Kapitel ML: X

X. Cluster-Analyse

- Einordnung Data Mining
- Einführung in die Cluster-Analyse
- Hierarchische Verfahren
- Iterative Verfahren
- Dichtebasierte Verfahren
- Cluster-Evaluierung

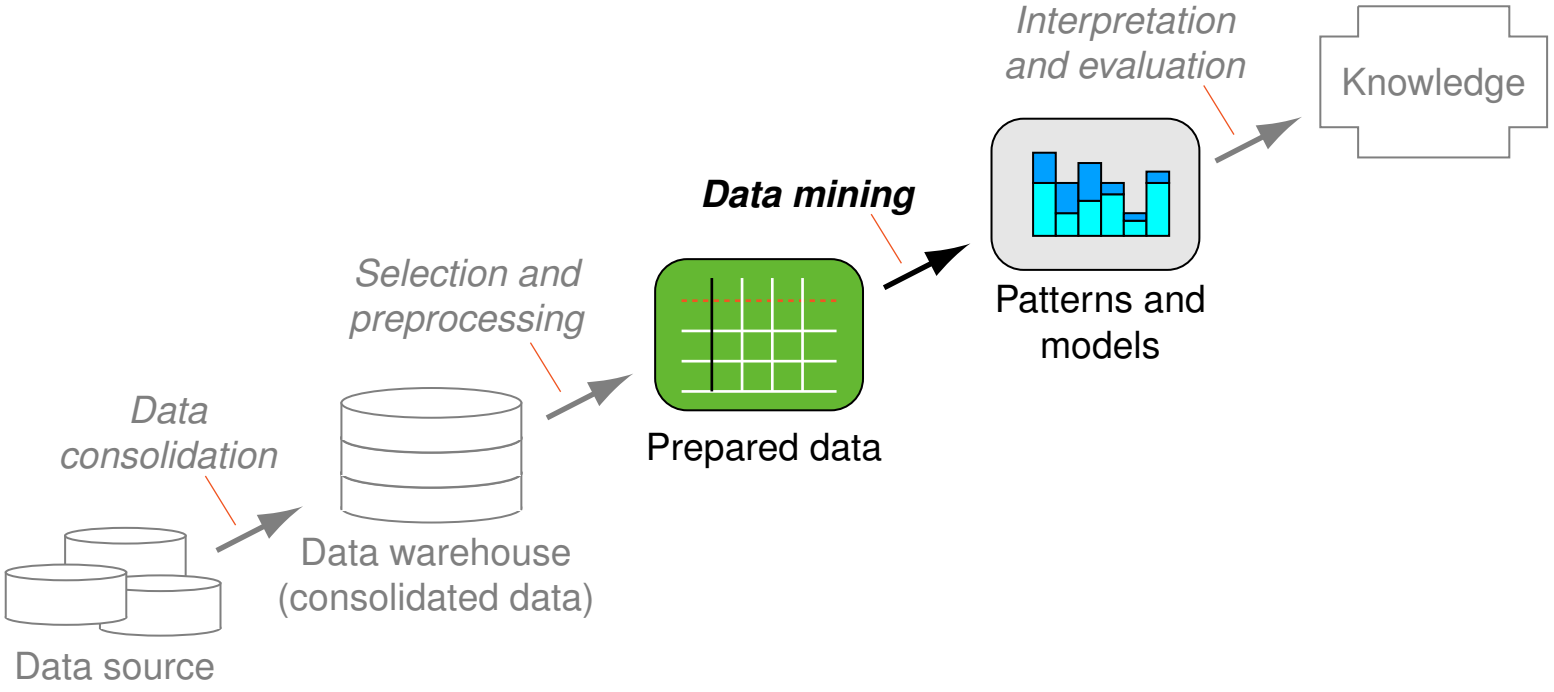
Einordnung Data Mining

Definition 1 (Data Mining)
Unter Data Mining versteht man das systematische, in der Regel automatisierte oder halbautomatische Entdecken und Extrahieren bislang unbekannter Zusammenhänge aus großen Mengen von Daten.

Data Mining umfasst folgende Schritte:

1. Aufgabendefinition
2. Datenselektion
3. Datenvorbereitung und -transformation
4. Mustererkennung
5. Kommunikation, Präsentation

Einordnung Data Mining



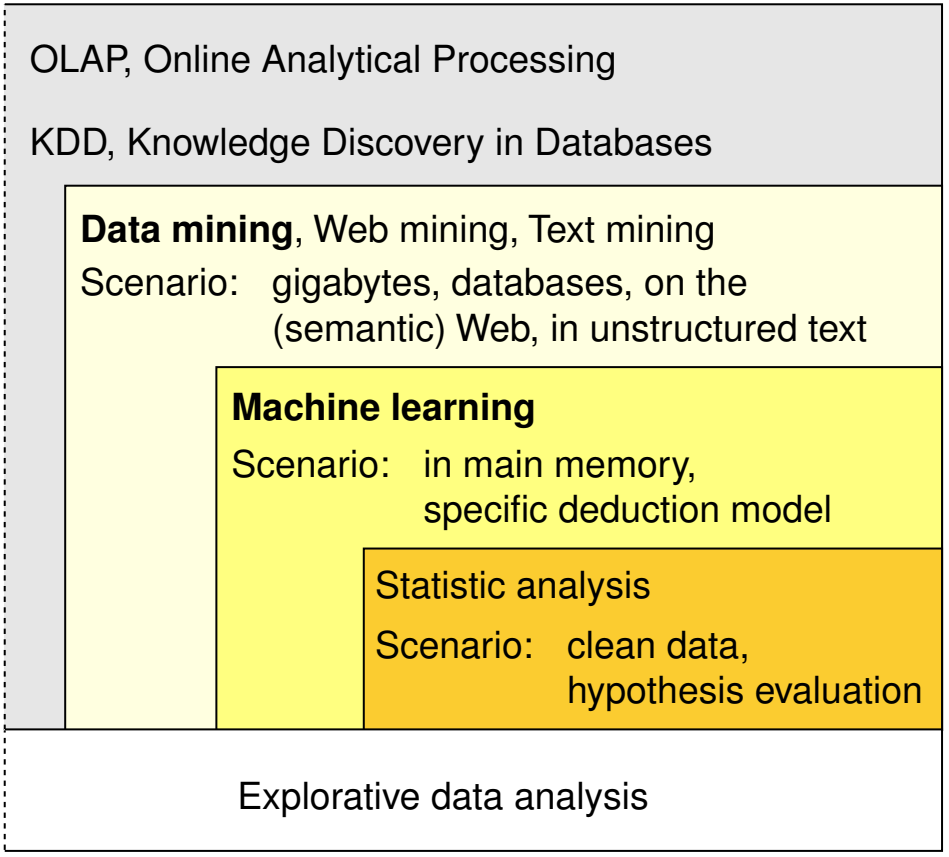
Definition 2 (Knowledge Discovery in Databases, KDD)
Wissensentdeckung in Datenbanken *(Knowledge Discovery in Databases)* ist der nichttriviale Prozess der Identifikation gültiger, neuer, potentiell nützlicher und schlussendlich verständlicher Muster in großen Datenbeständen.

[vgl. Fayyad 1996, Wrobel 1998]

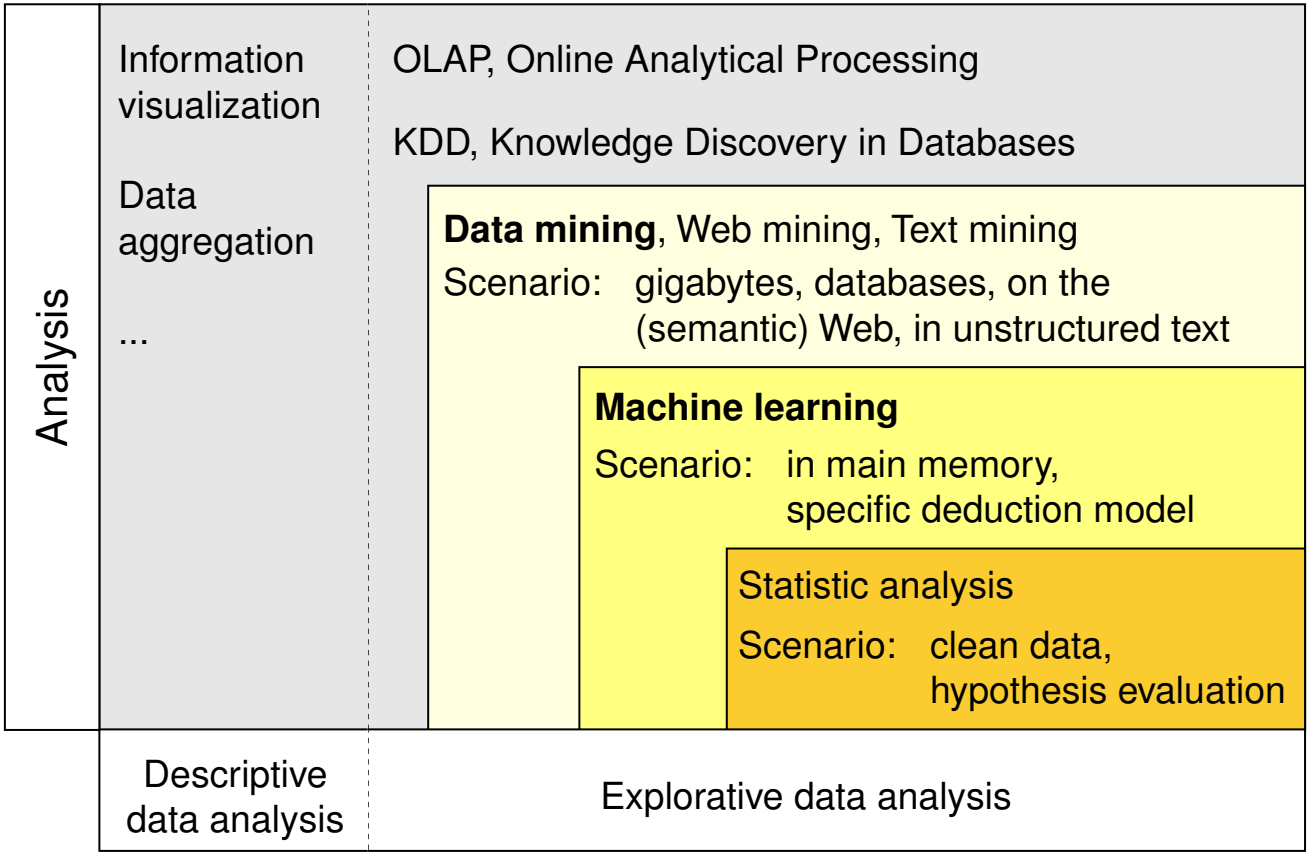
Bemerkungen:

- Data-Mining-Techniken werden der *explorativen Datenanalyse* zugeordnet. Ziel der explorativen Datenanalyse ist – über die Darstellung der Daten hinaus – die Suche nach Strukturen und Besonderheiten. Die explorative Datenanalyse wird eingesetzt, wenn die Fragestellung oder die Wahl eines geeigneten statistischen Modells unklar ist.
- In der Data-Mining-Definition wird auf den Begriff der Information verzichtet: Data Mining wird der sigmatischen Ebene der Semiotik zugeordnet. Die Interpretation der entdeckten Muster, also die im Rahmen der explorativen Datenanalyse stattfindende Auseinandersetzung mit Informationen im Sinne eines *subjektiven Wissenszuwachses*, die auf der pragmatischen Ebene abläuft, gehört in das Gebiet des Knowledge Discovery in Databases, KDD.
- Vor allem im kommerziellen Bereich wird der Begriff des Data Mining synonym zu Wissensentdeckung in Datenbanken (KDD) verwendet. Data Mining ist aber nur ein Teilschritt innerhalb des KDD-Prozesses, nämlich der Analyseschritt zur Mustererkennung.
- Unter Web Mining versteht man die Übertragung von Techniken des Data Mining zur (teil)automatischen Extraktion von Informationen aus dem Internet, speziell dem World Wide Web. Mit Text Mining wird die Entdeckung neuer und für den Benutzer relevanter Informationen aus Textdaten bezeichnet.

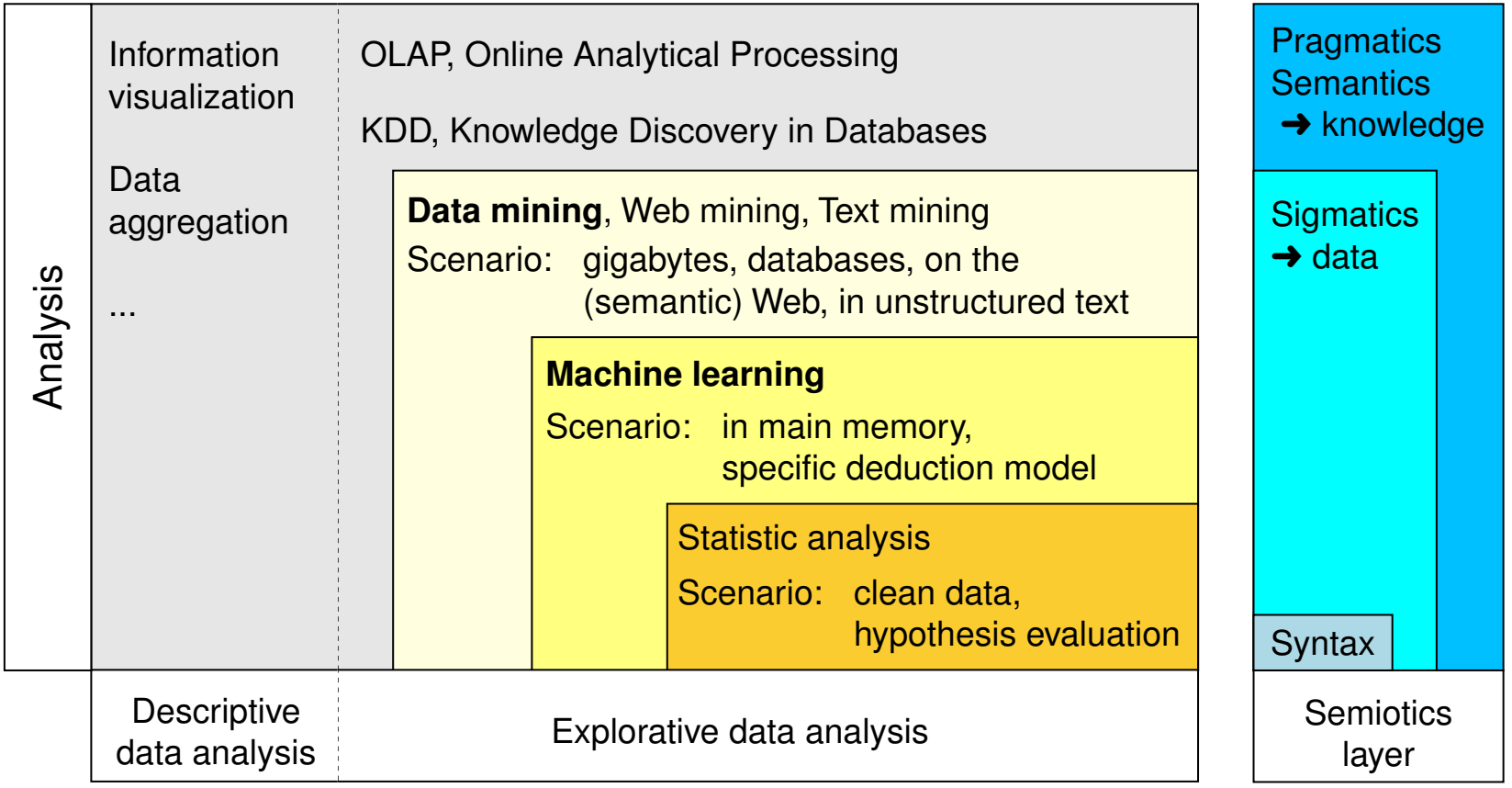
Einordnung Data Mining



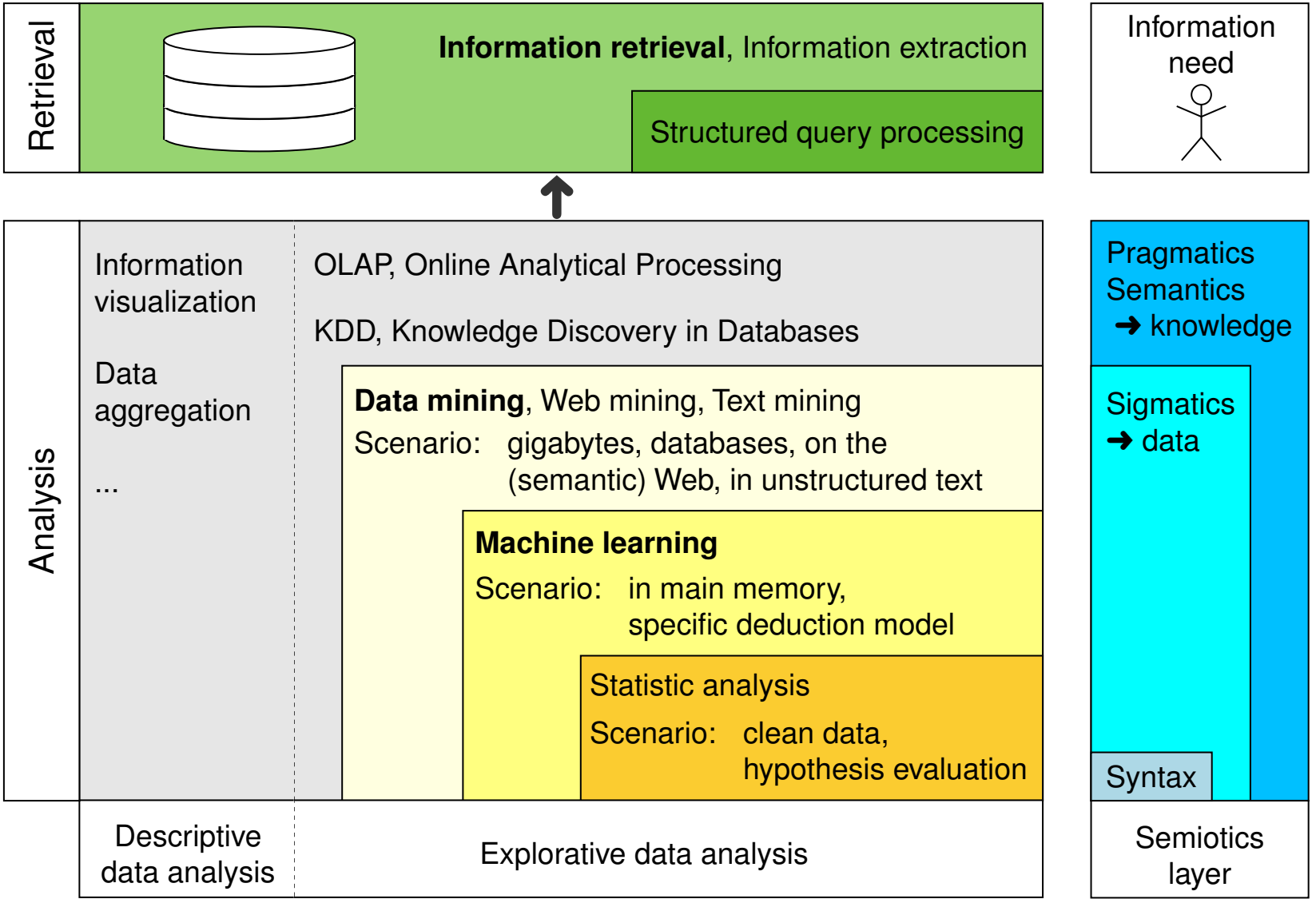
Einordnung Data Mining



Einordnung Data Mining



Einordnung Data Mining



Bemerkungen:

- Eine klare Abgrenzung von Machine Learning und Data Mining ist nicht immer möglich; ein wichtiger Unterschied resultiert aus der Größe der behandelten Datenmengen: Anwendungen des Machine Learning laufen typischerweise im Hauptspeicher ab; die Disziplin des Data Mining entstand aus der Notwendigkeit, maschinelle Analyseverfahren auf riesige Datenbanken anzuwenden.
- Ein Schwerpunkt des Machine Learning ist der eigentliche Lern- bzw. Deduktionsprozess wie die Theorie des analogen Schließens, das Lernen aus Beispielen, oder der Einfluss der Umwelt auf das Lernen. Hingegen ist die treibende Kraft hinter Data Mining die Industrie- und Geschäftswelt mit ihren großen Datenbanken.
- Zu den bekannten Aufgabenstellung des Data Mining gehören: ungerichtete Abhängigkeitsanalyse zur Identifikation signifikanter Dependenzen zwischen den Attributen eines Informationsobjektes (Beispiel: Warenkorbanalyse), Gruppenbildung und Klassifikationsprobleme, Filtern von Prozessdaten, Prognoseaufgaben.

Einordnung Data Mining

Methoden und Techniken

- Cluster-Analyse
- propositionale (oder relative) Regellernverfahren
- assoziative Regellernverfahren
- Hauptkomponenten- und Faktoranalyse

Kapitel ML: X

X. Cluster-Analyse

- Einordnung Data Mining
- Einführung in die Cluster-Analyse
- Hierarchische Verfahren
- Iterative Verfahren
- Dichtebasierte Verfahren
- Cluster-Evaluierung

Einführung in die Cluster-Analyse

Cluster-Analyse ist die unüberwachte Klassifikation einer Menge von Objekten in Gruppen; dabei wird folgendes Ziel verfolgt:

1. Ähnlichkeit innerhalb der Gruppen maximieren
2. Ähnlichkeit zwischen den Gruppen minimieren

Einführung in die Cluster-Analyse

Cluster-Analyse ist die unüberwachte Klassifikation einer Menge von Objekten in Gruppen; dabei wird folgendes Ziel verfolgt:

1. Ähnlichkeit innerhalb der Gruppen maximieren
2. Ähnlichkeit zwischen den Gruppen minimieren

Anwendungen

- Identifikation gleichartiger Käufergruppen
- „höhere“ Bildverarbeitung, im Sinne von Objekterkennung
- Suche nach ähnlichen Genprofilen
- Spezifikation von Syndromen
- Analyse von Verkehrsdaten in Computernetzen
- Visualisierung komplexer Graphen
- Textkategorisierung im Information-Retrieval

Bemerkungen:

- Die Problemstellung der Cluster-Analyse ist umgekehrt zur Problemstellung der Varianzanalyse. Ziel der Varianzanalyse ist die Überprüfung, ob eine gegebene nominalskalierte Variable Gruppen definiert, deren Mitglieder sich in abhängigen (intervallskalierten) Variablen unterscheiden. Ziel der Cluster-Analyse ist die Erzeugung einer solchen nominalskalierten Variable durch die Entdeckung der Ausprägungen dieser Variable. Jedes Cluster korrespondiert zu einer Variablenausprägung.
- Die Cluster-Analyse ist ein Verfahren zur Struktur*erzeugung*: Man weiß nichts über die zu erzeugende Variable, insbesondere nichts über die Anzahl ihrer Werteausprägungen.
  Die Varianzanalyse ist ein Verfahren zur Struktur*prüfung*.

Einführung in die Cluster-Analyse

$\mathbf{x}_1, \ldots \mathbf{x}_n$ sind die zu n Objekten gehörenden p-dimensionalen Merkmalvektoren:

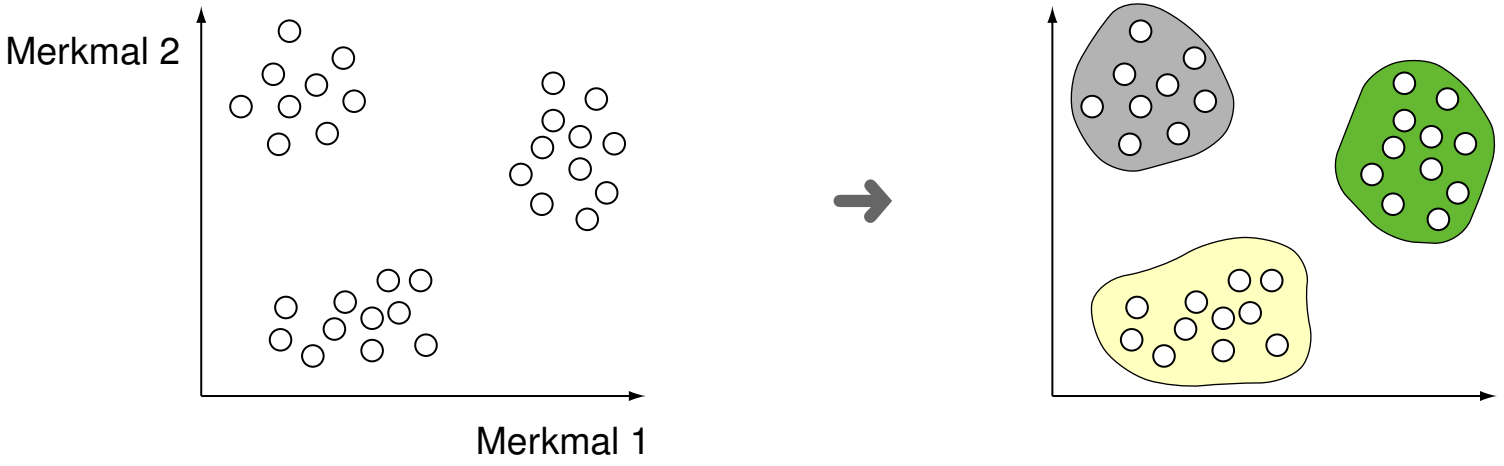
	Merkmal 1	Merkmal 2	...	Merkmal p
$\mathbf{x}_1$	x_{1_1}	x_{1_2}	...	x_{1_p}
$\mathbf{x}_2$	x_{2_1}	x_{2_2}	...	x_{2_p}
$\vdots$				
$\mathbf{x}_n$	x_{n_1}	x_{n_2}	...	x_{n_p}

Einführung in die Cluster-Analyse

$\mathbf{x}_1, \ldots \mathbf{x}_n$ sind die zu n Objekten gehörenden p-dimensionalen Merkmalvektoren:

	Merkmal 1	Merkmal 2	...	Merkmal p
$\mathbf{x}_1$	x_{1_1}	x_{1_2}	...	x_{1_p}
$\mathbf{x}_2$	x_{2_1}	x_{2_2}	...	x_{2_p}
$\vdots$				
$\mathbf{x}_n$	x_{n_1}	x_{n_2}	...	x_{n_p}

30 zweidimensionale Merkmalvektoren ($n = 30$, $p = 2$) :



Einführung in die Cluster-Analyse

Definition 18 (Clustering)

Sei X eine Menge von Merkmalvektoren. Ein (exklusives) Clustering $\mathcal{C}$ von X, $\mathcal{C} = \{C_1, C_2, \ldots, C_k\}$, $C_i \subseteq X$, ist eine Aufteilung von X in paarweise disjunkte Mengen C_i mit $\bigcup_{C_i \in \mathcal{C}} C_i = X$.

Einführung in die Cluster-Analyse

Definition 18 (Clustering)
Sei X eine Menge von Merkmalvektoren. Ein (exklusives) Clustering $\mathcal{C}$ von X, $\mathcal{C} = \{C_1, C_2, \ldots, C_k\}$, $C_i \subseteq X$, ist eine Aufteilung von X in paarweise disjunkte Mengen C_i mit $\bigcup_{C_i \in \mathcal{C}} C_i = X$.

Algorithmen zur Cluster-Analyse sind unüberwachte Lernverfahren:

- der Lernprozess ist selbstorganisiert
- es gibt keinen externen Lehrer
- es gibt ein aufgaben*unabhängiges* Optimierungskriterium

Einführung in die Cluster-Analyse

Definition 18 (Clustering)
Sei X eine Menge von Merkmalvektoren. Ein (exklusives) Clustering $\mathcal{C}$ von X, $\mathcal{C} = \{C_1, C_2, \ldots, C_k\}$, $C_i \subseteq X$, ist eine Aufteilung von X in paarweise disjunkte Mengen C_i mit $\bigcup_{C_i \in \mathcal{C}} C_i = X$.

Algorithmen zur Cluster-Analyse sind unüberwachte Lernverfahren:

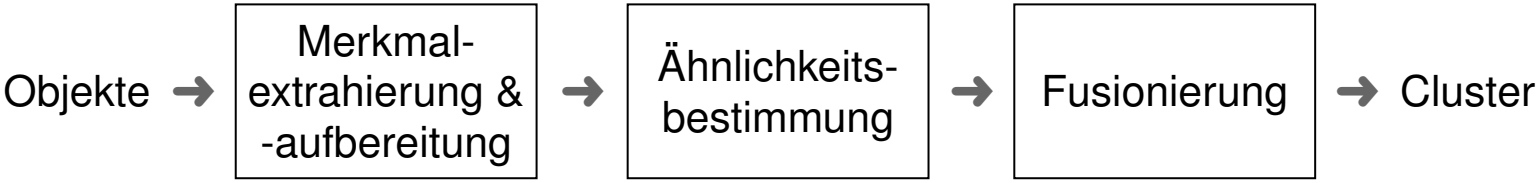
- der Lernprozess ist selbstorganisiert
- es gibt keinen externen Lehrer
- es gibt ein aufgaben*unabhängiges* Optimierungskriterium

Überwachtes Lernen:

- es gibt Lernziele: das Zielkonzept, gewünschte Reaktionen, etc.
- es gibt ein aufgaben*abhängiges* Optimierungskriterium
- es gibt Information darüber, *wie* eine Verbesserung im Optimierungskriterium zu erzielen ist. Stichwort: „Instructive Feedback“

Einführung in die Cluster-Analyse

Grundschritte einer Cluster-Analyse



Einführung in die Cluster-Analyse

Merkmalextrahierung und -aufbereitung

Gesucht sind evtl. neue Merkmale mit hoher Varianz. Techniken:

- Analyse von Streuungsparametern
- Dimensionsreduktion: Faktoranalyse, multidimensionale Skalierung
- Visuelle Analyse: Scatter-Plots, Box-plots

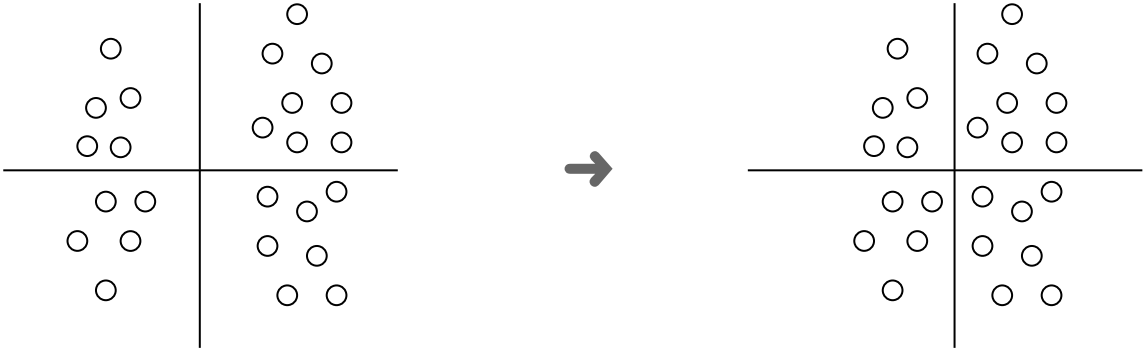
Einführung in die Cluster-Analyse

Merkmalextrahierung und -aufbereitung

Gesucht sind evtl. neue Merkmale mit hoher Varianz. Techniken:

- Analyse von Streuungsparametern
- Dimensionsreduktion: Faktoranalyse, multidimensionale Skalierung
- Visuelle Analyse: Scatter-Plots, Box-plots

Standardisierung von Variablen (Merkmalen) ist problematisch:



Einführung in die Cluster-Analyse

Berechnung von Distanzen oder Ähnlichkeiten

	Merkmal 1	Merkmal 2	...	Merkmal p
$\mathbf{x}_1$	x_{1_1}	x_{1_2}	...	x_{1_p}
$\mathbf{x}_2$	x_{2_1}	x_{2_2}	...	x_{2_p}
$\vdots$				
$\mathbf{x}_n$	x_{n_1}	x_{n_2}	...	x_{n_p}

	$\mathbf{x}_1$	$\mathbf{x}_2$	...	$\mathbf{x}_n$
$\mathbf{x}_1$	0	$d(\mathbf{x}_1, \mathbf{x}_2)$	...	$d(\mathbf{x}_1, \mathbf{x}_n)$
$\mathbf{x}_2$	-	0	...	$d(\mathbf{x}_2, \mathbf{x}_n)$
$\vdots$				
$\mathbf{x}_n$	-	-	...	0

Bemerkungen:

- Die Distanzmatrix ist oft implizit durch eine Metrik auf dem Merkmalsraum definiert.
- Die Distanzmatrix kann als Adjazenzmatrix eines gewichteten, ungerichteten Graphen G, $G = \langle V, E, w \rangle$, interpretiert werden: Die Menge X der Merkmalvektoren wird bijektiv auf eine Knotenmenge V abgebildet; eine Distanz $d(\mathbf{x}_i, \mathbf{x}_j)$ entspricht dem Gewicht $w(\{u, v\})$ der Kante $\{u, v\} \in E$ zwischen den mit $\mathbf{x}_i$ und $\mathbf{x}_j$ assoziierten Knoten u und v.

Einführung in die Cluster-Analyse

Berechnung von Distanzen oder Ähnlichkeiten (Fortsetzung)

Anforderungen an eine Distanzfunktion:

1. $d(\mathbf{x}_1, \mathbf{x}_2) \geq 0$
2. $d(\mathbf{x}_1, \mathbf{x}_1) = 0$
3. $d(\mathbf{x}_1, \mathbf{x}_2) = d(\mathbf{x}_2, \mathbf{x}_1)$
4. $d(\mathbf{x}_1, \mathbf{x}_3) \leq d(\mathbf{x}_1, \mathbf{x}_2) + d(\mathbf{x}_2, \mathbf{x}_3)$

Bei intervallskalierten Variablen Verwendung der Minkowsky-Metrik:

$$d(\mathbf{x}_1, \mathbf{x}_2) = \left(\sum_{i=1}^{p} |x_{1_i} - x_{2_i}|^r \right)^{1/r},$$

- $r = 1$. Manhatten- oder Hamming-Distanz, L_1-Norm
- $r = 2$. Euklidische Distanz, L_2-Norm
- $r = \infty$. Maximum-Distanz, L_∞-Norm bzw. L_{max}-Norm

Einführung in die Cluster-Analyse

Berechnung von Distanzen oder Ähnlichkeiten (Fortsetzung)

Eine Cluster-Analyse verlangt kein spezielles Skalenniveau der Merkmale.

➜ Verallgemeinerung der Distanzfunktion zur (Un)Ähnlichkeitsfunktion durch Verzicht auf die Dreiecksungleichung. (Un)Ähnlichkeiten lassen sich zwischen binären, nominalen und ordinalen Variablen quantifizieren.

Einführung in die Cluster-Analyse

Berechnung von Distanzen oder Ähnlichkeiten (Fortsetzung)

Eine Cluster-Analyse verlangt kein spezielles Skalenniveau der Merkmale.

➜ Verallgemeinerung der Distanzfunktion zur (Un)Ähnlichkeitsfunktion durch Verzicht auf die Dreiecksungleichung. (Un)Ähnlichkeiten lassen sich zwischen binären, nominalen und ordinalen Variablen quantifizieren.

Ähnlichkeitskoeffizienten für zwei Vektoren, $\mathbf{x}_1$, $\mathbf{x}_2$, mit binären Merkmalen:

$$\text{Simple Matching Coefficient (SMC)} = \frac{f_{11} + f_{00}}{f_{01} + f_{10} + f_{11} + f_{00}}$$

$$\text{Jaccard-Koefficient (J)} = \frac{f_{11}}{f_{01} + f_{10} + f_{11}}$$

mit

f_{00} = Anzahl der Merkmale mit Ausprägung 0 sowohl in $\mathbf{x}_1$ als auch in $\mathbf{x}_2$

f_{01} = Anzahl der Merkmale mit Ausprägung 0 in $\mathbf{x}_1$ und Ausprägung 1 in $\mathbf{x}_2$

f_{10} = Anzahl der Merkmale mit Ausprägung 1 in $\mathbf{x}_1$ und Ausprägung 0 in $\mathbf{x}_2$

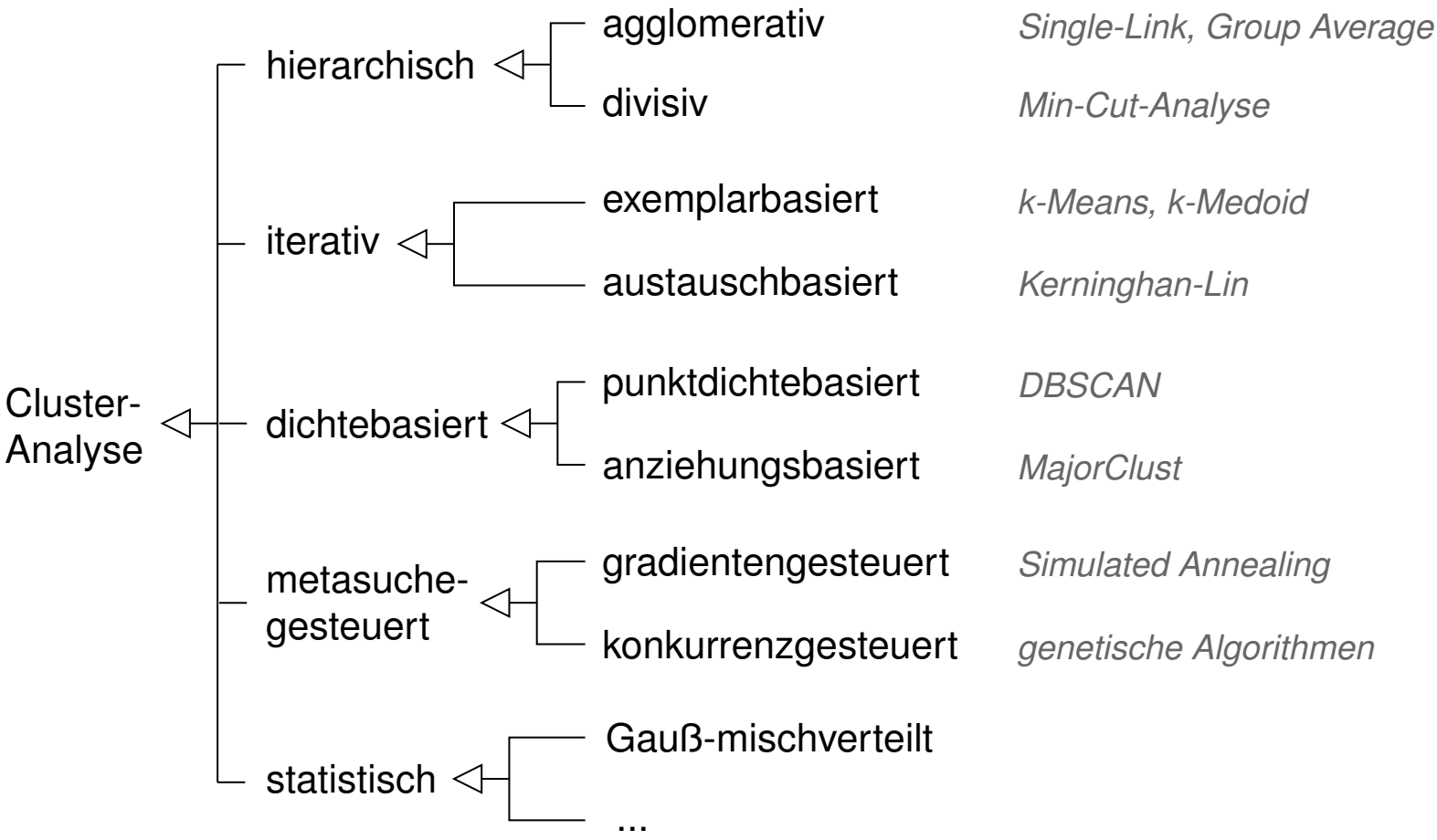
f_{11} = Anzahl der Merkmale mit Ausprägung 1 sowohl in $\mathbf{x}_1$ als auch in $\mathbf{x}_2$

Bemerkungen:

- Die Definition der Ähnlichkeitskoeffizienten lässt sich auf nominale Variablen erweitern.
- Heterogene Metriken (HEOM, HVDM) ermöglichen die kombinierte Verrechnung verschiedener Skalenniveaus.
- Die Berechnung von Korrelationskoeffizienten zwischen zwei Vektoren über alle Merkmale (also nicht zwischen zwei Merkmalen über alle Vektoren) ermöglicht den Vergleich von Profilen. Beispiel: Q-Korrelationskoeffizient
- Die Konstruktion eines Ähnlichkeitsmaßes kann die größte Herausforderung bei der Cluster-Analyse darstellen. Typische Problemfelder sind:
  - Normalisierung
  - Empfindlichkeit bei Ausreißern
  - Korrelationen zwischen Merkmalen
  - unterschiedliche Wichtigkeit der Merkmale
- Ähnlichkeitsmaße lassen sich kanonisch in Unähnlichkeitsmaße umrechnen – und umgekehrt.

