Vorlesung Maschinelles Lernen

Cluster Analyse

Katharina Morik

LS 8 Künstliche Intelligenz Fakultät für Informatik
Technische Universität Dortmund

6.1.2009

Gliederung

Lernaufgabe Clustering

Gegeben

- eine Menge $\mathcal{T} = \{\vec{x_1}, ..., \vec{x_N}\} \subset X$ von Beobachtungen,
- eine Anzahl K zu findender Gruppen $C_1, ..., C_K$,
- eine Abstandsfunktion $d(\vec{x}, \vec{x'})$ und
- eine Qualitätsfunktion.

Finde

- Gruppen $C_1, ..., C_K$, so dass
- alle $\vec{x} \in X$ einer Gruppe zugeordnet sind und
- die Qualitätsfunktion optimiert wird: Der Abstand zwischen Beobachtungen der selben Gruppe soll minimal sein; der Abstand zwischen den Gruppen soll maximal sein.

Bild

Der Abstand wurde zum Cluster-Zentrum gemessen. Dadurch ergibt sich der grüne Punkt neben den roten.

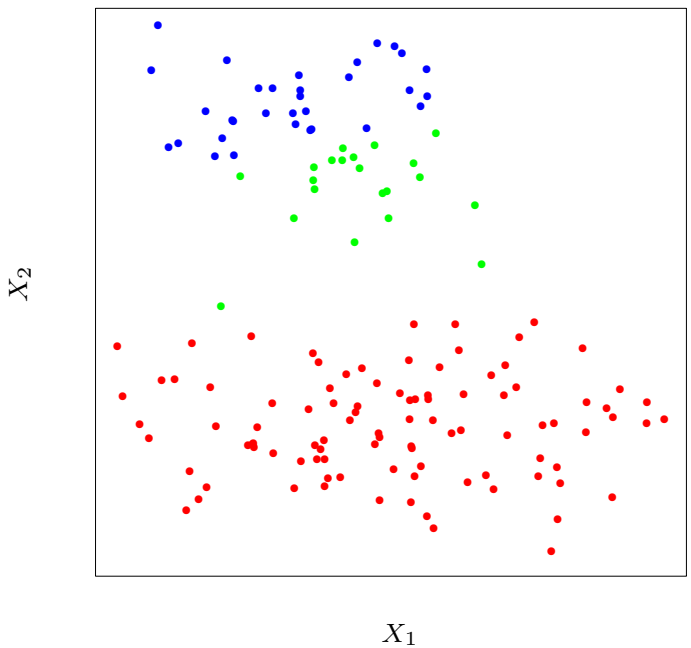


Figure 14.4: *Simulated data in the plane, clustered into three classes (represented by red, blue and green), by the K-means clustering algorithm*

- Könnte ein besseres Abstandsmaß den grünen Punkt dem roten Cluster zuweisen?
- Wenn nicht nur ein Punkt als Repräsentation eines Clusters gewählt wird, würde das Clustering dann besser?
- Wie kann man die Cluster verständlich beschreiben?
- Wäre $K = 2$ besser gewesen?

Die Probleme der Cluster-Analyse

1. Bestimmung des Abstandsmaßes
2. Formulierung des Optimierungsproblems
3. Repräsentation der Cluster
4. Bestimmung von K

Bestimmung des Abstandsmaßes

- Ähnlichkeitsmaße haben wir schon bei kNN gesehen.
- Im Allgemeinen ist der Abstand invers zur Ähnlichkeit:

$$D(\vec{x_1}, \vec{x_2}) = 1 - Sim(\vec{x_1}, \vec{x_2})$$

- Man kann aber irgendeine geeignete monoton absteigende Funktion zur Überführung der Ähnlichkeiten in Abstände wählen.

sim: Ähnlichkeit für einzelne Attribute (Erinnerung)

Numerische Attribute: Sei max_j der höchste Wert von X_j und min_j der niedrigste, sei x_{ij} der Wert des j-ten Attributs in der i-ten Beobachtung, dann ist die normalisierte Ähnlichkeit:

$$sim_j(x_{1j}, x_{2j}) = 1 - \frac{|x_{1j} - x_{2j}|}{max_j - min_j}$$

Nominale Attribute: Ganz einfach:

$$sim_j(x_{1j}, x_{2j}) = \begin{cases} 1 & falls \quad x_{1j} = x_{2j} \\ 0 & sonst \end{cases}$$

d: Abstand für einzelne Attribute

Numerische Attribute: Ohne Normalisierung durch $max_j - min_j$ ist der Betrag der Differenz:

$$d_j(x_{ij}, x_{i'j}) = \mid x_{ij} - x_{i'j} \mid$$

Der quadratische Abstand zwischen Beobachtungen x_i und x'_i bezüglich des Merkmals X_j gewichtet große Abstände stärker als kleine:

$$d_j(x_{ij}, x_{i'j}) = (x_{ij} - x_{i'j})^2 \quad (1)$$

Nominale Attribute: Man kann für jede Variable X_j mit M Attributwerten eine $M \times M$ Abstandsmatrix angeben oder einfach:

$$d_j(x_{1j}, x_{2j}) = \begin{cases} 1 & falls \quad x_{1j} \neq x_{2j} \\ 0 & sonst \end{cases}$$

