



Big Data Analytics – Week 3: Classification –

Knowledge Discovery and Data Mining
Prof. Dr. Emmanuel Müller

Classification

The systematic assignment of new observations to known categories according to criteria learned from a training set

Formally,

- a **classifier** K for a **model** $M(\theta)$ is a function $K_{M(\theta)}: D \rightarrow Y$, where
 - D : data space
 - Often d -dimensional space with attributes $a_i, i = 1, \dots, d$ (not necessarily vector space)
 - Some other space, e.g. metric space
 - $Y = \{y_1, \dots, y_k\}$: set of k distinct **class labels** $y_j, j = 1, \dots, k$
 - $O \subseteq D$: set of **training objects**, $o = (o_1, \dots, o_d)$, with known class labels $y \in Y$
- Classification: application of classifier K on objects from $D - O$

Model $M(\theta)$ is the “type” of the classifier, and θ are the model parameters

Supervised learning: find/learn optimal parameter θ for the model $M(\theta)$ from given training data

Supervised vs. Unsupervised Learning

Unsupervised learning (clustering)

- The class labels of training data are unknown
- Given a set of measurements, observations, etc. with the aim of establishing the existence of classes or clusters in the data
 - Classes (=clusters) are to be determined

Supervised learning (classification)

- Supervision: The training data (observations, measurements, etc.) are accompanied by labels indicating the class of the observations
 - Classes are known in advance (a priori)
- New data is classified based on information extracted from the training set

[WK91] S. M. Weiss and C. A. Kulikowski. Computer Systems that Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning, and Expert Systems. Morgan Kaufman, 1991.

Classification vs. Prediction

Related problem to classification: prediction

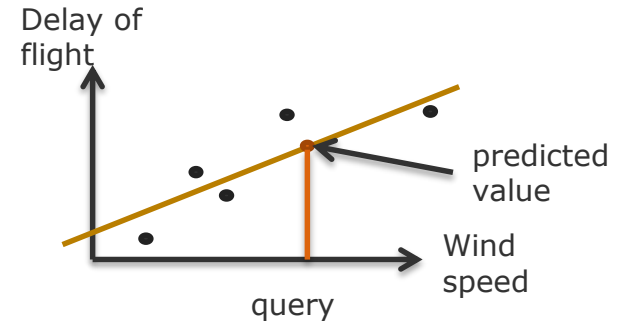
- Determine the numerical value of an object
- Method: e.g., regression analysis
- Example: prediction of flight delays

Prediction is **different** from classification

- Classification refers to predict categorical class label
- Prediction models continuous-valued functions

Prediction is **similar** to classification

- First, construct a model
- Second, use model to predict unknown value
 - Major method for prediction is regression
 - Linear and multiple regression
 - Non-linear regression



Introduction: Example

Training data

ID	age	car type	risk
1	23	family	high
2	17	sportive	high
3	43	sportive	high
4	68	family	low
5	32	truck	low

Simple classifier

if age > 50 then risk = low;

if age ≤ 50 and car type = truck then risk = low;

if age ≤ 50 and car type $\neq$ truck then risk = high.

Training Phase (Model Construction)

ID	age	car type	risk
1	23	family	high
2	17	sportive	high
3	43	sportive	high
4	68	family	low
5	32	truck	low

training data

training

unknown data

(age=60, familiy)

