wir gesehen, dass eine unitäre Diagonalisierung nur für normale Matrizen $\mathbf{A}^{\mathsf{H}}\mathbf{A} = \mathbf{A}\mathbf{A}^{\mathsf{H}}$ möglich ist. Die Schur-Zerlegung $\mathbf{A} = \mathbf{S}\mathbf{T}\mathbf{S}^{\mathsf{H}}$ funktioniert für alle quadratischen $n \times n$ Matrizen. Sie verwendet auch nur eine unitäre Matrix (eine orthonormale Basis in $\mathbb{C}^n$) und verrät die Eigenwerte auf der Diagonale von $\mathbf{T}$, die aber nur eine obere Dreiecksmatrix (nicht mehr eine einfache Diagonalmatrix) ist.

Im allgemeinen Fall, d. h. für Matrizen, die nicht normal sind, und sogar für Matrizen der Dimension $m \times n$ mit $m \neq n$, führen wir nun die *Singulärwertzerlegung* ein. Der Kompromiss, den wir für eine Diagonalisierung in diesem allgemeinen Fall machen müssen, ist, dass wir anstatt *einer* Orthonormalbasis, *zwei* Orthonormalbasen verwenden werden. Für $m \neq n$ sind das natürlicherweise je eine Orthonormalbasis in $\mathbb{R}^n$ und $\mathbb{R}^m$, für $m = n$ sind dies zwei orthonormale Basen in $\mathbb{R}^n$. Die spezielle Verbindung zwischen diesen Basen wird durch die Matrix $\mathbf{A}$ hergestellt.

**Satz 8.1.0.1. Singulärwertzerlegung (Singular Value Decomposition/SVD)**

*Sei $\mathbf{A} \in \mathbb{R}^{m\times n}$ eine beliebige Matrix. Es gibt zwei orthogonale Matrizen $\mathbf{U} \in \mathbb{R}^{m\times m}$ und $\mathbf{V} \in \mathbb{R}^{n\times n}$ sodass:*

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathsf{T}}, \tag{8.1.0.2}$$

*wobei die Matrix $\mathbf{\Sigma} = diag(\sigma_1, \sigma_2, \ldots, \sigma_p) \in \mathbb{R}^{m\times n}$ $(p = \min(m, n))$ die Singulärwerte der Matrix $\mathbf{A}$ zusammenfasst. Die Singulärwerte stehen auf der Hauptdiagonale von $\mathbf{\Sigma}$ und sind der Grösse nach absteigend geordnet: $\sigma_1$ ist der grösste und $\sigma_p$ ist der kleinste Singulärwert:*

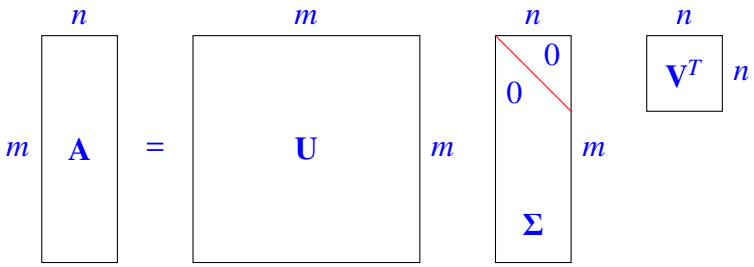
$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_p \geq 0\,. \tag{8.1.0.3}$$

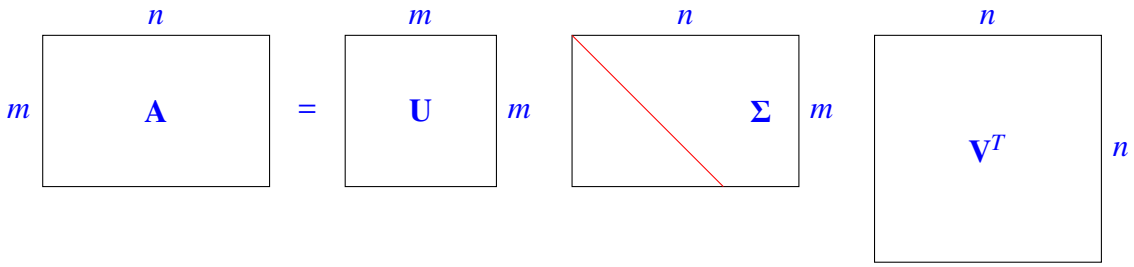
*Zudem gilt:*

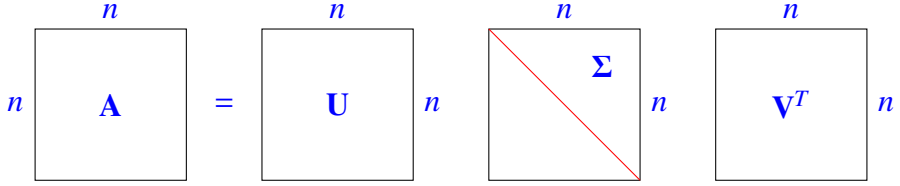
$$\mathbf{A} = \sum_{i=1}^{r} \sigma_i \mathbf{u}_i \mathbf{v}_i^{\mathsf{T}} \tag{8.1.0.4}$$

*mit $r = \operatorname{Rang}\mathbf{A}$.*

Die Singulärwerte sind in den folgenden Abbildungen mit rot dargestellt:



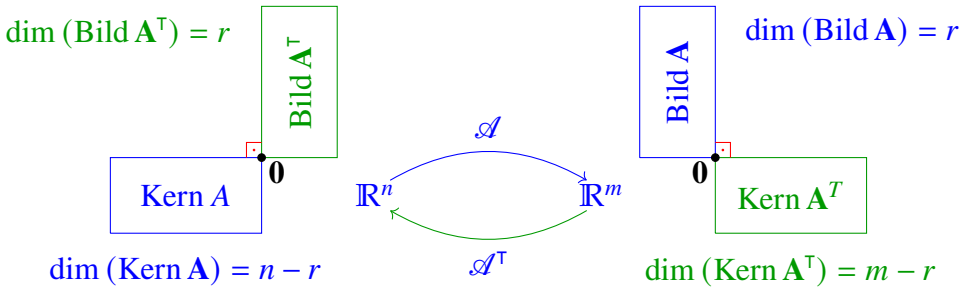



In der Literatur gibt es eine gewisse Konfusion: Manche betrachten die möglichen Nullen auf der Hauptdiagonale von $\mathbf{\Sigma}$ als Singulärwerte, andere nicht. Im folgenden Beweis wird die Bedeutung dieser Nullen klar.

*Beweis.* Wir teilen den Beweis in 5 Schritte auf.

1. Sei $\mathbf{A} \in \mathbb{R}^{m\times n}$. Die *gramsche Matrix* $\mathbf{A}^\mathsf{T}\mathbf{A} \in \mathbb{R}^{n\times n}$ ist symmetrisch. Dank des Spektralsatzes wissen wir, dass sie $n$ paarweise orthogonale Eigenvektoren hat.

   Wenn die Matrix $\mathbf{A}$ nicht vollen Rang hat, dann ist $0$ auch ein Eigenwert von $\mathbf{A}^\mathsf{T}\mathbf{A}$, was Vorsicht erfordert. Notieren wir also mit $r$ den Rang der Matrix $\mathbf{A}$ und erinnern uns an den Fundamentalsatz der linearen Algebra: $r = \mathrm{Rang}(\mathbf{A}) = \mathrm{Rang}(\mathbf{A}^\mathsf{T}\mathbf{A}) = \mathrm{Rang}(\mathbf{A}\mathbf{A}^\mathsf{T})$.




   Betrachten wir zuerst nur die $r$ Eigenwerte von $\mathbf{A}^\mathsf{T}\mathbf{A}$, die nicht null sind:

$$\lambda_1, \lambda_2, \ldots, \lambda_r \in \mathbb{R} \setminus \{0\}$$

   und die entsprechenden normierten Eigenvektoren

$$\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_r\,,$$

   die laut dem Spektralsatz paarweise orthogonal sind. Die restlichen $n - r$ paarweise orthogonalen und normierten Eigenvektoren entsprechen dem Eigenwert $0$ von $\mathbf{A}^\mathsf{T}\mathbf{A}$. Wir nummerieren sie weiter mit $\mathbf{v}_{r+1}, \ldots, \mathbf{v}_n$.

2. Nach der Definition der Eigenwerte und Eigenvektoren gilt für $j = 1, 2, \ldots, r$:

$$\mathbf{A}^\mathsf{T}\mathbf{A}\mathbf{v}_j = \lambda_j \mathbf{v}_j\,.$$

   Multiplizieren wir beide Seiten von links mit $\mathbf{A}$, so bekommen wir:

$$(\mathbf{A}\mathbf{A}^\mathsf{T})(\mathbf{A}\mathbf{v}_j) = \lambda_j(\mathbf{A}\mathbf{v}_j)\,.$$

   Hier sehen wir: $\mathbf{A}\mathbf{v}_j$ ist ein Eigenvektor der $m \times m$ Matrix $\mathbf{A}\mathbf{A}^\mathsf{T}$ zum Eigenwert $\lambda_j \neq 0$.
   Da $\mathbf{A}\mathbf{A}^\mathsf{T}$ symmetrisch ist, folgt aus dem Spektralsatz, dass seine $m$ Eigenvektoren orthogonal zueinander sein müssen. Demnach sind $\mathbf{A}\mathbf{v}_j$ paarweise orthogonal zueinander. Sie sind jedoch im Allgemeinen nicht normiert.

3. Wir berechnen im nächsten Schritt die euklidische Norm dieser Eigenvektoren $\mathbf{A}\mathbf{v}_j$ von $\mathbf{A}\mathbf{A}^\mathsf{T}$:

$$\begin{aligned}\|\mathbf{A}\mathbf{v}_j\|_2^2 &= (\mathbf{A}\mathbf{v}_j)^\mathsf{T}(\mathbf{A}\mathbf{v}_j) = \mathbf{v}_j^\mathsf{T}\mathbf{A}^\mathsf{T}\mathbf{A}\mathbf{v}_j \\ &= \mathbf{v}_j^\mathsf{T}(\mathbf{A}^\mathsf{T}\mathbf{A}\mathbf{v}_j) = \mathbf{v}_j^\mathsf{T}\lambda_j\mathbf{v}_j = \lambda_j\|\mathbf{v}_j\|_2^2 = \lambda_j\,.\end{aligned}$$

   Somit ist $\lambda_j = \|\mathbf{A}\mathbf{v}_j\|_2^2 > 0$, da wir den Eigenwert $0$ nicht berücksichtigen. Wir können nun die Singulärwerte definieren als:

$$\sigma_j = \|\mathbf{A}\mathbf{v}_j\| = \sqrt{\lambda_j} > 0\,.$$

4. Wir notieren die $r$ normierten Eigenvektoren von $\mathbf{A}\mathbf{A}^\top$, die nicht dem Eigenwert $0$ entsprechen, mit:

$$\mathbf{u}_j = \frac{1}{\sigma_j}\mathbf{A}\mathbf{v}_j, \quad j = 1, \dots, r\,.$$

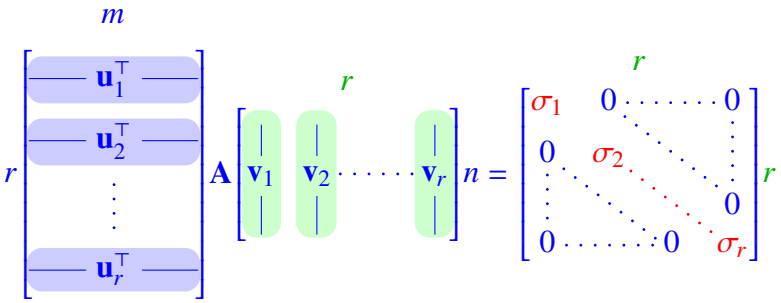
Die restlichen $m - r$ Elemente der paarweise orthogonalen und normierten Eigenvektoren von $\mathbf{A}\mathbf{A}^\top$ entsprechen dem Eigenwert $0$. Wir nummerieren sie weiter mit $\mathbf{u}_{r+1}, \dots, \mathbf{u}_m$.
Wir berechnen nun für $i, j \leq r$:

$$\begin{aligned}\mathbf{u}_i^\top \mathbf{A}\mathbf{v}_j &= \left(\frac{1}{\sigma_i}\mathbf{A}\mathbf{v}_i\right)^\top \mathbf{A}\mathbf{v}_j = \frac{1}{\sigma_i}\mathbf{v}_i^\top \mathbf{A}^\top \mathbf{A}\mathbf{v}_j \\ &= \frac{1}{\sigma_i}\mathbf{v}_i^\top \lambda_j \mathbf{v}_j = \frac{\lambda_j}{\sigma_i}\mathbf{v}_i^\top \mathbf{v}_j\,.\end{aligned}$$

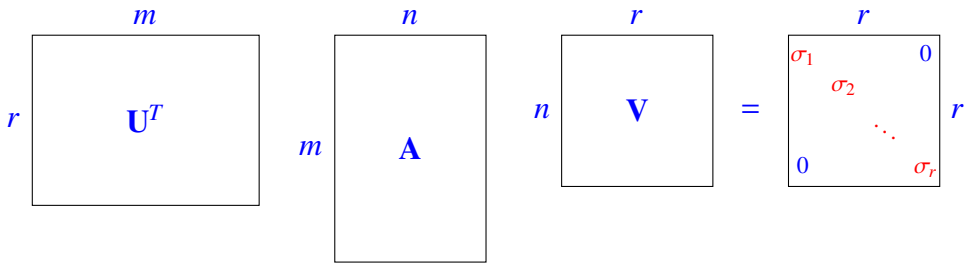
Falls $i \neq j$, führt die Orthogonalität der Vektoren $\mathbf{v}_i$ und $\mathbf{v}_j$ zu $\mathbf{u}_i^\top \mathbf{A}\mathbf{v}_j = 0$. Andernfalls gilt $\mathbf{u}_i^\top \mathbf{A}\mathbf{v}_j = \frac{\lambda_i}{\sigma_i}$, da wir die Eigenvektoren normiert haben: $\mathbf{v}_i^\top \mathbf{v}_i = \|\mathbf{v}_i\|_2^2 = 1$. Ausserdem gilt $\frac{\lambda_i}{\sigma_i} = \frac{\sigma_i^2}{\sigma_i} = \sigma_i$.
Das heisst:

$$\begin{aligned}\mathbf{u}_i^\top \mathbf{A}\mathbf{v}_j &= 0 \text{ für } i \neq j \\ &= \sigma_i \text{ für } i = j\,.\end{aligned}$$

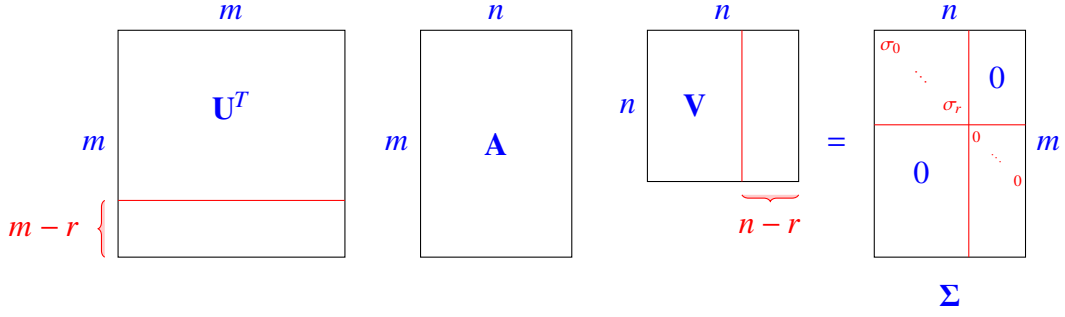
Wir haben also das Wesentliche bereits gebaut:




Diese Terme ergeben die reduzierte Singulärwertzerlegung (auch sparsame Singulärwertzerlegung genannt). Wir haben somit kürzere Matrizen $\mathbf{V}$ und $\mathbf{U}$ gefunden:




d. h. es fehlen nur $n - r$, bzw. $m - r$ Spalten in $\mathbf{V}$ und $\mathbf{U}$ zum Ziel:



5. Mithilfe der berechneten Vektoren aus Schritt 4 bilden wir die entsprechenden Matrizen. Die noch nicht verwendeten Teile der orthonormalen Basen in $\mathbb{R}^n$ und $\mathbb{R}^m$ von Eigenvektoren der Matrizen $\mathbf{A}^\top\mathbf{A}$ und $\mathbf{A}\mathbf{A}^\top$
$$\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_r, \mathbf{v}_{r+1}, \mathbf{v}_{r+2}, \ldots, \mathbf{v}_n \in \mathbb{R}^n$$
und
$$\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_r, \mathbf{u}_{r+1}, \mathbf{u}_{r+2}, \ldots, \mathbf{u}_m \in \mathbb{R}^m$$
kommen nun zum Einsatz.
Da $\mathbf{v}_{r+1}, \mathbf{v}_{r+2}, \ldots, \mathbf{v}_n$ Eigenvektoren von $\mathbf{A}^\top\mathbf{A}$ zum Eigenwert $0$ sind, gilt für jedes $j > r$: $\mathbf{A}^\top\mathbf{A}\mathbf{v}_j = \mathbf{0}$, also liegt $\mathbf{A}\mathbf{v}_j$ im Kern von $\mathbf{A}^\top$. Allerdings ist $\mathbf{A}\mathbf{v}_j$ offensichtlich auch im Bild von $\mathbf{A}$. Aus dem Fundamentalsatz der linearen Algebra wissen wir aber, dass das einzige gemeinsame Element des Kerns von $\mathbf{A}^\top$ und Bild von $\mathbf{A}$ der Nullvektor ist, d. h. $\mathbf{A}\mathbf{v}_j = \mathbf{0}$. Daher sind $\mathbf{A}\mathbf{v}_{r+1}, \ldots, \mathbf{A}\mathbf{v}_n$ alle $\mathbf{0}$, so dass die Ergänzung der Matrizen $\mathbf{V}$ und $\mathbf{U}$ mit den Spalten $\mathbf{v}_{r+1}, \ldots, \mathbf{v}_n$, bzw. $\mathbf{u}_{r+1}, \ldots, \mathbf{u}_m$ das Ziel erfüllt. Die Matrizen $\mathbf{V}, \mathbf{U}$ sind also die orthogonalen Matrizen, die aus den Eigenbasen von $\mathbf{A}^\top\mathbf{A}$ und $\mathbf{A}\mathbf{A}^\top$, in passender Reihenfolge geordnet, bestehen.
Es folgt:
$$\mathbf{U}^\top\mathbf{A}\mathbf{V} = \mathbf{\Sigma} \quad \Rightarrow \quad \mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$$
mit $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r > 0$.

