

INTELLIGENTE DATENANALYSE IN MATLAB

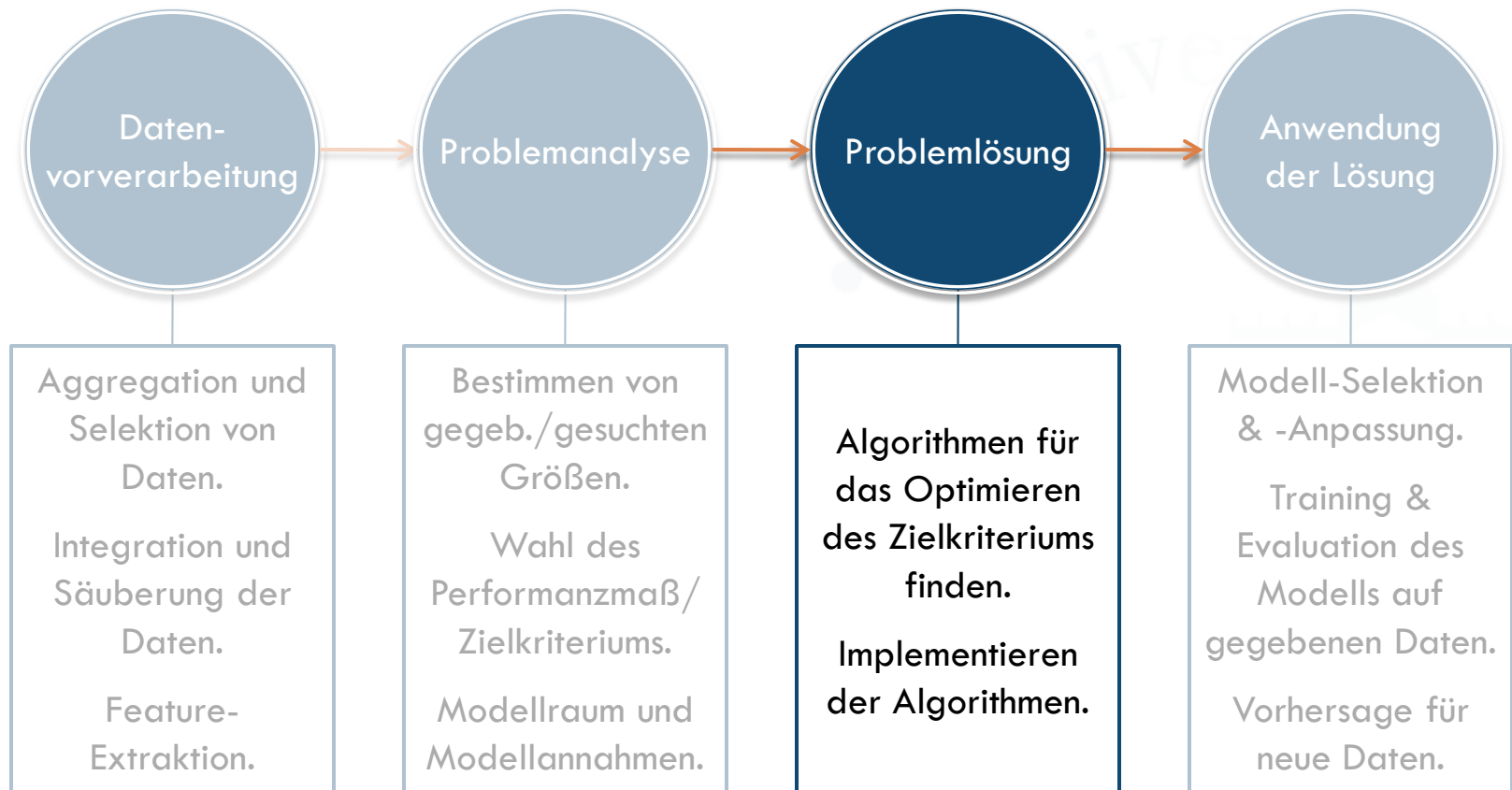
Unüberwachtes Lernen: Clustern von Instanzen

Literatur

- Chris Bishop: Pattern Recognition and Machine Learning.
- Jiawei Han und Micheline Kamber: Data Mining – Concepts and Techniques.
- Ulrike von Luxburg: A Tutorial on Spectral Clustering.
http://www.kyb.mpg.de/publications/attachments/Luxburg06_TR_%5B0%5D.pdf
- Matteo Matteucci: A Tutorial on Clustering Algorithms.
http://home.dei.polimi.it/matteucc/Clustering/tutorial_html

Überblick

□ Schritte der Datenanalyse:



Unüberwachtes Lernen

Arten von Modellen & Lernproblemen

- Clustern von Instanzen:
 - Finden und deterministische bzw. probabilistische Zuweisung zu Bereichen mit vielen Datenpunkten (Cluster).
- Clustern von Attributen:
 - Häufig zusammen auftretende (ähnliche bzw. korrelierte) Attribute finden.
- Clustern von Instanzen & Attributen (Co-Clustern):
 - Gleichzeitiges Clustern von Instanzen und Attributen.
- Outlier Detection:
 - Suche nach seltenen/auffälligen Datenpunkten.

Clustern von Instanzen

Problemstellung

- Gegeben: Menge von n Trainingsdaten $\mathbf{x}_i \in \mathbb{R}^m$ mit m Attributen (Datenmatrix $\mathbf{X} \in \mathbb{R}^{m \times n}$ mit Spaltenvektoren \mathbf{x}_i) und unbekannten Zielattributen (ungelabelte Daten).
- Gesucht:
 - Gruppierung der Instanzen, d.h. Finden von Bereichen mit vielen Datenpunkten (Cluster).
 - Eindeutige bzw. probabilistische Zuweisung der Datenpunkte zu den Clustern (Belegung der Zielattribute).

