LUDWIG MAXIMILIANS-UNIVERSITÄT MÜNCHEN

FAKULTÄT FÜR MATHEMATIK, INFORMATIK UND STATISTIK

INSTITUT FÜR STATISTIK

BACHELORARBEIT

Wissenschaftliche Arbeit zur Erlangung des akademischen Grades
Bachelor of Science

Interne Validierung für deskriptives Clustering von Genexpressionsdaten

Autorin: Anastasiia Holovchak

betreut durch
Prof. Dr. Anne-Laure Boulesteix
Theresa Ullmann

20. Januar 2020

Inhaltsverzeichnis

1. Motivation

Clusteralgorithmen werden oft für Analyse von Genexpressionsdaten verwendet. Man versucht in diesem Kontext die Gene sinnvoll in Gruppen anhand bestimmter Muster in den Variationen von Expressionsniveaus aufzuteilen. In der Anwendung wird man mit großer Vielzahl an Clusteralgorithmen konfrontiert und es gibt keine eindeutige Antwort auf die Frage, welcher von allen bevorzugt werden soll. Das heißt, dass verschiedene Algorithmen oder sogar verschiedene Konfigurationen von ein und demselben Algorithmus zu ganz unterschiedlichen Partitionen für denselben Datensatz führen können. Eine etablierte Vorgehensweise setzt Bestimmung von mehreren Partitionen voraus, wobei anschließend diejenige Partition bevorzugt wird, die zu den Daten am besten "passt". Dafür probiert man unzerschiedliche Algorithmen mit mehreren Inputparametern aus und man beurteilt diese mit geeigneten Clustervalidierungsmethoden.

Hierbei unterscheidet man zwischen *externen* und *internen* Validierungsmethoden. Externe Validierung basiert auf dem Vergleich der Clusteringergebnisse mit korrekter Partitionierung der Daten; alternativ vergleicht man zwei verschiedene Clusteringergebnisse untereinander. Interne Validierungsmethoden hingegen beziehen keine zusätzlichen Informationen über die Daten ein, sondern basieren ausschließlich auf Beurteilung der vom Algorithmus erhaltenen Partition. In der Anwendung sind aber "korrekte" Strukturen in den Daten unbekannt, das heißt also die wahre Gruppenzugehörigkeit der Objekte ist nicht vorhanden. Deswegen der Fokus in dieser Arbeit auf internen Validierungsmethoden. Die Idee interner Validierungskriterien liegt typischerweise auf Beurteilung von Kompaktheit sowie Separation von Clustern (Handl et al., 2005).

Wie bereits angedeutet wurde, probiert man in der Praxis verschiedene Clusteralgorithmen mit unterschiedlichen Parametern aus um das "optimale" Clustering zu finden. Es ist aber nicht auszuschließen, dass ein bestimmtes Clustering besser als alle anderen bei interner Validierung abschneidet, was aber nur an der Vielzahl der ausprobierten Clustering Methoden liegt. Das heißt, dass solch ein "gutes" Clustering nur durch Zufall zustande kommt und jedoch nicht die wahre Struktur der Daten beschreibt. Deshalb wird im Rahmen dieser Arbeit untersucht, ob solche möglichen Data-Dredging-Effekte aufgedeckt werden können. Eine im Fokus liegende Idee ist die Aufteilung des verwendeten Datensatzes in einen Trainings- und Testdatensatz. Dabei ist zu erwarten, dass eine Clustermethode, die auf dem Trainigsdatensatz gut abschneidet und somit den anderen Methoden bevorzugt wird, genauso ein gutes Resultat auf den Testdaten zeigt. Wird es nicht der Fall sein, liegt womöglich ein Data-Dredging-Effekt vor.

Weiterhin soll untersucht werden, welche Maßzahlen der internen Validierung dabei ausgewählt werden sollen. Das untersuchte Qualitätskriterium der Indizes wird ihre Stabilität auf dem Testdatensatz sein. Es wird also untersucht, ob Indizes, die auf den Trainingsdaten gut abschneiden, genauso gute Ergebnisse auf den Testdaten ausweisen können. Dabei werden nicht die einzelnen Werte der internen Validierungsindizes von Interesse sein, sondern es werden die Indizes für verschiedene Algorithmen und unterschiedliche Konfigurationen

der Inputparameter untereinander verglichen.

Der Rest von dieser Arbeit ist wie folgt aufgebaut. Im Kapitel 2 wird der für die Analyse verwendete Datensatz beschrieben. Ausgewählte Clustering Algorithmen sowie Validierungsindizes werden im Kapitel 3 diskutiert, danach werden im Kapitel 4 sämtliche Ergebnisse präsentiert und abschließend wird im Kapitel 5 das Fazit gegeben.

2. Datensatz

Der für die Clusteranalyse verwendete Datensatz ist ein Teil der Datensammlung *The Cancer Genome Atlas Breast Invasive Carcinoma* (Lingle et al., 2016), die dem Zweck der Untersuchung des Zusammenhangs von Krebsphänotypen mit Genotypen dient. Somit basieren sämtliche Ergebnisse dieser Arbeit auf Daten, die vom TCGA Research Network generiert wurden: `http://cancergenome.nih.gov/`. Als Basis der Datensammlung wurden klinische Bilder von Probanden aus dem *TCGA* bereitgestellt.

Der im Rahmen dieser Arbeit verwendete Datensatz enthält Genexpressionsdaten von 1092 Patientinnen mit Brustkrebs, die mit Hilfe der Sequentierungsmethode *RNA-Seq* gemessen wurden.

Bei der Formatierung des Datensatzes wurden mehrfache Samples gelöscht. Zusätzlich wurden niedrig exprimierte Gene entfernt. Im Rahmen der Normierung von Genexpressionsdaten wurden *log2-CPM* Werte berechnet.

Der formatierte Datensatz enthält die Genexpressionen von 22694 Genen für 1092 Patientinnen, die an Brustkrebs leiden. Man interessiert sich für Clustering der Gene als Objekte, es soll also untersucht werden, ob es Gruppen von Genen gibt, die sich bezüglich ihrer Expressionsniveaus ähnlich verhalten. In diesem Fall enthält der Datensatz eine feste Menge von Genen, die für Untersuchung von Brustkrebs vom Interesse sind. Somit spricht man vom *deskriptiven Clustering*, wobei Gene eine feste bekannte Menge an untersuchten Objekten bilden.

Für die Analyse wurde eine nicht zufällige Stichprobe von $n = 2000$ Genen gezogen, wobei maximale Variabilität der Daten als inhaltliches Auswahlkriterium genommen wurde.

Anschließend wurde dieser Datensatz mehrfach in einen Trainingsdatensatz und einen Testdatensatz aufgeteilt, wobei das Größenverhältnis 80%/20% bei jeder Aufteilung betrug.

3. Methoden

3.1 Clusteralgorithmen

Wie bereits erwähnt wurde, interessieren wir uns für Entdeckung der Gruppen von Genen, die funktional zusammenhängen beziehungsweise ähnliches Verhalten aufweisen. Die Idee vom Clustering liegt daran, Gruppen von Objekten im Datensatz zu finden. Formal gesehen, gegeben sei ein Datensatz $\mathcal{D} = \{x_1, ..., x_n\}$, der n Elemente enthält. Die Aufgabe von Clusteranalyse liegt darin, den Datensatz in K disjunkte Teilmengen $C_1, ..., C_K$ aufzuteilen, das heißt man sucht nach einer Partitionierung der Daten $\mathcal{C} = \{C_1, ..., C_K\}$.

Es gibt eine Vielzahl von Clusteralgorithmen und die zentrale Frage lautet, welcher Clusteralgorithmus am besten für die Analyse der Genexpressionsdaten geeignet wäre (Datta and Datta, 2003). Somit liegt der Vergleich von verschiedenen Algorithmen im Fokus dieser Arbeit.

Im Folgenden werden drei wichtige Ansätze von Clustermethoden vorgestellt und untereinander verglichen.

3.1.1 Partitionierende Verfahren (K-means Algorithmus)

Für Partitionsverfahren soll die Anzahl der Cluster K vorgegeben werden. Wir fokussieren uns zunächst auf dem *K-means* Algorithmus. Nachdem die gewünschte Anzahl an Cluster gewählt wurde, beschäftigt man sich mit der Wahl von K initialen Cluster-Repräsentanten, der sogenannten Zentroiden. Der Algorithmus ordnet dann die Objekte den einzelnen Clustern zu mit dem Ziel der Minimierung von totaler Quadratsumme innerhalb der Cluster. Nachdem die Objekte den einzelnen Clustern zugewiesen wurden, werden für jede Klasse neue Zentroide bestimmt und das Verfahren wird sukzessive wiederholt bis keine Veränderung bei der Anordnung von Objekten mehr vorliegt.

Dabei wird die Frage der Minimierung von Intra-Cluster Variation formal definiert als

$$S(\mathcal{D}, m_1, ..., m_K) = \sum_{i=1}^{n} d(x_i, m_{c(i)}),$$

wobei

$$c(i) = \underset{j \in \{1,...,K\}}{\arg\min}\ d(x_i, m_j), \quad j = 1, ..., K.$$

Dabei werden die Zentroide mit $m_1, ..., m_K$ bezeichnet und d ist ein geeignetes Unähnlichkeitsmaß. Typischerweise wird für d die quadrierte euklidische Distanz verwendet.

