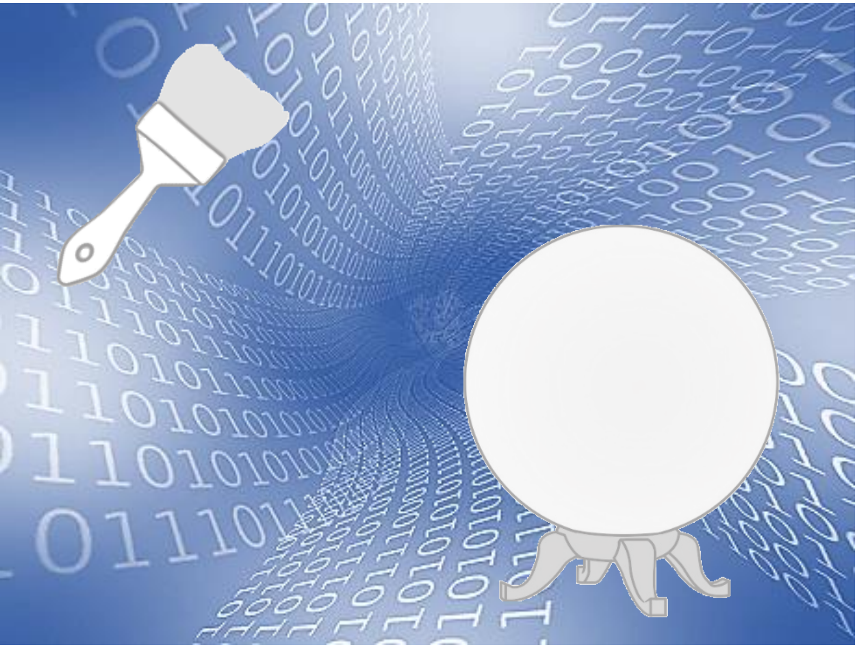
Prof. Dr. Bastian Amberg
School of Business & Economics
Department of Information Systems



# Business Intelligence

## 09 Predictive Modeling I

**Prof. Dr. Bastian Amberg**
**(summer term 2024)**
7.6.2024

# Schedule

| | | 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* | | Chap. 2 | |
| | W7 | 29.5. | Data Visualization II | 31.5. | *1.) Co-Create your exam*<br>*2.) Earn bonus points for the exam* | | | |
| | W8 | 5.6. | Data Preparation | 7.6. | Predictive Modeling I (10:00 -12:00) | BI-Project | Start | |
| | 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 | | \| | Case Study |
| | W11 | 26.6. | How to avoid overfitting | 28.6. | What is a good Model? | | \| | |
| Deep-ening | W12 | 3.7. | Project status update<br>Evidence and Probabilities | 5.7. | Similarity (and Clusters)<br>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 | |
| | | | | | *Klausur 1.Termin, 31.7.'24*<br>*Klausur 2.Termin, 2.10.'24* | Projektbericht | | |

Ref.

# Agenda

Freie Universität Berlin

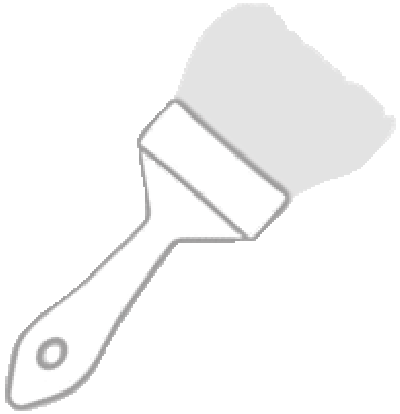
→ MCAR ✓

→ MAR ✓

Nonignorable ✓

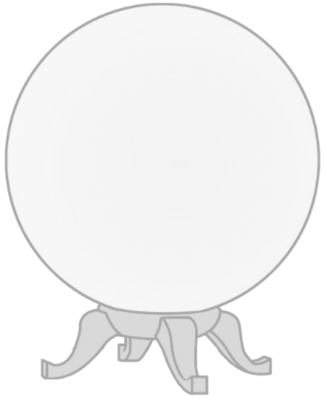
Data Preparation

Predictive Modeling I



- ✓ Data selection
- ✓ Data cleansing
- • **Data transformation**
- ✓ Data integration

Introductory example
Attribute Selection,
Decision Trees



Ref.

# Data transformation

## Categorical -> Numerical attributes



Some models can only handle numerical attributes, other models only categorical attributes.

In such cases, categorical attributes must be transformed into numerical ones or vice versa.

"Winfo" → 1
→ 0

### Categorical attribute -> Numerical attribute:

A binary attribute can be turned into a numerical attribute with the values 0 and 1 (aka dummy variable)

A categorical attribute with more than two values, say $a_1, \dots, a_k$, **should not be turned into a single numerical attribute** with the values $1, \dots, k$, unless the attribute is an ordinal attribute. It should be turned into $k$ attributes $A_1, \dots, A_k$ with values 0 and 1 (dummies). $a_1$ is represented by $A_i = 1$ and $A_j = 0$ for $i \neq j$.

"Winfo" ~~1.~~
"Facts" ~~2.~~
"M&M" ~~3.~~

| Winfo (ja / nein) | Facts | M&M |
|---|---|---|
| 0 oder 1 | 0 oder 1 | 0 oder 1 |

Ref.

