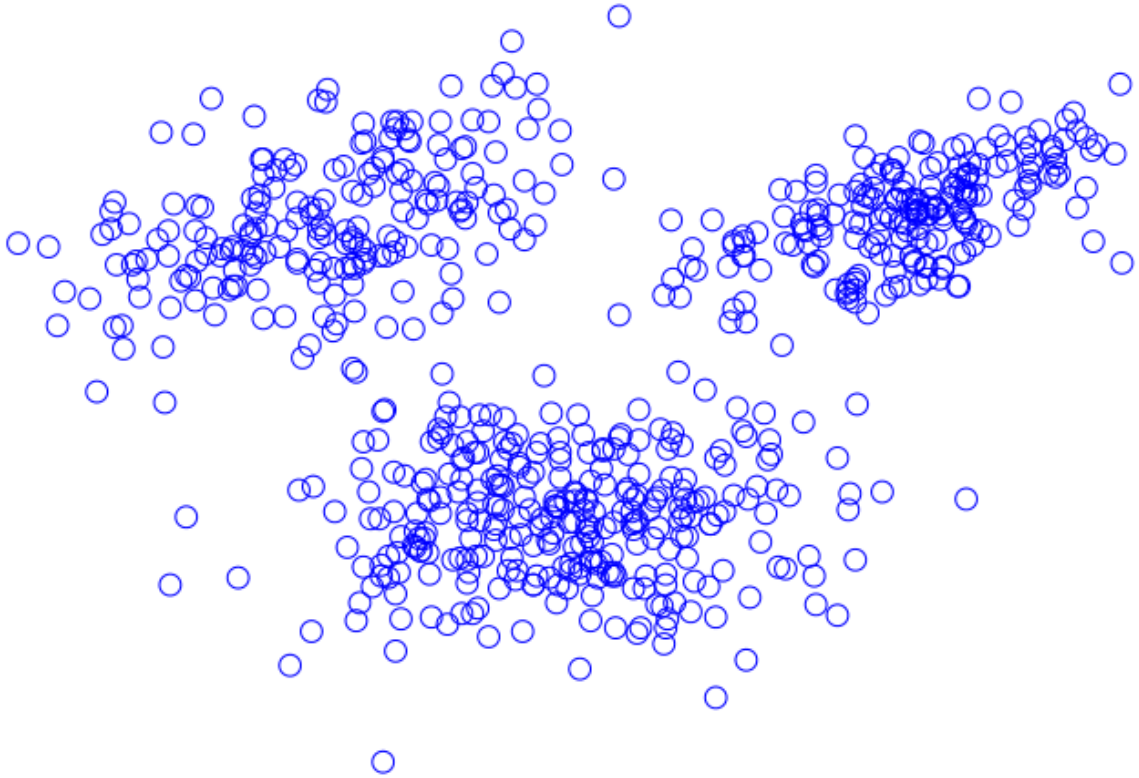
Universität Potsdam
Institut für Informatik
Lehrstuhl Maschinelles Lernen

