Vorlesungsplan

- 17.10. Einleitung
- 24.10. Ein- und Ausgabe
- 31.10. Reformationstag, Einfache Regeln
- 7.11. Naïve Bayes, Entscheidungsbäume
- 14.11. Entscheidungsregeln, Assoziationsregeln
- 21.11. Lineare Modelle, Instanzbasiertes Lernen
- 28.11. **Clustering**
- 5.12. Evaluation
- 12.12. Lineare Algebra für Data Mining
- 19.12. Statistik für Data Mining
- 9.1. Entscheidungsbäume, Klassifikationsregeln
- 16.1. Lineare Modelle, Numerische Vorhersage
- 23.1. Clustering
- 30.1. Attribut-Selektion, Diskretisierung, Transformationen
- 6.2. Kombination von Modellen, Lernen von nicht-klassifizierten Beispielen

Clustering-Problem

- Gegeben
  - Instanzen ohne Klasseninformation
  - Ähnlichkeits/Distanzmaß
- Gesucht
  - Einteilung der Instanzen in natürliche Gruppen
  - Instanzen aus derselben Gruppe sollen ähnlich sein => hohe Intra-Cluster Ähnlichkeit
  - Instanzen aus verschiedenen Gruppen sollen unähnlich sein => niedrige Inter-Cluster Ähnlichkeit

Varianten des Cluster-Problems

- Partitionierende Verfahren
  - Gruppen sind disjunkt
  - Repräsention der Cluster
    - durch einzelne Repräsentanten
    - durch die Instanzen, die zum Cluster gehören
- Hierarchische Verfahren
  - Hierarchie von verschachtelten Gruppen
- Probabilistische Verfahren
  - Instanz gehört zu jedem Cluster mit einer gewissen Wahrscheinlichkeit
- Moderne Varianten

k-Means

- Partitionierendes iteratives Verfahren
- Eingabe
  - Instanzen: $X = \{x_1, \ldots, x_n\} \subset \mathbb{R}^d$
  - Anzahl der Cluster: k
- Ausgabe
  - Cluster-Repräsentanten: $W = \{w_1, \ldots, w_k\} \subset \mathbb{R}^d$
- Findet lokales Optimum bezüglich des Fehlers

$$D(X, W) = \sum_{i=1}^{n} \|x_i - w_{I(x_i)}\|_2^2$$

$$I(x) = min\{j : \|x - w_j\|_2 \leq \|x - w_{j'}\|_2, \ \forall j' \in \{1, \ldots, k\}\}$$

k-Means Algorithmus

1. Initialisiere $w_1^{(0)}, \ldots, w_k^{(0)} \in \mathbb{R}^d, t = 0$
2. Berechne für alle $j = 1 \ldots, k$

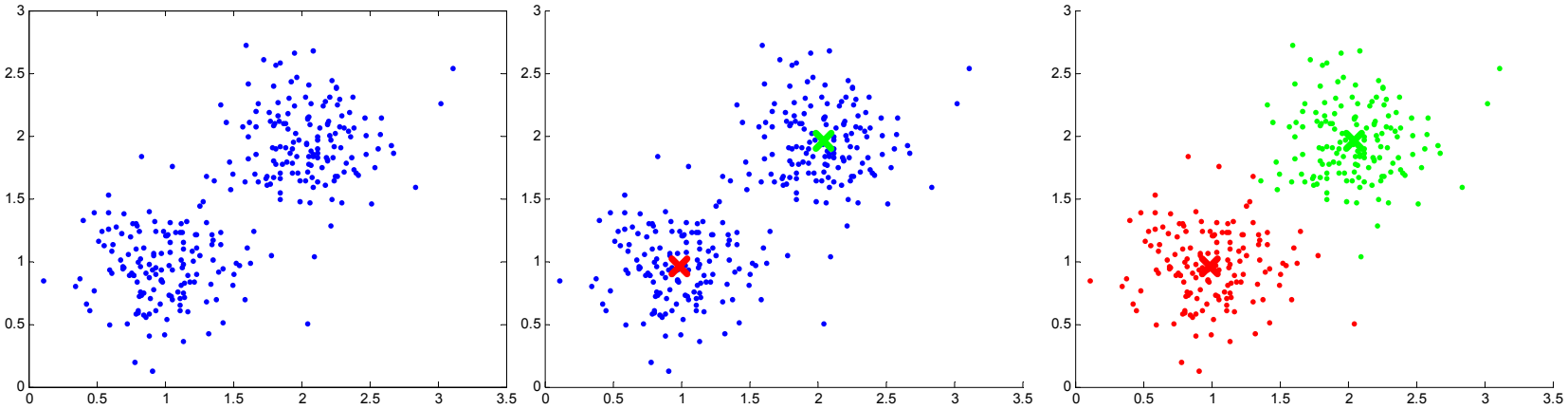
$$w_j^{(t+1)} = \frac{1}{|\{x_i : I(x_i, t) = j\}|} \cdot \sum_{x_i : I(x_i,t)=j} x_i$$

