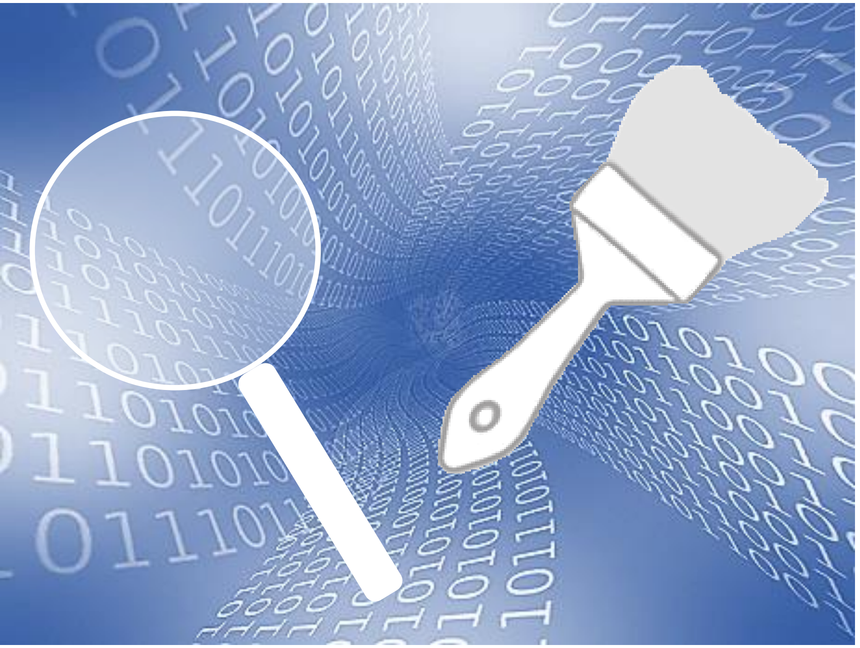
Prof. Dr. Bastian Amberg
School of Business & Economics
Department of Information Systems



Business Intelligence

07 Data Visualization II

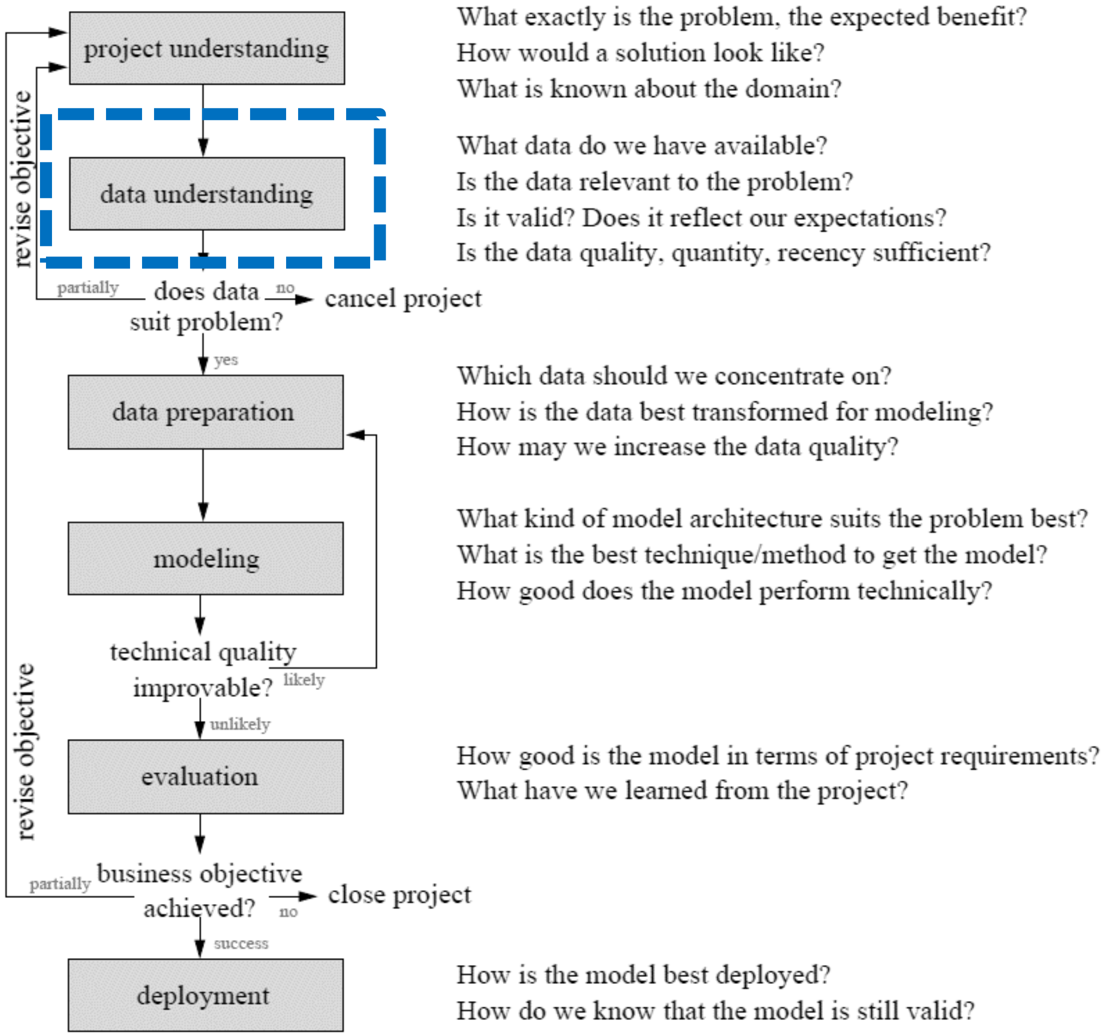
Prof. Dr. Bastian Amberg
(summer term 2024)
29.5.2024

Schedule (slightly adjusted)

		Wed., 10:00-12:00		Fr., 14:00-16:00 (Start at 14:30)		Self-study		
Basics	W1	17.4.	(Meta-)Introduction	19.4.		Python-Basics	Chap. 1	
	W2	24.4.	Data Warehouse – Overview & OLAP	26.4.	*[Blockveranstaltung SE Prof. Gersch]*		Chap. 2	
	W3	1.5.		3.5.			Chap. 3	
	W4	8.5.	Data Warehouse Modeling I & II	10.5.	Data Mining Introduction			
Main Part	W5	15.5.	CRISP-DM, Project understanding	17.5.	Python-Basics-Online Exercise	Python-Analytics	Chap. 1	
	W6	22.5.	Data Understanding, Data Visualization I	24.5.	*No lectures, but bonus tasks* *1.) Co-Create your exam* *2.) Earn bonus points for the exam*		Chap. 2	
	W7	29.5.	Data Visualization II	31.5.				
	W8	5.6.	Data Preparation	7.6.	Predictive Modeling I (10:00 -12:00)	BI-Project	Start	Case Study
	W9	12.6.	Predictive Modeling II, Fitting a Model I	14.6.	Python-Analytics-Online Exercise		|	
	W10	19.6.	*Guest Lecture Dr. Ionescu*	21.6.	Fitting a Model II		|	
	W11	26.6.	How to avoid overfitting	28.6.	What is a good Model?		|	
Deep-ening	W12	3.7.	Project status update Evidence and Probabilities	5.7.	Similarity (and Clusters) From Machine to Deep Learning I		|	
	W13	10.7.		12.7.	From Machine to Deep Learning II		|	
	W14	17.7.	Project presentation	19.7.	Project presentation		End	
Ref.					*Klausur 1.Termin, 31.7.'24* *Klausur 2.Termin, 2.10.'24*	Projektbericht		

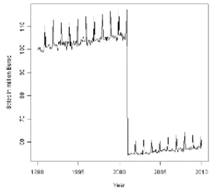
Last Lesson

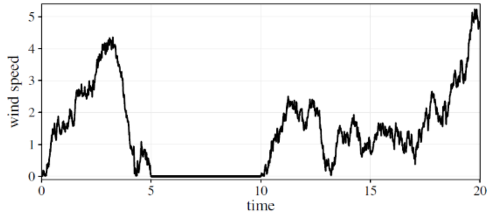
Data understanding I (attribute understanding, data quality)



Short Introduction

Data understanding II (data visualization, correlation analysis)





Low-dimensional relationships

Univariate Analysis

Bivariate Analysis

Higher-dimensional relationships

Principal Component Analysis

Parallel Coordinates

Ref.

Agenda

Low-dimensional relationships

Univariate Analysis

Bivariate Analysis

Higher-dimensional relationships

Principal Component Analysis

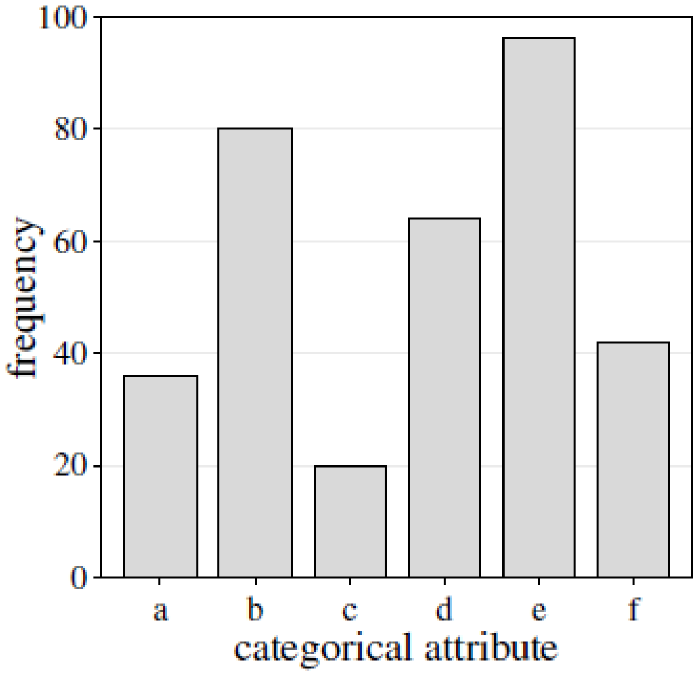
Parallel Coordinates

Ref.

Common visualizations

Bar charts and Histograms

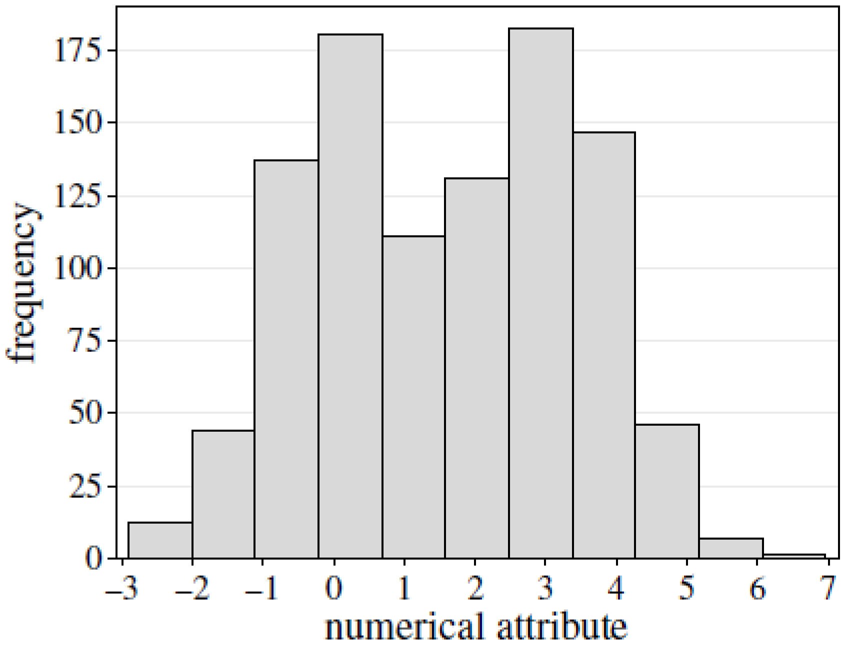
A **bar chart** is a simple way to depict the frequencies of the values of a categorical attribute.



