Intelligent Systems

Chapter 10: Classification

Winter Term 2019 / 2020

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

Introduction to Classification

- Goal:
  - Find a method to predict the class of observations.
- Learning:
  - Based on samples of known class (= label) – training patterns of the form $(x_1, \ldots, x_D, C_i)$
- In contrast to regression, labels are discrete (classes $C_1, \ldots, C_c$)
- Different methods
  - Decision Trees
  - Classification Rule Sets
  - Neuronal Networks
  - …

Introduction to Classification (2)

- In this lecture, only simple and basic classification methods are introduced
- Occasionally:
  - the assumption holds that the samples are distributed identically and indepently (iid)
  - one feature / a simple set of rules / a linear combination of features is enough for solving the classification problem

Ockham's razor

‘’Entia non sunt multiplicanda praeter necessitatem.”

→ ‘’There should not be made any assumptions beside the necessary.’’
(William of Ockham, 1287–1347)

Source: https://simple.wikipedia.org/wiki/File:William_of_Ockham.png

About this Chapter

Content

- Introduction to Classification
- 1-R Classifier
- Decision Trees
- Naïve Bayes Classification
- k-Nearest Neigbors
- Conclusion and further readings

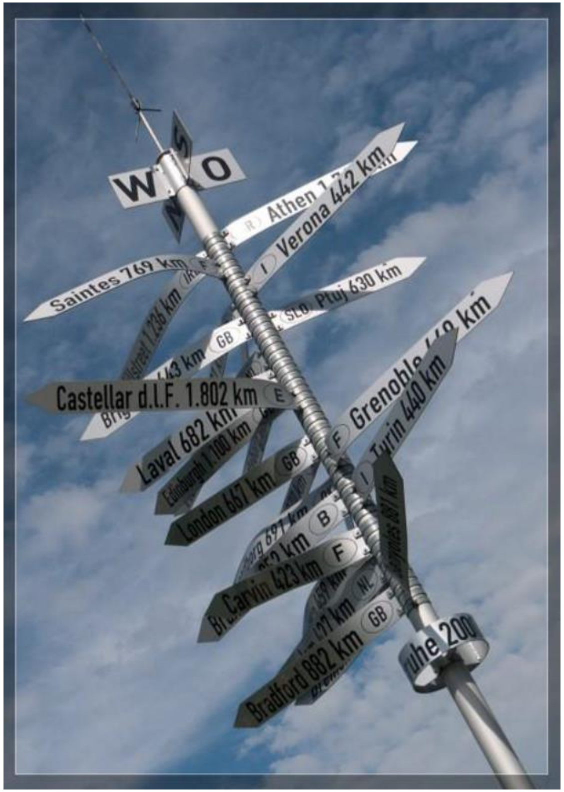
Goals

Students should be able to:

- Understand the concepts of learning
- Define the classification problem
- Explain Occam’s razor and the analogy to Machine Learning
- Explain for which tasks GMMs can be used in OC systems

Agenda

- Introduction to Classification
- **1-R Classifier**
- Decision Trees
- Naïve Bayes Classification
- k-Nearest Neigbors
- Combination of classifiers
- Conclusion and further readings



1-R Classifier

- Suitable for nominal features
  - Nominal features are in a discrete and finite value range and have no inherent ordering or preference structure
  - For example: gender (male or female), subject (economy, computer science, medicine, …), nationality (German, Italian, Austrian, British, …)
- Goal: Find a set of rules applied to one feature only
  - Set of rules correspond to a Decision Tree (see later) with one layer
- Inventor: Holte (University of Ottawa) 1993
  - Introduced in paper: Comparison of 16 benchmark data sets – similar performance as more complex Decision Trees
- Possible extension for ordinal features:
  - Ordinal features are in a finite value range with an ordering structure

1-R Classifier (2)

Algorithm 1-R Classifier:

- For all possible values of a feature:
  - Count the occurances of every class.
  - Find the most frequent class.
  - Produce a rule assigning the class to the feature value
- Calculate the failure rate of rules.
- Choose the rule of lowest failure rate

1-R Classifier (3)

Christian-Albrechts-Universität zu Kiel

Example: Playing golf?

Outlook	Temp	Humidity	Windy	Play
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Rainy	Mild	High	True	No

Attribute	Rules	Errors	Total errors
Outlook	Sunny → No	2/5	4/14
	Overcast → Yes	0/4	
	Rainy → Yes	2/5	
Temp	Hot → No*	2/4	5/14
	Mild → Yes	2/6	
	Cool → Yes	1/4	
Humidity	High → No	3/7	4/14
	Normal → Yes	1/7	
Windy	False → Yes	2/8	5/14
	True → No*	3/6	

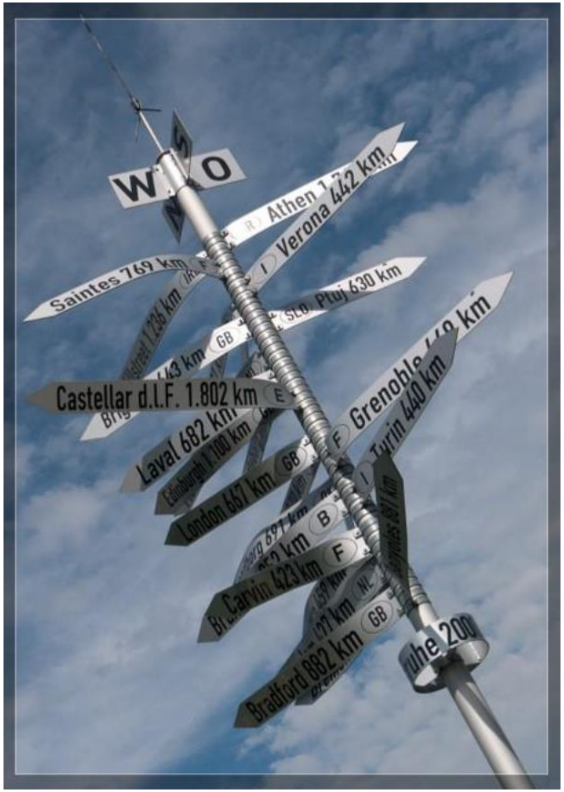
* Represents a preference in case of ties

Here, the rules of the features *humidity* or *outlook* are chosen.

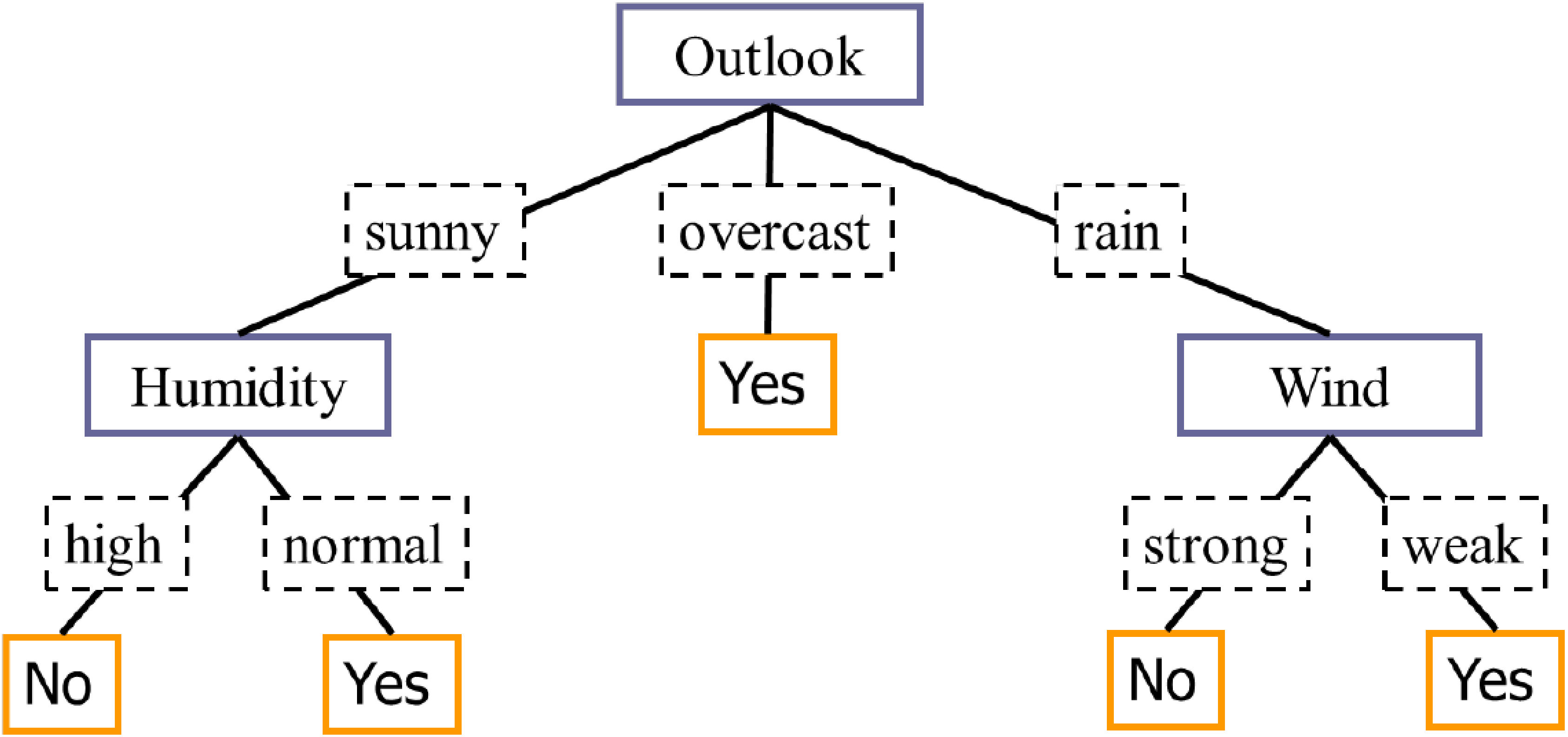
Agenda

Christian-Albrechts-Universität zu Kiel

- Introduction to Classification
- 1-R Classifier
- **Decision Trees**
- Naïve Bayes Classification
- k-Nearest Neigbors
- Combination of classifiers
- Conclusion and further readings

