

Intelligent Systems

Chapter 7: Representation and Features

Winter Term 2019 / 2020

Prof. Dr.-Ing. habil. Sven Tomforde
Institute of Computer Science / Intelligent Systems group

Content

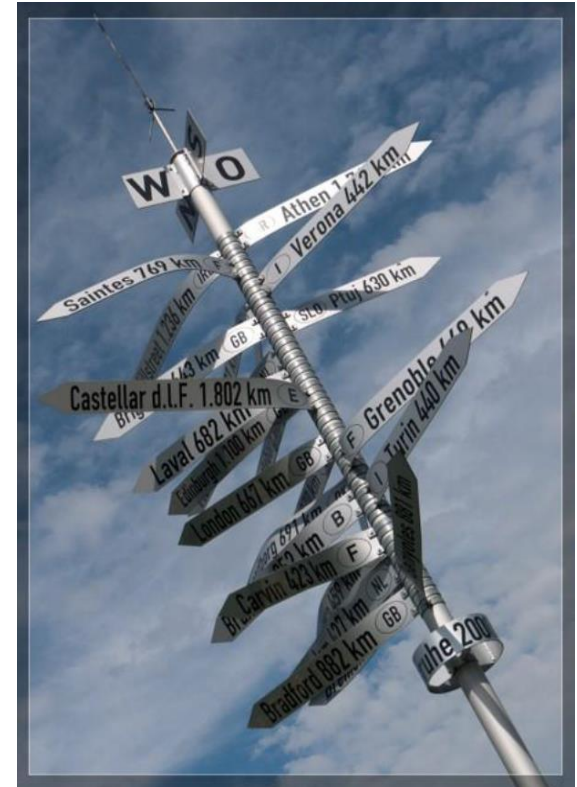
- Feature extraction
- Time-independent features
- Time-dependent features
- Feature selection
- Taxonomy of feature selection
- Feature selection algorithms
- Feature transformation
- Principle Component Analysis
- Conclusion
- Further readings

Goals

Students should be able to:

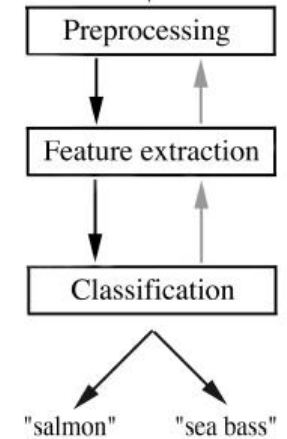
- motivate and define the three steps feature extraction, feature selection, and feature transformation.
- introduce different kinds of features.
- compare different algorithms and their applicability.
- distinguish filters and wrappers.
- apply accuracy, information, distance, dependency, and consistency measures.

- Feature extraction
- Time-independent features
- Time-dependent features
- Feature selection
- Taxonomy of feature selection
- Feature selection algorithms
- Feature transformation
- Principle Component Analysis
- Conclusion
- Further readings



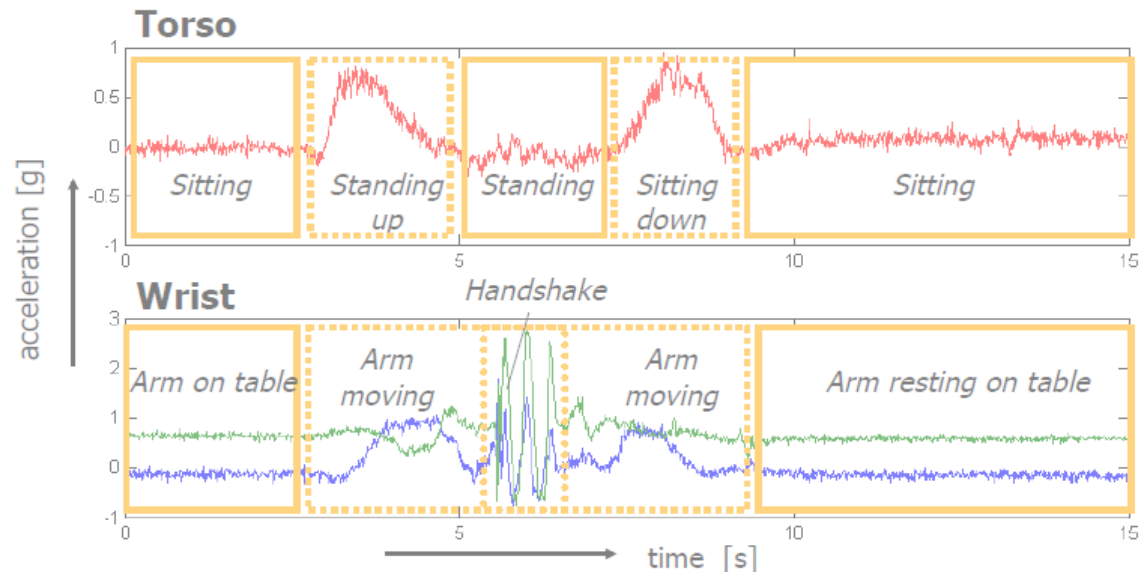
Feature extraction

- Example:
 - Classification of fish
 - Sensor: Camera
- Output signal: grey scale images
- Problems with raw data
 - High-dimensional (camera resolution e.g. 640x480)
Vector of length 307,200)
 - Contains a lot of unnecessary (i.e. not problem-relevant) information



Feature extraction

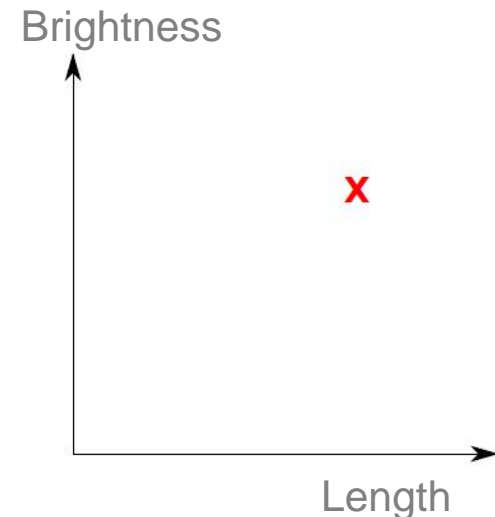
- Example: Activity detection
 - Sensors: Accelerometers
 - Output signals: Time series
- Problems with raw data
 - High-dimensional
→ Vector length = length of activities
 - Variable length



Feature extraction

Feature Extraction: Goals

- Abstract representation of input patterns by features
- Transformation of input data into the feature space (problem-specific representation)
 - Reduction of dimensionality to essential information (effort, robustness)
 - Elimination of "misleading" or irrelevant information
 - Constant dimensionality
- "Simplification" of the decision space



Feature extraction (2)

Features: Requirements

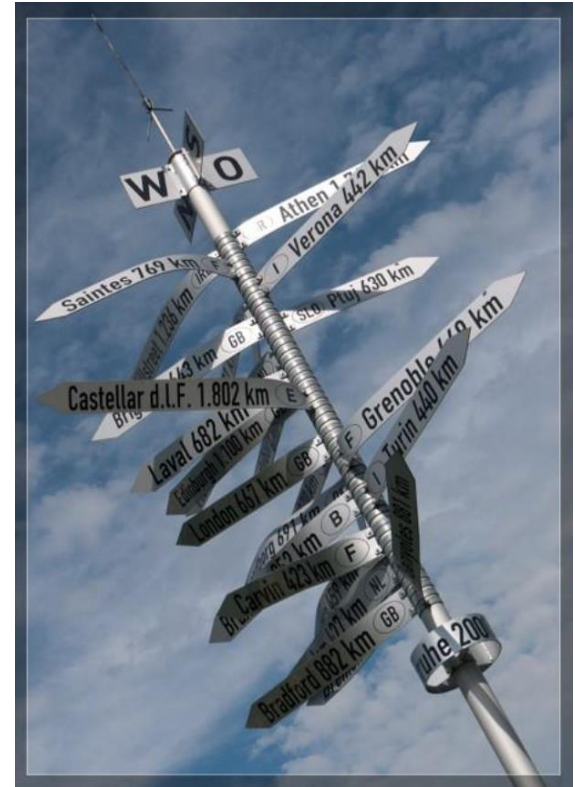
- Continue to describe input patterns in such a way that the actual task can still be performed (distinguishability).
 - For **classification**: patterns of the same class very similar, for different classes very dissimilar
 - For **clustering**: analogous for "natural" groups within data
- Invariance against problem-specific irrelevant transformations
 - Translation
 - Rotation
 - Scaling
 - ...
- E.g. characteristics describing properties such as shape, colour, texture, edges, curvature

Feature extraction (3)

Which features for which task?

- Features are derived from the input signals based on an analysis of the task.
 - Understanding the problem is critical!
 - Decision often based on experience and intuition
- Highly application-specific
- Possible features result from the type of the input signals
 - Time- and space-independent data
 - The input signal was recorded at a certain time and is only relevant for classification at this time.
 - Time- and space-dependent data
 - Continuously measured data where development over time is essential
 - Images (space-dependent)

- Feature extraction
- **Time-independent features**
- Time-dependent features
- Feature selection
- Taxonomy of feature selection
- Feature selection algorithms
- Feature transformation
- Principle Component Analysis
- Conclusion
- Further readings



Time-independent features

Time-independent features

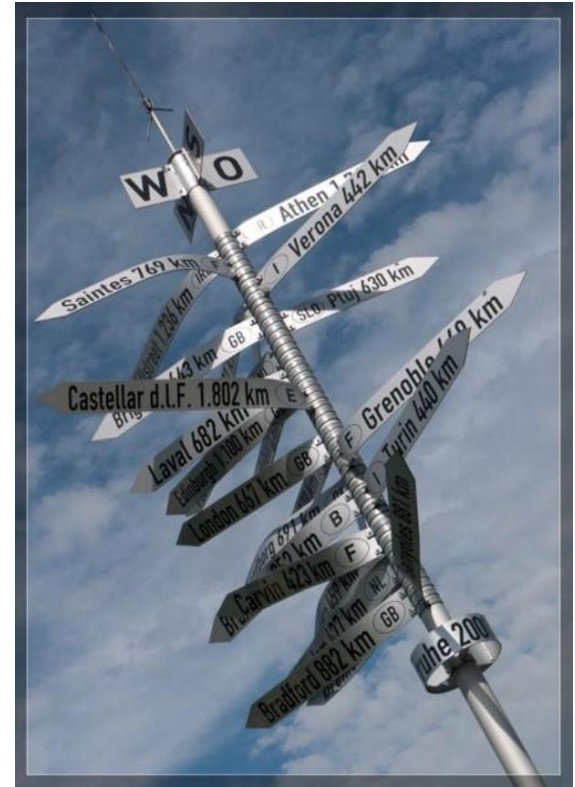
- Example: Absolute value of a vector as feature
- Acceleration sensor: Acceleration (a_x, a_y, a_z) depends on the orientation of the smartphone on the body.
→ It is not possible to separate the motion part from the gravity part.
- Also the change of the absolute value $|a| = \sqrt{a_x^2 + a_y^2 + a_z^2}$ contains characteristics of most movements
→ The absolute value is invariant to the orientation of the smartphone!
- Means: Use the absolute value as feature instead of the raw data

Time-independent features (2)

Time-independent data

- Further possibilities to extract features from raw data:
- Coordinate transformation
 - Selection of the reference system (e.g. absolute vs. relative velocity)
 - Use of polar coordinates
- General functions of subsets of the sensor signals used
 - Component ratios
 - Linear combinations of the components
 - Logical statements about the components

- Feature extraction
- Time-independent features
- **Time-dependent features**
- Feature selection
- Taxonomy of feature selection
- Feature selection algorithms
- Feature transformation
- Principle Component Analysis
- Conclusion
- Further readings



Time-dependent features

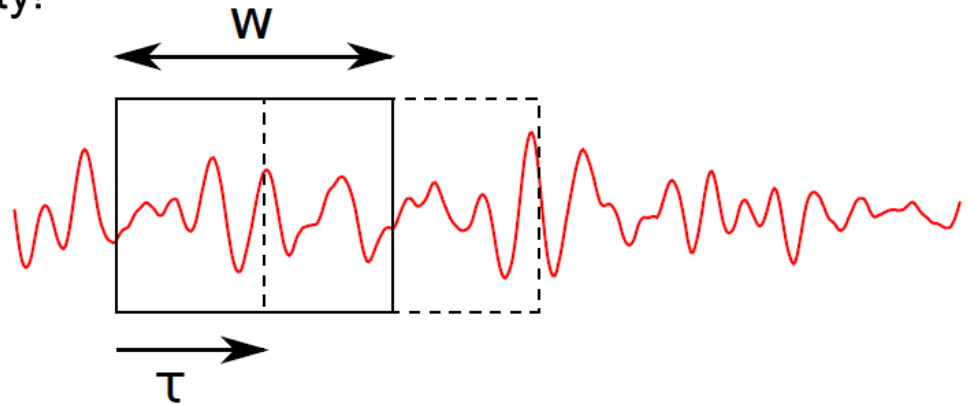
Processing time series data

- Various possibilities:
 - Consider the entire time series
 - Bad for "online" processing
 - Problematic if individual subsections (e.g. activities) are to be classified
 - Processing parts of the time series with window method
 - Sliding Window
 - Growing Window

Time-dependent features (2)

Sliding window

- Features are calculated in a fixed length window that moves across the input signal, each of which **affects the entire window**.
 - Hiding the remaining information
 - Reduction of dimensionality!