A **histogram** shows the frequency distribution for a numerical attribute.

The range of numerical attribute is discretized into a fixed number of intervals (“bins”), usually of equal length.

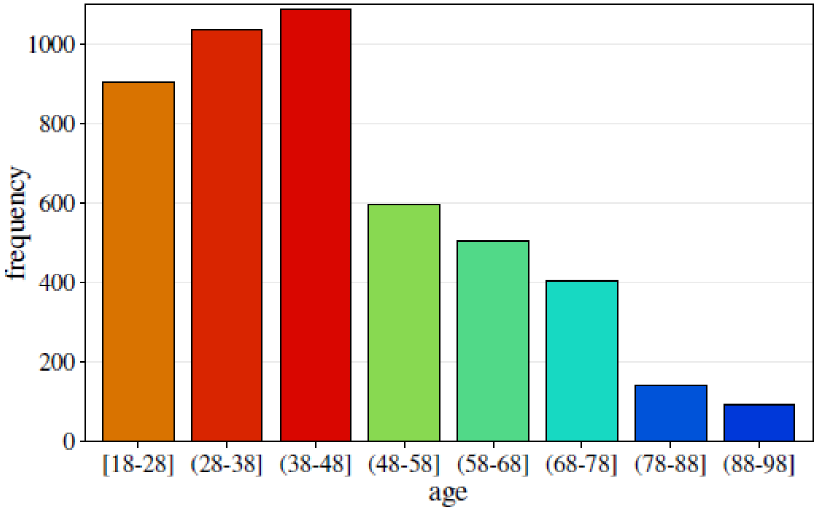
For each interval, the (absolute) frequency of values falling into it is indicated by the height of a bar.

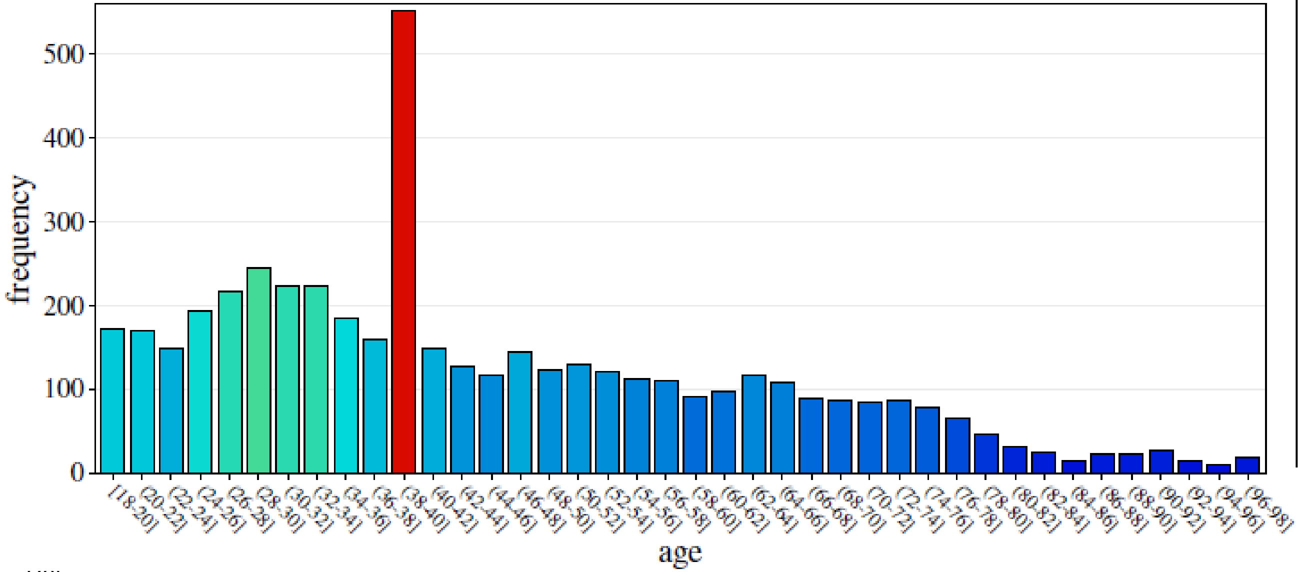


Ref.

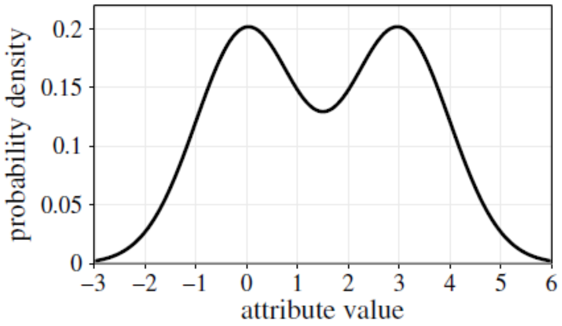
Common visualizations

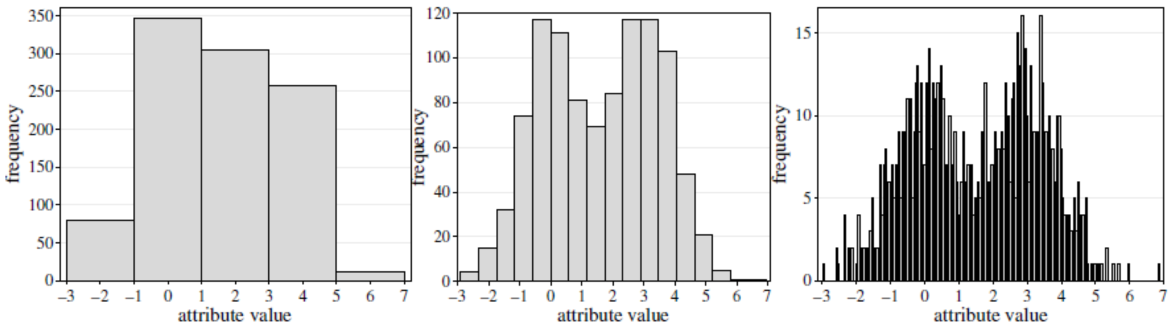
Histograms: The number of bins is very important.





Three histograms with 5, 17 and 200 bins for a sample from the same bimodal distribution. Sample size is n = 1000.





Ref.

Example data set

Iris data

Collected by E. Anderson in 1935

Contains measurements of four real-valued variables of 150 **iris flowers** of types Iris Setosa, Iris Versicolor, Iris Virginica

- Sepal length [Kelchblatt]
- Sepal widths
- Petal lengths [Blütenblatt]
- Petal widths

The fifth attribute is the name of the flower type

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	Iris-setosa
...				
...				
5.0	3.3	1.4	0.2	Iris-setosa
7.0	3.2	4.7	1.4	Iris-versicolor
...				
...				
5.1	2.5	3.0	1.1	Iris-versicolor
5.7	2.8	4.1	1.3	Iris-virginica
...				
...				
5.9	3.0	5.1	1.8	Iris-virginica

Ref.

Iris Setosa

Borsten-Schwertlilie

Iris Versicolor

Versch-f. Schwertl.

Iris Virginica

Sumpf-Schwertlilie

```python
import pandas as pd
# Create DataFrame using Pandas and set Column names
iris = pd.read_csv('irisData.csv', names=['sepal_length','sepal_width','petal_length','petal_width','species'])
# Show descriptive statistics on dimensional distributions
print(iris.describe())
# Show histogram
iris.hist(column='sepal_length', bins = (4.0,4.5,5.0,5.5,6.0,6.5,7.0,7.5,8))
```

Iris data set: boxplots

Boxplots are a very compact way to visualize and summarize main characteristics of a sample from a numerical attribute

Line in the middle = median

Box = interquartile range

Whiskers = 1.5 x interquartile range

```
import pandas as pd
import seaborn as sns

iris = pd.read_csv('irisData.csv', names=['sepal_length','sepal_width','petal_length','petal_width','species'])

sns.boxplot(x="species", y="sepal_length", data=iris, notch=True)
```

python

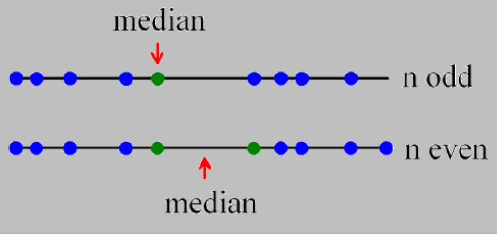
Reminder:

Median:
the value in the middle (for the values given in increasing order)

