Teaching Assistants!

- Looking for ***active*** involvement in our teaching?
- ***Apply for a Teaching Assistantship!***
  - For period 1 and/or 2
  - 391 ~ 456 euro per month (before tax)
  - 8 hours a week (4 contact hours @UU, 4 @home)
  - Use the form at:
    https://wwwsec.cs.uu.nl/education/sollicitatie.php
  - Deadline: June 20, 2018

Note: For Business Intelligence (and other courses in 3rd/4th period), please apply in December for becoming teaching assistants in 2019!

Georg Krempl g.m.krempl@uu.nl Utrecht University

Utrecht, May 22th 2018

Business Intelligence

Lecture 04 - Predictive Analystics
Data Mining

Georg Krempl

Algorithmic Data Analysis
Information and Computing Sciences
Utrecht University, The Netherlands

With particular thanks to

- Armel Lefebvre (tutor, A.E.J.Lefebvre@uu.nl)
- Vincent Goris (student teaching assistant, vincent.goris93@gmail.com)
- Kristof Fellegi (student teaching assistant, k.fellegi@uu.nl)

Summary of the Previous Lecture(s)

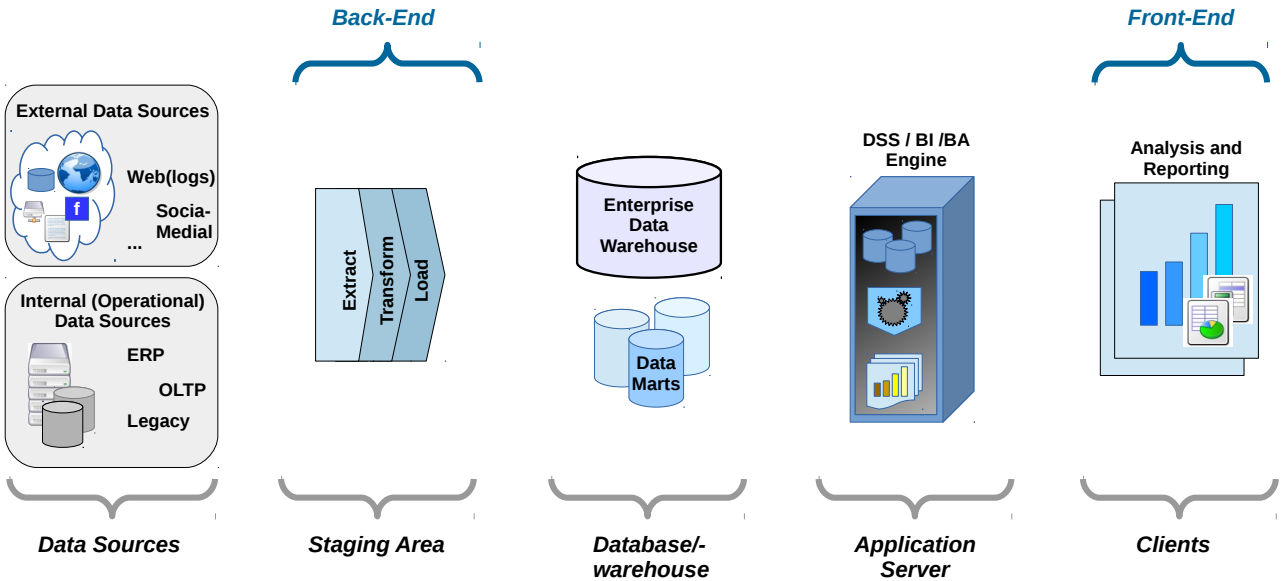


Figure: Data Warehouse Architecture

Summary of the Previous Lecture(s)

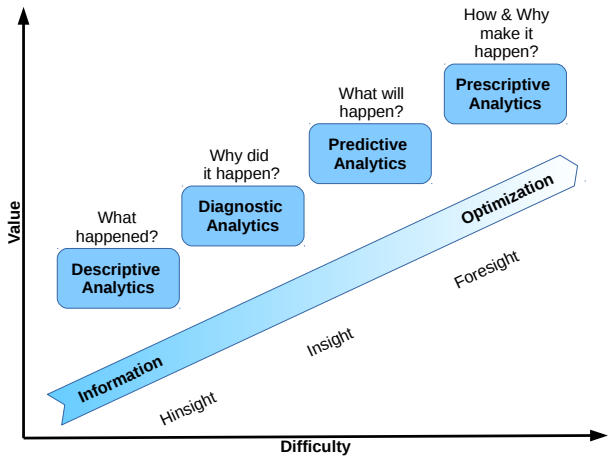


Figure: Analytic Ascendancy Model (based on Gartner's model [Laney, 2012])

- Data Integration: Data Warehousing
  See [Sharda et al., 2018, chapter 3]
- Descriptive Analytics
  See [Sharda et al., 2018, chapter 2]
- Predictive Analytics
  See [Sharda et al., 2018, chapters 4–5]
- Prescriptive Analytics
  See [Sharda et al., 2018, chapter 6]

Predictive Analytics: Data Mining

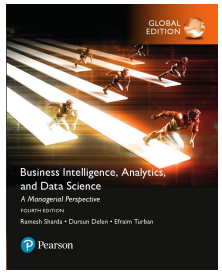


Figure: Textbook [Sharda et al., 2018, Chapter 3]
Sharda, Delen, Turban & King (2018). Business Intelligence, Analytics & Data Science: A Managerial Perspective 4th Global Edition, Pearson. ISBN-13: 9781292220567

Figure: Paper [Wu et al., 2008]
Wu et al. (2008)). Top 10 Algorithms in Data Mining. Knowledge Information Sys., vol. 14. DOI:10.1007/s10115-007-0114-2

Advanced Literature (Voluntary Further Reading):

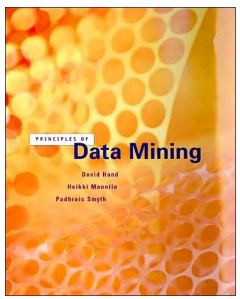


Figure: Textbook [Hand et al., 2001]
Hand, Mannila, Smyth (2001). Principles of Data Mining. The MIT Press. ISBN 978-0262082907

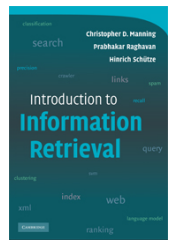


Figure: Textbook [Manning et al., 2008]
Manning, Raghavan, Schütze (2008) Introduction to Information Retrieval. Cambridge University Press. ISBN-13: 978-0124114616

Outline and Summary[1]

- Predictive Analystics in BI: Data Mining Process
  See [Sharda et al., 2018, chapter 4.4]
- Overview on Data Mining Tasks
  See [Sharda et al., 2018, chapter 4.2]
- Segmentation: Cluster Analysis
  See [Sharda et al., 2018, ch. 4.5], [Wu et al., 2008], [Hand et al., 2001, ch. 9]
- Prediction: Classification and Regression
  See [Sharda et al., 2018, ch. 4.5], [Wu et al., 2008], and
  [Hand et al., 2001, ch. 10-11]
- Association: Frequent Itemset and Association Rule Mining
  See [Sharda et al., 2018, ch. 4.5], [Wu et al., 2008], [Hand et al., 2001, ch. 13]
- Mining Semi- and "Unstructured" Data
  See [Sharda et al., 2018, chapter 5]

▸ Start ▸ Appendix

[1] Note: This lecture integrates knowledge from several further sources (cited where used, except if my own).

Outline and Summary[1]

- Predictive Analystics in BI: Data Mining Process
  See [Sharda et al., 2018, chapter 4.4]
- Overview on Data Mining Tasks
  See [Sharda et al., 2018, chapter 4.2]
- Segmentation: Cluster Analysis
  See [Sharda et al., 2018, ch. 4.5], [Wu et al., 2008], [Hand et al., 2001, ch. 9]
- Prediction: Classification and Regression
  See [Sharda et al., 2018, ch. 4.5], [Wu et al., 2008], and
  [Hand et al., 2001, ch. 10-11]
- Association: Frequent Itemset and Association Rule Mining
  See [Sharda et al., 2018, ch. 4.5], [Wu et al., 2008], [Hand et al., 2001, ch. 13]
- Mining Semi- and "Unstructured" Data
  See [Sharda et al., 2018, chapter 5]

▸ Start ▸ Appendix

[1] Note: This lecture integrates knowledge from several further sources (cited where used, except if my own).

Image not available due to copyright restrictions. Please refer to the source cited below.

Figure: Business Intelligence (Source: [Sharda et al., 2018, page 84])

The Cross-Industry Standard Process for Data Mining

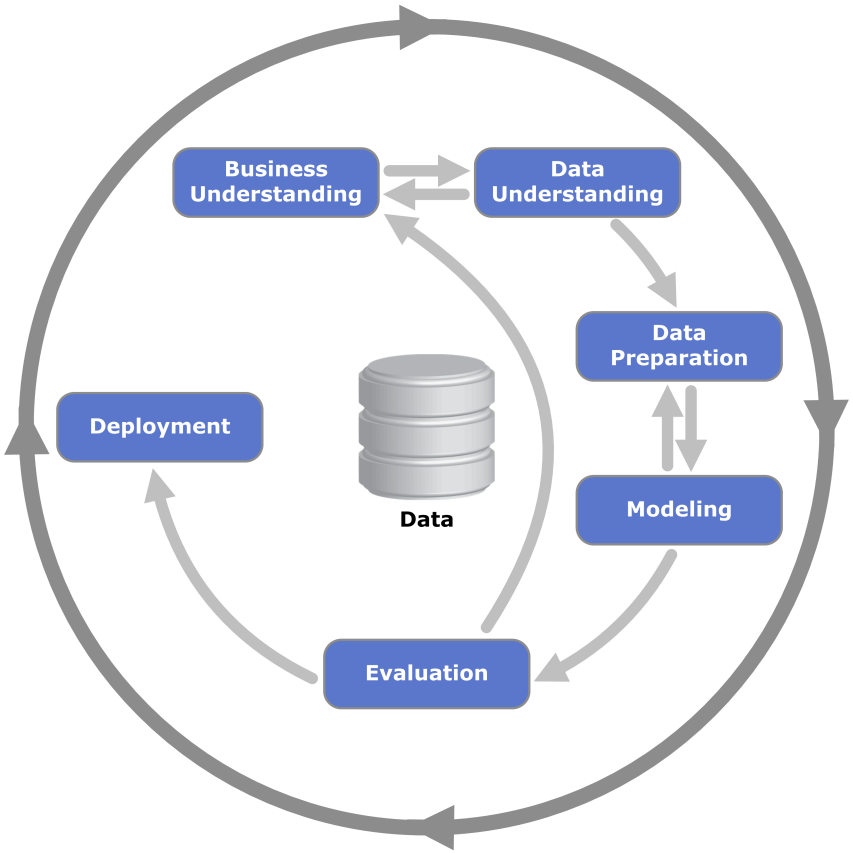


Figure: The CRISP Data Mining Model
(Image by K.Jensen, commons.wikimedia.org)

CRISP-DM Components

- **Business understanding**: Understand business aim & requirements
- **Data understanding** Collect & understand (raw) data (e.g., quality issues)
- **Data preparation** Select & transform raw data into final data
- **Modelling** Model the final data
- **Evaluation** Evaluate the model (reliability and usefulness)
- **Deployment** Put the model to use (e.g., report)

SAS' **S**ample-**E**xplore-**M**odify-**M**odel-**A**ssess Model

Image not available due to copyright restrictions. Please refer to the source cited below.

Figure: SAS Institute's SEMMA Model (Source: [Sharda et al., 2018, page 239])

SEMMA Components

- **Sample** Selecting & partitioning data for mining
- **Explore** Data understanding (e.g., relationships, anormalities)
- **Modify** Select, create & transform variables
- **Model** Model (transformed) data using Data Mining approaches
- **Assess** Evaluate the model (reliability and usefulness)

Critique

- Focus is on the modelling tasks of data mining projects
- Assumes *business understanding* as prerequisite
- Does not specify a deployment phase

Outline and Summary[1]

- Predictive Analystics in BI: Data Mining Process
  See [Sharda et al., 2018, chapter 4.4]
- Overview on Data Mining Tasks
  See [Sharda et al., 2018, chapter 4.2]
- Segmentation: Cluster Analysis
  See [Sharda et al., 2018, ch. 4.5], [Wu et al., 2008], [Hand et al., 2001, ch. 9]
- Prediction: Classification and Regression
  See [Sharda et al., 2018, ch. 4.5], [Wu et al., 2008], and
  [Hand et al., 2001, ch. 10-11]
- Association: Frequent Itemset and Association Rule Mining
  See [Sharda et al., 2018, ch. 4.5], [Wu et al., 2008], [Hand et al., 2001, ch. 13]
- Mining Semi- and "Unstructured" Data
  See [Sharda et al., 2018, chapter 5]

▸ Start ▸ Appendix

[1] Note: This lecture integrates knowledge from several further sources (cited where used, except if my own).

Predictive Analytics: Overview on Data Mining Tasks

Image not available due to copyright restrictions. Please refer to the source cited below.

Figure: Data Mining Tasks
(Source: [Sharda et al., 2018, page 227])

Segmentation

- Unsupervised (Machine Learning)
- Clustering, outlier analysis

Prediction

- Supervised (Machine Learning)
- Classification, Regression, Time-Series-Analysis

Association

- Unsupervised (Machine Learning)
- Frequent Itemset & Association Rule Mining, Link Analysis, Sequence Analysis

Discussion of DM Approaches: Foreword

- In the following subsections,
  we will discuss each of the three DM problem types in detail
- Later, we will extend our view to data of different structure
  (i.e., approaches for so-called semi-/"un-"structured data)

- For each DM problem, we will follow the following structure:
  1. Example of a problem instance
  2. Definition of the problem
  3. Discussion of ideas to solve the problem
  4. Overview on selected approaches & main ideas
  5. Evaluation methodology for models of this DM problem

- This part of the lecture will be more **interactive**:
  - Less "author"-driven, more "reader"-driven
  - Less slides, but more time to take your own notes!