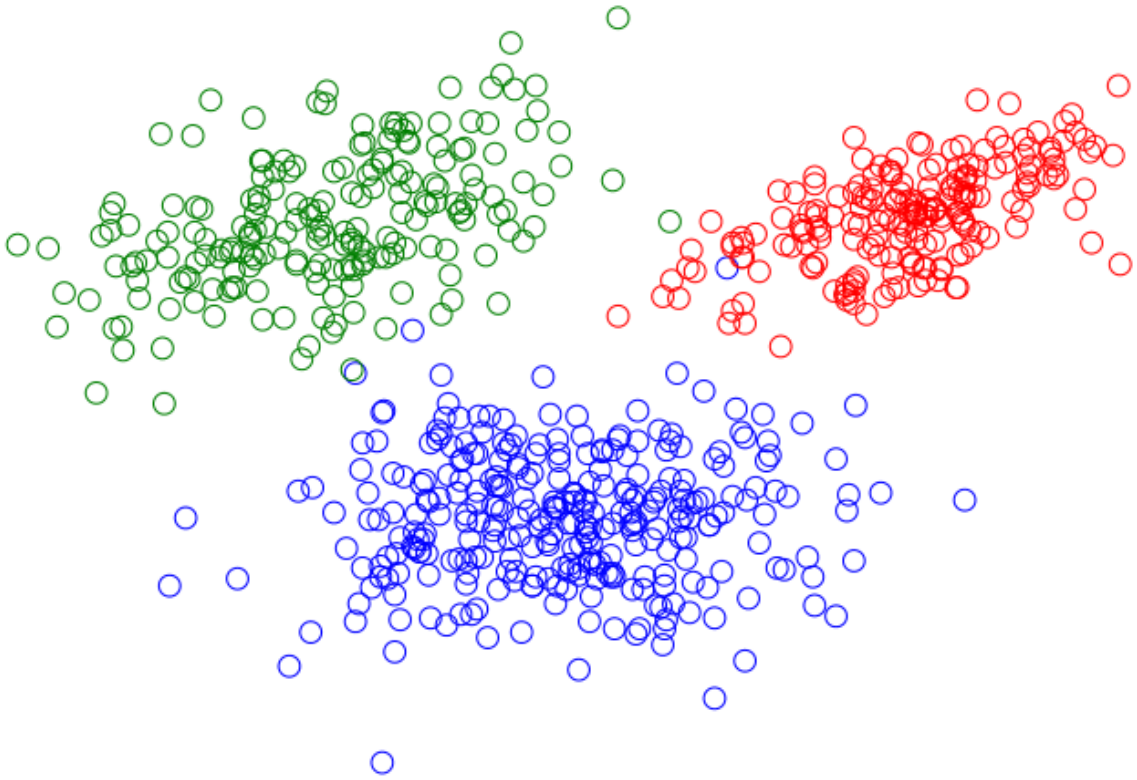
Maschinelles Lernen II
Clustering

Matthias Bussas / Niels Landwehr
Tobias Scheffer

Motivation



Motivation



Clustering

- Gegeben:
  - Objekte $V = \{x_1, ..., x_n\}$
  - Distanzfunktion $\text{dist}(x_i, x_j) \geq 0$ oder Ähnlichkeitsfunktion $w_{ij} = \text{sim}(x_i, x_j) \geq 0$
  - Erwartete Clusteranzahl k

- Ziel: Partition $P_1, ... P_k$, wobei $P_i \cap P_j = \varnothing, \bigcup_{i=1...n} P_i = V$ mit…
  - hoher intra-cluster-Ähnlichkeit
  - niedriger inter-cluster-Ähnlichkeit

Inter-Cluster Metriken

- Einfacher Abstand

$$d_{min}(P_i, P_j) = \min_{v \in P, w \in P_j} dist(v, w)$$

- Kompletter Abstand

$$d_{max}(P_i, P_j) = \max_{v \in P_i, w \in P_j} dist(v, w)$$

- Durchschnittsabstand

$$d_{mean}(P_i, P_j) = \frac{1}{|P_i||P_j|} \sum_{v \in P_i} \sum_{w \in Pj} dist(v, w)$$

- Abstand der Zentroide

$$d_{cent}(P_i, P_j) = dist\left(\frac{1}{|P_i|} \sum_{v \in P_i} v, \frac{1}{|P_j|} \sum_{v \in Pj} v \right)$$

Optimales Clustering

- Problem: Berechnung des globalen Optimum bezüglich der inter- und intra-Cluster-Ähnlichkeit ist NP – schwer.

- Approximation notwendig:
  - Heuristik (Hierarchisches Clustering)
  - Relaxation (Spectral Clustering)
  - EM-Algorithmus (nächste VL)

Überblick

- Hierarchisches Clustern
  - Bottom Up
  - Top Down
- Graph-basiertes Clustern
  - Ähnlichkeitsgraph
  - Minimaler Schnitt

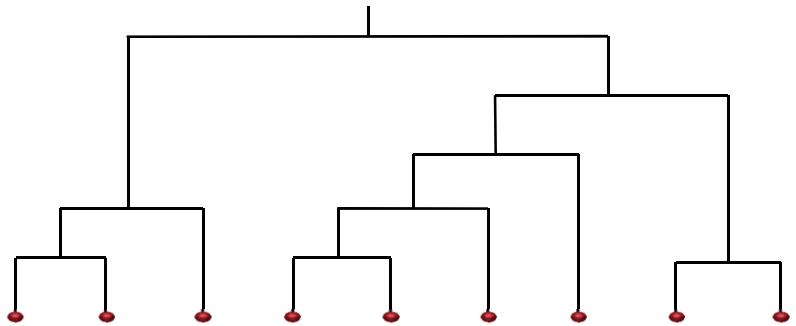
Überblick

- Hierarchisches Clustern
  - Bottom Up
  - Top Down
- Graph-basiertes Clustern
  - Ähnlichkeitsgraph
  - Minimaler Schnitt

Hierarchisches Clustern
Agnes (Algorithmus)

- Geg.: Objekte V, Inter-Cluster Metrik d
- Setze $C_0 = \{\{x\} \mid \forall x \in V\}$
- Solange unterschiedliche Cluster existieren
  - berechne min. Distanz über alle $c^v, c^w \in C_{i-1}$

    $$(s,t) = \arg\min_{v,w} d(c^v, c^w);\ D_i = \min_{v,w} d(c^v, c^w)$$
  - Setze $C_i = \{c^v \mid \forall v \neq s,t\} \cup \{c^s \cup c^t\}$
- Liefere $C_0, C_1, \ldots$ zurück



Hierarchisches Clustern
Agglomerative Coefficient

- Sei $m_k = d\left(c^s, \{x_k\}\right)$, wobei c^s das Cluster ist, mit dem x_k im i-ten Schritt verschmolzen wurde

$$C_i = \left\{c^v \mid \forall v \neq s,t\right\} \cup \left\{c^s \cup \{x_k\}\right\}$$

- Agglomerative Coefficient :

$$AC = \frac{1}{n}\sum_{i=1}^{n}\left(1 - \frac{m_i}{D_{final}}\right) \in [0,1]$$

- Ein Maß für die Qualität eines Clusterings
- Nicht geeignet um Datensätze unterschiedlicher Größe zu vergleichen

Überblick

- Hierarchisches Clustern
  - Bottom Up
  - Top Down
- Graph-basiertes Clustern
  - Ähnlichkeitsgraph
  - Minimaler Schnitt

Hierarchisches Clustern
Diana

- Bottom up: alle $\binom{n}{2}$ möglichen Fusionen werden betrachtet
- Top down: $2^{n-1}-1$ mögliche Splits

Hierarchisches Clustern
Diana (Algorithmus)

- Geg.: Objekte V, Inter-Cluster Metrik d
- Setze $C_0 = \{V\}$
- Solange mehr-elementige Cluster existieren
  - Bestimme Cluster mit höchsten Durchmesser