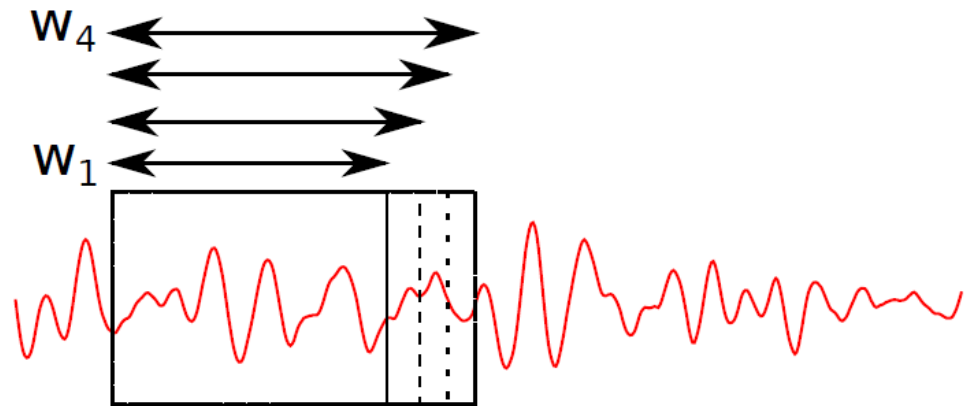


- Two degrees of freedom
 - Window size w : must be adapted to the typical duration of the phenomena to be found
 - Time offset τ : one or more time steps

Time-dependent features (3)

Growing window

- In a **window of increasing length**, characteristics are calculated that affect the entire window.
 - Hiding the remaining information
 - Reduction of dimensionality!
 - Length of the window grows: $w_1 < w_2 < w_3 < w_4$
 - Starting position is fixed.
- After detection: window is shifted (time steps) and reduced, procedure repeats itself
- Phenomena of variable length can be easily detected



Time-dependent features (3)

Statistical features if time series

→ Determined based on the entire time series or in windows

- Mean

$$\mu = \frac{1}{N} \sum_{i=1}^N x_i$$

- RMS (root mean square), “effective value“

$$\text{rms} = \sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2}$$

- Variance

$$\sigma^2 = \frac{1}{N-1} \sum_{i=1}^N (x_i - \mu)^2$$

- Standard deviation

$$\sigma = \sqrt{\sigma^2}$$

Time-dependent features (4)

Statistical features if time series

→ Determined based on the entire time series or in windows

- **Minimum and maximum** values
 - Difference Max-Min (span)
- **Median** or **Mode**
- **Threshold** values (e.g. zero crossings)
 - Over/underflow yes/no?
 - How often?
- Average of the absolute values of the **first differences**:

$$\delta = \frac{1}{N-1} \sum_{i=1}^{N-1} |x_{i+1} - x_i|$$

- Mean value of the absolute values of the **second differences**:

$$\gamma = \frac{1}{N-2} \sum_{i=1}^{N-2} |x_{i+2} - x_i|$$

- If necessary normalised with σ (standard deviation)

Time-dependent features (5)

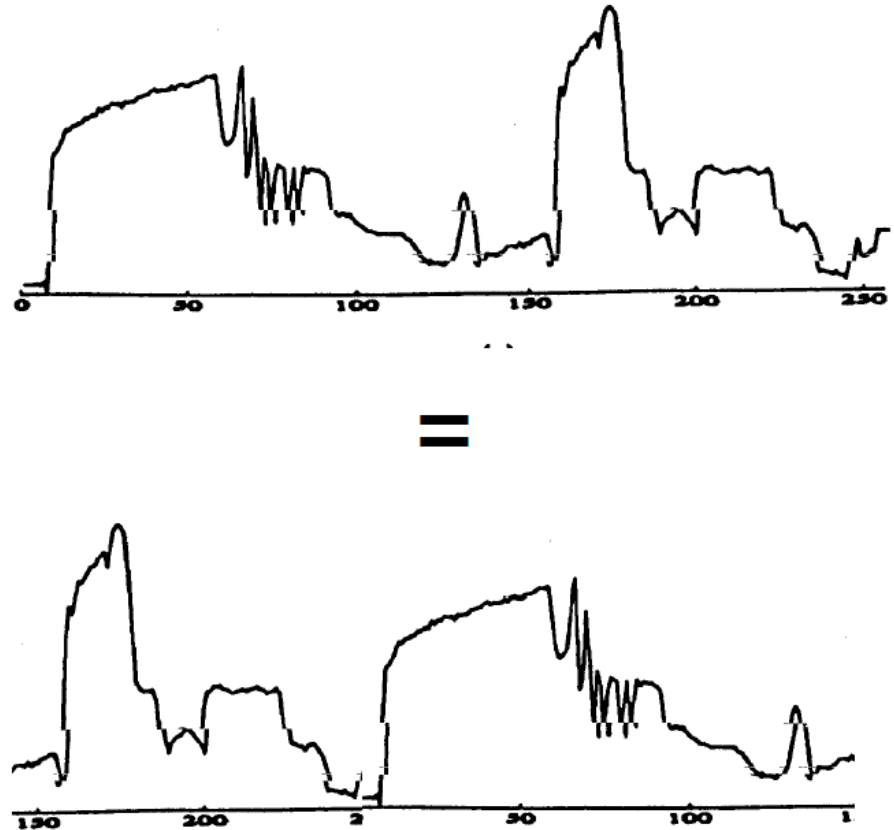
Statistical features: Assessment

- Advantages:
 - All time series are mapped to vectors of the same length
 - (Relative) insensitive to typical disturbances (outliers, noise, ...)
- Disadvantages:
 - Sliding window: Choice of window size critical and problem-specific
 - Too small: Relevant phenomena are not recorded
 - Too big: class changes are overlooked
 - Growing window:
 - The longer the window becomes, the more phenomena are smoothed out and possibly no longer detected.

Time-dependent features (6)

Further disadvantage:

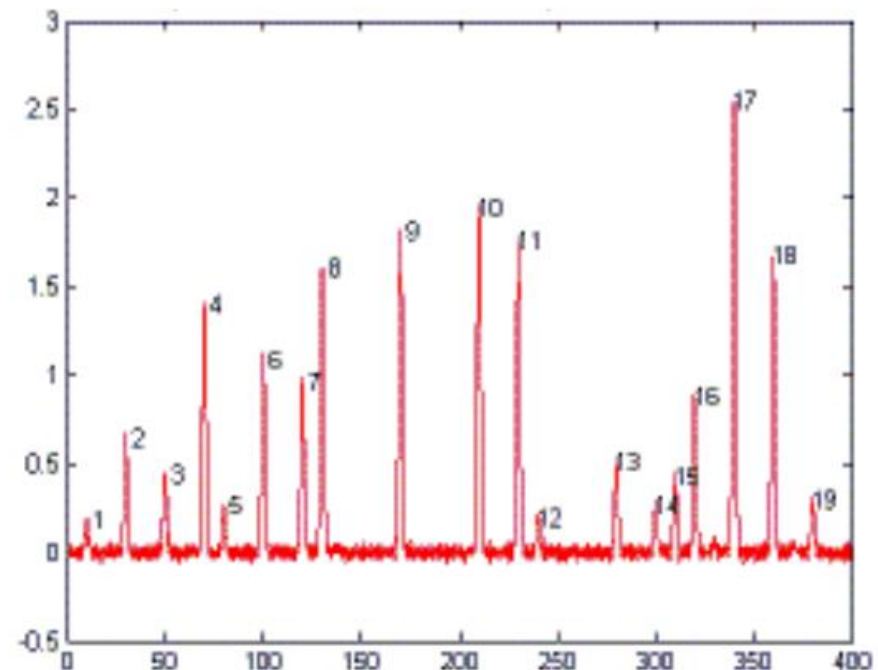
- Features are cumulated over the entire window length
- Temporal structure / behaviour is not or only insufficiently covered



Dynamic features

Description by peaks (signal peaks, deflection)

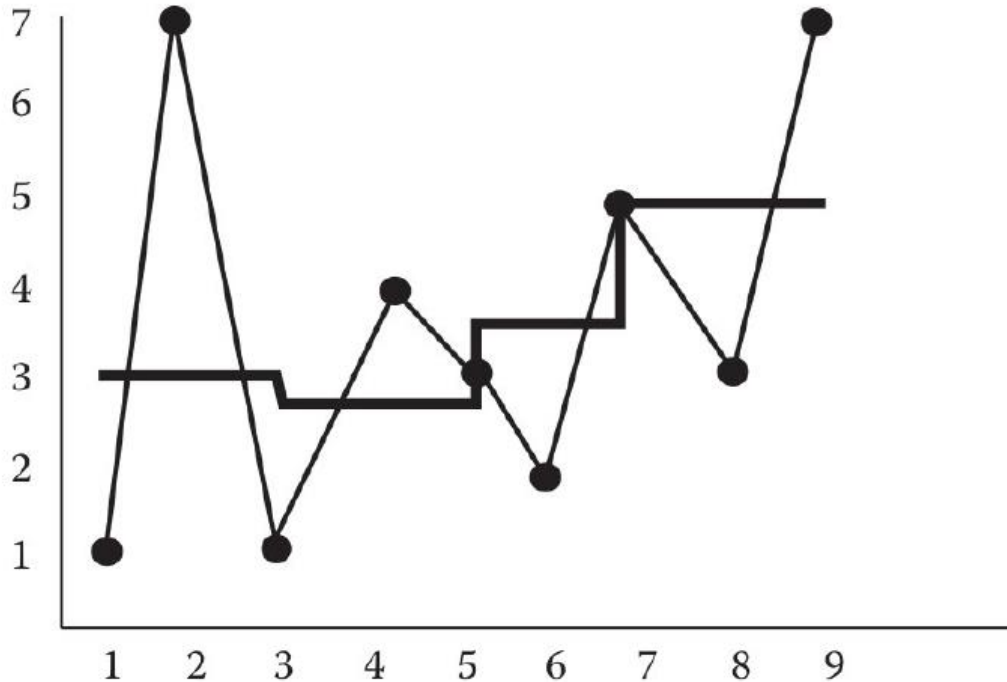
- Possible parameters
 - Position of the peaks
 - Height of peaks
 - Width of peaks
- For a constant number of dimensions:
 - n highest peaks of the window



Dynamic features (2)

Piecewise Aggregate Approximation (PAA)

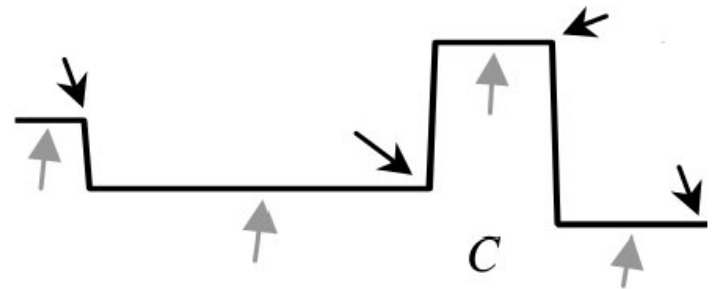
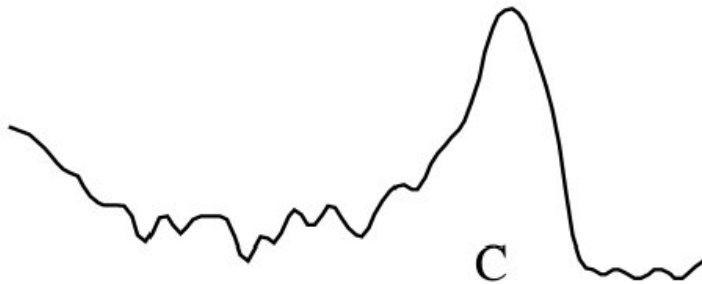
- Time series is divided into sections of equal length and these are replaced by a constant value (average) resulting from the sections.



Dynamic features (3)

Adaptive Piecewise Aggregate Approximation (APAA)

- Main difference to PAA: variable length of sections
- Basic idea: Adaptive to local details of a time series, i.e. phases with more “dynamics” are divided into smaller sections, phases without significant information into longer sections.

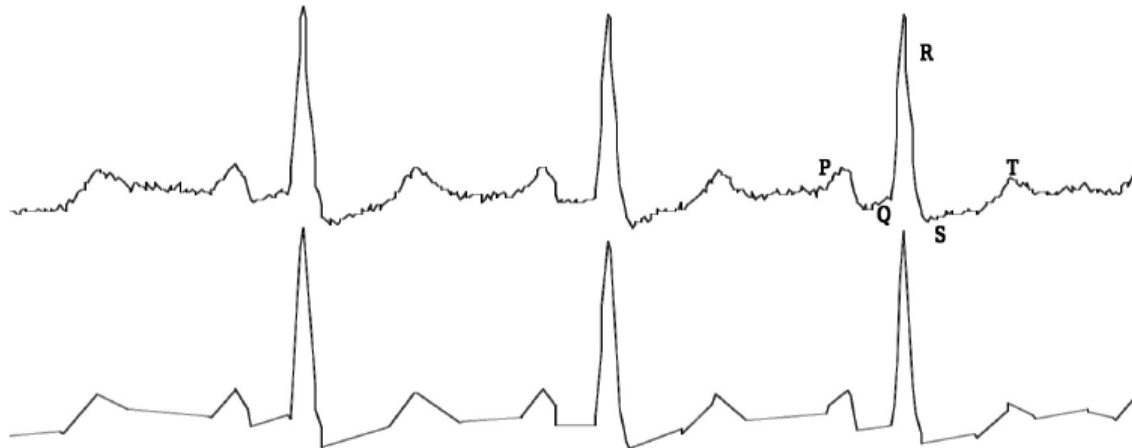


- Finding the optimal sections not trivial
 - Optimal with respect to an error measure that evaluates the difference between the approximation and the true signal
- Solution e.g. by means of heuristics or dynamic programming

Dynamic features (4)

Piecewise Linear Approximation (PLA)

- Basic idea:
 - Again divided into sections
 - Each section is represented by a straight line
 - Vector: length and slope



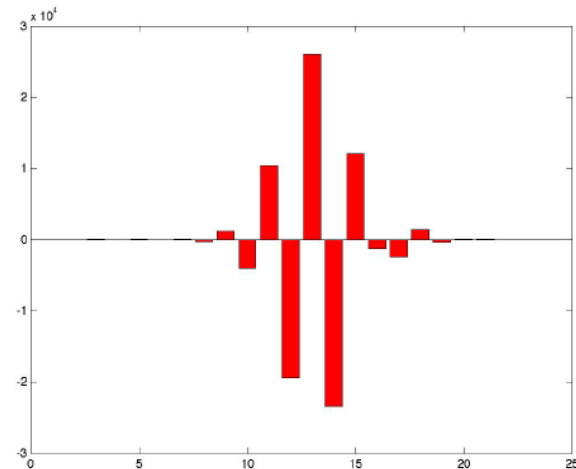
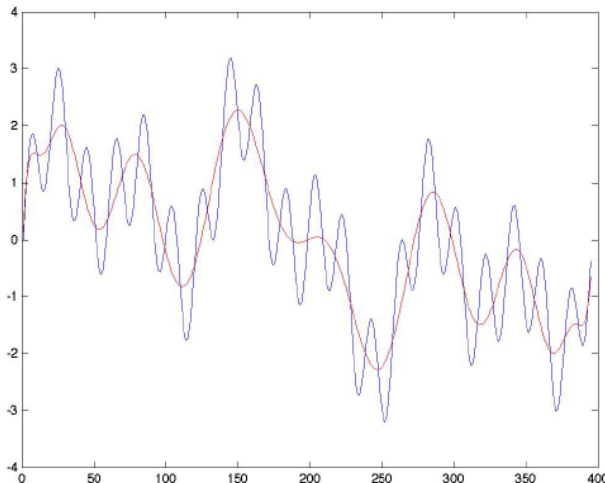
Dynamic features (5)

