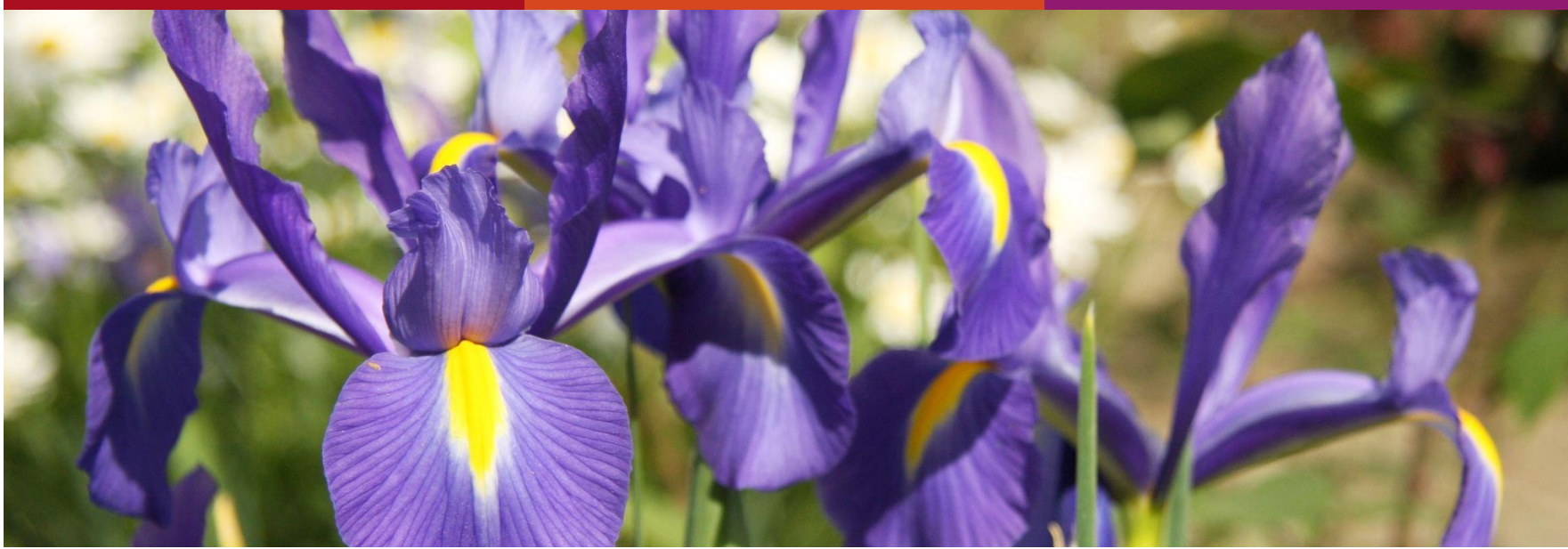


Abbildung: *Blaue Schwertlilie (Iris)* - Maja Dumat, CC-BY 2.0

Principal Component Analysis – Theorie und Anwendung

Komplexität reduzieren - Information bewahren

Bachelor Studiengänge im 3. Semester

Überblick

Principal Components Analysis (PCA)

- “Curse of Dimensionality”
- Iris Datensatz
- PCA - Die Idee am 2D Beispiel
- PCA auf dem Iris Datensatz
    - Schritt 1: Standardisierung
    - Schritt 2: Kovarianz-Matrix, Eigenwerte und Eigenvektoren
    - Schritt 3: Ladungsmatrix
    - Schritt 4: Wahl der Hauptkomponenten
    - Schritt 5: Projektion
- PCA auf dem MNIST Datensatz
- Anwendung
- Grenzen

“Curse of Dimensionality”

Räume mit vielen Dimensionen haben ungünstige Eigenschaften für maschinelles Lernen:

- Sehr geräumig
- Lose besetzt
- Punkte haben viele Nachbarn
- Abstände verlieren an Aussagekraft

Entstehende Nachteile:

- Probleme bei vielen Modellen / Overfitting
- Hohe Rechenkosten und Speicherbedarf
- Visualisierung erschwert

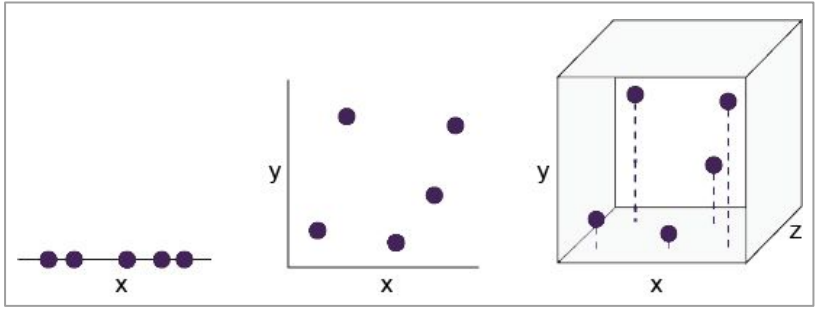


Abbildung: H. I. Rhys; *Machine Learning with R, the tidyverse, and mlr,* Manning Verlag, 2020, https://www.manning.com/books/machine-learning-with-r-the-tidyverse-and-mlr

Ansatz: Dimensionsreduktion

- **Ziel:** Dimension reduzieren
- **Aber**: Information beibehalten

a	b	c
-8	-4	1
12	6	1
-18	-9	1
-14	-7	1
-6	-3	0

Ansatz: Dimensionsreduktion

- **Ziel:** Dimension reduzieren
- **Aber**: Information beibehalten

➡ **Principal Component Analysis** (PCA)

- Lineare Zusammenhänge zwischen Variablen
- Varianz = Information

a	b	c
-8	-4	1
12	6	1
-18	-9	1
-14	-7	1
-6	-3	0

Iris Datensatz

- Klassisches Benchmark-Dataset für Klassifikation und Visualisierung
- Edgar Anderson erhob den Datensatz, um die strukturellen Unterschiede in den Blütenmerkmalen zwischen drei verwandten Iris-Arten zu erfassen.
- 3 Arten von Schwertlilien (Iris):
  - Iris setosa
  - Iris versicolor
  - Iris virginica

Abbildungen: *Iris Setosa, Iris Versicolor, Iris Virginica.* Quelle: Wikipedia.org

Iris Datensatz

- Klassisches Benchmark-Dataset für Klassifikation und Visualisierung
- Edgar Anderson erhob den Datensatz, um die strukturellen Unterschiede in den Blütenmerkmalen zwischen drei verwandten Iris-Arten zu erfassen.
- 3 Arten von Schwertlilien (Iris):
  - Iris setosa
  - Iris versicolor
  - Iris virginica
- 150 Beobachtungen (je 50 pro Klasse)
- Features (numerisch, in cm):
  - Kelchblattlänge (sepal length)
  - Kelchblattbreite (sepal width)
  - Kronblattlänge (petal length)
  - Kronblattbreite (petal width)

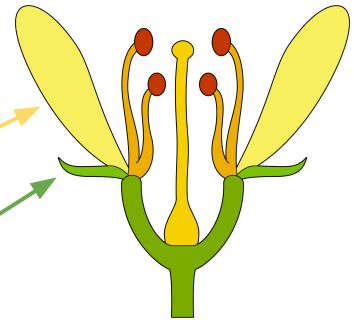


Abbildung: *Schematische Darstellung einer Blüte*. Quelle: Wikipedia.org

Abbildungen: *Iris Setosa, Iris Versicolor, Iris Virginica*. Quelle: Wikipedia.org

Iris Datensatz

	Kelchblattlänge sepal length (cm)	Kelchblattbreite sepal width (cm)	Kronblattlänge petal length (cm)	Kronblattbreite petal width (cm)	target
0	5.10	3.50	1.40	0.20	setosa
1	4.90	3.00	1.40	0.20	setosa
2	4.70	3.20	1.30	0.20	setosa
3	4.60	3.10	1.50	0.20	setosa
4	5.00	3.60	1.40	0.20	setosa
5	5.40	3.90	1.70	0.40	setosa
6	4.60	3.40	1.40	0.30	setosa
7	5.00	3.40	1.50	0.20	setosa
8	4.40	2.90	1.40	0.20	setosa
9	4.90	3.10	1.50	0.10	setosa
10	5.40	3.70	1.50	0.20	setosa
11	4.80	3.40	1.60	0.20	setosa

Abbildungen: *Iris Setosa, Iris Versicolor, Iris Virginica.* Quelle: Wikipedia.org

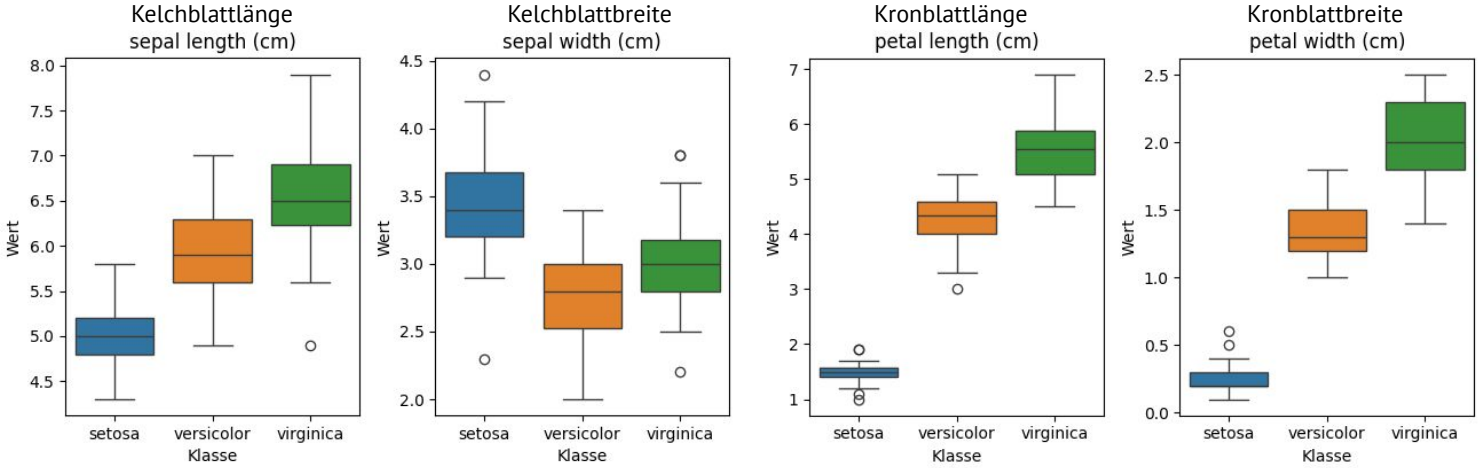
Iris Datensatz

	Kelchblattlänge sepal length (cm)	Kelchblattbreite sepal width (cm)	Kronblattlänge petal length (cm)	Kronblattbreite petal width (cm)	target
0	5.10	3.50	1.40	0.20	setosa
1	4.90	3.00	1.40	0.20	setosa
2	4.70	3.20	1.30	0.20	setosa
3	4.60	3.10	1.50	0.20	setosa
4	5.00	3.60	1.40	0.20	setosa
5	5.40	3.90	1.70	0.40	setosa
6	4.60	3.40	1.40	0.30	setosa
7	5.00	3.40	1.50	0.20	setosa
8	4.40	2.90	1.40	0.20	setosa
9	4.90	3.10	1.50	0.10	setosa
10	5.40	3.70	1.50	0.20	setosa
11	4.80	3.40	1.60	0.20	setosa

