

Fuzzy Clustering

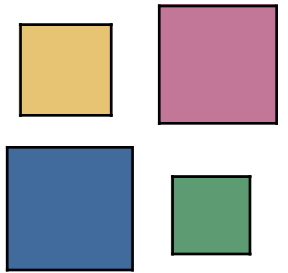
Fuzzy Clustering vergibt keine scharfe Zugehörigkeit zu einem bestimmten Cluster, sondern nur eine prozentuale Zugehörigkeit.

Somit ergibt sich für n Beobachtungen und k Cluster eine Zugehörigkeitsmatrix U

$$U = \begin{pmatrix} u_{11} & u_{12} & \dots & u_{1k} \\ u_{21} & u_{22} & \dots & u_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ u_{n1} & u_{n2} & \dots & u_{nk} \end{pmatrix}$$

mit

1. $0 \leq u_{ij} \leq 1$ für $1 \leq i, j \leq n$
2. $0 < \frac{1}{n} \sum_{i=1}^n u_{ij} < 1$ (kein Cluster darf leer sein)
3. $\sum_{j=1}^k u_{ij} = 1$ (total Membership)



Fuzzy Clustering: Minimierung

- Sind Distanzen zwischen den Beobachtungen gegeben, so nimmt die Minimierung folgende Form an:

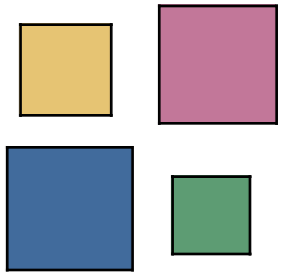
$$\sum_{j=1}^k \frac{\sum_{r,s}^n u_{rj}^2 u_{sj}^2 d_{rs}}{2 \sum_i^n u_{ij}^2}$$

- Werden die Distanzen aus den Entfernungen zu den Cluster Zentren ($\mathbf{z}_1, \dots, \mathbf{z}_k$) bestimmt, so ergibt sich:

$$\sum_{j=1}^k \sum_{i=1}^n u_{i,j}^2 d^2(\mathbf{x}_i, \mathbf{z}_j)$$

für Daten $\mathbf{x}_i, i = 1, \dots, n$.

Diese Variante ist unter Fuzzy c-Means bekannt.



Fuzzy C-Means Algorithmus

Fuzzy c-Means arbeitet iterativ:

1. Berechnung der Clusterzentren:

$$\mathbf{z}_j = \frac{\sum_{i=1}^n u_{ij}^2 \mathbf{x}_i}{\sum_{i=1}^n u_{ij}^2}$$

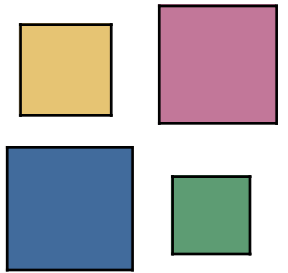
2. Neuberechnung der Distanzen:

$$d(\mathbf{x}_i, \mathbf{z}_j) = (\mathbf{x}_i - \mathbf{z}_j)' (\mathbf{x}_i - \mathbf{z}_j)$$

3. Update der Matrix U :

$$u_{ij} = \frac{1}{\sum_{r=1}^k \frac{d(\mathbf{x}_i, \mathbf{z}_j)}{d(\mathbf{x}_i, \mathbf{z}_r)}}$$

Iteriere 1-3 bis $||\Delta U|| < \epsilon$.