Sim: Ähnlichkeit der Beobachtungen als Kombination der Attributähnlichkeiten

Im einfachsten Fall mitteln wir die Einzelähnlichkeiten:

$$Sim(\vec{x_1}, \vec{x_2}) = \frac{1}{p} \sum_{j=1}^{p} sim(x_{1j}, x_{2j})$$

Die Korrelation verwendet das Mittel $\bar{x}_i$ über allen p Variablen:

$$Sim(\vec{x_1}, \vec{x_2}) = \frac{\sum_{j=1}^{p}(x_{1j} - \bar{x}_1)(x_{2j} - \bar{x}_2)}{\sqrt{\sum_{j=1}^{p}(x_{1j} - \bar{x}_1)^2 \sum_{j=1}^{p}(x_{2j} - \bar{x}_2)^2}} \quad (2)$$

Vielleicht sind einige Attribute wichtiger als andere?

$$Sim(\vec{x_1}, \vec{x_2}) = \frac{\sum_{j=1}^{p} w_j sim(x_{1,j}, x_{2,j})}{\sum_{j=1}^{p} w_j}$$

Wie bestimmt man w_j?

Abstandsmaß

- Verwendet wird eine $N \times N$ Matrix **D** für die N Beobachtungen, wobei d_{12} der Eintrag für $D(\vec{x_1}, \vec{x_2})$ ist.
- Die Matrix hat keine negativen Einträge.
- Die Diagonale der Matrix: $d_{ii} = 0$
- Der Abstand soll symmetrisch sein – falls nicht: $(\mathbf{D} + \mathbf{D}^T)/2$.

D: Abstand der Beobachtungen als Kombination der Attributabstände

- Gewichteter Durchschnitt:

$$D(\vec{x_1}, \vec{x_2}) = \sum_{j=1}^{p} w_j d_j(x_{1j}, x_{2j}); \sum_{j=1}^{p} w_j = 1 \quad (3)$$

- Bei quadratischem Abstand d_{12} ergibt sich:

$$D(\vec{x_1}, \vec{x_2}) = \sum_{j=1}^{p} w_j (x_{1j} - x_{2j})^2 \quad (4)$$

- Man kann die Korrelation (Gleichung 2) verwenden:

$$1 - Sim(\vec{x_1}, \vec{x_2}) \quad (5)$$

Einfluss einer Variablen auf das Clustering

- Wenn für alle Variablen $w_j = 1$ wäre, hätten doch nicht alle Variablen den gleichen Einfluss auf das Clustering!
- Der Einfluss einer Variable X_j richtet sich vielmehr nach ihrer durchschnittlichen Unähnlichkeit:

$$\bar{d}_j = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{i'=1}^{N} d_j(x_{ij}, x_{i'j}) \tag{6}$$

- Beim gewichteten quadratischen Abstand

$$\bar{d}_j = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{i'=1}^{N} (x_{ij} - x_{i'j})^2 = 2 \cdot var_j \tag{7}$$

  wobei var_j die anhand der Beobachtungmenge $\mathcal{T}$ geschätzte Varianz von X_j ist.
- Der Einfluss einer Variablen auf das Clustering richtet sich also nach der Varianz! Der relative Einfluss ist $w_j \, \bar{d}_j$.

Beispiel für Nachteil gleichen Einflusses der Variablen

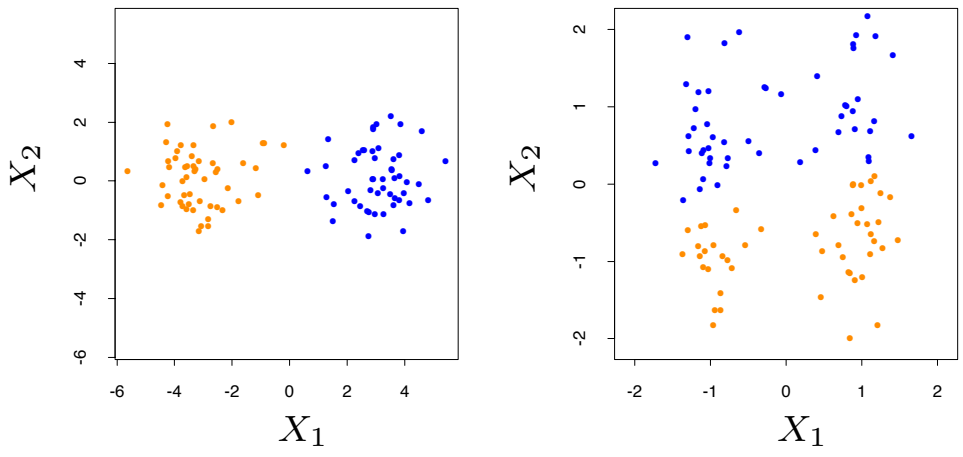


Figure 14.5: *Simulated data: on the left, K-means clustering (with K=2) has been applied to the raw data. The two colors indicate the cluster memberships. One the right, the features were first standardized before clustering. This is equivalent to using feature weights* $1/[2 \cdot \mathrm{var}(X_j)]$. *The standardization has obscured the two well-separated groups. Note that each plot uses the same units in the horizontal and vertical axes.*