	Kelchblattlänge sepal length (cm)	Kelchblattbreite sepal width (cm)	Kronblattlänge petal length (cm)	Kronblattbreite petal width (cm)
count	150.000	150.000	150.000	150.000
mean	5.843	3.057	3.758	1.199
std	0.828	0.436	1.765	0.762
min	4.300	2.000	1.000	0.100
25%	5.100	2.800	1.600	0.300
50%	5.800	3.000	4.350	1.300
75%	6.400	3.300	5.100	1.800
max	7.900	4.400	6.900	2.500

Abbildungen: *Iris Setosa, Iris Versicolor, Iris Virginica.* Quelle: Wikipedia.org

Iris Datensatz



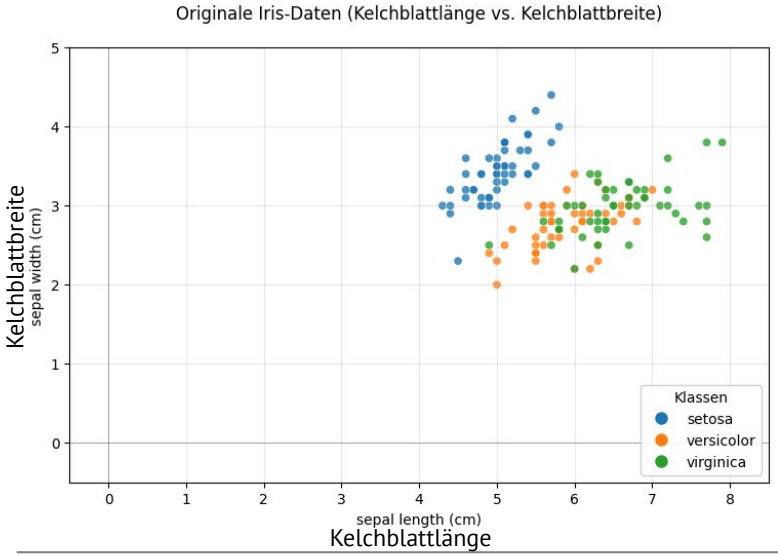
Abbildungen: *Iris Setosa, Iris Versicolor, Iris Virginica.* Quelle: Wikipedia.org

Grundidee Boxplot:

- Visuelle Zusammenfassung der Verteilung numerischer Daten
- Zeigt Median, Quartile, Ausreißer auf einen Blick
- Ideal zum Vergleich mehrerer Gruppen

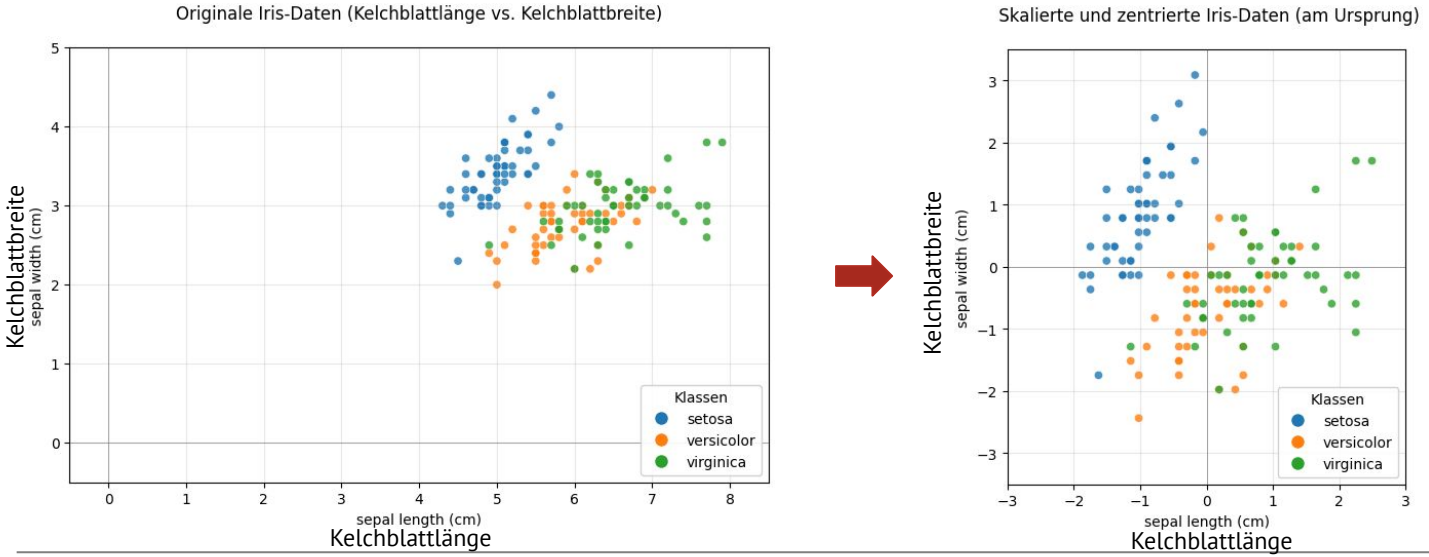
PCA - Die Idee am 2D Beispiel

- **Schritt 1**: Standardisierung (Skalieren & Zentrieren)
- **Schritt 2**: Kovarianz-Matrix, Eigenwerte und Eigenvektoren
- **Schritt 3**: Ladungs-Matrix
- **Schritt 4**: Wahl der Komponenten
- **Schritt 5**: Projektion



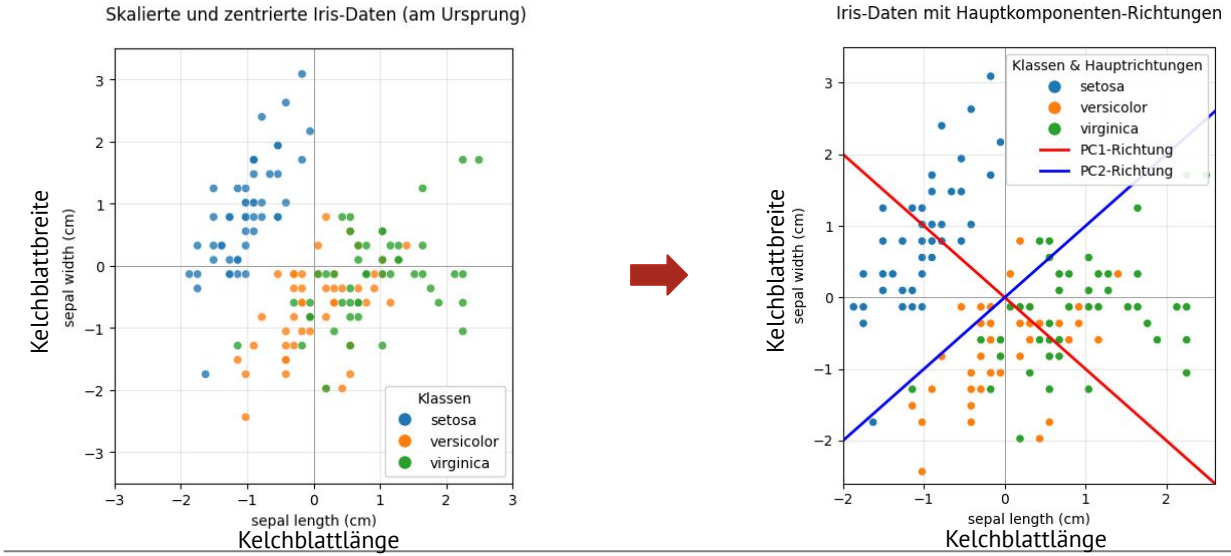
PCA - Die Idee am 2D Beispiel

- **Schritt 1**: Standardisierung (Skalieren & Zentrieren)
- **Schritt 2**: Kovarianz-Matrix, Eigenwerte und Eigenvektoren
- **Schritt 3**: Ladungs-Matrix
- **Schritt 4**: Wahl der Komponenten
- **Schritt 5**: Projektion



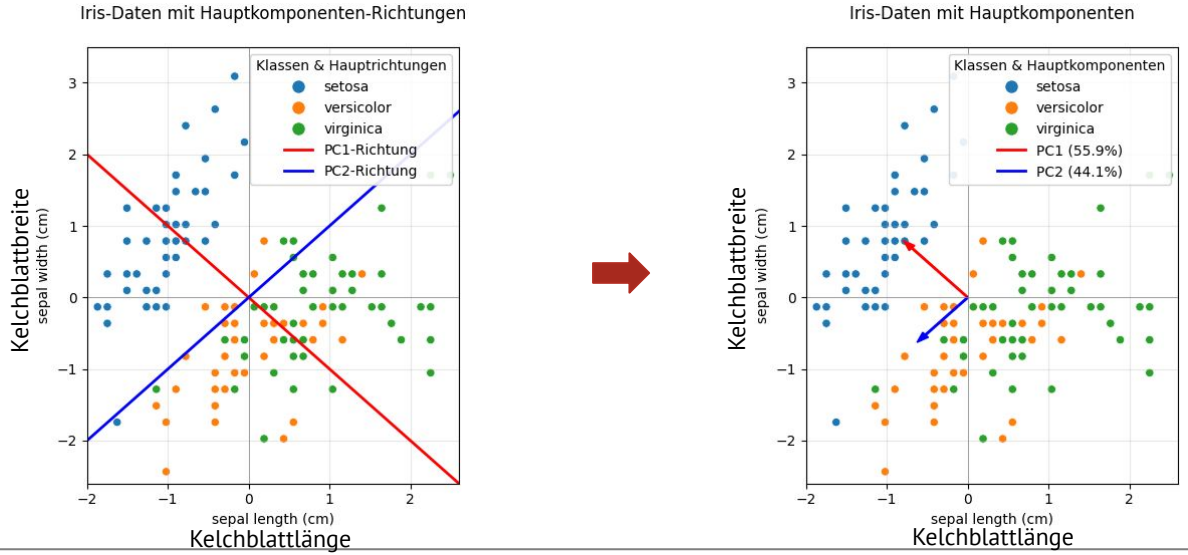
PCA - Die Idee am 2D Beispiel

- **Schritt 1**: Standardisierung (Skalieren & Zentrieren)
- **Schritt 2**: Kovarianz-Matrix, Eigenwerte und Eigenvektoren
- **Schritt 3**: Ladungs-Matrix
- **Schritt 4**: Wahl der Komponenten
- **Schritt 5**: Projektion