Outline and Summary[1]

- Predictive Analystics in BI: Data Mining Process
  See [Sharda et al., 2018, chapter 4.4]
- Overview on Data Mining Tasks
  See [Sharda et al., 2018, chapter 4.2]
- Segmentation: Cluster Analysis
  See [Sharda et al., 2018, ch. 4.5], [Wu et al., 2008], [Hand et al., 2001, ch. 9]
- Prediction: Classification and Regression
  See [Sharda et al., 2018, ch. 4.5], [Wu et al., 2008], and
  [Hand et al., 2001, ch. 10-11]
- Association: Frequent Itemset and Association Rule Mining
  See [Sharda et al., 2018, ch. 4.5], [Wu et al., 2008], [Hand et al., 2001, ch. 13]
- Mining Semi- and "Unstructured" Data
  See [Sharda et al., 2018, chapter 5]

▸ Start ▸ Appendix

[1] Note: This lecture integrates knowledge from several further sources (cited where used, except if my own).

Segmentation: Examplary Problem and Task Definition

Clustering: Exemplary Problem

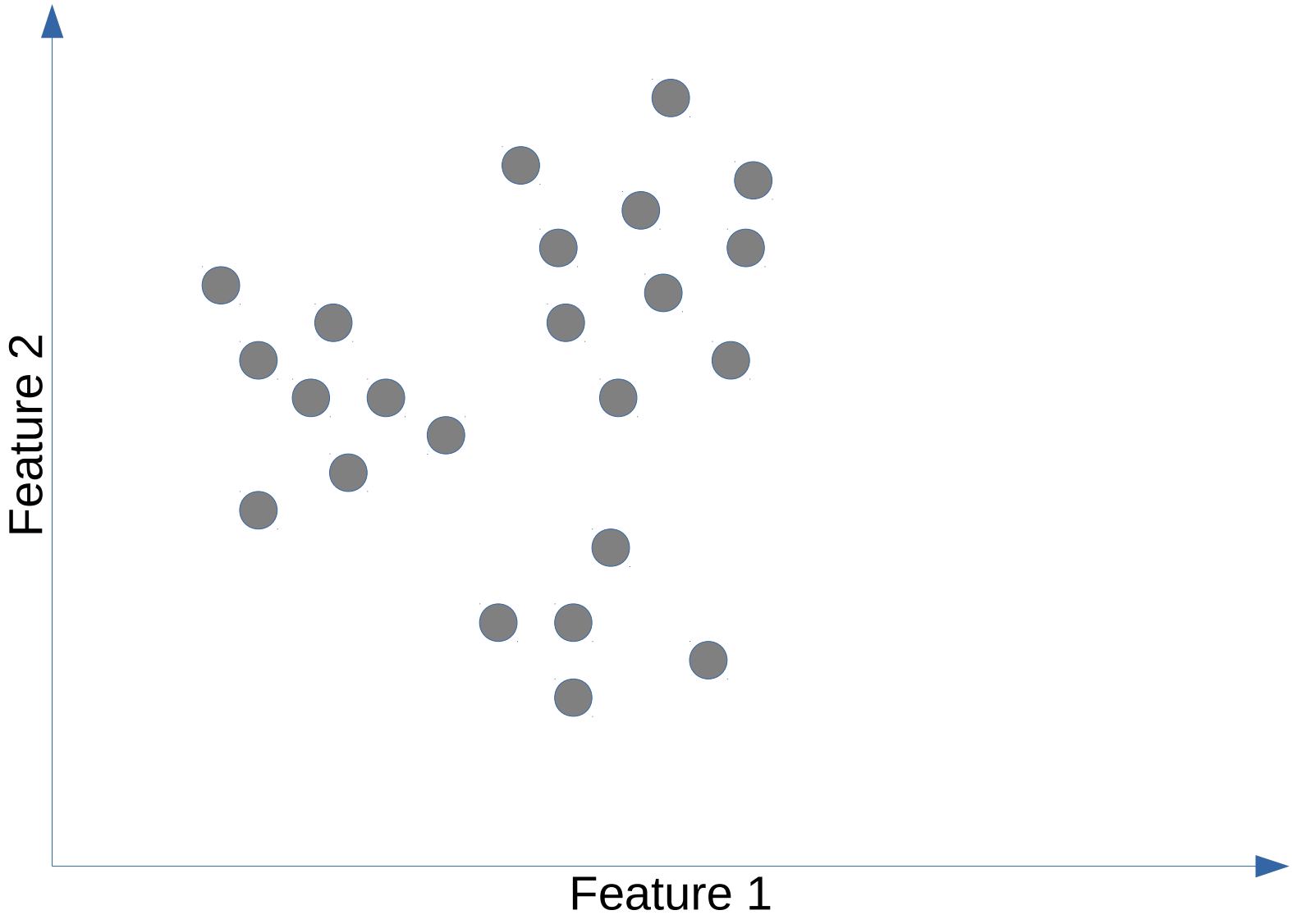


Figure: Clustering Task

Clustering: Further Exemplary Problems

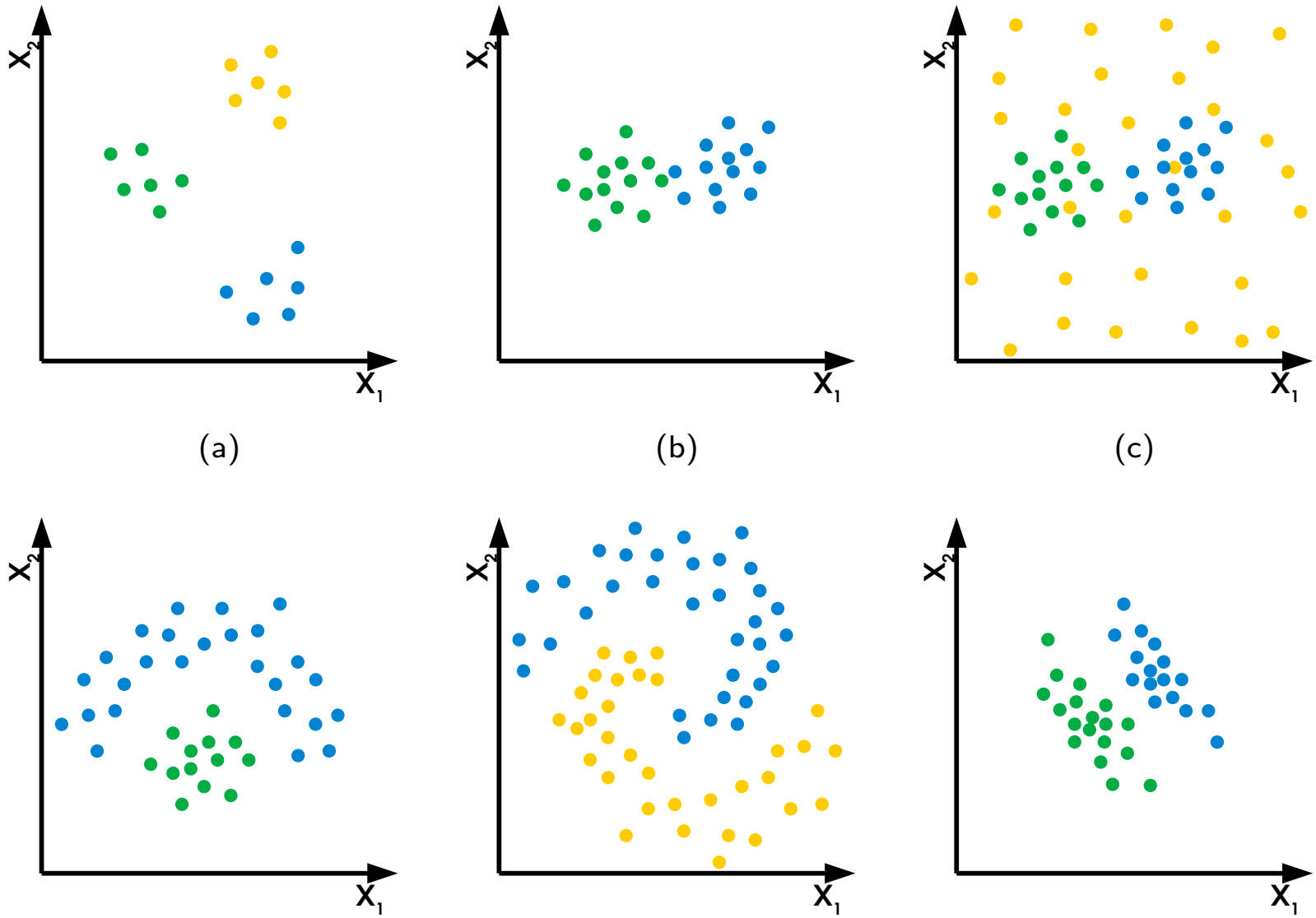


Figure: Scatter plots of different data sets.

Segmentation: Overview on Selected Approaches

Distinctive Characteristics

Flat vs. Hierarchical Is the output

- a hierarchy: there exists a hierachy between the instances based on their proximity
- a flat partitioning: clusters are simple partitions (no hierarchy between instances)

Hard vs. Soft Are assignments between instances and clusters

- hard: every instance belongs to one and only one cluster
- soft: an instance belongs to all clusters with varying membership probability
  (also denoted a **probabilistic** clustering)

Segmentation: Overview on Flat Partitioning Approaches

Shape-based Flat Clustering

- Starts with an assumption about the shape of the cluster(s)
- Examples:
  - Spheres (k-means), in R: `kmeans()` (which is a R core function)
  - Gaussians or other distributions (Expectation-Maximisation for Mixture Models) In R: `emcluster()` from the package `EMCluster`
  - ...
- Parameters:
  - Shape (or proximity) & parameters) of components
  - Number of components

Density-based Flat Clustering

- Assumes that clusters are separated by low-density regions
- Example Algorithms:
  - DBSCAN [**?**], in R: `dbscan()` from the package `dbscan`
  - ...
- Parameters:
  - Density estimation technique
  - Density threshold

Segmentation: Overview on Hierarchical Approaches

Agglomerative Hierarchical Clustering

- Starts with each instance being its own cluster, iterative merging
- Parameters:
  - Proximity (similarity or distance) measure
  - Linkage
- Different criteria for linking instances & clusters:

  Single Linkage: Given clusters A, B with instances $x_a \in A$, $x_b \in B$, consider the distance between the closest pair (x_a^*, x_b^*) of instances

  Complete Linkage: Given clusters A, B with instances $x_a \in A$, $x_b \in B$, consider the distance of the most distant pair (x_a^*, x_b^*) of instances

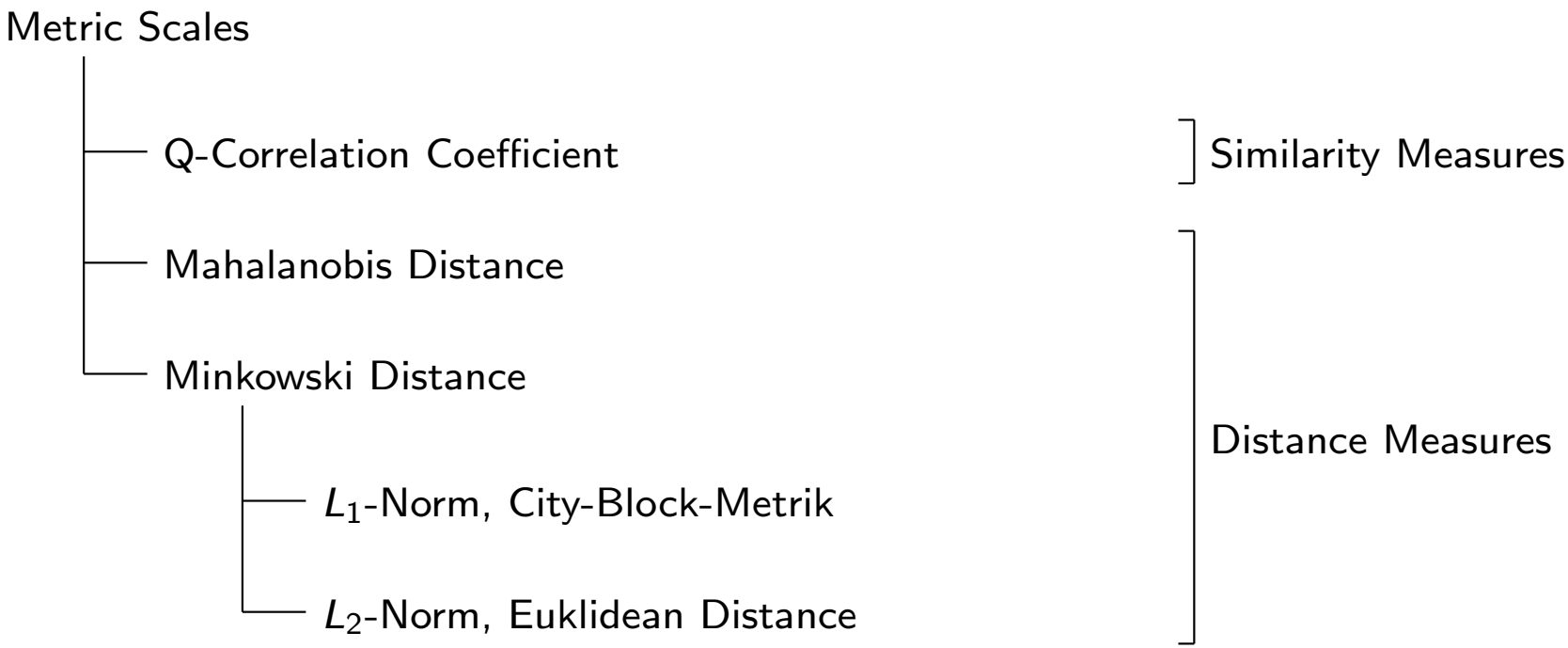
  Average Linkage: Given clusters A, B with instances $x_a \in A$, $x_b \in B$, consider the average distance of all instance pairs (x_a, x_b)
- R: `agnes()` in the package *cluster*