- Alle Variablen haben den selben Einfluss auf das Clustering, wenn $w_j \sim 1/\bar{d}_j$.
- Wenn als Gewichte $w_j = \frac{1}{2 \cdot var_j}$ gewählt wird, hat man den Einfluss der Varianz ausgeschaltet und erhält manchmal keine gute Separierung mehr.

Es hängt von der Anwendung ab, wie man w_j wählt!

Für eine Anwendung kann man vor dem Clustern

1. gar nichts tun, d.h. die Rohdaten ohne Gewichtung und ohne Normalisierung clustern,
2. die Rohdaten normalisieren (Werte im selben Wertebereich, z.B. $[0, 1]$, oder jeweils $max_j - min_j$ in den Abständen),
3. $\bar{d}_j$ für jedes Merkmal berechnen (Varianz-Gleichung 7),
4. die Rohdaten standardisieren, so dass alle Variablen den gleichen Einfuss haben,
5. Gewichte w_j, die dem Sachbereich entsprechen könnten oder dem Clustering-Ziel, direkt auf die Daten als Transformation der Eingabe anzuwenden. (Implizites w_j!)
6. Dann die Ergebnisse vergleichen!

Qualitätsfunktionen

Sei die Anzahl K der Cluster gegeben und jedes Cluster durch eine ganze Zahl $k \in \{1, 2, ..., K\}$ eindeutig ausgezeichnet. Die Abbildung $C(i) = k$ weist der i-ten Beobachtung das k-te Cluster zu.

Innerer Abstand Within: Minimiert werden soll der Abstand innerhalb eines Clusters C:

$$W(C) = \frac{1}{2} \sum_{k=1}^{K} \sum_{C(i)=k} \sum_{C(i')=k} D(\vec{x_i}, \vec{x_{i'}}) \tag{8}$$

Zwischenunähnlichkeit Between: Maximiert werden soll der Abstand zwischen Clustern:

$$B(C) = \frac{1}{2} \sum_{k=1}^{K} \sum_{C(i)=k} \sum_{C(i')\neq k} d(x_i, x_{i'}) \tag{9}$$

Optimierungsproblem der Cluster-Analyse

- Gegeben die Summe aller Abstände $T = \frac{1}{2} \sum_{i=1}^{N} \sum_{i'=1}^{N} d_{ii'}$, ergänzen sich $W(C) + B(C) = T$, so dass die Minimierung von $W(C)$ der Maximierung von $B(C)$ entspricht.
- Man hat so nur ein Optimierungsproblem.
- Sei $\bar{x}_k = (\bar{x}_{1k}, ..., \bar{x}_{pk})$ der Vektor der Mittelwerte aller Variablen in Cluster k und $N_k = \sum_{i=1}^{N} I(C(i) = k)$, dann ist das Optimierungsproblem:

$$C^* = min_C \sum_{k=1}^{K} N_k \sum_{C(i)=k} \| \vec{x_i} - \bar{x_k} \|^2 \qquad (10)$$

Iteratives Lösen des Optimierungsproblems – K-Means

Algorithmus K-Means$(\mathcal{T}, K)$

1. Wähle K Beobachtungen aus $\mathcal{T}$ zufällig als Mittelpunkte $\vec{m_1}, ..., \vec{m_K}$ von Clustern aus.
2. Berechne das Clustering anhand der Mittelpunkte:

$$C(i) = argmin_{1 \leq k \leq K} \parallel \vec{x_i} - \vec{m_k} \parallel^2 \tag{11}$$

3. Berechne die Mittelpunkte entsprechend $C(i)$:

$$\vec{m_i} := argmin_m \sum_{i=1}^{N} \parallel \vec{x_i} - \vec{m} \parallel^2 \tag{12}$$