# Data transformation

Größe

Discretization: Numerical -> Categorical attributes

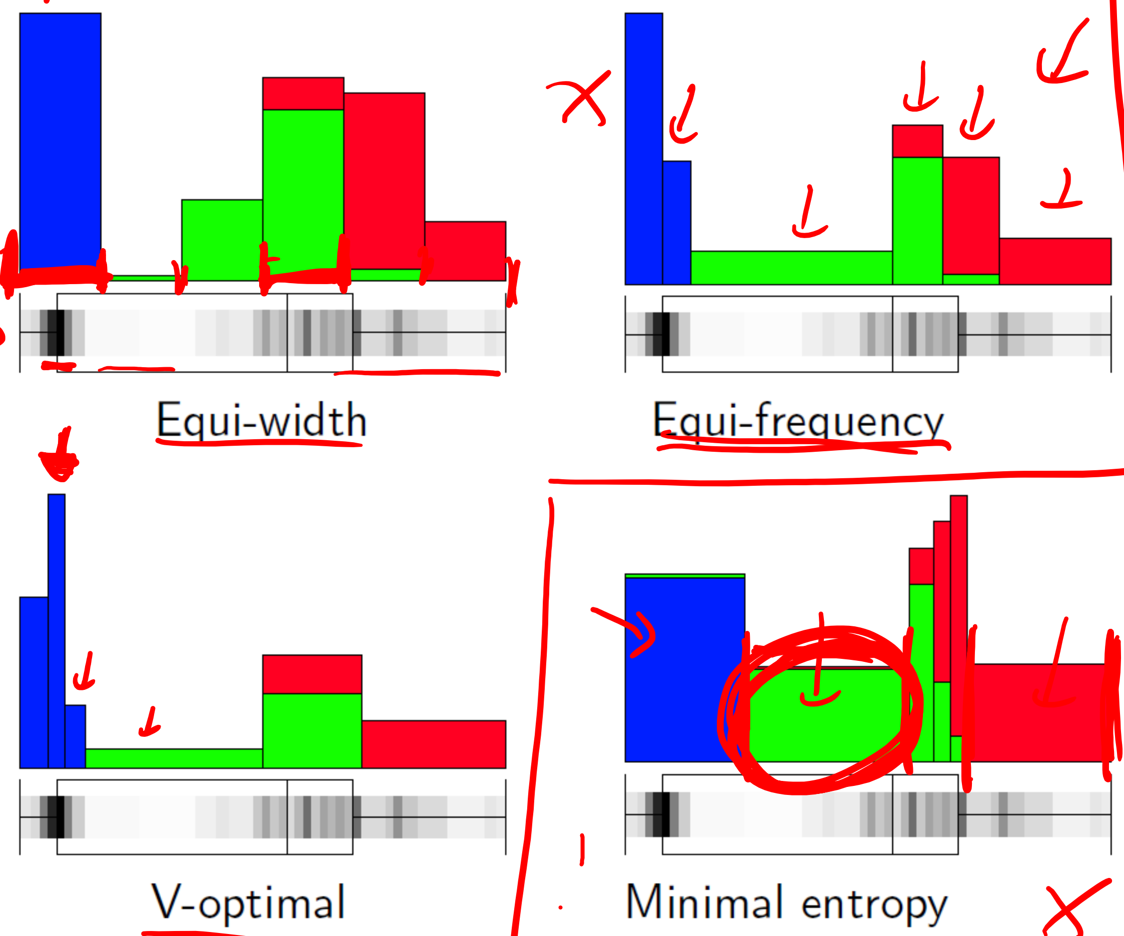
Sumpf

Kelchblattlänge

6x

**Discretization techniques** refer to splitting a numerical range into a number of finite bins.

- **Equi-width discretization.** Splits the range into intervals (bins) of the *same width*.
- **Equi-frequency discretization.** Splits the range into intervals such that each interval (bin) contains (roughly) the *same number of records*.
- **V-optimal discretization.** Minimizes $\sum_i n_i V_i$ where $n_i$ is the *number of data objects* in the $i$th interval and $V_i$ is the sample *variance* of the data in this interval.
- **Minimal entropy discretization.** Minimizes the *entropy*. (Only applicable in the case of classification problems, we'll dive deeper into this with decision trees)




Sumpf-
Borsten-
Verschiedenfarbig-

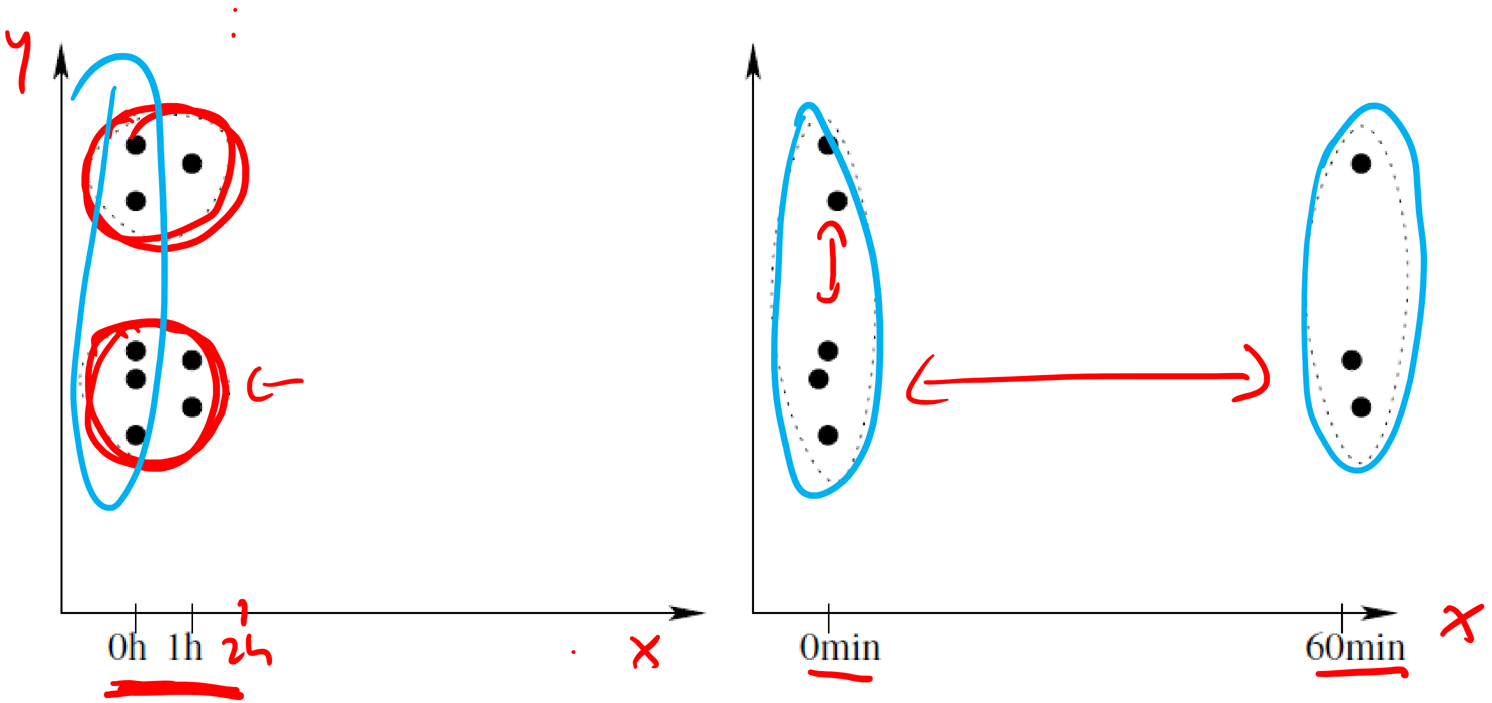
Ref.

# Normalization | Standardization (1/2)

[0...1]

For some data analysis techniques (PCA, MDS, cluster analysis) the influence of an attribute depends on the **scale** or measurement unit.

To guarantee impartiality, some kind of standardization or normalization should be applied.




Ref.

# Normalization | Standardization (2/2)

## Min-max normalization:

For a numerical attribute $X$ with $min_x$ and $max_x$ being the minimum and maximum value in the sample, the min-max normalization is defined as

$$n: domX \to [0,1], \qquad x \to \boxed{\frac{x - min_X}{max_X - min_X}}$$

## Z-score standardization:

For a numerical attribute $X$ with sample mean $\hat{\mu}_X$ and empirical standard deviation $\hat{\sigma}_X$, the z-score standardization is defined as

$$s: domX \to \mathbb{R}, \qquad x \to \frac{x - \overline{\hat{\mu}_X}}{\hat{\sigma}_X}$$

## Robust z-score standardization:

The sample mean and empirical standard deviation are easily affected by outliers. A more robust alternative is (see also boxplots):

$$s: domX \to \mathbb{R}, \qquad x \to \frac{x - \tilde{x}}{IQR_X}$$

## Decimal scaling:

For a numerical attribute $X$ and the smallest integer value $s$ that is larger than $log_{10}(max_X)$, the decimal scaling is defined as

$$d: domX \to [0,1], \qquad \boxed{x \to \frac{x}{10^s}}$$

Ref.

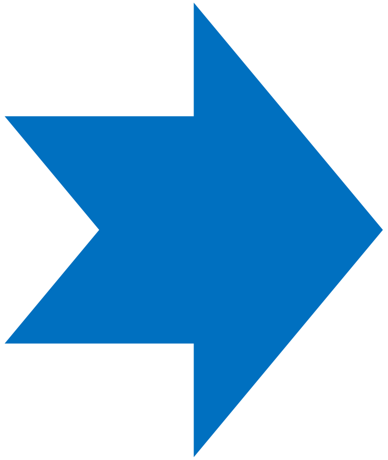
# Agenda

## Data Preparation

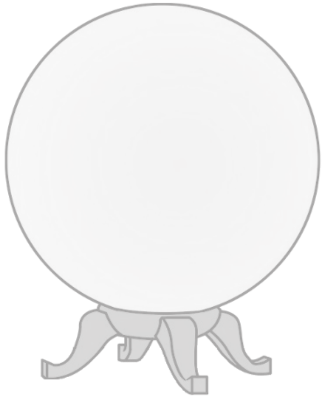


- ✓ Data selection
- ✓ Data cleansing
- ✓ Data transformation
- ✓ Data integration

## Predictive Modeling I



Introductory example
Attribute Selection,
Decision Trees



Ref.