    $c = \arg\max_{c \in C_{i-1}} \max_{s,t \in c} d(s,t)$
  - Bestimme unähnlichstes Element

    $s = \arg\max_{v \in c} d(v, c \setminus \{v\})$ und setze $\bar{c} = \{s\}$
  - Solange $\max_{v \in c \setminus \bar{c}} D(v) > 0$, wobei $D(v) = d(v, c \setminus \bar{c}) - d(v, \bar{c})$
    - $t = \arg\max_{v \in c \setminus \bar{c}} D(v)$
    - $\bar{c} = \bar{c} \cup \{t\}$
  - Setze $C_i = (C_{i-1} \setminus \{c\}) \cup \{c \setminus \bar{c}\} \cup \{\bar{c}\}$
- Liefere $C_0, C_1, \ldots$ zurück

Hierarchisches Clustern
Divisive Coefficient

- Sei dia_i, der Durchmesser des Cluster aus dem das Objekt v_i zu letzt herausgelöst wurde (bis es einzeln war)

- Divisive Coefficient:

$$DC = \frac{1}{n}\sum_{i=1}^{n} \mathrm{dia}_i$$

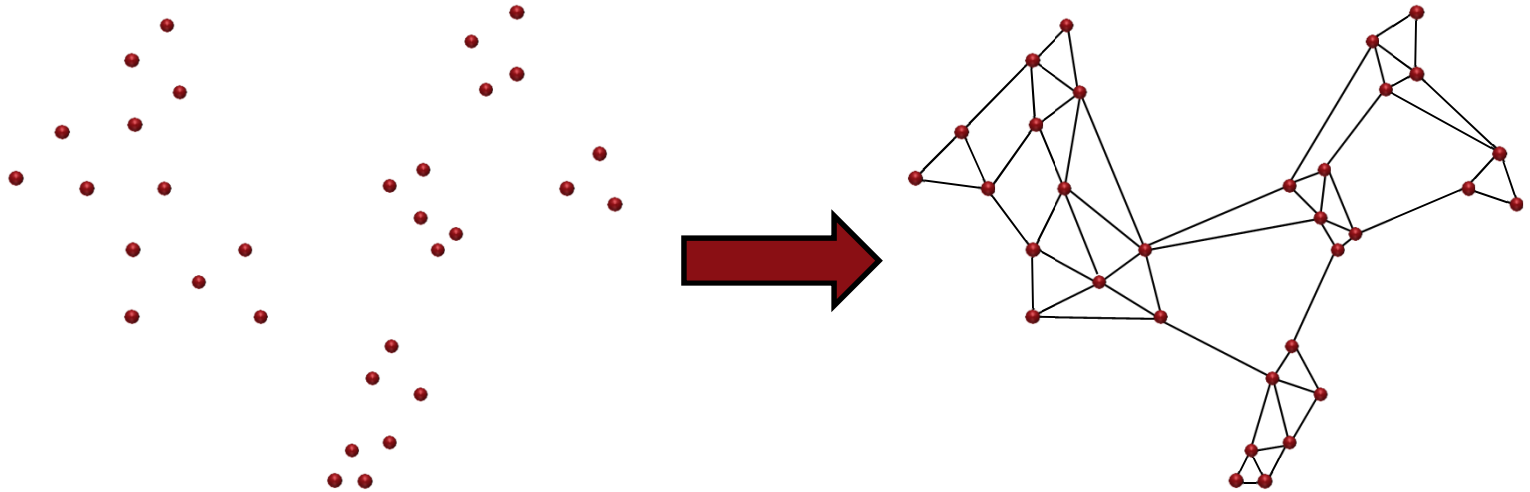
- Ein Maß für die Qualität eines Clusterings
- Nicht geeignet um Datensätze unterschiedlicher Größe zu vergleichen

Überblick

- Hierarchisches Clustern
  - Bottom Up
  - Top Down
- Graph-basiertes Clustern
  - Ähnlichkeitsgraph
  - Minimaler Schnitt

Graphen-basiertes Clustern
Ähnlichkeitsgraph

- Ähnlichkeiten zwischen Datenpunkten (Knoten) bilden gewichtete Kanten:



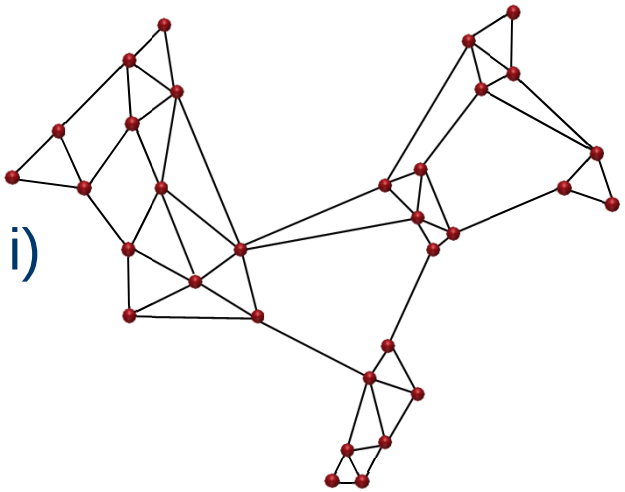
Graphen-basiertes Clustern
Ähnlichkeitsgraph

- Ähnlichkeiten zwischen Datenpunkten (Knoten) bilden gewichtete Kanten:

  - Vollständiger Graph: Kantengewichte = Ähnlichkeit

  - knn-Graph: Kante, wenn Knoten i (oder j) einer der k nächsten Nachbarn von j (bzw. i)

  - ε-Nachbarschaftsgraph: Kante, wenn $\mathrm{dist}\left(v_i, v_j\right) < \varepsilon$

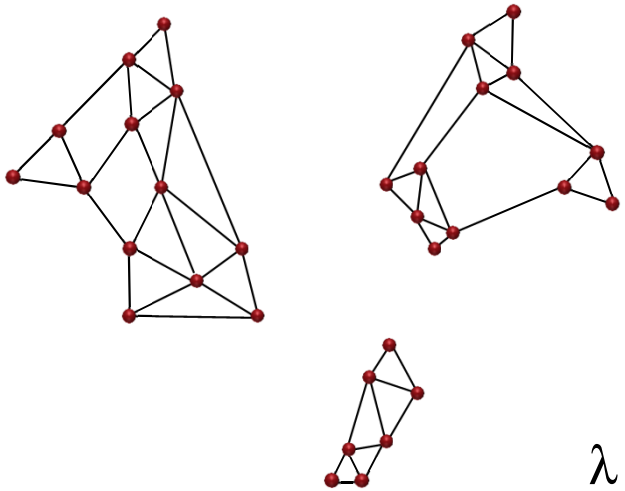


Graphen-basiertes Clustern
Definitionen

- Gewichtete Adjazenzmatrix $W = \begin{pmatrix} w_{11} & \cdots & w_{1n} \\ \vdots & \ddots & \vdots \\ w_{1n} & \cdots & w_{nn} \end{pmatrix}$
- Knotengrad-Matrix $\mathbf{D} = \begin{pmatrix} d_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & d_n \end{pmatrix}$ $\quad d_i = \sum_{j=1}^{n} w_{ij}$
- Laplace-Matrix
  - unnormalisiert $L_{un} = D - W$
  - Symmetrisch normalisiert $L_{sym} = I - D^{-1/2} W D^{-1/2}$

Beobachtung

- Zusammenhängende Teilgraphen…
  - entspricht Anzahl Eigenwerte von $\mathbf{L}$ mit Wert 0.
  - zugehörige (unnormierte) Eigenvektoren enthalten Indikatorvektoren der Teilgraphen.
  - Erkenntnis für schwach zusammenhäng. Teilgraphen?



$$\lambda_1 = \lambda_2 = \lambda_3 = 0$$

$$f_1 = (1,\ldots 1,0,\ldots 0,0,\ldots 0) / \sqrt{\#\text{Bsp. in } C_1}$$

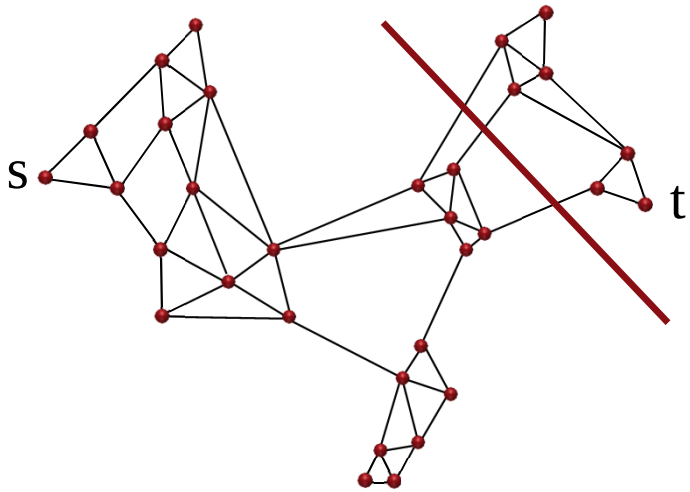
$$f_2 = (0,\ldots 0,1,\ldots 1,0,\ldots 0) / \sqrt{\#\text{Bsp. in } C_2}$$

$$f_3 = (0,\ldots 0,0,\ldots 0,1,\ldots 1) / \sqrt{\#\text{Bsp. in } C_3}$$

$$\lambda = f^T L_{un} f = f^T D f - f^T W f = \frac{1}{2}\sum_{i,j=1}^{n} w_{i,j}\left(f_i - f_j\right)^2$$

Minimaler Schnitt
Spezialfall k=2

- Betrachten Ähnlichkeitsgraphen mit zwei unterschiedlichen ausgezeichneten Knoten $s, t \in V$

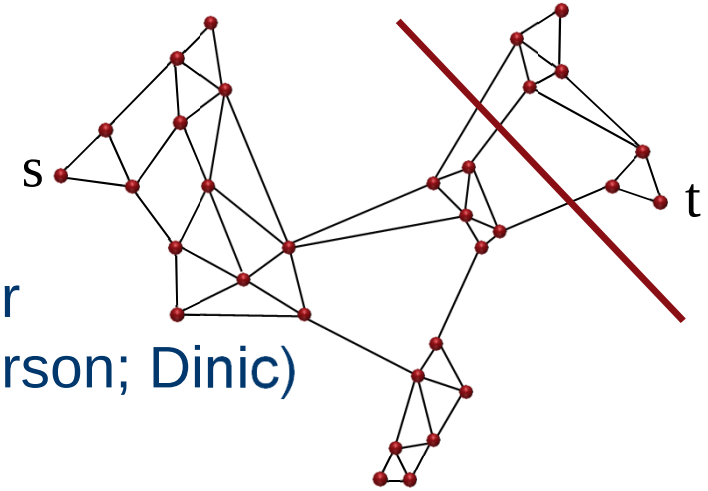


- Ein s-t-Schnitt ist eine Partitionierung der Knoten, wobei $s \in P$ und $t \in \overline{P} = V \setminus P$ mit

$$Cut^{s,t}(P) = \sum_{v_i \in P, v_j \in \overline{P}} w_{ij}$$

Minimaler Schnitt
Spezialfall k=2

- Der minimale s-t-Schnitt ist $P^* = \arg\min_{P \subset V} Cut^{s,t}(P)$
  - Problem ist in polynomieller Laufzeit lösbar (Ford/Fulkerson; Dinic)



