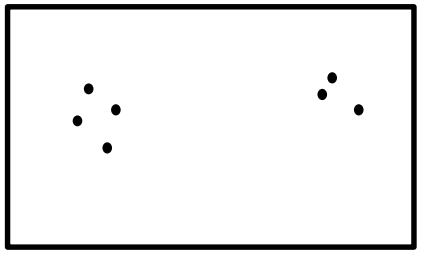
ETH zürich

Clustering

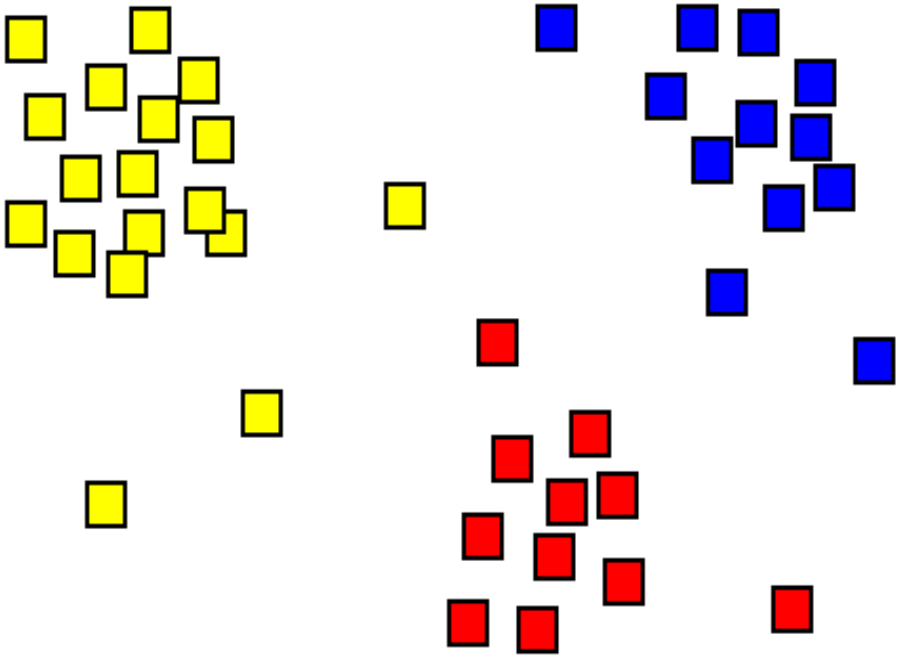
Ziel von Clustering



- Finde Gruppen, sodas Elemente innerhalb der gleichen Gruppe möglichst ähnlich sind und Elemente von verschiedenen Gruppen möglichst verschieden sind
- Bsp:
  - Gruppen von ähnlichen Kunden um effizientere Werbung zu machen
  - Gruppen von Krankheiten um effizientere Therapie zu finden

- “Unsupervised Learning”: Es sind keine Gruppenzugehörigkeiten bekannt

Clustering mit Computer: 3+ Dimensionen



Auge ist extrem gut im Clustering !

Verwende Clustering nur, Wenn man die Daten nicht von Auge inspizieren kann.

Problem von Clustering

N Beobachtungen, k Cluster: k^N mögliche Zuordnungen

Bsp: Bei N=100 und k=5 sind $5^{100} = 7*10^{69}$ Zuordnungen möglich.

Es ist also nicht möglich, alle Zuordnungen zu bewerten

→ Heuristische Methoden

Viele Methoden…

Alle zeigen einen wahren Aspekt der Realität!
Welcher Aspekt ist in der gegebenen Anwendung am wichtigsten?

Fahrplan

- Partitionierungsmethoden:
  - k-Means
  - PAM
- Agglomerative Methoden:
  - Hierarchical Clustering
- Distanzmasse
- Kontrolle: Silhouette-Plot

K-Means Algorithmus

- Anzahl Cluster ist vorgegeben (K)