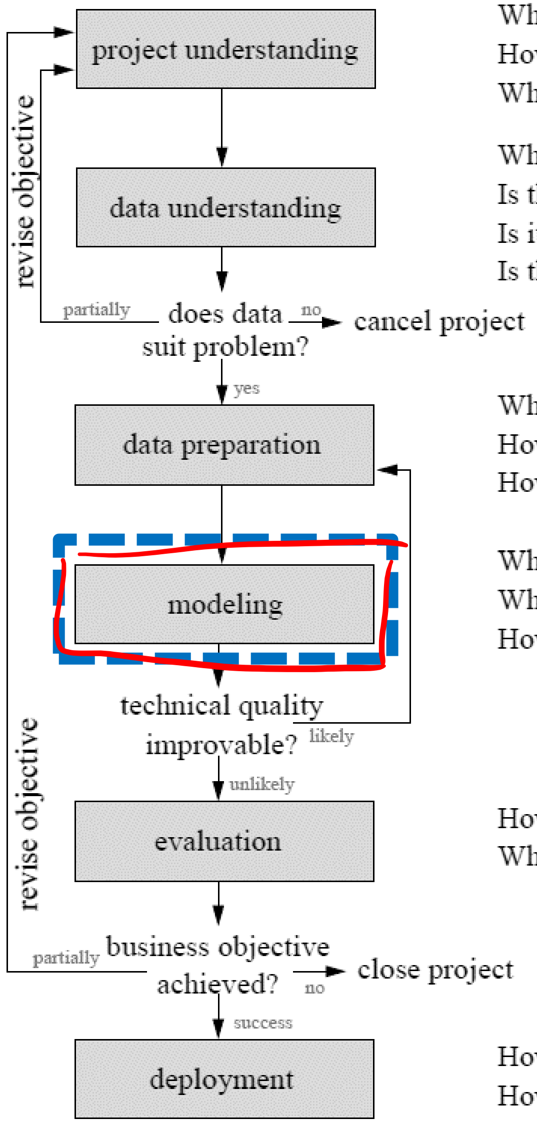
# CRISP-DM

**C**ross
**I**ndustry
**S**tandard
**P**rocess for
**D**ata
**M**ining

Iteration as a rule

Process of data exploration

Implementation of the KDD Process




What exactly is the problem, the expected benefit?
How would a solution look like?
What is known about the domain?

What data do we have available?
Is the data relevant to the problem?
Is it valid? Does it reflect our expectations?
Is the data quality, quantity, recency sufficient?

Which data should we concentrate on?
How is the data best transformed for modeling?
How may we increase the data quality?

What kind of model architecture suits the problem best?
What is the best technique/method to get the model?
How good does the model perform technically?

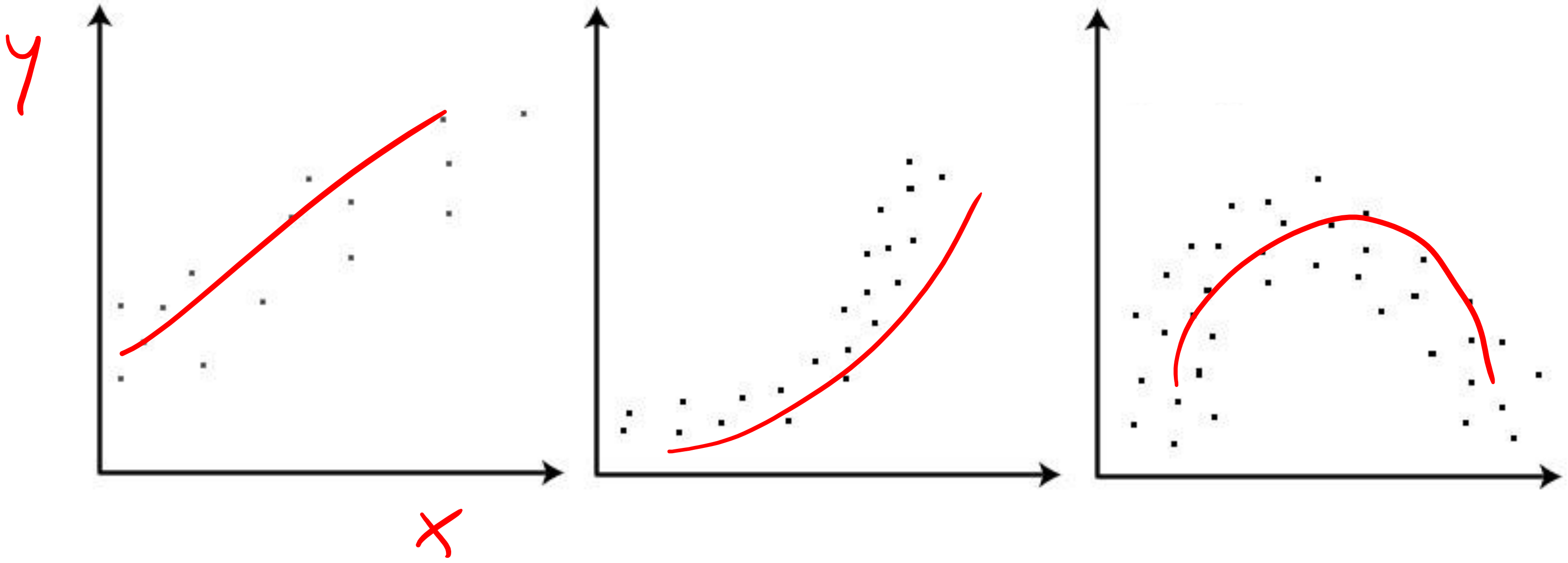
How good is the model in terms of project requirements?
What have we learned from the project?

How is the model best deployed?
How do we know that the model is still valid?

Ref. Wirth / Hipp (2000), Azevedo (2008)

# Let‘s revisit data understanding

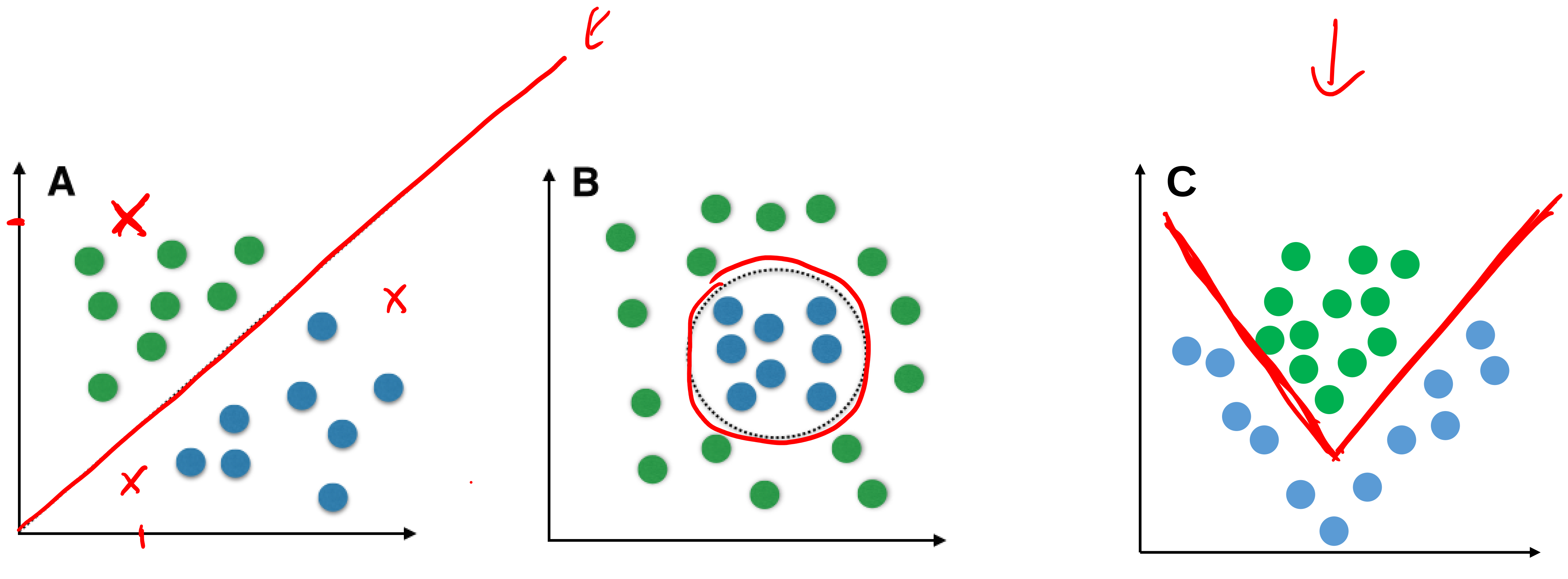
Types of relationships




Ref.