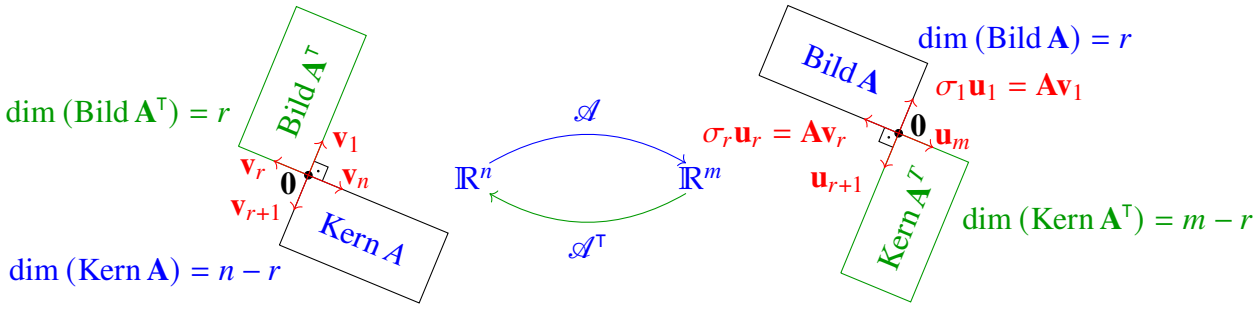
□

Im obigen Beweis der Singulärwertzerlegung haben wir auch den zweiten Teil des Fundamentalsatzes der linearen Algebra erhalten:

**Satz 8.1.0.5. Fundamentalsatz der linearen Algebra - Teil II.**

*Mit den Notationen aus der Singulärwertzerlegung gilt:*

1. $\mathbf{v}_{r+1}, \ldots, \mathbf{v}_n$ *ist eine Orthonormalbasis des Kerns von* $\mathbf{A}$*;*
2. $\mathbf{v}_1, \ldots, \mathbf{v}_r$ *spannen das Bild von* $\mathbf{A}^\top$ *auf und heissen rechte Singulärvektoren von* $\mathbf{A}$*;*
3. $\mathbf{u}_1, \ldots, \mathbf{u}_r$ *spannen das Bild von* $\mathbf{A}$ *auf und heissen linke Singulärvektoren von* $\mathbf{A}$*;*
4. $$\mathbf{u}_1^\top\mathbf{A}\mathbf{v}_1 = \sigma_1\,, \quad \mathbf{u}_2^\top\mathbf{A}\mathbf{v}_2 = \sigma_2\,, \ldots, \mathbf{u}_r^\top\mathbf{A}\mathbf{v}_r = \sigma_r\,;$$
5. $$\begin{aligned} \mathbf{u}_1 = \frac{1}{\sigma_1}\mathbf{A}\mathbf{v}_1 \quad &\Rightarrow \quad \mathbf{A}\mathbf{v}_1 = \sigma_1\mathbf{u}_1 \\ \vdots \qquad &\qquad \vdots \\ \mathbf{u}_r = \frac{1}{\sigma_r}\mathbf{A}\mathbf{v}_r \quad &\Rightarrow \quad \mathbf{A}\mathbf{v}_r = \sigma_r\mathbf{u}_r \\ &\qquad \mathbf{A}\mathbf{v}_{r+1} = \mathbf{0} \\ &\qquad \vdots \\ &\qquad \mathbf{A}\mathbf{v}_n = \mathbf{0}\,. \end{aligned}$$




**Bemerkung 8.1.0.6 (Eindeutigkeit der SVD)**
Aus dem Beweis sehen wir: Für eine gegebene Matrix $\mathbf{A}$ sind die Singulärwerte eindeutig. Wenn die Singulärwerte von $\mathbf{A}$ unterschiedlich sind, dann sind die entsprechenden Singulärvektoren aus der reduzierten Form eindeutig bis auf die Orientierung $\mathbf{u}'_j = \pm\mathbf{u}_j$, $\mathbf{v}'_j = \pm\mathbf{v}_j$. Für einen multiplen Singulärwert gibt es mehr Freiheit, denn wir haben viele Möglichkeiten, eine orthonormale Basis im entsprechenden Eigenraum der gramschen Matrix $\mathbf{A}^T\mathbf{A}$ auszuwählen.

⌟

**Beispiel 8.1.0.7 (SVD einer nicht-symmetrischen Matrix)** [1]
Sei eine Matrix $\mathbf{A}$ gegeben, die wir nun mittels Singulärwertzerlegung zerlegen möchten:

$$\mathbf{A} = \begin{bmatrix} 2 & 2 \\ -1 & 1 \end{bmatrix}.$$

Wir berechnen zuerst die Eigenwerte und Eigenvektoren von $\mathbf{A}^\top\mathbf{A}$ (für die Matrix $\mathbf{V}$):

$$\mathbf{A}^\top\mathbf{A} = \begin{bmatrix} 5 & 3 \\ 3 & 5 \end{bmatrix} \Rightarrow \text{EW}: \det\begin{bmatrix} 5-\lambda & 3 \\ 3 & 5-\lambda \end{bmatrix} = (\lambda-2)(\lambda-8)$$
$$\Rightarrow \lambda_1 = 8,\ \lambda_2 = 2$$
$$\Rightarrow \sigma_1 = \sqrt{8} = 2\sqrt{2},\ \sigma_2 = \sqrt{2}.$$

Die Eigenvektoren zu den berechneten Eigenwerten lauten demnach:

$$\mathbf{v}_1 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ 1 \end{bmatrix},\ \mathbf{v}_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} -1 \\ 1 \end{bmatrix}.$$

Wir können demnach die Matrix $\mathbf{V}$ konstruieren:

$$\mathbf{V} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix},\ \mathbf{V}^\top = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}.$$

Da wir die Spalten von $\mathbf{V}$, die Singulärwerte und die ursprüngliche Matrix kennen, können wir die Spaltenvektoren von $\mathbf{U}$ berechnen:

$$\mathbf{u}_1 = \frac{1}{\sigma_1}\mathbf{A}\mathbf{v}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix},\ \mathbf{u}_2 = \frac{1}{\sigma_2}\mathbf{A}\mathbf{v}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

Somit lautet die Singulärwertzerlegung von $\mathbf{A}$:

[1] Beispiel 3 aus Kapitel 7.7, Seite 366 in Str09.

$$\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top$$
$$\begin{bmatrix} 2 & 2 \\ -1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 2\sqrt{2} & 0 \\ 0 & \sqrt{2} \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}.$$

⌟

**Beispiel 8.1.0.8 (SVD einer singulären Matrix)** [2]
Sei die Matrix

$$\mathbf{A} = \begin{bmatrix} 1 & 1 \\ 7 & 7 \end{bmatrix}.$$

Wir gehen gleich wie im obigen Beispiel vor und berechnen zuerst die Spaltenvektoren von $\mathbf{V}$:

$$\mathbf{A}^\top\mathbf{A} = \begin{bmatrix} 50 & 50 \\ 50 & 50 \end{bmatrix} \Rightarrow \text{EW}: \det \begin{bmatrix} 50-\lambda & 50 \\ 50 & 50-\lambda \end{bmatrix} = \lambda(\lambda - 100)$$
$$\Rightarrow \lambda_1 = 100,\ \lambda_2 = 0$$
$$\Rightarrow \sigma_1 = \sqrt{100} = 10,\ \sigma_2 = 0.$$

Die Eigenvektoren berechnen wir folgendermassen:

$$\begin{bmatrix} -50 & 50 \\ 50 & -50 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \mathbf{0} \Leftrightarrow \begin{bmatrix} -50 & 50 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \mathbf{0}.$$

Durch den Rangverlust bekommen wir nur einen Eigenvektor

$$\mathbf{v}_1 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

Da wir zwei Vektoren für die Berechnung von $\mathbf{U}$ brauchen, können wir durch das Ergänzen von $\mathbf{v}_1$ zu einer Orthonormalbasis in $\mathbb{R}^2$ einen zweiten Vektor berechnen:

$$\mathbf{v}_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ -1 \end{bmatrix} \perp \mathbf{v}_1.$$

Damit können wir die Spaltenvektoren von $\mathbf{U}$ berechnen:

$$\mathbf{u}_1 = \frac{1}{\sigma_1}\mathbf{A}\mathbf{v}_1 = \frac{\sqrt{2}}{10} \begin{bmatrix} 1 \\ 7 \end{bmatrix}.$$

Den zweiten Vektor bekommen wir, genau wie $\mathbf{v}_2$, durch Erweitern von $\mathbf{u}_1$ zu einer ONB in $\mathbb{R}^2$:

$$\mathbf{u}_2 = \frac{1}{\sqrt{50}} \begin{bmatrix} -7 \\ 1 \end{bmatrix}.$$

[2] Dieses Beispiel wird in Abschnitt 9.8 mit Python gelöst.

Somit lautet die Singulärwertzerlegung von $\mathbf{A}$:

$$\mathbf{A} = \frac{1}{\sqrt{50}}\begin{bmatrix}1 & -7\\ 7 & 1\end{bmatrix}\begin{bmatrix}10 & 0\\ 0 & 0\end{bmatrix}\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}.$$

⌟

**Beispiel 8.1.0.9 (Die SVD ist nicht eindeutig.)**
Sei die Matrix

$$\mathbf{A} = \begin{bmatrix}4 & -2 & 1 & 1 & 1 & 1\\ -2 & 4 & 1 & 1 & 1 & 1\\ 1 & 1 & 4 & -2 & 1 & 1\\ 1 & 1 & -2 & 4 & 1 & 1\\ 1 & 1 & 1 & 1 & 1 & 1\end{bmatrix}.$$

Die Singulärwerte sind $\sigma_1 = 6, \sigma_2 = 6, \sigma_3 = \sqrt{30} \approx 5.48$. Der Rang der Matrix $\mathbf{A}$ ist 3, somit ist die Dimension des Kerns von $\mathbf{A}$ genau $6-3=3$ und die Dimension des Kerns von $\mathbf{A}^T$ ist $5-3=2$. Somit gibt es unendlich viele orthogonale Basen im Kern von $\mathbf{A}$ und im Kern von $\mathbf{A}^T$. Auch zum doppelten Singulärwert 6 entsprechen unendlich viele orthogonale Basen von linken und rechte Singulärvektoren. Hier sind je zwei Beispiele von solche Paaren von Basen:

- Ein Paar Singulärvektoren zu $\sigma_1 = \sigma_2 = 6$ ist:

$$\text{Rechts } (\mathbf{V}) \quad \mathbf{v}_1 = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\-1\\0\\0\\0\\0\end{bmatrix}, \quad \mathbf{v}_2 = \frac{1}{\sqrt{2}}\begin{bmatrix}0\\0\\1\\-1\\0\\0\end{bmatrix}$$

$$\text{Links } (\mathbf{U}) \quad \mathbf{u}_1 = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\-1\\0\\0\\0\end{bmatrix}, \quad \mathbf{u}_2 = \frac{1}{\sqrt{2}}\begin{bmatrix}0\\0\\1\\-1\\0\end{bmatrix}.$$

- Ein zweites Paar Singulärvektoren zu $\sigma_1 = \sigma_2 = 6$ ist:

$$\text{Rechts } (\tilde{\mathbf{V}}) \quad \tilde{\mathbf{v}}_1 = \frac{\mathbf{v}_1+\mathbf{v}_2}{\sqrt{2}} = \frac{1}{2}\begin{bmatrix}1\\-1\\1\\-1\\0\\0\end{bmatrix}, \quad \tilde{\mathbf{v}}_2 = \frac{\mathbf{v}_1-\mathbf{v}_2}{\sqrt{2}} = \frac{1}{2}\begin{bmatrix}1\\-1\\-1\\1\\0\\0\end{bmatrix}$$

$$\text{Links } (\tilde{\mathbf{U}}) \quad \tilde{\mathbf{u}}_1 = \frac{\mathbf{u}_1+\mathbf{u}_2}{\sqrt{2}} = \frac{1}{2}\begin{bmatrix}1\\-1\\1\\-1\\0\end{bmatrix}, \quad \tilde{\mathbf{u}}_2 = \frac{\mathbf{u}_1-\mathbf{u}_2}{\sqrt{2}} = \frac{1}{2}\begin{bmatrix}1\\-1\\-1\\1\\0\end{bmatrix}.$$

- Singulärvektoren zu $\sigma_3 = \sqrt{30}$:

$$\mathbf{v}_3 = \frac{1}{\sqrt{6}}\begin{bmatrix}1\\1\\1\\1\\1\\1\end{bmatrix}$$

$$\mathbf{u}_3 = \frac{1}{\sqrt{5}}\begin{bmatrix}1\\1\\1\\1\\1\end{bmatrix}.$$

Diese sind aufgrund der Vielfachheit von $\sigma_3 = \sqrt{30}$ bis auf das gemeinsame Vorzeichen eindeutig. Das heisst, es gibt genau zwei Möglichkeiten: $(\mathbf{v}_3, \mathbf{u}_3)$ und $(-\mathbf{v}_3, -\mathbf{u}_3)$

- Ein Paar Orthonormalbasen: im Kern von $\mathbf{A}$ und im Kern von $\mathbf{A}^T$:

$$\text{Rechts } (\mathbf{V}) \quad \mathbf{v}_4 = \frac{1}{\sqrt{2}}\begin{bmatrix}0\\0\\0\\0\\1\\-1\end{bmatrix}, \quad \mathbf{v}_5 = \frac{1}{2}\begin{bmatrix}1\\1\\-1\\-1\\0\\0\end{bmatrix}, \quad \mathbf{v}_6 = \frac{1}{\sqrt{12}}\begin{bmatrix}1\\1\\1\\1\\-2\\-2\end{bmatrix}$$

$$\text{Links } (\mathbf{U}) \quad \mathbf{u}_4 = \frac{1}{2}\begin{bmatrix}1\\1\\-1\\-1\\0\end{bmatrix}, \quad \mathbf{u}_5 = \frac{1}{\sqrt{20}}\begin{bmatrix}1\\1\\1\\1\\-4\end{bmatrix}.$$

- Ein zweites Paar Orthonormalbasen: im Kern von $\mathbf{A}$ und im Kern von $\mathbf{A}^T$:

$$\text{Rechts } (\tilde{\mathbf{V}}) \quad \tilde{\mathbf{v}}_4 = \frac{1}{\sqrt{2}}\begin{bmatrix}0\\0\\0\\0\\1\\-1\end{bmatrix}, \quad \tilde{\mathbf{v}}_5 = \frac{1}{2}\begin{bmatrix}1\\1\\0\\0\\-1\\-1\end{bmatrix}, \quad \tilde{\mathbf{v}}_6 = \frac{1}{\sqrt{12}}\begin{bmatrix}-1\\-1\\2\\2\\-1\\-1\end{bmatrix}$$

$$\text{Links } (\tilde{\mathbf{U}}) \quad \tilde{\mathbf{u}}_4 = \frac{1}{\sqrt{6}}\begin{bmatrix}1\\1\\0\\0\\-2\end{bmatrix}, \quad \tilde{\mathbf{u}}_5 = \frac{1}{\sqrt{30}}\begin{bmatrix}-2\\-2\\3\\3\\-2\end{bmatrix}.$$

Die erste (aus unendlich vielen) Singulärwertzerlegung hat die Matrizen

$$\mathbf{V} = \begin{bmatrix} \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{6}} & 0 & \frac{1}{2} & \frac{1}{\sqrt{12}} \\ -\frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{6}} & 0 & \frac{1}{2} & \frac{1}{\sqrt{12}} \\ 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} & 0 & -\frac{1}{2} & \frac{1}{\sqrt{12}} \\ 0 & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} & 0 & -\frac{1}{2} & \frac{1}{\sqrt{12}} \\ 0 & 0 & \frac{1}{\sqrt{6}} & \frac{1}{\sqrt{2}} & 0 & -\frac{2}{\sqrt{12}} \\ 0 & 0 & \frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{2}} & 0 & -\frac{2}{\sqrt{12}} \end{bmatrix}$$

und

$$\mathbf{U} = \begin{bmatrix} \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{5}} & \frac{1}{\sqrt{6}} & -\frac{2}{\sqrt{30}} \\ -\frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{5}} & \frac{1}{\sqrt{6}} & -\frac{2}{\sqrt{30}} \\ 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{5}} & 0 & \frac{3}{\sqrt{30}} \\ 0 & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{5}} & 0 & \frac{3}{\sqrt{30}} \\ 0 & 0 & \frac{1}{\sqrt{5}} & -\frac{2}{\sqrt{6}} & -\frac{2}{\sqrt{30}} \end{bmatrix}.$$

Rotationen in den Unterräumen generieren automatisch neue Zerlegungen, z.B.:

$$\tilde{\mathbf{V}} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{\sqrt{6}} & 0 & \frac{1}{2} & -\frac{1}{\sqrt{12}} \\ -\frac{1}{2} & -\frac{1}{2} & \frac{1}{\sqrt{6}} & 0 & \frac{1}{2} & -\frac{1}{\sqrt{12}} \\ \frac{1}{2} & -\frac{1}{2} & \frac{1}{\sqrt{6}} & 0 & 0 & \frac{2}{\sqrt{12}} \\ -\frac{1}{2} & \frac{1}{2} & \frac{1}{\sqrt{6}} & 0 & 0 & \frac{2}{\sqrt{12}} \\ 0 & 0 & \frac{1}{\sqrt{6}} & \frac{1}{\sqrt{2}} & -\frac{1}{2} & -\frac{1}{\sqrt{12}} \\ 0 & 0 & \frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{2}} & -\frac{1}{2} & -\frac{1}{\sqrt{12}} \end{bmatrix}$$

und

$$\tilde{\mathbf{U}} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{\sqrt{5}} & \frac{1}{2} & \frac{1}{\sqrt{20}} \\ -\frac{1}{2} & -\frac{1}{2} & \frac{1}{\sqrt{5}} & \frac{1}{2} & \frac{1}{\sqrt{20}} \\ \frac{1}{2} & -\frac{1}{2} & \frac{1}{\sqrt{5}} & -\frac{1}{2} & \frac{1}{\sqrt{20}} \\ -\frac{1}{2} & \frac{1}{2} & \frac{1}{\sqrt{5}} & -\frac{1}{2} & \frac{1}{\sqrt{20}} \\ 0 & 0 & \frac{1}{\sqrt{5}} & 0 & -\frac{4}{\sqrt{20}} \end{bmatrix}.$$

⌟

**Bemerkung 8.1.0.10 (Reduzierte SVD)**
In der Praxis genügt es, nur die reduzierte Singulärwertzerlegung anzugeben:

$$\mathbf{A} = \begin{bmatrix} \mathbf{u}_1 & \cdots & \cdots & \mathbf{u}_r \end{bmatrix} \begin{bmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_r \end{bmatrix} \begin{bmatrix} \mathbf{v}_1^T \\ \vdots \\ \mathbf{v}_r^T \end{bmatrix}.$$

⌟

## 8.2 Anwendungen

### 8.2.1 Ausgleichsrechnung

Jetzt sind wir in der Lage, den Ansatz professioneller Löser für die lineare Ausgleichsrechnung 5.1 zu verstehen. Im Unterschied zu den Lösungsansätzen mit der Normalengleichung und mit der QR-Zerlegung gibt der Ansatz mit der Singulärwertzerlegung immer stabile Ergebnisse, auch im Fall $\operatorname{Rang}\mathbf{A} = r < n$. Wir starten mit der Singulärwertzerlegung

$$\mathbf{A} = \begin{bmatrix}\mathbf{U}_1 & \mathbf{U}_2\end{bmatrix}\begin{bmatrix}\mathbf{\Sigma}_r & 0\\ 0 & 0\end{bmatrix}\begin{bmatrix}\mathbf{V}_1^\top\\ \mathbf{V}_2^\top\end{bmatrix}.$$

Hier haben wir die wesentlichen Teile der orthogonalen Matrizen $\mathbf{U}$ und $\mathbf{V}$ und der Diagonalmatrix $\mathbf{\Sigma}$ gekennzeichnet: $\mathbf{U}_1$ und $\mathbf{V}_1$ enthalten die ersten $r$ Spalten und $\mathbf{\Sigma}_r$ ist der obere linke $r \times r$-Block, sie entsprechen also der reduzierten Singulärwertzerlegung. Aus dem zweiten Teil des Fundamentalsatzes der linearen Algebra ist klar, dass nur $\Sigma_r, \mathbf{U}_1, \mathbf{V}_1$ wesentlich zur Lösung des linearen Ausgleichsproblems beitragen. Mit dieser Zerlegung kann man ähnlich wie beim Ansatz mit der QR-Zerlegung argumentieren:

$$\begin{aligned}\|\mathbf{Ax}-\mathbf{b}\|_2^2 &= \left\|\mathbf{U\Sigma V}^\top\mathbf{x}-\mathbf{UU}^\top\mathbf{b}\right\|_2^2 = \left\|\mathbf{\Sigma V}^\top\mathbf{x}-\mathbf{U}^\top\mathbf{b}\right\|_2^2\\ &= \|\begin{bmatrix}\mathbf{\Sigma}_r & 0\\ 0 & 0\end{bmatrix}\begin{bmatrix}\mathbf{V}_1^\top\\ \mathbf{V}_2^\top\end{bmatrix}\mathbf{x}-\begin{bmatrix}\mathbf{U}_1^\top\\ \mathbf{U}_2^\top\end{bmatrix}\mathbf{b}\|_2^2\\ &= \left\|\mathbf{\Sigma}_r\mathbf{V}_1^\top\mathbf{x}-\mathbf{U}_1^\top\mathbf{b}\right\|_2^2+\left\|\mathbf{U}_2^\top\mathbf{b}\right\|_2^2,\end{aligned}$$

was offensichtlich minimal für

$$\mathbf{\Sigma}_r\mathbf{V}_1^\top\mathbf{x} = \mathbf{U}_1^\top\mathbf{b}$$

ist. Das Minimum ist $\left\|\mathbf{U}_2^\top\mathbf{b}\right\|_2^2$.
Da $\mathbf{\Sigma}_r$ eine Diagonalmatrix ist, bei der alle Elemente auf der Diagonale nicht null sind, können wir die Bedingung für das Minimum etwas umschreiben:

$$\mathbf{V}_1^\top\mathbf{x} = \mathbf{\Sigma}_r^{-1}\mathbf{U}_1^\top\mathbf{b}\,.$$

Da $\mathbf{V}_1$ aus den ersten $r$ Spalten der orthogonalen Matrix $\mathbf{V}$ besteht, schneidet das Produkt $\mathbf{V}_1^\top\mathbf{V}$ die ersten $r$ Zeilen der $n \times n$-Identitätsmatrix ab: $\mathbf{V}_1^\top\mathbf{V} = \mathbf{I}_{r,n}$. Damit lässt sich die Lösung

$$\mathbf{x} = \mathbf{V}_1\mathbf{\Sigma}_r^{-1}\mathbf{U}_1^\top\mathbf{b}$$

direkt darstellen.