Clustern von Instanzen

Beispiel

□ Tabellendarstellung:

Monat	Bewölkung	Temperatur	Luftfeuchtigkeit	Wind	Jahreszeit
Juli	sonnig	warm	hoch	wenig	?
September	sonnig	warm	hoch	stark	?
August	bedeckt	warm	hoch	wenig	?
April	Regen	mild	hoch	wenig	?
Oktober	Regen	kühl	normal	wenig	?
Dezember	Regen	kühl	normal	stark	?
Januar	bedeckt	kühl	normal	stark	?
Juli	sonnig	mild	hoch	wenig	?
Februar	sonnig	kühl	normal	wenig	?
März	Regen	mild	normal	wenig	?
November	sonnig	mild	normal	stark	?
August	bedeckt	mild	hoch	stark	?
Juni	bedeckt	warm	normal	wenig	?
April	Regen	mild	hoch	stark	?

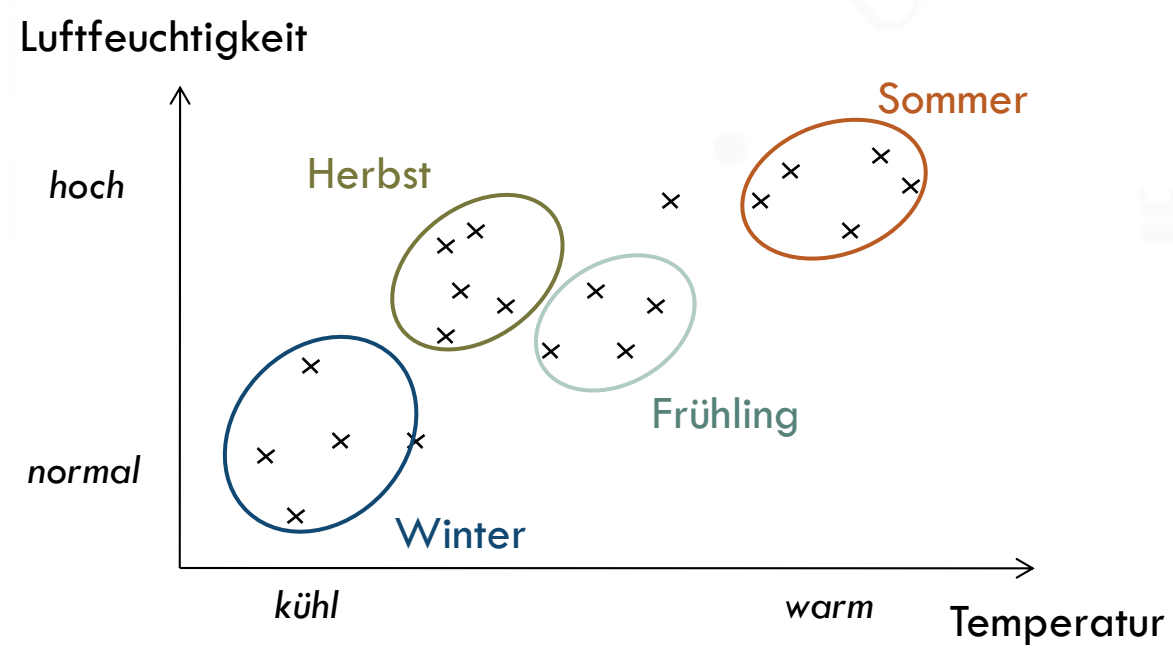
Trainingsdaten

Zielgröße

Clustern von Instanzen

Beispiel

- Diagramm bzgl. der Attribute Luftfeuchtigkeit und Temperatur:



Clustern von Instanzen

Motivation

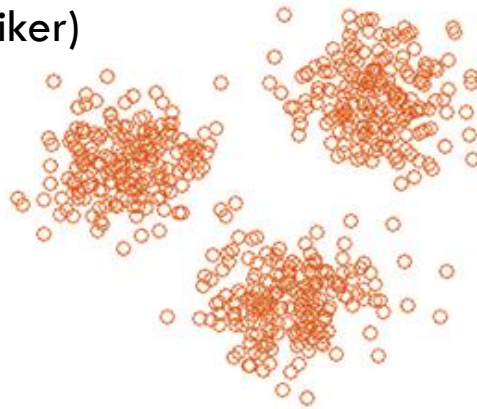
- Besseres Verständnis/Beschreibung der Daten:
 - Kundensegmentierung: Zielgerichtetes Marketing, Produktentwicklung usw. für einzelne Zielgruppen.
 - Landnutzung/Stadtplanung: Identifizieren von Regionen mit ähnlichen geographischen, klimatischen, städtebaulichen Eigenschaften.
 - Risikoanalyse: Klassen von Kunden mit unterschiedlichen Versicherungsrisiken erkennen.
- Teil der Datenvorverarbeitung für weitere Analyse:
 - Diskretisierung von numerischen Attributen.
 - Outlier-Detection.

Clustern von Instanzen

Anwendung

- Überblick über eine Dokumentenkollektion.
 - Suche nach Stichwort „Kohl“ liefert viele Dokumente.
 - Idee: Zeige dem Nutzer Cluster um genauere Auswahl des Themas zu ermöglichen.

Helmut Kohl (Politiker)



Kohl's (US Kaufhaus)

Kohl (Gemüse)

Clustern von Instanzen

Anwendung

- Spam-Kampagnen identifizieren.
 - Spam-Kampagne ist große Menge ähnlicher (aber nicht gleicher) Emails.
 - Idee: Email-Attribute (z.B. enthaltene Wörter) clustern und Spam-Cluster durch nachgeschalteten Klassifikator erkennen.