3. $$D^{(t+1)} = \sum_{i=1}^{n} \|x_i - w_{I(x_i,t+1)}^{(t+1)}\|_2^2$$
4. Stop falls $\frac{D^{(t)} - D^{(t+1)}}{D^{(t)}} \leq \epsilon$

   Sonst t=t+1 und gehe zu 2.

Beispiel

- Instanzen werden Repräsentanten mittels Nächster-Nachbar-Regel zugeordnet



- Ergebnis hängt von der Initialisierung ab
- Laufzeit: Iterationen * O(kn)

Verbesserung, LBG-U

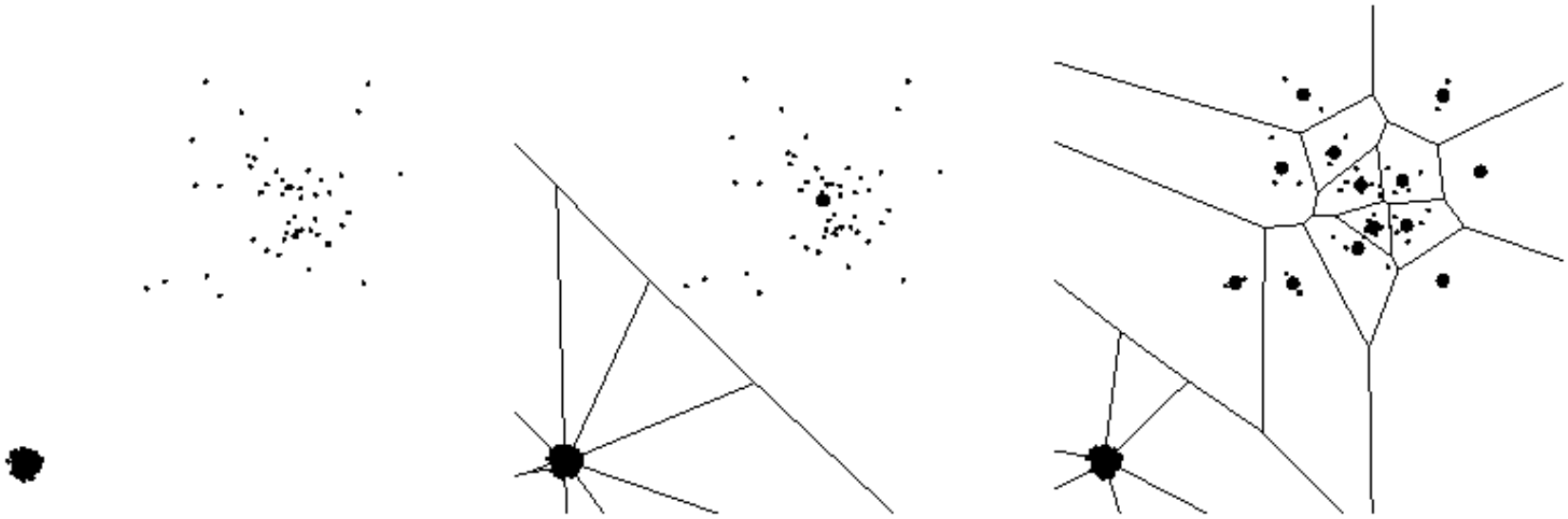
- Idee
  - verschiebe schlecht plazierte Repräsentanten mit nicht lokalen Sprüngen in Gebiete mit hohem Fehler