Der K-means Algorithmus wurde im *R*-Paket `stats` als Funktion `kmeans()` implementiert. Dabei sollten der Datensatz, die Anzahl an Cluster, die maximale Anzahl an Iterationen sowie die Anzahl an zufälligen Startpartitionen als Input-Parameter der Funktion

übergeben werden. Letztes Parameter ist besonders wichtig um Stabilität des Verfahrens sicherzustellen.

3.1.2 Agglomerative hierarchische Verfahren

Im Gegensatz zu den Partitionsverfahren wird von hierarchischen Verfahren keine feste Anzahl an Clustern sondern eine Hierarchie von Clustern bestimmt. Im ersten Schritt bildet jedes Objekt sein eigenes Cluster. In jedem nächsten Schritt werden zwei "ähnlichste" Cluster zu einem gemeinsamen Cluster zusammengefügt. Das Zusammenfügen der Cluster wird in jedem Schritt sukzessive wiederholt bis alle Objekte ein einziges Cluster bilden. Man unterscheidet zwischen *Single Linkage*, *Complete Linkage* und *Average Linkage* Verfahren, jedes von denen ein anderes Distanzmaß D für das Zusammenfügen der Cluster verwendet

- *Single Linkage* berechnet die kürzeste Distanz zwischen Objekten aus zwei Clustern

  $$D(C_1, C_2) = \min_{x_1 \in C_1, x_2 \in C_2} d(x_1, x_2)$$

  und neigt somit zur Kettenbildung, was oft der Identifikation von Ausreißern dient

- *Complete Linkage* bestimmt den größten Abstand zwischen Objekten aus zwei Clustern

  $$D(C_1, C_2) = \max_{x_1 \in C_1, x_2 \in C_2} d(x_1, x_2)$$

  und ist somit empfindlicher gegenüber kleinen Änderungen in den Daten als Single Linkage

- *Average Linkage* ist ein Kompromiss zwischen den oberen zwei Verfahren und berechnet die durchschnittliche Distanz zwischen Objekten aus zwei Clustern

  $$D(C_1, C_2) = \frac{1}{|C_1||C_2|} \sum_{x_1 \in C_1, x_2 \in C_2} d(x_1, x_2)$$

  und führt zu sehr homogenen Clustern

Um die Partition der Daten in K Cluster zu erhalten, "schneidet" man die Hierarchie an einer bestimmten Stufe ab.

Agglomerative hierarchische Verfahren wurden in R ebenfalls im Standard-Paket `stats` implementiert. Dafür ruft man die Funktion `hclust()` auf. Folgende zwei Parameter sollen der Funktion übergeben werden: die Distanzmatrix und die Agglomerationsmethode.

3.1.3 Spektrales Clustering-Verfahren

Ähnlich wie bei hierarchischen Verfahren, berechnen wir zuerst die paarweisen Distanzen zwischen den Datenobjekten. Wie vorher angedeutet wurde, gehören zwei Objekte zu verschiedenen Clustern, falls sie weit voneinander liegen oder anders ausgedrückt eine große Distanz aufweisen. Zwei Objekte, die "weit auseinander" liegen, können aber auch dem gleichen Cluster gehören, falls sie dichteverbunden sind, das heißt falls es zwischen den repräsentativen Punkten im Raum eine Sequenz von anderen Punkten gibt, die solche zwei Punkte untereinander "verbindet". Bei solchen Fragestellungen liefern Partitionierungsverfahren sowie hierarchische Verfahren meistens kein gutes Clusterergebnis. Spektrales

Clustering-Verfahren kann aber mit solchen Daten korrekt umgehen.

Spektrales Clustering nutzt Methoden, die auf Spektralzerlegung der sogenannten *Laplace-Matrix* basieren. Die Laplace-Matrix wird aus der Distanzmatrix der Daten gebildet. Das Ziel der Spektralzerlegung liegt daran, mittels der K kleinsten (also den K kleinsten Eigenwerten korrespondierenden) Eigenvektoren der Laplace-Matrix eine Abbildung der Daten in einen K-dimensionalen Raum zu konstruieren. Anschließend wird ein Standardverfahren wie zum Beispiel K-means auf die K-dimensionalen Vektoren angewendet.

Das *R*-Paket `kknn` bietet eine Möglichkeit für die Durchführung des Spektralen Clustering-Verfahrens mittels der Funktion `specClust()`. Als Parameter werden die Datenmatrix sowie die gewünschte Anzahl an Clustern benötigt. Wichtig ist hierbei anzumerken, dass die Affinitätsmatrix, die für die Bildung der oben eingeführten Laplace-Matrix benötigt wird, auf dem *kNN* Verfahren basiert.

Wie man an dieser Stelle erkennen kann, kommt zusätzlich zu der Frage der Wahl eines "richtigen" Clustering-Verfahrens auch die Wahl der Input-Parameter wie die Anzahl der Cluster oder die Methode der Bildung einer Distanzmatrix. Somit kommt man auf die Frage, ob man nicht einfach verschiedene Algorithmen mit unterschiedlichen Input-Parametern ausprobieren sollte und dann aus der Vielzahl der Ergebnisse das "beste" Ergebnis auswählen. Dabei kann man aber nicht ausschließen, dass ein so gutes Ergebnis nur durch das Ausprobieren von Vielzahl an Methoden zustande kommt. Eine weitere Frage ist auch, wie man das beste Ergebnis findet. Damit beschäftigen wir uns im nächsten Abschnitt dieser Arbeit.

3.2 Validierungsindizes

Jeder Clustering-Algorithmus erzeugt eigene Partition der Daten, es heißt wir erhalten verschiedene Clusterings, die nicht notwendig alle gleich sind. Somit stellt sich der Anwender die Frage, wie gut jedes einzelne Clustering eigentlich ist? Also inwiefern "passt" das Ergebnis zu den Daten und ob die Struktur, die dahinter steckt, korrekt abgebildet wird.

Um die Güte eines Clusterings zu beurteilen, betrachtet man verschiedene Eigenschaften. Beispielsweise werden die Kompaktheit, die Verbundenheit sowie die räumliche Separation der einzelnen Cluster meistens begutachtet.

Es gibt unterschiedliche Validierungsmethoden von Clusterergebnissen. Dabei unterscheidet man hauptsächlich zwischen *externen* und *internen* Validierungsmethoden.

Externe Validierungsindizes sind unabhängig von den Clusteralgorithmen und dienen zum Vergleich von zwei Clusterings mit denselbsen Clusterobjekten. Im Rahmen dieser Arbeit werden externe Validierungsindizes benötigt um die Übereinstimmung von Clusterings auf dem Trainigsdatensatz mit den Ergebnissen auf dem Testdatensatz zu vergleichen. Es wird also untersucht, inwiefern ein Clusteralgorithmus die gleichen Cluster von Genen auf den Trainigs- sowie Testdaten bildet.

Interne Validierungsmethoden bewerten hingegen einzelne Clusterings. Dabei wird die Qualität eines Clusterings nur anhand der Informationen, die den Daten zugrunde liegen, geschätzt (Handl et al., 2005). Der Fokus dieser Arbeit liegt darauf zu überprüfen, ob Verfahren, die auf dem Trainingsdatensatz bezüglich der internen Validierungsindizes besonders gut abschneiden, ein ähnlich gutes Ergebnis auch auf dem Testdatensatz aufweisen

können. Sollte es nicht der Fall sein, liegt eventuell ein Data-Dredging-Effekt vor.

Folgende Validierungsmethoden wurden auf diejenigen eingeschränkt, die bei der Analyse von Genexpressionsdaten etabliert wurden und am meisten benutzt werden. Sämtliche Notationen wurden von Hennig et al. (2015, Kapitel 26-27) übernommen.

3.2.1 Externe Validierungsindizes

Seien $\mathcal{C}$ und $\mathcal{C}'$ zwei Clusterings mit K und K' Clustern entsprechend. Wir betrachten eine Klasse von externen Indizes, die auf der Anzahl von Objektepaaren basieren, die im Resultat der beiden Clusterings übereinstimmen. Dabei unterscheidet man zwischen vier Fällen:

- N_{11} Anzahl an Paaren von Objekten, die im gleichen Cluster unter $\mathcal{C}$ sowie $\mathcal{C}'$ sind
- N_{00} Anzahl an Objektepaaren, die unter $\mathcal{C}$ und $\mathcal{C}'$ in unterschiedlichen Clustern sind
- N_{10} Anzahl von Paaren, die unter $\mathcal{C}$ im gleichen Cluster sind aber unter $\mathcal{C}'$ zu unterschiedlichen Clustern gehören
- N_{01} Anzahl von Paaren, die unter $\mathcal{C}'$ im gleichen Cluster sind aber unter $\mathcal{C}$ zu unterschiedlichen Clustern gehören

Jaccard Index

Jaccard Index wird definiert als

$$\mathcal{J}(\mathcal{C}, \mathcal{C}') = \frac{N_{11}}{N_{11} + N_{01} + N_{10}}$$

und ist eine Maßzahl für den Anteil der Übereinstimmungen von Objekten in zwei Mengen. Dabei wird die Anzahl N_{00} bei der Berechnung nicht berücksichtigt um die Invarianz des Index gegenüber der Anzahl an Cluster K gewährleisten zu können, da in Hennig et al. (2015, Kapitel 27) gezeigt wird, dass N_{00} mit zunehmendem K steigt.
Die Werte für den Jaccard Index liegen zwischen 0 und 1, wobei große Werte für gute Übereinstimmung der Objekte stehen.