Polynomial approximation

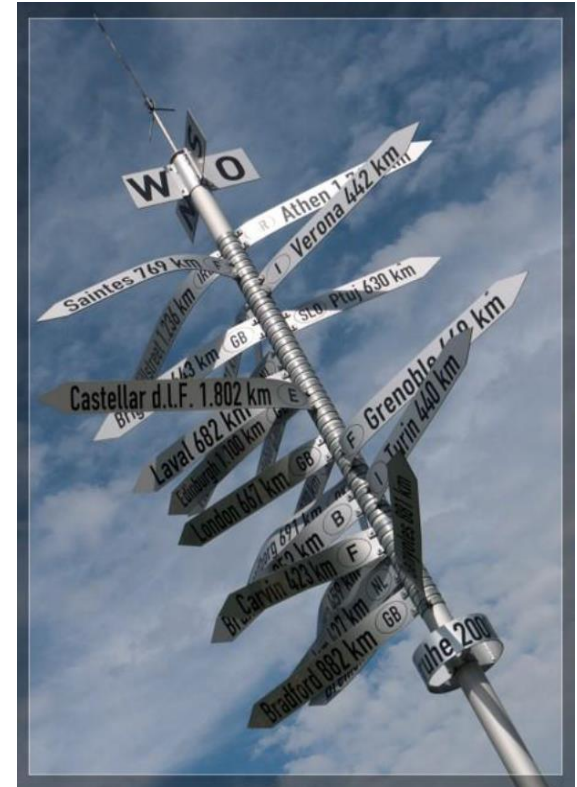
- Representation of a certain interval of a time series as polynomial of degree n :

$$p(x) = a_0 + a_1x^1 + a_2x^2 + \dots + a_nx^n$$

- Signal is reduced to the coefficients $a_0, a_1, a_2, \dots, a_n$.
 - Considers temporal behaviour
 - But: Still able to represent a long signal with only a few values



- Feature extraction
- Time-independent features
- Time-dependent features
- **Feature selection**
- Taxonomy of feature selection
- Feature selection algorithms
- Feature transformation
- Principle Component Analysis
- Conclusion
- Further readings



Feature selection

Task of feature selection:

- The task of feature selection is to select an optimal subset from the set of available features in a data set.
- Depending on the task, the set of "important" features (to be selected) should
 - offer highest accuracy (classification, regression, ...),
 - be minimum with a given minimum accuracy,
 - have low costs (monetary costs or calculation costs),
 - be better understandable,
 - ...
- Here: Focus on accuracy aspect for classification tasks

Feature selection (2)

Feature selection: Task

- Important (good) characteristics need to be distinguished from unimportant (bad) characteristics.
- Other and related tasks are:
 - Reduction of the sample number after feature selection
 - Establishment of a ranking of features (feature ranking)
 - Improvement of the visualisation of data
 - Construction of (meta-)features, e.g. by main axis transformation and selection of the most important main components (feature construction)

Feature selection (3)

What is the benefit of feature selection?

- Time savings due to smaller amounts of data (fewer features, possibly fewer patterns); the calculation effort for the subsequent ML algorithms (e.g. for classification) is reduced.
- Better data quality through elimination of unimportant (e.g. redundant or heavily noisy) information
- Improved generalisation performance due to smaller number of features
- Cost savings e.g. through the well directed use of relevant features (no need for unnecessary sensor technology) or in the case of time-consuming calculation of features (e.g. Signal processing)

Feature selection (4)

Definition (for classification tasks)

Definition: Feature Selection

Let F be a full set of features and $G \subseteq F$. In general, the goal of feature selection can be formalised as selecting a minimum subset G such that $P(C|G)$ is equal or as close as possible to $P(C|F)$, where $P(C|G)$ is the probability distribution of different classes given the feature values in G and $P(C|F)$ is the original distribution given the feature values in F . We call such a minimum subset an optimal subset.

[Yu, Liu 2004]

Feature selection (5)

Conditional probabilities:

- $P(C|G)$: Probability for C assuming (precondition) that G has occurred.
- $P(C, G)$: Probability for the occurrence of C and G (compound probability).
- It applies:

$$P(C, G) = P(C|G) \cdot P(G) = P(G|C) \cdot P(C)$$

Feature selection (6)

Informal clarification of terms

- Target of the characteristic selection:

Its objective is to select a minimal subset of features according to some reasonable criteria so that the original task can be achieved equally well, if not better. [Liu, Motoda 1998]

- Why better?
 - If the number of characteristics is lower, a better generalization performance may be possible!
 - Cf. overfitting of a classifier (later)

Feature selection (7)

Relevance of characteristics

- Definition (highly relevant)

A feature F_i is highly relevant if the following applies:

$$P(C|F_i, S_i) \neq P(C|S_i)$$

- Feature F_i is absolutely necessary for an optimal subset. It cannot be removed without changing the original distribution.
- Whereby S_i : set of all features without F_i

Feature selection (8)

Relevance of characteristics

- Definition (weakly relevant)

A feature F_i is weakly relevant if the following applies:

$$P(C|F_i, S_i) = P(C|S_i)$$

and: $\exists S'_i \subset S_i: P(C|F_i, S'_i) \neq P(C|S'_i)$

- Feature F_i is not always necessary for an optimal subset, but may become necessary under certain conditions.

Feature selection (9)

Relevance of characteristics

- Definition (irrelevant)

A feature F_i is irrelevant if the following applies:

$$\forall S'_i \subseteq S_i: P(C|F_i, S'_i) = P(C|S'_i)$$

- Feature F_i is never necessary for an optimal subset.

Feature selection (10)

Relevance of characteristics

- Definition (redundant)

Two feature F_i and F_k are redundant if they are fully correlated.

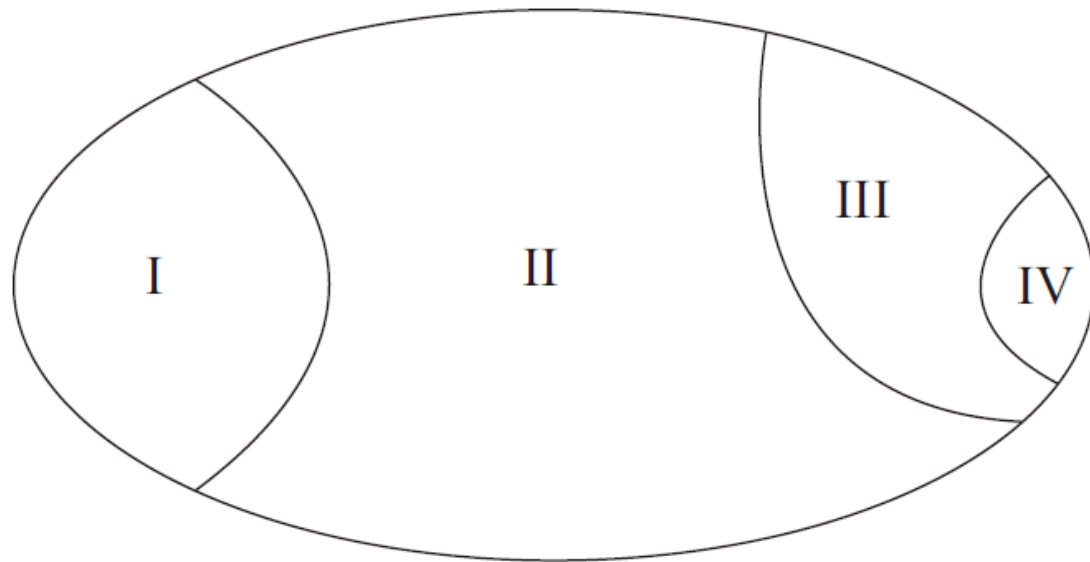
- Examples:

- $F_i = F_k$
- $F_i = \overline{F_k}$
- $F_i = F_k + 1$
- ...

Feature selection (11)

Relevance and redundancy

- I: irrelevant features
 - II: weakly relevant features
 - III: weakly relevant but non-redundant features
 - IV: strongly relevant features
 - III+IV: optimal subset
-
- An optimal subset contains all strongly relevant features, no irrelevant features and a subset of weakly relevant features!



Feature selection (12)

Challenge

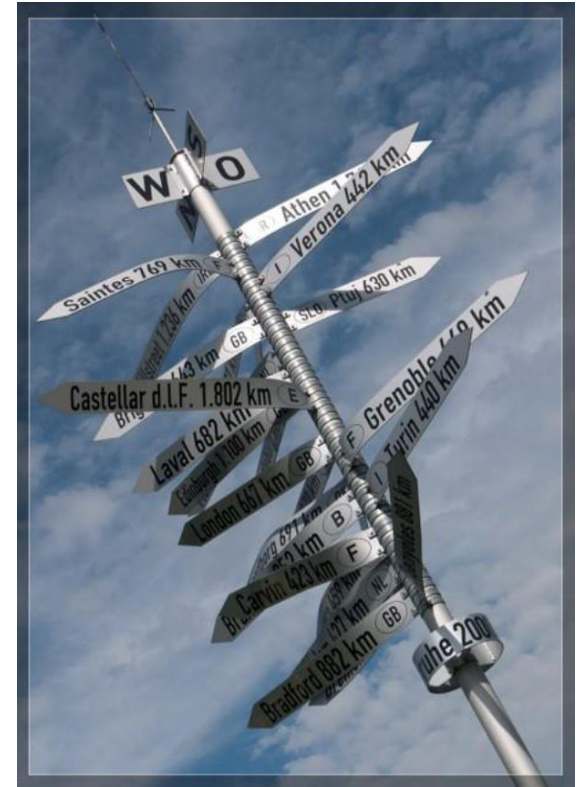
- In practice ...
 - there can be non-linear relationships between features,
 - groups of features can be interchangeable, not just individual features,
 - accuracy is difficult to measure when it comes to generalisation performance,
 - ...
- So how can "important" features be identified?

Feature selection (13)

Approaches to feature selection

- **Manual feature selection**: Selection of relevant features by hand, i.e. expert knowledge is required.
- **Automatic feature selection**: Selection of relevant features automatically using suitable procedures.
- Here: automatic feature selection

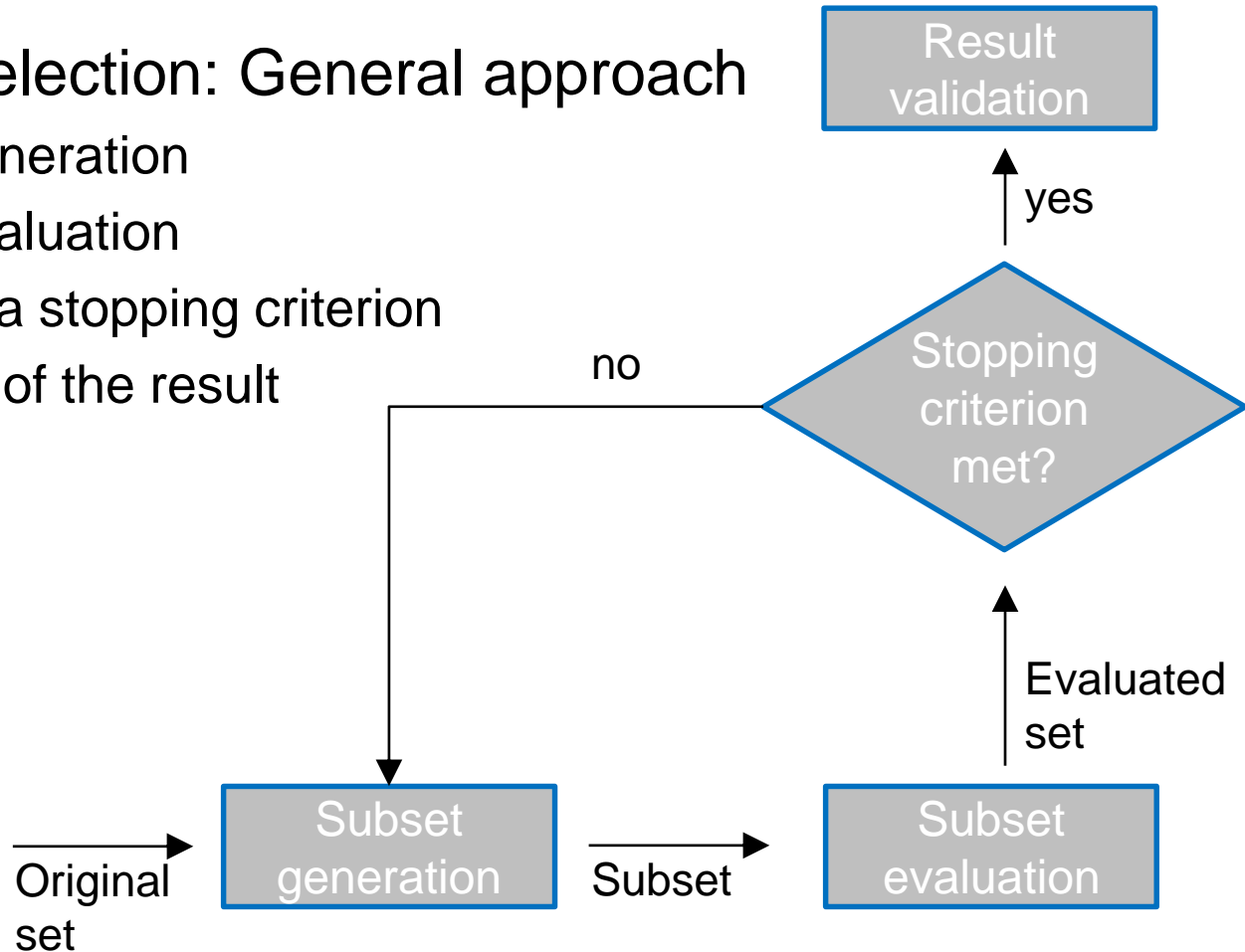
- Feature extraction
- Time-independent features
- Time-dependent features
- Feature selection
- **Taxonomy of feature selection**
- Feature selection algorithms
- Feature transformation
- Principle Component Analysis
- Conclusion
- Further readings



Taxonomy: Feature selection

Feature selection: General approach

1. Subset generation
2. Subset evaluation
3. Checking a stopping criterion
4. Validation of the result



Taxonomy: Feature selection (2)

Taxonomy

- Feature selection procedures can be classified using the following criteria:
 - Filters, wrappers and hybrid solutions
 - Monitored or unmonitored approaches
 - Generation of subsets (search directions and search strategies in the feature space)
 - Evaluation measures for selection algorithms
 - Stopping criteria

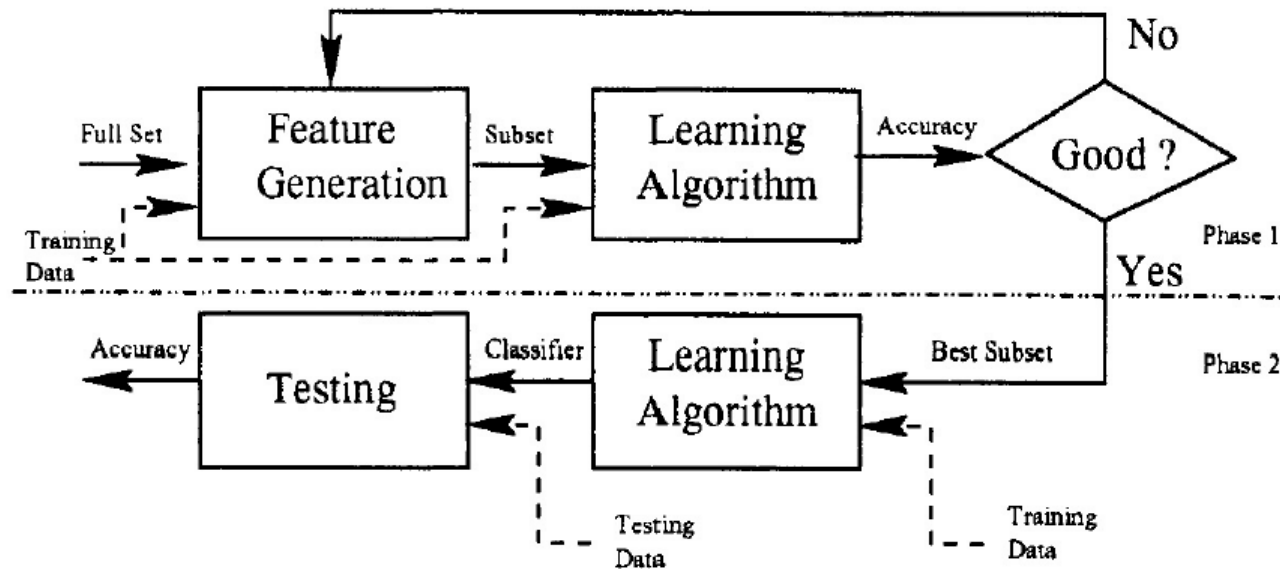
Taxonomy: Feature selection (3)

Wrappers and Filters

- Filters evaluate the quality of features based on an analysis of the data in the feature space.
- Wrappers evaluate the quality of a model instance found with selected features and a suitable ML algorithm (e.g. classifier).
- In principle, wrappers can deliver better results because the actual task is taken into account.
- However, the time expenditure is considerably higher, possibly too high in some tasks.
- Hybrid approaches are possible.