$$\text{Utility } U(w_j) = D(X, W \setminus \{w_j\}) - D(X, W)$$
$$= \sum_{x \in R_j} d(w_{j'}, x) - d(w, x)$$

$$\text{Fehler } E(w_j) = \frac{1}{|R_j|} \sum_{x \in R_j} \|x - w_j\|$$

  - Wähle den Repräsentant w mit der kleinsten Utility (Nützlichkeit) und verschiebe ihn in die Nähe des Repräsentanten w' mit dem größten Fehler.
- Wende k-Means wiederholt an bis zur Konvergenz

Beispiel



a) Datenmenge $\mathcal{D}$ mit 1000 Elementen

b) LBG-Kodebuch, $\sqrt{\text{MSE}}$: 0.001

c) „besseres“ Kodebuch, $\sqrt{\text{MSE}}$: 0.00009

Fehlermaß

http://www.neuroinformatik.ruhr-uni-bochum.de/ini/VDM/research/gsn/DemoGNG/GNG.html

Diskussion k-Means

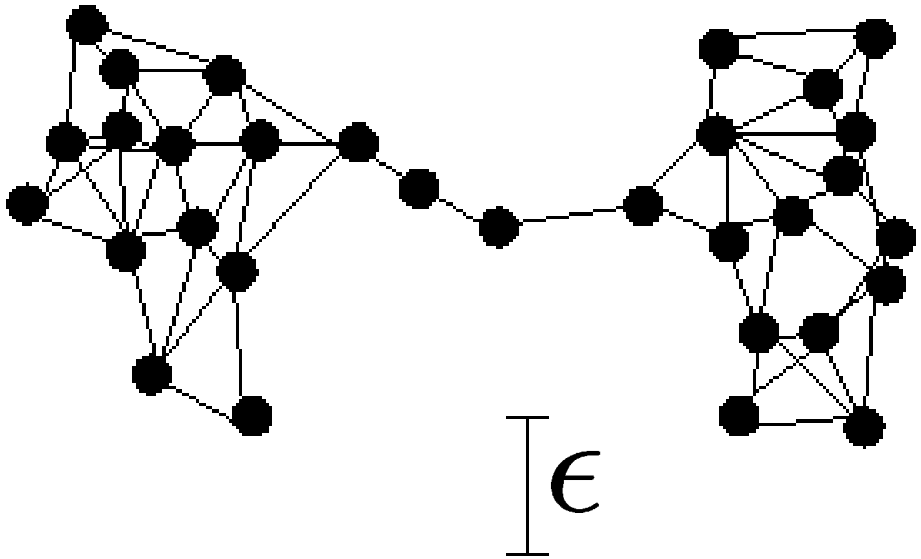
- Vorteile
  - schnelle Laufzeit
  - klare Optimierungsfunktion
    - Ziel: reduziere die Daten auf Repräsentanten
- Nachteile k-Means
  - kompakte Cluster, natürliche Cluster können geteilt werden
  - Anzahl der Cluster k ist vorgegeben
  - keine theoretischen Grundlagen

Single Linkage

- Single Linkage (partitionierend)
  - Instanzen sind Knoten eines Graphen
  - Kante existiert, falls $d(x, x') \leq \epsilon$
  - Cluster sind die Zusammenhangs-komponenten (ZHK) des Graphen
- Eigenschaften
  - Cluster nur durch Instanzen beschrieben
  - erkennt geformte Cluster

