Universität Potsdam
Mathematisch-Naturwissenschaftliche Fakultät
Institut für Mathematik

Diplomarbeit

Statistische Eigenschaften von Clusterverfahren

Andrea Schorsch
Matrikelnummer: 715466
Sommersemester 2008

Erstkorrektor: Prof. Dr. Henning Läuter

Zweitkorrektor: Apl. Prof. Dr. Hannelore Liero

Eigenständigkeitserklärung

Hiermit versichere ich, die vorliegende Arbeit „Statistische Eigenschaften von Clusterverfahren“ eigenständig verfasst und alle verwendeten Hilfsmittel und Quellen angegeben zu haben.

Die Diplomarbeit hat in dieser oder ähnlicher Form keiner anderen Prüfungsbehörde vorgelegen.

Ort, Datum Unterschrift

Danksagung

Mein erster Dank geht an Herrn Prof. Dr. Henning Läuter und Frau Prof. Dr. Hannelore Liero für die Themenstellung, Betreuung und Unterstützung während dieser Diplomarbeit.

Für die grammatikalische Durchsicht bedanke ich mich herzlich bei meinem Freund und meinen Eltern. Ebenso möchte ich mich an dieser Stelle herzlich für deren Unterstützungen in jeglicher Hinsicht während der Dauer der Diplomarbeit und des Studiums bedanken.

Inhaltsverzeichnis

1 Einleitung

Clusterverfahren zählen zu den wesentlichen Bestandteilen der beschreibenden Statistik. Das Grundanliegen dieser statistischen Analysen ist das Auffinden von Beobachtungsvektoren in einer gegebenen Menge von mehrdimensionalen Daten, die einander ähnlich sind. Angestrebt wird eine Zerlegung der Gesamtmenge so dergestalt, dass homogene Gruppen entstehen. Die Beobachtungsvektoren innerhalb dieser homogenen Gruppen weisen untereinander eine gewisse Ähnlichkeit auf, wohingegen Beobachtungsvektoren unterschiedlicher Gruppen einen wesentlichen Unterschied zueinander aufweisen.

Diese Herangehensweise der Clusterung einer Gesamtmenge findet unter anderem Anwendung in den Sozialwissenschaften, in der Politikwissenschaft oder in der Soziologie, bei denen z. B. bestimmte Personenschichten nach Verhaltensweisen, Einstellungen oder Kommunikationsformen gruppiert werden sollen.

Besonders vielfältigen Einsatz findet die Clusteranalyse in den Wirtschaftswissenschaften. Hierbei sind neben Konsumenten und Unternehmen auch Produkte oder andere marktrelevante (absatzrelevante) Sachverhalte Gegenstand der Analyse. Das Marketing hat großes Interesse an der Clusteranalyse. Mittels der Verfahren kann eine Marktsegmentierung, eine Marktabgrenzung und die Identifizierung von homogenen Zielgruppen vorgenommen werden, aber auch die Identifikation von Konkurrenten. Die Clusteranalyse wird somit zur Typologisierung und Bestimmung von Marktstrukturierungen herangezogen.

Vielfältiger Einsatz der Clusteranalyse ist in den Naturwissenschaften zu finden. Mögliche

zu gruppierende Objektmengen sind Bakterienstämme, geografische Regionen, Patienten einer Klinik oder Schadensfälle eines Versicherungsunternehmens.

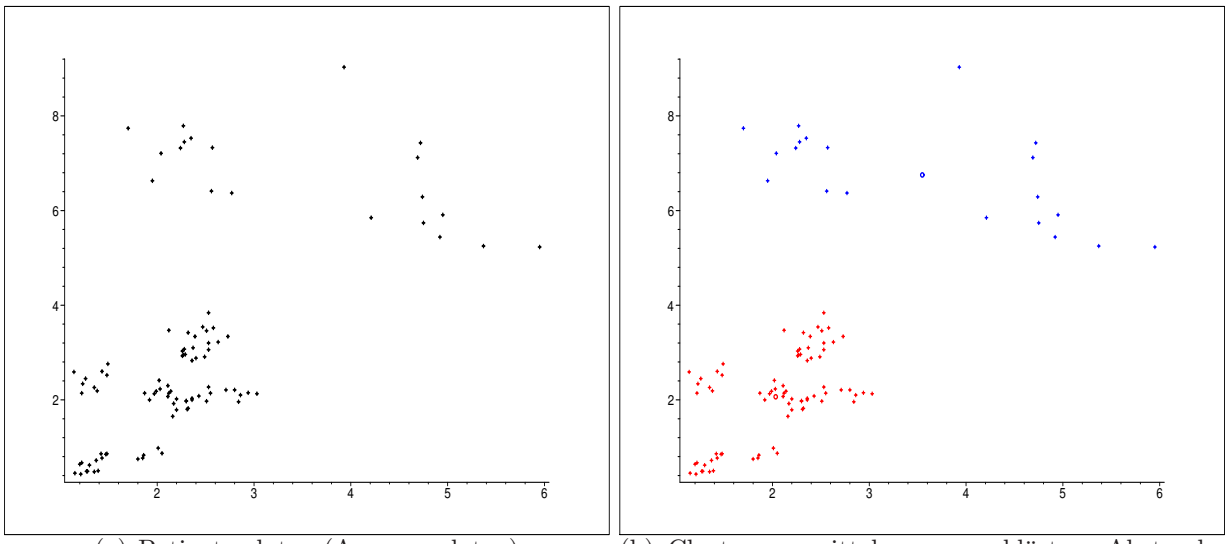
Bei der Untersuchung von mathematischen Eigenschaften geht es hauptsächlich um die Untersuchung des Verhaltens der Verfahren unter wohldefinierten unterschiedlichen Bedingungen. Die grundlegende Einflussgröße für die Wirksamkeit von Clusterverfahren sind die Abstandsmaße, mit denen die erhaltenen Cluster bewertet werden.

Die folgenden drei Fälle sollen kurz verdeutlichen, wie unterschiedlich der Einsatz der Abstandsmaße ist und wie abhängig die Wahl des Abstandsmaßes von der gegebenen Fragestellung ist.

1. Ausgehend von gegebenen Beobachtungen $x_1, ..., x_n \in \mathbb{R}^2$ lässt sich aus fachlichem Hintergrund annehmen, dass diese Beobachtungen in Gruppen zerfallen. Bei der Betrachtung von zum Beispiel Patientendaten in Bezug auf Veränderungen der Nervenzellen[1] erhofft man sich eine Zerlegung in 2 Gruppen, die mit 2 Krankheiten in Verbindung gebracht werden können. Im Falle der gezeigten Abb. 1 und der verwendeten Daten erfolgt eine Zerlegung in 2 Cluster, die mit der entsprechenden Erkrankung bzw. nicht Erkrankung in Verbindung gebracht werden können. Ausgehend von diesen Patientendaten gibt es 2 Werte $\mu_1,\ \mu_2 \in \mathbb{R}^2$, um die sich die Punkte der jeweiligen Cluster anordnen. Dem zugrunde liegend wird das entsprechende Abstandsmaß gewählt, welches die für diese Gruppen charakteristischen Mittelwerte berücksichtigt. Es wird somit ein Abstandsmaß verwendet, welches das Varianzkriterium nachbildet. Die in Abb. 1 gezeigten Grafiken sollen die Grundidee hinter diesem Modellansatz und dem Abstandsmaß verdeutlichen. In Abb. 1(b) ist eine klare Einteilung der Daten in Gruppen erkennbar. Die blauen markierten Beobachtungen können Patienten mit der Alzheimererkrankung in Verbindung gebracht werden, wohin gegen die rot markierten Daten alle anderen Patienten kennzeichnen. Die Clusterung kann unter Heranziehung weiterer Merkmalsausprägungen bzw. in Bezug auf andere Erkrankungen durchgeführt werden. Anhand des verwendeten Abstandsmaßes lassen sich die Daten in Bezug auf die verwendeten Merkmale der Patienten bzw. der Nervenzellen klar in Gruppen zerlegen. Diese Vorgehensweise lässt sich zum einen auf mehrdimensionale Beobachtungen übertragen und

[1] vgl. Läuter, Pincus; S. 20/21 oder Anhang A.4, Abb. 28 (A/N, A/O)

Abbildung 1: Zerlegung der Daten mittels Varianzkriterium



(a) Patientendaten (Ausgangsdaten)

(b) Clusterung mittels zuvor erklärtem Abstandsmaß

zum anderen auf die Zerlegung in mehr als 2 Cluster anwenden, d. h. Betrachtung von g Zentren $\mu_1, ..., \mu_g$, die charakteristisch für die g Cluster sind.

2. Der zweite Fall betrachtet Beobachtungen, die ebenfalls aus inhaltlicher Sicht eine Einteilung erzwingen, die sich jedoch von dem ersten Fall unterscheiden. Die Beobachtungen $x_1, ..., x_n \in \mathbb{R}^2$ sind zum Beispiel Beobachtungen, die zu Qualitätseigenschaften von technischen Produkten gehören. Diese sollen in Bezug auf ihre Qualitätsausprägung geclustert werden. Die gesuchten Cluster sollen also mit guter bzw. schlechter Qualität in Verbindung gebracht werden können. Daher ergibt sich der folgende Modellansatz (Modell mit parametrischer Struktur):
   Durch die Betrachtung des Qualitätsverhaltens ist es sinnvoll, die Cluster durch eine Kontur zu beschreiben, d. h. die Beobachtungen werden jeweils der ihnen entsprechenden Konturen zugeordnet. In diesem Fall ist die Kontur charakteristisch für das Cluster und nicht das entsprechende Zentrum, wie in Fall 1. Die angesprochene Kontur kann beispielsweise durch eine Kreislinie um den Mittelpunkt μ mit dem Radius r angegeben werden. Es ergibt sich dann für die Kontur folgende Darstellung:

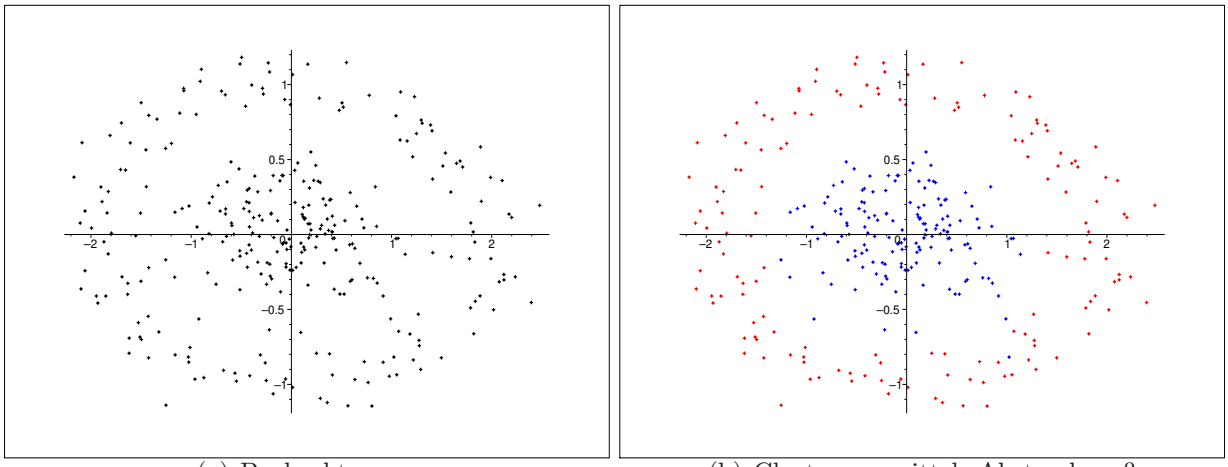
$$\mathcal{K}(\mu, r) = \{x : |x - \mu| = r\}.$$

Das entsprechende Abstandsmaß, das sich hieraus ergibt, hat dann für den Fall von 2 Clustern die Form

$$D^2(C_1, C_2) = \sum_{j=1}^{2} \sum_{\substack{i=1 \\ x_i \in C_j}}^{n_j} d(x_i, \mathcal{K}(\mu_j, r_j)),$$

wobei $d(x_i, \mathcal{K}(\mu_j, r_j))$ der minimale Abstand zwischen x_i und $\mathcal{K}(\mu_j, r_j)$ ist. Es wird somit ein Abstandsmaß verwendet, welches sich auf die beiden Cluster C_1, C_2 und auf ihre jeweiligen Mittelpunkte und entsprechende Radien bezieht. Für die zugrunde liegenden Beobachtungen dieses Sachverhaltes ergibt sich nach Anwendung des Abstandsmaßes und der Verwendung von 2 Mittelpunkten und 2 Radien, die Abb. 2(b). In diesem Fall liegen beide Mittelwerte im Nullpunkt und einer der Radien ist gleich Null.

Abbildung 2: Zerlegung der Daten mittels „Konturkriterium“



(a) Beobachtungen (b) Clusterung mittels Abstandsmaß

Für das Auffinden von 2 Cluster werden 2 Konturen benötigt. Dieses Abstandsmaß lässt sich auf mehrere Cluster übertragen, die diesem konkreten zweiten Fall entsprechen, d. h. Sachverhalte, bei denen eine gewisse Struktur der Cluster erwartet wird, die durch eine Kontur im Beobachtungsraum beschrieben wird.

3. In einigen Fällen der Clusteranalyse geht man davon aus, dass die Beobachtungsmenge in Cluster derart zerlegt werden kann, dass in den Clustern geeignete statistische Modelle gelten. Weiter wird davon ausgegangen, dass die Cluster durch Konturen, die im Beobachtungsraum existieren, beschrieben werden. In diesem Fall werden Regressionsstrukturen zugelassen. Die gegebenen Beobachtungen $x_1, ..., x_n$ haben die Gestalt:

$x_i = \binom{z_i}{y_i}$. Für z_i wird ein parametrischer Regressionsmodellansatz gewünscht, wodurch sich dann

$$z_i = f_j(y_i, \beta_j) + \varepsilon_i$$

ergibt, für $i = 1, ..., n_j$, wobei j das entsprechende Cluster bezeichnet. Die Beobachtungen liegen im Beobachtungsraum vor und scharen sich um eine Regressionskurve. Hieraus ergibt sich der Ansatz für das Abstandsmaß für 2 Cluster:

$$D^2(C_1, C_2; \beta_1, \beta_2) = \sum_{j=1}^{2} \sum_{\substack{i=1 \\ x_i \in C_j}}^{n_j} |z_i - f_j(y_i, \beta_j)|^2.$$

Die Clusterung bedeutet das Auffinden der Gruppen C_1 und C_2 und das Bestimmen der Schätzungen für β_1 und β_2. Diese Herangehensweise lässt sich ebenfalls auf mehrere Cluster übertragen, d. h. es handelt sich dann um das Auffinden der Cluster $C_1, ..., C_g$ und den entsprechenden Schätzungen. Dieser Ansatz geht davon aus, dass die Beobachtungen aus einem Cluster in einer höchstens $(d-1)$-dimensionalen Mannigfaltigkeit vorliegen. Da das Auffinden der niedrigerdimensionalen Mannigfaltigkeiten sehr aufwendig ist, kann man zur Vereinfachung das nearest neighbour Verfahren oder Single-Linkage-Verfahren nutzen.

Die aufgezeigten drei möglichen Vorgehensweisen machen deutlich, dass die Abstandsmaße einen wesentlichen Einfluss auf die Struktur der optimalen Cluster haben.

Die vorliegende Arbeit geht auf das Verfahren der Clusteranalyse ein. Es werden allgemeine Definitionen und grundlegende Erklärungen und Erläuterungen zum Ablauf der Clusteranalyse vorgestellt. Die Arbeit geht näher auf strukturentdeckende Eigenschaften und das asymptotische Verhalten des K-means Verfahrens ein.

Kapitel 2 liefert zu Beginn eine Definition der Clusteranalyse, deren Grundprinzipien, die bei jeder Clusteranalyse berücksichtigt werden müssen, und deren Einteilung in die vorhandenen statistischen Analyseverfahren. Es wird ein kurzer Überblick über den Ablauf einer Clusteranalyse gegeben, welcher nicht nur die reine rechnerische Komponente enthält, sondern auch inhaltliche Argumente berücksichtigt. Die Präzisierung des eigentlichen Untersuchungsziels, die Aufbereitung der gegebenen Daten, die Auswahl des zu verwendenden

Clusteranalysealgorithmus und auch die nach der technischen Durchführung erfolgende Interpretation der Ergebnisse sind unter anderem wesentliche Bestandteile einer Clusteranalyse.

Für die Durchführung der Clusteranalyse werden die ausgewählten Daten in eine Ähnlichkeits- bzw. Distanzmatrix aufgenommen. Für deren Ermittlung gibt es verschiedene Distanzfunktionen, die ebenfalls beispielhaft kurz in Kapitel 2 vorstellt werden. Nach der Aufbereitung der Daten lassen sich verschiedene Clusteranalyseverfahren wählen. Die Arbeit gibt einen kurzen Überblick über die grundlegenden Eigenschaften und Vorgehensweisen der hierarchischen und partitionierenden (nichthierarchischen) Clusteranalyseverfahren unter Berücksichtigung der gewählten Clustermethoden.

Kapitel 3 liefert Betrachtungen zu den Clustermethoden aus Fall 1 und 3, die zuvor beschrieben wurden. Es werden Verfahren erläutert, mit denen die Clustermethoden realisiert werden und Beispiele gegeben, die die Grundgedanken der Methoden verdeutlichen.

In Kapitel 4 geht es um statistische Eigenschaften des K-means Verfahrens. Für dieses, welches die Clustermethode aus Punkt 1 verwendet, werden asymptotische Eigenschaften aufgezeigt.

Für das asymptotische Verhalten des K-means Verfahrens liegen nur wenige Veröffentlichungen in der Literatur vor. Aufgrund dieser Tatsache stützt sich diese Arbeit stark auf die Veröffentlichungen von Pollard[2]. Seine Arbeiten nutzen Grundlagen aus der Wahrscheinlichkeitstheorie unter anderem für den Beweis des gleichmäßigen SLLN[3] oder den Beweis der Donsker-Klassen. Im Wesentlichen ist es in dieser Arbeit gelungen Beweisschritte aus diesen Veröffentlichungen nachzuvollziehen. Bei einigen Sachverhalten muss jedoch an den entsprechenden Stellen auf die Originalliteratur verwiesen werden.

[2] vgl. Pollard 1981, ders. 1982, ders. 1984, ders. 1990

[3] Strong Law Of Large Numbers

2 Clusteranalyse

2.1 Definition

Die Clusteranalyse befasst sich mit der Analyse einer heterogenen Gesamtheit von Objekten mit dem Ziel, homogene Teilmengen von Objekten aus dieser Gesamtheit zu identifizieren. Das primäre Ziel besteht also darin, eine Menge von Objekten in homogenen Gruppen zusammenzufassen. Die Unterschiede zwischen den Objekten innerhalb der Cluster sollen minimal und zwischen den Clustern maximal sein.

Allgemein gehört die Clusteranalyse zu den Verfahren der beschreibenden Statistik. Es existiert kein absolutes Kriterium zur Bestimmung der optimalen Anzahl von Clustern. Die Vielzahl der unterschiedlichen Distanzmaße, der Clusteralgorithmen, sowie die Notwendigkeit der Abstimmung der Clusterung auf den jeweiligen Untersuchungszweck verhindern die Bestimmung eines allgemeingültigen Kriteriums.

Die Clusteranalyse kommt insbesondere dann zum Einsatz, wenn Mengen von Objekten nicht mehr überblickt werden können, d. h. wenn entweder zu viele Objekte vorliegen, oder diese durch mehrdimensionale Merkmalsbeschreibungen dargestellt werden. Die Clusteranalyse hat somit die Funktion der Datenstrukturierung (explorativen Verfahren), d. h. es erfolgt eine systematische Informationsverdichtung, um aus einer Fülle von verschiedenen Einzelobjekten wesentliche Merkmale der Struktur der vorgegebenen Menge (Objektmenge) erkennen zu können. „Man sucht eine in den Daten möglicherweise latent verborge-

ne Gruppenstruktur.“[4] Es soll ein erster Überblick über die vorliegenden Daten, deren Auffälligkeiten und Regelmäßigkeiten gegeben werden.

Es handelt sich bei den Clusteranalyseverfahren, wie bereits erwähnt, um rein deskriptive Verfahren, welches keinerlei Aussagen über die Beziehungen der erstellten Lösungen zur Gesamtheit möglich machen, da keine Verteilungseigenschaften der Objekte genutzt werden.

Die Clusteranalyse unterliegt verschiedenen Grundprinzipien:

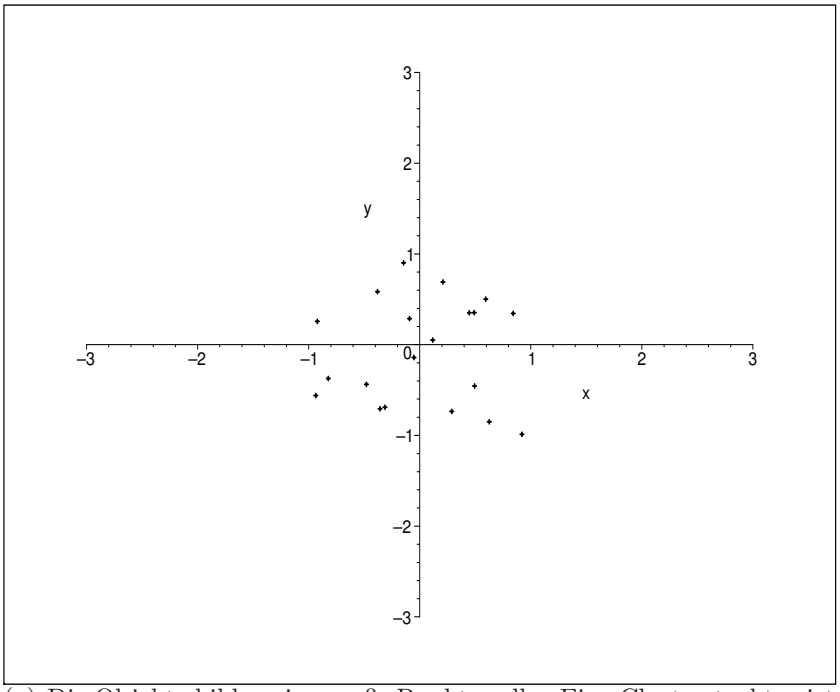
- Homogenität innerhalb der Gruppen und Heterogenität zwischen den Gruppen
- die Anzahl der Gruppen soll möglichst klein sein
- Monotonie

Jede Clusterbildung unterliegt der Homogenität innerhalb der Cluster und Heterogenität zwischen den Clustern. Liegt Homogenität innerhalb der Gruppe vor, dann sind die Objekte, die dieser Gruppe angehören, untereinander ähnlich. Liegt Heterogenität zwischen den Clustern vor, dann sollen sich diejenigen Objekte, die unterschiedlichen homogenen Gruppen angehören, in Bezug auf ihre Merkmalsausprägungen von den anderen unterscheiden.

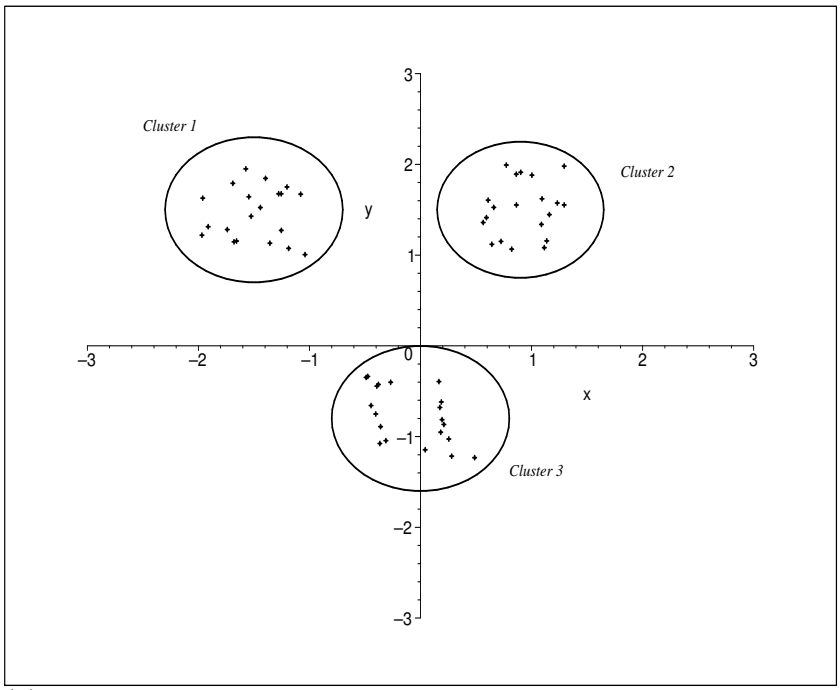
Abb. 3 gibt allgemein Beispiele für die Trennung zwischen möglichen Clustern in Bezug auf das Homogenitäts- und Heterogenitätsmaß.

[4] vgl. Deichsel, Trampisch; S. 26

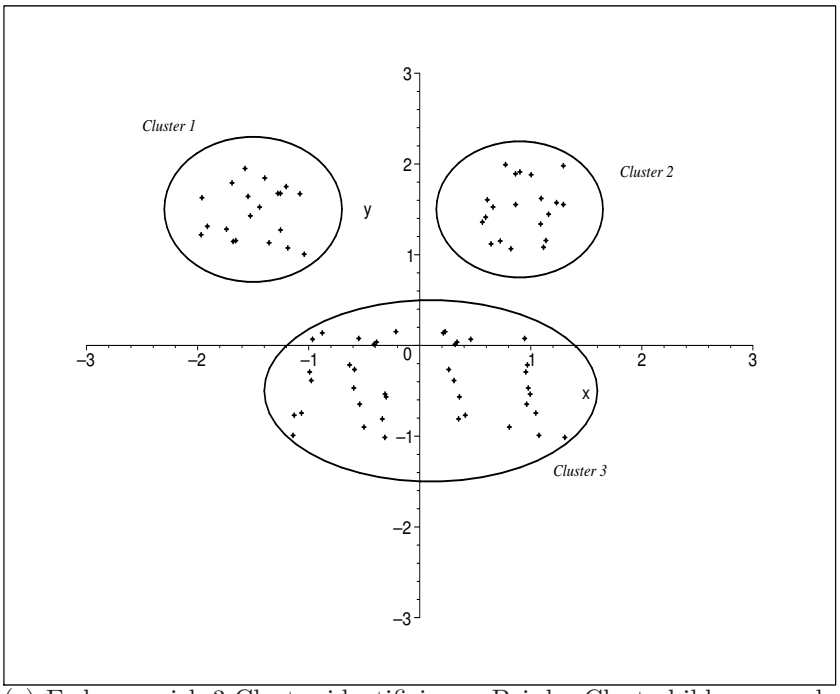
Abbildung 3: (a)-(d) Clusterbildung bzgl. Homogenitäts- bzw. Heterogenitätsmaßes[5]



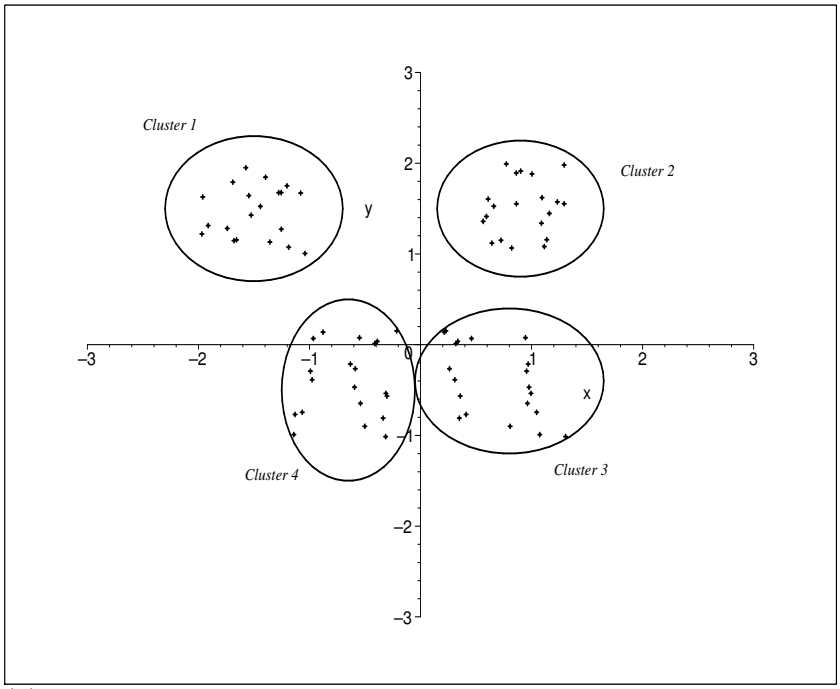
(a) Die Objekte bilden eine große Punktewolke. Eine Clusterstruktur ist nicht erkennbar.



(b) Es sind drei Cluster erkennbar. Grundprinzip der Homogenität und Heterogenität wurde gewahrt.



(c) Es lassen sich 3 Cluster identifizieren. Bei der Clusterbildung wurde der Heterogenität zwischen den Cluster ein größeres Gewicht beigemessen und daher das Cluster 3 nicht getrennt.



(d) Hier wurde der Homogenität innerhalb der Cluster bei der Clusterbildung ein größeres Gewicht beigemessen. Das langgestreckte Cluster aus Abb. (c) wurde getrennt.

Eng verbunden mit dem eben erläuterten Prinzip der Homogenität und Heterogenität ist das Grundprinzip der Clusteranzahl. Diese sollte aus inhaltlichen Gründen möglichst klein sein. Je weniger interpretierbare Cluster vorhanden sind, desto besser lässt sich das erhaltene Ergebnis praktisch umsetzen. Steigt jedoch die Anzahl der Cluster, kommt es zu einer Vereinzelung der Objekte. Dies ist nicht im Sinne des Untersuchungsziels und der Clusteranalyse.

Für die sinnvolle Interpretation über die Gruppenzugehörigkeit sollte ebenfalls das Monotoniekriterium erfüllt sein. Die Monotonie ist gegeben, wenn die Heterogenität in den Clustern mit abnehmender (zunehmender) Zahl der Gruppen steigt (sinkt).

2.2 Anwendung der Clusteranalyse

Die Anwendung der Clusteranalyse lässt sich nicht allein auf den Algorithmus reduzieren. Es müssen auch inhaltliche Sachverhalte berücksichtigt werden. Somit lässt sich die Durchführng einer Clusteranalyse vereinfacht in acht Phasen darstellen.[6]

(i) Zu Beginn der Clusteranalyse erfolgt eine ausführliche Präzisierung der Untersuchungsfragestellung, d. h. eine Präzisierung des inhaltlichen Problems bzw. des Verwendungszweckes der Clusteranalyse, um spätere Ergebnisse zielgerichtet, in Bezug auf den Stellenwert und die Funktion der Gruppierung, interpretieren zu können. Es müssen dementsprechend inhaltliche Kriterien festgelegt werden, um im weiteren Verlauf feststellen zu können, welche Ähnlichkeitsfunktion zu wählen ist und um beurteilen zu können, ob die Clusterung „brauchbar“ oder „unbrauchbar“ in Bezug auf einen bestimmten Zweck ist.

(ii) Als zweiten Schritt werden die Elemente und Variablen ausgewählt, die für den untersuchten Bereich hinreichend repräsentativ sein sollen. Diese Variablen müssen sich eindeutig auf das Untersuchungsziel beziehen.

(iii) Anschließend erfolgt die Aufbereitung der ausgewählten Daten. Für jedes der gegebenen n Objekte $\Omega_1, ..., \Omega_n$ liegen die Messwerte von p Merkmalsvariablen vor, wobei

[5] vgl. Bacher, S. 3

[6] vgl. Steinhausen, Langer; S. 20ff

die Clusteranalyse Ähnlichkeiten zwischen diesen Eigenschaften erkennen soll. Die Merkmalsvariablen werden in einer $n \times p$ Rohdatenmatrix $X^R = (x_{ij}^R)$ angeordnet, hierbei stellt x_{ij}^R den Messwert der j-ten Variable für das i-te Element dar.
Durch Datenvorbehandlungen können Daten standardisiert und normiert oder fehlende Werte berücksichtigt werden. Man erhält somit eine zu analysierende Datenmatrix $X = (x_{ij})$.

(iv) Um mit den ausgewählten Variablen in den eigentlichen Clusteralgorithmus starten zu können, werden die angemessenen Ähnlichkeitsfunktionen festgelegt bzw. definiert (Proximitätsmaße). Die Ähnlichkeitsfunktionen - auch Distanzfunktionen genannt - sind entsprechend den verschiedenen Variablentypen quantitativ oder qualitativ zu wählen. Man erhält dann eine Ähnlichkeitsmatrix $S = (s_{ij})$ bzw. eine Distanzmatrix (Unähnlichkeitsmatrix) $D = (d_{ij})$.

(v) Im nächsten Schritt erfolgt die Wahl der geeigneten Clusteranalysemethode im Hinblick auf die zu erwartenen Resultate. Pauschal lassen sich somit keine Aussagen zum „besten“ Clusteranalyseverfahren treffen. Die Auswahl hängt sehr stark von der Aufgabenstellung und der Datenstruktur ab. Je nach implementiertem Algorithmus können für dieselben Objekte ganz verschiedene und unterschiedliche Gruppierungen entstehen. Die Einteilung der gegebenen n Objekte in Gruppen wird in der Angabe einer Partition $\mathscr{C} = \{C_1, C_2, ..., C_g\}$ dargestellt, wobei g, die Anzahl der Cluster und jedes C_i, für $i = 1, ..., g$, wiederum eine Menge von Elementen ist. Für jedes Cluster der Partition $\mathscr{C}$ gilt

$$C_i \cap C_j = \emptyset, \ i \neq j.$$

Bei der Wahl der Verfahren und der daraus resultierenden (damit zusammenhängenden) Startpartition lässt sich zwischen hierarchischen und nichthierarchischen Algorithmen unterscheiden.