4. Wiederhole Schritt 2 und 3 bis die Zuweisungen sich nicht mehr ändern. Gib zurück $C(1), ..., C(K)$.

K-Means im Bild

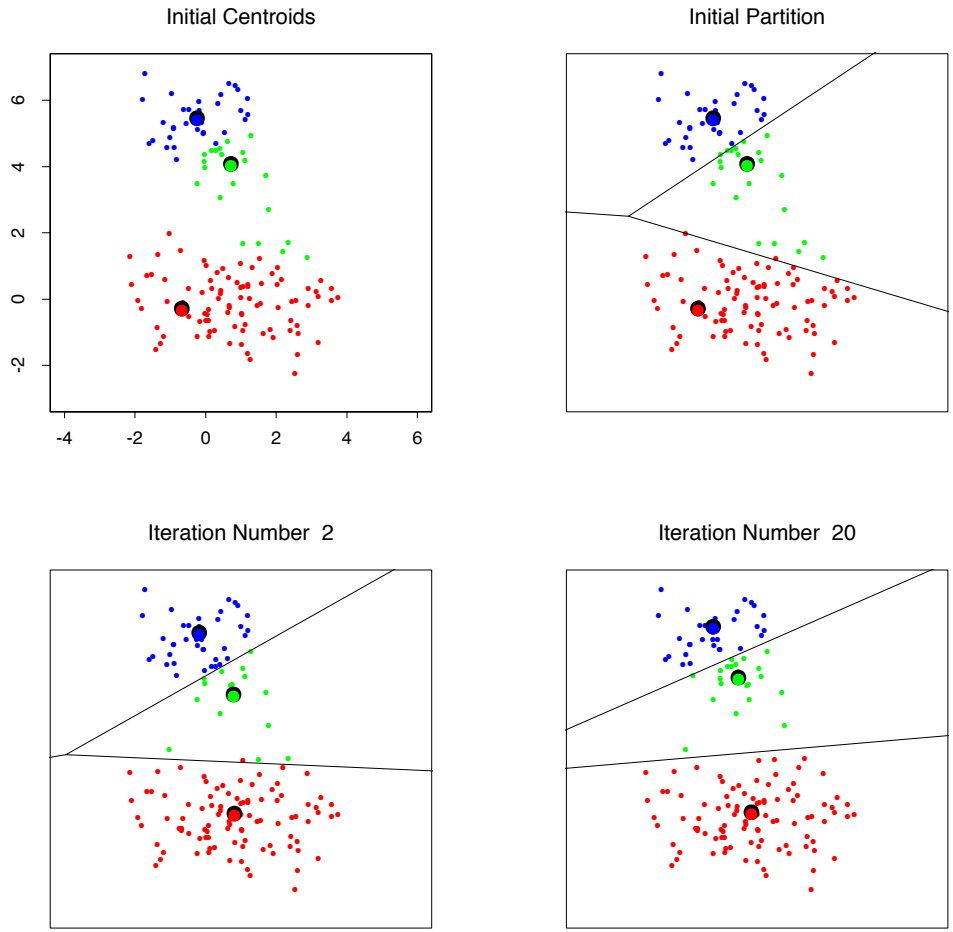


Figure 14.6: *Successive iterations of the K-means clustering algorithm for the simulated data of Fig-*

Eigenschaften von K-Means

- K-Means ist für numerische Variablen gemacht.
- Als Abstandsmaß wird der quadratische Euklidsche Abstand genutzt.
  - Den größten Einfluss haben Datenpunkte mit dem größten Abstand.
  - Das Verfahren ist daher anfällig für Ausreißer.
- Der Aufwand ist proportional zu $N \cdot K$.
  - Für jedes Cluster wird der Mittelpunkt berechnet anhand der zugeordneten Beobachtungen. Ein Cluster ist also nur durch einen Punkt repräsentiert.
  - Für alle Beobachtungen wird der Abstand zu den K Mittelpunkten berechnet.
- Es kann sein, dass die Lösung von K-Means nicht optimal ist (lokales Optimum).

Repräsentation der Cluster

- K-Means repräsentiert ein Cluster durch einen errechneten Punkt. Dies ist effizient.
- K-Medoid wählt eine Beobachtung als Repräsentation eines Clusters. Dafür muss über allen Punkten optimiert werden – ineffizient.
- Rajeev Rastogi hat vorgeschlagen einige Punkte als Repräsentation eines Clusters zu wählen (well scattered points).
- Oft möchte man eine interpretierbare Charaktierisierung der Cluster haben.
  - Aufgabe des labeling: finde eine (logische) Charakterisierung der Cluster. Man betrachtet die Cluster als Klassen und wendet z.B. Entscheidungsbaumlernen an.
  - Ryszard Michalski hat ein logisches Cluster-Verfahren vorgeschlagen, die Star-Methode (AQ-Algorithmus), bei dem direkt über den nominalen Werten der Beobachtungen gearbeitet wird.

Bestimmung der vorgegebenen Mittelpunkte

Die Lösung von K-Means hängt von den gewählten Start-mittelpunkten ab. Dafür gibt es mindestens zwei Auswege:

- Mehrfach mit zufällig gewählten Startmittelpunkten den Algorithmus starten!
- Optimierungskriterium

$$min_{C,\{m_k\}_1^K} \sum_{k=1}^{K} N_k \sum_{C(i)=k} \| \vec{x_i} - m_k \|^2$$

Für $k = 1, ..., K$:
Wähle einen Mittelpunkt i_k so, dass das Kriterium minimiert wird gegeben $i_1, ..., i_{k-1}$.
Starte K-Means mit den so gefundenen K Mittelpunkten.