Einführung in die Cluster-Analyse

Prinzipien der Fusionierung



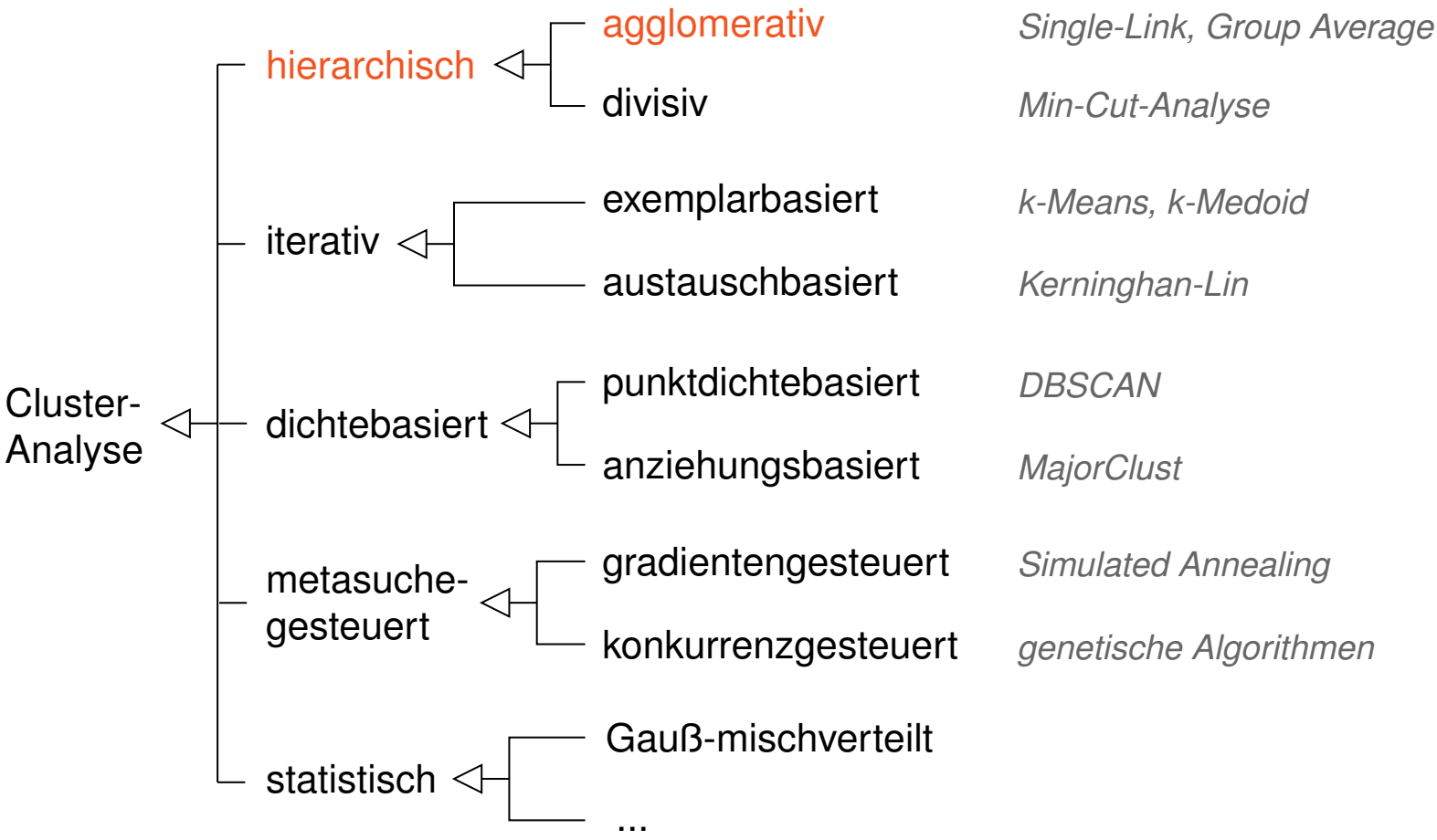
Kapitel ML: X (Fortsetzung)

X. Cluster-Analyse

- Einordnung Data Mining
- Einführung in die Cluster-Analyse
- Hierarchische Verfahren
- Iterative Verfahren
- Dichtebasierte Verfahren
- Cluster-Evaluierung

Hierarchische Verfahren

Prinzipien der Fusionierung



Hierarchische Verfahren

Algorithmus zur hierarchisch-agglomerativen Cluster-Analyse

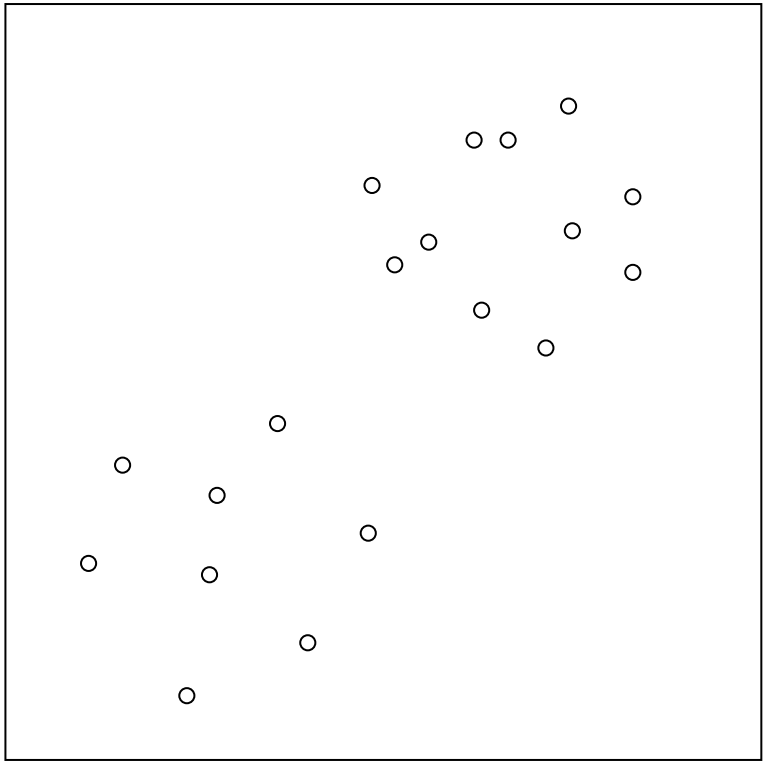
Input: $G = \langle V, E, w \rangle$. Weighted graph.
$d_{\mathcal{C}}$. Distance measure between two clusters.

Output: $T = \langle V_T, E_T \rangle$. Cluster hierarchy or dendrogram.

1. $\mathcal{C} = \{\{v\} \mid v \in V\}$ // define initial clustering
2. $V_T = \{v_C \mid C \in \mathcal{C}\}, \quad E_T = \emptyset$ // define initial dendrogram
3. **WHILE** $|\mathcal{C}| > 1$ **DO**
4. *update_distance_matrix*$(\mathcal{C}, G, d_{\mathcal{C}})$
5. $\{C, C'\} = \underset{\{C_i, C_j\} \in \mathcal{C}: C_i \neq C_j}{\text{argmin}} d_{\mathcal{C}}(C_i, C_j)$
6. $\mathcal{C} = (\mathcal{C} \setminus \{C, C'\}) \cup \{C \cup C'\}$ // merging
7. $V_T = V_T \cup \{v_{C,C'}\}, \quad E_T = E_T \cup \{\{v_{C,C'}, v_C\}, \{v_{C,C'}, v_{C'}\}\}$ // dendrogram
8. **ENDDO**
9. **RETURN**(T)

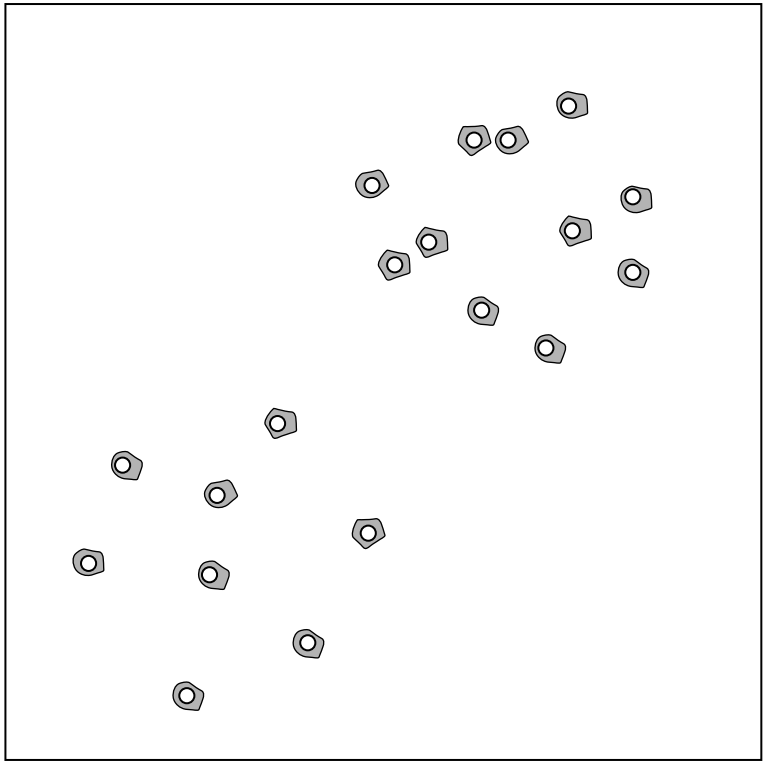
Hierarchische Verfahren

Single-Link: Cluster-Distanzmaß $d_{\mathcal{C}}$ = Nearest-Neighbor



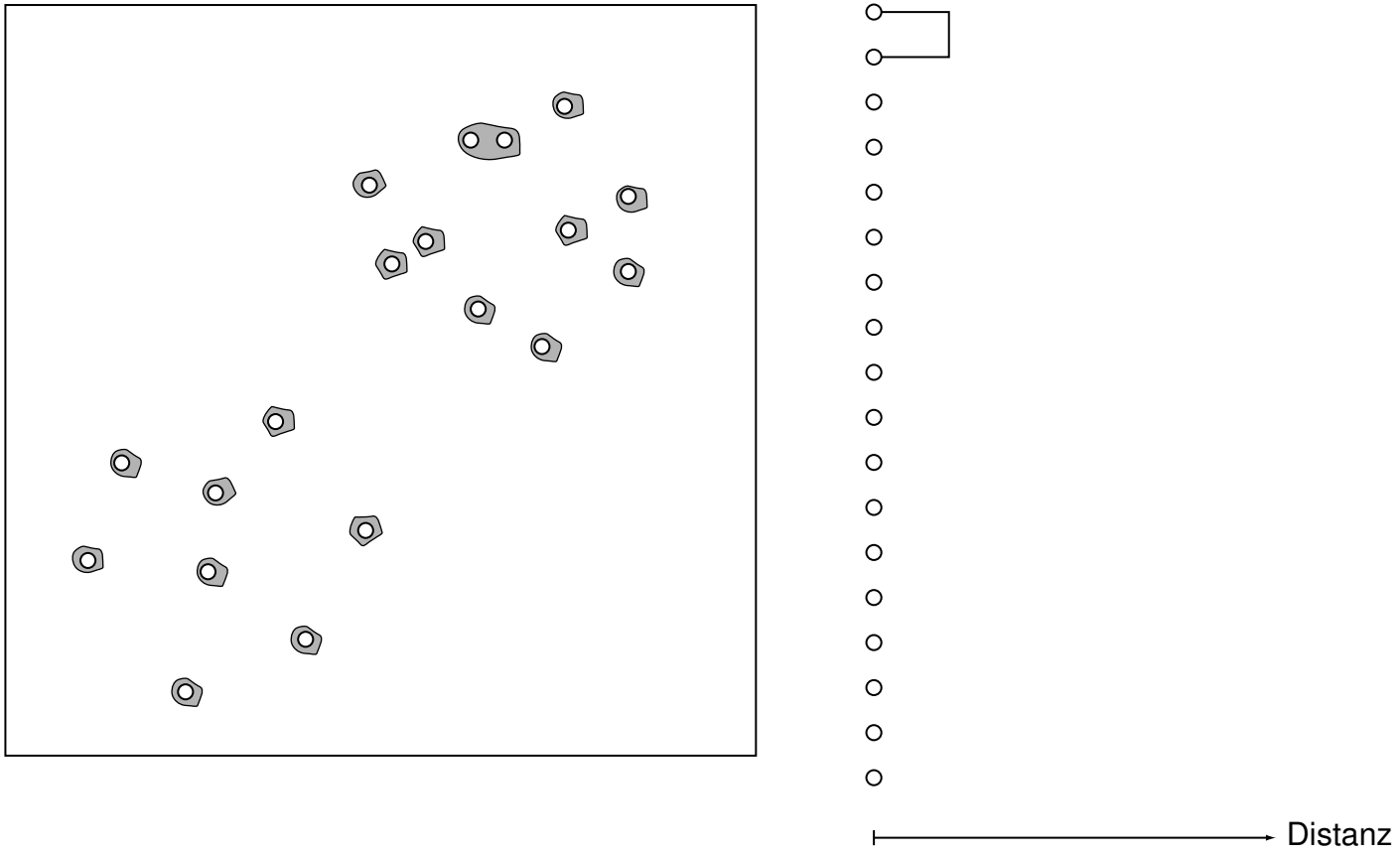
Hierarchische Verfahren

Single-Link: Cluster-Distanzmaß $d_{\mathcal{C}}$ = Nearest-Neighbor



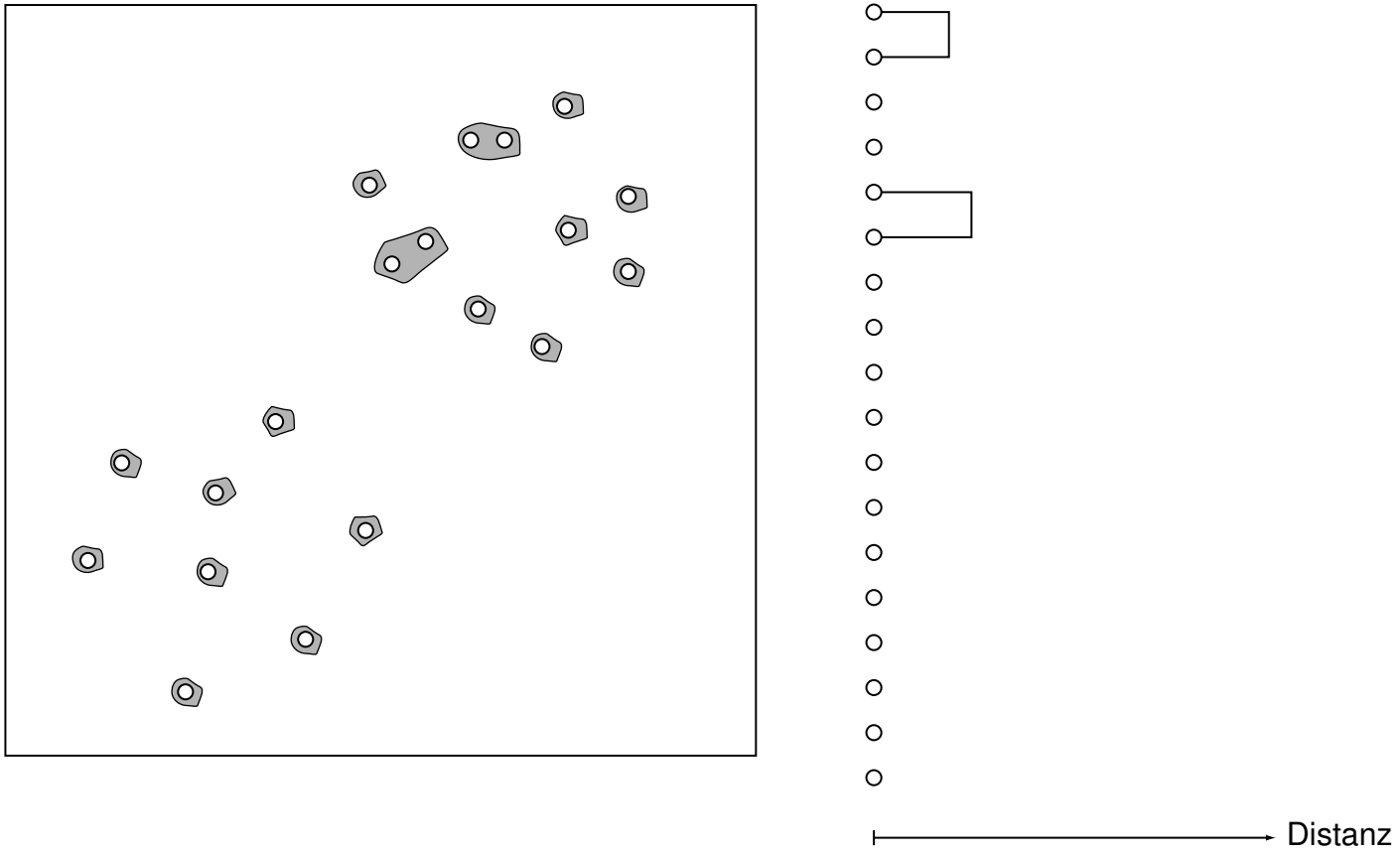
Hierarchische Verfahren

Single-Link: Cluster-Distanzmaß $d_{\mathcal{C}}$ = Nearest-Neighbor



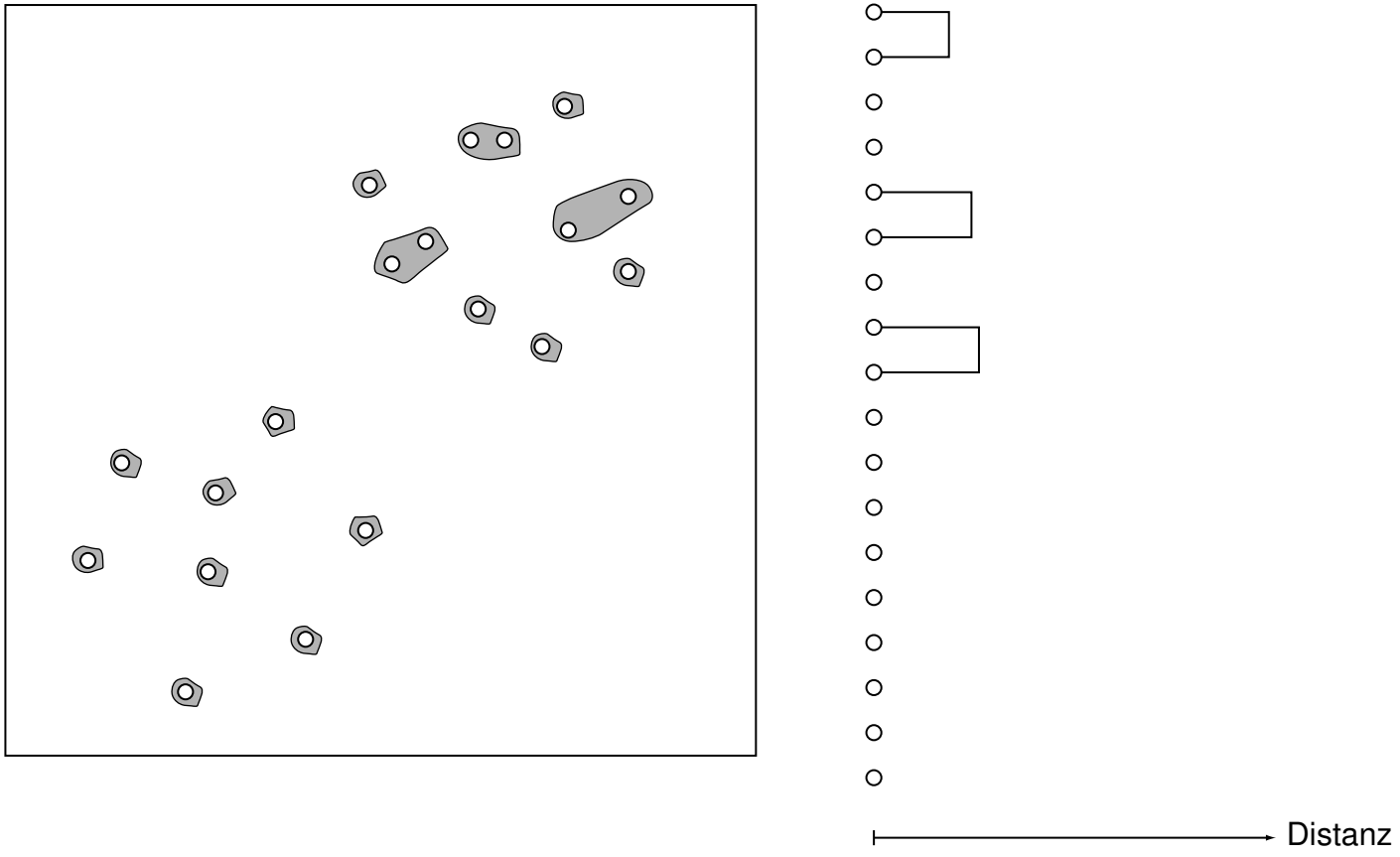
Hierarchische Verfahren

Single-Link: Cluster-Distanzmaß $d_{\mathcal{C}}$ = Nearest-Neighbor



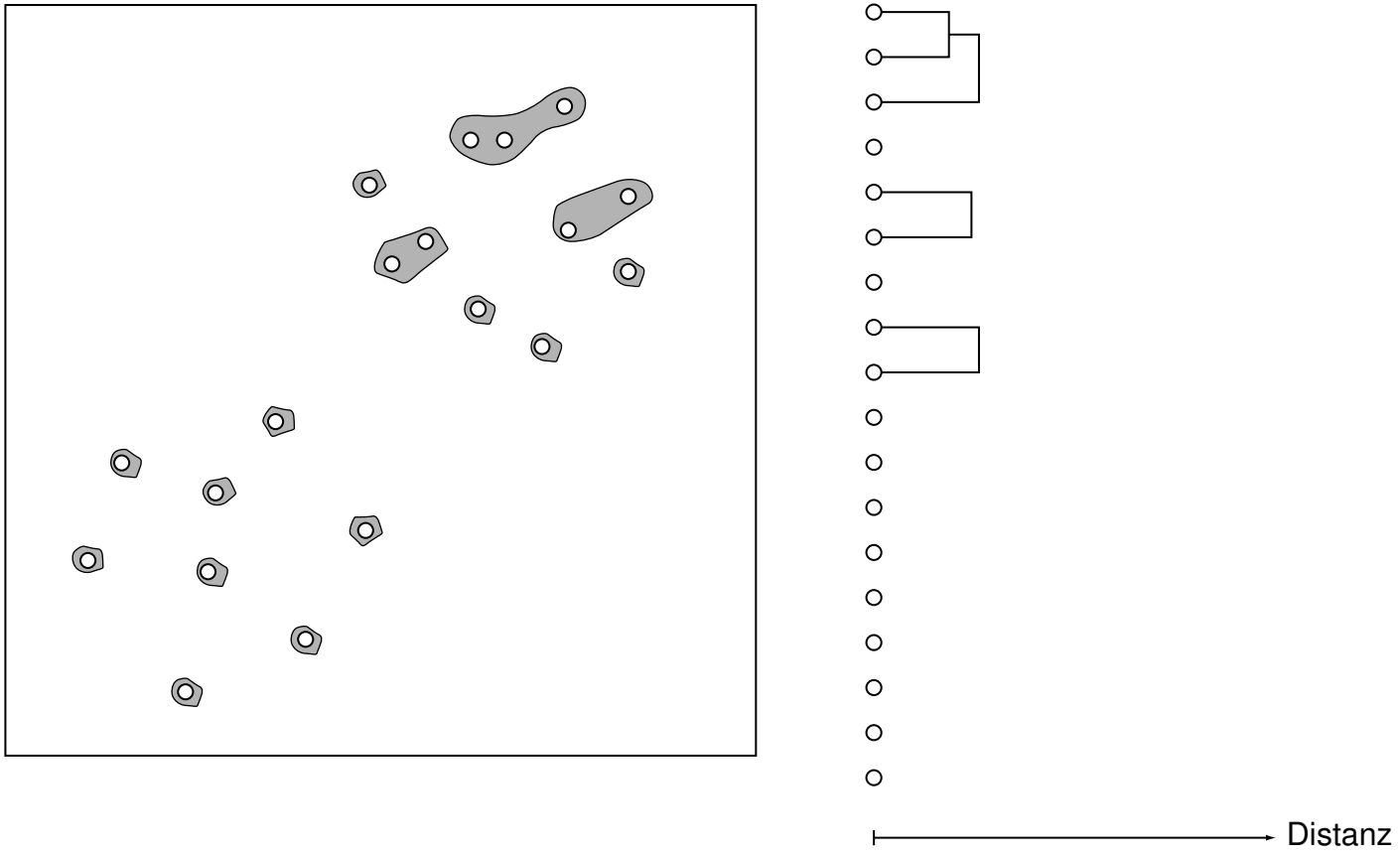
Hierarchische Verfahren

Single-Link: Cluster-Distanzmaß $d_{\mathcal{C}}$ = Nearest-Neighbor



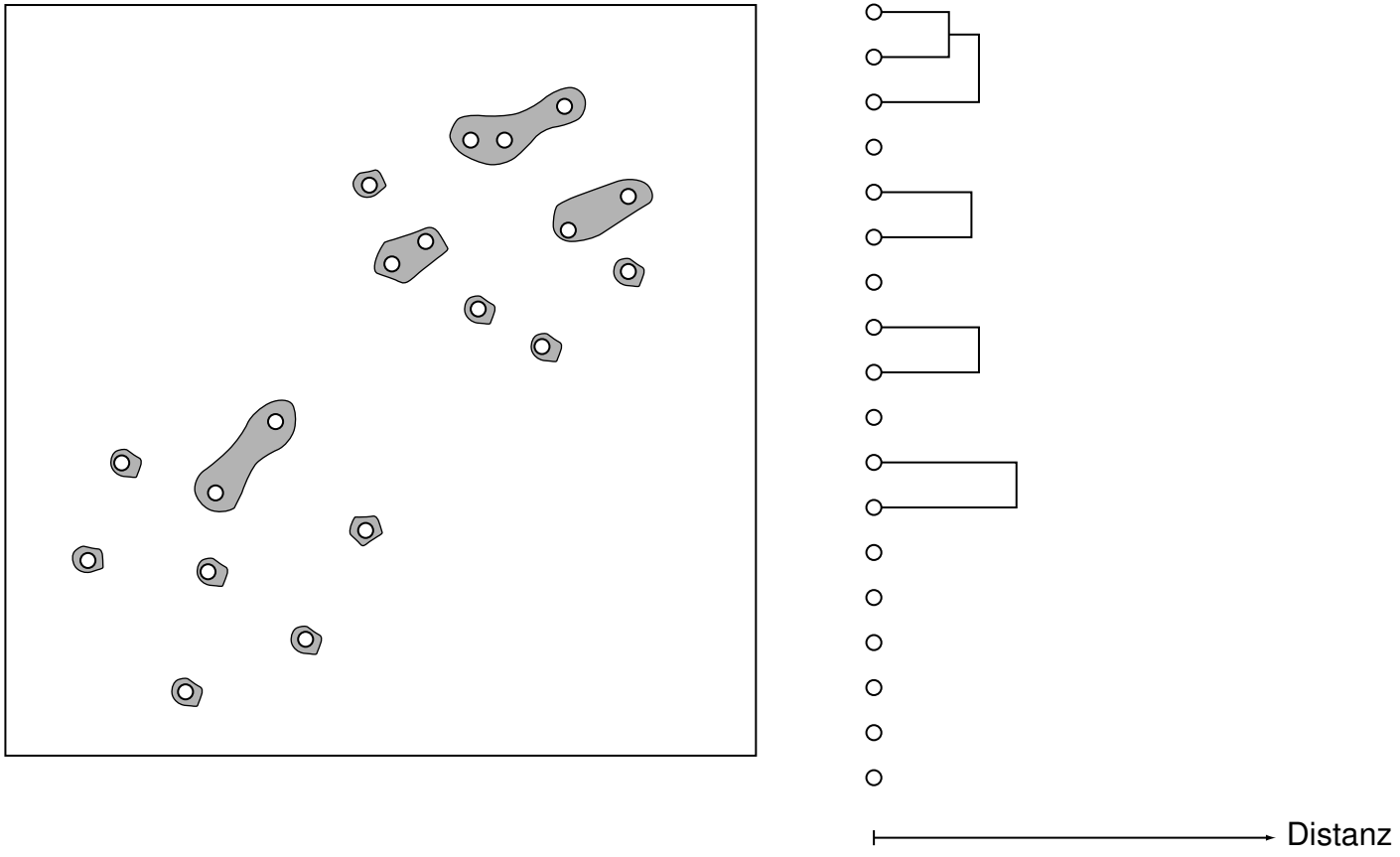
Hierarchische Verfahren

Single-Link: Cluster-Distanzmaß $d_{\mathcal{C}}$ = Nearest-Neighbor



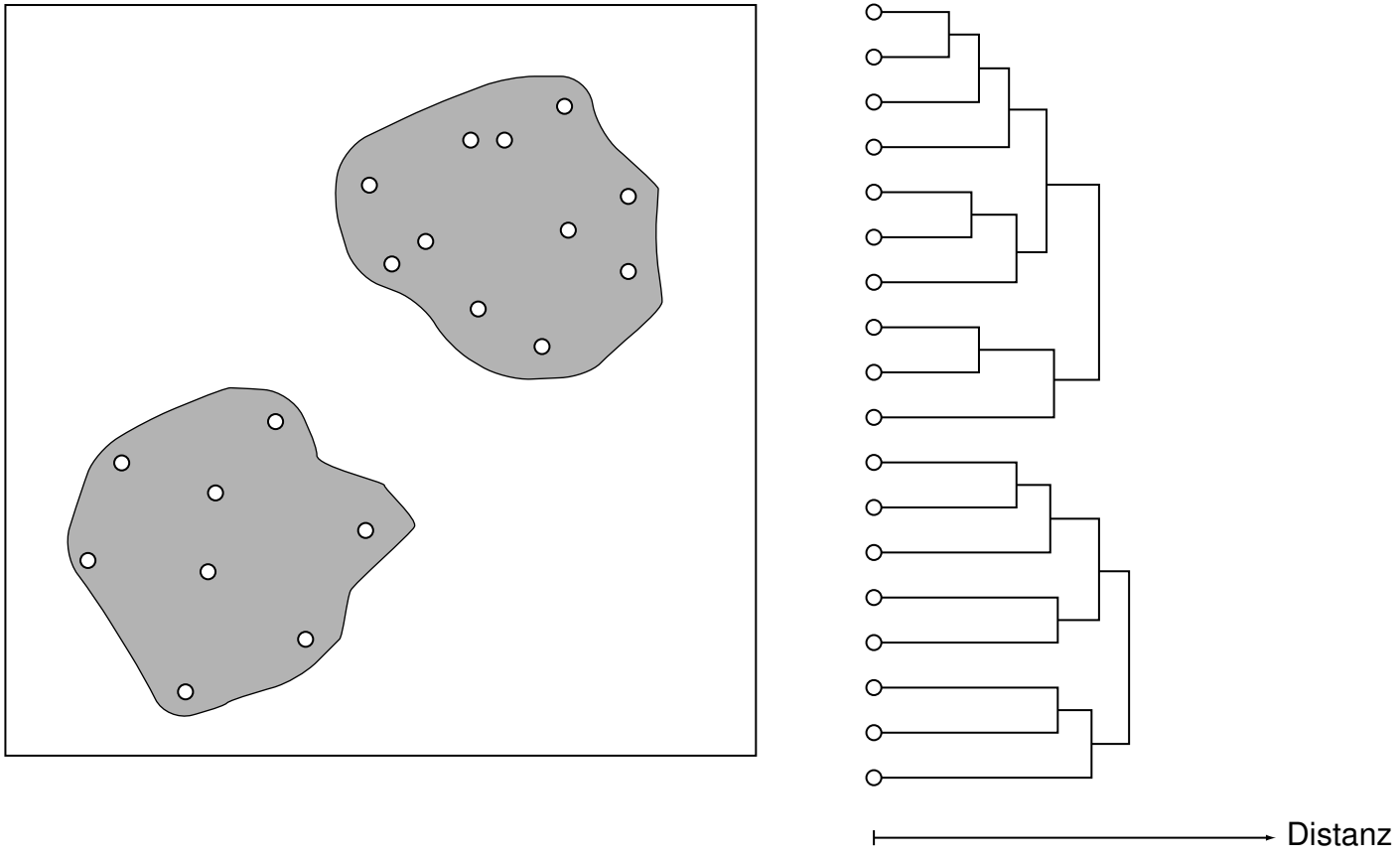
Hierarchische Verfahren

Single-Link: Cluster-Distanzmaß $d_{\mathcal{C}}$ = Nearest-Neighbor



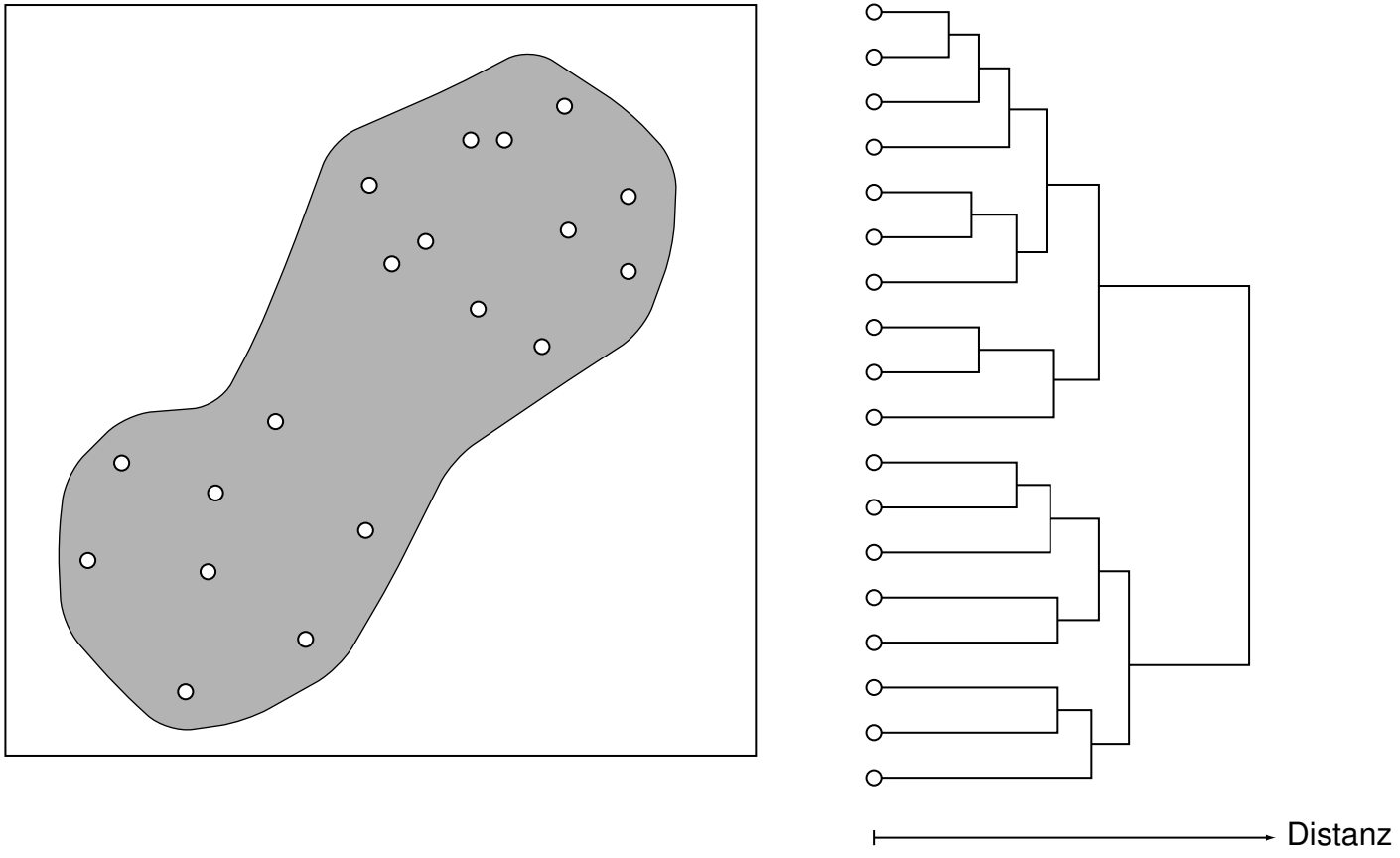
Hierarchische Verfahren

Single-Link: Cluster-Distanzmaß $d_{\mathcal{C}}$ = Nearest-Neighbor



Hierarchische Verfahren

Single-Link: Cluster-Distanzmaß $d_{\mathcal{C}}$ = Nearest-Neighbor



Hierarchische Verfahren

Einteilung hierarchisch-agglomerativer Verfahren hinsichtlich $d_{\mathcal{C}}$

$d_{\mathcal{C}}(C, C') = \min_{\substack{u \in C \\ v \in C'}} d(u, v)$	Single-Link (Nearest-Neighbor)
$d_{\mathcal{C}}(C, C') = \max_{\substack{u \in C \\ v \in C'}} d(u, v)$	Complete-Link (Furthest-Neighbor)
$d_{\mathcal{C}}(C, C') = \frac{1}{\lvert C \rvert \cdot \lvert C' \rvert} \sum_{\substack{u \in C \\ v \in C'}} d(u, v)$	(Group-)Average-Link
$d_{\mathcal{C}}(C, C') = \sqrt{\frac{2 \cdot \lvert C \rvert \cdot \lvert C' \rvert}{\lvert C \rvert + \lvert C' \rvert}} \cdot \lvert\lvert \bar{u} - \bar{v} \rvert\rvert$	Ward (Varianz)

Hierarchische Verfahren

Ward-Kriterium

Ward ist ein Varianzkriterium; es entspricht der doppelten Zunahme der Wurzel aus der Fehlerquadratsumme, SSE, in dem neuen Cluster, der durch die Vereinigung der beiden Cluster C und C' entsteht. Herleitung:

$$SSE(C) = \sum_{u \in C} ||\bar{u} - u||^2 = \sum_{u \in C} (||\bar{u}||^2 - 2 \cdot \langle u, \bar{u} \rangle + ||u||^2)$$
$$= |C| \cdot ||\bar{u}||^2 - 2|C| \cdot ||\bar{u}||^2 + \sum_{u \in C} ||u||^2 = \sum_{u \in C} ||u||^2 \ - |C| \cdot ||\bar{u}||^2$$

$$SSE(C') = \sum_{v \in C'} ||v||^2 \ - |C'| \cdot ||\bar{v}||^2$$

$$SSE(C \cup C') = \sum_{w \in (C \cup C')} ||w||^2 \ - |C \cup C'| \cdot ||\bar{w}||^2, \quad \text{mit } \ w = \frac{|C| \cdot \bar{u} + |C'| \cdot \bar{v}}{|C| + |C'|}$$

$$SSE(C \cup C') - SSE(C) - SSE(C') \ = \ldots = \ \frac{|C| \cdot |C'|}{|C| + |C'|} \cdot ||\bar{u} - \bar{v}||^2$$

$\bar{u}$ bzw. $\bar{v}$ bezeichnen den Mittelwert der Punkte $u \in C$ bzw. $v \in C$'.

Hierarchische Verfahren

Update-Formel für Cluster-Distanzen

Eine effiziente Berechnung neuer Cluster-Distanzen $d_{\mathcal{C}}$ ermöglicht folgende Update-Formel (Lance-Williams-Formel):

$$\begin{aligned} d_{\mathcal{C}}(C \cup C', C_i) = \; & \alpha \cdot d_{\mathcal{C}}(C, C_i) + \\ & \beta \cdot d_{\mathcal{C}}(C', C_i) + \\ & \gamma \cdot d_{\mathcal{C}}(C, C') + \\ & \delta \cdot |d_{\mathcal{C}}(C, C_i) - d_{\mathcal{C}}(C', C_i)| \end{aligned}$$

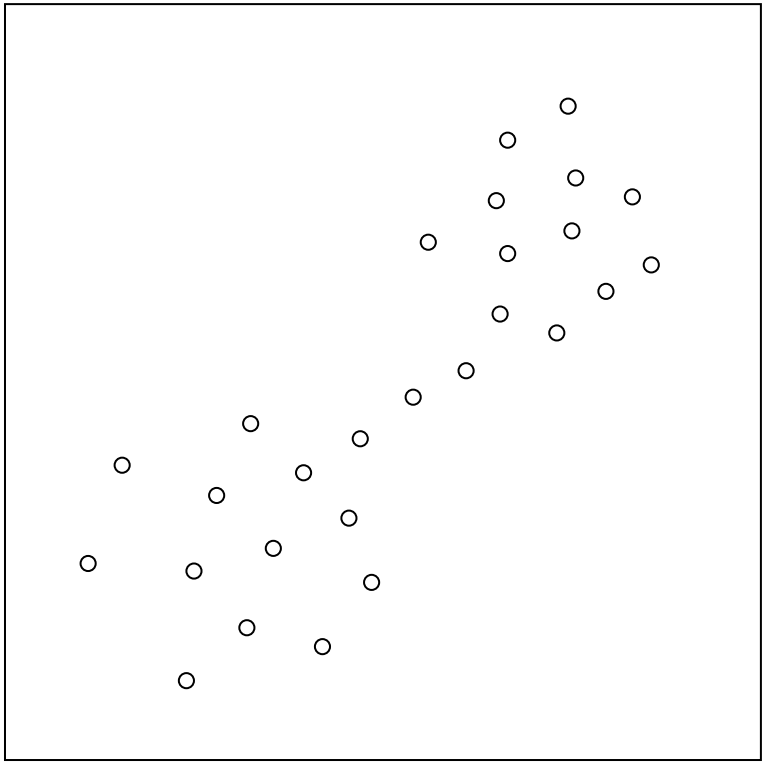
Idee: Anstatt immer wieder alle Elemente von zwei Clustern zu betrachten, nimmt die Update-Formel Bezug auf die vorherigen Distanzen.

