# Mathematische Grundlagen III

# Cluster-Analyse

*Garance PARIS*

*14. Juli 2011*

## *Clustering vs. Klassifikation*

- Bei der Klassifikation werden Instanzen vordefinierten Klassen zugeordnet
- Beim Clustering entdeckt der Algorithmus “natürliche” Klassen, die die Instanzen in Gruppen mit ähnlichen Eigenschaften teilen
- Deutscher Begriff: *Ballungsanalyse*

## *Betreutes und unbetreutes Lernen*

Betreutes Lernen (Engl. “supervised learning”):

Der Algorithmus lernt, indem er Eingabe-Ausgabe-Beispiele analysiert, die die Rolle eines “Lehrers” übernehmen

Unbetreutes Lernen (Engl. “unsupervised learning”):

Vorkategorisierte Beispiele sind nicht verfügbar

## Betreutes und unbetreutes Lernen

### Halb-betreutes Lernen (Engl. “semi-supervised”)

- Nur ein Teil der Trainingsbeispiele ist bereits mit einer Kategorie annotiert
- Kombination aus betreutes und unbetreutes Lernen

### Bestärkendes Lernen (Engl. “reinforcement learning”)

- Die Lernkomponente ist Teil eines Systems oder Agents, das lernen soll, wie in potenziell auftretenden Situationen zu handeln ist
- Der Algorithmus erfährt aber nicht explizit, was die richtige Kategorisierung wäre
- Stattdessen lernt er durch Feedback von der Umwelt, das ihm in der Form von Belohnung und Bestrafung gegeben wird, den Erfolg des Agenten zu maximieren

## *Wozu Clustering verwenden?*

### Explorative Datenanalyse

Um ein Gefühl für die vorhandenen Daten
und ihre Eigenschaften zu gewinnen

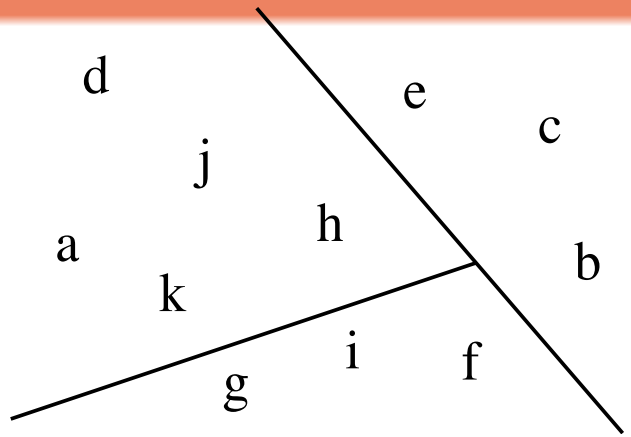
### Binning

Instanzen entdecken, die sich ähnlich verhalten
und daher ähnlich behandelt werden können,
um Abhilfe bei Sparse-Data-Problemen zu schaffen

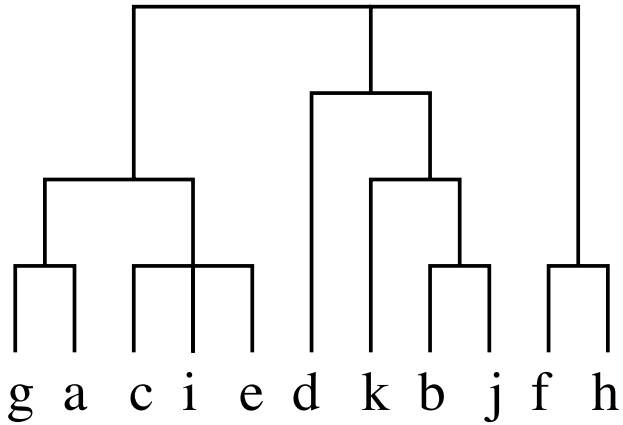
- In einem Korpus findet man die Sequenzen
  *“am Donnerstag”* und *“am Freitag”*
  sowie *donnertags* und *freitags*
- Außerdem hat man *“am Montag”*,
  aber *montags* kommt nicht vor
- Wenn wir wissen, dass *Donnerstag*, *Freitag*
  und *Montag* sich syntaktisch ähnlich verhalten,
  können wir *montags* inferieren

# Verschiedene Clustering-Arten

## Flach



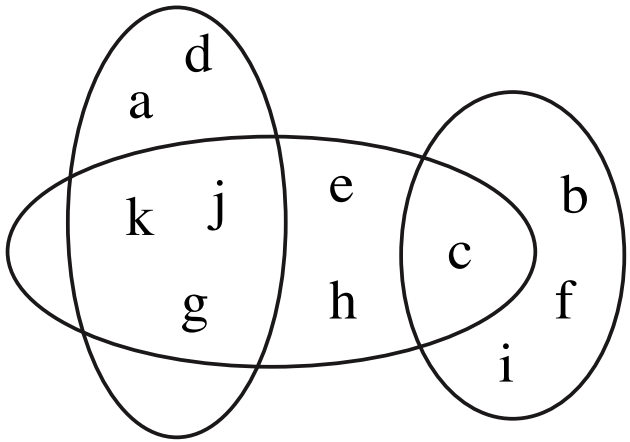

## Hierarchisch (Dendrogramme)



# Verschiedene Clustering-Arten

## Disjunktiv

Instanzen können mehreren Cluster angehören




## Probabilistisch oder “soft”

Für jeden Cluster wird eine Wahrscheinlichkeit angegeben, dass eine Instanz ihm zugeordnet wird

| | 1 | 2 | 3 |
|---|---|---|---|
| a | 0.4 | 0.1 | 0.5 |
| b | 0.1 | 0.8 | 0.1 |
| c | 0.3 | 0.3 | 0.4 |
| d | 0.1 | 0.1 | 0.8 |
| e | 0.4 | 0.2 | 0.4 |
| f | 0.1 | 0.4 | 0.5 |
| g | 0.7 | 0.2 | 0.1 |
| h | 0.5 | 0.4 | 0.1 |