Wie viele Cluster sollen gebildet werden?

- Vielleicht geht aus der Anwendung hervor, wie viele Cluster nötig sind. Z.B. sollen Kunden so auf K Vertriebsmitarbeiter aufgeteilt werden, dass ein Mitarbeiter ähnliche Fälle bearbeitet.
- Oft soll K^* anhand der Daten so ermittelt werden, dass die Clustering-Qualität optimiert wird (Gleichung 8).

$$W(C) = \frac{1}{2} \sum_{k=1}^{K} \sum_{C(i)=k} \sum_{C(i')=k} D(\vec{x_i}, \vec{x_{i'}})$$

Man bestimmt $W_1, ..., W_{Kmax}$ für $K = 1, ..., Kmax$.

Daten-gestützte Bestimmung von K

- Wenn $K < K^*$, dann ist meist eine Teilmenge der Beobachtungen in einem Cluster schon richtig zugeordnet, das Cluster müsste aber weiter aufgeteilt werden.
  - $W_{K+1} << W_K$
- Wenn $K > K^*$, dann ist ein 'richtiges' Cluster zerteilt worden.
  - $W_{K+1} < W_K$.
- Man sucht also nach einem Knick in der Kurve der $W_1, ..., W_{Kmax}$-Werte und wählt als K den Wert mit dem geringsten Abstieg $W_K - W_{K+1}$.
  - $\{W_K - W_{K+1} \mid K < K^*\} >> \{W_K - W_{K+1} \mid K \geq K^*\}$

Gap Heuristik

- Tibshirani et al. (2001) vergleichen die Kurve der anhand der Daten gemessenen W-Werte mit einer “normalen”.
- Es werden n Mal zufällig Datenpunkte erzeugt, die innerhalb einer Hülle um die Beobachtungen gleichmäßig verteilt sind.
- Für die simulierten Daten werden die W-Werte ausgerechnet und der Erwartungswert bestimmt.
- Die Kurven werden auf einer logarithmisierten Skala aufgetragen und verglichen: wo der Abstand zwischen den Kurven (gap) am größten ist, liegt das richtige K^*.

Gap Heuristik im Bild

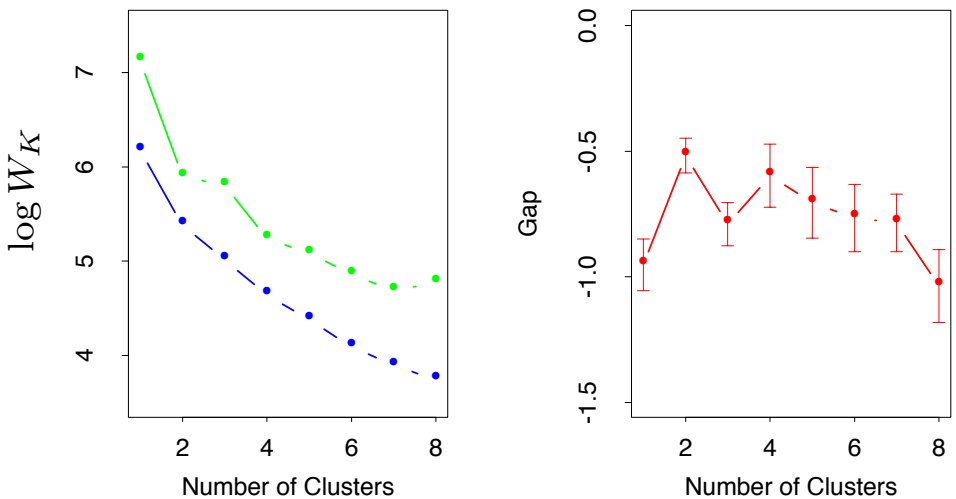


Figure 14.11: *Left panel: observed (green) and expected (blue) values of* $\log W_K$ *for the simulated data of Figure 14.4. Right panel: Gap curve, equal to the difference between the observed and expected values of* $\log W_K$*. The Gap estimate* K^* *is the smallest* K *producing a gap within one standard deviation of the maximum; here* $K^* = 2$*.*

Was wissen Sie jetzt?

- Sie haben die Abstandsmaße kennengelernt und sich dabei an die Ähnlichkeit bei kNN erinnert.
- Sie kennen das Optimierungsproblem des Clusterings (Gleichung 10).
- Sie kennen das Qualitätskriterium des inneren Abstands (Gleichung 8).
- Die Repräsentation eines Clusters kann durch alle zugeordneten Punkte, einige zugeordnete Punkte, einen zentralen zugeordneten Punkt oder ein berechnetes Zentrum sowie durch logische Formeln erfolgen.
- Zur Lösung des Optimierungsproblems kennen Sie K-Means: Euklidscher Abstand, Repräsentation durch berechnete Mittelpunkte, iteratives Vorgehen.
- Als Vorgehen zur Wahl der Anzahl K und zur Initialisierung der K Mittelpunkte haben Sie Heuristiken gesehen.