Bemerkungen:

- Link-basierte Verfahren arbeiten sowohl mit beliebigen Distanz- als auch Ähnlichkeitsmaßen.
- Single-Link kann unmittelbar mit einem Minimum-Spanning-Tree-Algorithmus realisiert werden.
- Varianz-basierte Verfahren sind nur sinnvoll, falls alle Merkmale Intervallskalennivau besitzen.

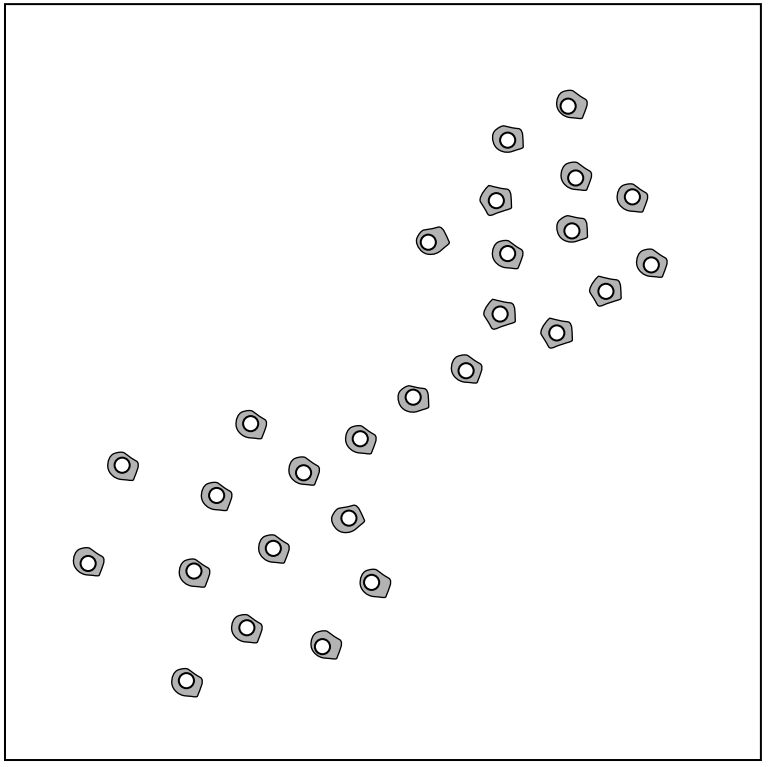
Hierarchische Verfahren

Chaining-Problematik bei Single-Link ($d_{\mathcal{C}}$ = Nearest-Neighbor)



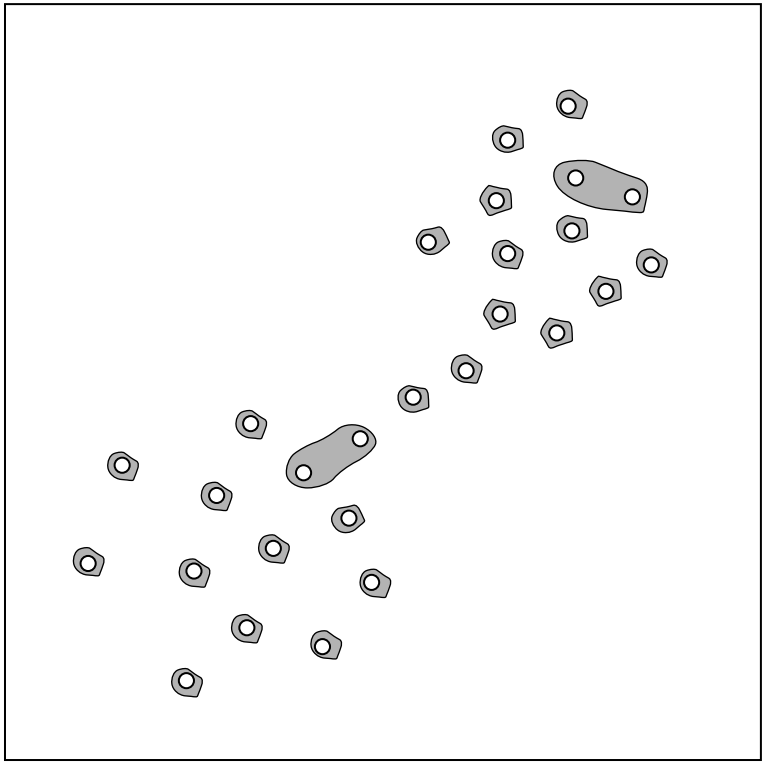
Hierarchische Verfahren

Chaining-Problematik bei Single-Link ($d_{\mathcal{C}}$ = Nearest-Neighbor)



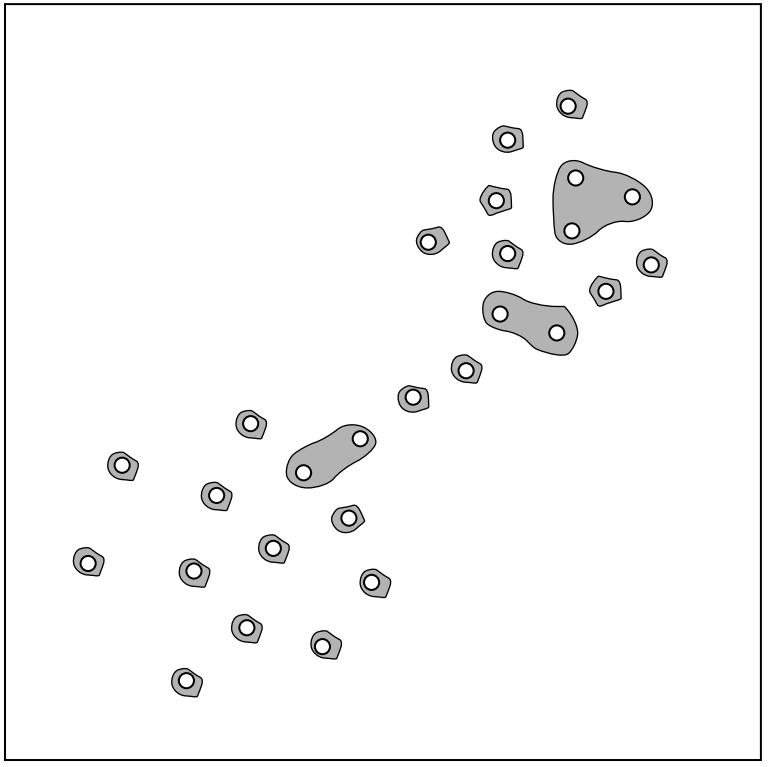
Hierarchische Verfahren

Chaining-Problematik bei Single-Link ($d_{\mathcal{C}}$ = Nearest-Neighbor)



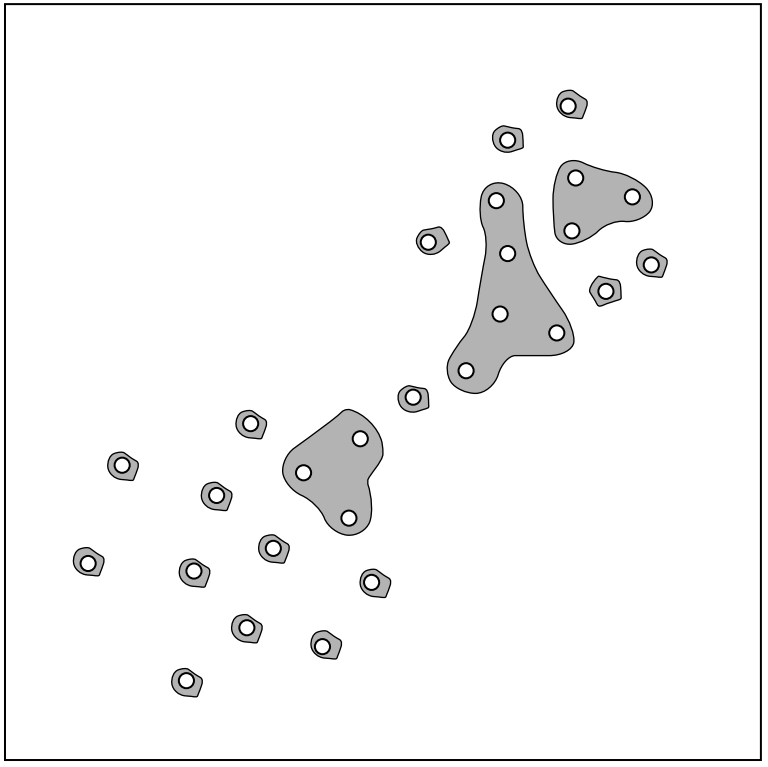
Hierarchische Verfahren

Chaining-Problematik bei Single-Link ($d_{\mathcal{C}}$ = Nearest-Neighbor)



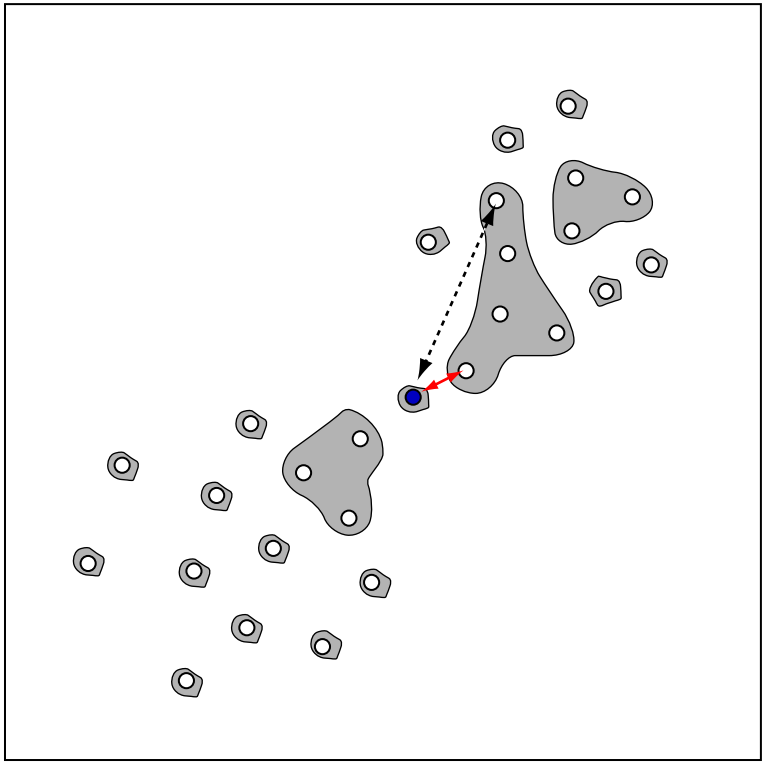
Hierarchische Verfahren

Chaining-Problematik bei Single-Link ($d_{\mathcal{C}}$ = Nearest-Neighbor)



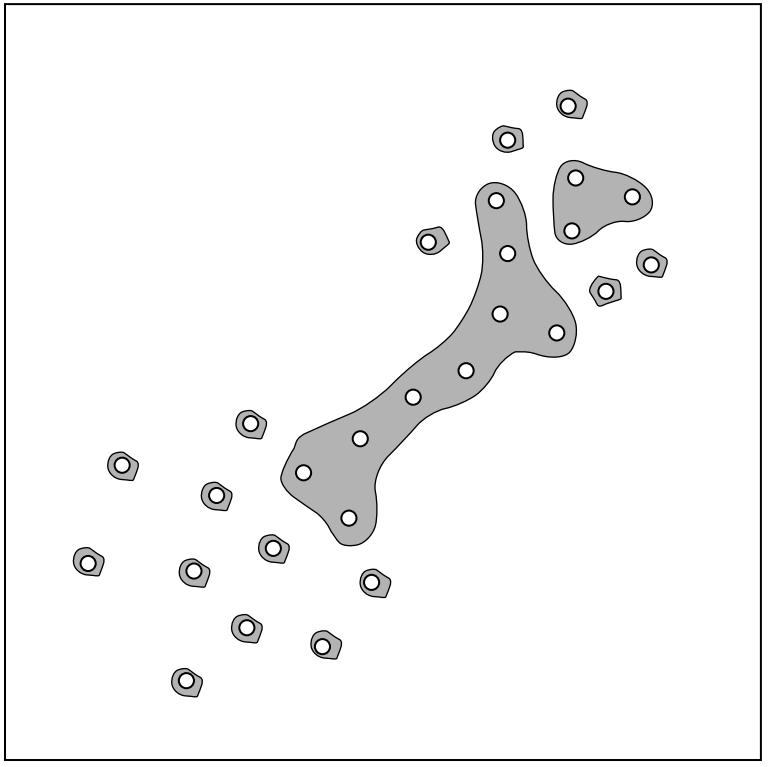
Hierarchische Verfahren

Chaining-Problematik bei Single-Link ($d_{\mathcal{C}}$ = Nearest-Neighbor)



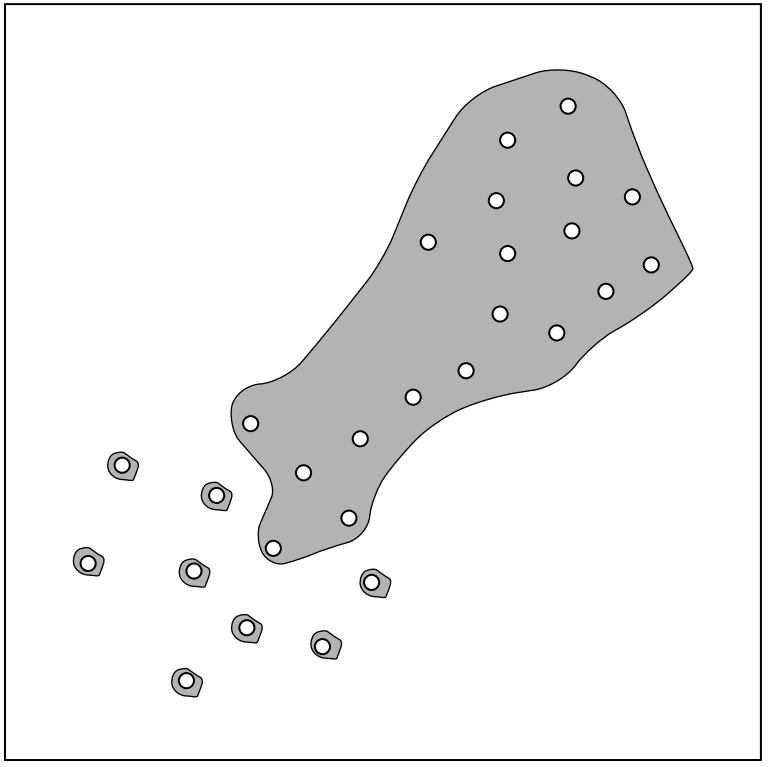
Hierarchische Verfahren

Chaining-Problematik bei Single-Link ($d_{\mathcal{C}}$ = Nearest-Neighbor)



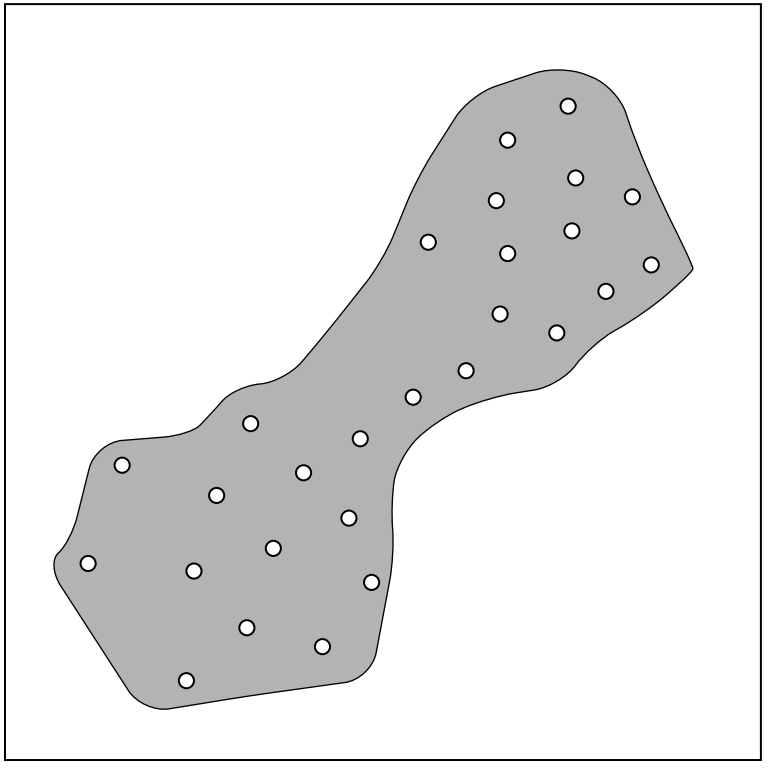
Hierarchische Verfahren

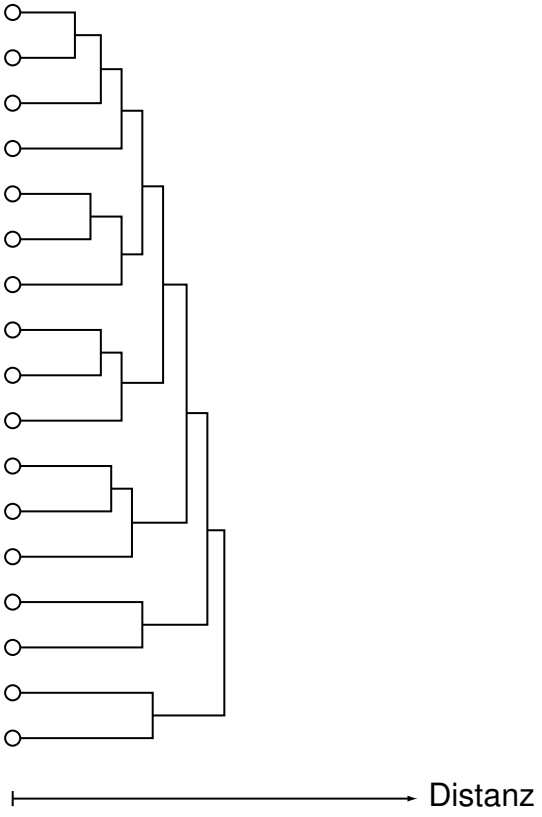
Chaining-Problematik bei Single-Link ($d_{\mathcal{C}}$ = Nearest-Neighbor)



Hierarchische Verfahren

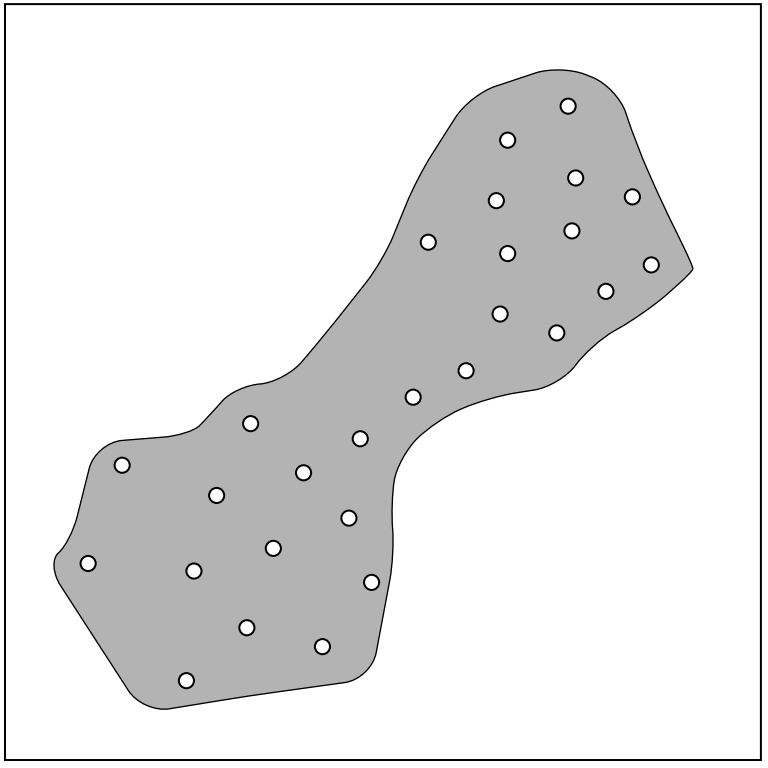
Chaining-Problematik bei Single-Link ($d_{\mathcal{C}}$ = Nearest-Neighbor)

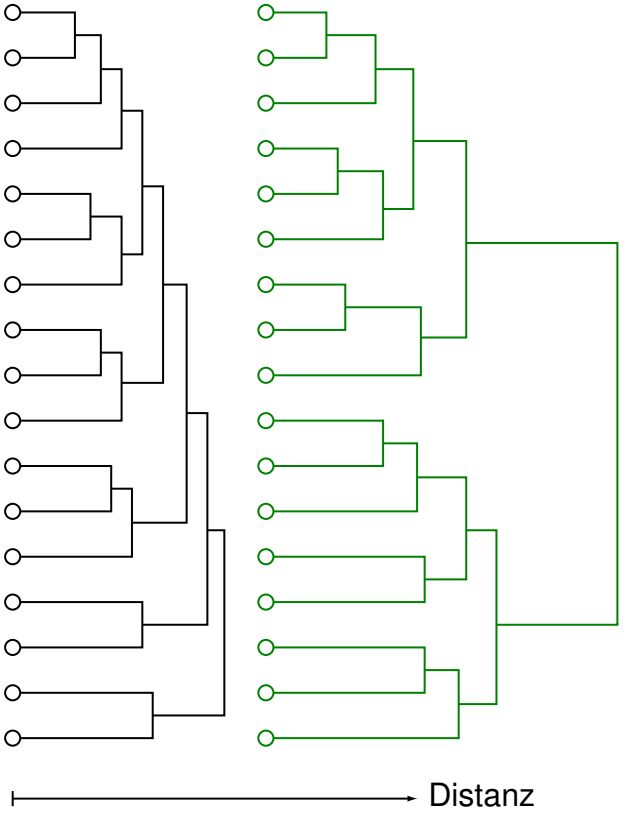




Hierarchische Verfahren

Chaining-Problematik bei Single-Link ($d_{\mathcal{C}}$ = Nearest-Neighbor)



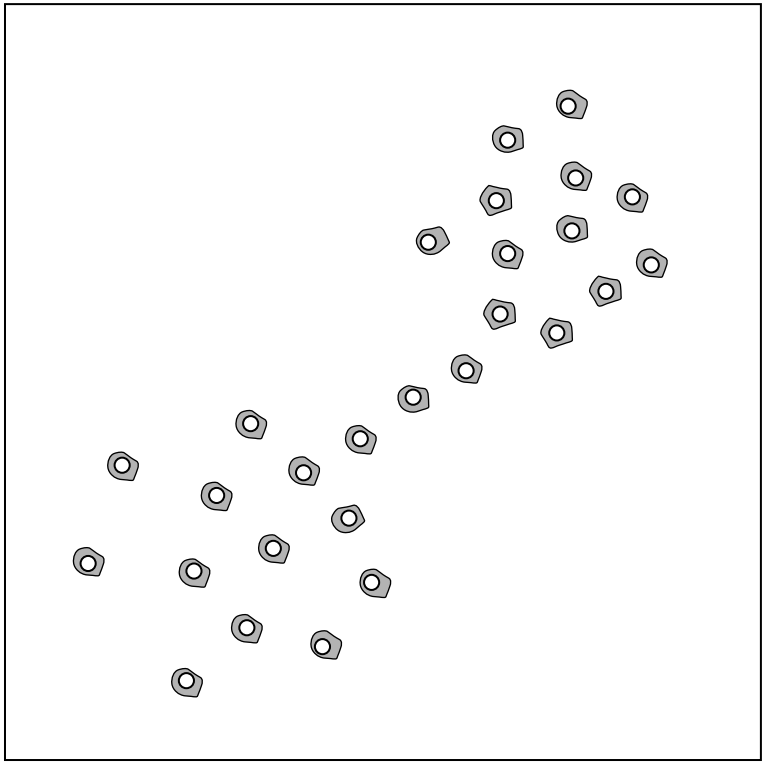


Bemerkungen:

- Mit einer k-Nearest-Neighbor-Variante könnte man das Problem entschärfen.
- Bei k-Nearest-Neighbor werden größere Cluster bei der Agglomeration bevorzugt, da sie mehr und damit – statistisch gesehen – auch mehr nähere Nachbarn besitzen.

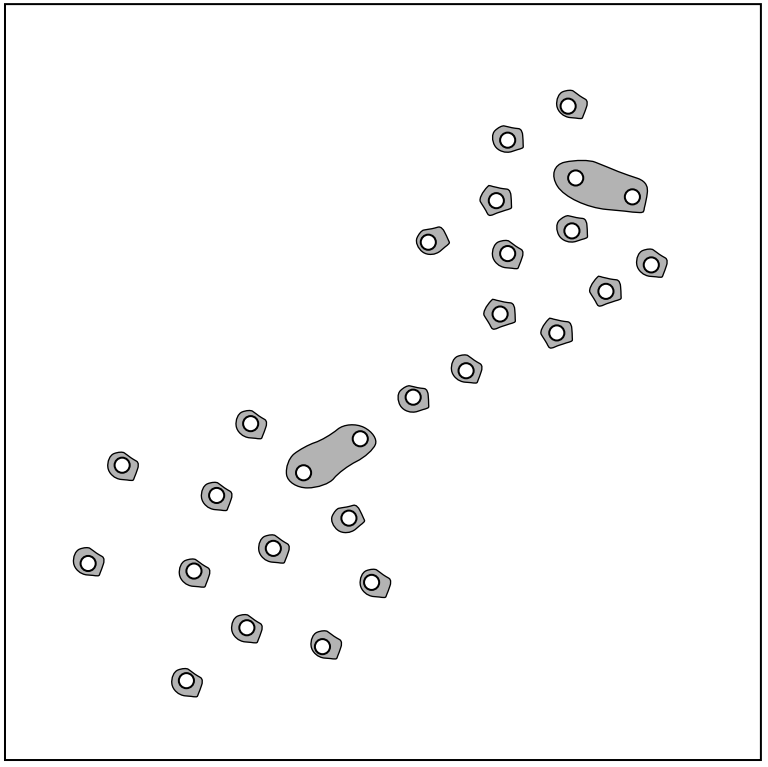
Hierarchische Verfahren

Chaining-Problematik bei Single-Link ($d_{\mathcal{C}}$ = k-Nearest-Neighbor)



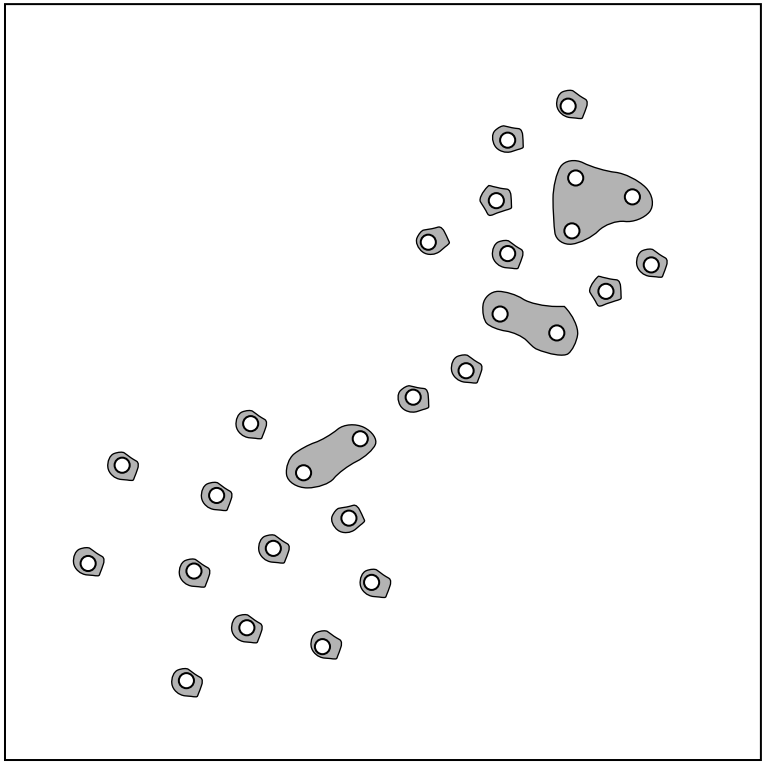
Hierarchische Verfahren

Chaining-Problematik bei Single-Link ($d_{\mathcal{C}}$ = k-Nearest-Neighbor)



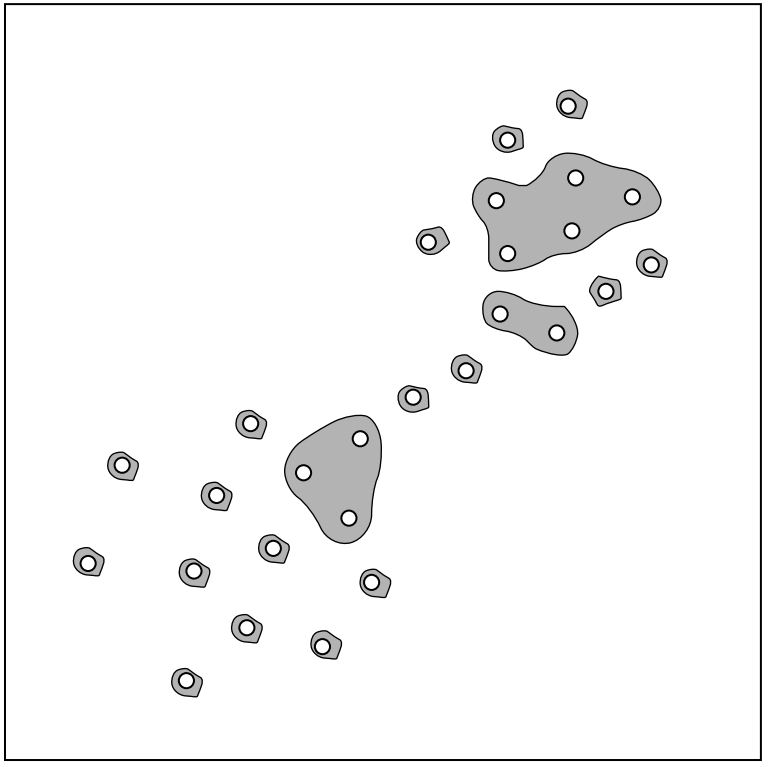
Hierarchische Verfahren

Chaining-Problematik bei Single-Link ($d_{\mathcal{C}}$ = k-Nearest-Neighbor)



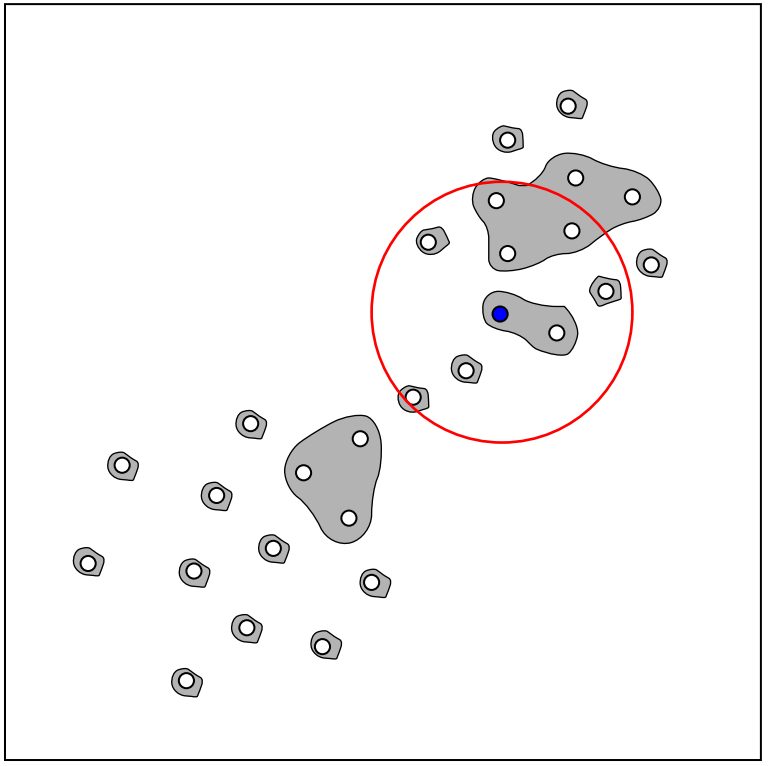
Hierarchische Verfahren

Chaining-Problematik bei Single-Link ($d_{\mathcal{C}}$ = k-Nearest-Neighbor)



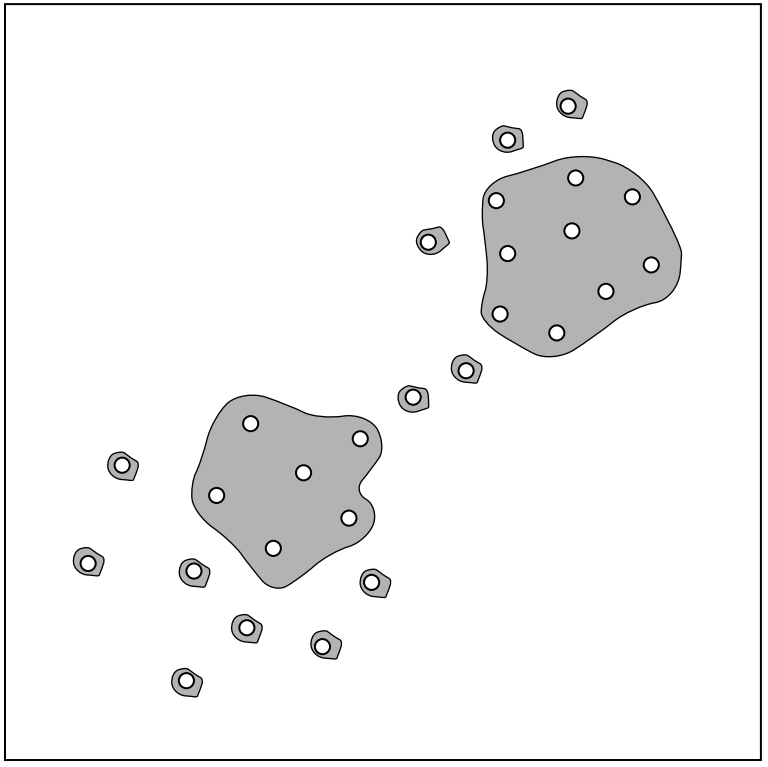
Hierarchische Verfahren

Chaining-Problematik bei Single-Link ($d_{\mathcal{C}}$ = k-Nearest-Neighbor)



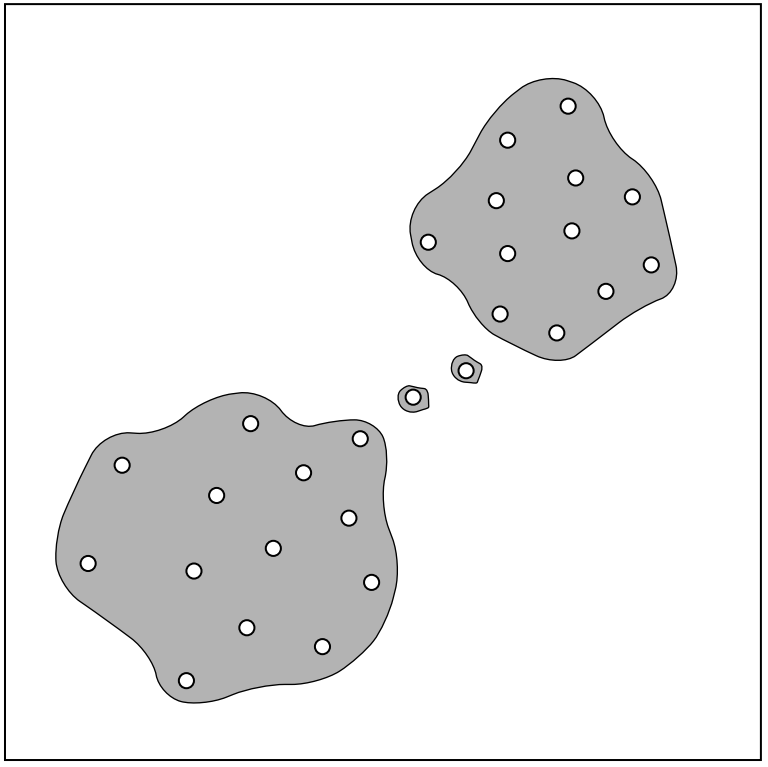
Hierarchische Verfahren

Chaining-Problematik bei Single-Link ($d_{\mathcal{C}}$ = k-Nearest-Neighbor)



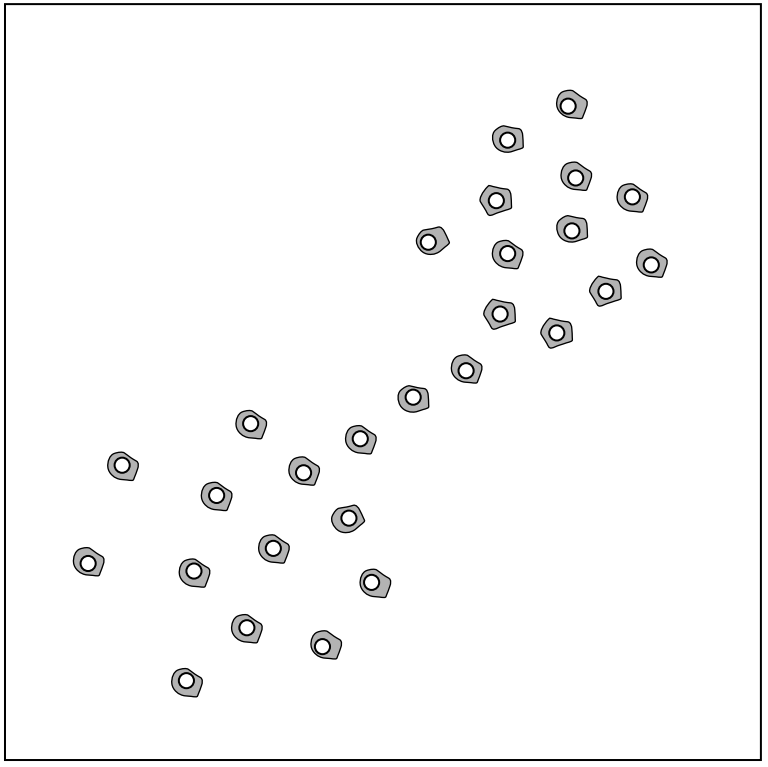
Hierarchische Verfahren

Chaining-Problematik bei Single-Link ($d_{\mathcal{C}}$ = k-Nearest-Neighbor)



Hierarchische Verfahren

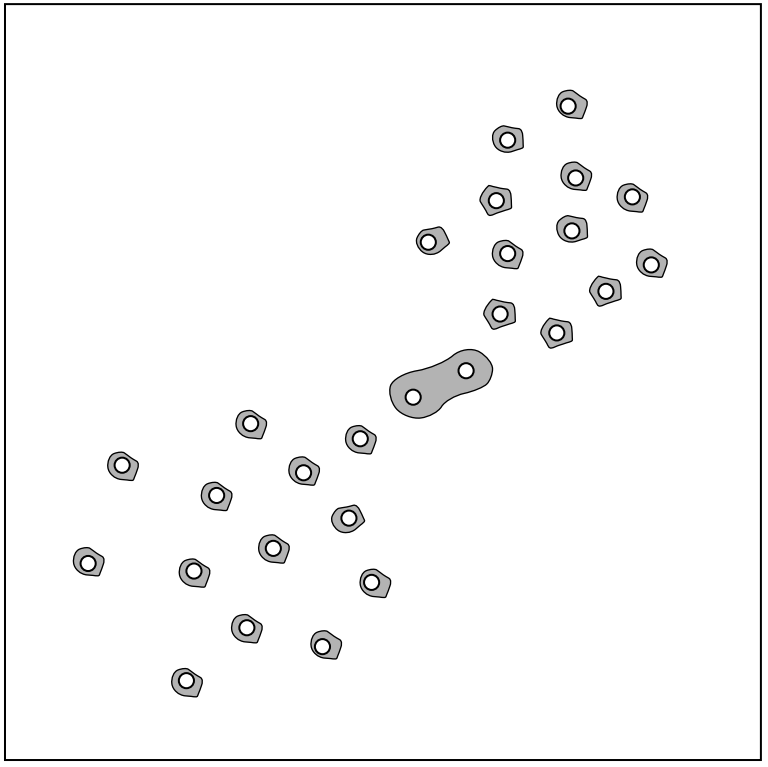
Chaining-Problematik bei Single-Link ($d_{\mathcal{C}}$ = k-Nearest-Neighbor)



In speziellen Situationen kann auch k-Nearest-Neighbor versagen.

