Database and Information Systems

What is Data Mining?

- What is Data Mining?
- Data-Mining as a Process
- Data Preprocessing
- Data Mining Tasks

Data Mining

Readings:

Dunham, Margaret H.: Data Mining - Introductory and Advanced Topics. Pearson Education, 2003, Chapter 3-6.

Kantardzic, Mehmet: Data Mining - Concepts, Models, Methods, and Algorithms. Wiley-Interscience 2003, Chapter 2-9.

Larose, Daniel T.: Data mining methods and models, Wiley-Interscience, 2006.

Tan, Pang-Ning ; Steinbach, Michael ; Kumar, Vipin: Introduction to data mining, Pearson/Addison-Wesley, 2006.

What is Data Mining?

- What is Data Mining?
- Data-Mining as a Process
- Data Preprocessing
- Data Mining Tasks

What is Data Mining?

- mining as a metaphor:
  - the search for the precious thing amongst the overburden
  - goal is not always clearly specified
    - what's precious?
    - successful even if looking for gold but found diamonds
  - mining is an explorative activity
    - finding cues
    - making hypotheses
    - evaluating hypotheses
    - getting the precious stuff

What is Data Mining?

- "data mining" is a misleading metaphor!

	mining	data mining
target	coal, gold, ore, ...	trends, associations
overburden	dirt, rock	"useless" data

- alternative notion:
  - KDD: Knowledge Discovery in Databases
- alternative view:
  - KDD is a complex process, DM only one step in it

Example Tasks

- customer relationship management
    - grouping of customer populations (tailored marketing)
    - prediction of customer behaviour (individualized marketing)
    - risk assessment (risky credits, fraudulent credit card use)
- fault analysis
    - interdependencies between faults
    - interdependencies between maintenance procedures and faults
- time-series analysis
    - stock market development
    - event prediction (stock market crashes, bankruptcies)

Why is Data Mining special?

- There is no silver bullet for data mining!
- quering a data base:
    - precise semantics
    - 100% correct results must be guaranteed
    - issues: performance, maintenance, …
- data mining:
    - vague task descriptions
    - no perfect solution methods
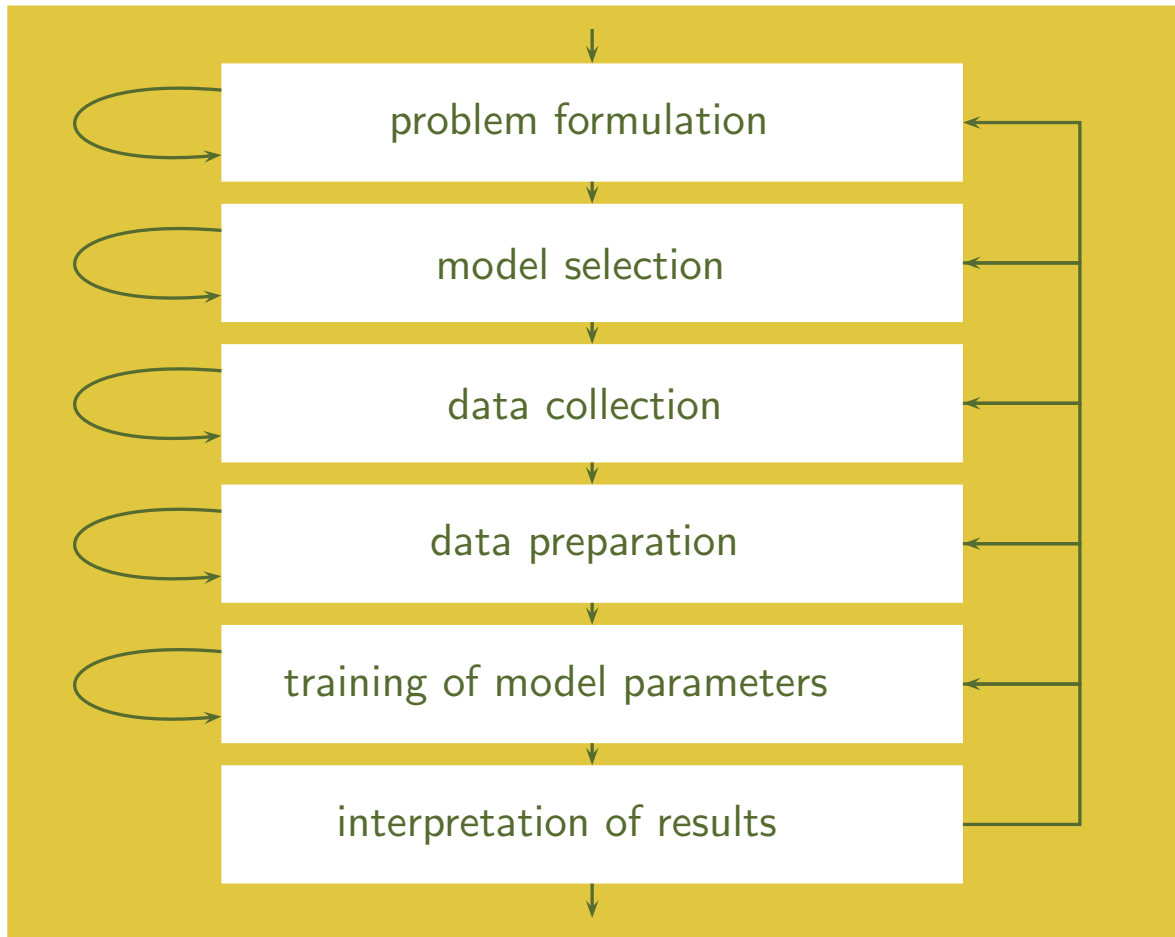   - issues: quality, portability

Evaluation

- models in data-mining are usually developed and evaluated on data collections
- for development purposes clear separation between training and test data necessary

Data Mining as a Process

Data Mining as a Process

problem formulation

model selection

data collection

data preparation

training of model parameters

interpretation of results

Data-Mining as a Process

1. problem formulation
   - based on domain-specific knowledge and experience
   - Which $\left\{ \begin{array}{c} \text{categories} \\ \text{dependencies} \end{array} \right\}$ are of particular interest?