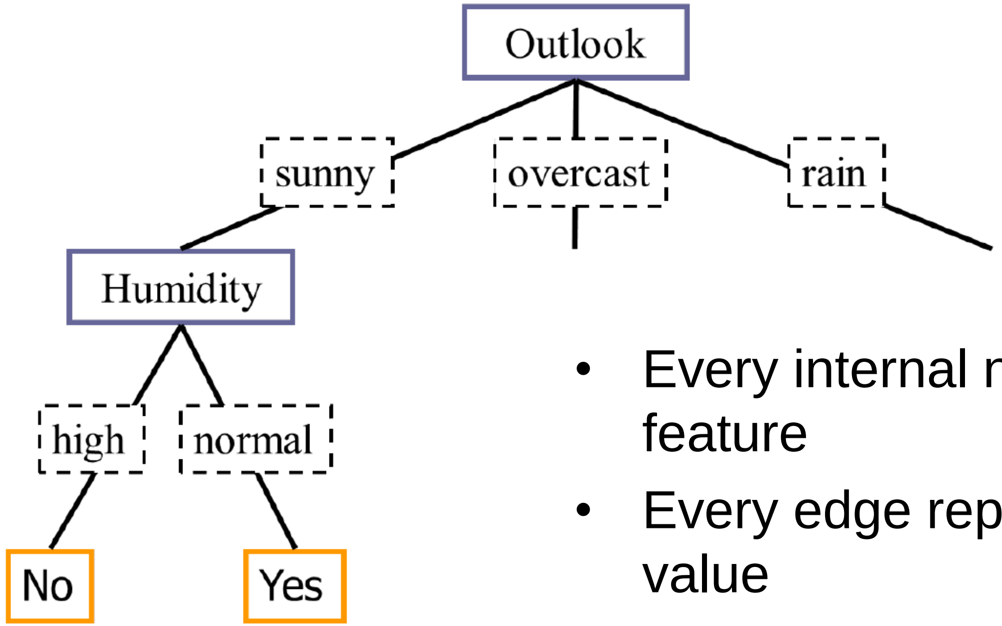


Playing golf: Yes/No?



Decision Trees (2)

Internal nodes, edges, leaf nodes



- Every internal node checks a feature
- Every edge represents a feature value
- Every leaf node represents a class assignement

Decision Trees (3)

Traversing decision trees

Algorithm:

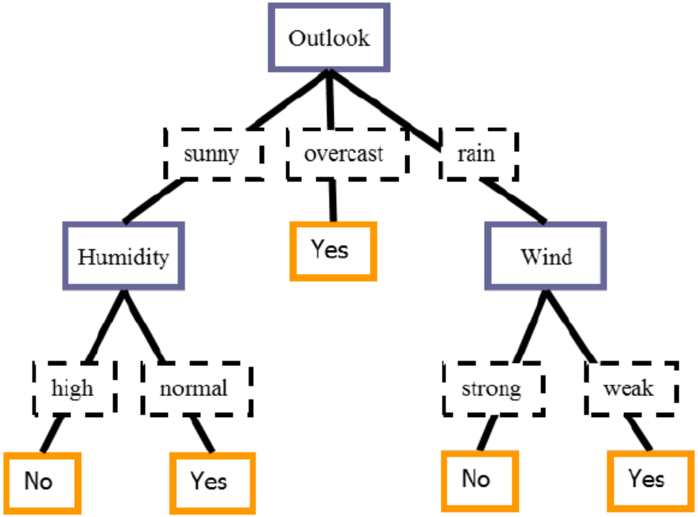
1) Begin with the root node
2) While current node is no leaf node
   - Answer the question current node
   - Follow edge with observed feature value to the next current node
3) Result can be read from the leaf node

Decision Trees (3)

Christian-Albrechts-Universität zu Kiel

Traversing – Example

- 4 features: outlook, temperature, humi
- Sample: [rain, cool, normal, strong]
- Outlook = rain → choose right edge
- Wind = strong → choose left edge
- Decision: No



Decision Trees (4)

Avertability

- Discrete and continuous features
- Noisy data
- The classification process shall be interpretable (rule extraction)

Construction of Decision Trees

- Manually: developping Decision Trees with the help of experts
  - Rules are often redundant, incomplete, or inefficient
  - Time-consuming and expensive process
- Induction: derive Decision Trees automatically from sample data (training data)

Decision Trees (5)

Methods of induction:

- Enumerative approach:
  - Produce all possible Decision Trees
  - Choose the tree with the minimal number of nodes

  → Optimal tree will be found

  → But: Very inefficient proceeding
- Heuristical approach:
  - Extend an existing tree with additional internal nodes
  - Terminate when stop criterion is fulfilled

  → More efficient

  → Optimal tree is not found generally

Decision Trees (6)

Decision Tree construction

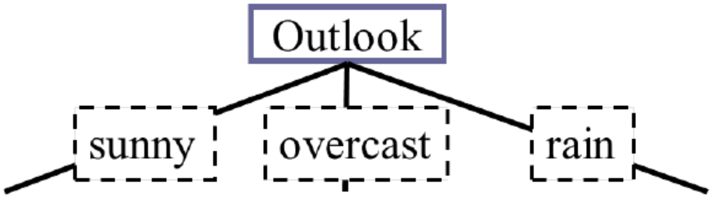
Simple algorithm:

1) Begin with an empty tree
2) Partition the training set recursively by selecting a single feature step by step
3) Stop when no more features are available or another stop criterion is fulfilled

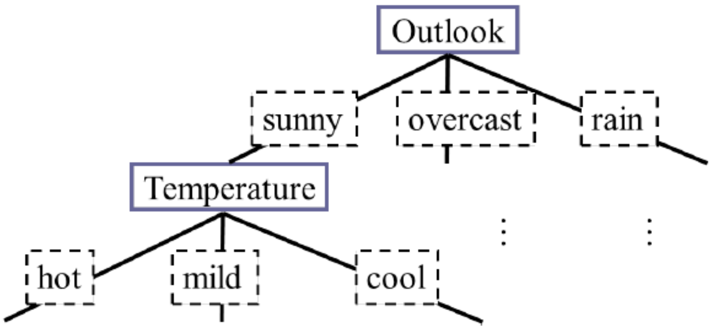
Decision Trees (7)

Applying the algorithm:

1) Feature: Outlook
   Possible feature values:
   sunny, overcast, rain



2) Feature: Temperature
   Possible feature values:
   hot, mild, cool



Decision Trees (8)