(vi) Nach Festlegung des Clusteranalysealgorithmus erfolgt die technische Durchführung, die aufgrund der meist hohen Variablenzahl durch den Computer stattfindet. Am Ende der Rechenzeit erhält man die Partition $\mathscr{C}$, die dann als Clusterlösung bezeichnet wird.

(vii) Die nun folgende formale Analyse der Ergebnisse richtet sich auf die Beurteilung der Homogenität der gebildeten Cluster, der Differenz der Clustermittelpunkte, des

Einflusses bestimmter Variablen oder der Bedeutung der Startnäherung. Ebenfalls wird die Optimalität der gefundenen Lösung aus statistischer Sicht beurteilt.

(viii) Der letzte Schritt der Clusteranalyse beinhaltet die Interpretation der gefundenen Ergebnisse. Es erfolgt die genaue Analyse und Interpretation der Gruppierung in Bezug auf das anfangs festgelegte Untersuchungsziel bzw. die Untersuchungsfragestellung. Die Ergebnisse werden mit dem inhaltlichen Thema des Untersuchungszwecks verknüpft, wodurch weitere inhaltliche Aufschlüsse über die erhaltenen Cluster gewonnen werden können.

Wie schon zu Beginn erwähnt ist die Clusteranalyse ein deskriptives Verfahren und bildet, um eine Beziehungsaussage der erhaltenen Ergebnisse zur Gesamtheit zu ermöglichen, die Grundlage für weiterführende multivariate Verfahren.[7]

2.3 Ähnlichkeits- und Distanzfunktionen

Die Unähnlichkeit bzw. Ähnlichkeit von zwei Elementen i und j werden mittels einer Distanzfunktion berechnet. Rein formal handelt es sich um eine reellwertige Funktion, wobei $d_{ij} = d(x_i, x_j)$ einer reellen (nicht-negativen) Zahl entspricht. $x_i = (x_{i1}, ..., x_{ip})^T$ und $x_j = (x_{j1}, ..., x_{jp})^T$ sind die Beobachtungsvektoren (Merkmalsvektoren) der jeweiligen Objekte $\Omega_i, \Omega_j \in \Omega$ aus der i-ten und j-ten Zeile der Datenmatrix. Die Funktionswerte liegen in dem Intervall $d_0 \leq d \leq d_1$, d. h. die maximale Unähnlichkeit (Distanz) zwischen den Elementen bedeutet $d_{ij} = d_1$ und die minimale Unähnlichkeit $d_{ij} = d_0$.

Für die Distanzfunktion d gelten folgende Axiome:

$$d_{ij} \geq 0 \ (= d_0) \tag{2.1}$$

$$d_{ij} = d_{ji} \tag{2.2}$$

$$d_{ii} = 0 \ (= d_0) \tag{2.3}$$

Von einer metrischen Distanzfunktion wird gesprochen, wenn die beiden folgenden Bedingungen

$$d_{ij} = 0 \rightarrow x_i = x_j \tag{2.4}$$

$$d_{ik} \leq d_{ij} + d_{jk} \text{ (Dreiecksungleichung)} \tag{2.5}$$

[7] Beispiele hierfür sind unteranderem die Varianzanalyse und die Diskriminanzanalyse

erfüllt sind. Oft ist $d_0 = 0$ und $d_1 = 1$, wobei, wie schon zuvor erwähnt, d_0 die minimale und d_1 die maximale Distanz bedeutet.

Nach Berechnung der Distanzen/Ähnlichkeiten erhält man eine Ähnlichkeits- bzw. Unähnlichkeitsmatrix $S = (s_{ij})$ oder $D = (d_{ij})$. Bei D und S handelt es sich um $(n \times n)$-Matrizen. Aufgrund der Axiome (2.2) und (2.3) sind es ebenfalls symmetrische Diagonalmatrizen.

Bei den Distanzfunktionen und deren Anwendung muss man eine Unterscheidung in Bezug auf die gegebenen Variablentypen berücksichtigen. Man unterscheidet hierbei zwischen qualitativen und quantitativen Variablen. Diese Arbeit berücksichtigt quantitative Variablen.

Bei quantitativen Merkmale werden häufig die L_r-Distanzen (Minkowski-Metriken, L_r-Normen), die euklidische Distanz oder auch die Mahalanobis-Distanz verwendet.

Die Minkowski-Metrik lässt sich mittels

$$d_{ij} := \|x_i - x_j\|_r \tag{2.6}$$

$$:= \Big[\sum_{l=1}^{p} |x_{il} - x_{jl}|^r\Big]^{\frac{1}{r}} \tag{2.7}$$

berechnen. Hierbei handelt es sich bei d_{ij} um die Distanz der Elemente i und j. Die positive ganzzahlige Konstante r wird auch Minkowski-Konstante genannt. Die Wahl von r hängt von der Wahl der Gewichtung ab. Bei der Festlegung großer r soll mehr Gewicht auf große Distanzen gelegt werden, wobei bei r klein eine relative Ausgewogenheit zwischen den Distanzen der Elemente besteht.

Für $r = 1$ handelt es sich um die 'City-Block'-Metrik, die bei der Clusterung von Standorten eine große Bedeutung hat.

Die euklidische Distanz ergibt sich für $r = 2$:

$$d_{ij} = \|x_i - x_j\|_2 = \sqrt{\sum_{l=1}^{p} |x_{il} - x_{jl}|^2}. \tag{2.8}$$

Häufig wird aus rechnerischer Einfachheit die quadrierte euklidische Distanz verwendet.

Bei allen Distanzen müssen sämtliche Daten dieselben Größenordnung aufweisen. Um eine einheitliche Maßeinheit zu erhalten werden die Daten standardisiert.

Die Mahalonobis-Distanz ist im Vergleich zur euklidischen Distanz skaleninvariant und ein Spezialfall der quadratischen Distanzfunktionen

$$d_B(x_l, x_j) := \left[(x_l - x_j)^T B(x_l - x_j)\right]^{\frac{1}{2}}.$$

Die Mahalanobis-Distanz hat dann die Form:

$$\begin{aligned} d_{K^{-1}}(x_l, x_j) &:= \|x_l - x_j\|_{K^{-1}} \\ &:= \left[(x_l - x_j)^T K^{-1}(x_l - x_j)\right]^{\frac{1}{2}}, \end{aligned} \tag{2.9}$$

die quadrierte Mahalanobis-Distanz lautet dementsprechend

$$\begin{aligned} d_{K^{-1}}(x_l, x_j) &:= \|x_l - x_j\|^2_{K^{-1}} \\ &:= (x_l - x_j)^T K^{-1}(x_l - x_j), \end{aligned} \tag{2.10}$$

wobei

$$K := \frac{1}{n}\sum_{l=1}^{n}(x_l - \bar{x})(x_l - \bar{x})^T$$

mit

$$\bar{x} = \frac{1}{n}\sum_{l=1}^{n} x_l$$

und

$$k_{ij} = \frac{1}{n}\sum_{l=1}^{n}(x_{li} - \bar{x}_{.i})(x_{lj} - \bar{x}_{.j})$$

die Varianzmatrix ist, bei der jedoch gewisse Probleme in Bezug auf die Invertierbarkeit auftreten können.

Die Mahalonobis-Distanz findet selten eine Anwendung in der Clusteranalyse, da sie stark von den Startwerten abhängt. Ebenfalls hängt die Matrix K von den gegebenen Daten ab, daher sind die erhaltenen Lösungen relativ willkürlich.

2.4 Clusteranalyseverfahren

Die nach der Aufbereitung der Daten gewonnenen Ähnlichkeits- oder Distanzmatrizen bilden den Ausgangspunkt für die Anwendung der Clusteralgorithmen und Clusterverfahren. Mittels dieser werden die Elemente in Gruppen zusammengefasst.

Die Gruppierungsverfahren lassen sich nach Zahl der Variablen, die beim Fusionierungsprozess berücksichtigt werden, einteilen. Es handelt sich hierbei um monothetische und polythetische Verfahren. Monothetische Verfahren ziehen zur Gruppierung jeweils nur eine Variable heran, wohingegen polythetische Verfahren simultan alle relevanten Merkmale zur Gruppierung der Objekte berücksichtigen.

Eine weitere Einteilung erfolgt über die Vorgehensweise im Fusionierungsprozess. Man unterscheidet hierbei zwischen hierarchischen und nicht-hierarchischen (partitionierenden) Verfahren.

Die hierarchischen Verfahren lassen sich in agglomerativ und divisiv einteilen. Bei den partitionierenden Verfahren gibt es beispielsweise das Austauschverfahren und das Minimaldistanzverfahren. Alle Verfahren können ein gewähltes Distanzmaß nutzen. Eine der möglichen Methoden zum Auffinden der (sub-) optimalen Lösung ist die K-means Methode.

Losgelöst von den Abstandsmaßen gibt es noch weitere Ansätze, wie man zu einer Clusterung kommen kann. Das Single-Linkage-, Complete-Linkage-, Average-Linkage-Verfahren, Cebtroid-Verfahren oder Verfahren nach Ward lassen sich unabhängig von Clustermethoden auf zu clusternde Daten anwenden.[8]

2.4.1 Hierarchische Verfahren

Wie bereits erwähnt, lassen sich die hierarchischen Verfahren in agglomerative (bottom-up) und divisive (top-down) Verfahren unterscheiden. Bei beiden Varianten werden Anforderungen an die Homogenität der Klassen gestellt.

[8] Die dabei erhaltenen Ergebnisse bilden jedoch keine (sub-) optimale Lösung.

Divisive Verfahren wählen als Ausgangspunkt die gröbste Partition $\mathscr{C}_1$, bei der alle Untersuchungsobjekte aus der Objektgesamtheit in einem Cluster liegen. Diese Ausgangspartition wird sukzessiv in mehrere Cluster unterteilt. Dadurch wird eine höhere Homogenität innerhalb der Klassen erreicht. Man unterscheidet hierbei zwischen monothetischen und polythetischen Verfahren. Die Clusterbildung bei den monothetischen Verfahren stützt sich auf das Vorhandensein oder Nicht-Vorhandensein eines Divisionsmerkmals. Die Aufteilung der Cluster verläuft nur anhand eines Merkmals. Daher besteht die Gefahr, dass die gebildeten Gruppen zwar homogen sind, jedoch können sich die Elemente innerhalb eines Clusters bezüglich anderer Merkmale stark voneinander unterscheiden. Divisiv-polythetische Verfahren berücksichtigen alle Merkmale. Allerdings erfordern diese im Vergleich zu den agglomerativen Verfahren einen hohen Rechenaufwand.

Bei beiden Varianten kann nur eine Gruppenzerlegung stattfinden, wenn die Cluster mindestens zwei Elemente enthalten. Divisive Verfahren sind im Vergleich zu agglomerativen Verfahren rechenzeitaufwendiger und im Vergleich zu den partitionierenden Verfahren liefern sie weniger gute Ergebnisse.

Bei den agglomerativen Verfahren bildet die feinste Partition den Ausgangspunkt, d. h. $\mathscr{C} = (\{x_1\}, ..., \{x_n\})$, bei der jedes Objekt aus der Objektgesamtheit ein Cluster für sich bildet. Durch die sukzessive Vereinigung der Objekte wird die Homogenität innerhalb der Klassen verringert.

Man sucht das Paar x_i, x_j, $(i \neq j)$ von Elementen mit minimaler Distanz d_{ij}, d. h. mit kleinster Gruppenunähnlichkeit, aus der zuvor berechneten Distanzmatrix und vereinigt dieses Paar, falls das gewählte Abstandsmaß in diesem Fall ein Minimum hat, zur Gruppe $C = \{x_i, x_j\}$. Zuvor muss allerdings über das Distanzmaß entschieden werden, wie auch über die Vereinigungsregel, auf die etwas später eingegangen wird.

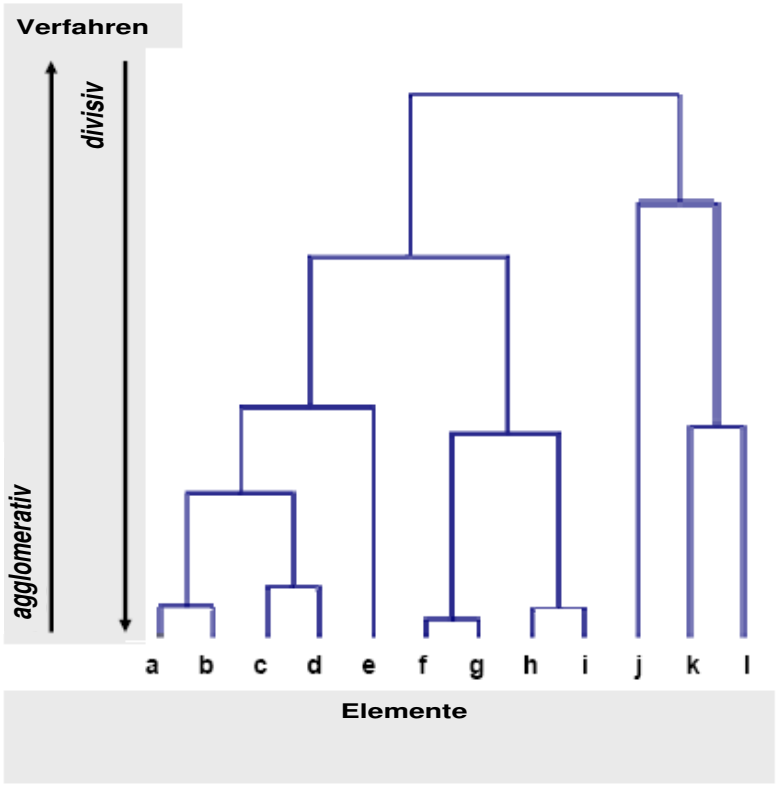
Der Algorithmus hat nun eine Partition $\mathscr{C}_k = \{C_1, C_2, ..., C_k\}$ mit k Gruppen erzeugt. Die beiden Gruppen $C_v, C_w, v \neq w$ werden zu einer neuen Gruppe vereinigt, wenn zwischen diesen beiden die minimale Distanz (d_{C_v,C_w}) in Bezug auf das Distanzmaß, d. h. die größte Ähnlichkeit besteht.

Nach diesem Schritt entsteht eine neue Partition $\mathscr{C}_{k-1}$ mit $k-1$ Clustern. Es werden nun

die Distanzen zwischen der neugebildeten Gruppe und den übrigen Gruppen berechnet, in einer reduzierten Distanzmatrix aufgenommen und wiederum die Gruppen mit der minimalen Distanz vereinigt. Dieses Schema wird solange wiederholt, bis die gröbste Partition $\mathscr{C}_1$, bei der alle Objekte in einem Cluster vereint sind, erreicht ist.

Die Ergebnisse bzw. den Verlauf der hierarchischen Clusterverfahren, sowohl divisiv als auch agglomerativ, lassen sich visuell durch ein sogenanntes Dendrogramm veranschaulichen (Vgl. Abb. 4). Es liefert eine gute grafische, interpretierbare und komplette Beschreibung der hierarchischen Clusterung.

Abbildung 4: Konstruktionsprinzip der hierarchischen Verfahren - Dendrogramm



Ein Vorteil der hierarchischen Clusterverfahren liegt in der Klassenanzahl, die zu Beginn nicht vorzugeben ist. Generell arbeiten hierarchische Verfahren sehr schnell, d. h. sie weisen einen geringeren Rechenaufwand als die nichthierarchischen Verfahren auf. Im Vergleich zu agglomerativen Verfahren sind divisive Verfahren rechentechnisch zeitaufwendiger und generell schwerer zu handhaben. Sie werden aus diesem Grund sowohl in der Literatur als auch in der Praxis nicht oft angewendet.

Bei den hierarchischen Verfahren können mögliche Fehler innerhalb der einzelnen Schritte auftreten, die nicht mehr korrigierbar sind. So kann bei agglomerativen Verfahren beispielsweise bei in einem früheren Schritt bereits fusionierte Objekte, d. h. ihre Fehlclusterung und damit ihre Clusterzugehörigkeit nicht mehr geändern werden. Gleiches gilt bei den divisiven Verfahren. Cluster, die einmal durch den Algorithmus getrennt wurden, können nicht mehr zusammengefügt werden. Nach Beendigung des Verfahrens sind also mögliche Fehler, die sich wiederum negativ auf das Endergebnis auswirken, nicht mehr korrigierbar.

Der starre Ablauf der hierarchischen Methode kann sich zum Vorteil, aber auch zum Nachteil entwickeln. Ein Vorteil dieser Verfahren ist vor allem der geringe Rechenaufwand, wohingegen die Unfähigkeit Fehler rückgängig zu machen einen großen Nachteil darstellt. Während partitionierende Verfahren versuchen die beste Clusterung auszuwählen, ermitteln die hierarchischen Verfahren nicht nur eine Clusterunterteilung, sondern auch noch eine ganze Hierarchie ineinander geschachtelter Cluster.

Es lassen sich unabhängig von Abstandsmaßen Clusterverfahren finden. Zu diesen Verfahren gehören zum Beispiel das Single-Linkage-Verfahren, das Complete-Linkage-Verfahren und das Centroid-Verfahren, welche zu den agglomerativ hierarchischen Clusterverfahren zählen. Sie unterscheiden sich ausschließlich in dem Fusionierungsmaß, welches für die Ermittlung der neuen Clusterung zugrunde gelegt wird.

Das *Single-Linkage-Verfahren*, welches auch nearest neighbour oder minimum distance method oder connectedness method genannt wird, vereinigt zunächst diejenigen Elemente miteinander, die die kleinste Distanz zueinander aufweisen. Im weiteren Verlauf des Verfahrens bildet die Distanz zwischen zwei Clustern C_f und C_g die kleinste Distanz zwischen

den Elementen aus C_f und den Elementen aus C_g:

$$d(C_f, C_g) = \min_{\substack{x_m \in C_f \\ x_l \in C_g}} \|x_m - x_l\|_2.$$

Im nächsten Vereinigungsschritt s werden nun diejenigen Cluster aus dem $(s-1)$-Schritt miteinander vereinigt für die gilt:

$$d(C_r, C_t) = \min_{\forall r,t} \min_{\substack{x_v \in C_r \\ x_z \in C_t}} d(x_v, x_z),$$

wobei $r \neq t$.[9]

Das Single-Linkage-Verfahren eignet sich gut um isolierte Punkte (Ausreißer) zu erkennen, da diese im Sinne des vorgegebenen Fusionierungsmaßes am weitesten entfernt liegen und daher, im Dendrogramm sichtbar, erst sehr spät mit anderen Elementen oder Clustern vereinigt werden.

Durch dieses Verfahren können beliebige Formen der Clusterung der Beobachtungen gefunden werden. Bei großen Klassen mit sehr weit enfernten Elementen kann die Gruppierung ellipsenförmig, gekrümmt, verzweigt oder kreisförmig aussehen.

Ein großer Nachteil des Single Linkage-Verfahrens ist die Neigung zur Kettenbildung. Durch „dicht“ aneinander liegende Elemente (sog. „Brücken“), die zu unterschiedlichen Gruppen gehören, kann das Verfahren keine genaue Trennung vornehmen. Es werden dadurch viele große Cluster gebildet, wodurch eben diese „schlecht“ getrennten Cluster nicht aufgedeckt werden, obwohl sie sich ansonsten deutlich voneinander unterscheiden.

Bei dem *Complete-Linkage-Verfahren* (furthest neighbour, maximum distance method, diameter method) ist die Distanz zwischen zwei Gruppen C_f und C_g gleich der größten Distanz zwischen den Elementen aus diesen Gruppen:

$$d(C_f, C_g) \quad = \quad \max_{\substack{x_m \in C_f \\ x_l \in C_g}} \|x_m - x_l\|_2$$

Analog zum Single-Linkage-Verfahren gilt dann für die nächste Partitionsstufe s:

$$d(C_r, C_t) = \min_{\forall r,t} \max_{\substack{x_v \in C_r \\ x_z \in C_t}} d(x_v, x_z),$$

[9] vgl. Kaufmann, Pape; S. 461

wobei $r \neq t$ und $C_r,\ C_t \in \mathscr{C}^{s-1}$.[10]

Dieses Verfahren erzeugt homogene, jedoch weniger gut voneinander separierte Cluster, d. h. man erhält tendenziell Gruppen mit ähnlicher Größe und Gestalt. Im Gegensatz zum Single-Linkage-Verfahren kommt es hierbei zur Bildung von vielen kleinen Gruppen. Nachteilig ist, dass isolierte Punkte durch das Complete-Linkage unentdeckt bleiben.

Diese beiden soeben vorgestellten Verfahren erzeugen ähnliche Ergebnisse, falls alle Beobachtungen innerhalb der einzelnen Cluster im Vergleich zu den Beobachtungen in den anderen Clustern verhältnismäßig dicht beieinander liegen und gut von den anderen Clustern abgetrennt sind. Man spricht dann von der Kompaktheit jedes Clusters.

Das *Centroid-Verfahren* beruht auf den Abständen zwischen den Mittelwerten der Cluster. Es erfolgt eine Vereinigung von Cluster, falls gilt

$$d(C_r, C_t) = \min_{\forall r,t} \|\bar{x}_r - \bar{x}_t\|_2$$

mit

$$\bar{x}_i = \frac{1}{n_i} \sum_{\substack{j=1 \\ x_j \in C_i}}^{n_i} x_j,\ i = 1, ..., k.$$

2.4.2 Partitionierende Verfahren

Die partitionierenden Clusterverfahren bilden die zweite Gruppe der Clusteranalyseverfahren. Bei ihnen wird eine Anzahl K von Startclustern vorgegeben. Es gilt $K \leq n$. Die Ausgangsbasis der nichthierarchischen Verfahren bildet somit eine vorgegebene Gruppierung $\mathscr{C} = \{C_1, C_2, ..., C_K\}$ der untersuchten Elemente. Hierfür wird häufig die Lösung eines hierarchischen Clusterverfahrens ohne Distanzmaß verwendet, die dann mittels eines partitionierenden Verfahrens verbessert wird.

Die Algorithmen versuchen iterativ die jeweilige Gruppierung durch Verschieben einzelner Elemente von einem Cluster in ein anderes Cluster zu verbessern. Die vorgegebene Clusterzahl ändert sich dabei nicht. Das Verfahren endet, wenn sich eine Gruppierung durch

[10] vgl. Kaufmann, Pape; S. 463

weiteres Verschieben von Beobachtungen nicht mehr verbessern lässt. Im Gegensatz zu den hierarchischen Verfahren ermöglichen die partitionierenden Verfahren, dass während des Optimierungsprozesses die Elemente zwischen den Clustern getauscht werden können. Die Zuordnung eines Elementes zu einem Cluster kann somit beliebig oft verändert werden. Es wird jedoch gefordert, dass jedes Element zu höchstens einer Gruppe gehört.

Die Vorteile dieser Verfahren liegen klar in der schnellen Angabe von Ergebnissen auch bei großen Datenmengen und in der Flexibilität der Umgruppierung von Elementen. Wie schon erwähnt lässt sich die Clusterzugehörigkeit von Beobachtungen jederzeit wieder ändern. Nachteilig ist jedoch die Voraussetzung der Vorabdefinition einer bestimmten Clusterzahl.

Die partitionierenden Verfahren sind zum Beispiel das k-nearest neighbour Verfahren, das Minimaldistanz- und das K-means Verfahren.

Dem k-nearest neighbour Verfahren lassen sich in der Literatur zwei Varianten zuordnen. Die erste Variante beruht auf der Ermittlung der nächsten Nachbarn aus der gesamten Objektmenge. Hierzu werden mittels der Abstandsberechnung (Euklidische Distanz) die „Entfernungen“ zwischen den zuzuordnenden Elementen $x_j,\ j = 1, ..., n$, wobei $x_j \in C_i,\ i = 1, ..., m$ und aller anderen Elemente gesucht. Anschließend erfolgt die Ermittlung der jeweiligen k Nachbarn, in Abhängigkeit vom gewählten Abstandsmaß, unabhängig von deren jeweiligen Clusterzugehörigkeit. D. h. die nächsten Nachbarn müssen nicht zwangsläufig aus demselben Cluster des neu zuzuordnenden Elementes stammen. Liegen nun alle k Elemente in einem Cluster, so wird x_j diesem Cluster zugeordnet. Stammen aber nicht alle k Elemente aus ein und demselben Cluster, so kann man dann nach dem Mehrheitsprinzip entscheiden. x_j wird dabei demjenigen Cluster zugeordnet, aus dem die Mehrzahl der nächsten Nachbarn stammen.

Die Ermittlung der nächsten Nachbarn nach der zuvor erwähnten zweiten Variante beruht auf der Abstandsberechnung zu den Elementen innerhalb des eigenen Clusters. Die Berechnung erfolgt für alle Beobachtungen und alle Cluster. Findet nach der Neuzuordnung eines Elementes eine Verbesserung des Abstandsmaßes statt, dann entsteht eine neue Clusterung.

Das Minimaldistanzverfahren setzt sich aus vier Schrittfolgen zusammen. Zu Beginn erfolgt die Vorgabe der Anfangspartition und die Berechnung der Gruppenschwerpunkte (Centro-

ide) der einzelnen Partitionen. Nun wird jedes Element in das Cluster verschoben, die mit dem im Sinne der euklidischen Distanz am nächsten liegenden Schwerpunkt wiederum unter der Berücksichtigung des minimalen Abstandsmaßwertes. Erst nach dem Wechsel aller Objekte werden die Gruppenschwerpunkte neu berechnet. Der Algorithmus wird beendet, wenn kein Element in einem Durchgang die Gruppe gewechselt hat.

Beim K-means Verfahren, dem Austauschverfahren in Kombination mit der K-means Methode, werden nach der Vorgabe der Anfangspartition $\mathscr{C} = \{C_1, C_2, ..., C_K\}$ die Gruppenschwerpunkte (Centroide) der einzelnen Cluster berechnet. In jedem Iterationsschritt wird ein Element in die Gruppe mit dem ihm im Sinne der euklidischen Distanz am nächsten liegenden Mittelwert bzgl. des Abstandsmaßes verschoben. Nach jedem Clusterwechsel werden, anders als beim Minimaldistanzverfahren, die Gruppenschwerpunkte neu berechnet. Mit diesem Ablaufschritt wird solange fortgefahren, bis l-mal hintereinander kein Element seine Gruppe gewechselt hat. Die Clusterzentren $\bar{x}_i$, $i = 1, ..., K$ werden für K Cluster so berechnet bzw. die Clusterneuzuordnung findet so statt, dass die Streuungsquadratsumme in den Clustern ein Minimum ist.

Der K-means Algorithmus liefert für unterschiedliche Startpartitionen möglicherweise unterschiedliche Ergebnisse. Auch kann es passieren, dass ein Cluster in einem Schritt leer bleibt und somit mangels Berechenbarkeit eines Clusterzenrums nicht mehr gefüllt werden kann. Um diese Probleme zu umgehen, startet man den K-means Algorithmus einfach neu und erhält beim nächsten Durchlauf durch andere zufällige Clusterzentren bzw. einer anderen Clusterausgangspartition ein anderes Ergebnis. Trotz dieser möglicherweise auftretenden Unzulänglichkeiten liefert das K-means Verfahren fast immer gute Ergebnisse.

Leider verfehlen Verfahren, die kein Abstandsmaß zugrunde legen, oft eine (sub-)optimale Lösung. Einerseits führen sie in jedem Schritt diejenige Aktion durch, welche die gegenwärtige Teillösung optimiert, aber nicht unbedingt zu einer optimalen Gesamtlösung führen muss.

3 Clustermethoden

Die Clusteranalyse ist neben dem allgemeinen Auffinden von homogenen Objektgruppen ein Mittel zur Modellbildung. Innerhalb einer gegebenen Objektmenge lassen sich mittels Abstandsmaßen gewisse Strukturen finden. Diese Abstandsmaße beschreiben Gütekriterien, welche die optimale Endclusterung in Bezug auf das verwendete Verfahren anstreben. Die Qualität der gefundenen Partition wird daher anhand der Gütekriterien (Optimalitätskriterien) ermittelt. Allgemein lässt sich feststellen, dass für die Bestimmung der optimalen Partition zwei Tatsachen zu beachten sind:[11]

- Wahl des Gütekriteriums (Gütefunktion)
- rechnerische Ermittlung einer (sub-)optimalen Partition.

Bei der richtigen Wahl des Abstandsmaßes muss vor allem das gesamte Untersuchungsziel und die inhaltliche Komponente der Untersuchung berücksichtigt werden. Wie in der Einleitung dargestellt, weisen die unterschiedlichen Abstandsmaße unterschiedliches Klassifikationsverhalten auf.

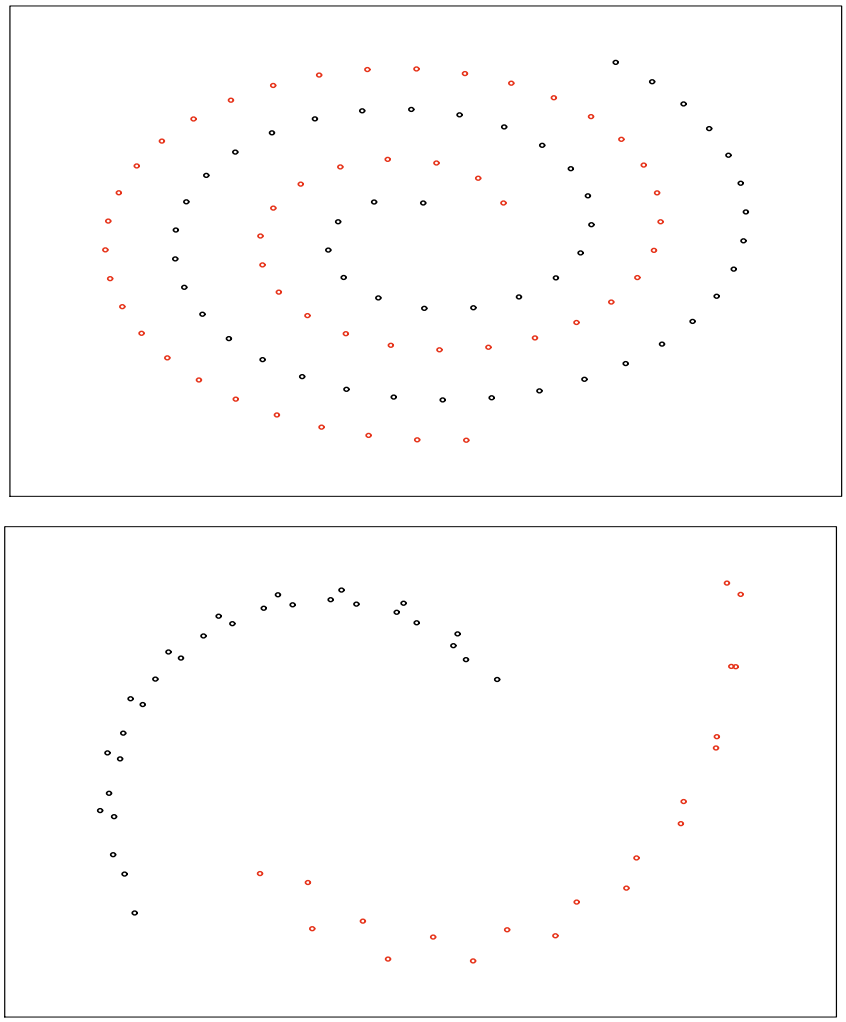
Es sollte die Frage berücksichtigt werden: „Ermöglicht mein Abstandsmaß das Erreichen des gewünschten Untersuchungsziels?“. Speziell in diesem Fall: „Führt das gewählte Abstandsmaß zur gewünschten Endstruktur bedingt durch die inhaltliche Betrachtungsweise?“

[11] vgl. Kaufmann, Pape; S. 470

Für die rechnerische Ermittlung der gewünschten Endpartition und der damit gewünschten Endstruktur lassen sich die Clusteranalyseverfahren verwenden.