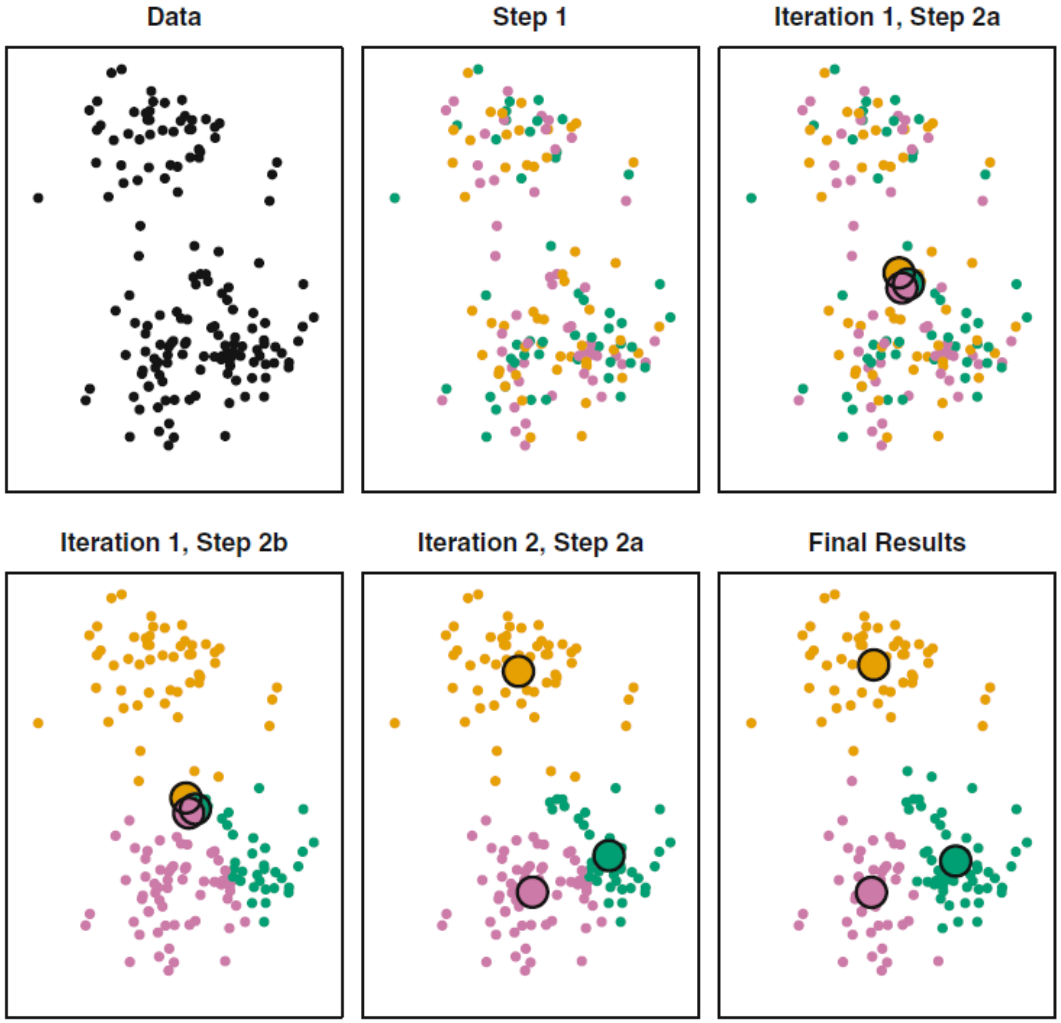
- Finde K Clusterzentren μ_C und Zuordnungen, sodas within-groups Sum of Squares (WGSS) minimal ist:

$$WGSS = \sum_{all\ Cluster\ C} \sum_{Point\ i\ in\ Cluster\ C} (x_i - \mu_C)^2$$

WGSS klein

WGSS gross

K-Means Algorithmus

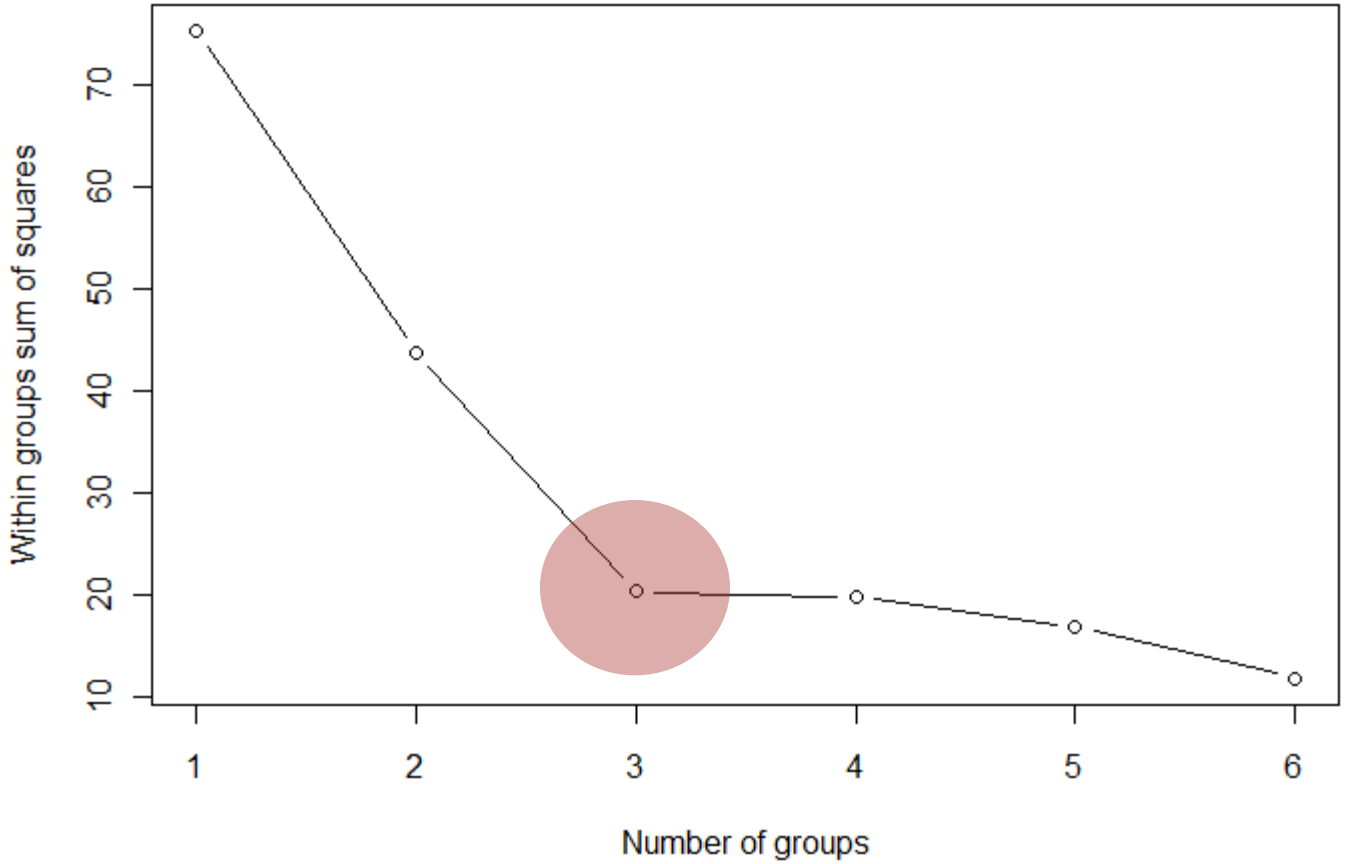


K-Mean: Zufällige Startwerte

K-Means findet nur das
lokale Optimum
→ berechne K-Means mehrmals mit unterschiedlichen, zufälligen Anfangsgruppierungen
→ verwende das Ergebnis mit dem besten WGSS

K-Means: Wie viele Cluster ?

