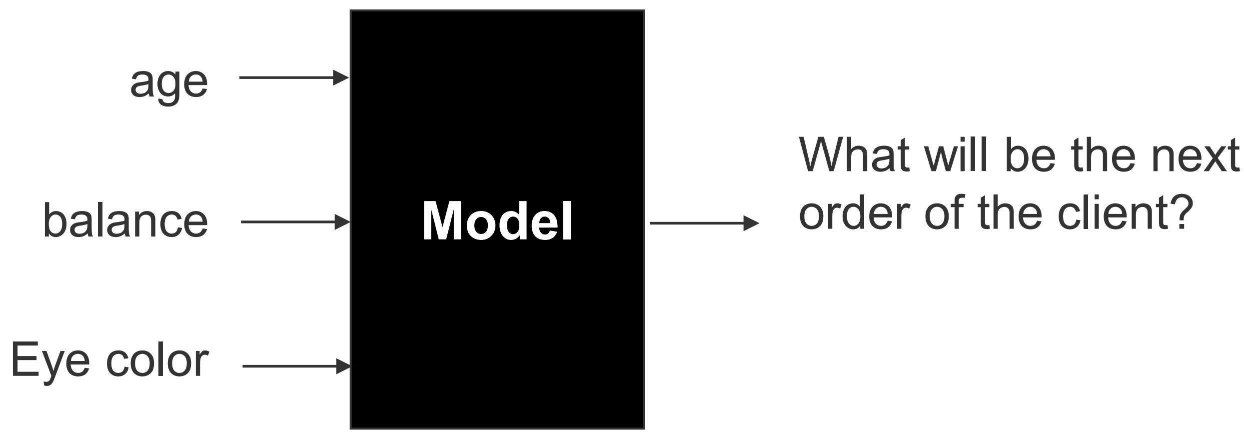
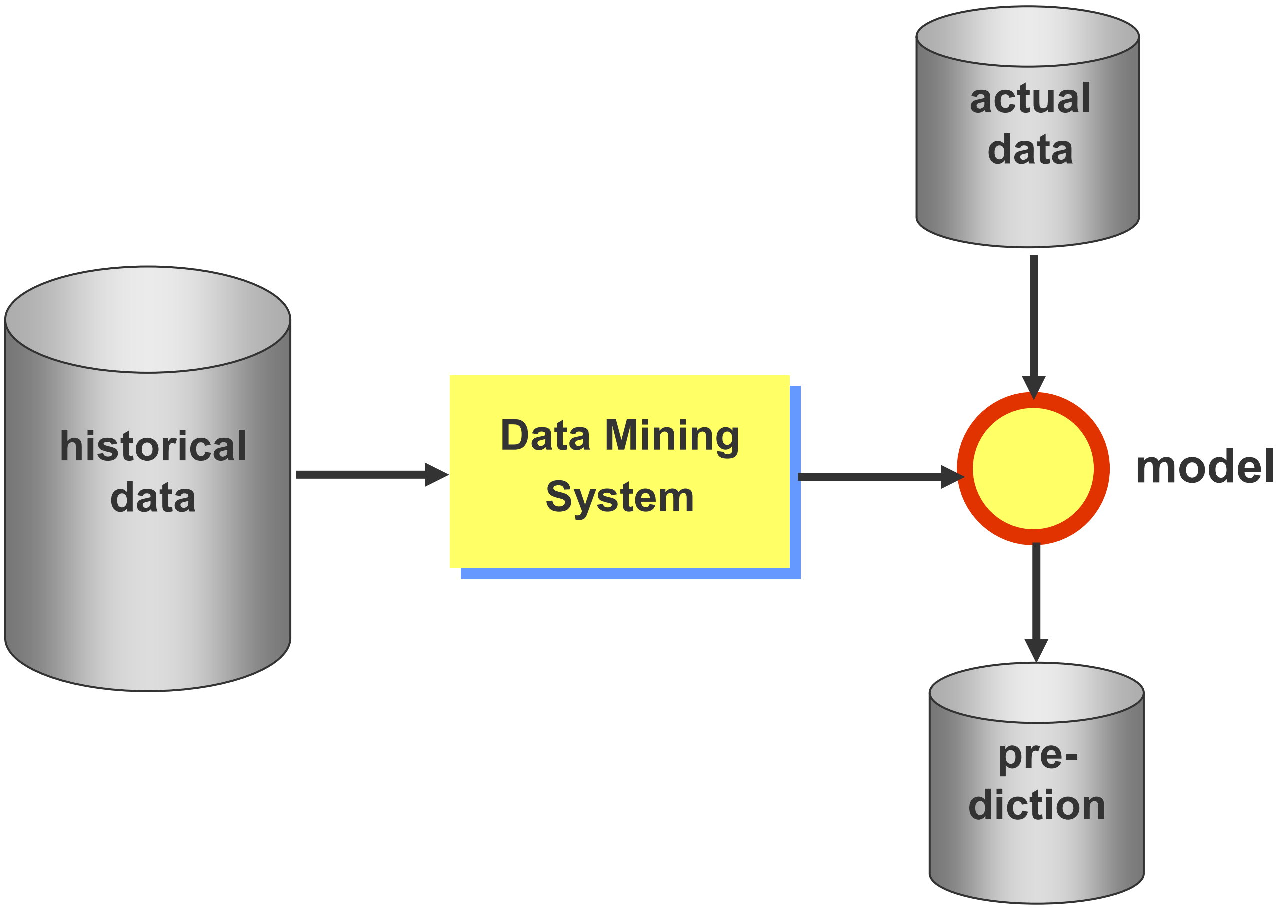
What is a predictive model