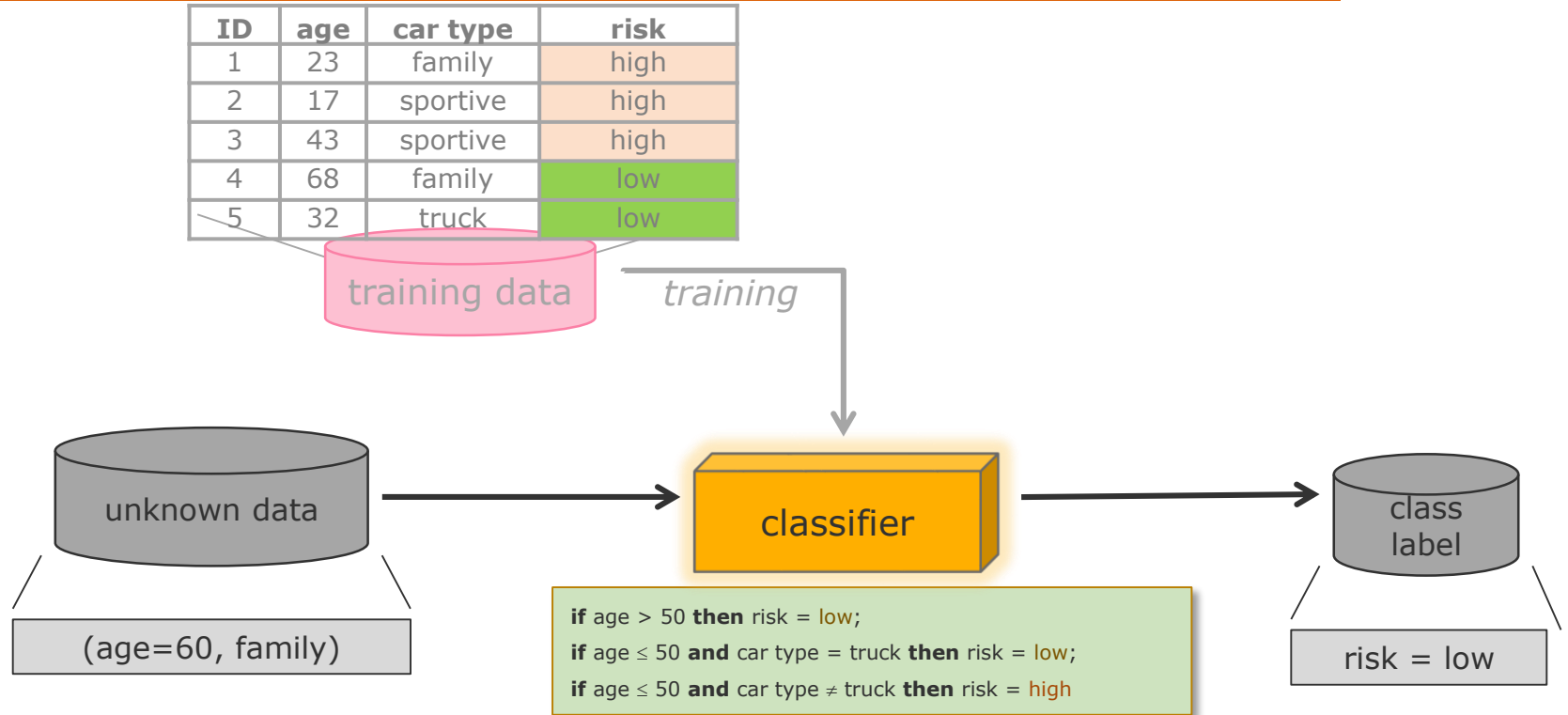
classifier

class label

```

if age > 50 then risk = low;
if age ≤ 50 and car type = truck then risk = low;
if age ≤ 50 and car type ≠ truck then risk = high
    
```

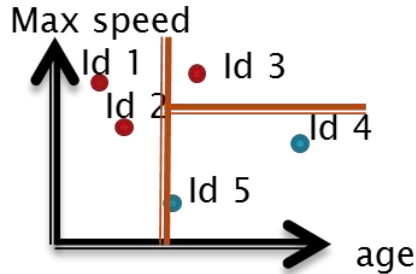
Classification Phase (Application)



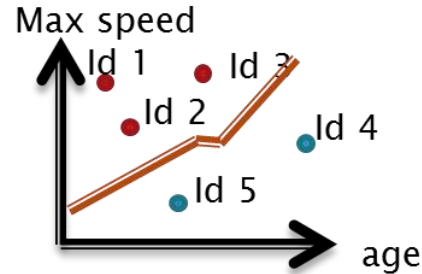
Goals of this course

ID	age	car type	max speed	risk
1	23	family	180	high
2	17	sportive	240	high
3	43	sportive	246	high
4	68	family	173	low
5	32	truck	110	low

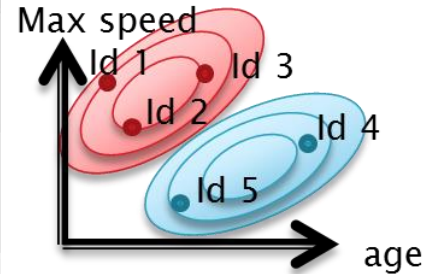
Introduction of different classification models



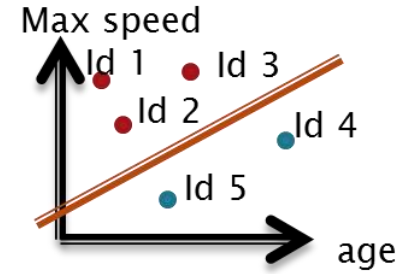
Decision trees



*k-nearest
neighbor*



*Bayes
classifier*



*Linear discriminant
function &
SVM*

Learning techniques for these models

Quality Measures for Classifiers

- Classification accuracy or classification error (complementary)
- Compactness of the model
 - decision tree size; number of decision rules
- Interpretability of the model
 - Insights and understanding of the data provided by the model
- Efficiency
 - Time to generate the model (training time)
 - Time to apply the model (prediction time)
- Scalability for large databases
 - Efficiency in persistent databases
- Robustness
 - Robust against noise or missing values

Evaluation of Classifiers – Notions

- Using training data to build a classifier and to estimate the model's accuracy may result in misleading and overoptimistic estimates
 - due to overspecialization of the learning model to the training data
- ***Train-and-Test***: Decomposition of labeled data set O into two partitions
 - *Training data* is used to train the classifier
 - construction of the model by using information about the class labels
 - *Test data* is used to evaluate the classifier
 - temporarily hide class labels, predict them anew and compare results with original class labels
- Train-and-Test is not applicable if the set of objects for which the class label is known is very small

Evaluation of Classifiers – Cross Validation

m-fold Cross Validation

- Decompose data set evenly into m subsets of (nearly) equal size
- Iteratively use $m - 1$ partitions as training data and the remaining single partition as test data.
- Combine the m classification accuracy values to an overall classification accuracy, and combine the m generated models to an overall model for the data.