Taxonomy: Feature selection (4)

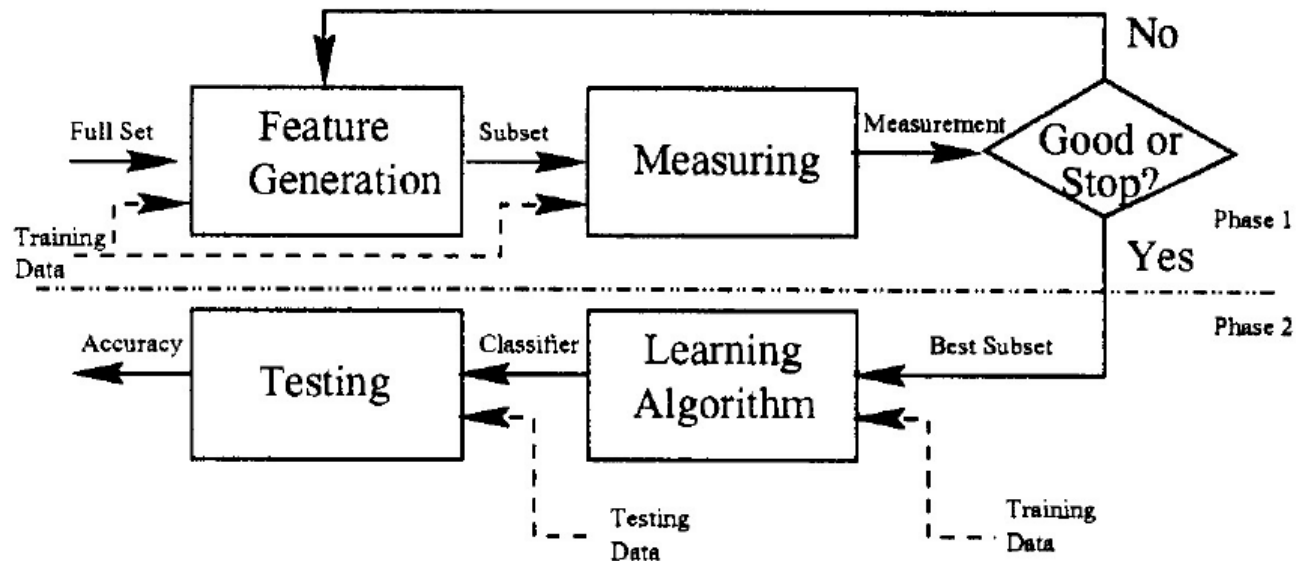
Wrapper



[Liu, Motoda 1998]

Taxonomy: Feature selection (5)

Filter



[Liu, Motoda 1998]

→ We are primarily interested in filters!

Taxonomy: Feature selection (6)

Supervised vs. unsupervised approaches

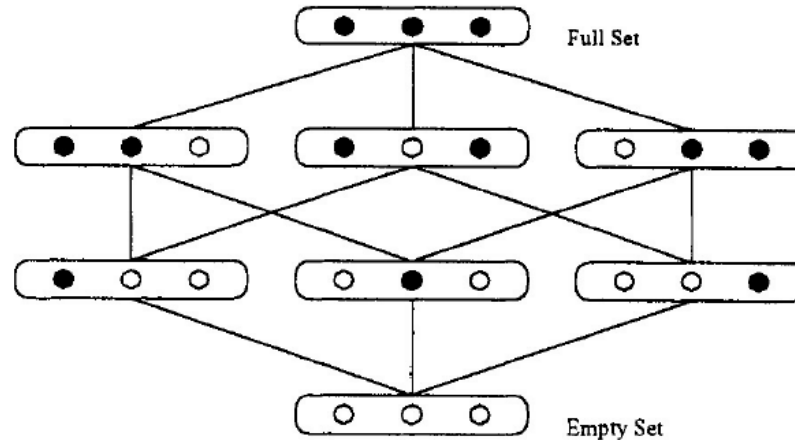
- **Supervised procedures** for feature selection (here: classification!) work with patterns with predefined class assignments (that is, data with labels).
- **Unsupervised procedures** for feature selection work with patterns without class assignment.
(Techniques: cluster analysis, entropy determination, etc.)

→ We are primarily interested in supervised methods!

Taxonomy: Feature selection (7)

Generation of subsets

- The solution space for feature selection consists of all subsets of the set of available features:



[Liu, Motoda 1998]

- The number N of features is sometimes $\gg 1000 \dots$

Taxonomy: Feature selection (8)

Generation of subsets

- Questions:
 - Where is the search started?
 - In which direction is the search conducted?
- According to which criterion is the next possible solution selected?
- In order to generate solutions ("generating" subsets of the feature set),
 - the **search direction** and
 - the **search strategy**
- must be defined.

Search direction

Generation of subsets

- Search directions:
 - Sequential Forward Generation (SFG)
 - Sequential Backward Generation (SBG)
 - Bidirectional Generation (BG)
 - Random Generation (RG)
- Search strategies:
 - Exhaustive / Complete Search
 - Heuristic Search
 - Nondeterministic search

Search direction (2)

Generation of subsets

- Search direction **Sequential Forward Generation** (SFG):
 - Starting from an empty set of features, exactly one feature (the "best" in each case) is added in each run.
 - The search ends when either all features are selected or a certain criterion is reached regarding the quality of the selected features.
 - The result is a ranking of features; if required, a fixed number (the "best" k features) can be selected.
 - In the first step, the **forward generation** variant considers all one element subsets, then all two element subsets, and so on.

Search direction (3)

Generation of subsets

- Search direction **Sequential Backward Generation** (SBG):
 - Starting from the total set of features, exactly one feature (the "most unimportant" feature) is removed in each run.
 - The search ends when there is only one feature left or when the evaluation criterion falls below a specified value.
 - A ranking of features is also created here.
 - In the first step, the **Backward Generation** variant considers all N -element subsets, then all $(N - 1)$ -element subsets, and so on.

Search direction (4)

Generation of subsets

- Search direction **Bidirectional Generation** (BG):
 - Here you start competing in both directions, i.e. with SFG and SBG.
 - Goal of BG: Acceleration of the search, since the search in one of the two directions often ends early.
 - The search ends when either a given number of k features in one of the two directions has been reached or when both methods perform a run with $\frac{N}{2}$ –elementary subsets.

Search direction (5)

Generation of subsets

- Search direction **Random Generation (RG)**:
 - Problem with SFG, SBG, BG: Accepting the best feature in the respective step or removing the feature least important in the respective step does not have to lead to the globally optimal solution!
 - Goal of RG: Avoidance of local optima
 - Here you start with a random selection of features.
 - In each run a feature is either removed or added (more or less) randomly.
 - Termination criterion here is the time or a given number of runs.

Search strategy

Generation of subsets

- Search Strategy **Exhaustive Search**:
 - All possible feature combinations are examined.
 - E.g. forward generation: $\binom{N}{0} + \binom{N}{1} + \binom{N}{2} + \dots + \binom{N}{M}$
 - Cf. width search
 - The runtime complexity is therefore $O(2^N)$.
 - Global optimum is always found.
 - Hardly feasible in practice, only for very small N .
 - Possibility to find the global optimum without explicitly evaluating all solutions (**Complete Search**). A special property of the evaluation criterion for selection algorithms (see below) is assumed: monotony.

Search strategy (2)

Generation of subsets

- Search Strategy **Heuristic Search**:
 - Heuristics (e.g. based on expert knowledge) are used.
 - Usually searches a specific path from the full set of characteristics to the empty subset or vice versa; then the runtime is $O(N)$.
 - Cf. depth search
 - Finding the global optimum cannot be guaranteed; however, very good optima are often found.

Search strategy (3)

Generation of subsets

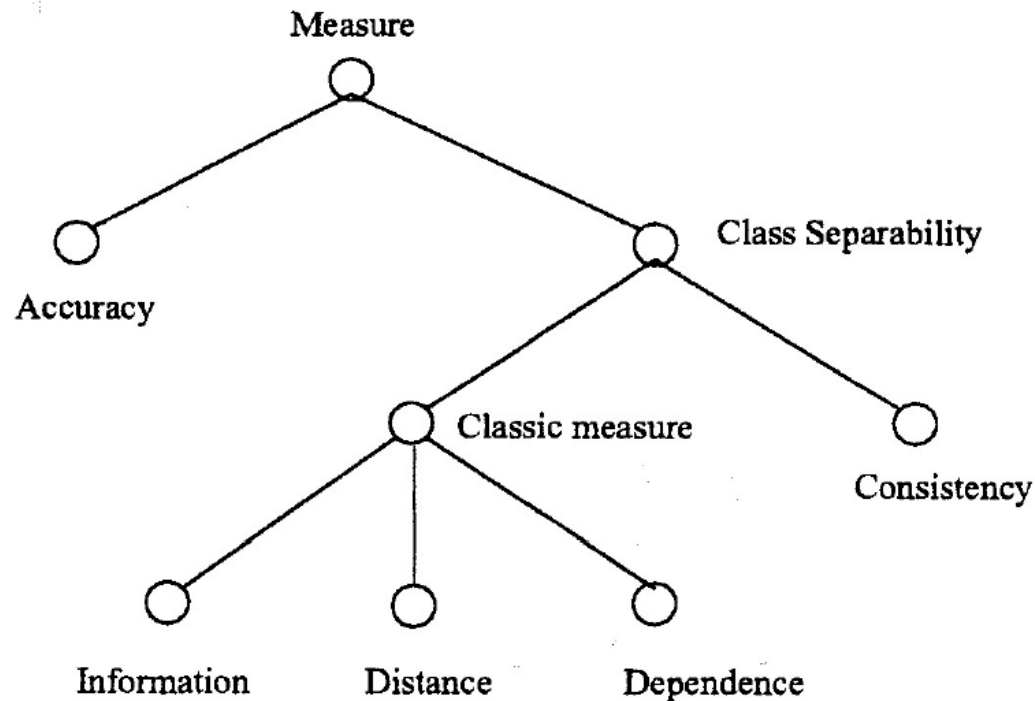
- Search Strategy **Non-deterministic Search**:
 - Selection of the next set of features not according to a fixed rule, but randomly.
 - I. e., next subset is not created by deterministic addition/removal of features.
 - Global optimum cannot be guaranteed either.
 - The stopping criterion is typically a fixed number of runs.

Evaluation measures

Evaluation measures

- Evaluation measures ([selection criteria](#)) are required to evaluate the quality of a generated subset of characteristics.
- Classes of measures:
 - Accuracy Measures
 - Classic measures
 - Information Measures
 - Distance measures
 - Dependence Measures
- Consistency Measures

Evaluation measures



[Liu, Motoda 1998]

Evaluation measures (3)

Accuracy measures ...

- ... evaluate the quality of the model instance created with selected features (e.g. by the classification rate).
- ... are therefore generally used for wrapper approaches, i.e. the following problems occur:
 - Selection of an appropriate ML paradigm (e.g. neural net, SVM, decision tree, ...)
 - Time required for model creation in each run
 - Avoid over-fitting in each run
 - Evaluation of the generalisation performance in each run

Evaluation measures (4)

Information measures ...

- ... evaluate (originally) the uncertainty of a recipient when receiving a message:
 - If the recipient knows which message he will receive, his surprise (uncertainty) is small.
 - If the recipient does not know which message is coming (assumption: all messages are equally likely to come), his uncertainty is high.
 - For classification tasks: Messages are classes.
- .. are often based e.g. on Shannon's entropy measure.

Evaluation measures (5)

Evaluation measures

- Definition (Shannon's Entropy E)

$$E(D) = - \sum_{i=1}^d p_i \log_2 p_i$$

- D : Data volume
- d : Number of classes
- p_i : Probability for class i
- Measure of the information content (impurity, disorder) of a statement.
- The greater the entropy, the greater the information content.