Hello. This is Terry Hagan. We are accepting your mortgage application. Our company confirms you are eligible for a \$250,000 loan for a \$380.00/month. Approval minute, so please fill out the form of Best Regards, Terry Hagan; Senior Trades/Finance Department North

Dear Mr/Mrs, This is Brenda Dunn. We are accepting your mortgage application. Our office confirms you can get a \$228,000 loan for a \$371.00 per month payment. Follow the link to our website and submit your contact information. Best Regards, Brenda Dunn; Accounts Manager Trades/Finance Department East Office

Clustern von Instanzen

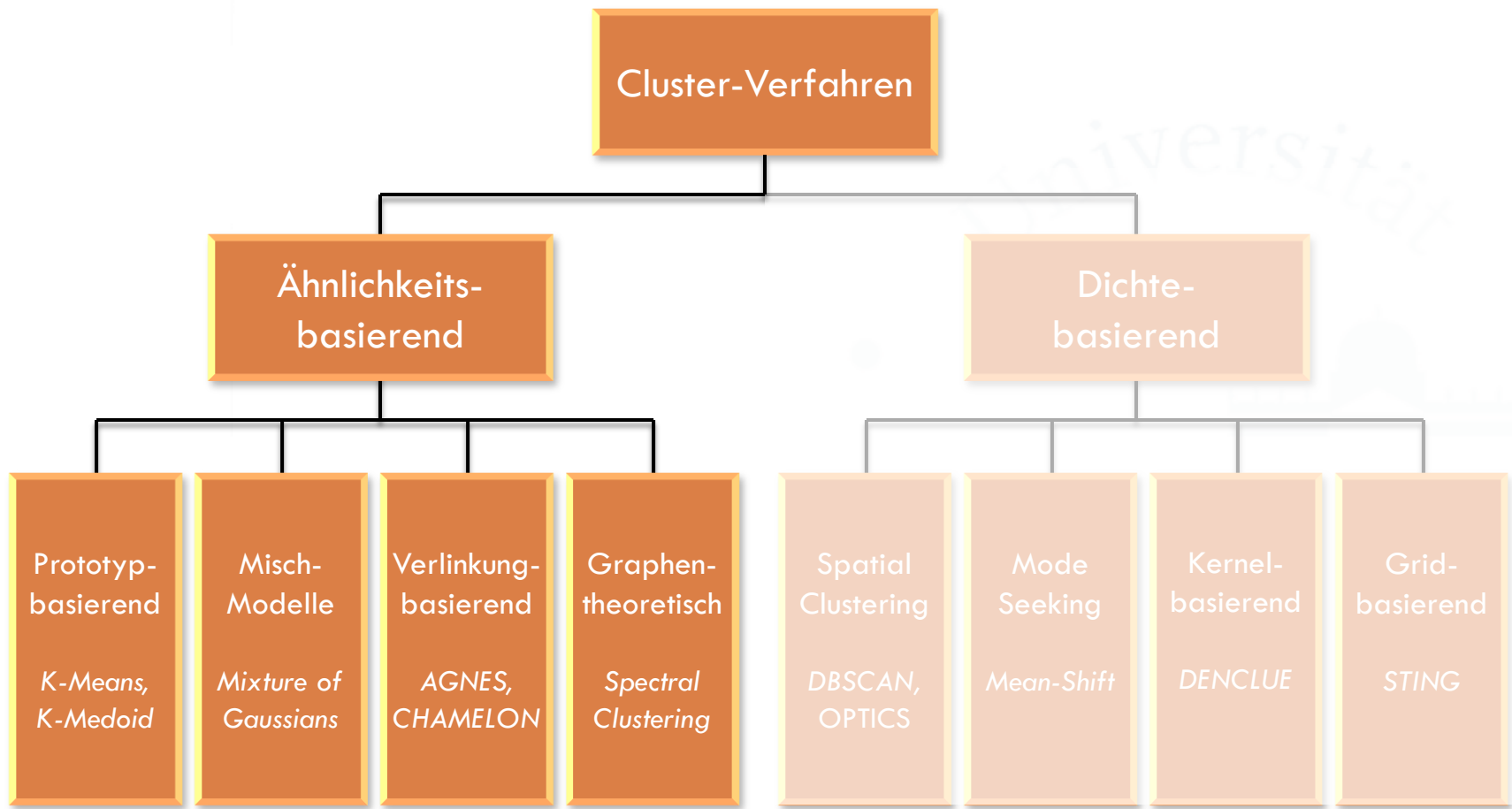
Evaluierung

- Qualitätsmerkmale eines Clusterings:
 - Hohe Ähnlichkeit zweier Datenpunkte eines Clusters (*intra-cluster similarity*).
 - Geringe Ähnlichkeit zwischen Datenpunkten verschiedener Cluster (*inter-cluster similarity*).
 - Anzahl, Form und Größenvarianz der Cluster.
 - Interpretierbarkeit, d.h. gefundene Cluster entsprechen echten (versteckten) Clustern.

Stichprobenartig durch Experten prüfen.

Clustern von Instanzen

Verfahren



Prototyp-Verfahren

- Gegeben:
 - ▣ Ungelabelte Daten $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$.
 - ▣ Anzahl vermuteter Cluster k mit $1 < k < n$.
 - ▣ Ähnlichkeitsmaß zwischen Datenpunkten.
- Gesucht: Partitionierung der Daten in k Cluster.
- Ziel: Kleiner Abstand zw. Punkten im selben Cluster und großer Abstand zw. Punkten verschiedener Cluster.
 - ▣ Exponentiell viele Partitionierungen \Rightarrow Suche NP-hart!
 - ▣ Heuristische Suche (lokal optimal): *K-Means* und *K-Medoids*.

Prototyp-Verfahren

K-Means

- Gesucht ist eine Zuweisung $\mathbf{R} = \{\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_n\}$ der Daten $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ zu den Clustern mit

$$\mathbf{r}_i \in \{0, 1\}^k \quad r_{ij} = \begin{cases} 1 & \mathbf{x}_i \text{ in Cluster } j \\ 0 & \text{sonst} \end{cases}$$

z.B. $\mathbf{r}_5 = [0 \ 0 \ 1 \ 0]^T$ falls das 5. Beispiel in Cluster 3 liegt.