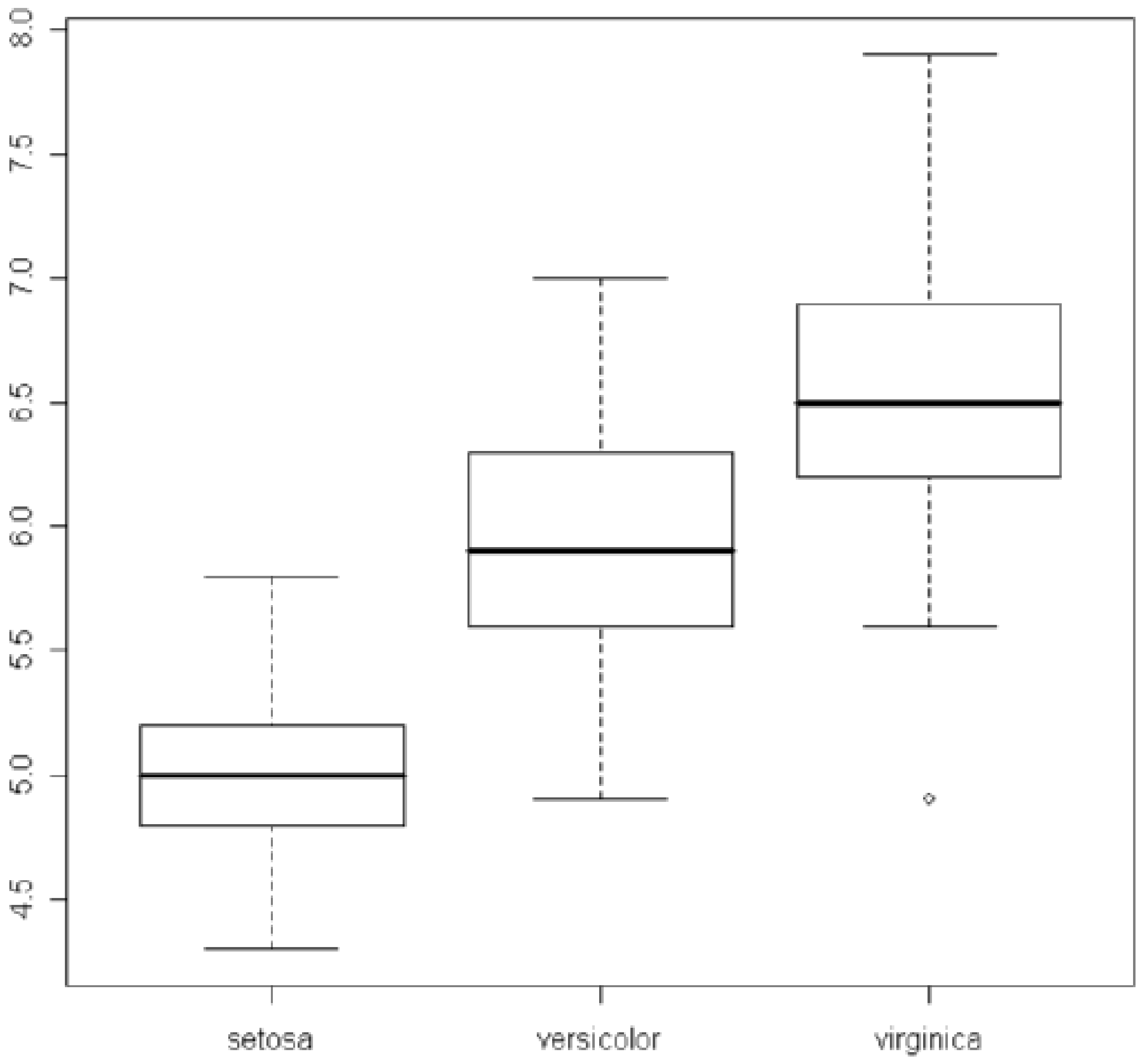
q%-quantile (0<q<100):
The value for which q% of the values are smaller and 100-q% are larger. The median is the 50%-quantile.

Quartiles:
25%-quantile (1st), median (2nd), 75%-quantile (3rd)

Interquartile range:
3rd quantile – 1st quantile



Sepal length (Kelchblattlänge)



Agenda

Low-dimensional relationships

Univariate Analysis

Bivariate Analysis

Higher-dimensional relationships

Principal Component Analysis

Parallel Coordinates

Ref.

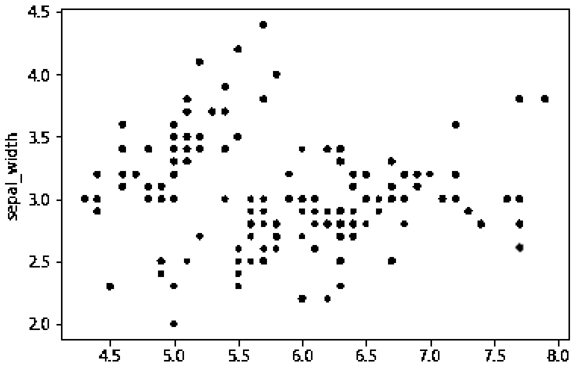
Common visualizations

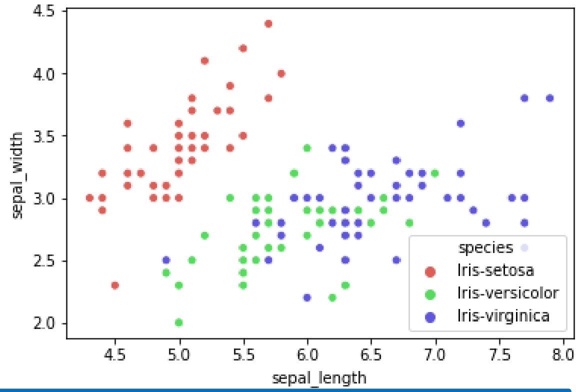
Scatter plots

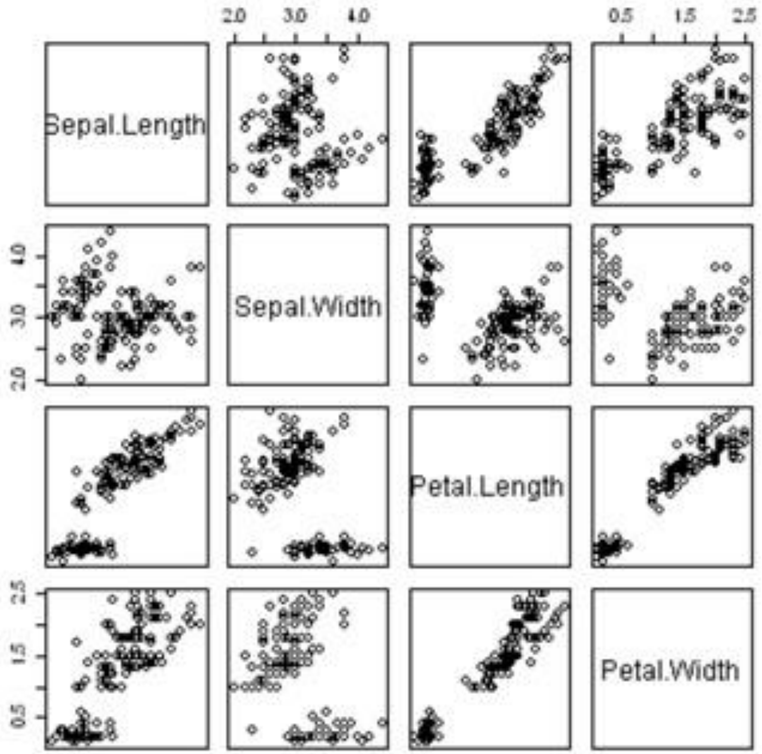
Scatter plots visualize two variables in a two-dimensional plot

Each axes corresponds to one variable

Not suited for larger data sets








```
import pandas as pd
import seaborn as sns

iris = pd.read_csv('irisData.csv', names=['sepal_length','sepal_width','petal_length','petal_width','species'])

# Describe relationships amoung variables in scatter plot
# hue: Variable used for color mapping
sns.scatterplot(data=iris, x="sepal_length", y="sepal_width", hue="species", palette="hls")

# Plot pairwise relationships in a dataset.
sns.pairplot(iris, hue="species", palette="hls")
# see https://seaborn.pydata.org/generated/seaborn.pairplot.html
```

Common visualizations

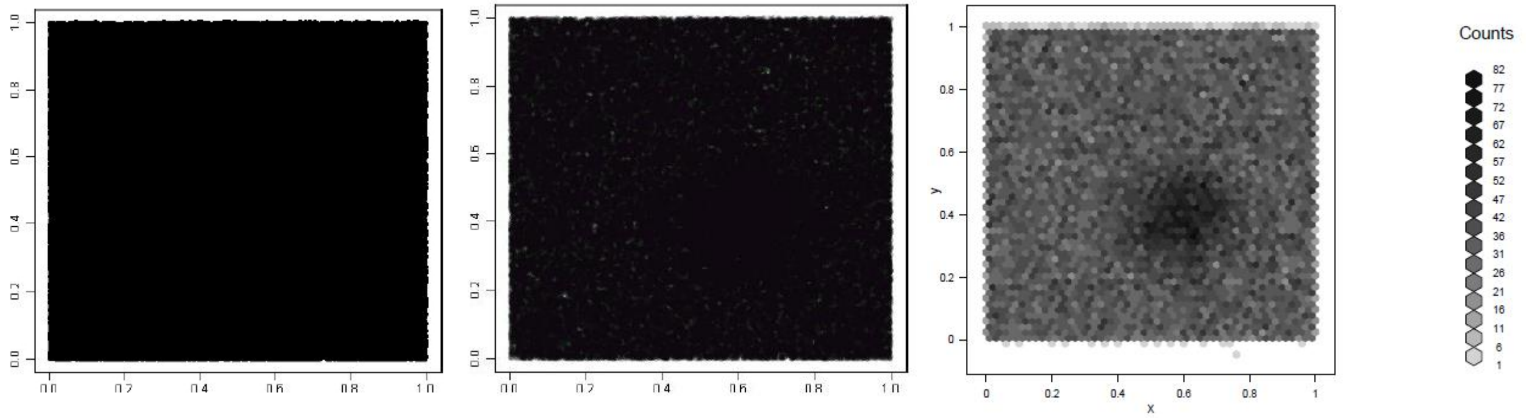
Scatter plots: density

For large data sets, points are plotted over each other and density information is lost.

Left:
1000000 objects

Middle:
Instead of solid points, semitransparent points are plotted

Right:
hexagonal binning. Grey intensity denotes number of points



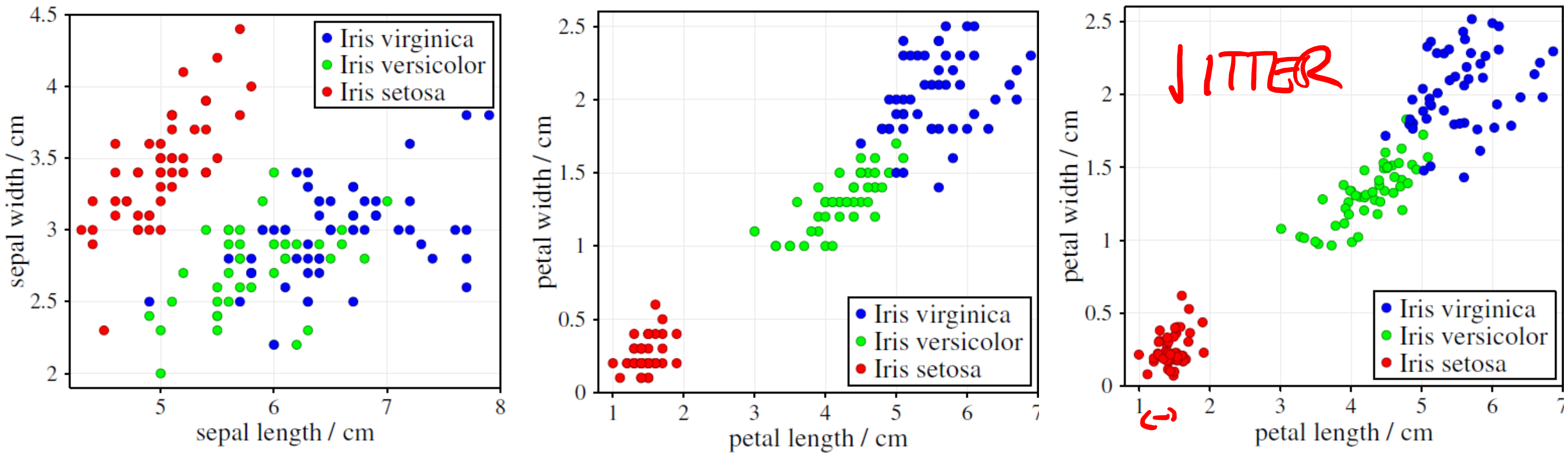
Iris Data Set Example

```
import pandas as pd
import seaborn as sns

iris = pd.read_csv('irisData.csv', names=['sepal_length','sepal_width',
'petal_length','petal_width','species'])

iris.plot.hexbin(x="sepal_length", y="sepal_width", gridsize=20)
sns.jointplot(data=iris, x="sepal_length", y="sepal_width", kind="hex",
color="k",joint_kws=dict(gridsize=20), marginal_kws=dict(bins=15, rug=True))
```

Ref.