2. model selection
   - based on available and feasible data mining techniques
   - Which kind of information can be derived from the available data?

Data-Mining as a Process

3. data collection
   - designed experiment:
     - data can be generated on demand
     - sampling distribution is known
   - observational approach:
     - random data collection
     - sampling distribution is unknown or implicit in the data collection procedure
   - data collection affects the outcome

Data-Mining as a Process

4. data preparation
   - data cleansing
     - outliers: measurement, coding or recording errors
     - outliers: abnormal but natural values
     - missing values
   - outlier treatment
     - only if outlier detection is not the goal
     - either removal as part of preprocessing
     - or application of techniques insensitive to outliers
   - data transformation
     - rescaling: making features comparable
     - dimensionality reduction: gaining efficiency
     - binning: map numerical values to qualitative classes

Data-Mining as a Process

5. training of model parameters
   - estimation of stochastic parameters
   - adjustment of threshold values
   - adjustment of weights
   - ...

   - training is usually an optimization problem

6. evaluation and interpretation of results
   - simple models ...
     - ... are better trainable
     - ... are better interpretable
     - ... but less accurate

   - summarization
   - visualization

CRISP-DM

- Cross industry standard process for data mining
- life-cycle model

1. Business understanding phase
   - analysis of objectives and requirements
   - problem definition
   - initial strategy development
2. Data understanding phase
   - data collection
   - exploratory data analysis
   - assessment of data quality
3. Data preparation phase
   - cleansing, transformation etc.

CRISP-DM

4. Modelling phase
   - selection of modelling techniques and tools
   - parameter tuning / optimization
   - data analysis

5. Evaluation Phase
   - evaluation of the model
   - comparison of the outcome to the initial objectives
   - deployment decision

6. Deployment phase
   - reporting
   - transfer to other application cases
   - if applicable: introduction into day-to-day business

Data Preprocessing

- What is Data-Mining?
- Data-Mining as a Process
- Data Preprocessing
- Data Mining Tasks

Data Preprocessing

- Data Types
- Metrics
- High-Dimensional Data Spaces
- Missing Data
- Outlier Detection
- Time-Dependent Data
- Dimensionality Reduction
- Number of Values Reduction
- Sample Size Reduction

Data Types

Dienstag 24. April 2007, 09:00 Uhr

Berlin (AP) Mit seiner rekordverdächtigen Trockenheit fällt der Frühling 2007 auch unter den Aspekten der globalen Erwärmung aus der Rolle. "Wir gehen eher davon aus, dass im Frühjahr etwas mehr Niederschlag ist", sagte Wolfgang Kusch, der Präsident des Deutschen Wetterdienstes, am Dienstag im ZDF-Morgenmagazin. Ein signifikanter Rückgang der Niederschläge werde dagegen für den Sommer erwartet. "Das zeigen auch unsere Statistiken jetzt schon."

Der Anstieg der Temperaturen lasse sich vor allem bei der langfristigen Wetterbeobachtung feststellen, sagte Kusch. Seit Beginn der Wetteraufzeichnungen seien die Temperaturen stets angestiegen. Seit 1901 habe sich die durchschnittliche Temperatur in Deutschland von knapp acht auf knapp neun Grad plus um 0,9 Grad erwärmt. "Das sind immerhin mehr als zehn Prozent", sagte Kusch. "Das sind ganz gravierende Veränderungen."

Von den zehn wärmsten Jahren dieser Periode lägen neun in der Zeit zwischen 1989 und jetzt. Der frühere Ausreißer war 1934. Der Anstieg habe sich beschleunigt. "Von 1990 bis 2000 war das absolut wärmste Jahrzehnt", sagte der Chef des Wetterdiensts. "Und dieses Jahrzehnt ist wieder auf Rekordkurs." Nicht nur die Mitteltemperatur nehme zu, auch die Extreme würden zunehmen, sagte er voraus. Gerade der Nordosten Deutschlands werde deutlich trockener.
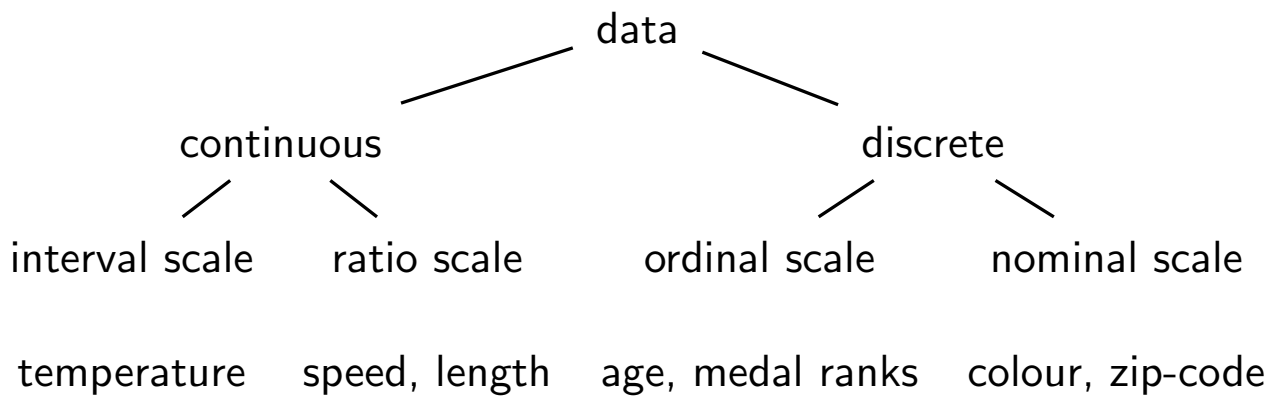
Data Types

- continuous (quantitative, metric)
    - measurement with (theoretically) infinite precision
    - interval scale
        - no problem-specific zero point
        - example: temperature
    - ratio scale
        - problem specific zero point (absence of the feature)
        - example: speed, length