Christian-Albrechts-Universität zu Kiel

Order of the feature selection

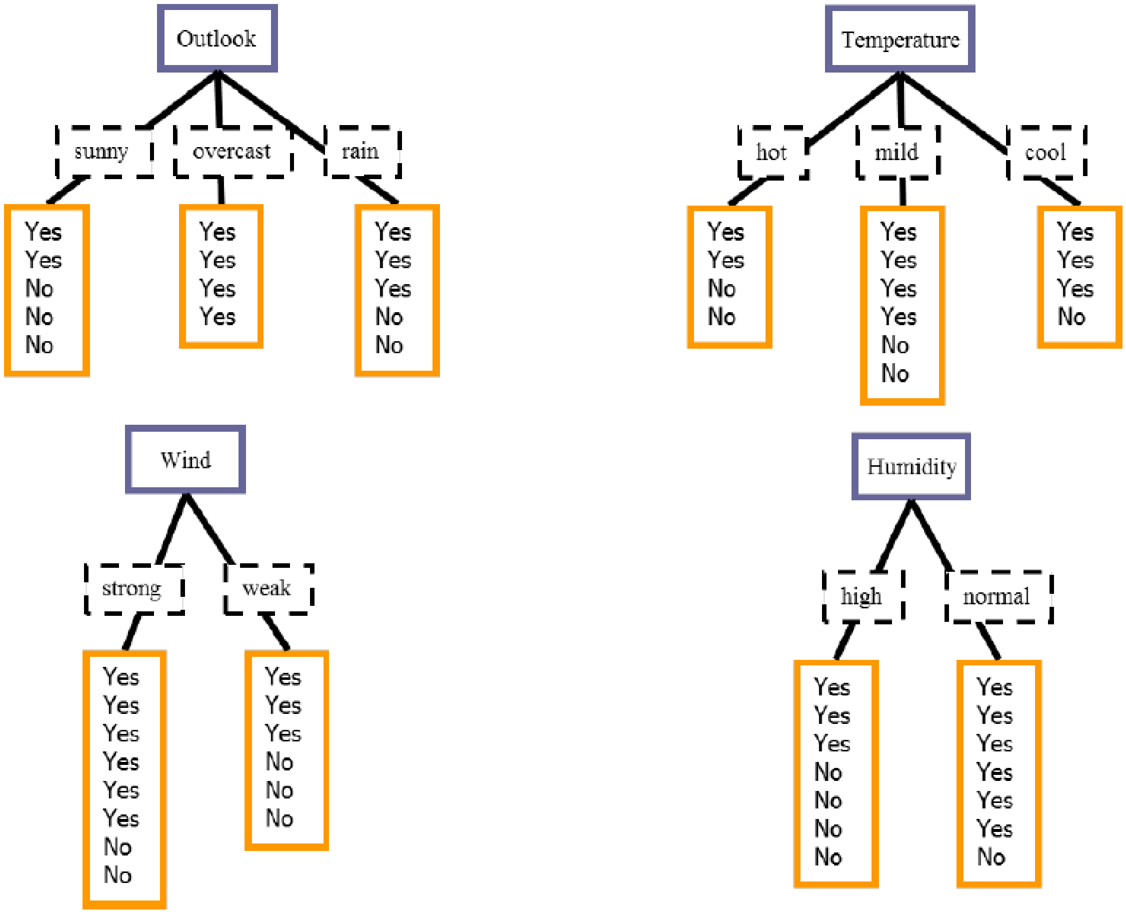
- What happens if one choose a different order?

  → You get a different tree.
- Which order is the best? Which feature shall be selected next?
- Splitting strategies:
  - Information gain
  - Gain ratio
  - Gini index

Decision Trees (9)

Christian-Albrechts-Universität zu Kiel

Splitting strategies – Which feature shall be selected next?



Decision Trees (10)

Christian-Albrechts-Universität zu Kiel

What is the best feature?

- The feature producing a minimal tree
- Heuristical: Choose the feature which produces the “purest“ class distribution (e.g. only Yes or only No)

Splitting strategy Information Gain (IG)

- Popular method
- Already known; feature selection
- Property: The more average “purity“ the partitioned sub sets have, the higher is the IG
- Strategy: Choose the feature with the highest IG

Decision Trees (11)

Christian-Albrechts-Universität zu Kiel

Repitition: Measure of Information – Information Gain (IG)

Partition the set D with the feature X in p subsets D_{X_j} with $j = 1, \ldots, p$.
Additionally, there are classes c_i with $i = 1, \ldots, d$.
Information Gain (IG) of feature X:

$$IG(X) \stackrel{\text{def}}{=} E(D) - \sum_{j=1}^{p} \frac{\left|D_{X_j}\right|}{|D|} \cdot E\left(D_{X_j}\right)$$

$$E\left(D_{X_j}\right) \stackrel{\text{def}}{=} -\sum_{i=1}^{d} P_{D_{X_j}}(c_i) \cdot \log_2 P_{D_{X_j}}(c_i)$$

$$E(D) \stackrel{\text{def}}{=} -\sum_{i=1}^{d} P_D(c_i) \cdot \log_2 P_D(c_i)$$

Decision Trees (12)

Christian-Albrechts-Universität zu Kiel

Example: Feature Outlook

- Outlook = sunny : 5 samples, 3x No, 2x Yes

$$E(Outlook = sunny) = -\frac{3}{5} \cdot \log_2 \frac{3}{5} - \frac{2}{5} \cdot \log_2 \frac{2}{5} = 0.971$$

- Outlook = overcast : 4 samples, 0x No, 4x Yes

$$E(Outlook = overcast) = -\frac{0}{4} \cdot \log_2 \frac{0}{4} - \frac{4}{4} \cdot \log_2 \frac{4}{4} = 0$$

- Outlook = rain : 5 samples, 2x No, 3x Yes

$$E(Outlook = rain) = -\frac{2}{5} \cdot \log_2 \frac{2}{5} - \frac{3}{5} \cdot \log_2 \frac{3}{5} = 0.971$$

Decision Trees (13)

Christian-Albrechts-Universität zu Kiel

Example: Feature Outlook

- Entropy of the whole data set D: 14 samples, 5x No, 9x Yes

$$E(D) = -\frac{5}{14} \cdot \log_2 \frac{5}{14} - \frac{9}{14} \cdot \log_2 \frac{9}{14} = 0.940$$

- Hence:

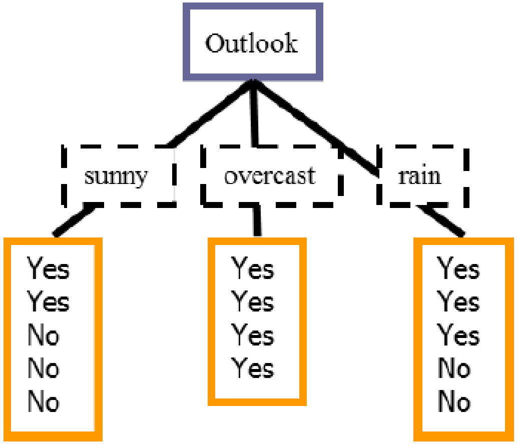
$$\begin{aligned} IG(Outlook) = {} & E(D) \\ & - \frac{5}{14} E(Outlook = sunny) \\ & - \frac{4}{14} E(Outlook = overcast) \\ & - \frac{5}{14} E(Outlook = rain) \\ = {} & 0.247 \end{aligned}$$

Decision Trees (14)

Results for all features:

- $IG(Outlook) = 0.247$
- $IG(Temperature) = 0.029$
- $IG(Humidity) = 0.152$
- $IG(Wind) = 0.048$

Therefore, Outlook is the first partitioning feature



Decision Trees (15)

Christian-Albrechts-Universität zu Kiel

What comes next?

- Consider left branch: Outlook = sunny
- New data set D:

Outlook	Temperature	Humidity	Wind	Play
sunny	hot	high	weak	No
sunny	hot	high	strong	No
sunny	mild	high	weak	No
sunny	cool	normal	weak	Yes
sunny	mild	normal	strong	Yes

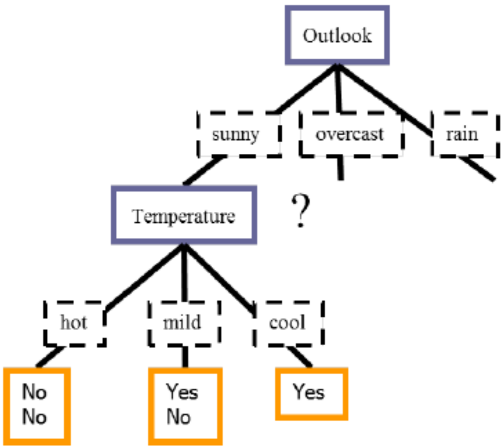
- Entropy of the new data set:

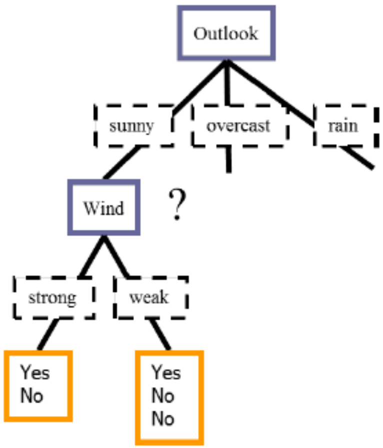
$$E(D) = -\frac{3}{5} \cdot \log_2 \frac{3}{5} - \frac{2}{5} \cdot \log_2 \frac{2}{5} = 0.971$$
$$= E(Outlook = sunny)$$

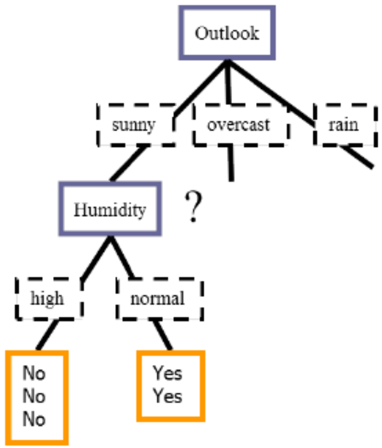
Christian-Albrechts-Universität zu Kiel

Other possible partitions

- Only 3 features: temperature, humidity, and wind







$IG(Temp.) = 0.571$ $IG(Wind) = 0.020$ $IG(Humidity) = 0.971$

Decision Trees (16)