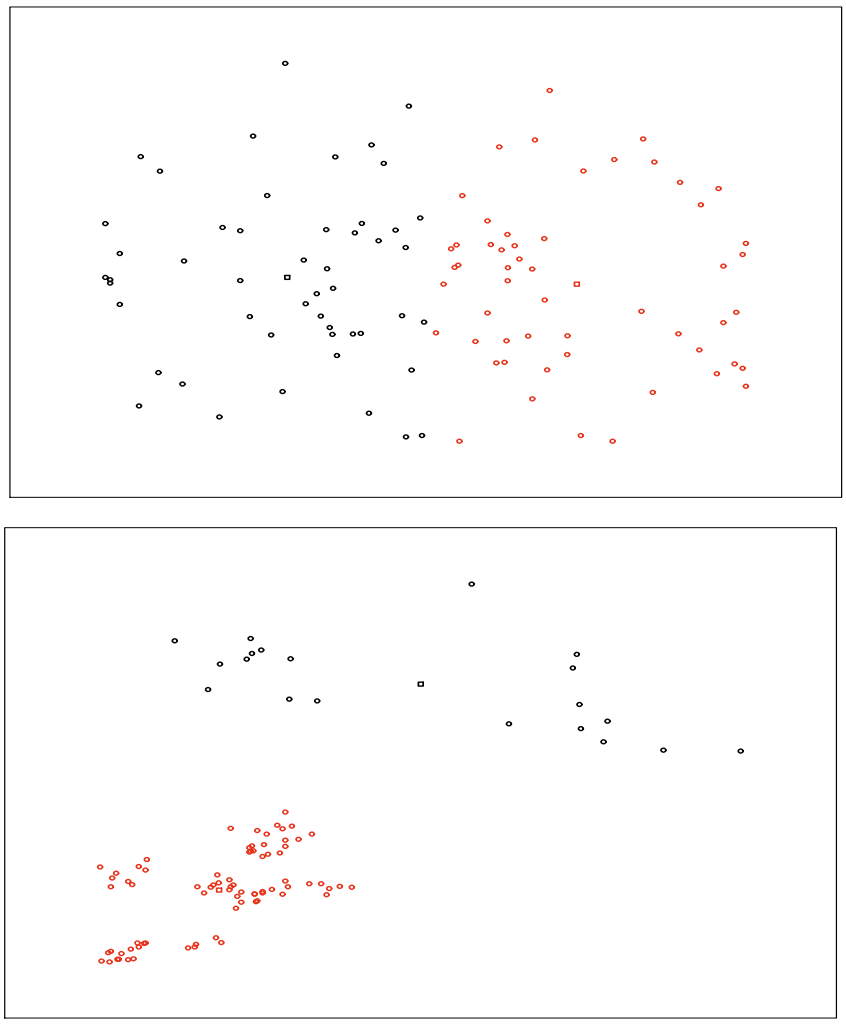
Dieses Kapitel geht auf zwei Clustermethoden ein, die mittels Anwendung der Clusteranalyseverfahren die optimalen Lösungen für diese Verfahren ermitteln (welche lokale Optima sind). Zum einen wird der Abstand zwischen Mannigfaltigkeiten und dessen rechentechnische Realisierung näher betrachtet. Es werden Beispiele gezeigt, die verdeutlichen, dass dieses Abstandsmaß spiralförmige, „geschwungene“ Strukturen findet. D. h. vereinfacht für den 2-dimensionalen Fall werden geschwungene Punktewolken als Cluster erkannt. (Vgl. Abb. 5)

Abbildung 5: mögliche Endcluster beruhend auf dem Abstand zwischen Mannigfaltigkeiten



Des weiteren wird in Abschnitt 3.2 die K-means Methode (siehe Punkt 1 in Einleitung) und dementsprechend auch das K-means Verfahren vorgestellt. An grafischen Beispielen wird verdeutlicht, wie die Methode arbeitet und zu welchen Ergebnissen man hierbei kommt, nämlich konvexe Mengen. (Vgl. Abb. 6)

Abbildung 6: mögliche Endcluster beruhend auf der K-means Methode



3.1 Abstand zwischen Mannigfaltigkeiten

Ausgehend von den Ausführungen in der Einleitung (Fall 3) wird die vereinfachte Variante betrachtet, die folgendes Abstandsmaß suggeriert:

$$d_{\cdot} := \min_{x_i \in C_1, x_j \in C_2} |x_i - x_j|, \tag{3.1}$$

wobei C_1, C_2 Cluster darstellen.

Das Verfahren, welches sich am besten für die Realisierung dieses Abstandsmaßes eignet, ist das k-nearest neighbour Verfahren in einer etwas modifizierten Art. In diesem Fall ist $k = 1$ zu wählen. Es liegen alle Beobachtungen in einer gesamten Ausgangsmenge $\mathscr{C} = \{x_1, ..., x_n\}$, die somit die Ausgangsclusterung darstellt. Die Ermittlung der in unserem Fall nächsten $k = 1$ Nachbarn erfolgt mittels

$$d_i := \min_{i \neq j} d(x_i, x_j) \quad = \quad \min_{i \neq j} |x_i - x_j|^2. \tag{3.2}$$

Für die genaue Vorgehensweise bezogen auf das Untersuchungsziel wird das ursprüngliche 1-nearest Neighbour Verfahren (bzw. Single-Linkage-Verfahren) modifiziert. Es müssen für die Durchführung hierbei einige Dinge berücksichtigt bzw. ergänzt werden. Wird das Element x_a als Startelement gewählt, so wird zu diesem mittels (3.2) der nächstliegende Nachbar x_b ermittelt. Nun wird mit x_b fortgefahren, d. h. der nächste Nachbar ermittelt, wobei x_a von der Auswahl ausgenommen wird. Formell gilt somit

$$\begin{aligned}
d_a &= \min_{j \neq a} d(x_a, x_j) \\
&= \min_{j \neq a} |x_a - x_j|^2, \quad \text{d. h. Ermittlung des Minimums } \forall\, x_j \in \mathscr{C}\ , j = 1...n \\
&\hookrightarrow \text{ nächster Nachbar von } x_a\text{: } x_b \\
\Rightarrow d_{b|a} &= \min_{j \neq a,b} d(x_b, x_j) \\
&= \min_{j \neq a,b} |x_b - x_j|^2, \quad \text{d. h. Ermittlung des Minimums } \forall\, x_j \in \mathscr{C} \setminus \{x_a\}, j = 1...n \\
&\vdots \\
&\vdots \\
\Rightarrow d_{t|a...} &= d(x_t, x_a) \\
&= |x_t - x_a|^2
\end{aligned}$$

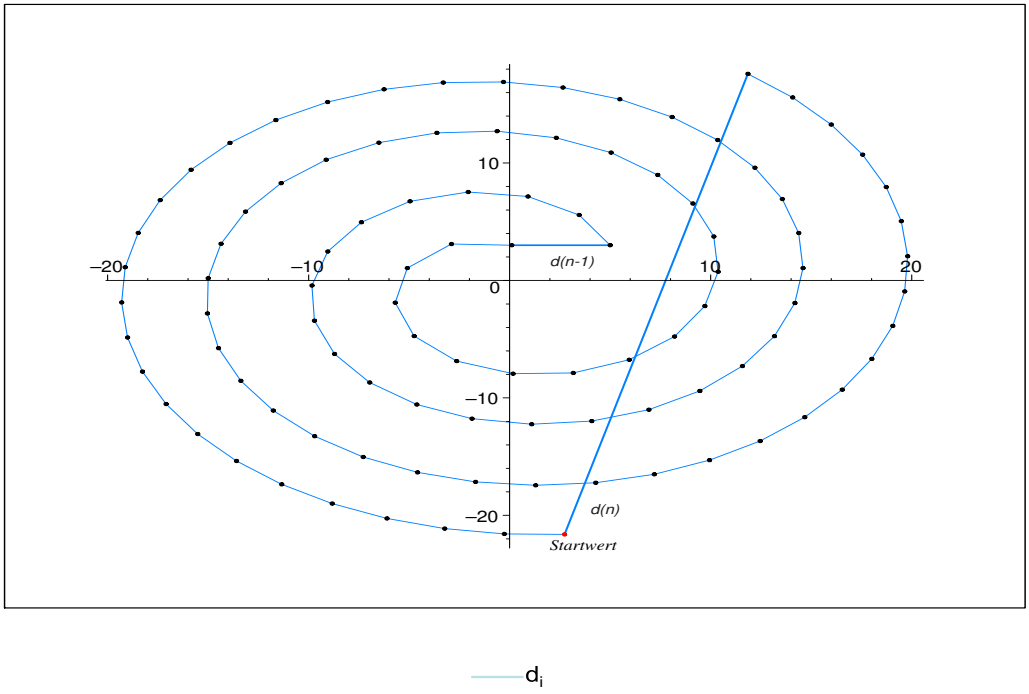
Es wird solange fortgefahren, bis für jedes Element der nächste Nachbar ermittelt wurde.

Die Entscheidung für die Neuordnung der Elemente erfolgt anhand der ermittelten Distanzen. Durch die „Punkt für Punkt-Vorgehensweise“ lässt sich klar erkennen, wann das Cluster gewechselt wird. Beim Übergang von einem möglichen Cluster zum anderen treten erhebliche Distanzänderungen auf, die sich wesentlich von den anderen unterscheiden. Nach Ordnung der erhaltenen Distanzwerte und Umbenennung

$$d_{(1)}, d_{(2)}, ..., d_{(n)}, \text{ mit } d_{(1)} \leq d_{(2)} \leq ... \leq d_{(n)}$$

erhält man die Trennstelle für die Zuordnung zu den Clustern.[12] Alle Elemente, die zwischen $d_{(n)}$ und $d_{(n-1)}$ liegen, werden jeweils in ein Cluster eingeordnet. Folgende Abb. 7 verdeutlicht den Verlauf des Verfahrens und rechtfertigt die Entscheidung über die Festlegung der Cluster.

Abbildung 7: Verlauf

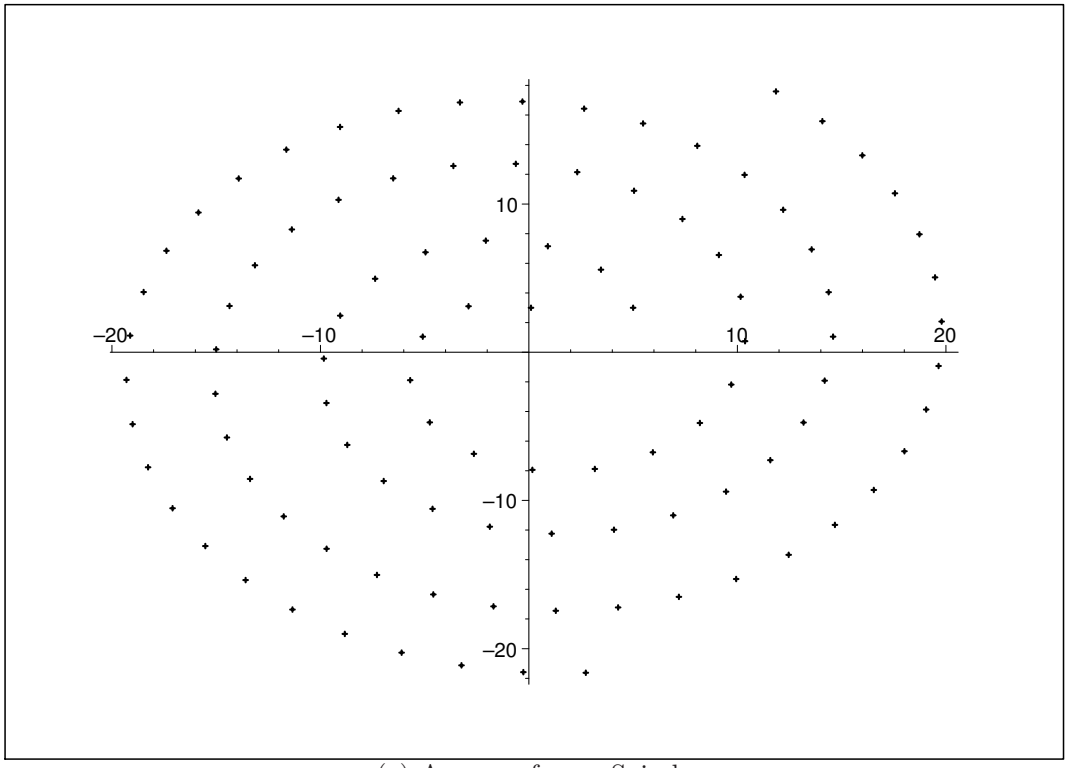


[12] vgl. Anhang A.4; S. 90-94

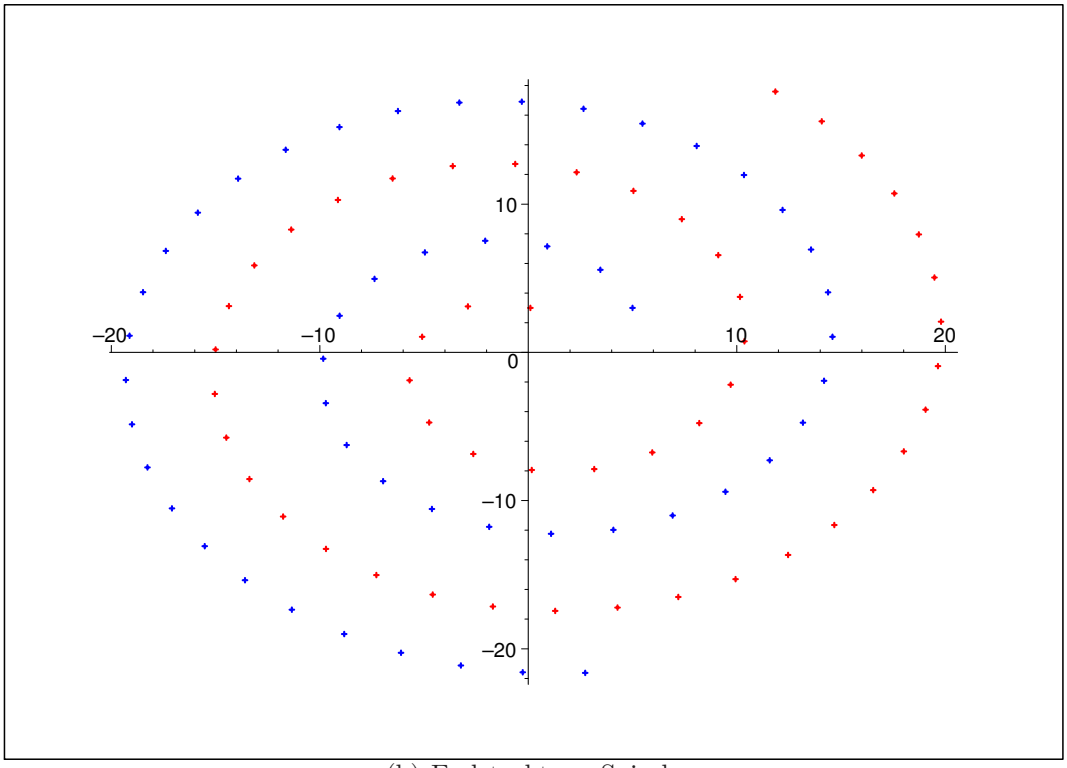
Für die unterschiedlichen Ausgangsmengen ergeben sich dann die zuvor erwähnten spiralförmigen, gekrümmten Endclusterungen.

Ist als Ausgangsmenge eine Punktewolke bestehend aus zwei Wurzelspiralen (Abb. 3.1(a)) gegeben, ergibt sich dann mittels der Vorgehensweise des mod. 1-nearest neighbour Verfahrens die Endstruktur in Abb. 3.1(b).

Abbildung 8: Struktur 1



(a) Ausgangsform - Spirale



(b) Endstruktur - Spirale

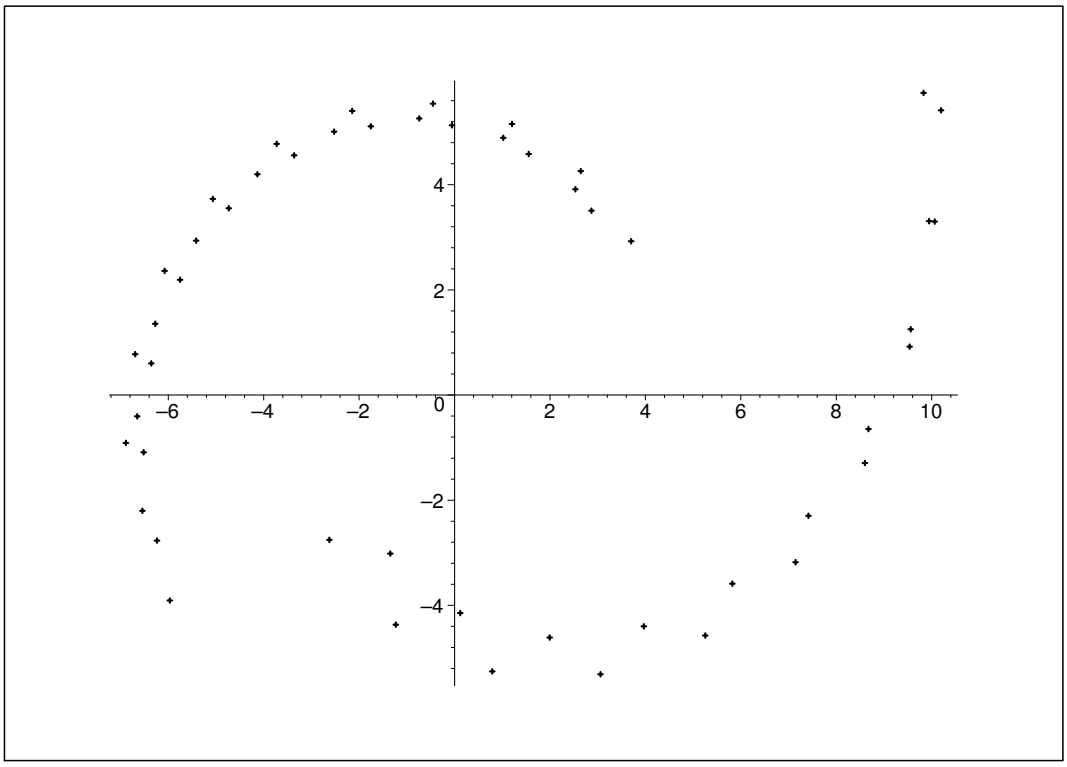
Um von vornherein mögliche Fehler bei der Trennung der Gesamtausgangsmenge zu vermeiden, ist es ratsam, wie bereits bei Abb. 7 erkennbar, einen geeigneten Startwert für die Durchführung dieses Algorithmus zu wählen.[13]

Ebenfalls muss gewährleistet werden, dass der Abstand zwischen den Objekten auf den Spiralen kleiner ist, als der Abstand zwischen den Spiralen. D. h., das Erhalten der optimalen Lösung dieses aufgezeigten Verfahrens ist abhängig von der Anzahl der betrachteten Elemente.

[13] In diesem Fall ist der Startwert das am weitesten vom Nullpunkt entfernte Element zu wählen.

Unter den gerade erwähnten Bedingungen lassen sich ebenfalls gekrümmte Strukturen erkennen. Die Ausgangsmenge stellt eine Punktewolke bestehend aus zwei „Halbkreisen“ dar (vgl. Abb. 9).

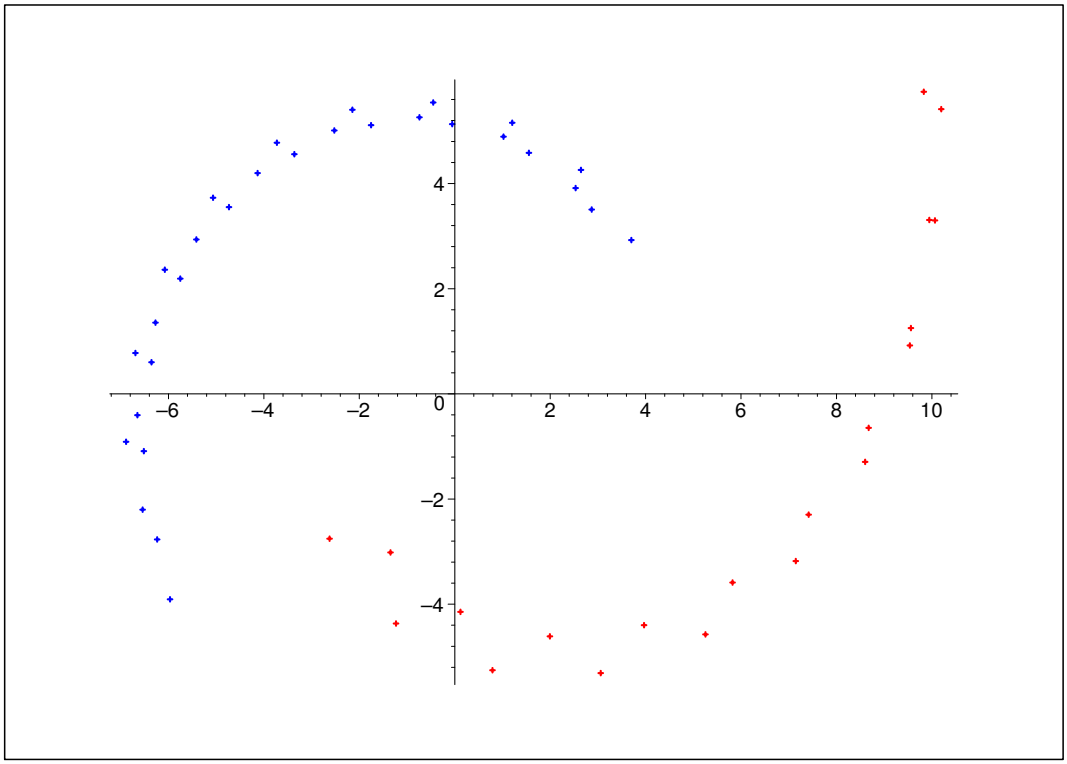
Abbildung 9: Ausgangsstruktur gekrümmt



Vom gewählten Startwert werden nach und nach alle Punkte der unteren Punktewolke und dann der oberen durchlaufen.

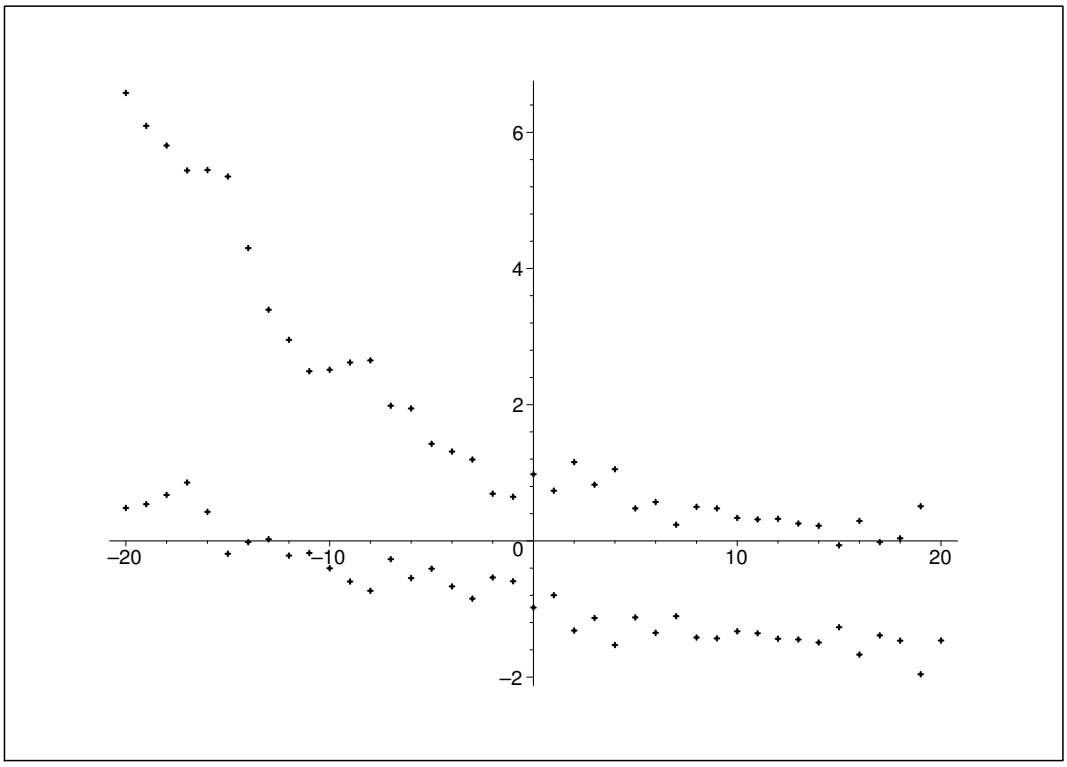
Es ist klar erkennbar, dass eine eindeutige Trennung zwischen den beiden Punktewolken stattfindet (vgl. Abb. 10).

Abbildung 10: Endstruktur gekrümmt



Ein weiteres Beispiel für Strukturerkennung des vorgestellten Verfahrens ist in Abb. 12 zu sehen.[14] Die Ausgangsmenge stellt eine Punktewolke dar, bei der sich die Beobachtungen entlang zweier Kurven anordnen, die als diese erkannt werden sollen (vgl. Abb. 11).

Abbildung 11: Ausgangsstruktur

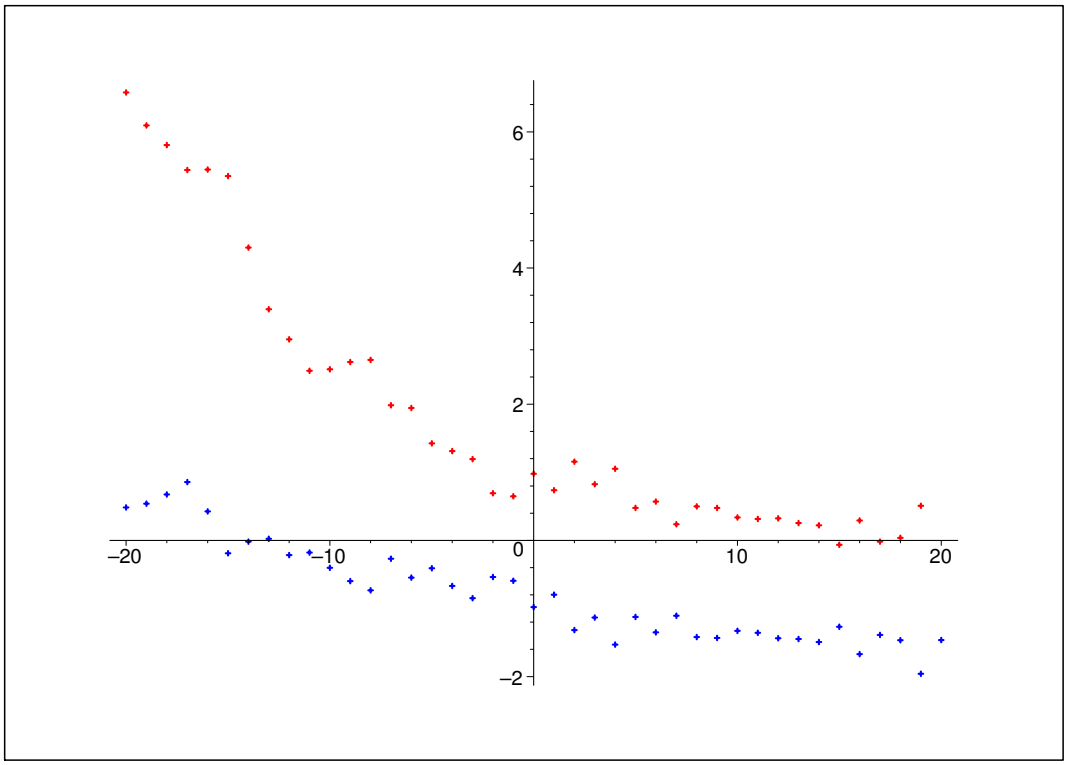


Wie bereits erwähnt, muss für die Durchführung des Verfahrens der richtige Startwert festgelegt werden.

Das Verfahren erkennt eine Kurvenstruktur und teilt die „Punktewolke“ in die zuvor genannten zwei Endcluster ein, wobei die Cluster jeweils die beiden Punktewolken darstellen (siehe Abb. 12).

[14] vgl. weitere Beispiele: Anhang A.2

Abbildung 12: Endstruktur

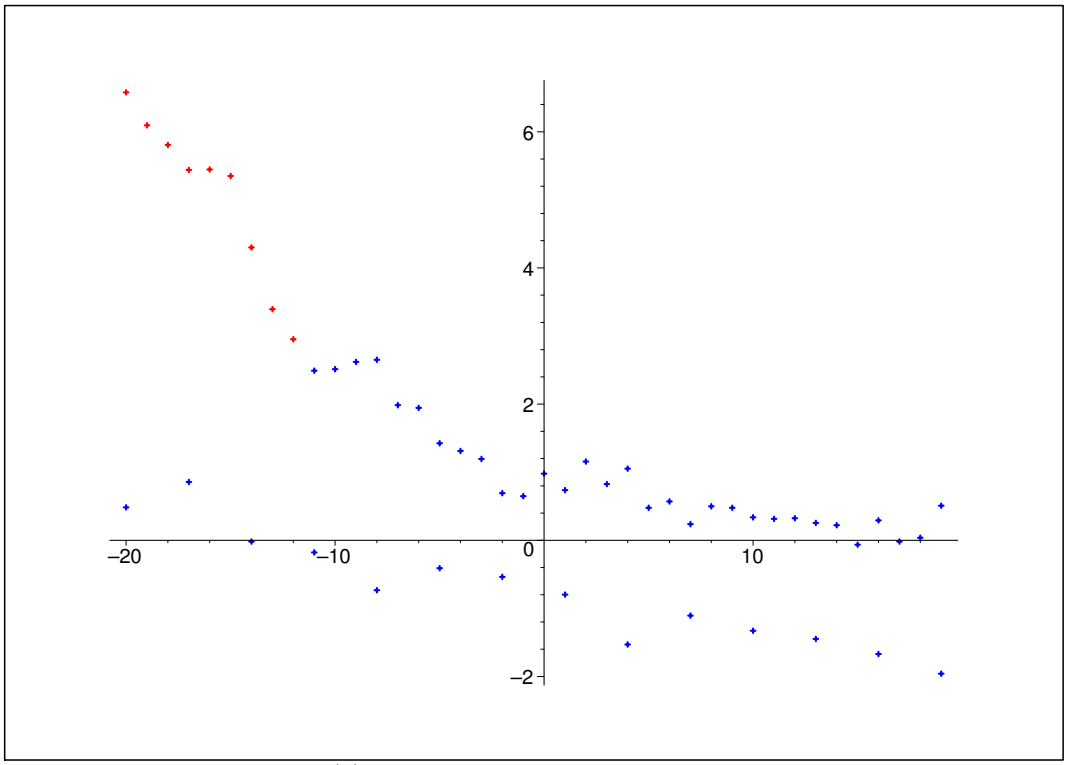


Bei einem größeren Abstand zwischen den Elementen bzw. bei einer kleinere Elementanzahl kann es zu einer „verfälschten“ Endclusterung kommen. Es findet ein Übergang zum anderen Cluster statt, der an dieser Stelle nicht vorgesehen ist. Es erfolgt somit eine Mischung der Cluster untereinander und damit keine klare strukturerkennende Endclusterung. Die Distanzen $d_{(n)}$ und $d_{(n-1)}$ werden an Positionen ermittelt, die zu einer falschen Trennung[15] zwischen den Elementen führt.

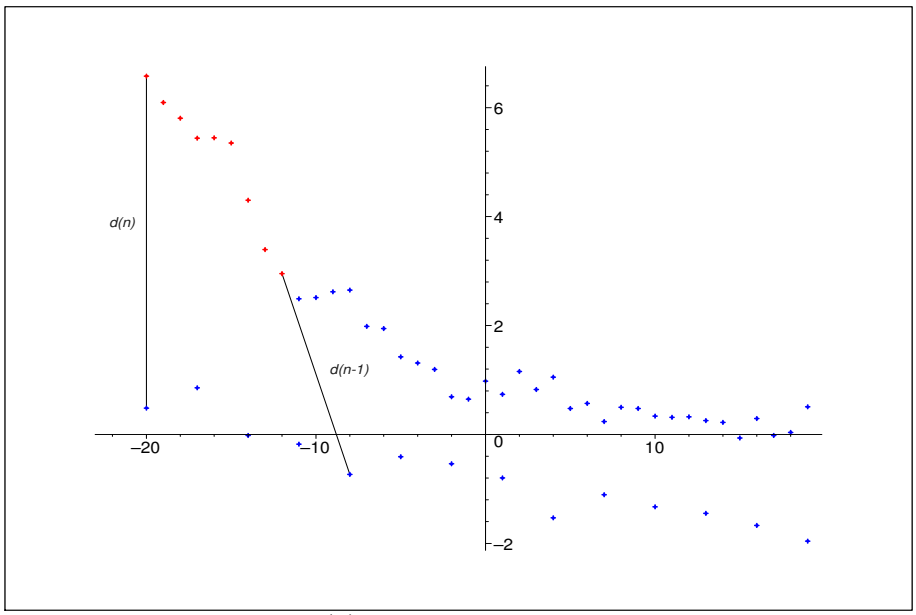
Die Abb. 13 zeigt eine mögliche Endclusterung bedingt durch eine zu geringe Elementanzahl (vgl. Abb. 12).

[15] im Sinne des mod. 1-nearest neighbour Verfahrens

Abbildung 13: Gegenbeispiel[16]



(a) Polynome - verzerrtes Ergebnis



(b) Polynome - Verlauf

Die Durchführung des mod. 1-nearest neighbour Verfahrens und dessen Ergebnis ist demzufolge nicht nur abhängig von der Wahl des richtigen Startwertes, sondern auch von der „richtigen“ Anzahl der Beobachtungsvektoren. Wie zu sehen war, kommt es bei einer zu geringen Anzahl zu keiner optimalen Clusterung.

[16]Vgl. Anhang A.4; S. 95/96 (Tabelle 2)

3.2 K-means Methode

Das K-means Verfahren dient der Realisierung der K-means Methode. Das Abstandsmaß legt das Varianzkriterium

$$D^2(\mathscr{C}^{(s)}) = \sum_{i=1}^{K} \sum_{x_j \in C_i} |x_j - \bar{x}_i|^2 \tag{3.3}$$

zugrunde, wobei

$$\bar{x}_i := \frac{1}{n_i} \sum_{\substack{l=1 \\ x_l \in C_i}}^{n_i} x_l$$

den entsprechenden Mittelwert des jeweiligen Clusters darstellt.

Das Varianzkriterium dient nicht nur der Strukturfindung, sondern auch, wie brereits erwähnt, der Ermittlung der (sub-)optimalen Clusterlösung. Es erfolgt eine schrittweise Verbesserung der Ausgangsgruppierung.

Die Ausgangsclusterung $\mathscr{C}^{(s)} = (C_1^{(s)}, ..., C_K^{(s)})$ mit Umfängen $n_1, ..., n_K$ mit der vorgegebenen Startanzahl K der Cluster stellt den Startpunkt des Algorithmus dar. Es erfolgt die Ermittlung der jeweiligen Mittelwerte der Cluster. Liegt nun im s-ten Iterationsschritt $x_r \in C_a$ im Sinne der euklidischen Distanz näher am Mittelwert eines anderen Clusters, als am Mittelwert des eigenen

$$|x_r - \bar{x}_a|^2 > |x_r - \bar{x}_b|^2$$

und ist für dieses Element das Varianzkriterium minimal, so wird x_r umgeordnet.

