Seminar zum Thema Künstliche Intelligenz: Clusteranalyse

Wolfgang Ginolas

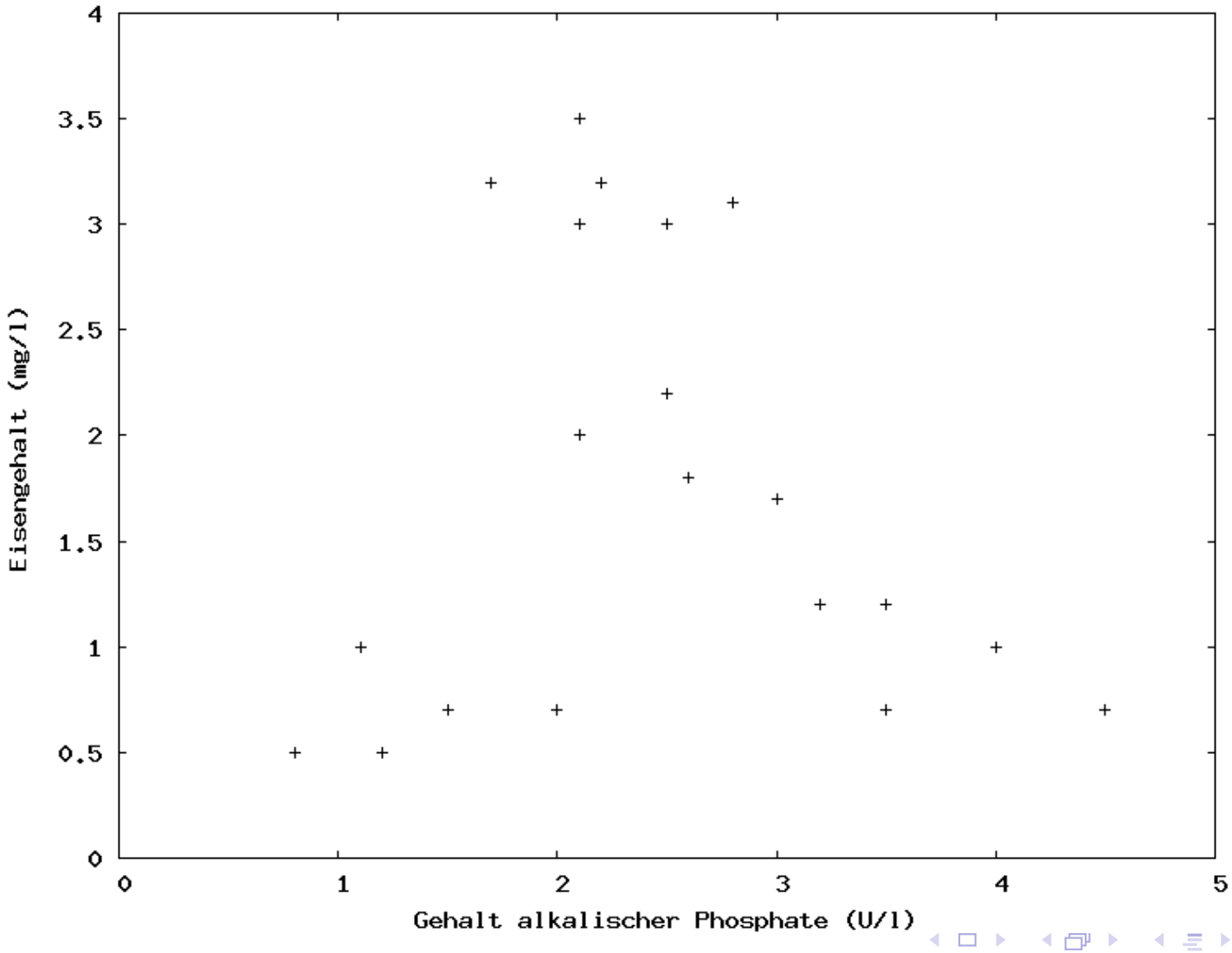
11.5.2005

Blutanalysen von 20 Patienten

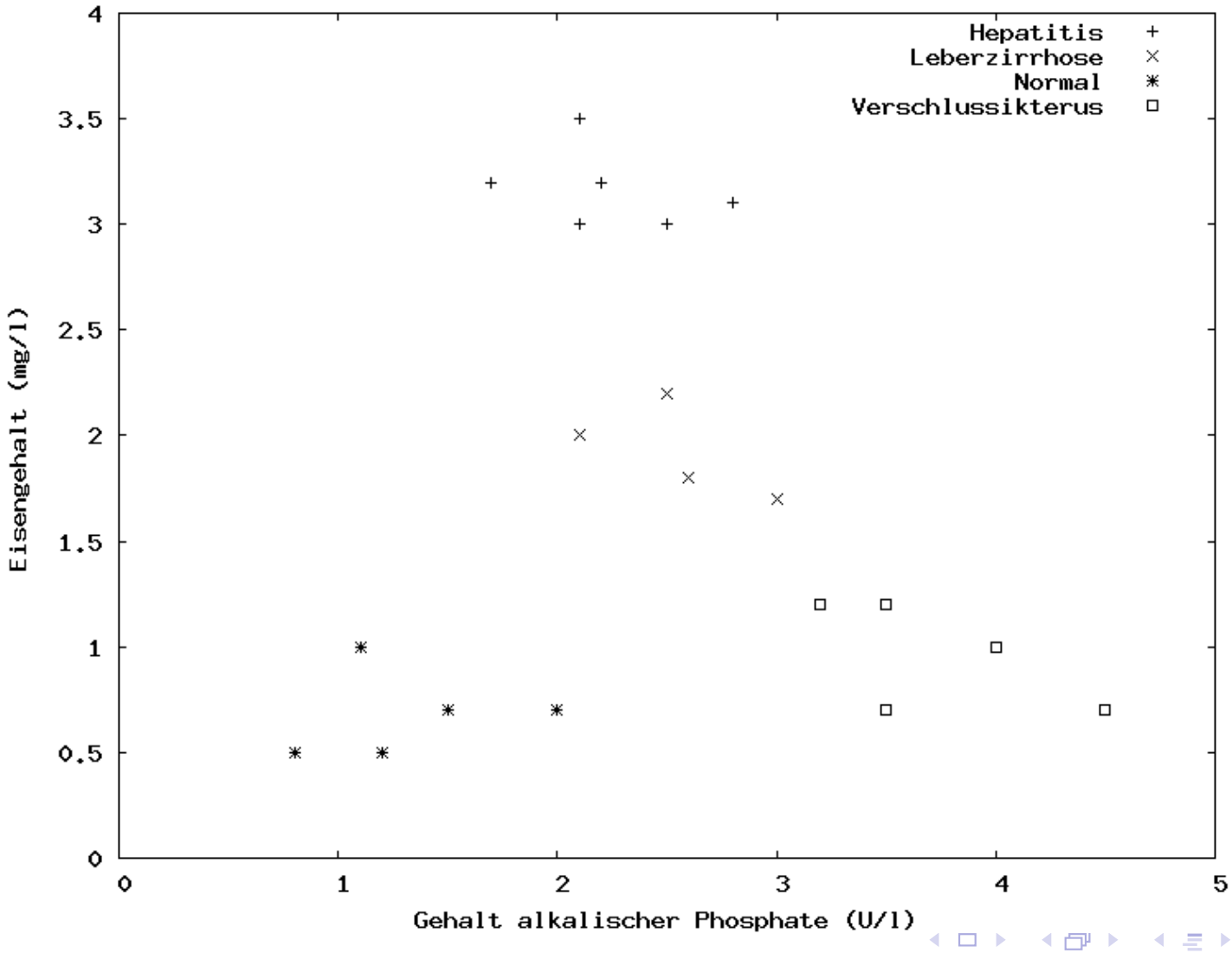
Gemessen wurden der „Gehalt alkalischer Phosphate“ und der „Eisengehalt“

Pat. Nr.	AP (U/l)	Fe (mg/l)	Pat. Nr.	AP (U/l)	Fe (mg/l)
1	4.0	1.0	11	2.0	0.7
2	3.0	1.7	12	1.2	0.5
3	2.6	1.8	13	4.5	0.7
4	1.5	0.7	14	2.5	3.0
5	2.5	2.2	15	3.5	0.7
6	1.1	1.0	16	2.2	3.2
7	2.8	3.1	17	2.1	3.5
8	1.7	3.2	18	2.1	2.0
9	0.8	0.5	19	3.5	1.2
10	2.1	3.0	20	3.2	1.2

Grafische Darstellung



Grafische Darstellung, Clustern zugeordnet



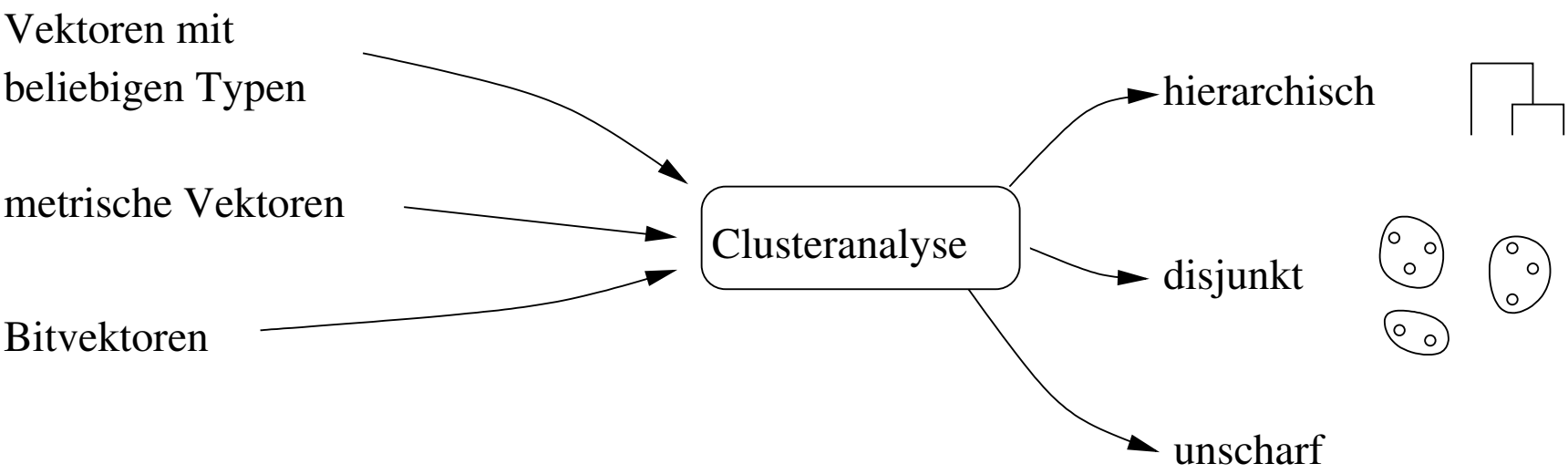
Was ist eine Clusteranalyse

Gegeben: Stichprobe von Objekten, die sich in eine noch unbekannte Anzahl von Gruppen ähnlicher Objekte unterteilen lässt.