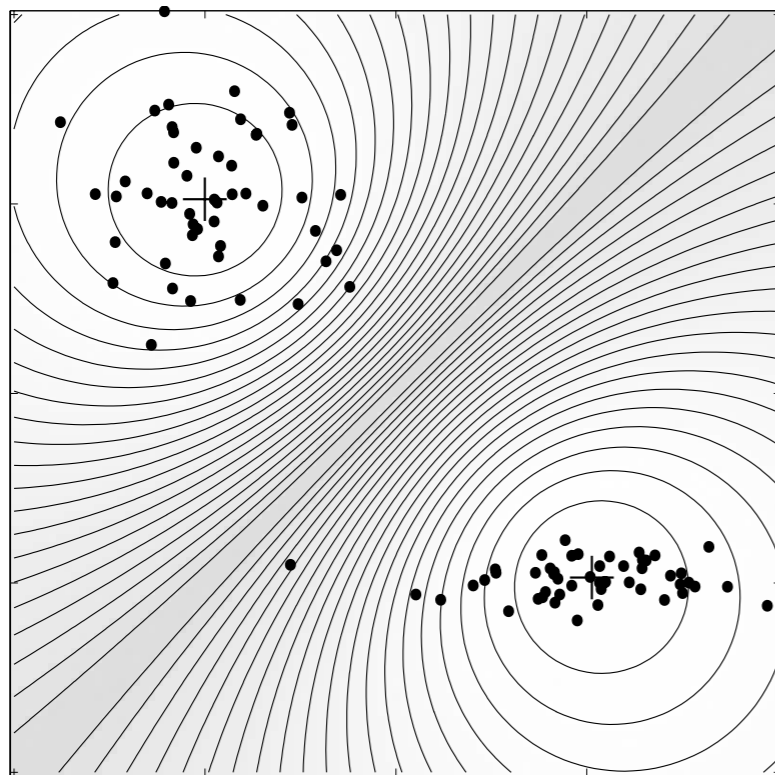
Bemerkungen

In dieser Form kann Fuzzy c-Means nur kreisförmige Clusterstrukturen definieren.

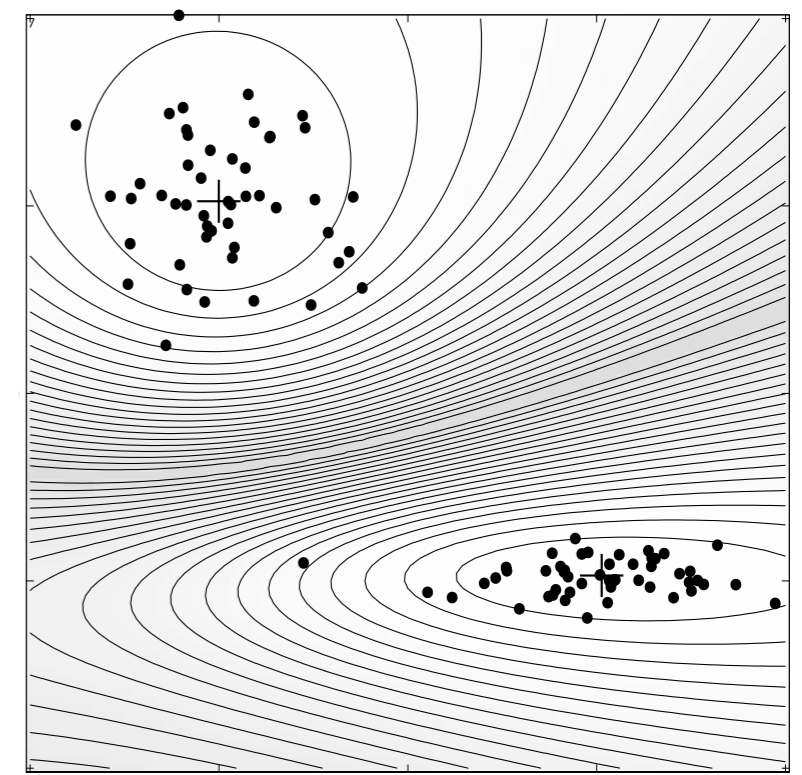
Ersetzt man die Distanzberechnung in Schritt 2 durch

$$d(\mathbf{x}_i, \mathbf{z}_j) = (\mathbf{x}_i - \mathbf{z}_j)' \Sigma_j^{*-1} (\mathbf{x}_i - \mathbf{z}_j) \quad j = 1, \dots, k$$

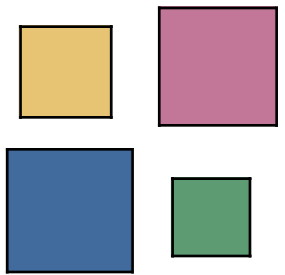
wobei Σ_j^* die VC-Matrix des j -ten Cluster ist (unter Berücksichtigung der Gewichte u_{ij}).