Common visualizations

Scatter plots: further elaboration



Scatter plots can be **enriched** with additional information:
color or different symbols incorporate **a third attribute** in the scatter plot.

What differences does this reveal?

Data objects with the same values cannot be distinguished in a scatter plot → **jitter** (adding random noise)

Ref.

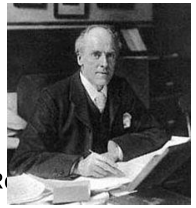
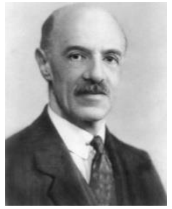
Correlation analysis

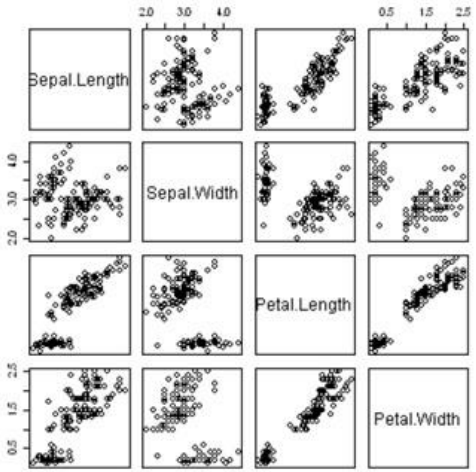
Scatter plots can **“visually” reveal correlations** or dependencies between two attributes.

Statistical measures for correlation are a more formal approach to correlation analysis and can be carried out automatically.

We briefly sketch…

- Pearson’s correlation coefficient
  - >> video for explanation
- Rank correlation coefficients
  - >> video for explanation
  - Spearman’s rho
  - Kendall’s tau


```
import pandas as pd

iris = pd.read_csv('irisData.csv', names=….)

print("Show Pearson's correlation:")
print(iris.corr())
#
print()
print("Show Spearman's rho correlation:")
print(iris.corr('spearman'))
#
print()
print("Show Kendal's tau correlation:")
print(iris.corr('kendall'))
```

python

Show Pearson's correlation:

	sepal_length	sepal_width	petal_length	petal_width
sepal_length	1.000000	-0.109369	0.871754	0.817954
sepal_width	-0.109369	1.000000	-0.420516	-0.356544
petal_length	0.871754	-0.420516	1.000000	0.962757
petal_width	0.817954	-0.356544	0.962757	1.000000

Show Spearman's rho correlation:

	sepal_length	sepal_width	petal_length	petal_width
sepal_length	1.000000	-0.159457	0.881386	0.834421
sepal_width	-0.159457	1.000000	-0.303421	-0.277511
petal_length	0.881386	-0.303421	1.000000	0.936003
petal_width	0.834421	-0.277511	0.936003	1.000000

Show Kendal's tau correlation:

	sepal_length	sepal_width	petal_length	petal_width
sepal_length	1.000000	-0.072112	0.717624	0.654960
sepal_width	-0.072112	1.000000	-0.182391	-0.146988
petal_length	0.717624	-0.182391	1.000000	0.803014
petal_width	0.654960	-0.146988	0.803014	1.000000

Pearson's correlation coefficient

The (sample) Pearson's correlation coefficient is a measure for a linear relationship between two numerical attributes X and Y and is defined as

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{(n-1)s_x s_y}$$

where $\bar{x}$ and $\bar{y}$ are the mean values of the attributes X and Y, respectively. s_x and s_y are the corresponding (sample) standard deviations.

The larger the absolute value of the Pearson correlation coefficient, the stronger the **linear relationship** between the two attributes.

$$-1 \leq r_{xy} \leq 1$$

Pearson's correllation assumes normal distribution (vulnerable to skewed data) and linear relationships.

Applicable to continuous variables.

Ref.

Rank correlation coefficient

Please read on your own

Pearson's correlation coefficient measures linear correlation.

Even for monotone functional, but non-linear relationship Pearson's correlation coefficient will not be -1 or 1. It can even be close to zero despite a monotone functional relationship.

Rank correlation coefficients avoid this by ignoring the exact numerical values of the attributes and *considering only the ordering* of the values.

They intend to measure monotonous correlations between attributes, where the monotonous function does not have to be linear.

Example: Aggregate Single Sales (US)

Pos	Artist and Title	Sales estimate	This year
1	Mark Ronson - Uptown Funk	7,470,000	120,000
2	Pharrell Williams - Happy	7,280,000	40,000
3	Katy Perry - Dark Horse	6,230,000	20,000
4	Taylor Swift - Shake It Off	5,840,000	60,000
5	Meghan Trainor - All About That Bass	5,710,000	20,000
ordinal		continuous	

Rank correlation coefficients

Spearman's rho

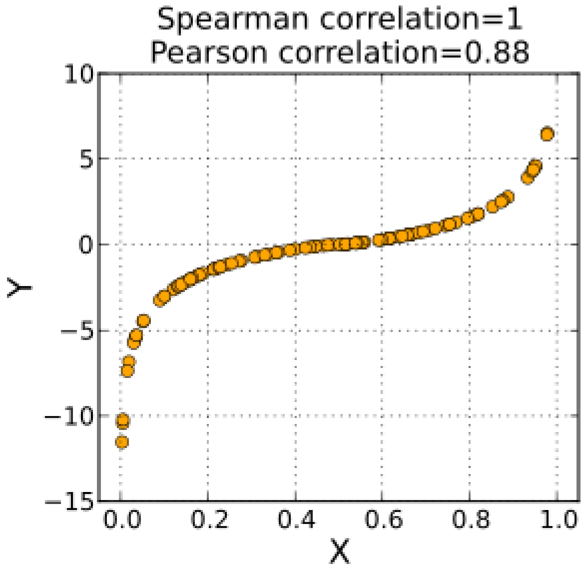
Spearman‘s rank correlation coefficient (**Spearman‘s rho**) is defined as

$$\rho = 1 - 6\frac{\sum_{i=1}^{n}\left(r(x_i)-r(y_i)\right)^2}{n(n^2-1)},$$

where we sum the deviations between $r(x_i)$ – the rank of value x_i when we sort the list $(x_1, \dots, x_n)$ in increasing order – and $r(y_i)$.

When the rankings of the x- and y-values are exactly in the same order, Spearman‘s rho will yield the value 1.

If they are in reverse order, we will obtain the value -1.



Spearman‘s rho makes no assumption on the distribution and is applicable to continuous and discrete (ordinal) variables.

It is sensitive to large deviations.

Ref.

Image: Skbkekas (2009) | Wikimedia

Please read on your own

Rank correlation coefficients

Kendall's tau

Kendall‘s tau rank correlation coefficient (Kendall‘s tau) is defined as

$$\tau_a = \frac{C-D}{\frac{1}{2}n(n-1)}$$

where C and D denote the numbers of concordant (similar rank order) and discordant pairs with similar ranks, respectively.

$$C = \left|\left\{(i,j) | x_i < x_j \text{ and } y_i < y_j\right\}\right|$$

$$D = \left|\left\{(i,j) | x_i < x_j \text{ and } y_i > y_j\right\}\right|$$

Kendall‘s tau makes no assumption on the distribution.
Kendall‘s tau$_a$ is applicable to continuous and discrete (incl. ordinal) variables

Less sensitive to errors and discrepancies in the data as Spearman.

Ref.

Agenda

Low-dimensional relationships

Univariate Analysis

Bivariate Analysis

Higher-dimensional relationships

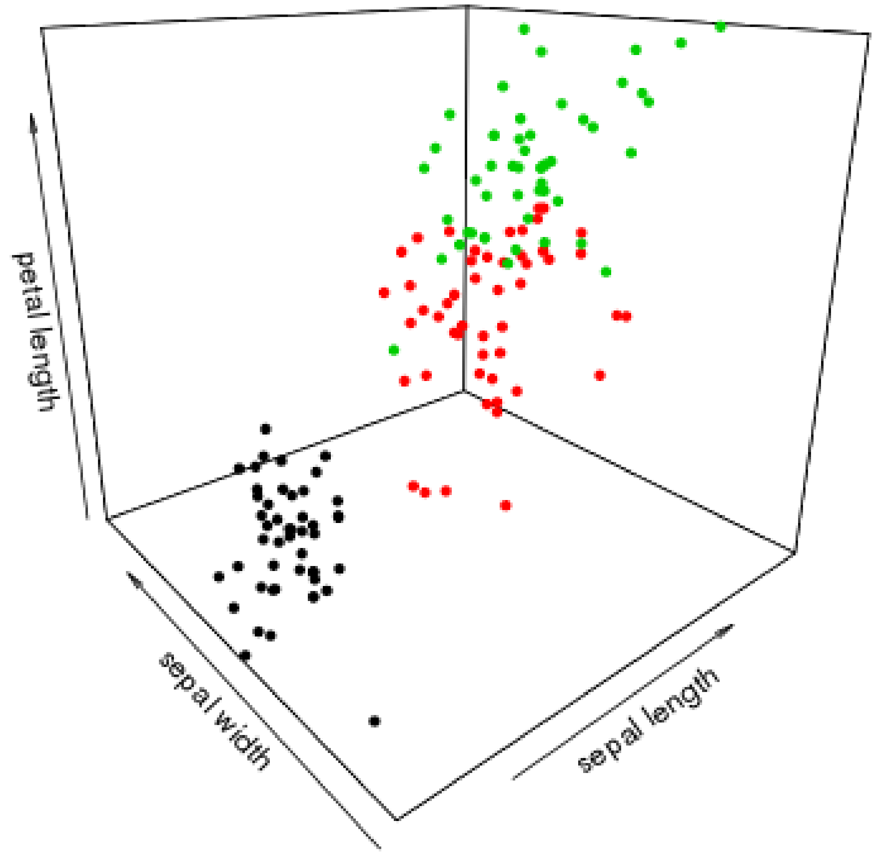
Principal Component Analysis

Parallel Coordinates

Ref.

3D scatter plots

For data sets of moderate size, scatter plots can be extended to **three dimensions**.



Ref. https://www.kaggle.com/andytran/rotating-3d-scatter-plot-for-iris-data

Methods for higher-dimensional data

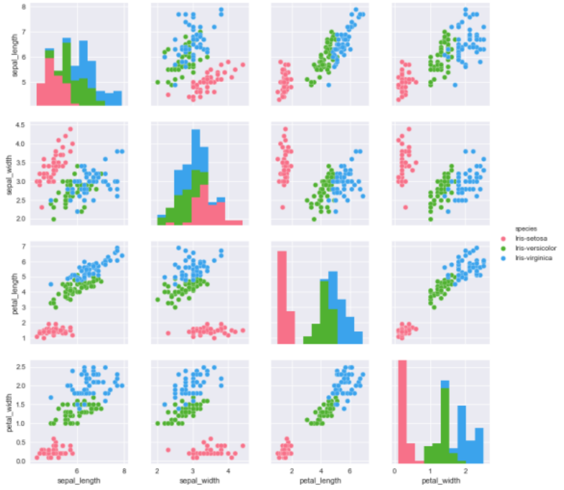
How do we visualize more then 3 dimensions?