- Der minimale Schnitt ist der minimale s-t-Schnitt über alle s-t-Schnitte: $Cut(P) = \sum_{v_i \in P, v_j \in \bar{P}} w_{ij}$
  - Problem ist in polynomieller Laufzeit lösbar

$$\mathcal{O}\left(nm + n^2 \log n\right)$$

Minimaler Schnitt
Balanzierung

- Problem: MinCut-Lösung separiert häufig einzelne Knoten

Minimaler Schnitt
Balanzierung

- Problem: MinCut-Lösung separiert häufig einzelne Knoten

- Balanzierung:

$$\mathrm{RatioCut}(P) = \frac{\mathrm{Cut}(P)}{|P|} + \frac{\mathrm{Cut}(\bar{P})}{|\bar{P}|}$$ wobei $|P|$ Anzahl der Knoten in P

$$\mathrm{Ncut}(P) = \frac{\mathrm{Cut}(P)}{\mathrm{vol}(P)} + \frac{\mathrm{Cut}(\bar{P})}{\mathrm{vol}(\bar{P})}$$ wobei $$\mathrm{vol}(P) = \sum_{v_i \in P} d_i$$

- Balanziertes MinCut-Problem ist NP-hart

Minimaler Schnitt
Balanzierung

- Lemma 1: Sei $f_i = \begin{cases} \sqrt{|\bar{P}| / |P|} & , \text{wenn } v_i \in P \\ -\sqrt{|P| / |\bar{P}|} & , \text{sonst} \end{cases}$ dann gilt

$$|V| \cdot \text{RatioCut}(P) = f^T L_{un} f$$

- Lemma 2: Sei $f_i = \begin{cases} \sqrt{\text{vol}(\bar{P}) / \text{vol}(P)} & , \text{wenn } v_i \in P \\ -\sqrt{\text{vol}(P) / \text{vol}(\bar{P})} & , \text{sonst} \end{cases}$ dann gilt

$$\text{vol}(V) \cdot \text{NCut}(P) = f^T L_{sym} f$$

Spectral-Clustering (unnormalisiert)
Relaxation

- RatioCut

$$\min_{P \subset V} f^T L f, \text{ wobei } \sum_{i=1}^{n} f_i = 0, \sum_{i=1}^{n} f_i^2 = n$$

Spectral-Clustering (unnormalisiert)
Relaxation

- RatioCut

f_i kann nur 2 Werte annehmen

$$\min_{P \subset V} f^T L f, \text{ wobei } \sum_{i=1}^{n} f_i = 0, \sum_{i=1}^{n} f_i^2 = n$$

$$f_i = \begin{cases} \sqrt{|\bar{P}| / |P|} & , \text{wenn } v_i \in P \\ -\sqrt{|P| / |\bar{P}|} & , \text{sonst} \end{cases}$$

Spectral-Clustering (unnormalisiert)

Relaxation

NP-hart

- RatioCut

$$\min_{P \subset V} f^T L f, \text{ wobei } \sum_{i=1}^{n} f_i = 0, \sum_{i=1}^{n} f_i^2 = n$$

Eigenwertproblem

- (Unnormalisiertes) Spectral-Clustering

$$\min_{f \in \mathbb{R}^n} f^T L f, \text{ wobei } \sum_{i=1}^{n} f_i = 0, \sum_{i=1}^{n} f_i^2 = n$$

Spectral-Clustering (unnormalisiert)

Relaxation

NP-hart

- RatioCut

$$\min_{P \subset V} f^T L f, \text{ wobei } \sum_{i=1}^{n} f_i = 0, \sum_{i=1}^{n} f_i^2 = n$$

Eigenwertproblem

- (Unnormalisiertes) Spectral-Clustering

$$\min_{f \in \mathbb{R}^n} f^T L f, \text{ wobei } \sum_{i=1}^{n} f_i = 0, \sum_{i=1}^{n} f_i^2 = n$$

- Diskretisierung: $\text{sign}(f_i)$

Spectral-Clustering (unnormalisiert)
Verallgemeinerung auf k>2

- $$\mathrm{Cut}(P_1,\ldots P_k) = \frac{1}{2}\sum_{i=1\ldots k} \mathrm{Cut}\left(P_i\right)$$

$$\mathrm{RatioCut}(P_1,\ldots P_k) = \frac{1}{2}\sum_{i=1\ldots k} \mathrm{RatioCut}\left(P_i\right)$$

$$\mathrm{Ncut}(P_1,\ldots P_k) = \frac{1}{2}\sum_{i=1\ldots k} \mathrm{Ncut}\left(P_i\right)$$