Evaluation measures (6)

Shannon's Entropy

- Example: Coin toss

- $p_{head} = 0.5, p_{tail} = 0.5, d = 2$

SOLUTION

- Example: Faked coin toss

- $p_{head} = 0.75, p_{tail} = 0.25, d = 2$

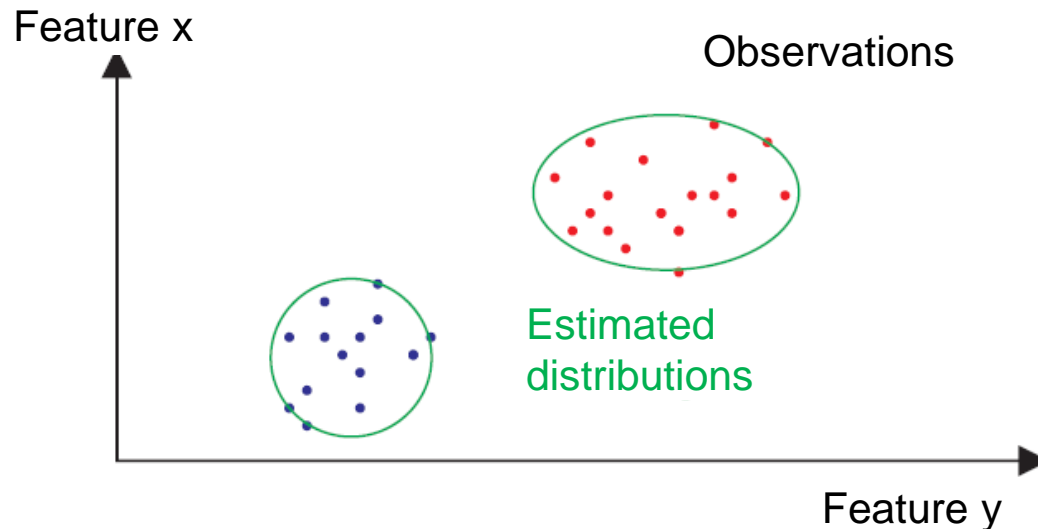
SOLUTION

- The more "predictable" the result, the smaller the information content!

Evaluation measures (7)

Distance measures ...

- ... evaluate the separability of classes, e.g. on the basis of estimated distribution functions.
- ... are therefore also known as separability measures, divergence measures, or discrimination measures.



- Feature y is probably better suited for separation than feature x

Evaluation measures (8)

Dependency measures ...

- ... evaluate (originally) the correlation of features with the correlation coefficient ϱ , i.e. they answer the question: How predictable is the value of a feature from the value of another feature?

$$\varrho = \frac{Cov(X, Y)}{\sigma(X)\sigma(Y)}$$

- ... investigate the correlation of features with classes.
- ... are also known as association measures or correlation measures.

Evaluation measures (9)

Consistency measures ...

- avoid a problem of classical measures: They cannot recognise redundant features, i.e. they cannot distinguish between equally good features.
- ... examine contradictory patterns (i.e., the same input features), but different class assignments.
- ... are also able to identify irrelevant features.

Evaluation measures (10)

Evaluation measures

- Specific examples for different evaluation measures will be presented in the following by means of a simple application example.
- The question is whether the fact that a person gets a sunburn depends:
 - on his hair colour,
 - on his size,
 - on his weight,
 - on the use of sun lotion.

Evaluation measures (11)

Example

[Liu, Motoda 1998]

- Sunburn, depending on hair colour, size, weight, use of sun lotion:

Hair colour	Size	Weight	Use of sun lotion	Result
Blonde	Average	Light	No	Sunburned
Blonde	Large	Average	Yes	None
Brown	Small	Average	Yes	None
Blonde	Small	Average	No	Sunburned
Red	Average	Heavy	No	Sunburned
Brown	Large	Heavy	No	None
Brown	Average	Heavy	No	None
Blonde	Small	Light	Yes	None

Evaluation measures (12)

Example

[Liu, Motoda 1998]

- Coding based on probabilities

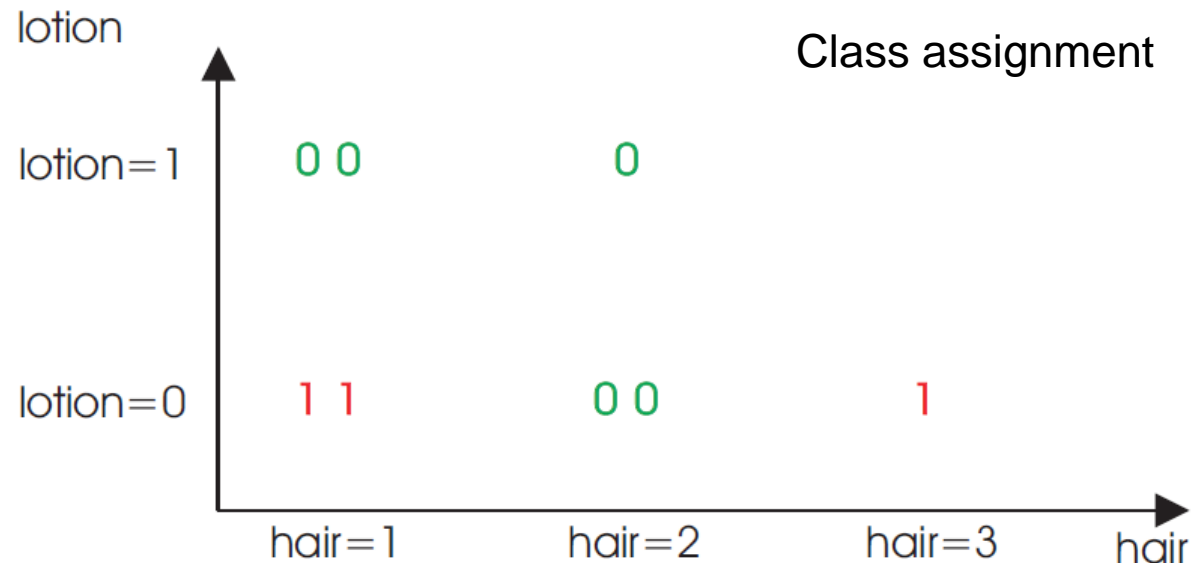
	Hair	Height	Weight	Lotion	Result
i_1	1	2	1	0	1
i_2	1	3	2	1	0
i_3	2	1	2	1	0
i_4	1	1	2	0	1
i_5	3	2	3	0	1
i_6	2	3	3	0	0
i_7	2	2	3	0	0
i_8	1	1	1	1	0

	Result (Sunburn)	
	No	Yes
$P(\text{Result})$	$5/8$	$3/8$
$P(\text{Hair}=1 \text{Result})$	$2/5$	$2/3$
$P(\text{Hair}=2 \text{Result})$	$3/5$	0
$P(\text{Hair}=3 \text{Result})$	0	$1/3$
$P(\text{Height}=1 \text{Result})$	$2/5$	$1/3$
$P(\text{Height}=2 \text{Result})$	$1/5$	$2/3$
$P(\text{Height}=3 \text{Result})$	$2/5$	0
$P(\text{Weight}=1 \text{Result})$	$1/5$	$1/3$
$P(\text{Weight}=2 \text{Result})$	$2/5$	$1/3$
$P(\text{Weight}=3 \text{Result})$	$2/5$	$1/3$
$P(\text{Lotion}=0 \text{Result})$	$2/5$	$3/3$
$P(\text{Lotion}=1 \text{Result})$	$3/5$	0

Evaluation measures (13)

Example

- The data set is small, the relationships simple, therefore a simple analysis is possible here:



- The features 'lotion' and 'hair' are sufficient for an unambiguous class allocation
- However: Both features are required!

Evaluation measures (14)

Information measure „Information Gain“ (IG)

- Partitioning of a data set X by a feature d in L subsets X_{d_l} with $l = 1, \dots, L$. There are C classes corresponding to the manifestations (possible instances) of the feature.
- Information Gain (IG) of a feature d :

$$IG(d) := I(X) - \sum_{l=1}^L \frac{|X_{d_l}|}{|X|} I(X_{d_l})$$

$$I(X_{d_l}) := - \sum_{c=1}^C p_{X_{d_l}}(c) \cdot \log_2 p_{X_{d_l}}(c)$$

$$I(X) := - \sum_{c=1}^C p_X(c) \cdot \log_2 p_X(c)$$

Evaluation measures (15)

Information measure „Information Gain“ (IG)

- Example:

$$\begin{aligned} I(X) &= - \sum_{c=1}^2 px(c) \cdot \log_2 px(c) \\ &= -\frac{5}{8} \log_2 \frac{5}{8} - \frac{3}{8} \log_2 \frac{3}{8} \\ &= 0,954 \end{aligned}$$

$$\begin{aligned} I(X_{hair_1}) &= - \sum_{c=1}^2 px_{hair_1}(c) \cdot \log_2 px_{hair_1}(c) \\ &= -\frac{2}{4} \log_2 \frac{2}{4} - \frac{2}{4} \log_2 \frac{2}{4} \\ &= 1 \end{aligned}$$

Evaluation measures (16)

Information measure „Information Gain“ (IG)

- Example:

$$\begin{aligned} I(X_{hair_2}) &= - \sum_{c=1}^2 px_{hair_2}(c) \cdot \log_2 px_{hair_2}(c) \\ &= -1 \log_2 1 - 0 \log_2 0 \\ &= 0 \end{aligned}$$

$$\begin{aligned} I(X_{hair_3}) &= - \sum_{c=1}^2 px_{hair_3}(c) \cdot \log_2 px_{hair_3}(c) \\ &= -0 \log_2 0 - 1 \log_2 1 \\ &= 0 \end{aligned}$$

→ No unpredictable assignment in both cases!

Evaluation measures (17)

Information measure „Information Gain“ (IG)

- Example:

$$\begin{aligned} IG(hair) &= I(X) - \sum_{c=1}^3 \frac{|X_{hair_c}|}{|X|} I(hair_c) \\ &= 0,954 - \frac{4}{8} \cdot 1 \\ &= 0,454 \end{aligned}$$

Since $IG(hair) = 0,454$, $IG(lotion) = 0,348$, $IG(height) = 0,266$ and $IG(weight) = 0,016$, is the feature *hair* the first one in the rank list.

Evaluation measures (18)

Consistency measure "inconsistency rate" (IR)

- Two patterns are inconsistent if they are identical except for the class assignment.
- The inconsistency number IZ of a pattern is the number of occurrences of this pattern minus the largest number with a particular class assignment.
- The inconsistency rate IR of a data set is the sum of all inconsistency numbers divided by the total number of samples.

Please note:

- IR is calculated during feature selection for various subsets that are created from the original data set by omitting features.

Evaluation measures (19)

Consistency measure "inconsistency rate" (IR)

- Example:

Sample	Class A	Class B	Class C	
00	1	1	2	$\rightarrow IZ(00) = 4 - 2 = 2$
01	2	3	0	$\rightarrow IZ(01) = 5 - 3 = 2$
10	0	0	1	$\rightarrow IZ(10) = 1 - 1 = 0$
11	2	8	0	$\rightarrow IZ(11) = 10 - 8 = 2$

$$\Rightarrow IR = \frac{2+2+0+2}{20} = 0.3$$

Remarks:

- IR is calculable in $O(N)$ (N: number of samples)
- IR is a monotonous measure

Evaluation measures (20)

Consistency measure "inconsistency rate" (IR)

- Selection criterion for subset based on IR:
 - Be D_i and D_j two (sub)sets of features with inconsistency rates $IR(D_i)$ and $IR(D_j)$. Then applies:
 - D_i and D_j are indistinguishable if $IR(D_i) = IR(D_j)$ and $|D_i| = |D_j|$
 - D_i will be preferred to D_j if
 - $IR(D_i) = IR(D_j)$ and $|D_i| < |D_j|$ or
 - $IR(D_i) < IR(D_j)$ and $|D_i| \leq |D_j|$

Evaluation measures (21)

Consistency measure "inconsistency rate" (IR)

- Example sunburn: The inconsistency rate is 0 for
 - the two features hair and lotion
 - the three characteristics hair, height and weight
 - any supersets of these two quantities
- Since the smallest quantity of features is sought, the two features hair and lotion are chosen.

Evaluation measures (22)

There are many other dimensions, for example:

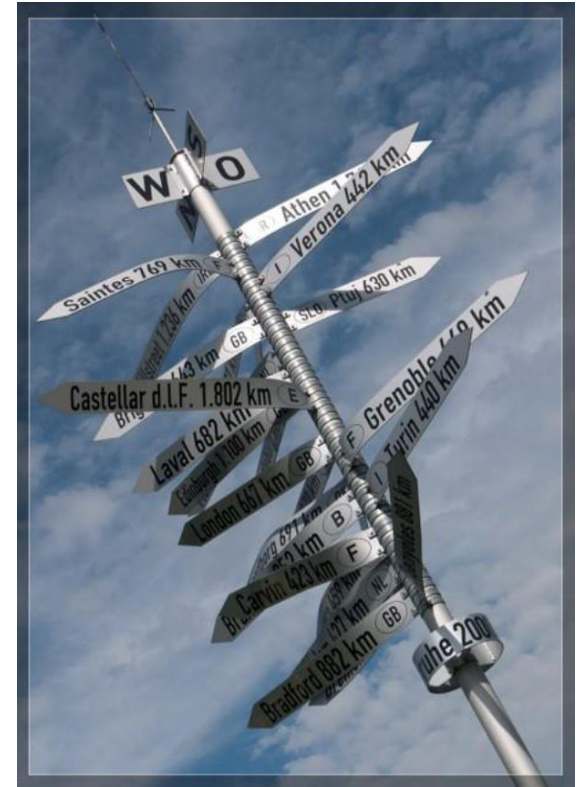
- Minimum Description Length
- Distance measure Directed Divergence
- Distance dimension Variance
- Bhattacharyya dependence measure
- etc.

Evaluation measures (23)

Different stopping criteria are possible:

- **Time**: Termination when time limit is reached
- **Number of features**: Termination if predefined number of features is selected (forwards and backwards possible)
- **Iterations**: Termination if a given number of iterations have been passed through in the algorithm
- **Algorithm-Specific** stopping criteria
- **Application-dependent** stopping criteria

- Feature extraction
- Time-independent features
- Time-dependent features
- Feature selection
- Taxonomy of feature selection
- **Feature selection algorithms**
- Feature transformation
- Principle Component Analysis
- Conclusion
- Further readings



Selected Algorithms

- Focus
- Automated Branch and Bound (ABB)
- Relief

Selection algorithms (2)

FOCUS

- Idea:
 - Start with empty characteristic set
 - Consider all $\binom{D}{1}, \binom{D}{2}, \binom{D}{3}, \dots$ combinations of features of size 1, 2, 3, ... (D : Total number of features)
 - Stop if the inconsistency rate equals 0 (alternatively: predefined inconsistency rate)
 - Subset must be examined $\sum_{d=1}^{D'} \binom{D}{d}$ in order to find a subset of the size $D' \leq D$.