Gesucht: Charakterisierung dieser potentiellen Gruppen und die Angabe, welches Objekt welcher Gruppe zugewiesen ist.

Lösungsverfahren: Clusteranalyse

Ein- und Ausgabe



Allgemeines Vorgehen bei der Clusteranalyse

1. Definition der Objekte
2. Auswahl und Aufbereitung der Merkmale
3. Auswahl des Clusterverfahrens
4. Die Interpretation der Ergebnisse

Anwendungen der Clusteranalyse

- Datenanalyse
- Mustererkennung
- Data-Mining
- Medizin
- Bildverarbeitung
- Web-Controlling
- *Computergrafik*
- *Sortieren einer Musiksammlung*

Ein einfacher Algorithmus

1. Alle Objekte bilden jeweils einen eigenen Cluster.
2. Die beiden Cluster, die sich am nächsten sind, werden vereinigt.
3. Wiederhole 2, bis die gewünschte Clusterzahl erreicht ist.

Problem: Was bedeutet „sich am nächsten“?

Metrik

Falls für d gilt:

1. $d(x, y) \geq 0$
2. $d(x, y) = 0 \iff x = y$
3. $d(x, y) = d(y, x)$
4. $d(x, y) \leq d(x, z) + d(z, y)$

so definiert d ein Distanzmaß.

Metrische/Reelle Daten

Metrische Daten lassen sich in zwei Arten unterteilen:

Intervallskaliert: Intervallskalierte Daten haben keinen vorgegebenen Bezugs-/Nullpunkt (z.B. Temperatur, Datum etc.)

Verhältnisskaliert: Verhältnisskalierte Daten haben einen vorgegebenen Nullpunkt (z.B. Länge, Gewicht)

Distanz

$$d_r(x, y) = |x - y|$$

Ordinale Daten

Ordinale Daten haben eine Rangfolge, so dass Vergleichsoperationen angewandt werden können.

Um eine Distanz zu bestimmen, lassen sich ordinale Daten leicht in metrische umwandeln. Z.B.:

$Dialup = 0 \qquad ISDN = 1 \qquad Broadband = 2 \qquad Cable = 3$

Nominale Daten

Bei nominalen Typen lässt sich der „Abstand“ z.B. durch Gleichheit und Ungleichheit definieren: $d_n(x,y) = \begin{cases} 0 & \text{für} \quad x = y \\ 1 & \text{für} \quad x \neq y \end{cases}$

Merkmalsvektor

Ein Merkmalsvektor fasst alle Merkmale eines Objektes zusammen.

- Ein Zusammenhang zwischen Merkmalen (z.B. Körpergewicht und Körpergröße) sollte vermieden werden, da er das Ergebnis verzerrt
- Das Ausschließen von Merkmalen durch andere (z.B. „Treiben Sie Sport“, „Spielen Sie Tennis“) ist zu vermeiden oder beim Distanzmaß zu berücksichtigen

Euklidische Distanz

Euklidische Distanz

$$d_e(x, y) = \sqrt{\sum_i d_i(x_i, y_i)^2}$$

Die Euklidische Distanz ist nicht *skaleninvariant*, d.h. ändert man z.B. die Maßeinheit eines Merkmals (Gramm $\Leftrightarrow$ Kilogramm), so verschieben sich auch die Distanzen.

Normalisieren: z-Transformation

Bei metrischen Objekten lässt sich ein Merkmal x der Objekte i mittels der z-Transformation normieren:

$$\begin{aligned} \overline{x} &= \frac{1}{N} \cdot \sum_{i=1}^{N} x_i \\ \sigma x &= \sqrt{\frac{1}{N} \cdot \sum_{i=1}^{N} (x_i - \overline{x})^2} \\ x_i' &= \frac{x_i - \overline{x}}{\sigma x} \end{aligned}$$

Normalisieren der Distanzmatrix

1. Distanzmatrix bezüglich eines Merkmals berechnen:

d_{M1}	Objekt 1	Objekt 2	Objekt 3
Objekt 1	0	1	2
Objekt 2	1	0	1
Objekt 3	2	1	0

2. Die größte Distanz ermitteln: $d_{M1Max} = 2$
3. Die Matrix durch diese Distanz teilen:

$$\begin{pmatrix} 0 & 1 & 2 \\ 1 & 0 & 1 \\ 2 & 1 & 0 \end{pmatrix} \cdot \frac{1}{2} = \begin{pmatrix} 0 & \frac{1}{2} & 1 \\ \frac{1}{2} & 0 & \frac{1}{2} \\ 1 & \frac{1}{2} & 0 \end{pmatrix}$$

Gewichten

Gewichte für einzelne Merkmale:

- Multiplikation der Merkmale mit einem konstanten Faktor
- Multiplikation der Distanzmatrix eines Merkmals mit einem konstanten Faktor

Gewichte für einzelne Objekte:

- Muss von dem Clusterverfahren unterstützt werden

Fehlwerte in einem Objekt

Der Wert eines Merkmals in einem Objekt ist nicht bekannt:

- Das ganze Objekt wird verworfen.
- Man wählt ein Distanzmaß, das Fehlwerte berücksichtigt.

Fehlwerte: Beispiel Euklidische Distanz

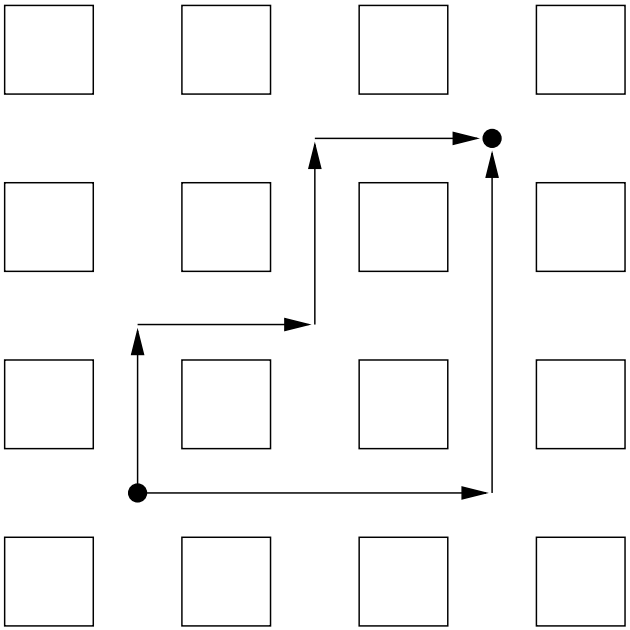
Fehlwerte könnten z.B. bei der Euklidischen Distanz durch weglassen des entsprechenden Merkmals bei der Summenbildung berücksichtigt werden.

$$x = (1, 1) \quad y = (2, *) \quad z = (4, 5)$$

$$\begin{aligned} d(x,y) &= \sqrt{(1-2)^2} &&= 1 \\ d(x,z) &= \sqrt{(1-4)^2 + (1-5)^2} &&= 5 \\ d(y,z) &= \sqrt{(2-4)^2} &&= 2 \end{aligned}$$