- Seien $\{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \dots, \boldsymbol{\mu}_k\}$ Cluster-Mittelpunkte (Prototypen).
- Ziel: Minimaler quadratischer Abstand zu den Cluster-Mittelpunkten:

$$\min \sum_{i=1}^n \sum_{j=1}^k r_{ij} \|\mathbf{x}_i - \boldsymbol{\mu}_j\|^2$$

Prototyp-Verfahren

K-Means

- Gleichzeitiges Minimieren über $\{\mu_1, \mu_2, \dots, \mu_k\}$ und $\{r_1, r_2, \dots, r_n\}$ sehr schwierig.
- Idee: Abwechselnde Minimierung.
- Algorithmus:

K-Means (*Instanzen* \mathbf{x}_i , *Clusteranzahl* k)

Setze $l=0$ und wähle zufällig $\forall i \mu_i^0 = \mathbf{x}_i$

DO

$$\{\mathbf{r}_1^{l+1}, \dots, \mathbf{r}_n^{l+1}\} = \arg \min_{\mathbf{r}_1, \dots, \mathbf{r}_n} \sum_{i=1}^n \sum_{j=1}^k r_{ij} \|\mathbf{x}_i - \mu_j^l\|^2$$

$$\{\mu_1^{l+1}, \dots, \mu_n^{l+1}\} = \arg \min_{\mu_1, \dots, \mu_n} \sum_{i=1}^n \sum_{j=1}^k r_{ij}^{l+1} \|\mathbf{x}_i - \mu_j\|^2$$

$l = l + 1$

WHILE $\{\mathbf{r}_1^l, \dots, \mathbf{r}_n^l\} \neq \{\mathbf{r}_1^{l-1}, \dots, \mathbf{r}_n^{l-1}\}$

RETURN $\{\mathbf{r}_1^l, \dots, \mathbf{r}_n^l\}$

Expectation

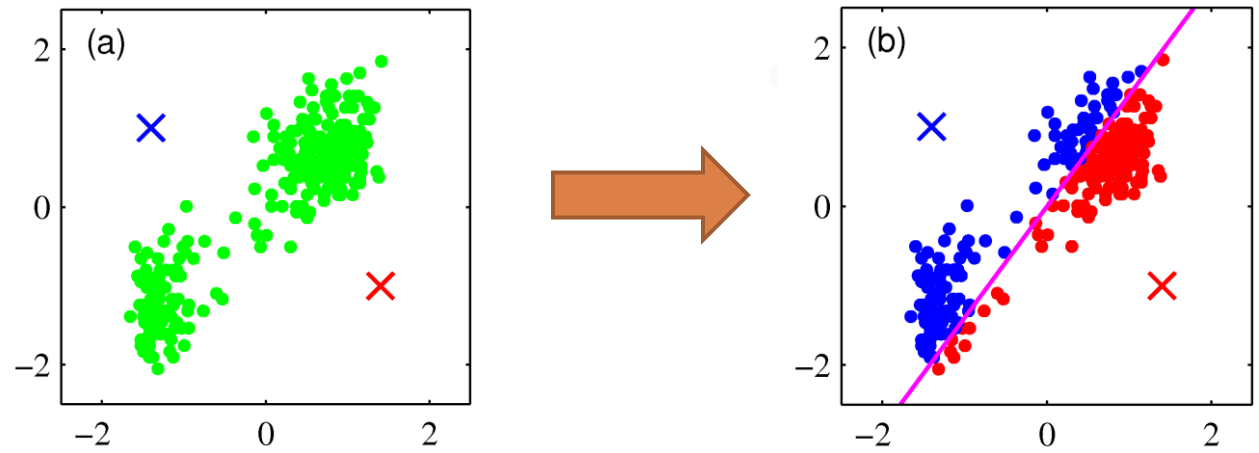
Maximization

Prototyp-Verfahren

K-Means

□ Expectation-Schritt: $\{\mathbf{r}_1^{l+1}, \dots, \mathbf{r}_n^{l+1}\} = \arg \min_{\mathbf{r}_1, \dots, \mathbf{r}_n} \sum_{i=1}^n \sum_{j=1}^k r_{ij} \|\mathbf{x}_i - \boldsymbol{\mu}_j^l\|^2$

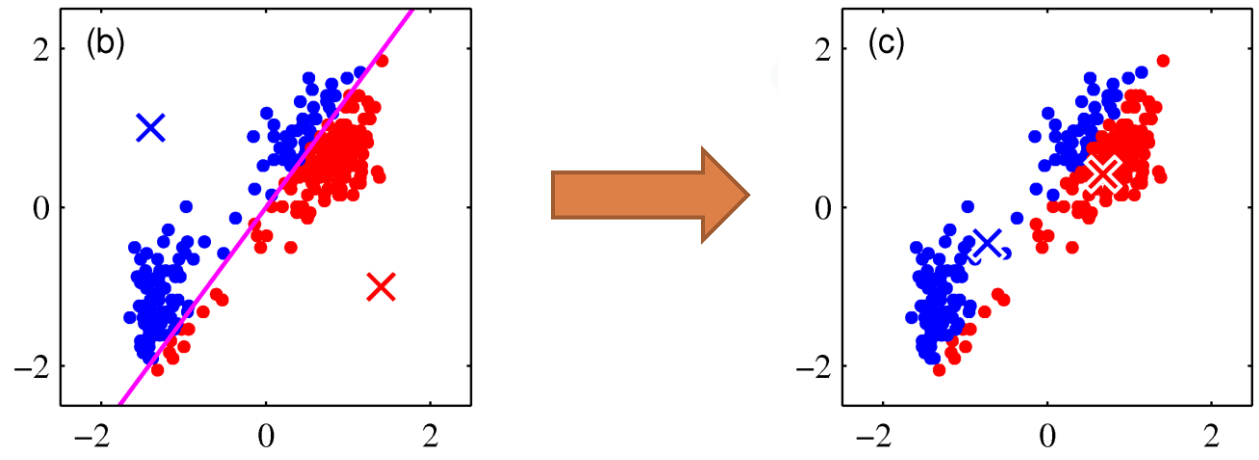
▣ Ordne jeden Punkt dem ihm nächsten Cluster-Mittelpunkt zu.