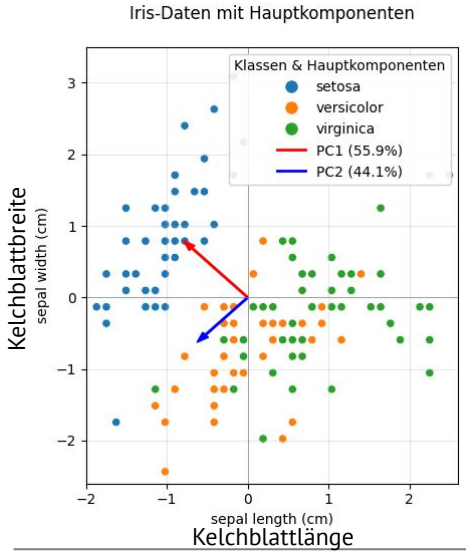
PCA - Die Idee am 2D Beispiel

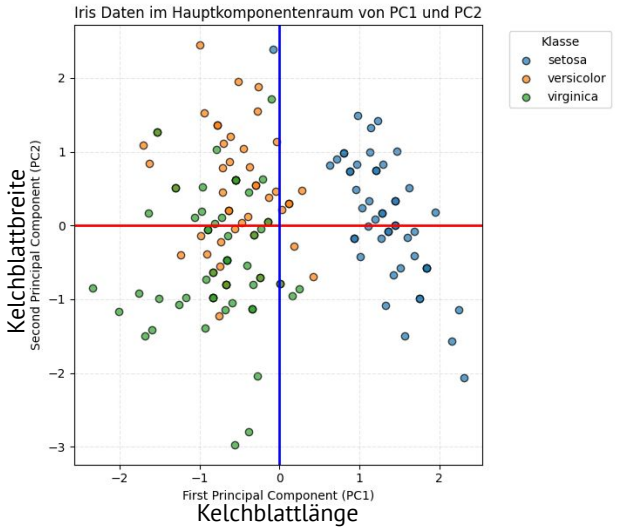
- **Schritt 1**: Standardisierung (Skalieren & Zentrieren)
- **Schritt 2**: Kovarianz-Matrix, Eigenwerte und Eigenvektoren
- **Schritt 3**: Ladungs-Matrix
- **Schritt 4**: Wahl der Komponenten
- **Schritt 5**: Projektion



PCA - Die Idee am 2D Beispiel

- **Schritt 1**: Standardisierung (Skalieren & Zentrieren)
- **Schritt 2**: Kovarianz-Matrix, Eigenwerte und Eigenvektoren
- **Schritt 3**: Ladungs-Matrix
- **Schritt 4**: Wahl der Komponenten
- **Schritt 5**: Projektion





Principal Component Analysis auf dem Iris Datensatz

	Kelchblattlänge sepal length (cm)	Kelchblattbreite sepal width (cm)	Kronblattlänge petal length (cm)	Kronblattbreite petal width (cm)	target
0	5.10	3.50	1.40	0.20	setosa
1	4.90	3.00	1.40	0.20	setosa
2	4.70	3.20	1.30	0.20	setosa
3	4.60	3.10	1.50	0.20	setosa
4	5.00	3.60	1.40	0.20	setosa
5	5.40	3.90	1.70	0.40	setosa
6	4.60	3.40	1.40	0.30	setosa
7	5.00	3.40	1.50	0.20	setosa
8	4.40	2.90	1.40	0.20	setosa
9	4.90	3.10	1.50	0.10	setosa
10	5.40	3.70	1.50	0.20	setosa
11	4.80	3.40	1.60	0.20	setosa

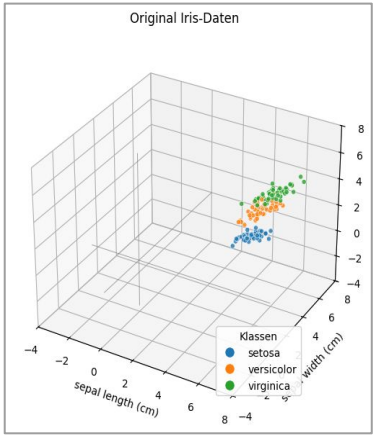
Abbildungen: *Iris Setosa, Iris Versicolor, Iris Virginica.* Quelle: Wikipedia.org

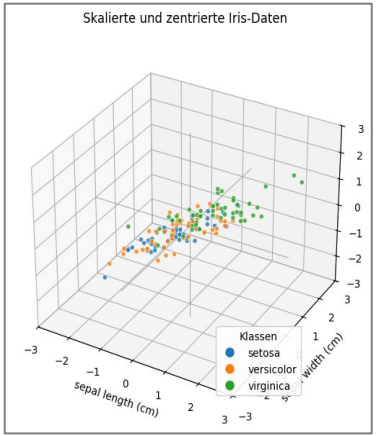
Schritt 1: Standardisieren (Zentrieren + Skalieren)

- Zentrierung: Mittelwert=0
- Skalierung: Varianz=1

	Kelchblattlänge sepal length (cm)	Kelchblattbreite sepal width (cm)	Kronblattlänge petal length (cm)	Kronblattbreite petal width (cm)
0	-0.901	1.019	-1.340	-1.315
1	-1.143	-0.132	-1.340	-1.315
2	-1.385	0.328	-1.397	-1.315
3	-1.507	0.098	-1.283	-1.315
4	-1.022	1.249	-1.340	-1.315