A black box to predict the future based on informations from the past and the present



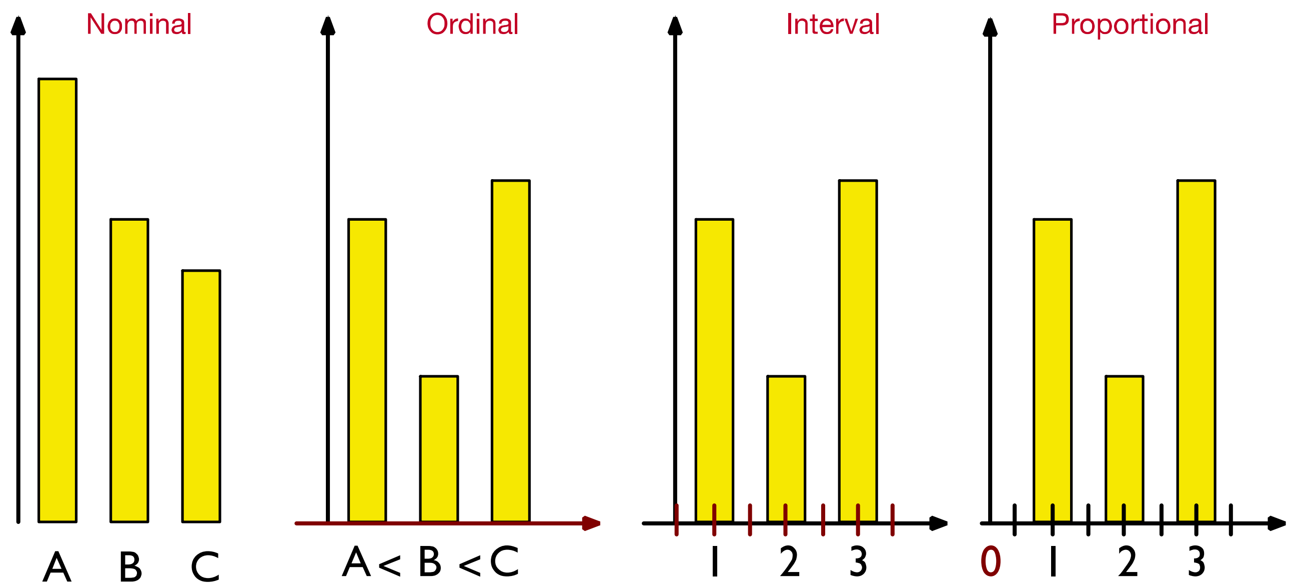
Development and use of predictive models



Quality aspects of a Data Mining Analyse

- The success of a data mining method depends on the relevance, reliability and validity of the variables (data quality)
- The validity (generalizability) of the results depends on whether the specialist department contributes hypotheses about attributes and relationships of the data population
- If only a selection of the data population is available as data elements (partial survey), the analyst must ask how reliably he can generalise the results of the sample
- A distinction is made between the learning set and the test set if a second set of data (the test set) is to validate the result of the learning set.