### 8.2.2 Pseudoinverse

[3] Das Lösen des linearen Ausgleichsproblems mittels Singulärwertzerlegung definiert einen linearen Operator, der die inverse Matrix verallgemeinert. Die Matrix dieses linearen Operators nennen wir *Pseudoinverse*.

[3] Die Berechnung der Pseudoinversen mit Python wird kurz in Abschnitt 9.5.1 aufgegriffen.

**Definition 8.2.2.1. Pseudoinverse nach Moore (1920) und Penrose (1955)**

Sei $\mathbf{A} \in \mathbb{R}^{m\times n}$ gegeben und sei $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^\mathsf{T}$ ihre Singulärwertzerlegung.
Die *Pseudoinverse (Moore-Penrose)* ist definiert als:

$$\mathbf{A}^\dagger = \mathbf{V}\mathbf{\Sigma}^\dagger\mathbf{U}^\mathsf{T},$$

mit der $n \times m$ Diagonalmatrix:

$$\mathbf{\Sigma}^\dagger = \begin{bmatrix} \sigma_1^{-1} & 0 & \cdots & \cdots & 0 \\ 0 & \sigma_2^{-1} & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ & & \sigma_r^{-1} & 0 & \vdots \\ 0 & \cdots & \cdots & 0 & 0 \end{bmatrix}.$$

## 8.2.3 Berechnung der Norm einer Matrix

Sei $\mathbf{A}$ eine Matrix. Wir wollen die von der euklidischen Vektornorm induzierte Matrixnorm

$$\|\mathbf{A}\|_2 = \max_{\|\mathbf{x}\|_2=1} \|\mathbf{A}\mathbf{x}\|_2$$

berechnen. Die Singulärwertzerlegung und die Orthogonalität der Matrizen $\mathbf{U}, \mathbf{V}$ ergeben:

$$\begin{aligned} \|\mathbf{A}\mathbf{x}\|_2^2 &= \|\mathbf{U}\mathbf{\Sigma}\mathbf{V}^\mathsf{T}\mathbf{x}\|_2^2 = \|\mathbf{\Sigma}\mathbf{V}^\mathsf{T}\mathbf{x}\|_2^2 \\ &= \sigma_1^2|\mathbf{v}_1^\mathsf{T}\mathbf{x}|^2 + \sigma_2^2|\mathbf{v}_2^\mathsf{T}\mathbf{x}|^2 + \ldots + \sigma_r^2|\mathbf{v}_r^\mathsf{T}\mathbf{x}|^2 . \end{aligned}$$

Wir notieren $\mathbf{V}^\mathsf{T}\mathbf{x} = \mathbf{y}$:

$$\|\mathbf{A}\|_2 = \max_{\|\mathbf{x}\|_2=1} \|\mathbf{A}\mathbf{x}\|_2 = \max_{\|\mathbf{x}\|_2=1} \|\mathbf{\Sigma}\mathbf{V}^\mathsf{T}\mathbf{x}\|_2 = \max_{\|\mathbf{y}\|_2=1} \|\mathbf{\Sigma}\mathbf{y}\|_2 .$$

Für den Term $\|\mathbf{\Sigma}\mathbf{y}\|_2^2$ gilt zudem:

$$\begin{aligned} \|\mathbf{\Sigma}\mathbf{y}\|_2^2 &= \sigma_1^2|y_1|^2 + \ldots + \sigma_r^2|y_r|^2 \\ &= \sigma_1^2\left(|y_1|^2 + \frac{\sigma_2^2}{\sigma_1^2}|y_2|^2 + \ldots + \frac{\sigma_r^2}{\sigma_1^2}|y_r|^2\right) \\ &\le \sigma_1^2\|\mathbf{y}\|_2^2 . \end{aligned}$$

Aus dieser Erkenntnis können wir schliessen, dass $\max_{\|\mathbf{y}\|_2=1} \|\mathbf{\Sigma}\mathbf{y}\|_2 = \sigma_1$ für $\mathbf{y} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$ erreicht wird. Das heisst:

$$\|\mathbf{A}\|_2 = \max_{\mathbf{x}\in\mathbb{R}^n\setminus\{\mathbf{0}\}} \frac{\|\mathbf{A}\mathbf{x}\|_2}{\|\mathbf{x}\|_2} = \sigma_1 \text{ erreicht für } \mathbf{x} = \mathbf{V}\begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \mathbf{v}_1 .$$

Analog können wir das entsprechende Maximum finden, wenn $\mathbf{x}$ auf gewisse Unterräume beschränkt ist. Wir betrachten $\mathbf{x} \in \operatorname{span}\{\mathbf{v}_1\}^\perp$. Solche $\mathbf{x}$ haben keine Komponente in $\mathbf{v}_1$ (da sie orthogonal auf $\mathbf{v}_1$ sind). Darum fehlt in der obigen Summe der erste Term, so dass wir anstelle von $\sigma_1$ den nächstgrösseren Singulärwert $\sigma_2$ ausklammern können, so dass

$$\max_{\mathbf{x}\in\operatorname{span}\{\mathbf{v}_1\}^\perp\setminus\{\mathbf{0}\}} \frac{\|\mathbf{A}\mathbf{x}\|_2}{\|\mathbf{x}\|_2} = \sigma_2 \text{ erreicht für } \mathbf{x} = \mathbf{V}\begin{bmatrix}0\\1\\0\\\vdots\\0\end{bmatrix} = \mathbf{v}_2\,.$$

Wir betrachten $\mathbf{x} \in \operatorname{span}\{\mathbf{v}_1, \mathbf{v}_2\}^\perp$. Solche $\mathbf{x}$ haben keine Komponente in $\mathbf{v}_1$ oder in $\mathbf{v}_2$ (da sie orthogonal auf $\mathbf{v}_1$ und $\mathbf{v}_2$ sind). Darum fehlen in der obigen Summe die ersten beiden Terme, so dass wir den nächstgrösseren Singulärwert $\sigma_3$ ausklammern können, so dass

$$\max_{\mathbf{x}\operatorname{span}\{\mathbf{v}_1,\mathbf{v}_2\}^\perp\setminus\{\mathbf{0}\}} \frac{\|\mathbf{A}\mathbf{x}\|_2}{\|\mathbf{x}\|_2} = \sigma_3 \text{ erreicht für } \mathbf{x} = \mathbf{V}\begin{bmatrix}0\\0\\1\\\vdots\\0\end{bmatrix} = \mathbf{v}_3\,,$$

usw. Daraus sieht man auch die Konditionszahl 7.3 einer invertierbaren Matrix:

$$k(\mathbf{A}) = \frac{\sigma_1}{\sigma_n}\,.$$

**Bemerkung 8.2.3.1 (Schlecht konditionierte Matrix)**
Eine Matrix ist schlecht konditioniert, falls ihr kleinster Singulärwert sehr nahe an $0$ liegt.

⌟

**Definition 8.2.3.2. Wichtige Matrizennormen**

Sei $\mathbf{A}$ eine $m \times n$ Matrix.
Die *euklidische Norm* oder (wie gerade bewiesen) auch *Spektralnorm* ist

$$\|\mathbf{A}\|_2 = \max_{\mathbf{x}\neq\mathbf{0}} \frac{\|\mathbf{A}\mathbf{x}\|_2}{\|\mathbf{x}\|_2} = \sigma_1\,.$$

Die *Frobenius Norm* ist

$$\|\mathbf{A}\|_F = \sqrt{\sum_{i,j} |A_{i,j}|^2} = \sqrt{\operatorname{Spur}(\mathbf{A}^\intercal\mathbf{A})} = \sqrt{\sigma_1^2 + \sigma_2^2 + \ldots + \sigma_r^2}\,.$$

Die *Nuklearnorm* ist

$$\|\mathbf{A}\|_N = \sigma_1 + \sigma_2 + \ldots + \sigma_r\,.$$

**Satz 8.2.3.3. Nützliche Ungleichungen zwischen Matrixnormen**

*Für $\mathbf{A} \in \mathbb{R}^{m\times n}$ von Rang $r$ gelten die Ungleichungen:*

$$\begin{aligned}
\|\mathbf{A}\|_2 &\le \|\mathbf{A}\|_F \le \sqrt{r}\|\mathbf{A}\|_2\\
\|\mathbf{A}\|_F &\le \|\mathbf{A}\|_N \le \sqrt{r}\|\mathbf{A}\|_F\\
\|\mathbf{A}\|_\infty &\le \|\mathbf{A}\|_2 \le \sqrt{mn}\|\mathbf{A}\|_\infty\\
\frac{1}{\sqrt{n}}\|\mathbf{A}\|_\infty &\le \|\mathbf{A}\|_2 \le \sqrt{m}\|\mathbf{A}\|_\infty\\
\frac{1}{\sqrt{m}}\|\mathbf{A}\|_1 &\le \|\mathbf{A}\|_2 \le \sqrt{n}\|\mathbf{A}\|_1\,,
\end{aligned}$$

*und ausserdem:*

$$\|\mathbf{A}\|_2 \le \sqrt{\|\mathbf{A}\|_1\|\mathbf{A}\|_\infty}\,.$$

### 8.2.4 Numerischer Rang einer Matrix

Wir können mithilfe der Singulärwertzerlegung einfach ablesen, welchen Rang die ursprüngliche Matrix hatte. Andererseits haben wir gerade gesehen, dass eine Matrix schlecht konditioniert ist, falls ihr kleinster Singulärwert sehr nahe an $0$ liegt. D. h. beim Lösen von linearen Gleichungssystemen mit einer Matrix welche winzig kleine Singulärwerte hat, sind grosse Rundungsfehler zu erwarten. Wenn aber die letzten Singulärwerte sehr klein im Vergleich zu den ersten Singulärwerten sind, stellt sich die Frage, ob die Daten der Matrix nicht bereits gestört sind. Darum macht es Sinn, zwischen mathematischem Rang $r$ und *numerischem* Rang $k$ einer Matrix zu unterscheiden:

$$\sigma_1 \ge \sigma_2 \ldots \ge \sigma_k \gg \sigma_{k+1} \ge \ldots \ge \sigma_r > 0\,.$$

Am Term $\sigma_k$ können wir ablesen, dass der numerische Rang der Matrix $k$ ist, sowie am Term $\sigma_r$, dass der mathematisch exakte Rang der Matrix $r$ ist.
Darum muss man bei der Verwendung der Pseudoinverse in professioneller Software auch die Grenze angeben, ab der die Singulärwerte als Null betrachtet werden sollten.

### 8.2.5 Speicherreduktion

Eine Matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$, die als Tabelle gespeichert ist, besetzt $m \times n$ Speicherplätze. Bei grossen Matrizen kann das zu viel sein. Würde man z. B. die Inzidenzmatrix der Wörter/Begriffe, die auf Webseiten erscheinen, im Tabellenformat speichern, so würde kein Speicher ausreichen. Dank der reduzierten Singulärwertzerlegung können wir die Anzahl benötigter Speicherplätze in manchen Fällen aber drastisch verkleinern. Wir haben dabei $r \cdot m$ benötigte Speicherplätze für die Vektoren $\mathbf{u}_1, \ldots, \mathbf{u}_r$ und $r \cdot n$ Plätze für die Vektoren $\mathbf{v}_1, \ldots, \mathbf{v}_r$ und weitere $r$ Plätze für die Singulärwerte $\sigma_1, \ldots, \sigma_r$. Vergleichen wir nun die benötigten Plätze der reduzierten SVD mit der ursprünglichen Matrix, so sehen wir:

$$r(m+n+1) \ll m \cdot n\,.$$

D. h., es ist viel effizienter, anstelle der Matrix $\mathbf{A}$ in Tabellenform nur die Faktoren $\mathbf{V}, \mathbf{U}, \mathbf{\Sigma}$ zu speichern. Die Speicherreduktion ist umso interessanter, je grösser der Unterschied zwischen Rang $r$ und Dimensionen $m, n$ ist. Der Unterschied ist noch grösser, wenn wir den numerischen Rang $k \ll r$ in den Algorithmen berücksichtigen.
Die Schwierigkeit verschiebt sich: Vom Speicherproblem hin zu Algorithmen, die in der Lage sind, mit der Matrix nur als Produkt der Faktoren aus der Singulärwertzerlegung ($\mathbf{U\Sigma V}^\mathsf{T}$) zu arbeiten. Beispielsweise gehört die Gauss-Elimination nicht dazu. Das Lösen von linearen Gleichungssystemen in diesem Format

muss dann über iterative Methoden erfolgen, die nur einen Algorithmus für die Matrix-Vektor-Multiplikation brauchen! Insbesondere muss dabei die Matrix nie in der Tabellenform dargestellt werden! In einer Vorlesung über Numerische Methoden erfahren Sie mehr darüber.

### 8.2.6 Datenkompression

Wir haben gesehen, dass der numerische Rang $k$ über die Wichtigkeit der Informationen in der Matrix $\mathbf{A}$ entscheidet und sich die Matrix für kleine $k$ sehr sparsam darstellen lässt. Diese zwei Bemerkungen sind fundamental für Bereiche wie Data Science und Machine Learning, denn mit einer dynamischen Entscheidung über den numerischen Rang $k$ können wir die Information sparsam zur Verfügung stellen. Die Frage ist, was man über den so entstandenen Fehler sagen kann. Wir werden gleich sehen, dass dieser Ansatz den Fehler in allen wichtigen Matrix-Normen minimiert, d. h. diese Methode ideal ist!

Sei $\mathbf{A}$ eine diagonalisierbare quadratische Matrix mit reellen Eigenwerten. Wir können diese anordnen: $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$. Die Diagonalisierung ist

$$\mathbf{A} = \sum_{j=1}^{n} \lambda_j \mathbf{s}_j \hat{\mathbf{s}}_j^{\mathsf{T}} = \mathbf{S}\mathbf{\Lambda}\mathbf{S}^{-1} .$$

In der Bemerkung 7.3.0.10 sahen wir, dass die Diagonalisierung eine Darstellung der Matrix als Summe von Rang 1 Matrizen (bei denen es sich eventuell um schiefe Projektoren handelt) gibt.

Für eine beliebige Matrix $\mathbf{A}$ gilt mit der Singulärwertzerlegung:

$$\mathbf{A} = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^{\mathsf{T}} + \sigma_2 \mathbf{u}_2 \mathbf{v}_2^{\mathsf{T}} + \ldots + \sigma_r \mathbf{u}_r \mathbf{v}_r^{\mathsf{T}} .$$

Wir sehen also: Die Singulärwertzerlegung schreibt die Matrix $\mathbf{A}$ auch als eine Summe von Rang 1 Matrizen.

Im Gegensatz zur Darstellung via Diagonalisierung kann diese Formel immer verwendet werden, um erfolgreich Matrizen mit grossem Rang durch Matrizen mit kleinem Rang zu approximieren. In diesem Kontext redet man von der Kompression der Matrix $\mathbf{A}$:

$$\mathbf{A} \approx \sigma_1 \mathbf{u}_1 \mathbf{v}_1^{\mathsf{T}} + \ldots + \sigma_k \mathbf{u}_k \mathbf{v}_k^{\mathsf{T}} \quad \text{mit } k \ll r .$$

**Beispiel 8.2.6.1 (Kompression mit SVD)**
Gegeben sei die Matrix

$$\mathbf{A} = \begin{bmatrix} 2 & 100 \\ 0 & 1 \end{bmatrix} .$$

Die Matrix ist zwar nicht symmetrisch, hat aber reelle Eigenwerte: 2 und 1. Die entsprechenden Eigenvektoren sind:

$$\mathbf{s}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad \text{und} \quad \mathbf{s}_2 = \begin{bmatrix} 1 \\ -\frac{1}{100} \end{bmatrix} .$$

Nachdem wir die inverse Matrix von $\mathbf{S}$ berechnet haben, können wir die Diagonalisierung und die Darstellung als Summe von Rang 1 Matrizen schreiben:

$$\mathbf{A} = \mathbf{S}\mathbf{\Lambda}\mathbf{S}^{-1} = \lambda_1 \mathbf{s}_1 \hat{\mathbf{s}}_1^{\mathsf{T}} + \lambda_2 \mathbf{s}_2 \hat{\mathbf{s}}_2^{\mathsf{T}} ,$$

oder konkret:

$$\mathbf{A} = 2 \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 & 100 \end{bmatrix} + 1 \begin{bmatrix} 1 \\ -\frac{1}{100} \end{bmatrix} \begin{bmatrix} 0 & -100 \end{bmatrix} .$$

Wir wollen die Matrix $\mathbf{A}$ mit einer Matrix vom Rang 1 approximieren. Aufgrund der Diagonalisierung können wir das entweder mit $\lambda_1 \mathbf{s}_1 \hat{\mathbf{s}}_1^\mathsf{T}$ oder mit $\lambda_2 \mathbf{s}_2 \hat{\mathbf{s}}_2^\mathsf{T}$ probieren. Beide Versuche scheitern aber gnadenlos, denn die jeweiligen Fehler sind offensichtlich sehr gross:

$$\mathbf{A} - 2\begin{bmatrix}1\\0\end{bmatrix}\begin{bmatrix}1 & 100\end{bmatrix} = \begin{bmatrix}0 & -100\\0 & 1\end{bmatrix}$$

und

$$\mathbf{A} - 1\begin{bmatrix}1\\-\frac{1}{100}\end{bmatrix}\begin{bmatrix}0 & -100\end{bmatrix} = \begin{bmatrix}2 & 200\\0 & 0\end{bmatrix}.$$

Die Singulärwertzerlegung der Matrix $\mathbf{A}$ ist weniger schön:

$$\mathbf{A} = \begin{bmatrix}0.9995 & -0.01\\0.01 & 0.9995\end{bmatrix}\begin{bmatrix}100.025 & 0\\0 & 0.02\end{bmatrix}\begin{bmatrix}0.01999 & 0.9991\\-0.9998 & 0.01999\end{bmatrix},$$

was

$$\begin{aligned}\mathbf{A} &= \sigma_1 \mathbf{u}_1 \mathbf{v}_1^\mathsf{T} + \sigma_2 \mathbf{u}_2 \mathbf{v}_2^\mathsf{T}\\ &= 100.025\begin{bmatrix}0.9995\\0.001\end{bmatrix}\begin{bmatrix}0.01999 & 0.9991\end{bmatrix} + 0.02\begin{bmatrix}0.01\\0.9995\end{bmatrix}\begin{bmatrix}0.9998 & 0.01999\end{bmatrix}\end{aligned}$$

als Summe von Rang 1 Matrizen ergibt. Wenn wir in dieser Summe nur der Term mit dem grössten Singulärwert beibehalten, erhalten wir die Approximation

$$\mathbf{A} \approx \sigma_1 \mathbf{u}_1 \mathbf{v}_1^\mathsf{T},$$

die den wesentlich kleineren Fehler

$$\mathbf{A} - \sigma_1 \mathbf{u}_1 \mathbf{v}_1^\mathsf{T} = \sigma_2 \mathbf{u}_2 \mathbf{v}_2^\mathsf{T} = 10^{-4}\begin{bmatrix}2 & -0.04\\-200 & 4\end{bmatrix}$$

hat. Das ist kein Zufall.