A display or **plot** is by definition two-dimensional, so that only two axes (attributes) can be incorporated.

3D techniques can be used to incorporate three axes (attributes).

The number of possible scatter plots grows in a quadratic fashion with the number of attributes. For m attributes, there are $\binom{m}{2} = \frac{m(m-1)}{2}$ possible scatter plots.

- For instance, 50 attributes → 1225 scatter plots.



Ref.

Agenda

Low-dimensional relationships

Univariate Analysis

Bivariate Analysis

Higher-dimensional relationships

Principal Component Analysis

Parallel Coordinates

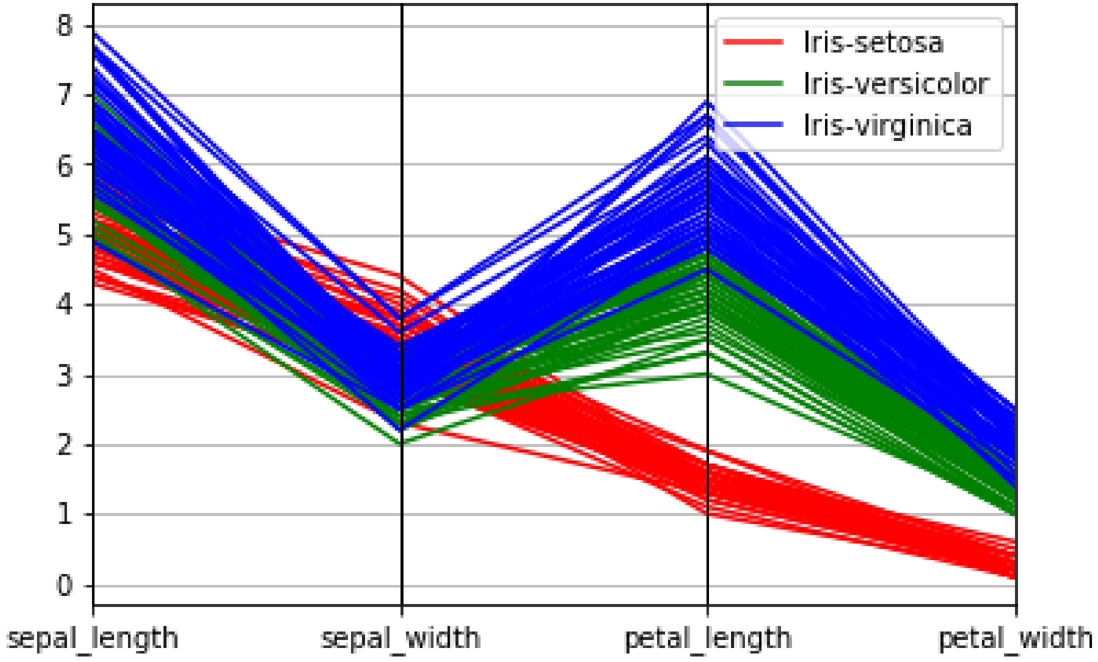
Ref.

Parallel coordinates

Parallel coordinates draw the coordinate axes parallel to each other

There is **no limitation** for the number of axes to be displayed

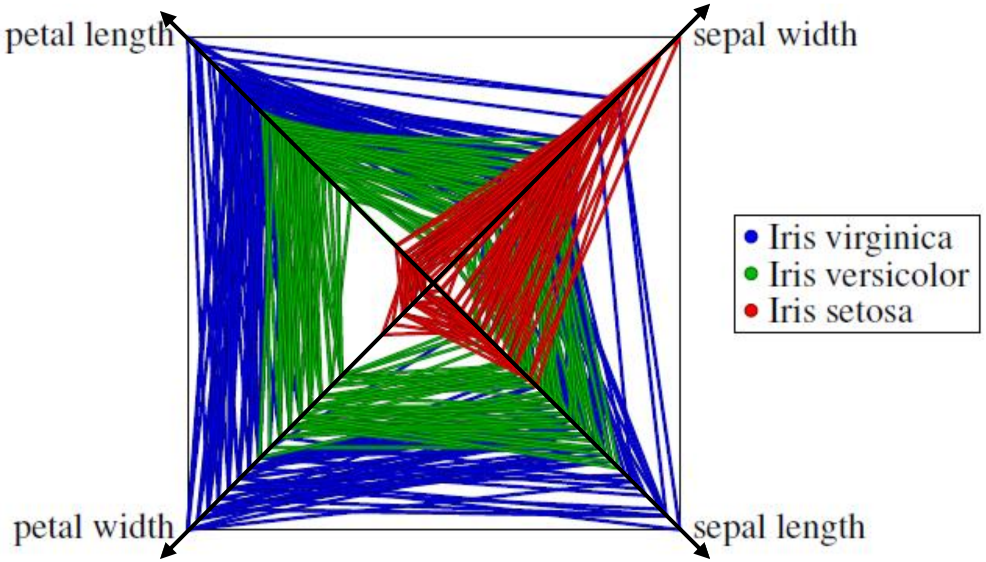
For a data object, a polyline is drawn connecting the values of the data object for the attributes on the corresponding axes



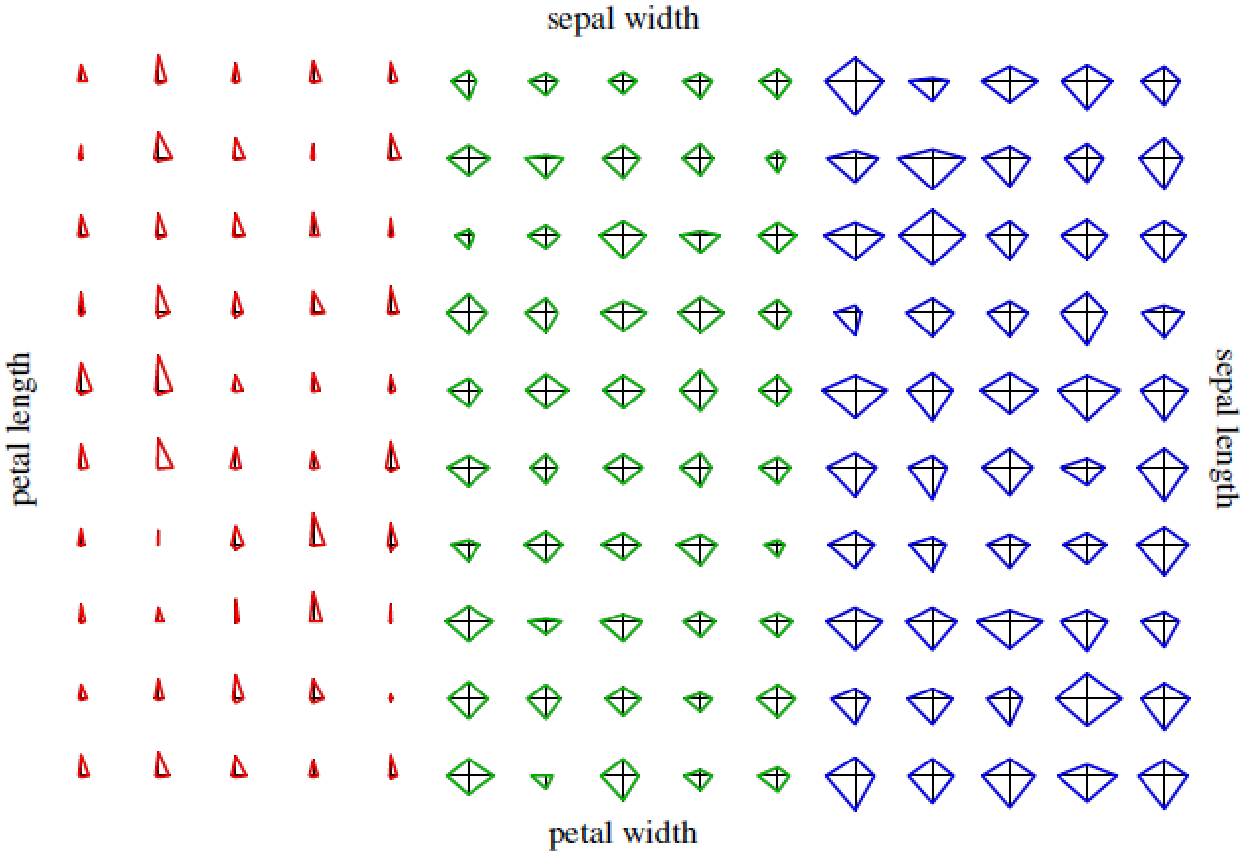

```
import pandas as pd
from pandas.plotting import parallel_coordinates

iris = pd.read_csv('irisData.csv', names=['sepal_length', 'sepal_width', 'petal_length', 'petal_width', 'species'])

parallel_coordinates(iris, 'species', color = ['r', 'g', 'b‘])

# Beispiel um spezifische Datensätze und Attribute auszuwählen
parallel_coordinates(iris[iris.species == "Iris-setosa"], 'species', cols=["sepal_length", "sepal_width", "petal_length", "petal_width"], color = ['r'])
```

Radar plots and star plots

Radar plots are based on a similar idea as parallel coordinates with the difference that the coordinate axes are drawn as parallel lines, but in a star-like fashion intersecting in one point.



Star plots are the same as radar plots where each data object is drawn separately.



Ref.

Agenda

Low-dimensional relationships

Univariate Analysis

Bivariate Analysis

Higher-dimensional relationships

Principal Component Analysis (PCA)

Parallel Coordinates

Ref.

Methods for higher-dimensional data

General approach for incorporating all attributes in a plot:

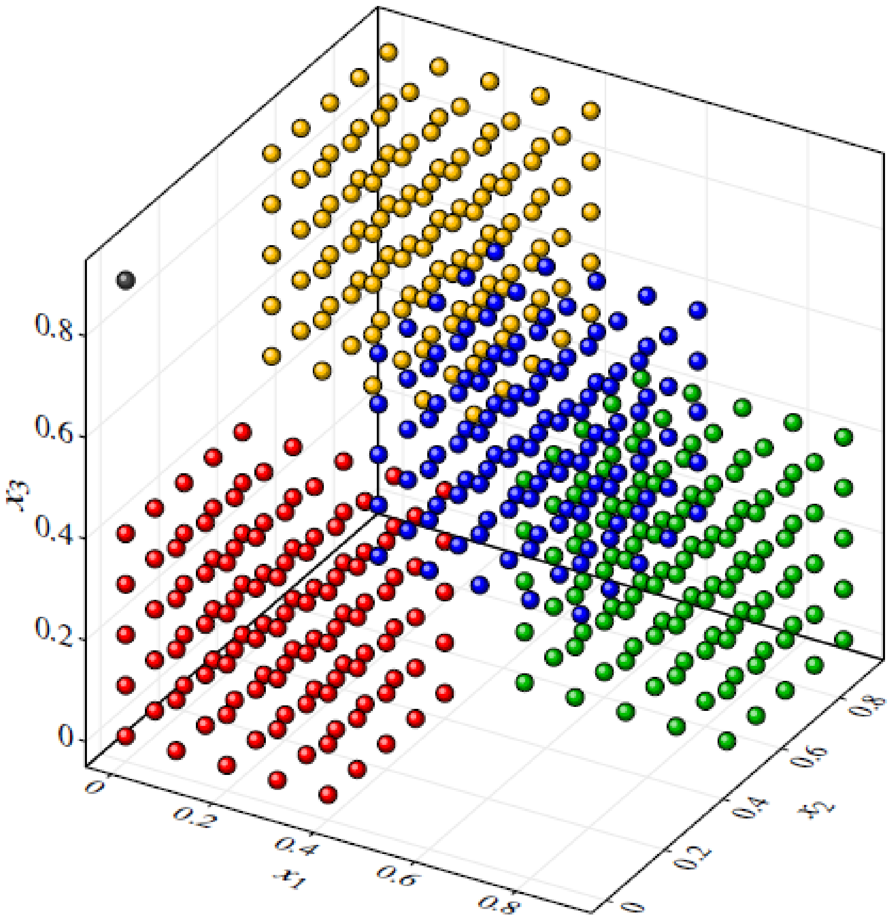
There is no unique measure for structure preservation.