# Let‘s revisit data understanding

On our way to classification problems




Decision Tree
SVM

Ref. Data from https://www.kaggle.com/deepmatrix/imdb-5000-movie-dataset (manipulated)

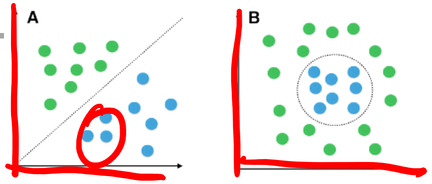
# Introduction

A B C D
\+ −

AB
\++ => yes
−−
−+ => no
\+−

Fundamental concept of DM: **predictive modeling**

Supervised segmentation: how can we segment the population with respect to something that we would like to predict or estimate e.g.




*„Which customers are likely to leave the company when their contracts expire?"*

*„Which potential customers are likely not to pay off their account balances?"*

Technique: find or **select important, informative variables / attributes** of the entities with respect to a target

Is there one or more other variables that reduces our uncertainty about the value of the target?

Select **informative subsets** in large databases

Ref.

# Agenda

Next week
(including the Python-exercise on Friday)

(1) Models and Induction

(2) Attribute Selection

(3a) Decision Trees Algorithmic View

(3b) Probability Estimation

(3c) Decision Tree Examples

Ref.

# Models and Supervised Learning

OR

A model is a simplified representation of reality created to serve a purpose

A predictive model is a formula for **estimating the unknown value of interest**: the target

- Classification/class-probability estimation and regression models

**Prediction = estimate an unknown value**

- Credit scoring, spam filtering, fraud detection

Descriptive modeling: gain insight into the underlying phenomenon or process

→ Clustering

Unsupervised

**Supervised learning**

- Model creation where the model describes a relationship between a set of selected variables (attributes/features) and a **predefined variable** (target)
- The model estimates the value of the target variable as a function of the features

Ref.

# Supervised Learning and Induction

## Supervised learning

Model creation where the model describes a relationship between a set of selected variables (attributes/features) and a **predefined variable** (target)

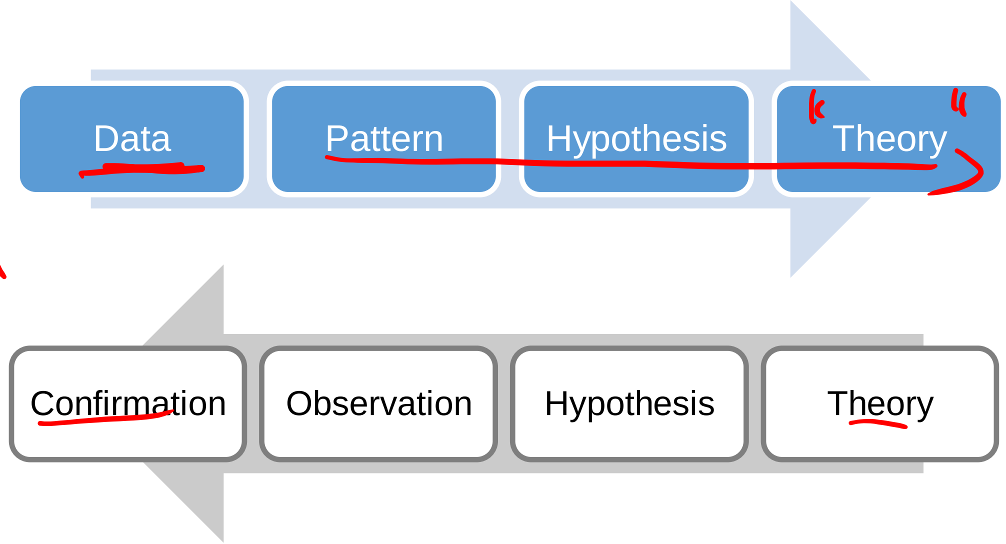
The model estimates the value of the target variable as a function of the features

## Induction

Creation of models from data

Refers to generalizing from specific case to general rules

How can we select one or more attributes / features / variables that will best divide the sample w.r.t. our target variable of interest?




Aufbau

Unlike deduction:

Einsatz

Ref.

# Now: From Data to Decision Trees

? ? ?

*Target attribute*

*Attributes*

| Name | Balance | Age | Employed | Write-off |
|---|---|---|---|---|
| Mike | $200,000 | 42 | no | yes |
| Mary | $35,000 | 33 | yes | no |
| Claudio | $115,000 | 40 | no | no |
| Robert | $29,000 | 23 | yes | yes |
| Dora | $72,000 | 31 | no | no |

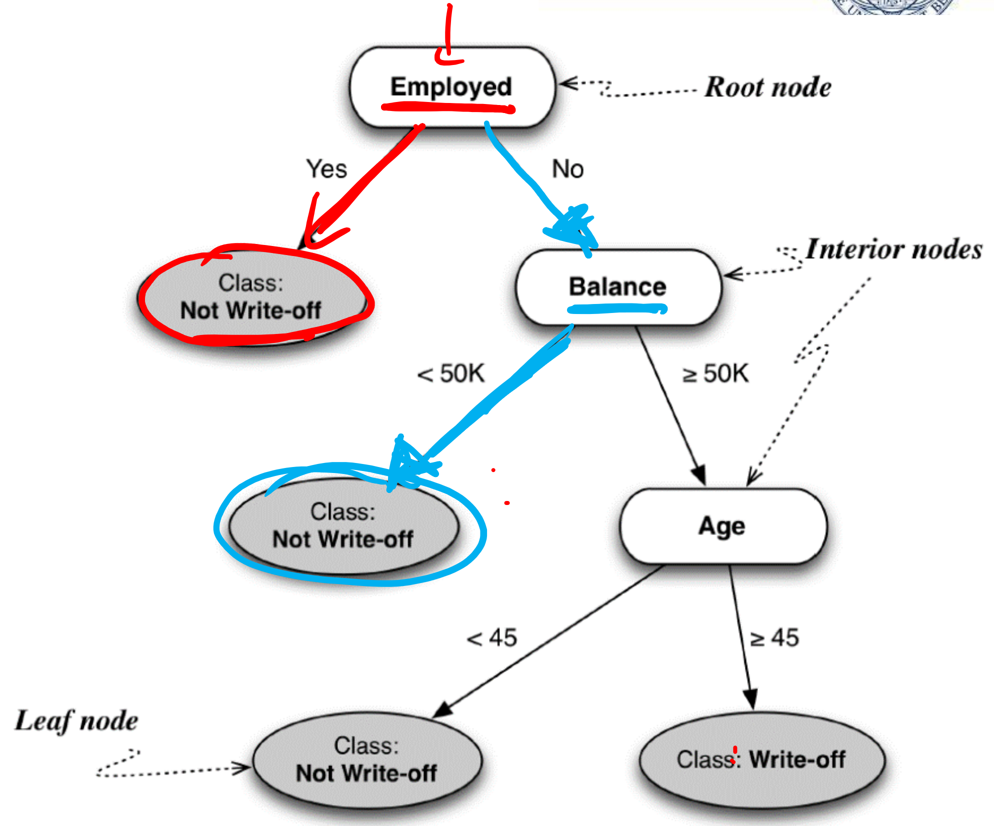
*This is one row (example).*
*Feature vector is:* **<Claudio,115000,40,no>**
*Class label (value of Target attribute) is* **no**

e.g., Michi, 40,000, 24, ~~yes~~ no, Write-off?