Nach der Umordnung von x_r müssen die Mittelwerte der neu entstandenen Cluster

$$C_{a,neu} := C_a \setminus \{x_r\}$$

und

$$C_{b,neu} := C_b \cup \{x_r\}$$

neu berechnet werden. Man erhält daher

$$\begin{aligned}\bar{x}_{a,neu} &= \frac{1}{n_a-1}\sum_{\substack{l=1\\ l\neq r}}^{n_a-1} x_l = \frac{1}{n_a-1}(n_a\bar{x}_a - x_r)\\ \bar{x}_{b,neu} &= \frac{1}{n_b+1}\big(\sum_{l=1}^{n_b} x_l\big) + x_r\\ &= \frac{1}{n_b+1}(n_b\bar{x}_b + x_r).\end{aligned}$$

Dann gilt für diese Clusterung, die sich immer noch in der Iterationsebene (s) befindet, die Bezeichnung $\mathscr{C}^{(s)}_{r,a,b}$. Für die Umordnung gilt dann folgendes Distanzmaß

$$\begin{aligned}D^2(\mathscr{C}^{(s)}_{r,a,b}) &= D^2(C^{(s)}_1,...,C^{(s)}_{a,neu},...,C^{(s)}_{b,neu},...,C^{(s)}_K)\\ &= D^2(\mathscr{C}^{(s)}) - \sum_{x_j\in C_a}|x_j-\bar{x}_a|^2 - \sum_{x_j\in C_b}|x_j-\bar{x}_b|^2\\ &\quad + \sum_{x_j\in C_{a,neu}}|x_j-\bar{x}_{a,neu}|^2 + \sum_{x_j\in C_{b,neu}}|x_j-\bar{x}_{b,neu}|^2.\end{aligned}$$

Da die restlichen Maße innerhalb der nichtbetroffenen Cluster keine Veränderung erfahren, genügt es, sich auf die eben dargestellten Teile der Berechnung zu konzentrieren. Um diese Berechnung zu vereinfachen, betrachten wir zunächst die Einzelkomponenten dieser Gleichung.

Nach diversen Umrechnungen ergibt sich dann für

$$\begin{aligned}\sum_{x_j\in C_a}|x_j-\bar{x}_a|^2 - \sum_{x_j\in C_{a,neu}}|x_j-\bar{x}_{a,neu}|^2 &= \sum_{j=1}^{n_a}|x_j-\bar{x}_a|^2 - \sum_{j=1}^{n_a-1}|x_j-\bar{x}_{a,neu}|^2\\ &= \big(\sum_{j=1}^{n_a-1}|x_j-\bar{x}_a|^2\big) + |x_r-\bar{x}_a|^2\\ &\quad - \sum_{j=1}^{n_a-1}|x_j-\frac{1}{n_a-1}(n_a\bar{x}_a-x_r)|^2\\ &= \frac{n_a}{n_a-1}|x_r-\bar{x}_a|^2.\end{aligned}$$

Analog gilt

$$\sum_{x_j\in C_b}|x_j-\bar{x}_b|^2 - \sum_{x_j\in C_{b,neu}}|x_j-\bar{x}_{b,neu}|^2 = -\frac{n_b}{n_b+1}|x_r-\bar{x}_b|^2.$$

Es ergibt sich nun

$$D^2(\mathscr{C}^{(s)}) - D^2(\mathscr{C}^{(s)}_{r,a,b}) = \frac{n_a}{n_a - 1}|x_r - \bar{x}_a|^2 - \frac{n_b}{n_b + 1}|x_r - \bar{x}_b|^2. \tag{3.4}$$

Nach der Berechnung von (3.4) für alle r, für die $|x_r - \bar{x}_a|^2 > |x_r - \bar{x}_b|^2 \ \forall a, b$ erfüllt ist, erfolgt die Entscheidung über die endgültige Umordnung/Neuzuordnung innerhalb des s-ten Iterationschrittes. Hierzu wird geprüft, bei welchem Element die maximale Differenz zwischen der alten und der neuen Clusterung besteht, d. h. bei welchem Element die größte Veränderung auftritt.

Die neue Gruppierung $\mathscr{C}^{(s+1)}$ ist demzufolge diejenige Clusterung, bei der x_{r_s} aus $C^{(s)}_{a_s}$ nach $C^{(s)}_{b_s}$ umgeordnet wird, wobei

$$(r_s, a_s, b_s) = \arg\max_{r,a,b} \left(D^2(\mathscr{C}^{(s)}) - D^2(\mathscr{C}^{(s)}_{r,a,b}) \right). \tag{3.5}$$

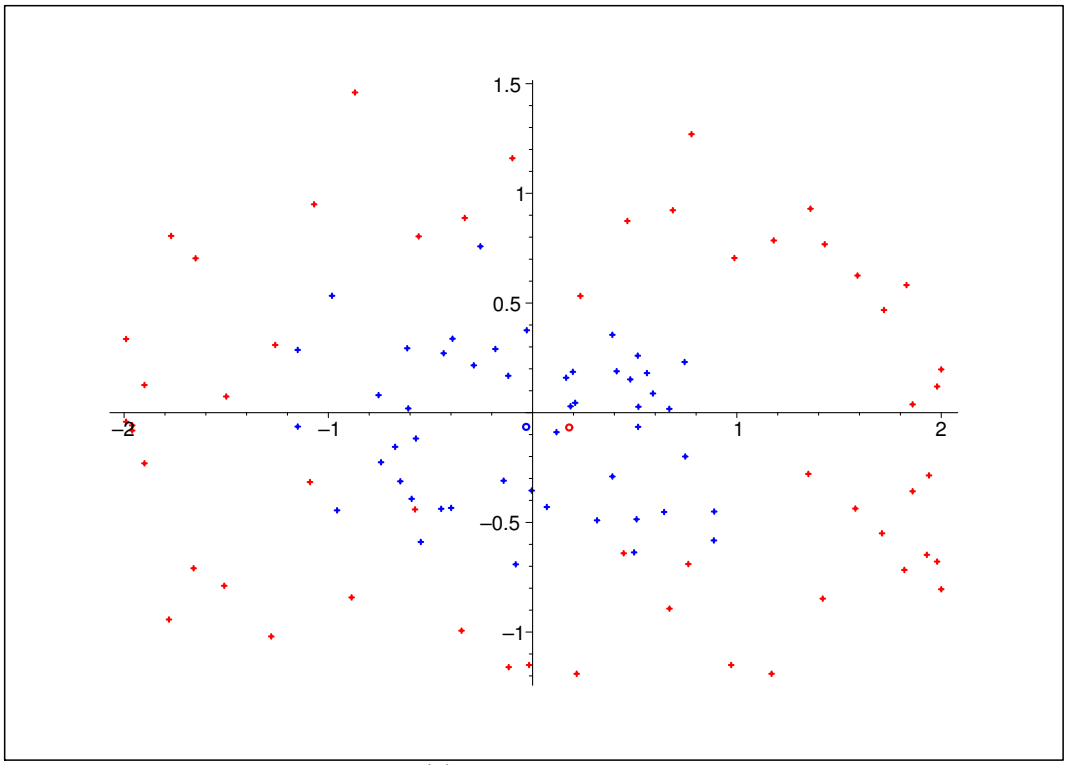
Nun werden die neuen Mittelwerte innerhalb dieser Clusterung berechnet und wiederum dasjenige Element umgeordnet, welches das Varianzkriterium verbessert. Der Algorithmus wird solange fortgeführt bis keine Verbesserung des Kriteriums eintritt und auch kein Gruppenwechsel mehr vonstatten gegangen ist. Es ist erkennbar, dass eine ständige Ausrichtung der Elemente in Richtung der Mittelwerte erfolgt, welche die Entwicklung der einzelnen Cluster hin zu konvexen Endmengen erklärt. Das K-means Verfahren liefert ebenfalls Endcluster mit einer ähnlichen Anzahl von Beobachtungen.

Die zuvor ausführlich erläuterte Vorgehensweise des K-means Verfahrens und der daraus resultierenden Endstruktur der Ergebnisse sollen durch die folgenden aufgeführten Beispiele verdeutlicht werden.[17]

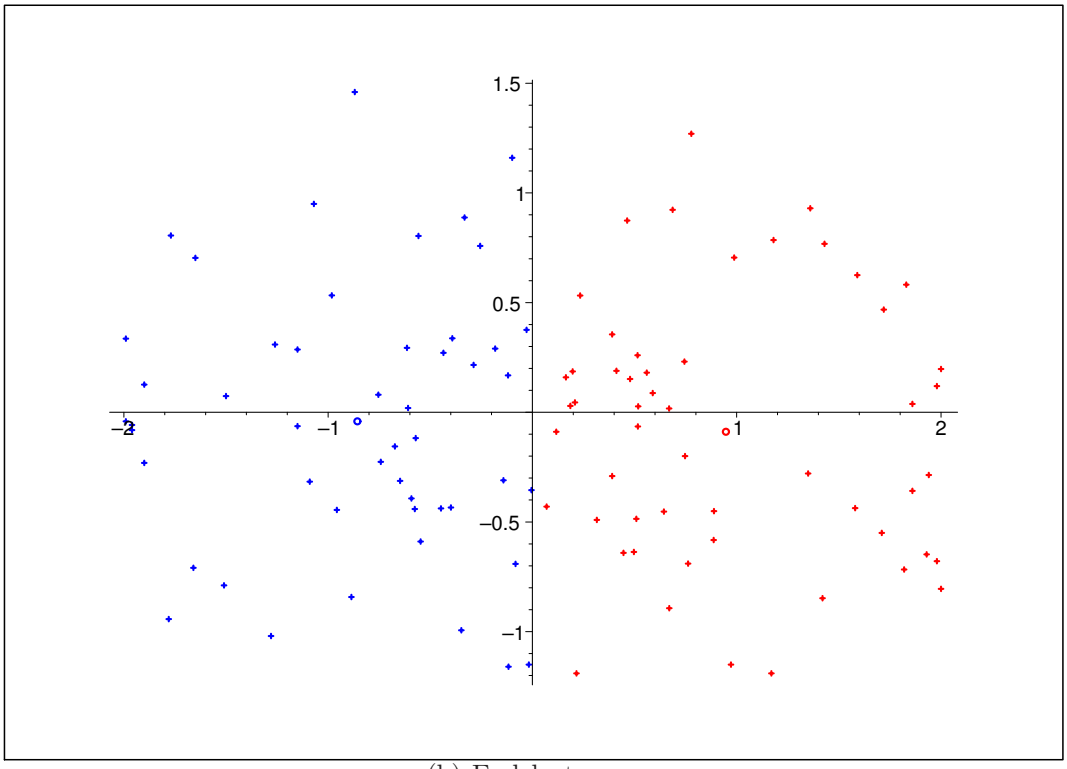
Die gegebenen Daten aus dem ersten Beispiel werden zur vereinfachten Darstellung in zwei Ausgangscluster der Gesamtmenge geteilt (siehe Abb. 14(a)). Nach der Ermittlung der entsprechenden Mittelwerten, ergibt sich während der Durchführung eine Verschiebung der Mittelwerte und auch eine Neuordnung der Objekt, aus der dann die Endclusterung Abb. 14(b) resultiert.

[17] vgl. weitere Beispiele: Anhang A.3

Abbildung 14: Beispiel 1 für das K-means Verfahren



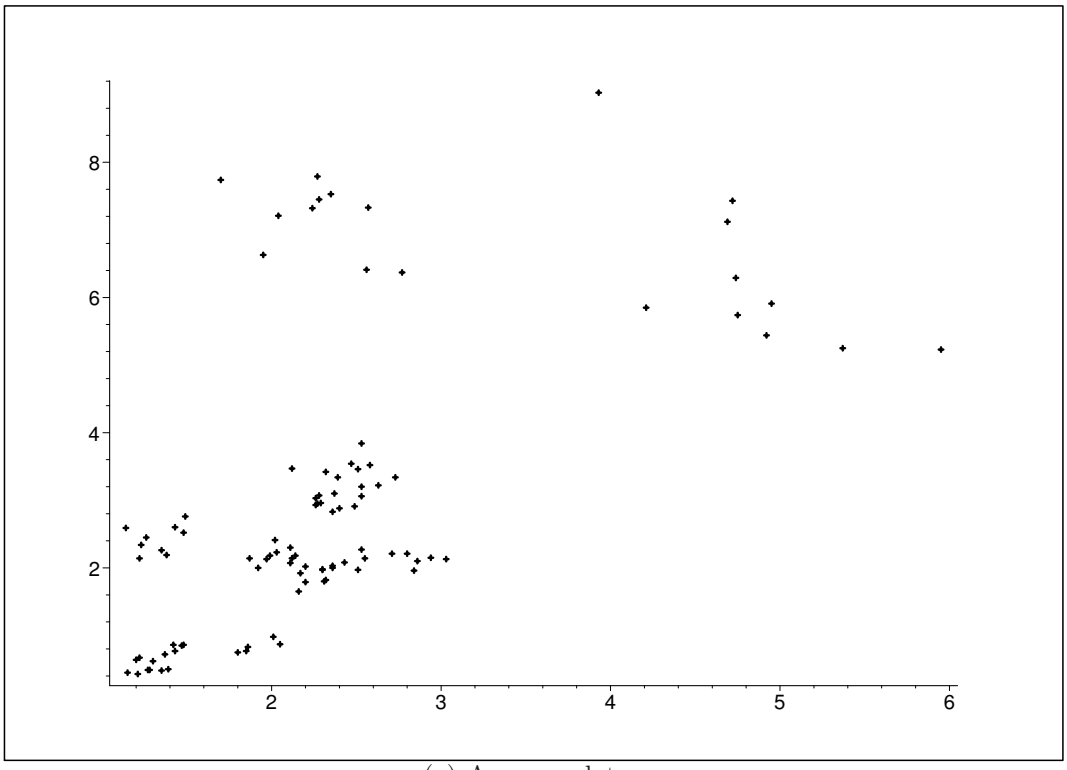
(a) Startclusterung



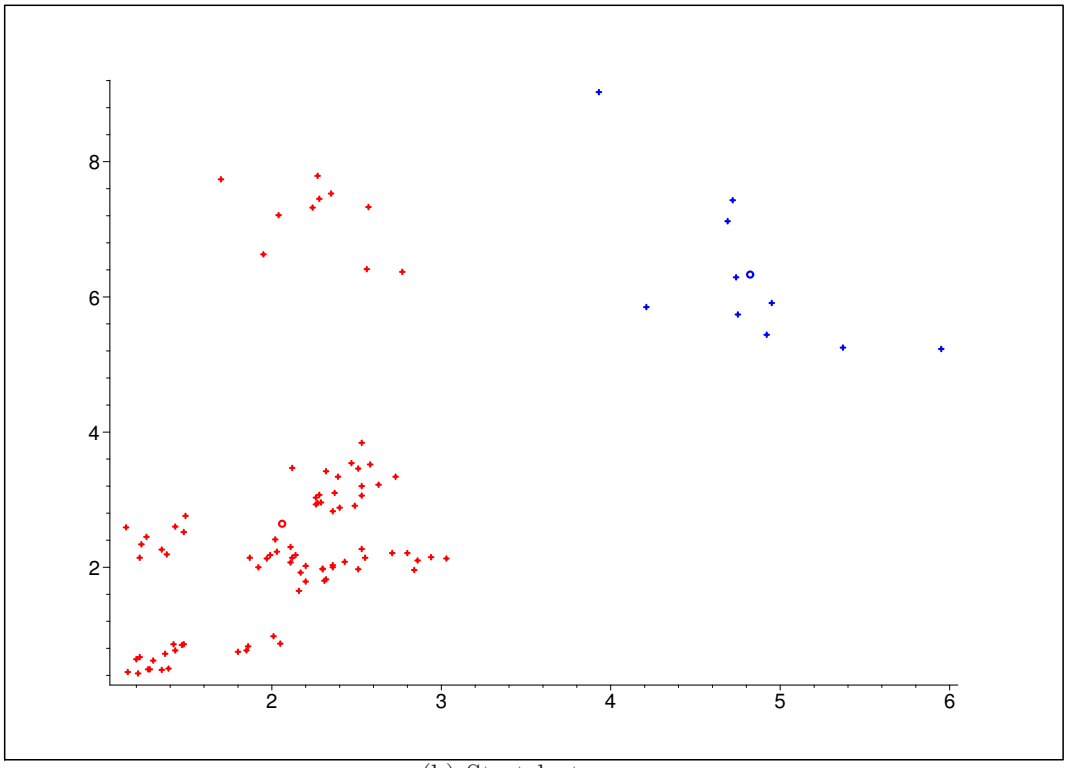
(b) Endclusterung

Ein anderes Beispiel soll zeigen, dass eine optisch erkennbare Clusterung in den Ausgangsdaten nicht notwendigerweise die optimale Clusterung im Sinne des Untersuchungsziels ist. Unter Verwendung der Nervenzellendaten aus Kapitel 1 wird die Ausgangsclusterung Abb. 15(b) gewählt. Wie bereits aus Kapitel 1 bekannt, ergibt sich jedoch eine andere Endclusterung. Die Daten zerfallen in 2 Cluster (Vgl. Abb. 15(c)), die aus inhaltlicher Sicht mit Krankheiten in Verbindung gebracht werden können. Das K-means Verfahren liefert somit die suboptimale Clusterung der Daten.

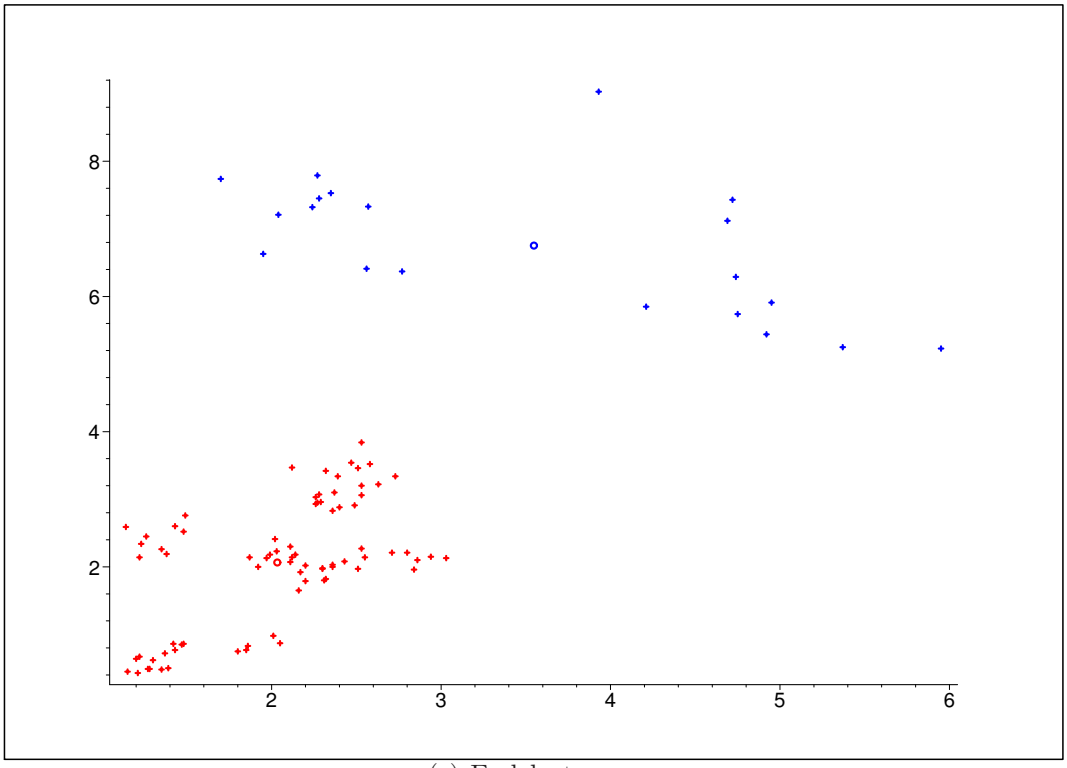
Abbildung 15: Beispiel 2 für das K-means Verfahren



(a) Ausgangsdaten



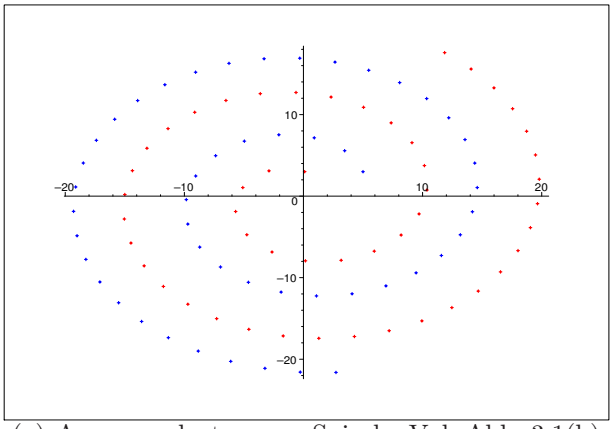
(b) Startclusterung



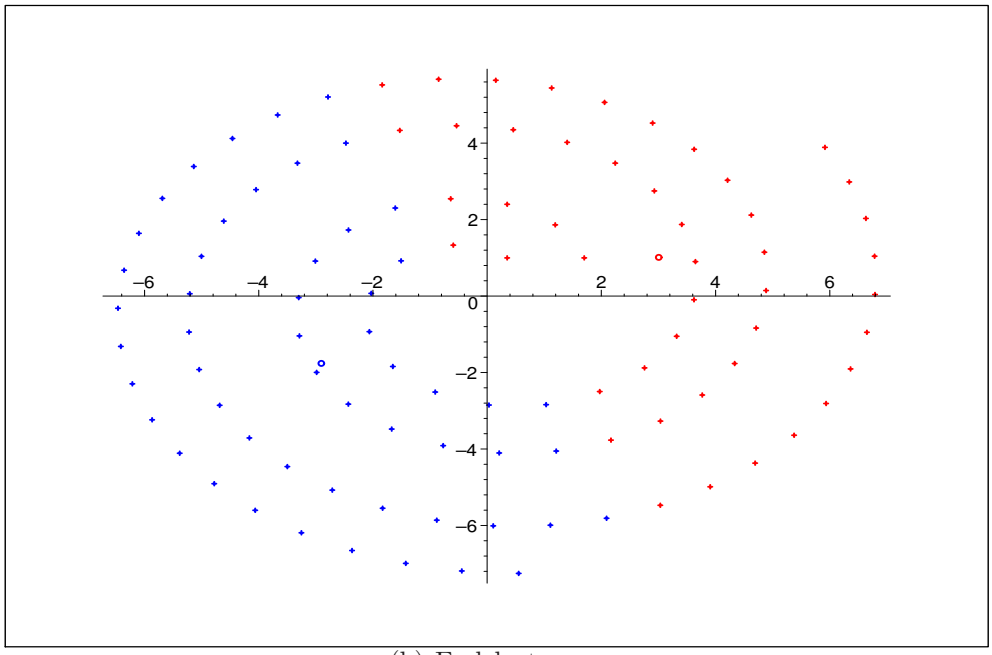
(c) Endclusterung

Die nun folgenden Abbildungen 16-18 sollen vergleichend zum mod. 1-nearest neighbour Verfahren betrachtet werden. Durch den direkten Vergleich wird nochmals deutlich, dass das K-means Verfahren Cluster mit konvexer Gestalt findet.

Abbildung 16: Beispiel 3 für das K-means Verfahren



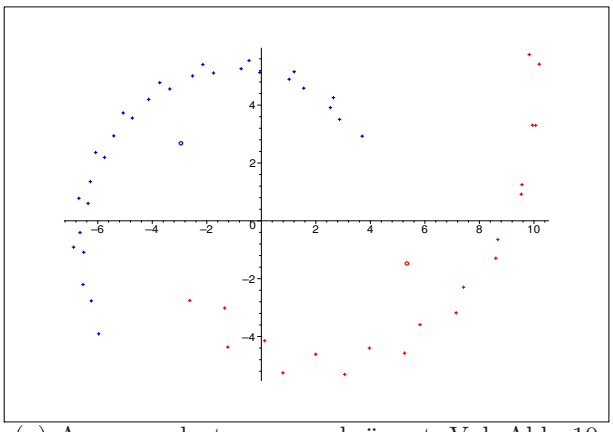
(a) Ausgangsclusterung - Spirale; Vgl. Abb. 3.1(b)



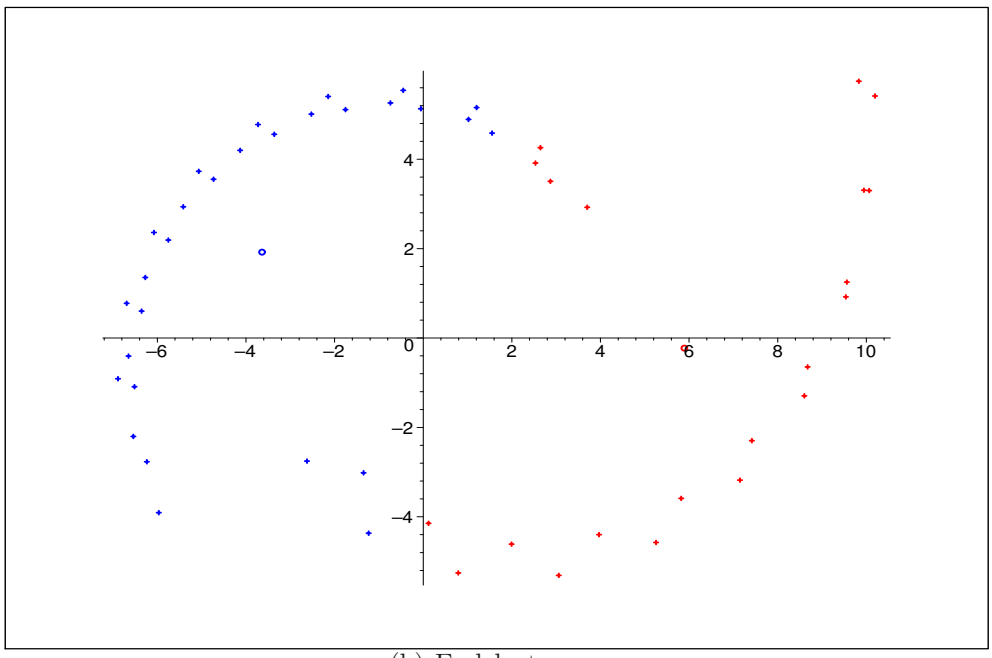
(b) Endclusterung

Die Ausgangscluster sind jeweils die Endcluster des 1-nearest neigbour Verfahrens. Ausgehend von diesen ergeben sich durch die Einbeziehung der Mittelwerte eine andere Clusterung als zuvor. (Vgl. auch Abb. 18)

Abbildung 17: Beispiel 4 für das K-means Verfahren

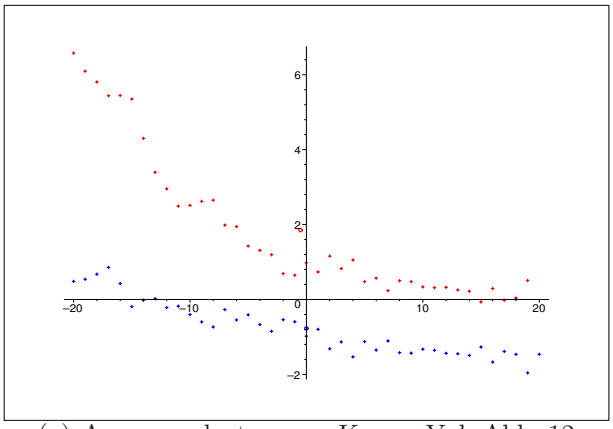


(a) Ausgangsclusterung - gekrümmt; Vgl. Abb. 10

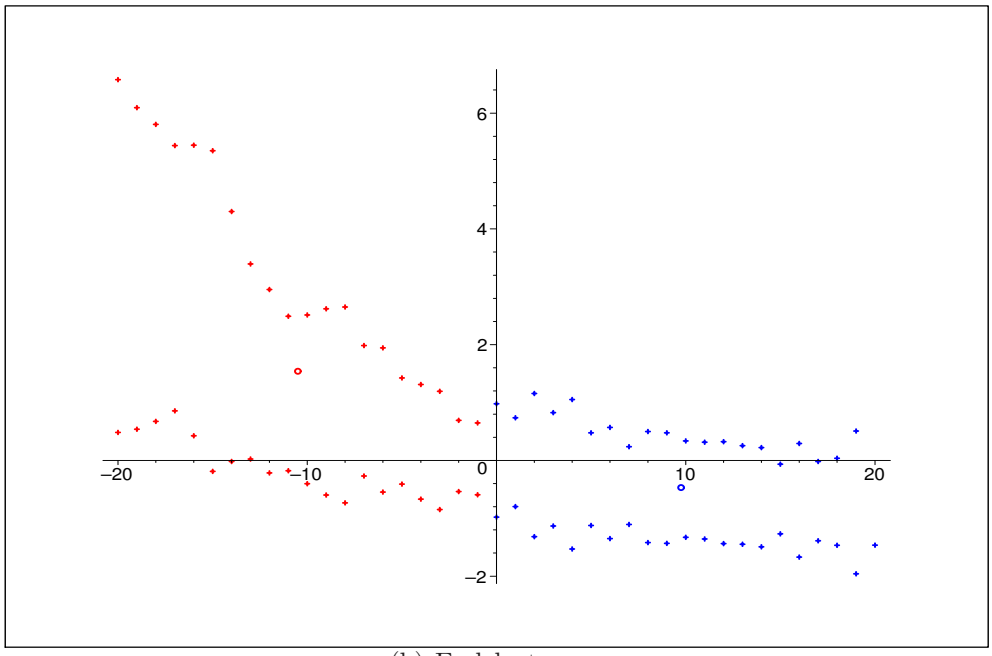


(b) Endclusterung

Abbildung 18: Beispiel 5 für das K-means Verfahren



(a) Ausgangsclusterung - Kurve; Vgl. Abb. 12



(b) Endclusterung

Die Beispiele lassen deutlich erkennen, dass das Varianzkriterium ungeeignet ist, um langgetreckte oder gekrümmte Cluster zu finden.

4 Asymptotische Eigenschaften

Dieses Kapitel geht auf die statistischen Eigenschaften der Clusterverfahren ein. Es wird das asymptotische Verhalten des K-means Verfahrens betrachtet, speziell die Konsistenz und die asymptotische Normalität des Vektors der Clusterzentren des K-means Verfahrens. Es wird eine wahre Clusterstruktur und eine Verteilung vorausgesetzt.

Es soll das Verhalten des K-means Verfahrens unter wohl definierten Bedingungen untersucht werden. Dabei sind die Beobachtungen $x_1, ..., x_n$ Realisierungen von unabhängig, identisch verteilter Zufallsgrößen $X_1, ..., X_n$ mit der Verteilung P. Das asymtotische Verhalten wird in Abhängigkeit von P bestimmt.

4.1 Konsistenz

Dieser Abschnitt zeigt auf, dass die Menge der optimalen Clusterzentren der K Cluster fast sicher konvergiert, falls die Anzahl der Beobachtungen steigt.

Die asymptotische Eigenschaft der Konsistenz betrachtet, inwieweit ein Schätzer bei unendlich großer Stichprobe vom wahren Wert entfernt liegt. Hierbei stellt der Vektor bzw. die Menge der optimalen Clusterzentren den Schätzer in Bezug auf die Clusterzentren, die das Varianzkriterium minimieren, dar.

Das K-means Verfahren beschreibt die Clusterung einer Menge von Punkten in K Gruppen. Gegeben sind die Beobachtungen $x_1, x_2, ..., x_n \in \mathbb{R}^s$. Zu Beginn werden die Clusterzentren $a_1, a_2, ..., a_K$ gewählt, die

$$W_n(a) = \frac{1}{n} \sum_{i=1}^{n} \min_{1 \leq j \leq K} \|x_i - a_j\|^2 \tag{4.1}$$

minimieren sollen, wobei $\|.\|$ die euklidische Norm ist. Die Menge $A = \{a_1, a_2, ..., a_K\}$ ist hierbei die Menge der Clusterzentren und enthält K oder weniger Zentren. Danach wird, wie schon zuvor erwähnt, jedes x_i seinem nächsten Cluster zugeordnet.

Die Menge $\{x_1, x_2, ..., x_n\}$ ist eine Menge von Realisierungen von unabhängigen ZG mit beliebiger Verteilung P und P_n, dem entsprechenden empirischen Maß.

Das eigentliche Problem (4.1) liegt in der Minimierung von

$$W(A, \mathsf{P}_n) := \int \min_{a \in A} \|x - a\|^2 \mathsf{P}_n(\,\mathrm{d}x).$$

In diesem Abschnitt der Arbeit wird gezeigt, dass

$$W(A, \mathsf{P}_n) \to W(A, \mathsf{P}) := \int \min_{a \in A} \|x - a\|^2 \mathsf{P}(\,\mathrm{d}x) \ f.s.$$

und dass die Menge der optimalen Clusterzentren $B_n = \{b_{n1}, b_{n2}, ..., b_{nK}\}$ f.s. gegen $\bar{A} = \{\bar{a}_1, \bar{a}_2, ..., \bar{a}_K\}$, der Menge der Zentren die $W(., \mathsf{P})$ minimiert, konvergiert. Es soll bewiesen werden, dass

$$b_{ni} \to \bar{a}_i \ f.s.,$$

für $i = 1, ..., K$.

Für den weiteren Verlauf der Arbeit und die darin enthaltenen Theoreme und Beweise wird vorausgesetzt, dass die optimale Menge der Clusterzentren B_n in einem kompakten Bereich in $\mathbb{R}^d$ liegt.

Bevor näher auf das Konsistenztheorem eingegangen werden kann, müssen zuvor einige wesentliche Dinge definiert werden.

$X_1, X_2, ...X_n \in \mathbb{R}^d$ sind reellwertige unabhängige Zufallsvariablen mit der Verteilung P und dem dazugehörigen empirischen Maß P_n. Wie schon erwähnt, ist die Einteilung von

$\{x_1, ..., x_n\}$ in K Cluster das Ziel.