Scaling of Features



- nominal: only frequencies
- ordinal: + order
- interval: + distances
- proportional: + zero point

Scale levels in comparison.
Red: The properties newly added at the respective scale level.

Data Preprocessing

Data preprocessing consists of five tasks

1. Reduce data objects («sampling»)
2. Reduce features («feature selection» or «dimensionality reduction»)
3. Treat defective and missing features
4. Normalize features
5. Adjust scaling

Reducing the Size (Sampling)

Why sampling?

- Counteract performance problems.
- Some methods are not applicable to too many records.

Requirement: the sample should reflect the context of the raw data

- i.e., it should not be biased.

Sampling methods:

- Different selection procedures are available
- Mostly: random selection of data objects

Reducing the Dimensions

Why reducing dimensions?

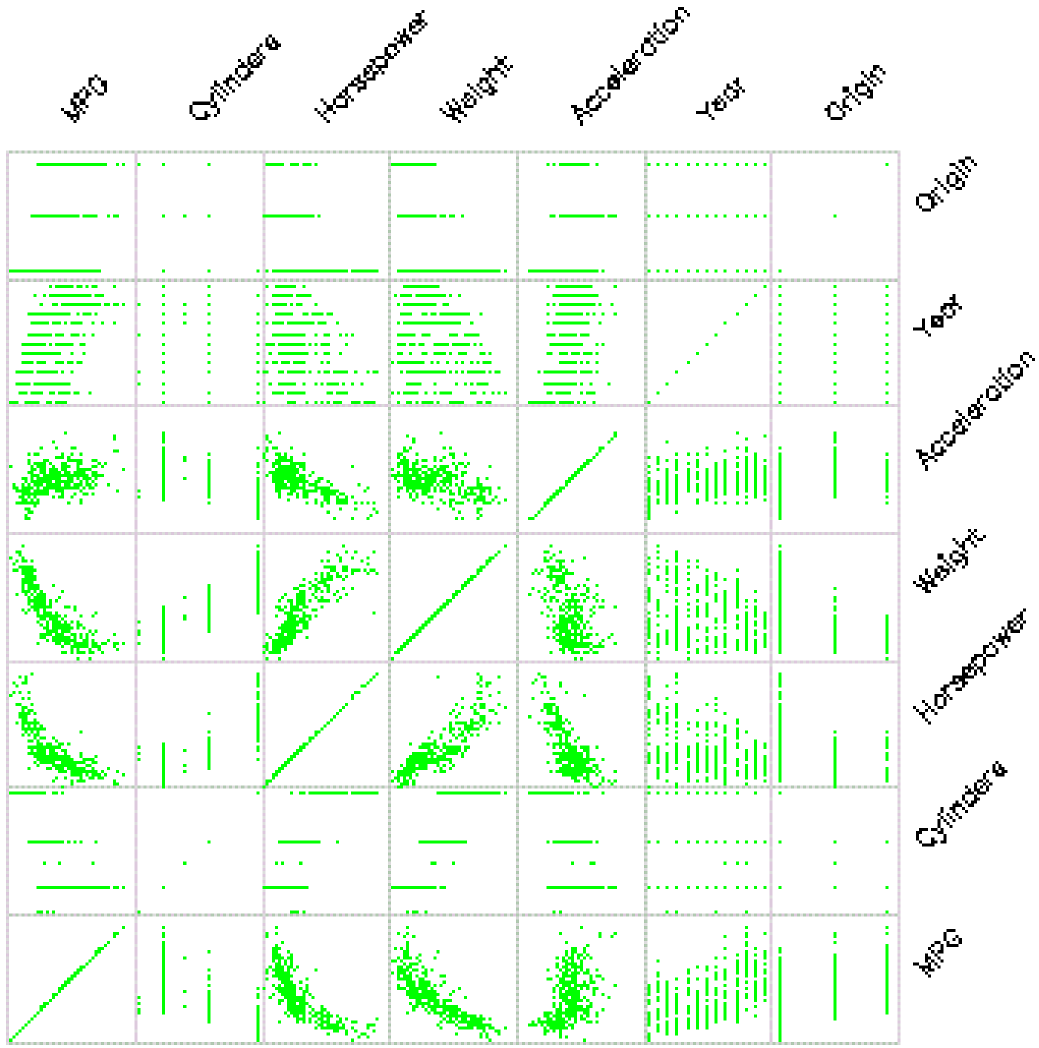
- Many data mining methods work better with less features.
- Some features may be redundant, irrelevant, or disruptive (noisy).
- Resulting models are more comprehensible.
- Visualization is only possible with 2 or 3 dimensions.
- Danger of overfitting if the number of features is much larger than the number of records