# Iteratives distanz-basiertes Clustering



## *Intuition bei k-means*

- Bestimme $k$, die Anzahl der gewünschten Cluster
- Wähle $k$ beliebige Punkte als Cluster-Zentren aus
- Weise jede Instanz dem nächsten Cluster-Zentrum zu
- Berechne den Mittelpunkt für jeden Cluster und verwende ihn als neues Zentrum
- Weise alle Instanzen wieder dem nächsten Cluster-Zentrum zu
- Iteriere, bis alle Cluster stabil sind

# Iteratives distanz-basiertes Clustering

## *Der Algorithmus*

- Jede Instanz $\vec{x}$ im Training Set wird als Vektor mit einem Wert pro Attribut repräsentiert

$$\vec{x} = (x_1, x_2, \ldots, x_n)$$

- Die Distanz zwischen zwei Vektors $\vec{x}$ and $\vec{y}$ ist definiert als (euklidische Distanz):

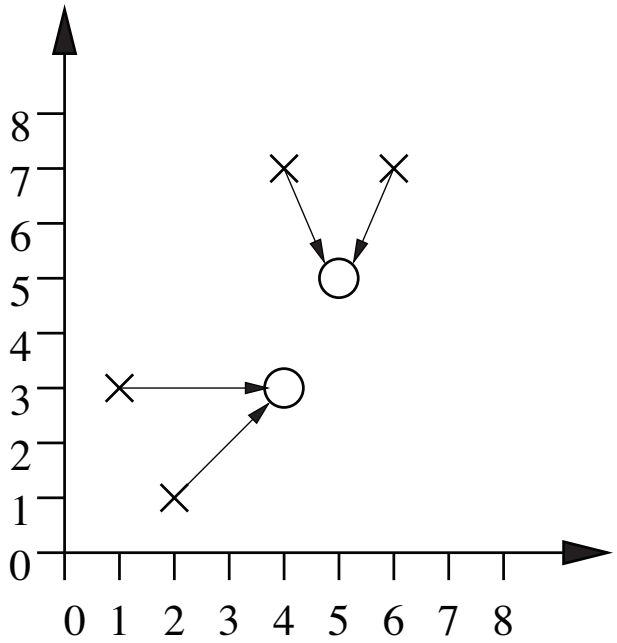
$$|\vec{x} - \vec{y}| = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

- Der Mittelpunkt $\vec{\mu}$ einer Menge Vektoren $c_j$ ist definiert als:

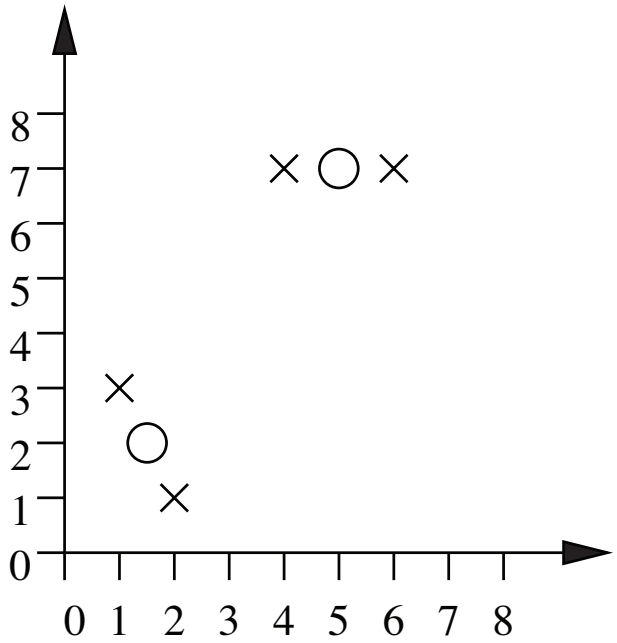
$$\vec{\mu} = \frac{1}{|c_j|} \sum_{\vec{x} \in c_j} \vec{x}$$

# Iteratives distanz-basiertes Clustering

## *Beispiel*




Die Instanzen (Kreuzchen) werden anfangs zum nächsten Cluster-Zentrum (Kreise) zugewiesen




Der Mittelpunkt jedes Cluster wird dann berechnet und als neuen Zentrum verwendet

# Iteratives distanz-basiertes Clustering

## *Eigenschaften von k-means*

- Flaches Clustering-Verfahren
- Konzeptuell einfach
- Effizient bei großen Datenmengen
- Nicht geeignet für Nominaldaten
- Findet nur ein lokales Maximum, keine globales
- Die Cluster hängen sehr von der initialen Wahl der Cluster-Zentren
- Kann man für hierarchisches Clustering verwenden
- Andere Distanzmaße können verwendet werden (z. B. der Cosinus)
- Alternative: der EM-Algorithmus
  - Legt iterativ die Parameter eines Modells so fest, dass es die gesehenen Daten optimal erklärt
  - HMMs sind eine Anwendung des EM-Algorithmus