- Berechne K-means für mehrere Anzahl Gruppen
- Plotte WGSS vs. Anzahl Gruppen
- Suche “Knick”



Robuste Alternative: PAM

- “Partinioning around Medoids” (PAM)
- K-Means: Zentrum des Clusters ist ein beliebiger Punkt im Raum
  PAM: Zentrum des Clusters muss eine Beobachtung sein (“medoid”)
- Vorteile von PAM vs. K-Means:
  - robuster bzgl. Ausreisser
  - funktioniert mit jedem Distanzmass
  - ein Repräsentant pro Cluster: Einfache Interpretation

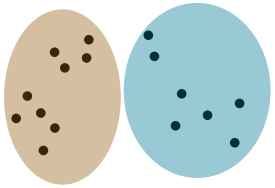
Partitionierungsmethoden in R

- Funktion “kmeans” in package “stats”
  Funktioniert *nur mit euklidischer Distanz*
- Funktion “pam” in package “cluster”
  Funktioniert *mit beliebiger Distanz*

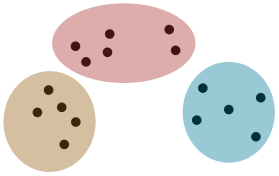
Agglomerative Methode: Hierarchical Clustering

- Löse Clusteringproblem gleichzeitig für alle möglichen Anzahlen von Clustern
- Setze Cluster aus Einzelbeobachtungen zusammen

- “Hierarchisch” ist eine Einschränkung:
  K-Means produziert evtl. bessere Cluster



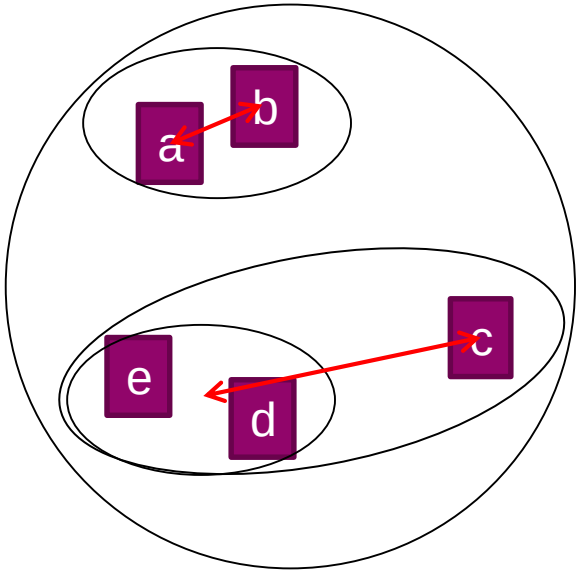
2 Cluster



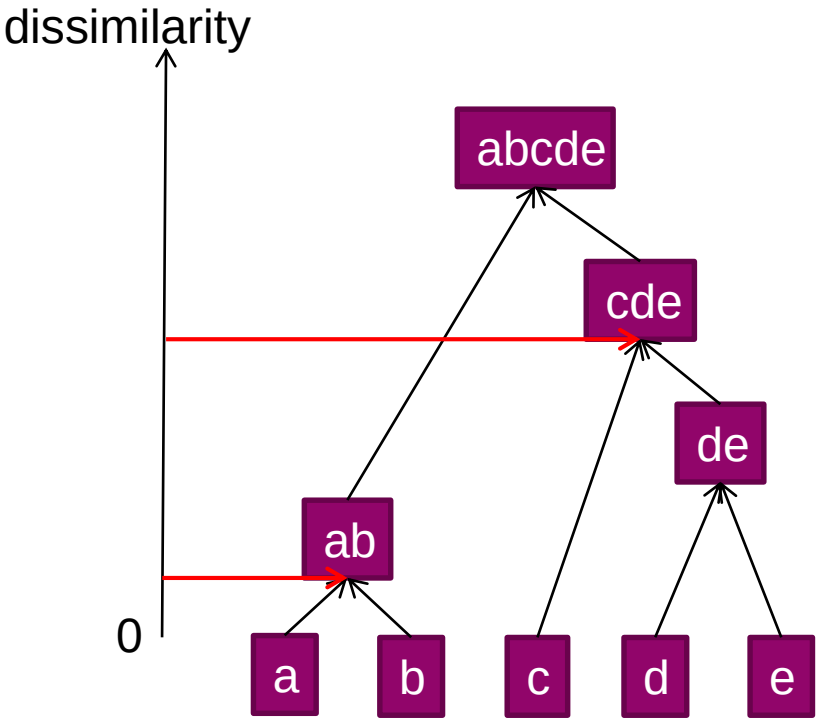
Bsp: 3 Cluster
Nicht “hierarchisch”