Prototyp-Verfahren

K-Means

- Maximization-Schritt: $\{\mu_1^{l+1}, \dots, \mu_n^{l+1}\} = \arg \min_{\mu_1, \dots, \mu_n} \sum_{i=1}^n \sum_{j=1}^k r_{ij}^{l+1} \|\mathbf{x}_i - \mu_j\|^2$
- Bestimme neue Cluster-Mittelpunkte $\mu_j^{l+1} = \frac{\sum_{i=1}^n r_{ij}^{l+1} \mathbf{x}_i}{\sum_{i=1}^n r_{ij}^{l+1}}$



Prototyp-Verfahren

K-Means

□ Vorteile:

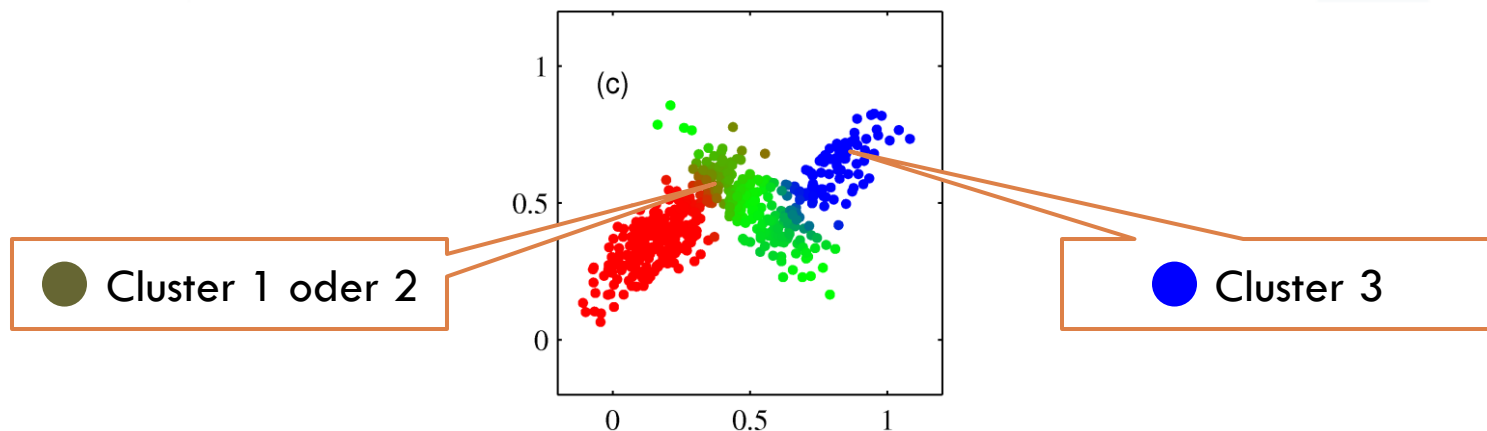
- Einfach zu implementieren
- Relativ schnell.
 - In $O(n \cdot k)$ pro Iteration.
 - Effizienten Berechnung neuer Cluster-Mittelpunkte möglich.

□ Nachteile:

- Nur lokales Optimum garantiert (unterschiedliche Startwerte = unterschiedliche Lösungen).
- Harte Zuweisung zu Clustern (nicht-probabilistisch).
- Anzahl Cluster muss vorgeben werden.
- Nur für numerische Attribute geeignet.

Mischmodelle

- Gegeben:
 - Ungelabelte Daten $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$.
 - Anzahl vermuteter Cluster k muss nicht bekannt sein.
 - Verteilungsannahme der Datenpunkte.
- Gesucht: Probabilistische Partitionierung der Daten.



Mischmodelle

□ Idee:

- Generatives (Misch-)Modell welches Daten \mathbf{X} erzeugt hat mit Modell-Parameter Θ .
- Cluster-Zuordnungen $\mathbf{R} = \{r_1, r_2, \dots, r_n\}$ sind versteckte Variablen des Modells.

□ Ziel:

- Parameter Θ mit maximaler A-Posteriori-Wahrscheinlichkeit (MAP):

$$\Theta^* = \arg \max_{\Theta} p(\Theta | \mathbf{X}) = \arg \max_{\Theta} p(\mathbf{X} | \Theta) p(\Theta)$$

Mischmodelle

Mixture of Gaussians

- Gesucht ist eine Zuweisung $\{\pi_1, \pi_2, \dots, \pi_n\}$ der Daten $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ zu den Clustern mit

$$\pi_i \in [0, 1]^k \quad \sum_{j=1}^k \pi_{ij} = 1$$

- Annahme:
 - ▣ Daten-erzeugendes Modell ist Kombination von Gauß-Verteilungen mit unterschiedlichen Mittelwerten und Kovarianzen