Divisive Hierarchical Clustering

- Starts with a single cluster containing all points. Each iteration, split one cluster
- R: `diana()` in the package *cluster*

Segmentation: Measuring Distance & Similarity

Measuring Proximity



Minkowski Distance[2]

$$d_{ij} = \left[\sum_{m=1}^{M} |x_{im} - x_{jm}|^r \right]^{\frac{1}{r}} \tag{1}$$

d_{ij} Distance between the instances i and j

x_{im} Value of attribute m for instance i

M Number of attributes

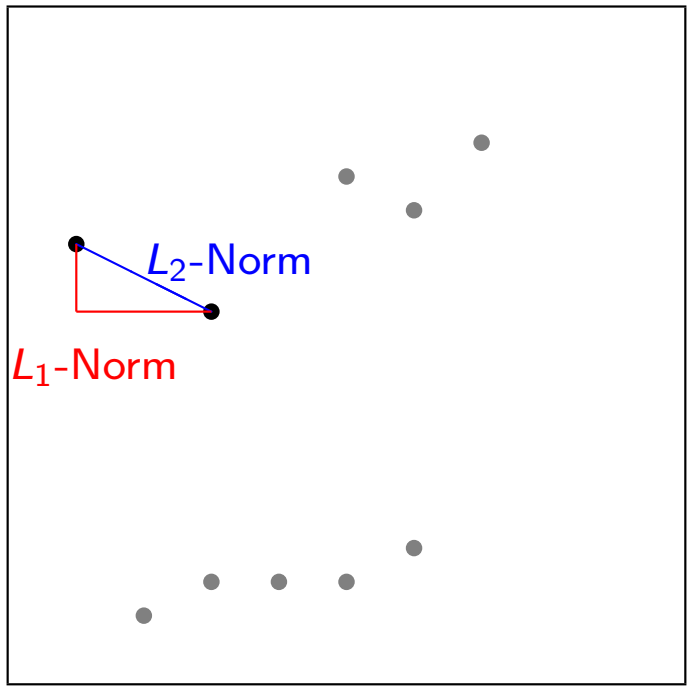
r *Minkowski Constant*

- $r = 1$ *L_1-Norm*, *City-Block Metric*
  City-Block Metric. Simple addition of the absolute distances (on each dimension) between the instances
- $r = 2$ *L_2-Norm*, *Euklidean Distance*
  Eukl. Dist. weights larger distances overproportionally

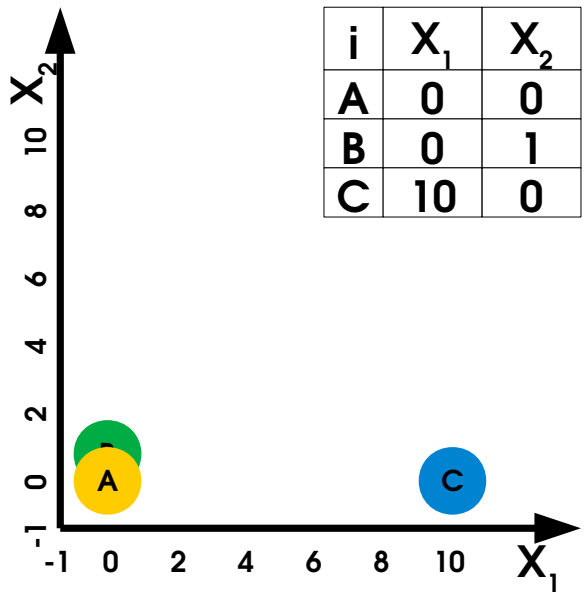
[2] Equation from [Backhaus et al., 2006, page 503].

Minkowski Distance: Example

Minkowski Distance Example:



Minkowski Distances: Scaling Issues



i	X_1	X_2
A	0	0
B	0	1
C	10	0

Figure: Scale Dependency Illustrated

- Note that the Minkowski-Distances (Euclidean, City-Block/Manhattan, ...) are scaling-dependent!
- Example: Three data points A,B,C
- **Question: Is B or C closer to A?**
- Depends on the scaling of X_1 relative to X_2!
- If we rescale only X_1 to a range $[0; 1]$, then A and C are (seemingly) close
- If we rescale only X_2 to a range $[0; 1]$, then A and B are (seemingly) close
- Either adjust scale, or use a scale-invariant Mahalanobis distance are (seemingly) close

Standardised Euclidean and Mahalanobis Distances

Standardised Euclidean Distance

$$d_{ij} = \left(\sum_{m=1}^{M} \frac{(x_{im} - x_{jm})^2}{s_m^2} \right)^{\frac{1}{2}} \tag{2}$$

d_{ij} Distance between the instances i and j

x_{im} Value of attribute m for instance i

M Number of attributes

s_m *Standard deviation* of m-th attribute

For zero covariance between attributes, this equals the Mahalanobis Distance.

Standardised Euclidean and Mahalanobis Distances

Mahalanobis Distance

$$d_{ij} = \left((\vec{x_i} - \vec{x_j})^T \Sigma^{-1} (\vec{x_i} - \vec{x_j}) \right)^{\frac{1}{2}} \tag{3}$$

d_{ij} Distance between the instances i and j

$\vec{x_i}$ Feature vector of instance i

M Number of attributes

Σ *Covariance Matrix* of the data

For zero covariance between attributes,
this equals the Standardised Euclidean Distance.

Q-Correlation Coefficient[3]

$$a_{ij} = \frac{\sum_{m=1}^{M} (x_{im} - \bar{x_i})(x_{jm} - \bar{x_j})}{\left[\sum_{m=1}^{M} (x_{im} - \bar{x_i})^2 \cdot \sum_{m=1}^{M} (x_{jm} - \bar{x_j})^2\right]^{\frac{1}{2}}} \quad (4)$$

a_{ij} Similarity between instances i and j

x_{im} Value of attribute m of instance i

M Number of attributes

$\bar{x_i}$ Arithmetic average of all of instance i's attributes

- Denominator: compare with covariance (like Pearson's correlation coefficient's denominator)
- Nominator: compare to Pearson's correlation coefficient's nominator

- Similarity Measure: Compares the similarity of the profiles, not their location!
- In particular useful, if one is interested in comparing the relative attribute values between instances, and not in their absolute values.

[3] Equation from [Backhaus et al., 2006, page 505].

Segmentation: Selected Clustering Approaches

Clustering: k-Means Algorithm

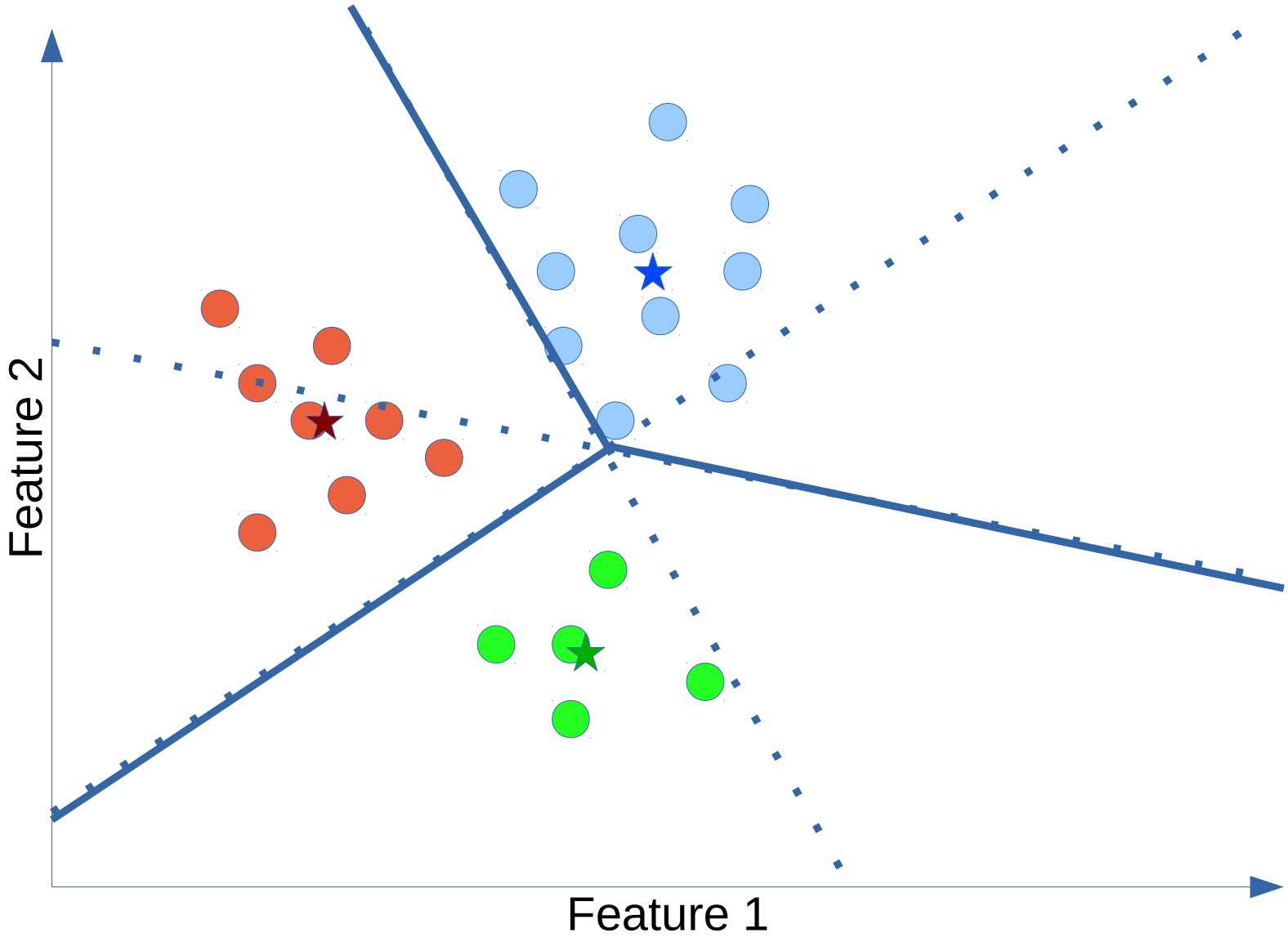


Figure: k-Means Clustering

Clustering: k-Means Algorithm

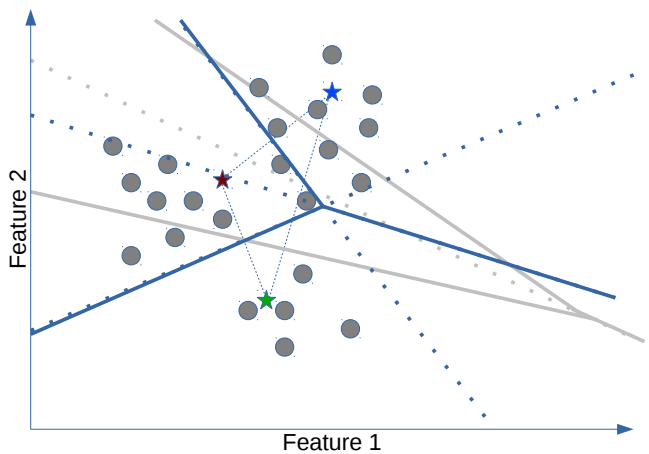


Figure: k-Means in progress ...

k-Means Algorithm

1. Parameters: n, distance measure, stopping criterion
2. Initalise cluster randomly
3. Compute distances between instances & centers
4. Assign each instance to its nearest center
5. Compute new cluster centre as mean of its instances
   Alternative: median
6. Repeat with step 3 until stopping criterion is met

Clustering: Expectation-Maximisation Algorithm

EM Algorithm

- Assume data to be generated by a **Mixture Model**
  e.g., Gaussian Mixture Model (GMM)
- Given: Instances with features $D = \{x(1), x(2), \cdots, x(n)\}$
- Probabilistic Clustering:
  Instances are members of all clusters, but with varying probability
- Views cluster-assignment as hidden variable Z
  with $H = \{z(1), z(2), \cdots z(n)\}$ being the instances' hidden values
- Aim is to determine the **Likelihood-maximising** Model
  Likelihood of observed data:

$$l(\theta) = \log p(D|\theta) = \log \sum_H p(D, H|\theta)$$

  of probabilistic model $p(D, H|\theta)$
  with unknown parameters θ
  $Q(H)$ probability distribution of missing data H
- Iterate between E(xpectation) and M(aximisation) Steps
  E-Step: $Q^{k+1} = \arg\max_Q F(Q^k, \theta^k)$
  M-Step: $\theta^{k+1} = \arg\max_\theta F(Q^{k+1}, \theta^k)$