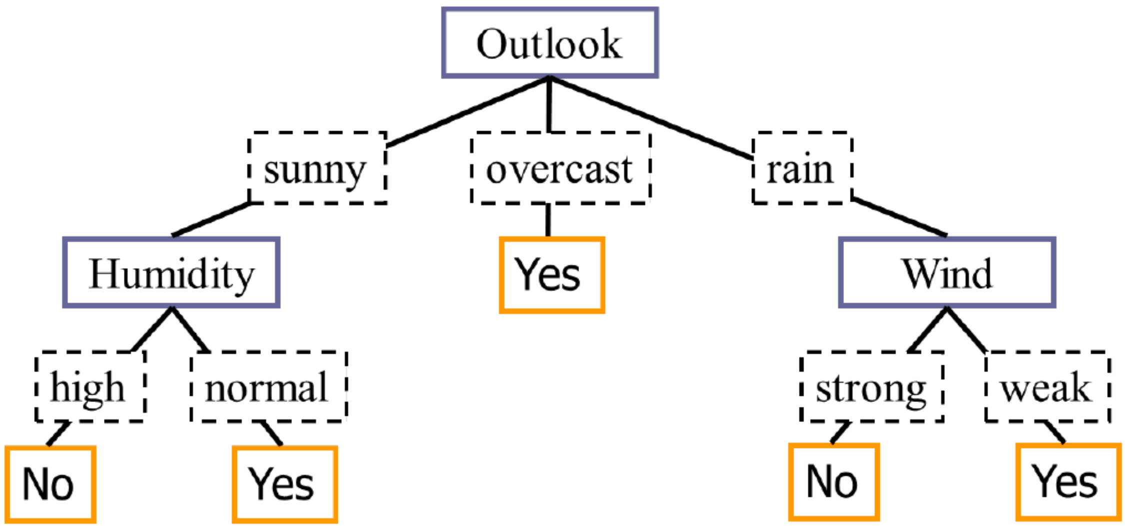
- Select feature humidity, because it corresponds to the largest Information Gain (0.971)
- No further separation of this branch necessary (entropy is respectively zero)

Next steps

- Analogously with Outlook = overcast and Outlook = rain
- Recursive steps until all features are treated or IG=0

Decision Trees (17)

Result:



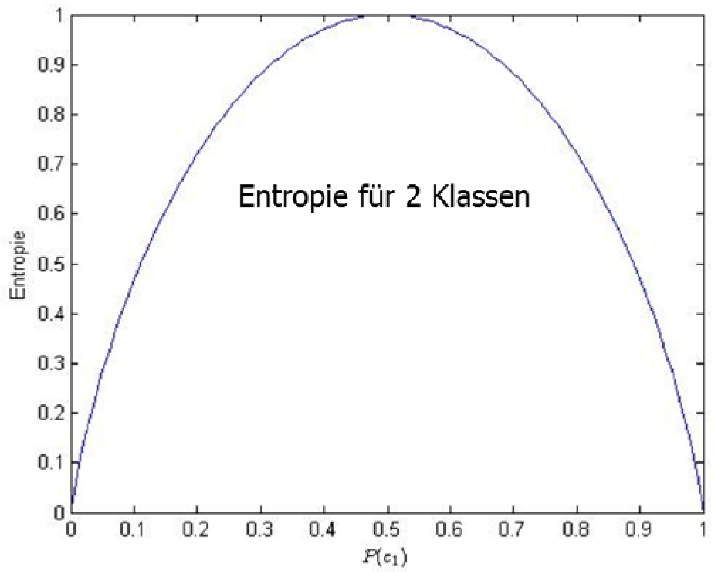
Remarks:

At the end, there might be nodes left containing more than one class (e. g. same samples with different class assignment)

Christian-Albrechts-Universität zu Kiel

Properties of Entropy

- Entropy is maximal if the classes are equally frequent
- If only one class is left, the entropy is zero (e.g. see $E(Outlook = overcast) = 0$)
- **Problem:** What happens with features of large range of values?



Christian-Albrechts-Universität zu Kiel

Extrem case: Column of indices

Index	Outlook	Temperature	Humidity	Wind	Play
D1	sunny	hot	high	weak	No
D2	sunny	hot	high	strong	No
D3	overcast	hot	high	weak	Yes
D4	rain	mild	high	weak	Yes
D5	rain	cool	normal	weak	Yes
D6	rain	cool	normal	strong	No
D7	overcast	cool	normal	strong	Yes
D8	sunny	mild	high	weak	No
D9	sunny	cool	normal	weak	Yes
D10	rain	mild	normal	weak	Yes
D11	sunny	mild	normal	strong	Yes
D12	overcast	mild	high	strong	Yes
D13	overcast	hot	normal	weak	Yes
D14	rain	mild	high	strong	No

Decision Trees (20)

Compute IG for feature Index

$$E(Index = D1) = -\frac{1}{1} \cdot \log_2 \frac{0}{1} \cdot \backslash\log_2 \frac{0}{1} = 0$$

$$\vdots$$

$$E(Index = D14) = -\frac{1}{1} \cdot \log_2 \frac{1}{1} - \frac{0}{1} \cdot \log_2 \frac{0}{1} = 0$$

$$\begin{aligned} IG(Index) = E(D) & \\ & - \frac{1}{14} \cdot E(Index = D1) \\ & \vdots \\ & - \frac{1}{14} \cdot E(Index = D14) \\ = 0.940 & \end{aligned}$$

Decision Trees (21)

Christian-Albrechts-Universität zu Kiel

Resulting IG for alle features

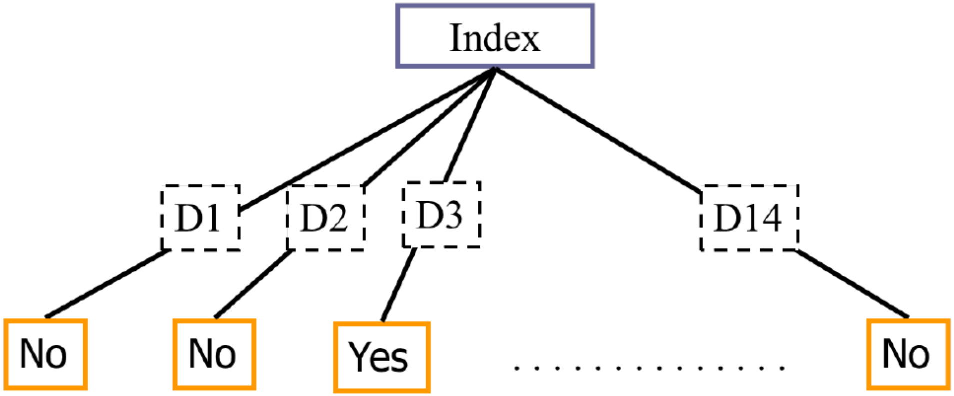
- $IG(Index) = 0.940$
- $IG(Outlook) = 0.257$
- $IG(Temperature) = 0.029$
- $IG(Humidity) = 0.152$
- $IG(Wind) = 0.048$

Index is always selected.

Decision Trees (22)

Christian-Albrechts-Universität zu Kiel

Corresponding tree:



- Bias: Features with a high number of distinct values are always selected
- Is this senseful?

→ No! Good classification for training data but worse for unkown samples (new index values).

Decision Trees (23)

Christian-Albrechts-Universität zu Kiel

Overfitting:

A tree b is overfitted, if there is another tree b' with:

$$error_{train}(b) < error_{train}(b')$$

and

$$error_{test}(b) > error_{test}(b') \quad ,$$

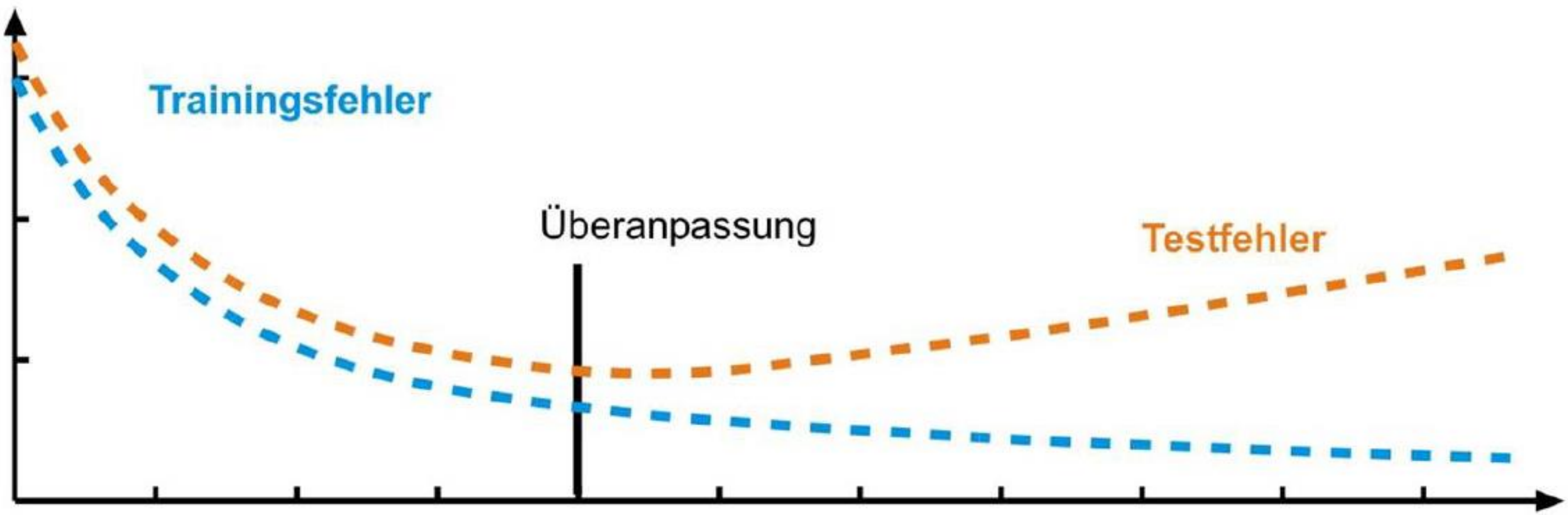
where

- $error_{train}(b)$ Classification error of tree b on training data
- $error_{train}(b')$ Classification error of tree b' on training data
- $error_{train}(b)$ Classification error of tree b on test data
- $error_{train}(b')$ Classification error of tree b' on test data