kreisförmig

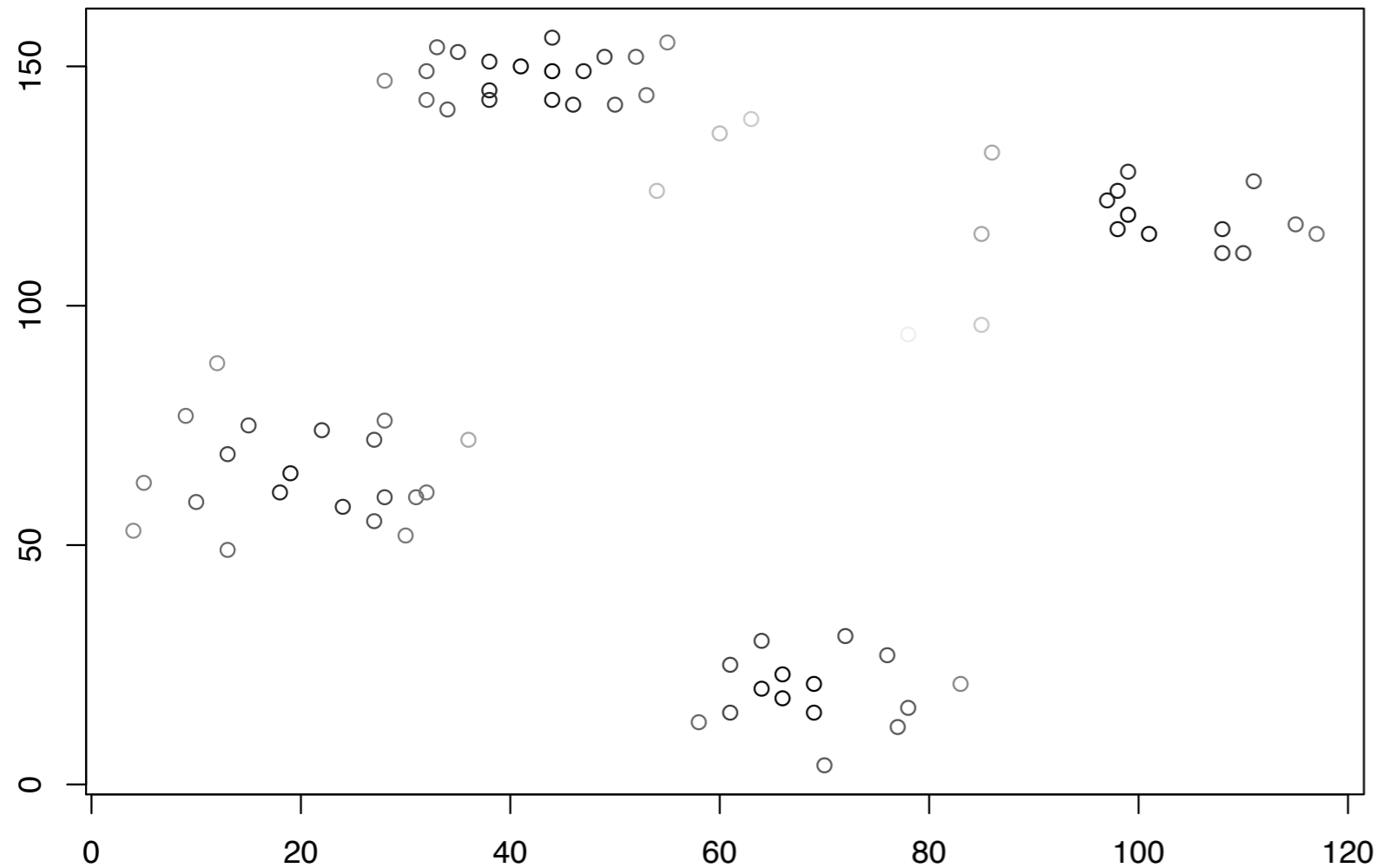


ellipsoid

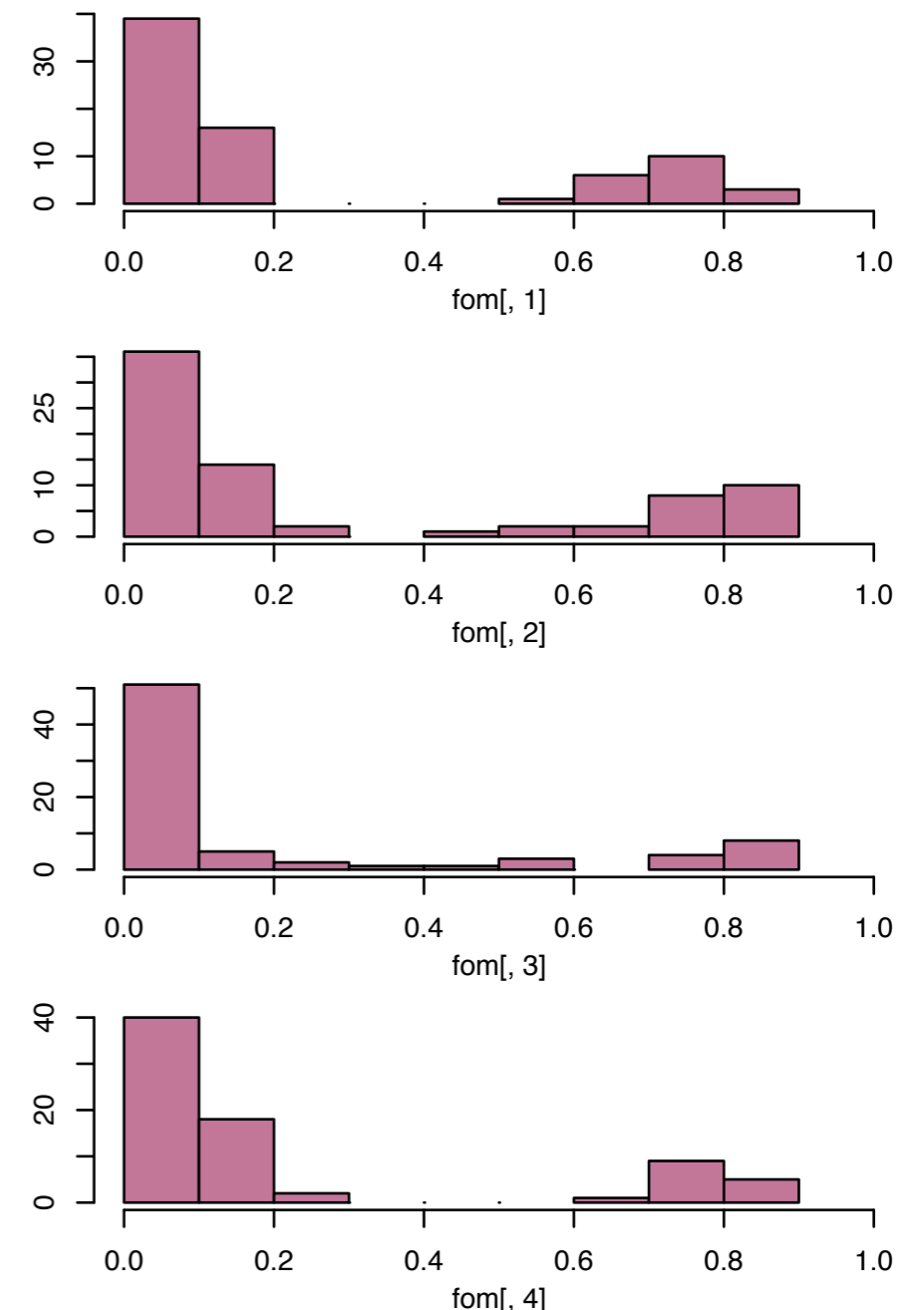


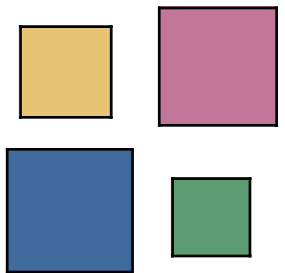
Ruspini Daten

- 4 Cluster vorgegeben
- Maximale Zugehörigkeit (via Grauschattierung)



