1 Knowledge Discovery and Data Mining Process (KDD)

Definition Data Mining:
Process of discovering patterns in large data sets. Methods involved at the intersection of (machine learning, statistics and database systems.

Goal of KDD: Extract hidden, potentialy useful knowledge and actionable information from data.

2 Machine Learning Paradigms

- Unsupervised learning (unlabeled)
- Supervised learning (labeled)
- Reinforcement learning

2.1 Unsupervised Learning

Clustering Grouping a set of objects in such a way that objects in the same group are more similar to each other than to those in other clusters.

2.2 Supervised Learning

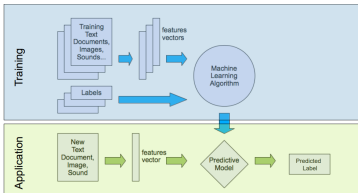

Abbildung 1: Supervised Learning.

Example
Data: Historic data from bank clients (Income, credit scores etc.)
Goal: Forecast wheter a new client should be granted a loan or not.

2.3 Reinforcement-Learning

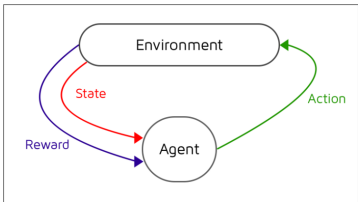

Abbildung 2: Reinforcement Learning.

2.4 Data

- **Data in Analaytics**
  - **Structured Data**
    - **Categorical**
      - Nominal
      - Ordinal
    - **Numerical**
      - Continous
      - Discrete
  - **Unstructured or Semistructured Data**
    - **Textual**
    - **Multimedia**
      - Image
      - Audio
      - Video
    - **XML/JSON**

2.4.1 Data Classes

- One-Dimensional data
- Multi-Dimensional data
- Network data
- Hierarchical data
- Time-Series
- Geographic data

3 Data Preprocessing

Tasks:

- Data Integration/consolidation
  - Collects and merges data from multipe sources into a coherent data store
- Data Cleaning
  - Removing or modifying incorrect data, identify and reduce noise in data
- Data transformations
  - Normalize, discretize or aggregate the data
- Data reduction
  - Reduce data size by reducing the number of samples or reducing the number of attributes, balance skewed data

3.1 Data Cleaning

3.1.1 Detect (Near Duplicates)

For numeric values:
Cosine Similarity of feature vectors
For text:
Levensthein distance between the texts

3.1.2 Levensthein Distance

Computes the minimum number of **Edit Operations** that are necessary to transform a word into another word.
Operations: Insert, Delete, Replace
Runtime: O(m*n) $->$ for 2 words of length m and n, with a dynamic programming algorithm

3.1.3 Cosine Similiraty

Given:

$$A = \begin{pmatrix} a_1 \\ a_2 \\ . \\ . \\ . \\ a_n \end{pmatrix} \quad B = \begin{pmatrix} b_1 \\ b_2 \\ . \\ . \\ . \\ b_n \end{pmatrix}$$

$$cos(\theta) = \frac{A*B}{||A||*||B]]}$$

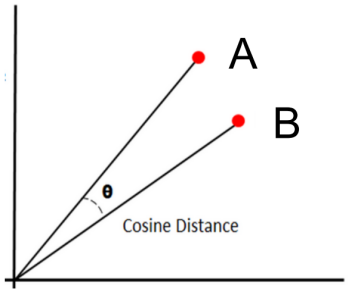

Abbildung 3: Cosine Similarity.

3.1.4 Missing Values

- Interpolation
- Durchschnitt
- Regression
- Werte vordefinieren zum anwählen

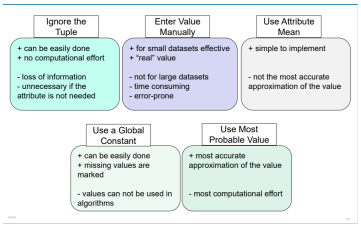

Abbildung 4: Missing Values.

3.1.5 Outlier Detection

Outlier Detection with Clustering (DBSCAN)

3.1.6 Smoothing

Goal: Make patterns noticeable, eliminate disturbances and outliers from data.
Methods:

- Binning
- Regression
- Clustering

Binning

Equal-width Binning

$Width = \frac{(max-min)}{N}$

- N = number of intervals (Example N=3)
- 24, 28, 29, 35, 41, 41, 44, 45, 46, 48, 49, 54
- outliers may dominate result
- Width= 10

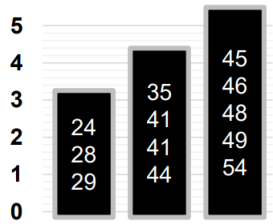

Abbildung 5: Equal-Width Binning.

Equal-depth Binning

$\frac{(Dataamount)}{N}$

- first sort Data
  - 24, 28, 29, 35, 41, 41, 44, 45, 46, 48, 49, 54
- N = number of intervals (example N=3)
  - $[24, 28, 29, 35], [41, 41, 44, 45], [46, 48, 49, 54]$
- outliers may dominate result

Smoothing by bin means: Replace each value by the mean value of the bin
$[29, 29, 29, 29], [43, 43, 43, 43], [49, 49, 49, 49]$

Smoothing by bin boundaries: Replace each value by closest boundary value
$[24, 24, 24, 35], [41, 41, 45, 45], [46, 46, 46, 54]$

3.2 Data Transformation

Discretization: Convert continous variables to discrete using binning

Aggregation: Reduce the number of categories for categorical variables applying proper concept hierarchies

Construct new features: Derive new, potentially more informative feature from the existing ones using different mathematical functions (e.g. multiplication)

4 Feature Scaling

Problem: Features have different values (e.g weights between 70-100kg vs height between 1.6-20m)

Goal: Make all features (columns of X) approximateley of equal size typically around zero