Decision Trees (24)

Christian-Albrechts-Universität zu Kiel

Overfitting – Example



Number of feature values

Decision Trees (25)

Gain Ratio

- Modification in order to reduce bias provoked by features with lots of distinct values
- Gain Ratio (GR) concerns the number and size of branches of a node

Concrete

- IG is corrected by regarding the information of the branching itself (how much information is necessary to say to which branch a sample belongs?)
- Intrinsic Information (II)

Decision Trees (26)

Christian-Albrechts-Universität zu Kiel

Intrinsic Information

- II: Entropy of the distribution of samples on branches
- II for the partition of a dataset D given by the feature X:

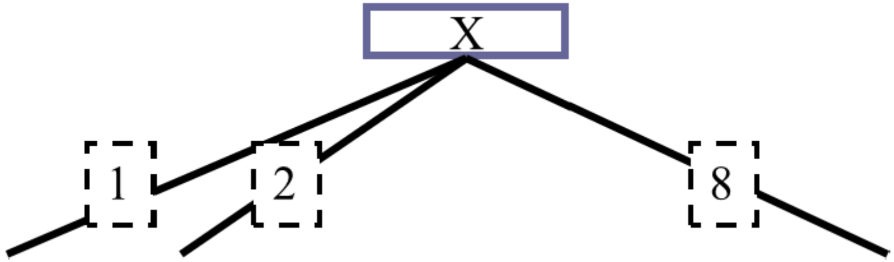
$$II(X) = -\sum_{j=1}^{p} \frac{|D_j|}{|D|} \log_2 \frac{|D_j|}{|D|}$$

- p: Number of different feature values of a feature X
- D_j: Subsets of D with respectively the same feature value of X

Christian-Albrechts-Universität zu Kiel

Intrinsic Information – Example

- Simple Tree: 1 feature, 1 node, 8 samples, 8 possible feature values



- How much information is necessary to encode the feature value of a certain sample?

$$II(X) = -\sum_{j=1}^{8} \frac{1}{8} \log_2 \frac{1}{8} = 3\ (Bit)$$

Decision Trees (28)

Christian-Albrechts-Universität zu Kiel

Example: II for feature index

- Partitioning into 14 subsets
- Subset size: 1

$$
\begin{aligned}
II(Index) &= -\sum_{j=1}^{14} \frac{1}{14} \log_2 \frac{1}{14} \\
&= 14 \cdot \left(-\frac{1}{14} \log_2 \frac{1}{14} \right) \\
&= 3.807
\end{aligned}
$$

Decision Trees (29)

Christian-Albrechts-Universität zu Kiel

Example: II for feature outlook

- Partitioning into 3 subsets
- Subset size: 5 (sunny), 4 (overcast), 5 (rain)

$$
\begin{aligned}
II(Outlook) &= -\frac{5}{14}\cdot\log_2\frac{5}{14} \\
&\quad -\frac{4}{14}\cdot\log_2\frac{4}{14} \\
&\quad -\frac{5}{14}\cdot\log_2\frac{5}{14} \\
&= 1.577
\end{aligned}
$$

Decision Trees (30)

Christian-Albrechts-Universität zu Kiel

Gain Ratio Definition

- Correction (Normalisation) of the IG
- Gain Ratio of a feature X

$$GR(X) = \frac{IG(X)}{II(X)}$$

- Strategy: Select feature with highest Gain Ratio

Decision Trees (31)

Christian-Albrechts-Universität zu Kiel

Gain Ratio for golf example:

- $GR(Index) = \frac{0.940}{3.807} = 0.247$
- $GR(Outlook) = \frac{0.247}{1.577} = 0.157$
- $GR(Temp.) = \frac{0.029}{1.362} = 0.021$
- $GR(Humidity) = \frac{0.152}{1} = 0.152$
- $GR(Wind) = \frac{0.048}{3.958} = 0.050$

Observations:

- Original data set (without index): Outlook is still the best feature
- With index: despite correction, feature index has the largest GR
- Solution: Test procedure detecting special features like index
- Then, why at least GR?
  → Works for features with lots of distinct values – only struggles in the extreme case of index columns

Decision Trees (32)

Christian-Albrechts-Universität zu Kiel

Extension to numerical features

- So far only nominal and discrete features were taken into account
- Not applicable for a practical use case
  → E. g. sensor data such as length [m], weight [kg], speed [km/h]
- Extension required: Numerical features has to be processed differently

Decision Trees (33)

Christian-Albrechts-Universität zu Kiel

Example: weather data with numerical feature

So far: Temperature values categorisable into (hot, mild, cold)
Now: Integer values representing degrees Celsius

Outlook	Temperature	Humidity	Wind	Play
sunny	85	high	weak	No
sunny	80	high	strong	No
overcast	83	high	weak	Yes
rain	75	high	weak	Yes
rain	68	normal	weak	Yes
rain	65	normal	strong	No
overcast	64	normal	strong	Yes
sunny	72	high	weak	No
sunny	69	normal	weak	Yes
rain	70	normal	weak	Yes
sunny	75	normal	strong	Yes
overcast	72	high	strong	Yes
overcast	81	normal	weak	Yes
rain	71	high	strong	No

Decision Trees (33)

Approach: Formation of intervals

- Sorting of values
- Formation of “new features“ introducing interval borders
- Then: Apply Splitting Strategy

Decision Trees (33)

Christian-Albrechts-Universität zu Kiel

Example: feature temperature

1) Sort feature values
2) Determine interval border, for example at 71.5
   - Temperature < 71.5 : 2x No, 4x Yes
   - Temperature ≥ 71.5 : 3x No, 5x Yes
3) Calculate IG for interval border $split = 71.5$

Temp.	64	65	68	69	70	71	72	72	75	75	80	81	83	85
Play?	Yes	No	Yes	Yes	Yes	No	No	Yes	Yes	Yes	No	Yes	Yes	No

Decision Trees (34)

Christian-Albrechts-Universität zu Kiel

IG for interval border $split = 71.5$

$$E(Temperature < 71.5) = -\frac{2}{6} \cdot \log_2 \frac{2}{6} - -\frac{4}{6} \cdot \log_2 \frac{4}{6} = 0.918$$

$$E(Temperature \geq 71.5) = -\frac{3}{8} \cdot \log_2 \frac{3}{8} - -\frac{5}{8} \cdot \log_2 \frac{5}{8} = 0.954$$

Therefore:

$$\begin{aligned} IG(split = 71.5) &= E(D) \\ &- \frac{6}{14} \cdot 0.918 \\ &- \frac{8}{14} \cdot 0.954 \\ &= 0.940 - 0.939 = 0.001 \end{aligned}$$

Decision Trees (35)

Christian-Albrechts-Universität zu Kiel

IG for all possible interval borders

64	65	68	69	70	71	72	72	75	75	80	81	83	85
Yes	No	Yes	Yes	Yes	No	No	Yes	Yes	Yes	No	Yes	Yes	No

- Calculate IG for all possible borders
- Define border with maximal IG
- Maximal IG corresponds to IG for feature temperature

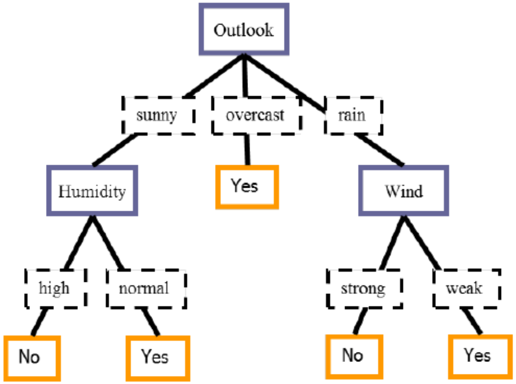
Optimisation

- Do we really need to score all borders?
- No, cause borders within a target class cannot be possible
  - Only 7 interval borders left instead of 13

Decision Trees (36)