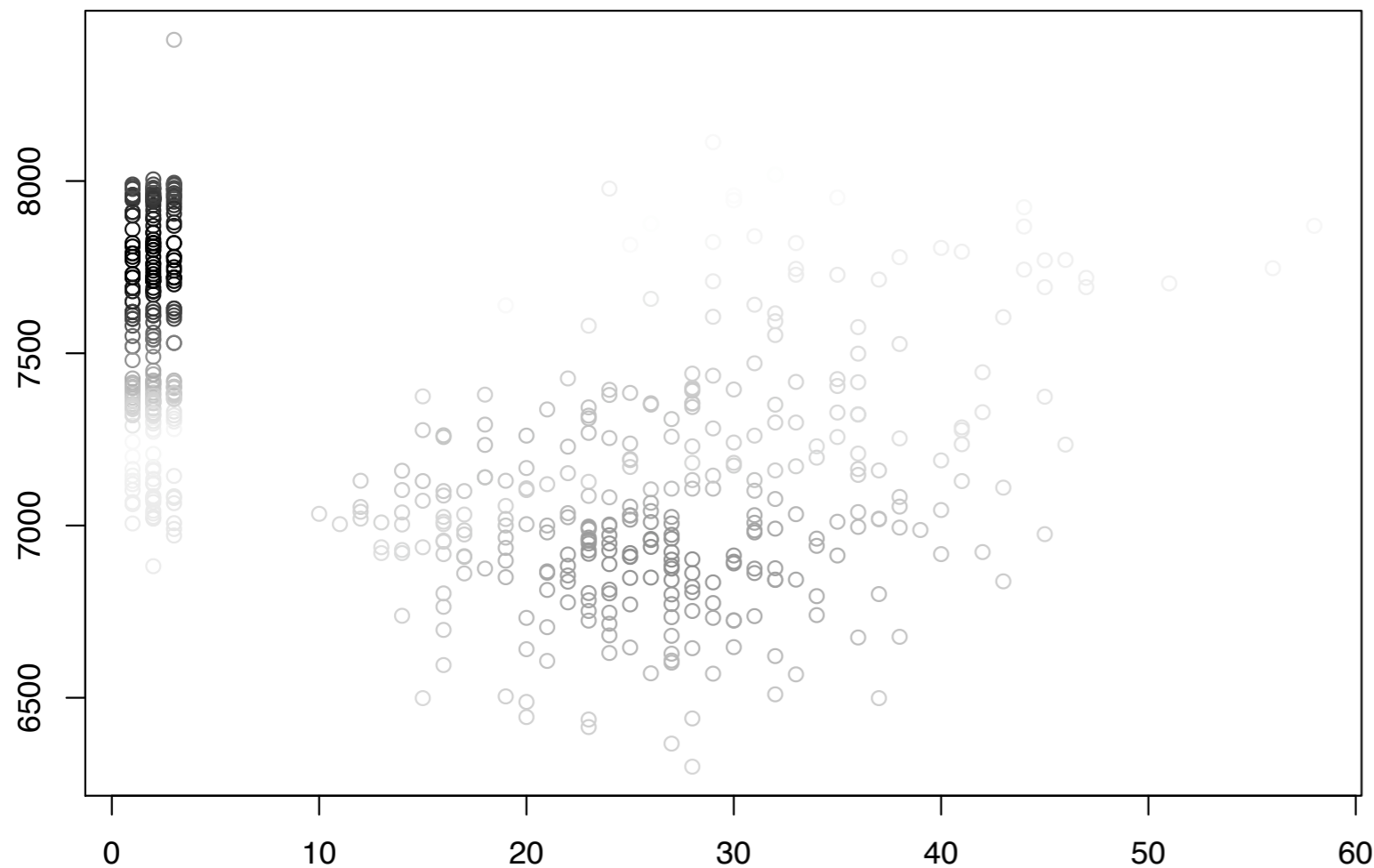
- Verteilung d. Zugehörigkeit





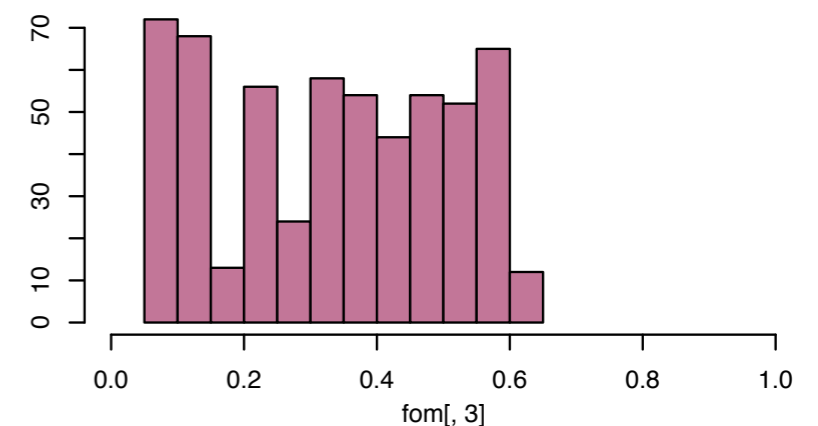
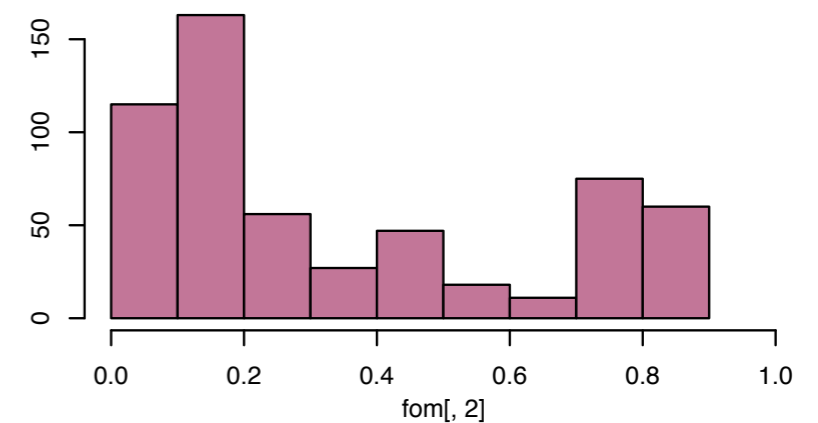
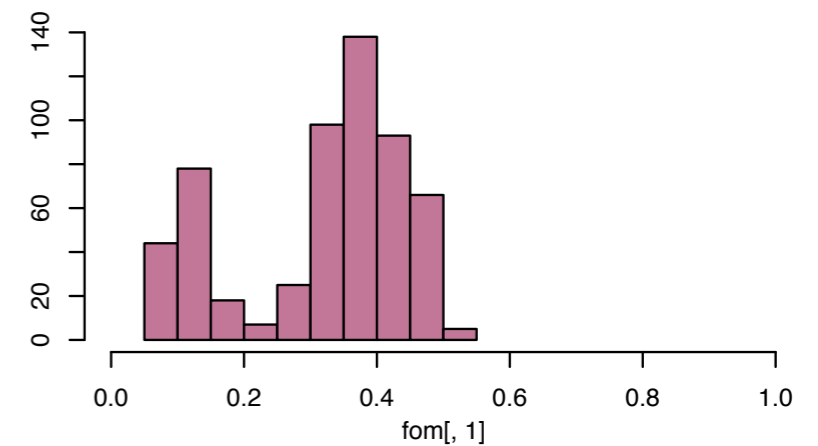
Olive Oils