Leave-one-out is a special case of cross validation ($m=n$)

- For each of the objects o in the data set O :
 - Use set $O \setminus \{o\}$ as training set
 - Use the singleton set $\{o\}$ as test set
- Compute classification accuracy by dividing the number of correct predictions through the database size $|O|$
- Particularly well applicable to nearest-neighbor classifiers

Quality Measures: Accuracy and Error

Let K be a classifier

Let $\mathcal{C}(o)$ denote the correct class label of an object o

Measure the quality of K :

- Predict the class label for each object o from a data set $T \subseteq \mathcal{O}$
- Determine the fraction of correctly predicted class labels
- *Classification Accuracy* of K :

$$G_T(K) = \frac{|\{o \in T, K(o) = \mathcal{C}(o)\}|}{|T|}$$

- *Classification Error* of K :

$$F_T(K) = \frac{|\{o \in T, K(o) \neq \mathcal{C}(o)\}|}{|T|}$$

Quality Measures: Accuracy and Error

Let K be a classifier

Let $TR \subseteq O$ be the training set – used to build the classifier

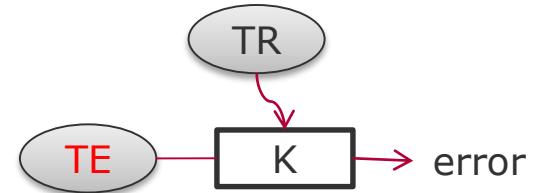
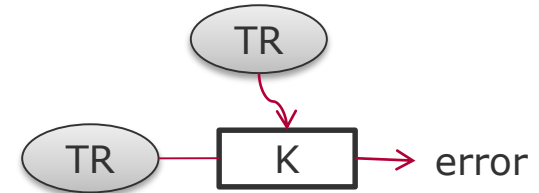
Let $TE \subseteq O$ be the test set – used to test the classifier

– *Apparent classification error of K :*

$$F_{TR}(K) = \frac{|\{o \in TR, K(o) \neq C(o)\}|}{|TR|}$$

– *True classification error of K :*

$$F_{TE}(K) = \frac{|\{o \in TE, K(o) \neq C(o)\}|}{|TE|}$$

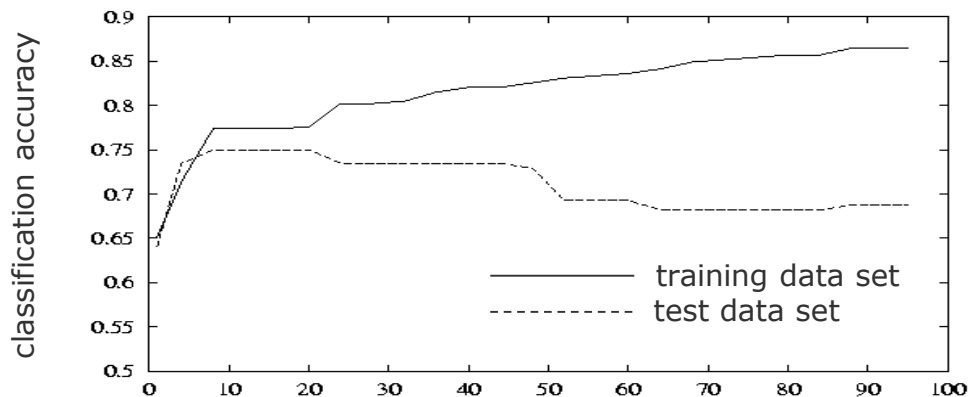


Overfitting

Characterization of overfitting:

There are two classifiers K and K' for which the following holds:

- on the training set, K has a smaller error rate than K'
- on the overall test data set, K' has a smaller error rate than K



Parameter of classifier

generalization ← **classifier** → **specialization**
 "overfitting"

Overfitting (2)

Overfitting

- occurs when the classifier is optimized to noisy training data
- The classifier yields worse results for test data set
- Potential reasons
 - bad quality of training data (noise, missing values, wrong values)
 - different statistical characteristics of training data and test data

Overfitting avoidance

- Removal of *noisy* and *erroneous* training data; in particular, remove contradicting training data
- Choice of an appropriate *size* of the training set: not too small, not too large
- Choice of appropriate sample:
sample should describe all aspects of the domain and not only parts of it