**Satz 8.2.6.2. Beste Approximation an einer Matrix: Schmidt 1907 wiederentdeckt von Eckart 1933 und Young 1936**

*Sei $\mathbf{A} \in \mathbb{R}^{m\times n}$ beliebig. Für jedes $k \le \operatorname{Rang}\mathbf{A}$ gibt die abgebrochene Singulärwertzerlegung*

$$\mathbf{A}_k = \sum_{j=1}^{k} \sigma_j \mathbf{u}_j \mathbf{v}_j^\mathsf{T}$$

*die beste Approximation von Rang $k$ an $\mathbf{A}$ im Sinne von:*

$$\|\mathbf{A} - \mathbf{A}_k\| = \min_{\substack{\mathbf{B}\in\mathbb{R}^{m\times n}\\ \operatorname{Rang}\mathbf{B}\le k}} \|\mathbf{A} - \mathbf{B}\| = \sigma_{k+1}.$$

*Dabei kann $\|\cdot\|$ die euklidische, die Frobenius- oder Nuklearnorm der Matrix sein.*

*Beweis. (Fall der euklidischen/Spektralnorm.)* Sei $\mathbf{B} \in \mathbb{R}^{m\times n}$ mit $\operatorname{Rang}\mathbf{B} \le k$ beliebig. Wir zeigen die Eigenschaft für die euklidische Norm. Da $\|\mathbf{A} - \mathbf{A}_k\|_2 = \sigma_{k+1}$, reicht es zu zeigen, dass

$$\sigma_{k+1} \le \|\mathbf{A} - \mathbf{B}\|_2 = \max_{\mathbf{x}\neq\mathbf{0}} \frac{\|(\mathbf{A}-\mathbf{B})\mathbf{x}\|_2}{\|\mathbf{x}\|_2}.$$

Da $\operatorname{Rang}\mathbf{B} \leq k$ muss $\operatorname{Kern}\mathbf{B} \subset \mathbb{R}^n$ mindestens Dimension $n-k$ haben. Der von den ersten $k+1$ rechten Singulärvektoren aufgespannte Unterraum $V_{k+1} = \operatorname{span}\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_{k+1}\}$ hat Dimension $k+1$.

Die Summe der Dimensionen der zwei Unterräume von $\mathbb{R}^n$ ist

$$\dim \operatorname{Kern}\mathbf{B} + \dim V_{k+1} \geq (n-k) + (k+1) = n+1\,,$$

so dass deren Schnitt $\operatorname{Kern}\mathbf{B} \cap V_{k+1}$ ein Unterraum von Dimension mindestens 1 ist. Somit gibt es mindestens ein $\mathbf{x} \in \operatorname{Kern}\mathbf{B} \cap V_{k+1}$:

$$\mathbf{x} = c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \ldots + c_{k+1}\mathbf{v}_{k+1}\,,$$

mit $\mathbf{x} \neq \mathbf{0}$. Für dieses $\mathbf{x}$ rechnen wir:

$$\begin{aligned}\|(\mathbf{A}-\mathbf{B})\mathbf{x}\|_2^2 = \|\mathbf{A}\mathbf{x}\|_2^2 &= \|c_1\sigma_1\mathbf{u}_1 + c_1\sigma_2\mathbf{u}_2 + \ldots + c_1\sigma_{k+1}\mathbf{u}_{k+1}\|_2^2 \\ &= c_1^2\sigma_1^2 + c_2^2\sigma_2^2 + \ldots + c_{k+1}^2\sigma_{k+1}^2 \\ &\geq \left(c_1^2 + c_2^2 + \ldots + c_{k+1}^2\right)\sigma_{k+1}^2 = \|\mathbf{x}\|_2^2\sigma_{k+1}^2\,.\end{aligned}$$

Somit haben wir gezeigt, dass es für jedes beliebige $\mathbf{B}$ mindestens ein $\mathbf{x} \neq \mathbf{0}$ gibt, so dass

$$\frac{\|(\mathbf{A}-\mathbf{B})\mathbf{x}\|_2}{\|\mathbf{x}\|_2} \geq \sigma_{k+1}\,.$$

Das bedeutet $\|\mathbf{A}-\mathbf{B}\|_2 \geq \sigma_{k+1} = \|\mathbf{A}-\mathbf{A}_k\|_2$, was den Beweis abschliesst. □

### 8.2.7 Trends in den Daten

Die Singulärwertzerlegung kann auch dazu benutzt werden, um Trends in den Daten festzustellen. [4] Stellen wir uns vor, eine $\mathbf{A} \in \mathbb{R}^{m\times n}$ Matrix enthält Bewertungen von $m$ Benutzer für $n$ Filme. Man möchte gerne wissen, wie viele Trends es in dieser Community gibt, welche Filme und welche Benutzer zu welchem Trend gehören.

**Beispiel 8.2.7.1 (Filme)** [5] Ein kleines Beispiel mit $m = 4$ Benutzern und $n = 6$ Filmen ist

$$\begin{bmatrix} 5 & 5 & 1 & 1 & 1 & 1 \\ 5 & 4 & 1 & 1 & 1 & 1 \\ 1 & 1 & 5 & 5 & 5 & 4 \\ 1 & 1 & 4 & 5 & 5 & 5 \end{bmatrix}.$$

Normalerweise zentriert man die Ergebnisse pro Spalte, indem man den Mittelwert davon subtrahiert:

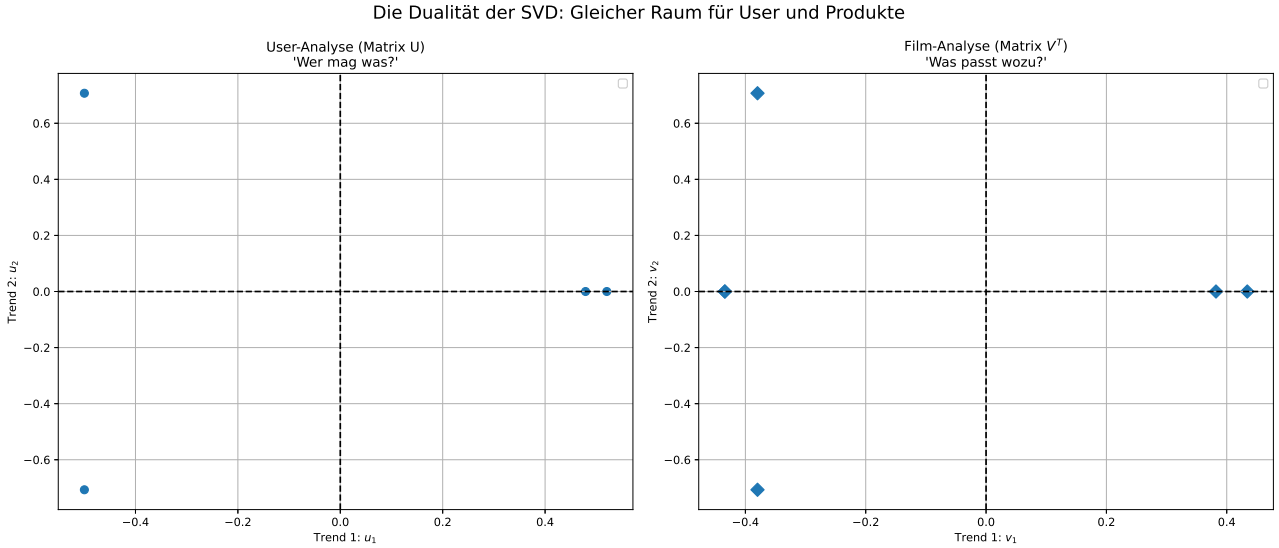
$$\mathbf{A} = \begin{bmatrix} 2. & 2.25 & -1.75 & -2. & -2. & -1.75 \\ 2. & 1.25 & -1.75 & -2. & -2. & -1.75 \\ -2. & -1.75 & 2.25 & 2. & 2. & 1.25 \\ -2. & -1.75 & 1.25 & 2. & 2. & 2.25 \end{bmatrix}.$$

Die Singulärwerte sind ungefähr: 9.21, 1., 0.65, 0.

- Der erste (dem Singulärwert 9.21 entsprechende) linke Singulärvektor ist $[0.52, 0.48, -0.5, -0.5]^T$. Er zeigt dank der Vorzeichen eine klare Trennung der Zuschauer in zwei Gruppen an: Die ersten beiden und die letzten beiden bilden jeweils eine Gruppe.
- Der erste rechte Singulärvektor ist $[0.43, 0.38, -0.38, -0.43, -0.43, -0.38]$. Auch hier ergibt sich dank der Vorzeichen eine klare Trennung der Filme in zwei Gruppen: Die ersten beiden und die letzten vier ergeben jeweils eine Gruppe.

[4] Diese Art der Datenauswertung entspricht im Wesentlichen der Hauptkomponentenanalyse.
[5] Die Aspekte der Datenauswertung und Visualisierung mit Python dieses Beispiels werden in Abschnitt 9.8.1 beleuchtet.

Mit der Singulärwertzerlegung kann man somit erkennen, dass die ersten beiden Zuschauer eher die ersten beiden Filme mögen während die anderen zwei Zuschauer eher die letzten vier Filme bevorzugen. Damit kann man sowohl die Filme als auch die Zuschauer kategorisieren, was für weitere Empfehlungen verwendet werden kann. Der zweite linke bzw. rechte Singulärvektor zeigt den zweiten Trend, der innerhalb der Zuschauer bzw. der Filme differenziert. Die folgenden Bilder zeigen die Aufteilung in diesem kleinen Beispiel und dann in etwas grösseren Datenbanken mit mehreren Unterkategorien.
Die Singulärwertzerlegung unterscheidet zwischen Trends (positiven oder negativen Vorzeichen in den Daten), wobei die konkreten Vorzeichen eine Konsequenz der verwendeten Konvention sind und die Namen der Trends natürlich von uns gegeben sind.




**Beispiel 8.2.7.2 (Mehrfache Singulärwerte)** In diesem Kontext ist die Matrix aus dem Beispiel 8.1.0.9 spannend: der doppelte Singulärwert 6 sagt aus, dass es keinen dominanten Trend gibt, sondern zwei gleich starke. (Auch der Dritte zum Singulärwert $\sqrt{30}$ ist fast gleich stark); da es unendlich viele Basen für die entsprechenden Räume gibt, kann man keine Aufteilungen und keine Empfehlungen machen. Solche Fälle sind in der Praxis aber kaum anzutreffen. Wenn sie trotzdem auftreten, deutet das auf eine strukturelle Symmetrie hin. Für die Datenanalyse heisst das: wenn wir einen doppelten Singulärwert finden, sollten wir nach zwei unabhängigen Mechanismen suchen, die sich zufällig oder systembedingt genau gleich stark auf das analysierte System auswirken. ⌟

## 8.2.8 Totales lineares Ausgleichsproblem

Sei $\mathbf{A} \in \mathbb{R}^{m\times n}$ von Rang $n$ und $\mathbf{b} \in \mathbb{R}^m$. Die Lösung des linearen Ausgleichsproblems mit $\mathbf{A}$ und $\mathbf{b}$ beruht auf der Projektion der rechten Seite $\mathbf{b}$ auf Bild $\mathbf{A}$. Hier verfolgen wir einen anderen Ansatz: Wir suchen $\widehat{\mathbf{A}}, \widehat{\mathbf{b}}$ und $\widehat{\mathbf{x}}$, so dass das lineare Gleichungssystem $\widehat{\mathbf{A}}\widehat{\mathbf{x}} = \widehat{\mathbf{b}}$ so nah wie möglich an $\mathbf{Ax} = \mathbf{b}$ im Sinne von

$$\min_{\mathbf{B},\operatorname{Rang}\mathbf{B}=n} \|\mathbf{C}-\mathbf{B}\| = \left\|\mathbf{C}-\widehat{\mathbf{C}}\right\|$$

mit erweiterten Matrizen $\mathbf{C} = [\mathbf{A}\,\mathbf{b}]$, $\widehat{\mathbf{C}} = [\widehat{\mathbf{A}}\,\widehat{\mathbf{b}}]$, und $\widehat{\mathbf{b}} \in \operatorname{Bild}\widehat{\mathbf{A}}$ ist.
Wir verwenden die Singulärwertzerlegung von $\mathbf{C}$:

$$\mathbf{C} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathsf{T}} = \sum_{j=1}^{n+1} \sigma_j \mathbf{u}_j \mathbf{v}_j^{\mathsf{T}}\,.$$

Wir verwenden die optimale Approximation an $\mathbf{C}$ in der Menge der Rang $n$ Matrizen:

$$\widehat{\mathbf{C}} = \sum_{j=1}^{n} \sigma_j \mathbf{u}_j \mathbf{v}_j^{\mathsf{T}}\,.$$

Die Orthogonalität von $\mathbf{V}$ sichert $\widehat{\mathbf{C}}\mathbf{v}_{n+1} = 0$, d. h. $\operatorname{Kern}\widehat{\mathbf{C}}$ hat Dimension 1 und es geht nur darum, eine passende Skalierung von $\mathbf{v}_{n+1}$ zu finden. Falls das Element $v_{n+1\,n+1} \neq 0$ ist, dann löst $\widehat{\mathbf{x}} = -\dfrac{1}{v_{n+1\,n+1}}\mathbf{v}_{n+1}$ das Problem $\widehat{\mathbf{A}}\widehat{\mathbf{x}} = \widehat{\mathbf{b}}$, denn $[\widehat{\mathbf{x}}, -1]^\mathsf{T} \in \operatorname{Kern}\widehat{\mathbf{C}}$.

### 8.2.9 Lineare Ausgleichsrechnung mit linearen Nebenbedingungen

Sei $\mathbf{A} \in \mathbb{R}^{m\times n}$ mit $m \geq n = \operatorname{Rang}\mathbf{A}$, und $\mathbf{b} \in \mathbb{R}^m$, dazu noch $\mathbf{C} \in \mathbb{R}^{p\times n}$ mit $n > p = \operatorname{Rang}\mathbf{C}$ und $\mathbf{d} \in \operatorname{Bild}\mathbf{C} \subset \mathbb{R}^p$. Wir wollen nun das lineare Ausgleichsproblem $\|\mathbf{Ax}-\mathbf{b}\|_2^2 = \min!$ mit der linearen Nebenbedingung $\mathbf{Cx} = \mathbf{d}$ lösen.

Eine erste Methode beruht auf der Verwendung von Lagrange-Multiplikatoren. Diese sind nicht auf lineare Probleme limitiert. Um das Optimierungsproblem mit Nebenbedingungen zu lösen werden *Lagrange-Multiplikatoren* $\mathbf{m} \in \mathbb{R}^p$ und die Lagrange-Funktion

$$L(\mathbf{x},\mathbf{m}) = \frac{1}{2}\|\mathbf{Ax}-\mathbf{b}\|_2^2 + \mathbf{m}^\mathsf{T}(\mathbf{Cx}-\mathbf{d})$$

eingeführt und das Sattelpunktproblem

$$\min_{\mathbf{x}\in\mathbb{R}^n}\max_{\mathbf{m}\in\mathbb{R}^p} L(\mathbf{x},\mathbf{m})$$

gelöst. Intuitiv: Die Gradienten der zu optimierenden Funktion und der Nebenbedingung müssen am Optimum parallel sein, sonst könnte man einen besseren Punkt finden. Geschrieben als partielle Ableitungen:

$$\frac{\partial L}{\partial \mathbf{x}}(\mathbf{x},\mathbf{m}) = \mathbf{0}$$
$$\frac{\partial L}{\partial \mathbf{m}}(\mathbf{x},\mathbf{m}) = \mathbf{0}$$

d. h.

$$\mathbf{A}^\mathsf{T}(\mathbf{Ax}-\mathbf{b}) + \mathbf{C}^\mathsf{T}\mathbf{m} = \mathbf{0}$$
$$\mathbf{Cx}-\mathbf{d} = \mathbf{0}$$

oder

$$\begin{bmatrix}\mathbf{A}^\mathsf{T}\mathbf{A} & \mathbf{C}^\mathsf{T}\\ \mathbf{C} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{x}\\ \mathbf{m}\end{bmatrix} = \begin{bmatrix}\mathbf{A}^\mathsf{T}\mathbf{b}\\ \mathbf{d}\end{bmatrix},$$

was mit einer Block-LU-Zerlegung gelöst werden kann.
Der Nachteil dieser Methode ist, dass die Kondition der gramschen Matrix $\mathbf{A}^\mathsf{T}\mathbf{A}$ sehr gross sein kann. Der Vorteil dieser Methode ist, dass eine eventuell vorhandene Struktur in den Matrizen $\mathbf{A}, \mathbf{C}$ leicht auszunutzen ist.

Die alternative Methode verwendet die Singulärwertzerlegung von $\mathbf{C}$, ist also besser für die Kondition, verliert aber die Struktur der Matrizen. Sei

$$\mathbf{C} = \mathbf{U}\begin{bmatrix}\boldsymbol{\Sigma} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{V}_1^\mathsf{T}\\ \mathbf{V}_2^\mathsf{T}\end{bmatrix}$$

so dass $\operatorname{Kern}\mathbf{C} = \operatorname{Bild}\mathbf{V}_2$. Ähnlich der Technik bei der linearen Ausgleichsrechnung machen wir den Ansatz:

$$\mathbf{x}_0 = \mathbf{V}_1\boldsymbol{\Sigma}^{-1}\mathbf{U}^\mathsf{T}\mathbf{d}$$

und sichern

$$\mathbf{x} = \mathbf{x}_0 + \mathbf{V}_2\mathbf{y}$$

mit $\mathbf{y} \in \mathbb{R}^{n-p}$, da die Nebenbedingung $\mathbf{Cx} = \mathbf{d}$ automatisch für solche $\mathbf{x}$ erfüllt ist. Die zu minimierende Quantität ist für solche $\mathbf{x}$ dann

$$\|\mathbf{Ax}-\mathbf{b}\|_2^2 = \|\mathbf{Ax}_0 + \mathbf{AV}_2\mathbf{y} - \mathbf{b}\|_2^2 = \|\mathbf{AV}_2\mathbf{y} - (\mathbf{b}-\mathbf{Ax}_0)\|_2^2,$$

was ein standardisiertes lineares Ausgleichsproblem für $\mathbf{y}$ ist.

### 8.2.10 Polarzerlegung

Wir können uns der Singulärwertzerlegung bedienen, um die Polarzerlegung einer Matrix auszurechnen. Die Polarzerlegung ist eine Verallgemeinerung der Polarform einer komplexen Zahl ($z = |z|e^{i\varphi} = |z|(\cos\varphi + i\sin\varphi)$). Sie repräsentiert eine gegebene Matrix als Produkt einer Rotationsmatrix mit einer Streckungsmatrix.

**Definition 8.2.10.1. Polarzerlegung**

Die Polarzerlegung einer Matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ mit $\operatorname{Rang}\mathbf{A} = n$ ist

$$\mathbf{A} = \mathbf{QH}\,,$$

mit einer orthogonalen Matrix $\mathbf{Q}$ und einer symmetrisch positiv definiten Matrix $\mathbf{H}$.

Das kommt direkt aus der Singulärwertzerlegung:

$$\mathbf{A} = \mathbf{U\Sigma V}^\mathsf{T} = \mathbf{UV}^\mathsf{T}(\mathbf{V\Sigma V}^\mathsf{T})\,,$$

mit $\mathbf{Q} = \mathbf{UV}^\mathsf{T}$, $\mathbf{H} = \mathbf{V\Sigma V}^\mathsf{T}$. Klar gilt auch: $\mathbf{A}^\mathsf{T}\mathbf{A} = \mathbf{H}^2$. Die Matrix $\mathbf{Q}$ beschreibt eine Drehung, die Matrix $\mathbf{H}$ eine Streckung. Die Eigenwerte von $\mathbf{H}$ sind die Faktoren der Streckungen in den Richtungen der entsprechenden Eigenvektoren.
Natürlich kann man diese Rechnung auch auf der anderen Seite machen, sodass die Polarzerlegung auch $\mathbf{A} = \mathbf{KQ}$ (mit einer symmetrisch positiv definiten Matrix $\mathbf{K}$ und orthogonalen Matrix $\mathbf{Q}$) geschrieben werden kann.