- Also useable for **missing value imputation**

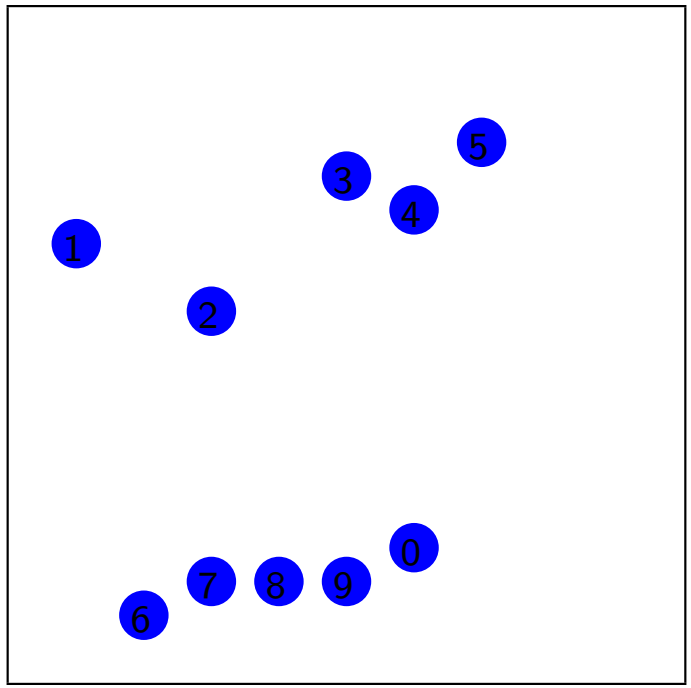
Clustering: Divisive Hierarchical Clustering

Clusteringverfahren

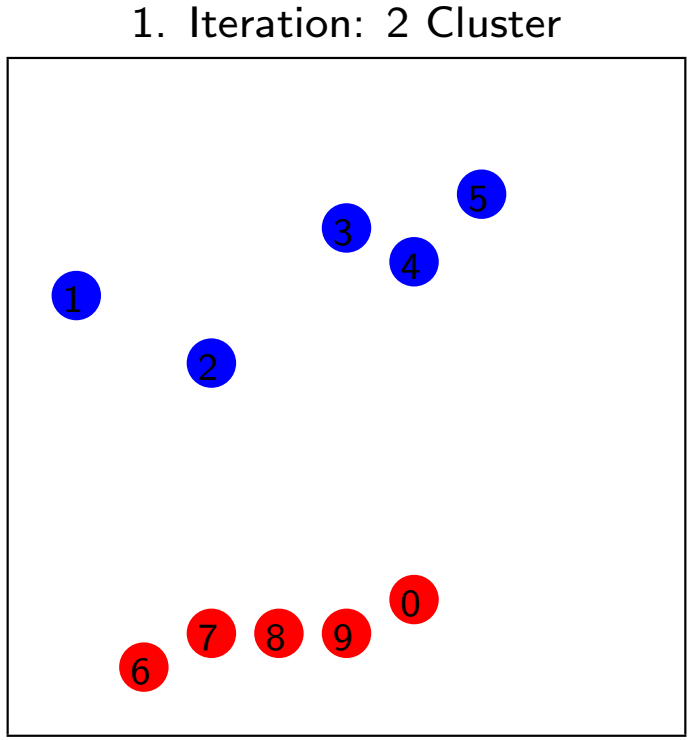
- Hierarchical
  - Divisive
  - Agglomerative
    - Single-Linkage: Nearest-Neighbor
    - Complete-Linkage: Furthest-Neighbor
    - Average-Linkage
    - Ward: Variance Minimising

Clustering: Divisive Hierarchical Clustering

Initialisierung:

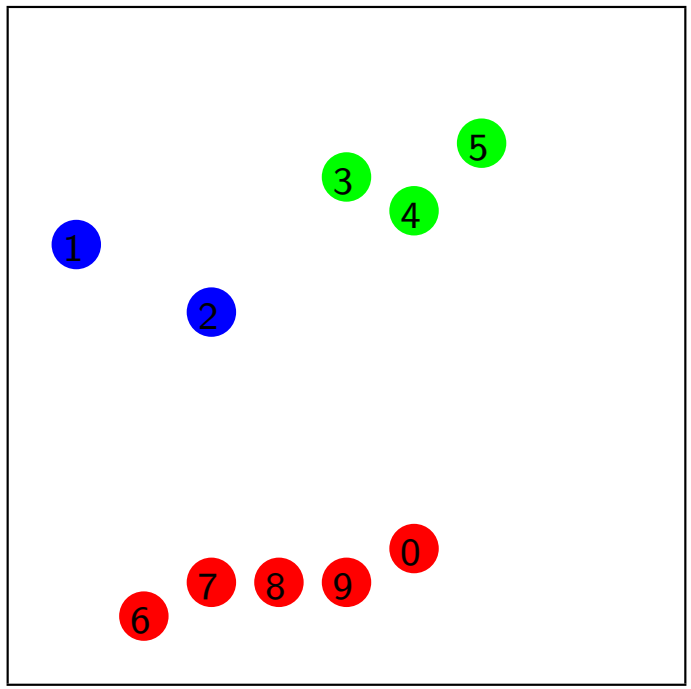


Clustering: Divisive Hierarchical Clustering



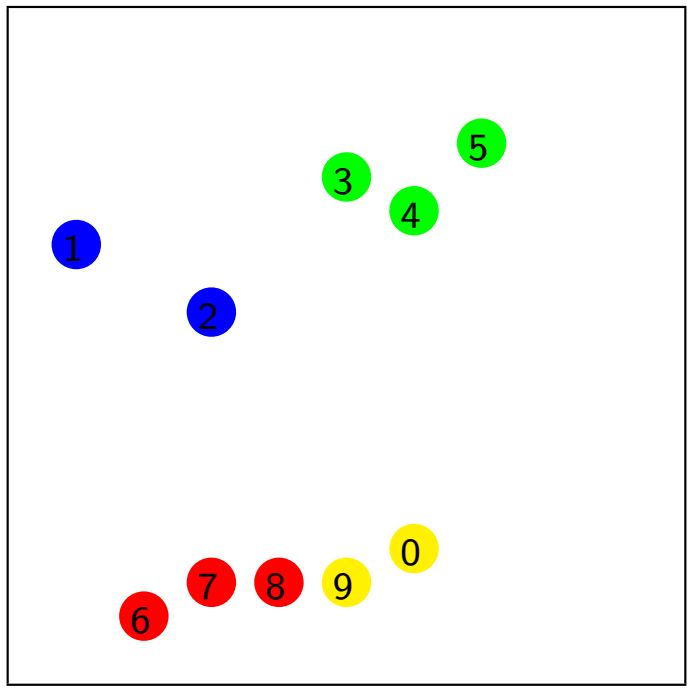
Clustering: Divisive Hierarchical Clustering

2. Iteration: 3 Cluster



Clustering: Divisive Hierarchical Clustering

3. Iteration: 4 Cluster



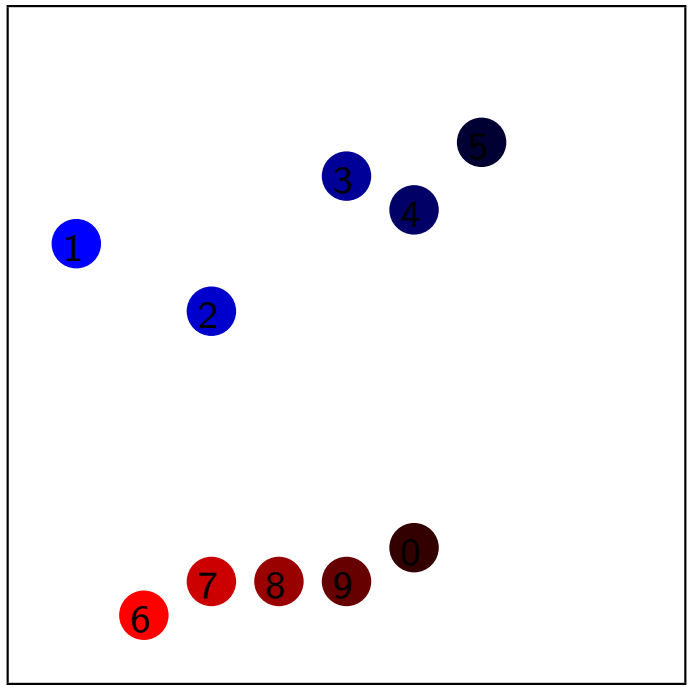
Clustering: Agglomerative Hierarchical Clustering

Clusteringverfahren

- Hierarchical
  - Divisive
  - Agglomerative
    - Single-Linkage: Nearest-Neighbor
    - Complete-Linkage: Furthest-Neighbor
    - Average-Linkage
    - Ward: Variance Minimising

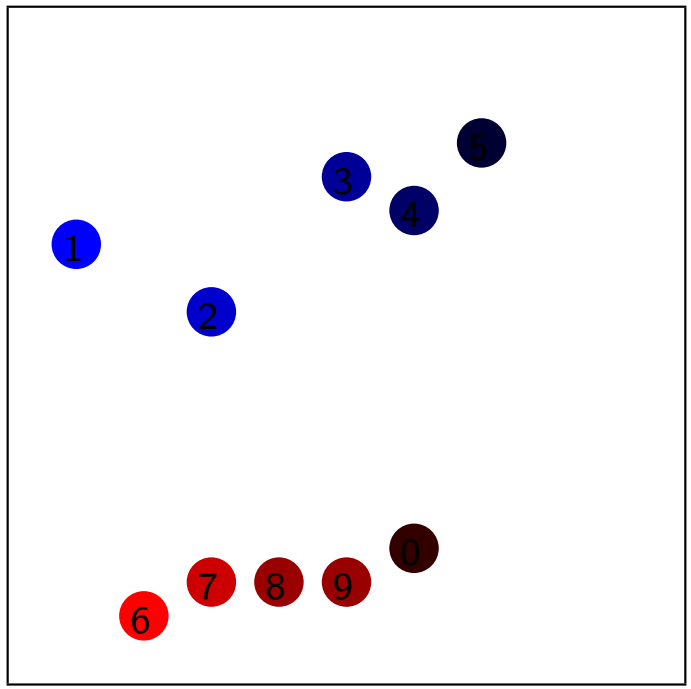
Clustering: Agglomerative Hierarchical Clustering

Initialisierung: N Cluster



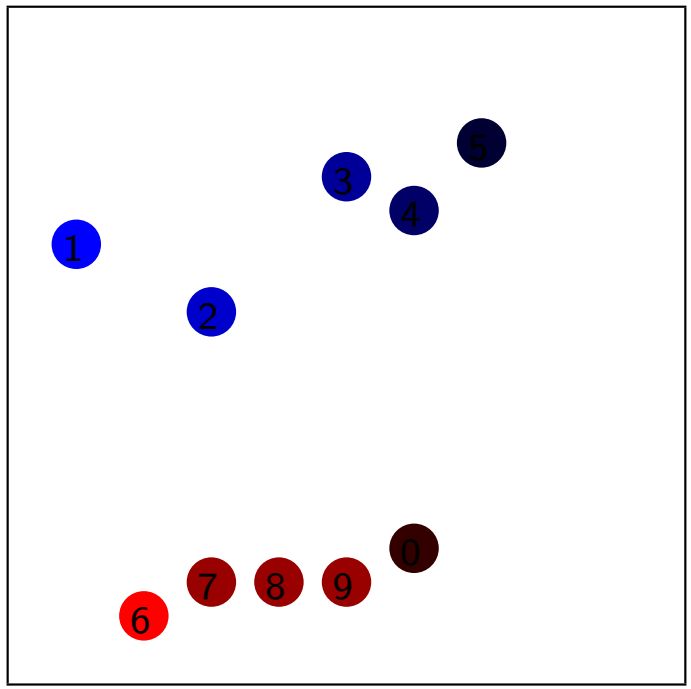
Clustering: Agglomerative Hierarchical Clustering

1.Iteration: $N - 1$ Cluster



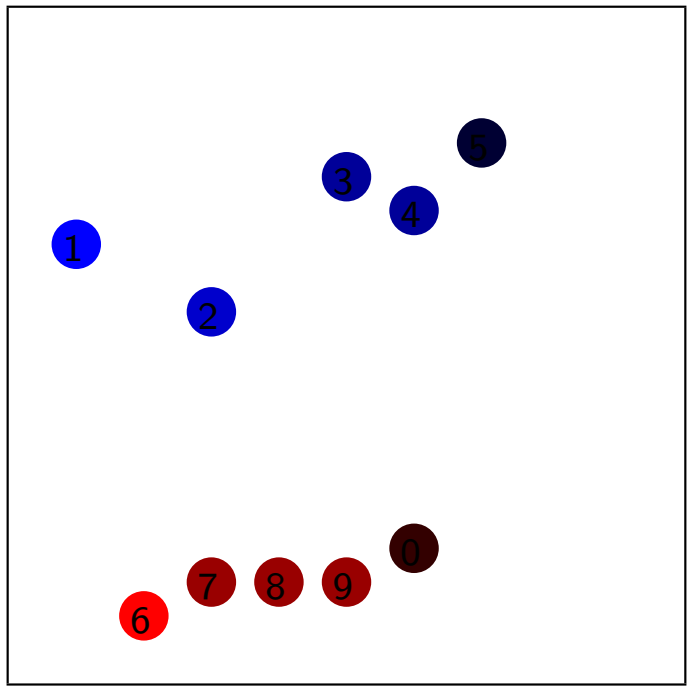
Clustering: Agglomerative Hierarchical Clustering