Hierarchisches Clustering

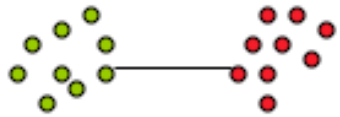
- Die Cluster sollen nicht auf einer Ebene liegen, sondern eine Taxonomie bilden.
- Die unterste Ebene enthält einzelne Beobachtungen.
- Jede Ebene enthält Cluster, die (zwei) Cluster der Ebene darunter subsummieren.
- Die oberste Ebene enthält ein Cluster mit allen Beobachtungen.
- Man unterscheidet ein Vorgehen bottom-up (agglomerativ) und top-down (aufteilend).

Agglomeratives Clustering

- Stufenweise werden Beobachtungen zu übergeordneten Clustern verschmolzen.
- Oft wird ein binärer Baum erzeugt, d.h. immer je 2 Cluster werden verschmolzen.
- Der Benutzer sucht die aussagekräftigste Ebene aus.
- Grundlage ist die Unähnlichkeit von Clustern: solche mit geringster Unähnlichkeit werden verschmolzen.
- Die Unähnlichkeit $d(G, H)$ der Cluster G, H wird berechnet durch den Abstand $d_{gh} = D(\vec{x_g}, \vec{x_h})$, wobei $\vec{x_g} \in G, \vec{x_h} \in H$.
- Welche Beobachtungen genutzt werden, macht den Unterschied zwischen den 3 wichtigsten Maßen zur Cluster-Unähnlichkeiten aus.

Single Linkage Clustering

Die Unähnlichkeit zwischen Cluster G und H ist die Unähnlichkeit der nächsten Punkte.

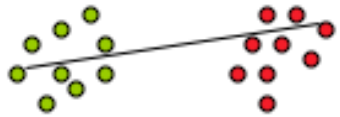


$$\begin{aligned} d_{SL}(G,H) &= min_{\vec{x_g} \in G, \vec{x_h} \in H} D(\vec{x_g}, \vec{x_h}) \\ &= min_{g \in G, h \in H}\ d_{gh} \end{aligned}$$

- Problem: Single Linkage ergibt eventuell Cluster, die nicht kompakt sind mit großer Unähnlichkeit innerhalb eines Clusters.

Complete Linkage Clustering

Die Unähnlichkeit zwischen Cluster G und H ist die Unähnlichkeit der entferntesten Punkte.



$$\begin{aligned} d_{CL}(G,H) &= max_{\vec{x_g} \in G, \vec{x_h} \in H} D(\vec{x_g}, \vec{x_h}) \\ &= max_{g \in G, h \in H}\ d_{gh} \end{aligned}$$

- Problem: Complete Linkage produziert kompakte Cluster, aber eventuell sind die Beobachtungen eines Clusters G näher zu denen eines anderen H als zu denen in G.

Average Linkage Clustering

Die Unähnlichkeit zwischen Cluster G und H ist die durchschnittliche Unähnlichkeit aller Punkte in G von allen in H.