Adjustierter Rand-Index

Adjustierter Rand-Index ist einer der am häufigsten verwendeten externen Validierungsindizes. Der Index ist eine korrigierte Version vom Rand-Index, der den Nachteil einer höheren *Baseline*, die den Erwartungswert des Index unter der Annahme von unabhängigen Clusterings bezeichnet, mit steigender Anzahl an Cluster K hat. Die Korrektur vom Rand Index stellt eine sogenannte Normierung des Index mit der Baseline dar

$$\mathcal{ARI}(\mathcal{C}, \mathcal{C}') = \frac{\sum_{k=1}^{K}\sum_{k'=1}^{K'}\binom{n_{kk'}}{2} - [\sum_{k=1}^{K}\binom{n_k}{2}][\sum_{k'=1}^{K'}\binom{n'_{k'}}{2}]/\binom{n}{2}}{[\sum_{k=1}^{K}\binom{n_k}{2} + \sum_{k'=1}^{K'}\binom{n'_{k'}}{2}]/2 - [\sum_{k=1}^{K}\binom{n_k}{2}][\sum_{k'=1}^{K'}\binom{n'_{k'}}{2}]/\binom{n}{2}},$$

wobei n die Gesamtanzahl an Objekten bezeichnet und $n_{kk'}, n_{k'}$ sowie n_k aus der sogenannten *Kontingenztabelle* herausgenommen werden können. Die Kontingenztabelle ist eine $K \times K'$ Matrix $\mathbf{N} = [n_{kk'}]$, wobei das kk'-te Element die Anzahl der Objekte im Schnitt von Clustern C_k von $\mathcal{C}$ und $C'_{k'}$ von $\mathcal{C}'$ bezeichnet

$$n_{kk'} = |C_k \cap C'_{k'}|$$

und

$$\sum_{k=1}^{K} n_{kk'} = |C'_{k'}| = n_{k'} \qquad \sum_{k'=1}^{K'} n_{kk'} = |C_k| = n_k$$

sowie

$$\sum_{k=1}^{K} n_k = \sum_{k'=1}^{K'} n_{k'} = n$$

Adjustierter Rand-Index ist bei 1 beschränkt, wobei Werte nahe bei 1 gute Übereinstimmung der Clusterings andeuten (Hennig et al., 2015, Kapitel 27).

3.2.2 Interne Validierungsindizes

Interne Validierungsindizes werden für verschiedene Clustering-Algorithmen sowie verschiedene Input-Parameter ausgerechnet und deren Werte werden verglichen um das "beste" Clustering zu bestimmen. Interne Indizes werden besonders oft benutzt um die optimale Anzahl an Cluster zu bestimmen. Im Rahmen dieser Arbeit verwendete Indizes sind nur für metrische Daten definiert und basieren auf Quantifizierung von Unähnlichkeiten zwischen den Objekten (Arbelaitz et al., 2013).

Dabei betrachten wir weiterhin den Datensatz $\mathcal{D} = \{x_1, ..., x_n\}$, der n Elemente enthält. Angenommen das Clustering $\mathcal{C}_K = \{C_1, ..., C_K\}$ sei gegeben und die Funktion c sei die Zuweisungsfunktion, wobei $c(i) = j$ gleichbedeutend ist zu $x_i \in C_j$. Weiterhin bezeichnen wir mit $n_j,\ j = 1, ..., K$ die Anzahl der Elemente in den jeweiligen Clustern. Für metrische Daten definieren wir $\bar{x}_j,\ j = 1, ..., K$ als Mittelwertsvektor für den Cluster C_j und $\bar{x}$ als Gesamtmittelwert.

Calinski-Harabasz Index

Die Idee basiert auf Minimierung der Intra-Cluster-Quadratsumme. Dafür werden die *Intra-Cluster-Varianz*

$$\mathbb{W}_{\mathcal{C}_K} = \sum_{j=1}^{K} \sum_{c(i)=j} (x_i - \bar{x}_j)(x_i - \bar{x}_j)^\top$$

und die *Inter-Cluster-Varianz*

$$\mathbb{B}_{\mathcal{C}_K} = \sum_{j=1}^{K} n_j (\bar{x}_j - \bar{x})(\bar{x}_j - \bar{x})^\top$$

untereinander verglichen.

Calinski-Harabasz Index ist definiert als

$$CH(\mathcal{C}_K) = \frac{trace(\mathbb{B}_{\mathcal{C}_K})}{trace(\mathbb{W}_{\mathcal{C}_K})} \cdot \frac{n - K}{K - 1}$$

Dabei sind große Werte von CH-Index ein Zeichen für ein gutes Clustering, da in diesem Fall die Objekte innerhalb der Cluster homogen sind und gleichzeitig die Cluster sich stark untereinander unterscheiden.

Dabei muss man anmerken, dass CH-Index implizit sphärische Cluster mit Objekten, die um die Clusterzentren konzentriert sind, erfordert. Dabei sollen einzelne Clusterzentren möglichst weit auseinander liegen. In vielen praktischen Anwendungen sind sphärische Cluster jedoch oft nicht der Fall (Hennig et al., 2015).

Davies-Bouldin Index

Davies-Bouldin Index basiert ebenfalls auf dem Vergleich von Intra- und Inter-Cluster-Varianzen. Dabei wird

$$\mathcal{S}_k = \Big(\frac{1}{n_k} \sum_{c(i)=k} ||x_i - \bar{x}_k||_2^q\Big)^{\frac{1}{q}}, \quad k = 1, ..., K$$

als Maß für die Variation innerhalb der Cluster benutzt. Es werden paarweise Ähnlichkeiten zwischen den Clustern berechnet als

$$\mathcal{R}_{ij} = \frac{\mathcal{S}_i + \mathcal{S}_j}{M_{ij}}$$

wobei

$$M_{ij} = ||\bar{x}_i - \bar{x}_j||_p$$

die Distanz zwischen den Zentroiden bezeichnet. Für jedes Cluster wird

$$\mathcal{D}_i = \max_{j \neq i} \mathcal{R}_{ij}$$

berechnet, was als Maß für paarweise "ähnlichste" Cluster dient, und Davies-Bouldin Index ist definiert als

$$DB(\mathcal{C}_k) = \frac{1}{K} \sum_{i=1}^{K} \mathcal{D}_i$$

Dabei wählt man typischerweise $p = q = 2$. Für jedes Cluster wird somit ein Maß für die Ähnlichkeit mit dem "nächstgelegenen" Cluster berechnet und die Cluster werden entsprechend ihrer Größe gewichtet. Die Separation der Cluster wird analog zum CH-Index über die Distanz zwischen den Zentroiden bestimmt. Kleine Werte von DB-Index besagen, dass die einzelnen Cluster homogen und gleichzeitig gut separiert sind.

Dunn Index

Der Index von Dunn verfolgt ebenfalls die Idee des Vergleiches vom Verhältnis der Separation und der Kompaktheit einzelner Cluster

$$DI(\mathcal{C}_K) = \min_{i=1,...,K} \Big\{ \min_{j=i+1,...,K} \Big(\frac{d_C(C_i, C_j)}{\max_{k=1,...,K}(\Delta(C_k))} \Big) \Big\}$$

Dabei ist $d_C(C_i, C_j) = \min_{x \in C_i, y \in C_j} d(x, y)$ ein Maß für die Distanz zwischen zwei Cluster und $\Delta(C) = \max_{x,y \in C} d(x, y)$ bestimmt den Durchmesser oder anders ausgedrückt die "Verbreitung" der einzelnen Cluster. Das Ziel ist Maximierung vom Dunn Index, da große Werte für gute Separation sowie Homogenität sorgen.

Average Silhouette Width Kriterium

Ein weiterer Validierungsindex, der ebenfalls auf Visualisierung interner Clusterhomogenität sowie Separation der Cluster beruht, ist Silhouettenkoeffizient. Für jeden einzelnen Datenpunkt wird berechnet, wie viel deutlicher der Punkt zu dem ihm zugeordneten Cluster gehört als zu dem nächstgelegenen Cluster. Für eine Beobachtung $x_i \in C_k$ ist die Silhoutte definiert als

$$s_i = \frac{b_i - a_i}{\max\{a_i, b_i\}},$$

wobei $a_i = \frac{1}{n_k - 1} \sum_{c(j)=k} d(x_i, x_j)$ die mittlere Distanz vom Objekt x_i zu anderen Objekten aus gleichem Cluster und $b_i = \min_{l \neq k} \frac{1}{n_l} \sum_{c(j)=l} d(x_i, x_j)$ die mittlere Distanz zu dem

nächsten Cluster bezeichnen.