Agglomerative Clustering

Daten in 2 Dimensionen



Clustering tree = Dendrogramm



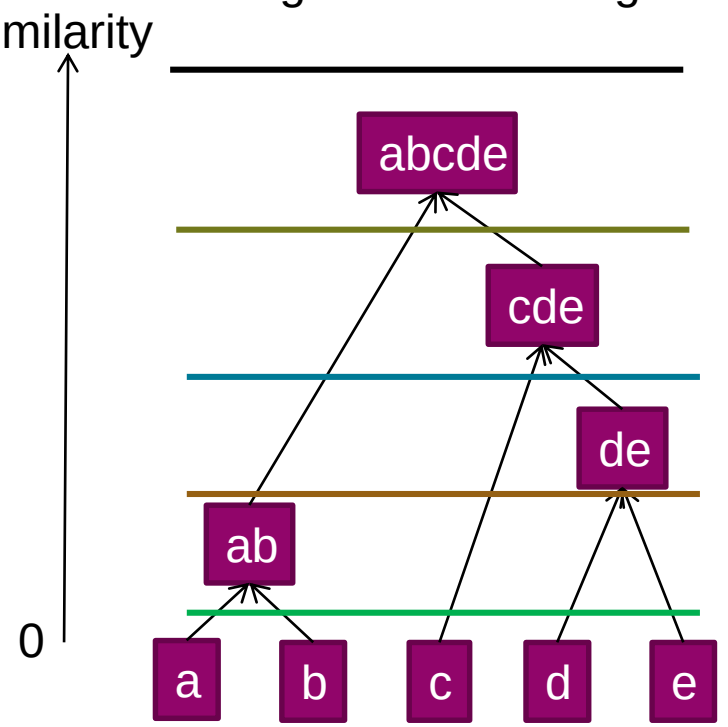
Verbinde Beobachtungen/Cluster die am nächsten sind, bis nur noch ein Cluster übrig ist

Agglomerative Clustering: Konkrete Cluster

Konkrete Cluster indem man Dendrogramm abschneidet:

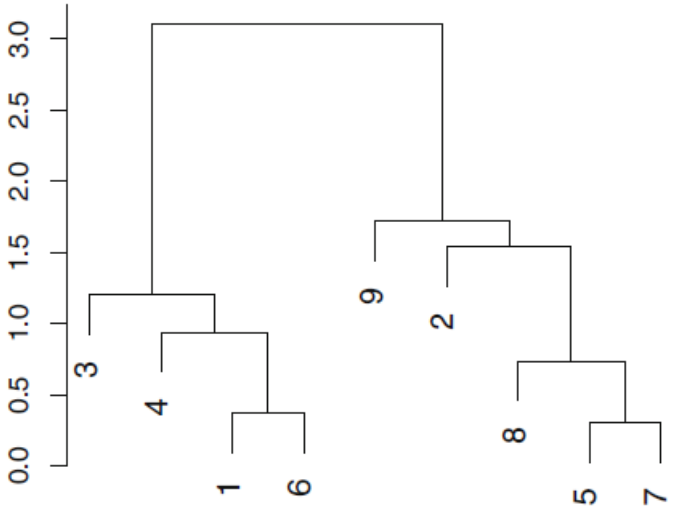
- 1 Cluster: abcde (trivial)
- 2 Cluster: ab - cde
- 3 Cluster: ab – c – de
- 4 Cluster: ab – c – d – e
- 5 Cluster: a – b – c – d – e (trivial)

Clustering tree = Dendrogramm



“cut the tree”

Dendrogramm interpretieren

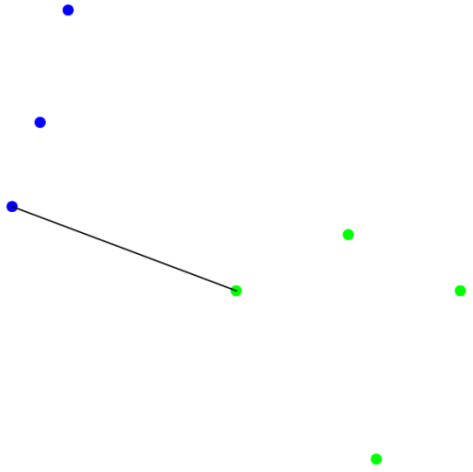


Richtig oder falsch ?
Beobachtungen 9 und 2 sind ähnlicher als Beobachtungen 9 und 7.

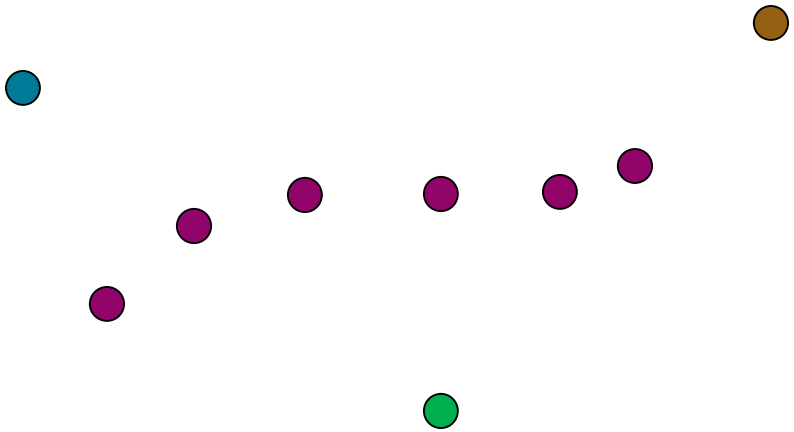
Distanzmass zwischen Cluster

- Basiert auf Distanzmass zwischen Beobachtungen (dazu später mehr)
- Häufigste Methoden:
  - single linkage
  - complete linkage
  - average linkage
- Kein richtig oder falsch: Jede Methode zeigt einen Aspekt der Realität
- Im Zweifelsfall nehme ich “average”