Achtung: Dies ist keine Metrik mehr, da: $d(x,z) > d(x,y) + d(y,z)$

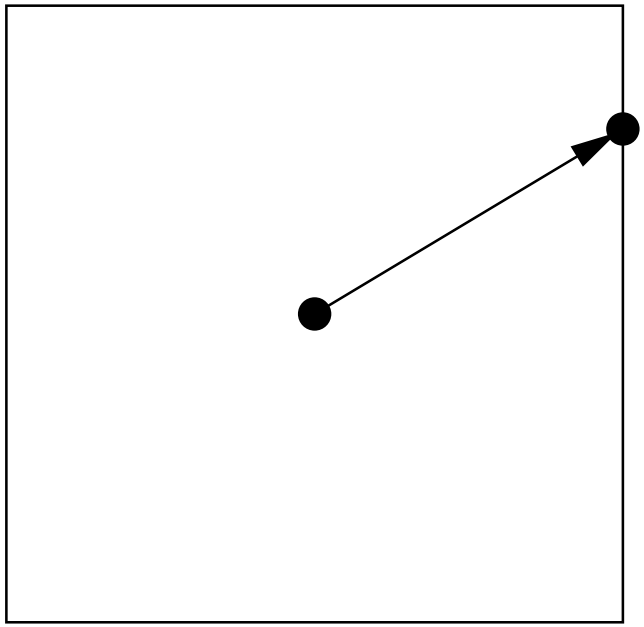
Häuserblockmetrik



Häuserblockmetrik

$$d_h(x, y) = \sum_i |d_i(x_i, y_i)|$$

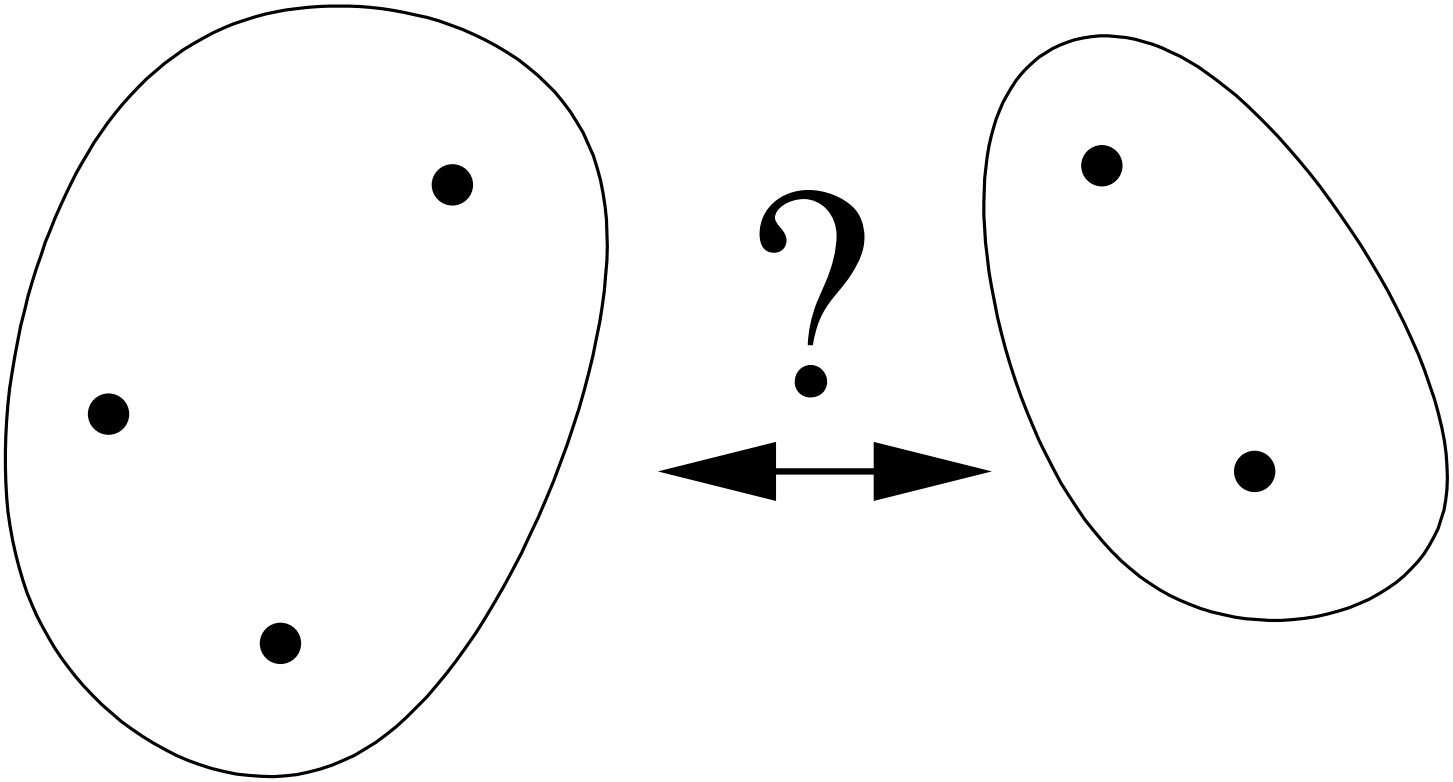
Maximum-Abstand



Maximum-Abstand

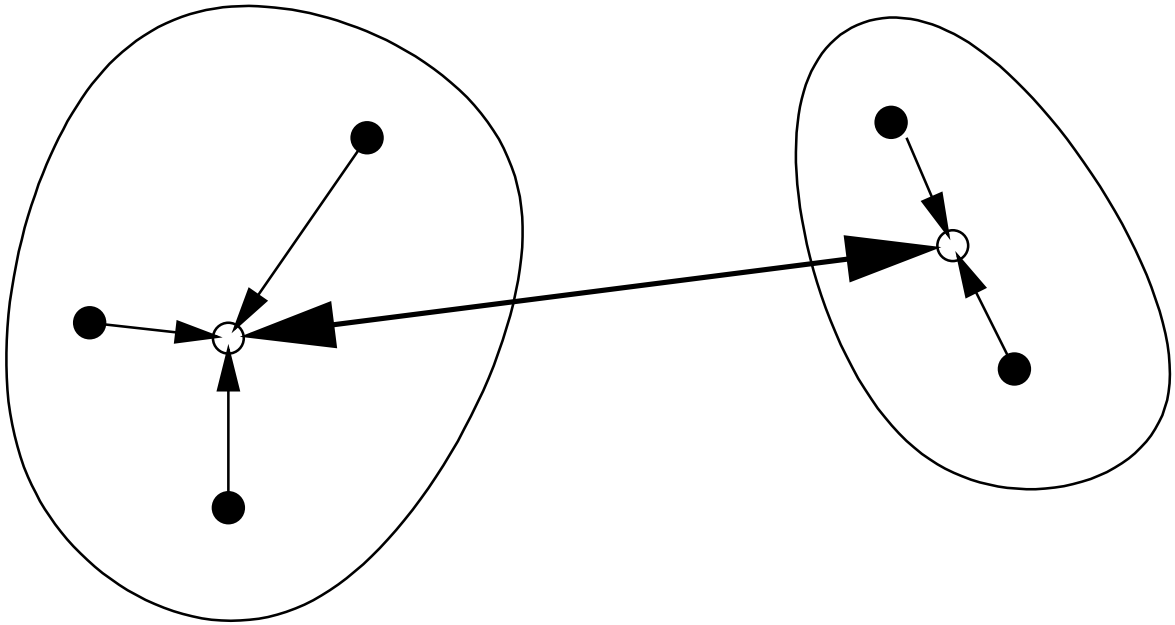
$$d_m(x, y) = \max(d_1(x_1, y_1), d_2(x_2, y_2), \ldots, d_n(x_n, y_n))$$

Abstände zwischen Clustern



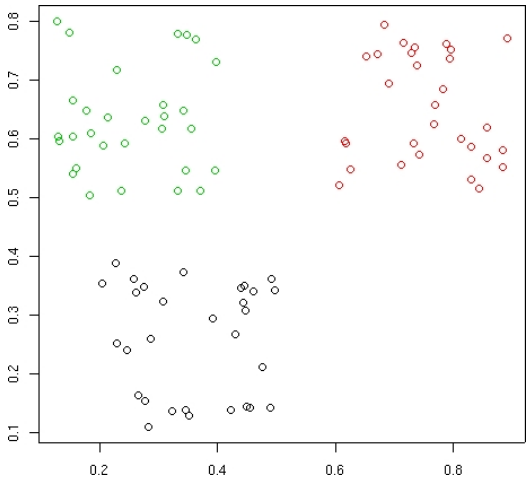
Zentroid Verfahren

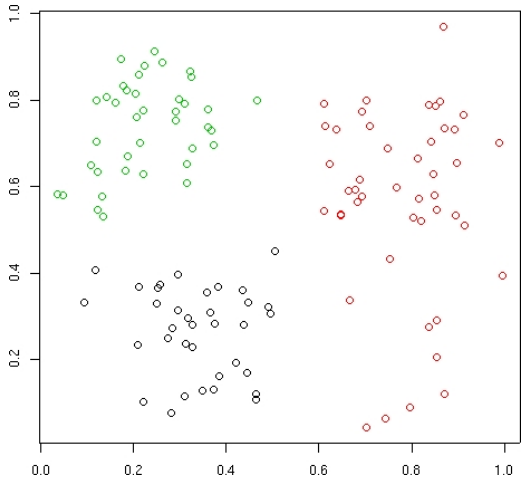
Bei dem *Zentroid Verfahren* entspricht die Distanz dem Abstand der Schwerpunkte (Zentroide) der beiden Cluster.

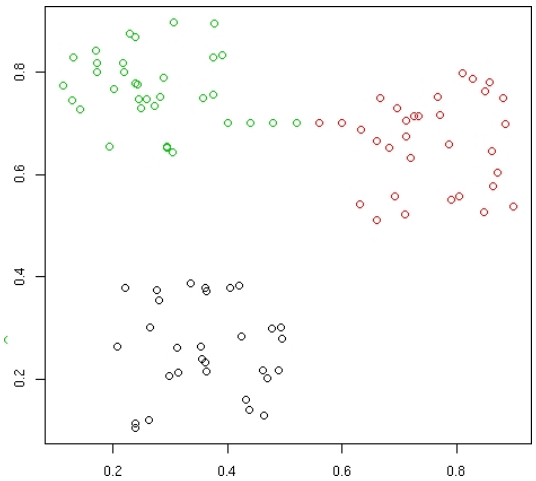


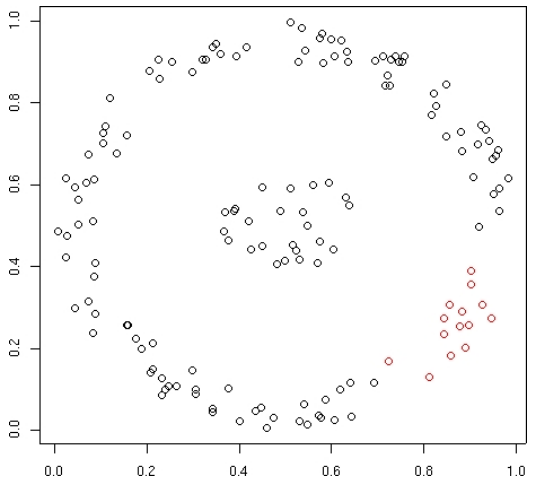
Achtung: Ein Schwerpunkt lässt sich nur bei metrischen Daten berechnen!