Try to preserve as much of the “structure” of the high-dimensional data set when **representing (plotting) the data in two (or three) dimensions**

Define a measure that evaluates lower-dimensional representations (plots) of the data in terms of **how well a representation preserves the original “structure”** of the high-dimensional data set.

Find the representation (plot) that gives the best value for the defined measure.

PCA – Chessboard example (1/2)



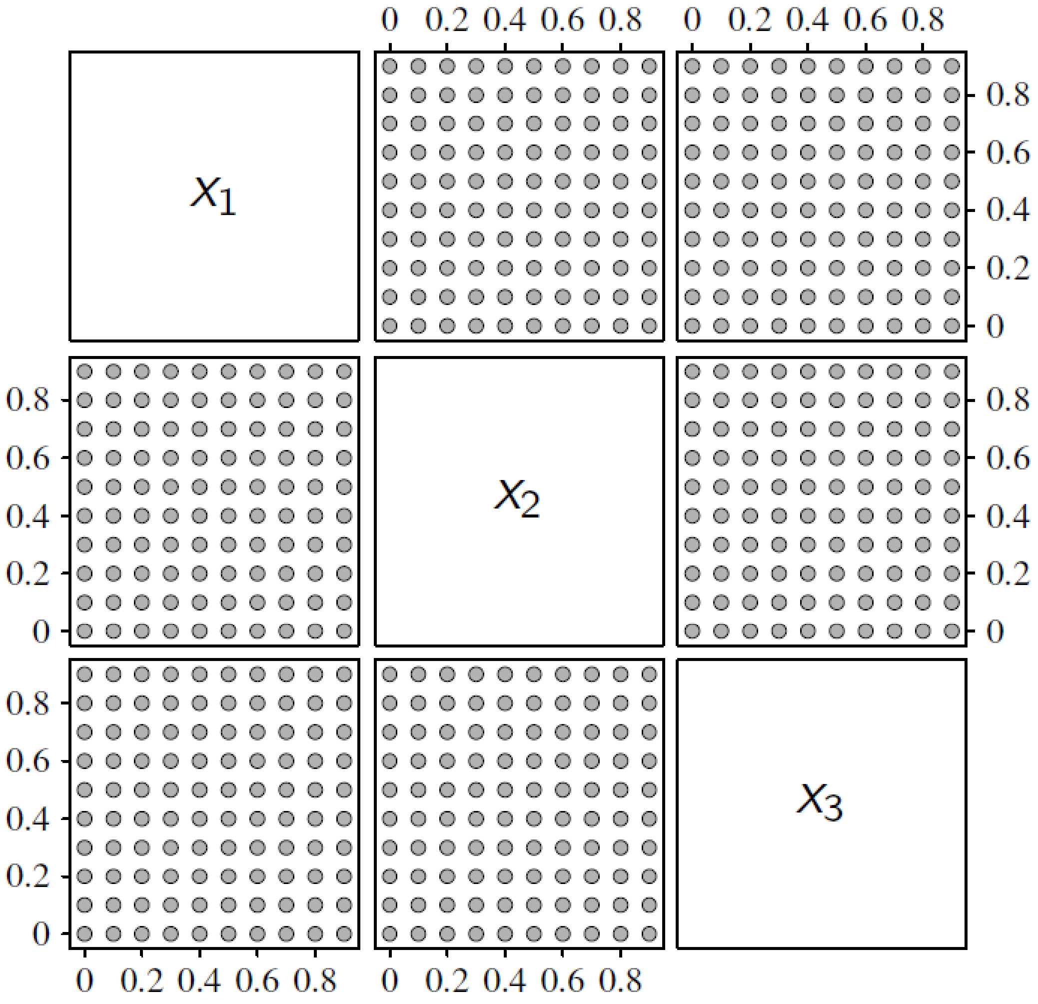
How to preserve “structure” in 2D ?

Ref. Berthold et al. (2010)

PCA – Chessboard example (2/2)

Scatter plots

Is data uniformly distributed over the grid?



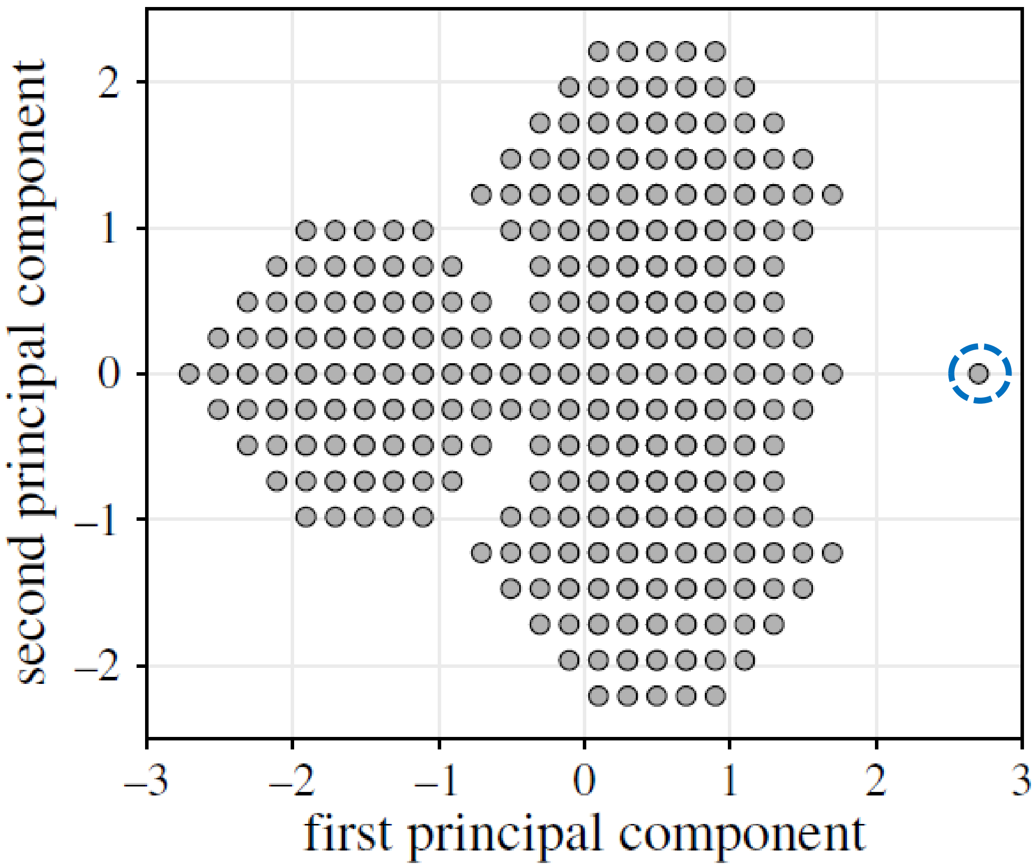
Ref. Berthold et al. (2010)

Projection to the first two **principal components**

Data is not uniformly distributed.

There is a pattern in the data set.

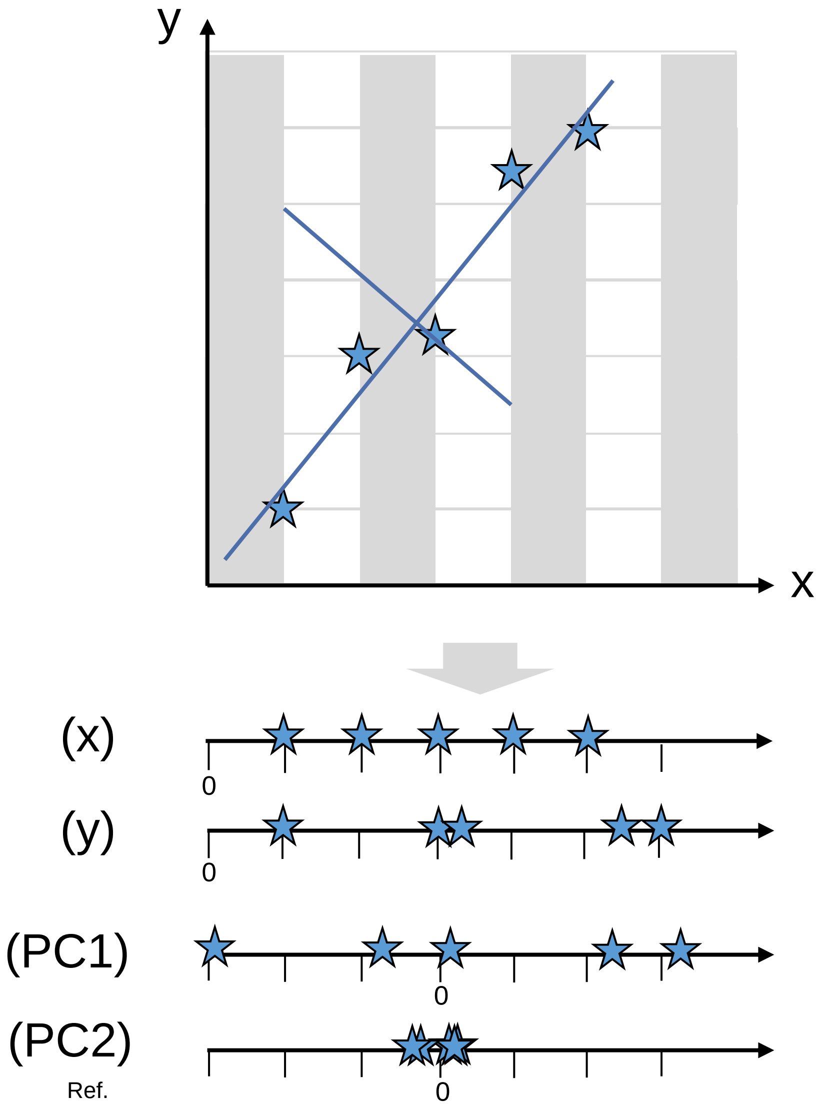
Data can be recreated from PCA.