Folgende Voraussetzungen müssen gegeben werden:

(V1) Für jedes Wahrscheinlichkeitsmaß Q über $\mathbb{R}^d$ und jeder endlichen Teilmenge $A \subset \mathbb{R}^d$ sei:

$$\Phi(A, Q) := \int \min_{a \in A} \phi(\|x - a\|) Q(\mathrm{d}x),$$

$$m_K(Q) := \inf\{\Phi(A, Q) : \ A \text{ enthält } K \text{ oder weniger Punkte}\}$$

(V2) Für das gegebene K, sind die Mengen $B_n = B_n(K)$ und $\bar{A} = \bar{A}(K)$ so zu wählen, dass jeweils

$$\Phi(B_n, \mathsf{P}_n) = m_K(\mathsf{P}_n)$$

und

$$\Phi(\bar{A}, \mathsf{P}) = m_K(\mathsf{P})$$

erfüllt ist.

(V3) Eine weitere Bedingung für das nachfolgende Konsistenztheorem ist die Stetigkeit der Funktion ϕ. Ebenfalls ist ϕ eine nicht fallende Funktion und es gilt $\phi(0) = 0$.

(V4) Weiterhin ist vorauszusetzen, dass

$$\exists\ \lambda = const. \ : \ \phi(2r) \leq \lambda \phi(r) \ \ \forall r > 0. \tag{4.2}$$

(V5) Es gilt

$$\int \phi(\|x\|) \mathsf{P}(\mathrm{d}x) < \infty \ \Rightarrow \ \Phi(A, \mathsf{P}) < \infty \ \ \forall A.$$

(V6) $\forall\ a \in \mathbb{R}^d$

$$\begin{aligned}
\int \phi(\|x - a\|) \mathsf{P}(\mathrm{d}x) &\leq \int \phi(\|x\| - \|a\|) \mathsf{P}(\mathrm{d}x) \leq \int \phi(\|x\|) + \phi(\|a\|) \mathsf{P}(\mathrm{d}x) \\
&\leq \int \phi(2\|x\|) \mathsf{P}(\mathrm{d}x) + \int \phi(2\|a\|) \mathsf{P}(\mathrm{d}x) \\
&\leq \phi(2\|a\|) + \int\limits_{\|x\| \geq \|a\|} \phi(2\|x\|) \mathsf{P}(\mathrm{d}x) \\
&\overset{(4.2)}{\leq} \phi(2\|a\|) + \lambda \int \phi(\|x\|) \mathsf{P}(\mathrm{d}x).
\end{aligned} \tag{4.3}$$

(V7) M ist groß genug, so dass

$$\lambda \int\limits_{\|x\| \geq 2M} \phi(\|x\|) \mathsf{P}(\,\mathrm{d}x) \; < \varepsilon \tag{4.4}$$

gilt. Hierbei ist ε so zu wählen, dass $\varepsilon + m_K(\mathsf{P}) < m_{K-1}(\mathsf{P})$.

Wie schon erwähnt ist das eigentliche Ziel dieses Abschnittes der Beweis des folgenden Theorems.

Theorem 1 [18] *Es gelten die Voraussetzungen (V1)-(V4). Dann gilt*

$$B_n \to \bar{A}(K) \; f.s.$$

und

$$\Phi(B_n, \mathsf{P}_n) \to \; \Phi(\bar{A}(K), \mathsf{P}) = m_K(\mathsf{P}) \; f.s.$$

Beweis: Zu Beginn wird ein M so groß gewählt, so dass für ein hinreichend großes n, mindestens ein Punkt aus B_n zu einem abgeschlossenen Bereich $K(M)$ gehört, wobei $K(M)$ ein abgeschlossenen Bereich vom Radius M um den Ursprung darstellt. Ebenfalls kann M so groß gesetzt werden, so dass

$$\phi(M - r)\mathsf{P}(G) > \int \phi(\|x\|)\mathsf{P}(\,\mathrm{d}x)$$

gilt, wobei r der Radius eines Gebietes G mit positivem P-Maß darstellt.
Es gilt

$$\phi(r) \to \infty \text{ für } r \to \infty.$$

Es gilt

$$\Phi(B_n, \mathsf{P}_n) \leq \Phi(A_0, \mathsf{P}_n) \tag{4.5}$$

für beliebige A_0, wobei A_0 wiederum höchstens K Punkte enthält.

Sei nun A_0 eine Menge bestehend aus einem einzigen Punkt im Ursprung, dann folgt

$$\Phi(A_0, \mathsf{P}_n) = \int \phi(\|x\|)\mathsf{P}_n(\,\mathrm{d}x) \; \to \; \int \phi(\|x\|)\mathsf{P}(\,\mathrm{d}x) \; f.s.$$

[18] vgl. Pollard 1981

da $\mathsf{P}_n \rightarrow \mathsf{P}$. Falls für unendlich viele Werte von n kein Element aus B_n in $K(M)$ enthalten ist, dann gilt

$$\limsup_n \Phi(B_n, \mathsf{P}_n) \geq \lim_n \phi(M-r)\mathsf{P}_n(G) = \phi(M-r)\mathsf{P}(G) \ f.s.$$

$$\Rightarrow \ \Phi(B_n, \mathsf{P}_n) > \Phi(A_0, \mathsf{P}_n) \ \lightning$$

Das erhaltene Ergebnis stellt einen Widerspruch zur vorher gegebenen Voraussetzung (4.5) dar. Daher kann man nun o.B.d.A. annehmen, dass B_n immer mindestens einen Punkt aus $K(M)$ enthält.

Als nächstes wird gezeigt, dass das abgeschlossene Gebiet $K(5M)$ alle Punkte aus B_n (für n groß genug) enthält.

Falls im entgegengesetzten Fall B_n nicht in $K(5M)$ enthalten ist, können die Clusterzentren, die außerhalb dieses Bereiches liegen, entfernt werden. Die neue Menge von höchstens $K-1$ Elementen kann ebenfalls $\Phi(., \mathsf{P}_n)$ reduzieren. (Hier ist 5 ein beliebiger Fehler. Es kann auch eine beliebige Zahl > 1 gewählt werden.)

Falls nun also mindestens ein Punkt aus B_n außerhalb von $K(5M)$ liegt, kann es im schlimmsten Fall dazu kommen, dass eine große Anzahl von Beobachtungen den Clusterzentren außerhalb von $K(5M)$ zugerechnet werden. Das Clusterzentrum $b_{n1} \in K(M)$ liegt innerhalb des Gebietes, wobei alle anderen Beobachtungen, die den außerhalb liegenden Zentren zugeordnet werden sollen, mindestens eine Distanz von $2M$ zum Ursprung aufweisen müssen. Wie bereits erwähnt werden die Zentren, die außerhalb liegen, entfernt. Sie stehen dann für die Zuordnung der Beobachtungen nicht mehr zur Verfügung. Durch dieses Auslöschen erfährt $\Phi(B_n, \mathsf{P}_n)$ einen kleinen Zuwachs. Es ergibt sich daher unter Berücksichtigung dieser Tatsache

$$\begin{aligned}
\int_{\|x\| \geq 2M} \phi(\|x - b_{n1}\|)\mathsf{P}_n(\,\mathrm{d}x) &\leq \int_{\|x\| \geq 2M} \phi(\|x\| + \|b_{n1}\|)\mathsf{P}_n(\,\mathrm{d}x) \\
&\leq \int_{\|x\| \geq 2M} \phi(2\|x\|)\mathsf{P}_n(\,\mathrm{d}x) \\
&\leq \lambda \int_{\|x\| \geq 2M} \phi(\|x\|)\mathsf{P}_n(\,\mathrm{d}x) \qquad (4.6)
\end{aligned}$$

als Extrabetrag, wobei $b_{n1} \in K(M)$.

Die so neu erhaltene Menge B_n^*, die durch den Löschvorgang entstanden ist, ist somit eine mögliche Variante zur Minimierung von $\Phi(., \mathsf{P}_n)$. B_n^* enthält $K-1$ oder weniger Elemente.

Betrachtet man also die Menge $B_n \nsubseteq K(5M)$, so erhält man für eine Teilfolge von Werten $\{n_i\}$ von n

$$
\begin{aligned}
m_{K-1}(\mathsf{P}) &\leq \liminf_i \Phi(B_{n_i}^*, \mathsf{P}_n) \ f.s. \\
&\leq \limsup_n \left[\Phi(B_n, \mathsf{P}_n) + \lambda \int_{\|x\| \geq 2M} \phi(\|x\|) \mathsf{P}_n(\,\mathrm{d}x) \right] \text{ wegen dem Zuwachs (4.6)} \\
&\leq \limsup_n \ \Phi(A, \mathsf{P}_n) + \lambda \int_{\|x\| \geq 2M} \phi(\|x\|) \mathsf{P}(\,\mathrm{d}x) \\
&\quad \text{für beliebige Menge } A \text{ mit } K \text{ oder weniger Elementen.}
\end{aligned}
$$

Wir setzen nun $A = \bar{A}(K)$ als optimale Menge von K Zentren für $\Phi(., \mathsf{P})$. Dann gilt für die Ungleichung, auch mittels (V7)

$$
\begin{aligned}
&\leq \Phi(\bar{A}(K), \mathsf{P}) + \varepsilon \\
&\quad \text{(nach Voraussetzung gilt } \Phi(\bar{A}(K), \mathsf{P}) = m_K(\mathsf{P})) \\
&= m_K(\mathsf{P}) + \varepsilon \\
&\overset{(4.4)}{<} m_{K-1}(\mathsf{P}) \qquad \lightning
\end{aligned}
$$

Dieser erhaltene Widerspruch macht somit deutlich, dass alle Elemente von B_n in einem kompakten Bereich liegen müssen. D. h. es genügt nach einem B_n (den optimalen Clusterzentren) inmitten der Klasse

$$\mathscr{E}_K := \{A \subseteq K(5M) : \ A \text{ enthält } K \text{ oder weniger Punkte}\}$$

zu suchen.

Nun kann also die Annahme getroffen werden, dass M so groß gewählt wird, dass $\bar{A}(K) \in \mathscr{E}_K$. Die Funktion $\Phi(., \mathsf{P})$ erreicht ihr Minimum bei $\bar{A}(K)$ aus $\mathscr{E}_K$.

Für jede Umgebung $\mathscr{N}$ von $\bar{A}(K)$ $\exists\ \eta > 0$ $(\eta = \eta(\mathscr{N}))$:

$$\Phi(A, \mathsf{P}) \geq \Phi(\bar{A}(K), \mathsf{P}) + \eta \quad ,\forall\ A \in \mathscr{E}_K\ \backslash \mathscr{N} \tag{4.7}$$

Es muss nun noch gezeigt werden, dass die Menge der optimalen Clusterzentren B_n innerhalb dieser Umgebung $\mathscr{N}$ liegt:

$$\Phi(B_n, \mathsf{P}) < \Phi(\bar{A}(K), \mathsf{P}) + \eta.$$

Mittels des gleichmäßigen Starken Gesetzes der Großen Zahlen (glm. SLLN)[19] lässt sich der Beweis beenden. Aus der Definition der optimalen Menge ist bekannt, dass

$$\Phi(B_n, \mathsf{P}_n) \ \leq\ \Phi(\bar{A}(K), \mathsf{P}_n).$$

Ebenfalls gilt:

$$\Phi(B_n, \mathsf{P}_n) - \Phi(B_n, \mathsf{P}) \to 0\ f.s. \text{ und}$$

$$\Phi(\bar{A}(K), \mathsf{P}_n) - \Phi(\bar{A}(K), \mathsf{P}) \to 0\ f.s.$$

$$\Rightarrow\ \Phi(B_n, \mathsf{P}_n) \to \Phi(B_n, \mathsf{P}) \leq \Phi(\bar{A}(K), \mathsf{P})$$

$$\Rightarrow\ \Phi(B_n, \mathsf{P}) < \Phi(\bar{A}(K), \mathsf{P}) + \eta$$

Somit lässt sich mittels der Umgebungseigenschaft (4.7) B_n in die Umgebung $\mathscr{N}$ von $\bar{A}(K)$ einordnen. Daraus lässt sich schließen, dass

$$B_n \to \bar{A}(K)\ f.s.$$

Ebenfalls wissen wir, dass

$$\Phi(B_n, \mathsf{P}_n) = m_K(\mathsf{P}_n) = \inf\{\Phi(A, \mathsf{P}_n) : A \in \mathscr{E}_K\}.$$

Es gilt nun

$$\inf\{\Phi(A, \mathsf{P}_n) : A \in \mathscr{E}_K\} \xrightarrow{f.s.} \inf\{\Phi(A, \mathsf{P}) : A \in \mathscr{E}_K\}, \text{ d. h.}$$

$$\Phi(B_n, \mathsf{P}_n) \xrightarrow{f.s.} m_K(\mathsf{P}).$$

[19] vgl. Anhang A.1.1; vgl. Pollard 1981

Aus der f.s. Konvergenz folgt die Konvergenz in Wahrscheinlichkeit und daraus wiederum die Konsistenz eines Schätzers. Der Schätzer B_n, für die optimalen Clusterzentren, ist konsistent.

□

4.2 Asymptotische Normalität

Die zweite statistische Eigenschaft, die betrachtet werden soll, ist die asymptotische Normalität der optimalen Clusterzentren.

Wir werden als Vorbereitung auf den eigentlichen Beweis der asymptotischen Normalität die quadratische Approximation

$$W_n(a) \approx W_n(\mu) - n^{-\frac{1}{2}} Z_n^T(a-\mu) + \frac{1}{2}(a-\mu)^T \Gamma (a-\mu) \tag{4.8}$$

beweisen, wobei a in der Umgebung eines festen Vektors μ liegt, Γ eine feste positiv definite Matrix und Z_n ein asymptotisch normalverteilter Zufallsvektor ist.

Es gelten zunächst folgende Voraussetzungen: Die Menge der optimalen Clusterzentren aus Kapitel 4.1 wird als Vektor geschrieben. Dieser Vektor b_n bestehend aus den Komponenten $(b_{n1}, ..., b_{nK})$ kann gewählt werden, um

$$W_n(a) = \frac{1}{n} \sum_{i=1}^{n} \min_{1 \le j \le K} \|x_i - a_j\|^2 \tag{4.9}$$

zu minimieren. Zu jedem Vektor $a = (a_1, ..., a_K) \in \mathbb{R}^{Kd}$ und für alle $x \in \mathbb{R}^d$ wird ebenfalls wieder eine Funktion ϕ definiert

$$\phi(x, a) = \min_{1 \le j \le K} \|x - a_j\|^2. \tag{4.10}$$

Die quadratische Approximation (4.8) beruht auf der Taylor-Entwicklung für quadratische Terme. Zunächst zerlegen wir $W_n(.)$ in zwei Terme. Aus dem empirischen Maß P_n und dem zugehörigen empirischen Prozess

$$\mathbf{X}_n(.) = n^{\frac{1}{2}}(\mathsf{P}_n(.) - \mathsf{P}(.))$$

folgt

$$\mathsf{P}_n = n^{-\frac{1}{2}} \mathbf{X}_n(.) + \mathsf{P}(.).$$

Damit ergibt sich

$$\begin{aligned} W_n(a) &= \frac{1}{n}\sum_{i=1}^{n} \min_{1\le j\le K} \|x_i - a_j\|^2 \\ &= \int \phi(.,a)\,\mathrm{d}\mathsf{P}_n \\ &= \int \phi(.,a)\,\mathrm{d}\mathsf{P} + n^{-\frac{1}{2}}\mathbf{X}_n\phi(.,a). \end{aligned} \tag{4.11}$$

Diese Darstellung von $W_n(a)$ bildet die Grundlage für die Entwicklung der quadratischen Approximation (4.8). Ausgehend von dieser Darstellung wird zunächst die erste und auch die zweite Ableitung von $\int \phi(.,a)\,\mathrm{d}\mathsf{P}$ ermittelt und eine Entwicklung für $\mathbf{X}_n\phi(.,a)$ an der Stelle μ gefunden.

Lemma 4.1 *Angenommen, $\int \|x\|^2\,\mathrm{d}\mathsf{P} < \infty$ und P stetig. Dann ist die Abbildung $a \to \phi(.,a)$ von $\mathbb{R}^{Kd}$ in $\mathscr{L}^2(\mathsf{P})$ differenzierbar im quadratischen Mittel. Demzufolge ist die Abbildung $a \to \mathsf{P}\phi(.,a)$ differenzierbar.*

Beweis: Für A_j, eine Umgebung um den Punkt a_j, gilt falls $x \in intA_j$, d. h. x liegt im Inneren von A_j, für A_j offen und h klein genug

$$\begin{aligned} \phi(x, a+h) &= \|x - a_j - h_j\|^2 \\ &= \|x - a_j\|^2 - 2(x-a_j)h_j + \|h_j\|^2 \\ &= \phi(x,a) - 2h_j^T(x-a_j) + \|h_j\|^2. \end{aligned}$$

Wir definieren $\Delta(x,a)$ als einen K Vektor von $\mathscr{L}^2$-Funktionen, mit den Komponenten $\Delta_j(x,a) = -2\mathbb{1}_{A_j}(x)(x-a_j)$. Die Funktion $\phi(.,a+h)$ kann für alle $x \in \mathbb{R}^d$ erweitert werden, da die Grenzen der A_j stetig sind

$$\phi(x,a+h) = \phi(x,a) + h^T\Delta(x,a) + \|h\|R(x,a,h) \tag{4.12}$$

$\forall h$ und $R(x,a,h) \to 0$ für P fast alle x und $h \to 0$.
Nun gilt

$$\begin{aligned} |R(x,a,h)| &\le \|h\|^{-1}\Big[\ \|h^T\Delta(x,a)\| + \max_j \big|\ \|x-a_j-h_j\|^2 - \|x-a_j\|^2\ \big|\ \Big] \\ &\le \underbrace{\|\Delta(x,a)\| + \|h\|^{-1}\sum_{j=1}^{K} \big|\ \|x-a_j-h_j\|^2 - \|x-a_j\|^2\ \big|}_{\star}. \end{aligned}$$

Wir definieren die „Hilfsfunktion“ $g(x) := \sum_{j=1}^{K} \|x - a_j\|^2$, die differenzierbar und quadratisch in x ist. Nun gilt für den Differenzenquotient, der $(\star)$ darstellt

$$\begin{aligned} \frac{g(x+h) - g(x)}{\|h\|} &= \frac{h^T g'(x)}{\|h\|} + \underbrace{\frac{1}{\|h\|}\, o(\|h\|)}_{\to\, 0} \\ &\qquad \text{für } g(x+h) = g(x) + h^T g'(x) + o(\|h\|) \\ &\leq \|g'(x)\|. \end{aligned}$$

Es kann nun $\|R(x, a, h)\|$ mittels dieser Erkenntnisse abgeschätzt werden

$$\|R(x, a, h)\| \leq C(1 + \|x\|) \ \in \mathscr{L}^2(\mathsf{P})$$

für h klein genug und einer beliebigen Konstante C. Mit (4.12) folgt, dass $\phi(., a)$ differenzierbar im quadratischen Mittel ist.

Aus (4.12) folgt ebenfalls, dass $\mathsf{P}\phi(., a)$ differenzierbar mit den Ableitungskomponenten

$$\begin{aligned} \frac{\partial}{\partial a_j} \int \phi(., a) \ \mathrm{d}\mathsf{P} &= -2 \int_{A_j} (\xi - a_j) \mathsf{P}(\,\mathrm{d}\xi) \\ &= \int \Delta_j(., a) \ \mathrm{d}\mathsf{P} \\ &=: \ \gamma_j(., a), \quad \text{wobei } \gamma_j \in \mathbb{R}^d \end{aligned}$$

ist.

□

Für den nächsten Schritt des Beweises der quadratischen Approximation benötigt man die zweite Ableitung von $\mathsf{P}\phi(., a)$.

Lemma 4.2 *Angenommen, $\int \|x\|^2 \,\mathrm{d}\mathsf{P} < \infty$ und P hat eine stetige Dichte f mit Hinsicht auf das d-dimensionale Lebesgue Maß λ. Es wird vorausgesetzt, dass das Integral*

$$\int_{F_{ij}} f(\xi)(\xi - m)(\xi - m)^T \ \sigma(\,\mathrm{d}\xi)$$

existiert und ununterbrochen von der Position der Zentren abhängt, für alle i, j und festes $m \in \mathbb{R}^d$. Hierbei ist F_{ij} die Schnittfläche der A_i's und $\sigma(.)$ das $(d-1)$-dimensionale Lebesgue Maß. Falls nun alle Clusterzentren a_i verschieden voneinander sind, hat die Abbildung $a \mapsto \mathsf{P}\phi(., a)$ eine zweite Abbildung Γ zusammengesetzt aus $(d \times d)$ Blöcken:

$$\Gamma_{ij} = \begin{cases} 2 \int\limits_{A_i} I_d \ \mathsf{P}(\,\mathrm{d}\xi) - 2 \sum\limits_{\alpha \neq i} r_{i\alpha}^{-1} \int\limits_{F_{i\alpha}} f(\xi)(\xi - a_i)(\xi - a_i)^T \ \sigma(\,\mathrm{d}\xi) & \textit{für } i = j \\ -2 r_{ij}^{-1} \int\limits_{F_{ij}} f(\xi)(\xi - a_i)(\xi - a_i)^T \ \sigma(\,\mathrm{d}\xi) & \textit{für } i \neq j \end{cases} \tag{4.13}$$

wobei $r_{ij} = \|a_i - a_j\|$.

Beweis: Die Komponenten der ersten Ableitung von $\int \phi(., a) \ \mathrm{d}\mathsf{P}$ sind bereits aus Lemma 4.1 bekannt und lauten:

$$\gamma_i(a) = \int \Delta_i(., a) \ \mathrm{d}\mathsf{P} = -2 \int\limits_{A_i} (\xi - a_i) \ \mathsf{P}(\,\mathrm{d}\xi), \quad i = 1, \ldots, K.$$

Die Differenzierbarkeit von $\gamma_i(.)$ bei beliebigem festem μ wird mittels

$$\frac{\partial}{\partial a_i} \int\limits_{A_i} (\xi - a_i) \ \mathsf{P}(xi) = - \int\limits_{A_i} I_d \ \mathsf{P}(\,\mathrm{d}\xi)$$

gegeben.

Es gilt

$$\int\limits_{A_i} (\xi - \mu_i) \ \mathsf{P}(\,\mathrm{d}\xi) = \int\limits_{A_i} f(\xi)(\xi - \mu_i) \ \lambda(\,\mathrm{d}\xi).$$

Dann ergibt sich mittels Anwendung des Integralsatzes von Stokes bei $a = \mu$

$$\begin{aligned} \mathrm{d} \int\limits_{A_i} (\xi - \mu_i) \ \mathsf{P}(\,\mathrm{d}\xi) &= \mathrm{d} \int\limits_{A_i} f(\xi)(\xi - \mu_i) \ \lambda(\,\mathrm{d}\xi) \\ &= \int\limits_{\partial M_i} f(\xi)(\xi - \mu_i) \ v_i^T \ \sigma(\,\mathrm{d}\xi) \ \mathrm{d}a, \end{aligned} \tag{4.14}$$

wobei σ das $(d-1)$-dimensionale Lebesgue Maß ist und v_i^T den Geschwindigkeitsvektor für die Bewegung von A_i orthogonal zu seinem Rand ∂A_i darstellt. Die Gestalt des Vektors $v_i^T \in \mathbb{R}^{Kd}$ hängt für den Fall $a = \mu$ von der Schnittfläche (Grenzfläche) F_{ij} ab, wobei F_{ij} als die Schnittfläche zwischen A_i und A_j definiert ist.

v_i^T stellt die Größe der Projektion der Grenzgeschwindigkeit in den Tangentialraum von A_i dar. Es findet eine Verschiebung der Menge ∂M_i statt und damit eine Änderung hin zu ∂A_i. Sämtliche Elemente aus ∂M_i werden mittels $v_i^T \ \mathrm{d}a$ folgendermaßen verschoben:

$$v_i^T \ \mathrm{d}a = -(x - \frac{1}{2}(\mu_i + \mu_j))^T \, \mathrm{d}n_{ij} + \frac{1}{2} n_{ij}^T (\, \mathrm{d}a_i + \, \mathrm{d}a_j), \tag{4.15}$$

wobei $n_{ij} = r_{ij}^{-1}(\mu_j - \mu_i) = \frac{(\mu_j - \mu_i)}{\|a_i - a_j\|}$ den Normalenvektor darstellt und $\frac{1}{2}(\, \mathrm{d}a_i + \, \mathrm{d}a_j)$ sich aus

$$(x - \frac{1}{2}(\mu_i + \mu_j)) - (x - \frac{1}{2}(\mu_i + \, \mathrm{d}a_i + \mu_j + \, \mathrm{d}a_j))$$

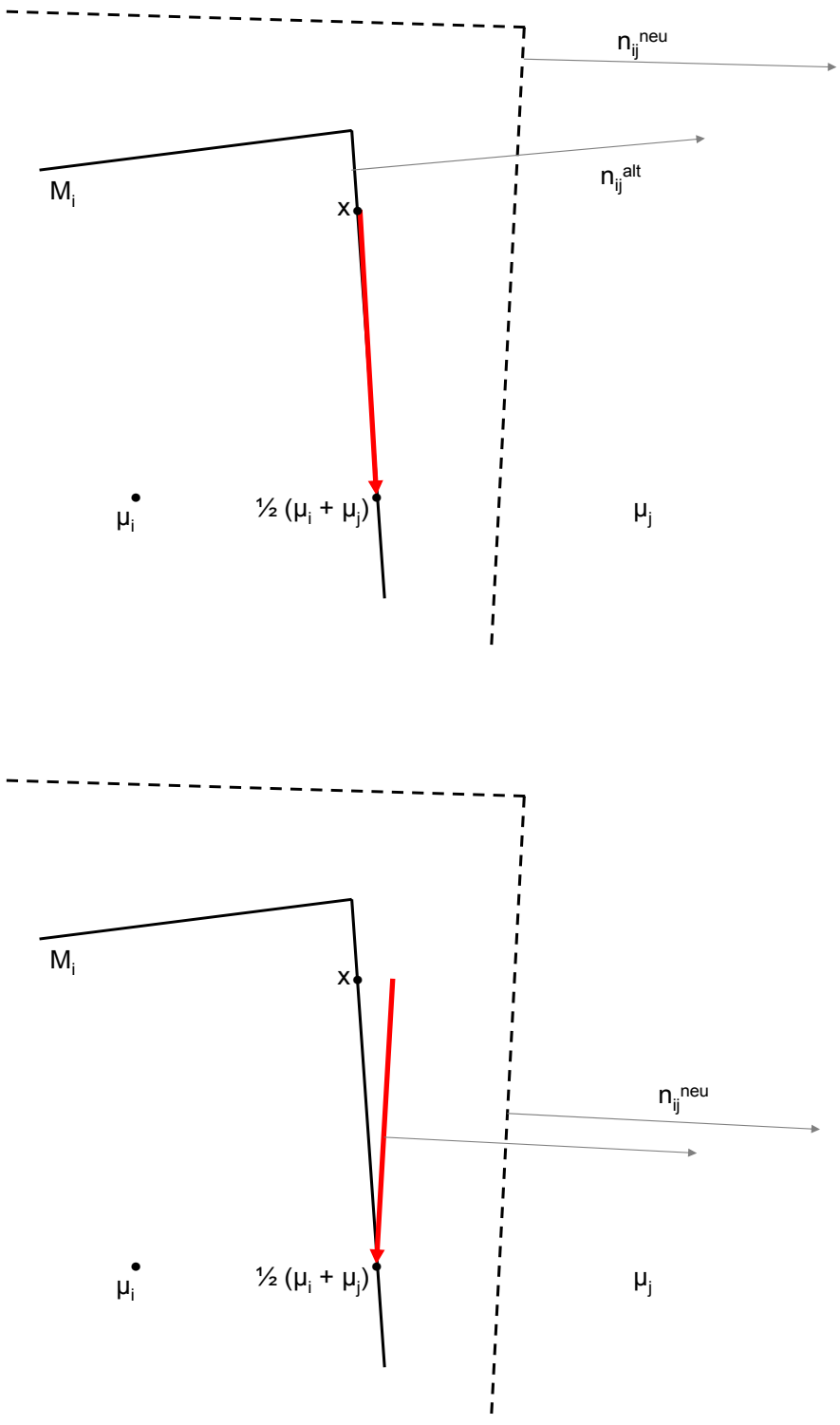
ergibt.

Im erste Term von (4.15) stellt $\mathrm{d}n_{ij}$ die Veränderung des Normalenvektors dar, welches rein formal ausgedrückt werden kann als

$$\mathrm{d}n_{ij} = r_{ij}^{-1} P_{ij}(\, \mathrm{d}a_j - \, \mathrm{d}a_i).$$

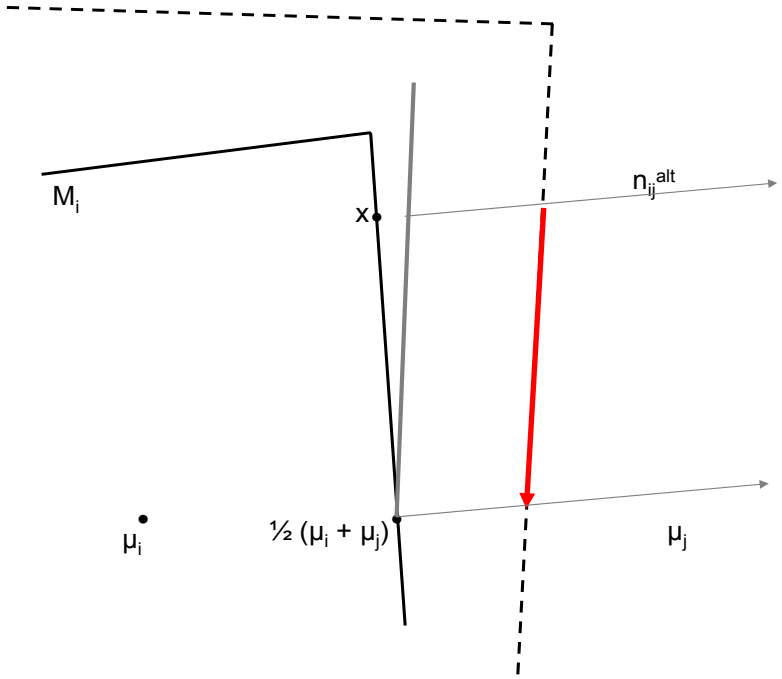
P_{ij} ist die Projektionsmatrix auf die affine Hülle von F_{ij}. $(\, \mathrm{d}a_j - \mathrm{d}a_i)$ ist der Richtungsänderungsvektor für die Verschiebung des „alten“ Normalenvektors in Richtung des „neuen“ Normalenvektors. Der erste Term sorgt somit für die Drehung des Vektor $-(x - \frac{1}{2}(\mu_i + \mu_j))^T$ in Richtung des neuen Normalenvektors. (vgl. Abb. 19)

Abbildung 19: Verlauf der Drehung



Der zweite Term von (4.15) stellt die endgültige Verschiebung des Vektors in Richtung des „alten“ Normalenvektors um den Verschiebungsvektor $\frac{1}{2}(\,\mathrm{d}a_i + \,\mathrm{d}a_j)$ dar. (Vgl. Abb. 20)

Abbildung 20: Verschiebung



Es erfolgt somit zuerst eine Rotation der Fläche und dann eine Translation.

Nun kann $v_i^T \;\mathrm{d}a$ mittels der Substitution durch $\mathrm{d}n_{ij} = r_{ij}^{-1} \prod_{ij}(\,\mathrm{d}a_j - \,\mathrm{d}a_i)$ umgeformt werden:

$$\begin{aligned}
v_i^T \;\mathrm{d}a &= -(x - \frac{1}{2}(\mu_i + \mu_j))^T \,\mathrm{d}n_{ij} + \frac{1}{2} n_{ij}^T(\,\mathrm{d}a_i + \,\mathrm{d}a_j) \\
&= -(x - \frac{1}{2}(\mu_i + \mu_j))^T r_{ij}^{-1}(\,\mathrm{d}a_j - \,\mathrm{d}a_i) + \frac{1}{2} r_{ij}^{-1}(\mu_j - \mu_i)^T(\,\mathrm{d}a_i + \,\mathrm{d}a_j) \\
&= -r_{ij}^{-1}\big[x^T(\,\mathrm{d}a_j - \,\mathrm{d}a_i) - \frac{1}{2}(\mu_i + \mu_j)^T(\,\mathrm{d}a_j - \,\mathrm{d}a_i) - \frac{1}{2}(\mu_j - \mu_i)^T(\,\mathrm{d}a_i + \,\mathrm{d}a_j)\big] \\
&= -r_{ij}^{-1}\big[x^T(\,\mathrm{d}a_j - \,\mathrm{d}a_i) + \mu_i \,\mathrm{d}a_i - \mu_j \,\mathrm{d}a_j\big] \\
&= -r_{ij}^{-1}\big[(x - \mu_j)^T \,\mathrm{d}a_j - (x - \mu_i)^T \,\mathrm{d}a_i\big] \\
&= -r_{ij}^{-1}(x - \mu_j)^T \,\mathrm{d}a_j + r_{ij}^{-1}(x - \mu_i)^T \,\mathrm{d}a_i.
\end{aligned}$$

Ausgehend von (4.14) erhält man durch Multiplikation

$$\begin{aligned} f(x)(x-\mu_i)v_i &= f(x)(x-\mu_i)\big(r_{ij}^{-1}(x-\mu_i)^T - r_{ij}^{-1}(x-\mu_j)^T\big) \\ &= r_{ij}^{-1}f(x)(x-\mu_i)(x-\mu_i)^T - r_{ij}^{-1}f(x)(x-\mu_i)(x-\mu_j)^T. \end{aligned}$$

Nun erfolgt die Bildung der Integrale über alle Mengen von ∂M_i

$$r_{ij}^{-1}\int\limits_{F_{ij}} f(\xi)(\xi-\mu_i)(\xi-\mu_i)^T\ \sigma(\,\mathrm{d}\xi) - r_{ij}^{-1}\int\limits_{F_{ij}} f(\xi)(\xi-\mu_i)(\xi-\mu_j)^T\ \sigma(\,\mathrm{d}\xi)$$

und deren Aufsummierung

$$\sum_{i\neq\zeta} r_{i\zeta}^{-1}\int\limits_{F_{i\zeta}} f(\xi)(\xi-\mu_i)(\xi-\mu_i)^T\ \sigma(\,\mathrm{d}\xi) - r_{ij}^{-1}\int\limits_{F_{ij}} f(\xi)(\xi-\mu_i)(\xi-\mu_j)^T\ \sigma(\,\mathrm{d}\xi),$$

wobei der erste Term für den Fall $i=j$ und der zweite Term für $i\neq j$ gilt.