Beispiel: Zentroid Verfahren



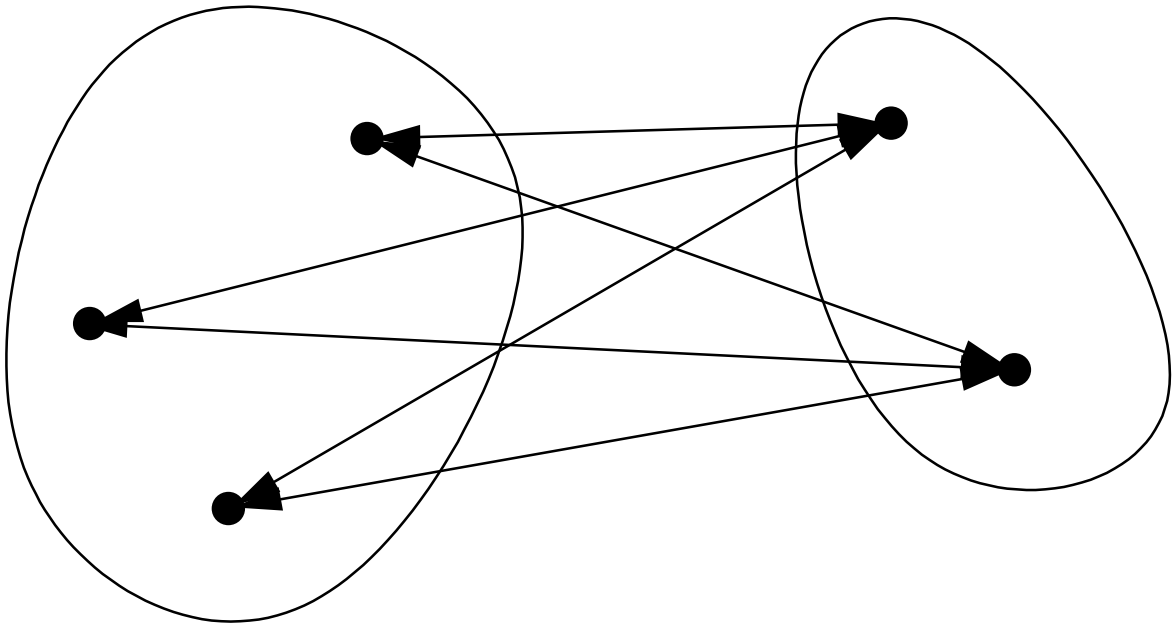




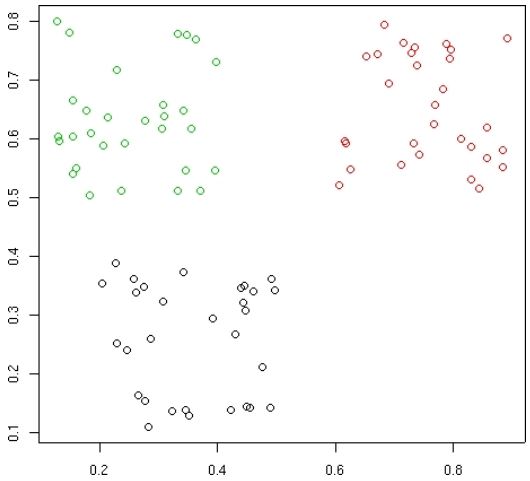


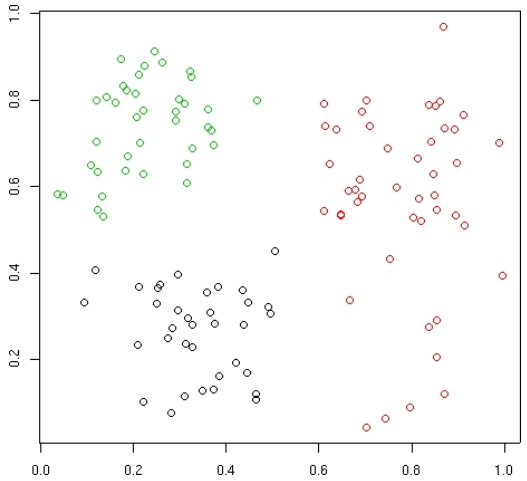
Average-Linkage-Verfahren

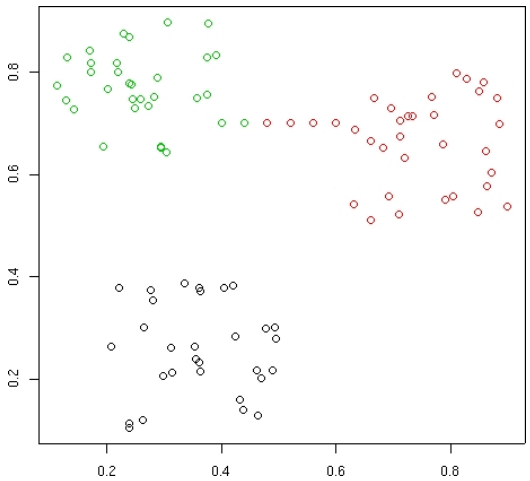
Bei dem *Average-Linkage-Verfahren* entspricht die Distanz dem Mittelwert aller Abstände zwischen den Objekten beider Cluster.

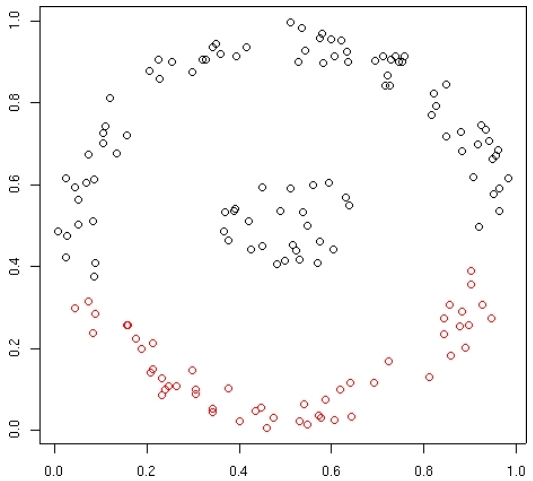


Beispiel: Average-Linkage-Verfahren



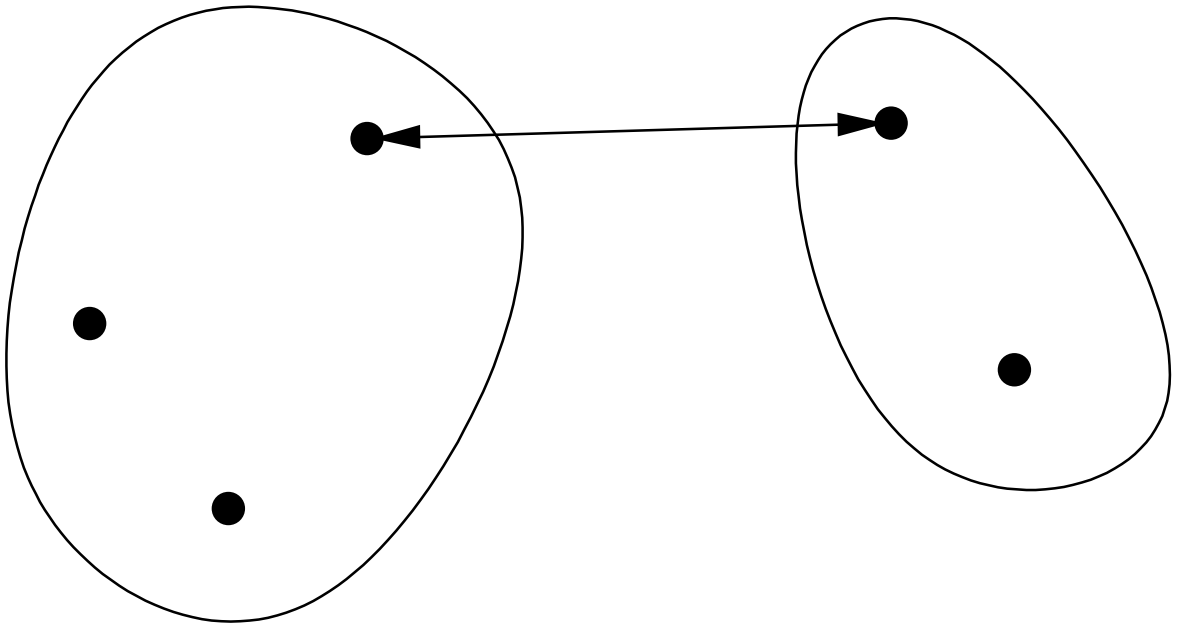




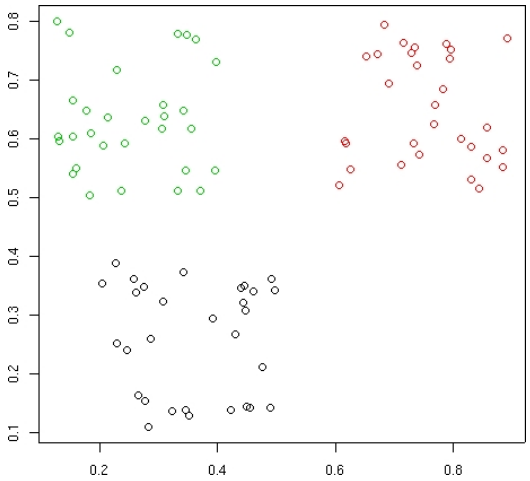


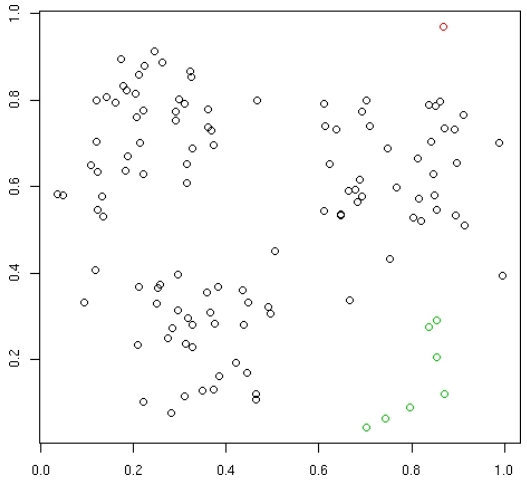
Single-Linkage-Verfahren

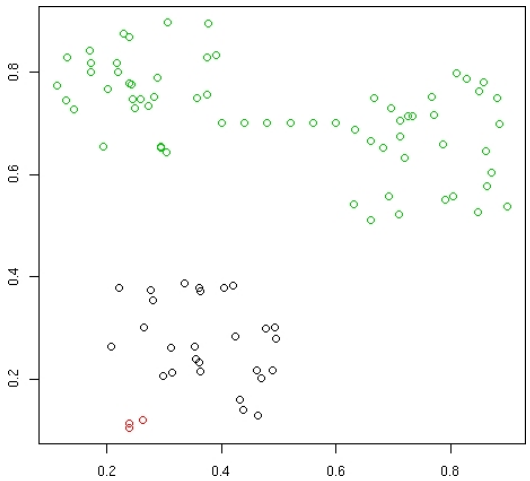
Bei dem *Single-Linkage-Verfahren* entspricht die Distanz dem Abstand der Objekte, die sich am nächsten sind.