Hierarchische Verfahren

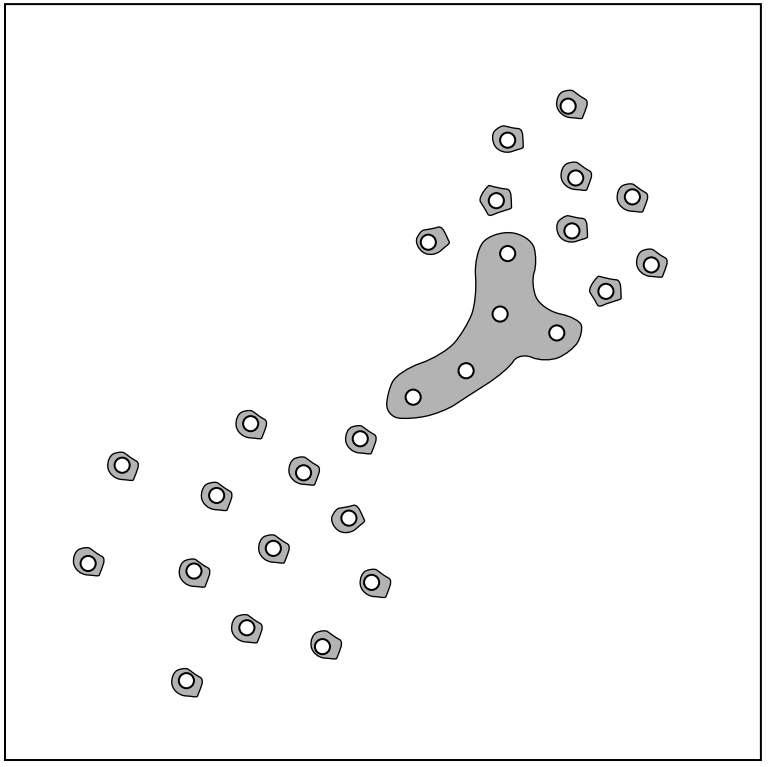
Chaining-Problematik bei Single-Link ($d_{\mathcal{C}}$ = k-Nearest-Neighbor)



In speziellen Situationen kann auch k-Nearest-Neighbor versagen.

Hierarchische Verfahren

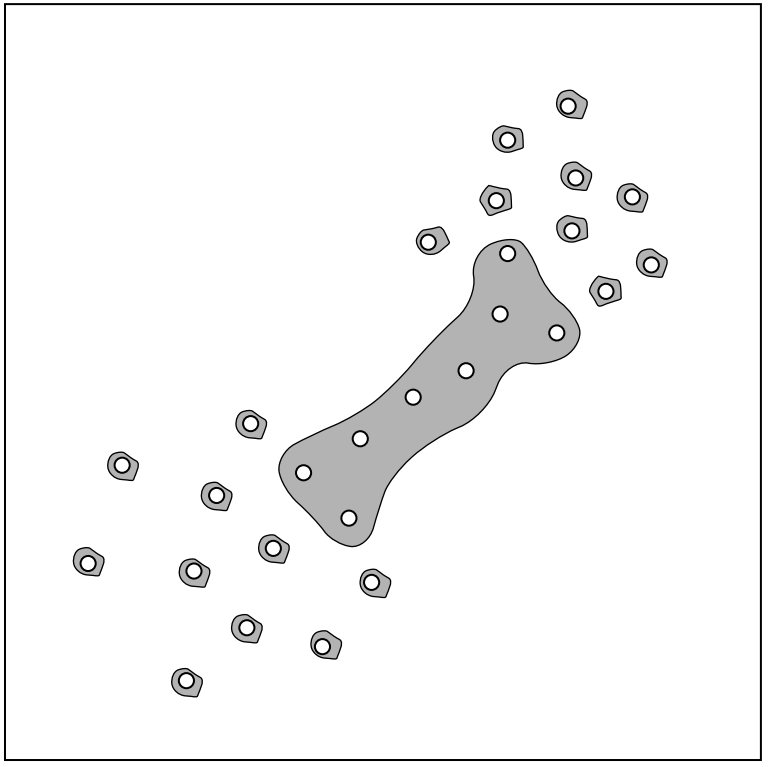
Chaining-Problematik bei Single-Link ($d_{\mathcal{C}}$ = k-Nearest-Neighbor)



In speziellen Situationen kann auch k-Nearest-Neighbor versagen.

Hierarchische Verfahren

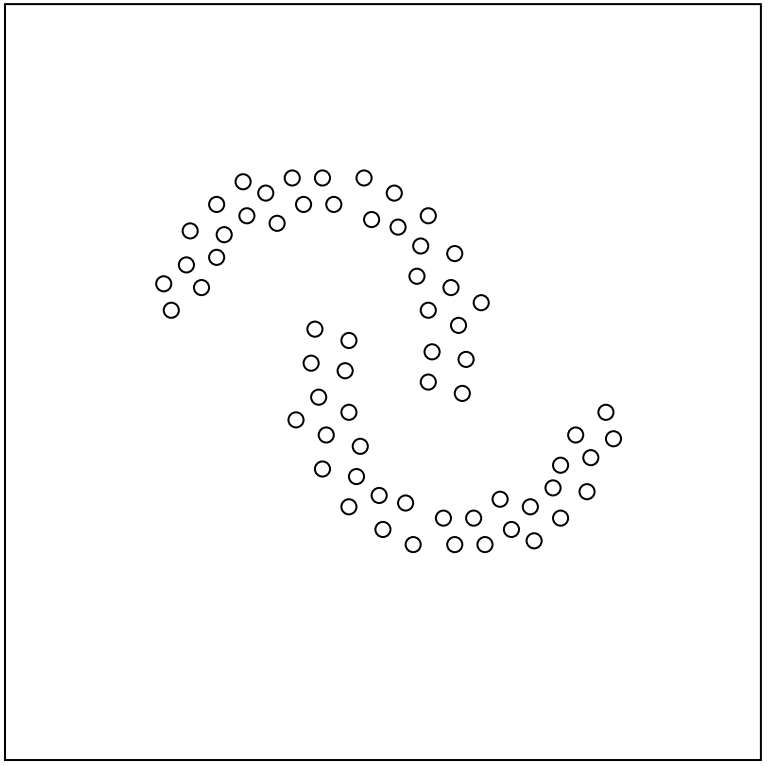
Chaining-Problematik bei Single-Link ($d_{\mathcal{C}}$ = k-Nearest-Neighbor)



In speziellen Situationen kann auch k-Nearest-Neighbor versagen.

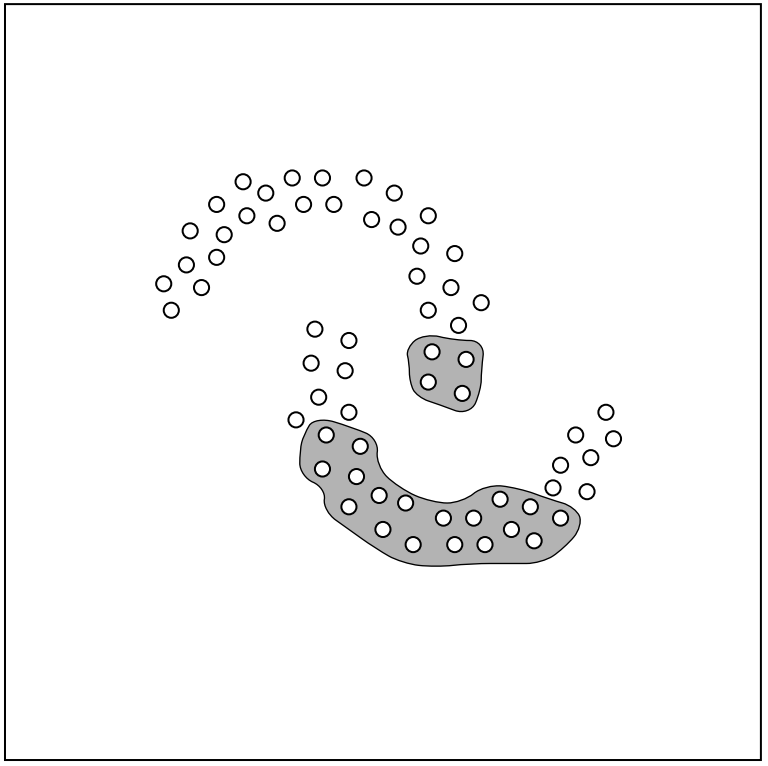
Hierarchische Verfahren

Überlappungsproblematik bei Complete-Link ($d_{\mathcal{C}}$ = Furthest-Neighbor)



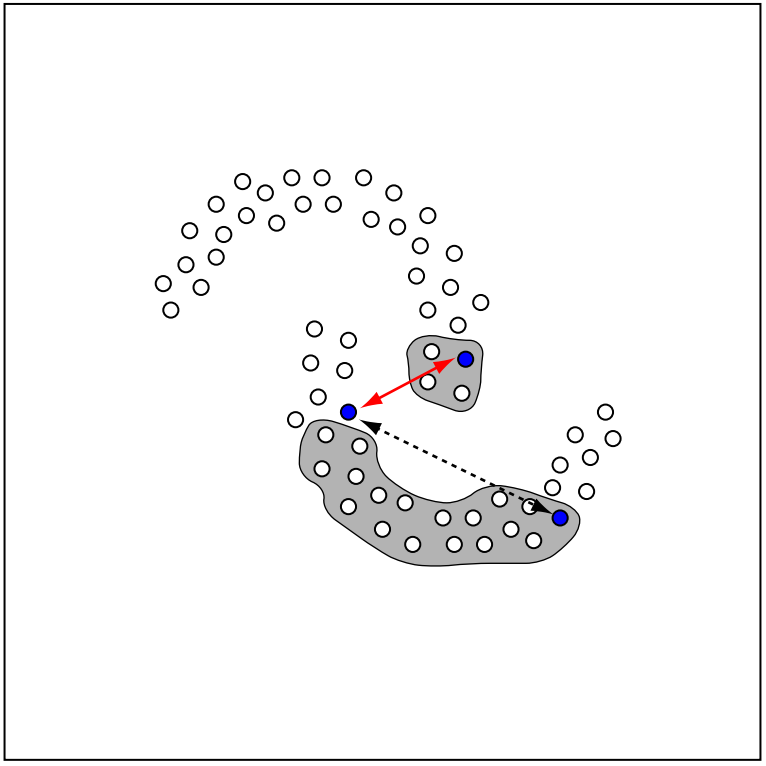
Hierarchische Verfahren

Überlappungsproblematik bei Complete-Link ($d_{\mathcal{C}}$ = Furthest-Neighbor)



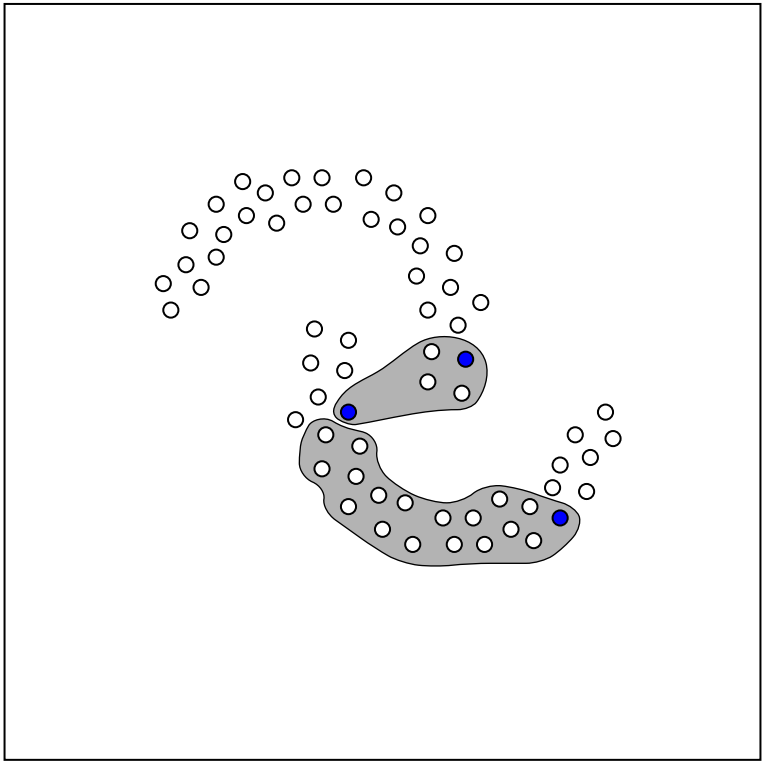
Hierarchische Verfahren

Überlappungsproblematik bei Complete-Link ($d_{\mathcal{C}}$ = Furthest-Neighbor)



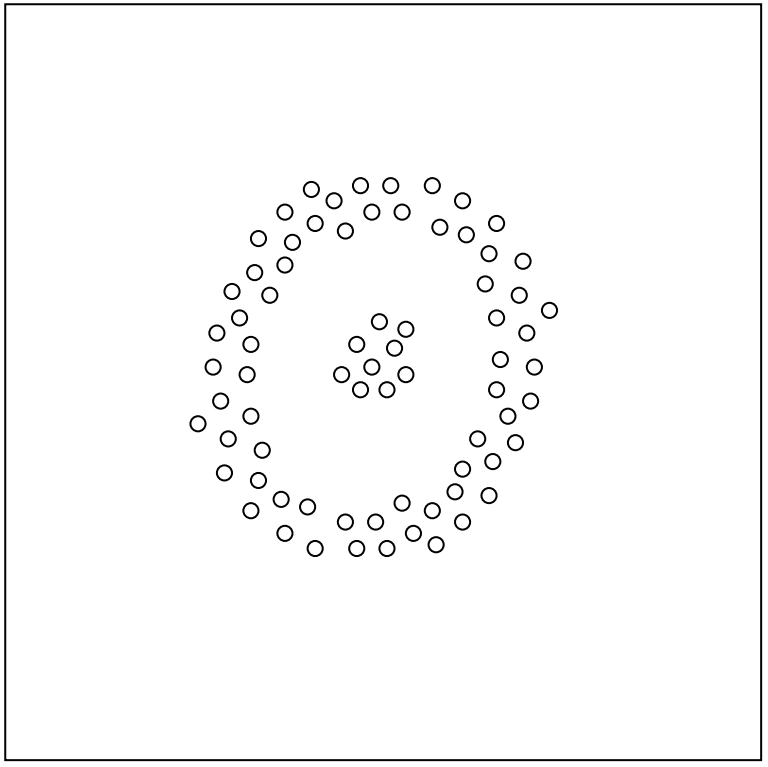
Hierarchische Verfahren

Überlappungsproblematik bei Complete-Link ($d_{\mathcal{C}}$ = Furthest-Neighbor)



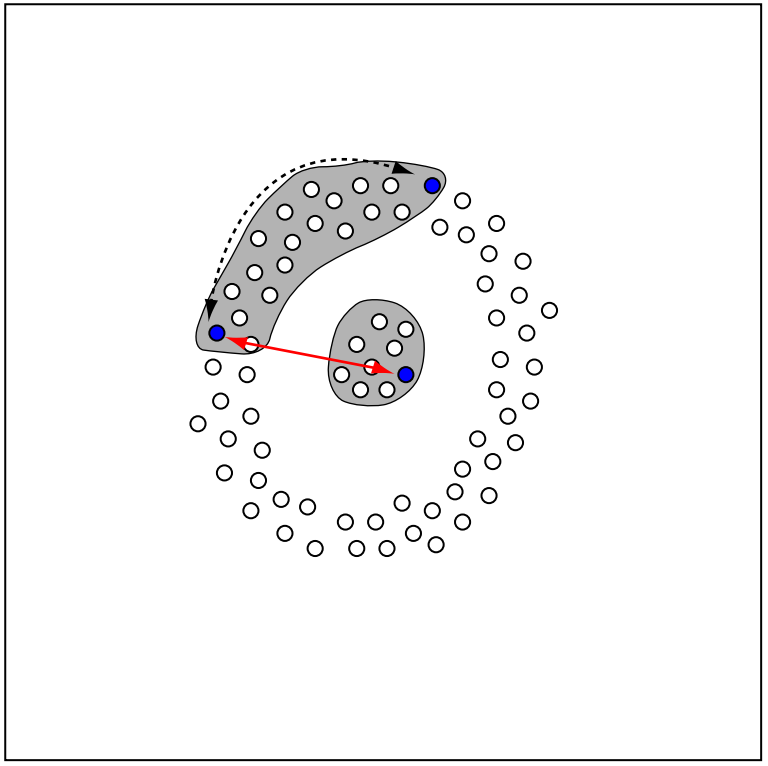
Hierarchische Verfahren

Überlappungsproblematik bei Complete-Link ($d_{\mathcal{C}}$ = Furthest-Neighbor)



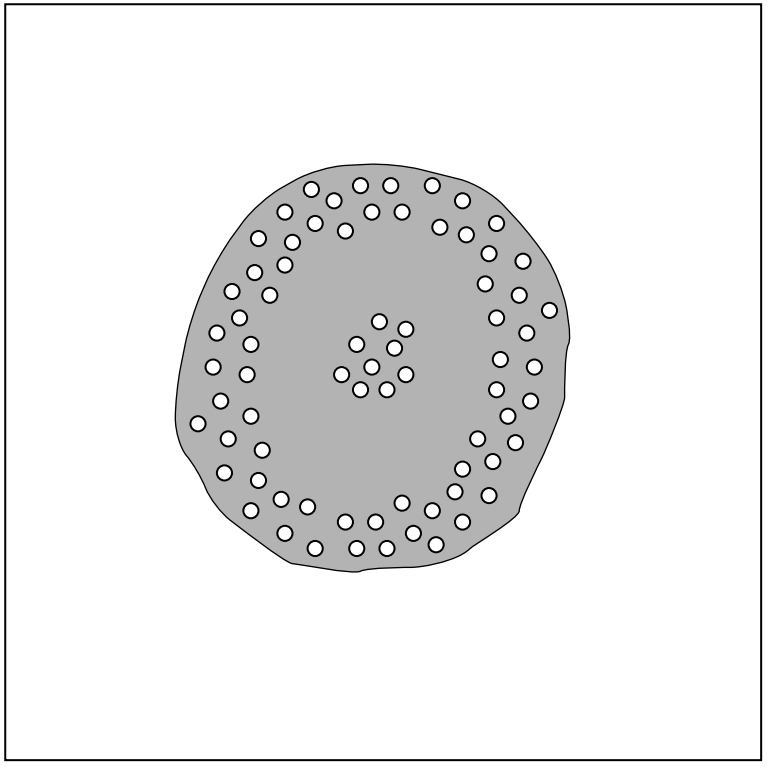
Hierarchische Verfahren

Überlappungsproblematik bei Complete-Link ($d_{\mathcal{C}}$ = Furthest-Neighbor)



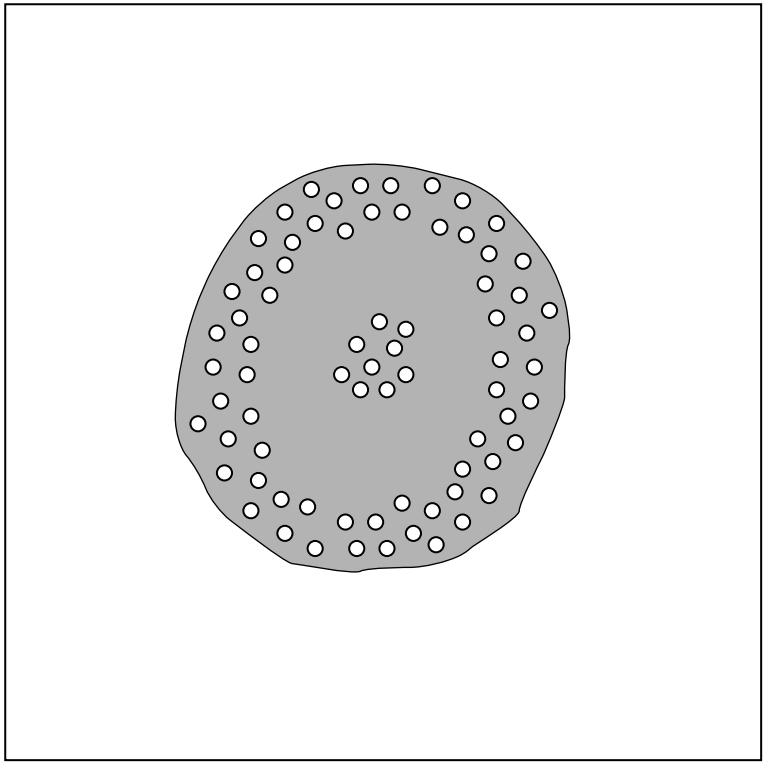
Hierarchische Verfahren

Überlappungsproblematik bei Complete-Link ($d_{\mathcal{C}}$ = Furthest-Neighbor)

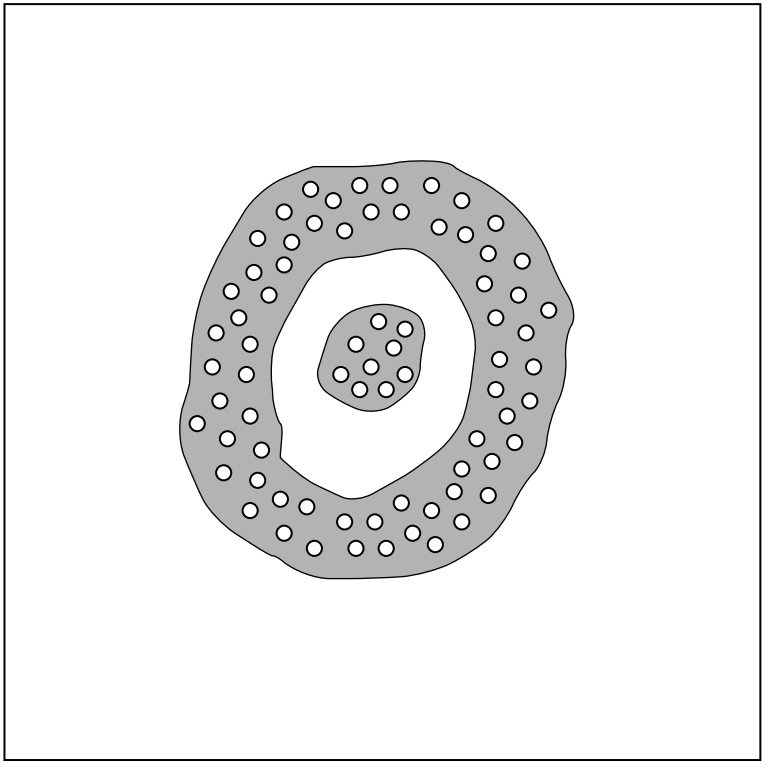


Hierarchische Verfahren

Überlappungsproblematik bei Complete-Link ($d_{\mathcal{C}}$ = Furthest-Neighbor)



Wirklichkeit



Wunsch

Hierarchische Verfahren

Gegenüberstellung bekannter hierarchisch-agglomerativer Verfahren

Geometrische Eigenschaften:

	Single-Link	Complete-Link	Average-Link	Ward
Charakteristik	kontrahierend:	dilatierend:	konservativ:	konservativ:
Cluster-Zahl	niedrig	hoch	mittel	mittel
Cluster-Form	ausgedehnt	klein	kompakt	sphärisch
Verkettungstendenz	stark	niedrig	niedrig	niedrig
ausreißerentdeckend	hoch	sehr niedrig	niedrig	niedrig

Hierarchische Verfahren

Gegenüberstellung bekannter hierarchisch-agglomerativer Verfahren

Geometrische Eigenschaften:

	Single-Link	Complete-Link	Average-Link	Ward
Charakteristik	kontrahierend:	dilatierend:	konservativ:	konservativ:
Cluster-Zahl	niedrig	hoch	mittel	mittel
Cluster-Form	ausgedehnt	klein	kompakt	sphärisch
Verkettungstendenz	stark	niedrig	niedrig	niedrig
ausreißerentdeckend	hoch	sehr niedrig	niedrig	niedrig

Datenbezogene Eigenschaften:

verrauschte Daten	empfindlich	empfindlich	unbeeinflusst	unbeeinflusst
Merkmaltransformation	invariant	invariant	–	–

Hierarchische Verfahren

Gegenüberstellung bekannter hierarchisch-agglomerativer Verfahren

Geometrische Eigenschaften:

	Single-Link	Complete-Link	Average-Link	Ward
Charakteristik	kontrahierend:	dilatierend:	konservativ:	konservativ:
Cluster-Zahl	niedrig	hoch	mittel	mittel
Cluster-Form	ausgedehnt	klein	kompakt	sphärisch
Verkettungstendenz	stark	niedrig	niedrig	niedrig
ausreißerentdeckend	hoch	sehr niedrig	niedrig	niedrig

Datenbezogene Eigenschaften:

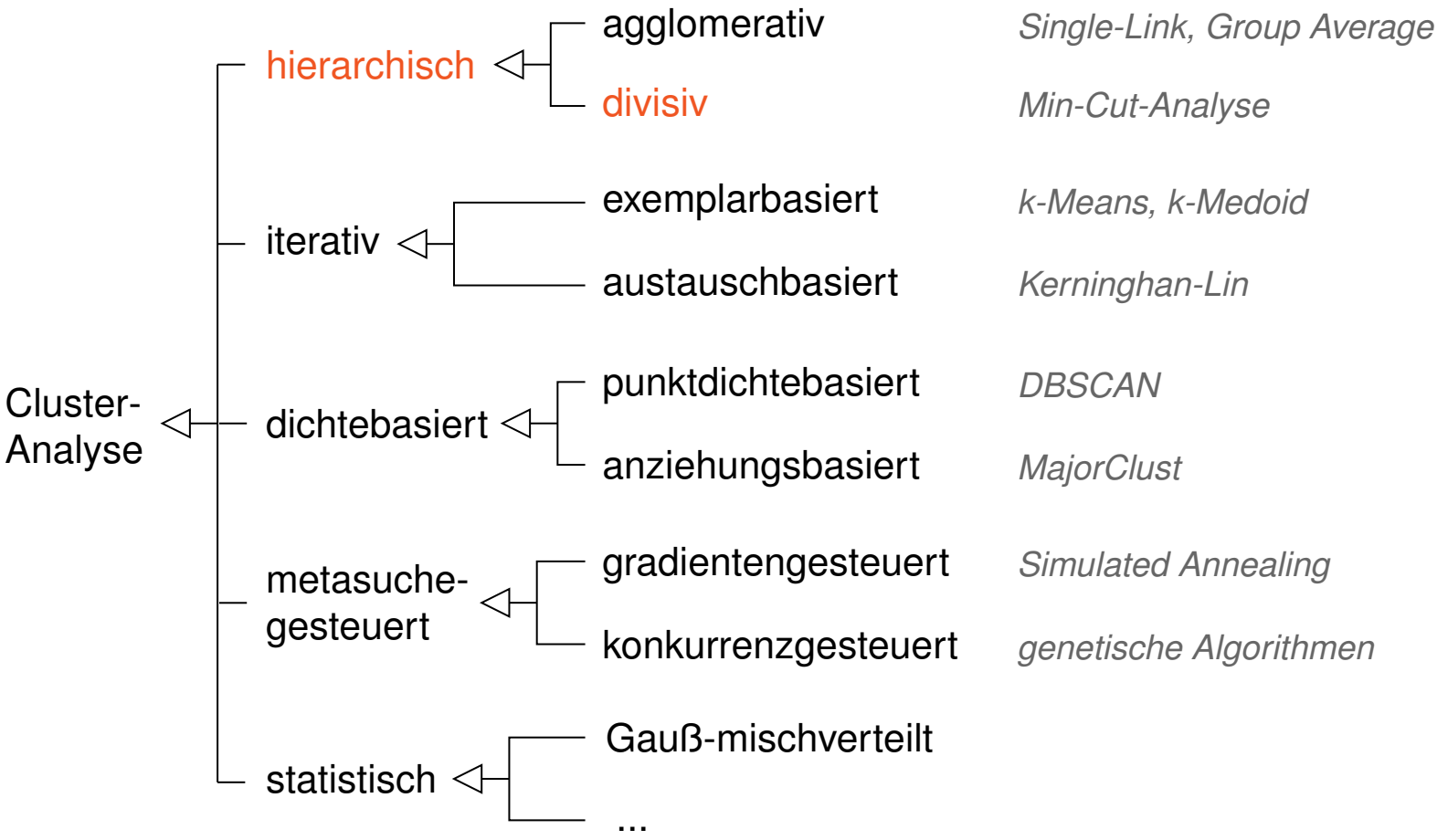
verrauschte Daten	empfindlich	empfindlich	unbeeinflusst	unbeeinflusst
Merkmaltransformation	invariant	invariant	–	–

Eigenschaften des Cluster-Distanzmaßes:

$d_{\mathcal{C}}$ monoton	✓	✓	✓	✓
$d_{\mathcal{C}}$ reihenfolgeunabh.	✓	✓	✓	✓
$d_{\mathcal{C}}$ konsistent	$\longrightarrow 0$	$\longrightarrow \infty$	✓	$\longrightarrow \infty$

Hierarchische Verfahren

Prinzipien der Fusionierung



Hierarchische Verfahren

Algorithmus zur hierarchisch-divisiven Cluster-Analyse

Input: $G = \langle V, E, w \rangle$. Weighted graph.
$d_{\mathcal{C}}$. Distance measure between two clusters.

Output: $T = \langle V_T, E_T \rangle$. Cluster hierarchy or dendrogram.

1. $\mathcal{C} = \{V\}$ `// define initial clustering`
2. $V_T = \{v_C \mid C \in \mathcal{C}\}, \quad E_T = \emptyset$ `// define initial dendrogram`
3. **WHILE** $\exists C_x : (C_x \in \mathcal{C} \ \wedge \ |C_x| > 1)$ **DO**
4. $\{C, C'\} = \underset{\{C_i, C_j\}: \ C_i \cup C_j = C_x \ \wedge \ C_i \cap C_j = \emptyset}{\text{argmax}} d_{\mathcal{C}}(C_i, C_j)$
5. $\mathcal{C} = (\mathcal{C} \setminus \{C_x\}) \ \cup \ \{C, C'\}$
6. $V_T = V_T \cup \{v_C, v_{C'}\}, \quad E_T = E_T \cup \{\{v_{C_x}, v_C\}, \{v_{C_x}, v_{C'}\}\}$
7. **ENDDO**
8. **RETURN**(T)

Vergleiche hierzu den Algorithmus zur hierarchisch-agglomerativen Cluster-Analyse.

Bemerkungen:

- Im Prinzip kann $d_{\mathcal{C}}$ wie bei den hierarchisch-agglomerativen Verfahren gewählt werden. Die Worst-Case-Komplexität ist exponentiell statt quadratisch.
- Hierarchisch-divisive Verfahren werden oft als *monothetische* Variante konstruiert: In jedem Entscheidungsschritt wird nur *eine* Variable betrachtet.
- Im Gegensatz zu hierarchisch-agglomerativen Verfahren darf ein hierarchisch-divisives Verfahren keinen „Fehler“ bei den ersten Schritten machen.
- Ein mächtiges hierarchisch-divisives Cluster-Analyse-Verfahren entsteht mit

$$sim_{\mathcal{C}}(C, C') = \sum_{e \in cut(\{C, C'\})} w(e) \quad \text{bzw.} \quad d_{\mathcal{C}}(C, C') = \frac{1}{sim_{\mathcal{C}}(C, C')}$$

Hierarchische Verfahren

MinCut-Cluster-Analyse

Definition 19 (Cut, minimaler Cut)
Sei $G = \langle V, E, w \rangle$ ein Graph mit nicht-negativer Gewichtsfunktion w; weiterhin sei $U \subset V$ eine nichtleere Teilmenge der Knotenmenge V und $\bar{U} = V \setminus U$. Der Cut zwischen U und $\bar{U}$ ist wie folgt definiert:

$$cut(\{U, \bar{U}\}) = \{\{u, v\} \mid \{u, v\} \in E, u \in U, v \in \bar{U}\}$$

Hierarchische Verfahren

MinCut-Cluster-Analyse

Definition 19 (Cut, minimaler Cut)
Sei $G = \langle V, E, w \rangle$ ein Graph mit nicht-negativer Gewichtsfunktion w; weiterhin sei $U \subset V$ eine nichtleere Teilmenge der Knotenmenge V und $\bar{U} = V \setminus U$. Der Cut zwischen U und $\bar{U}$ ist wie folgt definiert:

$$cut(\{U, \bar{U}\}) = \{\{u, v\} \mid \{u, v\} \in E, u \in U, v \in \bar{U}\}$$

Weiterhin bezeichne $w(\{U, \bar{U}\})$ das Gewicht oder die Kapazität von $cut(\{U, \bar{U}\})$:

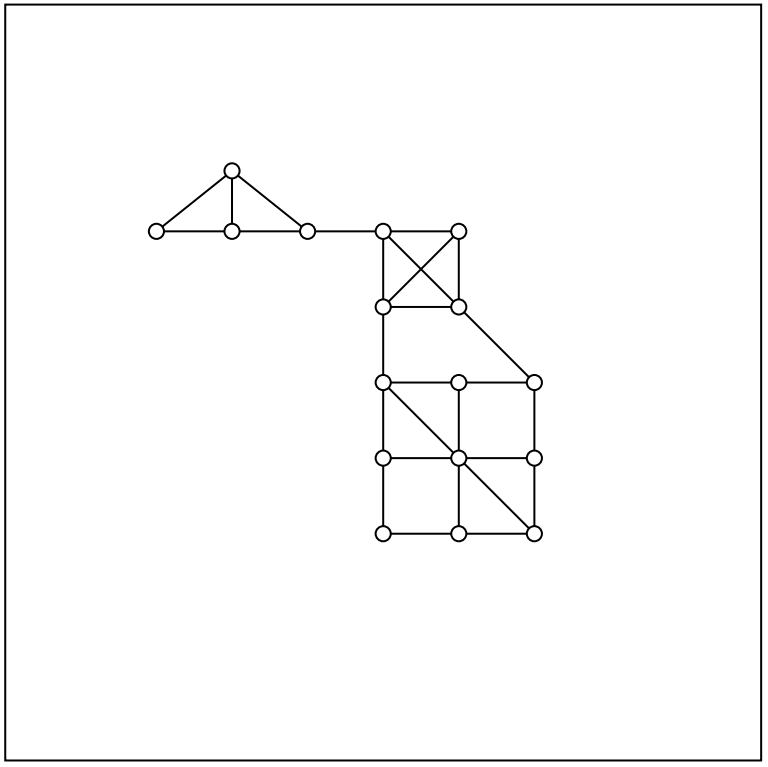
$$w(\{U, \bar{U}\}) = \sum_{e \in cut(\{U, \bar{U}\})} w(e)$$

$cut(\{U, \bar{U}\})$ heißt minimaler Cut von G (minium capacity cut), wenn für alle Zerlegungen $\{W, \bar{W}\}$, $W, \bar{W} \neq \emptyset$ gilt:

$$w(\{U, \bar{U}\}) \leq w(\{W, \bar{W}\})$$

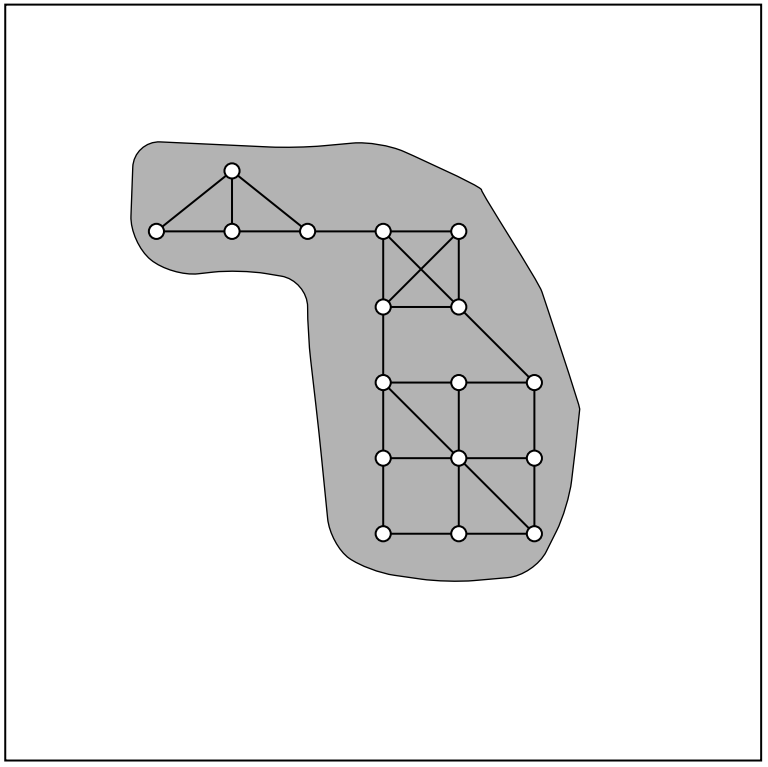
Hierarchische Verfahren

MinCut-Cluster-Analyse



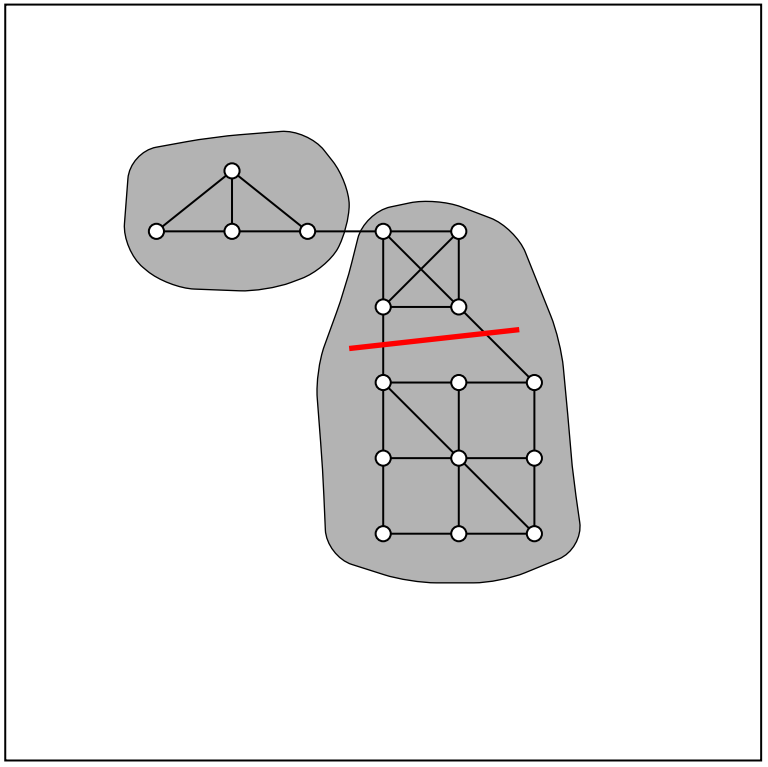
Hierarchische Verfahren

MinCut-Cluster-Analyse



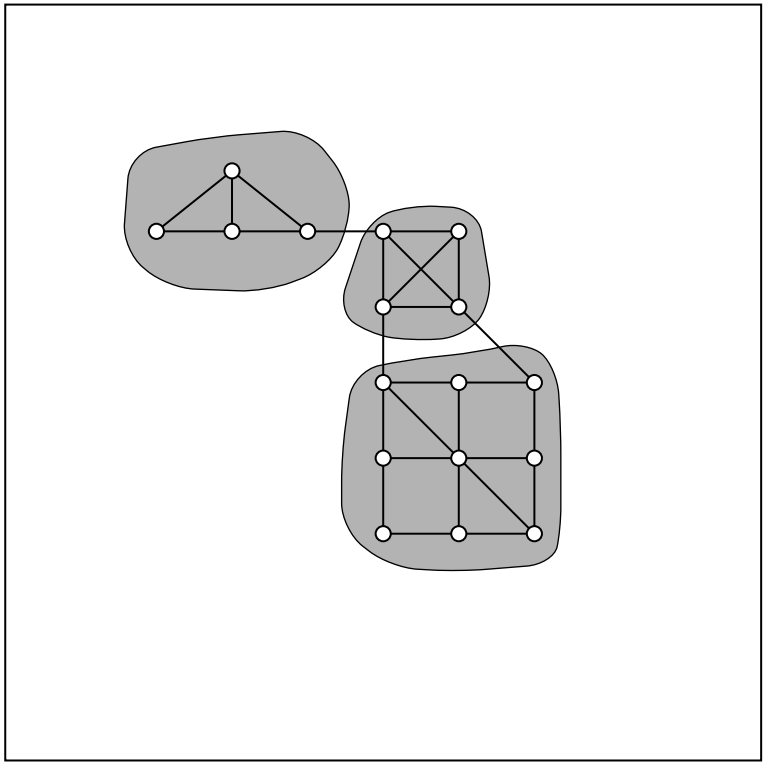
Hierarchische Verfahren

MinCut-Cluster-Analyse



Hierarchische Verfahren

MinCut-Cluster-Analyse

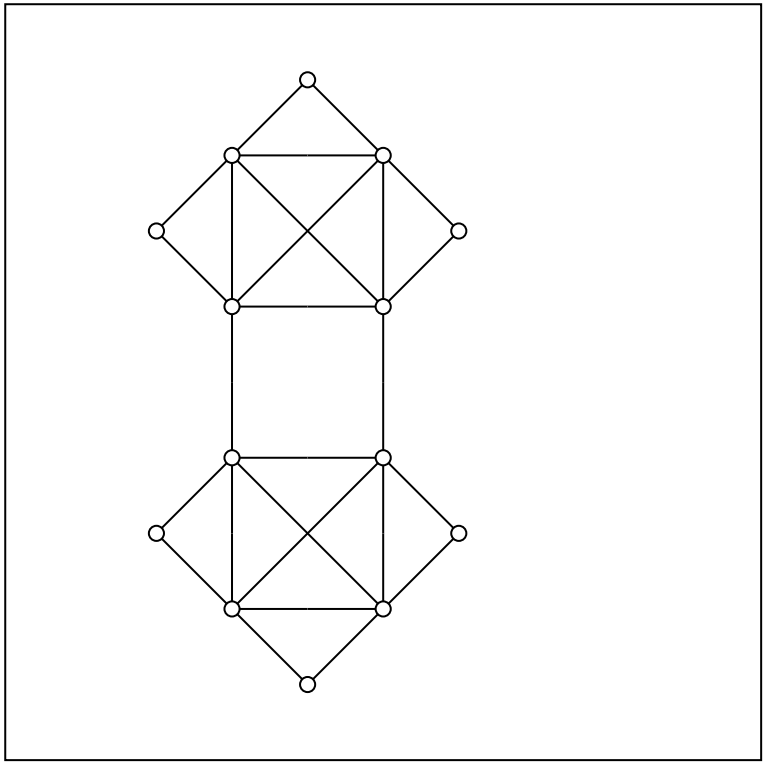


Bemerkungen:

- Jede Aufteilung erfordert die Berechnung eines Cuts minimaler Kapazität.
- Der beste bekannte Algorithmus zur Berechnung eines minimalen Cuts ist in $O(|V| \cdot |E| + |V|^2 \cdot \log|V|)$. [Nagamochi/Ono/Ibaraki 1994]
- Es sind dass $|V|$ Berechnungen eines Minimum-Capacity-Cuts notwendig, um eine vollständige Zerlegung (= ein Knoten pro Cluster) herzustellen.
- Der Aufwand zur Berechnung eines Minimum s-t-Cut ist in $O(|V|^2 \log(|E|)$.
- Der Aufwand zur Berechnung eines Balanced Min-Cut (k-way, $k \geq 2$) ist NP-vollständig.