Christian-Albrechts-Universität zu Kiel

Rule extraction from trees



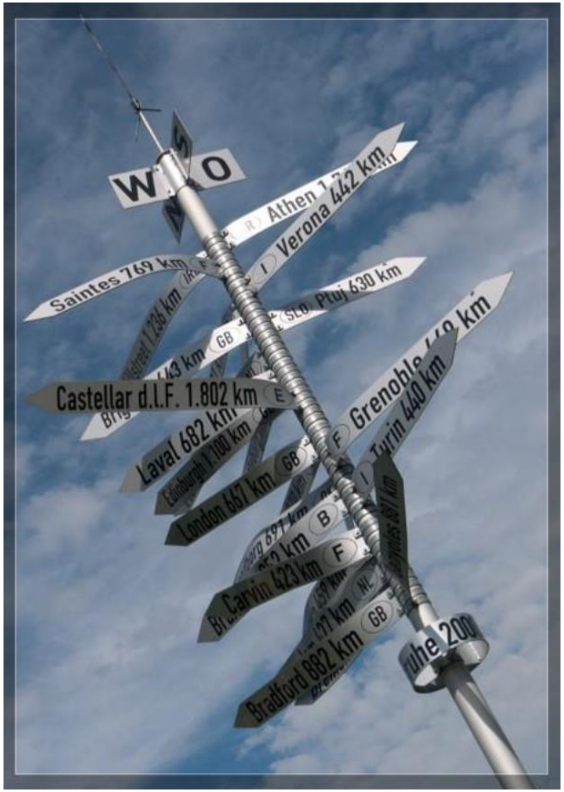
- **IF** … **THEN** … rules
- General:
- **IF** $test_1$ **AND** … **AND** $test_n$ **THEN** Decision C
- One rule per leaf

- **IF** Outlook=sunny **AND** Humidity=high **THEN** Decision No
- **IF** Outlook=sunny **AND** Humidity=normal **THEN** Decision Yes
- **IF** Outlook=overcast **THEN** Decision Yes
- **IF** Outlook=rain **AND** Wind=strong **THEN** Decision No
- **IF** Outlook=rain **AND** Wind=weak **THEN** Decision Yes

Agenda

Christian-Albrechts-Universität zu Kiel

- Introduction to Classification
- 1-R Classifier
- Decision Trees
- **Naïve Bayes Classification**
- k-Nearest Neigbors
- Combination of classifiers
- Conclusion and further readings



Naïve Bayes Classification

Christian-Albrechts-Universität zu Kiel

- Takes all features into account (in contrast to 1-R Classifier)
- Probabilistic classifier:

$$P(\mathcal{C}|x_1, \dots, x_D)$$

Source: https://simple.wikipedia.org/wiki/File:William_of_Ockham.png

- Based on Bayes theorem by Thomas Bayes (1702-1761)
- Assumption: All features are equally important
- Originally for nominal features, but can be modified for ordinal and other features

Naïve Bayes Classification (2)

Christian-Albrechts-Universität zu Kiel

Reminder: Probability theory

- Single random variable A – corresponding probability $P(A)$

- Here more interesting: multiple random variables A, B, ...

- **Compound probability**: $P(A \cap B)$

  The probability that A and B commonly occurr.

  (Alternative notation: $P(A, B)$)

- **Conditional probability**: $P(A|B)$

  The probability of occurance of event A under the condition that event B was previously observed. If one assumes event B, the probability of observing A is $P(A|B)$, hence it is not a (logical) condition for A

Naïve Bayes Classification (3)

Christian-Albrechts-Universität zu Kiel

Reminder: Probability theory

- For arbitrary events A and B and $P(B) > 0$ it holds that:

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

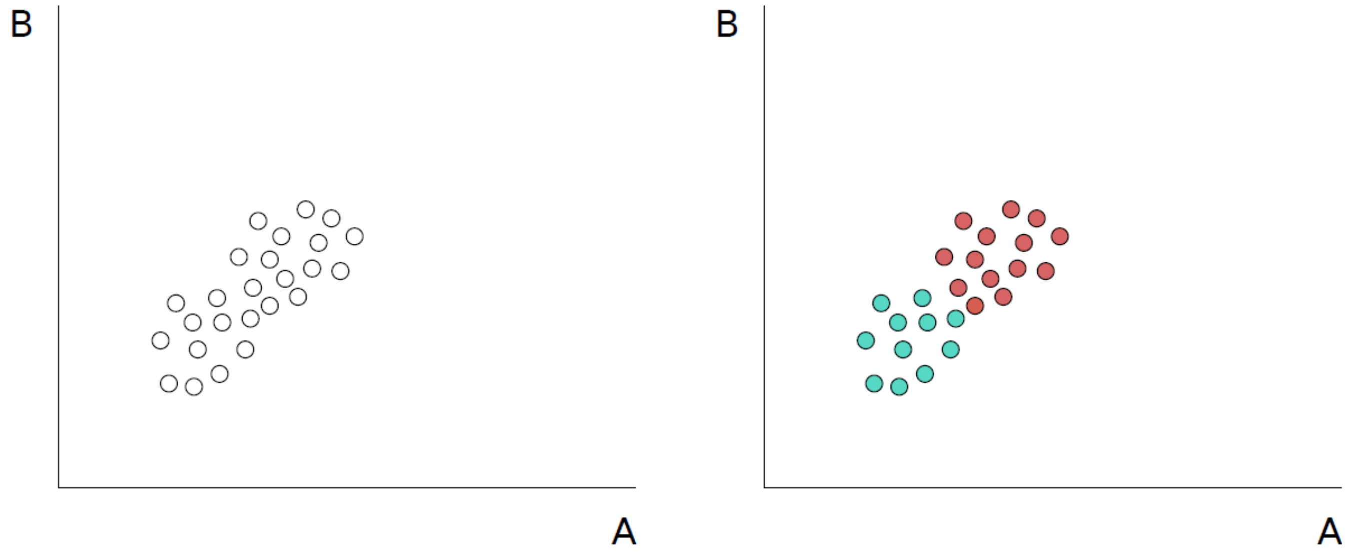
- By transforming the formula we derive the **mutliplication axiom**

$$P(A \cap B) = P(A|B)P(B) = P(B|A)P(A)$$

- **Independence**: If A and B are independent from each other, then

$$P(A|B) = P(A)$$
$$P(A \cap B) = P(A)$$

Naïve Bayes Classification (4)



- Conditional independent: Given C A and B are conditional independent, if it holds that $P(A, B|C) = P(A|C) \cdot P(B|C)$
  → Attention: Conditional independence does not imply independence

Naïve Bayes Classification (5)

Christian-Albrechts-Universität zu Kiel

- Bayes theorem:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

- $P(A)$: A priori probability for event A
- $P(B|A)$: Probability for event B given occurrance of A (also known as likelihood)
- $P(A|B)$: A posteriori probability of event A
- $P(B)$: Evidence
- Usage: Roughly spoken, Bayes theorem allows the inversion of conclusions
  - Calculation of $P(Event|Cause)$ often easy
  - But usually required: $P(Cause|Event)$
  - Therefore: “Exchange“ of arguments

Naïve Bayes Classification (6)

For countable many events $A_i (i = 1, \ldots N)$ the Bayes Theorem can be extended to

$$P(A_i|B) = \frac{P(B|A_i)P(A_i)}{P(B)} = \frac{P(B|A_i)P(A_i)}{\sum_{i=1,\ldots,N} P(B|A_i)P(A_i)}$$

Whereas the relation

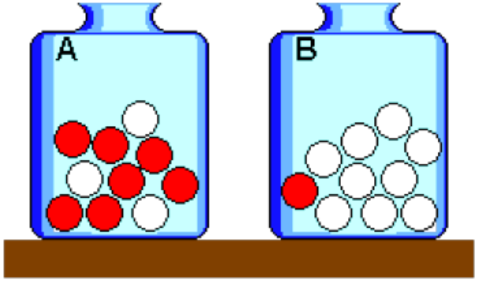
$$\begin{aligned} P(B) &= P(A_1 \cap B) + P(A_2 \cap B) + \cdots \\ &= P(B|A_1)P(A_1) + P(B|A_2)P(A_2) + \cdots \end{aligned}$$

is denoted as the law of total probability.