5	-0.537	1.940	-1.170	-1.052
6	-1.507	0.789	-1.340	-1.184
7	-1.022	0.789	-1.283	-1.315
8	-1.749	-0.362	-1.340	-1.315
9	-1.143	0.098	-1.283	-1.447
10	-0.537	1.479	-1.283	-1.315
11	-1.264	0.789	-1.227	-1.315

```
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```



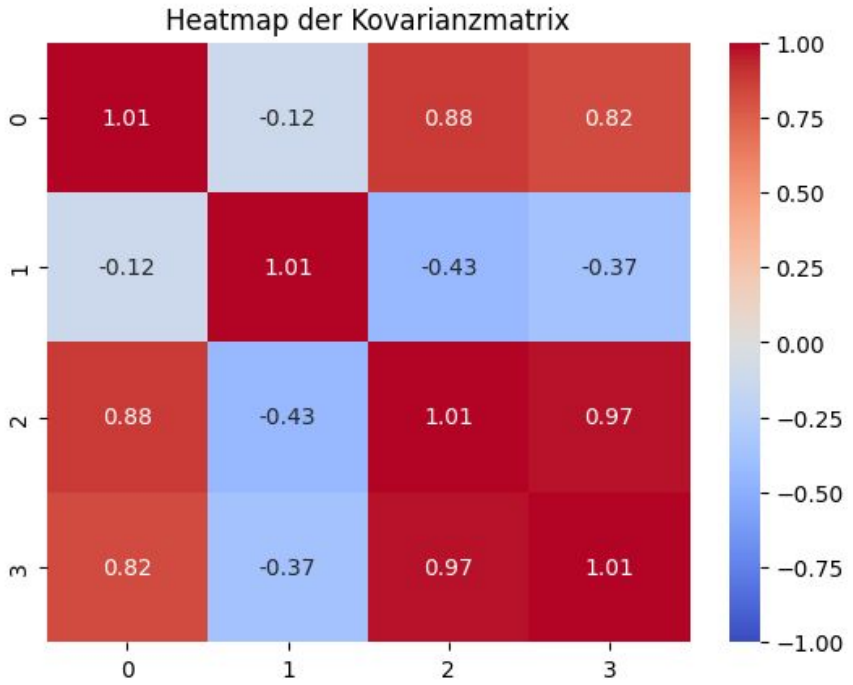


Schritt 2: Kovarianz-Matrix, Eigenwerte und Eigenvektoren

Kovarianz-Matrix:

- Die Hauptdiagonale enthält **Varianzen**
- Die übrigen Stellen enthalten die **Kovarianzen** zwischen verschiedenen Variablen.
- Die Matrix ist **symmetrisch** zur Hauptdiagonalen
- Sonderfall wegen Standardisierung: Kovarianzmatrix ähnelt Korrelationsmatrix

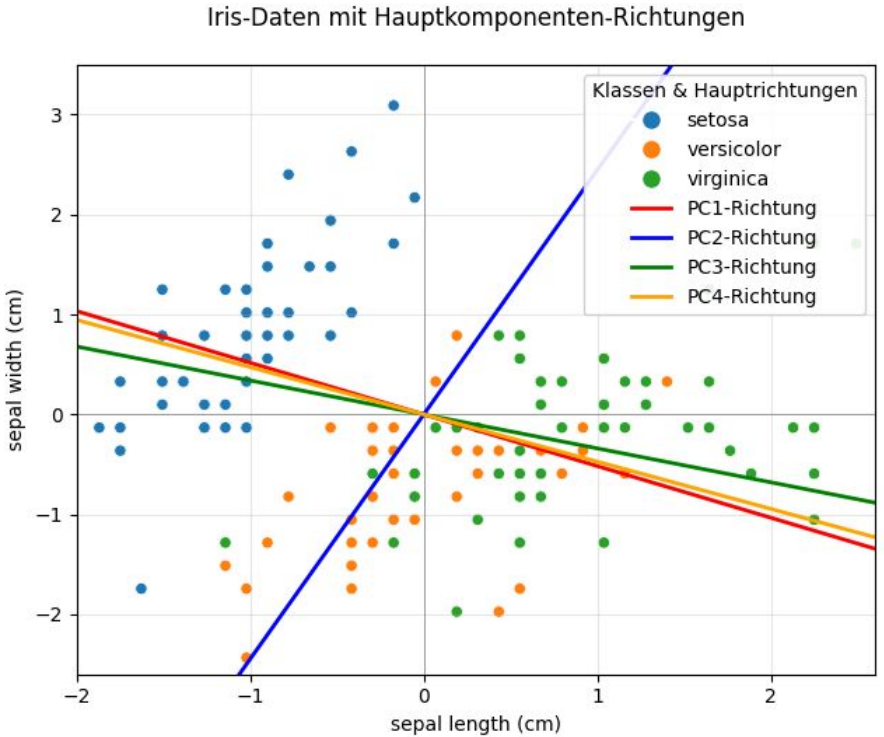
```
# Kovarianzmatrix berechnen
cov_matrix = np.cov(X_scaled.T)
```



Schritt 2: Kovarianz-Matrix, Eigenwerte und Eigenvektoren

- **Eigenwerte und Eigenvektoren** berechnen sich aus der Kovarianzmatrix der (standardisierten) Daten.
- Sie helfen dabei, Richtungen mit der größten Varianz in den Daten zu identifizieren
- Die **Eigenvektoren** definieren die Richtungen der neuen Hauptkomponentenachsen im ursprünglichen Merkmalsraum.

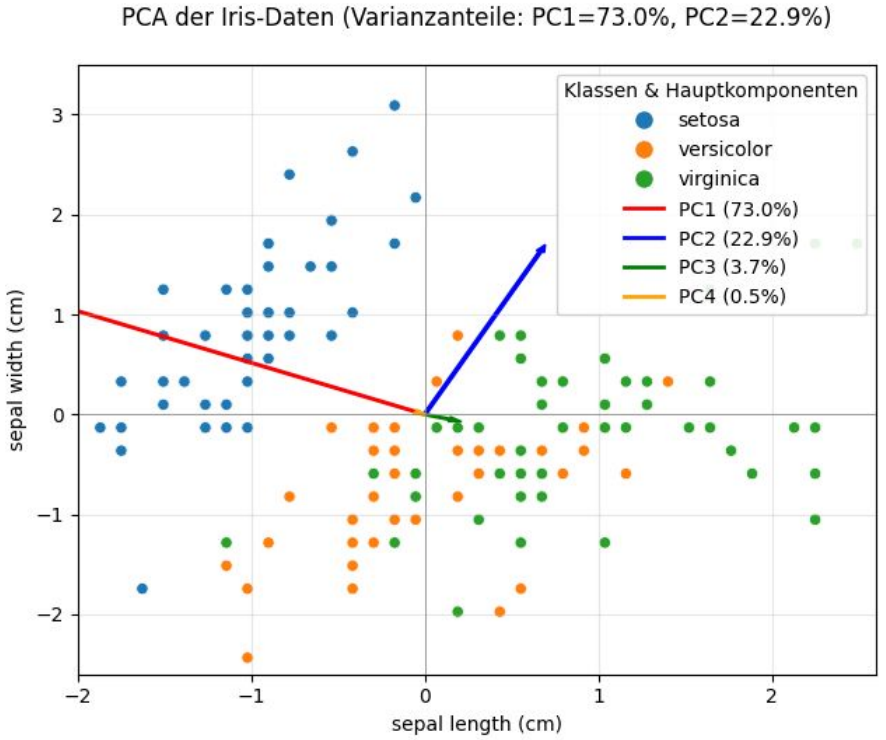
```
# Eigenwerte und Eigenvektoren berechnen
eigenvalues, eigenvectors = np.linalg.eigh(cov_matrix)
```



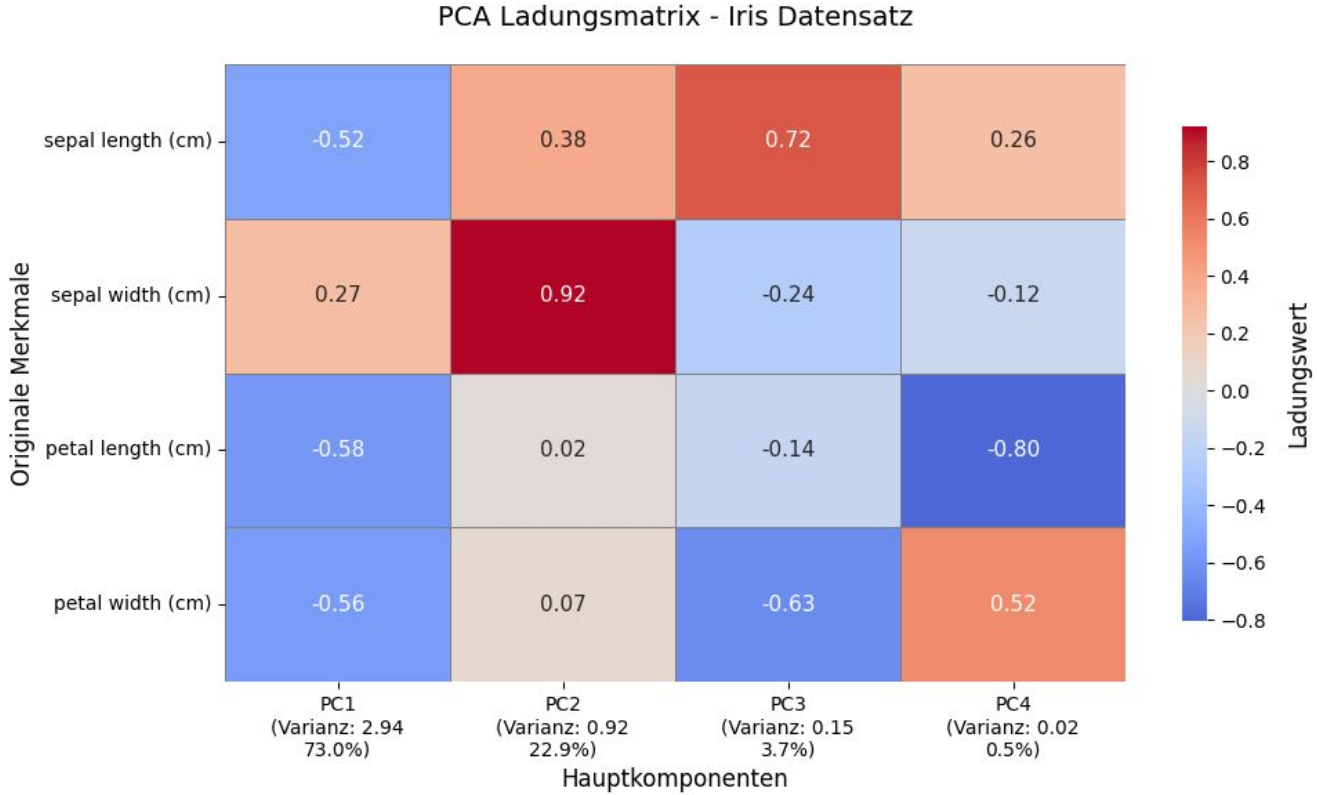
Schritt 2: Kovarianz-Matrix, Eigenwerte und Eigenvektoren

- Die **Eigenwerte** geben die Varianz der Daten entlang der neuen Hauptkomponentenachsen an.
- Da die Hauptkomponenten orthogonal zueinander stehen, enthalten sie **keine redundante Varianzinformation**.
- Die **Elemente eines Eigenvektors** zeigen an, wie stark jede der ursprünglichen Variablen zu der neuen Hauptkomponente beiträgt.