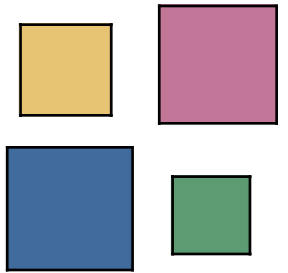
- Nur 2 Variablen: *oleic* und *eicosenoic*
- 3 Cluster vorgegeben
- Maximale Zugehörigkeit



Konvergenz Probleme ?!

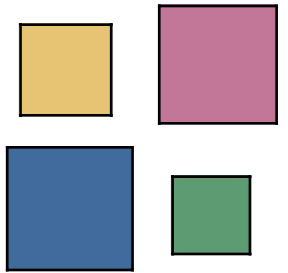
- Verteilung d. Zugehörigkeit





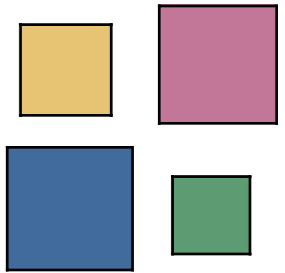
Cluster Library in R

- R hat eine eigene Library für Cluster Methoden:
 - Partitionierungsverfahren
 - **kmeans** (Teil der stats Library)
 - **pam** (Partitioning Around Medoids)
 - **clara** (wie pam, arbeitet aber mit Teilmengen zur Beschleunigung)
 - Hierarchische Verfahren
 - **agnes** (Agglomerative Nesting)
 - **diana** (Divisive Analysis Clustering)
 - **mona** (Monothetic Analysis Clustering of Binary Variables)
 - Fuzzy Clustering
 - **fanny** (Fuzzy Analysis Clustering)
- Für Model based Clustering gibt es die MClust Library
 - **Mclust**