Data Types

- discrete
    - ordinal scale (discretized)
        - problem-specific order relation
        - no distance relation
        - example: age, income, medal ranks
        - scale need not be linear
    - nominal scale (qualitative, categorical)
        - no problem-specific order relation
        - no distance relation
        - example: color, zip-code

Data Types

```
                              data
                 /                         \
          continuous                      discrete
          /        \                      /        \
interval scale   ratio scale    ordinal scale   nominal scale

temperature    speed, length    age, medal ranks    colour, zip-code
```

Data Types

- temporal characteristics
  - static
    - no change over time
  - dynamic (temporal)
    - change over time
  - periodic
    - no problem-specific order relation
    - distance relation
    - example: days of the week

Data Preprocessing

Metrics

- many data mining techniques are based on some notion of similarity/dissimilarity
- i.e. MINKOVSKIJ distance

$$d(\vec{x_1}, \vec{x_2}) = \left(\sum_{i=1}^{n} |x1i - x2i|^m \right)^{1/m}$$

- $m = 1$: Manhattan distance, city block distance
- $m = 2$: EUCLIDIAN distance
- $m = \infty$: max-distance, TCHEBYCHEV distance

Metrics

- special case: binary data $\rightarrow$ Hamming distance

$$|x1i - x2i| = \begin{cases} 0 & \text{if } x1i = x2i \\ 1 & \text{if } x1i \neq x2i \end{cases}$$

$$d(\vec{x_1}, \vec{x_2}) = \left(\sum_{i=1}^{n} |x1i - x2i|^m \right)^{1/m} = \left(\sum_{i=1}^{n} |x1i - x2i| \right)^{1/m}$$

$$= \left(n_{0/1} + n_{1/0} \right)^{1/m}$$

- root function is monotonic, m can be chosen as 1

$$d(\vec{x_1}, \vec{x_2}) = n_{0/1} + n_{1/0}$$

Metrics

- self identity: $\forall \vec{x} \, . \, d(\vec{x}, \vec{x}) = 0$
- positivity: $\forall \vec{x_1} \neq \vec{x_2} \, . \, d(\vec{x_1}, \vec{x_2}) > 0$
- symmetry: $\forall \vec{x_1}, \vec{x_2} \, . \, d(\vec{x_1}, \vec{x_2}) = d(\vec{x_2}, \vec{x_1})$
- triangle inequation: $\forall \vec{x_1}, \vec{x_2}, \vec{x_3} \, . \, d(\vec{x_1}, \vec{x_3}) = d(\vec{x_1}, \vec{x_2}) + d(\vec{x_2}, \vec{x_1})$

Metrics

- metrics for complex objects, i.e. strings
- string edit distance, LEVENSHTEIN metric
- minimal effort to transform a sequence into another one
- basic operations
  - substitution
  - insertion
  - deletion
- alignment: pairwise, order preserving mapping between the elements of the two strings
- alternative alignments with same distance possible

```
c   h   e   a   t
|  /|  /|  /|   |
c   o   a   s   t
```
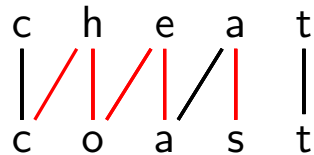
Metrics

- string edit distance is a non-deterministic, recursive function

$$d(x_{0:0}, y_{0:0}) = 0$$

$$d(x_{1:m}, y_{0:0}) = d(x_{0:0}, y_{1:m}) = m$$

$$d(x_{1:m}, y_{1:n}) = \min \begin{cases} d(x_{2:m}, y_{2:n}) + d(x_1, y_1) \\ d(x_{1:m}, y_{2:n}) + 1 \\ d(x_{2:m}, y_{1:n}) + 1 \end{cases}$$

Metrics

- finding the minimum distance is an optimization problem
  $\rightarrow$ dynamic programming

local distances

		c	h	e	a	t
	0	1	1	1	1	1
c	1	0	1	1	1	1
o	1	1	1	1	1	1
a	1	1	1	1	0	1
s	1	1	1	1	1	1
t	1	1	1	1	1	0

		c	h	e	a	t
	0	1	2	3	4	5
c	1	0	1	2	3	4
o	2	1	1	2	3	4
a	3	2	2	2	2	3

global distances

		c	h	e	a	t
	0	1	2	3	4	5
c	1					
o	2					
a	3					
s	4					
t	5					

Data Preprocessing

- Data Types
- Metrics
- **High-Dimensional Data Spaces**
- Missing data
- Outlier Detection
- Time-Dependent Data
- Dimensionality Reduction
- Number of Values Reduction
- Sample Size Reduction

High-Dimensional Data Spaces

- all high-dimensional data spaces are sparse
  - keeping the same data-point density in a space with more dimensions requires exponentially more data points
  - to enclose a prespecified portion of data points, an increasingly large part of the hypercube needs to be "encircled":

$$e(p) = p^{\frac{1}{n}}$$

portion p	dimensionality n	edge length $e(p)$
0.1	1	0.100
0.1	2	0.316
0.1	3	0.464
	...	
0.1	10	0.794

High-Dimensional Data Spaces

- paradoxes (cont.)
  - almost every point is closer to an edge of the cube than to another sample point
  - almost every point is an outlier

Data Preprocessing

Missing Data

- some data mining tools are insensitive to missing data

- ignore all incomplete tuples
    - might result in the loss of a substantial amount of data
- manual completion
- automatic completion
    - using a global constant
    - using the global mean value
    - using a class-dependent mean value
    - use a predictive model, e.g. based on a correlation with other features

Data Preprocessing

Outlier Detection

- not applicable if the application is aimed at outlier detection
  e.g. fraudulent credit card transactions
- the task: find k out of n tuples, which are
  - considerably dissimilar
  - exceptional
  - inconsistent with the remaining data

Outlier Detection

1. manual detection supported by visualization tools
   - only for low dimensional data
2. statistical methods
   - threshold for the variance, e.g. two times variance
   - only applicable if the distribution is known
3. using domain knowledge
   - value restrictions, e.g. age > 0
4. distance-based detection
   - applicable also for multi-dimensional data
   - a sample is an outlier if it has not enough neighbors