```
# Eigenwerte absteigend sortieren
sorted_idx = np.argsort(eigenvalues)[::-1]
eigenvalues = eigenvalues[sorted_idx]
eigenvectors = eigenvectors[:, sorted_idx]
```



Schritt 3: Ladungs-Matrix



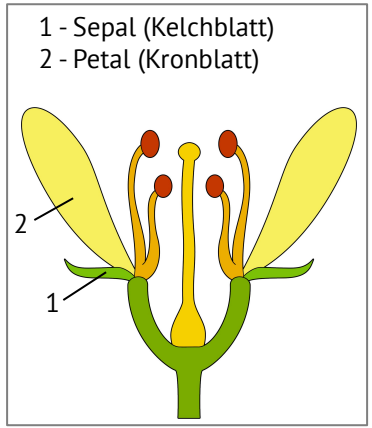
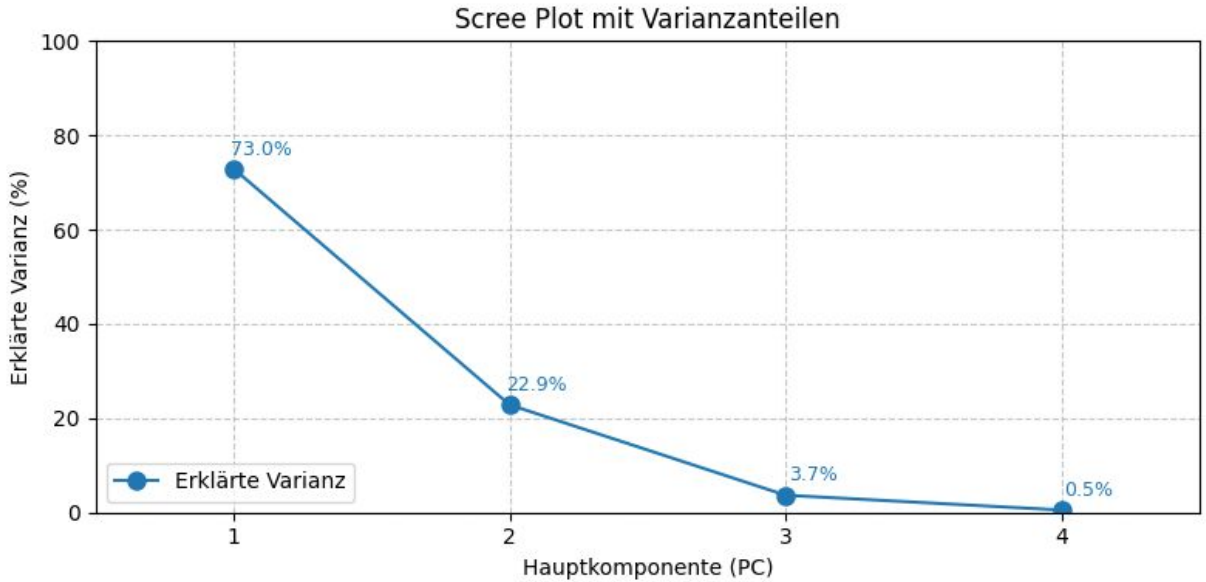


Abbildung: *Schematische Darstellung einer Blüte.*
Quelle: Wikipedia.org

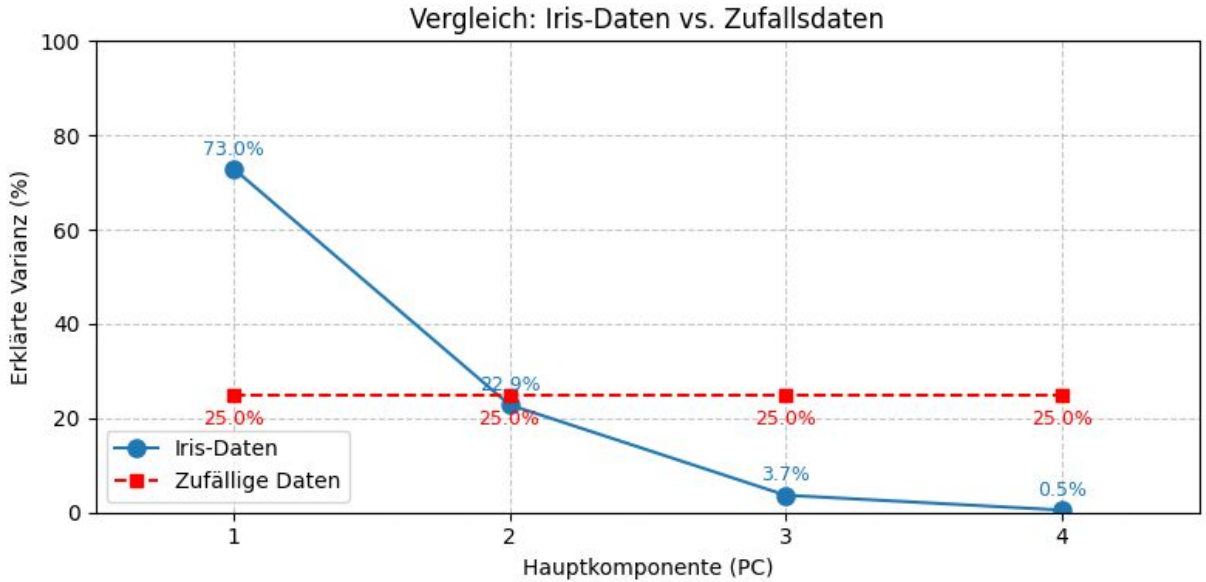
Schritt 4: Wahl der Hauptkomponenten

- Cattells Scree-Test
    - Annahme: PCs nach dem Knick erklären oft nur noch zufällige Variation.



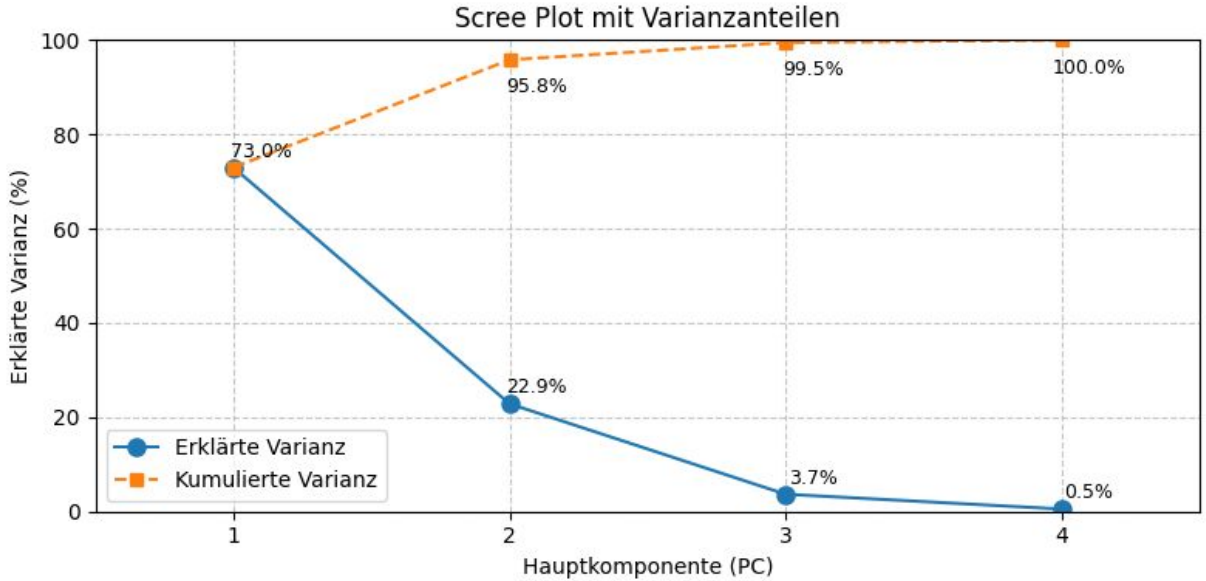
Schritt 4: Wahl der Hauptkomponenten

- Cattells Scree-Test
  - Annahme: PCs nach dem Knick erklären oft nur noch zufällige Variation.
  - Die Eigenwerte von Zufallszahlen verlaufen typischerweise annähernd konstant
  - Deshalb: Nur Komponenten links vom Knick (Knie) wählen



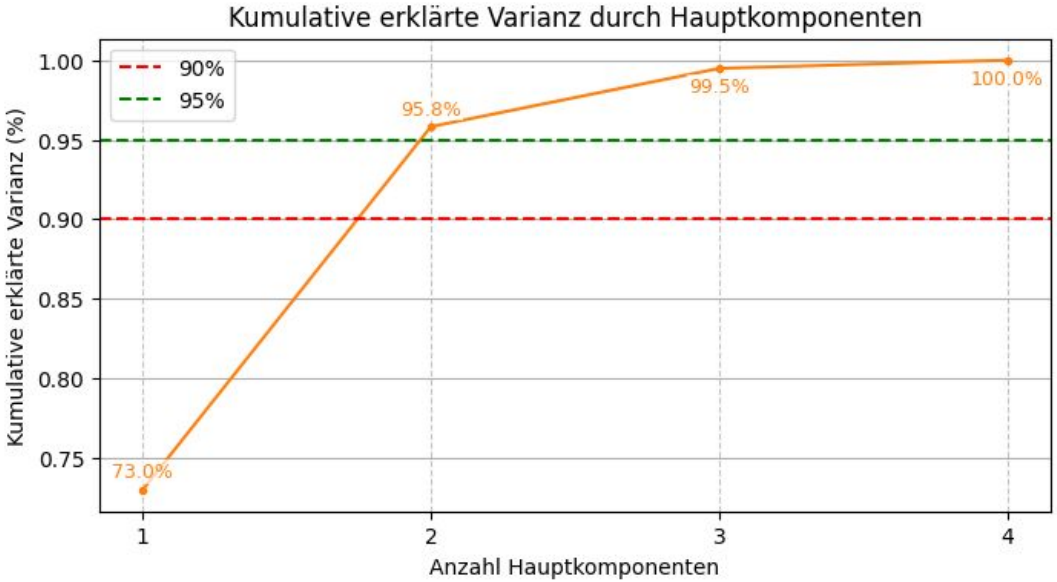
Schritt 4: Wahl der Hauptkomponenten

- Cattells Scree-Test
  - Annahme: PCs nach dem Knick erklären oft nur noch zufällige Variation.
  - Die Eigenwerte von Zufallszahlen verlaufen typischerweise annähernd konstant
  - Deshalb: Nur Komponenten links vom Knick (Knie) wählen



Schritt 4: Wahl der Hauptkomponenten

- Kumulierte Varianz (z. B. 95% Regel)
    - Idee: Wähle so viele PCs, dass ein festgelegter Varianzanteil erklärt wird.



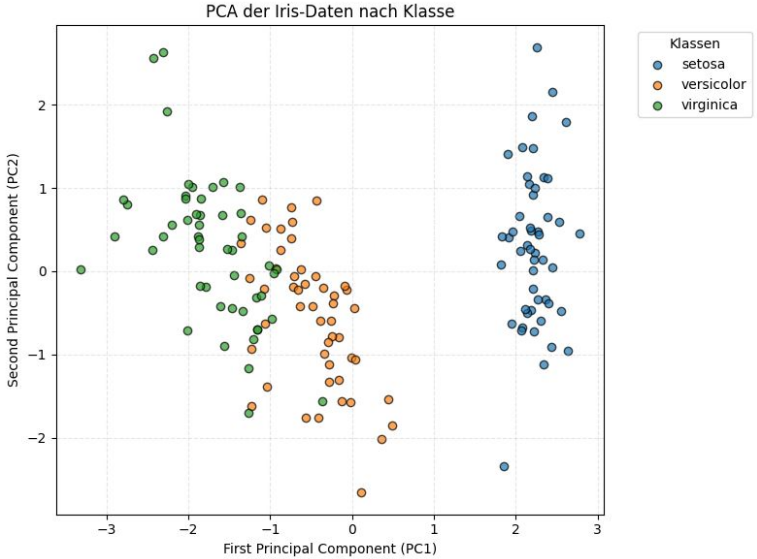
Schritt 5: Projektion

- Die Projektion erfolgt durch Multiplikation der zentrierten Daten mit der Eigenvektor-Matrix