Für ein Clustering $\mathcal{C}_K$ ist somit Average Silhouette Width definiert als

$$ASW(\mathcal{C}_k) = \frac{1}{n} \sum_{i=1}^{n} s_i$$

Dabei wäre bei einem guten Clustering zu erwarten, dass für jedes Objekt x_i mittlere Distanz b_i zum nächsten Cluster größer ist als mittlere Distanz a_i zu den Punkten aus eigenem Cluster. Somit spricht eine große Differenz $b_i - a_i$ für gutes Clustering. Die Werte von s_i sind normiert und liegen zwischen -1 und 1. Entsprechend sind wir an Maximierung von ASW-Kriterium interessiert, da in diesem Fall ein Kompromiss zwischen Homogenität und Separation erreicht wird.

4. Ergebnisse

4.1 Einfache Aufteilung des Datensatzes

Die Daten wurden mit Hilfe aller fünf betrachteten Clustering-Algorithmen partitioniert. Maximale Anzahl an Cluster wurde gleich 10 gewählt, damit inhaltliche Interpretation der gebildeten Cluster gewährleistet werden kann. Es ist wichtig anzumerken, dass Single Linkage Verfahren kein sinnvolles Ergebnis lieferte, da für jede von uns überprüfte Anzahl an Cluster k alle Objekte in ein Cluster eingeordnet wurden und die restlichen $k-1$ Cluster nur jeweils ein Objekt enthielten. Da ein solches Ergebnis für Clustervalidierung wenig interessant ist, wird das Single Linkage Verfahren bei weiterer Analyse nicht berücksichtigt.

Für die restlichen vier Clustering-Algorithmen berechnen wir jeweils vier interne Validierungsindizes, die im Unterabschnitt 3.2.2 beschrieben wurden, einmal für den Trainingsdatensatz und einmal für den Testdatensatz. Vom primären Interesse ist der Vergleich interner Indizes für den Trainings- sowie den Testdatensatz. Zusätzlich werden für jedes Verfahren und jede uns interessierende Anzahl an Cluster externe Validierungsindizes berechnet, die die Frage beantworten sollten, inwieweit die Clusteringergebnisse für Trainings- und Testdaten übereinstimmen. Es macht in diesem Kontext besonders viel Sinn, da für deskriptives Clustering von Genexpressionsdaten eine feste Menge von Genen und nicht die Patienten als Datenobjekte betrachtet werden und somit die Elemente einzelner Cluster sinnvoll untereinander verglichen werden können.

Die Ergebnisse werden in Form der Abbildungen dargestellt, wobei in jeder Abbildung jeweils oben die Ergebnisse für den Trainingsdatensatz und unten für den Testdatensatz dargestellt werden. Für jede Abbildung wurden auf der horizontalen Achse die Anzahl an Cluster und auf der vertikalen Achse die Werte für einzelne Indizes abgetragen. Die Tabellen, die einzelne Werte (gerundet auf fünf Nachkommastellen) für alle verwendeten Indizes enthalten, sind im Anhang A zu finden.

Calinski-Harabasz Index liefert ähnliche Ergebnisse bei jedem der vier Clustering-Algorithmen, wie es aus Abbildung 4.1 abzulesen ist. Wir erinnern uns daran, dass große Werte vom Calinski-Harabasz Index ein Zeichen für gutes Clustering sind. Dabei können wir sehen, dass bei K-means Algorithmus sowie Complete Linkage Clustering der Index für zwei Cluster maximal wird. Diese Ergebnisse werden sowie vom Trainingsdatensatz als auch vom Testdatensatz geliefert, was auf gewisse Stabilität des Indizes hindeutet. Im Gegensatz dazu wird der Index für Partitionen, die von Average Linkage sowie spektralem Clustering für die Trainingsdaten erzeugt wurden, für Anzahl Cluster $k = 3$ maximal, wenngleich der Index immer noch für $k = 2$ auf dem Testdatensatz maximal wird. Eine Erklärung für den extrem kleinen Wert von Calinski-Harabasz Index bei der vom Average Linkage Clustering für zwei Cluster erzeugten Partition auf dem Trainingsdatensatz ist, dass alle bis auf ein Datenobjekt dem ersten Cluster zugeordnet wurden. In diesem Fall muss natürlich die Frage gestellt werden, ob das Ergebnis überhaupt Sinn macht und wie aussagekräftig interne Indizes für den Fall sind.

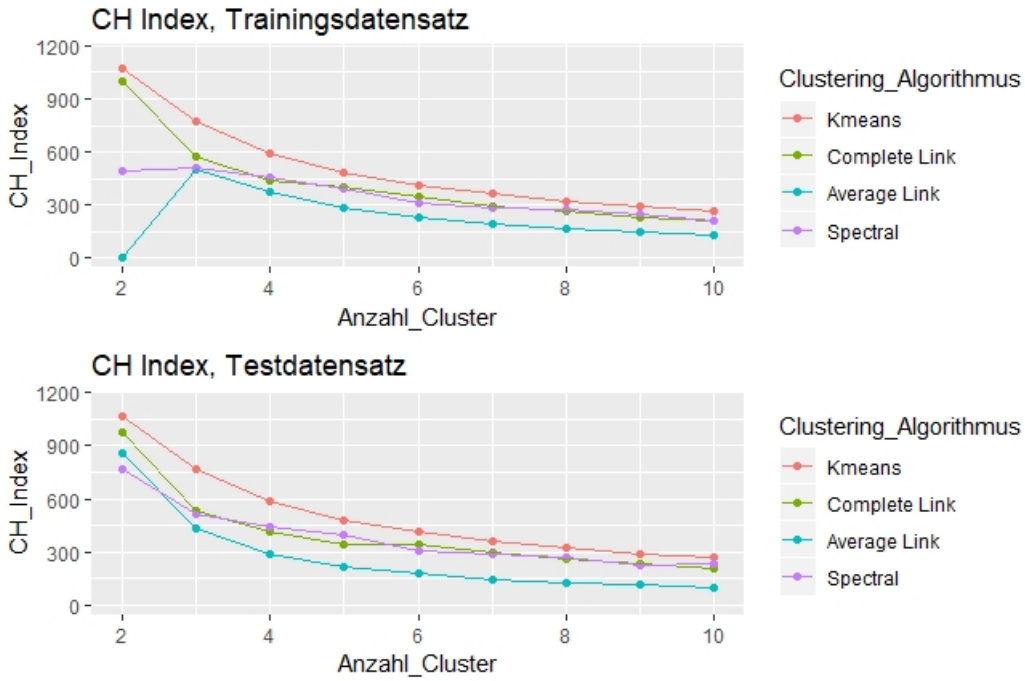

Abbildung 4.1: Calinski-Harabasz Index

Weiterhin analysieren wir Ergebnisse, die vom *Davies-Bouldin Index* geliefert werden (Abbildung 4.2). Eine Partition der Datenpunkte, die homogene aber gleichzeitig gut separierte Cluster liefert, weist einen kleinen Davies-Bouldin Index auf. Somit sehen wir, dass der Index minimal auf dem Trainingsdatensatz für $k = 2$ wird, wenn wir uns die Ergebnisse des K-means Verfahrens sowie des spektralen Clusterings anschauen. Gleichzeitig sehen wir, dass für die zwei Algorithmen der Davies-Bouldin Index auch auf dem Testdatensatz für zwei Cluster minimal bleibt. Also in dem Fall können wir davon ausgehen, dass ein Data-Dredging-Effekt ausgeschlossen ist. Analog können wir sehen, dass für das Complete Linkage Verfahren DB Index auf beiden Teildatensätzen für $k = 3$ Cluster minimal wird, wobei sich die Werte für den Index bei zwei und drei Cluster nicht stark voneinander unterscheiden. Etwas uneindeutig sind die Ergebnisse des Davies-Bouldin Index für Average Linkage Verfahren, da auf dem Trainingsdatensatz der Index seinen minimalen Wert für zwei Cluster annimmt, wobei der Wert für den Index bei den Testdaten für $k = 2$ am höchsten ist, was ein Zeichen für ein schlechtes Cluster Ergebnis ist. Es kann aber immer noch dadurch erklärt werden, dass in unserem Fall für den Trainingsdatensatz alle Datenpunkte laut dem Average Linkage Verfahren einem Cluster gehören und sich nur ein einziger Datenpunkt in dem anderen Cluster befindet. Wenn wir diesen Fall ignorieren würden, wäre der Davies-Bouldin Index für beide Teildatensätze bei $k = 7$ Cluster minimal.