Spectral-Clustering (unnormalisiert)
Verallgemeinerung auf k>2

- $$\mathrm{Cut}(P_1,\ldots P_k) = \frac{1}{2}\sum_{i=1\ldots k} \mathrm{Cut}(P_i)$$

$$\mathrm{RatioCut}(P_1,\ldots P_k) = \frac{1}{2}\sum_{i=1\ldots k} \mathrm{RatioCut}(P_i)$$

$$\mathrm{Ncut}(P_1,\ldots P_k) = \frac{1}{2}\sum_{i=1\ldots k} \mathrm{Ncut}(P_i)$$

- $$f_i = \begin{cases} \sqrt{|\overline{P}|/|P|} & , \text{wenn } v_i \in P \\ -\sqrt{|P|/|\overline{P}|} & , \text{sonst} \end{cases} \Rightarrow F_{ij} = \begin{cases} \sqrt{1/|P_j|} & , \text{wenn } v_i \in P_j \\ 0 & , \text{sonst} \end{cases}$$

- $$\mathrm{RatioCut}(P_1,\ldots P_k) = \mathrm{Tr}\left(F^T L F\right)$$

Spectral-Clustering (unnormalisiert)

Relaxierung (k>2)

NP-hart

- RatioCut

$$\min_{P_1,\ldots,P_k} \mathrm{Tr}\left(F^T L F\right), \text{ wobei } F^T F = I$$

Eigenwertproblem

- (Unnormalisiertes) Spectral-Clustering

$$\min_{F \in \mathbb{R}^{n \times k}} \mathrm{Tr}\left(F^T L F\right), \text{ wobei } F^T F = I$$

Spectral-Clustering (unnormalisiert)

Relaxierung (k>2)

NP-hart

- RatioCut

$$\min_{P_1,\ldots,P_k} \mathrm{Tr}\left(F^T L F\right), \text{ wobei } F^T F = I$$

Eigenwertproblem

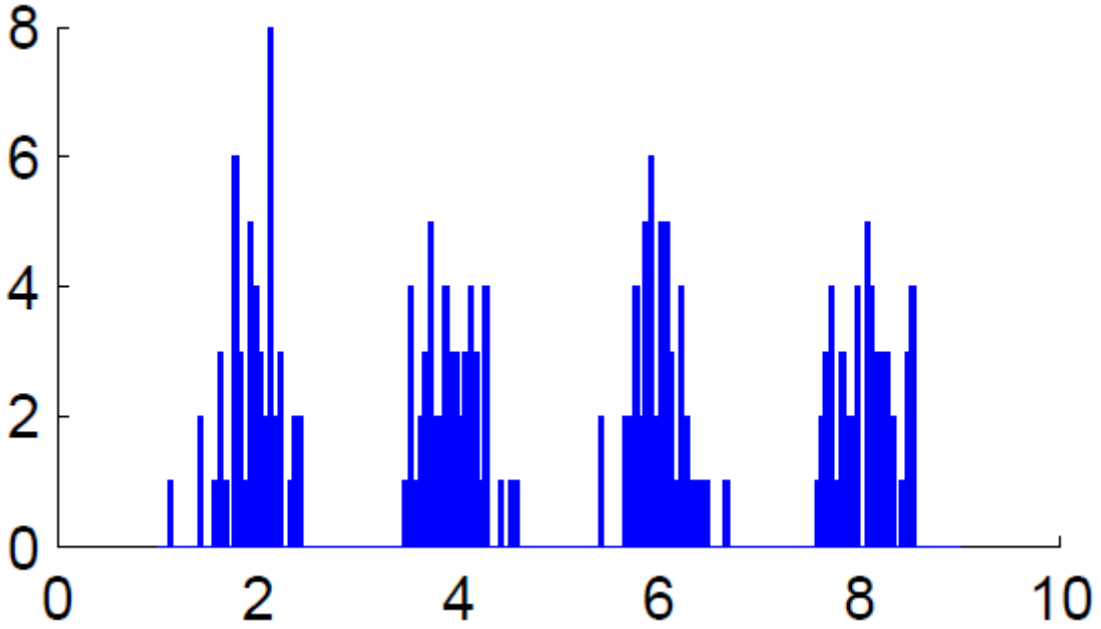
- (Unnormalisiertes) Spectral-Clustering

$$\min_{F \in \mathbb{R}^{n \times k}} \mathrm{Tr}\left(F^T L F\right), \text{ wobei } F^T F = I$$