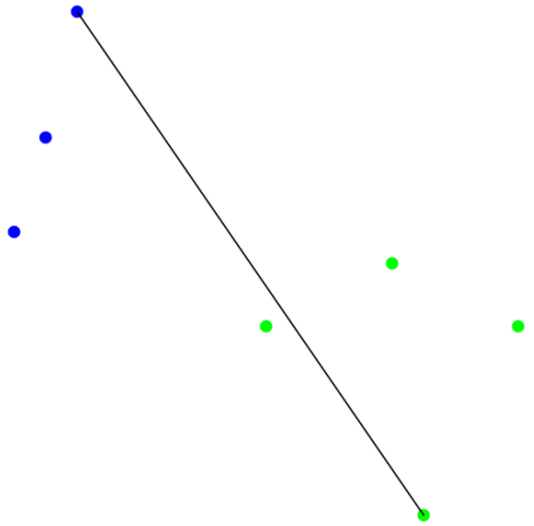
Single linkage



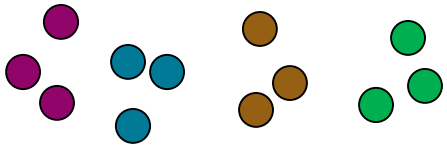
- Distanz zwischen zwei Cluster = minimale Distanz zwischen allen Elementen
- Findet auch längliche Cluster



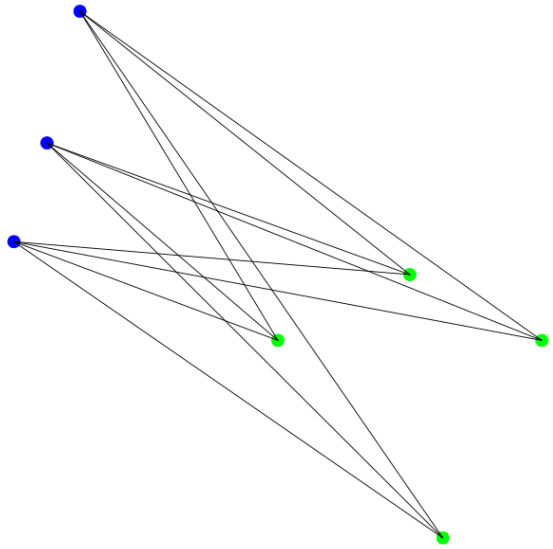
Complete linkage



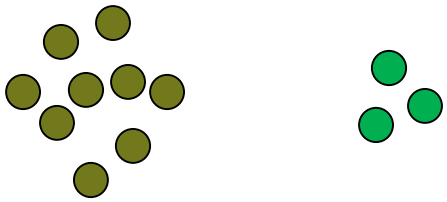
- Distanz zwischen zwei Cluster = maximale Distanz zwischen allen Elementen
- Findet kompakte Cluster, auch wenn nicht gut separiert



Average linkage

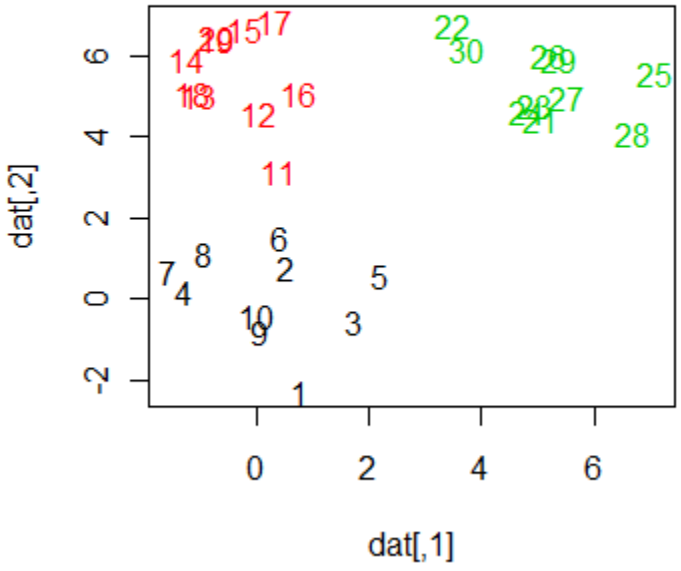


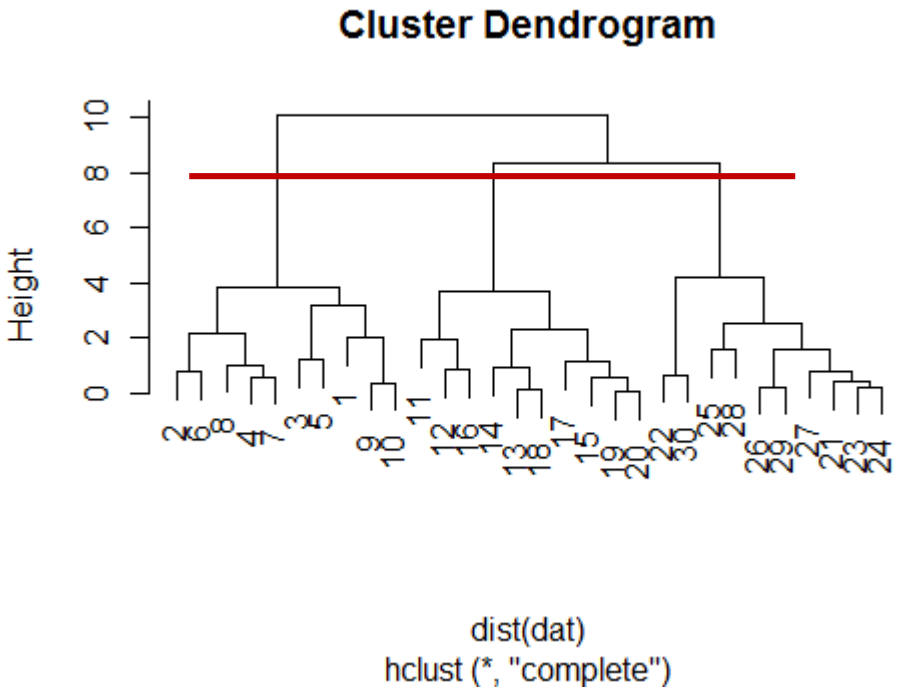
- Distanz zwischen zwei Cluster = mittlere Distanz zwischen allen Elementen
- Findet gut separierte, rundliche Cluster