Der Index, der am meisten unterschiedliche Ergebnisse lieferte, ist der *Dunn Index*. Der Index wird berechnet als Verhältnis von kleinster Distanz zwischen zwei Beobachtungen, die unterschiedlichen Clustern zugewiesen wurden, zu der größten Intraclusterdistanz (Brock et al., 2008). In der Abbildung 4.3 können wir nachsehen, dass Werte, die der Index für den Trainings- sowie den Testdatensatz liefert, am weitesten auseinander liegen. Bei K-means Algorithmus sehen wir als Beispiel, dass laut dem Dunn Index Clustering mit 10 Clustern bevorzugt werden sollte, wobei auf den Testdaten der Index für sechs Cluster maximal wird. Beim Average Linkage Verfahren (unter Beachtung des Problems für Partition in

Abbildung 4.2: Davies-Bouldin Index

zwei Cluster) sind die Indexwerte auf beiden Teildatensetzen bis auf mehrere Nachkommastellen für unterschiedliche Anzahl an Cluster nicht unterscheidbar. Für Analyse von Genexpressionsdaten scheint der Index somit wenig geeignet zu sein und soll genauer für weitere Datensätze untersucht werden. Mohanty et al. (2013) zeigte ebenfalls, dass Dunn Index im Vergleich mit Average Silhouette Width eine deutlich schlechtere Validierungsmethode für Genexpressionsdaten ist.

Besondere Aufmerksamkeit müssen wir dem Average Silhouette Width schenken, da der Index als der am meisten verbreitete Validierungsindex für Clustering bekannt ist. Die Ergebnisse für Average Silhouette Width werden in Abbildung 4.4 dargestellt. Der Index wird für Anzahl an Cluster $k = 2$ für drei Clusteralgorithmen sowie auf dem Trainings- als auch auf dem Testdatensatz maximal, und zwar für K-means, Complete Linkage sowie spektrales Clustering. Beim Average Linkage Verfahren wird AWS auf dem Trainingsdatensatz für $k = 3$ Cluster maximal, wobei Clustering auf dem Testdatensatz laut dem Silhoutte Index für $k = 2$ sinnvoller zu sein scheint. Dabei muss man anmerken, dass die Werte für zwei und drei Cluster auf den beiden Teildatensätzen sich nur marginal unterscheiden. Somit schneidet Average Silhouette Index für alle vier Algorithmen bezüglich seiner Stabilität auf Trainings- und Testdaten am besten ab. Der Vorteil beim ASW liegt auch an seiner Normiertheit, da nur Werte vom Intervall $[-1, 1]$ angenommen werden können. Da maximale Werte für Average Silhouette Width für jede der vier von uns verwendeten Clustering Algorithmen den Wert von 0.32 nicht überschreiten, deutet es auf kein eindeutiges Clusterergebnis hin, da Werte von Silhouette Index, die nahe an Null sind, auf Zwischenposition der Datenobjekte zwischen zwei Clustern schließen lassen (Liu and Graham, 2019).

Weiterhin wollen wir zwei Clusterings mit denselben Objekten für jeweils die gleichen Kombinationen von Algorithmus und Inputparameter vergleichen. Es wird also geprüft, inwieweit die Ergebnisse von einzelnen Clustering-Verfahren auf dem Trainings- und dem Testdatensatz übereinstimmen. Im Rahmen dieser Arbeit wurden zwei externe Validie-

Abbildung 4.3: Dunn Index

rungsindizes ausgewählt, nämlich Jaccard Index und Adjustierter Rand Index, jedoch existiert eine Vielzahl an anderen Indizes, die ausführlich in Hennig et al. (2015, Kapitel 27) beschrieben werden.

Jaccard Index deutet mit dem Wert von 0.94 auf sehr gute Übereinstimmung der Objekte für K-means Algorithmus bei zwei Clustern hin. Analog sieht es für Adjustierten Rand Index aus, da der Wert von 0.93 auch für sehr gute Übereinstimmung der Clustering-Resultate spricht. Diese Ergebnisse liefern eine sehr positive Tendenz, da auch interne Validierungsindizes bei K-means Algorithmus für zwei Cluster ihre besten Ergebnisse lieferten.

Beim Complete Linkage Verfahren erhalten wir generell etwas niedrigere Werte für die beiden Indizes, und zwar ist Jaccard Index mit dem Wert von circa 0.73 für Clusterings mit zwei und drei gebildeten Clustern am höchsten, und Adjustierter Rand Index schwankt zwischen den Werten von 0.64 und 0.66 für zwei, drei oder vier Cluster. Auch hier sieht man, dass Ergebnisse mit internen Validierungsindizes gewissermaßen übereinstimmen. Calinski-Harabasz Index und Average Silhouette Width waren für $k = 2$ Cluster maximal, wobei Davies-Bouldin Index für drei Cluster minimal wurde.

Im Fall des Average Linkage Algorithmus sind auch Ergebnisse der internen Validierungsindizes nicht eindeutig interpretierbar, da wir in Abbildung 4.5 sehen können, dass die Werte für alle Anzahlen von Clustern $k = 3, ..., 10$ in etwa gleich groß für beide Indizes sind. Bei zwei Clustern ist jedoch der Wert vom Jaccard Index am kleinsten und ARI nimmt sogar einen leicht negativen Wert an. Genau Werte für die Indizes sind aus Tabelle A.9 zu entnehmen.

Ergebnisse vom spektralen Clustering stimmen am meisten bei drei Clustern laut den beiden externen Validierungsindizes überein. Sowie Jaccard Index als auch ARI nehmen für $k = 3$ Clustern ihre maximalen Werte von ungefähr 0.88 an. Ergebnisse für interne Vali-

Abbildung 4.4: Average Silhouette Width

dierungsindizes waren für zwei und drei Cluster am besten, wie es aus oberen Abbildungen zu entnehmen ist. In diesem Fall ist es etwas überraschend, dass die Ergebnisse für zwei Cluster deutlich weniger übereinstimmen, vor allem liefert Adjustierter Rand Index einen ersichtlich niedrigeren Wert von 0.59.

	Calinski-Harabasz	Davies-Bouldin	Dunn	Silhouette
K-means	0.99998893	0.99492905	0.73728853	0.99939261
Complete Linkage	0.99633344	0.90541478	0.95654027	0.98639493
Average Linkage	-0.15742780	-0.20721416	NA	0.88334083
Spektrales Clus.	0.87611757	0.98760686	-0.40734129	0.98596622

Tabelle 4.1: Korrelationskoeffizienten zwischen internen Validierungsindizes (berechnet auf dem Trainings- und Testdatensatz)

Die nächste für uns interessante Frage wäre, wie gut interne Indizes übereinstimmen oder anders gesagt, ob es einen (linearen) Zusammenhang zwischen den einzelnen internen Validierungsindizes für Trainingsdaten und Testdaten gibt. Dafür wird der Korrelationskoeffizient nach Bravais-Pearson (Fahrmeir et al., 2016, Kapitel 3) berechnet. Die Ergebnisse sind in Tabelle 4.1 nachzusehen. Hierbei sehen wir, dass Calinski-Harabasz Index, Davies-Bouldin Index, Dunn Index sowie Average Silhouette Width für k-Means Algorithmus sowie für Complete Linkage Verfahren einen starken positiven Zusammenhang haben. Also steigen die Indizes auf dem Trainingsdatensatz, so werden sie auch auf dem Testdatensatz größer, also können wir davon ausgehen, dass gelieferte Werte von den Indizes nicht zufällig sind und somit kein Data-Dredging-Effekt vorliegt. Anders sieht es für Average Linkage Verfahren aus, da die Korrelationskoeffizienten für verschiedene Indizes sowie stark positive als auch leicht negative Werte annehmen. Für Dunn Index wird kein Wert für den Korrelationskoeffizienten geliefert, da laut der Warnung, die vom Software **R** bei der Berechnung ausgegeben wurde, eine Standardabweichung von Null vorliegt und somit der Koeffizient nicht berechnet werden kann. Da es keinen stark positiven Zusam-

Abbildung 4.5: Externe Validierungsindizes: Jaccard Index (oben) und Adjustierter Rand Index (unten)

menhang zwischen den Ergebnissen interner Validierungsindizes gibt, deutet es nochmal darauf hin, dass Ergebnisse, die Average Linkage Algorithmus für den Datensatz liefert, nicht wirklich sinnvoll zu sein scheinen. Beim spektralen Clustering sehen wir einen starken positiven Zusammenhang zwischen Calinski-Harabasz Indizes auf beiden Teildatensätzen sowie zwischen Davies-Bouldin und Silhouette Width Indizes. Somit können wir sehen, dass Calinski-Harabasz Index, Davies-Bouldin Index und Average Silhouette Width als drei besonders stabile interne Validierungsindizes gemäß der Analyse des im Rahmen dieser Arbeit verwendeten Datensatzes ausgezeichnet werden können.