Problem

- Verkettungseffekt

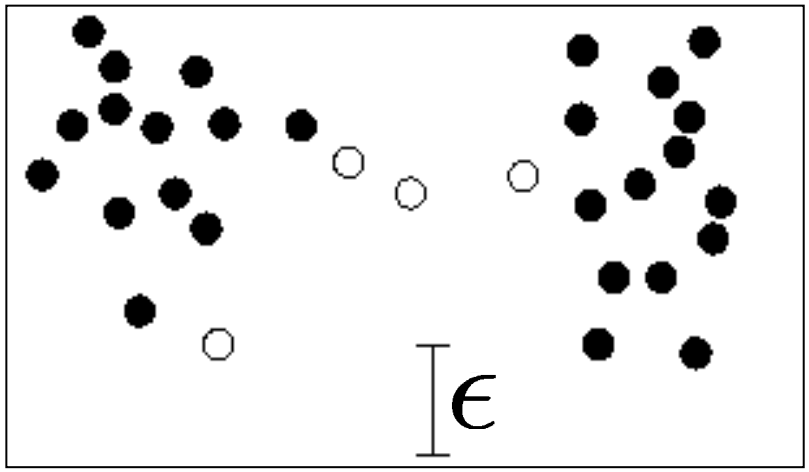


- Abhilfe
  - entferne alle Punkte x mit weniger als k Punkten in ihrer Epsilon Nachbarschaft: $|\{x' : d(x, x') \leq \epsilon\}| \leq k$

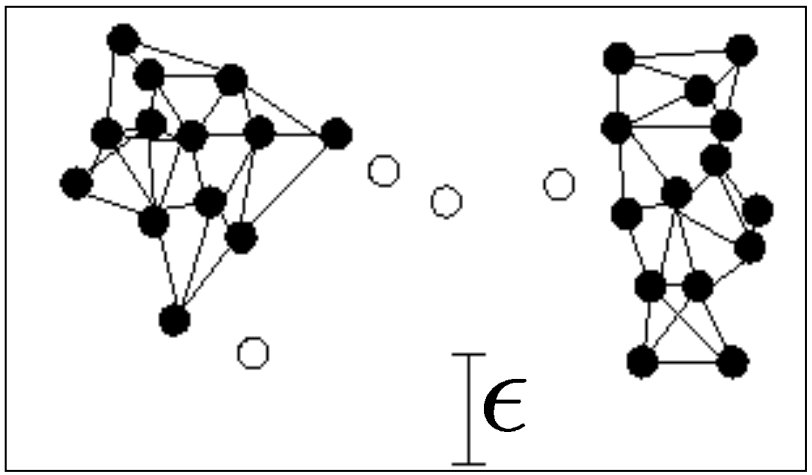
Beispiel

- Wisharts Methode (k=4, epsilon=d)

Reduziere die Daten



Wende Single Linkage an



Diskussion Single-Linkage

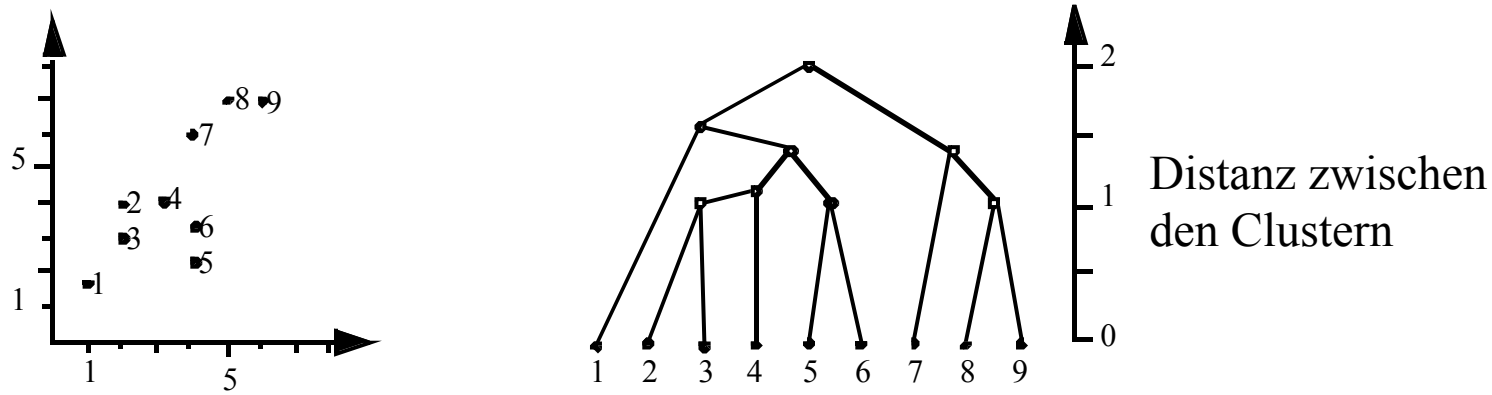
- Vorteile
  - erkennt natürliche geometrisch geformte Cluster
  - Anzahl der Cluster wird vom Verfahren bestimmt
- Nachteile
  - Laufzeit ist quadratisch
    - bei niedrig-dimensionalen Instanzen kann ein Suchindex (z.B. R*-Baum) zur Beschleunigung genutzt werden
  - Verkettungseffekt
  - Cluster sind keine homogene Gruppe
  - keine theoretischen Grundlagen