Welche Anzahl Cluster ?

- Keine strikte Regel
- Finde grössten vertikalen Abstand im Dendrogramm





Agglomeratives Clustering in R

- Funktionen “hclust”, “cutree” in package “stats”

Distanzmass zwischen Beobachtungen

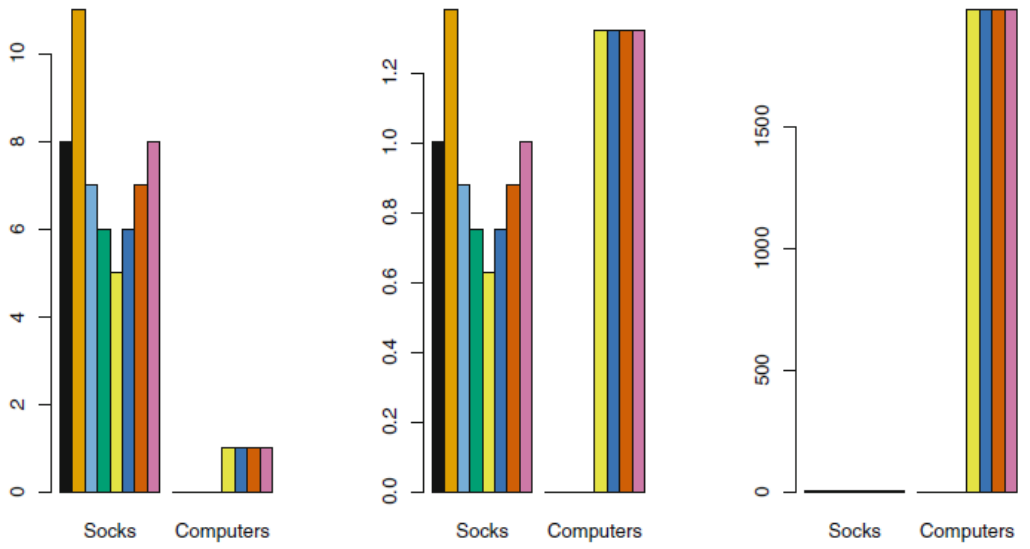
- Kontinuierliche Variablen:
  Euklidsche Distanz (standardisieren oder nicht?)
  oder Korrelation
- Diskrete Variablen:
  Simple Matching Coefficient
- Gemischte Variablen:
  Gower's Dissimilarity Index

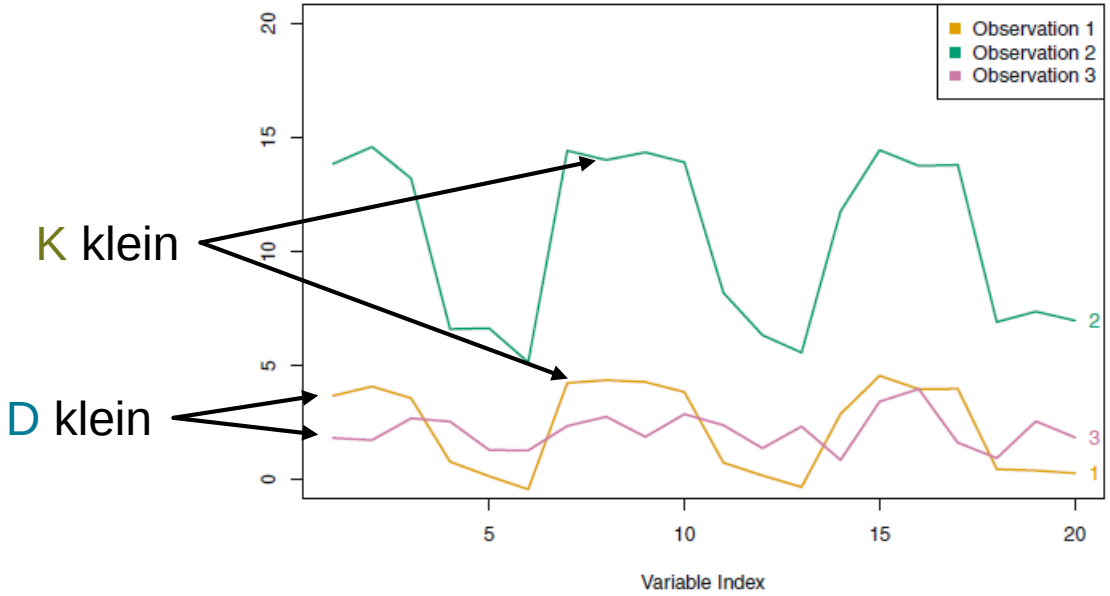
- Pragmatische Auswahl; es gibt sehr viele Alternativen

Distanzmass zwischen Beobachtungen: Kontinuierliche Daten

- Euklidische Distanz D
  Skalieren oder nicht ?