Naïve Bayes Classification (7)

Example:

- A Ball is randomly drawn from an (a priori uniformly random) urn (A or B) with red (R) and white (W) balls.
- One may ask oneself what the probability is for having drawn a red ball (R) from urn A: $P(A|R)$

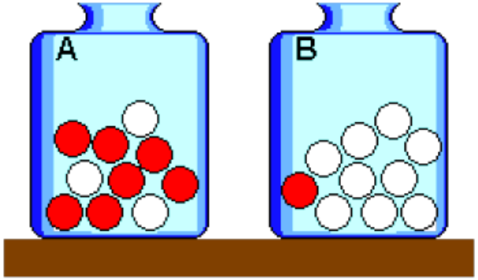


[Quelle: de.wikipedia.org]

Naïve Bayes Classification (8)

- $P(A) = P(B) = \frac{1}{2}$
- $P(R|A) = \frac{7}{10}$ (There are 7 red balls in urn A)
- $P(R|B) = \frac{1}{10}$ (There is 1 ball in urn B)
- $P(R) = P(R|A) \cdot P(A) + P(R|B) \cdot P(B)$

$$= \frac{7}{10} \cdot \frac{1}{2} + \frac{1}{10} \cdot \frac{1}{2} = \frac{2}{5}$$ (total probability)



[Quelle: de.wikipedia.org]

Naïve Bayes Classification (9)

Christian-Albrechts-Universität zu Kiel

- Application of the Bayes theorem for classification

$$P(C|x_1, \dots, x_D) = \frac{P(x_1, \dots, x_D|C) \cdot P(C)}{P(x_1, \dots, x_D)}$$

- Likelihood $P(x_1, \dots, x_D|C)$ and class-a-priori probability C
  - In principle: Determinable from training data (counting, calculate ratios)
  - Corresponds to maximum likelihood estimators of the parameters
- Evidence $P(x_1, \dots x_D|C)$ of occurance of the sample $(x_1, \dots, x_D)$ – normalisation factor
  - Actually: Approximately derivable from the training data
  - But: Not relevant for the classification decision dsaf
  - Because: Independet from C – hence constant for all classes
  - Absolute value of $P(C|x_1, \dots, x_D)$ not important but inter-class difference
  - Assignment to the class with maximum value

Naïve Bayes Classification (10)

- So far: no naïve assumption/restriction introduced
  - Fundamental mathematical/statistical foundation
  - Why then the name?
- Calculation of $P(x_1, \ldots, x_D|\mathcal{C})$
  - Number of free parameters $\mathcal{O}(K^D \cdot C)$
  - Where K is the average number of distinct feature values of a feature
  - In typical realistic applications: combinatorial explosion
- Therefore: "Naïve" assumption of conditional independence of the features of a class

$$P(x_1, \ldots, x_D|\mathcal{C}) = \prod_{i=1}^{D} P(x_i|\mathcal{C})$$

  - Only $\mathcal{O}(K^D \cdot C)$ parameters left to determine

Naïve Bayes Classification (11)

Christian-Albrechts-Universität zu Kiel

Example:

Outlook	Temp	Humidity	Windy	Play
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Rainy	Mild	High	True	No

Calculation of the probabilities:

- A priori possibility
  - $P(Play = Yes) = ?$
  - $P(Play = No) = ?$
- For samples:
  - $P(Outlook = Sunny|Play = Yes)$
  - $P(Outlook = Rainy|Play = Yes)$
  - $P(Outlook = Sunny|Play = No)$
  - $P(Outlook = Rainy|Play = No)$

Naïve Bayes Classification (12)

Example:

Outlook			Temperature			Humidity			Windy			Play	
	Yes	*No*		*Yes*	*No*		*Yes*	*No*		*Yes*	*No*	*Yes*	*No*
Sunny	2	3	Hot	2	2	High	3	4	False	6	2	9	5
Overcast	4	0	Mild	4	2	Normal	6	1	True	3	3		
Rainy	3	2	Cool	3	1								
Sunny	2/9	3/5	Hot	2/9	2/5	High	3/9	4/5	False	6/9	2/5	9/14	5/14
Overcast	4/9	0/5	Mild	4/9	2/5	Normal	6/9	1/5	True	3/9	3/5		
Rainy	3/9	2/5	Cool	3/9	1/5								

A new day starts with an “Event“…

Outlook	Temp.	Humidity	Windy	Play
Sunny	Cool	High	True	?

Naïve Bayes Classification (13)

Christian-Albrechts-Universität zu Kiel

Example:

Outlook			Temperature			Humidity			Windy			Play	
	Yes	*No*		*Yes*	*No*		*Yes*	*No*		*Yes*	*No*	*Yes*	*No*
Sunny	2	3	Hot	2	2	High	3	4	False	6	2	9	5
Overcast	4	0	Mild	4	2	Normal	6	1	True	3	3		
Rainy	3	2	Cool	3	1								
Sunny	2/9	3/5	Hot	2/9	2/5	High	3/9	4/5	False	6/9	2/5	9/14	5/14
Overcast	4/9	0/5	Mild	4/9	2/5	Normal	6/9	1/5	True	3/9	3/5		
Rainy	3/9	2/5	Cool	3/9	1/5								

A new day starts with an “Event“…

Outlook	Temp.	Humidity	Windy	Play
Sunny	Cool	High	True	?

Likelihood of the two classes

For “yes” = $2/9 \times 3/9 \times 3/9 \times 3/9 \times 9/14 = 0.0053$

For “no” = $3/5 \times 1/5 \times 4/5 \times 3/5 \times 5/14 = 0.0206$

Conversion into a probability by normalization:

P(“yes”) = $0.0053 / (0.0053 + 0.0206) = 0.205$

P(“no”) = $0.0206 / (0.0053 + 0.0206) = 0.795$

Naïve Bayes Classification (14)

Christian-Albrechts-Universität zu Kiel

- Event E (fix values for 4 features):

Outlook	Temp.	Humidity	Windy	Play
Sunny	Cool	High	True	?

$$P(yes|E) = \frac{P(E|yes) \cdot P(yes)}{P(E)}$$
$$= P(outlook$$
$$= sunny|yes \cdot P(temperature = cool|yes) \cdot P(humidity = high|yes)$$
$$\cdot P(windy = true|yes) \cdot \frac{P(yes)}{P(E)}$$
$$= \frac{\frac{2}{9} \cdot \frac{3}{9} \cdot \frac{3}{9} \cdot \frac{3}{9} \cdot \frac{9}{14}}{P(E)}$$

Remark: In comparison to $P(no|E)$, $P(E)$ does not necessarily have to be calculated.

Naïve Bayes Classification (15)

Christian-Albrechts-Universität zu Kiel

Questions and Answers:

- What shall we do if a feature value does not appear for every class (probilities would vanish)?
  - Addition of a constant value $\alpha > 0$ (e.g. see *Laplace Smoothing)*
  - Generally, for a categorical dimension X with K possible distinct feature values $1, \ldots, K$ and N observations it holds

$$P_{Lap}(X = i) = \frac{|X = i| + \alpha}{N + K \cdot \alpha} \qquad k \in 1, \ldots, K$$

- How shall we treat missing values?
  - This feature will not be considered for the calculation of the dependent probability

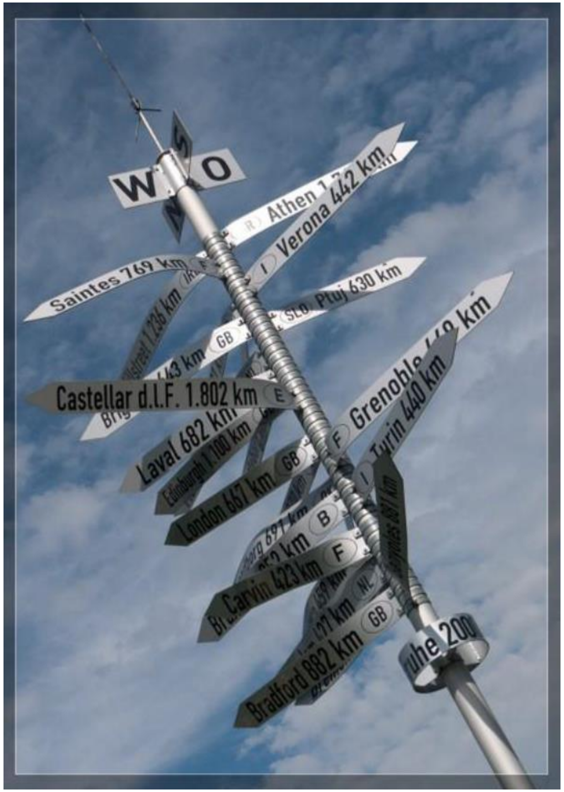
Naïve Bayes Classification (16)

Christian-Albrechts-Universität zu Kiel

- Why does the Naïve Bayes Classification performs unexpectedly well even if the assumptions are not fulfilled?
  - The Classification does not require a good estimator of the probabilities, because the event with maximum probability will be assigned to the correct class!
  - Real application: Spam filtering
- Hint for the implementation: Multiple multiplications with probability values (i.e. < 0) results in a fall of values below the available numerical precision
  - Solution: Dealing with logarithmic expressions → products become sums
- How to deal with numerical features?
  - Discretisation: partitioning into bins
  - Assumption of normal distribution: For each class calculate mean μ_C and variance σ_C^2

Agenda

- Introduction to Classification
- 1-R Classifier
- Decision Trees
- Naïve Bayes Classification
- k-Nearest Neigbors
- Combination of classifiers
- Conclusion and further readings