	Calinski-Harabasz	Davies-Bouldin	Dunn	Silhouette
K-means	0.76996417	-0.70792480	-0.74167305	0.68184923
Complete Linkage	0.54841736	-0.66122482	-0.54963256	0.76625044
Average Linkage	0.56886563	0.83928682	-0.97859633	-0.31616044
Spektrales Clus.	0.61341368	0.12699897	-0.59254925	0.40003892

Tabelle 4.2: Korrelationskoeffizienten zwischen internen Validierungsindizes (berechnet auf dem Trainingsdatensatz) und Adjustierten Rand Indizes

In Tabelle 4.2 werden Korrelationskoeffizienten zwischen internen Validierungsindizes auf dem Trainingsdatensatz und dem Adjustierten Rand Index dargestellt. Wenn kein Data-Dredging-Effekt vorliegt, das heißt wenn gute Ergebnisse von internen Validierungsindizes nicht zufällig sind und nicht nur durch mehrfaches Ausprobieren verschiedener Konstellationen zustande kommen, würde man erwarten, dass es starke Korrelationen zwischen internen Validierungsindizes und dem Adjustierten Rand Index gibt. In einem solchen Fall wird also erwartet, dass gute Clusteringergebnisse auf den Trainings- und auf den Testdaten übereinstimmen und somit Adjustierter Rand Index Werte nahe bei 1 annimmt. Dabei wäre es wichtig anzumerken, dass wir an Maximierung von Calinski-Harabasz Index, Dunn Index sowie Silhouette Index interessiert sind, bei Davies-Bouldin Index erwartet man hingegen kleine Werte bei einem guten Clusteringergebnis. Also gehen wir im Idealfall davon

aus, dass bei den drei oben genannten Indizes starke positive Korrelationen nachgewiesen werden können und bei dem Davies-Bouldin Index starke negative Korrelationen vorliegen. Wie wir sehen können, gibt es einen mittleren positiven Zusammenhang zwischen dem Calinski-Harabasz Index und dem Adjustierten Rand Index, ebenso wie erwartet sind Silhouette Width und ARI positiv korreliert, wobei die einzelne negative Korrelation für den Average Linkage Algorithmus durch die fragwürdigen Ergebnisse (wenig sinnvolle Clustergrößen) erklärbar sein könnte. Ähnlich sieht die Situation bei dem Davies-Bouldin Index aus, wobei die Korrelation für Average Linkage Verfahren wieder auf einen gegenläufigen Trend hindeutet. Die meisten Fragen entstehen jedoch bei den Ergebnissen vom Dunn Index, da dieser in allen vier Fällen eine mittlere bis auf eine starke negative Korrelation mit dem ARI aufweist. Dies stützt die oben erwähnte Vermutung, dass Dunn Index für die Analyse der Genexpressionsdaten weniger geeignet ist und seine Anwendung in diesem Bereich genauer untersucht werden soll.

4.2 Mehrfache Aufteilung des Datensatzes

Um die Ergebnisse der internen Validierungsindizes besser evaluieren zu können, wird der Datensatz mehrfach zufällig in Trainings- und Testdatensatz aufgeteilt. Im Rahmen dieser Arbeit wurde die Anzahl der Wiederholungen auf $n = 10$ festgelegt, wobei die Größenverhältnisse von Trainings- und Testdatensätzen für jeden Split mit 80%/20% gleich bleiben. Da Average Linkage Verfahren ähnlich wie für den Fall mit der einfachen Aufteilung des Datensatzes keine vernünftigen Clusteringergebnisse für den Datensatz lieferte, wurde der Algorithmus von weiteren Analysen ausgeschlossen. Somit werden interne Validierungsindizes in diesem Abschnitt nur für K-means Algorithmus, Complete Linkage Algorithmus und spektrales Clustering untersucht.

Zuerst wird untersucht, wie sich einzelne interne Validierungsindizes für Trainings- und Testdaten unterscheiden sowie welche Kombinationen von Algorithmen und Inputparametern die besten Ergebnisse liefern. Für mehrfache Aufteilung des Datensatzes wurden solche Indizes für jede uns interessierende Kombination und jede Partition des Datensatzes einzeln berechnet, wobei wir uns im Folgenden für mittlere Werte der Validierungsindizes (gemittelt über alle 10 Splits) interessieren.

Mittlere Werte vom *Calinski-Harabasz Index*, die in Abbildung 4.6 zu sehen sind, liefern insgesamt sehr gute Ergebnisse, da für den K-means Algorithmus und für den Complete Linkage Verfahren mittlere Werte des Index sich für alle Anzahlen der Cluster ähnlich verhalten und maximale Werte bei $k = 2$ Clustern erreicht werden. Beim spektralen Clustering wird der mittlere Index für $k = 3$ Cluster auf dem Trainingsdatensatz maximal, wobei auf dem Testdatensatz der maximale Wert für zwei Cluster vorliegt. Da aber die Indexwerte auf den Trainingsdaten für zwei und drei Cluster recht ähnlich sind, sollte dies eher kein Grund für einen Verdacht auf Data-Dredging sein. Der Index bleibt auch bei mehrfacher Aufteilung des Datensatzes stabil und zuverlässig. Insgesamt nimmt der Calinski-Harabasz Index seinen größten Wert für K-means Clustering mit zwei Clustern an, also sollte genau die Kombination von Algorithmus und Inputparameter gegenüber allen anderen bevorzugt werden.

Auch *Davies-Bouldin Index* liefert sehr ähnliche Ergebnisse auf beiden Teildatensätzen (Abbildung 4.7). Der mittlere Index wird minimal für Clustering mit zwei Gruppen bei der Ergebnissen von allen drei Algorithmen und der Trend bleibt für beide Teildatensätze der gleiche. Der Unterschied zu den Ergebnissen von Calinski-Harabasz Index liegt daran, dass der über alle Clustering-Methoden hinweg beste Wert vom Davies-Bouldin Index für

Abbildung 4.6: Mittlerer Calinski-Harabasz Index

Complete Linkage Verfahren angenommen wird. Die Werte des Index bei K-means und Complete Linkage Verfahren unterscheiden sich aber für zwei Cluster nur marginal.

Im Gegensatz zu den ersten zwei internen Validierungsindizes liefert der *Dunn Index* leider auch bei mehreren Splits keine zuverlässigen Ergebnisse. Wie man aus Abbildung 4.8 sofort erkennt, wird der Maximum für verschiedene Algorithmen bei ganz unterschiedlichen Anzahlen an Cluster angenommen und auch für die einzelnen Clustering-Algorithmen stimmen die Ergebnisse vom mittleren Dunn Index auf dem Trainings- und Testdatensatz nicht überein. Insgesamt wird der Index bei Complete Linkage Clustering mit zwei Clustern maximal, jedoch unterscheiden sich die Werte vom Dunn Index sehr stark für verschiedene Algorithmen. Zudem liegen Ergebnisse von diesem Index mit den etwas mehr eindeutigen Ergebnissen von anderen ausgewählten Validierungsindizes sehr weit auseinander. Somit sehen wir auch bei mehrfacher Aufteilung des Datensatzes, dass der Dunn Index als Validierungsmethode für Clustering von Genexpressionsdaten schlecht geeignet ist.

Mittlerer *Silhouette Width* (Abbildung 4.9) liefert eindeutige Ergebnisse, da das Verhalten des mittleren Index über für alle drei Clustering-Verfahren gleich bleibt und die Werte maximal für $k = 2$ Cluster werden. Über die drei Algorithmen hinweg wird Average Silhouette Width auf beiden Teildatensätzen für Complete Link mit zwei Clustern maximal. Auch hier sind aber die entsprechenden Werte für K-means Algorithmus nur etwas geringer. Jedoch muss man darauf hinweisen, dass maximale Werte für Average Silhouette Width für den von uns betrachteten Datensatz bei ungefähr 0.3 liegen, was gleichzeitig heißt, dass Objekte der einzelnen Cluster nicht äußerst homogen sind und die Cluster an sich nicht sehr heterogen zueinander stehen.

Wir interessieren uns auch für mittlere Werte der externen Validierungsindizes wie Jaccard Index und Adjustierter Rand Index. Kurz zusammengefasst kann man sagen, dass die Übereinstimmung der Ergebnisse für Trainings- und Testdaten bei zwei und drei Clustern

Abbildung 4.7: Mittlerer Davies-Bouldin Index

für alle Clustering-Algorithmen am besten ist, wie es aus Abbildung 4.10 zu erkennen ist. Es heißt, dass bei den Cluster Ergebnissen, die laut den meisten internen Validierungsindizes bevorzugt werden sollten, die Zuordnungen einzelner Datenobjekte am besten übereinstimmen, somit also kein Verdacht auf Zufälligkeit der Ergebnisse vorliegt.