- Korrelation (Ähnlichkeit):
  “Distanz”:
  K = 1 - Korrelation





Distanzmass zwischen Beobachtungen: Diskrete Daten

- Simple Matching Coefficient
  Für jede Variable: Ist Eintrag gleich (ja → 0; nein → 1)?
- Lieblingsfarbe: $Dist(P1, P2; Farbe) = 0$
- Lieblingsessen: $Dist(P1, P2; Essen) = 1$
- Total: $Dist(P1, P2) = \frac{0+1}{1+1} = \frac{1}{2}$

Person	Lieblingsfarbe	Lieblingsessen
P1	Rot	Pizza
P2	Rot	Pasta
P3	Gelb	Pizza
P4	Blau	Pasta

Distanzmass zwischen Beobachtungen: Gemischte Daten

- Gower's Dissimilarity Index: Für beliebige Variablen
  Gower, J. C. (1971) A general coefficient of similarity and some of its properties, *Biometrics* **27**, 857–874
- Distanz zwischen zwei Zeilen:
  Für jede Variable wird eine Distanz in [0,1] berechnet
  Die Distanzen werden dann gemittelt
- Diskrete Variablen: Simple Matching Coefficient

  Kont. Variablen: $\frac{\left|Differenz(X_{ik}, X_{jk})\right|}{Range(X_{.k})}$

Gower's Dissimilarity: Beispiel

Person	Lieblingsfarbe	Alter
P1	Rot	12
P2	Rot	27
P3	Gelb	30
P4	Blau	60

Range: 48

P1-P2: Lieblingsfarbe $Dist(P1, P2; Farbe) = 0$

Alter $Dist(P1, P2; Alter) = \frac{|27-12|}{48} = \frac{15}{48} = 0.3125$

Total: $Dist(P1, P2) = \frac{0+0.3125}{2} \approx 0.15625$

P2-P4: Lieblingsfarbe $Dist(P2, P4; Farbe) = 1$

Alter $Dist(P2, P4; Alter) = \frac{|60-27|}{48} = \frac{33}{48} = 0.6875$

Total: $Dist(P2, P4) = \frac{1+0.6875}{2} \approx 0.84375$

Gower's Dissimilarity

- Wie gross ist die Gower's Dissimilarity zwischen P1 und P4?

Person	Lieblingsfarbe	Alter
P1	Rot	12
P2	Rot	27
P3	Gelb	30
P4	Blau	60

Range: 48

Distanzmasse in R

- Funktion «daisy» in package «cluster» (euklidisch mit/ohne standardisierung, simple matching coeficient, gower's dissimilarity)
- Korrelation: Funktion «cor» um Korrelationsmatrix zu berechnen

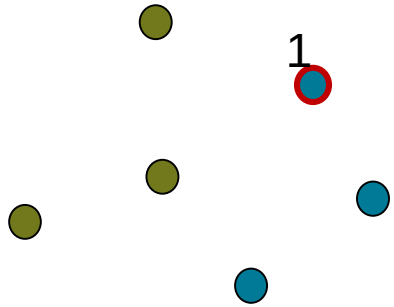
Qualität der Cluster: Silhouette plot

- Ein Wert S(i) in [0,1] für jede Beobachtung
- Berechne für jede Beobachtung i:
  a(i) = mittlere Distanz von i zu allen anderen Punkten im gleichen Cluster
  b(i) = mittlere Distanz zwischen i und allen Punkten im "Nachbarcluster", d.h. im nächsten Cluster, zu dem i nicht gehört

  Dann ist S(i) = $\frac{(b(i)-a(i))}{\max(a(i),b(i))}$
- S(i) gross: gute Trennung
  S(i) klein: schlechte Trennung
  S(i) negativ: Punkt zu "falschem" Cluster zugeordnet