2.Iteration: $N - 2$ Cluster



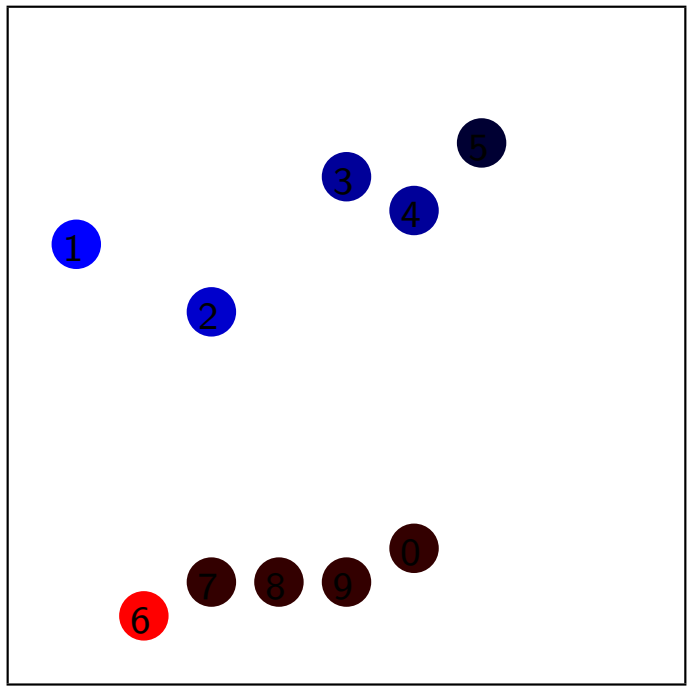
Clustering: Agglomerative Hierarchical Clustering

3.Iteration: $N - 3$ Cluster



Clustering: Agglomerative Hierarchical Clustering

4.Iteration: $N - 4$ Cluster



Segmentation: Evaluation Methodology

Clustering Objective

Maximise

- Intra-Cluster-Homogenity, i.e. similarity of instances in the same cluster
- Inter-Cluster-Heterogenity, i.e. dissimilarity of instances from different clusters

Internal Measures

- Based on the clustered data only, not on external class information
- Various measures exist, see R package *ClusterCrit* and description by Desgraupes (2013)
  `https://cran.r-project.org/web/packages/clusterCrit/vignettes/clusterCrit.pdf`
- Example: Silhouette index

External Measures

- Based on an external class label
- Again, various measures exist, see R package *ClusterCrit* and description by Desgraupes (2013)
  `https://cran.r-project.org/web/packages/clusterCrit/vignettes/clusterCrit.pdf`
- Example: Rand Index, Purity, NMI

Silhouette Index

For each instance x_i,

- for each cluster k, calculate the arithmetic average distance of x_i to all instances in that cluster k:

$$d(x_i, k) = \frac{1}{n_k - 1} \sum_{x \in z_k, x \neq x_i} dist(x_i, x)$$

  where $dist()$ is a distance function, n_k is the number of instances in cluster k
- determine its average distance within its own cluster l:

$$a(x_i) = d(x_i, l)$$

- from the average distances to other clusters $k \neq l$, determine the minimum:

$$b(x_i) = \min_{k \neq l} d(x_i, k)$$

- calculate the silhouette width of that instance x_i as

$$s(x_i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}$$

Segmentation: Clustering Evaluation Measures (2)

Silhouette Index

Having computed the silhouette width of each instance, we compute now the aggregates:

- For each cluster k, calculate the mean silhouette width s_k

$$s_k = \frac{1}{n_k} \sum_{x_i \in z_k} s(x_i)$$

- Calculate Silhouette Index as the mean of the clusters' mean silhouettes:

$$SI = \frac{1}{|Z|} \sum_{k=1}^{|Z|} s_k$$

Segmentation: Clustering Evaluation Measures

Purity[4]

$$purity(Y, Z) = \frac{1}{n} \sum_{i=1}^{|Z|} \max_j \left| z_i \cap y_j \right|$$

- $Z = \{z_1, z_2, \ldots, z_{|Z|}\}$ set of clusters
- $Y = \{y_1, y_2, \ldots, y_{|Y|}\}$ set of classes
- n is the number of instances (in all clusters/classes)
- Assigns each cluster to the most frequent class in it
- Purity is value in $[0; 1]$, with 1 being a perfect clustering
- Depends on the number of clusters (1 if one cluster per instance)

[4] See [Manning et al., 2008, Eq. 182].

Segmentation: Clustering Evaluation Measures

Normalised Mutual Information NMI[5]

$$NMI(Y, Z) = \frac{I(Y, Z)}{\frac{H(Z)+H(Y)}{2}}$$

- $Z = \{z_1, z_2, \ldots, z_{|Z|}\}$ set of clusters
- $Y = \{y_1, y_2, \ldots, y_{|Y|}\}$ set of classes
- Mutual information $I(Y, Z) = \sum_{i=1}^{|Z|} \sum_{j=1}^{|Y|} \frac{|z_i \cap y_j|}{n} \log \frac{N \cdot |z_i \cap y_j|}{|z_i| \cdot |y_j|}$
- Entropy $H(Z) = -\sum_{i=1}^{|Z|} \frac{|z_i|}{n} \log \frac{|z_i|}{n}$
- n is the number of instances (in all clusters/classes)
- Assigns each cluster to the most frequent class in it
- NMI is value in $[0; 1]$, with 1 being a perfect clustering
- Normalised for the number of clusters (i.e., independent of $|Z|$)

[5] See [Manning et al., 2008, Eq. 183–187].

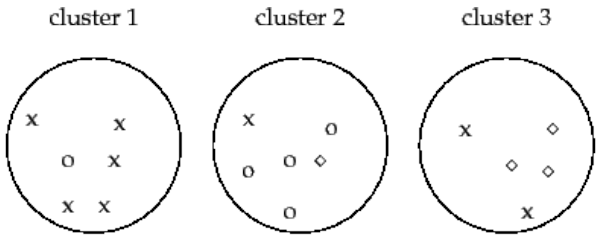
Rand Index (or Accuracy)[6]

$$RI(Y, Z) = \frac{TP + TN}{TP + FP + FN + TN}$$

- True positives, the number of pairs in the same cluster and class:
- True negatives TN: number of pairs in different clusters and classes
- False positives FP: number of pairs in the same cluster but different classes
- False negatives FN: number of pairs in different clusters but same class
- Measures the percentage of correct decisions
- RI is value in $[0; 1]$, with 1 being a perfect clustering

[6] See [Manning et al., 2008, Eq. 188–189].

Segmentation: Clustering Evaluation Example



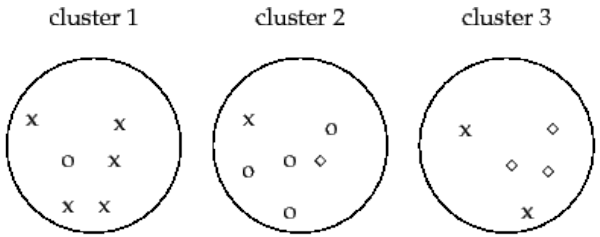
► **Figure 16.1** Purity as an external evaluation criterion for cluster quality. Majority class and number of members of the majority class for the three clusters are: x, 5 (cluster 1); o, 4 (cluster 2); and ◇, 3 (cluster 3). Purity is $(1/17) \times (5+4+3) \approx 0.71$.

Figure: Clustering Example from [Manning et al., 2008, Fig. 16.1]

	$y_1 : x$	$y_2 : o$	$y_3 : \diamond$	$\sum$ **cols**
z_1	5	1	0	6
z_2	1	4	1	6
z_3	2	0	3	5
$\sum$ **rows**	8	5	4	17

- $TP = \sum_{ij} \binom{|z_i \cap y_j|}{2} = \binom{5}{2} + \binom{1}{2} + \cdots = 20$
- $TP + FP = \sum_i \binom{\Sigma \text{cols}_i}{2} = \binom{6}{2} + \binom{6}{2} + \binom{5}{2} = 40$
- $FP = TP + FP - TP = 40 - 20 = 20$
- $TP + FN = \sum_j \binom{\Sigma \text{rows}_j}{2} = \binom{8}{2} + \binom{5}{2} + \binom{4}{2} = 44$

Segmentation: Clustering Evaluation Example



► **Figure 16.1** Purity as an external evaluation criterion for cluster quality. Majority class and number of members of the majority class for the three clusters are: x, 5 (cluster 1); o, 4 (cluster 2); and ◇, 3 (cluster 3). Purity is $(1/17) \times (5+4+3) \approx 0.71$.

Figure: Clustering Example from [Manning et al., 2008, Fig. 16.1]

	$y_1 : x$	$y_2 : o$	$y_3 : \diamond$	$\sum$ **cols**
z_1	5	1	0	6
z_2	1	4	1	6
z_3	2	0	3	5
$\sum$ **rows**	8	5	4	17

- from before: $TP = 20$, $FP = 20$, $TP + FN = 44$
- $FN = TP + FN - TP = 44 - 20 = 24$
- $TN = \binom{n}{2} - TP - FP - FN =$ $\binom{17}{2} - 20 - 20 - 24 = 136 - 64 = 72$
- $Purity = 0.71$
- $NMI = 0.36$
- $RI = \frac{20+72}{136} = 0.676$

Segmentation: Choosing the Number of Clusters

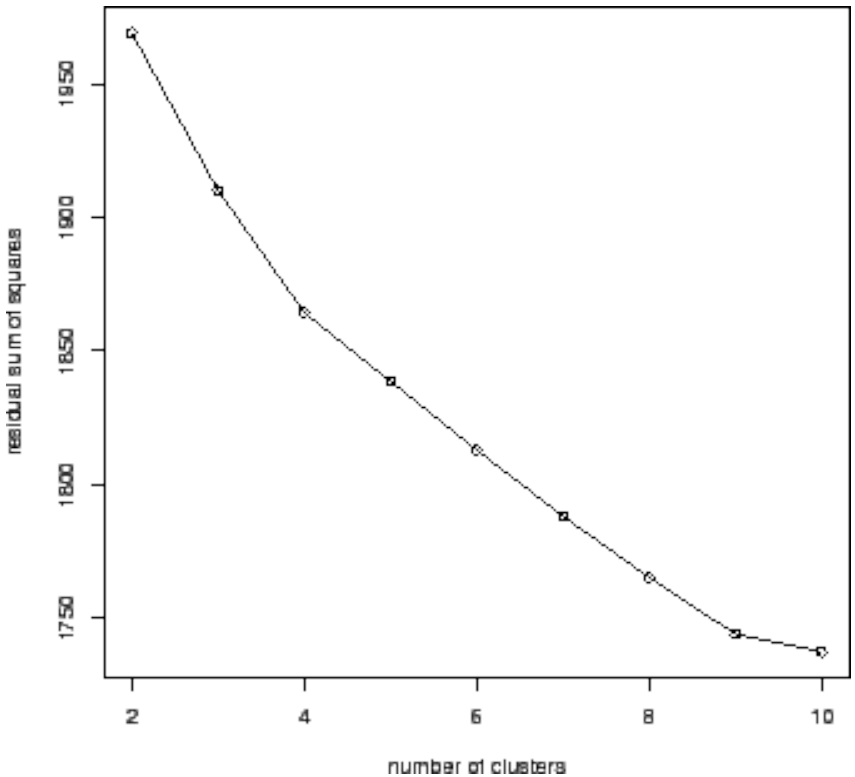


Figure: Estimated Residual Sum of Squares as Function of k in k-means [Manning et al., 2008, Fig. 16.8]

Elbow Method

- Iterate over different values of k, for each k, perform several clusterings with different random initialisations for each of them, compute a performance measure,
- E.g., compute the residual sum of squares
  $RSS_k = \sum_{z_i \in Z} \sum_{x \in z_i} (x - \bar{z}_i)^2$
  (where $\bar{z}_i$ is the i-th cluster's center)
- Plot the averaged performance values against the number of clusters, inspect the curve for "knees" (points where it flattens)
- Curve flattens at $k = 4$ and $k = 9$, i.e. these are candidates for k^*

Segmentation: Cluster Analysis

Questions?

Outline and Summary[1]

- Predictive Analystics in BI: Data Mining Process
  See [Sharda et al., 2018, chapter 4.4]
- Overview on Data Mining Tasks
  See [Sharda et al., 2018, chapter 4.2]
- Segmentation: Cluster Analysis
  See [Sharda et al., 2018, ch. 4.5], [Wu et al., 2008], [Hand et al., 2001, ch. 9]
- Prediction: Classification and Regression
  See [Sharda et al., 2018, ch. 4.5], [Wu et al., 2008], and
  [Hand et al., 2001, ch. 10-11]
- Association: Frequent Itemset and Association Rule Mining
  See [Sharda et al., 2018, ch. 4.5], [Wu et al., 2008], [Hand et al., 2001, ch. 13]
- Mining Semi- and "Unstructured" Data
  See [Sharda et al., 2018, chapter 5]

▸ Start ▸ Appendix

[1] Note: This lecture integrates knowledge from several further sources (cited where used, except if my own).