```
# Projektion der Daten durch Matrix-Multiplikation der
# skalierten Daten mit der Matrix der gewählten Hauptkomponenten
X_pca = X_scaled.dot(eigenvectors[:, :num_components])
```

Projizierte Datenform: (150, 2)

	PC1	PC2
0	2.265	0.480
1	2.081	-0.674
2	2.364	-0.342
3	2.299	-0.597
4	2.390	0.647
5	2.076	1.489
6	2.444	0.048
7	2.233	0.223
8	2.335	-1.115
9	2.184	-0.469
10	2.166	1.044
11	2.326	0.133
12	2.218	-0.729
13	2.633	-0.962
14	2.199	1.860



PCA auf MNIST

- **MNIST**-Datenbank (Modified National Institute of Standards and Technology database)[3]
- öffentlich verfügbare Datenbank handgeschriebener Ziffern
- jede Ziffer: 28 × 28 Pixel großes Graustufen-Bild
- Die MNIST-Datenbank besteht aus 60.000 Beispielen im Trainingsdatensatz und 10.000 Beispielen im Testdatensatz
- Ziel: Training von Klassifikatoren

Abbildung: Wikimedia Commons. *A few samples from the MNIST test dataset*, https://commons.wikimedia.org/wiki/File:MnistExamples.png

Zusammenfassung

Schritt 1	Schritt 2	Schritt 3	Schritt 4	Schritt 5
Standardisierung	**Kovarianzmatrix, Eigenwerte & Eigenvektoren**	**Ladungsmatrix**	**Wahl der Komponenten**	**Projektion**
Zentrieren & Skalieren der Features (Mittelwert 0, Varianz 1)	Berechnung der paarweisen Kovarianzen zwischen Variablen (Lineare Abhängigkeiten), Eigenwerte (Varianz) & Eigenvektoren (Richtungen) extrahieren.	Beiträge der Variablen zu den Hauptkomponenten (Interpretation)	Wichtige PCs auswählen. (Dimensionsreduktion)	Daten in neuen Raum abbilden. (Transformation)

Anwendungen

PCA für Dimensionreduktion

(Ziel: Wesentliche Strukturen mit weniger Dimensionen darstellen)

- Computer Vision: Gesichtserkennung (Eigenfaces)
- ML/DS: Feature-Reduktion für bessere Modelle
- ML/DS: Datenvisualisierung (2D/3D-Projektion)
- Signalverarbeitung: Sprach- & Tonanalyse
  (z. B. Speaker Recognition)
- NLP: Word Embedding-Reduktion

PCA für Rekonstruktion/Kompression

(Ziel: Daten mit möglichst geringem Informationsverlust komprimieren)

- Bildverarbeitung: Bildkompression
- Big Data / DB: Effiziente Datenspeicherung
- Big Data / DB: Beschleunigung von Datenbankabfragen
- Netzwerke: Traffic-Komprimierung

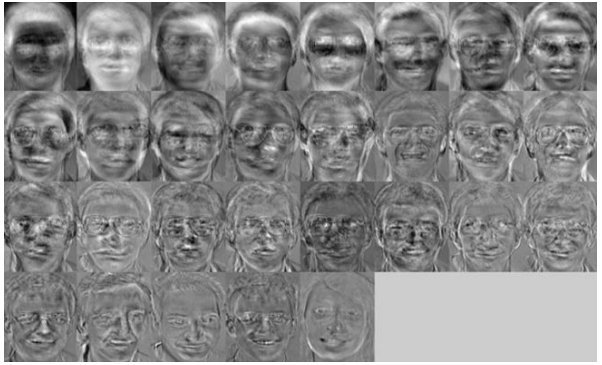


Abbildung: *Face Recognition: A set of M orthogonal face matrices (called Eigenfaces) are used to represent the original images, permitting significant reduction in computation recognition. 29 Eigenfaces shown in this figure are calculated for the original 500 face templates.* [4]

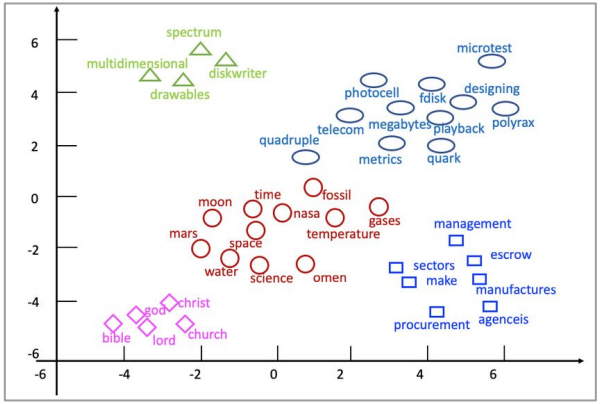


Abbildung: *2D PCA projection of word embeddings* [5]

Anwendungen

PCA für Dimensionreduktion

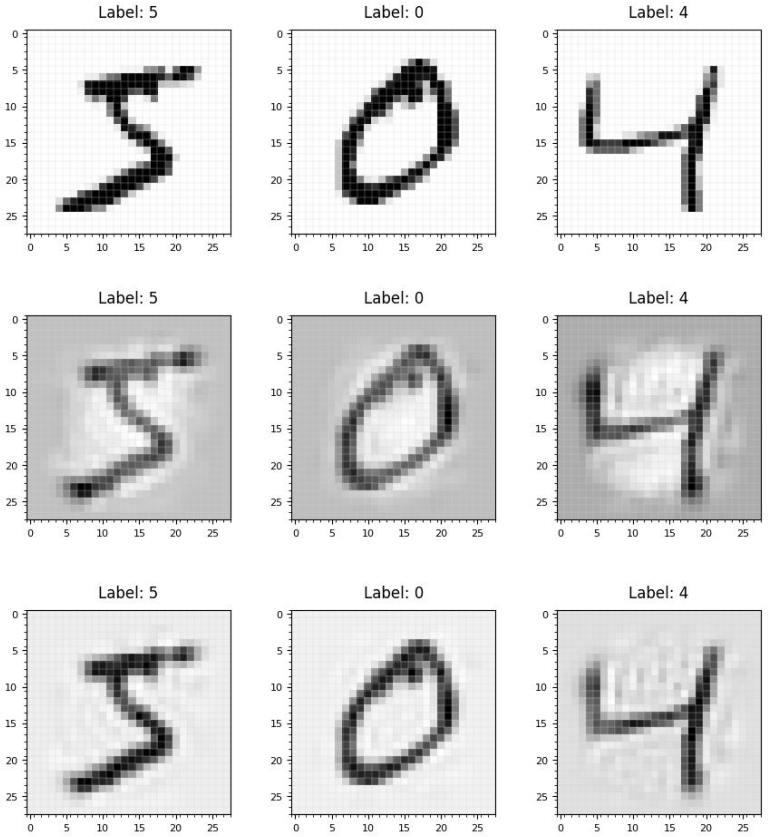
(Ziel: Wesentliche Strukturen mit weniger Dimensionen darstellen)

- Computer Vision: Gesichtserkennung (Eigenfaces)
- ML/DS: Feature-Reduktion für bessere Modelle
- ML/DS: Datenvisualisierung (2D/3D-Projektion)
- Signalverarbeitung: Sprach- & Tonanalyse
  (z. B. Speaker Recognition)
- NLP: Word Embedding-Reduktion

PCA für Rekonstruktion/Kompression

(Ziel: Daten mit möglichst geringem Informationsverlust komprimieren)

- Bildverarbeitung: Bildkompression
- Big Data / DB: Effiziente Datenspeicherung
- Big Data / DB: Beschleunigung von Datenbankabfragen
- Netzwerke: Traffic-Komprimierung



Grenzen der PCA

- Für numerische Daten
- Lineare Abhängigkeiten
- Sensitivität gegenüber Skalierung und Ausreißern
- Verlust von Informationen (wenn zu viele Komponenten entfernt werden)
- Hohe Rechenkomplexität bei großen Datensätzen
- Verlust der Interpretierbarkeit

Vielen Dank!

Fragen & Diskussion

Vorlesungsfolien
Übung mit Lösung
Jupyter Notebooks
Ergänzende Materialien

https://github.com/MartinaEchtenbruck/Principal-Component-Analysis

Kontakt:
Dr. Martina Echtenbruck
martina.echtenbruck@th-koeln.de

Referenzen

[1] Bellman, R. E. , *Adaptive Control Processes: A Guided Tour,* Princeton University Press. (1961)

[2] R. A. FISHER Sc.D., F.R.S., *THE USE OF MULTIPLE MEASUREMENTS IN TAXONOMIC PROBLEMS*, Annals of Eugenics, Wiley, https://doi.org/10.1111/j.1469-1809.1936.tb02137.x, (1936)

[3] Y. LeCun, *The MNIST Database of handwritten digits*, Courant Institute, NYU Corinna Cortes, Google Labs, New York Christopher J.C. Burges, Microsoft Research, Redmond. (orig. link broken, avail. via: https://web.archive.org/web/20200430193701/http://yann.lecun.com/exdb/mnist/)

[3] T. Soyata, R. Muraleedharan, C. Funai, M. Kwon, W. Heinzelman, *Cloud-Vision: Real-time Face Recognition Using a Mobile-Cloudlet-Cloud Acceleration Architecture*, Proceedings of the IEEE Symposium on Computers and Communications ISCC ‘12 (pp. 59-66), (2012)

[4] D. Li, J. Zhang, P. Li, *TMSA: A Mutual Learning Model for Topic Discovery and Word Embedding*, Proceedings of the 2019 SIAM International Conference on Data Mining (SDM), p. 684-692, https://epubs.siam.org/doi/10.1137/1.9781611975673.77

Weiterführende Inhalte

Fun Facts

- **Die erste Hauptkomponente war schon 1733 bekannt**
  Das mathematische Konzept hinter PCA (Eigenvektoren der Kovarianzmatrix) wurde bereits von *Leonhard Euler* und *Joseph-Louis Lagrange* im 18. Jahrhundert entwickelt. Die erste praktische Anwendung erfolgte aber erst 1936 durch *Harold Hotelling* – für Psychometrie!

- **PCA kann unmögliche Daten erzeugen**
  Bei der Rekonstruktion aus zu wenigen Komponenten produziert PCA manchmal "Geisterbilder". Bei MNIST können Ziffern plötzlich *zusätzliche Pixel* oder *falsche Striche* enthalten, die in keinem Originalbild vorkamen. Die KI "halluziniert" quasi fehlende Details.

- **PCA ist ein heimlicher Clustering-Algorithmus**
  In der Genetik wird PCA regelmäßig zur Entdeckung von Bevölkerungsgruppen genutzt. Die ersten Hauptkomponenten trennen oft automatisch ethnische Gruppen – ohne dass man Cluster-Algorithmen wie k-Means anwenden müsste. Ein Beispiel: Die erste PC der menschlichen DNA trennt meist Afrika von Eurasien.

- **Bonus: PCA hat einen geheimen Zwillingsalgorithmus**
  Die *Singulärwertzerlegung (SVD)* ist mathematisch äquivalent zu PCA, wird aber völlig anders berechnet.

Ansatz: Dimensionsreduktion

- **Ziel:** Dimension reduzieren
- **Aber**: Information beibehalten

a	b	c	d
8	3	0	17
5	0	1	11
5	0	1	11
3	-2	0	7
3	-2	1	7
8	3	1	17
4	-1	0	9
6	1	0	13
3	-2	1	7
2	-3	1	5

Ansatz: Dimensionsreduktion

- **Ziel:** Dimension reduzieren
- **Aber**: Information beibehalten

- Standardisierung:
  - Zentrierung: Mittelwert = 0
  - Skalierung: Standardabweichung = 1
  - Macht unsere unterschiedlichen Variablen (mathematisch) vergleichbar

a	b	c	d
8	3	0	17
5	0	1	11
5	0	1	11
3	-2	0	7
3	-2	1	7
8	3	1	17
4	-1	0	9
6	1	0	13
3	-2	1	7
2	-3	1	5

Mittelwert :	4,7	-0,3	0,6	10,4
Standardabweichung:	2	2	0,49	4

Ansatz: Dimensionsreduktion

- **Ziel:** Dimension reduzieren
- **Aber**: Information beibehalten

- Standardisierung:
  - Zentrierung: Mittelwert = 0
  - Skalierung: Standardabweichung = 1
  - Macht unsere unterschiedlichen Variablen (mathematisch) vergleichbar

a	b	c	d
1,65	1,65	-1,22	1,65
0,15	0,15	0,82	0,15
0,15	0,15	0,82	0,15
-0,85	-0,85	-1,22	-0,85
-0,85	-0,85	0,82	-0,85
1,65	1,65	0,82	1,65
-0,35	-0,35	-1,22	-0,35
0,65	0,65	-1,22	0,65
-0,85	-0,85	0,82	-0,85
-1,35	-1,35	0,82	-1,35

Ansatz: Dimensionsreduktion

- **Ziel:** Dimension reduzieren