Example

- sample data

$$S = [s_1, ..., s_7] = [(2, 4), (3, 2), (1, 1), (4, 3), (1, 6), (5, 3), (4, 2)]$$

- distance matrix

	s_1	s_2	s_3	s_4	s_5	s_6	s_7
s_1		2.236	3.162	2.236	2.236	3.162	2.828
s_2	2.236		2.236	1.414	4.472	2.236	1.000
s_3	3.162	2.236		3.605	5.000	4.472	3.162
s_4	2.236	1.414	3.605		4.242	1.000	1.000
s_5	2.236	4.472	5.000	4.242		5.000	5.000
s_6	3.162	2.236	4.472	1.000	5.000		1.414
s_7	2.828	1.000	3.162	1.000	5.000	1.414	

Example

- neighborhood: $d \leq \theta = 3$

	s_1	s_2	s_3	s_4	s_5	s_6	s_7
s_1		2.236	3.162	2.236	2.236	3.162	2.828
s_2	2.236		2.236	1.414	4.472	2.236	1.000
s_3	3.162	2.236		3.605	5.000	4.472	3.162
s_4	2.236	1.414	3.605		4.242	1.000	1.000
s_5	2.236	4.472	5.000	4.242		5.000	5.000
s_6	3.162	2.236	4.472	1.000	5.000		1.414
s_7	2.828	1.000	3.162	1.000	5.000	1.414	

- number of points in the neighborhood

sample	s_1	s_2	s_3	s_4	s_5	s_6	s_7
	4	5	1	4	1	3	4

Example

Outlier Detection

5. deviation-based methods
    - measure the dissimilarity of a data set (e.g. variance)
    - determine the smallest subset of data that if removed results in the largest reduction of dissimilarity
    - combinatorics of subset selection $\rightarrow$ extremely expensive

Data Preprocessing

- Data Types
- Metrics
- High-Dimensional Data Spaces
- Missing data
- Outlier Detection
- **Time-Dependent Data**
- Dimensionality Reduction
- Number of Values Reduction
- Sample Size Reduction

Time-Dependent Data

- time series: $[v_1, v_2, v_3, ..., v_n]$
- typical task: predict the next value v_{n+1}
- predictions are based on a window of k preceding values
  $$v_{n+1} = f([v_{n-k+1}, v_{n-k+2}, ..., v_n])$$
- data can be rearranged (here $k = 5$):

sample	window					next value
1	v_1	v_2	v_3	v_4	v_5	v_6
2	v_2	v_3	v_4	v_5	v_6	v_7
3	v_3	v_4	v_5	v_6	v_7	v_8
4	v_4	v_5	v_6	v_7	v_8	v_9

Time-Dependent Data

- postponed prediction: predict a value further in the future
- offset o (here $k = 5$, $o = 3$):

sample	window					next value
1	v_1	v_2	v_3	v_4	v_5	v_8
2	v_2	v_3	v_4	v_5	v_6	v_9
3	v_3	v_4	v_5	v_6	v_7	v_{10}
4	v_4	v_5	v_6	v_7	v_8	v_{11}

Time-Dependent Data

- smoothing time-series data: moving average

$$MA(i, k) = \frac{1}{k} \sum_{j=i-k+1}^{i} v_j$$

- different weighting schemas
  - exponential moving average: putting more emphasis on the most recent values

$$EMA(i, k) = \begin{cases} v_i & \text{if } k = 1 \\ p \cdot v_i + (1 - p) \cdot EMA(i - 1, k - 1) & \text{else} \end{cases}$$

  - p controls the influence of more recent data ($p = 0...1$)

Time-Dependent Data

- comparative features: emphasize changes in the data, e.g.

$$v_i - MA(i, k)$$

$$MA(i, k) - MA(i - l, k)$$

$$\frac{v_i}{MA(i, k)}$$

Time-Dependent Data

- survival data: How long does in take until an event occurs?

- applications
    - medicine: outbreak of a desease, death
    - industry: component failure

- problems
    - censored observation:
      the event does not always occur until the end of a (limited)
      observation period
    - input values are time-dependent:
      the object under observation changes

Data Preprocessing

- Data Types
- Metrics
- High-Dimensional Data Spaces
- Missing Data
- Outlier Detection
- Time-Dependent Data
- **Dimensionality Reduction**
- Number of Values Reduction
- Sample Size Reduction

Dimensionality Reduction

- Which data can be discarded without sacrificing the quality of the data mining results?
- too many dimensions
    - mining results degrade (insufficient data)
    - resulting model is incomprehensible
    - problem becomes untractable
- too few dimensions
    - data dependencies are lost
    - mining results degrade (limited expressiveness)

Dimensionality reduction

- feature selection approaches
    - feature ranking
    - minimum subset selection
- feature composition approaches
    - principal component analysis

Feature Ranking

- needed: evaluation measure based on
    - accuracy of data
    - consistency
    - information content
    - distances between samples
    - statistical dependencies between features

Feature Ranking

- idea: discard features which contribute least to the overall entropy of the sample set $S = \{\vec{x_1}, \ldots \vec{x_n}\}$

- entropy of a sample set is the sum over the entropy of all sample pairs

$$E(S) = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} E(\vec{x_i}, \vec{x_j})$$

- sample pair is a coin-tossing experiment
  similar − dissimilar: $p_{ij} = p(\vec{x_i} \sim \vec{x_j})$

$$E = -\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} p_{ij} \cdot \log p_{ij} + (1 - p_{ij}) \cdot \log(1 - p_{ij})$$

Feature Ranking

- numerical data:
  probability of being similar is estimated by the inverse of the
  distance D_{ij} between the two samples

$$p_{ij} = e^{-\alpha \cdot D_{ij}/D}$$

D: average distance among samples in the data set
D_{ij}: distance, e.g. normalized Euclidian distance

$$D_{ij} = \sqrt{\sum_{k=1}^{n} \left(\frac{v_{ki} - v_{kj}}{\max_m(v_{km}) - \min_m(v_{km})} \right)^2}$$

$\rightarrow$ avoids the dominance of dimensions
   with great variance over low-variance dimensions