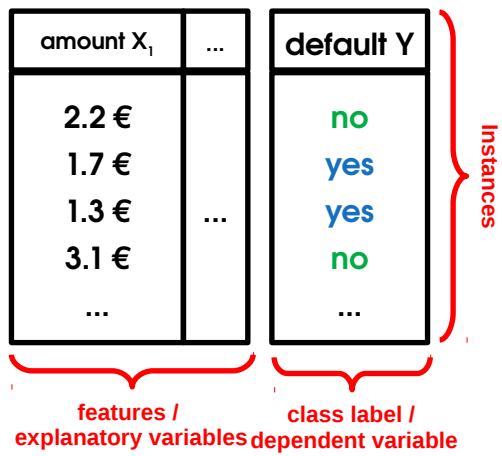
Classification: Credit Scoring Example

Let the business objective be minimising risk of loss due to unpaid orders by customers, i.e., by identifying risky orders and acting accordingly (accepting or declining)

- Each new order constitutes an **instance**
- For each instance, certain characteristics are **known before the decision** has to be taken to accept or decline the order. These characteristics correspond to **features (or explanatory variables)** X
- Instances are classified by their outcome, e.g., default vs. non-default. This **outcome** corresponds to the **class label** or **dependent variable** Y
- Aim: A **classifier (predictor) function** $f : X \to Y$ that optimises a desired performance measure, e.g., the 0-1-loss

$$L(y, f(x)) = \begin{cases} 0 & f(x) = y \\ 1 & \text{otherwise} \end{cases}$$

over a **set of instances**.



Classification in a Nutshell

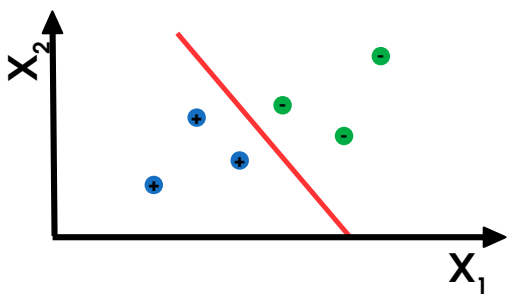
Credit Scoring: Will a new order by paid?

- **Historical Data**
  e.g. previous customer's records
- **Generate Training Sample** $\mathcal{L}$ with feature variables (e.g. order value X) and class label (e.g. default Y)
- **Aim: Classifier function** $f : X \to Y$
- **How do we get there?**

Classification in a Nutshell

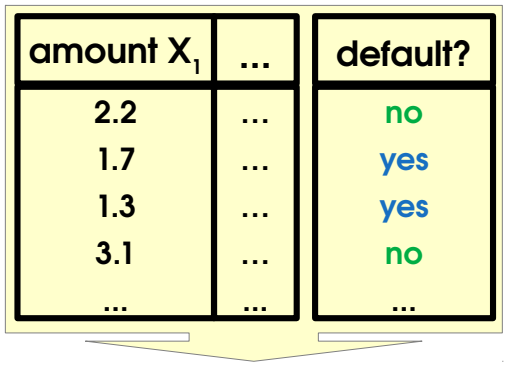
Credit Scoring: Will a new order by paid?

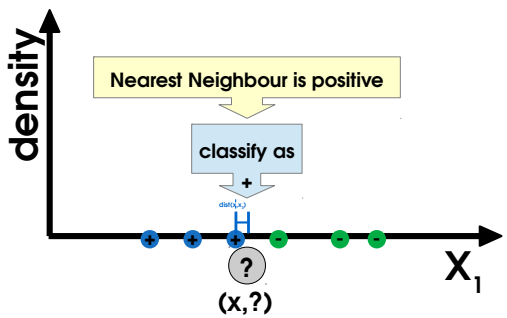
X_1	X_2	default?
2.2	52	no
1.7	28	yes
1.3	33	yes
3.1	42	no
...	...	...



- **Historical Data**
  e.g. previous customer's records
- **Generate Training Sample** $\mathcal{L}$ with feature variables (e.g. order value X) and class label (e.g. default Y)
- **Aim: Classifier function** $f : X \to Y$
- **Discussion: How do we get there?**
  - Nearest Neighbour Classifier
  - Bayes' Classifier (Histogram-Based)
  - Bayes' Classifier (Kernel-Based)
  - Naive Bayes (multivariate data)
  - Multivariate Bayes
  - Decision Trees
- many more ...

Classification Approaches: k-Nearest Neighbour





Training:

- Memorize all training instances

Prediction:

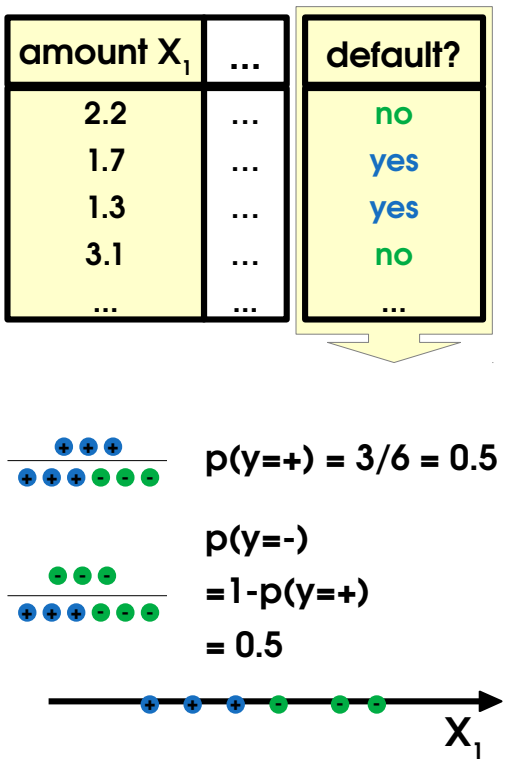
- Upon arrival of new instance $(x, ?)$, calculate its distance to all training instances
- Sort the neighbours by their distance
- Assign the label shared by majority of k nearest neighbours

Characteristics:

- This is a **lazy learner**
- No model is build, no abstraction
- Fast training, slow prediction

▸ Classifier Overview

Classification Approaches: Univariate Bayes (Using Histogram/Discretisation)



- **Calculate posterior $\mathcal{P}r(Y|X)$ using the Bayes Theorem:**

$$\mathcal{P}r(Y|X) = \frac{\mathcal{P}r(X|Y) \cdot \mathcal{P}r(Y)}{\mathcal{P}r(X)}$$

How to derive the quantities therein?

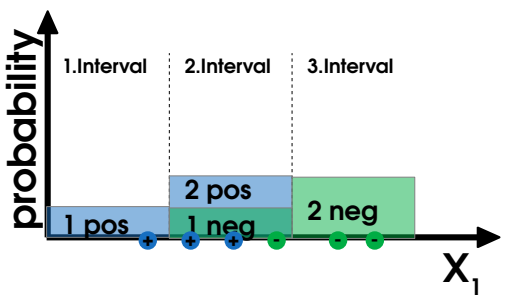
- Class prior probability:

$$\mathcal{P}r(Y = +) = \frac{|\mathcal{L}_+|}{|\mathcal{L}_+ \cup \mathcal{L}_-|}$$

▸ Classifier Overview

Classification Approaches: Univariate Bayes (Using Histogram/Discretisation)

amount X_1	...	default?
2.2	...	no
1.7	...	yes
1.3	...	yes
3.1	...	no
...	...	...



Conditional Feature Probability

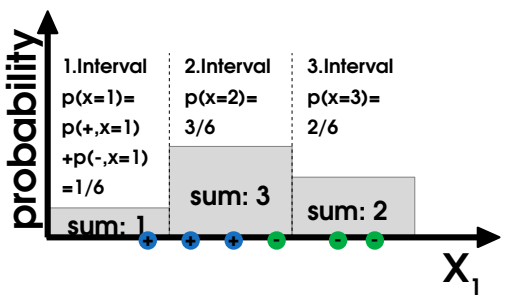
Calculate $\mathcal{P}r(X = x|Y = +)$ by discretisation / histograms:

- Discretise feature X into disjoint intervals
- In each interval, count class' occurences
- Use frequencies as estimate for $\mathcal{P}r(X = x|Y = +)$
- Repeat for the other class(es)

▸ Classifier Overview

Classification Approaches: Univariate Bayes (Using Histogram/Discretisation)

amount X_1	...	default?
2.2	...	no
1.7	...	yes
1.3	...	yes
3.1	...	no
...	...	...



(Unconditional) Feature Probability

Calculate $\mathcal{P}r(X = x)$ by **marginalisation** over y:

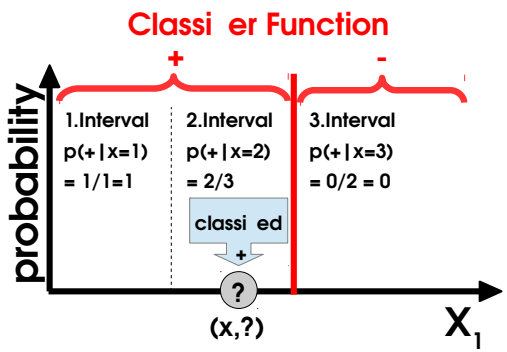
$$\mathcal{P}r(X = x) = \sum_y \mathcal{P}r(X, Y = y) \quad (5)$$

$$= \sum_y \mathcal{P}r(X|Y = y) \cdot \mathcal{P}r(Y = y) \quad (6)$$

▸ Classifier Overview

Classification Approaches: Univariate Bayes (Using Histogram/Discretisation)

amount X_1	...	default?
2.2	...	no
1.7	...	yes
1.3	...	yes
3.1	...	no
...	...	...



Posterior Probability

Calculate posterior $\mathcal{P}r(Y|X)$ using the Bayes Theorem

$$\mathcal{P}r(Y|X) = \frac{\mathcal{P}r(X|Y) \cdot \mathcal{P}r(Y)}{\mathcal{P}r(X)}$$

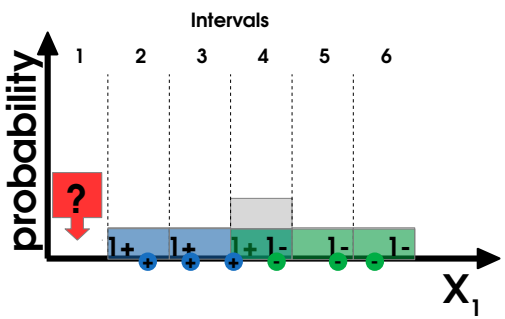
Prediction:

- Use posterior for classification:
- Assign most probable label

▸ Classifier Overview

Classification Approaches: Univariate Bayes (Using Histogram/Discretisation)

amount X_1	...	default?
2.2	...	no
1.7	...	yes
1.3	...	yes
3.1	...	no
...	...	...



Posterior Probability

Calculate posterior $\mathcal{P}r(Y|X)$ using the Bayes Theorem

$$\mathcal{P}r(Y|X) = \frac{\mathcal{P}r(X|Y) \cdot \mathcal{P}r(Y)}{\mathcal{P}r(X)}$$

Prediction:

- Use posterior for classification:
- Assign most probable label

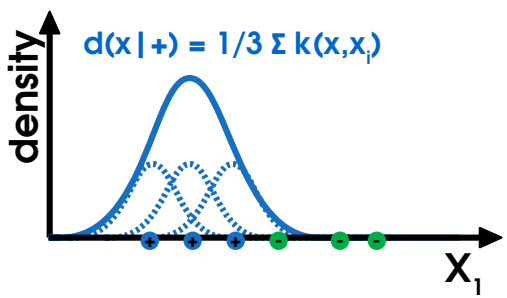
Prediction:

- Challenge: Interval size
- Too big: loss of detail
- Too small: lack of data

▶ Classifier Overview

Classification Approaches: Univariate Bayes (Using Kernel Density Estimation)

amount X_1	...	default?
2.2	...	no
1.7	...	yes
1.3	...	yes
3.1	...	no
...	...	...



- **Calculate posterior $\mathcal{P}r(Y|X)$ using the Bayes Theorem:**

$$\mathcal{P}r(Y|X) = \frac{\mathcal{P}r(X|Y) \cdot \mathcal{P}r(Y)}{Pr(X)}$$

- Derive conditional feature probability $\mathcal{P}r(X = x|Y = +)$ by **Kernel Density Estimation**
  - On the location of each instance x_i, place a (Gaussian) Kernel $k(x_i, x)$

$$K(x, x_i) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x - x_i)^2}{2\sigma^2}\right)$$

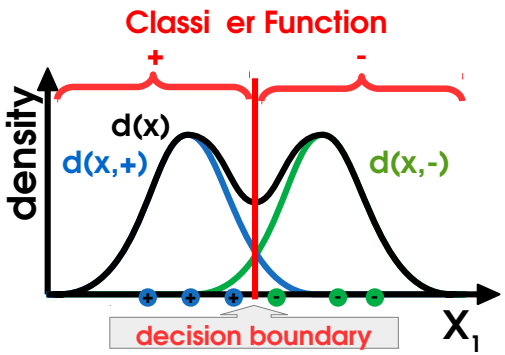
    cmp. [Hand et al., 2001, 10.2.2 Probabilistic Models for Classification]
  - Calculate $\mathcal{P}r(X = x|Y = +)$ by summing over the kernels:

$$\mathcal{P}r(X = x|Y = +) = \frac{\sum_{x_i \in \mathcal{L}_+} K(x, x_i)}{|\mathcal{L}_+|}$$

▸ Classifier Overview

Classification Approaches: Univariate Bayes (Using Kernel Density Estimation)

amount X_1	...	default?
2.2	...	no
1.7	...	yes
1.3	...	yes
3.1	...	no
...	...	...



- Repeat Kernel Density Estimation on other classes
- Unconditional density of features:

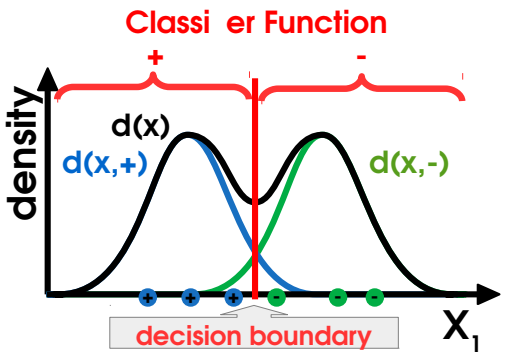
$$d(X) = d(X|Y = +) + d(X|Y = -)$$

- **Derive Decision Boundary** at intersections of joint distributions

▸ Classifier Overview

Classification Approaches: Univariate Bayes (Using Kernel Density Estimation)

amount X_1	...	default?
2.2	...	no
1.7	...	yes
1.3	...	yes
3.1	...	no
...	...	...