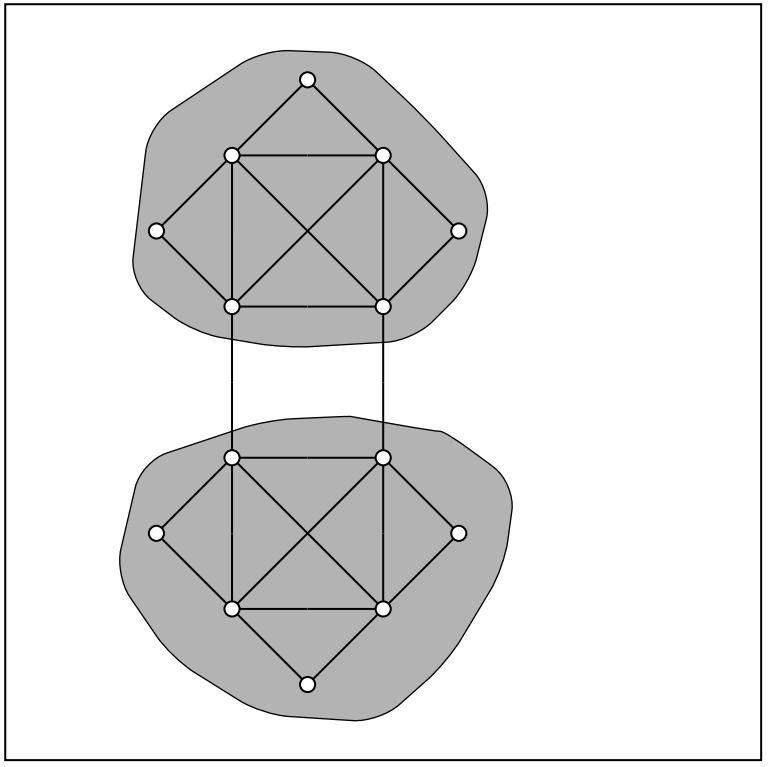
Hierarchische Verfahren

Splitting-Problematik bei MinCut-Cluster-Analyse



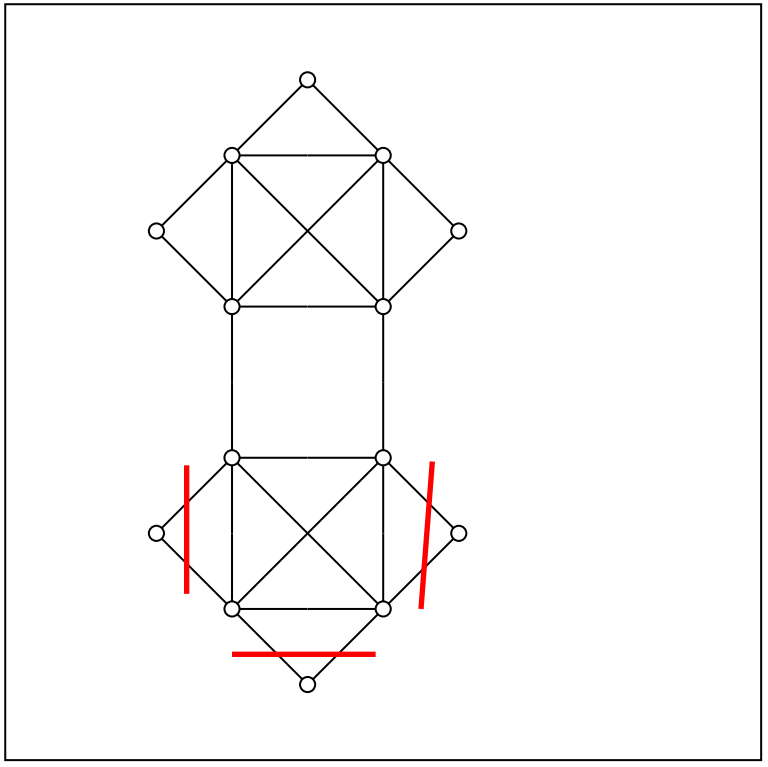
Hierarchische Verfahren

Splitting-Problematik bei MinCut-Cluster-Analyse



Hierarchische Verfahren

Splitting-Problematik bei MinCut-Cluster-Analyse



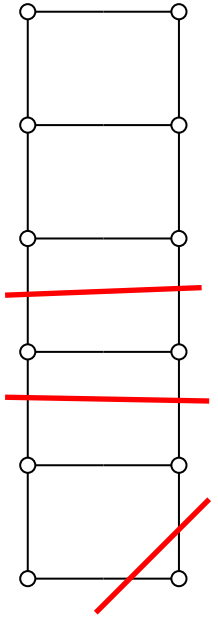
Ausweg: Normalisierung der Cut-Kapazität bzgl. der Größe der Knotenmengen.

Hierarchische Verfahren

Splitting-Problematik bei MinCut-Cluster-Analyse

Normalisierte Cut-Kapazität: $\overline{w}(\{U, \bar{U}\}) = \frac{w(\{U, \bar{U}\})}{w(\{U, V\})} + \frac{w(\{U, \bar{U}\})}{w(\{\bar{U}, V\})}$

Illustration von $\overline{w}$:



$\overline{w}$ = 2/9 + 2/9 ≈ 0.44

$\overline{w}$ = 2/6 + 2/12 = 0.5

$\overline{w}$ = 2/2 + 2/16 = 1.125

Bemerkungen:

- Die Bestimmung eines Cuts minimaler, normalisierter Kapazität ist NP-vollständig.
- Es gibt effiziente Näherungslösungen zur Berechnung von $\overline{w}(\{U, \bar{U}\})$. Das Verfahren wird angewandt im Bereich der Bildsegmentierung und der Gene-Expression-Cluster-Analyse. [Shi/Malik 2000]

Hierarchische Verfahren

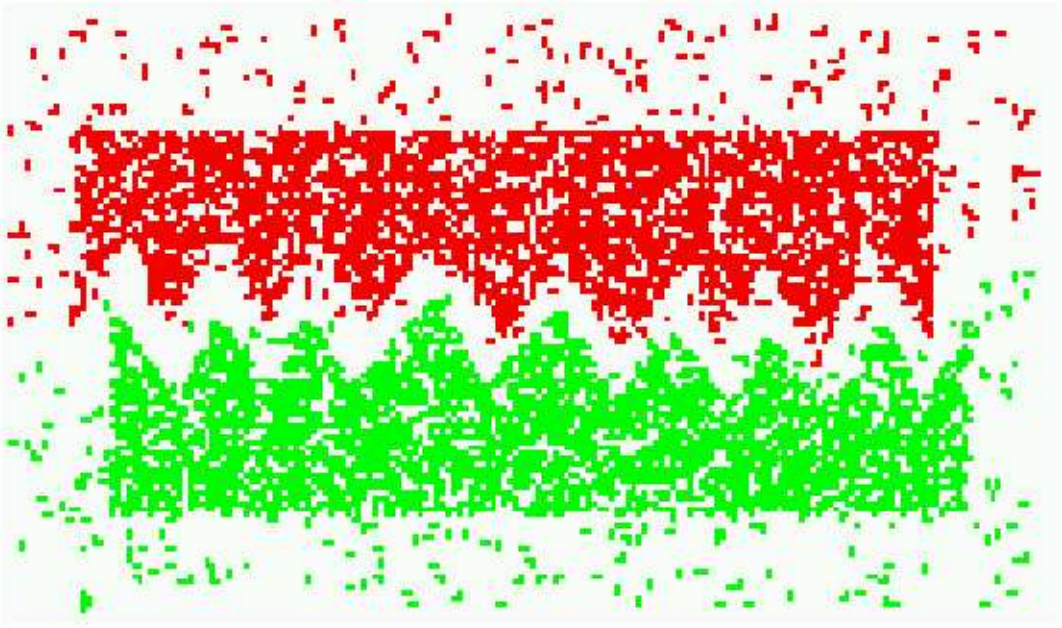
Kombination hierarchischer Verfahren

Das System Chameleon kombiniert die Schritte Graphausdünnung, Graphpartionierung und hierarchische Cluster-Analyse [Karypis/Han/Kumar 2000] :

Hierarchische Verfahren

Kombination hierarchischer Verfahren

Das System Chameleon kombiniert die Schritte Graphausdünnung, Graphpartionierung und hierarchische Cluster-Analyse [Karypis/Han/Kumar 2000] :

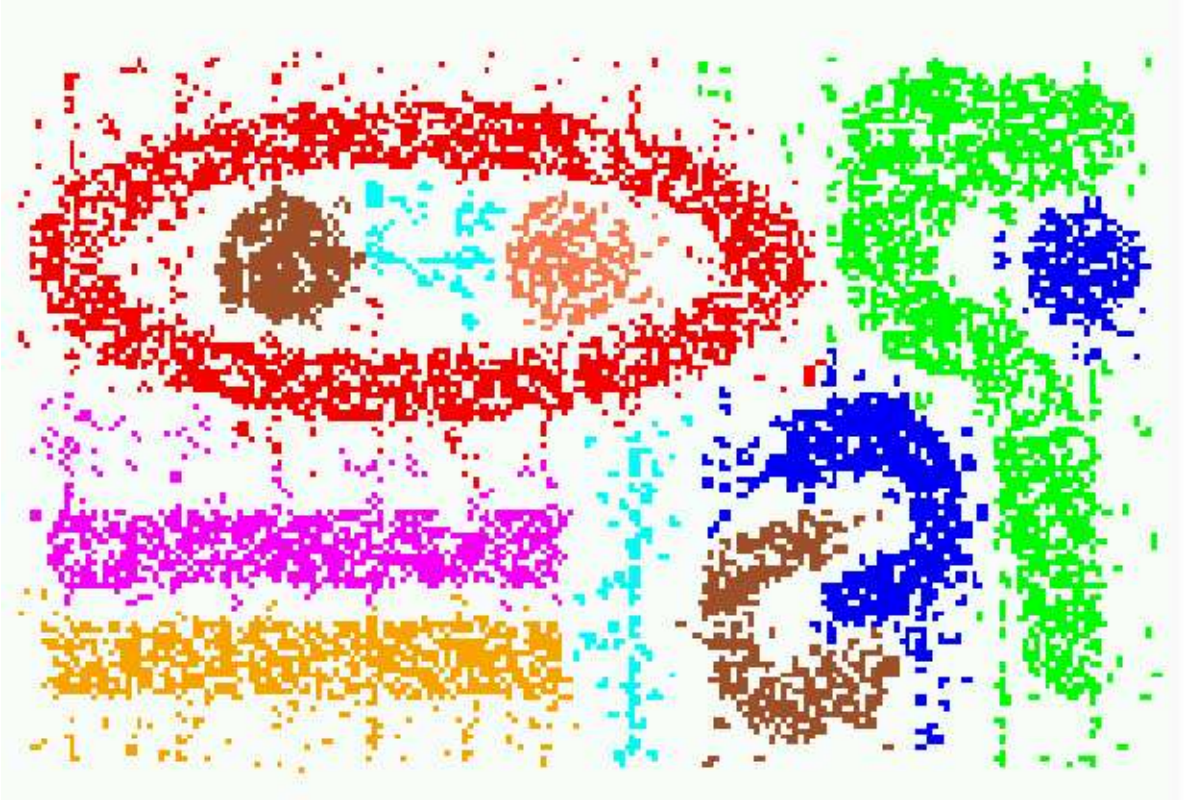


Die Clusterdistanz $d_{\mathcal{C}}(C, C')$ ist definiert als $d_{\mathcal{C}} = \frac{1}{R_I(C,C') \cdot (R_C(C,C'))^{\alpha}}$

Hierarchische Verfahren

Kombination hierarchischer Verfahren

Chameleon [Karypis/Han/Kumar 2000] :



Der Parameter α in $d_{\mathcal{C}}$ ist vom Anwender problemabhängig zu bestimmen.

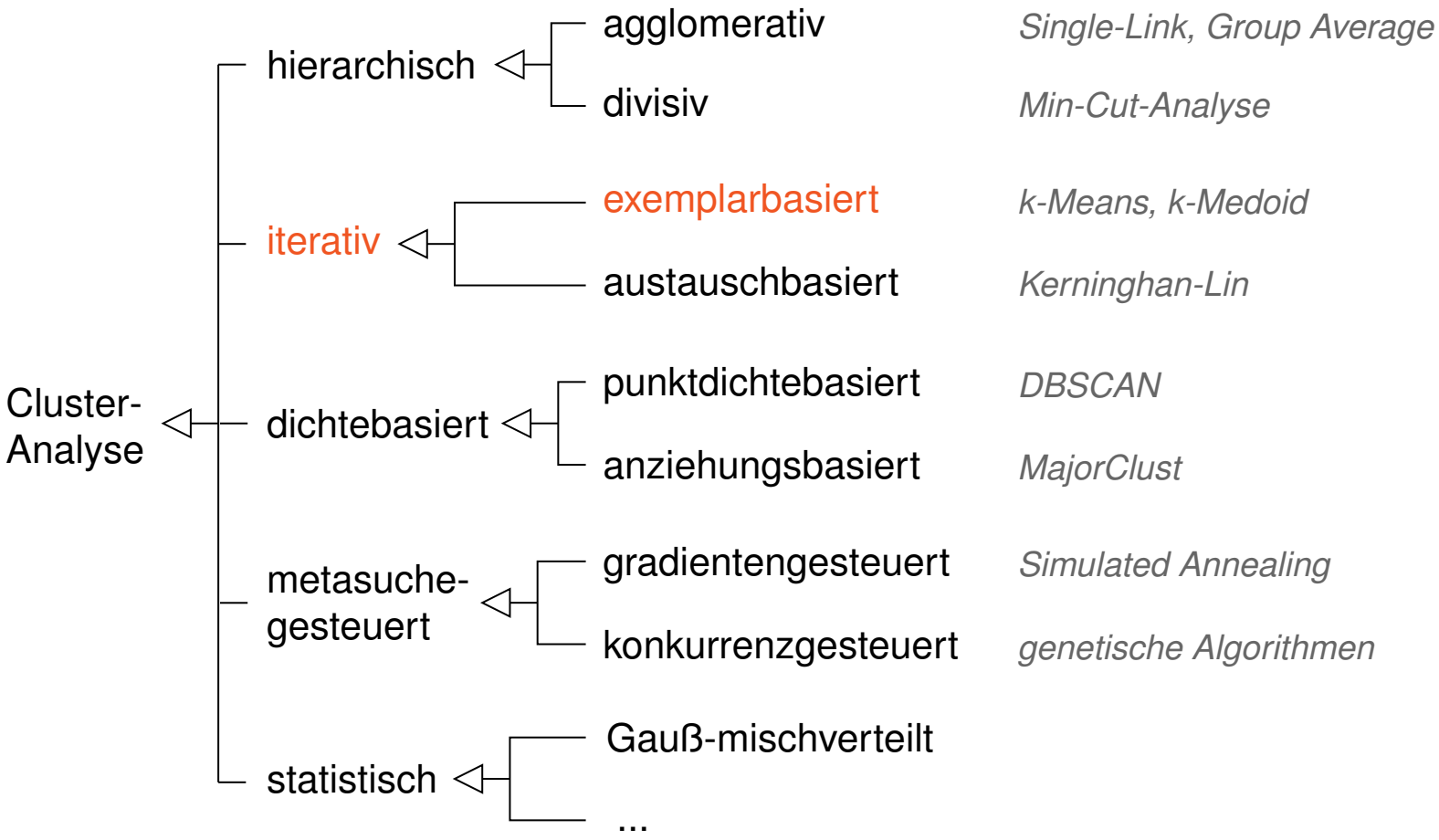
Kapitel ML: X (Fortsetzung)

X. Cluster-Analyse

- Einordnung Data Mining
- Einführung in die Cluster-Analyse
- Hierarchische Verfahren
- Iterative Verfahren
- Dichtebasierte Verfahren
- Cluster-Evaluierung

Iterative Verfahren

Prinzipien der Fusionierung



Iterative Verfahren

Algorithmus zur exemplarbasierten Cluster-Analyse

Input: $G = \langle V, E, w \rangle$. Weighted graph.
d. Distance function for nodes in V.
e. Minimization criterion for cluster representatives, based on d.
k. Number of desired clusters.

Output: $r_1, \ldots, r_k$. Cluster representatives.

```
1.
2. FOR i = 1 to k DO r_i(t) = choose(V)   // init representatives
3.
4.
5.   FOREACH v ∈ V DO   // find nearest representative (cluster)
6.     x = argmin_{i: i∈{1,...,k}} d(r_i(t), v), C_x = C_x ∪ {v}
7.   ENDDO
8.   FOR i = 1 to k DO r_i(t) = minimize(e, C_i)   // update
9.
10.
```

Iterative Verfahren

Algorithmus zur exemplarbasierten Cluster-Analyse

Input: $G = \langle V, E, w \rangle$. Weighted graph.
d. Distance function for nodes in V.
e. Minimization criterion for cluster representatives, based on d.
k. Number of desired clusters.

Output: $r_1, \ldots, r_k$. Cluster representatives.

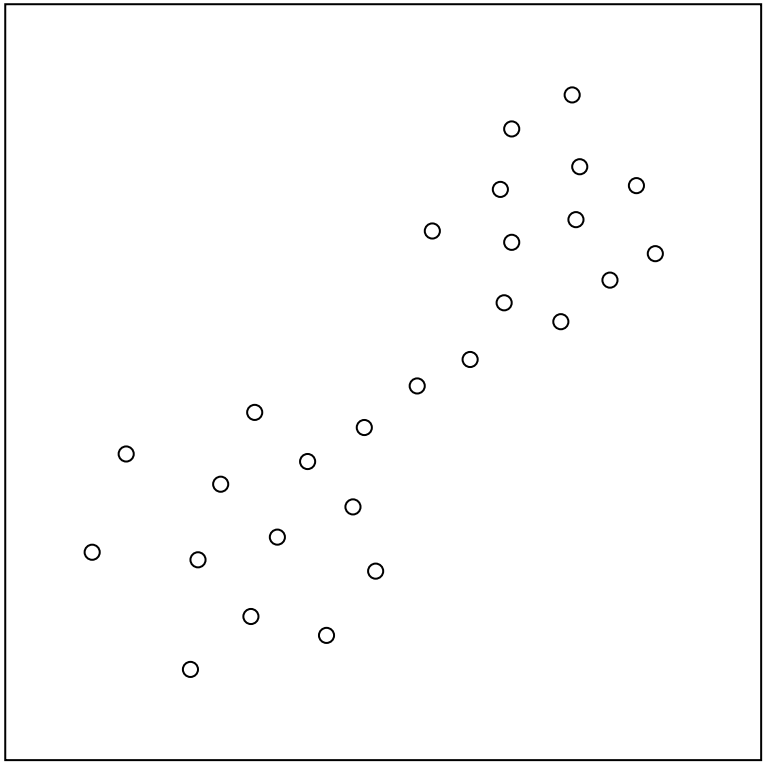
1. $t = 0$
2. **FOR** $i = 1$ to k **DO** $r_i(t) = \mathit{choose}(V)$ // init representatives
3. **REPEAT**
4. **FOR** $i = 1$ to k **DO** $C_i = \emptyset$
5. **FOREACH** $v \in V$ **DO** // find nearest representative (cluster)
6. $x = \underset{i:\, i \in \{1,\ldots,k\}}{\text{argmin}}\ d(r_i(t),\, v)$, $C_x = C_x \cup \{v\}$
7. **ENDDO**
8. **FOR** $i = 1$ to k **DO** $r_i(t) = \mathit{minimize}(e, C_i)$ // update
9. **UNTIL**$(\forall r_i : d(r_i(t),\, r_i(t-1)) < \varepsilon \ \vee \ t > t_{\text{max}})$
10. **RETURN**$(\{r_1(t), \ldots, r_k(t)\})$

Bemerkungen:

- Die Cluster-Repräsentanten werden Centroide bzw. allgemein Medoide genannt.
- Die Funktion *choose*(V) realisiert eine zufällige Auswahl ohne Zurücklegen.
- Als Distanzfunktion d wird bei metrischen Daten meistens der euklidische Abstand zwischen zwei Punkten gewählt. Ein alternativer und allgemeiner Ansatz ist die Verwendung des kürzesten Weges in G.
- Als Minimierungskriterium e wird bei metrischen Daten meistens die Summe der quadrierten Abweichungen zum Cluster-Repräsentanten gewählt (Varianzkriterium). Sind die Punkte $v \in V$ Vektoren aus dem $\mathbf{R}^m$, lässt sich der optimale Cluster-Repräsentant durch komponentenweise Berechnung des arithmetischen Mittels ermitteln.

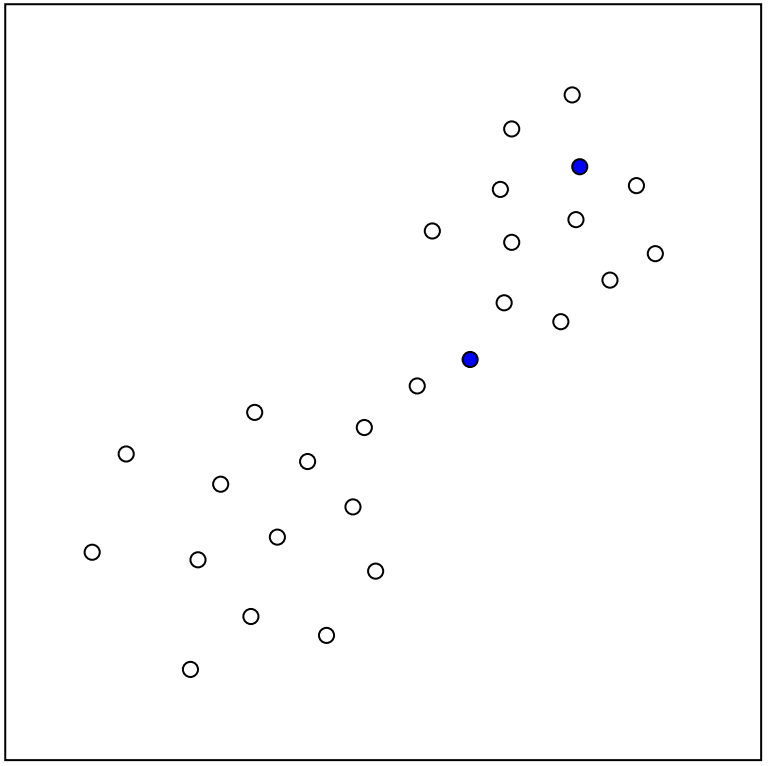
Iterative Verfahren

k-Means mit Minimierungskriterium e = Varianz



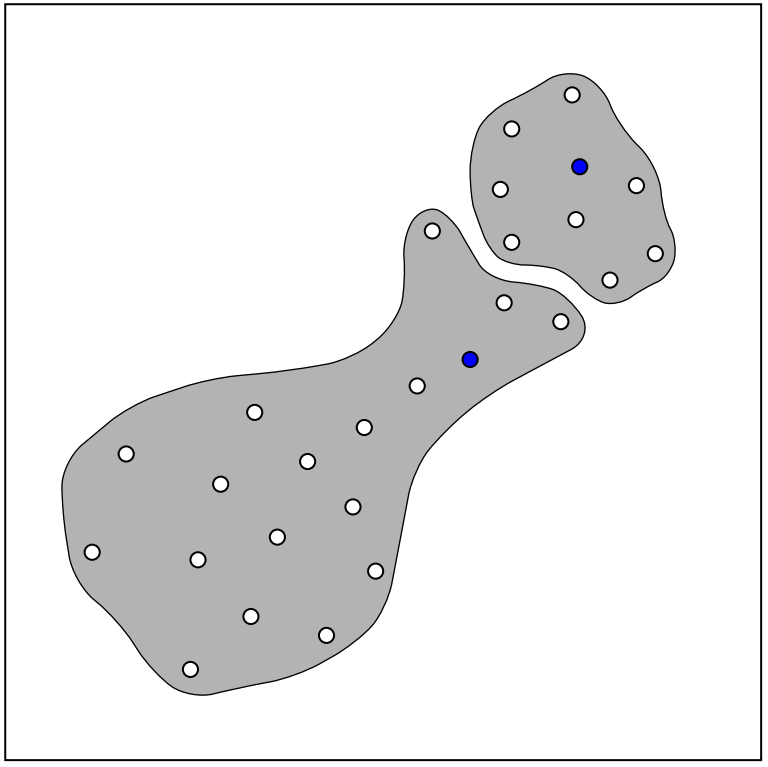
Iterative Verfahren

k-Means mit Minimierungskriterium e = Varianz



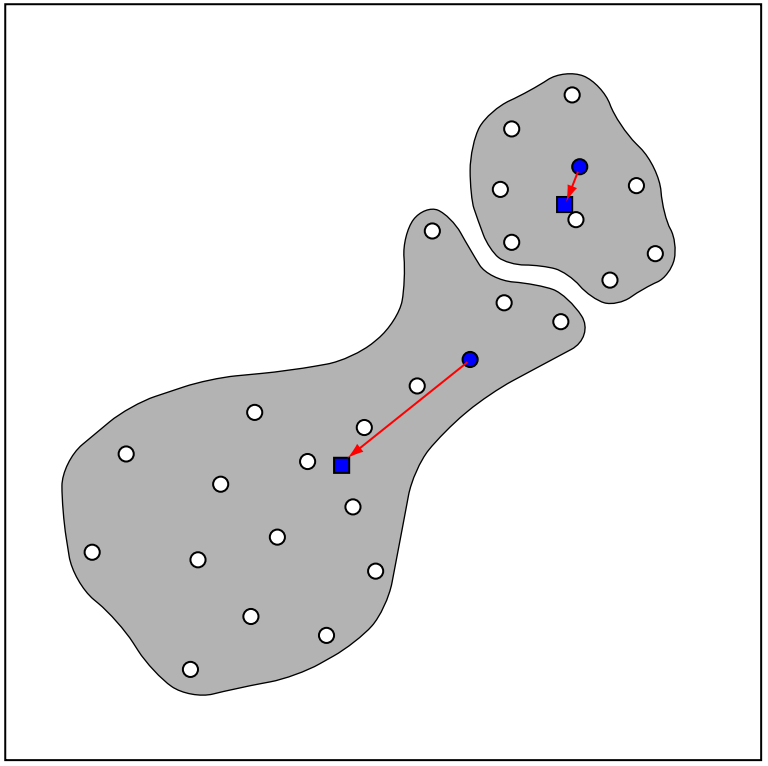
Iterative Verfahren

k-Means mit Minimierungskriterium e = Varianz



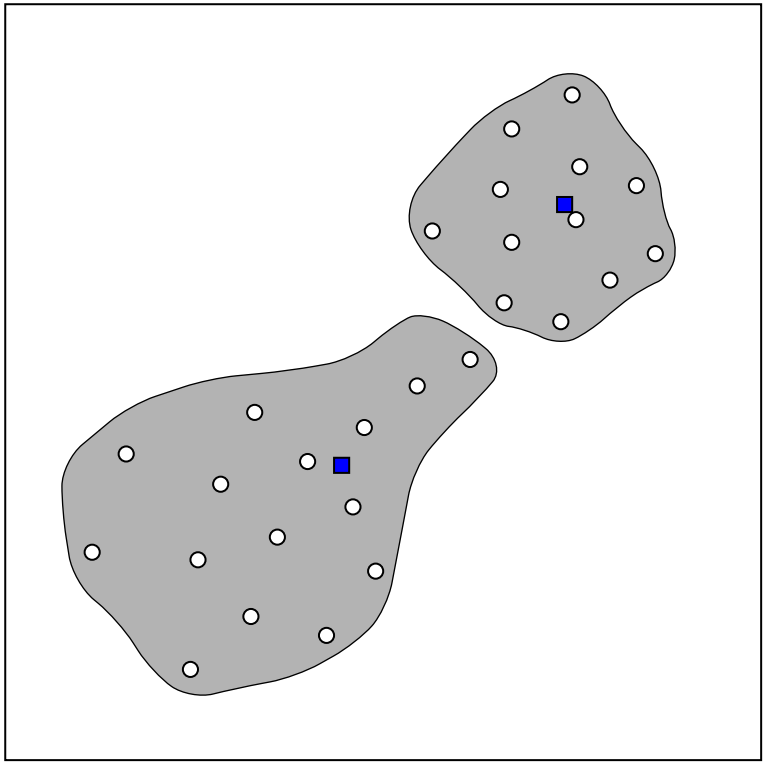
Iterative Verfahren

k-Means mit Minimierungskriterium e = Varianz



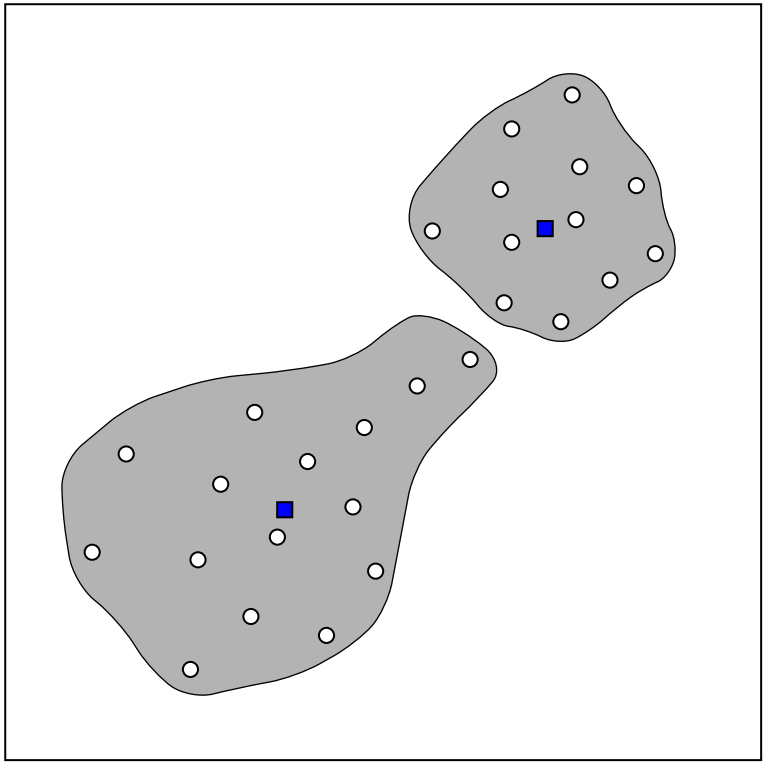
Iterative Verfahren

k-Means mit Minimierungskriterium e = Varianz



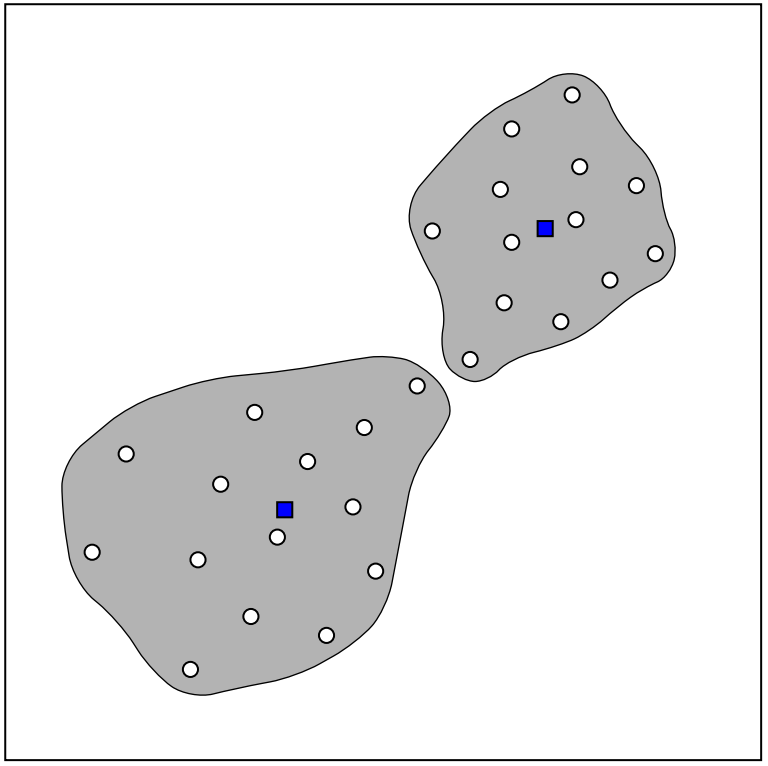
Iterative Verfahren

k-Means mit Minimierungskriterium e = Varianz



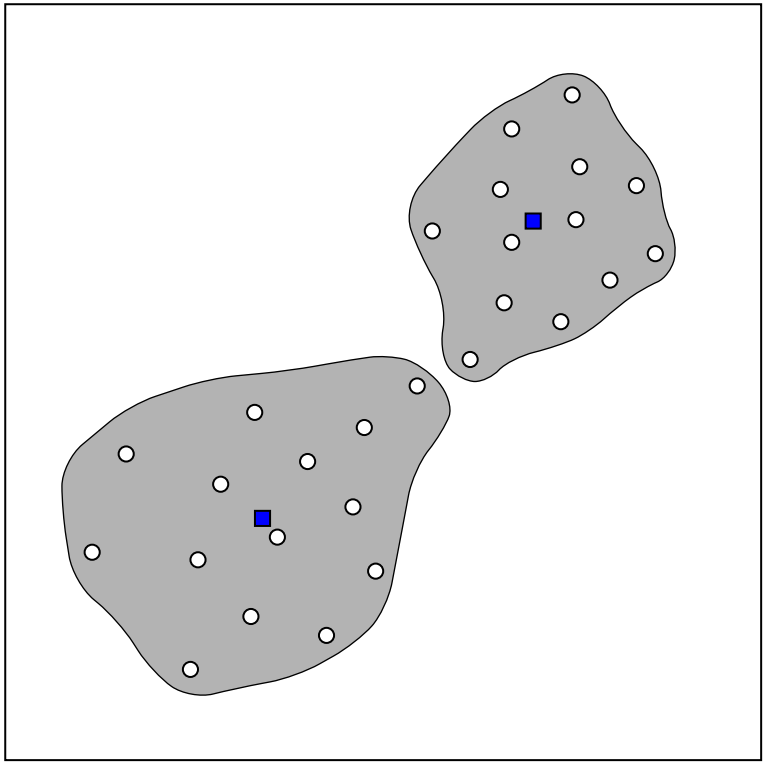
Iterative Verfahren

k-Means mit Minimierungskriterium e = Varianz



Iterative Verfahren

k-Means mit Minimierungskriterium e = Varianz



Iterative Verfahren

Einteilung exemplarbasierter Verfahren hinsichtlich e

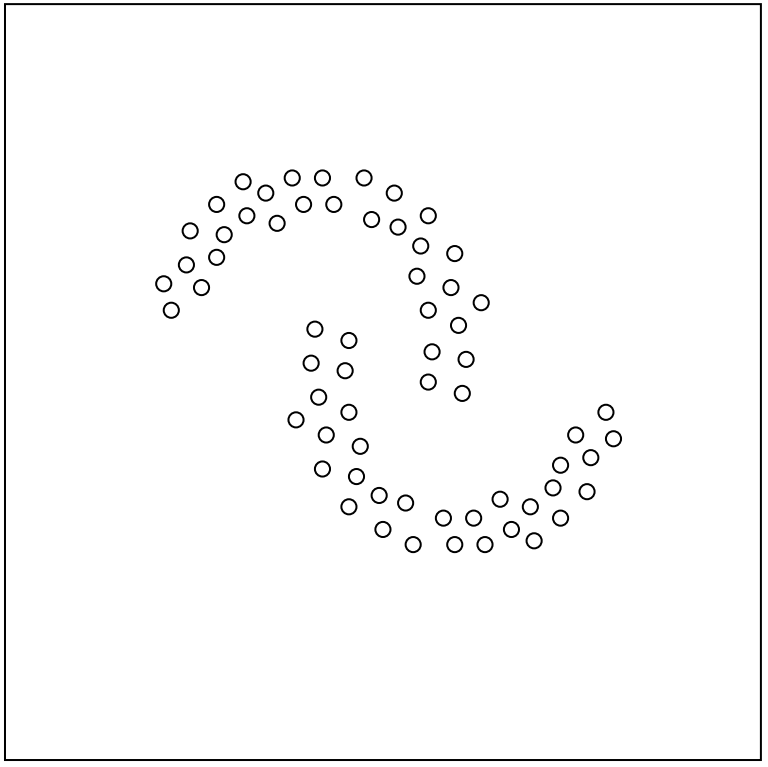
$e(\mathcal{C}) = \sum_{i=1}^{k} \sum_{v \in C_i} (v - r_i)^2$	$r_i = \bar{v}_{C_i}$	Centroid-Berechnung (k-Means)
$e(\mathcal{C}) = \sum_{i=1}^{k} \sum_{v \in C_i} \lvert v - r_i \rvert$	$r_i \in C_i$	Medoid-Berechnung (k-Medoid)
$e(\mathcal{C}) = \sum_{i=1}^{k} \max_{v \in C_i} \lvert v - r_i \rvert$	$r_i \in C_i$	k-Center
$e(\mathcal{C}) = \sum_{i=1}^{k} \sum_{v \in V} \mu_{v_i}^2 \cdot (v - r_i)^2$	$r_i = \frac{\sum_{v \in V} \mu_{v_i}^2 \cdot v}{\sum_{v \in V} \mu_{v_i}^2}$	Fuzzy-k-Means

Bemerkungen:

- $\bar{v}_{C_i}$ bezeichne den Mittelwert der Punkte $v \in C_i$.
- Der Cluster-Repräsentant ist vereinfachend mit r_i anstatt mit $r_i(t)$ bezeichnet.
- Die Centroid-Berechnung bei k-Means als Mittelwert der Cluster-Elemente entspricht einer lokalen, also Cluster-spezifischen Varianzminimierung.
- Der Medoid (Zentralelement) eines Cluster ist das Element, für das die Summe aller Distanzen zu diesem Element minimal ist. Ein Vorteil bei Verwendung von Medoiden ist das robustere Verhalten gegenüber Ausreißern und damit eventuell eine in der Anzahl der Iterationen schnellere Konvergenz.
- Bei Fuzzy-k-Means bezeichnet μ_{v_i} den Zugehörigkeitswert von $v \in V$ zu Repräsentant r_i.
- k-Medoid und k-Center arbeiten sowohl mit beliebigen Distanz- als auch mit Ähnlichkeitsmaßen.
- k-Means und Fuzzy-k-Means setzen intervallskalierte Merkmale voraus.
- k-Means kann unmittelbar als Kohonen Self-Organizing-Map (SOM), einem speziellen neuronalen Netz, umgesetzt werden:
  - Die Netztopologie besteht aus einer Eingangsschicht mit Knoten in der Anzahl der Merkmale und einer Verarbeitungsschicht, dem „competitive Layer“, mit k Knoten.
  - Der Lernalgorithmus bestimmt für einen Merkmalvektor auf Basis der Kantengewichte das „Winning Neuron“, dessen Gewichte gemäß des Lernparameters η erhöht werden.