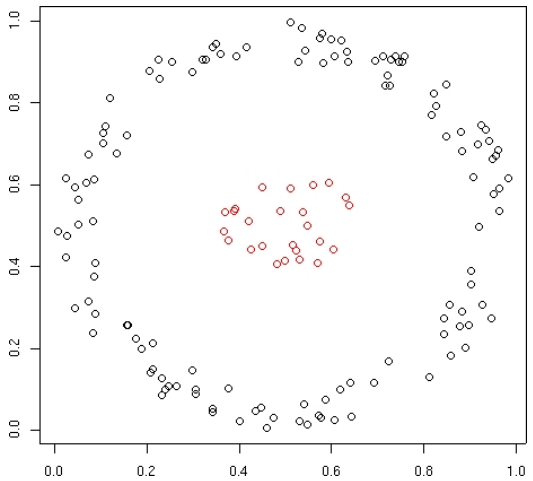


Beispiel: Single-Linkage-Verfahren



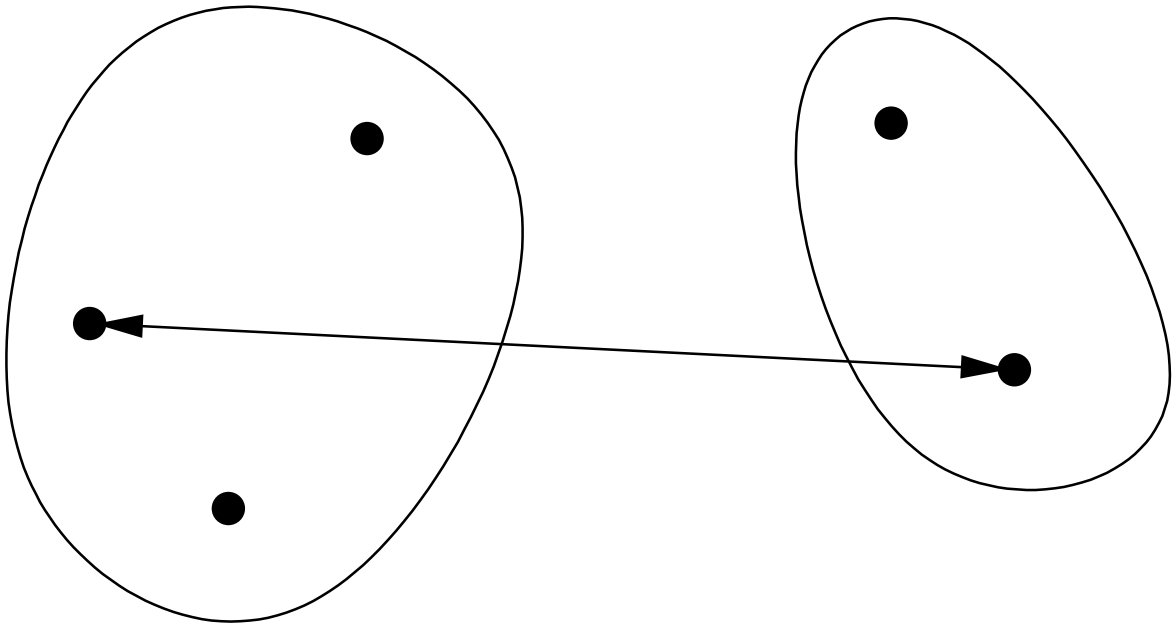




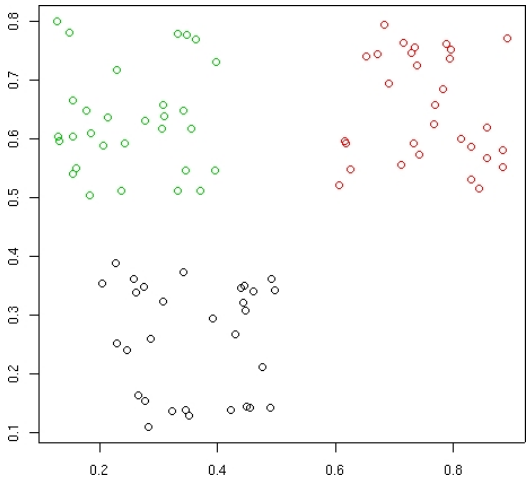


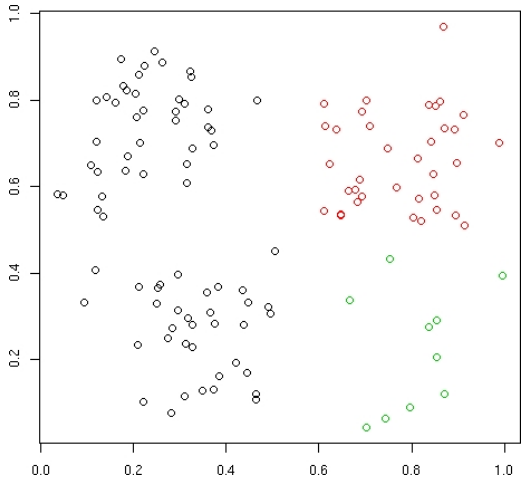
Complete-Linkage-Verfahren

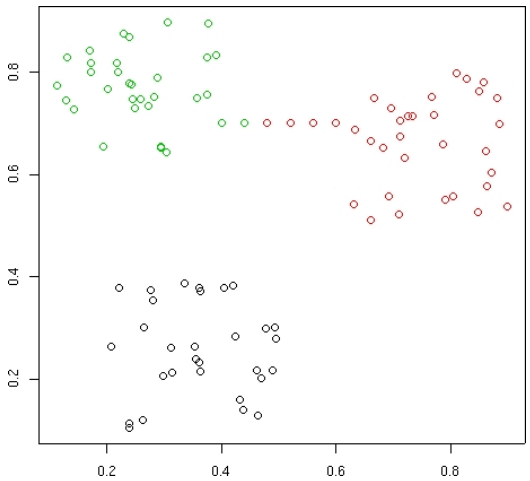
Bei dem *Complete-Linkage-Verfahren* entspricht die Distanz dem Abstand der Objekte, die am weitesten voneinander entfernt sind.

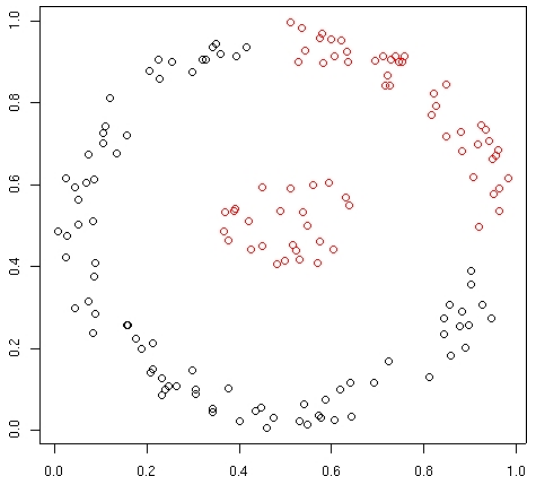


Beispiel: Complete-Linkage-Verfahren

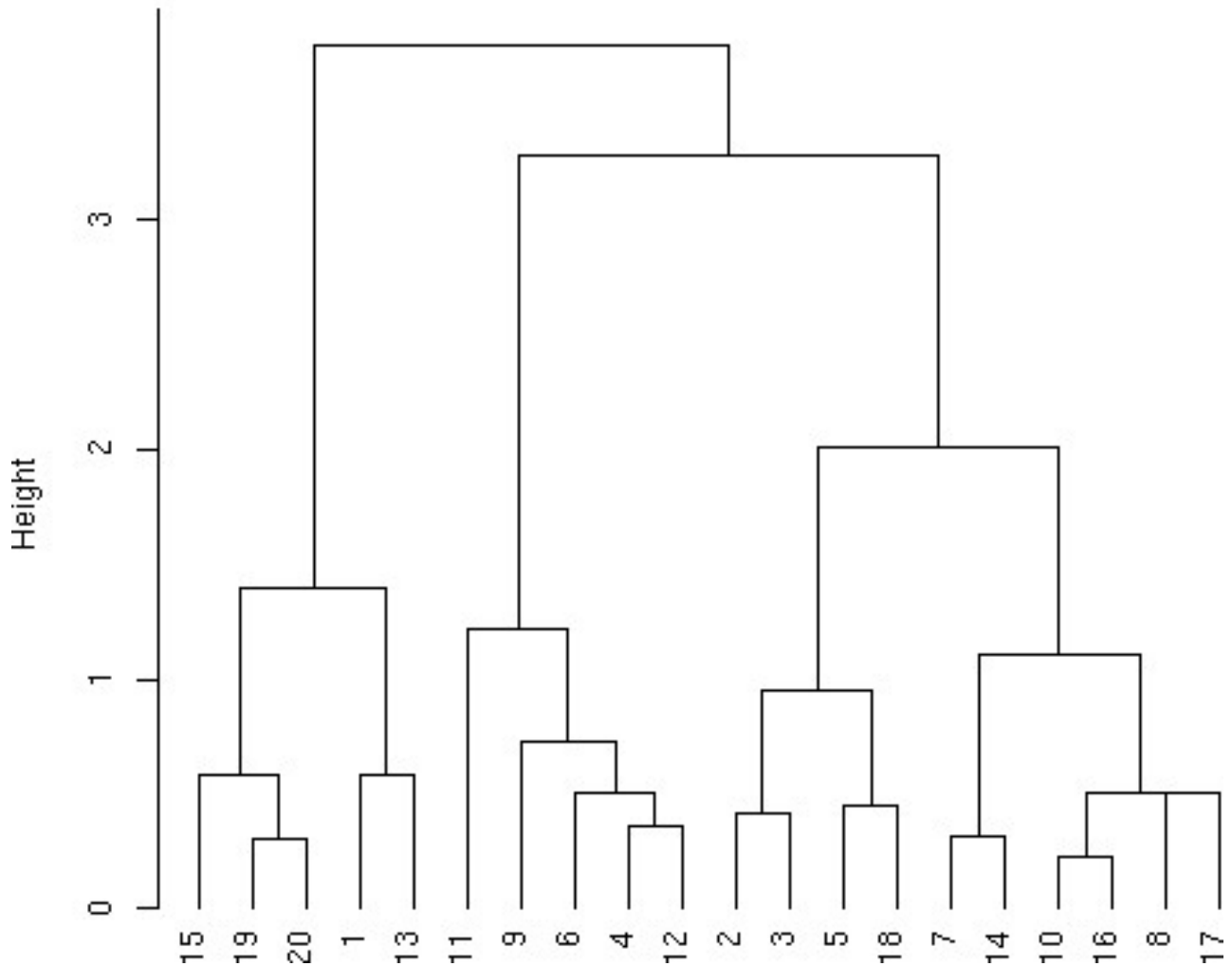








Dendrogramm



Zwei verschiedene Herangehensweisen

Agglomerativ

- Ausgangszustand: Jedes Objekt bildet einen eigenen Cluster.
- Wiederholen: Die beiden Cluster, die sich am nächsten sind, werden verschmolzen.

Divisiv

- Ausgangszustand: Alle Objekte bilden einen großen Cluster.
- Wiederholen: Ein „optimaler“ Cluster wird gewählt und sinnvoll in zwei neue zerteilt.