- Repeat Kernel Density Estimation on other classes
- Unconditional density of features:

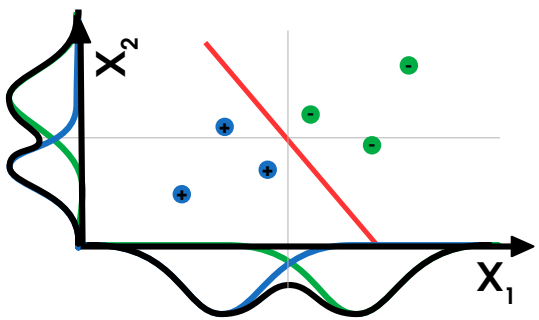
$$d(X) = d(X|Y = +) + d(X|Y = -)$$

- **Derive Decision Boundary** at intersections of posterior distributions
- **Derive Posterior** using the Bayes Theorem as above
- **Predict new data, i.e., apply f to instances from a test sample $\mathcal{U}$**

▸ Classifier Overview

Classification Approaches: Naive Bayes (on Multivariate Data)

X_1	X_2	default?
2.2	52	no
1.7	28	yes
1.3	33	yes
3.1	42	no
...	...	...



- Posterior $\mathcal{P}r(Y|X_1, X_2, \dots)$ of multivariate data:

$$\mathcal{P}r(Y|X_1, X_2) = \frac{\mathcal{P}r(X_2|Y, X_1) \cdot \mathcal{P}r(X_1|Y) \cdot \mathcal{P}r(Y)}{\mathcal{P}r(X_1, X_2)}$$

- Calculating multivariate joint probabilities poses a challenge: **Curse of dimensionality**
- **Conditional Independence Assumption:** X_i and X_j assumed to be conditionally independent given class label

$$\mathcal{P}r(X_i|X_j, Y) = \mathcal{P}r(X_i|Y)$$

- Simplifies Posterior into:

$$\mathcal{P}r(Y|X_1, X_2) \stackrel{*}{=} \frac{\mathcal{P}r(X_2|Y) \cdot \mathcal{P}r(X_1|Y) \cdot \mathcal{P}r(Y)}{\mathcal{P}r(X_1, X_2)}$$

$$\propto \mathcal{P}r(Y) \cdot \prod_{i=1}^{2} \mathcal{P}r(X_i|Y)$$

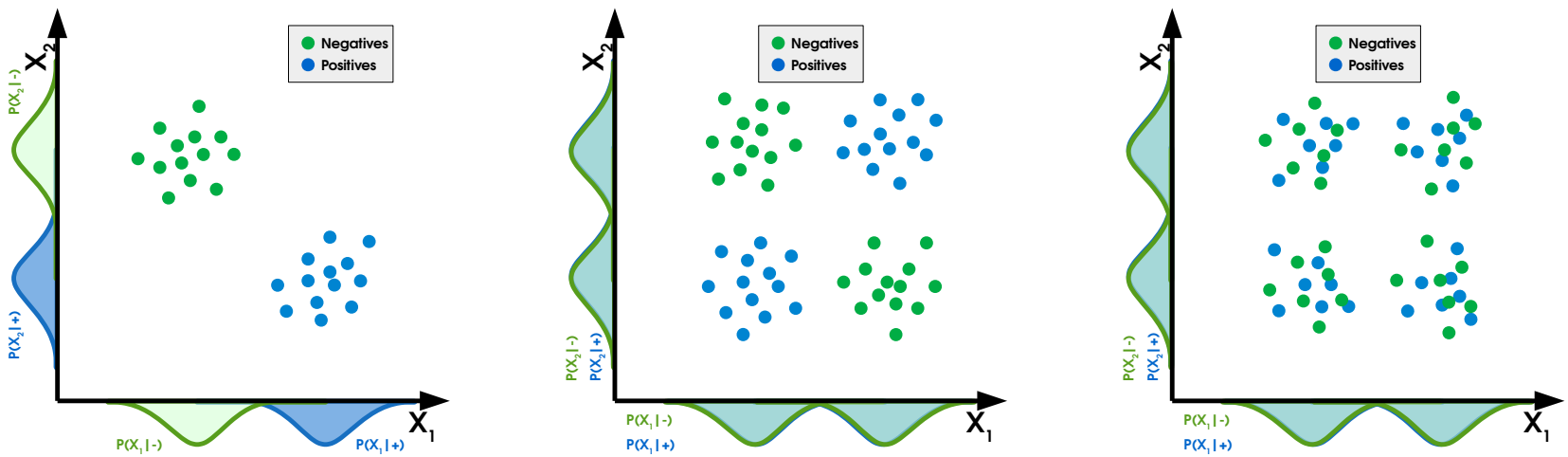
▸ Classifier Overview

Recap Chapter 2: Conditional Independence[7]

Two variables X_i and X_j are conditionally independent given a third variable Y, if for all values of X_i, X_j, Y holds

$$\mathcal{P}r(X_i, X_j|Y) = \mathcal{P}r(X_i|Y) \cdot \mathcal{P}r(X_j|Y)$$

$$\mathcal{P}r(X_i|X_j, Y) = \mathcal{P}r(X_i|Y)$$



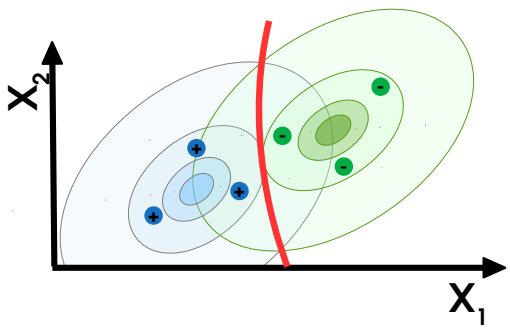
(a) (b) (c)

Figure: In scatterplots (a,c), X_1, X_2 are cond. indep. given Y. In (b), they are not.

[7] See [Hand et al., 2001, Section 4.3.1].

Classification Approaches: Multivariate Bayes

X_1	X_2	default?
2.2	52	no
1.7	28	yes
1.3	33	yes
3.1	42	no
...	...	...



- Posterior $\mathcal{P}r(Y|X_1, X_2, \dots)$ of multivariate data:

$$\mathcal{P}r(Y|X_1, X_2) = \frac{\mathcal{P}r(X_2|Y, X_1) \cdot \mathcal{P}r(X_1|Y) \cdot \mathcal{P}r(Y)}{\mathcal{P}r(X_1, X_2)}$$

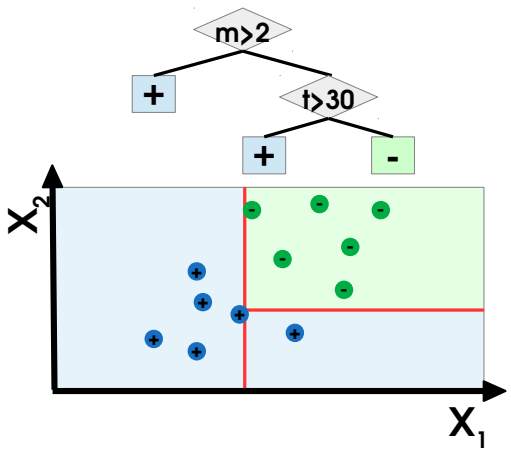
$$= \frac{\mathcal{P}r(X_1, X_2|Y) \cdot \mathcal{P}r(Y)}{\mathcal{P}r(X_1, X_2)}$$

- Alternative to Naive Bayes and the assumption of conditional independence?
- Use a **parametric model** for density estimation e.g., assume cluster structure (e.g., Gaussian Mixture)

▸ Classifier Overview

Classification Approaches: Decision Trees

X_1	X_2	default?
2.2	52	no
1.7	28	yes
1.3	33	yes
3.1	42	no
...	...	...



▸ Classifier Overview

Training:

- Iteratively build decision tree,
- by splitting on different features
- Greedy choice of split attribute
  e.g., using Information Gain

Prediction:

- Upon arrival of new instance $(x, ?)$,
  assign it to a leaf and classify based on majority class there

Classification Approaches: Overview

Lazy Learner

- **Lazy Learner** do not abstract from training data, but rather store all the training data for later comparison with new data
- (No) complexity in training, but large one in prediction!

Generative vs. Predictive

- **Generative:** Provide a (distributional) model
- **Predictive:** Learn directly a classification rule
- Learning predictive classifiers is considered the simpler problem

Single Classifiers vs. Ensembles

- **Ensembles:** Combine several classifiers
  Also denoted as Multiple Classifier Systems
- Advantage: Robustness, better accuracy
- Disadvantage: Space/Time Complexity, Loss of Interpretability

Classification Approaches: Overview

Assumptions

Many (all?) classifiers make assumptions, e.g., on the structure of the data

Examples:

- Naive Bayes: Conditional Independence Assumptions
- Clustering Assumption (all instances within a cluster are of the same class)
- Linear Relationships (e.g., regression-based classifiers)
- ...

Prediction: Evaluation Methodology

Classification Objective

- Most accurate prediction on **new** data
- Further aspects to consider in classifier evaluation:
  - Interpretability of the model and its predictions
  - Runtime (and memory consumption) for training, parameter tuning, and testing
  - Parameter tuning and sensitivity of parameters
  - Insights into model's reliability/confidence and the use of instances/features

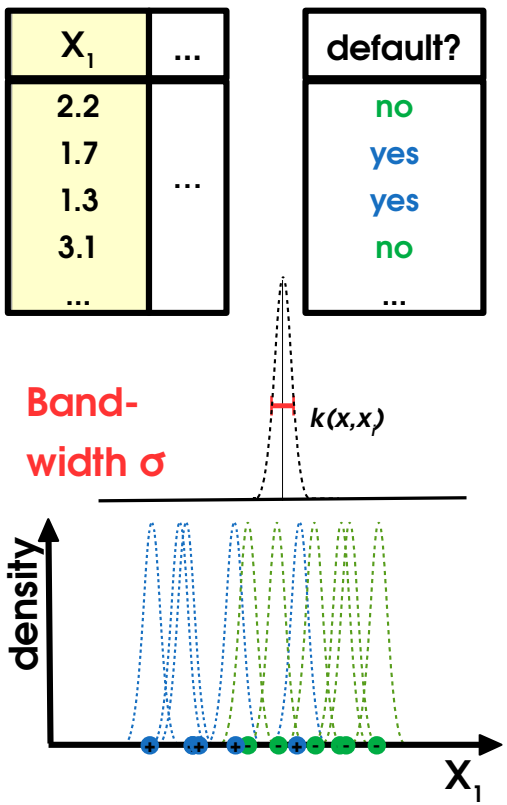
Design of Experimental Evaluations

- Split into training and hold-out (validation and test) sets
- Example: (k-Fold) Cross-Validation

Performance Measures

- Confusion Matrix
- Accuracy, Error Rate
- Area under the Receiver Operating Characteristic Curve (AUC ROC Curve)

Classifier Evaluation: Experimental Design



Challenge:

- Which parameter (or classifier) is the best?
- How to get a **realistic** performance estimation?

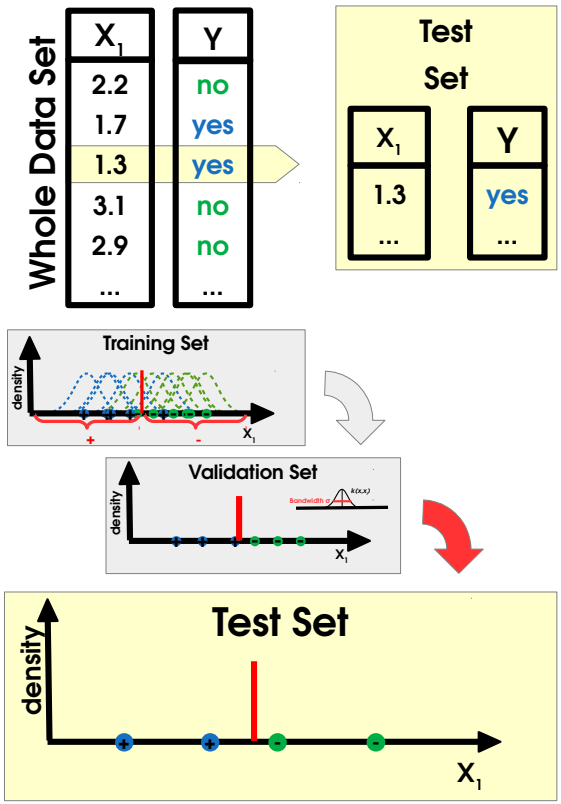
Overfitting:

- Occurs when classifier models even noise/outliers
- Does not generalise well to new data

Gold Standard:

- Use separate sets for training, validation and testing

Classifier Evaluation: Experimental Design



Using Hold-Out Sets:

- Split data into training, validation and test sets
- Training set:
  Used for training classfier
- Validation set:
  Used for evaluating classfier with given parametrisation
  i.e., comparing different parameter values
- Test set:
  Used for final evaluation of the chosen classifier and parameter combination
  Should provide a realistic estimate of the **out-of-sample** performance on new, unseen data

Classifier Evaluation: Experimental Design

Image not available due to copyright restrictions. Please refer to the source cited below.