- **Aber**: Information beibehalten

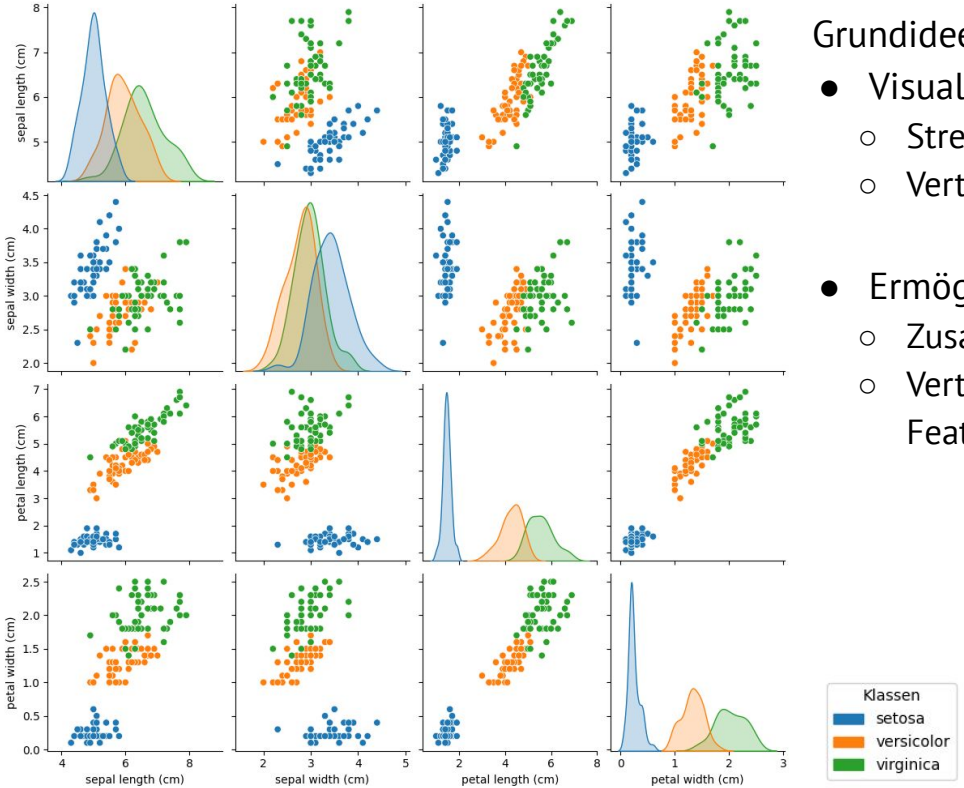
- Standardisierung:
  - Zentrierung: Mittelwert = 0
  - Skalierung: Standardabweichung = 1
  - Macht unsere unterschiedlichen Variablen (mathematisch) vergleichbar

➡ **Principal Component Analysis** (PCA)

- Lineare Zusammenhänge zwischen Variablen
- Varianz = Information

a	b	c	d
1,65	1,65	-1,22	1,65
0,15	0,15	0,82	0,15
0,15	0,15	0,82	0,15
-0,85	-0,85	-1,22	-0,85
-0,85	-0,85	0,82	-0,85
1,65	1,65	0,82	1,65
-0,35	-0,35	-1,22	-0,35
0,65	0,65	-1,22	0,65
-0,85	-0,85	0,82	-0,85
-1,35	-1,35	0,82	-1,35

Iris Datensatz



Grundidee Pairplot:

- Visualisierung aller Variablen-Paare
  - Streudiagramme zwischen Variablen
  - Verteilungen auf der Diagonalen

- Ermöglicht schnelle Einsicht:
  - Zusammenhänge (Korrelationen)
  - Verteilungen und Cluster zwischen Features

Abbildungen: *Iris Setosa, Iris Versicolor, Iris Virginica*. Quelle: Wikipedia.org

Schritt 1
Standardisieren

Schritt 1: Standardisieren

- Bei der Standardisierung werden folgende Schritte durchgeführt:
- Zentrierung:
  - Verschiebung der Daten so, dass gilt: Mittelwert=0
  - Zwingend erforderlich, da die erste Hauptkomponente sonst nicht in Richtung der größten Varianz, sondern in Richtung der Mitte der Daten zeigen würde.
- Skalierung:
  - Skalierung der Daten so, dass gilt: Standardabweichung=1
  - Sinnvoller Schritt, da sonst unausgewogene Daten für unausgewogene Gewichtungen in den Kovarianzen sorgen würden

Schritt 1: Standardisieren

Ein zufällig gestreute Variable X:

- Der Erwartungswert (Mittelwert) von X : $E(X) = \mu$ $(= \bar{x})$,
- die Varianz von X: $Var(X) = \sigma^2$
- die Standardabweichung X: σ

Die zugehörige standardisierte Zufallsvariable Z erhält man durch Zentrierung und anschließende Division durch die Standardabweichung:

$$Z = \frac{X - \mu}{\sigma}$$

Für die Zufallsvariable Z gilt:

- Der Erwartungswert von Z: $E(Z) = 0$
- Die Varianz von Z: $Var(Z) = 1$
- Z ist also Standardnormalverteilt.

Schritt 1: Standardisieren

Ohne Zentrierung wird die erste Hauptkomponente (PC1) durch die Lage des Mittelwerts verzerrt und zeigt nicht entlang der Richtung größter Varianz, sondern in Richtung des Mittelpunkts der Daten.

Warum?

1. Die PCA sucht Richtungen (Hauptkomponenten), in denen die Varianz der Daten maximal ist. Dazu wird die Kovarianzmatrix der Daten berechnet, und deren Eigenvektoren geben die Hauptkomponenten.
2. Die Kovarianz-Matrix basiert auf der Formel: $\mathrm{Cov}(X) = \frac{1}{n-1}(X - \bar{X})^T(X - \bar{X})$
3. Ohne Zentrierung wird diese Matrix berechnet: $\frac{1}{n-1}X^T X$
4. Diese enthält dann Informationen über die Lage der Daten im Raum, nicht über deren Streuung.
5. Der so entstandene “Offset” vom Mittelpunkt wird fälschlicherweise als Varianz interpretiert.

Schritt 2

Kovarianz-Matrix, Eigenwerte und Eigenvektoren

Schritt 2: Kovarianz-Matrix, Eigenwerte & Eigenvektoren

- Die **Varianz** misst die Streuung einer Variable um ihren Mittelwert

- Die Varianz einer Stichprobe als Schätzwert für die Grundgesamtheit: $$Var(a) = \frac{1}{n-1} \sum_{i=1}^{n} (a_i - \bar{a})^2$$

  Mit $\bar{a}$, den Mittelwert von a_i, und n die Anzahl der Datenpunkte.

Schritt 2: Kovarianz-Matrix, Eigenwerte & Eigenvektoren

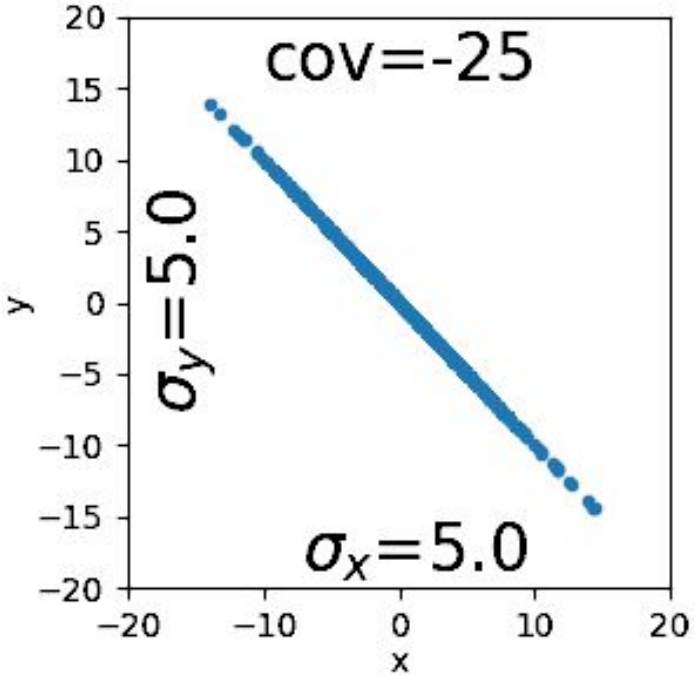
- Die **Kovarianz** misst den linearen Zusammenhang zwischen zwei Variablen x und y.

- Die Kovarianz zweier Stichproben als Schätzwert für die Grundgesamtheit: $$Cov(a,b) = \frac{1}{n-1}\sum_{i=1}^{n}(a_i - \bar{a})(b_i - \bar{b})$$

  Mit $\bar{a}$ und $\bar{b}$, den Mittelwerten von a_i und b_i, und n die Anzahl der Datenpunkte.

- *Cov(a,b) > 0* ⟶ x und y steigen / fallen gemeinsam.
- *Cov(a,b) = 0* ⟶ kein linearer Zusammenhang.
- *Cov(a,b) < 0* ⟶ x und y steigen / fallen entgegengesetzt.

Schritt 2: Kovarianz-Matrix, Eigenwerte & Eigenvektoren



- *Cov(a,b)* > *0* → x und y steigen / fallen gemeinsam.
- *Cov(a,b)* = *0* → kein linearer Zusammenhang.
- *Cov(a,b)* < *0* → x und y steigen / fallen entgegengesetzt.

Abbildung: "*Normalverteilungen zweier Variablen mit unterschiedlicher Kovarianz*"
Quelle: Physikinger, CC0, via Wikimedia Commons,
https://commons.wikimedia.org/wiki/File:Varianz.gif

Schritt 2: Kovarianz-Matrix, Eigenwerte & Eigenvektoren

- Die Hauptdiagonale enthält Varianzen
- Die übrigen Stellen enthalten die Kovarianzen zwischen verschiedenen Variablen.
- Die Matrix ist symmetrisch zur Hauptdiagonalen

$$\Sigma = \begin{pmatrix} Var(a) & Cov(a,b) & \cdots & Cov(a,l) & Cov(a,m) \\ Cov(b,a) & Var(b) & & Cov(b,l) & Cov(b,m) \\ \vdots & & \ddots & & \vdots \\ Cov(l,a) & Cov(l,b) & & Var(l) & Cov(l,m) \\ Cov(m,a) & Cov(m,b) & \cdots & Cov(m,l) & Var(m) \end{pmatrix}$$

Schritt 2: Kovarianz-Matrix, Eigenwerte & Eigenvektoren

Berechnung der Kovarianz-Matrix:

- Die Datenbasis notieren wir formell als Datenmatrix X
  - X ist eine n x p-Matrix mit:
    - n Zeilen (Anzahl Beobachtungen / Einträge)
    - p Spalten (Anzahl der Variablen)

- $\bar{X}$ ist die Matrix der Mittelwerte
  (jede Spalte enthält den Mittelwert der Variable)

- Die Kovarianz-Matrix Σ wird aus X berechnet:

$$\Sigma = \frac{1}{n-1}(X - \bar{X})(X - \bar{X})^T$$

Kovarianz-Matrix

Die Kovarianz für eine Stichprobe:

$$Cov(a,b) = \frac{1}{n-1}\sum_{i=1}^{n}(a_i - \bar{a})(b_i - \bar{b})$$

i	a	b	c
1	-1	-1	1
2	1	-1	-5
3	3	2	1
4	5	3	-1
Mittelwert	2	3	-1

$$\Sigma = \begin{pmatrix} Var(a) & Cov(a,b) & Cov(a,c) \\ Cov(b,a) & Var(b) & Cov(b,c) \\ Cov(c,a) & Cov(c,b) & Var(c) \end{pmatrix}$$

Kovarianz-Matrix

Cov(a,b) = ⅓ ((-3)·(-4) + (-1)·(-4) + 1·(-1) + 3·0)
= ⅓ (12 + 4 - 1 + 0)
= ⅓ 15 = 5

Cov(a,c) = ⅓ ((-3)·2 + (-1)·(-4) + 1·2 + 3·0)
= ⅓ (- 6 + 4 + 2 + 0)
= ⅓ 0 = 0

Cov(b,c) = ⅓ ((-4)·2 + (-4)·(-4) + (-1)·2 + 0)
= ⅓ (- 8 + 16 - 2)
= ⅓ 6 = 2

$$\Sigma = \begin{pmatrix} Var(a) & Cov(a,b) & Cov(a,c) \\ Cov(b,a) & Var(b) & Cov(b,c) \\ Cov(c,a) & Cov(c,b) & Var(c) \end{pmatrix}$$

i	$a_i - \bar{a}$	$b_i - \bar{b}$	$c_i - \bar{c}$
1	-3	-4	2
2	-1	-4	-4
3	1	-1	2
4	3	0	0

$$Cov(a,b) = \frac{1}{n-1}\sum_{i=1}^{n}(a_i - \bar{a})(b_i - \bar{b})$$

Kovarianz-Matrix

Cov(a,b) = ⅓ ((-3)·(-4) + (-1)·(-4) + 1·(-1) + 3·0)
= ⅓ (12 + 4 - 1 + 0)
= ⅓ 15 = 5

Cov(a,c) = ⅓ ((-3)·2 + (-1)·(-4) + 1·2 + 3·0)
= ⅓ (- 6 + 4 + 2 + 0)
= ⅓ 0 = 0

Cov(b,c) = ⅓ ((-4)·2 + (-4)·(-4) + (-1)·2 + 0)
= ⅓ (- 8 + 16 - 2)
= ⅓ 6 = 2

$$\Sigma = \begin{pmatrix} Var(a) & 5 & 0 \\ 5 & Var(b) & 2 \\ 0 & 2 & Var(c) \end{pmatrix}$$

i	$a_i - \bar{a}$	$b_i - \bar{b}$	$c_i - \bar{c}$
1	-3	-4	2
2	-1	-4	-4
3	1	-1	2
4	3	0	0

$$Cov(a,b) = \frac{1}{n-1}\sum_{i=1}^{n}(a_i - \bar{a})(b_i - \bar{b})$$

Kovarianz-Matrix

Var(a) = ⅓ ($(-3)^2 + (-1)^2 + 1^2 + 3^2$)
= ⅓ (9 + 1 + 1 + 9)
= ⅓ 20 ≈ 6,67

Var(b) = ⅓ ($(-4)^2 + (-4)^2 + (-1)^2 + 0^2$)
= ⅓ (16 + 16 + 1)
= ⅓ 33 = 11

Var(c) = ⅓ ($2^2 + (-4)^2 + 2^2 + 0^2$)
= ⅓ (4 + 16 + 4)
= ⅓ 24 = 8

$$\Sigma = \begin{pmatrix} Var(a) & 5 & 0 \\ 5 & Var(b) & 2 \\ 0 & 2 & Var(c) \end{pmatrix}$$

i	$a_i - \bar{a}$	$b_i - \bar{b}$	$c_i - \bar{c}$
1	-3	-4	2
2	-1	-4	-4
3	1	-1	2
4	3	0	0

$$Var(a) = \frac{1}{n}\sum_{i=1}^{n}(a_i - \bar{a})^2$$

Kovarianz-Matrix

Var(a) = ⅓ ($(-3)^2 + (-1)^2 + 1^2 + 3^2$)
= ⅓ (9 + 1 + 1 + 9)
= ⅓ 20 ≈ 6,67

Var(b) = ⅓ ($(-4)^2 + (-4)^2 + (-1)^2 + 0^2$)