→ no
→ no

If we select multiple attributes each giving some information gain, it‘s not clear how to put them together
→ **decision trees**

Decision trees are often used as **predictive models**




The tree creates a segmentation of the data

Each *node* in the tree contains a test of an attribute

Each *path* eventually terminates at a *leaf*

Each leaf corresponds to a *segment*, and the attributes and values along the path give the characteristics

Each leaf contains a value for the target variable

Ref.

# Agenda

Next week
(including the Python-exercise on Friday)

(1) Models and Induction

(2) Attribute Selection

(3a) Decision Trees Algorithmic view

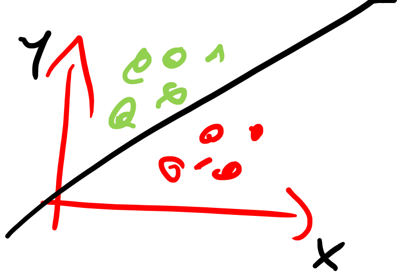
(3b) Probability Estimation

(3c) Decision Tree Examples

Ref.

# Supervised segmentation

## Intuitive approach




Segment the population into **subgroups** which have different values for the target variable (*high inter-group discrimination*) and similar values for the target variable within the subgroup (*low intra-group discrimination*)

Segmentation may provide a **human-understandable set of segmentation patterns**
(e.g., *„Middle-aged professionals who reside in New York City on average have a churn rate of 5%“*)

How can we (automatically) judge whether a variable contains important information about the target variable?

*What variable gives us the most information about the future churn rate of the population?*

Ref.

# Supervised Segmentation – Decision Tree Example

How can we (automatically) judge whether a variable contains important information about the target variable?

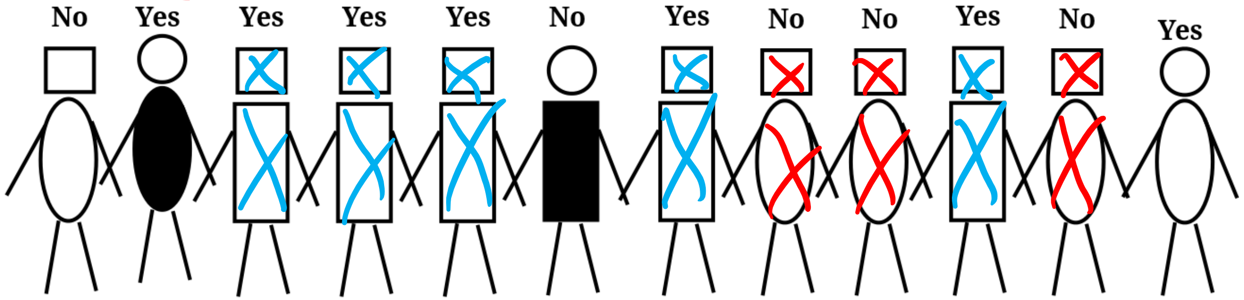
Yes
NO

Kopf und Körper
→ gleiche Form => yes
unterschiedlich => no

**Consider a binary (two class) classification problem**

Binary target variable: {“Yes”,”No”}

Attributes: head-shape, body-shape, shirt-color




*Which of the attributes would be the best to segment these people in groups such that write-offs will be distinguished from non-write-offs?*

Resulting groups should be as **pure** as possible!

Ref.

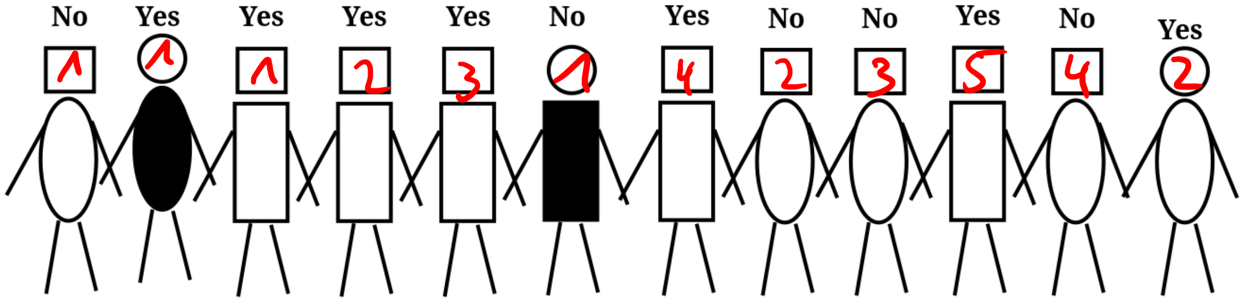
5 Min.

# Exercise – attribute selection

## And a first step to build decision trees

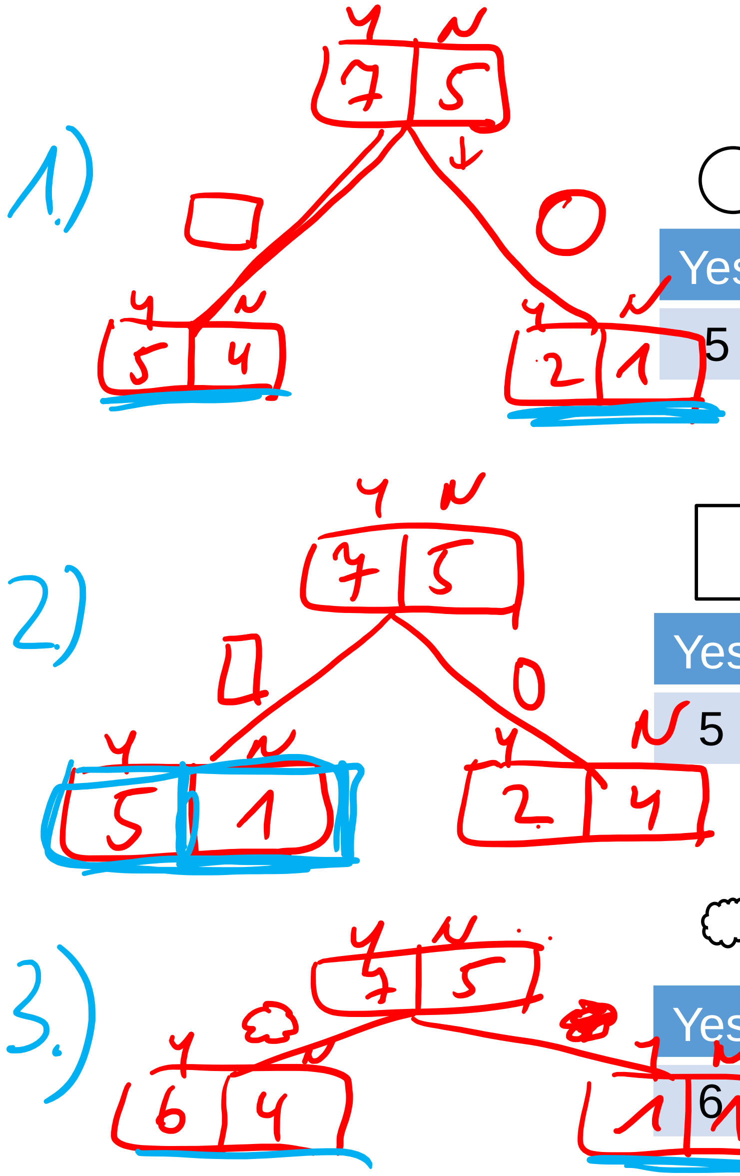
A B C D
++ --

Which attribute should be selected first?