Figure: k-Fold Cross Validation(Source: [Sharda et al., 2018, page ?])

K-Fold Cross Validation

- Split the whole data set into subsets
- Iterate over each subset, thereby use the selected subset for testing, and all

Classifier Performance Measures

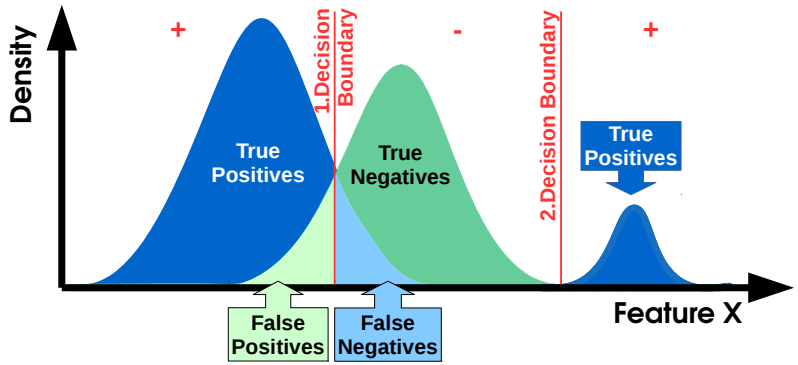


Figure: Classification and Error Rates

Classification Errors

- **True Positives (TP):** correctly classified pos.
- **True Negatives (TN):** correctly classified neg.
- **False Positive (FP):** Actual negative instance classified as positive
- **False Negative (FN):** Actual positive instance classified as negative
- **Bayes Error Rate**: Unavoidable error of a Bayes-optimal classifier
- **Question**: Possible to achieve zero FPs (or FNs)?

Classifier Performance Measures

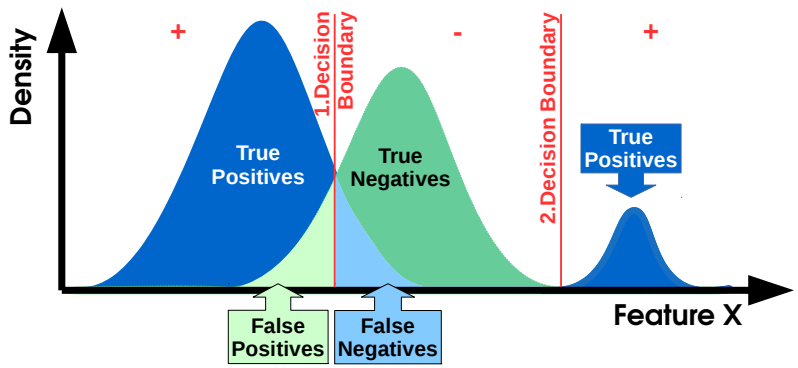


Figure: Classification and Error Rates

		Actual Class: Positive	Actual Class: Negative
Predicted Class	Positive	True Positive	False Positive
Predicted Class	Negative	False Negative	True Negative

Figure: Confusion Matrix

Confusion Matrix

- Quadratic table where
- columns = true classes
- rows = predicted classes
- Diagonal elements: Correct predictions
- Off-diagonal elements: Misclassifications
- Might be weighted by multiplying with a **cost matrix** of same size

Classifier Performance Measures

Accuracy

Share of correct classifications

$$Acc = \frac{TP + TN}{FP + FN + TP + TN}$$

Error Rate ($= 1 - Acc$)

$$Err = \frac{FP + FN}{FP + FN + TP + TN}$$

Sensitivity or True Positive Rate (TPR)

$$Sens = TPR = \frac{TP}{TP + FN}$$

Specificity or 1-(False Pos. Rate (FPR))

$$Spec = 1 - FPR = \frac{FP}{FP + TN}$$

Misclassification Loss

- Accuracy and error rate weight all errors similarly
- For unequal misclassification costs:

$$MLoss = FP \cdot Cost_{FP} + FN \cdot Cost_{FN}$$

- Hereby, the FP-cost and the FN-cost are often normalized to sum up to one: $Cost_{FP} + Cost_{FN} = 1$, such that $Cost_{FP}$ corresponds to the cost-ratio.

Classifier Performance Measures

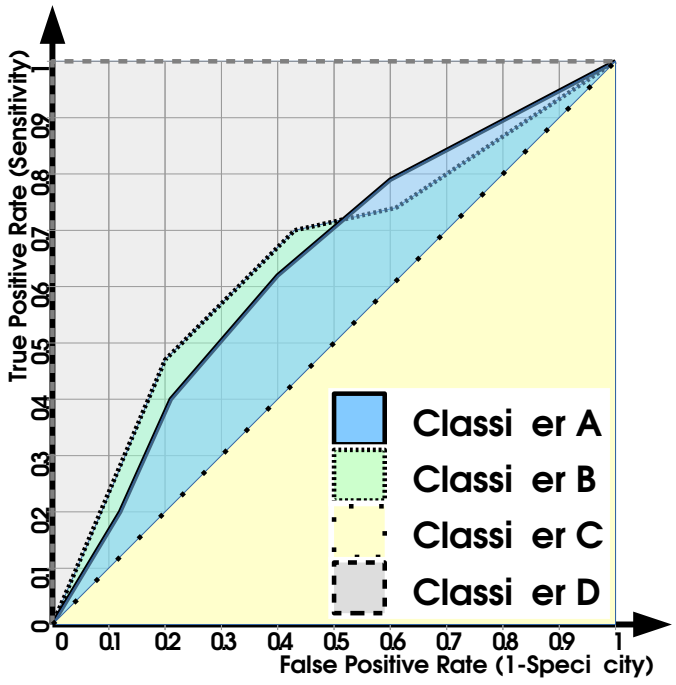


Figure: Plot of Examplary Area Under the Receiver Operating Characteristics (ROC) Curves (AUC)

AUC under the ROC Curve

The Receiver Operating Characteristic Curves

- Plots the false positive rate against the true positive rate
- the steeper the increase, the better
- Suited for problems with class imbalance
- Area under this Curve (AUC):
  Summarises the curve by integrating it
- Examplary characteristics from Fig. 24
  - Classifier D is a **perfect** classifier ($AUC = 1$), it dominates all other classifiers
  - Classifier C: similar to **random guessing** ($AUC = \frac{1}{2}$)
  - Neither classifier A is dominating B, nor classifier B is dominating A. Each is better for a different combination of FPR vs. TPR.

Prediction: Classification (and Regression)

Questions?

Outline and Summary[1]

- Predictive Analystics in BI: Data Mining Process
  See [Sharda et al., 2018, chapter 4.4]
- Overview on Data Mining Tasks
  See [Sharda et al., 2018, chapter 4.2]
- Segmentation: Cluster Analysis
  See [Sharda et al., 2018, ch. 4.5], [Wu et al., 2008], [Hand et al., 2001, ch. 9]
- Prediction: Classification and Regression
  See [Sharda et al., 2018, ch. 4.5], [Wu et al., 2008], and
  [Hand et al., 2001, ch. 10-11]
- Association: Frequent Itemset and Association Rule Mining
  See [Sharda et al., 2018, ch. 4.5], [Wu et al., 2008], [Hand et al., 2001, ch. 13]
- Mining Semi- and "Unstructured" Data
  See [Sharda et al., 2018, chapter 5]

▸ Start ▸ Appendix

[1] Note: This lecture integrates knowledge from several further sources (cited where used, except if my own).

Association Rule Mining - Motivating Questions

E-Commerce

As junior consultant, `bol.com` asks you to develop a model that recommends customers products that are "interesting" to them.

- How do you approach this problem?
- As potential data source, you can use data about the products the customer has already bought, or that other customers have bought.

[7] Illustration: Ecommerce sales, author MVCOSHOP, source: `commons.wikimedia.org`

Association Rule Mining - Motivating Questions

Space Medicine

In preparation for the first manned mission to Mars, NASA asks you to develop an onboard medical diagnosis system. In particular, it should help astronauts to identify potential illnesses from symptoms.

- How do you approach this problem?
- NASA agrees to provide you with (anonymised) data from military personel's medical records. This data contains lists symptoms, illnessess and treatment plans.

[7] Illustrations: NASA. WHO

Association Rule Mining

ID	Items in Basket
1	 
2	 
3	 
4	  
5	  
6	 
7	  
8	  

[7] Image Source: `Amazon.com`

Market Basket Analysis

- Identify interesting relationships (affinities)
- Between variables (items or events)
- Employs unsupervised learning, no output variable
- Input: transaction data from a point-of-sale
- Output: Most frequent affinities among items

Example:

"Customers who bought  *,*

also bought  *in 100% of the time."*

 ⇒ 

A Generic Rule

$$X \Rightarrow Y[S\%, C\%]$$

where

- X,Y: items (products, ...)
- X: Left-hand-side (LHS)
- Y: Right-hand-side (RHS)
- S: Support: how often X and Y go together
- C: Confidence: how often Y go together with the X

Example:

$$\mathit{LaptopComputer}, \mathit{AntivirusSoftware} \Rightarrow \mathit{ExtendedServicePlan}[30\%, 70\%]$$

Frequent Itemset Mining: Approaches

Overview

Several algorithms are developed for discovering (identifying) association rules

- Apriori
- Eclat
- FP-Growth
- and derivatives and hybrids of the three ...

Identify the frequent itemsets, which are then converted to association rules

Apriori Algorithm

- Finds subsets that are common to at least a minimum number of the itemsets
- Uses a bottom-up approach
  - frequent subsets are extended one item at a time (the size of frequent subsets increases from one-item subsets to two-item subsets, then three-item subsets, and so on), and
  - groups of candidates at each level are tested against the data for minimum support

Image not available due to copyright restrictions. Please refer to the source cited below.

Apriori Algorithm

Support

$$support(A) = \frac{|\{t \in \mathcal{T} \wedge contains(t, A)\}|}{|\mathcal{T}|}$$

$$confidence(A \rightarrow B) = \frac{support(A \cup B)}{support(A)}$$

Apriori Algorithm (Simplified)

- Input: baskets $\mathcal{T}$, set of items $\mathcal{I}$, threshold σ
- $L_1 \leftarrow$ All items that occur $\geq \sigma|\mathcal{T}|$
- Repeat
  - Expand each itemset $x \in L_i$ by an item from $L_1 - x$
  - Add this itemset to set of candidates C_{i+1}
  - $L_{i+1} \leftarrow$ All itemsets in C_{i+1} that occur $\geq \sigma|\mathcal{T}|$
- Until $L_{i+1} = \varnothing$
- For each $x \in \bigcup L_i$:
- Create rule $A \rightarrow B$ with $A \cup B = x, A \cap B = \varnothing$

Frequent Itemset Mining Example

BID	Items in Basket
1	 
2	 
3	 
4	  
5	  
6	 
7	  
8	  

Items:

-  Ides of March (IM)
-  Killer Elite (KE)
-  Salt (SA)
-  Time Traveler's Wife (TTW)
-  What's your Number (WYN)
-  Lake House (LH)

- **Recommendations** for a customer who has "Ides of March" already in the basket?
- **Support** of all itemsets?
- **A-Priori Algorithm**
  Support: $\sigma = 50\%$
  Confidence: $c = 25\%$

Questions?

Outline and Summary[1]

- Predictive Analystics in BI: Data Mining Process
  See [Sharda et al., 2018, chapter 4.4]
- Overview on Data Mining Tasks
  See [Sharda et al., 2018, chapter 4.2]
- Segmentation: Cluster Analysis
  See [Sharda et al., 2018, ch. 4.5], [Wu et al., 2008], [Hand et al., 2001, ch. 9]
- Prediction: Classification and Regression
  See [Sharda et al., 2018, ch. 4.5], [Wu et al., 2008], and
  [Hand et al., 2001, ch. 10-11]
- Association: Frequent Itemset and Association Rule Mining
  See [Sharda et al., 2018, ch. 4.5], [Wu et al., 2008], [Hand et al., 2001, ch. 13]
- Mining Semi- and "Unstructured" Data
  See [Sharda et al., 2018, chapter 5]

▸ Start ▸ Appendix

[1] Note: This lecture integrates knowledge from several further sources (cited where used, except if my own).

Outlook: Prescriptive Analytics

Next Lecture

- Prescriptive Analytics: Data Mining: [Sharda et al., 2018, chapter 6]

Any More Questions?

Thank you!

Appendix

Bibliography I


Backhaus, K., Erichson, B., Plinke, W., and Weiber, R. (2006).
Multivariate Analysemethoden: Eine anwendungsorientierte Einführung.
Springer, 11 edition.


Hand, D. J., Mannila, H., and Smyth, P. (2001).
Principles of Data Mining.
Adaptive Computation and Machine Learning. The MIT Press.


Laney, D. (2012).
Information economics, big data and the art of the possible with analytics.
Presentation copyrighted by Gartner.


Manning, C. D., Raghavan, P., and Schütze, H. (2008).
Introduction to Information Retrieval.
Cambridge University Press.


Sharda, R., Delen, D., and Turban, E. (2018).
Business Intelligence, Analytics, and Data Science: A Managerial Perspective.
Pearson, 4 edition.


Sherman, R. (2015).
Business Intelligence Guidebook: From Data Integration to Analytics.
Morgan Kaufmann.

Bibliography II

Winston, W. L. (1997).
Operations Research: Applications and Algorithms.
Wadsworth Publishing Company, 3rd edition edition.

Wu, X., Kumar, V., Quinlan, J. R., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G. J., Ng, A., Liu, B., and Yu, P. S. e. a. (2008).
Top 10 algorithms in data mining.
Knowledge and Information Systems, 14:1–37.