Sämtliche Ergebnisse des Beweises führen dann zu der im Lemma gegebenen Darstellung der zweiten Ableitung der Abbildung $a\mapsto \mathsf{P}\phi(.,a)$

$$\Gamma_{ij} = \begin{cases} 2\int\limits_{A_i} I_d\ \mathsf{P}(d\xi) - 2\sum\limits_{\alpha\neq i} r_{i\alpha}^{-1}\int\limits_{F_{i\alpha}} f(\xi)(\xi-a_i)(\xi-a_i)^T\ \sigma(\,\mathrm{d}\xi) & \text{für } i=j \\ -2r_{ij}^{-1}\int\limits_{F_{ij}} f(\xi)(\xi-a_i)(\xi-a_i)^T\ \sigma(\,\mathrm{d}\xi) & \text{für } i\neq j \end{cases}$$

□

Im Lemma 4.1 wurde die $\mathscr{L}^2$ Differenzierbarkeit von $\phi(.,a)$ gezeigt. Des weiteren wird für die Entwicklung der quadratischen Approximation die stochastische Differenzierbarkeit von $\mathbf{X}_n\phi(.,a)$ benötigt. Um eine Umformung von der $\mathscr{L}^2$ Differenzierbarkeit in stochastische Differenzierbarkeit vornehmen zu können, benötigt man die Eigenschaften von Donsker Klassen von Funktionen.[20]

Lemma 4.3 *Sei $\{a_n\}$ eine Folge von Zufallsvektoren mit $\|a_n-\mu\| = o_{\mathsf{P}}(1)$, für feste Vektoren μ. Dann gilt unter den Bedingungen von Lemma* (4.1)

$$\mathbf{X}_n\phi(.,a_n) = \mathbf{X}_n\phi(.,\mu) + (a_n-\mu)^T\mathbf{X}_n\Delta(.,\mu) + o_{\mathsf{P}}(r_n), \tag{4.16}$$

wobei $r_n = \|a_n-\mu\|$.

[20] vgl. Anhang A.1.2

BEWEIS: Aus den Betrachtungen aus Lemma 4.1, besonders aus der Entwicklung (4.12), d. h.

$$\phi(x, a+h) = \phi(x, a) + h^T \Delta(x, a) + \|h\| R(x, a, h)$$

kann der Restbetrag $o_{\mathsf{P}}(r_n)$ aus (4.16) als

$$\|a_n - \mu\| \mathbf{X}_n R(., \mu, a_n - \mu)$$

geschrieben werden. Dann gilt

$$\phi(., \mu + (a_n - \mu)) = \phi(., \mu) + (a_n - \mu)^T \Delta(., \mu) + \|a_n - \mu\| R(., \mu, a_n - \mu).$$

(4.16) kann nach $\mathbf{X}_n R(., \mu, a_n - \mu)$ (siehe Beweis Lemma (4.1)) umgestellt werden.
Es gilt

$$R(., \mu, a - \mu) \to 0$$

in $\mathscr{L}^2$ für $a \to \mu$. Diese Tatsache ergibt sich aus der Konvergenz in $\mathscr{L}^2$ von $R(x, a, h)$ gegen 0 für $h \to 0$ aus Lemma 4.1.

Nach den Erklärungen der Donsker Klassen gilt dann, dass die Funktionen $R(., \mu, a - \mu)$ und 0 in $\mathscr{G}$ liegen und die Eigenschaften der Donsker Klasse erfüllen.
Somit lässt sich schlussfolgern, dass $\mathbf{X}_n \phi(., a)$ stochastisch differenzierbar ist.

□

Das folgende Lemma liefert unter Einbeziehung der bis jetzt erhaltenden Ergebnisse die Entwicklung der Approximation (4.8) und einen ersten Beweis der asymptotischen Normalität.

Lemma 4.4 *Es wird vorausgesetzt, dass $W(.)$ ein Minimum in μ besitzt. Sei nun $\{a_n\}$ eine beliebige Folge von Zufallsvektoren in $\mathbb{R}^{Kd}$ für die gilt $\|a_n - \mu\| = o_{\mathsf{P}}(1)$.*
Dann gilt unter den Voraussetzungen von Lemma 4.2

$$W_n(a_n) = W_n(\mu) - n^{-\frac{1}{2}} Z_n^T (a_n - \mu) + \frac{1}{2}(a_n - \mu)^T \Gamma (a_n - \mu) + o_{\mathsf{P}}(n^{-\frac{1}{2}} r_n) + o_{\mathsf{P}}(r_n^2) \quad (4.17)$$

wobei $r_n = \|a_n - \mu\|$. Wenn $\Delta(., \mu) \in \mathscr{G}$, dann hat Z_n eine asymptotische $\mathsf{N}(0, V)$-Verteilung mit V gegeben durch die Blöcke $V_i = 4 \int\limits_{M_i} (\xi - \mu_i)(\xi - \mu_i)^T \, \mathsf{P}(\, \mathrm{d}\xi)$.

BEWEIS: Aus den vorangegangenen Lemma 4.1 und 4.2 ergibt sich die Entwicklung (vgl. Taylor-Entwicklung)

$$\int \phi(.,a)\,\mathrm{d}\mathsf{P} = \int \phi(.,\mu)\,\mathrm{d}\mathsf{P} + (a-\mu)^T\gamma(\mu) + \frac{1}{2}(a-\mu)^T\Gamma(a-\mu) + o(\|a-\mu\|^2).$$

Da $a = \mu$ $W(.)$ minimiert, verschwindet der lineare Term, d. h. $\gamma(\mu) = 0$. Setzen wir nun $a = a_n$ erhält man die Entwicklung

$$\begin{aligned}\int \phi(.,a_n)\,\mathrm{d}\mathsf{P} &= \int \phi(.,\mu)\,\mathrm{d}\mathsf{P} + \frac{1}{2}(a_n-\mu)^T\Gamma(a_n-\mu) + o(\|a_n-\mu\|^2) \\ &= \int \phi(.,\mu)\,\mathrm{d}\mathsf{P} + \frac{1}{2}(a_n-\mu)^T\Gamma(a_n-\mu) + o_{\mathsf{P}}(r_n^2). \qquad (4.18)\end{aligned}$$

Betrachtet man nun die Entwicklung (4.16), d. h.

$$\mathbf{X}_n\phi(.,a_n) = \mathbf{X}_n\phi(.,\mu) + (a_n-\mu)^T\mathbf{X}_n\Delta(.,\mu) + o_{\mathsf{P}}(r_n)$$

und führt eine Substitution dieser Darstellung mit Formel (4.18) in (4.11)

$$W_n(a) = \int \phi(.,a)\,\mathrm{d}\mathsf{P} + n^{-\frac{1}{2}}\mathbf{X}_n\phi(.,a)$$

durch, folgt für $a = a_n$

$$\begin{aligned}W_n(a_n) &= \int \phi(.,\mu)\,\mathrm{d}\mathsf{P} + \frac{1}{2}(a_n-\mu)^T\Gamma(a_n-\mu) + o_{\mathsf{P}}(r_n^2) \\ &\quad + n^{-\frac{1}{2}}\mathbf{X}_n\phi(.,\mu) + n^{-\frac{1}{2}}(a_n-\mu)^T\mathbf{X}_n\Delta(.,\mu) + n^{-\frac{1}{2}}o_{\mathsf{P}}(r_n).\end{aligned}$$

Durch die Umgruppierung der erhaltenen Terme ergibt sich

$$\begin{aligned}W_n(a_n) &= \overbrace{\int \phi(.,\mu)\,\mathrm{d}\mathsf{P} + n^{-\frac{1}{2}}\mathbf{X}_n\phi(.,\mu)}^{W_n(\mu)} + n^{-\frac{1}{2}}(a_n-\mu)^T\mathbf{X}_n\Delta(.,\mu) + o_{\mathsf{P}}(n^{-\frac{1}{2}}r_n) \\ &\quad + \frac{1}{2}(a_n-\mu)^T\Gamma(a_n-\mu) + o_{\mathsf{P}}(r_n^2) \\ &\overset{\text{setze } Z_n = -\mathbf{X}_n\Delta(.,\mu)}{=} W_n(\mu) - n^{-\frac{1}{2}}(a_n-\mu)^TZ_n + o_{\mathsf{P}}(n^{-\frac{1}{2}}r_n) \\ &\quad + \frac{1}{2}(a_n-\mu)^T\Gamma(a_n-\mu) + o_{\mathsf{P}}(r_n^2) \\ &\approx W_n(\mu) + n^{-\frac{1}{2}}Z_n^T(a_n-\mu) + \frac{1}{2}(a_n-\mu)^T\Gamma(a_n-\mu).\end{aligned}$$

Z_n besitzt den Erwartungswert

$$\begin{aligned}\mathsf{E}(-\mathbf{X}_n\Delta(.,\mu)) &\overset{\mathbf{X}_n \text{ emp.Prozess}}{=} -\int \Delta(.,\mu)\,\mathrm{d}\mathsf{P} \\ &= -\gamma(\mu) = 0.\end{aligned}$$

Als Varianzmatrix erhält man

$$\begin{aligned} \mathsf{Var}\,(Z_n) &= \mathsf{E}(Z_n^2) + \underbrace{(\mathsf{E}(Z_n))^2}_{=0} \\ &= \int \Delta(.,\mu)\ \Delta(.,\mu)^T \,\mathrm{d}\mathsf{P}, \end{aligned}$$

wobei die (i,j)te Stelle dargestellt wird durch

$$4 \int\limits_{M_i \cap M_j} (\xi - \mu_i)(\xi - \mu_j)^T \ \mathsf{P}(\,\mathrm{d}\xi),$$

welches für $i \neq j$ verschwindet, da $M_i \cap M_j = \emptyset$. Es gilt für $i = j$

$$V_i = 4 \int\limits_{M_i} (\xi - \mu_i)(\xi - \mu_i)^T \ \mathsf{P}(\,\mathrm{d}\xi), \tag{4.19}$$

wobei $i = 1, ..., K$.
V ist daher eine $(Kd \times Kd)$-dimensionale Diagonalmatrix. Aufgrund der Voraussetzung, dass $\Delta(.,\mu) \in \mathscr{G}$ lässt sich auf Z_n der ZGWS anwenden. Daher ergibt sich die Konvergenz in Verteilung gegen die Normalverteilung. Es gilt also

$$Z_n \xrightarrow{\mathscr{L}} \mathsf{N}(0, V).$$

Somit ist Z_n asymptotisch normalverteilt.

□

Bemerkung 1 *D. Pollard hat in seiner Veröffentlichung das Lemma 4.4 ohne die zusätzliche Voraussetzung $\Delta(.,\mu) \in \mathscr{G}$ bewiesen. Dieser Beweis konnte ohne diesen Zusatz nicht nachvollzogen werden.*

Werden nun alle erhaltenen Erkenntnisse aus Lemma 4.1 bis Lemma 4.3 und speziell aus dem Lemma 4.4 zusammengefasst, ergibt sich die gewünschte Approximation aus (4.8)

$$W_n(a) \approx W_n(\mu) - n^{-\frac{1}{2}} Z_n^T(a - \mu) + \frac{1}{2}(a - \mu)^T \Gamma (a - \mu).$$

Diese Approximation bildet die Grundlage für die folgende Vermutung, dass $\sqrt{n}(b_n - \mu)$ asymptotisch normalverteilt ist mit $\mathsf{N}(0, \Gamma^{-1} V \Gamma^{-1})$, wobei b_n der Vektor der optimalen Clusterzentren ist. Diese Vermutung wird im folgenden Theorem bewiesen.

Theorem 2 *Sei b_n der Vektor der optimalen Clusterzentren für unabhängige Stichproben einer Verteilung P aus $\mathbb{R}^d$. Es wird vorausgesetzt:*

(i) der Vektor μ ist eindeutig (bis auf Umbenennung seiner Koordinaten)

(ii) $\mathsf{P}\|x\|^2 < \infty$ (Voraussetzung in Lemma 4.1 und Lemma 4.2)

(iii) das Wahrscheinlichkeitsmaß P hat eine stetige Dichte f in Hinblick auf das Lebesgue-Maß λ in $\mathbb{R}^d$ (siehe Lemma 4.2)

(iv) das Integral

$$\sigma[\mathbb{1}_{F_{ij}}(x)\ f(x)(x-m)(x-m)^T]$$

hängt stetig von der Lage der Clusterzentren ab

(v) die Matrix Γ (definiert durch Auswerten von (4.13) bei $a = \mu$) ist positiv definit

Dann gilt $\sqrt{n}(b_n - \mu) \xrightarrow{\mathscr{L}} \mathsf{N}(0, \Gamma^{-1}V\Gamma^{-1})$, wobei V die $(Kd \times Kd)$-dimensionale Diagonalmatrix aus Lemma 4.4 ist.

Beweis: Aus den beiden ersten Bedingungen lässt sich, in Hinblick auf Theorem 1 (und Lemma 4.1/4.2) schließen, dass der Vektor der optimalen Clusterzentren gegen den Vektor der $W(.)$ minimiert, konvergiert. D.h.

$$b_n \xrightarrow{f.s.} \mu.$$

Aus der f.s.-Konvergenz folgt die Konvergenz in Wahrscheinlichkeit. Daher gilt schließlich:

$$b_n \xrightarrow{\mathsf{P}} \mu,$$

d.h. $\forall \varepsilon > 0 : \lim_{n\to\infty} P(|b_n - \mu| > \varepsilon) = 0$.

Es gilt

$$W_n(b_n) \leq W_n(\mu),$$

erklärbar durch die Definition von b_n. Setzen wir $\lambda_n = \|b_n - \mu\|$, dann ergibt sich unter Verwendung von Lemma 4.4 (speziell Darstellung (4.17)) und $a_n = b_n$:

$$-n^{-\frac{1}{2}} Z_n^T(b_n - \mu) + \frac{1}{2}(b_n - \mu)^T\Gamma(b_n - \mu) + o_{\mathsf{P}}(n^{-\frac{1}{2}}\lambda_n) + o_{\mathsf{P}}(\lambda_n^2) \leq 0. \tag{4.20}$$

Z_n ist der Ordnung $O_\mathsf{P}(1)$, da Z_n in Verteilung konvergiert und damit begrenzt auf eine Konstante in Wahrscheinlichkeit konvergiert bzw. in Wahrscheinlichkeit begrenzt ist.[21] Es gilt dann

$$\begin{aligned} n^{-\frac{1}{2}} Z_n^T (b_n - \mu) &= n^{-\frac{1}{2}} \lambda_n \, O_\mathsf{P}(1) \\ &= O_\mathsf{P}(n^{-\frac{1}{2}} \lambda_n). \ ^{22} \end{aligned}$$

Aus der Bedingung *(v)*, also aus der positiven Definitheit der Matrix Γ lässt sich eine Konstante τ finden, so dass

$$y^T \Gamma y \geq \tau \|y\|^2 \quad \forall y.$$

Damit ergibt sich folgende Abschätzung:

$$\begin{aligned} \frac{1}{2}(b_n - \mu)^T \Gamma (b_n - \mu) &\geq \tau \|b_n - \mu\|^2 \\ &= \tau \lambda_n^2. \end{aligned}$$

Aus der Tatsache (4.20) erhält man die Umformung

$$\begin{aligned} \tau \lambda_n^2 &\leq O_\mathsf{P}(n^{-\frac{1}{2}} \lambda_n) + o_\mathsf{P}(n^{-\frac{1}{2}} \lambda_n) + o_\mathsf{P}(\lambda_n^2) \\ &\leq O_\mathsf{P}(n^{-\frac{1}{2}} \lambda_n) + o_\mathsf{P}(\lambda_n^2) \ ^{23} \\ \Rightarrow \ \tau \lambda_n &\leq O_\mathsf{P}(n^{-\frac{1}{2}}) + o_\mathsf{P}(\lambda_n) \end{aligned}$$

$$\begin{aligned} &\Rightarrow \quad \tau n^{-\frac{1}{2}} \lambda_n \leq O_\mathsf{P}(1) + o_\mathsf{P}(n^{\frac{1}{2}} \lambda_n) \\ &\Rightarrow \quad \lambda_n = O_\mathsf{P}(n^{-\frac{1}{2}}). \end{aligned}$$

Setzen wir $\theta := n^{\frac{1}{2}}(b_n - \mu)$. Durch Einstetzen in (4.17) ergibt sich:

$$\begin{aligned} W_n(b_n) &= W_n(\mu) - n^{-\frac{1}{2}} Z_n^T (b_n - \mu) + \frac{1}{2}(b_n - \mu)^T \Gamma (b_n - \mu) + o_p(n^{-\frac{1}{2}} \lambda_n) + o_p(\lambda_n^2) \\ &= W_n(\mu) - n^{-1} Z_n^T \theta + \frac{1}{2} n^{-1} \theta^T \Gamma \theta + o_p(n^{-1}) \\ &= ^{24} \ W_n(\mu) - \frac{1}{2} n^{-1} \|\Gamma^{\frac{1}{2}} \theta_n - \Gamma^{-\frac{1}{2}} Z_n\|^2 - \frac{1}{2} n^{-1} Z_n^T \Gamma^{-1} Z_n + o_p(n^{-1}). \end{aligned} \tag{4.21}$$

[21] allg.: $Y_n \xrightarrow{\mathscr{L}} a \quad \Rightarrow \quad Y_n \xrightarrow{\mathsf{P}} a$, für $a \in \mathbb{R}$

[22] allg. gilt: $R_n O_\mathsf{P}(1) = O_\mathsf{P}(R_n)$, wobei R_n eine Folge von Zufallsvariablen ist

[23] da $O_\mathsf{P}(n^{-\frac{1}{2}} \lambda_n) + o_\mathsf{P}(n^{-\frac{1}{2}} \lambda_n) = O_\mathsf{P}(n^{-\frac{1}{2}} \lambda_n)$

[24] Nebenrechnung:

$$\begin{aligned} &\frac{1}{2} n^{-1} \|\Gamma^{\frac{1}{2}} \theta_n - \Gamma^{-\frac{1}{2}} Z_n\|^2 = \frac{1}{2} n^{-1} \theta_n^T \Gamma \theta_n - n^{-1} Z_n^T \theta_n + \frac{1}{2} n^{-1} Z_n^T \Gamma^{-1} Z_n \\ \Rightarrow \quad &-n^{-1} Z_n^T \theta_n + \frac{1}{2} n^{-1} \theta_n^T \Gamma \theta_n = \frac{1}{2} n^{-1} \|\Gamma^{\frac{1}{2}} \theta_n - \Gamma^{-\frac{1}{2}} Z_n\|^2 - \frac{1}{2} n^{-1} Z_n^T \Gamma^{-1} Z_n \end{aligned}$$

Setzen wir $a_n = \mu + n^{-\frac{1}{2}}\Gamma^{-1}Z_n$ in (4.17):

$$\begin{aligned} W_n(\mu + n^{-\frac{1}{2}}\Gamma^{-1}Z_n) &= W_n(\mu) - n^{-\frac{1}{2}}Z_n^T(n^{-\frac{1}{2}}\Gamma^{-1}Z_n) + \frac{1}{2}(n^{-\frac{1}{2}}\Gamma^{-1}Z_n)^T\Gamma(n^{-\frac{1}{2}}\Gamma^{-1}Z_n) \\ &\quad +o_p(n^{-1}) \\ &= W_n(\mu) - n^{-1}Z_n^T\Gamma^{-1}Z_n + \frac{1}{2}n^{-1}Z_n^T\Gamma^{-1}Z_n + o_p(n^{-1}) \\ &= W_n(\mu) - \frac{1}{2}n^{-1}Z_n^T\Gamma^{-1}Z_n + o_p(n^{-1}). \end{aligned}$$

Unter Berücksichtigung von (4.21) ergibt sich

$$W_n(b_n) = W_n(\mu + n^{-\frac{1}{2}}\Gamma^{-1}Z_n) + \frac{1}{2}n^{-1}\|\Gamma^{\frac{1}{2}}\theta_n - \Gamma^{-\frac{1}{2}}Z_n\|^2 + o_p(n^{-1}).$$

Laut Definition von b_n, der $W_n(.)$ minimiert, gilt

$$W_n(b_n) \leq W_n(\mu + n^{-\frac{1}{2}}\Gamma^{-1}Z_n).$$

Es folgt

$$\frac{1}{2}n^{-1}\|\Gamma^{\frac{1}{2}}\theta_n - \Gamma^{-\frac{1}{2}}Z_n\|^2 + o_p(n^{-1}) \leq 0.$$

Es gilt also

$$\begin{aligned} &\frac{1}{2}n^{-1}\|\Gamma^{\frac{1}{2}}\theta_n - \Gamma^{-\frac{1}{2}}Z_n\|^2 = o_p(n^{-1}) \\ &\Leftrightarrow \quad \theta_n = o_p(1) + \Gamma^{-1}Z_n \\ &\Rightarrow \quad \theta_n - \Gamma^{-1}Z_n = o_p(1) \\ &\Rightarrow \quad n^{\frac{1}{2}}(b_n - \mu) - \Gamma^{-1}Z_n \xrightarrow{\mathsf{P}} 0 \text{ [25]} \\ &\Rightarrow \quad n^{\frac{1}{2}}(b_n - \mu) \xrightarrow{\mathscr{L}} \mathsf{N}(0, \Gamma^{-1}V\Gamma^{-1}) \end{aligned}$$

□

Die Kovarianzmatrix $\Gamma^{-1}V\Gamma^{-1}$ besteht aus der Multiplikation zwischen der Inversen der Blockmatrix Γ mit der Diagonalblockmatrix V und der Inversen von Γ. Die Kovarianzmatrix ist eine $(Kd \times Kd)$-dimensionale Blockmatrix. Die Komponenten von $n^{\frac{1}{2}}(b_n - \mu)$ sind voneinander abhängig.[26]

[25] Die beiden Komponenten besitzen dieselbe Grenzverteilung.

[26] vgl. Anhang A.1.3

5 Fazit

Diese Arbeit beschäftigt sich mit statistischen Eigenschaften von Clusterverfahren. Zunächst wurde ein Überblick über die Vorgehensweise der Clusteranalyse gegeben. Sowohl hierarchische als auch nichthierarchische Clusteranalyseverfahren wurden in Verbindung mit Clusteranalysemethoden vorgestellt.

Die nähere Betrachtung der Clusteranalysemethoden hat gezeigt, dass es verschiedene Methoden zu den unterschiedlichen Fragestellungen gibt. Für den Abstand zwischen Mannigfaltigkeiten eignet sich zur Realisierung ein modifiziertes 1-nearest neighbour Verfahren. Anhand der gegebenen Beispiele lässt sich erkennen, dass das Auffinden der gewünschten Clusterstruktur mittels mod. 1-nearest neighbour Verfahrens abhängig von der Anzahl der gegebenen Beobachtungen ist.

Die K-means Methode, realisiert durch das K-means Verfahren, ist eine weitere Möglichkeit, Clusterstrukturen bezogen auf das gewünschte Untersuchungsziel zu erkennen. Es werden konvexe Strukturen gefunden, die sich anhand der ständigen Betrachtung und Berücksichtigung der Mittelwerte der Cluster ergeben. Bei diesem Verfahren besteht eine Abhängigkeit bezüglich der Wahl der Startpartitionen. Unterschiedliche Ausgangsmengen können zu unterschiedlichen Endclustern führen, d. h. es werden nur lokale Optima gefunden. Zur Beseitigung dieser Unzulänglichkeiten ist es daher ratsam, das K-means Verfahren für unterschiedliche Startpartitionen durchzuführen und die „beste“ Endclusterung im Sinne des Untersuchungsziels zu wählen.

Die Arbeit macht für das K-means Verfahren ebenfalls Aussagen bezüglich statistischer Eigenschaften. Es wurde gezeigt, dass die Menge der optimalen Clusterzentren konsistent ist. Sie konvergiert in Wahrscheinlichkeit bei einer Vergößerung des Stichprobenumfangs gegen die Menge der Clusterzentren, die das Varianzkriterium, welches der K-means Methode zugrunde liegt, minimiert. Eine weitere asymptotische Eigenschaft bezüglich der K-means Methode ist die asymptotische Normalität. Es wird mittels einer Abschätzung für das „Varianzkriterium“ gezeigt, dass der Vektor der optimalen Clusterzentren für $n \to \infty$ gegen eine Normalverteilung konvergiert. Hierbei hat sich ergeben, dass die Einzelkomponenten von $\sqrt{n}(b_n - \mu)$ bzw. b_n voneinander abhängen[27], was wiederum gut interpretierbar ist.

Die asymptotische Normalität lässt auf weiterführende statistische Eigenschaften schließen. Es ist daher möglich Aussagen über die Konfidenzintervalle der optimalen Clusterzentren b_n zu treffen.

Die Betrachtung der statistischen Eigenschaften bezieht sich in dieser Arbeit ausschließlich auf die K-means Methode. Inwieweit sich in dieser Studie erarbeiteten Ergebnisse bezüglich des asymptotischen Verhaltens auf andere Clustermethoden (Abstand zwischen Mannigfaltigkeiten, Abstandsmaß bzgl. Konturen) übertragen lassen, müssen weitere Studien zeigen.

[27] vgl. Anhang A.1.3

Literatur

[1] ***Bacher, Dr. J.;*** Clusteranalyse - Anwendungsorientierte Einführung, 2. Auflage 2002, R. Oldenbourg Verlag München Wien

[2] ***Backhaus, K.; Erichson, B.; Plinke, W.; Weiber, R.;*** Multivariate Analysemethoden, 6. Auflage 1990, Springer-Verlag Berlin Heidelberg

[3] ***Baddeley, A.;*** Integrals on a moving Manifold and geometrical Probability, 1977, Advances in Applied Probability 9 (588-603)

[4] ***Bock, H. H.;*** Automatische Klassifikation, 1974, Vandenhoeck & Ruprecht Göttingen

[5] ***Bronstein I. N.; Semendjajew K. A.; Musiol G.; Mühlig H.;*** Taschenbuch der Mathematik, 5. Auflage 2001, Verlag Harri Deutsch

[6] ***Cunningham, P.; Delany, S. J.;*** k-Nearest Neighbour Classifiers, Technical Report University College Dublin, School of Computer Science and Informatics

[7] ***Deichsel, G.; Trampisch, H. J.;*** Clusteranalyse und Diskriminanzanalyse, 1985, Gustav Fischer Verlag · Stuttgart (Biometrie)

[8] ***Everitt, B. S.;*** Cluster Analysis, 1993 (third edition), Hodder & Stoughton London

[9] ***Forster, O.;*** Analysis 2, 5. Auflage 1999, vieweg Verlag

[10] ***Gaenssler, P.; Stute, W.;*** Empirical Processes: A survey of results for independent and identically distributed random variables, 1979, The Annals of Probability Vol. 7, No. 2, 193-243

[11] ***Gordon, A. D.;*** Classification - 2nd Edition, 1999, Monographs on Statistics and Applied Probability, 82

[12] ***Hand, D. J.;*** Discrimination and Classification, 1992, Wiley Series in Probability and Mathematical Statistics

[13] ***Hartigan, J. A.;*** Clustering Algorithms, 1975, Wiley Series in Probability and Mathematical Statistics

[14] ***Hastie, T.; Tibshirani, R.; Friedman, J.;*** The Elements of Statistical Learning, 2001, Springer-Verlag, Springer Series in Statistics

[15] ***Jahnke, H.;*** Clusteranalyse als Verfahren der schließenden Statistik, 1988, Göttingen Vandenhoeck & Ruprecht

[16] ***Jain, A. K.; Dubes, R. C.;*** Algorithms for Clustering Data, 1988, Prentice Hall Advanced Reference Series

[17] ***Jajuga, K.; Sokolowski, A.; Bock, H.-H.;*** Classification, Clustering, and Data Analysis: Recent Advances and Applications, 2002, Springer-Verlag Berlin Heidelberg

[18] ***Kaufmann, H.; Pape, H.;*** Clusteranalyse, in: Fahrmeir, L.: Multivariate statistische Verfahren. 2. Auflage 1996, Walter de Gruyter

[19] ***Läuter, H.; Pincus, R.;*** Mathematisch-Statistische Datenanalyse, 1989, Akademie-Verlag Berlin

[20] ***MacQueen, J.;*** Some methods for classification and analysis of multivariate observations, 1967, Proc. Fifth Berkeley Symp. on Math. Statist. and Prob., Vol. 1, 281-297 (Univ. of Calif. Press)

[21] ***Marinell, G.;*** Multivariate Verfahren: Einführung für Studierende und Praktiker, 1990 (3. erw. Auflage), Wien : Oldenbourg

[22] ***Mucha, H.-J.;*** Clusteranalyse mit Mikrocomputern, 1992, Akademischer Verlag

[23] ***Mucha, H.-J.; Sofyan, H.;*** Cluster Analysis, 2000, Humboldt Universität Berlin, Sonderforschungsbericht 373, 2000-49

[24] ***Pollard, D.;*** Strong Consistency of K-means Clustering, 1981, The Annals of Statistics Vol. 9, No. 1, 135-140

[25] ***Pollard, D.;*** A Central Limit Theorem for the K-means Clustering, 1982, The Annals of Probability Vol. 10, No. 4, 919-926

[26] ***Pollard, D.;*** Convergence of Stochastic Processes, 1984, Springer-Verlag New York, Springer Series in Statistics

[27] ***Pollard, D.;*** Empirical Processes: Theory and Applications, 1990, NSF-CBMS Regional Conference Series in Probalbility and Statistics Vol. 2

[28] ***Pruscha, H.;*** Statistisches Methodenbuch (- Verfahren, Fallstudien, Programmcodes), 2006, Springer-Verlag Berlin Heidelberg

[29] ***Van Ryzin, J.;*** Classification and Clustering, 1977, Academic Press

[30] ***Sodeur, W.;*** Empirische Verfahren zur Klassifikation, 1974, B. G. Teubner (Studienskripten), Stuttgart

[31] ***Späth, H.;*** Cluster-Analyse-Algorithmen (zur Objektklassifizierung und Datenreduktion), 1975, R. Oldenbourg Verlag GmbH, München

[32] ***Stahl, H.;*** Clusteranalyse großer Objektmengen mit problemorientierten Distanzmaßen, 1985, Verlag Harri Deutsch · Thun · Frankfurt am Main

[33] ***Steinhausen, D.; Langer, K.;*** Clusteranalyse - Einführung in Methoden und Verfahren der automatischen Klassifikation, 1. Auflage 1977, Walter de Gruyter

[34] ***Zorich, V. A.;*** Mathematical Analysis II, 2004, Springer-Verlag Berlin Heidelberg

A Anhang

A.1 Ergänzungen

A.1.1 gleichmäßiges SLLN

Das gleichmäßige Gesetz der Großen Zahlen beruht hier auf P-integrierbaren Funktionen. Die Klasse $\mathscr{T}$ enthält P-integrierbaren Funktionen in $\mathbb{R}^d$ folgender Gestalt:

$$t_A(x) := \min_{a \in A} \phi(\|x - a\|),$$

wobei $A \in \mathscr{E}_K$. Für diese Funktionen gilt das gleichmäßige Gesetz der Großen Zahlen:

$$\sup_{t \in \mathscr{T}} \Big| \int t \, \mathrm{d}\mathsf{P}_n - \int t \, \mathrm{d}\mathsf{P} \Big| \longrightarrow \; 0 \; f.s.$$

Es gilt somit

$$\sup_{A \in \mathscr{E}_K} |\Phi(A, \mathsf{P}_n) - \Phi(A, \mathsf{P})| \longrightarrow \; 0 \; f.s.$$

bzw.

$$\sup_{A \in \mathscr{E}_K} \Big| \int \min_{a \in A} \phi(\|\xi - a\|) \mathsf{P}_n(\, \mathrm{d}\xi) - \int \min_{a \in A} \phi(\|\xi - a\|) \mathsf{P}(\, \mathrm{d}\xi) \Big| \longrightarrow \; 0 \; f.s.$$

Der Beweis hierfür wird ausführlich in Pollards „Strong Consistency of K-means Clustering“ besprochen mit Verweis auf die Arbeit von Gaenssler und Stute[28].

A.1.2 Donsker Klasse

Erfüllt $\mathbf{X}_n\phi$ die Eigenschaft von Donsker Klassen, lässt sich daraus folgern, dass $\mathbf{X}_n\phi$ stochastisch differenzierbar ist. Falls $\mathscr{G} \subseteq \mathscr{L}^2$ gilt, wird der empirische Prozess $\mathbf{X}_n$ als stochastischer Prozess indiziert mit $\mathscr{G}$ aufgefasst. Die Klasse $\mathscr{G}$ wird dann Donsker-Klasse für P genannt, falls der funktionale ZGWS über einer Folge von Prozessen $\{\mathbf{X}_n(g) : g \in \mathscr{G}\}$ betrachtet wird. Dabei gilt folgende Schlüsseleigenschaft für Donsker-Klassen:

$$\forall \varepsilon > 0, \; \eta > 0 \; \exists \; \delta > 0 \text{ und ein } n_0 : \; \forall n \geq n_0$$

$$\mathbb{P}\big(\sup_{[\delta]} |\mathbf{X}_n(g_1) - \mathbf{X}_n(g_2)| > \eta \big) < \varepsilon,$$

[28] vgl. Gaenssler, P.; Stute, W.

wobei das Supremum über alle Funktionspaare $g_1, g_2 \in \mathscr{G}$ läuft.