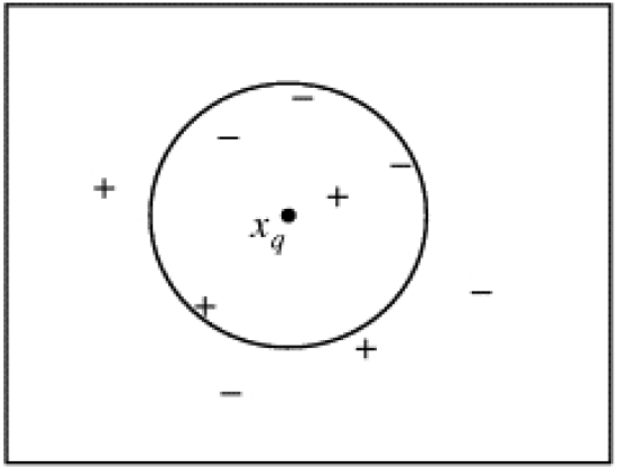
k-Nearest Neigbors

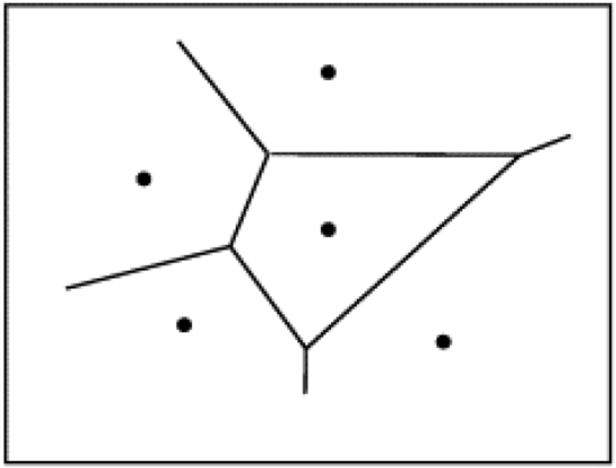
Christian-Albrechts-Universität zu Kiel

- kNN
- Very simple classification procedure
- Approach:
  - Use all training samples as model; no selection; no training
  - Classify an unkown sample by observing its k nearest neighbors
  - Application of a well known distance metric for samples
  - Discrimination of class via majority decision
- Only parameter k that determines the number of nearest neighbors taken into account
  - Typical values 1, 3 or 5 respectively for more than two classes ideally in such a way that a majority decision lead to a clear result

k-Nearest Neigbors (2)

Example:





5-NN assigns the sample to the class " – "

1-NN is represented by a Voronoi diagram (cmp. decision boundary)

k-Nearest Neigbors (3)

Christian-Albrechts-Universität zu Kiel

Choice of parameter k:

- Properties for large k
  - Procedure becomes more resistant against noise
  - But: Also not relevant samples will be taken into account
- Properties for small k
  - Nuances of the class distribution can be modelled
  - But: Sensitive against noise
- Challenge: Find a good trade-off
- Alternatively: Weighting according class affiliation of neighbors respectively their neighbors
  - Restriction to k neighbors is obsolet – all training samples will be affected

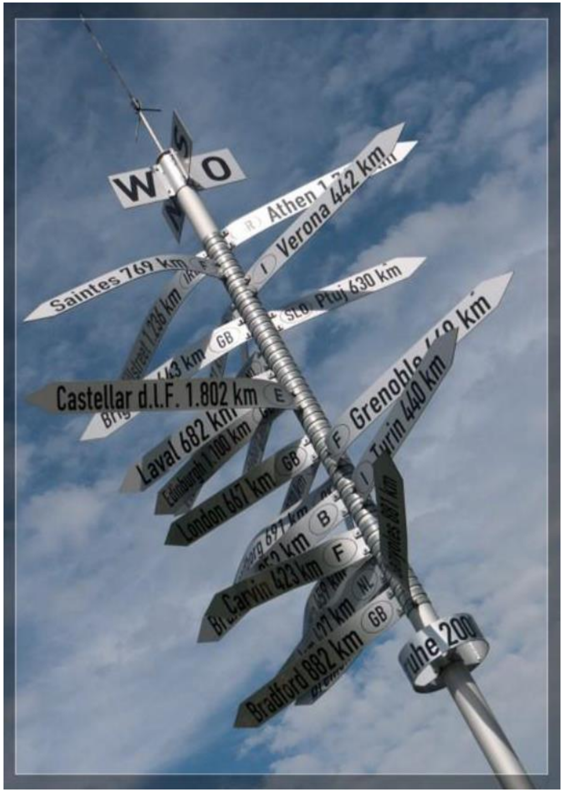
k-Nearest Neigbors (4)

Christian-Albrechts-Universität zu Kiel

- Evaluation
  - Training is not required
  - Fast, but storing of all the training samples is necessary
  - Classification process: Expensive, because all samples has to be taken into account (search nearest neighbors)
  - Model of class distribution of the known training samples
    → Only local approximation (for every sample to be classified)
  - Good for reference values of the the classification performance

Agenda

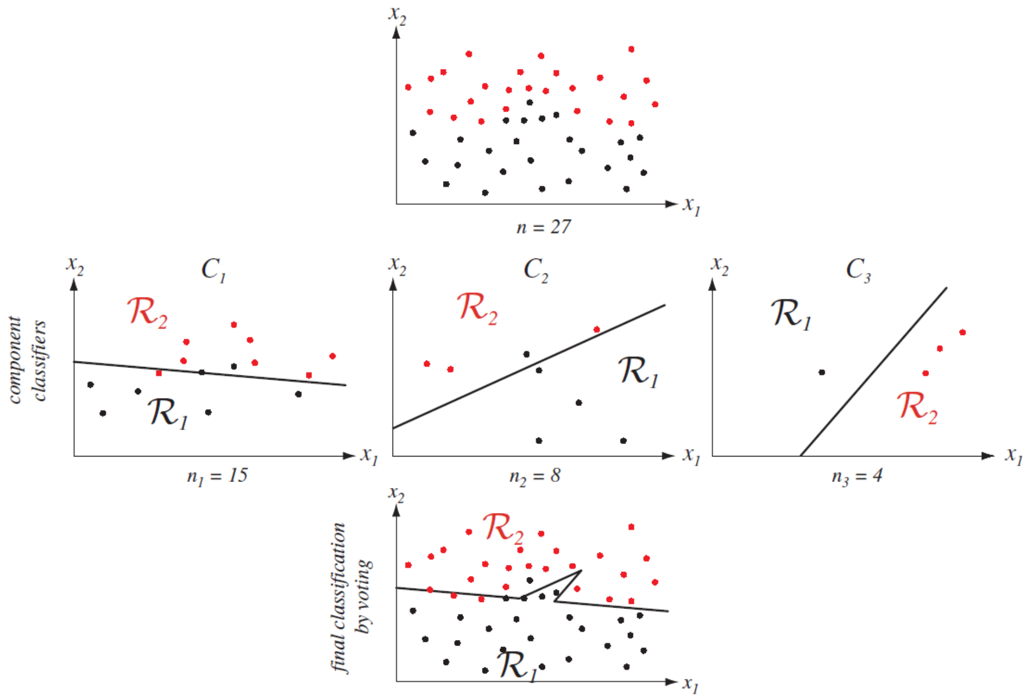
- Introduction to Classification
- 1-R Classifier
- Decision Trees
- Naïve Bayes Classification
- k-Nearest Neigbors
- **Combination of classifiers**
- Conclusion and further readings



Christian-Albrechts-Universität zu Kiel

Combination of classifiers

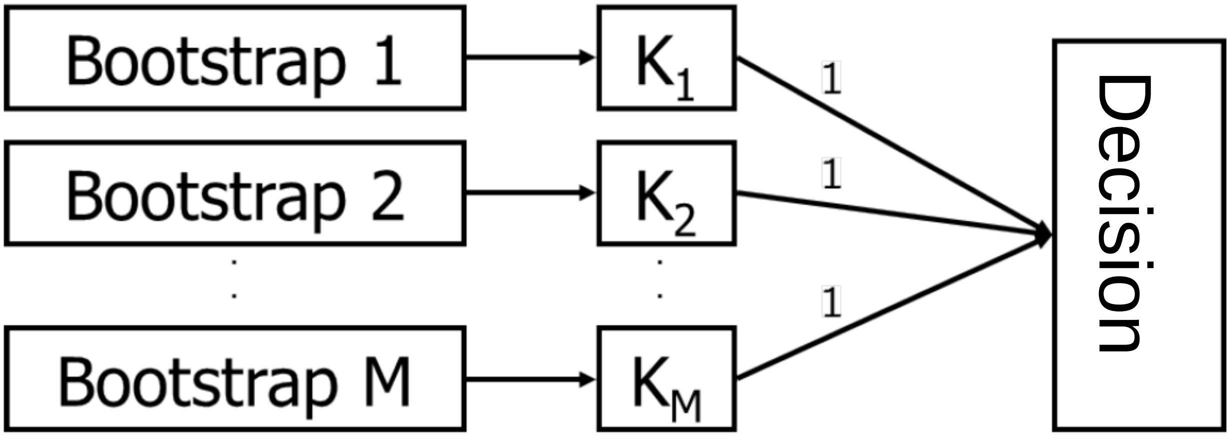
- Why to combine multiple classifiers?
  → Possible improvement of the classification performance
  → Example:



Combination of classifiers (2)

Bagging

- Bootstrap aggregation: for every classifier a new/own training set ("bootstrap") will be generated
  - Random draws with placing back
- Combintion of all classifiers via majority decision



Combination of classifiers (3)

Christian-Albrechts-Universität zu Kiel

Boosting

- The probability for selection of a sample are not constant (as in bagging), but will be recalculated in every Bootstrap iteration
- All classifiers will be generated step by step
- Sample which has been missclassified will be selected more likely
- For the total decision the classification performance of every single classifier is taken into account
- Alternative name: ARCing (Adaptive Reweighting and Combining)

End

Christian-Albrechts-Universität zu Kiel

- Questions….?