Two classes of methods:

1. Dimensionality Reduction
2. Feature Selection

Visual attribute reduction

Scatter matrix



2.3.2 Feature reduction

1. Remove obviously irrelevant or redundant features. Do manually:
   - E.g., the customer ID is irrelevant to predict a target variable
   - E.g., age and age group are redundant

2. For the rest, define an optimal subset of features. Apply systematic approach:
   - Total set of features: $F = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$.
   - Evaluate features sets using a quality measure:
     - For $F' \subseteq F$ let $J(F')$ be the quality of F'.
     - Goal: Find the subset F' with best possible quality (minimal or maximal $J(F')$).
   - two main approaches:
     - «Filter Method»
     - «Wrapper Method

Missing and Defective Features

A record x_i must be “repaired”, if….

… it is not complete.

I.e., in the data object $\boldsymbol{x}_i$, the j-th feature value x_{ij} is missing (for some j and i).

… it is erroneous.

I.e., in the data object $\boldsymbol{x}_i$, the j-th feature value x_{ij} is erroneous (for some j and i).

Missing and Defective Features

- Random errors occur very often during manual recordings of data
  - E.g., spelling errors, digit errors, comma wrongly set, etc.

- Systematic errors occur
  - E.g., when a sensor does not properly work during automatic data acquisition
  - E.g., when an employee systematically makes erroneous input during data collection.
  - E.g., change of the input system
  - E.g., change of the business meaning

Missing and Defective Features

- Erroneous records deviate very much from the rest of the data. In this case they are called outliers.
  - Note: Not all outliers are errors! The sample data may contain correct, but exceptional feature values that can provide valuable information.

- Methods of Outlier-Detection:
  - Visualization often sufficient (outliers are very far above or below the "normal" distribution of the feature values).
  - Statistical measures.
    - Example: k-sigma rule: An attribute value is considered an outlier when it deviates by more than k times the standard deviation from the mean value of the attribute. Typical are: k = 2, 3, 4, 5

Missing and Defective Features

1. Remove records with missing or erroneous values from the sample
   - I.e., remove a row from the data set.
   - Risk: Unintentional distortion of the sample.

2. Remove attributes with missing or erroneous values from the sample
   - I.e., remove a column from the data set.
   - Unfavorable if the attribute contains important information for modeling.

3. Encode missing / erroneous values with a special feature expression
   - Numerical Features: Special “out-of-range” number
     - E.g., “-1” or NaN (Not a Number)
   - Categorical Features: Special category
     - E.g., unknown, erroneous.
   - Note: The so coded attribute values must be treated particularly in the following data analysis!

4. Complete or replace missing / erroneous values with:
   - Statistical measure
     - E.g., min, max, mean, median, mode of the feature
   - Feature value of the most similar data object in the sample
     - Cf. nearest neighbor

Normalizing Features

Different attributes represent different variables.

→ Range of values can be very different.
→ May lead to problems during the data mining process.