- Diskretisierung: Clustern auf Basis der Vektoren F_i

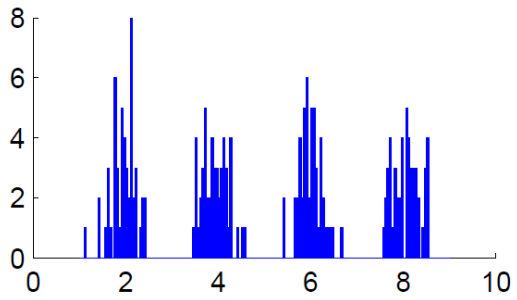
Spectral-Clustering
Beispiel

- Daten: Mixture of gaussian

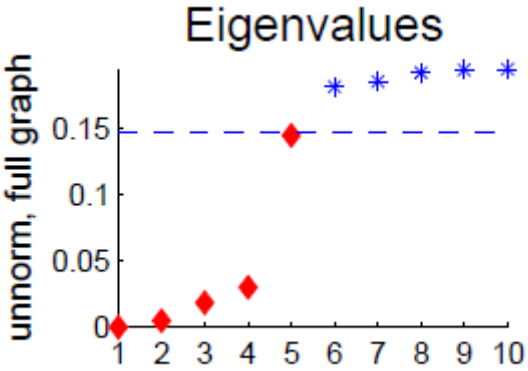


Spectral-Clustering
Beispiel

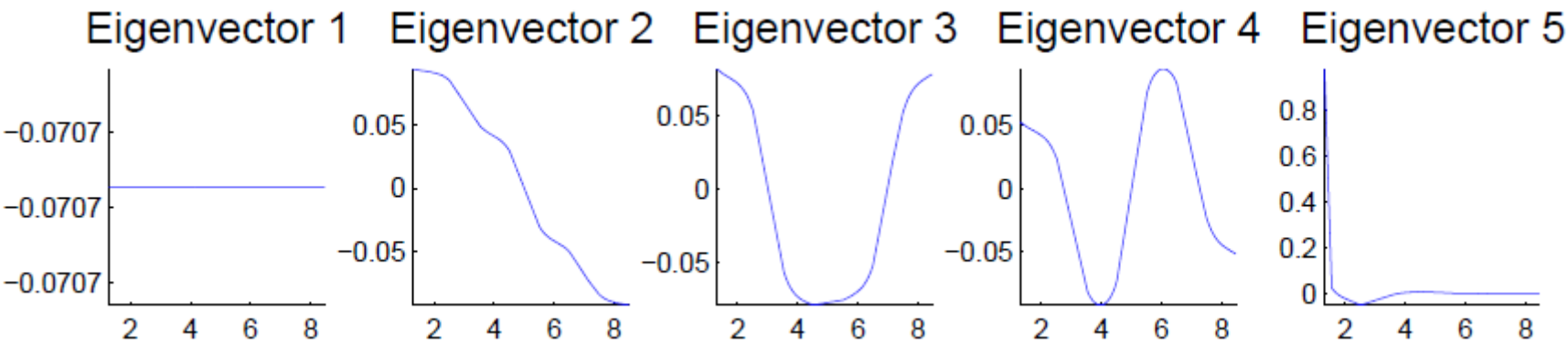
- sim: RBF mit $\sigma = 1$

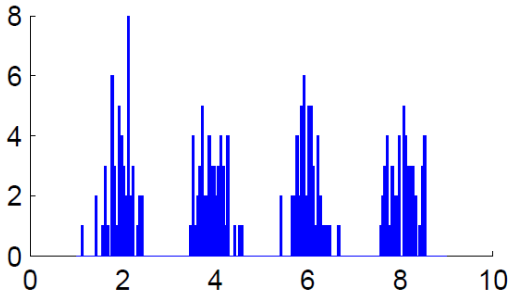


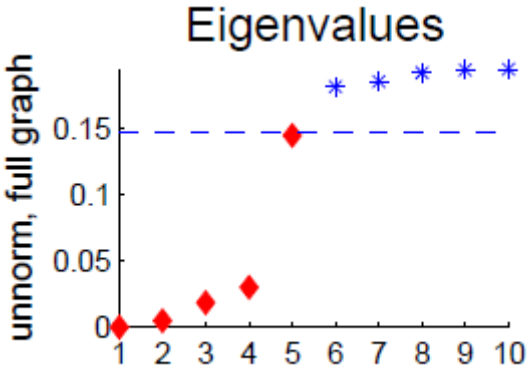
- Eigenwerte der zugehörigen Laplacematrix (fully connected Graph)