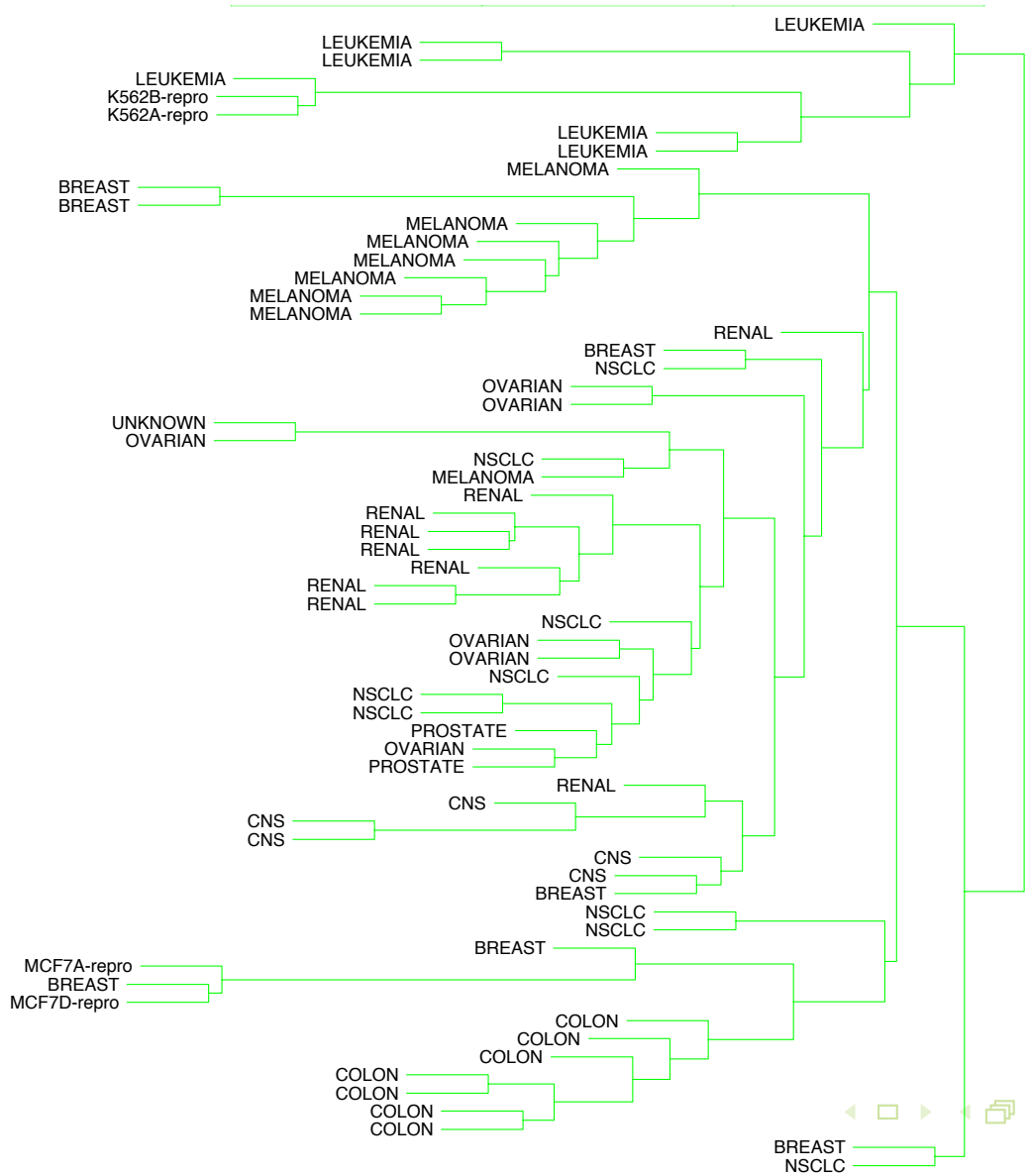
$$d_{AL}(G,H) = \frac{1}{N_G N_H} \sum_{g \in G} \sum_{h \in H} d_{gh}$$

- Kompromiss zwischen Single und Complete Linkage: relativ kompakte Cluster, die relativ weit von einander entfernt sind.
- Problem: Eine strikt monoton aufsteigende Transformation des Abstandsmaßes $h(d_{gh})$ kann das Ergebnis stark verändern.

Beispiel MicroArray-Daten über Krebs

Figure 1.3: *DNA microarray data: expression matrix of 6830 genes (rows) and 64 samples (columns), for the human tumor data. Only a random sample of 100 rows are shown. The display is a heat map, ranging from bright green (negative, under expressed) to bright red (positive, over expressed). Missing values are gray. The rows and columns are displayed*

Beispiel Average Linkage bei MicroArray-Daten über Krebs



Dendrogramme für agglomeratives Clustering der MicroArray-Daten über Krebs mit Average, Complete, Single Linkage