1. Transform each feature so that mean = 0 and standard deviation = 1:

$$\hat{x}_{i_j} = \frac{x_{i_j} - m_j}{\sigma_j}$$

where $\hat{x}_{ij}$ is the value of the *j*-th feature after normalization.

2. Incorporate the features into a fixed interval, e.g., $[0,1]$:

$$\hat{x}_{i_j} = \frac{x_{i_j} - min}{max - min}, \text{where} \quad \begin{array}{l} min = \min\{x_{i_j} | i = 1, \ldots, N\}; \\ max = \max\{x_{i_j} | i = 1, \ldots, N\}. \end{array}$$

Notice:

- For a new data object $x_{ij} \in [\min, \max]$ need not hold!
- Therefore $\hat{x}_{ij}$ can be outside [0,1].

Adjust Scaling

Changing the scale of measurement

- Always possible from higher to lower scale level.
  - E.g., from numerical to categorical.
  - Method, e.g.: discretization.
- Usually not possible from a lower to a higher scale level.
  - E.g., no exact numerical values can be derived from categories

Adjust Scaling

Discretization

- Summarizes numerical attributes.
- Result: finite number of subsets → categorical feature.

Why discretization?

- Often leads to more clarity of data structure → can simplify analysis
- Some data mining methods can only process categorical attributes.

Notice:

- Part of the original information is lost!
- Often already applied during sampling.
  - E.g. Surveys: one does not have to specify the exact income but only the income class.

Adjust Scaling

Discretization Methods

- Equal Width Binning
  - Approach:
    1. Specify number of categories *n*. Sort the feature values in ascending order. Divide the range of feature values into *n* intervals of same size (by defining $n - 1$ split points). The intervals represent the categories.
       - E.g., $\{ [0,10[, [10,20[, [20,30[\}$).
    2. Map all values within an interval to the corresponding category.
  - Disadvantage: Often unequal distribution of data objects in bins.
    - Some intervals may contain a lot of data objects, others may be almost empty.
    - This can negatively affect data mining results.

2.3.5 Adjust Scaling

Discretization Methods

- Equal Frequency Binning
  - Approach:
    - Define intervals so that each interval contains (approximately) the same number of data objects from the sample.
- Clustering based Binning
  - Approach:
    - Divide the feature values into categories using a clustering algorithm that takes into account the feature to be discretized

Adjust Scaling

Binarization can be used to transform categorical features

- Approach:
  - Categorical attribute with *k* possible values.
  - Replace it by *k* artificial *binary* attributes.
  - Each of these *k* artificial attributes represents a possible occurrence of the categorial attribute, and is equal to 1 if the value of the original attribute corresponds to the corresponding category (cf. bitmap indexing).
- Binarization is important for calculating similarities, or dissimilarities of features.
- Example 10: Binarization.

x_i
awful
poor
OK
good
great

$\Rightarrow$

x_i=*awful*	x_i=*poor*	x_i=*OK*	x_i=*good*	x_i=*great*
1	0	0	0	0
0	1	0	0	0
0	0	1	0	0
0	0	0	1	0
0	0	0	0	1

Imbalanced distributions

- In certain cases, the classes have very different frequencies
  - Prediction of quitting in telecommunication: 97% stay, 3% quit (per month)
  - Medical diagnosis: 90% healthy, 10% sick
  - eCommerce: 99% buy nothing, 1% buy
- Similar situations with multiple classes
- A classifier that predicts the majority has an accuracy of e.g. 97%, but is worthless

Treating imbalanced data (stratification)

- Assumption: Two classes with a positive value as a minority
- Divide the raw data into a residual set (e.g. 30% of the data) and training data
  - Separate the residual set and only use it at the end
- Find the remaining positive instances (e.g. 70% of all positive instances) from the training data
- Mix them with the same number of negative instances and sort them randomly to form a balanced data set.
- Divide the resulting data set into learning and test set

Learn with imbalanced data

- Build the model with the balanced learning and test sets
- Check and determine the result with the separate raw data
- Generalization for multiple classes
  - Stratify the data
  - Ensure that all classes are equally represented in the learning and test set