Agglomerativ: Algorithmus

1. Alle Objekte bilden jeweils einen eigenen Cluster
2. Die beiden Cluster, die sich am nächsten sind, werden vereinigt
3. Vereinigte Cluster und deren Distanz werden in das Dendrogramm eintragen
4. Wiederhole 2-3, bis nur noch ein Cluster übrig ist

Optimierung: Die neuen Distanzen, nach einer Vereinigung, können relativ einfach aus den alten berechnet werden. (Siehe: Nakhaeizadeh, S. 129ff)

Divisiv

Divisive Verfahren sind rechenaufwändiger als Agglomerative und werden deswegen kaum verwendet.

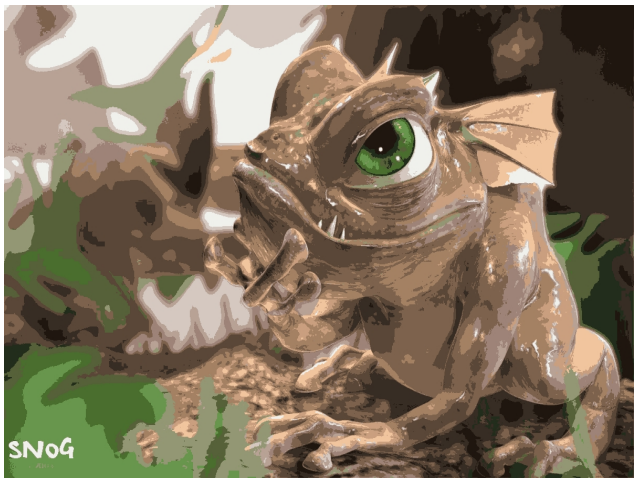
Aber...

Divisive Klassifikation in der Computergrafik

Problem: Ältere Grafik-HW kann nur eine bestimmte Anzahl von Farben gleichzeitig anzeigen.

Lösung: Die Anzahl der Farben mittels Clusteranalyse reduzieren.

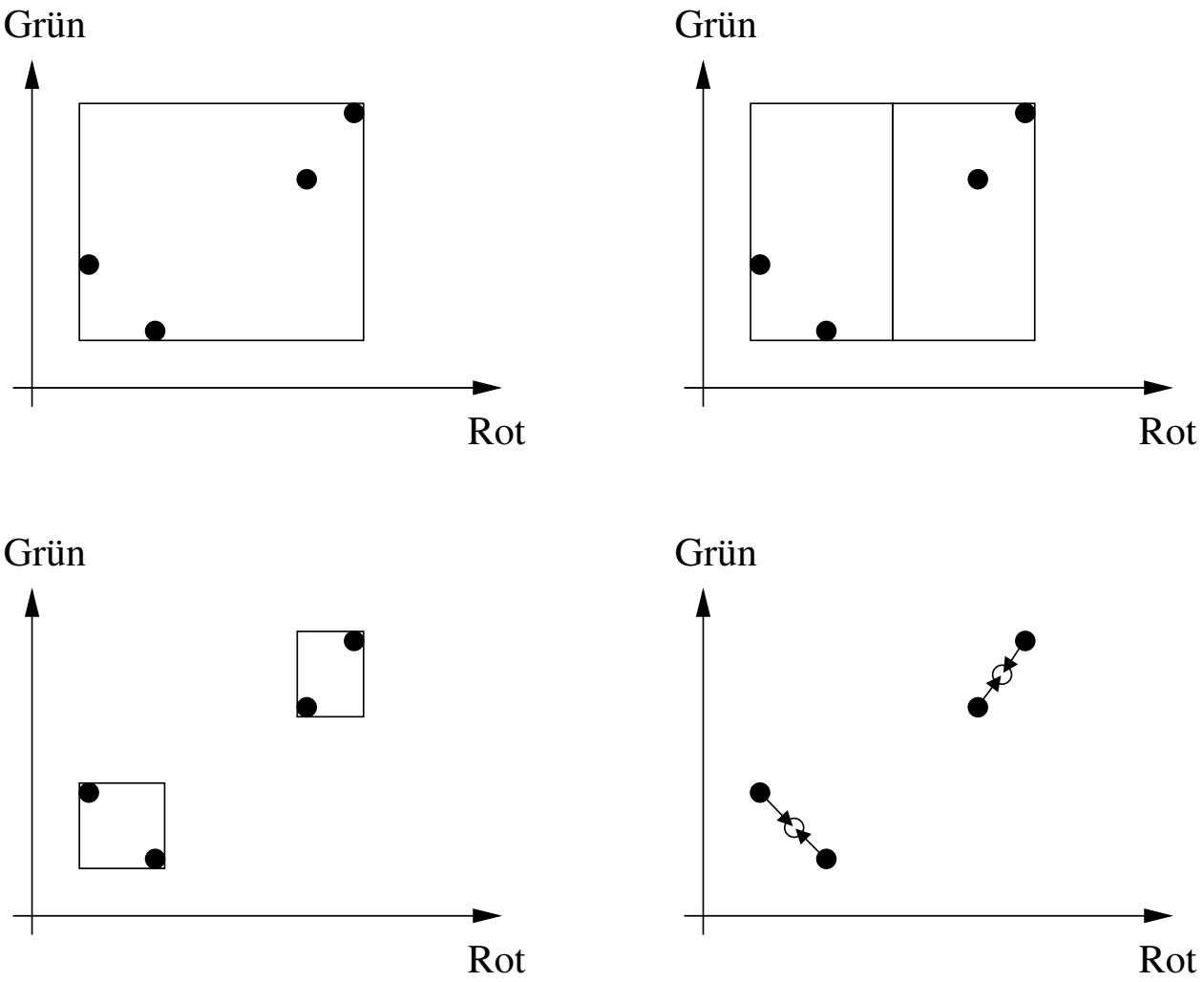
Reduzierung auf 16 Farben



Divisive Klassifikation in der Computergrafik: Algorithmus

1. *Ausgangssituation:* Ein möglichst kleiner Quader (Cluster), der alle Objekte umfasst.
2. Den Quader mit dem größten Volumen ermitteln.
3. Diesen Quader senkrecht zu seiner längsten Kante so teilen, dass in den neuen Quadern etwa gleich viele Objekte sind.
4. Die beiden neuen Quader soweit verkleinern, bis alle Objekte gerade noch enthalten sind.
5. Wiederhole 2-4, bis die gewünschte Cluster-/Farbanzahl erreicht ist.
6. Die Schwerpunkte der Cluster sind die neuen Farben.

Divisive Klassifikation in der Computergrafik: Beispiel



Disjunkte Klassifikation

- Jedes Objekt wird genau einem Cluster zugeordnet
- Cluster können in keiner Hierarchie angeordnet werden
- Disjunkte Verfahren sind in der Regel schneller als Hierarchische

Disjunkte Klassifikation: k-means

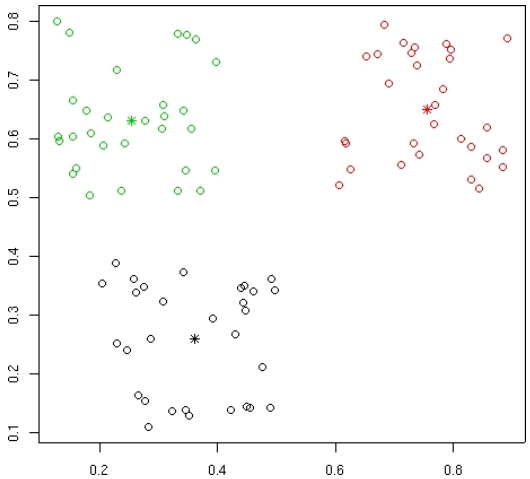
1. *Ausgangssituation:* (Zufällige) Auswahl von k Clusterzentren.
2. Jedes Objekt wird dem nächsten Clusterzentrum zugeordnet.
3. Neuberechnung der Clusterzentren.
4. Wiederhole 1-2 bis sich die Zuordnung der Objekte nicht mehr ändert.

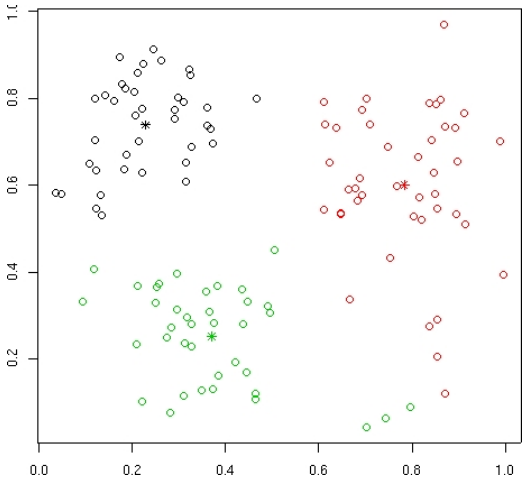
Disjunkte Klassifikation: k-means: Probleme

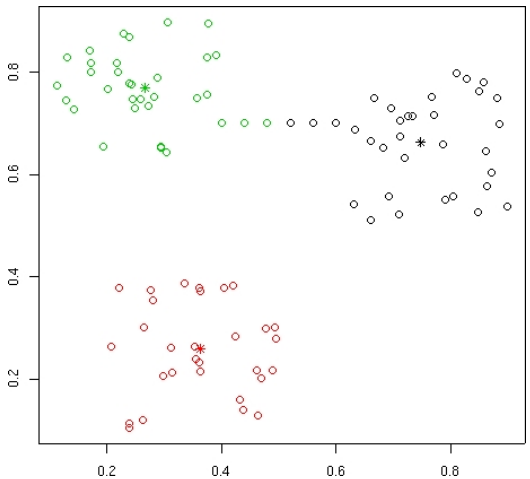
- k-means konvergiert nicht gezwungenermaßen.
- Cluster können leer bleiben, so dass kein neues Clusterzentrum berechnet werden kann.

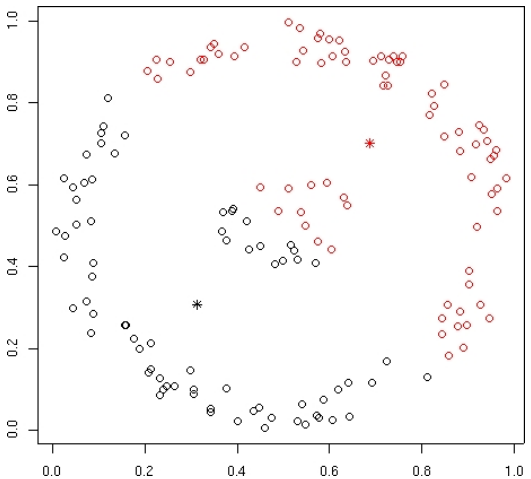
Aber: k-means gilt als „billig und gut“.