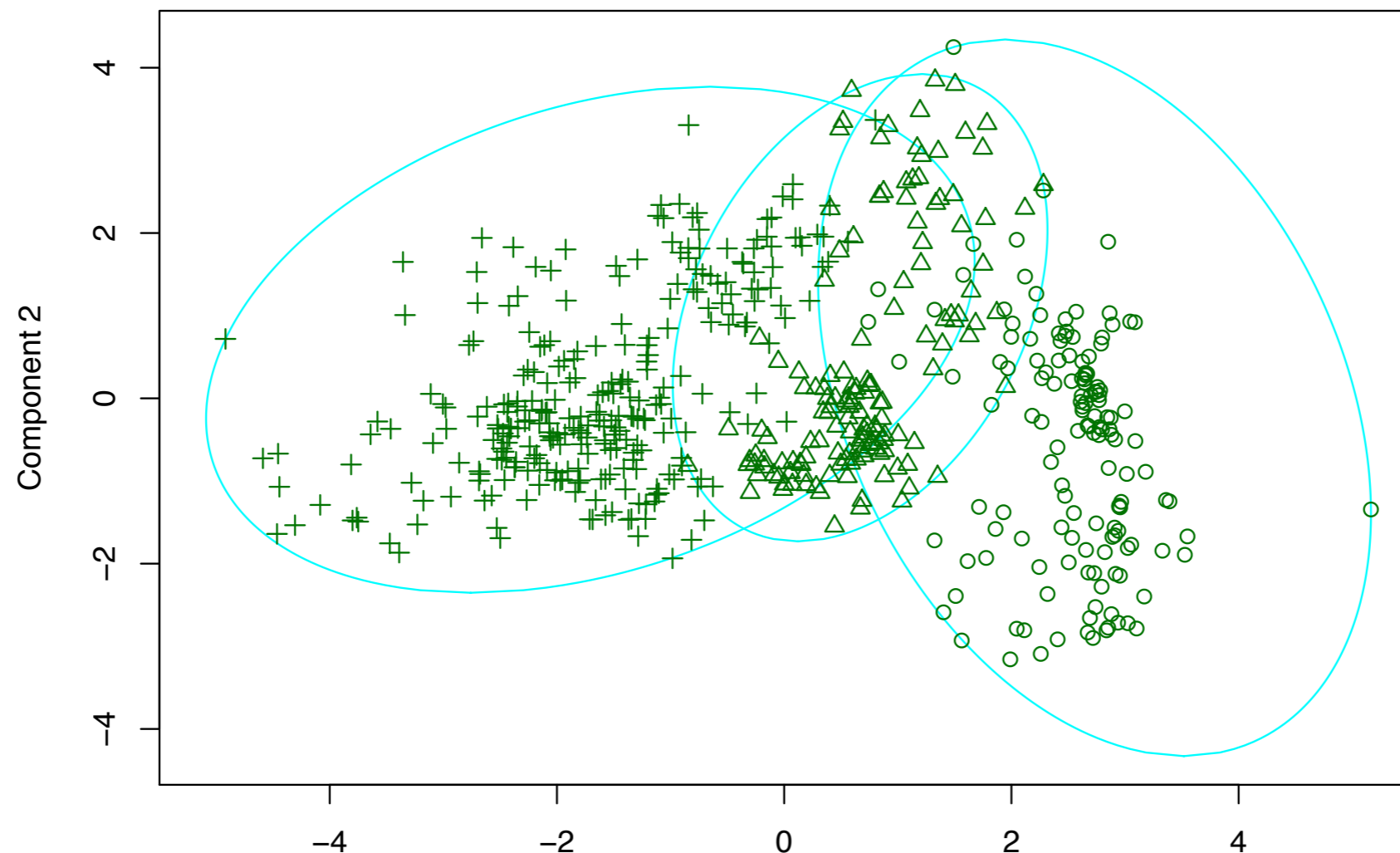
Objektstruktur

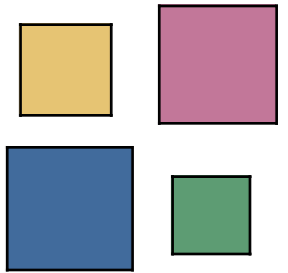
- Für die 6 Verfahren der Cluster Library gibt es 3 Objekttypen:
 - **partition**
Für alle Partitionierungsverfahren
 - **hierarchical**
Für alle hierarchischen Verfahren
 - **mona**
Für das monothetische Verfahren
- Diese Objekte können mit den Standardfunktionen
 - **print**
 - **summary**
 - **plot**verglichen und analysiert werden.
- **kmeans** hat keine weiteren Objekteigenschaften.
- **Mclust** bietet einen ganz eigenen Objekttyp.