Jede Funktion aus $\mathscr{G}$ kann als Summe von K^2 Elementen der Klasse $\mathscr{F}$ geschrieben werden. $\mathscr{F}$ liegen folgende Eigenschaften zugrunde:

(i) $|f(x)| \leq [C(1 + \|x\|) = F(x)] \quad \forall x$

(ii) $f = L \cdot \mathbb{1}_Q$, wobei L eine linieare Funktion und $\mathbb{1}_Q$ ein konvexer Bereich ist, der als Schnittfläche von (höchstens $2K$) offenen oder abgeschlossenen Halbräumen dargestellt werden kann.

Diese Tatsache soll nun angewendet werden auf unseren Sachverhalt. Sei $\mathscr{G}$ die Klasse aller Funktionen $R(., \mu, a - \mu)$. a liegt in einer Umgebung eines festen μ's, so dass

$$|R(x, \mu, a - \mu)| \leq C(1 + \|x\|)$$

$\forall a$ und $\forall x$ und beliebige Konstante C gilt.

Die Funktion $R(x, \mu, a-\mu)$ lässt sich schreiben als (im Vergleich zu (4.12) ist hier $h = a-\mu$)

$$\begin{aligned} R(x, \mu, a - \mu) &= \|a - \mu\|^{-1}[\phi(x, a) - \phi(x, \mu) - (a - \mu)^T \Delta(x, \mu)] \\ &= \sum_{i,j} \mathbb{1}_{M_i \cap A_j} \|a - \mu\|^{-1}[\|x - a_j\|^2 - \|x - \mu_i\|^2 + 2(a_i - \mu_i)^T (x - \mu_i)] \\ &= \sum_{i,j} \underbrace{\underbrace{\mathbb{1}_{M_i \cap A_j}}_{*} \underbrace{\|a - \mu\|^{-1}[2(a_i - a_j)^T x + \|\mu_i\| - 2\mu_i^T a_i \|a_j\|^2]}_{**}}_{=f}, \ i, j = 1, ..., k, \end{aligned}$$

wobei $A_1, ..., A_K$ konvexe Polyeder mit den dazugehörigen Zentren $a_1, ..., a_K$ sind und $M_1, ..., M_K$ Polyeder assoziert mit $\mu_1, ..., \mu_K$.

Ein konvexer Polyeder ist ein Schnitt von Halbräumen und der Schnitt eines konvexen Polyeders mit einem Polyeder ist damit wiederum konvex. Daher ergibt sich, dass $(*)$ eine Schnittfläche von höchstens $2K$, da $i, j = 1, ..., K$, (offenen oder abgeschlossenen) Halbräumen ist. Zum anderen ist $(**)$ eine lineare Funktion in x. Durch diese beiden Aspekte wird die zweite Eigenschaft der Klasse $\mathscr{F}$ erfüllt.

Durch Umformungen von $|f|$ erhält man die Abschätzung $C(\|x\| + 1)$, welches die erste Eigenschaft von $\mathscr{F}$ erfüllt. Daher gilt nun $f \in \mathscr{F}$ und R ist Summe von K^2 Funktionen

aus $\mathscr{F}$.

Der endgültige Beweis für die Gültigkeit, dass $\mathscr{G}$ eine Donsker Klasse ist, wird in den Schriften von Pollard [29] und LeCam[30] ausgeführt.

A.1.3 Kovarianzmatrix

Die Kovarianzmatrix ist eine $(Kd \times Kd)$-dimensionale Blockmatrix, bei der keine der enthaltenen Blöcke gleich Null ist. Für den vereinfachten Fall $d = 1$ ergibt sich:

Für $K = 2$, gilt

$$\Gamma^{-1} = \begin{pmatrix} \gamma_{11} & -\gamma_{12} \\ -\gamma_{21} & \gamma_{22} \end{pmatrix}$$

und

$$V = \begin{pmatrix} v_{11} & 0 \\ 0 & v_{22} \end{pmatrix}.$$

Es ergibt sich dann

$$\Gamma^{-1} \cdot V = \begin{pmatrix} \gamma_{11}v_{11} & -\gamma_{12}v_{22} \\ -\gamma_{21}v_{11} & \gamma_{22}v_{22} \end{pmatrix}$$

und

$$\begin{aligned} \Gamma^{-1} \cdot V \cdot \Gamma^{-1} &= \begin{pmatrix} \gamma_{11}v_{11} & -\gamma_{12}v_{22} \\ -\gamma_{21}v_{11} & \gamma_{22}v_{22} \end{pmatrix} \cdot \begin{pmatrix} \gamma_{11} & -\gamma_{12} \\ -\gamma_{21} & \gamma_{22} \end{pmatrix} \\ &= \begin{pmatrix} \gamma_{11}^2 v_{11} + \gamma_{12}\gamma_{21}v_{22} & -\gamma_{11}\gamma_{12}v_{11} - \gamma_{12}\gamma_{22}v_{22} \\ -\gamma_{21}\gamma_{11}v_{11} - \gamma_{22}\gamma_{21}v_{22} & \gamma_{21}\gamma_{12}v_{11} + \gamma_{22}^2 v_{22} \end{pmatrix}. \end{aligned}$$

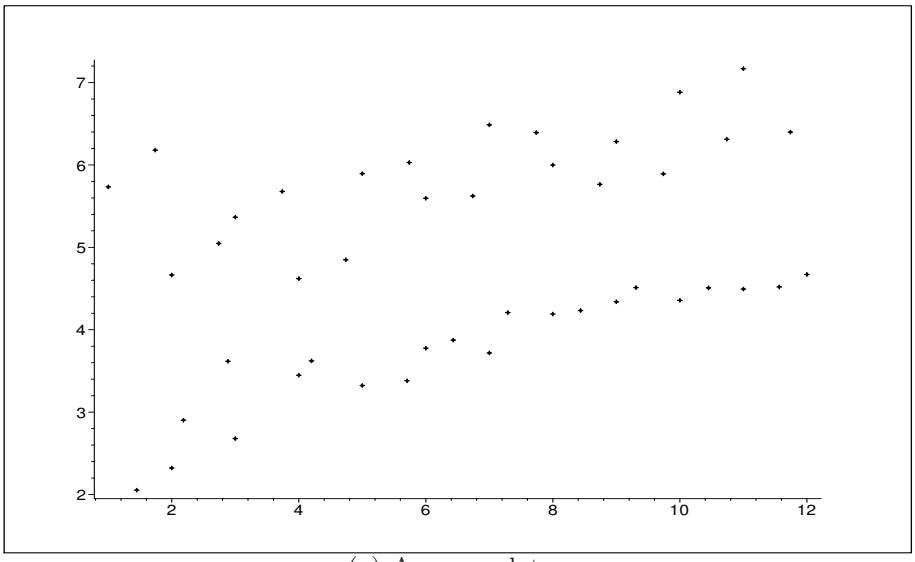
Es ist also erkennbar, dass die Kovarianzmatrix keine Diagonalmatrix ist. Es treten Kovarianzen auf. Die Clusterzentren b_1 und b_2 sind voneinander abhängig.

[29] Pollard, D. (1982a). A central limit theorem for empirical processes. *J. Austr. Math. Soc.* A 33(2)
[30] LeCam, L. (1981). A remark on empirical measures (preprint)

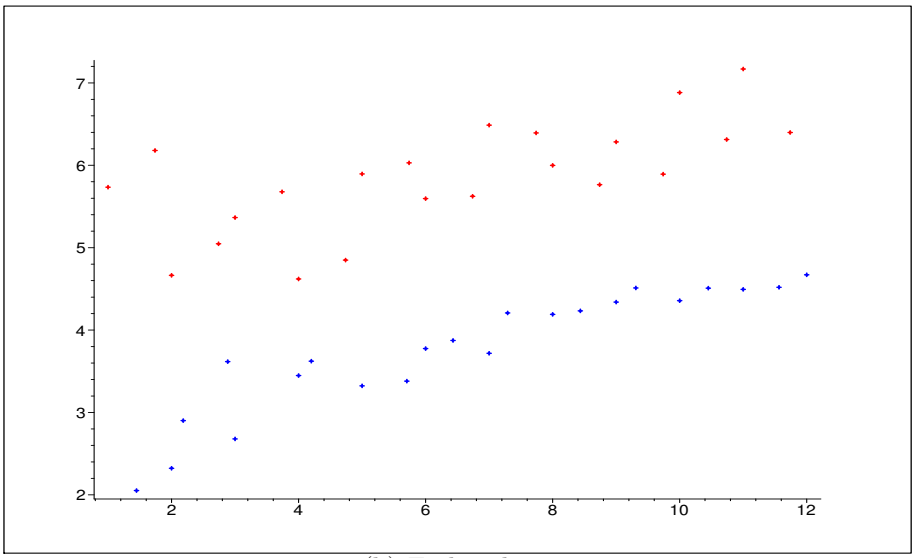
A.2 weitere Beispiele mod. 1-nearest neighbour Verfahren

Die folgenden 3 Beispiele (Abb. (21) bis (23)) verdeutlichen nochmals die Strukturfindung des Verfahrens. Es werden gekrümmte, kurvenförmige Endcluster gebildet, die der Clusterstruktur entsprechen.

Abbildung 21: Beispiel 1 - mod. 1-nearest neighbour Verfahren

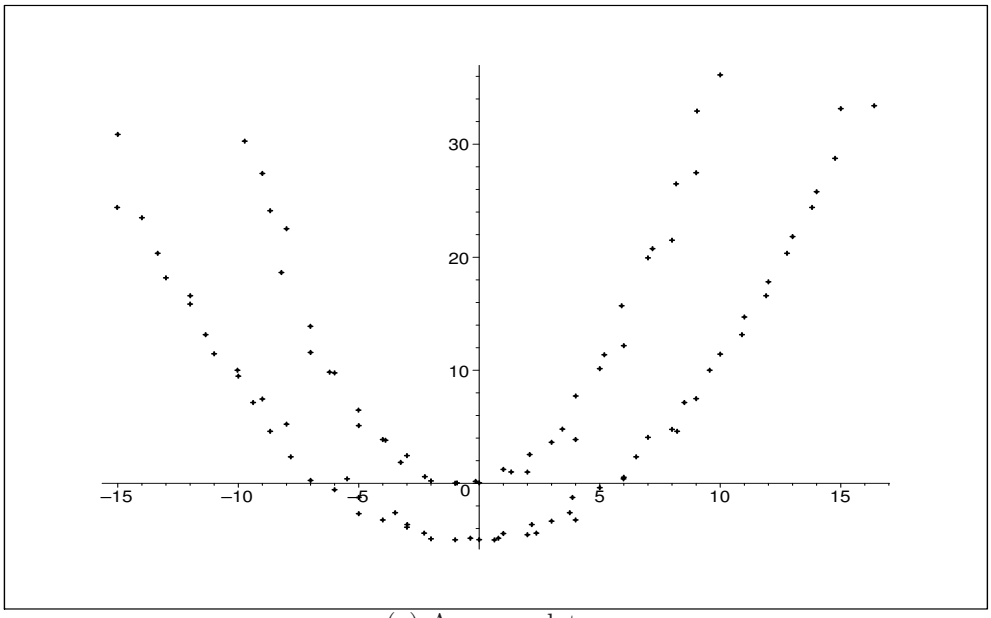


(a) Ausgangsdaten

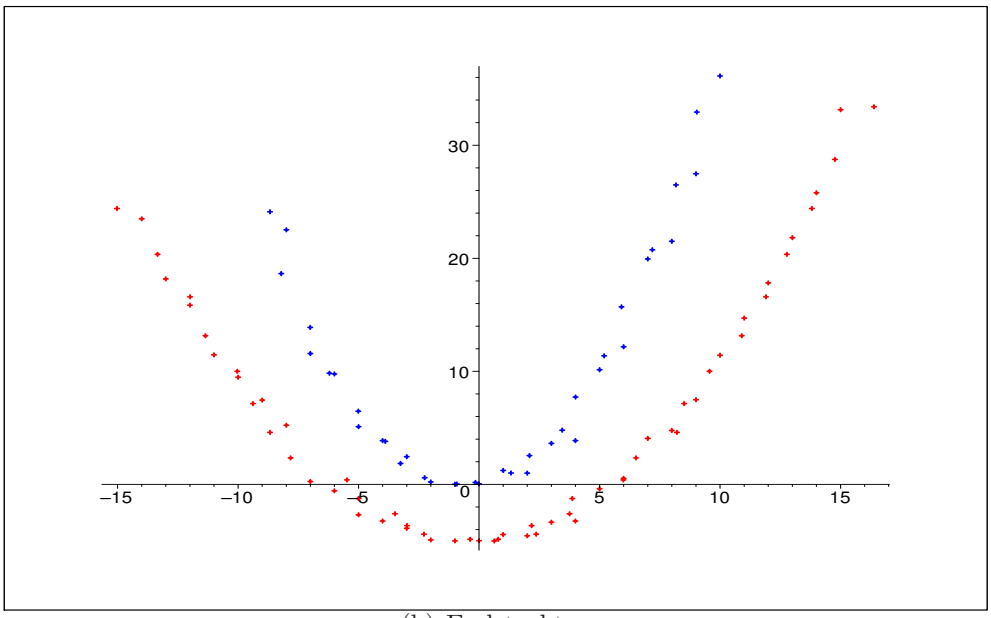


(b) Endstruktur

Abbildung 22: Beispiel 2 - mod. 1-nearest neighbour Verfahren

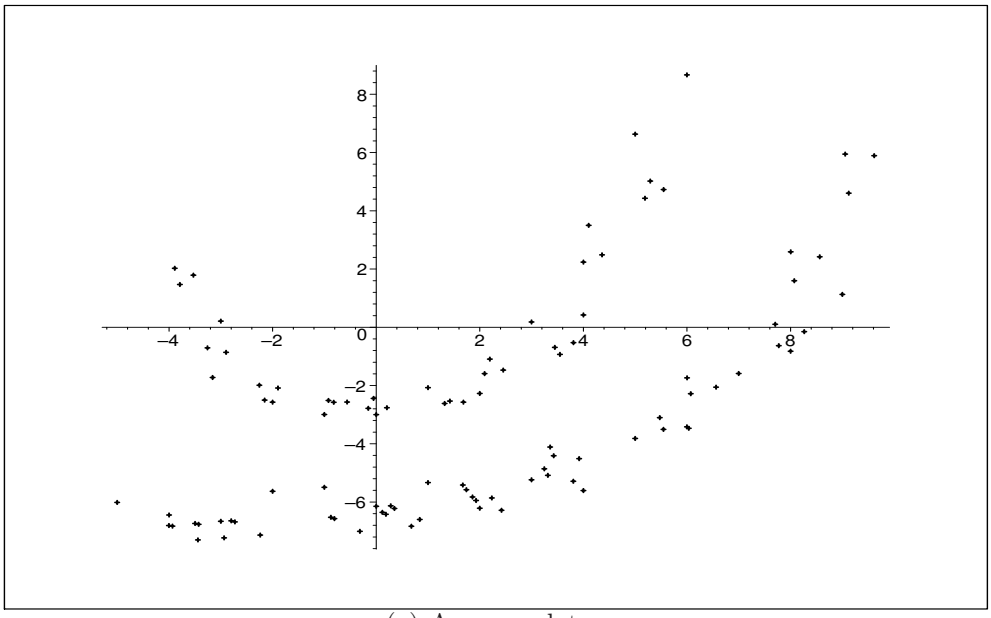


(a) Ausgansgdaten

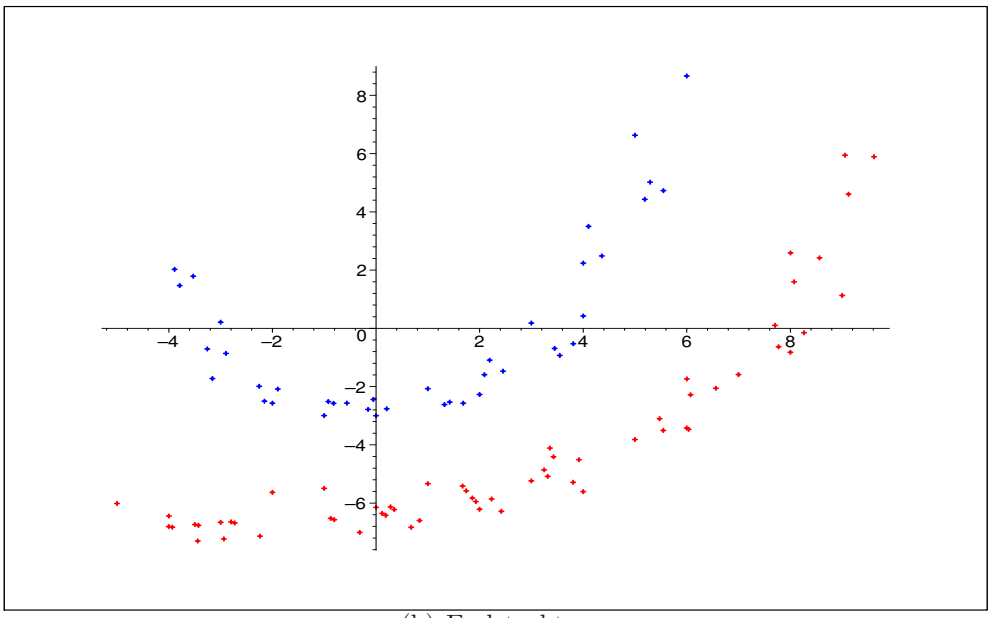


(b) Endstruktur

Abbildung 23: Beispiel 3 - mod. 1-nearest neighbour Verfahren



(a) Ausgangsdaten



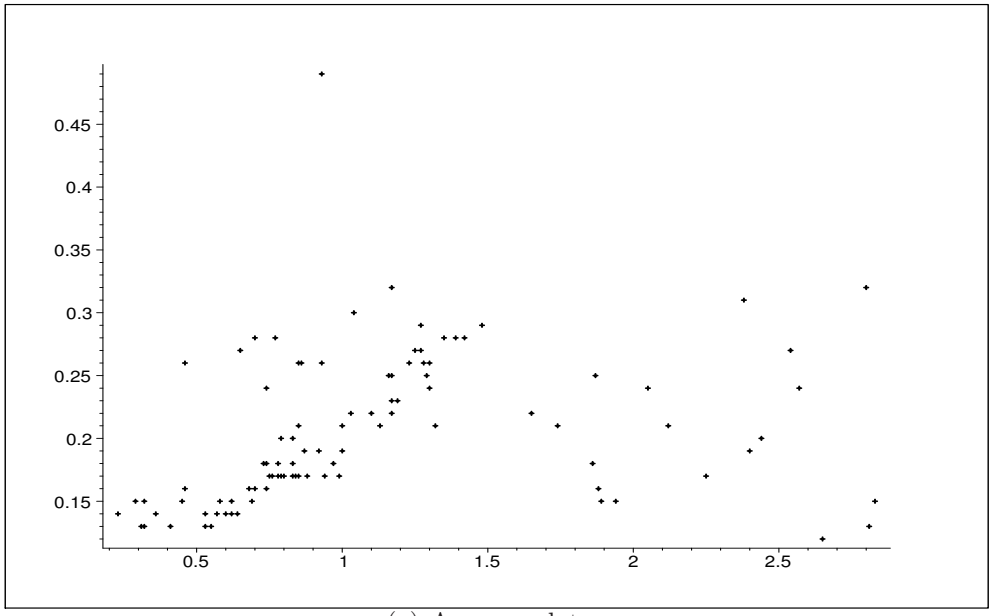
(b) Endstruktur

A.3 weitere Beispiele K-means Verfahren

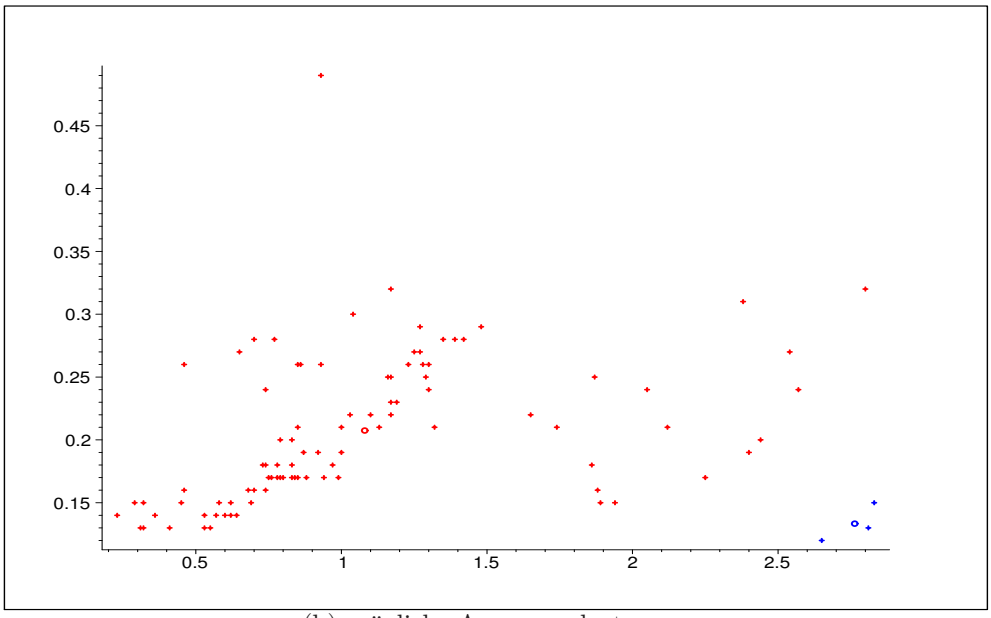
Folgende Grafiken beziehen sich auf die Patientendaten bezogen auf die Veränderung der Nervenzellen (vgl. Abb. 28).

Die Ausgangsdaten für das erste Beispiel sind die Daten bezogen auf die Quotienten (O/N, M/N). Die unterschiedlichen Ausgangsclusterungen in Abb. 24(b) und (d) führen zur gleichen Endclusterung (vgl. Abb.24(c) und (e)), welche die Daten in Erkrankte mit Morbus Pick (blaues Cluster) und Nichterkrankte an Morbus Pick (rotes Cluster) einteilt.

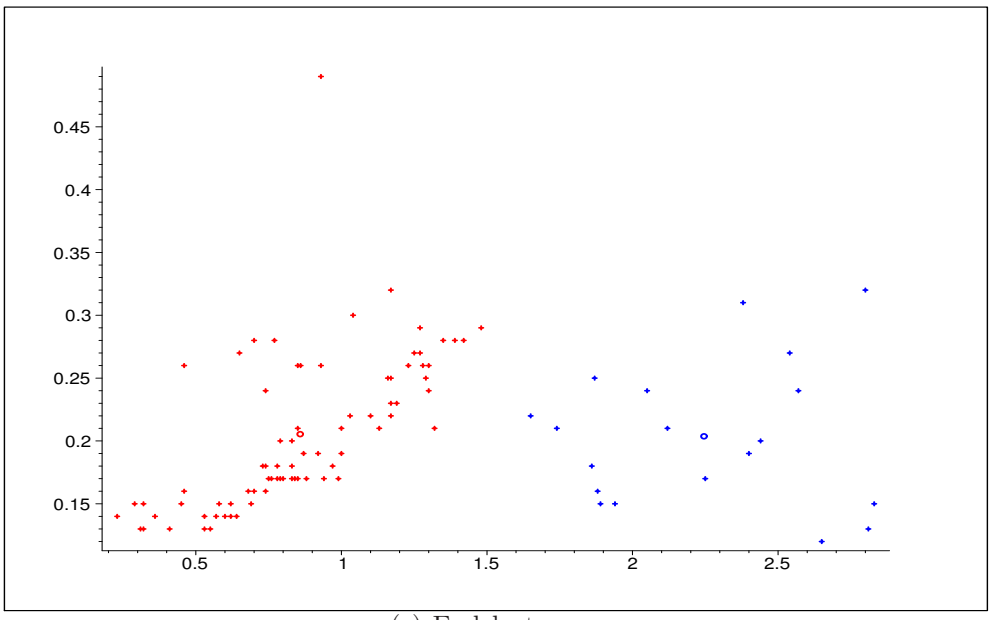
Abbildung 24: Beispiel 1 - K-means Verfahren



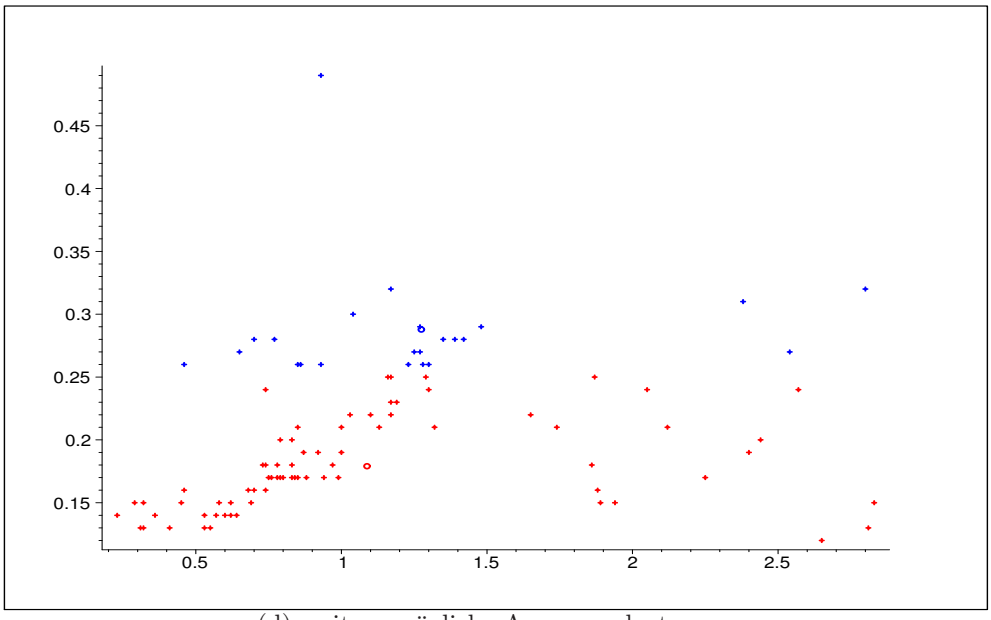
(a) Ausgangsdaten



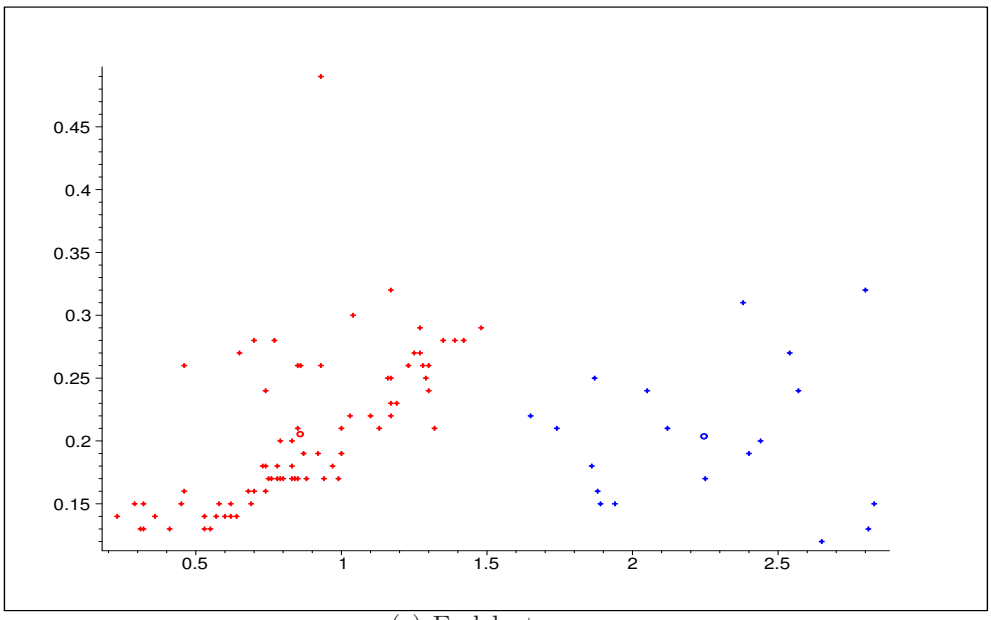
(b) mögliche Ausgangsclusterung



(c) Endclusterung



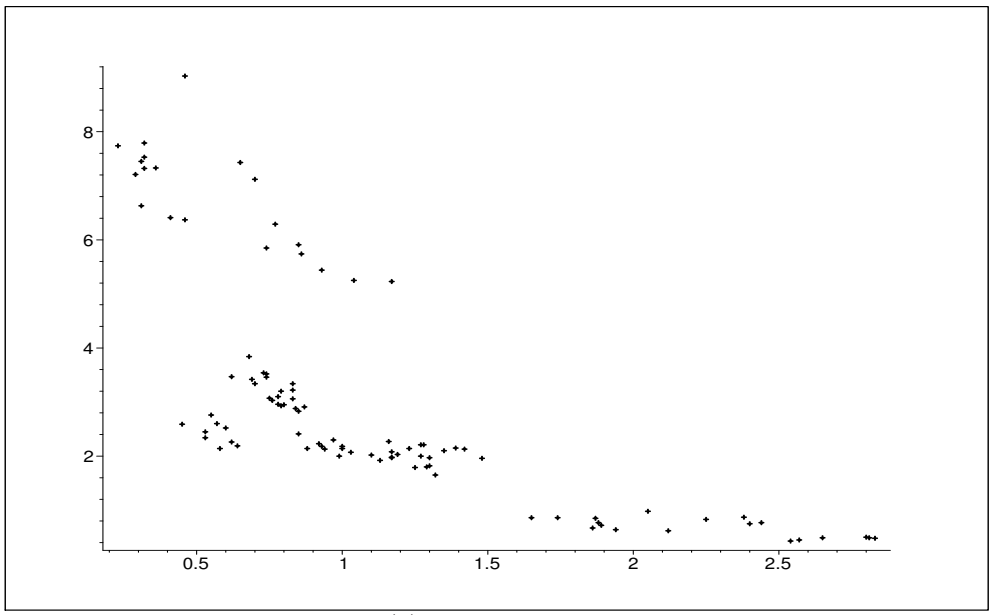
(d) weitere mögliche Ausgangsclusterung



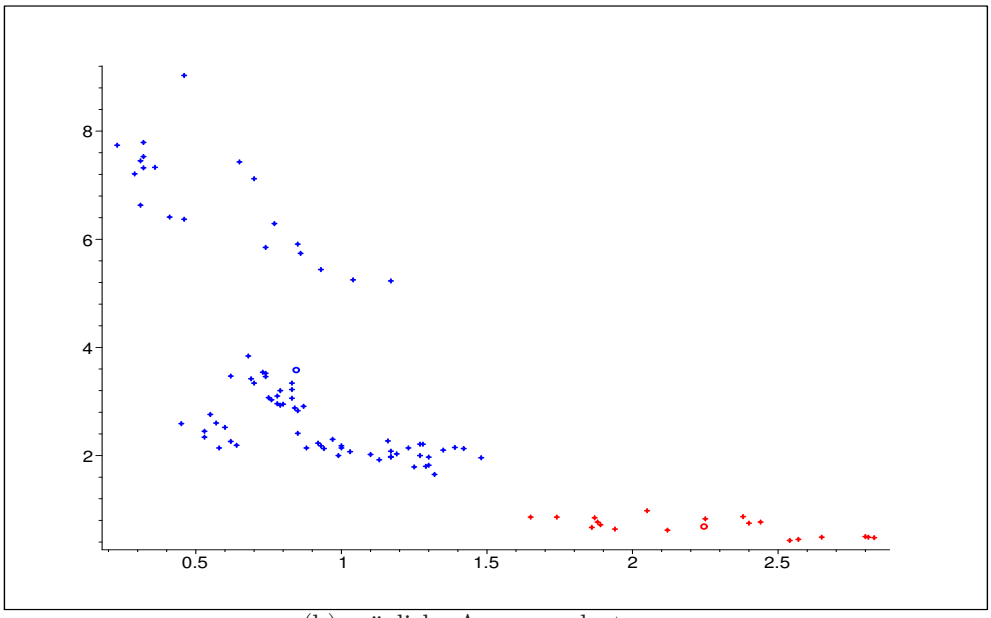
(e) Endclusterung

Ein weiteres Beispiel sind die Patientendaten bezogen auf die Quotienten (O/N, A/N). Die Ausgangsdaten (Abb. 25(a)) lassen sich in 2 Cluster einteilen (vgl. Abb. 25(b)). Es ist auch eine höhere Ausgangsclusterzahl möglich, jedoch sollen die Daten in Alzheimererkrankte und nicht Erkrankung eingeteilt werden. Die Endcluster lassen sich also mit der erwähnten Einteilung in Verbindung gebracht werden (vgl. Abb. 25(c); blau - Alzheimer; rot - nicht Alzheimer erkrankt)

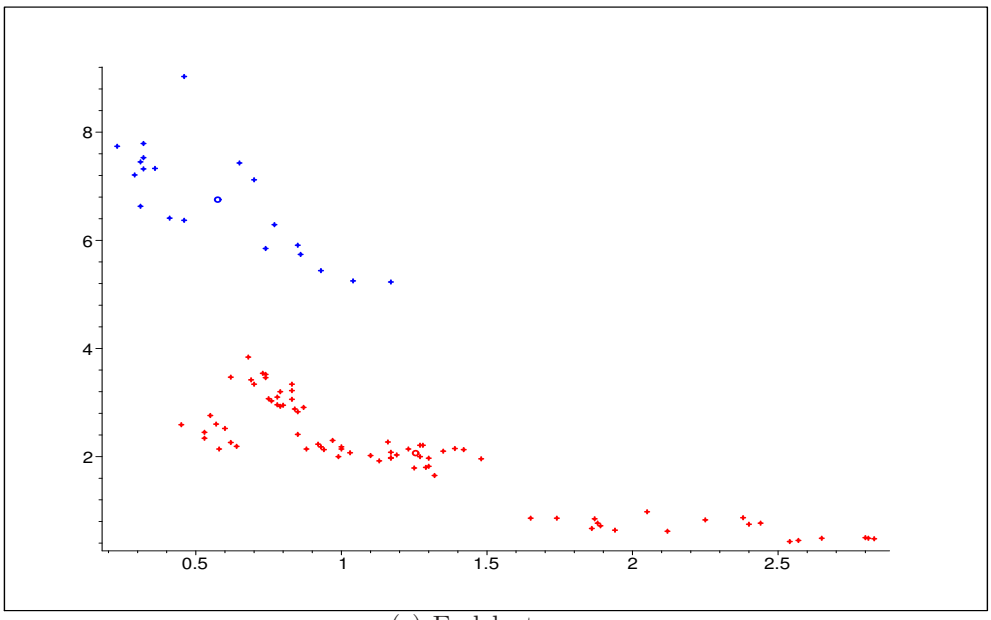
Abbildung 25: Beispiel 2 - K-means Verfahren



(a) Ausgangsdaten



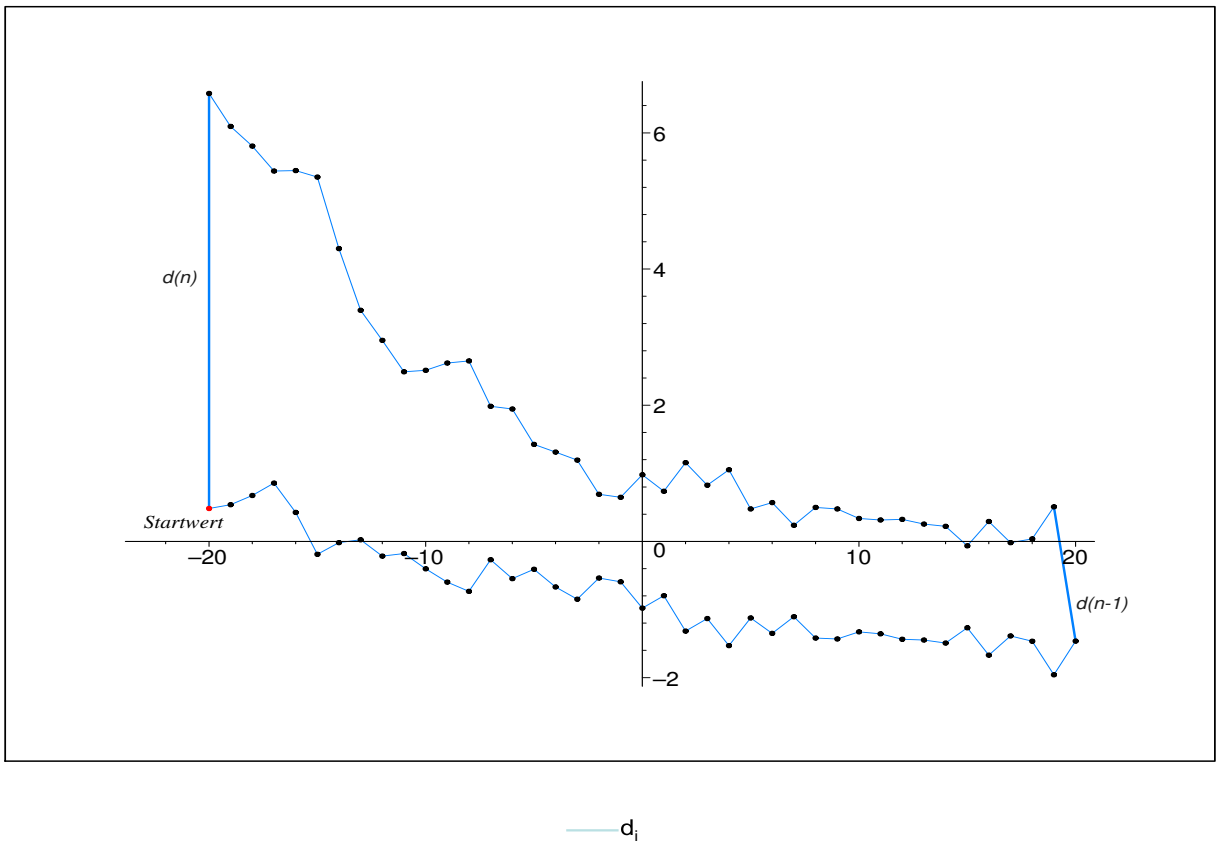
(b) mögliche Ausgangsclusterung



(c) Endclusterung

A.4 Sonstiges

Abbildung 26: grafische Darstellung des Verlaufen des mod. 1-nearest neighbour Verfahrens



Die folgende Tabelle liefert die berechneten Distanzen zwischen den Beobachtungen. Wobei die Tabelle ebenfalls den Verlauf des Verfahrens widerspiegelt, d. h. der nächste Nachbar des Startvektors ist das Element $[-19, .538629659]$. Der nächste Nachbar dieses Elementes ist dann wiederum $[-18, .675403111]$ usw.