Hierarchische Verfahren

- Ziel
  - Konstruktion einer Hierarchie von Clustern (*Dendrogramm*), so daß immer die Cluster mit minimaler Distanz verschmolzen werden
- Dendrogramm
  - ein Baum, dessen Knoten die Cluster repräsentieren, mit folgenden Eigenschaften:
    - die Wurzel repräsentiert die ganze DB
    - die Blätter repräsentieren einzelne Objekte
    - ein innerer Knoten repräsentiert die Vereinigung aller Objekte, die im darunterliegenden Teilbaum repräsentiert werden

Hierarchische Verfahren

- Beispiel eines Dendrogramms



- Typen von hierarchischen Verfahren
- Bottom-Up Konstruktion des Dendrogramms (agglomerative)
- Top-Down Konstruktion des Dendrogramms (divisiv)

Agglomeratives hierarchisches Clustering (bottom up)

1. Bilde initiale Cluster, die jeweils aus einem Objekt bestehen und bestimme die Distanzen zwischen allen Paaren dieser Cluster.

2. Bilde einen neuen Cluster aus den zwei Clustern, welche die geringste Distanz zueinander haben.

3. Bestimme die Distanz zwischen dem neuen Cluster und allen anderen Clustern.

4. Wenn alle Objekte sich in einem einzigen Cluster befinden: Fertig,
   andernfalls wiederhole ab Schritt 2.

Distanzfunktionen für Cluster

- Gegeben Distanzfunktion dist(x,y) für Paare von Objekten
- Seien X, Y Cluster, d.h. Mengen von Objekten.
- Centroid-Link

$$\text{centroidLinkDist}(X,Y) = \text{dist}(\bar{x},\bar{y}), \quad \bar{x} = \frac{1}{|X|}\sum_{x\in X} x, \quad \bar{y} = \frac{1}{|Y|}\sum_{y\in Y} y$$

- Single-Link

$$\text{singleLinkDist}(X,Y) = \min_{x\in X, y\in Y} dist(x,y)$$

- Complete-Link

$$\text{completeLinkDist}(X,Y) = \max_{x\in X, y\in Y} \text{dist}(x,y)$$

- Average-Link

$$\text{averageLinkDist}(X,Y) = \frac{1}{|X|\cdot|Y|}\cdot \sum_{x\in X, y\in Y} dist(x,y)$$

Eingabe

- Daten-Matrix (N x d)
  - große Datenmengen
  - Eigenschaften repräsentieren das Objekt nicht vollständig
  - Explizite Ähnlichkeitsfunktion (Distanzfkt.) notwendig
- Ähnlichkeits/Distanz-Matrix, (N x N)
  - kleine Datenmengen
  - Ähnlichkeitsfunktion nicht notwendig, da alle Werte-Paare schon gegeben sind.
  - Sehr komplexe Beziehungen zwischen den Objekten möglich
- N... Anzahl der Datenobjekte, d ... Eigenschaften

Diskussion: Hierarchische Verfahren

- Vor- und Nachteile

+ erfordert keine Kenntnis der Anzahl k der Cluster
+ findet nicht nur ein flaches Clustering, sondern verschachtelte Cluster
+ ein einzelnes Clustering kann aus dem Dendrogramm gewonnen werden, z.B. mit Hilfe eines horizontalen Schnitts durch das Dendrogramm (erfordert aber wieder Anwendungswissen)
+ geeignet für komplexe Objekte mit aufwändigen Distanzfunkionen

- Ineffizienz Laufzeitkomplexität von mindestens $O(n^2)$ für n Objekte
- Auswahl der Distanzfunktion

- Software:
  - http://www-users.cs.umn.edu/~karypis/cluto/ (Linux, Sun, Windows)