Feature Ranking

- nominal data: Hamming distance

$$p_{ij} = \frac{1}{n} \sum_{k=1}^{n} |x_{ik} = x_{jk}|$$

$$|x_{ik} = x_{jk}| = \begin{cases} 1 & \text{if } x_{ik} = x_{jk} \\ 0 & \text{else} \end{cases}$$

Feature Ranking

- algorithm
    - start with the full set of features F
    - S_F is the sample S using the feature set F
    - $\forall f \in F \,.\, F' = F - \{f\}$

    $$f_{min} = \arg \min_f E(S_F) - E(S_{F'})$$

    rank f_{min} lowest

    - continue with $F' = F - \{f_{min}\}$ until $F = \emptyset$
- features with lower ranks can be discarded

Minimum Subset Selection

- idea: select a feature set which maximises the class distinction capability
    - requires to assume a distribution: e.g. normal distribution
- class distinction defined as

$$cd_{A,B} = \frac{|\mu_A - \mu_B|}{\sqrt{\frac{\sigma_A}{n_A} + \frac{\sigma_B}{n_B}}}$$

i.e. class distinction is maximal if

- means are far away and
- variance is low
- despite many samples

Minimum Subset Selection

- simple case: features are considered as independent
- compute all pairwise class distinction values
- retain a feature, if it is relevant for any of the pairwise comparisons

Example

- sample data set (assumption: normal distribution)

X	Y	C	X	Y	C
0.3	0.7	A	0.2	0.9	B
0.6	0.6	A	0.7	0.7	B
0.5	0.5	A	0.4	0.9	B

Example

- data sets separated by class and dimension
  $X_A = \{0.3, 0.6, 0.5\}$, $X_B = \{0.2, 0.7, 0.4\}$,
  $Y_A = \{0.7, 0.6, 0.5\}$, $Y_B = \{0.9, 0.7, 0.9\}$

$$cd_{X_A, X_B} = \frac{|\mu_{X_A} - \mu_{X_B}|}{\sqrt{\frac{\sigma_{X_A}}{n_{X_A}} + \frac{\sigma_{X_B}}{n_{X_B}}}} = \frac{|0.4667 - 0.4333|}{0.3671} = 0.0908$$

$$cd_{Y_A, Y_B} = \frac{|\mu_{Y_A} - \mu_{Y_B}|}{\sqrt{\frac{\sigma_{Y_A}}{n_{Y_A}} + \frac{\sigma_{Y_B}}{n_{Y_B}}}} = \frac{|0.6 - 0.8333|}{0.268} = 0.8706$$

- $\rightarrow$ discard X

Minimum Subset Selection

- generalization to non-independent features
- μ vector of means, C covariance matrix
- $C_{i,j}$ correlation between each pair of features

$$C_{i,j} = \frac{1}{n} \sum_{k=1}^{n} (x_{ki} - \mu_i) \cdot (x_{kj} - \mu_j)$$

$$C_{i,j} = \frac{1}{n-1} \sum_{k=1}^{n} (x_{ki} - \bar{x}_i) \cdot (x_{kj} - \bar{x}_j)$$

Minimum Subset Selection

- class distinction for non-independent features

$$cd_{AB} = (\mu_A - \mu_B)(C_A + C_B)^{-1}(\mu_A - \mu_B)^T$$

- compute $cd_{A,B}$ for all subsets of features
- select the subset with a comparatively small loss of discriminating power
- 2^n combinations: full search is infeasible $\rightarrow$ heuristic search
  - examining only promising subsets
  - using a simplified distance metric
  - select features only on a subset of data

Principal Component Analysis

- KARHUNEN-LOEWE transformation
- features are composed into a new feature set with reduced dimensionality: construction of a new feature space
- new features are computed as a linear combination of old ones

$$v' = \sum_{i=1}^{n} w_i \cdot v_i$$

- up to n transformations are generated
- each vector of weights is called a principal component

Principal Component Analysis

- principal components are ranked according to their usefulness → components with lowest rank can be discarded
- idea: high information content corresponds to high variance
- transform a set of input dimensions x into a single new dimension y

$$\vec{y} = A \cdot \vec{x}^T$$

- choose A so that $\vec{y}$ has the largest possible variance
    - $\vec{y}$ is a new axis in the direction of maximum variance

Principal Component Analysis
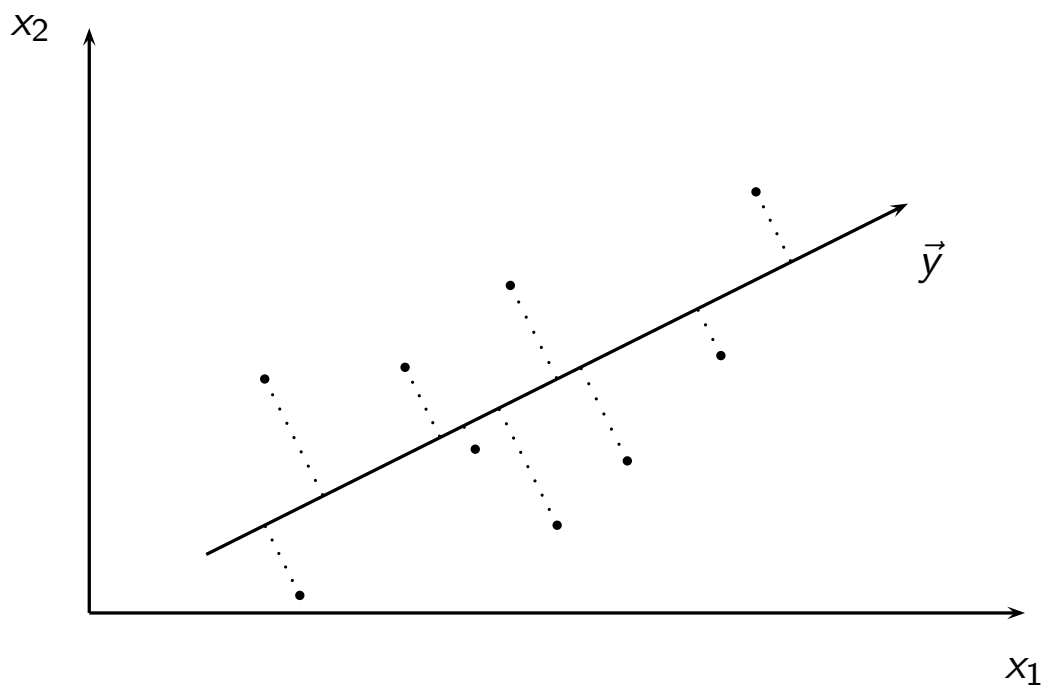
Principal Component Analysis