Spectral-Clustering
Beispiel





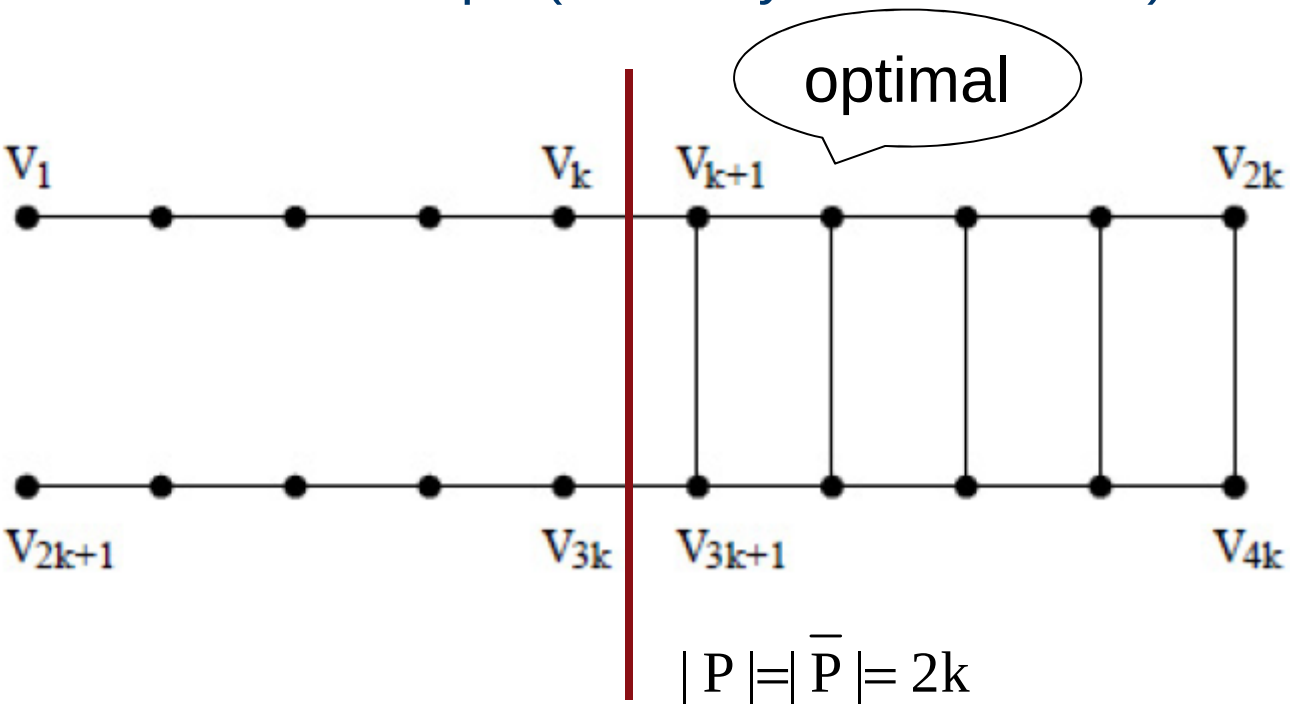


Spectral-Clustering (unnormalisiert)
Algorithmus

- Geg.: Adjazenzmatrix $W \in \mathbb{R}_{\geq 0}^{n \times n}$, Clusteranzahl k
- Berechne zugehörige Laplacematrix L_{un}
- Berechne die kleinsten k Eigenvektoren $u_i \in \mathbb{R}^n$ von L_{un}
- Setze $\begin{pmatrix} - & x_1 & - \\ & \vdots & \\ - & x_n & - \end{pmatrix} = \begin{pmatrix} | & & | \\ u_1 & \dots & u_k \\ | & & | \end{pmatrix}$
- Berechne Cluster C_j aus Datenpunkte x_i
- Liefere C_j zurück

Approximationsgüte
Balanzierte Schnitte

- Polynomieller Algorithmus mit konstanter Approximationsgüte existiert nicht
  - Cockroach Graph (Guattery & Miller 1998)

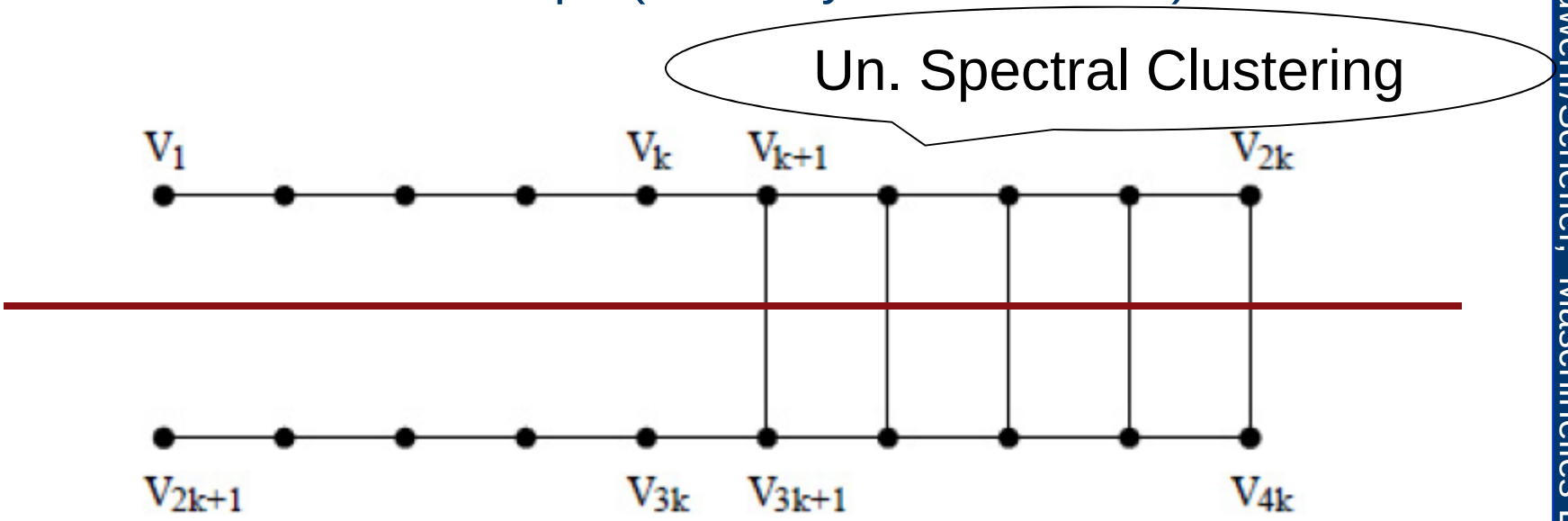


$$|P| = |\bar{P}| = 2k$$

$$\mathrm{cut}\left(P, \bar{P}\right) = 2$$

Approximationsgüte
Balanzierte Schnitte

- Polynomieller Algorithmus mit konstanter Approximationsgüte existiert nicht
  - Cockroach Graph (Guattery & Miller 1998)



$$|P| = |\bar{P}| = 2k$$

$$\mathrm{cut}\left(P, \bar{P}\right) = k$$

Anmerkungen

- Ncut führt zum verallgemeinerten Eigenvektorproblem (norm. Spectral clustering)

- Quelle:
  - H. Zha et al.: Spectral Relaxation for K-means Clustering; 2001
  - U. von Luxburg: A Tutorial on Spectral Clustering; 2007