| | # | target: yes | no |
|---|---|---|---|
| head-shape: | | | |
| square | 9 | 5 | 4 |
| circular | 3 | 2 | 1 |
| body-shape: | | | |
| rectangular | 6 | 5 | 1 |
| oval | 6 | 2 | 4 |
| shirt-color: | | | |
| white | 10 | 6 | 4 |
| black | 2 | 1 | 1 |

Σ 12




| Yes | No |
|---|---|
| 7 | 5 |

| Yes | No |
|---|---|
| 5 | 4 |

| Yes | No |
|---|---|
| 2 | 1 |

| Yes | No |
|---|---|
| 7 | 5 |

| Yes | No |
|---|---|
| 5 | 1 |

| Yes | No |
|---|---|
| 2 | 4 |

| Yes | No |
|---|---|
| 7 | 5 |

| Yes | No |
|---|---|
| 6 | 4 |

| Yes | No |
|---|---|
| 1 | 1 |

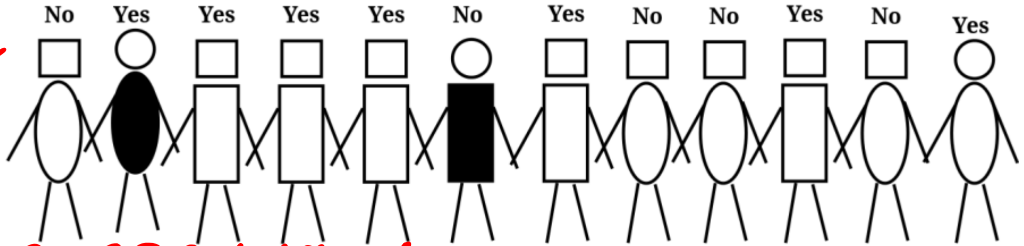
Ref.

# Reduce impurity

Freie Universität Berlin

Attributes rarely **split a group perfectly**

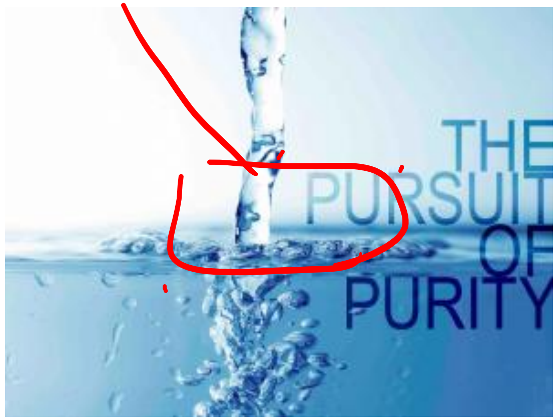
- Consider if the second person were not there
  – Then, *shirt-color* would create a pure segment where all individuals have (*write-off=no*)
- – Then, the condition *shirt-color=black* would only split off one single data point into the pure subset. Is this better than a split that does not produce any pure subset, but reduces the impurity more broadly?
- Not all attributes are binary. How do we compare the splitting into two groups with **splitting into more groups**?
- Some attributes take on **numeric values**. How should we think about creating supervised segmentations using numeric attributes?




32 28 21 19 16 22 4 21 20 17 32 24

[16-20] [20-30]

≥ [16-20)




Purity measure → **information gain / entropy**

Ref.

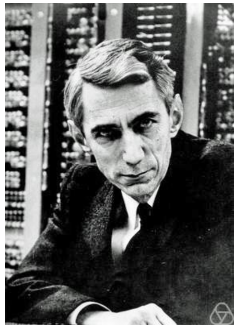
# Entropy

$-\sum p_i \log_{(2)}(p_i)$

$-$ $+$

Entropy is a **measure of disorder** that can be applied to a set

Disorder corresponds to how mixed (impure) a segment is w.r.t. the properties of interest (values of target)

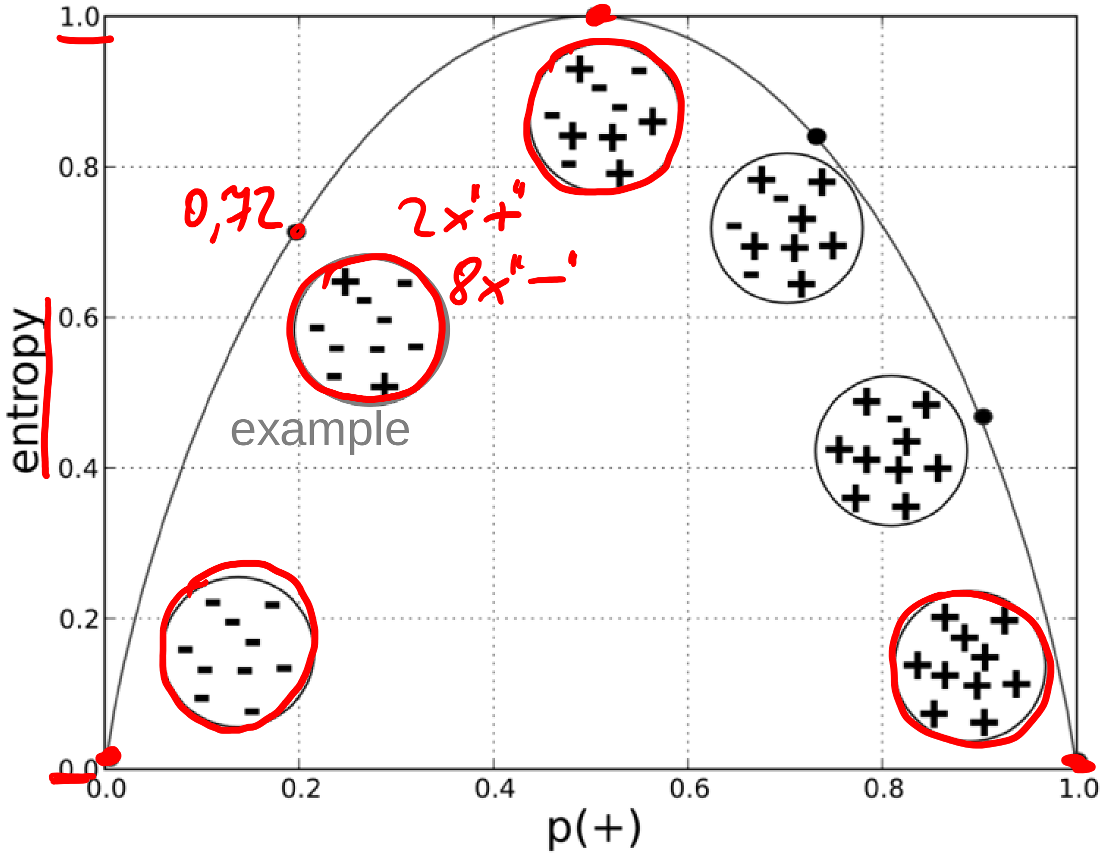


Claude E. Shannon

$$\boldsymbol{entropy = -p_1 \log_2(p_1) - p_2 \log_2(p_2) - \cdots - p_n \log_2(p_n)}$$

with $p_i$ as the relative percentage of property $i$ within the set, ranging from $p_i = 0$ to $p_i = 1$ (all have property $i$).