Disjunkte Klassifikation: k-means: Beispiele









Unscharfe Klassifikation

Zu jedem Objekt wird die Wahrscheinlichkeit ermittelt, mit der es in einem bestimmten Cluster ist.

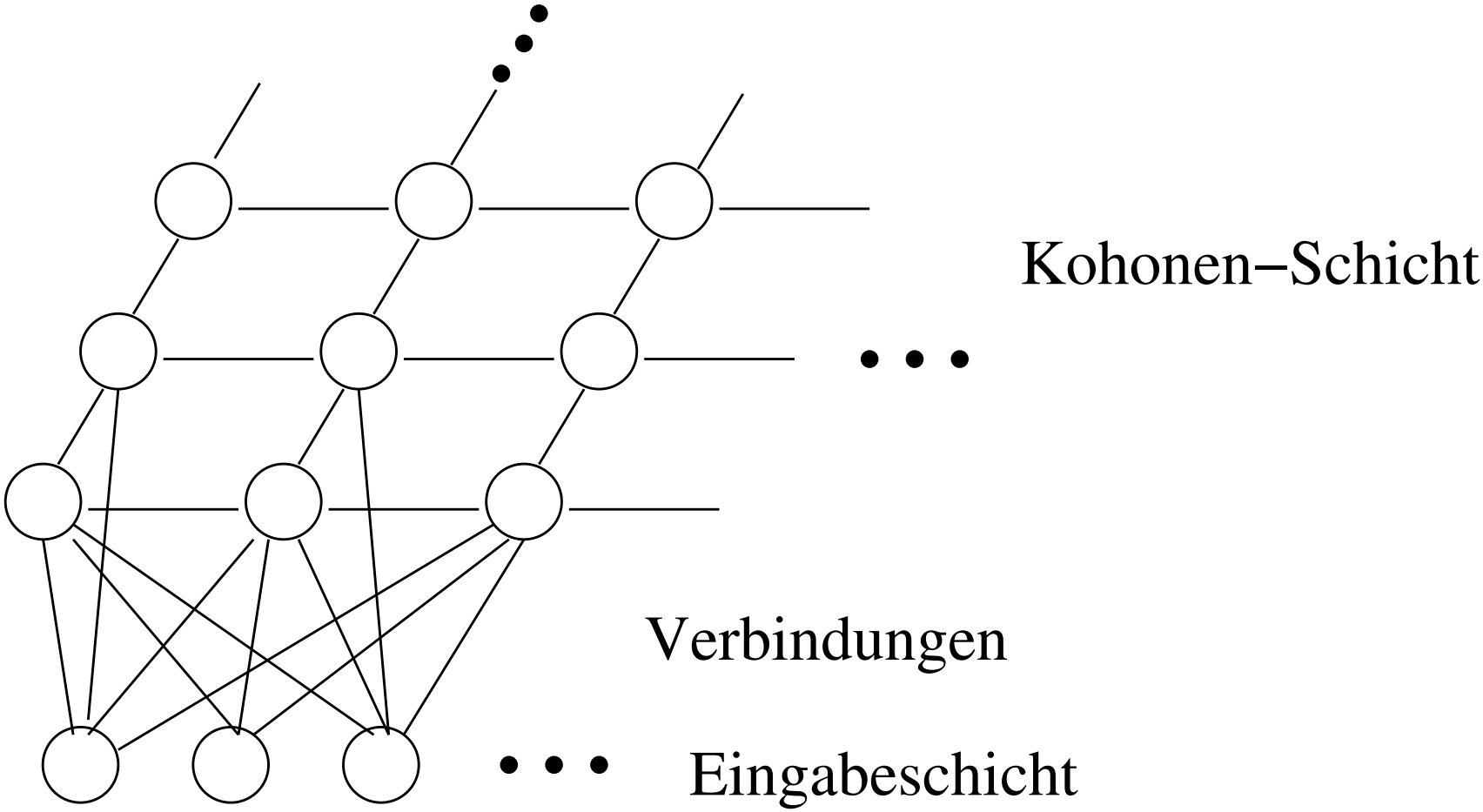
Ein mögliches Vorgehen:

1. *Ausgangssituation:* Es wurde bereits eine disjunkte Klassifikation durchgeführt.
2. Abstand zwischen Objekten o und Clustern c berechnen: $d(o, c)$
3. Die Wahrscheinlichkeit ist proportional zu: $e^{-d^2(o,c)}$
4. Wahrscheinlichkeiten normalisieren

Self-Organizing Maps/Kohonennetze

- Kohonennetze sind neuronale Netze mit 2 Schichten.
- Die zwei Schichten sind vollständig verbunden.
- Unüberwachtes Lernverfahren.
- Die Eingabeschicht ist der Merkmalsvektor.
- Die Ausgabeschicht ist eine Karte.
- Ähnliche Eingabevektoren aktivieren benachbarte Neuronen auf der Karte.

Self-Organizing Maps: Struktur



Self-Organizing Maps: Anwendung

Benutzung eines trainierten Kohonennetzes:

1. Anlegen eines Merkmalvektors an die Eingabeschicht.
2. Das Neuron der Ausgabeschicht, dessen Gewichtungen der Eingabeneuronen am ehesten entsprechen, ist das „Gewinner-Neuron“.

Self-Organizing Maps: Training 1

1. Alle Gewichte zufällig initialisieren.
2. Einen Trainingsvektor anlegen und Gewinner-Neuron ermitteln.
3. Die Gewichte des Gewinner-Neurons und seiner Nachtbarschaft dem Trainingsvektor ähnlicher machen.
4. 2-3 Wiederholen.

Self-Organizing Maps: Training 2

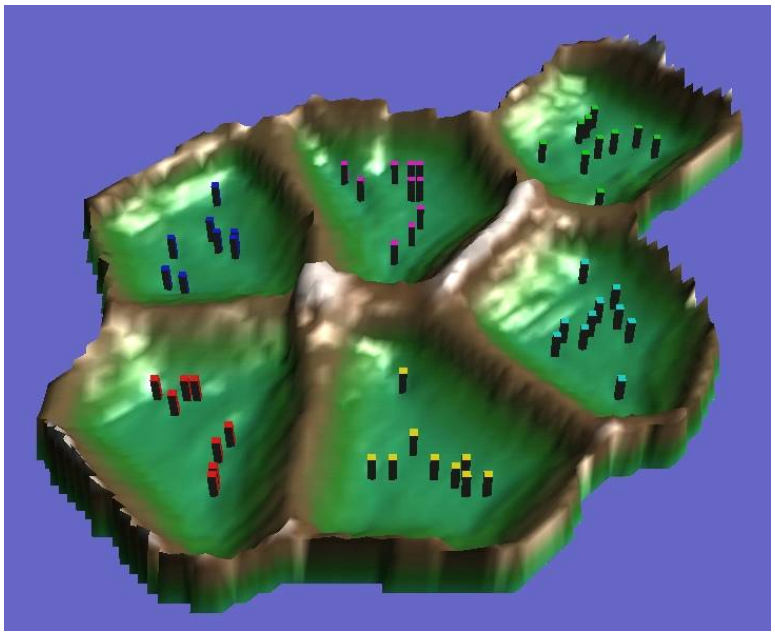
Nachtbarschaft: Die Nachtbarschaft eines Neurons kann z.B. durch eine Glockenkurve definiert werden.

Lernrate: Die Lernrate bestimmt, wieweit sich die Gewichte auf den Trainingsvektor zubewegen:
Lernrate = 1: Die Gewichte werden auf den Trainingsvektor gesetzt.
Lernrate = 0: Die Gewichte verändern sich nicht.

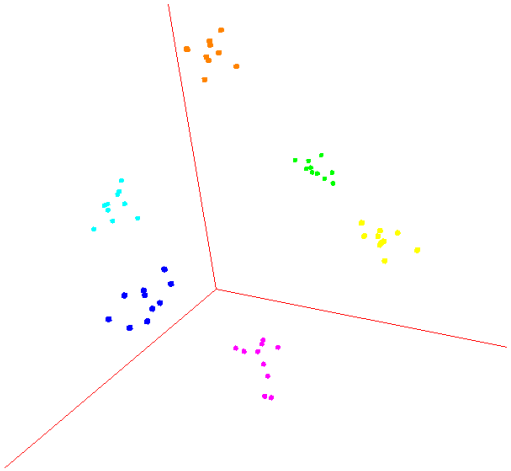
Die Größe der Nachtbarschaft und die Lernrate ist während des Trainings stetig zu verkleinern, bis das Netz stabil ist.

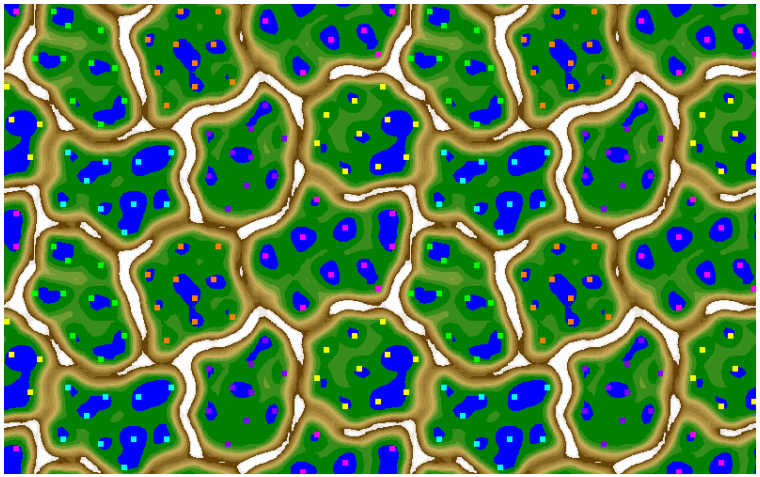
Wie lassen sich Self-Organizing Maps zur Clusteranalyse einsetzen?