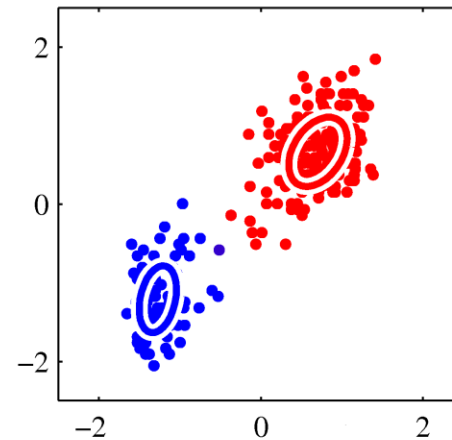
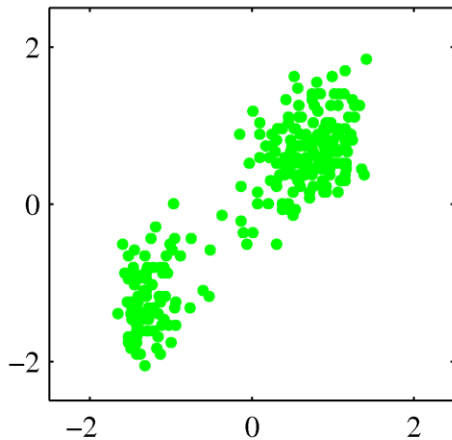
$$p(\mathbf{X} | \Theta) = \prod_{i=1}^n p(\mathbf{x}_i | \Theta) = \prod_{i=1}^n \left(\sum_{j=1}^k \pi_{ij} N(\mathbf{x}_i | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) \right)$$

mit Parametern $\Theta = (\{\pi_{ij}\}, \{\boldsymbol{\mu}_j\}, \{\boldsymbol{\Sigma}_j\})$.

Mischmodelle

Mixture of Gaussians

- Schätzen der Parameter durch abwechselnde Optimierung analog zu K-Means (EM-Algorithmus).



Mischmodelle

Mixture of Gaussians

- Vorteile:
 - Probabilistische Zuweisung zu Clustern.
 - Anzahl Cluster muss nicht vorgeben werden.
 - Automatischer Trade-off zwischen Anzahl Clustern und Anpassung an Daten.
- Nachteile:
 - Langsamer und komplexer als K-Means.
 - Nur für numerische Attribute geeignet.

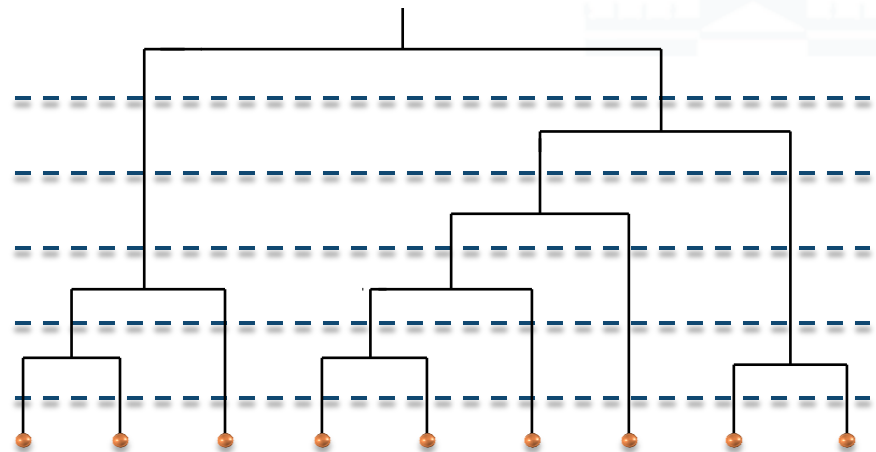
Verlinkungsbasierte Verfahren

□ Gegeben:

- Ungelabelte Daten $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$.
- Anzahl vermuteter Cluster k muss nicht bekannt sein.
- Abstandsmaß *dist* zwischen Datenpunkten.

□ Gesucht:

- Darstellung der Daten in Form eines *Dendrogramms*.



Verlinkungsbasierte Verfahren

□ Idee:

□ *Agglomerative Hierarchical Clustering.*

- Zu Beginn bildet jeder Datenpunkt einen eigenen Cluster.
- Benachbarte Cluster werden sukzessive verschmolzen (bottom-up).

□ *Divisive Hierarchical Clustering.*

- Zu Beginn bilden alle Daten einen gemeinsamen Cluster.
- Cluster werden sukzessive gesplittet (top-down).

□ Ziel:

- Iteratives Aufbauen des Clusterings bis vorgegebene Qualität erreicht ist.

Verlinkungsbasierte Verfahren

Agglomerative Nesting (AGNES)

Algorithmus:

AGNES (Instanzen \mathbf{x}_i)

Setze $C_i = \{\mathbf{x}_i\} \forall i$

DO

$D_{ij} = \text{dist}(C_i, C_j) \forall i, j$

$(i^*, j^*) = \arg \min_{i, j} (D_{ij})$

Verschmelze C_{i^*} und C_{j^*}

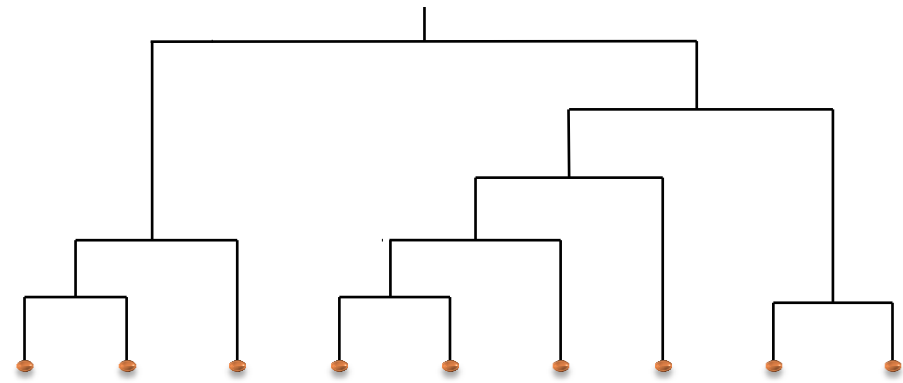
WHILE $D_{i^*j^*} < \varepsilon$

RETURN $\{C_i\}$

Jeder Datenpunkt ein eigener Cluster

Distanz zwischen zwei Clustern?