# Kapitel 9

# Python

## Inhalt


**9.1 Einleitung . . . . . 240**
9.1.1 NumPy und SciPy . . . . . 241
9.1.2 Leistung . . . . . 241
9.1.3 Quellcode . . . . . 241
**9.2 Lineare Gleichungssysteme . . . . . 242**
9.2.1 Vektoren und Matrizen . . . . . 242
9.2.2 Lineare Gleichungssysteme . . . . . 243
9.2.3 Rang einer Matrix . . . . . 244
9.2.4 LU-Zerlegung . . . . . 244
9.2.5 QR-Zerlegung . . . . . 245
9.2.6 Weitere Zerlegungen . . . . . 246
9.2.7 Gauss-Elimination . . . . . 246
**9.3 Lineare Räume . . . . . 259**
9.3.1 Bild und Kern einer Matrix . . . . . 259
**9.4 Normen und Skalarprodukte . . . . . 260**
9.4.1 Vektor- und Matrixnormen . . . . . 260
9.4.2 Skalarprodukt . . . . . 260
**9.5 Ausgleichsrechnung . . . . . 261**
9.5.1 Pseudoinverse . . . . . 262
**9.6 Determinante . . . . . 263**
**9.7 Eigenwertproblem . . . . . 264**
9.7.1 Konditionszahl . . . . . 265
9.7.2 Zirkulante Matrix . . . . . 265
9.7.3 Fourier-Matrix . . . . . 265
9.7.4 Faltung . . . . . 266
**9.8 Singulärwertzerlegung . . . . . 267**
9.8.1 Trends in den Daten . . . . . 267


## 9.1 Einleitung

Im folgenden Abschnitt wird eine kurze Einführung in die Verwendung von Python zum Lösen von Problemen der Linearen Algebra, wie sie in dieser Vorlesung besprochen werden, gegeben. Dafür werden insbesondere die Pakete NumPy und SciPy verwendet.

### 9.1.1 NumPy und SciPy

Zunächst sollen kurz die Zusammenhänge zwischen NumPy und SciPy beleuchtet werden.

NumPy stellt ein mehrdimensionales Array-Objekt zur Verfügung, welches beispielsweise zur Repräsentation von Vektoren und Matrizen verwendet werden kann. Zudem stellt es Operationen darauf zur Verfügung. Enthalten sind sowohl grundlegende Operationen, wie das Einlesen, die Multiplikation, die Transposition, als auch fortgeschrittene Operationen, wie die QR-Zerlegung, Singulärwertzerlegung und diskrete Fouriertransformation.

SciPy nutzt die Datenstruktur von NumPy und stellt dafür zahlreiche Algorithmen zur Verfügung. Zusätzlich steht eine deutlich vollständigere Sammlung an Funktionen aus der Linearen Algebra und weiteren Bereichen zur Verfügung. Insbesondere für den Bereich der Linearen Algebra ist die Empfehlung, die SciPy Algorithmen anstelle der analogen NumPy Funktionen zu verwenden.

### 9.1.2 Leistung

Da SciPy die Routinen der Bibliotheken BLAS und LAPACK nutzt und somit die Berechnungen in kompilierten Sprachen durchgeführt werden, kann eine deutlich bessere Rechenleistung im Vergleich zu einer äquivalenten Implementierung in Python erwartet werden.

### 9.1.3 Quellcode

Der gesamte Quellcode dieses Kapitels ist auf der Plattform Code-Expert im Abschnitt ‘‘Code examples’’ verfügbar. Den Zugang dazu erhalten Studierende zu Beginn des Semesters.

## 9.2 Lineare Gleichungssysteme

### 9.2.1 Vektoren und Matrizen

Wir verwenden das Beispiel 1.1.0.7, um grundlegende Operationen mit Vektoren und Matrizen zu veranschaulichen.

Zunächst initialisieren wir eine Matrix.

**Python-code 9.2.1.1: Matrixeigenschaften**

```
>>> A = np.array([[1, 0, 0], [-1, 1, 0], [0, 1, 1]])
>>> A
array([[ 1,  0,  0],
       [-1,  1,  0],
       [ 0,  1,  1]])
>>> A.ndim
2
>>> A.shape
(3, 3)
>>> A.dtype.name
'int64'
```

Hierbei sehen wir bereits drei wichtige Eigenschaften von np.array-Instanzen. Diese haben nämlich eine Dimension, eine Form und einen Elementtyp.

- In der Mathematik spricht man bei Dimension 0 von einem Skalar, bei Dimension 1 von einem Vektor und bei Dimension 2 von einer Matrix.
- Die Form beschreibt die Anzahl an Zeilen und Spalten einer Matrix.
- Der Datentyp legt fest, ob es sich beispielsweise um eine ganze Zahl, eine Fliesskommazahl oder eine komplexe Zahl handelt. Zusätzlich kann die Grösse (in Byte) und somit der Wertebereich und die Genauigkeit bestimmt werden.

Es gibt zwei Möglichkeiten Vektoren zu initialisieren, welche im Folgenden Beispiel demonstriert werden.

**Python-code 9.2.1.2: Vektoreigenschaften**

```
>>> b = np.array([1,1,1])
>>> b
array([1, 1, 1])
>>> b.ndim
1
>>> b.shape
(3,)
>>> b_col = np.array([[1],[1], [1]])
>>> b_col
array([[1],
       [1],
       [1]])
>>> b_col.ndim
2
>>> b_col.shape
```

```
(3, 1)
```

Die Konsequenz der unterschiedlichen Initialisierungsformen wird bei der Matrix-Vektor-Multiplikation klar.

**Python-code 9.2.1.3: Matrix-Vektor-Multiplikation**

```
>>> A @ b
array([1, 0, 2])
>>> b @ A
array([0, 2, 1])

>>> A @ b_col
array([[1],
       [0],
       [2]])
>>> b_col @ A
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ValueError: matmul: Input operand 1 has a mismatch in its core
   dimension 0, with gufunc signature (n?,k),(k,m?)->(n?,m?) (size
   3 is different from 1)
```

Sowohl für die Matrix-Vektor-Multiplikation als auch für die Matrix-Matrix-Multiplikation wird der Operator @ verwendet. Wenn die Matrix A von rechts mit b_col multipliziert wird, erhalten wir das erwartete Ergebnis. Eine Multiplikation von links resultiert in einer Fehlermeldung, da eine solche bekannterweise nicht definiert ist.

Eine Multiplikation mit b verhält sich jedoch anders. Während die Multiplikation von rechts ebenfalls das erwartete Ergebnis ausgibt, findet bei der Multiplikation von links eine implizite Konversion zu $b^T$ statt.

Da solche und ähnliche Konversionen in der Praxis nützlich sind, wird häufig die Variante b verwendet, jedoch sollte man sich bewusst sein, dass diese auch zu unerwarteten Verhalten führen kann.

### 9.2.2 Lineare Gleichungssysteme

Ein lineares Gleichungssystem kann mit der Funktion `linalg.solve` gelöst werden. Falls bekannt ist, dass die Matrix eine besondere Struktur aufweist, kann diese optional angegeben werden. Beispiele dafür sind:

- obere Dreiecksmatrix
- symmetrische Matrix
- symmetrisch positiv definite Matrix

In solchen Fällen können die Berechnungen mit eigenen Lösern effizienter und/oder stabiler gemacht werden.

Sollte eine inverse Matrix explizit benötigt werden ist das ebenfalls möglich. Jedoch ist zu beachten, dass für das Lösen von linearen Gleichungssystemen `linalg.solve` im Allgemeinen schneller und stabiler ist.

Als Beispiel soll 1.1.0.7 betrachtet werden.

**Python-code 9.2.2.1: Lösung linearer Gleichungssysteme**

```
>>> A = np.array([[1, 0, 0], [-1, 1, 0], [0, 1, 1]])
```

```
>>> b = np.array([1,1,1])
>>> linalg.solve(A, b)
array([ 1.,  2., -1.])
>>> linalg.inv(A)
array([[ 1.,  0.,  0.],
       [ 1.,  1.,  0.],
       [-1., -1.,  1.]])
>>> linalg.inv(A) @ b
array([ 1.,  2., -1.])
```

### 9.2.3 Rang einer Matrix

Der Rang einer Matrix kann mit der Funktion np. linalg .matrix_rank berechnet werden. Wichtig zu beachten ist, dass die Rückgabe dem numerischen Rang, wie er in Abschnitt 8.2.4 besprochen wird, entspricht und nicht zwingend mit der Definition 1.2.0.6 übereinstimmt. Durch die Verwendung des numerischen Rangs wird sichergestellt, dass ein gegenüber numerischen Fehlern (vor allem Darstellungs- und Rundungsfehler) robustes Ergebnis zurückgegeben wird.

Zur Veranschaulichung betrachten wir die Matrix aus Beispiel 1.2.0.18.

**Python-code 9.2.3.1: Rang**

```
>>> A = np.array([
... [1, 0, 5, 0, 4, 0, 0, 1, 0],
... [0, 0, 0, 5, 0, 4, 3, 0, 2],
... [1, 5, 9, 3, 6, 1, 0, 1, 0],
... [1, 10, 13, 11, 8, 6, 3, 1, 2],
... [0, 5, 4, 18, 2, 18, 12, 2, 7],
... [1, 10, 13, 21, 8, 24, 16, 5, 8],
... [0, 5, 4, 13, 2, 24, 17, 6, 7]
... ])
>>> np.linalg.matrix_rank(A)
np.int64(5)
```

### 9.2.4 LU-Zerlegung

Eine LU-Zerlegung kann mit der Funktion linalg .lu durchgeführt werden. Dazu betrachten wir nochmals das Zahlenbeispiel 1.2.0.18.

**Python-code 9.2.4.1: LU-Zerlegung**

```
>>> A = np.array([
... [1, 0, 5, 0, 4, 0, 0, 1, 0],
... [0, 0, 0, 5, 0, 4, 3, 0, 2],
... [1, 5, 9, 3, 6, 1, 0, 1, 0],
... [1, 10, 13, 11, 8, 6, 3, 1, 2],
... [0, 5, 4, 18, 2, 18, 12, 2, 7],
... [1, 10, 13, 21, 8, 24, 16, 5, 8],
... [0, 5, 4, 13, 2, 24, 17, 6, 7]
... ])
```

```
>>> P, L, U = linalg.lu(A)
>>> P
array([[1., 0., 0., 0., 0., 0., 0.],
       [0., 0., 0., 0., 1., 0., 0.],
       [0., 0., 1., 0., 0., 0., 0.],
       [0., 1., 0., 0., 0., 0., 0.],
       [0., 0., 0., 1., 0., 0., 0.],
       [0., 0., 0., 0., 0., 0., 1.],
       [0., 0., 0., 0., 0., 1., 0.]])
>>> L
array([[1. , 0. , 0. , 0. , 0. , 0. , 0. ],
       [1. , 1. , 0. , 0. , 0. , 0. , 0. ],
       [1. , 0.5, 1. , 0. , 0. , 0. , 0. ],
       [0. , 0.5, 0. , 1. , 0. , 0. , 0. ],
       [0. , 0. , 0. , 0.4, 1. , 0. , 0. ],
       [0. , 0.5, 0. , 0.6, 0. , 1. , 0. ],
       [1. , 1. , 0. , 0.8, 0. , 0.5, 1. ]])
>>> U
array([[ 1. ,  0. ,  5. ,  0. ,  4. ,  0. ,  0. ,  1. ,  0. ],
       [ 0. , 10. ,  8. , 11. ,  4. ,  6. ,  3. ,  0. ,  2. ],
       [ 0. ,  0. ,  0. , -2.5,  0. , -2. , -1.5,  0. , -1. ],
       [ 0. ,  0. ,  0. , 12.5,  0. , 15. , 10.5,  2. ,  6. ],
       [ 0. ,  0. ,  0. ,  0. ,  0. , -2. , -1.2, -0.8, -0.4],
       [ 0. ,  0. ,  0. ,  0. ,  0. , 12. ,  9.2,  4.8,  2.4],
       [ 0. ,  0. ,  0. ,  0. ,  0. ,  0. ,  0. ,  0. ,  0. ]])
>>> P @ L @ U
array([[ 1.,  0.,  5.,  0.,  4.,  0.,  0.,  1.,  0.],
       [ 0.,  0.,  0.,  5.,  0.,  4.,  3.,  0.,  2.],
       [ 1.,  5.,  9.,  3.,  6.,  1.,  0.,  1.,  0.],
       [ 1., 10., 13., 11.,  8.,  6.,  3.,  1.,  2.],
       [ 0.,  5.,  4., 18.,  2., 18., 12.,  2.,  7.],
       [ 1., 10., 13., 21.,  8., 24., 16.,  5.,  8.],
       [ 0.,  5.,  4., 13.,  2., 24., 17.,  6.,  7.]])
```

### 9.2.5 QR-Zerlegung

Eine QR-Zerlegung mittels Householder Transformationen kann ebenfalls durchgeführt werden. Zur Veranschaulichung verwenden wir 1.8.0.5.

**Python-code 9.2.5.1: QR-Zerlegung**

```
>>> A = np.array([[1, 1], [2, 0], [2, 0]])
>>> Q, R = linalg.qr(A)
>>> Q
array([[-3.33333333e-01,  9.42809042e-01, -7.40148683e-17],
       [-6.66666667e-01, -2.35702260e-01, -7.07106781e-01],
       [-6.66666667e-01, -2.35702260e-01,  7.07106781e-01]])
>>> R
array([[-3.        , -0.33333333],
       [ 0.        ,  0.94280904],
```

```
       [ 0.        ,  0.        ]])
>>> Q @ R
array([[ 1.00000000e+00,  1.00000000e+00],
       [ 2.00000000e+00, -2.20346637e-16],
       [ 2.00000000e+00, -1.10009756e-17]])
```

### 9.2.6 Weitere Zerlegungen

Auch für die weiteren Zerlegungen, die in dieser Vorlesung besprochen werden, gibt es analoge SciPy-Funktionen, welche hier nicht explizit besprochen werden.

Beispielsweise:

- *LDL-Zerlegung:* linalg . ldl
- *Cholesky-Zerlegung:* linalg . cholesky
- *Polar-Zerlegung:* linalg . polar
- *Schur-Zerlegung:* linalg . schur

Eine vollständige Auflistung findet sich in der SciPy-Dokumentation.

### 9.2.7 Gauss-Elimination

In diesem Abschnitt finden sich zwei Programme, welche die Gauss-Elimination mit didaktischem Fokus durchführen. Das Ziel dieses Beispiels ist einerseits zu veranschaulichen wie die Gauss-Elimination in Python implementiert werden kann, andererseits können diese Programme genutzt werden, um zu lernen, wie die Gauss-Elimination Schritt für Schritt durchgeführt werden kann.

**Python-code 9.2.7.1: Gauss-Elimination: Betragsgrösster Pivot**

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
Didaktische Gauss-Elimination mit Zeilenvertauschung für
   rechteckige Matrizen.

Besonderheit dieser Version:
  - Die Rechnung wird intern mit fractions.Fraction durchgeführt.
  - Dadurch werden Multiplikatoren exakt als Brüche ausgegeben.
  - Die Zeilenoperation wird didaktisch nur als Addition
     geschrieben:
         Zeile i := Zeile i + c * Zeile p
    mit c = -m.
  - Der Multiplikator m wird explizit als Bruch dargestellt,
     wobei der Pivot
    unter der Bruchlinie steht.

Konvention am Ende:
    P @ A = L @ U

Für A mit m Zeilen und n Spalten gilt:
```

```
  - P ist m x m
  - L ist m x m
  - U ist m x n
"""

import numpy as np
from fractions import Fraction

def to_fraction(x):
    """Wandelt Zahlen robust in Fraction um."""
    if isinstance(x, Fraction):
        return x
    # Bei NumPy-Zahlen zuerst nach Python-Zahl konvertieren.
    if hasattr(x, "item"):
        x = x.item()
    # Für ganze Zahlen direkt exakt.
    if isinstance(x, int):
        return Fraction(x, 1)
    # Für Floats über String, damit z. B. 0.1 nicht als binärer
      Floatbruch erscheint.
    return Fraction(str(x)).limit_denominator(10**9)

def matrix_to_fraction(A):
    """Konvertiert eine NumPy-Matrix oder Liste von Listen in
      Fraction-Einträge."""
    arr = np.array(A, dtype=object)
    return [[to_fraction(arr[i, j]) for j in range(arr.shape[1])]
      for i in range(arr.shape[0])]

def zeros(rows, cols):
    return [[Fraction(0, 1) for _ in range(cols)] for _ in
      range(rows)]

def identity(n):
    M = zeros(n, n)
    for i in range(n):
        M[i][i] = Fraction(1, 1)
    return M

def copy_matrix(A):
    return [row[:] for row in A]

def frac_str(x):
    """Gibt einen Fraction-Wert schön aus, z. B. -1/2 oder 3."""
    x = to_fraction(x)
    if x.denominator == 1:
        return str(x.numerator)
    return f"{x.numerator}/{x.denominator}"

def print_matrix(name, M):
```

```
    """Matrix in der Konsole lesbar ausgeben."""
    print(f"\n{name} =")
    if not M:
        print("[]")
        return
    string_matrix = [[frac_str(v) for v in row] for row in M]
    widths = [max(len(string_matrix[i][j]) for i in
       range(len(M))) for j in range(len(M[0]))]
    for row in string_matrix:
        print("[ " + "  ".join(row[j].rjust(widths[j]) for j in
           range(len(row))) + " ]")

def permutation_matrix(m, r1, r2):
    """Elementare Permutationsmatrix, die die Zeilen r1 und r2
       vertauscht."""
    S = identity(m)
    S[r1], S[r2] = S[r2], S[r1]
    return S

def matmul(A, B):
    """Matrixprodukt für Fraction-Matrizen."""
    rows_A = len(A)
    cols_A = len(A[0])
    rows_B = len(B)
    cols_B = len(B[0])
    if cols_A != rows_B:
        raise ValueError("Matrixgrössen passen nicht für
           Multiplikation.")
    C = zeros(rows_A, cols_B)
    for i in range(rows_A):
        for j in range(cols_B):
            C[i][j] = sum(A[i][k] * B[k][j] for k in
               range(cols_A))
    return C

def max_abs_difference(A, B):
    """Maximale absolute Differenz zweier gleich grosser
       Fraction-Matrizen."""
    max_diff = Fraction(0, 1)
    for i in range(len(A)):
        for j in range(len(A[0])):
            diff = abs(A[i][j] - B[i][j])
            if diff > max_diff:
                max_diff = diff
    return max_diff

def print_multiplier_definition(i, pivot_row, col, numerator,
   pivot, multiplier):
    """
    Druckt den Multiplikator als Bruch.
```

```
        Die Zeilen- und Spaltennummern werden didaktisch 1-basiert
           ausgegeben.
        """
        i1 = i + 1
        p1 = pivot_row + 1
        c1 = col + 1

        print("\nDefinition des Multiplikators:")
        print(f"m = Wert unter dem Pivot / Pivot")
        print(f"m = U[{i1},{c1}] / U[{p1},{c1}]")

        top_label = f"U[{i1},{c1}]"
        bottom_label = f"U[{p1},{c1}]  <- Pivot"
        top_value = frac_str(numerator)
        bottom_value = frac_str(pivot)
        line_len = max(len(top_label), len(bottom_label),
           len(top_value), len(bottom_value), 12)

        print("      " + top_label.center(line_len) + "      " +
           top_value.center(line_len))
        print("m =   " + ("-" * line_len) + "  =   " + ("-" *
           line_len) + f"  = {frac_str(multiplier)}")
        print("      " + bottom_label.center(line_len) + "      " +
           bottom_value.center(line_len))

    def gauss_lu_pivot_rectangular(A):
        """
        Gauss-Elimination mit partieller Pivotisierung für
           rechteckige Matrizen.