Iterative Verfahren

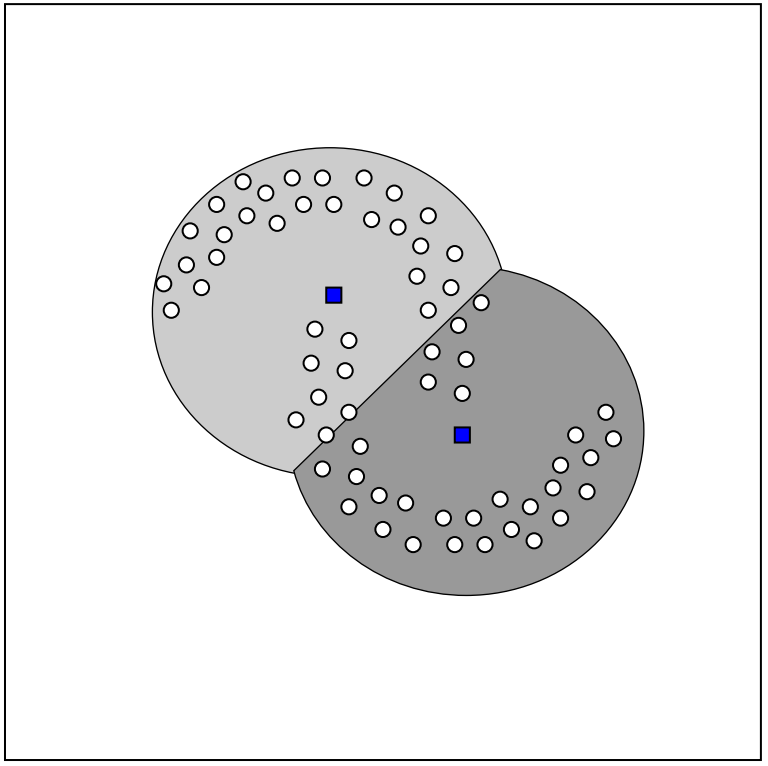
k-Means versus Single-Link



Exemplar-basierte Verfahren versagen bei verschränkt liegenden Clustern.

Iterative Verfahren

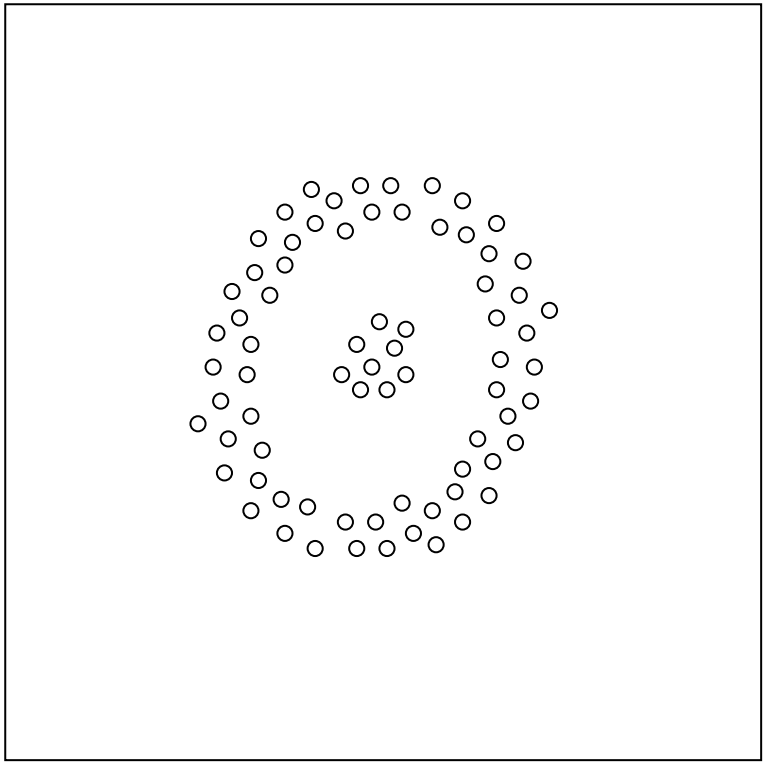
k-Means versus Single-Link



Exemplar-basierte Verfahren versagen bei verschränkt liegenden Clustern.

Iterative Verfahren

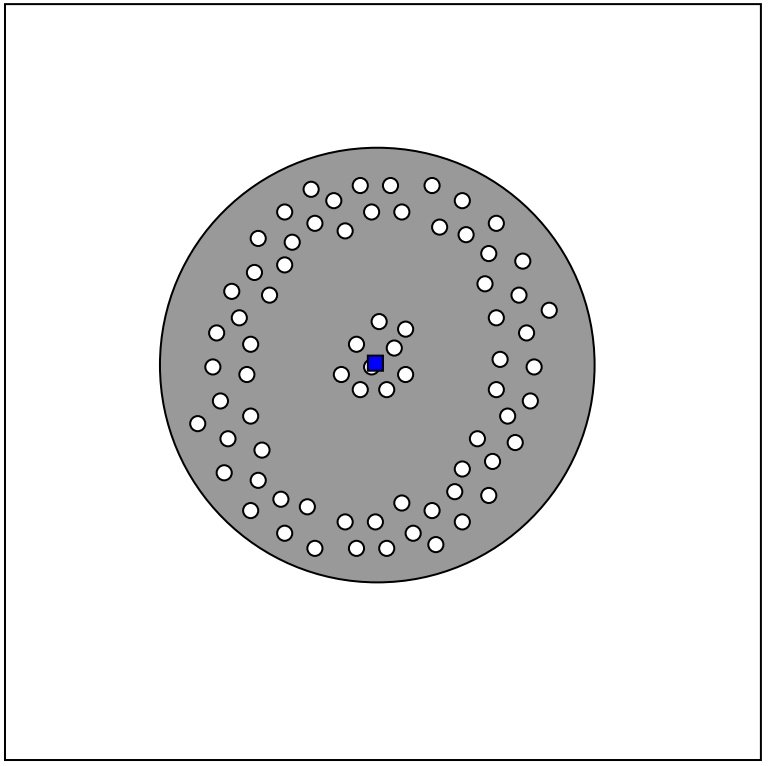
k-Means versus Single-Link



Exemplar-basierte Verfahren versagen bei verschränkt liegenden Clustern.

Iterative Verfahren

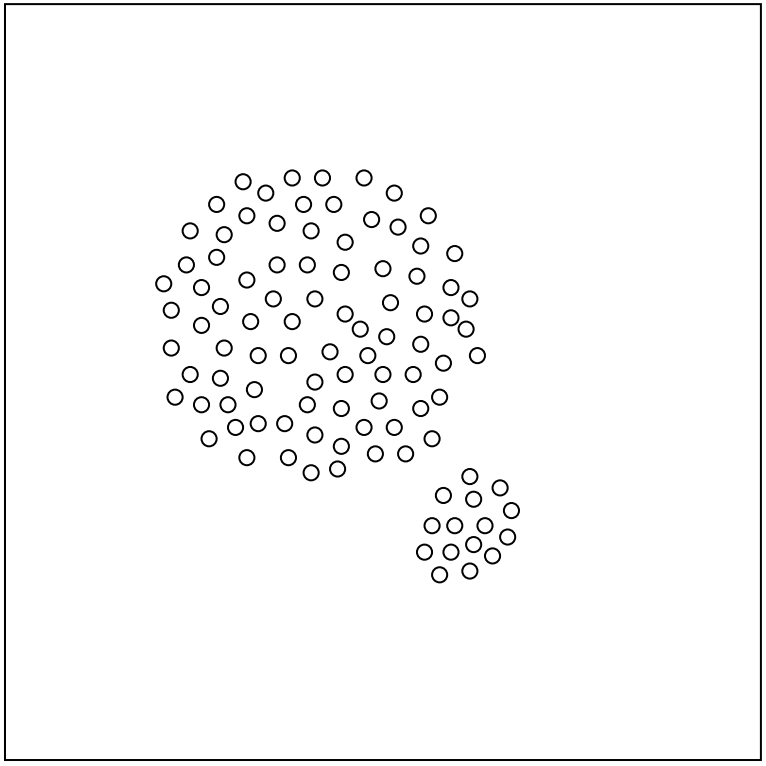
k-Means versus Single-Link



Exemplar-basierte Verfahren versagen bei verschränkt liegenden Clustern.

Iterative Verfahren

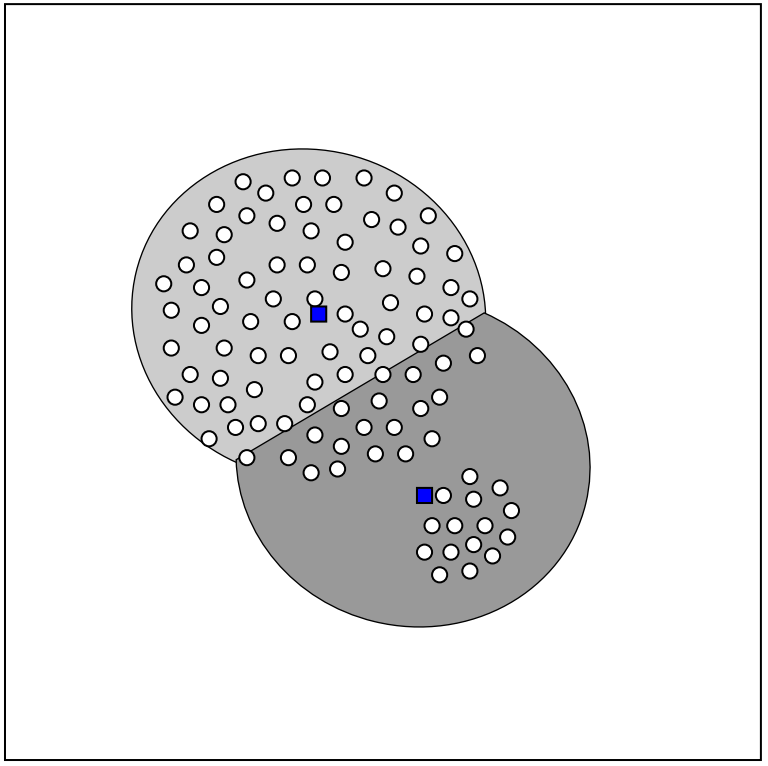
k-Means versus Single-Link



Exemplar-basierte Verfahren versagen bei extremen Größendifferenzen.

Iterative Verfahren

k-Means versus Single-Link



Exemplar-basierte Verfahren versagen bei extremen Größendifferenzen.

Iterative Verfahren

Exklusive versus nicht-exklusive Cluster-Analyse

Sei $\mathcal{C} = \{C_1, \ldots, C_k\}$ eine Partionierung einer Menge V mit $\bigcup_{i=1\ldots k} C_i = V$.

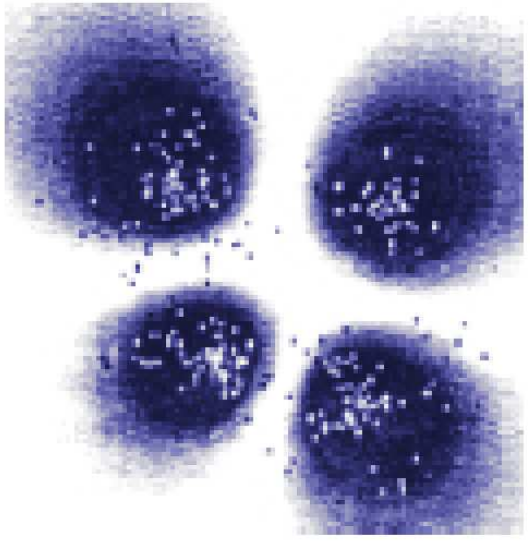
- exklusive Cluster-Analyse: $\forall_{i,j \in \{1,\ldots,k\}} : i \neq j \;\Rightarrow\; C_i \cap C_j = \emptyset$
- nicht-exklusive Cluster-Analyse erlaubt mehrfache Cluster-Zugehörigkeit

Iterative Verfahren

Exklusive versus nicht-exklusive Cluster-Analyse

Sei $\mathcal{C} = \{C_1, \ldots, C_k\}$ eine Partionierung einer Menge V mit $\bigcup_{i=1\ldots k} C_i = V$.

- exklusive Cluster-Analyse: $\forall_{i,j \in \{1,\ldots,k\}} : i \neq j \;\Rightarrow\; C_i \cap C_j = \emptyset$
- nicht-exklusive Cluster-Analyse erlaubt mehrfache Cluster-Zugehörigkeit
- Fuzzy-Cluster-Analyse quantifiziert die Cluster-Zugehörigkeit der $v \in V$ mit Zugehörigkeitsfunktionen $\mu_{v_i,\ i \in \{1,\ldots,k\}}$.

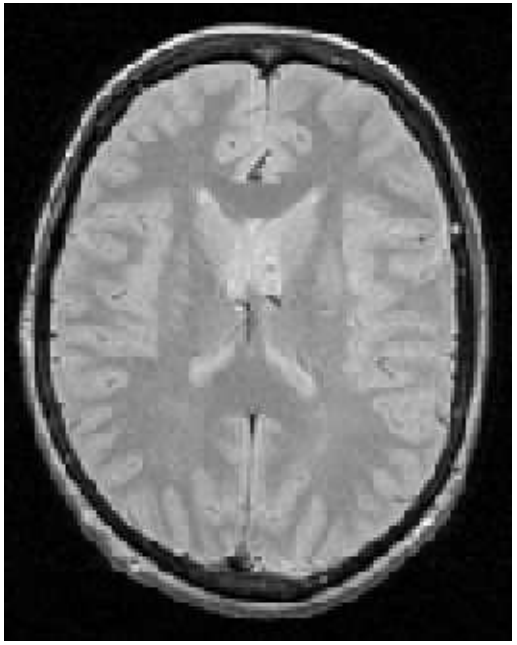


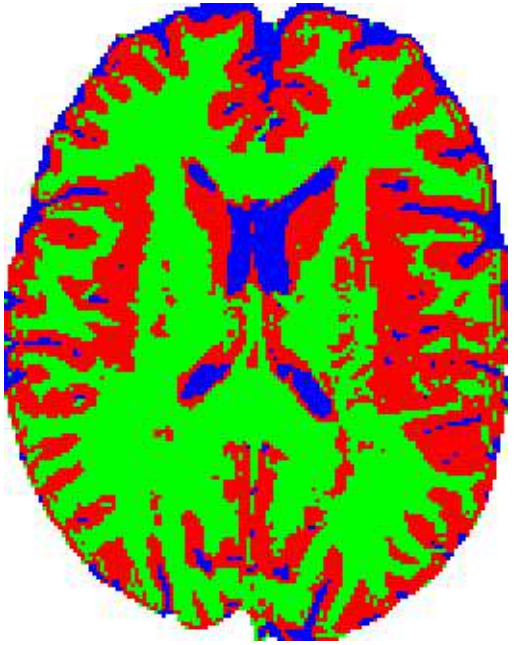
[Höppner/Klawonn/Kruse 1997]

Iterative Verfahren

Exklusive versus nicht-exklusive Cluster-Analyse

Anwendung der Fuzzy-Cluster-Analyse zur Darstellung von Gehirngewebe:





[Pham/Prince/Dagher/Xn 1996]

Bemerkungen:

- Die linguistische Variable bei der Fuzzy-Modellierung hat k Ausprägungen, entsprechend den Clustern $C_1, \ldots, C_k$.
- Üblich ist eine Normalisierungsrestriktion für Zugehörigkeitsfunktionen: $\sum_{i=1\ldots k} \mu_{v_i} = 1$
- Varianten von Fuzzy-k-Means ohne die Normalisierungsrestriktion haben den Nachteil, dass Punkte mit kleinen Zugehörigkeitswerten zu einem Cluster wie Ausreißer behandelt werden, anstatt den Cluster in ihre Richtung zu bewegen. Deshalb ist es sinnvoll, das Iterationsverfahren in einer Initialisierungsphase zunächst mit der Restriktion anzuwenden.
- Eine Klassifikation durch eine unscharfe Cluster-Analyse ist vorteilhaft, wenn keine klar ausgebildete Klassenstruktur vorliegt bzw., wenn sich nicht alle Vektoren eindeutig einer Klasse zuordnen lassen.
- Ein Nachteil der Fuzzy-Cluster-Analyse ist, dass keine Repräsentanten für Cluster ermittelt werden.

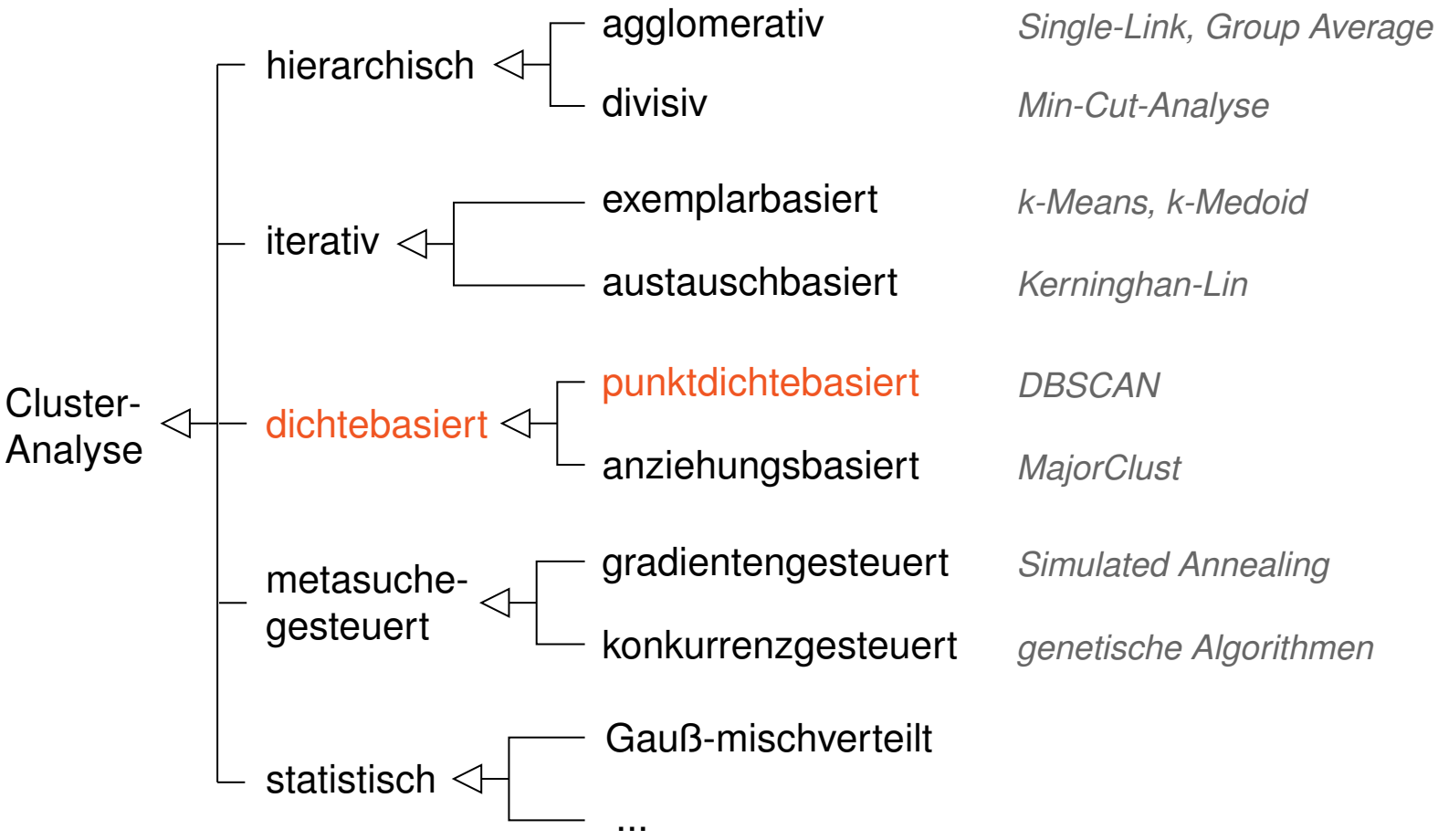
Kapitel ML: X (Fortsetzung)

X. Cluster-Analyse

- Einordnung Data Mining
- Einführung in die Cluster-Analyse
- Hierarchische Verfahren
- Iterative Verfahren
- Dichtebasierte Verfahren
- Cluster-Evaluierung

Dichtebasierte Verfahren

Prinzipien der Fusionierung



Dichtebasierte Verfahren

Dichtebasierte Verfahren versuchen, den Graphen $G = \langle V, E, w \rangle$ bzw. die Punktmenge V in Bereiche gleicher Dichte aufzuteilen.

Ansätze zur Dichteschätzung:

- parameterbasiert: die unterliegende Verteilung ist bekannt
- parameterlos: Histogramme (Konstruktion von Bar-Charts), Kerndichteschätzer (Überlagerung kontinuierlicher Funktionen)

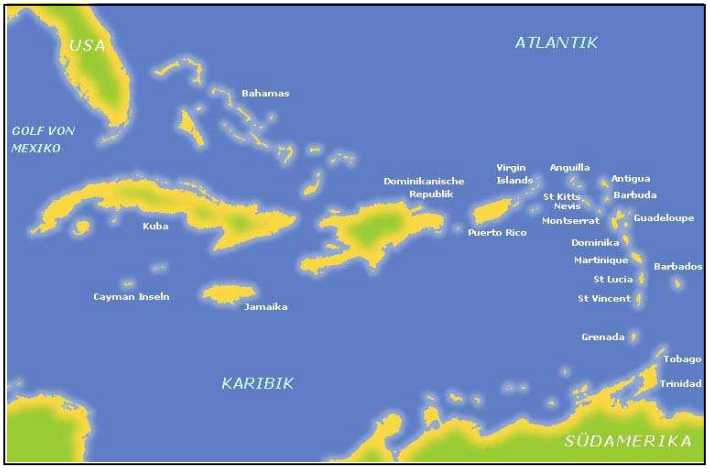
Dichtebasierte Verfahren

Dichtebasierte Verfahren versuchen, den Graphen $G = \langle V, E, w \rangle$ bzw. die Punktmenge V in Bereiche gleicher Dichte aufzuteilen.

Ansätze zur Dichteschätzung:

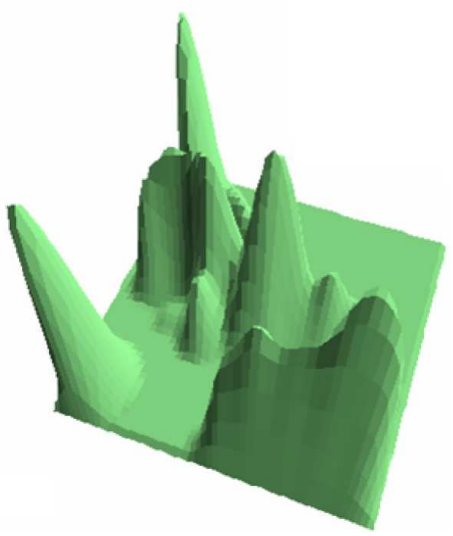
- parameterbasiert: die unterliegende Verteilung ist bekannt
- parameterlos: Histogramme (Konstruktion von Bar-Charts), Kerndichteschätzer (Überlagerung kontinuierlicher Funktionen)

Beispiel (karibische Inseln):



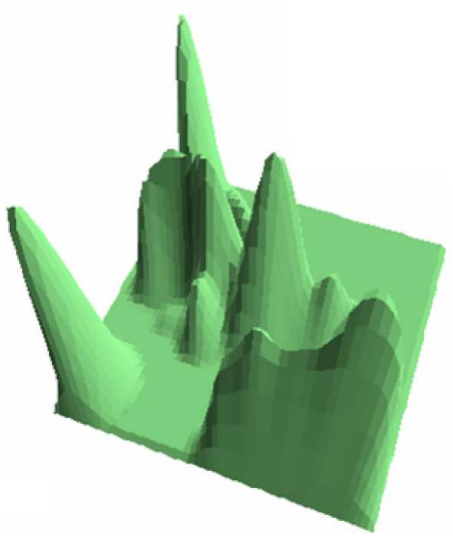
Dichtebasierte Verfahren

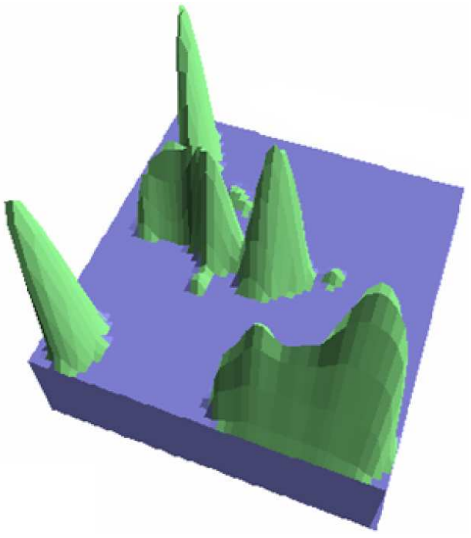
Dichteschätzung mit Gauß'schem Kern für das Beispiel



Dichtebasierte Verfahren

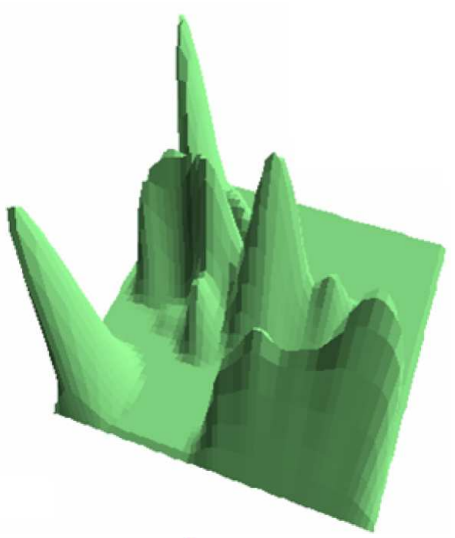
Dichteschätzung mit Gauß'schem Kern für das Beispiel

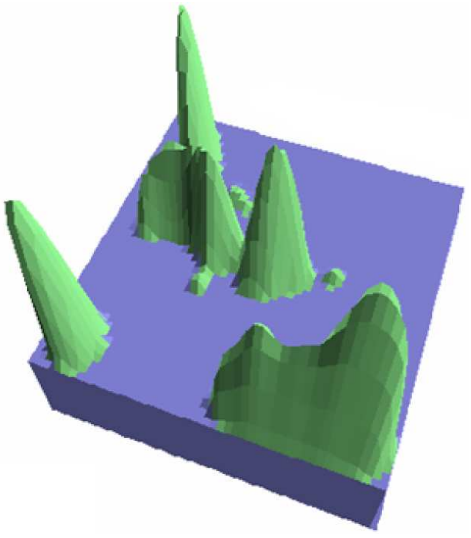


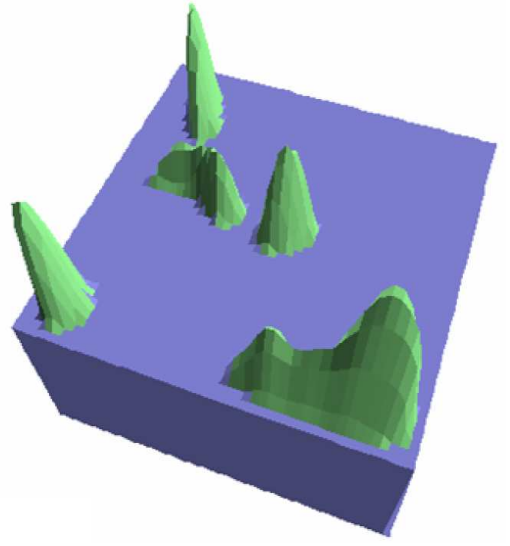


Dichtebasierte Verfahren

Dichteschätzung mit Gauß'schem Kern für das Beispiel

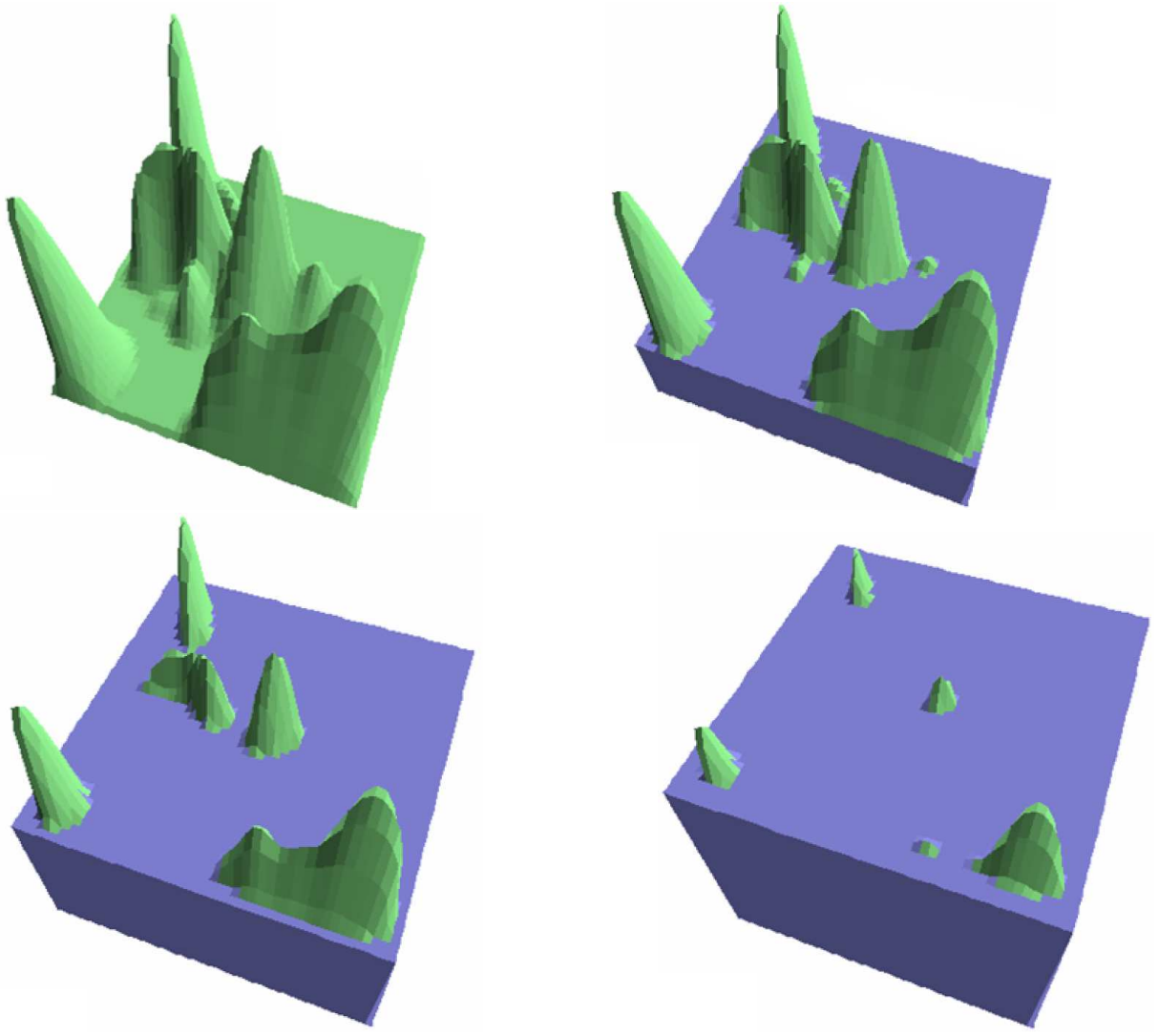






Dichtebasierte Verfahren

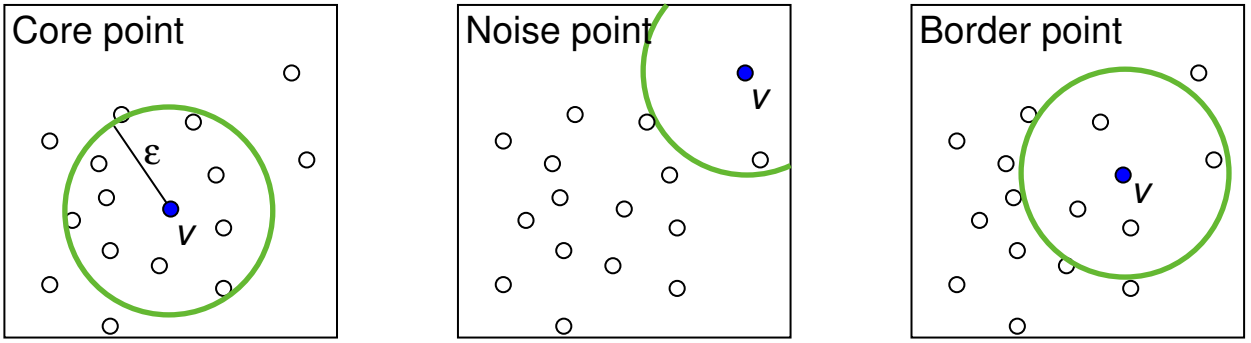
Dichteschätzung mit Gauß'schem Kern für das Beispiel



Dichtebasierte Verfahren

DBSCAN: Prinzip der Dichteschätzung [Ester et al. 1996]

Sei $|N_\varepsilon(v)|$ die ε-Nachbarschaft eine Punktes v. Unterscheidung von drei Punkttypen:



1. v ist Kernpunkt (core point) $\Leftrightarrow$ $|N_\varepsilon(v)| \geq$ *MinPts*

2. v ist Rauschen (noise point) $\Leftrightarrow$
   v ist von keinem Kernpunkt aus dichteerreichbar (density-reachable)

3. v ist Randpunkt (border point) in allen anderen Fällen

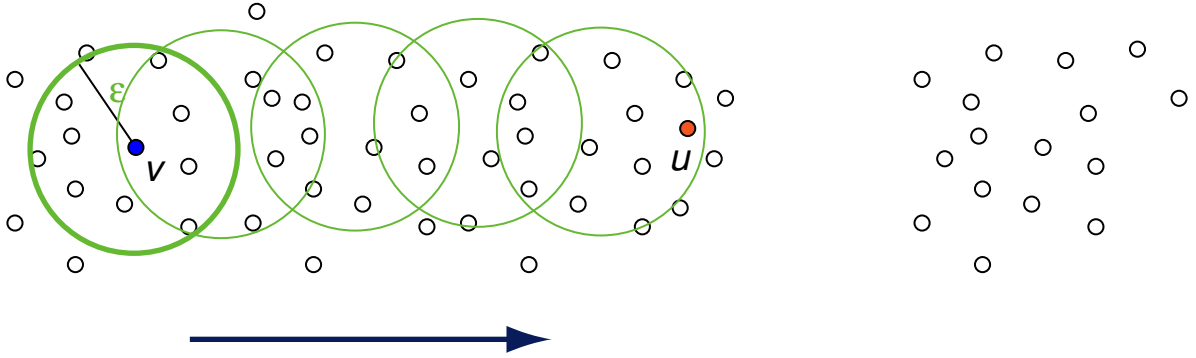
Dichtebasierte Verfahren

DBSCAN: Prinzip der Dichteschätzung

Ein Punkt u ist dichteerreichbar von einem Punkt v, falls gilt:

(a) $u \in |N_\varepsilon(v)|$, wobei v ein Kernpunkt ist, oder

(b) es gibt eine Menge von Punkten $\{v_1, \ldots, v_l\}$:
$v_{i+1} \in |N_\varepsilon(v_i)|$ und v_i ist Kernpunkt, $i = 1, \ldots, l\text{-}1$, mit $v_1 = v$, $v_l = u$.



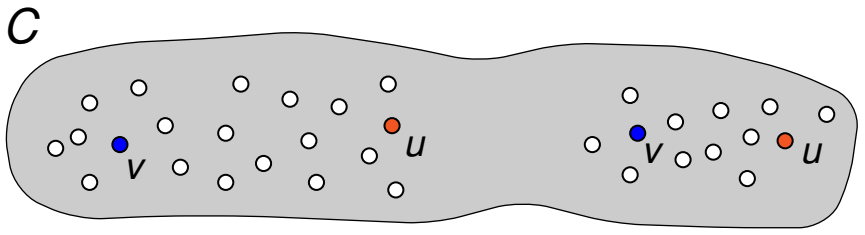
Bedingung (b) lässt sich als transitive Anwendung von Bedingung (a) auffassen.

Dichtebasierte Verfahren

DBSCAN: Cluster-Interpretation

Ein Cluster $C \subseteq V$ erfüllt folgende Bedingungen:

1. $\forall u, v$: Falls $v \in C$ und u dichteerreichbar von v, dann ist $u \in C$.



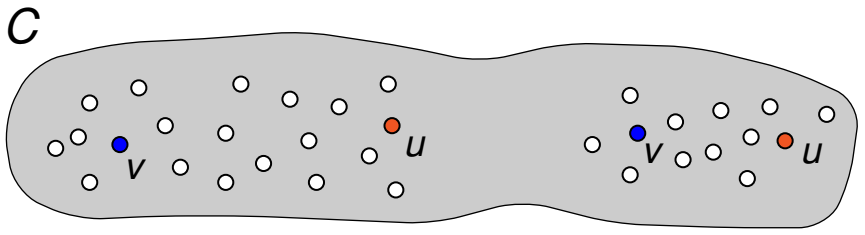
Maximalitäts-
bedingung

Dichtebasierte Verfahren

DBSCAN: Cluster-Interpretation

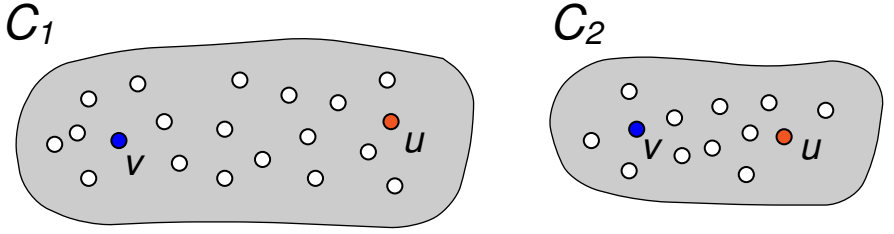
Ein Cluster $C \subseteq V$ erfüllt folgende Bedingungen:

1. $\forall u, v$: Falls $v \in C$ und u dichteerreichbar von v, dann ist $u \in C$.



Maximalitäts-
bedingung

2. $\forall u, v$: u ist dichteverbunden mit v, d. h.,
   es existiert ein Punkt t von dem u und v dichteerreichbar sind.



Zusammenhangs-
bedingung

Dichtebasierte Verfahren

DBSCAN: Algorithmus

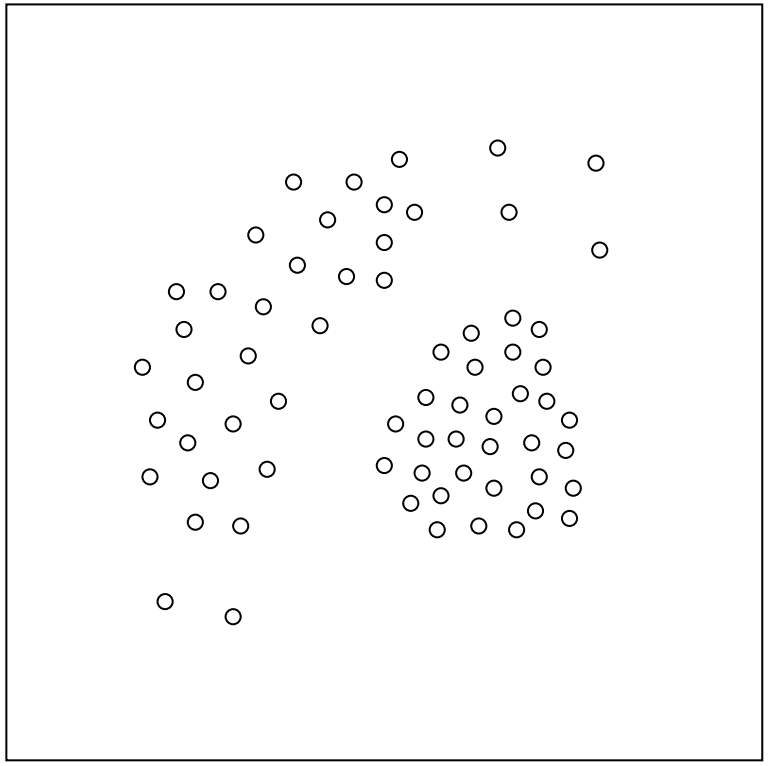
Input: $G = \langle V, E, w \rangle$. Weighted graph.
d. Distance function for nodes in V.
ε. Neigborhood radius.
MinPts. Lower bound for point number in ε-neigborhood.

Output: $\gamma : V \to \mathbf{Z}$. Cluster assignment function.