Mindest-Qualität (Abbruchbedingung)



Verlinkungsbasierte Verfahren

Agglomerative Nesting (AGNES)

□ Distanz zwischen zwei Clustern:

□ Single Linkage: $dist(C_i, C_j) = \min_{\mathbf{x} \in C_i, \mathbf{y} \in C_j} (dist(\mathbf{x}, \mathbf{y}))$

□ Complete Linkage: $dist(C_i, C_j) = \max_{\mathbf{x} \in C_i, \mathbf{y} \in C_j} (dist(\mathbf{x}, \mathbf{y}))$

□ Average Linkage: $dist(C_i, C_j) = \frac{1}{|C_i| |C_j|} \sum_{\mathbf{x} \in C_i, \mathbf{y} \in C_j} dist(\mathbf{x}, \mathbf{y})$

□ Average Group Linkage: $dist(C_i, C_j) = \frac{1}{|C_i \cup C_j|} \sum_{\mathbf{x}, \mathbf{y} \in C_i \cup C_j} dist(\mathbf{x}, \mathbf{y})$

□ Centroid: $dist(C_i, C_j) = dist \left(\frac{1}{|C_i|} \sum_{\mathbf{x} \in C_i} \mathbf{x}, \frac{1}{|C_j|} \sum_{\mathbf{y} \in C_j} \mathbf{y} \right)$

Verlinkungsbasierte Verfahren

Agglomerative Nesting (AGNES)

□ Vorteile:

- Einfach zu implementieren.
- Relativ schnell.
- Iteratives Verfahren, für Online-Clustering geeignet.
- Für nominale, ordinale und numerische Attribute geeignet.
- Anzahl Cluster muss nicht vorgegeben werden.

□ Nachteile:

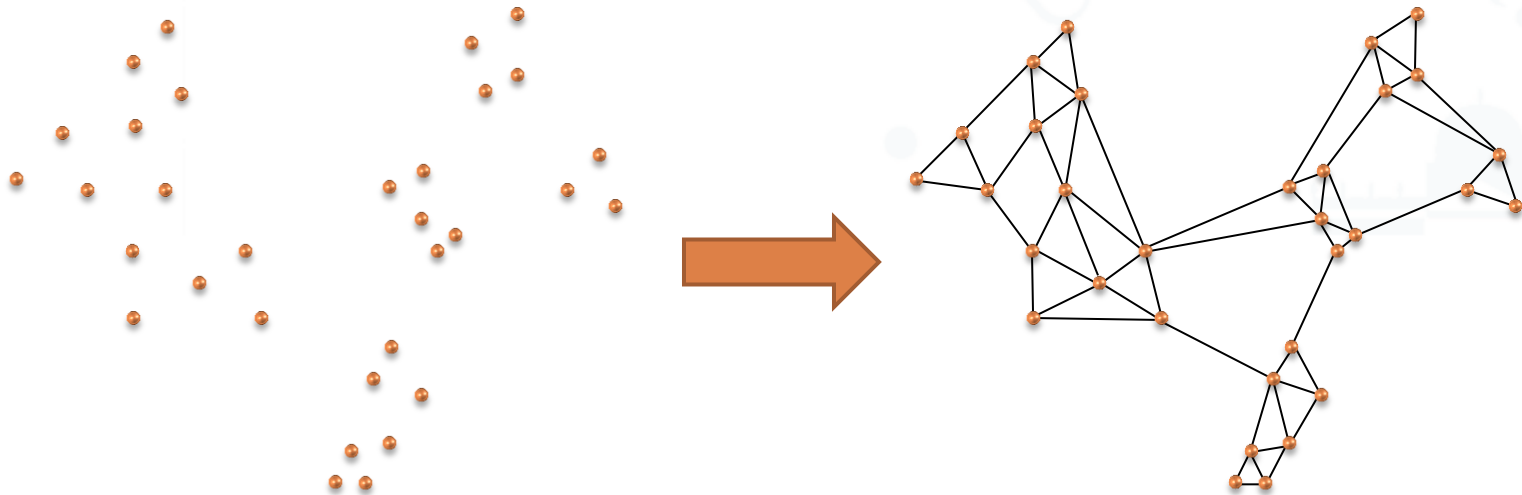
- Nur lokales Optimum garantiert.
- Harte Zuweisung zu Clustern (nicht-probabilistisch).
- Maß für Cluster-Distanz & Abbruchbedingung muss vorgegeben werden.

Graphentheoretische Verfahren

- Gegeben:
 - Ungelabelte Daten $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$.
 - Anzahl vermuteter Cluster k mit $1 < k < n$.
 - Ähnlichkeitsmaß (Kernel) sim zwischen Datenpunkten.
- Gesucht: Partitionierung der Daten in k Cluster.
- Ziel: Hohe Ähnlichkeit zw. Punkten im selben Cluster und geringe Ähnlichkeit zw. Punkten verschiedener Cluster.
 - Repräsentation der Daten als Graph und Partitionieren des Graphs.

Graphentheoretische Verfahren

- Ähnlichkeit zwischen Datenpunkten (Knoten) bilden gewichtete Kanten:



Graphentheoretische Verfahren

- Konstruktion des Graphen:
 - ε -Neighborhood-Graph: Verbinde Knoten x_i mit x_j falls $\text{sim}(x_i, x_j) > \varepsilon$, und setze Kantengewicht auf 1.
 - k -Nearest-Neighbor-Graph: Verbinde Knoten x_i mit x_j falls x_i einer der k -nächsten Nachbarn von x_j ist oder/und x_j einer der k -nächsten Nachbarn von x_i ist, und setze Kantengewicht auf $\text{sim}(x_i, x_j)$.
 - Vollständiger Graph: Verbinde alle Knoten x_i mit x_j , und setze Kantengewicht auf $\text{sim}(x_i, x_j)$.

Graphentheoretische Verfahren

- Partitionierung des Graphen:
 - Kanten zwischen Clustern (Teilgraphen) haben geringe Gewichte (geringe inter-cluster similarity).
 - Kanten innerhalb eines Clusters haben hohe Gewichte (hohe intra-cluster similarity).