Silhouette plot: Beispiel

S(1) gross

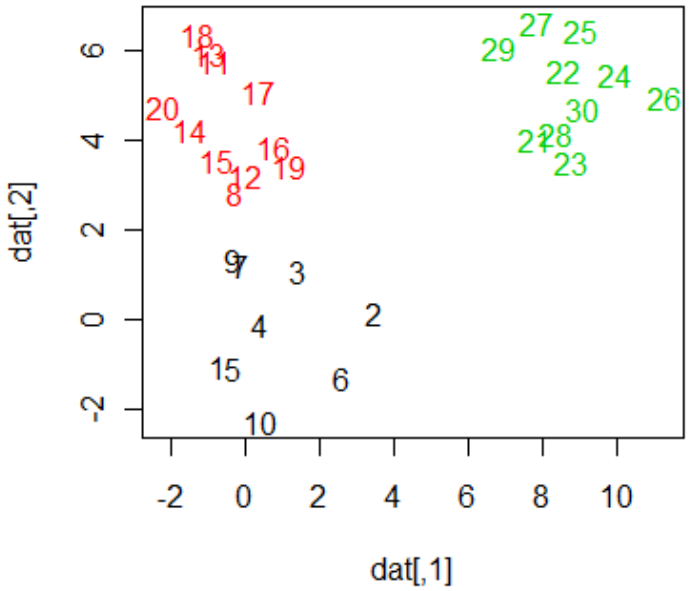


S(1) klein

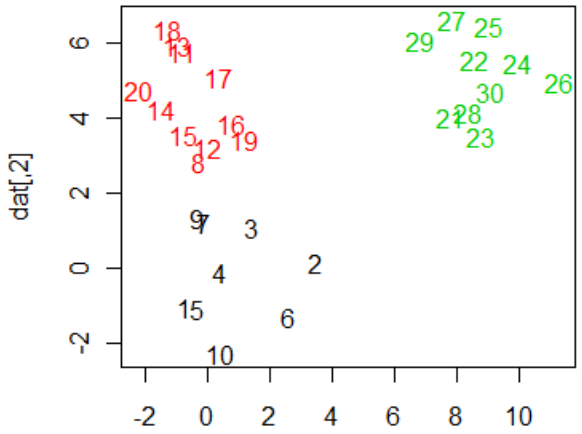
Faustregel: Mittleres S sollte über 0.5 sein

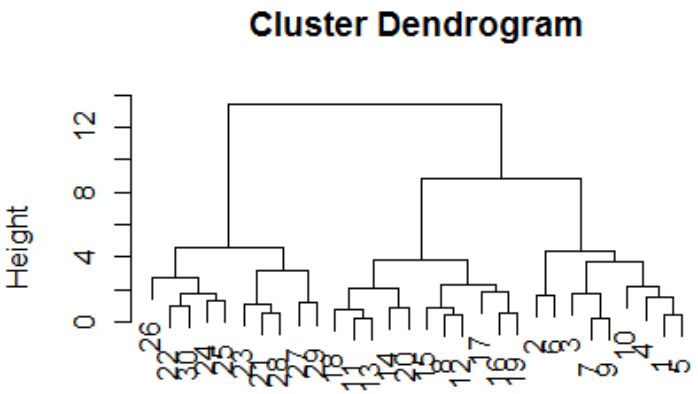
Silhouette plot: Beispiel

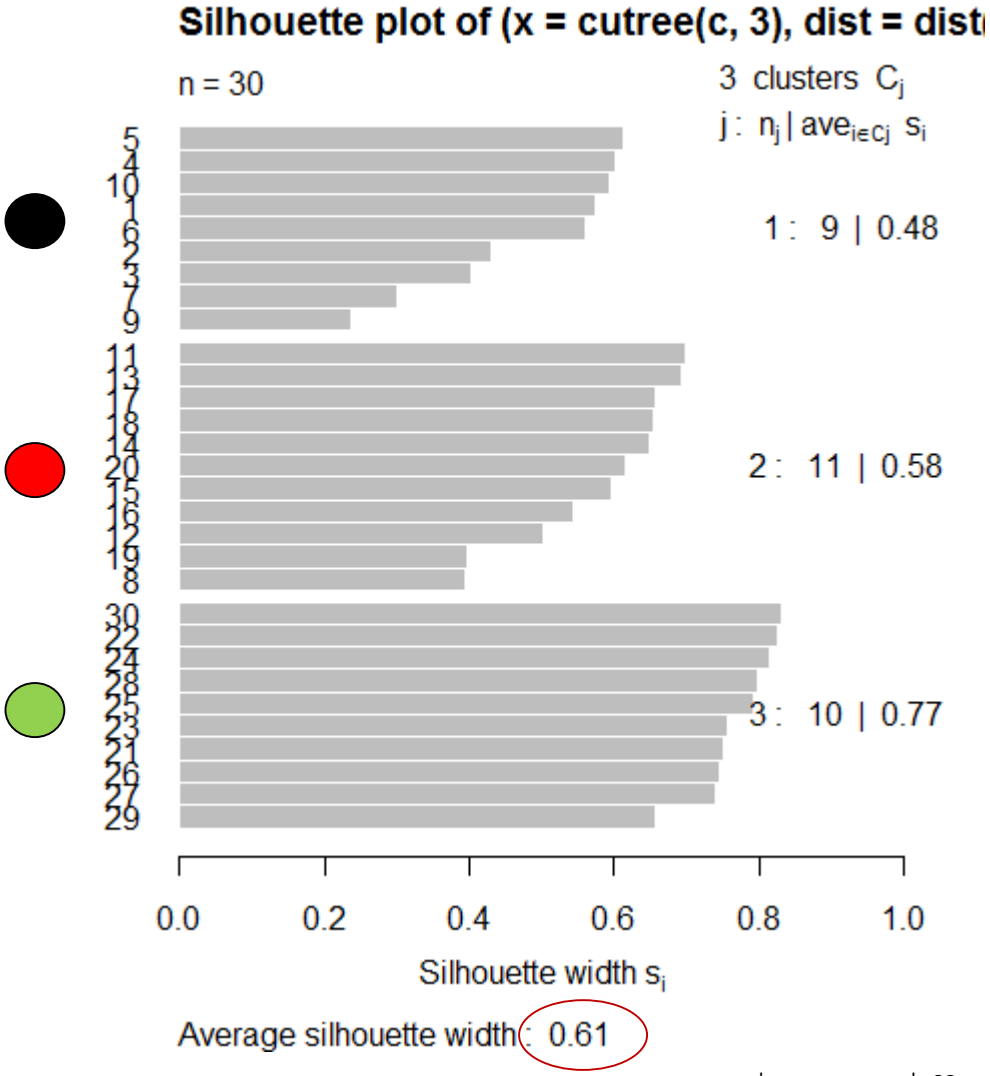
Welcher Punkt hat den grösseren Silhouette-Wert, 29 oder 9 ?



Silhouette plot: Bsp







Silhouette Plots in R

- Function “silhouette” in package “cluster”