Principal Component Analysis (PCA)

Structure preservation through variance in data set

From $\mathbb{R}^2$ to $\mathbb{R}^1$



PCA compresses a large data set to capture the *essence of the original data* through linear transformation

PCA constructs **a projection** from the high-dimensional space to a lower-dimensional space (plane or hyperplane)
using only the most relevant dimensions

PCA uses the **variance in the data** set as the structure preservation criterion.

Assumption: Large variances describe interesting dynamics, smaller noise.

PCA preserves as much of the original variance of the data when projected to a lower-dimensional space

(Sample) variance for a numerical attribute:

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2 = \frac{n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2}{n(n-1)}$$

Ref.
e.g., Kristensen & Terje (2016, p. 81 ff.)

Principal Component Analysis

Procedure: Objective

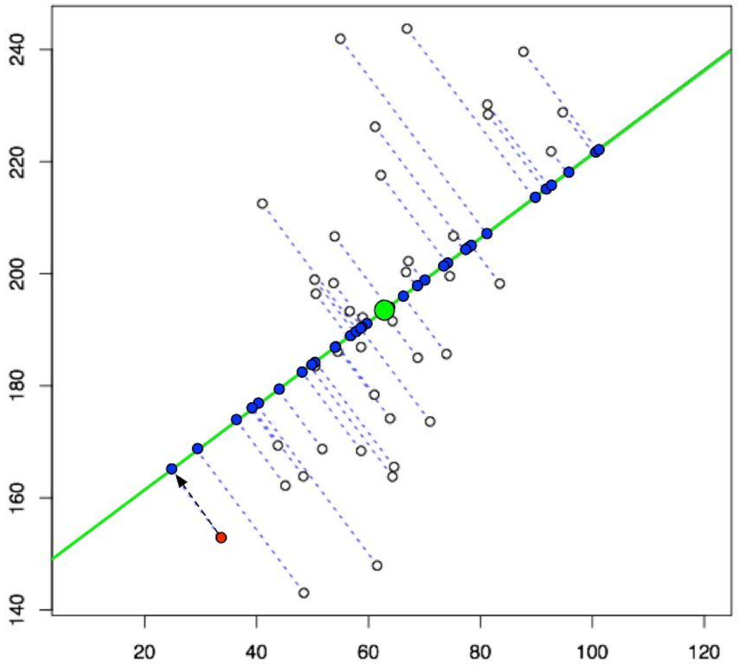
The data points are first **centered around the origin** by subtracting the mean values

Objective:

find a projection in the form of a linear mapping given by $\boldsymbol{y} = \boldsymbol{M}(\boldsymbol{x} - \overline{\boldsymbol{x}})$, where M is a $q \times m$ matrix such that the **variance** of the projected data $y_i = M(x_i - \bar{x})$ is **maximized**

($2 \times m$ for projections to a plane)

Projecting 2 dimensions on 1



PCA uses the covariance matrix which holds information on spread (variance) and orientation (covariance)

$$\Sigma = \begin{bmatrix} \sigma(x,x) & \sigma(x,y) \\ \sigma(y,x) & \sigma(y,y) \end{bmatrix}$$

See excursus for in-depth information

Ref. e.g., Kristensen & Terje (2016, p. 81 ff.)

Image: Pachter (2014)

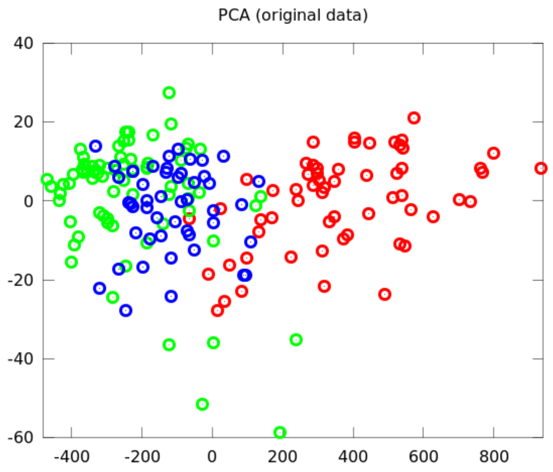
Principal Component Analysis

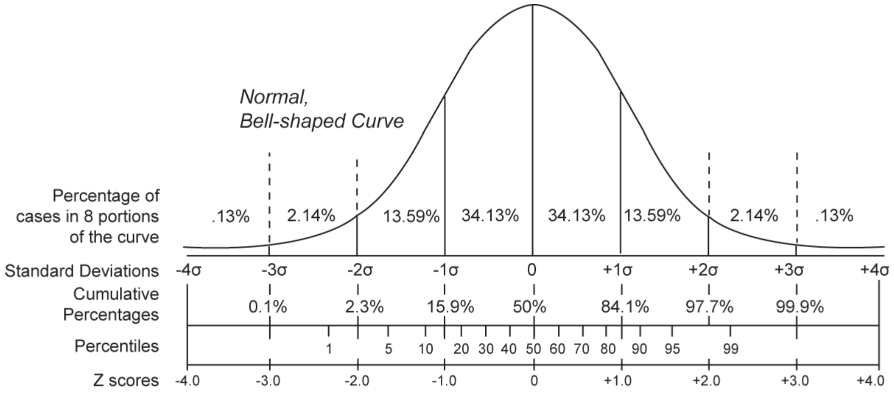
Procedure: Problem

Problem:
Without restriction for the matrix M, the entries in M can be chosen arbitrary large so that the data are not only projected, but also scaled, leading to an arbitrary large variance of the projected data.

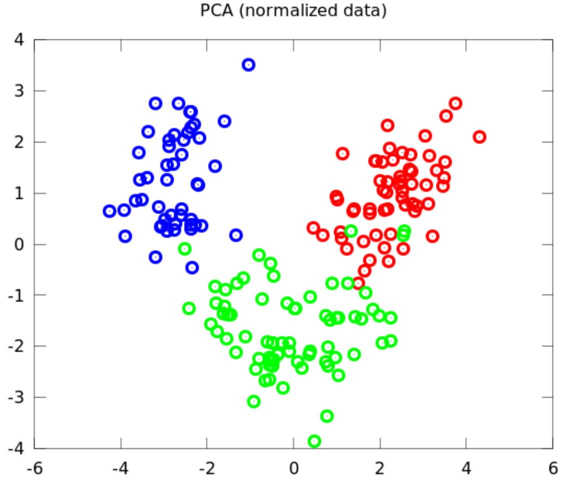
We introduce **constraints** such that the matrix M is only a projection:

The row v_i of the matrix $M = (v_1, \ldots, v_q)$ must be normalized, i.e., $||v_i|| = 1$.





Usually, the data should be **zero-score standardized** ($x \rightarrow \frac{x-\hat{\mu}_x}{\hat{\sigma}_x}$) to ensure that all attributes contribute equally to the overall variance
(with $\hat{\mu}_x$ being the mean value and and $\hat{\sigma}_x$ the sample standard deviation of attribute X, z-score: numeric distance of x in standard deviations from mean)



Ref. e.g., Kristensen & Terje (2016, p. 81 ff.)

Images: Wagner (2011)

Principal Component Analysis

Choosing principal components

Solution of the constraint optimization problem:

The projection matrix M is given by $M = (v_1, \dots, v_q)$,

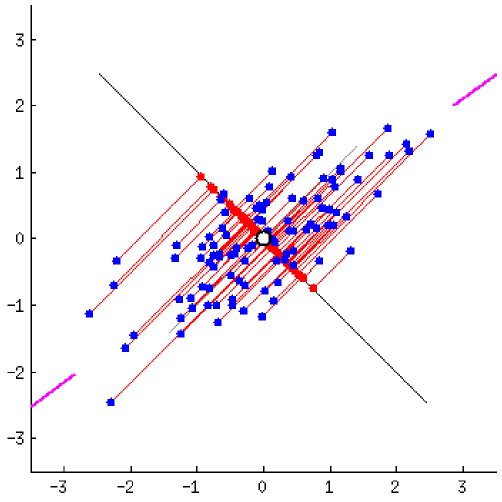
where the **principal components** $v_1, \dots, v_q$

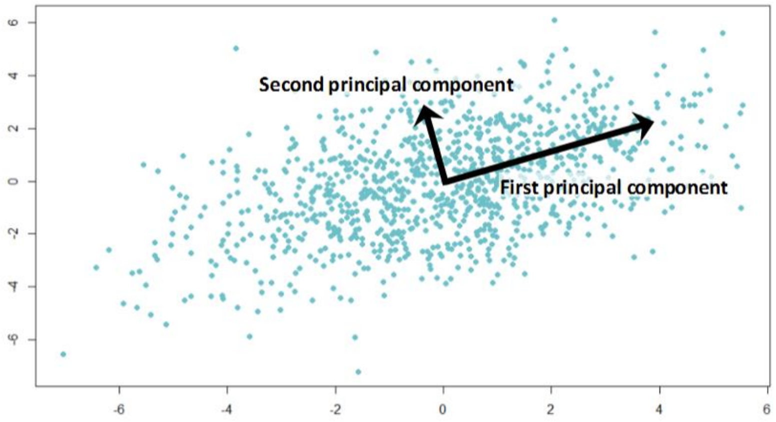
are the *normalized eigenvectors of the covariance matrix* of the attributes in the data set

$$\sigma_{jk} = \frac{1}{n-1}\sum_{i=1}^{n}(x_{ij} - \bar{x}_j)(x_{ik} - \bar{x}_k)^T$$

for the q **largest eigenvalues** $\lambda_1 \geq \cdots \geq \lambda_q$.

λ is called an eigenvalue of a matrix A, if there is a non-zero vector $\boldsymbol{v}$ such that $A\boldsymbol{v} = \lambda\boldsymbol{v}$ holds. The vector v is called eigenvector (direction of the data) to the eigenvalue λ (magnitude of its spread).





Ref. e.g., Kristensen & Terje (2016, p. 81 ff.), An illustrative explanation for Eigenvalue/Eigenvectors (in German)

Image. Gabasova (2014)

Principal Component Analysis

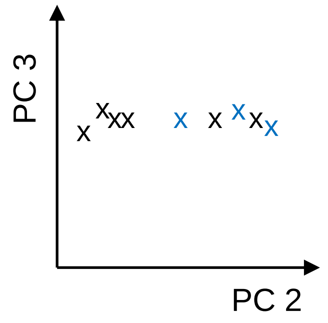
Dimension reduction

Let $\lambda_1 \geq \cdots \geq \lambda_m$ be the eigenvalues of the covariance matrix.