1. $i = 0$
2. **WHILE** $\exists v : (v \in V \ \wedge \ \gamma(v) = \perp)$ **DO** // $\perp$ = undefined, unclassified
3. $v =$ *choose_unclassified_point*(V)
4. $N_\varepsilon(v) =$ *neighborhood*(G, d, v, ε)
5. **IF** $|N_\varepsilon(v)| \geq$ *MinPts* **THEN** // v is core point
6. $i = i + 1$
7. $C_i =$ *density_reachable_hull*$(G, d, N_\varepsilon(v))$ // form a new cluster
8. **FOREACH** $v \in C_i$ **DO** $\gamma(v) = i$
9. **ELSE** $\gamma(v) = -1$ // v is _preliminarily_ classified as noise
10. **ENDDO**
11. **RETURN**(γ)

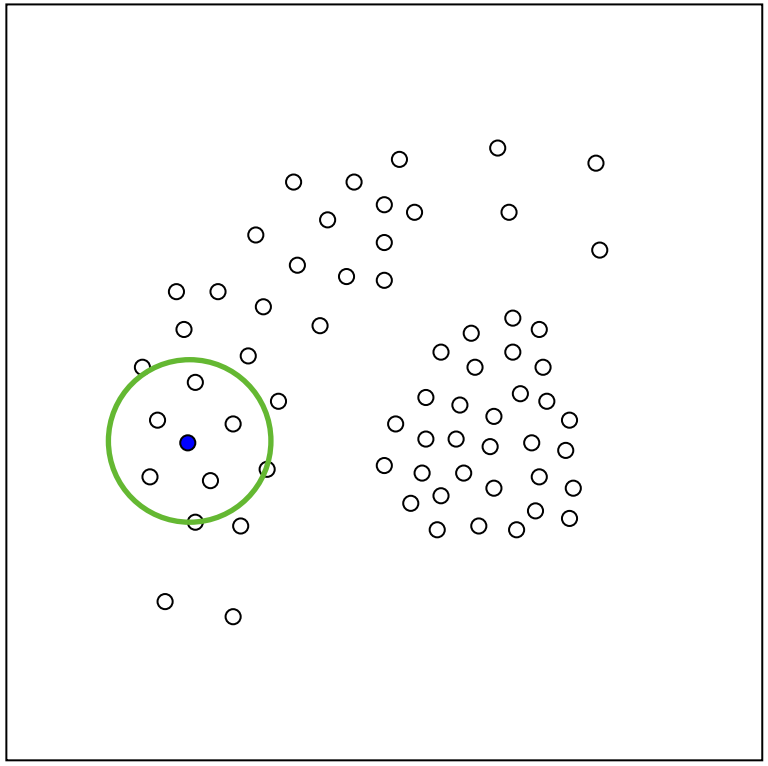
Dichtebasierte Verfahren

DBSCAN



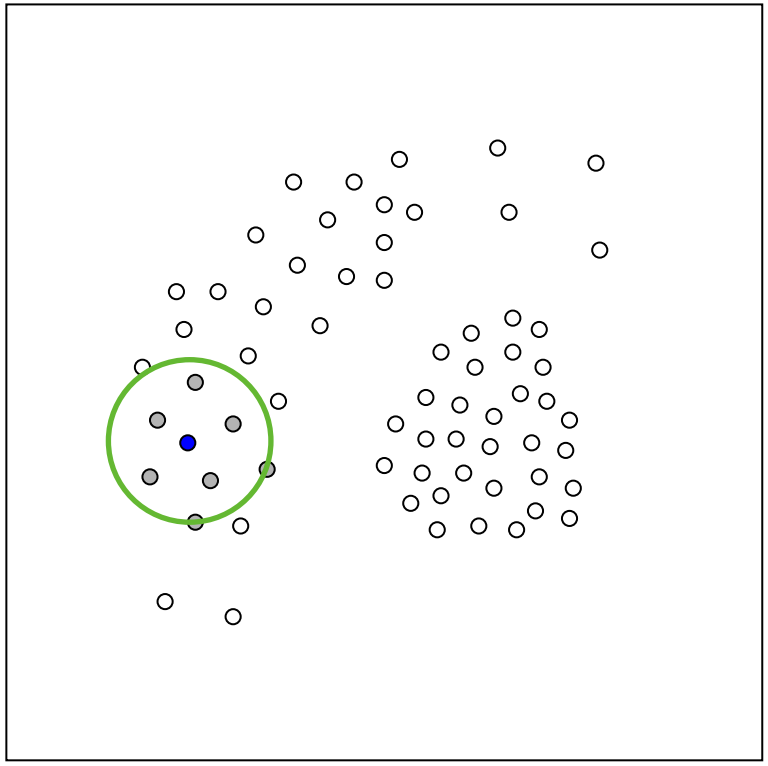
Dichtebasierte Verfahren

DBSCAN



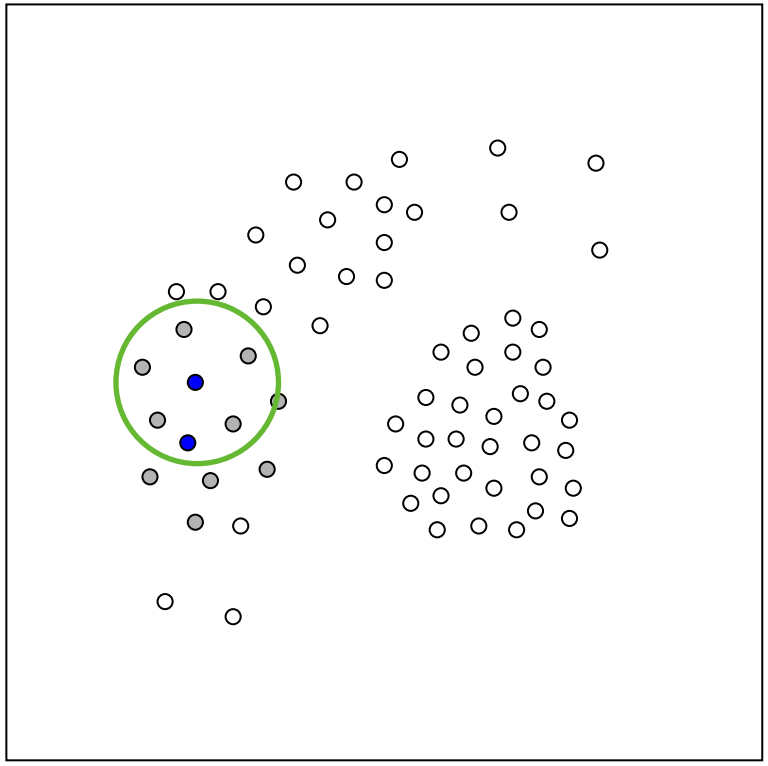
Dichtebasierte Verfahren

DBSCAN



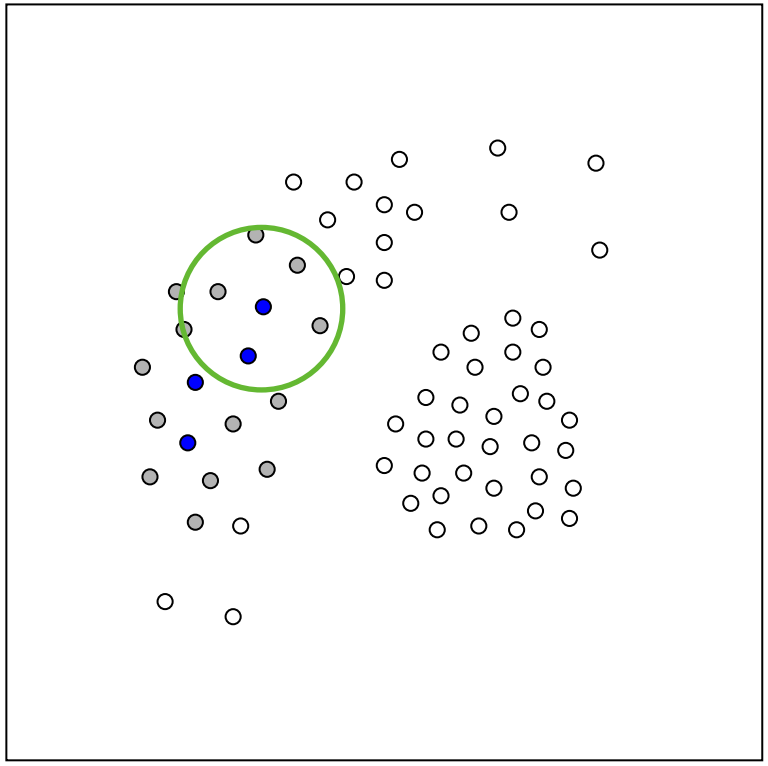
Dichtebasierte Verfahren

DBSCAN



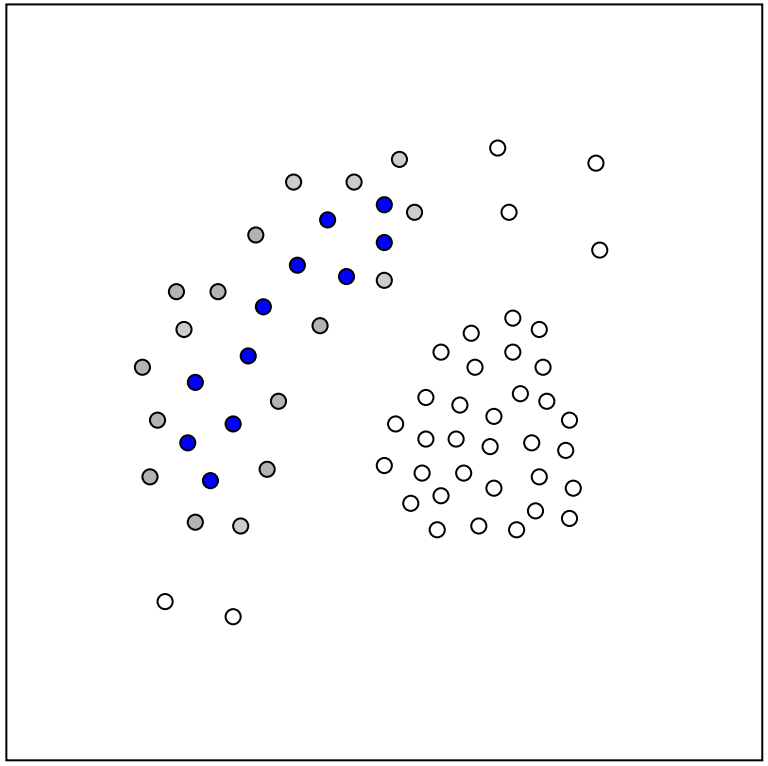
Dichtebasierte Verfahren

DBSCAN



Dichtebasierte Verfahren

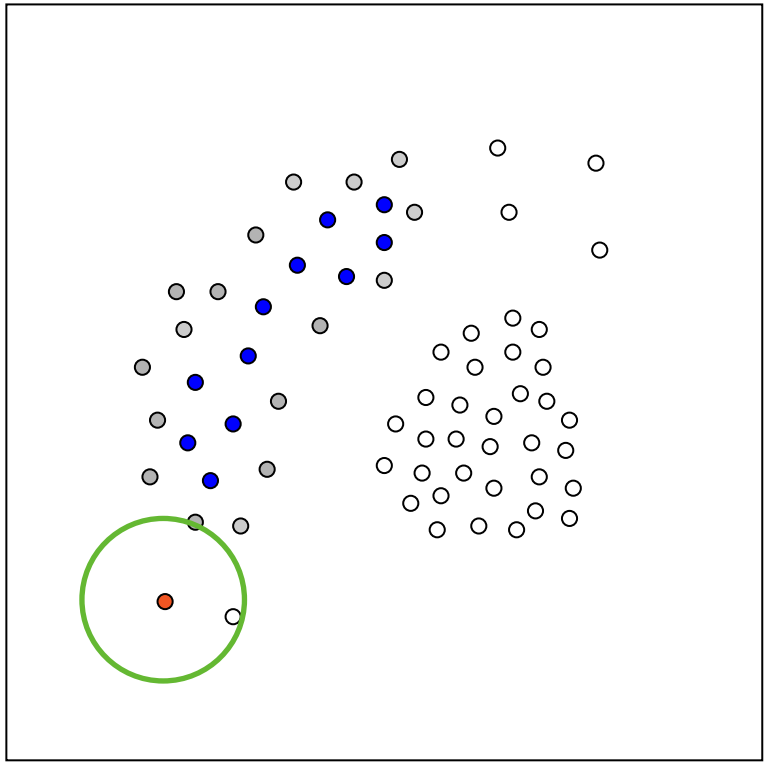
DBSCAN



- Core point
- Border point

Dichtebasierte Verfahren

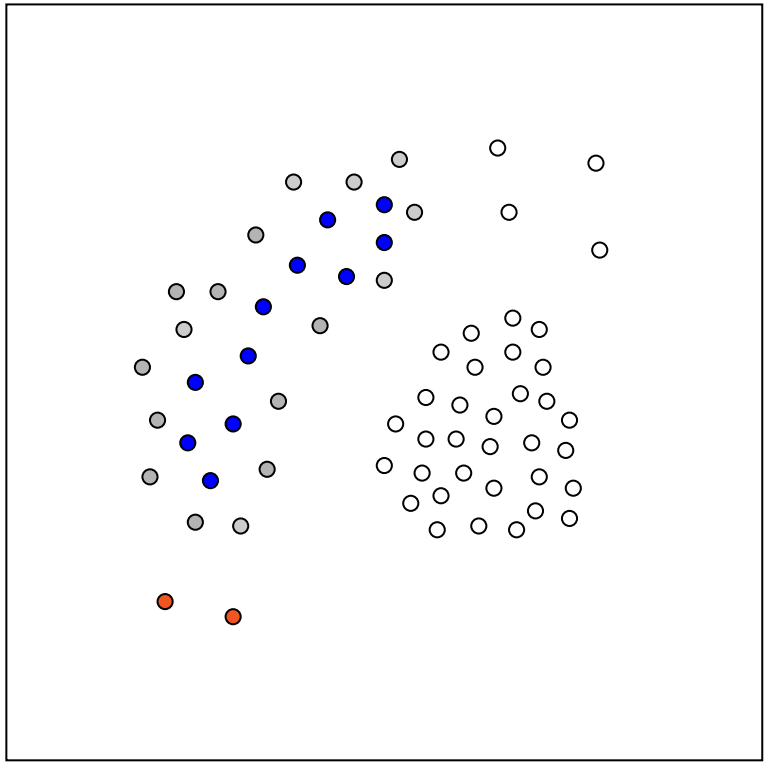
DBSCAN



- Core point
- Border point

Dichtebasierte Verfahren

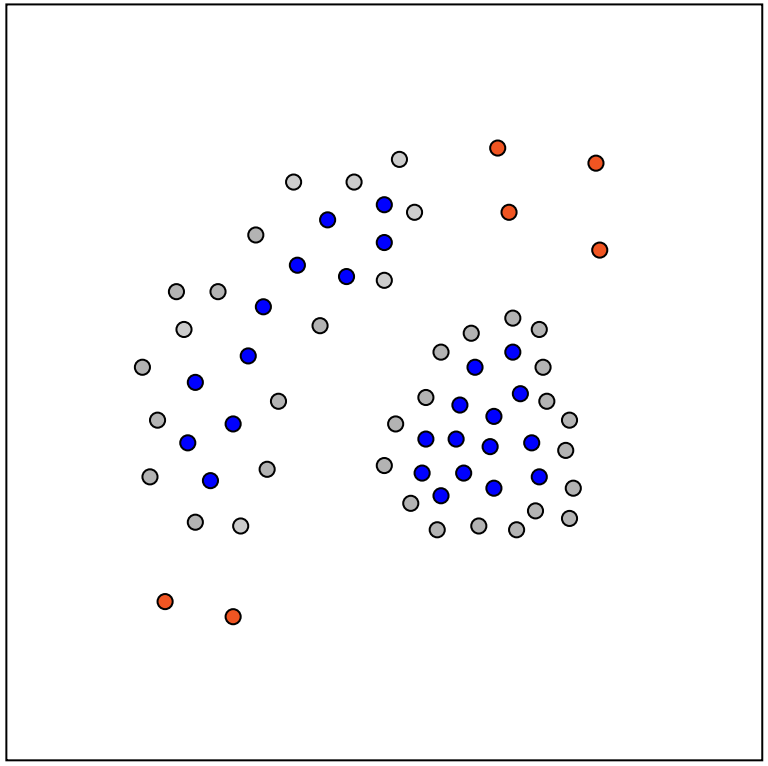
DBSCAN



- Core point
- Border point
- Noise point

Dichtebasierte Verfahren

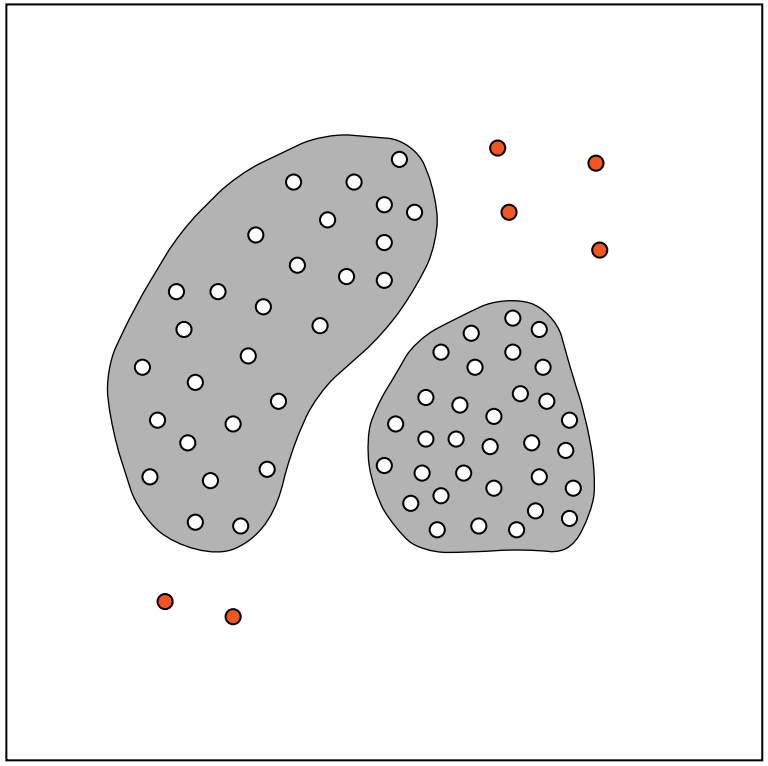
DBSCAN



- Core point
- Border point
- Noise point

Dichtebasierte Verfahren

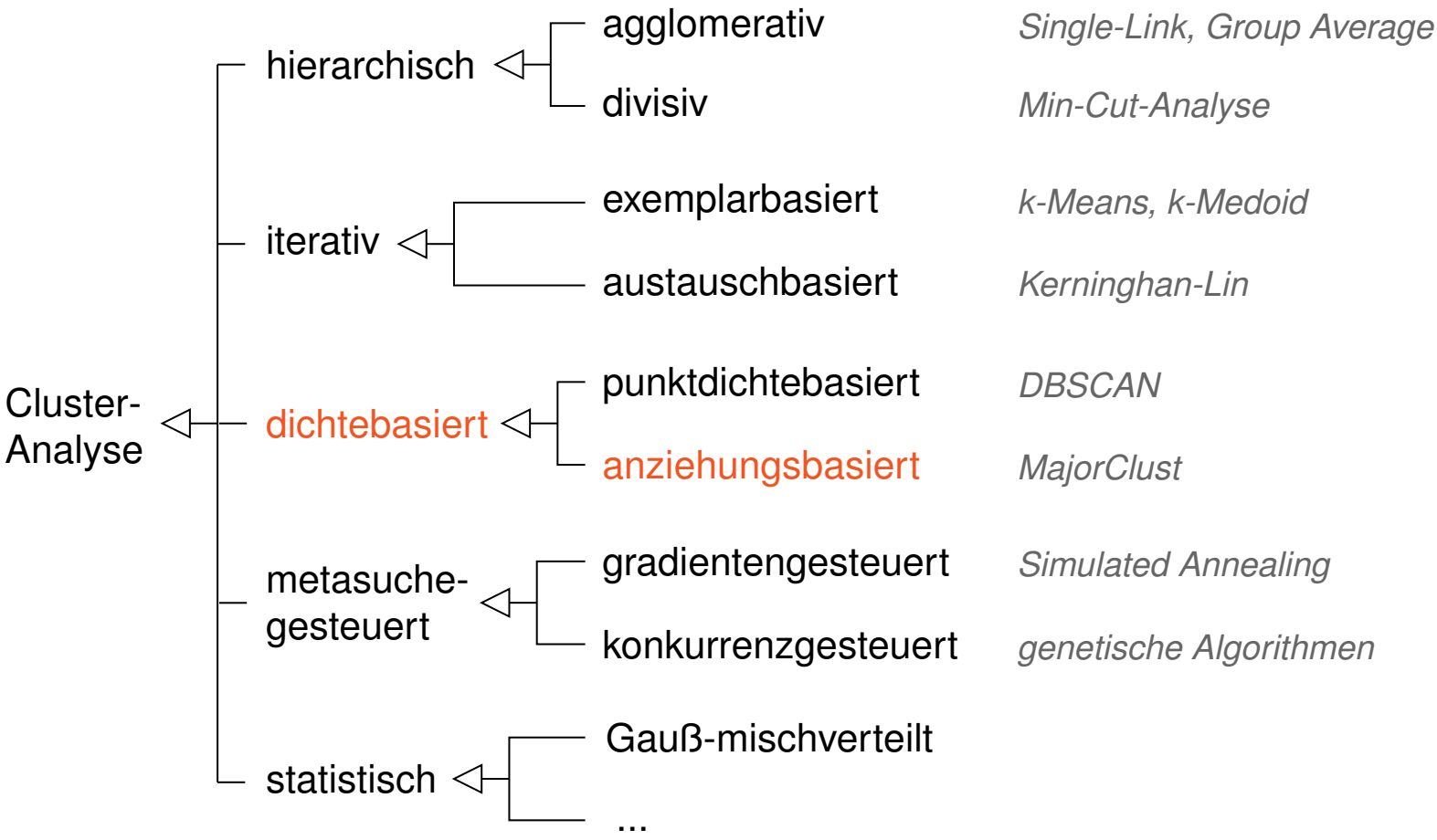
DBSCAN



- Noise point

Dichtebasierte Verfahren

Prinzipien der Fusionierung

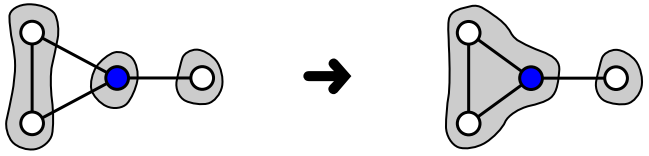


Dichtebasierte Verfahren

MajorClust: Prinzip der Dichteschätzung [Stein/Niggemann 1999]

Die gewichteten Kanten im Graph $G = \langle V, E, w \rangle$ werden als Kräfte interpretiert; Knoten desselben Clusters bündeln ihrer Kräfte. Illustration:

eindeutige Zugehörigkeitsentscheidung (mit Agglomeration):

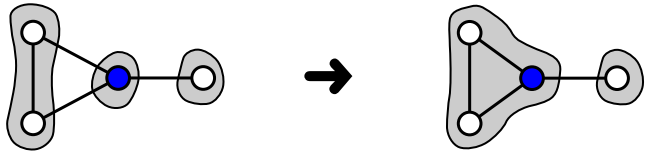


Dichtebasierte Verfahren

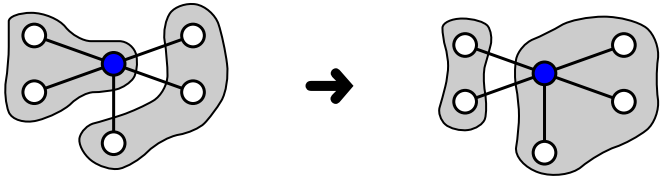
MajorClust: Prinzip der Dichteschätzung [Stein/Niggemann 1999]

Die gewichteten Kanten im Graph $G = \langle V, E, w \rangle$ werden als Kräfte interpretiert; Knoten desselben Clusters bündeln ihrer Kräfte. Illustration:

eindeutige Zugehörigkeitsentscheidung (mit Agglomeration):



eindeutige Zugehörigkeitsentscheidung
(mit Cluster-Wechsel):

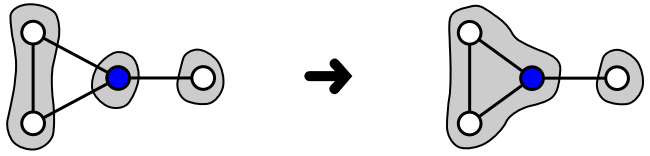


Dichtebasierte Verfahren

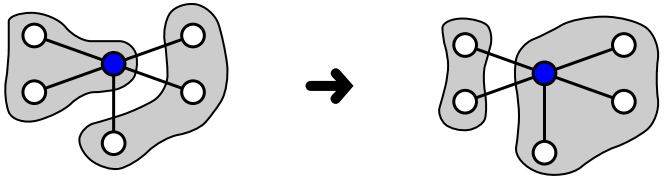
MajorClust: Prinzip der Dichteschätzung [Stein/Niggemann 1999]

Die gewichteten Kanten im Graph $G = \langle V, E, w \rangle$ werden als Kräfte interpretiert; Knoten desselben Clusters bündeln ihrer Kräfte. Illustration:

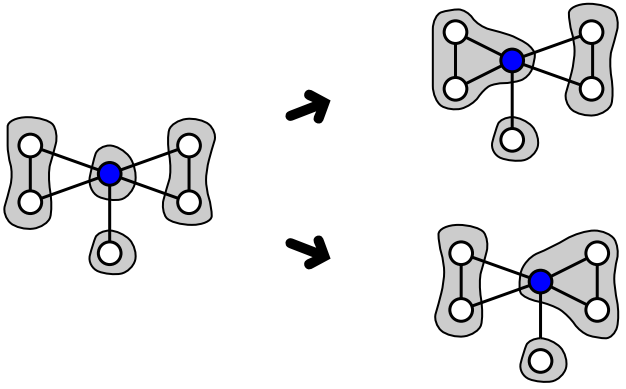
eindeutige Zugehörigkeitsentscheidung (mit Agglomeration):



eindeutige Zugehörigkeitsentscheidung (mit Cluster-Wechsel):



mehrdeutige Zugehörigkeit:



Dichtebasierte Verfahren

MajorClust: Algorithmus

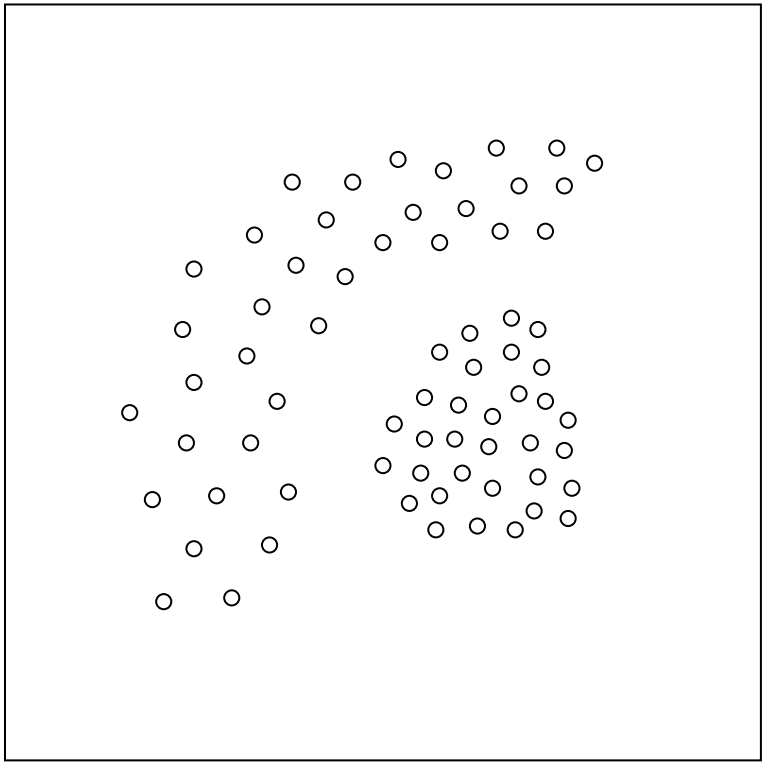
Input: $G = \langle V, E, w \rangle$. Weighted graph.
d. Distance function for nodes in V.

Output: $\gamma : V \to \mathbf{N}$. Cluster assignment function.

1. $i = 0$, $t =$ *False*
2. **FOREACH** $v \in V$ **DO** $i = i + 1$, $\gamma(v) = i$ **ENDDO**
3. **UNLESS** t **DO**
4. $t =$ *True*
5. **FOREACH** $v \in V$ **DO**
6. $\gamma^* = \underset{i:\, i \in \{1,\ldots,|V|\}}{\text{argmax}} \sum_{\{u,v\}:\{u,v\} \in E \wedge \gamma(u) = i} w(u, v)$
7. **IF** $\gamma(v) \neq \gamma^*$ **THEN** $\gamma(v) = \gamma^*$, $t =$ *False* **ENDIF** // relabeling
8. **ENDDO**
9. **ENDDO**
10. **RETURN**(γ)