- computation of the covariance matrix C_{ij}

$$C = [C_{i,j}] = \left[\frac{1}{n-1} \sum_{k=1}^{n} (x_{ki} - \bar{x}_i) \cdot (x_{kj} - \bar{x}_j) \right]$$

- computation of the eigenvalues $\lambda_1, \lambda_2, ..., \lambda_n$ of C
  - eigenvalue i is equal to the variance of the i-th component
- computation of the eigenvectors $\vec{e_1}, \vec{e_2}, ..., \vec{e_n}$ corresponding to $\lambda_1, \lambda_2, ..., \lambda_n$

Principal Component Analysis

- decomposition of the covariance matrix C_{ij}

$$C = A \times L \times A^T$$
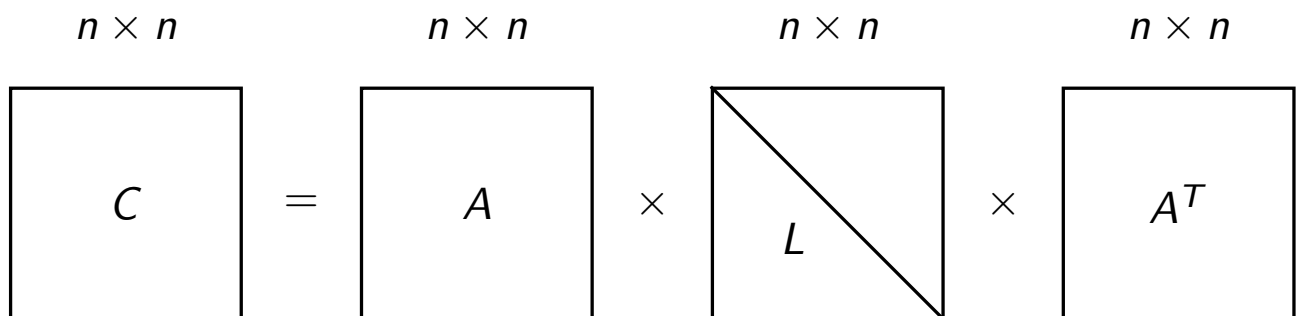
- L is a diagonal matrix containing the eigenvalues $\lambda_1, \lambda_2, ..., \lambda_n$ of C
- A contains the eigenvectors $\vec{e_1}, \vec{e_2}, ..., \vec{e_n}$ corresponding to $\lambda_1, \lambda_2, ..., \lambda_n$ as columns

Principal Component Analysis

- permutation of the matrices to order the elements of L in decreasing order
- selection of the first k of n eigenvectors
  until a threshold for

$$\left(\sum_{i=1}^{k} \lambda_i\right) / \left(\sum_{i=1}^{n} \lambda_i\right)$$
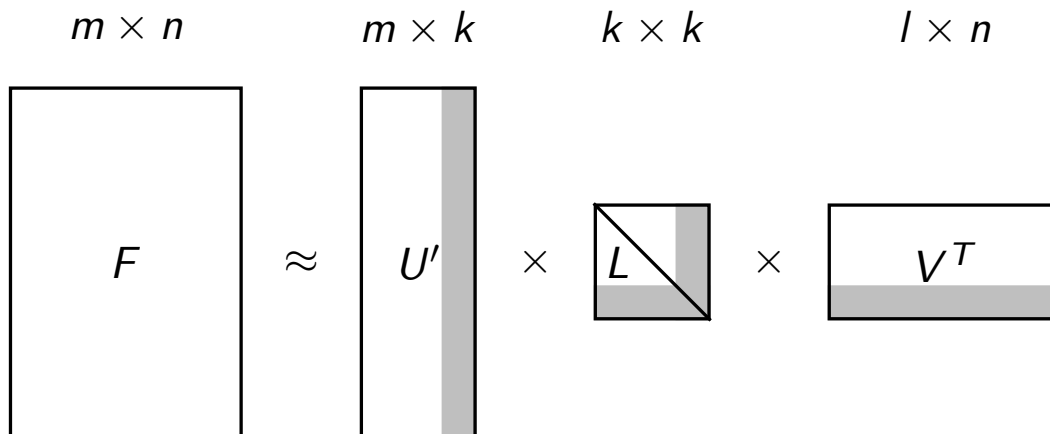
is reached
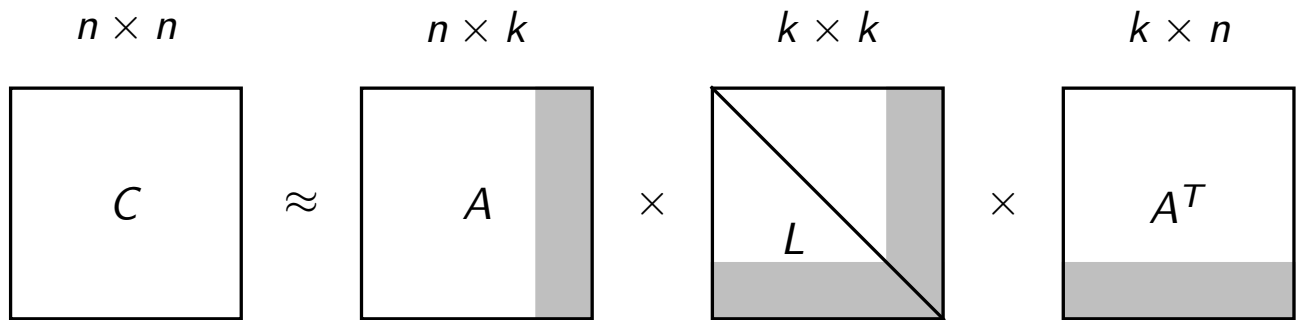
Latent Semantic Analysis