Diagnostische Plots

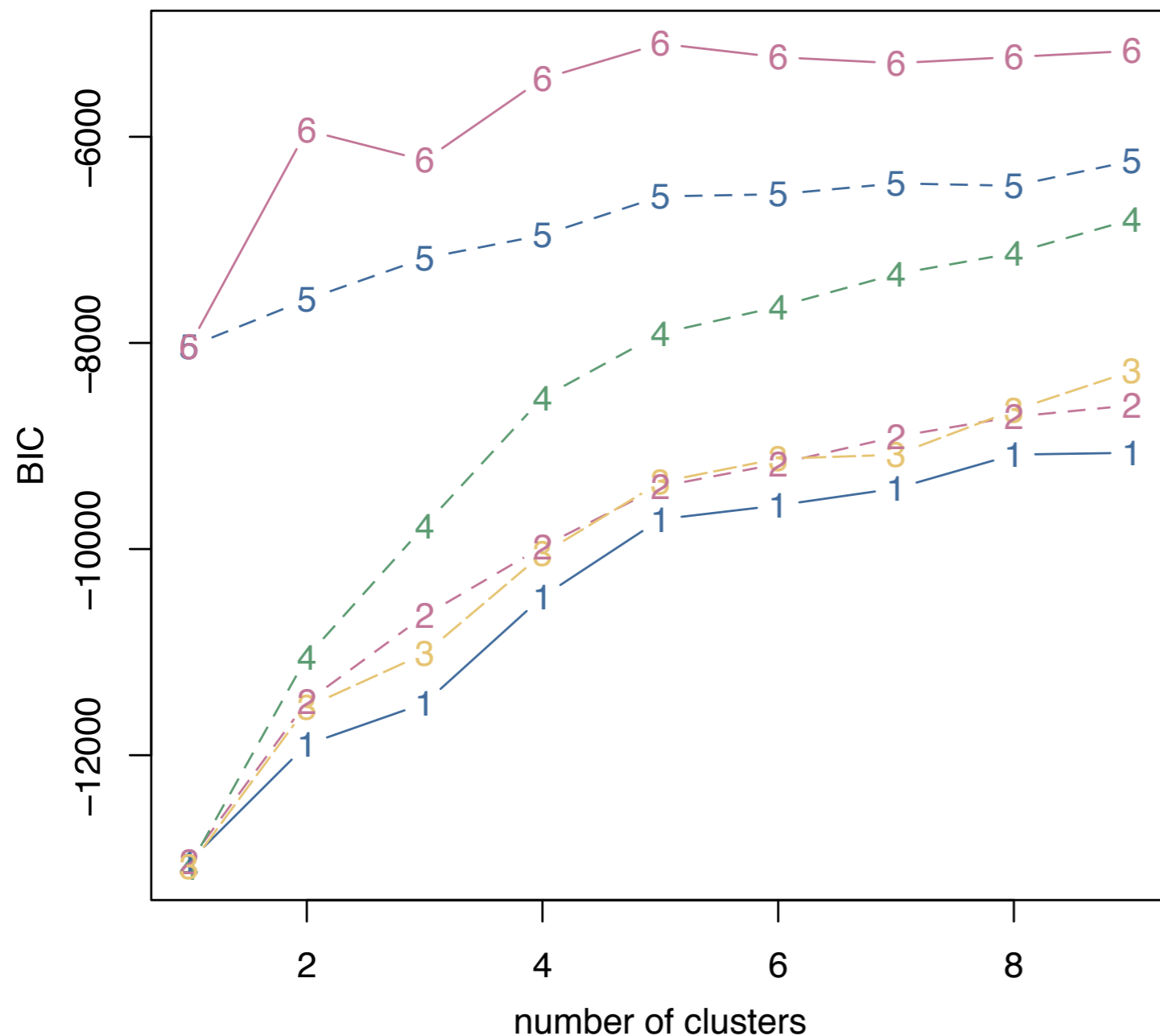
- Silhouette Plot: `silhouette`, oder `plot(partition.object)`
- `plot(hierarchical.object)` liefert ein Dendrogramm.
- `clusplot(hierarchical.object)` zeichnet einen 2-dim Plot der Daten via Hauptkomponenten und Multidimensionaler Skalierung und trägt dort Ellipsen “um” die Cluster ein.





Mclust Library

- Die `mclust` Library ist nicht Teil der Standarddistribution von R.
- Diagnostische Plots:



"VVV": ellipsoidal, varying volume, shape, and orientation

"EEE": ellipsoidal, equal volume, shape, and orientation

"VVI": diagonal, varying volume, varying shape

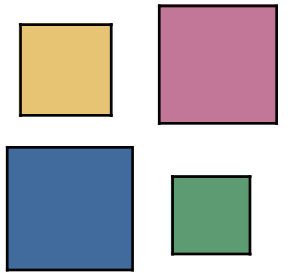
"EEI": diagonal, equal volume, equal shape

"VII": spherical, unequal volume

"EII": spherical, equal volume

Weitere Plots:

- Pairs
- Classification
- Uncertainty



Cluster Analyse: Zusammenfassung

- Der Vergleich von Cluster Ergebnissen mit Klassifikationsproblemen ist nützlich, aber mit Vorsicht zu betrachten.
- Ohne Vorkenntnisse über die Daten und potentielle Cluster sind die meisten Verfahren nicht sehr nützlich.
- Eine eingehende graphische und explorative Analyse der erhaltenen Cluster ist unbedingt notwendig.
- Die Integration von kategoriellen Variablen ist aufwendig.
- Alle Implementierungen der Verfahren sollten kritisch geprüft werden.
- Im Data Mining sind die klassischen Cluster Analyse Algorithmen die auf Metriken auf stetigen Daten basieren nicht anwendbar, da dort die Daten in einem hohen Maße diskret sind.
↳ andere Methoden, z.B. Assoziations Regeln etc.