= ⅓ (16 + 16 + 1)
= ⅓ 33 = 11

Var(c) = ⅓ ($2^2 + (-4)^2 + 2^2 + 0^2$)
= ⅓ (4 + 16 + 4)
= ⅓ 24 = 8

$$\Sigma = \begin{pmatrix} 6{,}67 & 5 & 0 \\ 5 & 11 & 2 \\ 0 & 2 & 8 \end{pmatrix}$$

i	$a_i - \bar{a}$	$b_i - \bar{b}$	$c_i - \bar{c}$
1	-3	-4	2
2	-1	-4	-4
3	1	-1	2
4	3	0	0

$$Var(a) = \frac{1}{n}\sum_{i=1}^{n}(a_i - \bar{a})^2$$

Schritt 2: Kovarianz-Matrix, Eigenwerte & Eigenvektoren

Berechnung der Eigenwerte und Eigenvektoren der Kovarianzmatrix Σ:

- Als einfaches Beispiel sei $\boldsymbol{\Sigma}$ ein 2x2 Matrix: $\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 2 \\ 3 & 2 \end{pmatrix}$
- Sei $\boldsymbol{E}$ die Einheitsmatrix. Die Matrix $\boldsymbol{\Sigma}$-$\lambda\boldsymbol{E}$ wird dann als charakteristische Matrix von $\boldsymbol{\Sigma}$ *bezeichnet:*
  $\boldsymbol{\Sigma} - \lambda\boldsymbol{E} = \begin{pmatrix} 1 & 2 \\ 3 & 2 \end{pmatrix} - \lambda \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 1-\lambda & 2 \\ 3 & 2-\lambda \end{pmatrix}$
- Zur Bestimmung der Eigenvektoren muss nun das Lineare Gleichungssystem gelöst werden:
  $\begin{pmatrix} 1-\lambda & 2 \\ 3 & 2-\lambda \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$
- Dies hat nur dann eine nicht-triviale Lösung, wenn die Determinante verschwindet: $det(\boldsymbol{\Sigma}\text{-}\lambda\boldsymbol{E}) = 0$.
  $det(\boldsymbol{\Sigma}\text{-}\lambda\boldsymbol{E}) = (1-\lambda)(2-\lambda) - 2 \cdot 3 = 0$
  Dies ist die Charakteristische Gleichung der Matrix $\boldsymbol{\Sigma}$.
- Die Lösungen der Charakteristischen Gleichung, $\lambda_1 = -1$ *und* $\lambda_2 = 4$, sind die Eigenwerte der Matrix $\boldsymbol{\Sigma}$.

Schritt 2: Kovarianz-Matrix, Eigenwerte & Eigenvektoren

Berechnung der Eigenvektoren der Kovarianzmatrix Σ:

- Setzen wir nun in das Gleichungssystem: $\begin{pmatrix} 1-\lambda & 2 \\ 3 & 4-\lambda \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$

  nacheinander die Eigenwerte ein, dann können wir die zugehörigen Eigenvektoren bestimmen:

- λ_1=-1 :

  $\boldsymbol{\Sigma} - (-\mathbf{1})\boldsymbol{E} = \begin{pmatrix} 1-(-1) & 2 \\ 3 & 2-(-1) \end{pmatrix} = \begin{pmatrix} 2 & 2 \\ 3 & 3 \end{pmatrix}$

  Es ergeben sich die Gleichungen:

  $2x_1 + 2x_2 = 0$

  $3x_1 + 3x_2 = 0$

- Auflösen nach x_1 ergibt: $x_1 = -x_2$

- Ein Eigenvektor für λ_1 ist also: $\overrightarrow{v_1} = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$

- λ_2=4 :

  $\boldsymbol{\Sigma} - \mathbf{4}\boldsymbol{E} = \begin{pmatrix} 1-4 & 2 \\ 3 & 2-4 \end{pmatrix} = \begin{pmatrix} -3 & 2 \\ 3 & -2 \end{pmatrix}$

  Es ergeben sich die Gleichungen:

  $-3x_1 + 2x_2 = 0$

  $3x_1 - 2x_2 = 0$

- Auflösen nach x_1 ergibt: $x_1 = 2/3x_2$

- Ein Eigenvektor für λ_2 ist also: $\overrightarrow{v_2} = \begin{pmatrix} 2 \\ 3 \end{pmatrix}$

Schritt 3
Ladungs-Matrix

Schritt 4

Wahl der Komponenten

Schritt 4: Wahl der Hauptkomponenten

- Kaiser-Guttman-Kriterium
  - Idee: nur diejenigen Faktoren beibehalten, die mehr Varianz erklären als die ursprünglichen Variablen
  - Dies ist nur bei Komponenten mit Eigenwerten größer eins gegeben

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10	PC11	PC12	PC13
alcohol	-0,1443	-0,4837	-0,2074	0,0179	-0,2657	-0,2135	0,0564	-0,3961	0,5086	0,2116	0,2259	0,2663	-0,0150
malic_acid	0,2452	-0,2249	0,0890	-0,5369	0,0352	-0,5368	-0,4205	-0,0658	-0,0753	-0,3091	-0,0765	-0,1217	-0,0260
ash	0,0021	-0,3161	0,6262	0,2142	-0,1430	-0,1545	0,1492	0,1703	-0,3077	-0,0271	0,4987	0,0496	0,1412
alcalinity_of_ash	0,2393	0,0106	0,6121	-0,0609	0,0661	0,1008	0,2870	-0,4280	0,2004	0,0528	-0,4793	0,0557	-0,0917
magnesium	-0,1420	-0,2996	0,1308	0,3518	0,7270	-0,0381	-0,3229	0,1564	0,2714	0,0679	-0,0713	-0,0622	-0,0568
total_phenols	-0,3947	-0,0650	0,1462	-0,1981	-0,1493	0,0841	0,0279	0,4059	0,2860	-0,3201	-0,3043	0,3039	0,4639
flavanoids	-0,4229	0,0034	0,1507	-0,1523	-0,1090	0,0189	0,0607	0,1872	0,0496	-0,1632	0,0257	0,0429	-0,8323
nonflavanoid_phenols	0,2985	-0,0288	0,1704	0,2033	-0,5007	0,2586	-0,5954	0,2333	0,1955	0,2155	-0,1169	-0,0424	-0,1140
proanthocyanins	-0,3134	-0,0393	0,1495	-0,3991	0,1369	0,5338	-0,3721	-0,3682	-0,2091	0,1342	0,2374	0,0956	0,1169
color_intensity	0,0886	-0,5300	-0,1373	-0,0659	-0,0764	0,4186	0,2277	0,0338	0,0562	-0,2908	-0,0318	-0,6042	0,0120
hue	-0,2967	0,2792	0,0852	0,4278	-0,1736	-0,1060	-0,2321	-0,4366	0,0858	-0,5224	0,0482	-0,2592	0,0899
od280/od315_of_diluted_wines	-0,3762	0,1645	0,1660	-0,1841	-0,1012	-0,2659	0,0448	0,0781	0,1372	0,5237	-0,0464	-0,6010	0,1567
proline	-0,2868	-0,3649	-0,1267	0,2321	-0,1579	-0,1197	-0,0768	-0,1200	-0,5758	0,1621	-0,5393	0,0794	-0,0144
Erklärte Varianz	4,7324	2,5111	1,4542	0,9242	0,8580	0,6453	0,5541	0,3505	0,2905	0,2523	0,2271	0,1697	0,1040
Anteil (%)	36,20	19,21	11,12	7,07	6,56	4,94	4,24	2,68	2,22	1,93	1,74	1,30	0,80

Erklärte Varianz: 66,53%

Schritt 5
Projektion

Projektion

- Die Projektion ist mathematisch eine Matrixmultiplikation mit den ausgewählten Eigenvektoren:

  $\boldsymbol{Z}=\boldsymbol{X}\cdot\boldsymbol{W}_k$

  - $\boldsymbol{X}$ die Originaldaten,
  - $\boldsymbol{W}$ die Matrix der k ausgewählten Hauptkomponenten
  - $\boldsymbol{Z}$ die Daten nach der Projektion in den neuen Raum

Weiterführende Inhalte

Residualmatrix

- Bei der PCA wird eine Datenmatrix $\boldsymbol{X}$ (mit *n* Beobachtungen und *p* Features) in **Hauptkomponenten** zerlegt.
- Die **Residualmatrix** $\boldsymbol{E}$ (auch *Fehlermatrix* genannt) misst, wie viel Information **nicht** durch die ausgewählten Hauptkomponenten erklärt wird.
    - $\boldsymbol{E}$ gibt die Differenz zwischen den **Originaldaten** X und der **Rekonstruktion** $\hat{X}$ aus den Hauptkomponenten an:

      $$E = X - \hat{X}$$

Skalenniveaus

Nominal
Ausprägungen können **unterschieden** werden

Ordinal
Ausprägungen können **sortiert** werden

A < B < C < D

Metrisch
Abstände zwischen den Ausprägungen können berechnet werden

Nominalskala (Nominal Scale)

- Rein qualitative Unterscheidung **ohne Rangordnung oder Abstände**.
- Werte sind "Labels" (Kategorien).
- **Beispiele:**
  - Augenfarbe (blau, grün, braun)
  - Berufsgruppen (Lehrer, Ingenieur, ...)
  - Binäre Werte (Ja/Nein, 0/1)

Ordinalskala (Ordinal Scale)

- **Definition:**
  - Kategorien mit **Rangfolge**, aber **nicht quantifizierbaren Abständen**.
  - "Besser/schlechter"-Relationen, aber nicht "wie viel besser".
- **Beispiele:**
  - Schulnoten (1, 2, 3, 4, 5, 6)
  - Likert-Skalen ("stimme nicht zu" bis "stimme voll zu")
  - Krankheitsstadien (leicht, mittel, schwer)

Metrische Skala (Metric Scale)

Umfasst **Intervall- und Verhältnisskalen**

a) Intervallskala (Interval Scale)
Numerische Werte mit **gleichen Abständen**, aber **keinem natürlichen Nullpunkt**.

- **Beispiele:**
  - Temperatur in °C (0°C ≠ "keine Temperatur")
  - IQ-Scores (IQ 0 ≠ "keine Intelligenz")

b) Verhältnisskala (Ratio Scale)
Numerische Werte mit **natürlichem Nullpunkt** und **interpretierbaren Verhältnissen**.

- **Beispiele:**
  - Alter (0 Jahre = "kein Alter")
  - Einkommen (0 € = "kein Einkommen")
  - Reaktionszeit in Sekunden (0 s = "keine Zeit")

Weitere Verfahren zur Dimensionsreduktion

- SVD
- Faktoranalyse
- t-SNE (t-distributed Stochastic Neighbor Embedding)
- Autoencoder (Neuronale Netze)