Selection algorithms (3)

FOCUS

Focus

Input: F - all features x in data D

U - inconsistency rate as evaluation measure

initialize: $S = \{\}$

for $i = 1$ to N

for each subset S of size i

 if $\text{Cal}U(S, D) = 0$ /* $\text{Cal}U(S, D)$ returns inconsistency */

return S

Output: S - a minimum subset that satisfies U

Selection algorithms (4)

Classification of Focus:

- Type: Filter
- Supervised: yes
- Search direction: SFG (sequential forward generation)
- Search strategy: comprehensive search (exhaustive search)
- Measure: Consistency measure (inconsistency rate)
- Termination criterion: all feature subset examined or subset found with $IR = 0$

Selection algorithms (5)

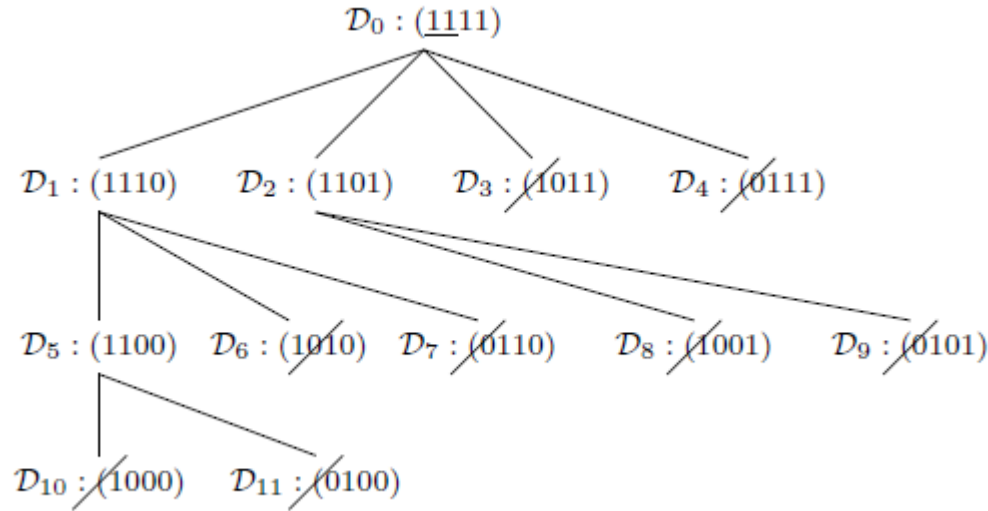
Automated Branch and Bound (ABB) :

- Idea:
 - Start with complete feature set.
 - If IR of the subset is larger, stop BS at this point and mark the subset as invalid; accordingly, all successors of this node (subsets) are also invalid, otherwise continue with BS.
 - When BS is finished, return the smallest valid feature set.

Selection algorithms (6)

Automated Branch and Bound (ABB) :

- Example:
 - Set of 4 features
 - Assumption: the first two features are important, the other two are not (omitting increases inconsistency rate)



Selection algorithms (7)

Automated Branch and Bound (ABB) :

- Comments on the method
 - Coding of the selected features:
 - Binary string (1: Feature) selected; 0: Feature not selected).
 - Generate subsets:
 - Set first "0" from left as marker.
 - Generate subsets by changing a "1" to a "0" from the marker to the left.
 - Find "non-legitimate" subsets:
 - Generated subsets are "non-legitimate" (no need to be evaluated) if their Hamming distance to a subset already marked as "invalid" is "1".
 - Find "invalid" subsets:
 - Generated subsets are "invalid" if their inconsistency rate is higher (Pruning: trimming the BS tree at this point).

Selection algorithms (8)

Hamming distance

- Defined as: $d_H(x, y) = |\{i = 1, \dots, l \mid x_i \neq y_i\}|$
- With (x, y) samples.

Determine the Hamming-Distance of these sample pairs:

- 000110 – 011110
- 101110 – 010010
- Complex – Komplex
- Komplex – Komplexe

Selection algorithms (9)

Automated Branch and Bound (ABB) :

- Characteristics of the approach
 - Complexity depends on the number of subsets considered, which is relatively small on average due to Pruning.
 - The complexity of the legitimacy check depends on the number of features, the complexity of the inconsistency check on the number of samples.
 - Here, too, a threshold value can be used for the inconsistency rate.

Selection algorithms (10)

Classification of ABB:

- Type: Filter
- Supervised: yes
- Search direction: SBG (sequential backward generation)
- Search strategy: complete search
- Measure: Consistency measure (inconsistency rate)
- Stopping criterion: algorithm specific

Selection algorithms (11)

Idea of RELIEF:

- Creates ranking of features by assigning weights to the features (Feature Ranking)
- Finds features that are statistically relevant by calculating distances on samples (!).
- Basic idea for two-class problems:
 - For samples the next neighbour (sample) of the same class (nearest-hit) and the other class (nearest-miss) is searched with a next neighbour algorithm.
 - Features are regarded as relevant if the respective values are similar in the considered sample and nearest-hit or different in the considered sample and nearest-miss.
- Weights are calculated iteratively.

Selection algorithms (12)

RELIEF:

- Be $x = (x_1, x_2, \dots, x_D)$ as well as $y = (y_1, y_2, \dots, y_D)$ each sample of the data set (D : number of features). Then the following dimensions are required:
 - Nearest-hit(x): Pattern $x^{(H)}$ with the smallest (usually for continuous, standardised features: Euclidean) distance to x among all samples with the same class assignment.
 - Nearest-miss(x): Pattern $x^{(M)}$ with the smallest (usually for continuous, standardised features: Euclidean) distance to x among all samples with different class assignments.
 - $||x_d - y_d||$: Distance between the two values x_d and y_d (e.g. 0/1 for discrete features or difference for continuous, standardised features).

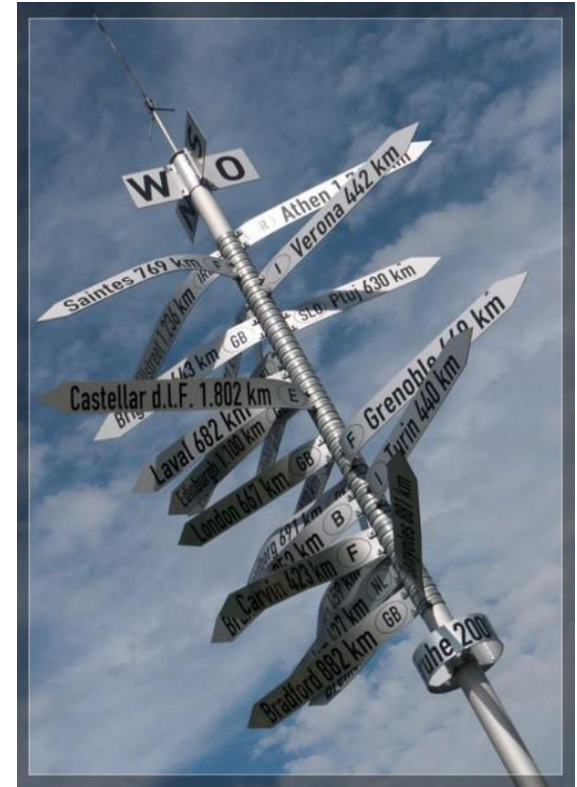
For this purpose w is the vector of the characteristic weights

Selection algorithms (13)

Classification of RELIEF:

- Type: Filter
- Supervised: yes
- Search direction: difficult to classify (looks at samples)
- Search strategy: difficult to classify (looks at samples)
- Dimension: distance dimension (e.g. Euclidean distance on samples)
- Stopping criterion: specified number of iterations

- Feature extraction
- Time-independent features
- Time-dependent features
- Feature selection
- Taxonomy of feature selection
- Feature selection algorithms
- **Feature transformation**
- Principle Component Analysis
- Conclusion
- Further readings



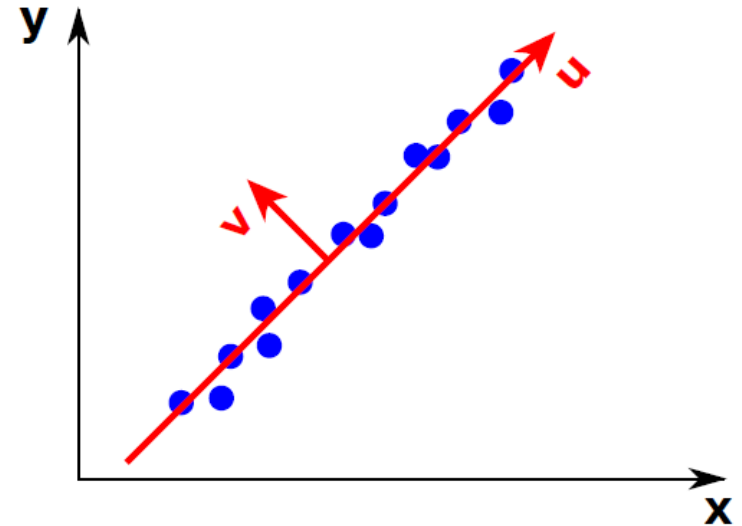
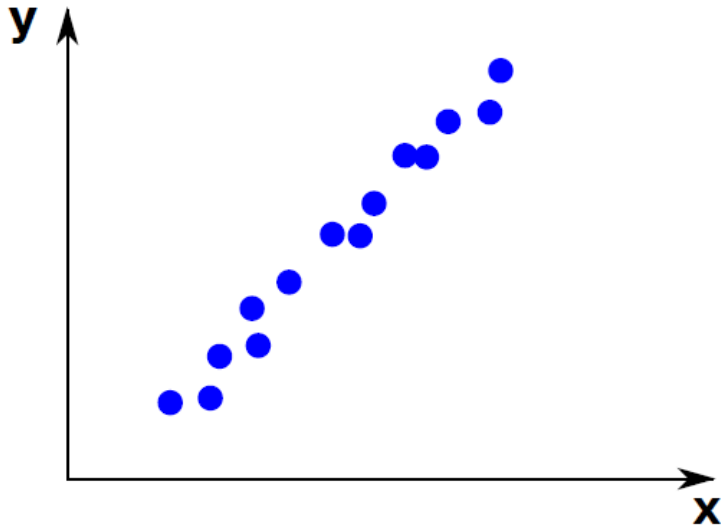
Feature transformation

Feature transformation: reduction of dimensions

- Previously: Reduction of existing information by selecting a subset of "important/relevant" features (attributes, dimensions).
 - Start with empty, complete or random subset of all features
 - Successive addition or omission of certain features
 - Until certain stopping criterion (quality, number of iterations, ...) fulfilled
- Now: Transformation of data
 - Projection into a space of smaller dimensionality
 - Objective: Summarisation of relevant information in a smaller number of features
 - Maintain data structure as well as possible
 - Transformed characteristics: Linear combination of the original features (main components)

Feature transformation (2)

Feature transformation



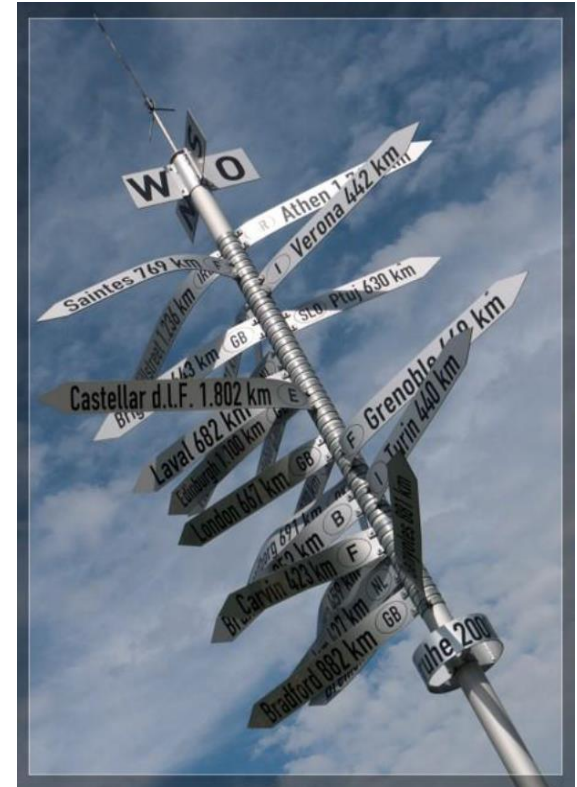
Feature transformation (3)

Motivation

- Given:
 - A data set X with samples $x_1, x_2, x_3, \dots, x_N$
 - The samples are D -dimensional, i.e. there are D features.
- Wanted:
 - A data set Y with samples $y_1, y_2, y_3, \dots, y_M$
 - The samples are also D -dimensional, i.e. there are D features
 - The following applies: $|X|=|Y|$, or $N=M$.
 - The information content is the same as in the dataset X , but: the information content of the dataset Y is stored in the first, few features (dimensions).
 - This is also referred to as meta-features.
- Principal Component Analysis (PCA):
 - A method to find such a data set

Agenda

- Feature extraction
- Time-independent features
- Time-dependent features
- Feature selection
- Taxonomy of feature selection
- Feature selection algorithms
- Feature transformation
- **Principle Component Analysis**
- Conclusion
- Further readings



Principal Component Analysis

What does information content mean?

- Assumption: high information content corresponds to high variance!
- Main objective of the Principal Component Analysis:
 - Dimension reduction
 - Less important dimensions can be omitted, i.e., the number D' of features in the transformed data set is $D' \ll D$.

Principal Component Analysis (2)

Benefits of Principal Component Analysis:

- **Time saving**: by using DM algorithms on reduced data sets.
- **Feature selection**: very simple by selecting the most important meta-features.
- **Understanding**: better recognition of structures in data, e.g. by visualising the data set in the space of the two or three most important meta-features.

Other names for Principal Component Analysis:

- Hotelling Transformation
- Karhunen-Loeve transformation
- ...

Principal Component Analysis (3)

Basics

- To transform a data set, the **arithmetic mean of each sample** i is calculated first:

$$\mu_i := \frac{1}{N} \sum_{n=1}^N x_{i,n}$$

- Instead of the original samples, the **average-cleaned samples** are then used again:

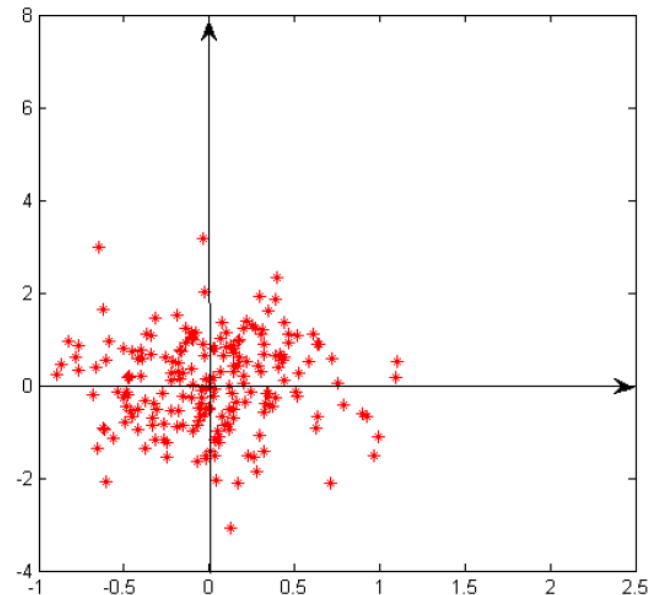
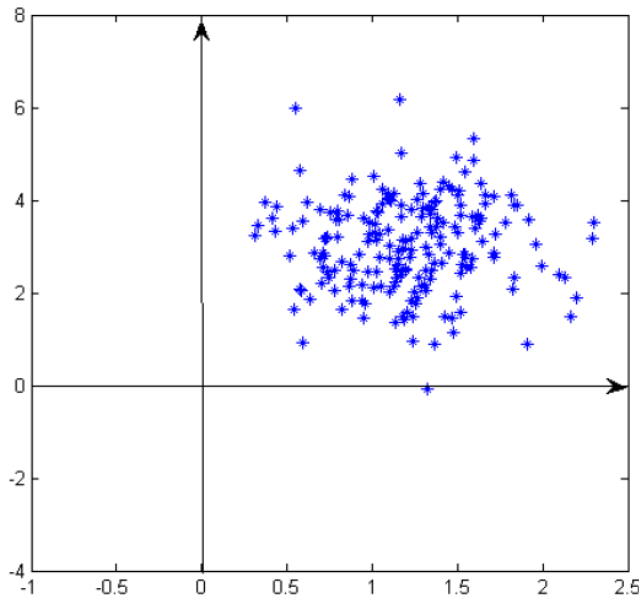
$$\forall_{n=1 \dots N} \forall_{i=1 \dots D} : x'_{i,n} = x_{i,n} - \mu_i$$

- Alternatively:

$$\mu := \begin{pmatrix} \mu_1 \\ \mu_2 \\ \dots \\ \mu_D \end{pmatrix}$$

- $\mu - x_n$ for $n = 1, 2, \dots, N$. This corresponds geometrically to a **translation** (displacement) of the data.