- singular value decomposition
- decomposition of the feature matrix

$$F = U \times L \times V^T$$
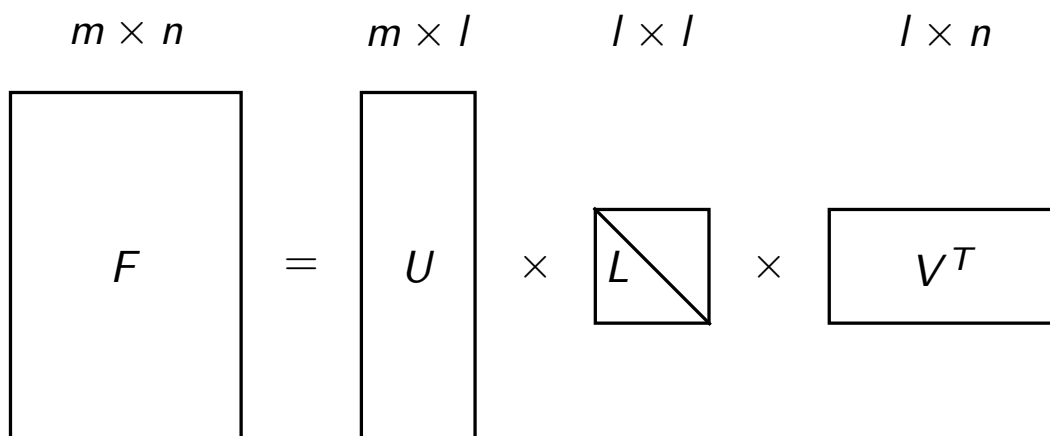
Latent Semantic Analysis

- selection of the most relevant features

$$
\underset{m \times n}{F} \approx \underset{m \times k}{U'} \times \underset{k \times k}{L} \times \underset{l \times n}{V^T}
$$

- elements of L can be interpreted as the underlying abstract concepts elements of the feature matrix are mapped to
- mainly used for document retrieval

Data Preprocessing

- Data Types
- Metrics
- High-Dimensional Data Spaces
- Missing data
- Outlier Detection
- Time-Dependent Data
- Dimensionality Reduction
- **Number of Values Reduction**
- Sample Size Reduction

Number of Values Reduction

- one dimensional: feature discretization (binning)
  $\rightarrow$ mapping values to intervals
    - value exchange techniques
    - merging techniques
- multi-dimensional: clustering of feature vectors
    - splitting techniques

Value Exchange

- given k, the number of bins, distribute the values to minimize the average distance to the mean of a bin

- algorithm
    - sort all values for a given feature
    - assign approximately equal values to each bin
    - move a border element v_i to the neighboring bin if that reduces the global average distance
    - continue until no further improvement can be achieved
- gradient descent search: optimum not guaranteed

Merging Techniques

- merging of bins: χ^2 merger
  - merge adjacent intervals if they are similar with respect to a classification task
  - independence of the class assignment from the intervals is measured by the χ^2 test
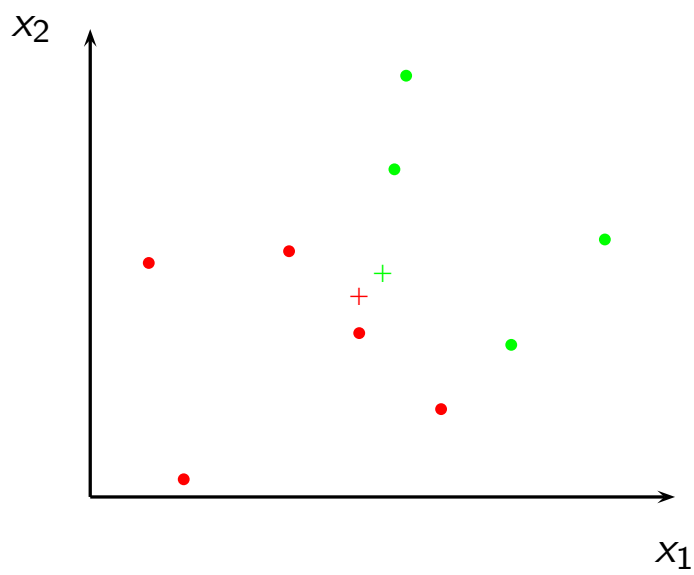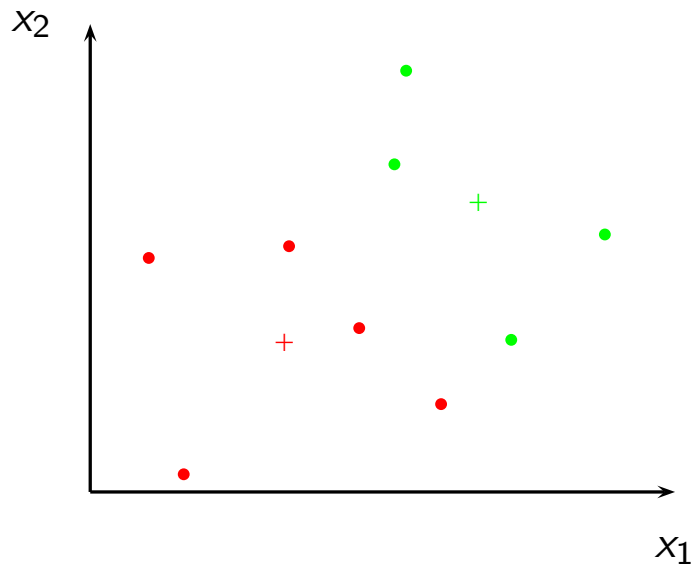 - adjacent intervals can be merged if their χ^2 value is below a threshold