**Entropy measures the general disorder of a set**, ranging from ***zero*** at minimum disorder (the set has members all with the same, single property) to ***one*** at maximal disorder (the properties are equally mixed)




$p(-) = 8/10 \quad p(+) = 2/10$

$entropy(S) = -[0.8 \times log_2(0.8) + 0.2 \times log_2(0.2)]$
$= -[0.8 \times (-0.32) + 0.2 \times (-2.32)] \approx 0.72$

Example

python™

```
#Entropy :
import math as m
def entropy(p1,p2):
     return - p1 * m.log2(p1) - p2 * m.log2(p2)

#Excursus (alternatively): gini-coefficient
def gini(p1,p2):
    return 1- (p1*p1 + p2*p2)
```

# Information gain (IG)

Basic idea behind information gain:
Measure how much an attribute improves (**decreases**) entropy over the whole segmentation it creates.

IG measures the change in entropy due to any amount of new information added

How much purer are the **children c** (split set) compared to their **parent** (original set)?

$$IG(parent, children) = entropy(parent) - [p(c_1) \times entropy(c_1) + p(c_2) \times entropy(c_2) + \cdots]$$

The entropy for each child $c_i$ is weighted by **the proportion of instances** belonging to that child

Ref.

# Information gain

## Example 1

$-\sum p_i \log_2(p_i)$

Two-class problem (• and ★)

Entropy parent:

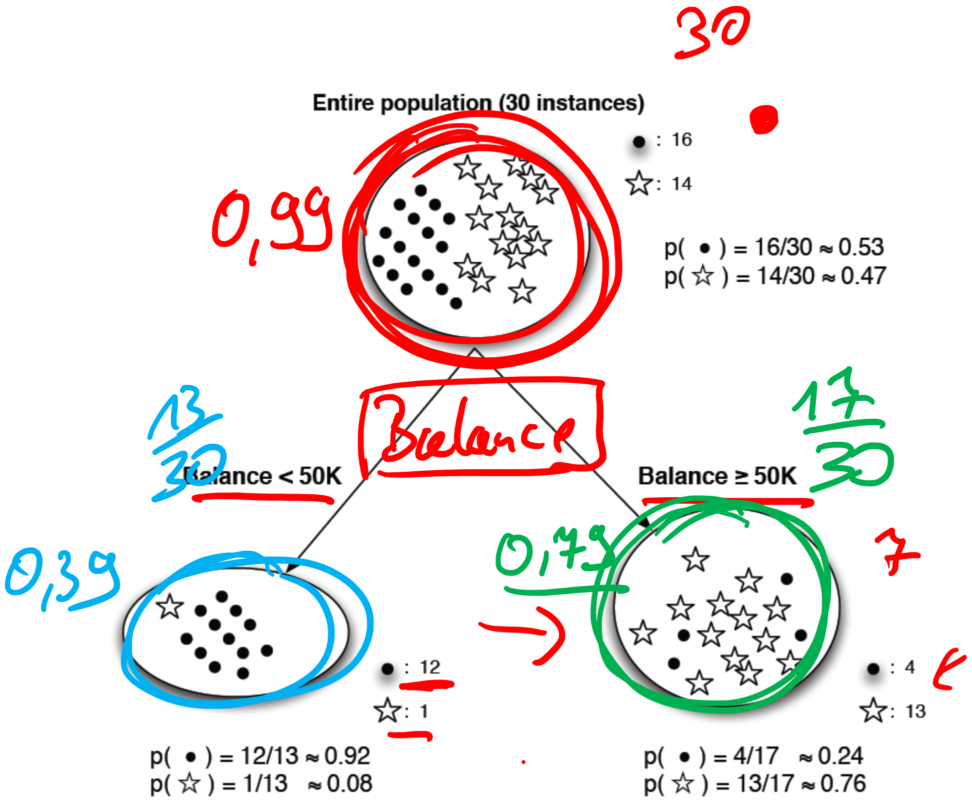
$$
\begin{aligned}
&= -[p(\bullet) \times \log_2 p(\bullet) + p(\star) \times \log_2 p(\star)] \\
&\approx -[0.53 \times -0.9 + 0.47 \times -1.1] \\
&\approx 0.99 \quad \text{(very impure)}
\end{aligned}
$$

Entropy left child:

$$
\begin{aligned}
&= -[p(\bullet) * \log_2 p(\bullet) + p(\star) * \log_2 p(\star)] \\
&\approx -[0.92 \times (-0.12) + 0.08 \times (-3.7)] \\
&\approx 0.39
\end{aligned}
$$

Entropy right child:

$$
\begin{aligned}
&= -[p(\bullet) * \log_2 p(\bullet) + p(\star) * \log_2 p(\star)] \\
&\approx -[0.24 \times (-2.1) + 0.76 \times (-0.39)] \\
&\approx 0.79
\end{aligned}
$$




IG:

Information Gain

0,99    13/30    0,39

$$
\begin{aligned}
&= \mathit{entropy}(\mathit{parent}) - [p(\textbf{Balance} < \textbf{50K}) \times \mathit{entropy}(\textbf{Balance} < \textbf{50K}) \\
&\quad + p(\textbf{Balance} \geq \textbf{50K}) \times \mathit{entropy}(\textbf{Balance} \geq \textbf{50K})] \\
&\approx 0.99 - [0.43 \times 0.39 + 0.57 \times 0.79] \\
&\approx 0.37
\end{aligned}
$$

0,79

17/30

Ref.

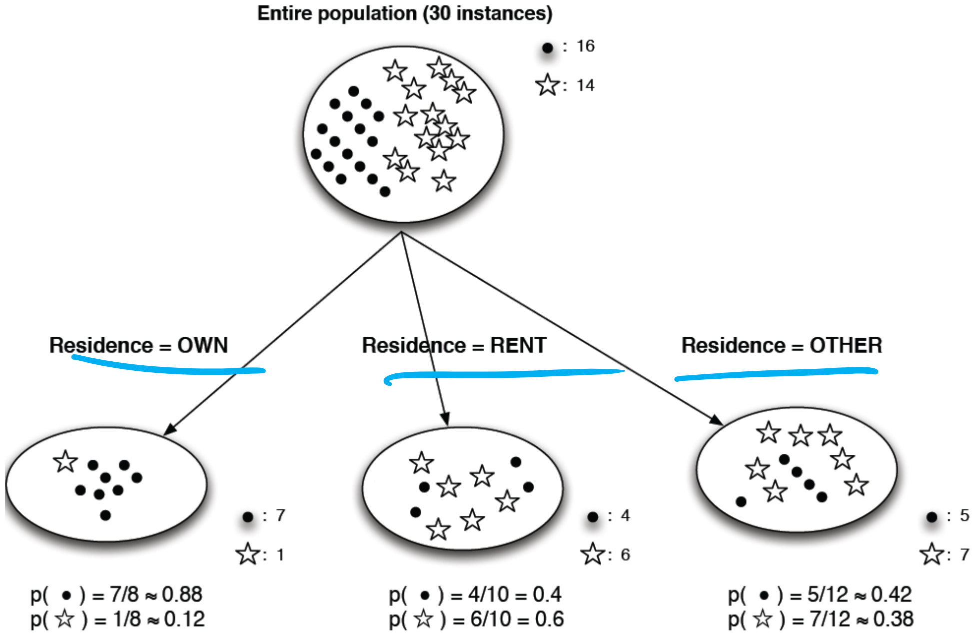
# Information gain

## Example 2

Same example, but different candidate split

attribute here: *residence*

entropy and information gain computations:




$$
\begin{aligned}
entropy(parent) &\approx 0.99 \\
entropy(\textbf{Residence=OWN}) &\approx 0.54 \\
entropy(\textbf{Residence=RENT}) &\approx 0.97 \\
entropy(\textbf{Residence=OTHER}) &\approx 0.98 \\
IG &\approx 0.13
\end{aligned}
$$

The *residence* variable does have a positive information gain, but it is lower than that of *balance*.