Example:



- The blue dots represent the original dataset, with the arithmetic means $\mu_1 = 1.2$ and $\mu_2 = 3$. The red dots represent the original dataset after translation

Principal Component Analysis (5)

Basics (continued)

- The empirical variance of a feature i is then:

$$\sigma_i^2 := \frac{1}{N-1} \sum_{n=1}^N x'^2_{i,n}$$

- Thus, the empirical standard deviation of the feature is i :

$$\sigma_i = \sqrt{\sigma_i^2}$$

Principal Component Analysis (6)

Basics (continued)

- The covariance of two features i and j is also required:

$$s_{i,j} := \frac{1}{N-1} \sum_{n=1}^N x'_{i,n} \cdot x'_{j,n}$$

- A covariance $s_{i,i}$ (i.e. a feature with itself) is of course the variance again. In addition applies: $s_{i,j} = s_{j,i}$

Principal Component Analysis (7)

Covariance: Interpretation

- Covariance is a measure of the linear relationship between two features.
- $s_{i,j} > 0$: Large values of feature i are associated with large values of feature j (analogous for small values).
- $s_{i,j} < 0$: Large values of feature i are associated with small values of feature j (and vice versa)
- $s_{i,j} = 0$: no linear relationship between the two features

Principal Component Analysis (8)

Covariance

- Covariance is always calculated in pairs, i.e. for two features.
- For D-dimensional data there are $\frac{D!}{(D-2)!2!}$ many covariances.
- If we write the covariances in a matrix:

$$\begin{pmatrix} s_{1,1} & \cdots & s_{1,D} \\ \vdots & \ddots & \vdots \\ s_{D,1} & \cdots & s_{D,D} \end{pmatrix}$$

this matrix is symmetrical. The diagonals show the variances of the features.

Principal Component Analysis (9)

Basics (continued)

- An eigenvector v of such a matrix C is a D -dimensional vector, for which applies:

$$C \cdot v = \lambda \cdot v$$

- $\lambda \in \mathbb{R}$ means eigenvalue to the eigenvector v .

It applies:

- C has D eigenvectors v_1, v_2, \dots, v_D with the corresponding eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_D$.
- Without limitation of the generality assume for v_1, v_2, \dots, v_D :
$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_D$$
- The eigenvectors are perpendicular to each other, i.e. they are orthogonal to each other.
- Multiples of an eigenvector are also eigenvectors, we use those that are normalised to length 1

Principal Component Analysis (10)

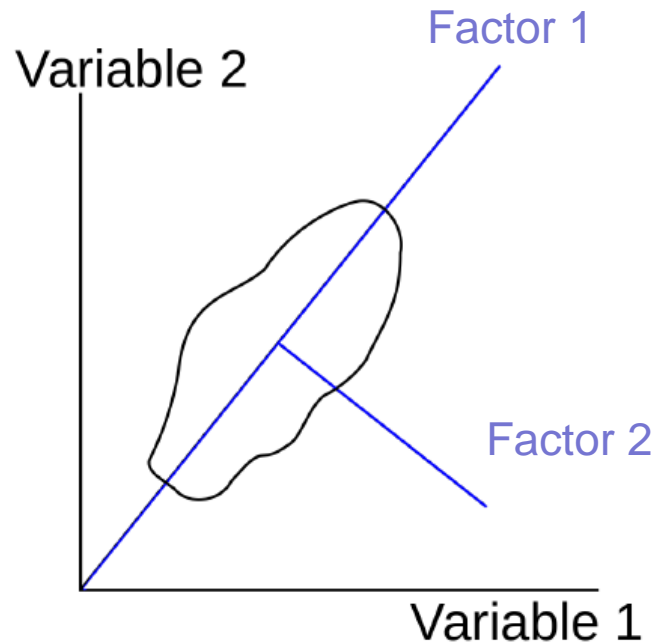
Most important property of the eigenvectors:

- The eigenvector with the highest eigenvalue specifies the direction in which the data set has the highest variance.
- The eigenvector with the second highest eigenvalue specifies an orthogonal direction in which the data set has the second highest variance.
- etc.

The variances are described by the respective eigenvalues!

- Variance → Information content!

Example



Example 2: Three farmers

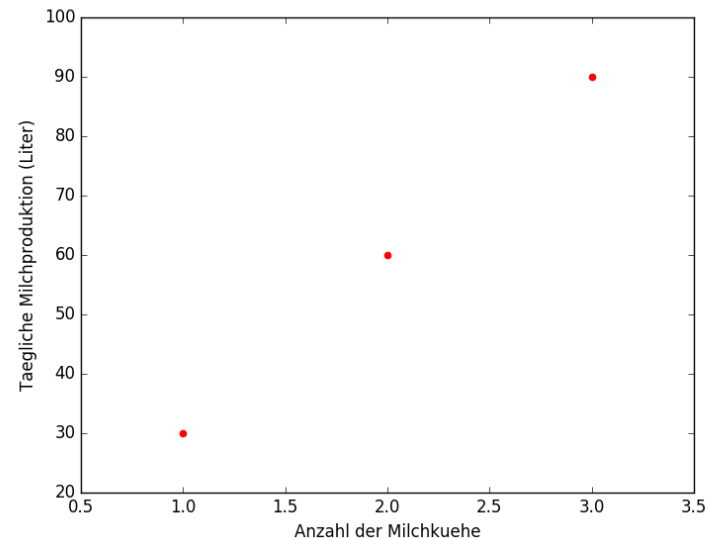
	Anton	Ben	Carl
Number of cows	1	2	3
Daily milk production (litre)	30	60	90
Deviation of #cows from average	-1	0	1
Deviation of #litres from average	-30	0	30
Product of deviations	30	0	30

- Question (to be answered by covariance calculation): Is there a causal relationship between the number of cows and milk production?

Principal Component Analysis (13)

Example 2: Three farmers

- Intuition suggest: Yes!
- See scatter plot



Principal Component Analysis (14)

Example:

- The 3 deviation products are added together: $30 + 0 + 30 = 60$; then divided by the number of feature carriers (3 dairy farmers): $60 / 3 = 20$.
- In the case of the 3 dairy farmers considered, the total population is to be considered here; if, on the other hand, a sample set is considered, the so-called empirical covariance is involved; in this case, it is not to be divided by the number of feature carriers, but by (number of feature carriers - 1) (that would be $3 - 1 = 2$ here).
- The product of the deviations for Anton results from the fact that the negative deviation of -1 is multiplied by the negative deviation of - 30 (litres); this results in +30.
- As a complete formula:

$$Cov = \frac{[(1 - 2) \cdot (30 - 60)] + [(2 - 2) \cdot (60 - 60)] + [(3 - 2) \cdot (90 - 60)]}{n} = \frac{30 + 0 + 30}{3}$$

$$= 20$$
- Positive values of the covariance mean that high values of one characteristic (here: milk production) are accompanied by high values of the other characteristic (here: number of cows) (and vice versa).

Principal Component Analysis (15)

Back to PCS: Basics (continued)

- Next, a specific number $D' \leq D$ is selected from eigenvectors to transform the data:
- All eigenvectors ($D' = D$) are selected if the goal of the Principal Component Analysis is, for example, a principal axis transformation to decorrelate the data.
- In this case, the mean-adjusted samples are transformed as follows:

$$\mathbf{y}_k = \begin{pmatrix} \mathbf{v}_1^T \\ \mathbf{v}_2^T \\ \vdots \\ \mathbf{v}_D^T \end{pmatrix} \mathbf{x}'_k$$

- This corresponds to a rotation of the data.

Principal Component Analysis (16)

- Basics (continued)
- A smaller number of eigenvectors (usually $D' \ll D$) is selected if the goal of the principal component analysis is data reduction.
- In this case, the mean-adjusted patterns are transformed as follows:

$$y_k = \begin{pmatrix} \mathbf{v}_1^T \\ \mathbf{v}_2^T \\ \vdots \\ \mathbf{v}_{D'}^T \end{pmatrix} \mathbf{x}'_k$$

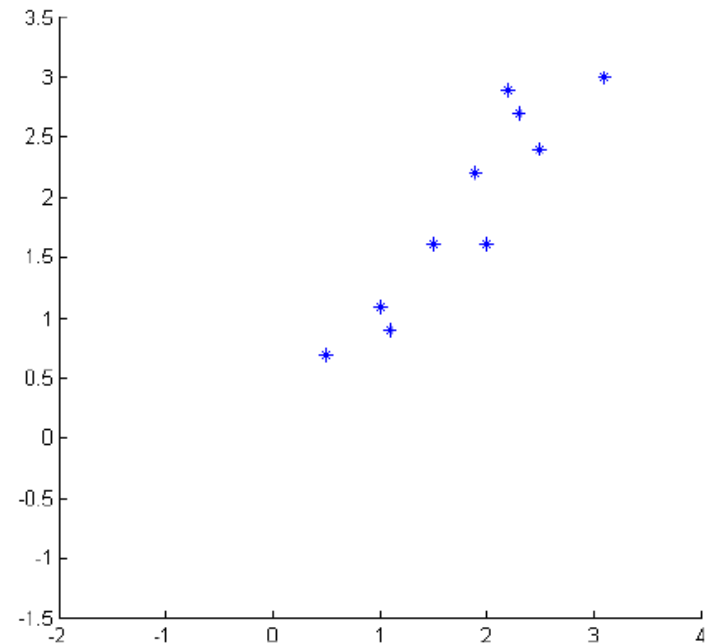
A retransformation is possible - but at D' dimensions only with loss of information

- So the transformed patterns y_k only have dimensions D' .

Principal Component Analysis (17)

Example of a 2-dimensional data set:

Sample	Feature 1	Feature 2
x_1	2.5	2.4
x_2	0.5	0.7
x_3	2.2	2.9
x_4	1.9	2.2
x_5	3.1	3.0
x_6	2.3	2.7
x_7	2.0	1.6
x_8	1.0	1.1
x_9	1.5	1.6
x_{10}	1.1	0.9



Mean A: 1,81

Mean B: 1,91

Principal Component Analysis (18)

PCA: Approach

- In each dimension, the mean value is subtracted from the data.
- The mean value of the transformed data is then 0.

x'_1	0.69	0.49
x'_2	-1.31	-1.21
x'_3	0.39	0.99
x'_4	0.09	0.29
x'_5	1.29	1.09
x'_6	0.49	0.79
x'_7	0.19	-0.31
x'_8	-0.81	-0.81
x'_9	-0.31	-0.31
x'_{10}	-0.71	-1.01

Principal Component Analysis (19)

- The covariance matrix is then calculated:

$$\mathbf{C} = \begin{pmatrix} 0.617 & 0.615 \\ 0.615 & 0.717 \end{pmatrix}$$

- Since the elements apart from the diagonals are positive, there is a positive correlation between the two features (cf. correlation coefficient).

- The eigenvalues and eigenvectors of the matrix C are:

$$\mathbf{v}_1 = \begin{pmatrix} -0.678 \\ -0.735 \end{pmatrix} \text{ mit } \lambda_1 = 1.284$$

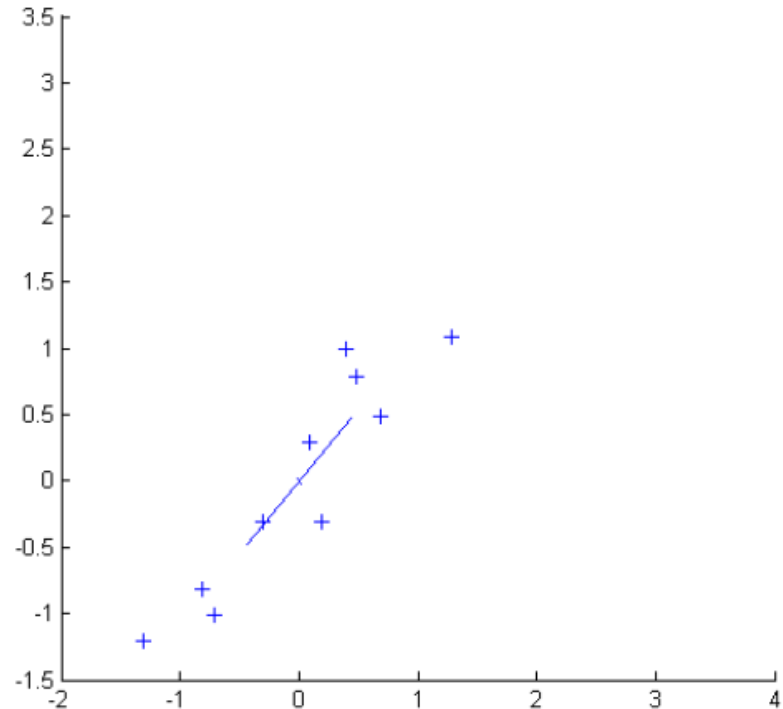
$$\mathbf{v}_2 = \begin{pmatrix} -0.735 \\ 0.678 \end{pmatrix} \text{ mit } \lambda_2 = 0.049$$

- The eigenvectors have length one and are orthogonal to each other; v_1 (higher eigenvalue) describes the first main component, v_2 the second.