Splitting Techniques

- splitting of bins: vector quantization
  - compute the mean of all data (centroid)
  - split the centroid(s)
  - assign data points to the nearest centroid (bin)
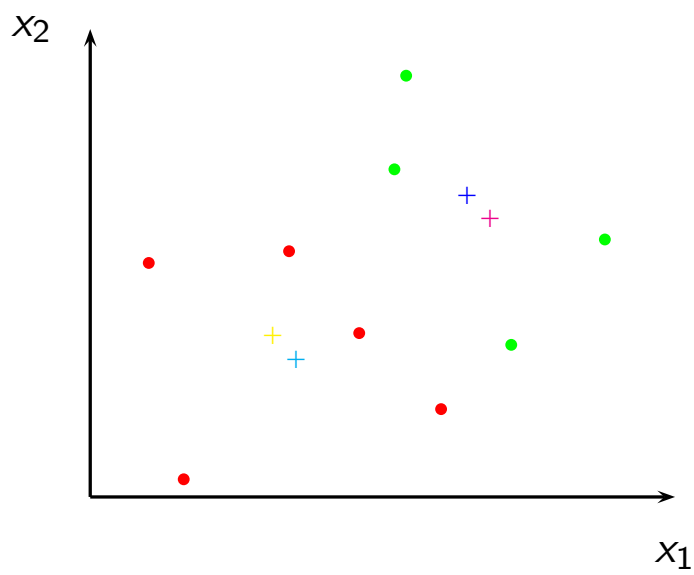 - continue until enough bins have been generated

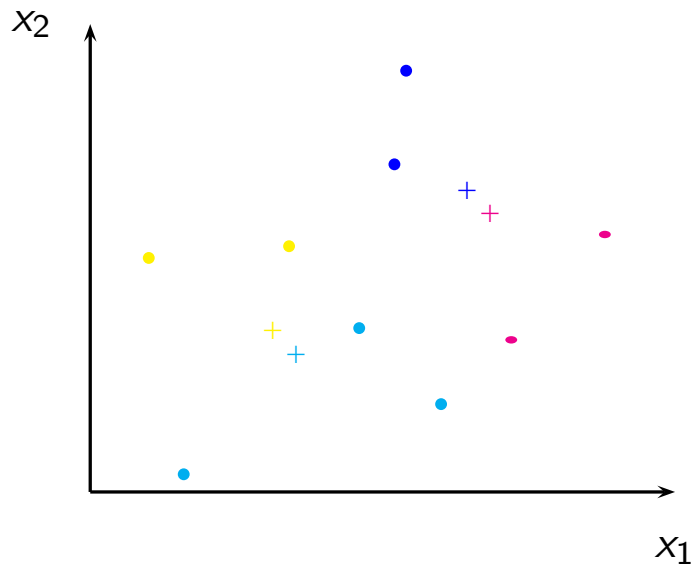# Number of Values Reduction
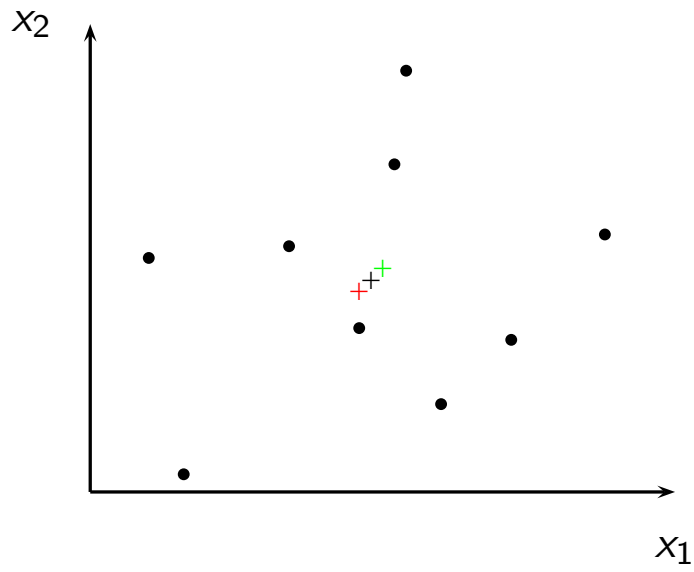
Number of Values Reduction

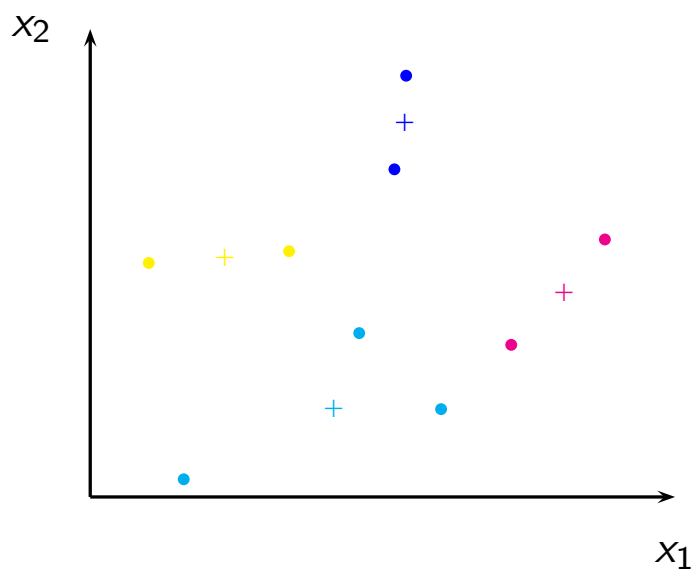# Number of Values Reduction

Number of Values Reduction

Number of Values Reduction

Number of Values Reduction

Number of Values Reduction

- "one-dimensional" vector quantisation:

  - mapping continuous values to discrete ones

- multi-dimensional vector quantisation:

  - dimensionality + number of values reduction

- vector quantisation is a special case of unsupervised data clustering:

  - learning without teacher

Data Preprocessing

- Data Types
- Metrics
- High-Dimensional Data Spaces
- Missing data
- Outlier Detection
- Time-Dependent Data
- Dimensionality Reduction
- Number of Values Reduction
- Sample Size Reduction

Sample Size Reduction

- systematic sampling, e.g. every other tuple
- random sampling
    - with replacement
    - without replacement
    - incremental: choose increasingly large subsets of the data
    - averaged: combine results from multiple subsets of data