        Rückgabe:
            P, L, U mit P @ A = L @ U

        Hinweis:
            Die Ausgabe der Zeilennummern ist didaktisch 1-basiert.
            Python selbst indexiert intern 0-basiert.
        """
        A = matrix_to_fraction(A)
        m = len(A)
        n = len(A[0]) if m > 0 else 0

        U = copy_matrix(A)
        L = identity(m)
        P = identity(m)

        pivot_row = 0

        print("Ausgangsmatrix")
        print_matrix("A", A)
        print_matrix("P", P)
```

```
    print_matrix("L", L)
    print_matrix("U", U)

    for col in range(n):
        if pivot_row >= m:
            break

        print("\n" + "=" * 86)
        print(f"Spalte {col + 1}: Pivot ab Zeile {pivot_row + 1}
            suchen")

        # Pivotzeile suchen: grösster Betrag in aktueller Spalte
            ab pivot_row
        pivot = max(range(pivot_row, m), key=lambda r:
            abs(U[r][col]))
        pivot_value = U[pivot][col]

        if pivot_value == 0:
            print(f"In Spalte {col + 1} gibt es unterhalb von
                Zeile {pivot_row + 1} keinen brauchbaren Pivot.")
            print("Diese Spalte wird übersprungen.")
            continue

        # Zeilenvertauschung, falls nötig
        if pivot != pivot_row:
            S = permutation_matrix(m, pivot_row, pivot)
            U[pivot_row], U[pivot] = U[pivot], U[pivot_row]
            P[pivot_row], P[pivot] = P[pivot], P[pivot_row]

            # Bereits gebildete L-Spalten müssen mitgetauscht
                werden.
            # Nur die Spalten 0 bis pivot_row-1 enthalten schon
                gültige Multiplikatoren.
            if pivot_row > 0:
                for j in range(pivot_row):
                    L[pivot_row][j], L[pivot][j] = L[pivot][j],
                        L[pivot_row][j]

            print(f"\nZeilenvertauschung: Zeile {pivot_row + 1}
                <-> Zeile {pivot + 1}")
            print_matrix("S, Permutationsmatrix dieses Tauschs",
                S)
            print_matrix("P, bisherige Gesamtpermutation", P)
            print_matrix("L, nach Anpassung bereits gebildeter
                Spalten", L)
            print_matrix("U, nach Zeilenvertauschung", U)
        else:
            print(f"\nKeine Zeilenvertauschung nötig. Pivot ist
                U[{pivot_row + 1},{col + 1}] =
                {frac_str(U[pivot_row][col])}")
```

```
        # Einträge unterhalb des Pivots eliminieren
        for i in range(pivot_row + 1, m):
            if U[i][col] == 0:
                print(f"\nZeile {i + 1}: Eintrag in Spalte {col +
                    1} ist bereits 0. Keine Elimination nötig.")
                continue

            numerator = U[i][col]
            pivot_value = U[pivot_row][col]
            multiplier = numerator / pivot_value
            coefficient = -multiplier
            L[i][pivot_row] = multiplier

            # Elementarmatrix E bewirkt:
            # Zeile_i := Zeile_i + coefficient * Zeile_pivot_row
            E = identity(m)
            E[i][pivot_row] = coefficient

            print_multiplier_definition(i, pivot_row, col,
                numerator, pivot_value, multiplier)
            print(f"\nAdditionsfaktor c = -m =
                {frac_str(coefficient)}")
            print(f"Operation: Zeile {i + 1} := Zeile {i + 1} +
                ({frac_str(coefficient)})*Zeile {pivot_row + 1}")

            # Zeilenoperation direkt auf U ausführen
            for j in range(n):
                U[i][j] = U[i][j] + coefficient * U[pivot_row][j]

            print_matrix("E, Elementarmatrix dieses
                Eliminationsschritts", E)
            print_matrix("U, nach diesem Eliminationsschritt", U)
            print_matrix("L, mit gespeichertem Multiplikator m",
                L)
            print_matrix("P, unverändert bei reiner Elimination",
                P)

        pivot_row += 1

    print("\n" + "=" * 86)
    print("Endergebnis")
    print_matrix("P", P)
    print_matrix("L", L)
    print_matrix("U", U)

    print("\nKontrolle der Zerlegung: P @ A und L @ U sollten
        gleich sein.")
    PA = matmul(P, A)
    LU = matmul(L, U)
```

```
    print_matrix("P @ A", PA)
    print_matrix("L @ U", LU)
    print(f"\nMaximaler Fehler ||P@A - L@U||_max =
       {frac_str(max_abs_difference(PA, LU))}")

    return P, L, U

if __name__ == "__main__":
    # Beispiel: nicht quadratische Matrix mit 7 Zeilen und 9
       Spalten
    A = np.array([
        [1, 0, 5, 0, 4, 0, 0, 1, 0],
        [0, 0, 0, 5, 0, 4, 3, 0, 2],
        [1, 5, 9, 3, 6, 1, 0, 1, 0],
        [1, 10, 13, 11, 8, 6, 3, 1, 2],
        [0, 5, 4, 18, 2, 18, 12, 2, 7],
        [1, 10, 13, 21, 8, 24, 16, 5, 8],
        [0, 5, 4, 13, 2, 24, 17, 6, 7]
    ], dtype=float)

    gauss_lu_pivot_rectangular(A)
```

Die folgende Programmvariante unterscheidet sich von der letzten vor allem durch die Strategie zur Auswahl des Pivot-Elements. Anstatt das betragsgrösste Element als Pivot zu wählen, wird versucht möglichst einfache Brüche zu behalten. Der Kürze halber werden nur jene Funktionen und Abschnitte inkludiert, welche sich von der vorherigen Variante unterscheiden. Die vollständige Version ist auf CodeExpert verfügbar.

**Python-code 9.2.7.2: Gauss-Elimination: Einfache Brüche**

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
Didaktische Gauss-Elimination mit Zeilenvertauschung für
   rechteckige Matrizen.

Besonderheit dieser Version:
  - Die Rechnung wird intern mit fractions.Fraction durchgeführt.
  - Dadurch werden Multiplikatoren exakt als Brüche ausgegeben.
  - Die Zeilenoperation wird didaktisch nur als Addition
     geschrieben:
        Zeile i := Zeile i + c * Zeile p
    mit c = -m.
  - Der Multiplikator m wird explizit als Bruch dargestellt,
     wobei der Pivot
    unter der Bruchlinie steht.
  - Der Pivot wird nicht einfach nach grösstem Betrag gewählt,
     sondern so,
    dass die Multiplikatoren in diesem Eliminationsschritt
       möglichst einfach
    werden. Insbesondere werden Pivots bevorzugt, die zu keinen
```

```
        oder möglichst
    wenigen echten Brüchen führen.

Konvention am Ende:
    P @ A = L @ U

Für A mit m Zeilen und n Spalten gilt:
  - P ist m x m
  - L ist m x m
  - U ist m x n
"""

def pivot_score_for_simple_fractions(U, pivot_row, col,
   candidate_row):
    """
    Bewertet einen möglichen Pivot nach didaktischer Einfachheit
       der Multiplikatoren.

    Ziel: Für die Multiplikatoren
        m_i = U[i,col] / Pivot
    möglichst wenige echte Brüche erzeugen.

    Ein kleinerer Score ist besser. Bewertet werden der Reihe
       nach:
      1. Anzahl echter Brüche, also Nenner != 1
      2. Summe der Nenner aller nicht-ganzzahligen Multiplikatoren
      3. Summe der Beträge der Zähler nicht-ganzzahliger
         Multiplikatoren
      4. Anzahl der nichtverschwindenden Multiplikatoren
      5. Betrag des Pivots, negativ verwendet, damit grössere
         Pivotbeträge
         bei Gleichstand bevorzugt werden
    """
    m = len(U)
    pivot_value = U[candidate_row][col]
    if pivot_value == 0:
        return None

    multipliers = []
    for r in range(pivot_row, m):
        if r == candidate_row:
            continue
        entry = U[r][col]
        if entry == 0:
            multipliers.append(Fraction(0, 1))
        else:
            multipliers.append(entry / pivot_value)

    true_fractions = [q for q in multipliers if q != 0 and
       q.denominator != 1]
```

```
    nonzero = [q for q in multipliers if q != 0]

    return (
        len(true_fractions),
        sum(q.denominator for q in true_fractions),
        sum(abs(q.numerator) for q in true_fractions),
        len(nonzero),
        -abs(pivot_value),
        candidate_row
    )

def choose_pivot_for_simple_fractions(U, pivot_row, col):
    """
    Wählt die Pivotzeile so, dass im aktuellen
       Eliminationsschritt möglichst
    einfache Multiplikatoren entstehen.
    """
    candidates = []
    for r in range(pivot_row, len(U)):
        if U[r][col] != 0:
            score = pivot_score_for_simple_fractions(U,
               pivot_row, col, r)
            candidates.append((score, r))

    if not candidates:
        return None

    candidates.sort(key=lambda pair: pair[0])
    return candidates[0][1]

def explain_pivot_choice(U, pivot_row, col, chosen):
    """Gibt didaktisch aus, warum ein Pivot gewählt wurde."""
    print("\nPivotwahl nach Einfachheit der Multiplikatoren:")
    print("Für jede mögliche Pivotzeile wird geprüft, welche
       Multiplikatoren entstehen.")
    print("Bevorzugt wird: möglichst wenige echte Brüche, dann
       kleine Nenner.")

    rows = []
    for r in range(pivot_row, len(U)):
        if U[r][col] == 0:
            continue
        pivot_value = U[r][col]
        multipliers = []
        for s in range(pivot_row, len(U)):
            if s == r:
                continue
            entry = U[s][col]
            if entry != 0:
                multipliers.append(entry / pivot_value)
```

```
            else:
                multipliers.append(Fraction(0, 1))
        true_fraction_count = sum(1 for q in multipliers if q !=
            0 and q.denominator != 1)
        denominator_sum = sum(q.denominator for q in multipliers
            if q != 0 and q.denominator != 1)
        rows.append((r, pivot_value, multipliers,
            true_fraction_count, denominator_sum))

    for r, pivot_value, multipliers, true_fraction_count,
        denominator_sum in rows:
        marker = "  <-- gewählt" if r == chosen else ""
        mult_text = ", ".join(frac_str(q) for q in multipliers)
        if mult_text == "":
            mult_text = "keine"
        print(
            f"  Pivotkandidat Zeile {r + 1}: Pivot =
                {frac_str(pivot_value)}, "
            f"Multiplikatoren = [{mult_text}], "
            f"echte Brüche = {true_fraction_count}, Nennersumme =
                {denominator_sum}{marker}"
        )

def gauss_lu_pivot_rectangular(A):
    """
    Gauss-Elimination mit didaktischer Pivotwahl für rechteckige
        Matrizen.

    Rückgabe:
        P, L, U mit P @ A = L @ U

    Hinweis:
        Die Ausgabe der Zeilennummern ist didaktisch 1-basiert.
        Python selbst indexiert intern 0-basiert.
    """
    A = matrix_to_fraction(A)
    m = len(A)
    n = len(A[0]) if m > 0 else 0

    U = copy_matrix(A)
    L = identity(m)
    P = identity(m)

    pivot_row = 0

    print("Ausgangsmatrix")
    print_matrix("A", A)
    print_matrix("P", P)
    print_matrix("L", L)
    print_matrix("U", U)
```

```
    for col in range(n):
        if pivot_row >= m:
            break

        print("\n" + "=" * 86)
        print(f"Spalte {col + 1}: Pivot ab Zeile {pivot_row + 1}
            suchen")

        pivot = choose_pivot_for_simple_fractions(U, pivot_row,
            col)

        if pivot is None:
            print(f"In Spalte {col + 1} gibt es unterhalb von
                Zeile {pivot_row + 1} keinen brauchbaren Pivot.")
            print("Diese Spalte wird übersprungen.")
            continue

        explain_pivot_choice(U, pivot_row, col, pivot)

        # Zeilenvertauschung, falls nötig
        if pivot != pivot_row:
            S = permutation_matrix(m, pivot_row, pivot)
            U[pivot_row], U[pivot] = U[pivot], U[pivot_row]
            P[pivot_row], P[pivot] = P[pivot], P[pivot_row]

            # Bereits gebildete L-Spalten müssen mitgetauscht
                werden.
            # Nur die Spalten 0 bis pivot_row-1 enthalten schon
                gültige Multiplikatoren.
            if pivot_row > 0:
                for j in range(pivot_row):
                    L[pivot_row][j], L[pivot][j] = L[pivot][j],
                        L[pivot_row][j]

            print(f"\nZeilenvertauschung: Zeile {pivot_row + 1}
                <-> Zeile {pivot + 1}")
            print_matrix("S, Permutationsmatrix dieses Tauschs",
                S)
            print_matrix("P, bisherige Gesamtpermutation", P)
            print_matrix("L, nach Anpassung bereits gebildeter
                Spalten", L)
            print_matrix("U, nach Zeilenvertauschung", U)
        else:
            print(f"\nKeine Zeilenvertauschung nötig. Pivot ist
                U[{pivot_row + 1},{col + 1}] =
                {frac_str(U[pivot_row][col])}")

        # Einträge unterhalb des Pivots eliminieren
        for i in range(pivot_row + 1, m):
```

```
            if U[i][col] == 0:
                print(f"\nZeile {i + 1}: Eintrag in Spalte {col +
                    1} ist bereits 0. Keine Elimination nötig.")
                continue

            numerator = U[i][col]
            pivot_value = U[pivot_row][col]
            multiplier = numerator / pivot_value
            coefficient = -multiplier
            L[i][pivot_row] = multiplier

            # Elementarmatrix E bewirkt:
            # Zeile_i := Zeile_i + coefficient * Zeile_pivot_row
            E = identity(m)
            E[i][pivot_row] = coefficient

            print_multiplier_definition(i, pivot_row, col,
                numerator, pivot_value, multiplier)
            print(f"\nAdditionsfaktor c = -m =
                {frac_str(coefficient)}")
            print(f"Operation: Zeile {i + 1} := Zeile {i + 1} +
                ({frac_str(coefficient)})*Zeile {pivot_row + 1}")

            # Zeilenoperation direkt auf U ausführen
            for j in range(n):
                U[i][j] = U[i][j] + coefficient * U[pivot_row][j]

            print_matrix("E, Elementarmatrix dieses
                Eliminationsschritts", E)
            print_matrix("U, nach diesem Eliminationsschritt", U)
            print_matrix("L, mit gespeichertem Multiplikator m",
                L)
            print_matrix("P, unverändert bei reiner Elimination",
                P)

        pivot_row += 1

    print("\n" + "=" * 86)
    print("Endergebnis")
    print_matrix("P", P)
    print_matrix("L", L)
    print_matrix("U", U)

    print("\nKontrolle der Zerlegung: P @ A und L @ U sollten
        gleich sein.")
    PA = matmul(P, A)
    LU = matmul(L, U)
    print_matrix("P @ A", PA)
    print_matrix("L @ U", LU)
    print(f"\nMaximaler Fehler ||P@A - L@U||_max =
```

```
        {frac_str(max_abs_difference(PA, LU))}")

    return P, L, U
```

## 9.3 Lineare Räume

### 9.3.1 Bild und Kern einer Matrix

Wir verwenden die Beispiele 2.2.0.12 und 2.2.0.13, um zu zeigen, wie mit Python ein Erzeugendensystem des Kerns bzw. des Bilds einer Matrix gefunden werden kann.

**Bemerkung 9.3.1.1 (Orthonormale Basis)** [1] Tatsächlich nutzt SciPy hier die in Kapitel 8 beschriebene Singulärwertzerlegung und den Fundamentalsatz der linearen Algebra (Satz 8.1.0.5), um sogar orthonormale Basen zu erhalten. ⌟

Die Funktion `linalg.orth` liefert ein Erzeugendensystem des Bildes, während `linalg.null_space` eines des Kerns liefert.

**Python-code 9.3.1.2: Bild und Kern einer Matrix**

```
>>> A = np.array([
... [1, 1, 2, 3],
... [2, 2, 8, 10],
... [3, 3, 10, 13]
... ])
>>> linalg.orth(A)
array([[ 0.17606524,  0.79728771],
       [ 0.6024388 , -0.55112083],
       [ 0.77850403,  0.24616689]])
>>> linalg.null_space(A)
array([[-0.27921667, -0.72252201],
       [ 0.72468056,  0.27356552],
       [ 0.44546388, -0.44895649],
       [-0.44546388,  0.44895649]])
```

[1] Wenn Sie diesen Abschnitt gemeinsam mit Kapitel 2 lesen, werden Ihnen diese Begriffe noch nicht vertraut sein. Sie dürfen diese Bemerkung ignorieren.

## 9.4 Normen und Skalarprodukte

### 9.4.1 Vektor- und Matrixnormen

Die gängigsten Vektor- und Matrixnormen sind in der Funktion linalg .norm abgebildet. Dazu zählen insbesondere die Frobenius-, Nuklear- und $p$-Norm.

**Python-code 9.4.1.1: Vektor- und Matrixnormen**

```
>>> A = np.array([
... [1, -2, -3],
... [2, 3, -1]
... ])
>>> linalg.norm(A, ord=np.inf)
np.float64(6.0)
>>> linalg.norm(A, ord=1)
np.float64(5.0)
```

### 9.4.2 Skalarprodukt

Bei der Berechnung des (euklidischen) Skalarprodukts mit NumPy wird dieselbe Konvention wie in dieser Vorlesung verwendet. Das heisst, dass das Skalarprodukt linear im zweiten Argument ist.

Konkret kann man das Skalarprodukt zweier Vektoren mit np.vdot berechnen. Im reellen Fall kann auch np.dot (oder äquivalent der @-Operator) verwendet werden. Hierbei wird keine komplexe Konjugation des ersten Arguments durchgeführt.

**Python-code 9.4.2.1: Skalarprodukt**

```
>>> a = np.array([1.j, 2.j, 3.j]) # j ist die imaginäre Einheit
>>> b = np.array([4., 5., 6.])
>>> a @ b
np.complex128(32j)
>>> np.dot(a, b)
np.complex128(32j)
>>> np.vdot(a, b)
np.complex128(-32j)
```

## 9.5 Ausgleichsrechnung

Um das Problem der Ausgleichsrechnung mit SciPy zu lösen, steht die Funktion linalg . lstsq zur Verfügung. Zu beachten ist, dass hierbei die Singulärwertzerlegung (Abschnitt 8.2.1) als Lösungsmethode und nicht etwa die Normalengleichung (Abschnitt 5.2) oder die QR-Zerlegung (Abschnitt 5.3) verwendet wird.

Als einfaches Beispiel dient 5.3.0.1.

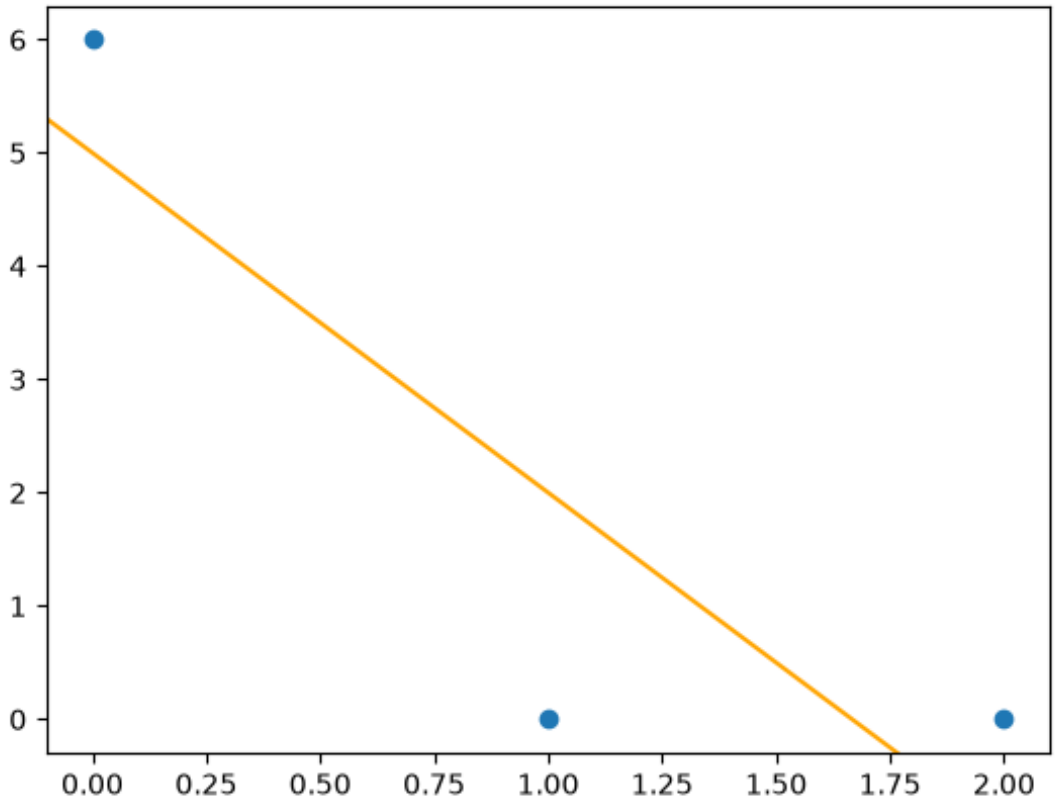
**Python-code 9.5.0.1: Ausgleichsrechnung**

```
import numpy as np
from scipy import linalg
import matplotlib.pyplot as plt

x = np.array([0, 1, 2])
y = np.array([6, 0, 0])
A = np.array([[1, 0],
              [1, 1],
              [1, 2]])

c, resid, rank, sigma = linalg.lstsq(A, y)

plt.scatter(x, y)
plt.axline((0, c[0]), slope=c[1], color ="orange")
plt.show()
```



**Bemerkung 9.5.0.2 (Ausgleichsrechnung)** Auch wenn `linalg.lstsq` das Problem der Ausgleichsrechnung, wie es in dieser Vorlesung behandelt wurde, direkt und korrekt löst, gibt es für häufige Probleme der Ausgleichsrechnung andere Funktionen, welche oft benutzerfreundlicher sind.