When we project the data to the first q principal components $v_1, \ldots, v_q$ corresponding to the eigenvalues $\lambda_{1,} \ldots, \lambda_q$, this projection will preserve a fraction of of the variance of the original data.

$$\frac{\lambda_1 + \cdots + \lambda_q}{\lambda_1 + \cdots + \lambda_m}$$

Omid principal components which explain little variance in the data, like…



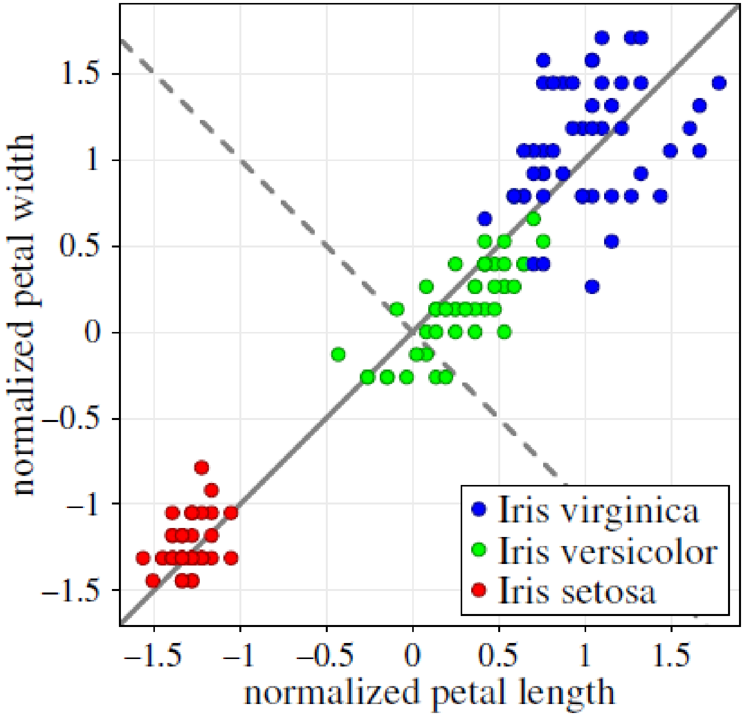
Iris data set:

	PC1	PC2	PC3	PC4
Proportion of variance	0.73	0.229	0.0367	0.00518
Cum. proportion	0.73	0.958	0.9948	1.00000

Ref.

PCA – Iris data set example (1/2)

PCA applied to the **Iris data set** restricted to the (normalized) petal length and width

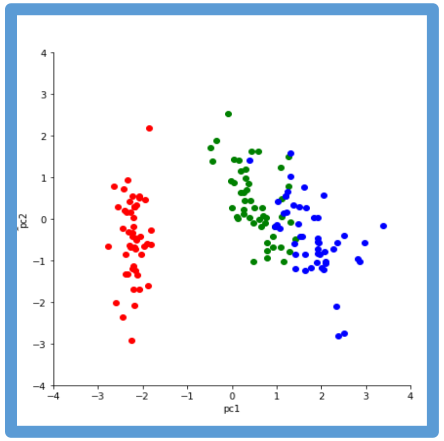


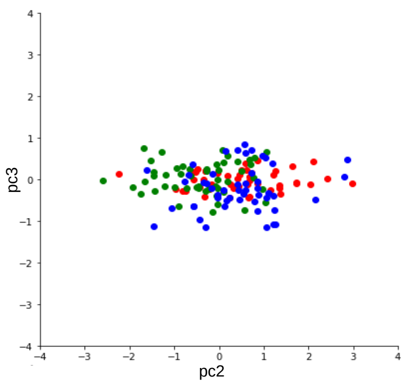
The principal components are always *orthogonal*

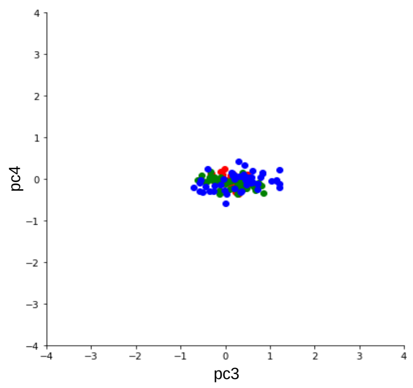
Ref.

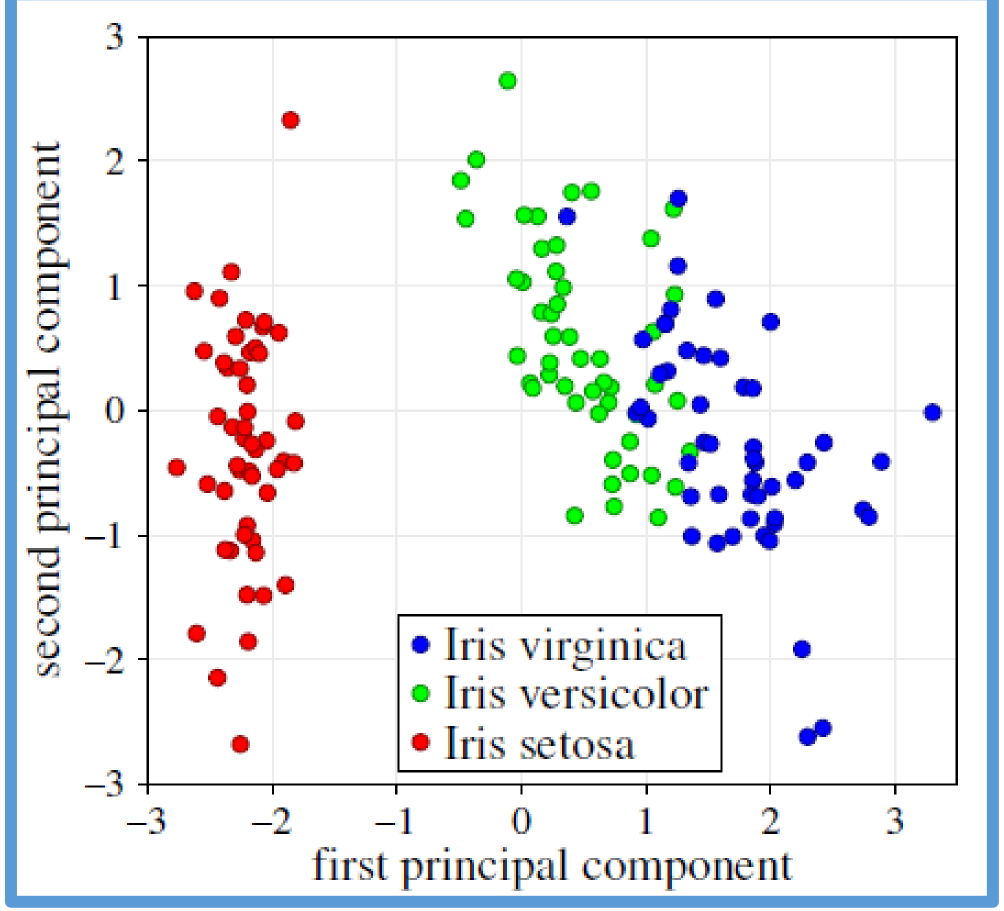
PCA – Iris data set example (2/2)

Projection to the first two principal components of PCA taking all four numerical attributes into account









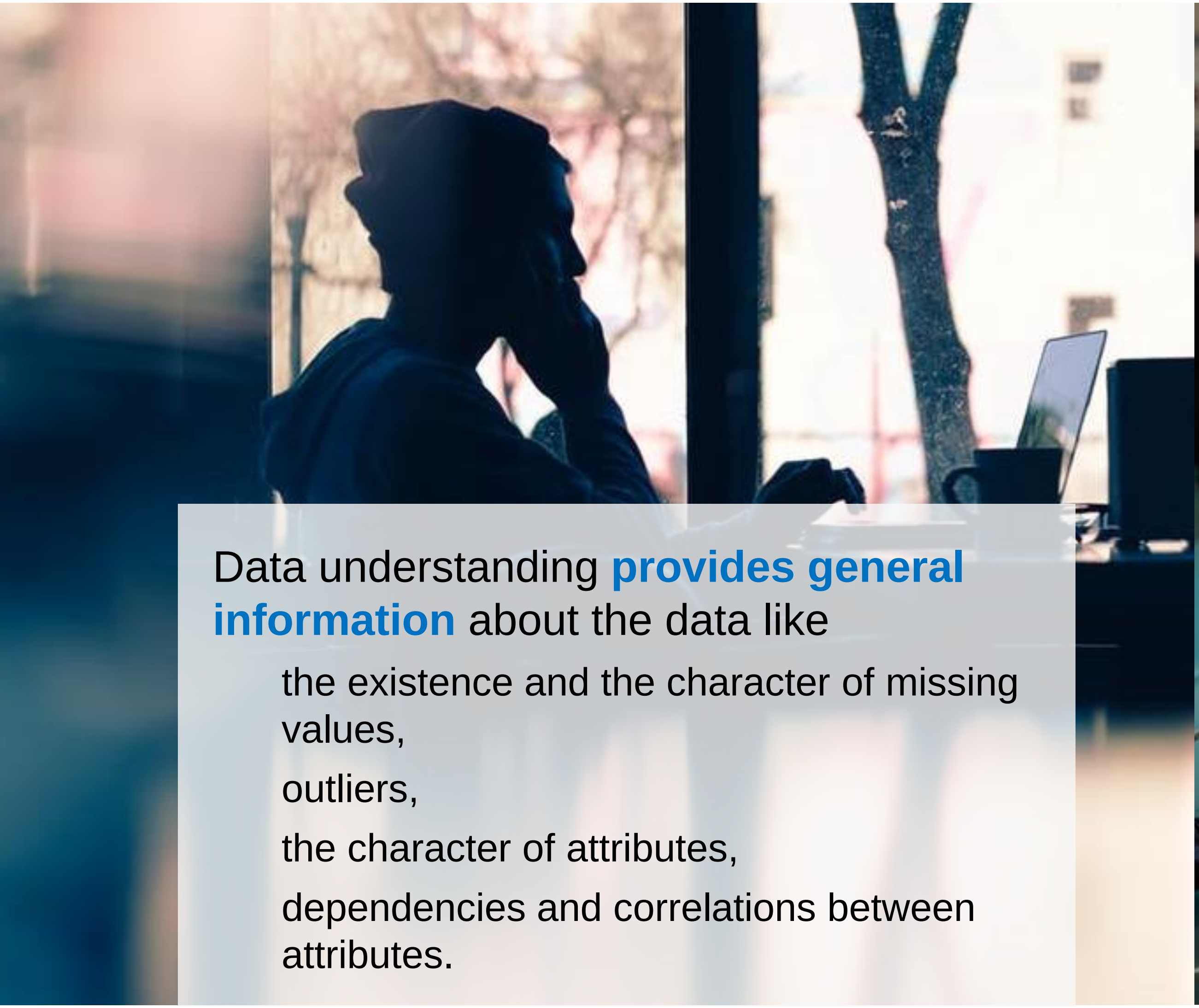
Original data is **reconstructable** from the principal components

Example in Python

python™

Ref.

Next Lesson
Data understanding vs. Data preparation



Data understanding **provides general information** about the data like

- the existence and the character of missing values,
- outliers,
- the character of attributes,
- dependencies and correlations between attributes.

Data preparation **uses this information** to

- select attributes and data records,
- reduce the dimension of the data set,
- treat missing values and outliers,
- integrate, unify and transform data,
- improve data quality.

Ref.

Fragen?

- ✓ Data visualization, correlation analysis (Data understanding II)

- ✓ Low-dimensional relationships
  - ✓ Univariate Analysis
  - ✓ Bivariate Analysis
- ✓ Higher-dimensional relationships
  - ✓ Principal Component Analysis
  - ✓ Parallel Coordinates

Todos for next Week

- Think about who you want to form a project group with (4 people per group)

Ref.

Recommended reading

Berthold et al.   Chapter 4

Han, J., Kamber, M., Pei, J.: Data Mining: Concepts and Techniques. Morgan Kaufmann, 2011

Ref.