Principal Component Analysis (21)

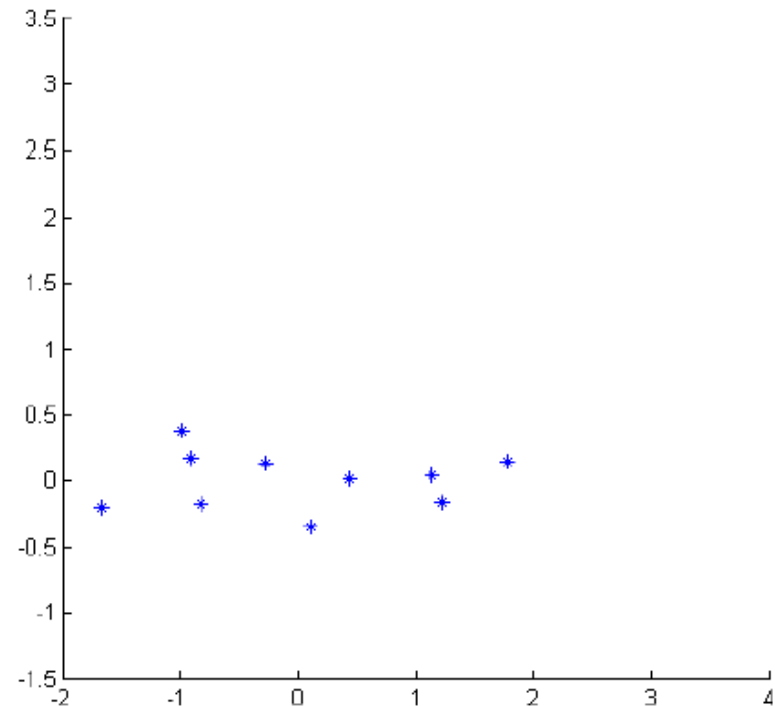
Illustration

- Starting from the mean value, each eigenvector is drawn in both directions; length corresponds to the eigenvalue.



- Transformation of the data using both eigenvectors:

y_1	-0.828	-0.175
y_2	1.778	0.143
y_3	-0.992	0.384
y_4	-0.274	0.130
y_5	-1.676	-0.209
y_6	-0.913	0.175
y_7	0.099	-0.350
y_8	1.145	0.046
y_9	0.438	0.018
y_{10}	1.224	-0.163



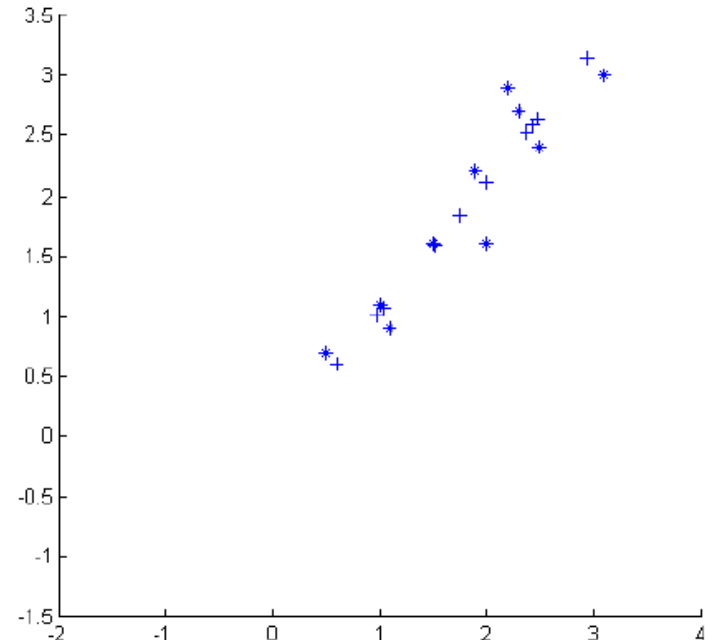
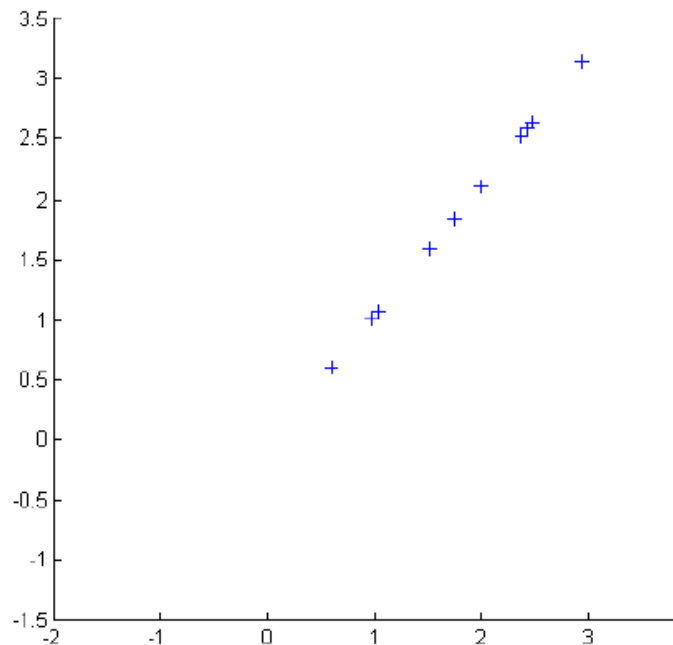
- Transformation of the data using the eigenvector with the higher eigenvalue:

y_1	-0.828
y_2	1.778
y_3	-0.992
y_4	-0.274
y_5	-1.676
y_6	-0.913
y_7	0.099
y_8	1.145
y_9	0.438
y_{10}	1.224

This of course
corresponds to the
first column in the
table of the previous
slide!

Principal Component Analysis (24)

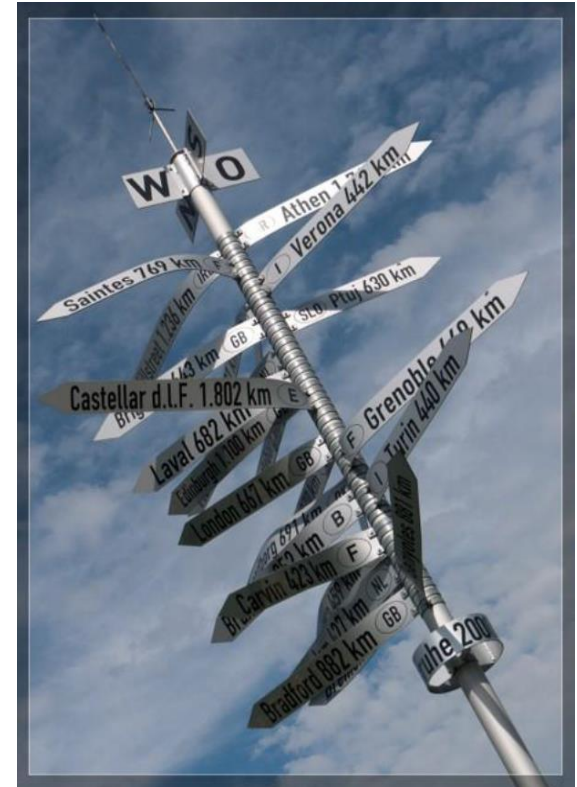
- Backtransformation of this data shows the loss of information!
- (+: backtransformed data, *: original data record)
- (Corresponds to projection of the data onto the axis described by the first principal component.)



Some final remarks to PCA

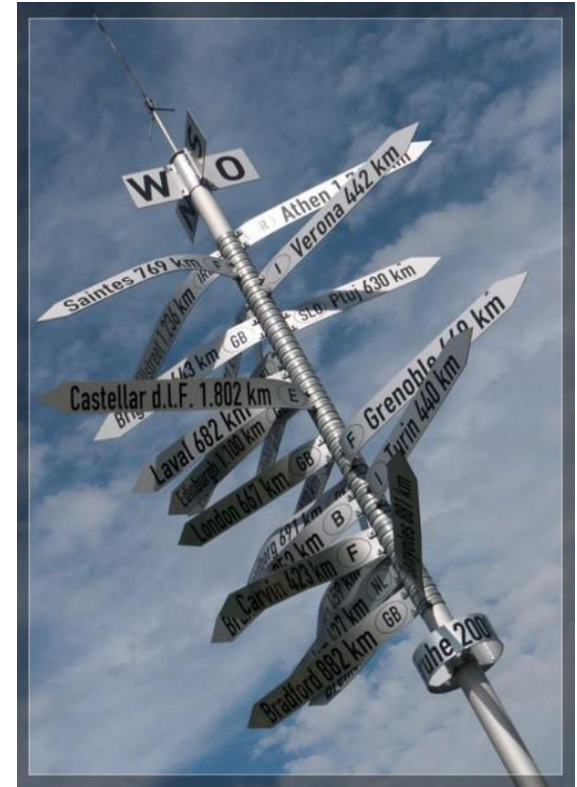
- Which criteria are used to determine a suitable number D' of main components for data reduction?
- The sum of the eigenvalues of the most important D' eigenvectors should account for a certain proportion (e.g. at least 0.75) of the sum of all D eigenvalues.
- Dimensions are omitted if the eigenvalues of the corresponding eigenvectors are lower than the average of all eigenvalues.
- The eigenvalues are represented according to the descending importance of the eigenvectors. If this curve becomes significantly flatter at one point, the corresponding dimensions are omitted (so-called elbow or knee criterion).
- ...

- Feature extraction
- Time-independent features
- Time-dependent features
- Feature selection
- Taxonomy of feature selection
- Feature selection algorithms
- Feature transformation
- Principle Component Analysis
- **Conclusion**
- Further readings



- Representations can be adjusted adaptively to the conditions of the time series.
- The Principal Component Analysis serves to transform the data, whereby information content is determined by variance.
- Piecewise forms of representation usually try a segmented approximation, e.g. via polynomials.
- Feature selection is used, among other things, to reduce the search space.
- A basic distinction is made between filters and wrappers
- This requires evaluation measures: Information, accuracy and consistency measures are the most important representatives here.

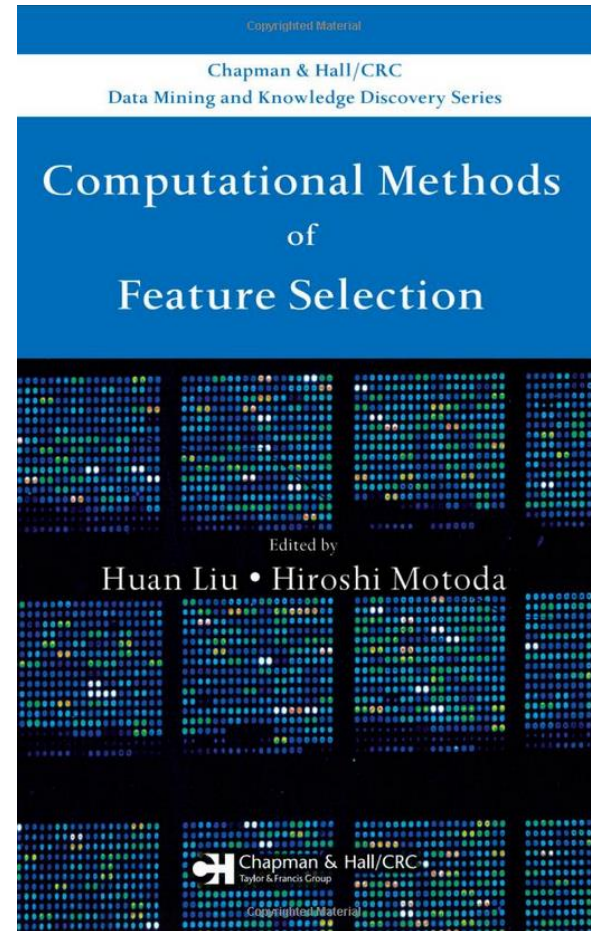
- Feature extraction
- Time-independent features
- Time-dependent features
- Feature selection
- Taxonomy of feature selection
- Feature selection algorithms
- Feature transformation
- Principle Component Analysis
- Conclusion
- Further readings



Further readings

Basic readings:

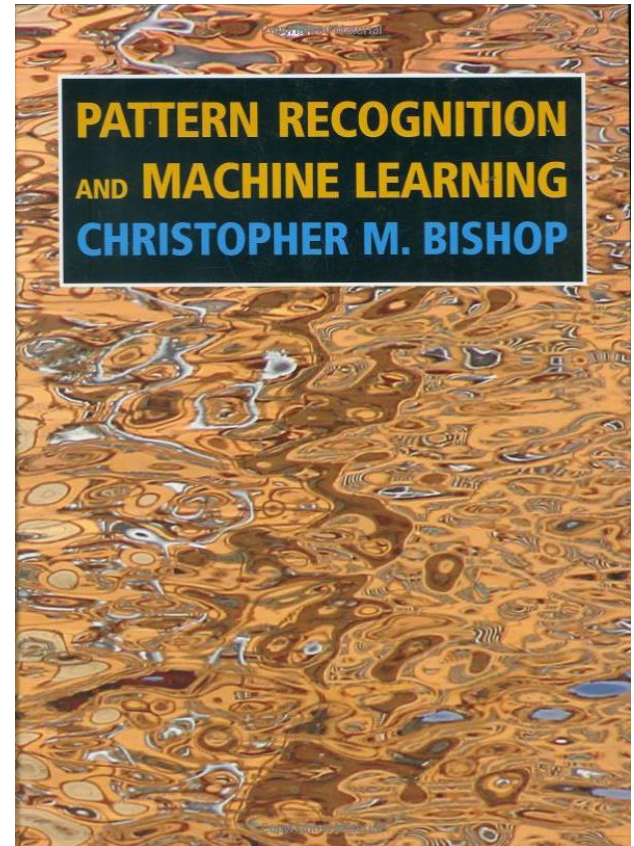
- Huan Liu, Hiroshi Motoda
- Computational Methods of Feature Selection
- Chapman & Hall/CRC Data Mining and Knowledge Discovery Series
- Knowledge Discovery Series
- 2007
- ISBN: 978-1584888789
- Contents and illustrations partly based on lecture “Intelligent Technical Systems” by Prof. Paul Lukowicz (DFKI Kaiserslautern)



Further readings (2)

Basic readings in general:

- Christopher M. Bishop
- Pattern Recognition and Machine Learning (Information Science and Statistics)
- Springer Verlag
- 2011, 2nd edition
- ISBN: 978-0387310732



Further readings (3)

- [Lin et al., 2003]: Lin, Keogh, Lonardi, Chiu, A Symbolic Representation of Time Series, with Implications for Streaming Algorithms 2003
- [Mitsa 2010]: Mitsa, Theophano. Temporal data mining. CRC Press, 2010.
- [Chakravarty 2000]: Chakravarty, Shahar, CAPSUL: A constraint-based specification of repeating patterns in time-oriented data, 2000
- [Fu 2008]: Fu, Chung, Luk, Ng, Representing financial time series based on data point importance 2008
- [Keogh et al., 2001]: Keogh, Chakrabarti, Mehrotra, Pazzani, Locally Adaptive Dimensionality Reduction for Indexing Large Time Series Databases 2001
- [Keogh 1998]: Keogh, Pazzani, An enhanced representation of time series which allows fast and accurate classification, clustering and relevance feedback 1998

End

- Questions....?