	Calinski-Harabasz	Davies-Bouldin	Dunn	Silhouette
K-means	0.99989	0.98887	0.55631	0.99842
Complete Linkage	0.93133	0.87831	0.90930	0.94814
Spektrales Clus.	0.89454	0.79001	0.28675	0.98141

Tabelle 4.3: Mittlere Korrelationskoeffizienten zwischen internen Validierungsindizes (berechnet auf dem Trainings- und Testdatensatz)

Im weiteren Verlauf der Analyse untersuchen wir mittlere Korrelationskoeffizienten, die Ergebnisse sind in Tabelle 4.3 dargestellt. Es wurden Korrelationen zwischen einzelnen internen Validierungsindizes auf den Trainings- und den Testdaten für alle drei Clustering-Algorithmen und für jede Datenaufteilung berechnet und die Werte für einzelne Splits wurden gemittelt. Die Überlegung, die dahinter steckt, ist dass Kombinationen von Algorithmen und Anzahlen an Clustern, die gute Ergebnisse auf Trainingsdaten aufweisen, ebenfalls auch gute Werte für Testdaten liefern sollten, und umgekehrt. Es wäre also zu erwarten, dass es einen starken positiven Zusammenhang zwischen den einzelnen Validierungsindizes gibt. Anders ausgedrückt wird somit analysiert, ob die Verläufe einzelner Kurven von den oben dargestellten Grafiken gleichen Trend haben. Wie man sehen kann, sind Calinski-Harabasz Index und Silhouette Width weit vorne bei den Werten von mittleren Korrelationskoeffizienten, Davies-Bouldin Index und Dunn Index haben zwar etwas niedrigere Werte, die aber immer noch auf einen starken linearen Zusammenhang hindeuten. Die Ausnahme stellt jedoch der Dunn Index für die Ergebnisse des spektralen Clusterings mit dem mittleren Korrelationswert von 0.28675 dar. Dabei muss man aber aufpassen, dass selbst stark positive mittlere Korrelation keinen Auskunft darüber gibt,

Abbildung 4.8: Mittlerer Dunn Index

ob die Indizes gleiche Ergebnisse für alle drei Algorithmen liefern, ein gutes Beispiel dafür ist die Problematik von Dunn Index, die aus Abbildung 4.8 klar erkennbar wird.

	Calinski-Harabasz	Davies-Bouldin	Dunn	Silhouette
K-means	0.75048	-0.71502	-0.48020	0.66761
Complete Linkage	0.45561	-0.48580	-0.40524	0.46744
Spektrales Clus.	0.65098	-0.01452	-0.52238	0.45718

Tabelle 4.4: Mittlere Korrelationskoeffizienten zwischen internen Validierungsindizes (berechnet auf dem Trainingsdatensatz) und Adjustierten Rand Indizes

Analoge Überlegung gilt auch für die Korrelationen zwischen den internen Validierungsindizes auf dem Trainingsdatensatz und dem Adjustierten Rand Index. Wie in ausführlich Abschnitt 4.1 beschrieben wurde, erwarten wir betragsmäßig große Werte für Korrelationen im Fall wenn interne Validierungsindizes aussagekräftig sind und kein Data-Dredging-Effekt vorliegt. Wir erwarten dabei große Werte für Calinski-Harabasz Index, Dunn Indes und Average Silhouette Width und kleine Werte für Davies-Bouldin Index. Die Ergebnisse werden in Tabelle 4.4 dargestellt. Wie wir sehen können, bleiben Calinski-Harabasz Index sowie Silhouette Width relativ stabil mit mittleren bis auf starken positiven Werten von Korrelationskoeffizienten bei allen drei Algorithmen. Der Dunn Index weist wie im Fall von einfacher Aufteilung des Datensatzes negative Korrelationen auf, somit können wir auch diesem Fall davon ausgehen, dass der Index bei der Analyse von Genexpressionsdaten nicht aussagekräftig ist. Ebenso fraglich ist der mittlere Korrelationskoeffizient von Davies-Bouldin Index beim spektralen Clustering, da der Wert nahe bei 0 besagt, dass es keinen Zusammenhang zwischen den Ergebnissen von diesem Index und der Übereinstimmung der Clusteringergebnisse auf dem Trainings- und dem Testdatensatz gibt.

Die letzte Frage, die im Rahmen dieser Arbeit analysiert wurde, ist welche Werte im Mittel der Adjustierte Rand Index für die Clusteringergebnisse annimmt, die die besten Werte

Abbildung 4.9: Mittlerer Average Silhouette Width

für einzelne internen Validierungsindizes auf dem Trainingsdatensatz aufweisen. Es wird also für jeden Split die Konstellation von Clustering-Algorithmus und Anzahl von Clustern rausgesucht, für die der ausgewählte interne Validierungsindex, beispielsweise Silhouette Width, den größten Wert annimmt. Weiterhin wird für jede der Kombinationen der Adjustierte Rand Index betrachtet, (also wie gut die Ergebnisse auf dem Trainingsdatensatz mit denen auf dem Testdatensatz übereinstimmen), und anschließend wird der mittlere Adjustierte Rand Index über alle Splits hinweg ermittelt. Zu erwarten dabei wäre, dass der mittlere ARI eher große Werte annimmt, da für "gute" Kombinationen von Algorithmus und Anzahl an Cluster die Ergebnisse auf den Trainingsdaten und auf den Testdaten übereinstimmen sollten und somit der ARI nahe bei 1 wäre. Solche Analyse wurde für zwei interne Validierungsindizes - den Calinski-Harabasz Index und den Silhouette Width - durchgeführt. Die Ergebnisse werden in Tabelle A.22 und Tabelle A.23 dargestellt. Mittlerer ARI für Average Silhouette Width nimmt den Wert von 0.64777 an. Man muss anmerken, dass die Werte für ARI bei einzelnen Splits sehr stark variieren und somit Werte zwischen 0.2 und 0.9 annehmen. Dadurch wird auch der nicht besonders hohe Werte für Adjustierten Rand Index erklärt. Bei dem Calinski-Harabasz Index schneidet der mittlere ARI deutlich besser ab und nimmt den mittleren Wert von 0.8977 an. Eine wichtige Besonderheit ist auch, dass der Calinski-Harabasz Index seinen größten Wert bei allen zehn Splits für K-means Algorithmus und zwei Cluster annimmt.

Abbildung 4.10: Mittlere externe Validierungsindizes: Jaccard Index (oben) und Adjustierter Rand Index (unten)

5. Fazit

Clustervalidierung stellt ein großes Problem in Clustering dar, da die Wahl der "richtigen" Kombination von Algorithmus und Inputparameter oft nicht einfach ist, da die wahre Gruppenzugehörigkeit der Datenobjekte unbekannt ist. Im Fokus dieser Arbeit lag Untersuchung von Stabilität ausgewählter interner Validierungsindizes. Für diesen Zweck wurde der Datensatz mehrfach zufällig in Trainingsdatensatz und Testdatensatz aufgeteilt und Ergebnisse der Validierungsindizes wurden auf den beiden Teildatensätzen untereinander verglichen.

Somit wurde festgestellt, dass Calinski-Harabasz Index und Average Silhouette Width für verschiedene Splits sehr ähnliche Ergebnisse lieferten, was auf Stabilität dieser Indizes hindeutet. Der Dunn Index zeichnete sich durch recht instabile Ergebnisse für mehrere Aufteilungen des Datensatzes aus, somit sollte für weitere Genexpressionsdatensätze geprüft werden, ob die Verwendung von Dunn Index als Validierungsmethode für Clustering von Genexpressionsdaten sinnvoll ist.

Weiterhin haben wir gesehen, dass Single Linkage sowie Average Linkage Verfahren für unseren Datensatz keine sinnvollen Clusteringergebnisse liefern, da bei der Mehrheit von Fällen Cluster mit jeweils einem Element entstanden sind, wobei alle anderen Datenobjekte einem Cluster zugeordnet wurden. Eine Analyse von solchen Partitionen ist nicht wirklich sinnvoll, deswegen wurden für diese zwei Verfahren interne Validierungsindizes nicht untersucht.

Eine weitere Klasse der internen Validierungsindizes bilden sogenannte Stabilitätsindizes, wobei *Figure of Merit* (FOM) ein in der Analyse der Genexpressionsdaten besonders verbreiteter Index ist. Die Idee von Figure of Merit liegt daran, den durchschnittlichen Abstand jedes Datenobjektes zu seinem Cluster Schwerpunkt auszurechnen, nachdem einzelne Variablen (im Fall unseres Datensatzes die Patienten) aus dem Datensatz entfernt wurden. Dabei wird die Methode bevorzugt, die kleinere Werte von FOM liefert. Das Problem von Stabilitätsindizes liegt daran, dass der Index mit der wachsenden Anzahl an Cluster tendenziell sinkt und somit das Minimum immer bei größeren Werten von k liegt. Somit wird in Yeung et al. (2001) darauf hingewiesen, dass diese Statistik nur für relative Vergleiche der Clusteringergebnisse von verschiedenen Algorithmen für ein festes k verwendet werden kann. Aus diesem Grund wurde das Figure of Merit im Rahmen dieser Arbeit nicht untersucht. Im Allgemeinen wäre die Analyse von Stabilitätsindizes für deskriptives Clustering von Genexpressionsdaten vom großen Interesse. Dabei sollte aber ein weiteres inhaltliches Kriterium überlegt werden, das einen Vergleich der Werte von Stabilitätsindizes für unterschiedliche Werte von Anzahl der Cluster k ermöglichen würde. Ein mögliches Kriterium wäre dabei ein fest gesetztes Threshold für relative Veränderung des Index für steigendes k.

A. Tabellen

Dieser Anhang enthält Tabellen mit den Ergebnissen sämtlicher Analysen, die im Rahmen dieser Arbeit durchgeführt wurden. Die Ergebnisse können mit Hilfe von den Daten sowie des **R**-Codes, die sich in dem dieser Arbeit beigefügten elektronischen Anhang befinden, reproduziert werden.