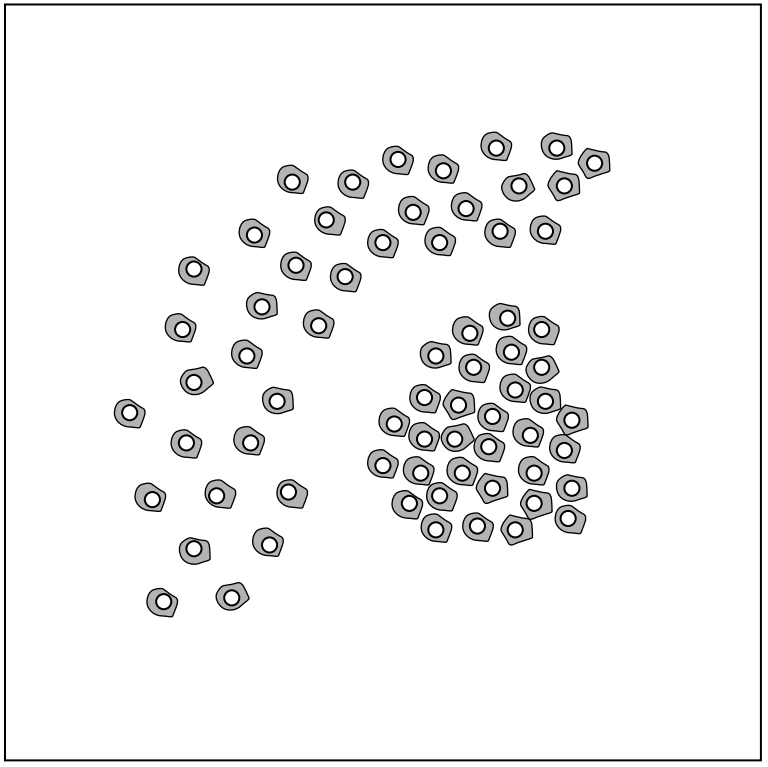
Dichtebasierte Verfahren

MajorClust



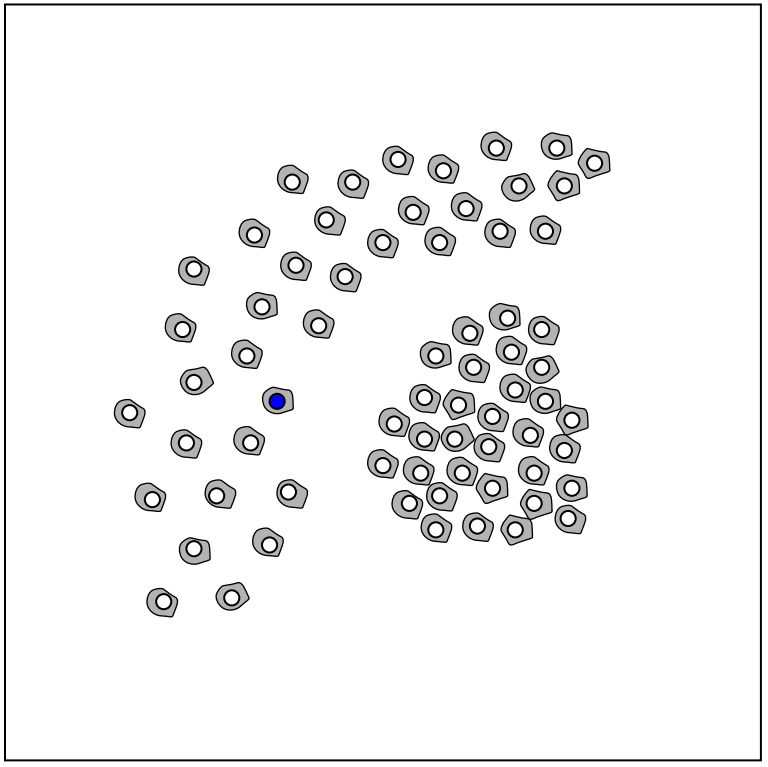
Dichtebasierte Verfahren

MajorClust



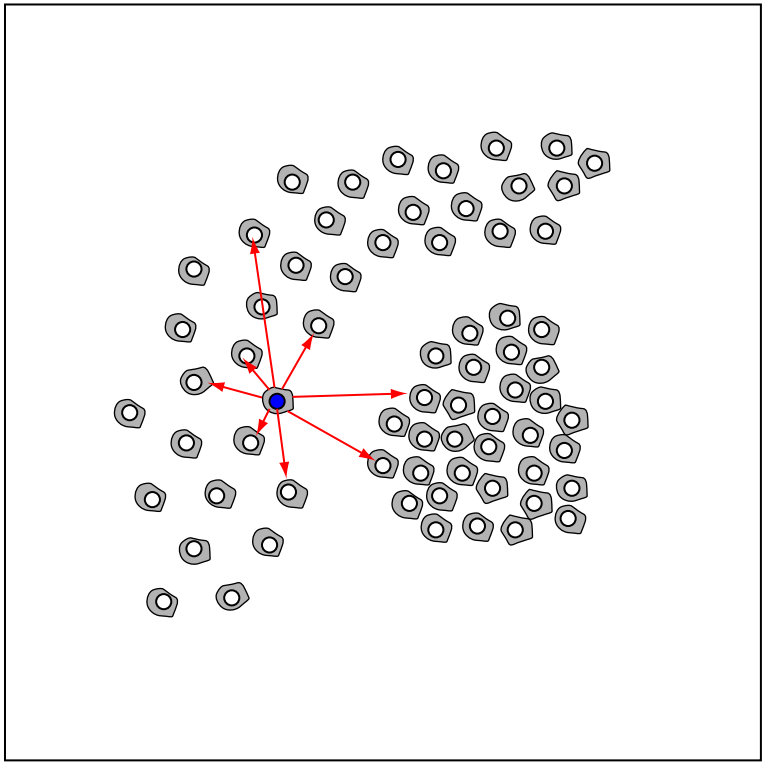
Dichtebasierte Verfahren

MajorClust



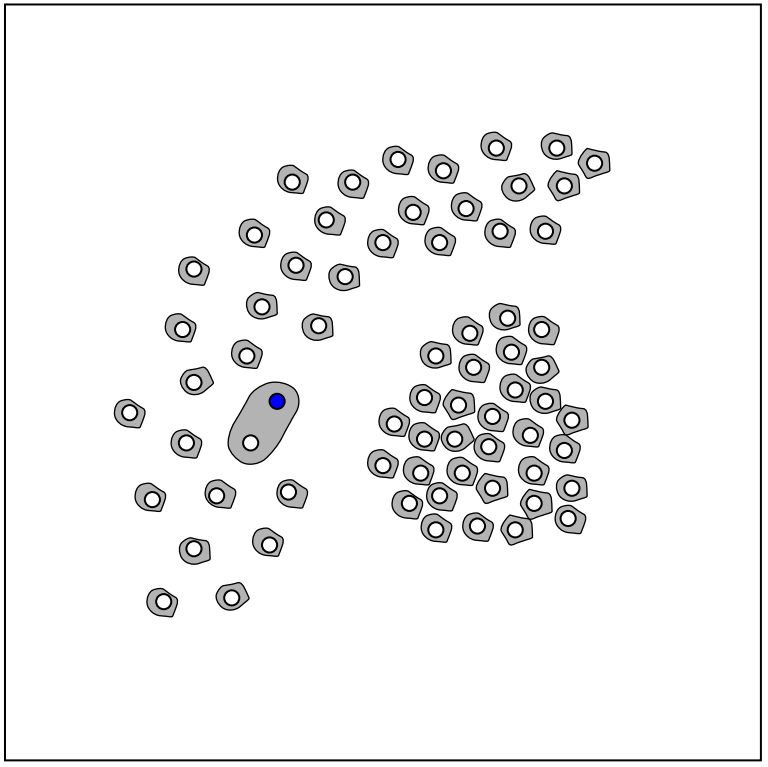
Dichtebasierte Verfahren

MajorClust



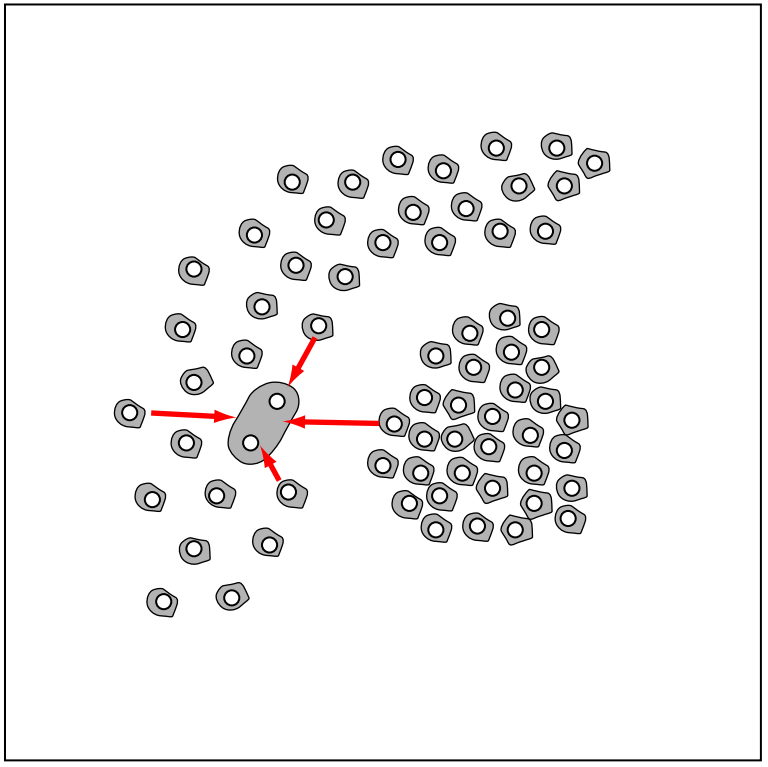
Dichtebasierte Verfahren

MajorClust



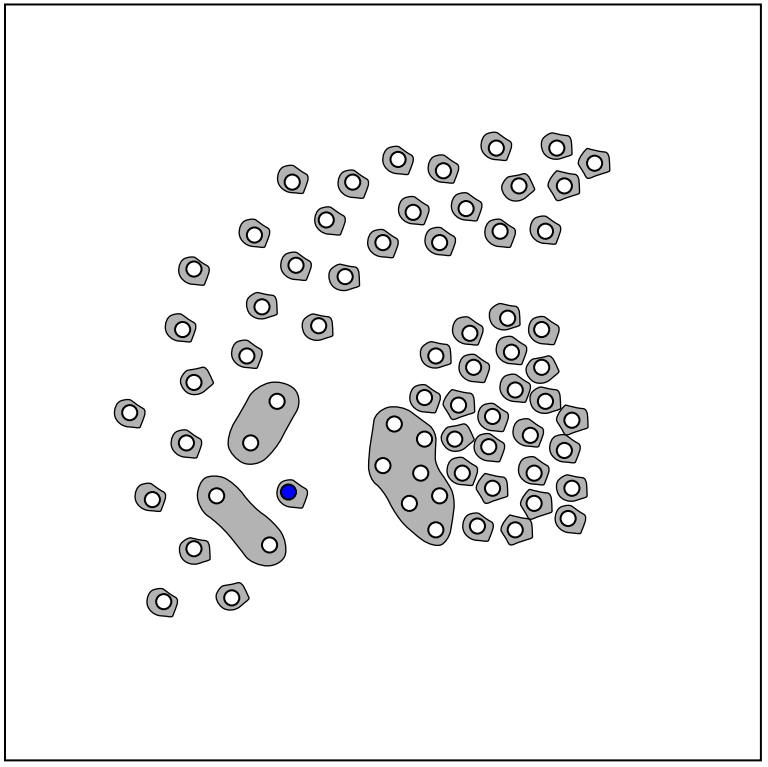
Dichtebasierte Verfahren

MajorClust



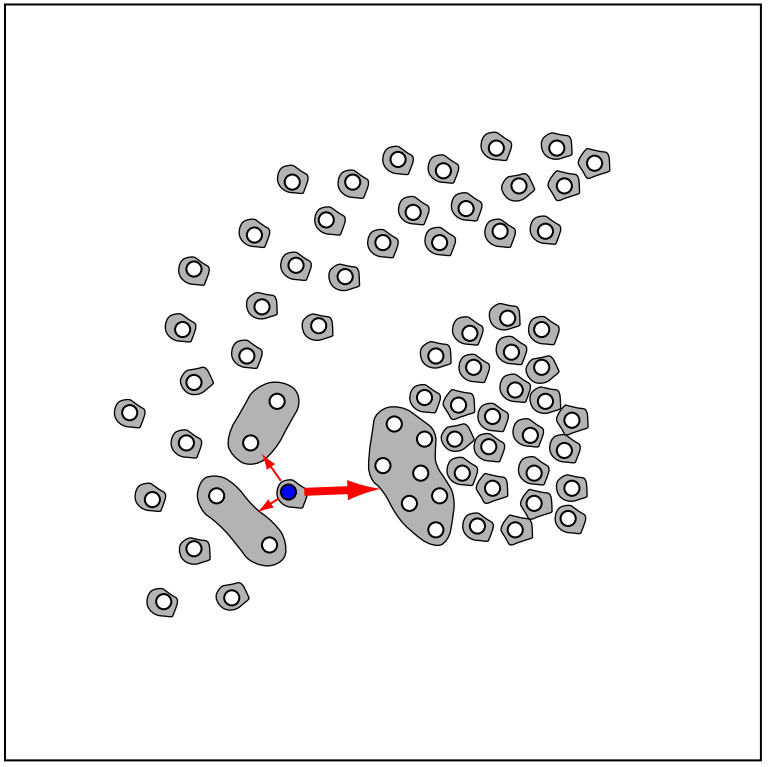
Dichtebasierte Verfahren

MajorClust



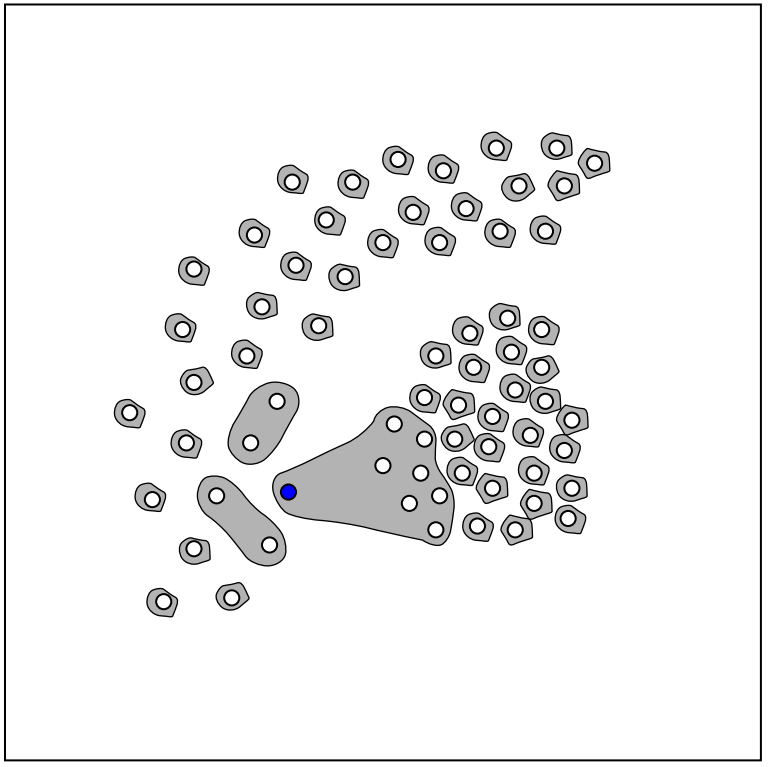
Dichtebasierte Verfahren

MajorClust



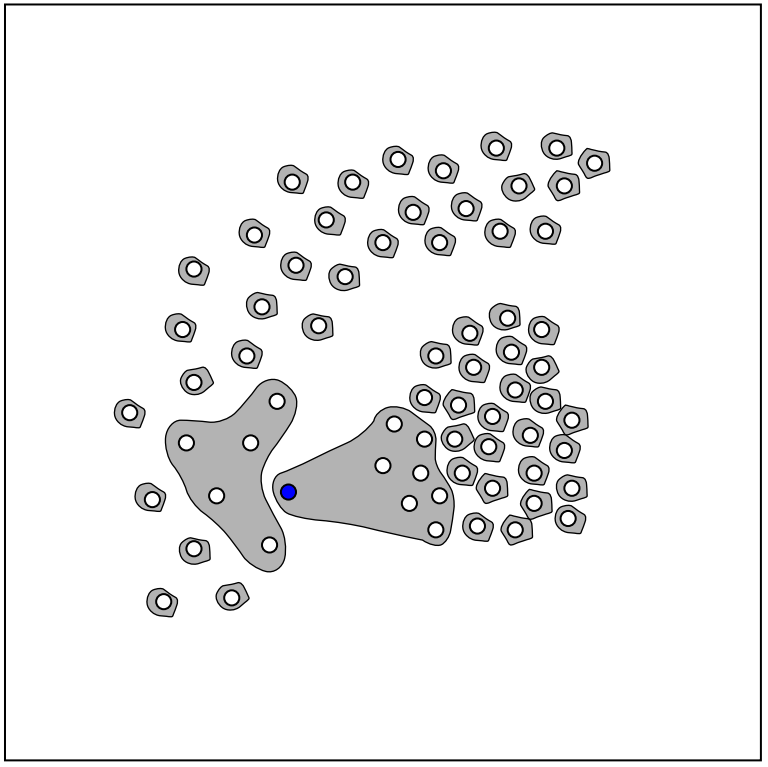
Dichtebasierte Verfahren

MajorClust



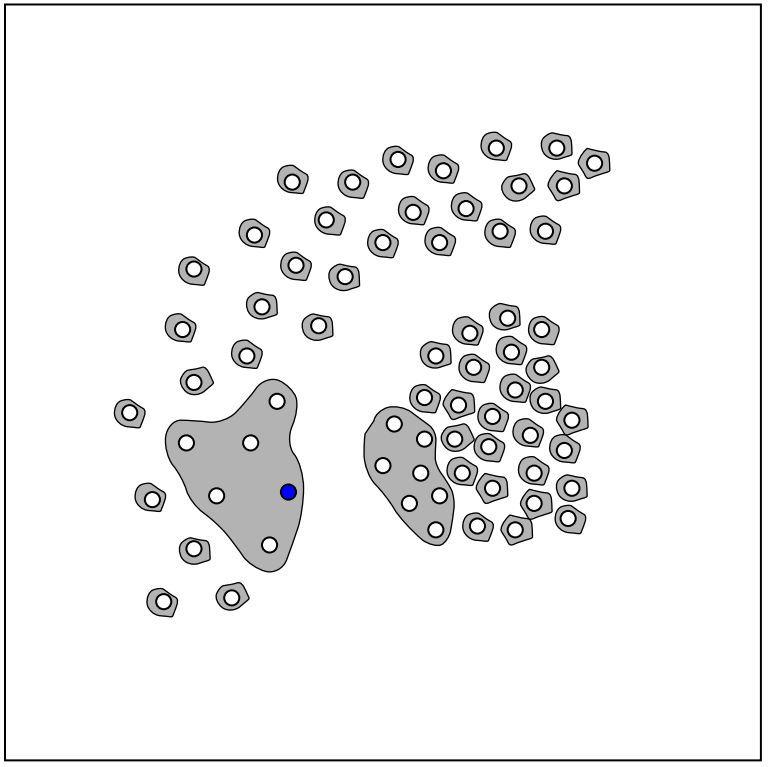
Dichtebasierte Verfahren

MajorClust



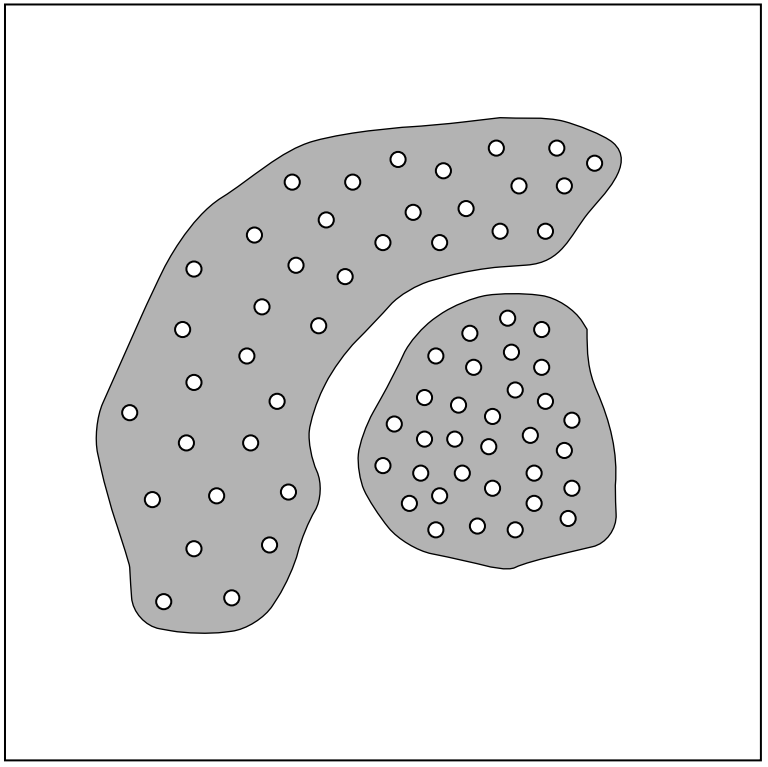
Dichtebasierte Verfahren

MajorClust



Dichtebasierte Verfahren

MajorClust



Dichtebasierte Verfahren

MajorClust: Prinzip der Dichteschätzung (Fortsetzung)

Jedes $\mathcal{C} = \{C_1, \ldots, C_k\}$ induziert k Teilgraphen. MajorClust ist eine Heuristik zur Maximierung des *gewichteten partiellen Kantenzusammenhangs*, $\Lambda(\mathcal{C})$.

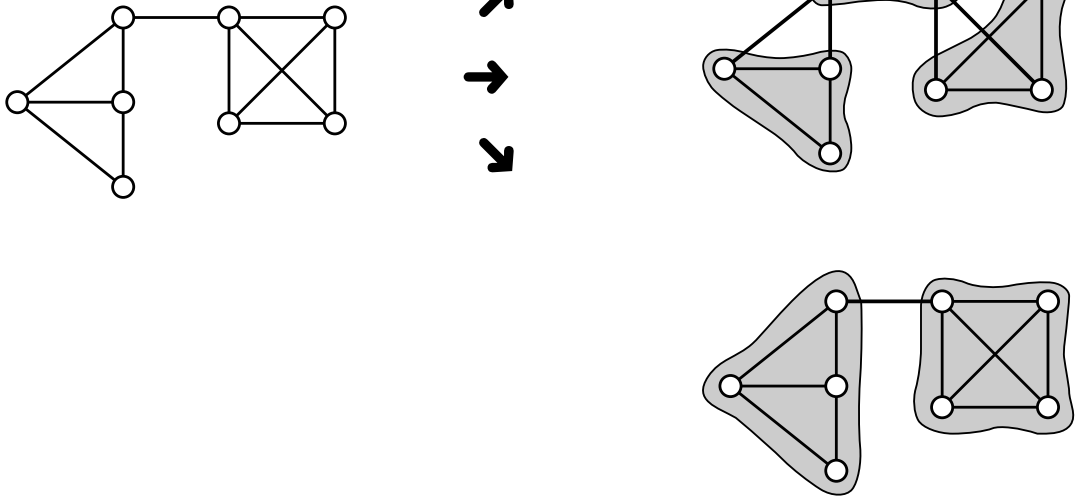
$$\Lambda(\mathcal{C}) = \sum_{i=1}^{k} |C_i| \cdot \lambda_i$$

Dichtebasierte Verfahren

MajorClust: Prinzip der Dichteschätzung (Fortsetzung)

Jedes $\mathcal{C} = \{C_1, \ldots, C_k\}$ induziert k Teilgraphen. MajorClust ist eine Heuristik zur Maximierung des *gewichteten partiellen Kantenzusammenhangs*, $\Lambda(\mathcal{C})$.

$$\Lambda(\mathcal{C}) = \sum_{i=1}^{k} |C_i| \cdot \lambda_i$$

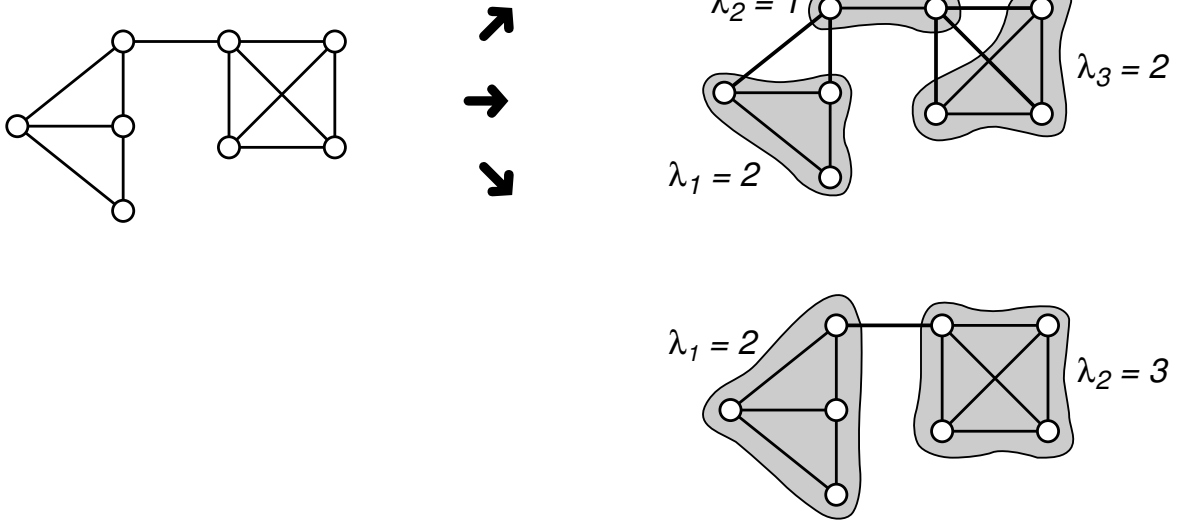


Dichtebasierte Verfahren

MajorClust: Prinzip der Dichteschätzung (Fortsetzung)

Jedes $\mathcal{C} = \{C_1, \ldots, C_k\}$ induziert k Teilgraphen. MajorClust ist eine Heuristik zur Maximierung des *gewichteten partiellen Kantenzusammenhangs*, $\Lambda(\mathcal{C})$.

$$\Lambda(\mathcal{C}) = \sum_{i=1}^{k} |C_i| \cdot \lambda_i$$

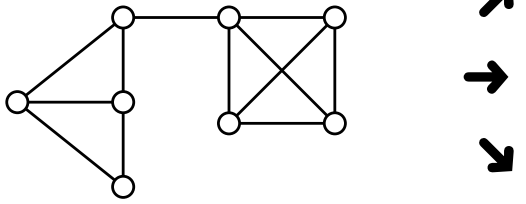


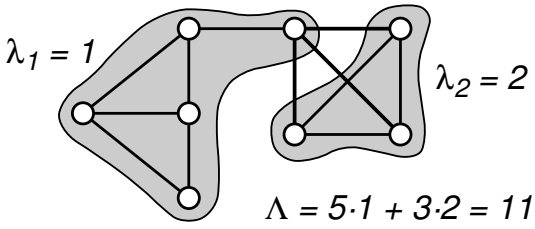
Dichtebasierte Verfahren

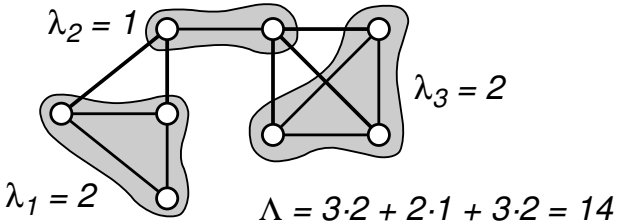
MajorClust: Prinzip der Dichteschätzung (Fortsetzung)

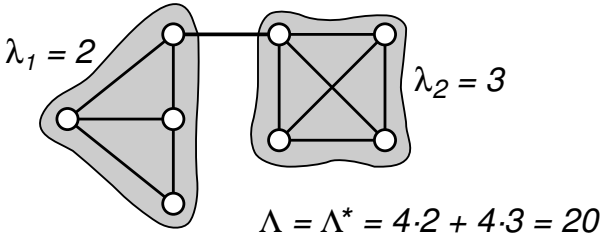
Jedes $\mathcal{C} = \{C_1, \ldots, C_k\}$ induziert k Teilgraphen. MajorClust ist eine Heuristik zur Maximierung des *gewichteten partiellen Kantenzusammenhangs*, $\Lambda(\mathcal{C})$.

$$\Lambda(\mathcal{C}) = \sum_{i=1}^{k} |C_i| \cdot \lambda_i$$



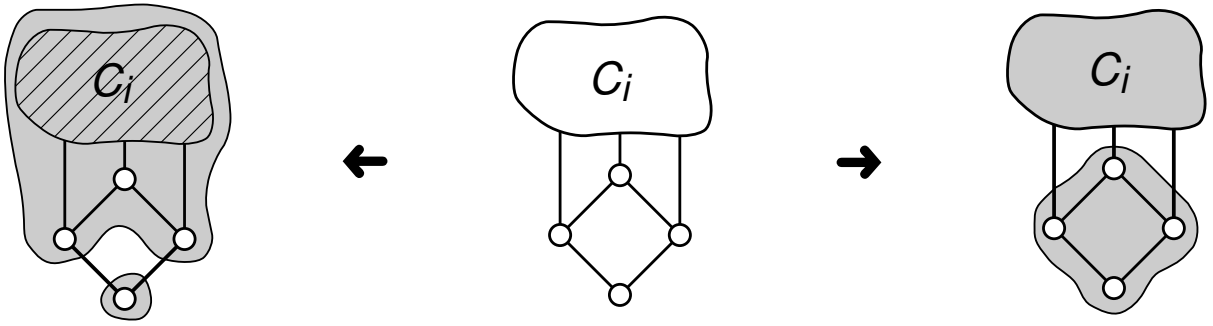






Dichtebasierte Verfahren

MajorClust: Prinzip der Dichteschätzung (Fortsetzung)

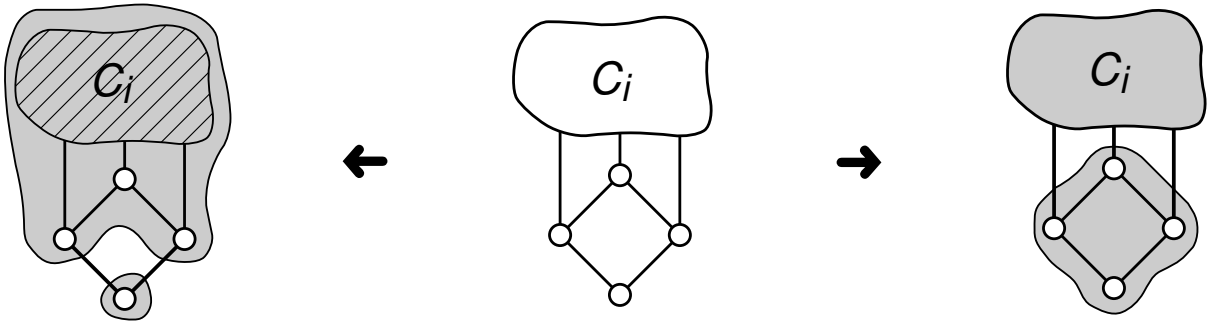


Min-Cut-Clustering

Λ-Maximierung

Dichtebasierte Verfahren

MajorClust: Prinzip der Dichteschätzung (Fortsetzung)



Min-Cut-Clustering

Λ-Maximierung

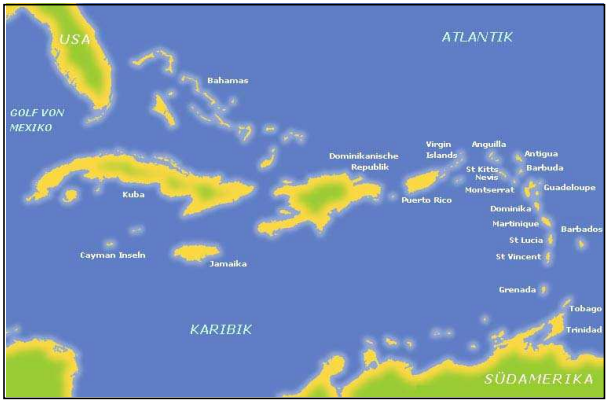
Satz 7 (Strong Splitting Condition [Stein/Niggemann 1999]**)**
Sei $\mathcal{C} = \{C_1, \ldots, C_k\}$ eine Partitionierung eines Graphen $G = \langle V, E, w \rangle$; weiterhin bezeichne $\lambda(G)$ den Kantenzusammenhang von G und $\lambda_1, \ldots, \lambda_k$ die Kantenzusammenhänge der von den $C_1, \ldots, C_k$ induzierten Subgraphen.

Gilt $\lambda(G) < \min\{\lambda_1, \ldots, \lambda_k\}$ (Strong Splitting Condition), so liefert Λ-Maximierung eine Aufteilung am minimalen Cut.

Dichtebasierte Verfahren

DBSCAN versus MajorClust: niedrigdimensionale Daten

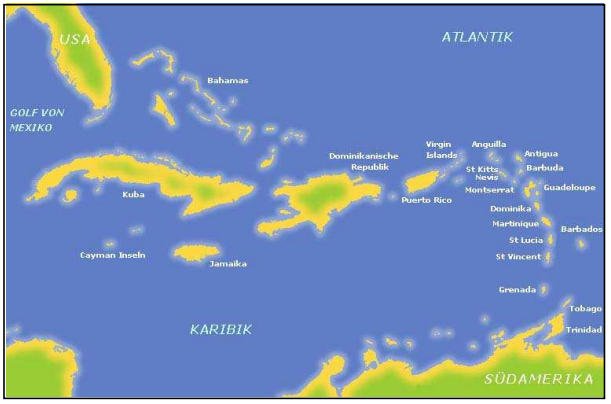
Karte der karibischen Inseln, etwa 20.000 Punkte:



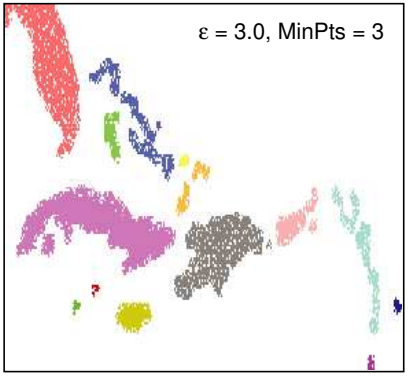
Dichtebasierte Verfahren

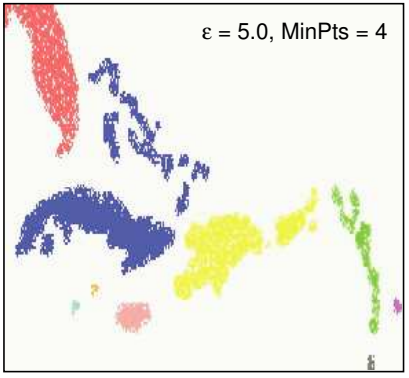
DBSCAN versus MajorClust: niedrigdimensionale Daten

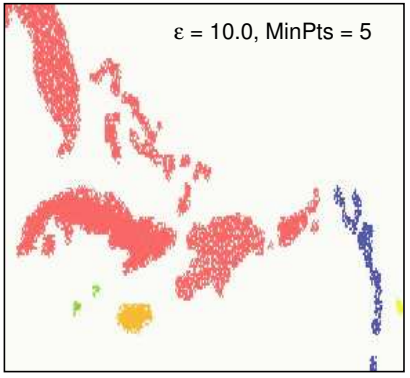
Karte der karibischen Inseln, etwa 20.000 Punkte:



DBSCAN:



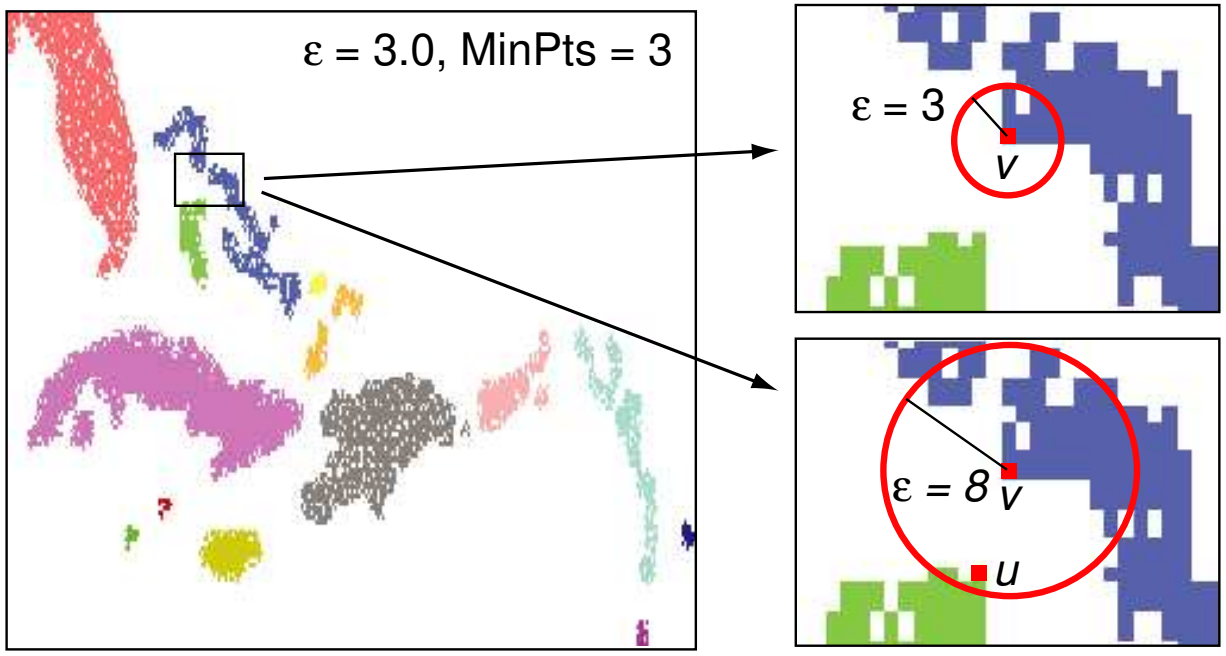




Dichtebasierte Verfahren

DBSCAN versus MajorClust: niedrigdimensionale Daten

Problematik geeigneter ε-Werte bei DBSCAN:



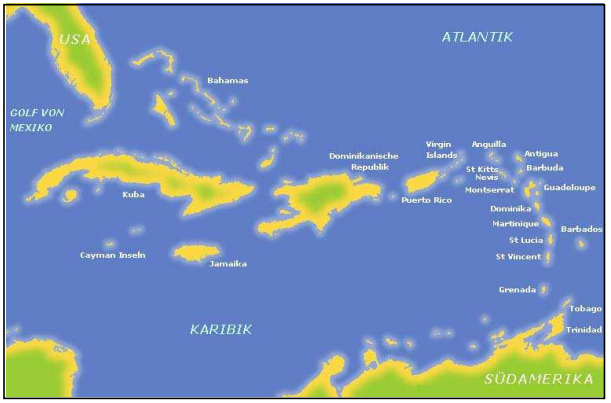
Zwei separate Cluster wurden gefunden.

Die Cluster werden vereinigt.

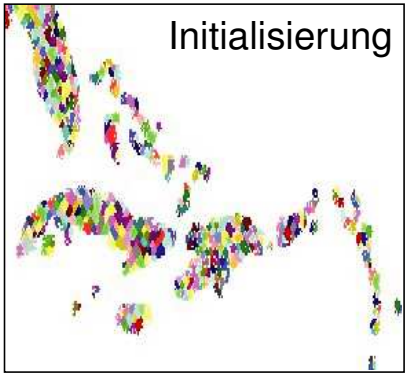
Dichtebasierte Verfahren

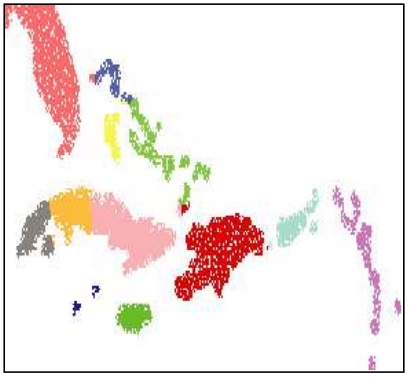
DBSCAN versus MajorClust: niedrigdimensionale Daten

Karte der karibischen Inseln, etwa 20.000 Punkte:



MajorClust:

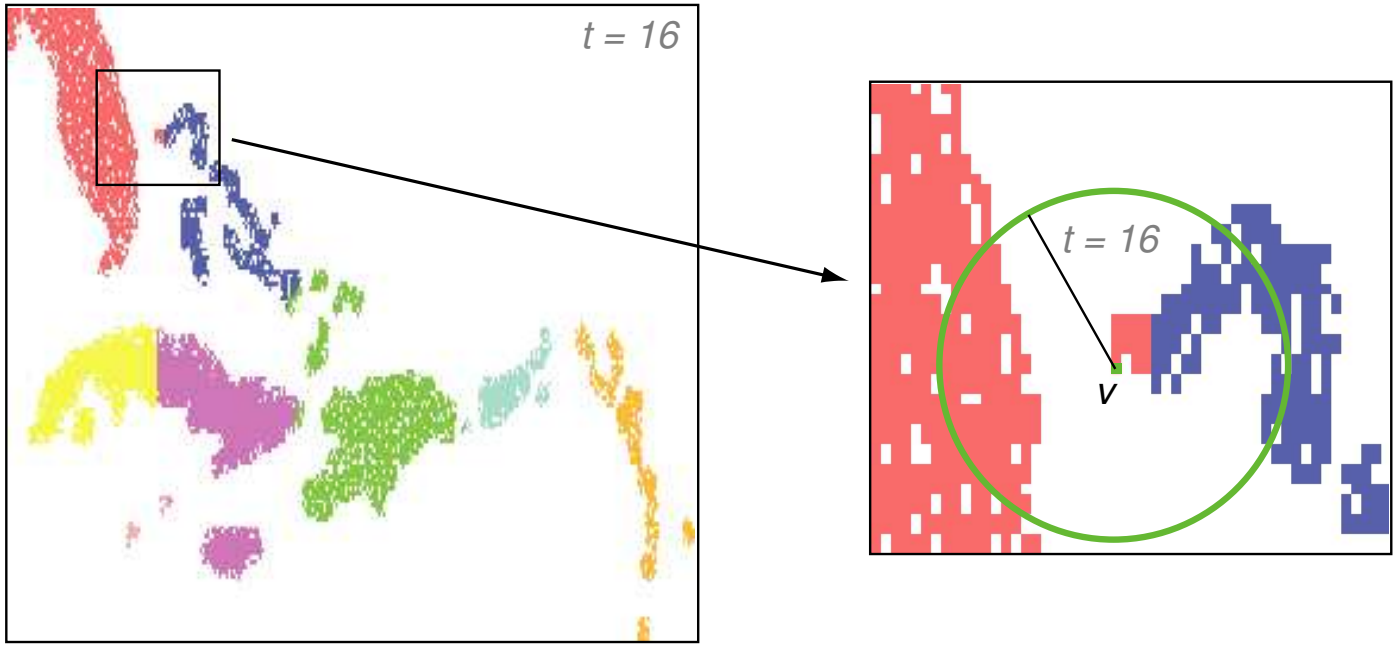




Dichtebasierte Verfahren

DBSCAN versus MajorClust: niedrigdimensionale Daten

Problematik der globalen Analyse (keine Beschränkung auf eine ε-Nachbarschaft) bei MajorClust:



Dichtebasierte Verfahren

DBSCAN versus MajorClust: hochdimensionale Daten

Dokumentenkategorisierung mit dem Reuters-Korpus:

- 1000 Dokumente
- 10 Kategorien: Politik, Kultur, Wirtschaft, etc.
- die Dokumente sind gleichverteilt, gehören genau zu einer Kategorie
- Dimension des Merkmalraums: > 10.000

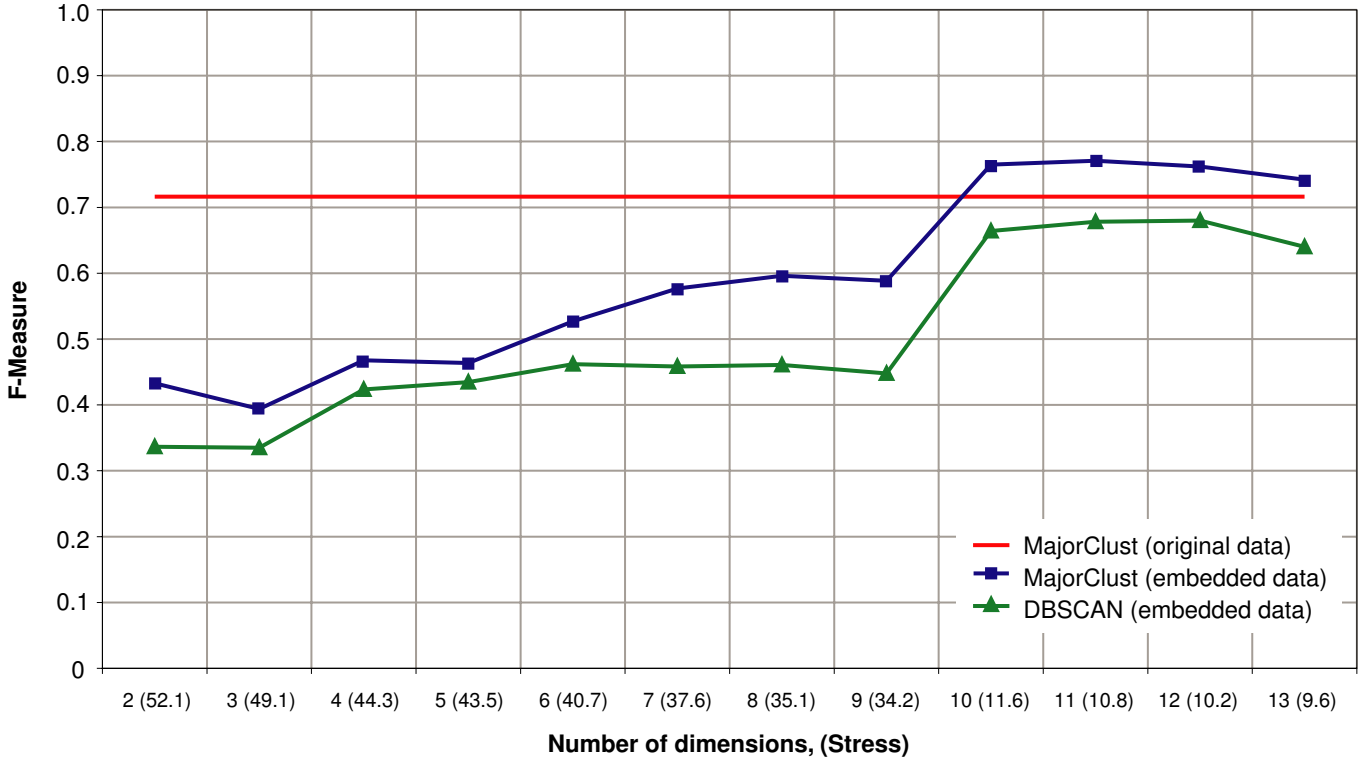
DBSCAN:

- degeneriert mit steigender Zahl der Dimensionen
- Ursache ist die Bestimmung der ε-Nachbarschaft
- Ausweg ist eine Dimensionreduktion, z. B. eine Einbettung der Daten mittels multidimensionaler Skalierung (MDS)

Dichtebasierte Verfahren

DBSCAN versus MajorClust: hochdimensionale Daten

Klassifikationsergebnisse (F-Measure), aufgetragen über Dimensionalität:



[Stein/Busch 2005]

Bemerkungen:

- Das Problem der Nachbarschaftssuche in hochdimensionalen Räumen ist meistens nicht effizient lösbar: Ab Dimensionen größer als 10-20 ist das lineare Durchsuchen aller Merkmalvektoren effizienter als die Verwendung von hochentwickelten, raumpartitionierenden Datenstrukturen wie R-Tree, X-Tree, Quadtree, KD-Tree, etc.

  Einen Ausweg bieten die Ansätze wie Locality-Sentitve-Hashing oder Fuzzy-Fingerprinting.
  Siehe auch: [Weber 99] [Gionis/Indyk/Motwani 99-04] [Stein 05] [Stein/SMZE 05]
- DBSCAN verwendet zur Bestimmung der ε-Nachbarschaft die R-Tree-Datenstruktur. Diese Datenstruktur leistet einen wesentlichen (wenn nicht den größten) Teil der Cluster-Analyse innerhalb von DBSCAN.
- Möchte man DBSCAN für hochdimensionale Daten verwenden, ist eine Einbettung der Daten in einen niedrigdimensionalen Raum unvermeidbar. Dabei ist zu bedenken, dass eine Dimensionsreduktion durch Einbettung rechenintensiv ist und das gute Laufzeitverhalten von DBSCAN zunichte macht.

Kapitel ML: X (Fortsetzung)

X. Cluster-Analyse

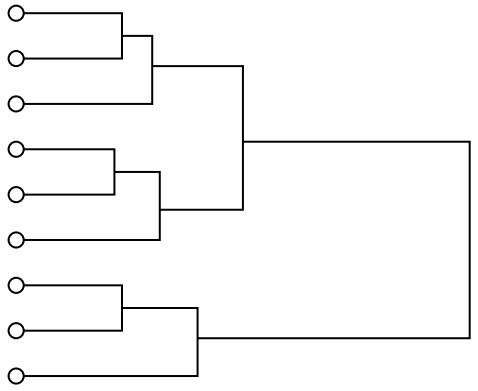
- Einordnung Data Mining
- Einführung in die Cluster-Analyse
- Hierarchische Verfahren
- Iterative Verfahren
- Dichtebasierte Verfahren
- Cluster-Evaluierung

Cluster-Evaluierung

Problematik hierarchischer Verfahren

Grenzen der Unüberwachtheit: Wann mit Aufteilen / Zusammenfassen aufhören?

Ergebnis eines hierarchischen Verfahrens:

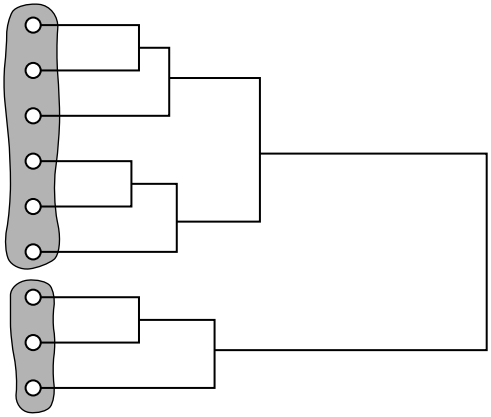


Cluster-Evaluierung

Problematik hierarchischer Verfahren

Grenzen der Unüberwachtheit: Wann mit Aufteilen / Zusammenfassen aufhören?

Ergebnis eines hierarchischen Verfahrens:



Festlegung (subjektiver) Schwellwerte für

- Kantengewichte
- Fehlerquadratsummen
- Clustergrößen

Cluster-Evaluierung

Problematik hierarchischer Verfahren (Fortsetzung)

> *"A hierarchical method suffers from the defect that it can never repair what was done in previous steps."*
>
> [Kaufmann/Rousseuw 1990]

Cluster-Evaluierung

Problematik hierarchischer Verfahren (Fortsetzung)

> *"A hierarchical method suffers from the defect that it can never repair what was done in previous steps."*
>
> [Kaufmann/Rousseuw 1990]

Verbesserung der Analysequalität durch zweistufiges Vorgehen:

1. (a) Identifikation von Ausreißern mittels Single-Link
   (b) Graph-Coarsening durch Knotenverschmelzung
   (c) Pre-Cluster-Analyse durch Berechnung von Nearest-Neighbor-Graph
2. Cluster-Analyse für vereinfachten Graph

Verbesserung der Verfahrensrobustheit durch Variation:

- mehrmalige Cluster-Analyse bei veränderter Distanzberechnung $d_{\mathcal{C}}$

Verbesserung eines Clusterings $\mathcal{C}$ durch Reparatur:

- lokale Knotenaustauschheuristiken mit Optimierungsfunktionen

Cluster-Evaluierung

Problematik exemplarbasierter Verfahren

Wieviele Repräsentanten soll man vorgeben – bzw. $k =$?

Cluster-Evaluierung

Problematik exemplarbasierter Verfahren

Wieviele Repräsentanten soll man vorgeben – bzw. $k =$?

Vollständigkeitsstrategie:

- alle Möglichkeiten ausprobieren; d. h., $\mathcal{C}_1, \ldots, \mathcal{C}_{|V|}$ generieren

➜ Kriterium zur Evaluierung der Qualität von Clusterings $\mathcal{C}_i$ erforderlich

Schwellwertstrategie:

- ausgehend von einer Überschätzung (Unterschätzung) von k erlaubt man das Vereinigen (Aufteilen) von Clustern $C \in \mathcal{C}$

➜ Schwellwerte erforderlich, die sogar Cluster-spezifisch sein sollten

Überwachungsstrategie:

- der Domänenexperte entscheidet

Cluster-Evaluierung

Problematik exemplarbasierter Verfahren (Fortsetzung)

Welche Repräsentanten soll man vorgeben?

Cluster-Evaluierung

Problematik exemplarbasierter Verfahren (Fortsetzung)

Welche Repräsentanten soll man vorgeben?

Informationsstrategie:

- Schätzung der Repräsentanten aus den Mittelwerten der Randverteilungen
- Schätzung durch Domänenexperten

Varianzminimierungsstrategie:

- Wiederholung der Cluster-Analyse mit zufällig oder systematisch variierten Repräsentanten