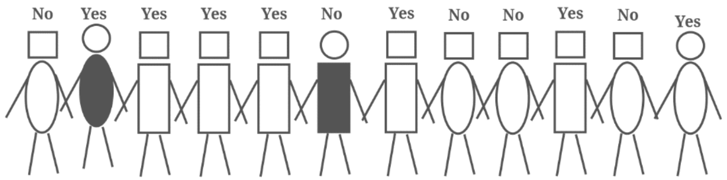
Ref.

# Exercise – Information gain

## Example 3, 4 and 5

**10 Min.**

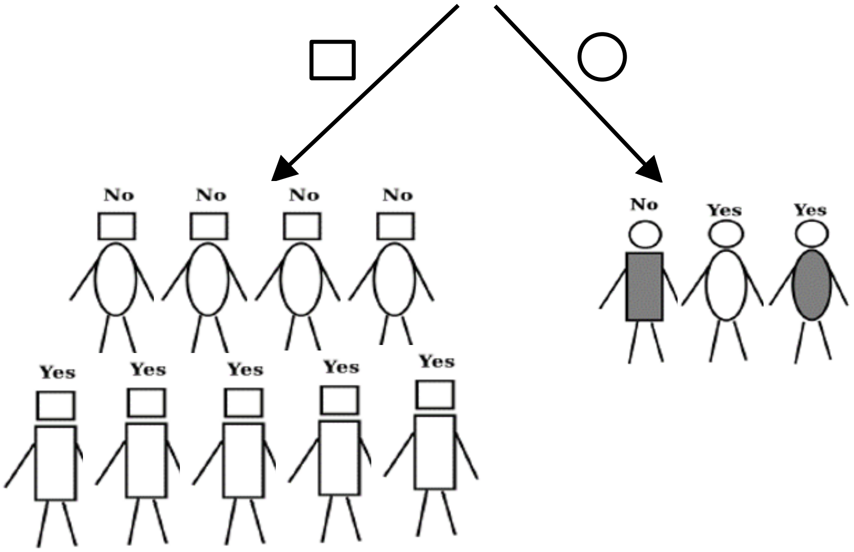
Which attribute to choose?



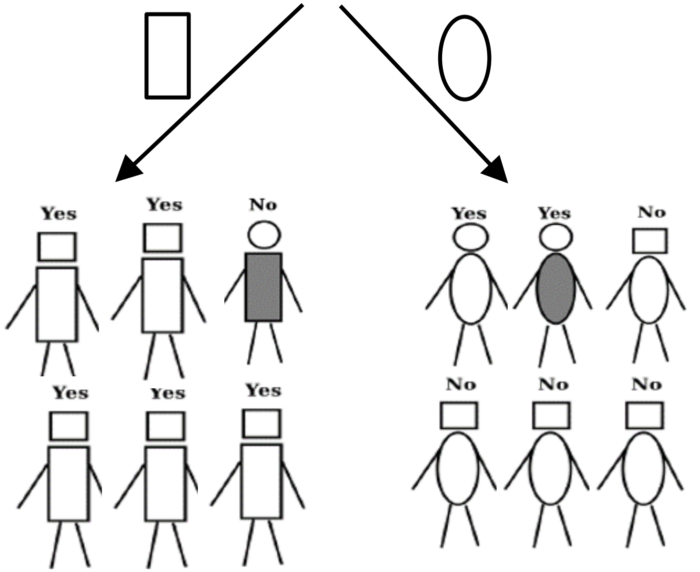

Yes: 7
No: 5

*entropy(parent)* ≈ 0,98

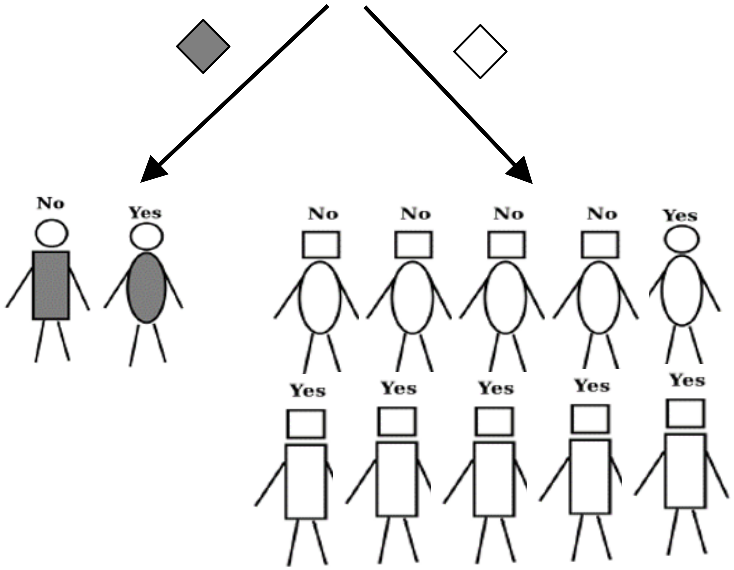
Head shape




Body shape




Body color




*entropy*(□)≈ 0,99 *entropy*(○)≈ 0,92

IG ≈ 0,007

*entropy*(▯)≈0,65 *entropy*(⬯)≈0,92

IG ≈ 0,196

*entropy*(◆)≈ 1,00 *entropy*(◇)≈0,97

IG ≈ 0,004

Ref.

# Agenda

Next week
(including the Python-exercise on Friday)

(1) Models and Induction

(2) Attribute Selection

(3a) Decision Trees Algorithmic view

(3b) Probability Estimation

(3c) Decision Tree Examples

Ref.

Freie Universität Berlin

# Fragen?

- ✓ Predictive modeling I

- ✓ Models and induction
- ✓ Attribute selection
- ✓ Decision trees - introduction

- ○ Algorithms for tree induction
- ○ Probability estimation tree

# Todos for this week

Choose your project and your project group (4 persons per group)

*See slides BI-Project ("Folien – Projektaufgabe – ab 7.6.'24" in Blackboard)*

>> Tagesspiegel


>> FAZ


SUSTAINABLE DEVELOPMENT GOALS

>> United Nations

## Costa Rican Household Poverty Prediction

Set of household characteristics from a representative sample of households
Make sure the right people are given enough aid

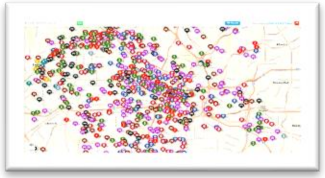
*Goal: Predict the level of need (income level)*

## West Nile Virus Prediction in Chicago

7 years of weather, location, testing, and spraying data

Goal: *predict the presence of West Nile virus for a given time, location, and species*

>> Crime Mapping



## Crime Classification in Los Angeles

4 years of crime reports from all across Los Angeles

Goal: *Predict the category of crime that occurred given a certain time and location*

## Crime Classification in San Francisco

12 years of crime reports from all across San Francisco

Goal: *Predict the category of crime that occurred given a certain time and location*

Ref.

# Recommended reading

## Data Preparation

| | |
|---|---|
| Berthold et al. | Chapter 4, 6 |

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

## Predictive Modeling

| | |
|---|---|
| Provost, F., Fawcett, T. | Data Science for Business<br>Chapter 3 |
| Berthold et al. | Guide to Intelligent Data Analysis<br>Chapter 8.1 |
| Hand, D. | Principles of Data Mining<br>Chapter 10 |
| Quinlan, J.R. | Induction of Decision Trees (in: Machine Learning, 1(1), p. 81-106, 1986) |

Ref.

# Bibliography


- J. Bertin (1983) *Semiology of graphics: diagrams, networks, maps*. University of Wisconsin Press. Originally in French: *Semiologie Graphique*, 1967
- Cairo, A. (2012). *The Functional Art: An introduction to information graphics and visualization*. New Riders.
- Mertens, P., & Meier, M. (2009). *Integrierte Informationsverarbeitung*. Wiesbaden: Gabler.
- Woolman, M. (2002). *Digital information graphics*. Watson-Guptill Publications, Inc..


Ref.