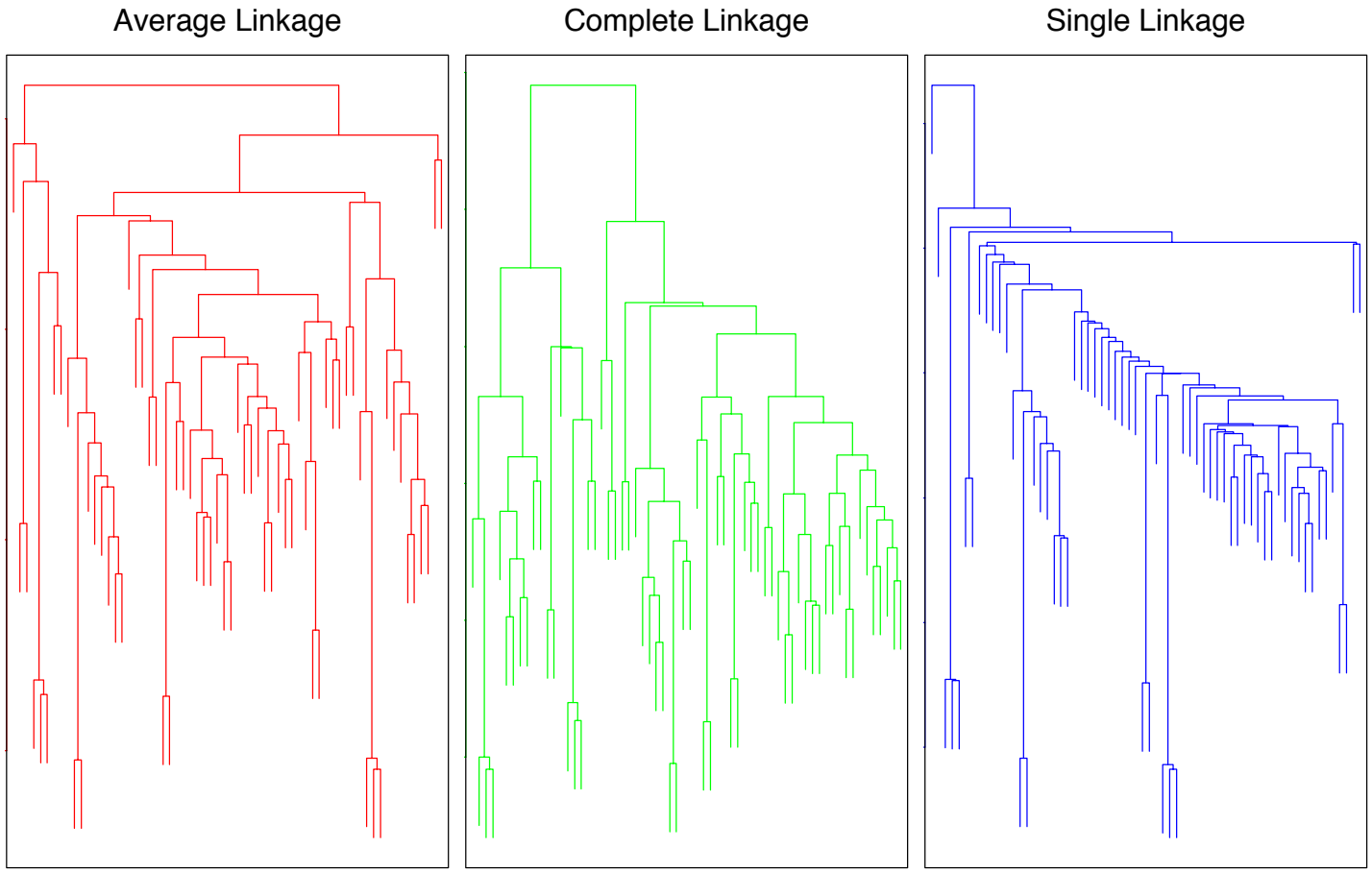


Figure 14.13: *Dendrograms from agglomerative hier-*

Dendogramme

- Monotonie: Die Unähnlichkeit steigt über die Ebenen von unten nach oben monoton an.
- Ein Dendogramm ist so angeordnet, dass die Höhe eines Knoten (Clusters) gerade proportional zur Unähnlichkeit zwischen den beiden Unterknoten ist.
- Deshalb kann der Benutzer eine Ebene auswählen, bei der die Unähnlichkeit zwischen Clustern einen Schwellwert übersteigt.

Aufteilendes Clustering durch rekursives K-Means

- Die rekursive Anwendung von K-Means mit $K = 2$ ergibt ein aufteilendes Verfahren.
- Allerdings ist das Ergebnis dann kein Dendogramm, bei dem die Unähnlichkeit mit den Ebenen immer monoton ansteigt.
- Deshalb gibt es ein anderes Verfahren.

Aufteilendes Clustering durch iteratives Verringern der Unähnlichkeit in einem Cluster

- Alle Beobachtungen sind im Wurzelknoten G.
- Aufteilung(G)
  1. Initialisierung:
     Wähle den Punkt $\vec{x_h}$ in G, der am unähnlichsten zu allen anderen ist. Dieser wird dem neuen Cluster H zugeordnet.
  2. Teile iterativ G auf solange es ein $\vec{x_i} \in G$ gibt, das im Durchschnitt ähnlicher zu allen $\vec{x_j} \in H$ ist als zu allen $\vec{x_g} \in G$:
     $H := H \cup \{\vec{x_i}\}$; $G := G \backslash \{\vec{x_i}\}$;
  3. Wähle Cluster zur Aufteilung aus:
     Solange $\mid G \mid > 1$ und $d_{ij} > 0$ für alle $\vec{x_i}, \vec{x_j} \in G$ Aufteilung(G).
     Solange $\mid H \mid > 1$ und $d_{ij} > 0$ für alle $\vec{x_i}, \vec{x_j} \in H$ Aufteilung(H).

Was wissen Sie jetzt?

- Top-down Clustering kann durch rekursives K-Means realisiert werden, ist aber aufwändig.
- Optimieren der Average Linkage $d_{AL}(G, H)$ für alle möglichen Aufteilungen wird angenähert durch ein iteratives Verfahren, bei dem in jeder Iteration eine Beobachtung von dem Ausgangscluster G dem neuen Cluster H zugeordnet wird.
- Kann man das effizienter machen?