So hätten wir für das vorhergehende Beispiel auch `scipy.stats.linregress` oder `numpy.polynomial.polynomial.polyfit` verwenden können und so die Matrixkonstruktion nicht explizit durchführen müssen. ⌟

### 9.5.1 Pseudoinverse

Die Lösung der Ausgleichsrechnung mithilfe der Singulärwertzerlegung entspricht, wie in Abschnitt 8.2.2 besprochen, der Multiplikation mit der Pseudoinversen. Diese kann mit `linalg.pinv` explizit berechnet werden.

## 9.6 Determinante

Die Berechnung der Determinante einer Matrix erfolgt in SciPy mithilfe der Funktion linalg.det. Exemplarisch wird das Beispiel 6.1.0.8 verwendet. Wie bereits im entsprechenden Abschnitt erwähnt, wird zur Berechnung der Determinante eine LU-Zerlegung durchgeführt und das Produkt der Diagonalelemente von $U$ ausgegeben.

**Python-code 9.6.0.1: Determinante**

```
>>> A = np.array([
... [ 4, 3, -1, -2],
... [ 2, 1, -1, 0],
... [-2, -2, 0, 2],
... [ 5, 1, 0, 2]
... ])
>>> linalg.det(A)
np.float64(0.0)
```

## 9.7 Eigenwertproblem

Das Eigenwertproblem kann ebenfalls mit SciPy numerisch gelöst werden. Als Beispiel betrachten wir die strikt diagonaldominante Matrix aus Beispiel 7.2.0.13.

**Python-code 9.7.0.1: Eigenwertproblem**

```
>>> A = np.array([
... [4, 1, 0, 0, 0],
... [1, 4, 1, 0, 0],
... [0, 1, 4, 1, 0],
... [0, 0, 1, 4, 1],
... [0, 0, 0, 1, 4]
... ])
>>> la, v = linalg.eig(A)
>>> la
array([2.26794919+0.j, 3.        +0.j, 4.        +0.j,
   5.73205081+0.j,
       5.        +0.j])
>>> v
array([[-2.88675135e-01, -5.00000000e-01,  5.77350269e-01,
         2.88675135e-01, -5.00000000e-01],
       [ 5.00000000e-01,  5.00000000e-01,  6.66458787e-16,
         5.00000000e-01, -5.00000000e-01],
       [-5.77350269e-01,  6.08319210e-16, -5.77350269e-01,
         5.77350269e-01, -6.54721069e-16],
       [ 5.00000000e-01, -5.00000000e-01, -5.87132450e-16,
         5.00000000e-01,  5.00000000e-01],
       [-2.88675135e-01,  5.00000000e-01,  5.77350269e-01,
         2.88675135e-01,  5.00000000e-01]])
>>> v @ np.diag(la) @ linalg.inv(v)
array([[ 4.00000000e+00+0.j,  1.00000000e+00+0.j,
   1.32898469e-15+0.j,
         8.32667268e-16+0.j,  5.55111512e-17+0.j],
       [ 1.00000000e+00+0.j,  4.00000000e+00+0.j,
          1.00000000e+00+0.j,
         6.10622664e-16+0.j,  7.21644966e-16+0.j],
       [ 7.20338136e-16+0.j,  1.00000000e+00+0.j,
          4.00000000e+00+0.j,
         1.00000000e+00+0.j,  1.04821970e-15+0.j],
       [ 2.22044605e-16+0.j, -3.88578059e-16+0.j,
          1.00000000e+00+0.j,
         4.00000000e+00+0.j,  1.00000000e+00+0.j],
       [ 1.11022302e-15+0.j,  2.05391260e-15+0.j,
          2.06230007e-15+0.j,
         1.00000000e+00+0.j,  4.00000000e+00+0.j]])
```

Wie bereits aus dem Satz 7.2.0.32 bekannt ist, gibt es Matrizen, die nicht diagonalisierbar sind. Es ist interessant zu betrachten, wie sich eine solche Matrix beim numerischen Lösen des Eigenwertproblems verhält. Als Beispiel betrachten wir dieselbe Matrix wie in Beispiel 7.2.0.30.

Bei der Betrachtung der errechneten Eigenvektoren wird klar, dass es zu Problemen gekommen ist, denn wir

erhalten zwei Eigenvektoren, die praktisch linear abhängig sind. Die absolute Gewissheit erhalten wir, wenn wir versuchen, die ursprüngliche Matrix aus der vermeintlichen Diagonalisierung zu rekonstruieren.

Zu beachten ist, dass NumPy bei der Berechnung der Inversen berechtigterweise eine Warnung ausgibt, denn die Matrix ist äusserst schlecht konditioniert. rcond gibt hierbei den Kehrwert der Konditionszahl an. rcond = 3.172065784643304e−17 bedeutet also, dass die Matrix beinahe singulär ist.

**Python-code 9.7.0.2: Eigenwertproblem bei nicht diagonalisierbarer Matrix**

```
>>> A = np.array([
... [1, 2, 0, 1],
... [0, -1, 0, 1],
... [0, 1, 1, 1],
... [0, 0, 0, -1]
... ])
>>> la, v = linalg.eig(A)
>>> la
array([ 1.+0.j,  1.+0.j, -1.+0.j, -1.+0.j])
>>> v
array([[ 0.00000000e+00,  1.00000000e+00, -6.66666667e-01,
         6.66666667e-01],
       [ 0.00000000e+00,  0.00000000e+00,  6.66666667e-01,
        -6.66666667e-01],
       [ 1.00000000e+00,  0.00000000e+00, -3.33333333e-01,
         3.33333333e-01],
       [ 0.00000000e+00,  0.00000000e+00,  0.00000000e+00,
         1.48029737e-16]])
>>> np.real(v @ np.diag(la) @ linalg.inv(v))
<stdin>:1: LinAlgWarning: An ill-conditioned matrix detected:
   slice 0 has rcond = 3.172065784643304e-17.
array([[ 1.  ,  2.  ,  0.  ,  1.  ],
       [ 0.  , -1.  ,  0.  , -0.25],
       [ 0.  ,  1.  ,  1.  ,  0.5 ],
       [ 0.  ,  0.  ,  0.  , -1.  ]])
```

### 9.7.1 Konditionszahl

Wie wir im letzten Beispiel, aber auch im Abschnitt 4.2, gesehen haben, ist die Konditionszahl eine sehr wichtige Grösse in der numerischen linearen Algebra.

Die Konditionszahl kann mit `linalg.cond` berechnet werden. Das Ergebnis ist abhängig von der verwendeten Norm. Weitere Informationen hierzu finden sich in Abschnitt 9.4 und in der SciPy-Dokumentation.

### 9.7.2 Zirkulante Matrix

Eine zirkulante Matrix kann mit `linalg.circulant` erstellt werden. Dabei entspricht die erste Spalte dem übergebenen Array.

### 9.7.3 Fourier-Matrix

Die $n \times n$ Fourier-Matrix kann mit `linalg.dft` erstellt werden. Diese verwendet die gleiche Konvention hinsichtlich der Normierung, wie sie in dieser Vorlesung behandelt wurde. Somit sind die Einträge Potenzen

von $\omega_n = e^{-2\pi i/n}$.

### 9.7.4 Faltung

Die Berechnungen aus Beispiel 7.5.0.13 zur Betrachtung der Multiplikation als Faltung können auf einfache Weise in SciPy umgesetzt werden. Für die Faltung steht die Funktion signal .convolve zur Verfügung. Für die Berechnung der zyklischen Faltung stehen zwei Möglichkeiten zur Verfügung. Einerseits kann eine Multiplikation mit einer zirkulanten Matrix verwendet werden. Andererseits kann das Faltungstheorem angewandt werden. Die zweite Variante ist bei grösseren Arrays deutlich zu bevorzugen, da sie mit einer asymptotischen Laufzeit von $\Theta(n \log n)$ deutlich schneller ist als die direkte Multiplikation mit einer asymptotischen Laufzeit von $\Theta(n^2)$.

**Python-code 9.7.4.1: Faltung**

```
>>> from scipy import linalg, signal
>>> from scipy.fft import fft, ifft
>>> def cyclic_convolution(signal, kernel):
... return linalg.circulant(signal) @ kernel
...
>>> def fft_cyclic_convolution(signal, kernel):
... return np.real(ifft(fft(signal) * fft(kernel)))
...
>>> signal.convolve([1, 2, 4], [3, 0, 5])
array([ 3,  6, 17, 10, 20])
>>> cyclic_convolution([1, 2, 4], [3, 0, 5])
array([13, 26, 17])
>>> fft_cyclic_convolution([1, 2, 4], [3, 0, 5])
array([13., 26., 17.])
```

## 9.8 Singulärwertzerlegung

Um eine Singulärwertzerlegung mit Python durchzuführen, kann die Funktion linalg .svd verwendet werden. Zur Veranschaulichung werden wir das Zahlenbeispiel 8.1.0.8 erneut betrachten.

**Python-code 9.8.0.1: Singulärwertzerlegung**

```
>>> A = np.array([[1, 1],
... [7, 7]])
>>> m, n = A.shape
>>> U, s, Vh = linalg.svd(A)
>>> Sig = linalg.diagsvd(s,m,n)
>>> U
array([[-0.14142136, -0.98994949],
       [-0.98994949,  0.14142136]])
>>> s
array([10.,  0.])
>>> Sig
array([[10.,  0.],
       [ 0.,  0.]])
>>> Vh
array([[-0.70710678, -0.70710678],
       [-0.70710678,  0.70710678]])
>>> U @ Sig @ Vh
array([[1., 1.],
       [7., 7.]])
```

### 9.8.1 Trends in den Daten

Für dieses Beispiel gehen wir in drei Schritten vor:

1. Zunächst initialisieren wir die Matrix A und zentrieren diese durch Abzug der spaltenweisen Mittelwerte.
2. Dann führen wir die Singulärwertzerlegung wie oben beschrieben durch.
3. Abschliessend geben wir nacheinander die Singulärwerte und Singulärvektoren aus und visualisieren die ersten beiden Singulärvektorpaare.

**Python-code 9.8.1.1: Filme**

```
import numpy as np
import matplotlib.pyplot as plt

A = np.array([[5,5,1,1,1,1], [5,4,1,1,1,1], [1,1,5,5,5,4],
   [1,1,4,5,5,5]], dtype=float)
print(f"{A=}")

B = A - A.mean(0)

print(f"{B=}")
UB,SB,VBT = np.linalg.svd(B)
```

```
for k  in range(3):
    print(f"{k=}")
    print(f"{SB[k]=}")
    print(f"{UB[:,k]=}")
    print(f"{VBT[k,:]=}")

fig, axes = plt.subplots(1, 2, figsize=(16, 7))
axes[0].scatter(UB[:,0], UB[:,1], alpha=0.6, s=50)
axes[0].set_title("User-Analyse (Matrix U)\n'Wer mag was?'")
axes[0].set_xlabel(r"Trend 1: $u_1$")
axes[0].set_ylabel("Trend 2: $u_2$")
axes[0].grid(True, alpha=0.3)

axes[1].scatter(VBT[0,:], VBT[1,:], alpha=0.6, s=70, marker = 'D')
axes[1].set_title("Film-Analyse (Matrix $V^T$)\n'Was passt wozu?'")
axes[1].set_xlabel(r"Trend 1: $v_1$")
axes[1].set_ylabel(r"Trend 2: $v_2$")
axes[1].grid(True, alpha=0.3)

# Visueller Vergleich: Hilfslinien durch den Ursprung
for ax in axes:
    ax.axhline(0, color='black', linestyle='--', alpha=0.3)
    ax.axvline(0, color='black', linestyle='--', alpha=0.3)

plt.suptitle("Die Dualität der SVD: Gleicher Raum für User und
   Produkte", fontsize=16)
plt.tight_layout()
plt.savefig('UandVmini.png')
plt.show()
```

# Liste der Symbole


$\|\mathbf{A}\|$, 124
$C^s$, 73
$C^\infty[a,b]$, 74
$C^s[a,b]$, 73
$L^2[0,1]$, 75
$L^2$, 75
$P_y x$, 136
$P^\perp_{\mathrm{Span}\{b_1,\ldots,b_k\}}$, 142
Bild $\mathscr{A}$, 80
$\mathscr{A}$, 55
$\mathscr{P}_n$, 75
$\langle\cdot,\cdot\rangle_{\mathbf{A}}$, 134
$\langle\cdot,\cdot\rangle$, 54
$\langle f,g\rangle$, 124
Kern $\mathbf{A}$, 83
$\|\mathbf{A}\|_1$, 125
$\|\mathbf{A}\|_2$, 125
$\|\mathbf{A}\|_F$, 124
$\|\mathbf{A}\|_\infty$, 124
$\|\mathbf{u}\|_{\mathbf{A}}$, 135
$\|\mathbf{x}\|_2$, 120
$\|\cdot\|_0$, 123
$\|\cdot\|_2$, 120, 123
$\|\cdot\|_\infty$, 122
$\|\cdot\|_p$, 122
$\|\cdot\|$, 54, 120
Span$\{\cdot\}$, 80
$\mathbf{A}^\mathsf{T}$, 29
$\mathbf{A}^{-1}$, 38
$\mathbf{A}^\mathsf{H}$, 30
$\mathbf{A}$, 11
$\mathbf{D}(\phi)$, 57, 58
$\mathbf{D}_{ij}(\phi)$, 59
$\mathbf{D}$, 18
$\mathbf{E}_{ij}$, 45
$\mathbf{G}(\phi)$, 58
$\mathbf{G}_{ij}(\phi)$, 59
$\mathbf{I}$, 18, 35
$\mathbf{L}$, 17
$\mathbf{P}_\mathbf{y}$, 137
$\mathbf{P}$, 47
$\mathbf{R}$, 17
$\mathbf{U}$, 17
$\mathbb{C}^n$, 73
$\mathbb{C}$, 7
$\mathbb{R}^n$, 8
$\mathbb{R}^n$, 73
$\mathbb{R}$, 7
$\cdot \perp \cdot$, 54
$\iota$, 29
$\mathscr{k}_{\mathscr{B}}$, 101
$\overline{z}$, 29
Rang$(\mathbf{A})$, 20
$\widehat{(\cdot,\cdot)}$, 54
$\widehat{\cdot,\cdot}$, 138
$e_j$, 9
$\ell$, 76

# Definitionen


Ähnlichkeit, 178

Abbildung
    bilineare, 164
    lineare, 106
    multilineare, 164
Abbildungsmatrix, 108
Addition, 73
Adjungierte Abblidung, 155
Algebraische Multiplizität (AM), 186
Algorithmus
    Gauss-Elimination, 18
allgemeine Lösung, 86
assoziativ, 32, 36
Assoziativitätsgesetz, 73
aufgespannt, 80

Basis, 92
    kanonische, 9
    Standard-, 9
Bild, 80
Bild der Abbildung, 80
Bild der Matrix, 80

Charakteristische Gleichung, 183
charakteristischesPolynom, 183
Cholesky-Zerlegung, 53

Determinante, 165
Determinante einer Matrix (Berechnung), 41
Diagonaldominante Matrix, 181
Diagonalisierung, 178
Diagonalmatrix, 18
Dimension eines linearen Raumes, 93
Diskrete Fourier Transformation, 208
distributiv, 36, 73
Drehmatrix, 57, 59
dünnbesetzte Matrizen, 66

Eigenraum, 185
Eigenvektor, 179
Eigenwert, 179
Eigenwertproblem, 179
Einheitsvektor, 138
Elimination, 18
Eliminationsmatrix, 45
Eliminationsmatrix $\mathbf{E}_{ij}$, 45
endlichdimensional, 88
Endlichdimensionale lineare Räume, 88
Erzeugendensystem, 83
erzeugt, 80
euklidische Norm, 54
euklidisches Skalarprodukt, 54
Exponentialmatrix $e^{\mathbf{D}t}$, 177

Faltung, 210
Fourier Matrix, 207
freie Variable, 14, 18
freie Variablen, 20
Frobenius-Norm, 124
Funktional, 153

Gauss-Elimination, 16, 18
    Blockweise, 50
Gauss-Jordan, 40
Geometrische Multiplizität (GM), 187
Gerschgorin-Gebiet, 181
Gerschgorin-Kreis, 181
Givens-Rotation, 58

Hermite transponierte Matrix, 30
homogen, 28
Homogenes LGS, 28
Householder-Matrix, 60

Identitätsmatrix, 16
Identitätsmatrix, 35, 275
Inverse, 37
Inverse einer Matrix, 37
Isomorphismen und Automorphismen, 111

Jordan-Matrix, 218
Jordanblock, 218

kanonische Basis, 9
Kern einer Matrix, 83
Kern und Bild, 113
Koeffizientenmatrix, 11
kommutativ, 32
Kommutatives Diagramm, 109

Kommutativitätsgesetz, 73
Kompatibilitätsbedingungen, 20
Kompatibilität mit der Multiplikation mit Skalaren, 73
Kompatibilitätsbedingung (KB), 14
Konditionszahl, 197
Koordinaten, 101
Koordinatenabbildung, 101
Koordinatenabbildung und Koordinaten, 101

Lösung
trivial, 28
LDU-Zerlegung, 52
LGS, 11
homogen, 28
linear abhängig, 90
linear im zweiten Argument, 133
linear unabhängig, 90
Lineare Abbildung, 106
Lineare Ausgleichsrechnung als lineares Gleichungssystem, 159
Lineare Unabhängigkeit, 90
linearer Raum, 73
Lineares Gleichungssystem, 11
Lineares Gleichungssystem (LGS), 11
Linearkombination, 8, 80
LR-Zerlegung, 43
LU-Zerlegung, 43

Matrix, 11
Addition, 31
antisymmetrisch, 30
dünnbesetzt, 66
Diagonal-, 18
diagonalisierbar, 178
Dreh-, 57, 59
Eliminations-, 45
Hermite symmetrisch, 30
Hermite transponierte, 30
Householder, 59
Inverse, 37
invertierbar, 37
Jordan-, 218
Multiplikation, 33
Multiplikation mit Skalare, 31
normal, 193
Obere Dreiecks-, 17
orthogonal, 55
Pseudoinverse, 231
regulär, 28
Rotations-, 57, 59
singulär, 28
Spiegelungs-, 59
symmetrisch, 30
transponiert, 29
Tridiagonal-, 18
unitäre, 55
Untere Dreiecks-, 17
voller Rang, 28
ähnlich, 178
Matrix-Abbildung, 55
Matrix-Vektor-Multiplikation, 12
Matrixmultiplikation, 33
Eigenschaften, 36
neutrales Element, 35
Matrixnorm aus der Vektornorm, 124
Matrizen
Potenzen, 35
Monome, 75
Multilineare Abbildung, 164
Multiplikation mit Skalaren, 73

neutrale Element, 32
Neutralelement, 73
Norm, 120
$L^2$-, 123
$\mathbf{A}$-, 135
$\infty$-, 122
$p$-, 122
euklidische, 54, 120
Frobenius-, 124
Spaltensummen-, 125
Spektral-, 125
sup-, 123
Zeilensummen-, 124
Norm aus einem Skalarprodukt, 133
Norm einer Matrix
euklidische, 232
Frobenius, 232
Nuklear-, 232
Spektral-, 232
Normale Matrix, 193
Normalengleichung, 161
Normierter linearer Raum, 120
Nullvektor, 73
Nullzeilen, 20

Obere Dreiecksmatrix, 17
Orthogonale Matrix, 55
Orthogonale Projektion auf einen Vektor, 136
Orthogonale und schiefe Projektoren, 149
Orthogonale Vektoren, 54
Orthogonaler Projektor auf einen Unterraum, 152