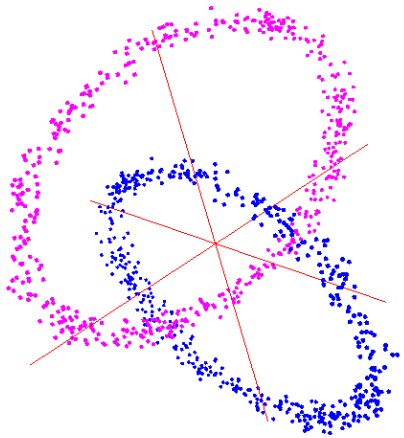
Die „Grenzen“ zwischen den Clustern können ermittelt werden, indem der Abstand der Gewichte der Kohonen berechnet wird.

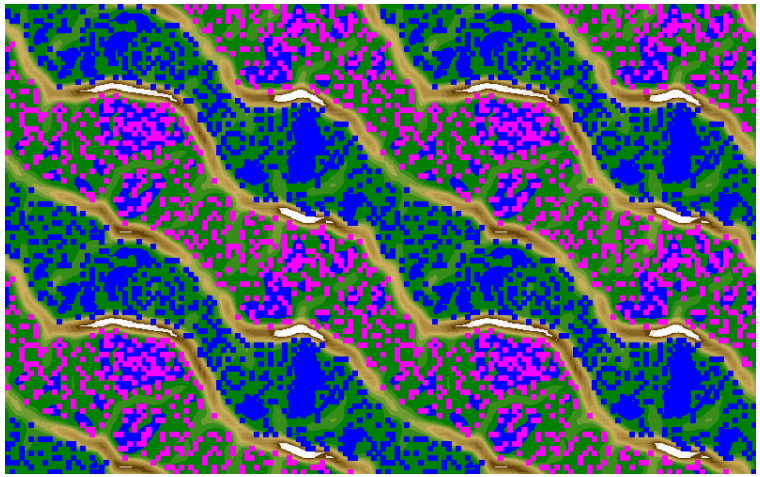


Self-Organizing Maps: Beispiele 1

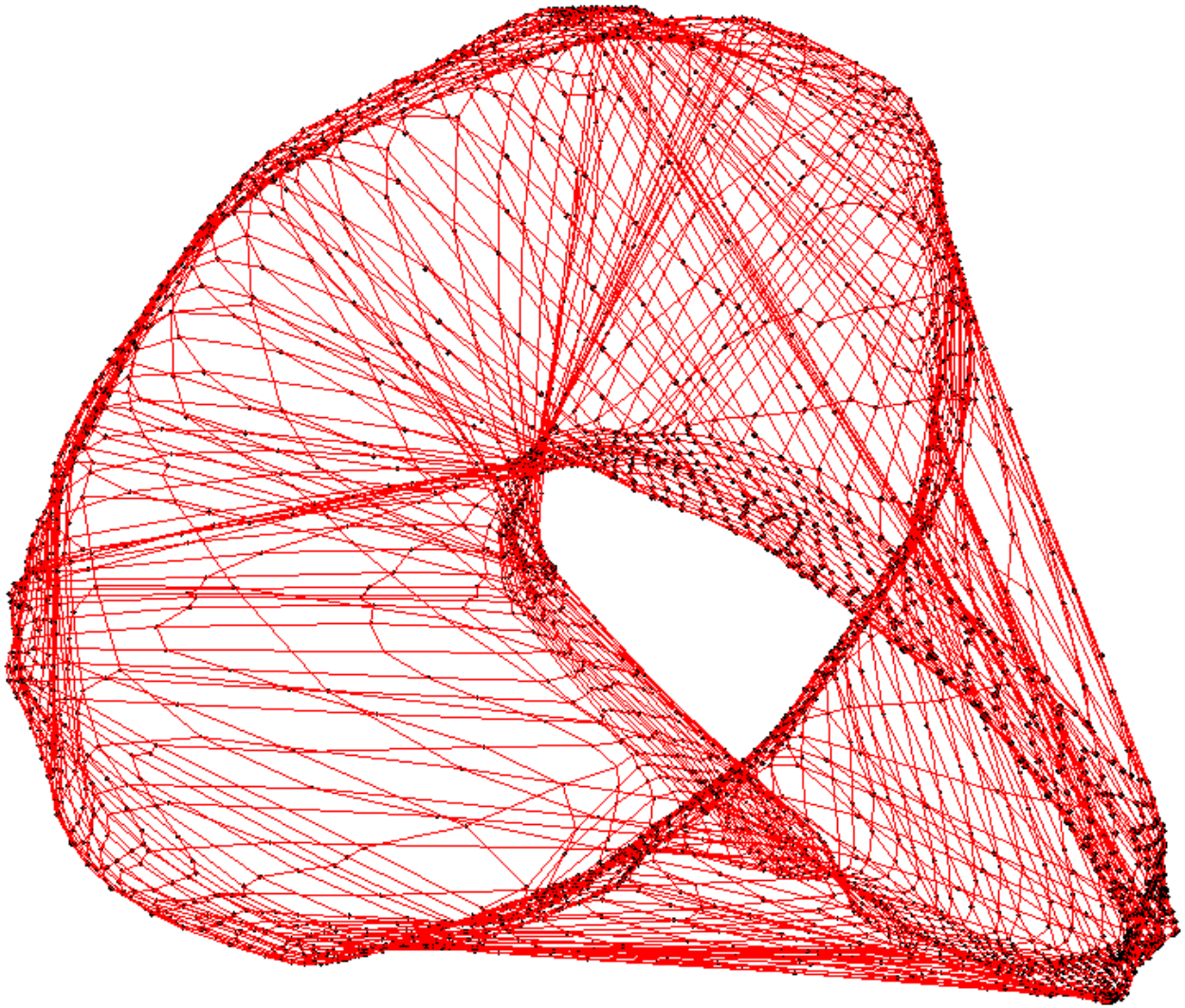








Self-Organizing Maps: Beispiele 2

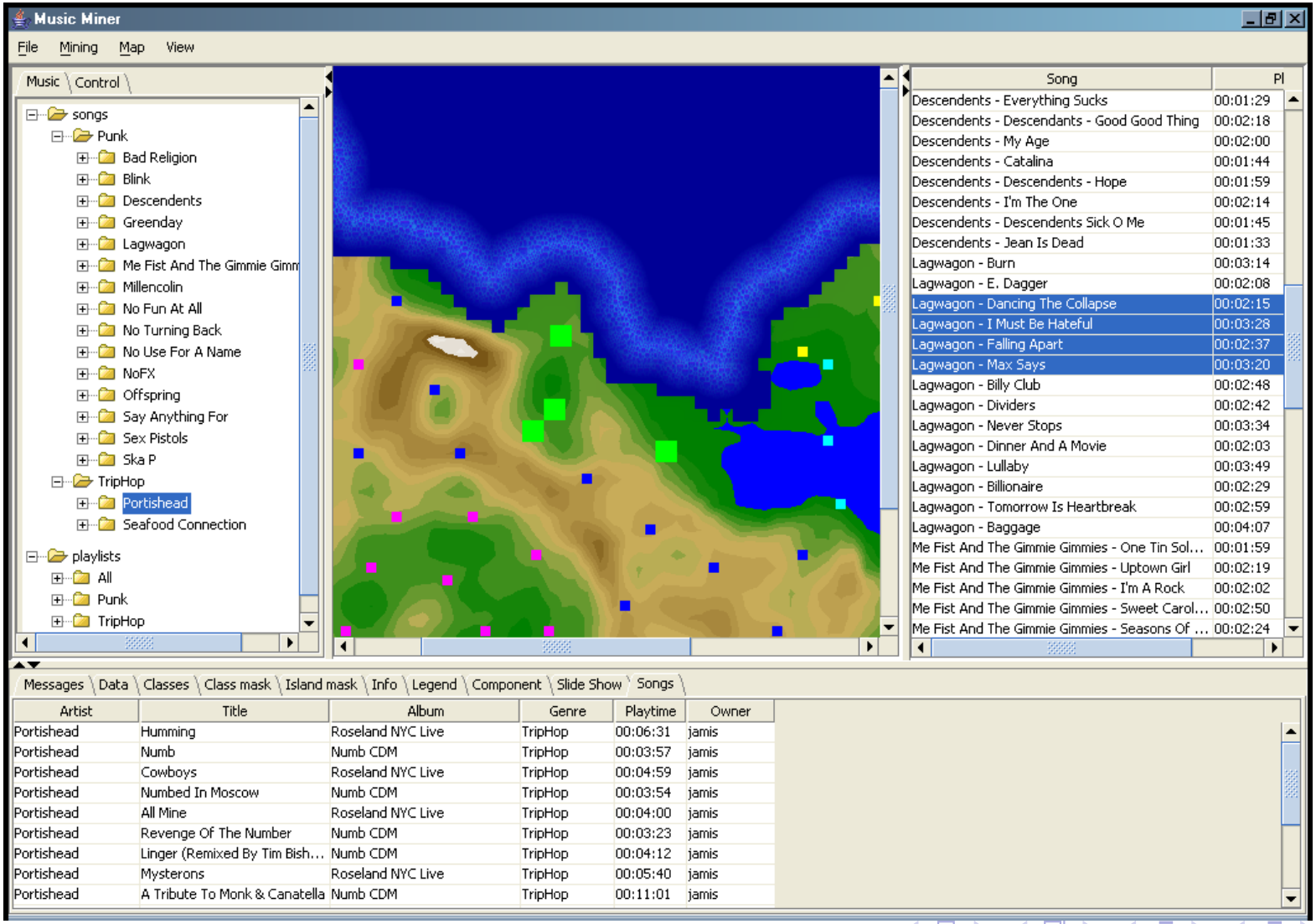


Self-Organizing Maps zur Analyse von Musik

Funktion der Software „MusicMiner“:

- *Objekte* sind einzelne Musikstücke.
- Die *Merkmale* werden anhand von Rhythmus und Frequenzverteilung ermittelt.
- Die Musikstücke werden in einer Kohonenkarte dargestellt.

MusicMiner: Beispiel 1



MusicMiner: Beispiel 2

Quellen

- Deichsel, Kap. 1-5
- Nakhaeizadeh, S. 109-141
- Diplomarbeit „Identifikation und Analyse von Besucherprofilen auf Websites“ von Michael Fait
- Scholl & Pfeiffer: „Natur als fraktale Grafik“, Markt&Technik
- *http://de.wikipedia.org/wiki/Clusteranalyse*
- *http://de.wikipedia.org/wiki/Self-Organizing_Maps*
- *http://www.mathematik.uni-marburg.de/~databionics/de//?q=esom*
- *http://musicminer.sourceforge.net/*
- *http://www.r-project.org/*