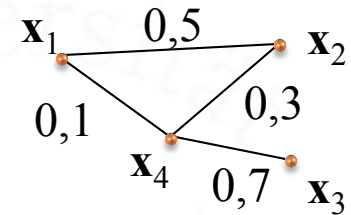
Graphentheoretische Verfahren

□ Repräsentation eines (ungerichteten) Graphen:

□ Adjazenzmatrix:

- Enthält Kantengewichte A_{ij} .

$$\mathbf{A} = \begin{bmatrix} 1 & 0,5 & 0 & 0,1 \\ 0,5 & 1 & 0 & 0,3 \\ 0 & 0 & 1 & 0,7 \\ 0,1 & 0,3 & 0,7 & 1 \end{bmatrix}$$



□ Knotengrad-Matrix:

$$\mathbf{D} = \begin{bmatrix} d_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & d_n \end{bmatrix}$$

$$d_i = \sum_{j=1}^n A_{ij}$$

□ Laplace-Matrix:

- Unnormalisiert:

$$\mathbf{L}_{un} = \mathbf{D} - \mathbf{A}$$

- Normalisiert (Random Walk):

$$\mathbf{L}_{rw} = \mathbf{I} - \mathbf{D}^{-1}\mathbf{A}$$

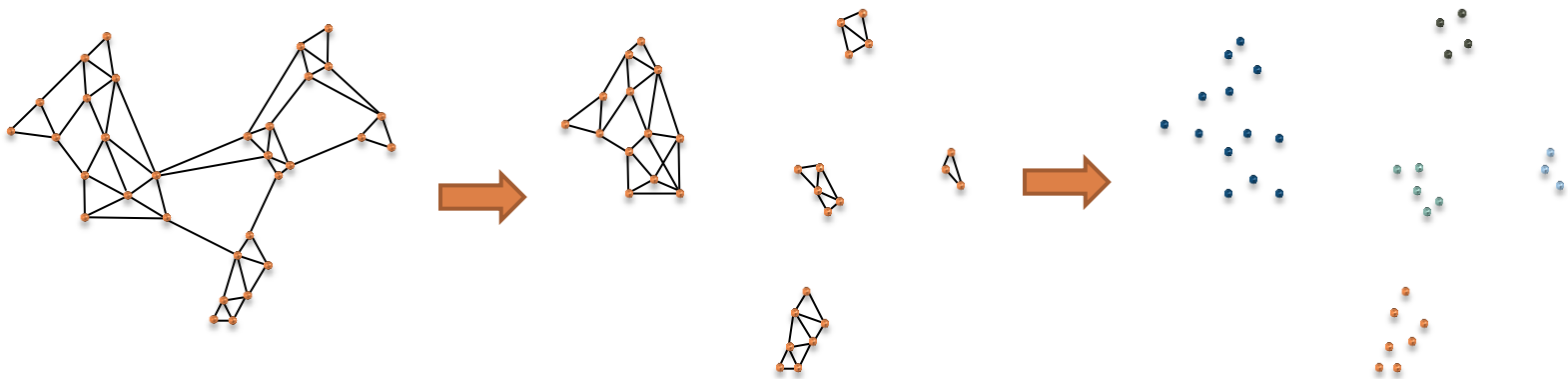
- Symmetrisch normalisiert:

$$\mathbf{L}_{sym} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$$

Graphentheoretische Verfahren

Spectral Clustering

- Eigenschaften der Laplace-Matrix L eines (ungerichteten) Graphen mit positiven Kantengewichten:
 - ▣ Anzahl zusammenhängender Teilgraphen = Anzahl Eigenwerte von L mit Wert 0.
 - ▣ Zugehörige (unnormierte) Eigenvektoren enthalten Einträge 0 und 1, und sind Indikatorvektoren der Teilgraphen.



Graphentheoretische Verfahren

Spectral Clustering



□ Algorithmus:

SpecClust(*Instanzen* \mathbf{x}_i , *Clusteranzahl* k)

Konstruiere Graph aus *Instanzen* \mathbf{x}_i

Berechne zugehörige Laplace-Matrix \mathbf{L}

Berechne die k Eigenvektoren $\mathbf{v}_i \in \mathbb{R}^n$ mit den k kleinsten
Eigenwerten

Setze $[\mathbf{x}'_1 \ \mathbf{x}'_2 \ \cdots \ \mathbf{x}'_n] = [\mathbf{v}_1 \ \mathbf{v}_2 \ \cdots \ \mathbf{v}_k]^T$

$\{\mathbf{r}_1, \dots, \mathbf{r}_n\} = \text{K-Means}(\text{Instanzen } \mathbf{x}'_i, \text{Clusteranzahl } k)$

RETURN $\{\mathbf{r}_1, \dots, \mathbf{r}_n\}$

Graphentheoretische Verfahren

Spectral Clustering

□ Vorteile:

- Cluster können beliebige Form haben.
- Einfach zu implementieren.
- Relativ schnell (falls Laplace-Matrix dünn besetzt).
- Meist hohe Qualität.

□ Nachteile:

- Nur lokales Optimum garantiert.
- Harte Zuweisung zu Clustern (nicht-probabilistisch).
- Anzahl Cluster muss vorgegeben werden.

Zusammenfassung

- Prototyp-Verfahren (z.B. K-Means):
 - Schnell, numerische Attribute, bekannte Clusteranzahl.
- Mischmodelle (z.B. Mixture of Gaussians):
 - Probabilistische Cluster-Zuweisung, numerische Attribute, unbekannte Clusteranzahl.
- Verlinkungsbasierte Verfahren (z.B. AGNES):
 - Iterativ, beliebige Attribute, unbekannte Clusteranzahl.
- Graphbasierte Verfahren (z.B. Spectral Clustering):
 - Beliebige Clusterform, oft gute Performance.