Orthogonalität von Elementen eines linearen Raumes mit Skalarprodukt, 134
Orthogonalität von Unterräumen, 98

partikuläre Lösung, 86
Permutationsmatrix, 47
Pivot, 18
Pivote, 20
Pivotspalten, 20
Pivotzeile, 18
Polarzerlegung, 239
Polynom, 75
  charakteristische, 183
positiv definit, 133
positiv semi-definit, 133
Potenzen von Matrizen, 35
Projektion
  auf einen Vektor, 136
  auf Unterraum, 142
Projektor, 149
  auf einen Unterraum
    orthogonaler, 152
    schiefer, 152
  orthogonaler, 149
  schiefer, 149
Pseudoinverse nach Moore (1920) und Penrose (1955), 231

QR-Zerlegung, 62
  mit Householder-Spiegelungen, 66

Rang, 20
Rang einer linearen Abbildung, 113
Rang einer Matrix, 20
Raum
  $C^s[a,b]$, 73
  $C^s$, 73
  $L^2[0,1]$, 75
  $L^2$, 75
  $\mathcal{P}_n$, 75
  $\mathbb{C}^n$, 73
  $\mathbb{R}^n$, 73
  der quadratisch integrierbaren Funktionen, 75
  der stetig Differenzierbaren Funktionen, 73
  linear, 73
rechte Seite, 11
regulär, 28
Rotationsmatrix, 57, 59
Rücksubstitution, 16

Schiefer Projektor auf einen Unterraum, 153
Schur
  Komplement, 51
Schur Komplement, 51
Sherman-Morrison-Woodbury Formel, 52
singulär, 28
Singulärvektoren
  linke, 224
  rechte, 224
Singulärwert, 221
Skalare, 7
Skalarprodukt, 133
  $L^2$-, 124
  $\mathbf{A}$-, 134
  euklidisches, 54
Spaltensummen-Norm, 125
Span, 80
Spektral-Norm, 125
Spiegelung, 59
Standardbasis, 9
Stufen, 20
Symmetriearten von Matrizen, 30
symmetrisch, 133
symmetrisch positiv definit, 198
Symmetrisch positiv definite Matrizen, 198
symmetrisch positiv semi-definit, 198

Transponierte Matrix, 29
Transponierte Vektoren, 31
Tridiagonalmatrix, 18
triviale Lösung, 28

Unbekannte, 11
Untere Dreiecksmatrix, 17
Unterraum, 76

Vektorraum, 73

Wichtige Matrizennormen, 232
Winkel in einem linearen Raum, 138
Winkel zwischen zwei Vektoren, 54

Zeilenstufenform, 20
Zeilensummen-Norm, 124
Zerlegung
  Cholesky, 53
  LDU, 52
  LR, 43
  LU, 43
  Polar-, 239
  QR, 62
  Schur-, 214
    komplexe, 214
    reelle, 214
Zirkulante Matrix, 205
Zyklische Faltung von periodischen Folgen, 210
Zyklische Faltung von Vektoren, 211

# Beispiele


$L^2$-Norm, 123
$\mathbf{A}$-Skalarprodukt, 134

Abbildungsmatrix bei Basiswechsel, 114
Abbildungsmatrix der Multiplikation, 109
Ableitungsoperator, 107

Basen, 92
Basiswechsel, 104
Beispiele und Gegenbeispiele von Unterräumen, 77
Berechnung der Determinante - Rechenführung, 167
Berechnung einer Determinante, 168
Berechnung vom Bild, 86
Berechnung vom Kern, 85
Bild einer Matrix / Bild der Matrixabbildung ist ein linearer Unterraum, 80

Das euklidische und das $L^2$-Skalarprodukt , 133
Diagonalisierbare Matrix mit multiplen Eigenwerte, 185
Die ideale Dämpfung eines harmonischen Oszillators braucht eine nicht-diagonalisierbare Matrix, 213
Die Räume $\mathbb{R}^n$ und $\mathbb{C}^n$, 73
Die SVD ist nicht eindeutig., 227
Differentialoperator, 107
Dimensionen, 94

Eigenwerte und Eigenvektoren einer singulären Matrix, 184
Eigenwerte und Eigenvektoren einer strikt diagonaldominanten Matrix, 182
Ein Unterraum in $\mathbb{R}^4$, 77
Eine Abbildung auf $\mathscr{P}$ mit Adjungierte, 156
Eine Abbildung auf $\mathscr{P}$ ohne Adjungierte, 156
Eine Abbildung mit Adjungierte., 156
Eine invertierbare $3 \times 3$-Matrix, 40
Eine Linearkombination von Vektoren liest man als lineares Gleichungssystem, 10
Eine Matrix mit rein imaginären Eigenwerten, 182
Einige lineare Abbildungen in 2 Dimensionen, 172
Elimination und Inverse, 12
Ellipse, 203
Endlichdimensionale lineare Räume, 88
Erzeugende Systeme, 88
Euklidische Norm, 120

Filme, 236
Finite Differenz für die zweite Ableitung., 168
Freie Variablen und Kompatibilitätsbedingungen, 25

Gauss-Elimination zur Lösung des Eigenwertproblems, 185
Gegenbeispiel von Unterraum, 78, 79

Householder-Matrix, 61

Jordan-Matrix, 218, 219

Kern einer Matrix / Kern der Matrixabbildung ist ein linearer Unterraum, 83
Kompatibilitätsbedingungen, 15
Kompatibilitätsbedingungen mit Parameter, 23
Kompression mit SVD, 234
Koordinaten in der Basis von Monomen, 102
Koordinaten in der Standardbasis, 101
Koordinatenabbildung, 106

Lösung einer Differentialgleichung mit einer oberen Dreiecksmatrix, 219
Legendre-Polynome aus der Gram-Schmidt Orthogonalisierung, 145
Linear abhängige Vektoren führen zu besonderen linearen Gleichungssystemen, 13
Lineare Ausgleichsrechnung - konkret, 163
Lineares Modell, 157
Lineares Modell - konkret, 159
LU-Zerlegung mit Permutationen., 47

Matrixabbildung, 106
Matrixnormen, die von einer Vektornorm induziert sind, 124
Mehrfache Singulärwerte, 237
Modell im Unterraum, 159
Multiplikation als Faltung von Ziffern, 210

Nützliche endlichdimensionale Unterräume, 88

Nicht diagonalisierbare Matrix, 188
Nullen mit Householder-Spiegelungen, 67
Nullen via Givens-Rotationen, 64

Obere Dreiecksmatrix, 214
Orthogonale Matrix, 55

Permutationsmatrizen sind orthogonal, 57
Positive quadratische Form, 201
Potenz einer Matrix, 176
Projektoren auf Unterräume, 151

QR-Zerlegung mittels Householder-Spiegelungen, 69

Räume von Polynomen, 75
Raum $\mathcal{P}_5$ wird auch von Legendre-Polynomen erzeugt, 81
Raum der quadratisch integrierbaren Funktionen $L^2$, 75
Raum der stetig differenzierbaren Funktionen, 73
Rechenführung beim Eigenwertproblem, 188
Rotationsmatrix in 2D, 57
Rotationsmatrix in 3D, 58

Schiefer Projektor, 150
Schiefer Projektor auf einem Unterraum, 152
Schur-Form, 215
Semi-positiv definierte bilineare Funktion, 135
Span von Monomen, 81
Spiegelung, 59
sup-Norm, 123
SVD einer nicht-symmetrischen Matrix, 225
SVD einer singulären Matrix, 226
System entkoppelter linearer Differenzialgleichungen 1. Ordnung, 177
System gekoppelter linearer Differenzialgleichungen 1. Ordnung, 177

Veränderte Ableitung, 169
Verschachtelung einiger Räume von Funktionen, 77

Wichtige Mengen linear unabhängiger Vektoren, 90

Zwei Pivote und eine einzige Gleichung als Kompatibilitätsbedingung, 22
Zwei Pivote und eine Gleichung als Kompatibilitätsbedingung, 22
Zyklische Faltung, 211

# Bemerkungen und Theoreme


Bild $\mathbf{A} \neq$ Bild $\mathbf{U}$, 96
dim Bild $\mathbf{A}$ = dim Bild $\mathbf{U}$ = Rang $\mathbf{A}$, 96
$n$ unterschiedliche Eigenwerte, 180
Blockweises Produkt, 36

Abbildungsmatrix und Skalarprodukt, 154
Allgemeine Lösung eines LGS, 86
Allgemeine sparsame Form der LU-Zerlegung, 50
Anzahl positiver Eigenwerte symmetrischer, invertierbarer Matrizen, 198
Ausgleichsrechnung, 262

Berechnung der Determinante, 167
Berechnung der Eigenvektoren, 182
Berechnung der Koordinaten in einer orthonormalen Basis, 139
Berechnung der orthogonalen Projektion in einem linearen Raum mit Skalarprodukt, 136
Betrag der Eigenwerte, 180
Blick auf die Multiplikation von Matrizen: Zeilenweise & Spaltenweise, 35
Blockweise Gauss-Elimination, 50

Cauchy-Schwarz Ungleichung, 121
Charakterisierungen von Unterräumen, 79

Das Produkt der Eigenwerte ist das Produkt der Diagonalelemente von $\mathbf{U}$, 198
Der Kern ist ein Unterraum, 84
Der Rang des Ergebnisses der Matrixmultiplikation ist maximal der kleinste Rang der Faktoren, 34
Der Rang ist charakteristisch für die ganze Matrix, 21
Der Rang ist gleich gross oder kleiner als die Matrixdimensionen, 21
Determinante erschien bereits in der Gauss-Elimination, 164
Diagonalisierbare Matrix, 178
Diagonalisierbare Matrizen sind Summen von (schiefen) Projektoren, 195
Diagonalisierung, Eigenwerte, Eigenvektoren, 178
Die Identitätsmatrix ist das neutrale Element für die Matrixmultiplikation, 35
Die Bedingung, dass beide Matrizen diagonalisierbar sind, ist notwendig., 212
Die Determinante ist eine multilineare, antisymmetrische Form, 165
Die ersten drei Eigenschaften legen die Determinante fest, 165
Die ersten Legendre-Polynome, 82
Die Gauss-Elimination liefert die LU-Zerlegung, 43
Die Kompatibilitätsbedingungen entscheiden über die Existenz von Lösungen, 27
Die Koordinatenabbildung ist ein Isomorphismus, 111
Die LU-Zerlegung (ohne Permutationen) ist eindeutig, 47
Die Matrixmultiplikation ist nicht kommutativ, 35
Die Pivotspalten sind die wesentlichen Spalten, 21
Die Pivotzeilen sind die wesentlichen Zeilen, 21
Die Struktur einer Matrix ist wichtig, 35
Dualer Raum, 154

Eigenschaften der Einheitswurzel, 208
Eigenschaften der Fourier-Matrix, 207
Eigenwerte der Fourier-Matrix, 209
Eindeutigkeit der SVD, 225
Eine Matrix ist regulär (invertierbar, voller Rang) genau dann, wenn ihre Determinante nicht Null ist, 41
Eine reguläre Matrix hat eine Inverse, 37
Eine sehr lange Formel für die Determinante, 169
Elimination berechnet lineare Kombinationen der Zeilen, 21
Elimination entspricht Linearkombinationen von Gleichungen/Zeilen, 15
Elimination und lineare Kombinationen der Spalten, 21
Elimination verwendet nur lineare Kombinationen und produziert spezielle Matrizen, 18

Faltungstheorem, 211
FFT, 209

Für die Matrixmultiplikation müssen die Dimensionen passen, 34

Geometrie in $n$ Dimensionen, 204
Gram-Schmidt in $\mathbb{R}^n$ gibt eine QR-Zerlegung, 146

Inverse DFT, 209
Inverse eines Isomorphismus ist auch Isomorphismus, 111

Jede Householder-Matrix ist orthogonal, 60

Kompatibilitätsbedingungen für mindestens eine Lösung, 21
Kondition einer symmetrischen Matrix, 197

LGS löst man mit der LU-Zerlegung, nicht mit der Inversen, 44
Lineare Abhängigkeit, 90

Matrix-Abbildung, 55
Matrix-Vektor-Multiplikation, 11
Matrizen mit besonderer Struktur haben spezielle Eigenwerte, 192
Multiplikation mit einer Diagonalmatrix, 34

Nicht alle Normen von Matrizen kommen aus einer Vektornorm, 124
Nicht diagonalisierbare Matrizen, 190
Nicht-triviale Lösungen eines homogenen Systems, 28
Normale Matrizen sind Summen von unitären/orthogonalen Projektoren, 194

Ohne KB bestimmen die freien Variablen die Menge der Lösungen, 28
Operationen mit Eliminationsmatrizen, 45
Optimierung mit einer Diagonalmatrix, 195
Orthogonale Matrix aus orthonormaler Basis, 139
Orthogonale Projektion in $\mathbb{R}^n$, 136
Orthogonale Zerlegung nach dem Fundamentalsatz der linearen Algebra - I, 100
Orthogonalität mancher Eigenvektoren, 185
Orthonormale Basis, 259

Projektion auf einen Unterraum, 141
Projektoren auf die Elemente einer orthonormalen Basis, 139

QR-Zerlegung mit orthogonalen Transformationen oder mit Gram-Schmidt?, 147
QR-Zerlegung via Givens-Rotationen ist günstig für dünnbesetzte Matrizen, 66
Quadratische Form als Summe von Quadraten für symmetrische $\mathbf{A}$, 201

Rechenregeln in linearen Räumen, 76
Reduzierte SVD, 229
Reelle Matrizen können komplexe Eigenwerte haben, 182

S.p.d.-Test via Pivotelemente., 199
Schlecht konditionierte Matrix, 232
Schur-Zerlegung einer Hermite-symmetrischen Matrix, 216
Sherman-Morrison-Woodbury Formel, 51
Skalarprodukt entsprechend einer s.p.d. Matrix, 198
Speicherplatz, 21
Spur & Determinante, 183
Symmetrische Matrizen sind normal, 193

Theorem: $n$ linear unabhängige Eigenvektoren diagonalisieren eine $n \times n$-Matrix, 179
Theorem:
Normale Matrizen sind die durch unitäre Transformationen diagonalisierbaren Matrizen, 217
Theorem:
Normale Matrizen sind die durch unitäre Transformationen diagonalisierbaren Matrizen, 193
Theorem: Ähnliche Matrizen haben gleiche Eigenwerte, 187
Theorem: Äquivalenz der Normen, 123
Theorem: Adjungierte Abbildung für endlich dimensionale Räume, 155
Theorem: Basis im Bild $\mathbf{A}$, 95
Theorem: Beste Approximation an einer Matrix: Schmidt 1907 wiederentdeckt von Eckart 1933 und Young 1936, 235
Theorem: Cauchy-Bunjakowski-Schwarz-Ungleichung, 121
Theorem: Diagonalisierung, Eigenwerte, Eigenvektoren, 179
Theorem: Die Eigenwerte der inversen Matrix, 197
Theorem: Dimension von Kern $\mathbf{A}^\intercal$, 98
Theorem: Dimension von Kern $\mathbf{A}$, 96
Theorem: Dimensionssatz, 113
Theorem: Eigenschaften der Inverse, 38
Theorem: Eigenschaften der Matrixaddition, 32
Theorem: Eigenschaften der Matrixmultiplikation, 36
Theorem: Eigenschaften einer Jordan-Matrix, 218
Theorem: Eigenwerte einer s.p.d. Matrix, 198
Theorem: Eindeutige Darstellung in gegebener Basis, 101

Theorem: Ellipse, 204
Theorem: Erhaltungssatz, 56
Theorem: Euklidische Norm und Eigenwerte, 196
Theorem: Existenz und Anzahl der Eigenwerte einer Matrix, 183
Theorem: Existenz und Eindeutigkeit der Inverse, 38
Theorem: Fundamentalsatz der linearen Algebra - Teil I, 99
Theorem: Fundamentalsatz der linearen Algebra - Teil II., 224
Theorem: Gleichzeitig diagonalisierbare Matrizen, 212
Theorem: Gleichzeitige Diagonalisierung aller zirkulanten Matrizen, 207
Theorem: Grösse der Basen, 93
Theorem: Grösse von erzeugenden Systemen in einem endlichdimensionalen Raum, 94
Theorem: Gram-Schmidt, 143
Theorem: Jordanzerlegung, 219
Theorem: Kern einer linearen Injektion, 113
Theorem: Kriteria für Existenz der Inverse, 39
Theorem: Linear unabhängige Spalten einer Matrix, 92
Theorem: Lösung der Ausgleichsrechnung mit der Normalengleichung, 161
Theorem: Lösungen für **Ax = 0**, 181
Theorem: Multiplizitätsregel, 188
Theorem: Nützliche Ungleichungen zwischen Matrixnormen, 233
Theorem: Normale obere Dreiecksmatrizen sind Diagonalmatrizen, 216
Theorem: Operationen mit orthogonalen Matrizen, 57
Theorem: Orthogonale und schiefe Projektoren, 150
Theorem: Orthogonale Vektoren sind linear unabhängig, 138
Theorem: Orthonormale Basis, 139
Theorem: Parseval, 141
Theorem: Pythagoras, 138
Theorem: QR-Zerlegung einer allgemeinen Matrix, 71
Theorem: Riesz Darstellungssatz, 153
Theorem: Satz von Gerschgorin, 181
Theorem: Schur-Zerlegung, 214
Theorem: Schwarz'sche Ungleichung, 137
Theorem: Singulärwertzerlegung (Singular Value Decomposition/SVD), 221
Theorem: Spektralsatz, 193
Theorem: Strikt diagonaldominante Matrizen sind regulär, 182
Theorem: Störungstheorem, 126
Theorem: Symmetrische Matrizen haben reelle Eigenwerte und orthogonale Eigenvektoren, 192
Theorem: Verschiedene Eigenwerte haben linear unabhängige Eigenvektoren, 179
Theorem: Zusammenhang zwischen GM und AM entscheidet, ob diagonalisierbar oder nicht, 191

Verschiedene Implementierungen können verschiedene Matrizen produzieren, 43
Vom Skalarprodukt zur Norm, 133
Vorzeitiges Ende in Gram-Schmidt-Algorithmus, 148

Weitere wichtige Eigenschaften der Determinante, 165
Weiteres zur Schur-Zerlegung, 215
Wenn alle Eigenwerte AM 1 haben, dann ist die Matrix diagonalisierbar, 187
Wenn eine Matrix eine Inverse hat, dann ist sie regulär (hat vollen Rang), 37

zyklische Faltung via Matrixmultiplikation, 211
Zyklische Faltung von Vektoren, 211

Äquivalenzschema quadratischer Matrizen, 28

# Pseudocode

Ausgleichsrechnung, 261

Bild und Kern einer Matrix, 259

Determinante, 263

Eigenwertproblem, 264
Eigenwertproblem bei nicht diagonalisierbarer Matrix, 265

Faltung, 266
Filme, 267

Gauss-Elimination: Betragsgrösster Pivot, 246
Gauss-Elimination: Einfache Brüche, 252
Gauss-Jordan, 40
Gram-Schmidt, Modifizierter, 148

LU-Zerlegung, 244
Lösung linearer Gleichungssysteme, 243

Matrix-Vektor-Multiplikation, 243
Matrixeigenschaften, 242
Modifizierter Gram-Schmidt Algorithmus, 148

QR-Zerlegung, 245

Rang, 244

Singulärwertzerlegung, 267
Skalarprodukt, 260

Vektor- und Matrixnormen, 260
Vektoreigenschaften, 242

# Literatur


[Gut07] Martin H Gutknecht. *Lineare Algebra*. de. Zürich, Schweiz: ETH Zürich, Juni 2007 (siehe S. 1, 25, 92).

[NS15] Kaspar Nipp und Daniel Stoffer. *Lineare Algebra*. vdf Hochschulverlag AG, 2015. DOI: 10.3218/3704-3 (siehe S. 1, 22, 23, 145, 185, 188).

[OS18] Peter J Olver und Chehrzad Shakiban. *Applied Linear Algebra*. en. 2. Aufl. Undergraduate Texts in Mathematics. Cham, Schweiz: Springer International Publishing, Juni 2018. DOI: 10.1007/978-3-319-91041-3 (siehe S. 1).

[Str09] Gilbert Strang. *Introduction to Linear Algebra*. en. 4. Aufl. Wellesley, MA, USA: Wellesley-Cambridge Press, 2009. DOI: 10.1007/978-3-642-55631-9 (siehe S. 1, 10, 12, 13, 40, 45, 159, 168, 169, 184, 203, 225).