Tabelle 1: Verlauf - mod. 1-nearest neighbour Verfahren

	Beobachtungsvektoren	**Distanzen** d_i
1	$[-20, .483081828]$	
2	$[-19, .538629659]$	$1.003085562 = d_1$(Distanz zwischen Vektor 1 und 2)
3	$[-18, .675403111]$	$1.018706977 = d_2$(Distanz zwischen Vektor 2 und 3)
4	$[-17, .856646852]$	1.032849294 (Distanz zwischen Vektor 3 und 4)
5	$[-16, .426140928]$	1.185335351 (Distanz zwischen Vektor 4 und 5)
6	$[-15, -.188199983]$	1.377414755 ...
7	$[-14, -.19047293e-1]$	1.028612633
8	$[-13, .23740829e-1]$	1.001830823
9	$[-12, -.214761200]$	1.056883218
10	$[-11, -.177006982]$	1.001425381
11	$[-10, -.401698729]$	1.050486381
12	$[-9, -.597287815]$	1.038255091
13	$[-8, -.732775302]$	1.018356859
14	$[-7, -.269732451]$	1.214408682
15	$[-6, -.546441192]$	1.076567727
16	$[-5, -.408774583]$	1.018952095
17	$[-4, -.669397242]$	1.067924170
18	$[-3, -.849375757]$	1.032392266
19	$[-2, -.537029082]$	1.097560445
20	$[-1, -.593528904]$	1.003192230
21	$[0, -.97900]$	1.148587966
22	$[1, -.797170576]$	1.033061939
23	$[2, -1.317962582]$	1.271224314
24	$[3, -1.132984024]$	1.034217067
25	$[4, -1.530069247]$	1.157676674
26	$[5, -1.122859217]$	1.165820008
27	$[6, -1.349861779]$	1.051530163
28	$[7, -1.105911910]$	1.059511539
29	$[8, -1.420279954]$	1.098827267
30	$[9, -1.432751848]$	1.000155548
31	$[10, -1.328029340]$	1.010966804
32	$[11, -1.357290190]$	1.000856197

	Beobachtungsvektoren	**Distanzen** d_i
33	$[12, -1.437448364]$	1.006425333
34	$[13, -1.448534223]$	1.000122896
35	$[14, -1.491704696]$	1.001863690
36	$[15, -1.268833447]$	1.049671594
37	$[16, -1.669071036]$	1.160190128
38	$[17, -1.387385068]$	1.079346985
39	$[18, -1.465330340]$	1.006075465
40	$[19, -1.957858976]$	1.242584457
41	$[20, -1.462620559]$	1.245261090
42	$[19, .5084752799]$	$4.885218807 = d_{(n-1)}$
43	$[18, .379928671e-1]$	1.221353701
44	$[17, -.186483996e-1]$	1.003208233
45	$[16, .2920547217]$	1.096536430
46	$[15, -.637107907e-1]$	1.126569100
47	$[14, .2219765867]$	1.081617278
48	$[13, .2555352857]$	1.001126186
49	$[12, .3242256754]$	1.004718370
50	$[11, .3141014595]$	1.000102500
51	$[10, .3369668502]$	1.000522826
52	$[9, .4779262342]$	1.019869548
53	$[8, .4994080520]$	1.000461468
54	$[7, .2374893711]$	1.068601395
55	$[6, .5711586467]$	1.111335186
56	$[5, .4777962622]$	1.008716535
57	$[4, 1.051198514]$	1.328790142
58	$[3, .8258646817]$	1.050775336
59	$[2, 1.156790467]$	1.109511875
60	$[1, .7365105992]$	1.176635167
61	$[0, .97900]$	1.058801110
62	$[-1, .648260825]$	1.109388402
63	$[-2, .691655470]$	1.001883095
64	$[-3, 1.194610127]$	1.252963387
65	$[-4, 1.311232756]$	1.013600838
66	$[-5, 1.424141809]$	1.012748454

	Beobachtungsvektoren	**Distanzen** d_i
67	$[-6, 1.945292314]$	1.271597849
68	$[-7, 1.985183651]$	1.001591319
69	$[-8, 2.650842319]$	1.443101462
70	$[-9, 2.619295179]$	1.000995222
71	$[-10, 2.513576866]$	1.011176362
72	$[-11, 2.490337248]$	1.000540080
73	$[-12, 2.951658595]$	1.212817385
74	$[-13, 3.393830017]$	1.195515566
75	$[-14, 4.300306560]$	1.821699723
76	$[-15, 5.350106623]$	2.102080172
77	$[-16, 5.445275144]$	1.009057047
78	$[-17, 5.440325968]$	1.000024494
79	$[-18, 5.804789238]$	1.132833475
80	$[-19, 6.093449763]$	1.083324899
81	$[-20, 6.578591339]$	1.235362349
1	$[-20, .483081828]$	$37.15523620 = d_{(n)}$ - Distanz zwischen Vektor 81 und 1

Nach diesen berechneten Distanzen und den Bezeichnungen der Elemente ergibt sich die Zuordnung für die Cluster folgendermaßen:

$$\begin{aligned} C_1 &= \{x_1, x_2, x_3, \ldots, x_{41}\} \\ C_2 &= \{x_{42}, x_{43}, x_{44}, \ldots, x_{81}\} \end{aligned}$$

Abbildung 27: grafische Darstellung der Distanzen

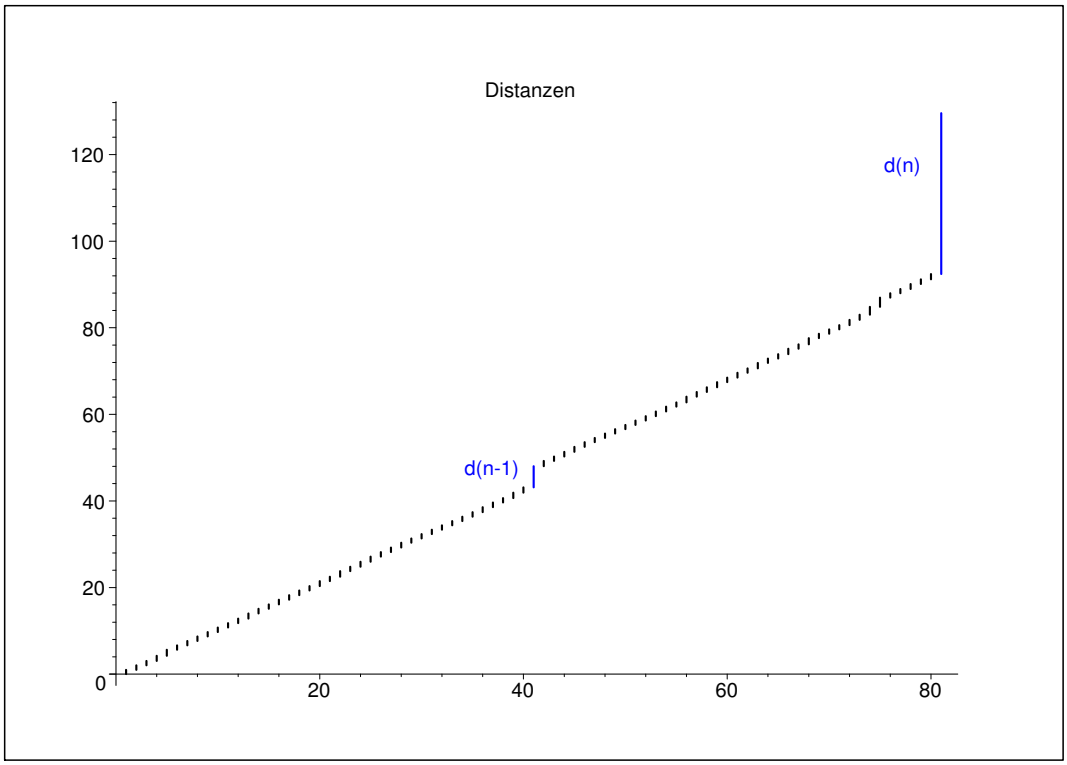


Tabelle 2: Verlauf Gegenbeispiel - mod. 1-nearest neighbour Verfahren

	Beobachtungsvektoren	**Distanzen** d_i
1	$[-20, .483081828]$	
2	$[-17, .856646852]$	9.139550827
3	$[-14, -.19047293e-1]$	9.766840236
4	$[-11, -.177006982]$	9.024951263
5	$[-11, 2.490337248]$	7.114725241
6	$[-10, 2.513576866]$	1.000540080
7	$[-9, 2.619295179]$	1.011176362
8	$[-8, 2.650842319]$	1.000995222
9	$[-7, 1.985183651]$	1.443101462
10	$[-6, 1.945292314]$	1.001591319
11	$[-5, 1.424141809]$	1.271597849
12	$[-4, 1.311232756]$	1.012748454
13	$[-3, 1.194610127]$	1.013600838
14	$[-2, .691655470]$	1.252963387
15	$[-1, .648260825]$	1.001883095
16	$[0, .97900]$	1.109388402
17	$[1, .7365105992]$	1.058801110
18	$[2, 1.156790467]$	1.176635167
19	$[3, .8258646817]$	1.109511875
20	$[4, 1.051198514]$	1.050775336
21	$[5, .4777962622]$	1.328790142
22	$[6, .5711586467]$	1.008716535
23	$[7, .2374893711]$	1.111335186
24	$[8, .4994080520]$	1.068601395
25	$[9, .4779262342]$	1.000461468
26	$[10, .3369668502]$	1.019869548
27	$[11, .3141014595]$	1.000522826
28	$[12, .3242256754]$	1.000102500
29	$[13, .2555352857]$	1.004718370
30	$[14, .2219765867]$	1.001126186
31	$[15, -.637107907e-1]$	1.081617278

	Beobachtungsvektoren	**Distanzen** d_i
32	$[16, .2920547217]$	1.126569100
33	$[17, -.186483996e-1]$	1.096536430
34	$[18, .379928671e-1]$	1.003208233
35	$[19, .5084752799]$	1.221353701
36	$[19, -1.957858976]$	6.082804662
37	$[16, -1.669071036]$	9.083398474
38	$[13, -1.448534223]$	9.048636486
39	$[10, -1.328029340]$	9.014521427
40	$[7, -1.105911910]$	9.049336153
41	$[4, -1.530069247]$	9.179909446
42	$[1, -.797170576]$	9.537140462
43	$[-2, -.537029082]$	9.067673597
44	$[-5, -.408774583]$	9.016449217
45	$[-8, -.732775302]$	9.104976466
46	$[-12, 2.951658595]$	$29.57505314 = d_{(n-1)}$
47	$[-13, 3.393830017]$	1.195515566
48	$[-14, 4.300306560]$	1.821699723
49	$[-15, 5.350106623]$	2.102080172
50	$[-16, 5.445275144]$	1.009057047
51	$[-17, 5.440325968]$	1.000024494
52	$[-18, 5.804789238]$	1.132833475
53	$[-19, 6.093449763]$	1.083324899
54	$[-20, 6.578591339]$	1.235362349
1	$[-20, .483081828]$	$37.15523620 = d_{(n)}$

Die vorliegende Tabelle 2 hat die berechneten Distanzen, bezogen auf eine geringere Elementanzahl, dargestellt. Es ist ein anderer Verlauf des Verfahrens und damit eine „verfälschte“ Clusterung der Daten klar erkennbar. Es wird keine geschwungene bzw. kurvenförmige Struktur gefunden.

Abbildung 28: Patientendaten; *Quelle: Läuter H., Pincus R.; S. 21*

Tabelle 1.1.1. Daten zum Beispiel 1.1.1

Pat.	A/N	O/N	M/N	G/N	A/O	Pat.	A/N	O/N	M/N	G/N	A/O
1	2.04	0.29	0.15	2.84	7.21	50	2.40	0.84	0.17	3.37	2.88
2	1.70	0.23	0.14	2.07	7.74	51	2.12	0.62	0.14	2.80	3.47
3	1.95	0.31	0.13	2.38	6.63	52	2.39	0.70	0.16	3.17	3.34
4	2.24	0.32	0.15	2.71	7.32	53	2.58	0.74	0.18	3.53	3.52
5	2.35	0.32	0.13	2.81	7.53	54	2.26	0.76	0.17	3.19	3.03
6	2.57	0.36	0.14	3.03	7.33	55	2.53	0.68	0.16	3.37	3.84
7	2.28	0.31	0.13	2.71	7.45	56	2.51	0.74	0.16	3.41	3.46
8	2.56	0.41	0.13	3.11	6.41	57	2.47	0.73	0.18	3.39	3.54
9	2.27	0.32	0.13	2.72	7.79	58	2.32	0.69	0.15	3.18	3.42
10	2.77	0.46	0.16	3.38	6.37	59	1.87	0.88	0.17	2.91	2.14
11	4.69	0.70	0.28	5.68	7.12	60	2.02	0.85	0.21	3.06	2.41
12	3.93	0.46	0.26	4.70	9.03	61	2.03	0.92	0.19	3.14	2.23
13	4.72	0.65	0.27	5.66	7.43	62	1.92	0.99	0.17	3.05	2.00
14	4.74	0.77	0.28	5.69	6.29	63	1.99	0.93	0.19	3.10	2.18
15	4.21	0.74	0.24	4.79	5.85	64	2.12	1.00	0.19	3.31	2.14
16	4.95	0.85	0.26	6.06	5.91	65	2.11	1.03	0.22	3.36	2.07
17	4.75	0.86	0.26	5.88	5.74	66	2.14	1.00	0.21	3.35	2.18
18	5.37	1.04	0.30	6.68	5.25	67	1.97	0.94	0.17	3.07	2.13
19	4.92	0.93	0.26	6.12	5.44	68	2.11	0.97	0.18	3.26	2.30
20	5.95	1.17	0.32	7.44	5.23	69	2.17	1.13	0.21	3.39	1.92
21	1.15	2.57	0.24	4.00	0.45	70	2.84	1.48	0.29	4.63	1.96
22	1.30	2.12	0.21	3.60	0.62	71	2.32	1.30	0.26	3.88	1.82
23	1.22	1.86	0.18	3.26	0.67	72	2.36	1.27	0.22	3.79	2.00
24	1.21	2.54	0.27	3.81	0.43	73	2.31	1.29	0.25	3.41	1.80
25	1.39	2.80	0.32	4.51	0.50	74	2.30	1.17	0.25	3.73	1.98
26	1.28	2.65	0.12	3.95	0.49	75	2.20	1.25	0.27	3.71	1.79
27	1.35	2.83	0.15	4.32	0.48	76	2.80	1.28	0.26	4.37	2.21
28	1.27	2.81	0.13	3.99	0.49	77	2.30	1.17	0.22	3.67	1.97
29	1.20	1.94	0.15	3.29	0.64	78	2.36	1.19	0.23	3.80	2.03
30	2.05	2.38	0.31	4.72	0.87	79	2.36	0.85	0.17	3.38	2.83
31	2.01	2.05	0.24	4.31	0.98	80	2.37	0.78	0.18	3.33	3.10
32	1.42	1.65	0.22	3.32	0.86	81	2.27	0.80	0.17	3.26	2.95
33	1.47	1.87	0.25	3.45	0.85	82	2.29	0.78	0.17	3.22	2.96
34	1.86	2.25	0.17	4.29	0.83	83	2.28	0.75	0.17	3.20	3.07
35	1.85	2.44	0.20	4.49	0.77	84	2.49	0.87	0.19	3.53	2.91
36	1.43	1.88	0.16	3.46	0.77	85	2.53	0.83	0.17	3.52	3.06
37	1.80	2.40	0.19	4.39	0.75	86	2.73	0.83	0.20	3.76	3.34
38	1.37	1.89	0.15	3.43	0.72	87	2.53	0.79	0.17	3.47	3.20
39	1.48	1.74	0.21	3.42	0.86	88	2.63	0.83	0.18	3.65	3.22
40	1.14	0.45	0.15	1.74	2.59	89	2.16	1.32	0.21	3.70	1.65
41	1.22	0.58	0.15	1.96	2.14	90	2.43	1.17	0.23	3.84	2.08
42	1.49	0.55	0.13	2.18	2.76	91	2.51	1.30	0.24	4.07	1.97
43	1.23	0.53	0.13	1.87	2.34	92	3.03	1.42	0.28	4.71	2.13
44	1.38	0.64	0.14	2.16	2.19	93	2.20	1.10	0.22	3.53	2.02
45	1.48	0.60	0.14	2.22	2.52	94	2.86	1.35	0.28	4.44	2.10
46	1.35	0.62	0.15	2.12	2.26	95	2.53	1.16	0.25	3.94	2.27
47	1.26	0.53	0.14	1.92	2.45	96	2.94	1.39	0.28	4.63	2.15
48	1.43	0.57	0.14	2.11	2.60	97	2.55	1.23	0.26	4.06	2.14
49	2.26	0.79	0.20	3.25	2.93	98	2.71	1.27	0.27	4.24	2.21

B Maple-Quellcode

B.1 Quellcode: mod. 1-nearest neighbour Verfahren

```
> C:=P1:nC:=nops(C);nP:=nops(P1):  # Umschreiben der Ausgangsdaten in ein Cluster
>                                  # nP wird als unveränderte Anzahl während des Ablaufes verwendet
>
> h:=1:N:=1:
> L:=C[ig];  # Startwert (abhängig von der Ausgangsmenge)
>
> for l from 1 to nP do      #äußere Schleife
>
>  distanz1:=10000:        # Festlegung eines Vergleichswertes, der nicht überschritten wird
>
>  if nC=1 then         # Schleife für das Ende des Verfahrens
>                       # Ermittlung des Abstandes zwischen dem letzten Element und dem  Startwert
>
>    distanz:=sum((L[mm]-Ctemp[1][mm])^2,mm=1..2);
>    Distanz[h]:=distanz:
>    C:=C minus {L}:nC:=nops(C);
>    Ctemp[h]:=L:
>    break:
>  end:
>
>  for j from 1 to nC do     # Ermittlung des nächsten Nachbarn
>
>    distanz:=sum((L[m]-C[j][m])^2,m=1..2);
>    if distanz=0 then
>     distanz:=10000^2;
>    end:
>
>   if distanz<distanz1 and (distanz>0) then
>     distanz1:=distanz:
>     N:=j:
>   end:
>
>  end do:
>
> LL:=C[N]:          # LL ist der nächste Nachbar von L
> Ctemp[h]:=L:       # L (für diesen Vektor wurde der nächste Nachbar im Iterationsschritt ermittel) wird in eine
>                    # temporäre Menge verschoben
>
> C:=C minus {L}:nC:=nops(C);   # L wird aus der Menge der zu betrachteten Elemente entnommen
>
> Distanz[h]:=distanz1:      # Merken der ermittelte Distanzen zwischen den Elementen (nächsten Nachbarn)
>
> h:=h+1:
> L:=LL:     # der gerade ermittelte Nachbar wird nun zum Ausgangselement des nächsten Iterationsschrittes
> end do:
> C;nC;
> print(Distanz);
> print(Ctemp);
>
> MaxAbstand1:=max(seq(Distanz[ds],ds=1..nP)); # Ermittlung der maximalen Distanz unter den Distanzen
>
>  for t from 1 to nP do
>  if MaxAbstand1=Distanz[t] then
>  break:
>  end:
> end do:
>
> # Ermittlung der "Trennstelle" zwischen den Elementen
> # erster Cut bei t (größter Distanzwert)
```

```
> if t=nP then
>  t;1;
>  print("maximaler Abstand zwischen: ",Ctemp[t],"und",Ctemp[1]);
> else
>  t;t+1;
>  print("maximaler Abstand zwischen: ",Ctemp[t],"und",Ctemp[t+1]);
> end;
>
> dd:=0:
> for ff from 1 to nP do        # Ermittlung des zweitgrößten Distanzwertes
>   if Distanz[ff]>dd and (ff<>t) then
>    dd:=Distanz[ff]:
>    MM:=ff:
>    end:
> end do:
> dd;MM;
> MaxAbstand2:=dd;
>
> # Fallunterscheidung für die Unterteilung in Cluster bezüglich der "Lage" des größten und zweitgrößten Distanzwertes
> C1:={}:C2:={}:
>
> if t>MM then
>   for hd from MM+1 to t do
>      C1:=C1 union {Ctemp[hd]}:
>   end do:
>  if t<>nP then
>   for hdd from t+1 to nP do
>      C2:=C2 union{Ctemp[hdd]}:
>   end do:
>  else
>   for hdg from 1 to MM do
>      C2:=C2 union {Ctemp[hdg]}:
>   end do:
>  end:
> end:
>
> if t<MM then
>   for hdf from t+1 to MM do
>      C1:=C1 union {Ctemp[hdf]}:
>   end do:
>  if MM<>nP then
>   for hddf from MM+1 to nP do
>      C2:=C2 union{Ctemp[hddf]}:
>   end do:
>  else
>   for hdgf from 1 to t do
>      C2:=C2 union {Ctemp[hdgf]}:
>   end do:
>  end:
> end:
>
> # Ausgabe der erhaltenen Ergebnisse; Endcluster, Anzahl der Elemente in den Cluster
> C1;C2;
> B1:=plot(C1,color=red,style=point,symbol=cross):
> B2:=plot(C2,color=blue,style=point,symbol=cross):
> display(B1,B2);
> nC1:=nops(C1);nC2:=nops(C2);
> n:=nC1+nC2;
```

B.2 Quellcode: K-means Verfahren

```
> # Umschreiben/Umbezeichnung der zuvor erstellten Mengen von Beobachtungen
>
> C1:=P1: nC1:=nops(C1); # C1 = Cluster 1; nC1 = Anzahl der Objekte in C1
> C2:=P: nC2:=nops(C2);  # C2 = Cluster 2; nC2 = Anzahl der Objekte in C2
>
> # Mittelwerte der Ausgangscluster
>
> C1M:=(1/nC1)*sum(C1[k],k=1..nC1);
> C2M:=(1/nC2)*sum(C2[k],k=1..nC2);
>
> # Gütekriterium, welches während des ganzen Verfahrens minimiert werden muss
>
> K1:=sum((C1[m][1]-C1M[1])^2+(C1[m][2]-C1M[2])^2,m=1..nC1):
> K2:=sum((C2[m][1]-C2M[1])^2+(C2[m][2]-C2M[2])^2,m=1..nC2):
> D2:=K1+K2;
>
> d2:=0:
> f:=0:
>
> for jj from 1 to 300 while f=0 do  # äußere Schleife: Festlegung der Iterationsschritte
>
> differenz1:=0:differenz2:=0:
> N1:=0:f1:=0:
> N2:=0:f2:=0:
>
> for i from 1 to nC1 do  # 1. Schleife
>
>   distanz1:=sqrt(sum((C1[i][m]-C1M[m])^2,m=1..2));  # Berechnung des Abstandes der Elemente aus C1 zu ihrem
>                                                      # Mittelwert
>
>   distanz2:=sqrt(sum((C1[i][m]-C2M[m])^2,m=1..2)); # Berechnung des Abstandes der Elemente aus C1 zum
>                                                     #Mittelwert des anderen Clusters
>
>   # Test, ob das jeweilige Element aus C1 näher zum eigenen oder "fremden" Clustermittelpunkt liegt
>
>   if distanz1 > distanz2 then
>     xt:=C1[i]:
>
>     # falls also das i-te Element näher am Mittelwert des Clusters 2 liegt, wird die Differenz
>     # (siehe Diplomarbeit Formel (3.5)) berechnet
>
>     differenz[i]:= nC1/(nC1-1)*sum((xt[m]-C1M[m])^2,m=1..2) –
>                    nC2/(nC2+1)*sum((xt[m]-C2M[m])^2,m=1..2):
>
>     # Suche nach der maximalen Abstandmaßänderung zwischen der alten Clusterung und der neuen Clusterung
>     # (bezogen auf potentielle Umordnung von Cluster 1 in Cluster 2)
>
>      if differenz[i]>differenz1 then
>        differenz1:=differenz[i]:
>        N1:=i:
>      end:
>   end:
>
> end do:
>
> if N1=0 then
>    f1:=1:
> end:
```

```
>
> for j from 1 to nC2 do    # 2. Schleife
>
>   distanz3:=sqrt(sum((C2[j][m]-C1M[m])^2,m=1..2)); # Berechnung des Abstandes der Elemente aus C2 zum
>                                                    # Mittelwert des anderen Clusters
>
>   distanz4:=sqrt(sum((C2[j][m]-C2M[m])^2,m=1..2)); # Berechnung des Abstandes der Elemente aus C2 zum
>                                                    # eigenen Mittelwert
>
>   # ebenfalls wieder der Test, ob das jeweilige Element näher zum eigenen oder zum "fremden"
>   # Clustermittelpunkt liegt
>
>   if distanz4 > distanz3 then
>
>    xt:=C2[j]:
>
>    # falls das j-te Element näher am "fremden" Mittelpunkt (C1M) liegt, wird die Differenz (siehe (3.5)) ermittelt
>
>    differenz[j]:= nC2/(nC2-1)*sum((xt[m]-C2M[m])^2,m=1..2) -
>                   nC1/(nC1+1)*sum((xt[m]-C1M[m])^2,m=1..2):
>
>    # Suche nach der maximalen Abstandmaßänderung zwischen der alten Clusterung und der neuen
>    # Clusterung  (bezogen auf potentielle Umordnung von Cluster 2 in Cluster1)
>
>     if differenz[j]>differenz2 then
>       differenz2:=differenz[j]:
>       N2:=j:
>     end:
>   end:
> end do:
>
> if N2=0 then
>    f2:=1:
> end:
>
> # Am Ende der 1. und 2. Schleife wurden jeweils die maximalen Differenzwerte und das dazugehörige Element
> # gespeichert. Um nun für diesen Iterationsschritt festlegen zu können welches Objekt neu zugeordnet wird,
> # erfolgt die endgültige Ermittlung, bei welcher potentiellen Elementumordnung die beste Verbesserung
> # eingetreten ist.
>
> if (f1=0) and (f2=0) and (differenz1 > differenz2) then
>
>        # Tritt bei der Umordnung von C1 in C2 die größte Verbesserung auf, dann wird das dazugehörige
>        # Element (welches zuvor mit N1=i deklariert wurde) dem Cluster 1 "entnommen" und dem Cluster 2
>        # zugeschrieben.
>
>        C2:= C2 union {C1[N1]}: nC2:=nops(C2):
>        C1:= C1 minus {C1[N1]}: nC1:=nops(C1):  # Neuberechnung der Elementanzahl in den Clustern
>
>        # Berechnung der neuen Clustermittelwerte
>
>        C1M:=(1/nC1)*sum(C1[k],k=1..nC1):
>        C2M:=(1/nC2)*sum(C2[k],k=1..nC2):
>
> end:
>
> if (f1=0) and (f2=1) and (differenz1 > d2) then
>
>        # Falls beim Tausch von Cluster1 in Cluster2 kein Maximum erreicht wird, dann wird das Maximum
```

```
>       # von differenz1 mit dem Ausgangswert d2 verglichen.
>
>       C2:= C2 union {C1[N1]}: nC2:=nops(C2):
>       C1:= C1 minus {C1[N1]}: nC1:=nops(C1):
>
>       # Berechnung der neuen Clustermittelwerte
>
>       C1M:=(1/nC1)*sum(C1[k],k=1..nC1):
>       C2M:=(1/nC2)*sum(C2[k],k=1..nC2):
>
> end:
>
> if (f1=0) and (f2=0) and (differenz2 > differenz1) then
>
>       # Tritt allerdings bei Umordnung von C2 in C1 die größte Verbesserung auf, dann wird dementsprechend
>       # das dazugehörige Element dem Cluster1 zugeordnet.
>
>       C1:= C1 union {C2[N2]}: nC1:=nops(C1):
>       C2:= C2 minus {C2[N2]}: nC2:=nops(C2):  # Neuberechnung der Elementanzahl in den Clustern
>
>       # Berechnung der neuen Clustermittelwerte
>
>       C1M:=(1/nC1)*sum(C1[k],k=1..nC1):
>       C2M:=(1/nC2)*sum(C2[k],k=1..nC2):
>
> end:
>
> if (f1=1) and (f2=0) and (differenz2 > d2) then
>
>       # Falls bei differenz1 kein Maximum erreicht wird, d.h. f1=1 gilt, dann wird differenz2
>       # nicht mit differenz1, sondern mit dem Ausgangswert verglichen.
>
>       C1:= C1 union {C2[N2]}: nC1:=nops(C1):
>       C2:= C2 minus {C2[N2]}: nC2:=nops(C2):
>
>       # Berechnung der neuen Clustermittelwerte
>
>       C1M:=(1/nC1)*sum(C1[k],k=1..nC1):
>       C2M:=(1/nC2)*sum(C2[k],k=1..nC2):
>
> end:
>
> # Falls bei beiden überprüften Neuzuordnungen (aus C1 in C2 und umgekehrt) keine erneute Verbesserung auftritt,
> # dann wird der Kontrollwert f auf 1 gesetzt und die äußere Schleife wird beendet.
>
> if (N1=0) and (N2=0) then
>   f:=1:
> end:
>
> end do:
>
> print("Iterationsschritte = ",jj);   # Ausgabe der benötigten Iterationsschritte
> BC1:=pointplot(C1,color=black):
> BC2:=pointplot(C2,color=red):
> display(BC1,BC2);   # Zeichnet das erhaltene Clusterungsergebnis
> print("nC1:"nC1,"nC2:"nC2);   # Ausgabe der neuen Elementanzahl in den Endclustern
```