A.1 Einfache Berechnungen

A.1.1 K-means

# Cluster	Calinski-Harabasz	Davies-Bouldin	Dunn	Silhouette Width
2	1075.49839	1.28391	0.12467	0.30004
3	773.55223	1.74172	0.10056	0.17669
4	591.85848	2.07022	0.15030	0.12813
5	480.79931	2.26496	0.14883	0.11421
6	411.84989	2.34933	0.16248	0.10039
7	362.33327	2.26800	0.16395	0.09257
8	322.42765	2.50358	0.16395	0.08048
9	290.79010	2.43954	0.16904	0.07771
10	265.55742	2.51038	0.17435	0.07328

Tabelle A.1: Interne Validierungsindizes für den Trainingsdatensatz

# Cluster	Calinski-Harabasz	Davies-Bouldin	Dunn	Silhouette Width
2	1067.95971	1.28742	0.12847	0.29951
3	766.33256	1.75630	0.10889	0.17479
4	589.09762	2.05417	0.15046	0.12932
5	481.23334	2.23430	0.10930	0.11912
6	411.47018	2.32413	0.17368	0.10250
7	361.82812	2.36770	0.16020	0.09040
8	321.35745	2.48475	0.14055	0.07998
9	290.56502	2.40677	0.15420	0.07740
10	266.49244	2.51584	0.15420	0.07796

Tabelle A.2: Interne Validierungsindizes für den Testdatensatz

# Cluster	Jaccard	ARI
2	0.94140	0.93234
3	0.89105	0.90929
4	0.85691	0.89091
5	0.80563	0.86105
6	0.78144	0.84783
7	0.60368	0.70271
8	0.68113	0.77952
9	0.65569	0.76034
10	0.46597	0.58904

Tabelle A.3: Externe Validierungsindizes

A.1.2 Complete Linkage

# Cluster	Calinski-Harabasz	Davies-Bouldin	Dunn	Silhouette Width
2	999.58320	1.34646	0.11571	0.27955
3	571.30328	1.27530	0.14356	0.23458
4	436.45423	1.83392	0.16375	0.19078
5	401.39202	2.19253	0.16492	0.08262
6	347.65481	2.49058	0.16924	0.06980
7	295.20167	2.50771	0.17277	0.06850
8	262.54957	2.75880	0.17471	0.06172
9	231.23132	2.67627	0.18391	0.06105
10	206.18277	2.44394	0.18657	0.06119

Tabelle A.4: Interne Validierungsindizes für den Trainingsdatensatz

# Cluster	Calinski-Harabasz	Davies-Bouldin	Dunn	Silhouette Width
2	977.02745	1.30145	0.12393	0.29243
3	531.69518	1.24747	0.14492	0.25066
4	411.13748	1.76439	0.15132	0.19288
5	338.40730	1.99446	0.17198	0.11438
6	340.38374	2.28839	0.17571	0.08155
7	294.10854	2.24078	0.17583	0.08146
8	258.33378	2.77575	0.18273	0.06895
9	234.11895	3.02784	0.18393	0.04517
10	209.24832	3.10936	0.18782	0.04238

Tabelle A.5: Interne Validierungsindizes für den Testdatensatz

# Cluster	Jaccard	ARI
2	0.73127	0.64626
3	0.72514	0.64407
4	0.71695	0.65657
5	0.41001	0.34430
6	0.48077	0.52672
7	0.47878	0.52550
8	0.42247	0.47623
9	0.43848	0.50833
10	0.43791	0.50795

Tabelle A.6: Externe Validierungsindizes

A.1.3 Average Linkage

# Cluster	Calinski-Harabasz	Davies-Bouldin	Dunn	Silhouette Width
2	2.42512	0.61043	0.36349	0.27089
3	500.16116	1.01164	0.12990	0.30283
4	372.56346	1.12150	0.17278	0.25883
5	280.36042	1.01772	0.17278	0.25583
6	225.66047	0.98552	0.17278	0.25460
7	188.60413	0.93309	0.17278	0.19953
8	165.64516	1.22974	0.17278	0.18178
9	145.32890	1.17181	0.17278	0.17329
10	129.72076	1.19133	0.17278	0.16975

Tabelle A.7: Interne Validierungsindizes für den Trainingsdatensatz

# Cluster	Calinski-Harabasz	Davies-Bouldin	Dunn	Silhouette Width
1	854.76177	1.18411	0.15944	0.31851
2	429.43307	0.97134	0.15944	0.27991
3	287.27365	0.88661	0.15944	0.22615
4	216.15747	0.84151	0.15944	0.20451
5	174.02182	0.86499	0.15944	0.19851
6	145.46197	0.83618	0.15944	0.16848
7	125.20734	0.96315	0.15944	0.15747
8	110.54698	1.12354	0.15944	0.14243
9	98.55303	1.07917	0.15944	0.13753

Tabelle A.8: Interne Validierungsindizes für den Testdatensatz

# Cluster	Jaccard	ARI
2	0.69137	-0.00076
3	0.79434	0.67554
4	0.78745	0.66911
5	0.78751	0.66942
6	0.78787	0.67022
7	0.78775	0.67070
8	0.78648	0.67114
9	0.78624	0.67265
10	0.78673	0.67456

Tabelle A.9: Externe Validierungsindizes

A.1.4 Spektrales Clustering

# Cluster	Calinski-Harabasz	Davies-Bouldin	Dunn	Silhouette Width
2	490.93573	1.65931	0.14348	0.22687
3	512.16572	2.02893	0.07476	0.16281
4	451.19152	2.29758	0.11231	0.12145
5	395.79979	2.48658	0.07476	0.09489
6	309.00340	5.63551	0.10158	0.08202
7	279.01643	2.84020	0.10058	0.05243
8	272.12075	2.61241	0.11622	0.04947
9	244.71827	2.75301	0.13112	0.03695
10	214.50311	2.92600	0.09112	0.04916

Tabelle A.10: Interne Validierungsindizes für den Trainingsdatensatz

# Cluster	Calinski-Harabasz	Davies-Bouldin	Dunn	Silhouette Width
2	768.04520	1.48810	0.11153	0.25121
3	512.83408	1.90931	0.15546	0.15000
4	439.79106	2.34724	0.10520	0.12210
5	395.69238	2.42087	0.10250	0.10040
6	307.69643	5.03613	0.09334	0.08705
7	285.99640	2.66320	0.10520	0.06575
8	266.33056	2.66421	0.09392	0.03853
9	227.75152	2.93206	0.10947	0.04604
10	235.53496	2.68745	0.16906	0.04247

Tabelle A.11: Interne Validierungsindizes für den Testdatensatz

# Cluster	Jaccard	ARI
2	0.72907	0.59161
3	0.88195	0.88100
4	0.83356	0.86816
5	0.75066	0.81236
6	0.73575	0.80134
7	0.42537	0.51558
8	0.38406	0.47472
9	0.39897	0.50245
10	0.52870	0.64982

Tabelle A.12: Externe Validierungsindizes

A.2 Mehrfache Berechnungen

A.2.1 K-means

# Cluster	Calinski-Harabasz	Davies-Bouldin	Dunn	Silhouette Width
2	1071.95154	1.28790	0.12252	0.29907
3	770.02880	1.73911	0.10987	0.17681
4	589.81662	2.06870	0.15596	0.12839
5	479.58576	2.26258	0.14347	0.11464
6	410.66436	2.34267	0.16298	0.09958
7	361.10795	2.33156	0.15593	0.08985
8	321.20162	2.50737	0.16240	0.07988
9	289.78438	2.45596	0.14955	0.07755
10	264.87291	2.49022	0.15369	0.07373

Tabelle A.13: Mittlere interne Validierungsindizes für den Trainingsdatensatz

# Cluster	Calinski-Harabasz	Davies-Bouldin	Dunn	Silhouette Width
2	1087.37836	1.28075	0.11289	0.30032
3	784.10926	1.71542	0.10867	0.18022
4	599.84928	2.07080	0.13952	0.12785
5	488.19245	2.24701	0.14509	0.11481
6	417.67465	2.33458	0.15698	0.09867
7	367.60789	2.31912	0.14816	0.09121
8	327.25780	2.47197	0.13965	0.08093
9	295.35328	2.47644	0.14593	0.07515
10	269.59868	2.51272	0.15422	0.07296

Tabelle A.14: Mittlere interne Validierungsindizes für den Testdatensatz

# Cluster	Jaccard	ARI
2	0.91245	0.89770
3	0.86228	0.88248
4	0.81436	0.85421
5	0.70669	0.77207
6	0.68717	0.76619
7	0.62552	0.71950
8	0.56624	0.65755
9	0.47441	0.57650
10	0.42358	0.53338

Tabelle A.15: Mittlere externe Validierungsindizes

A.2.2 Complete Linkage

# Cluster	Calinski-Harabasz	Davies-Bouldin	Dunn	Silhouette Width
2	808.82762	1.15573	0.14086	0.31932
3	577.51068	1.60205	0.15607	0.21514
4	434.54962	1.86484	0.17265	0.17121
5	381.13467	2.14162	0.17992	0.11386
6	319.68571	2.27687	0.18610	0.08628
7	284.78210	2.52825	0.18955	0.07104
8	255.86335	2.53616	0.19311	0.06785
9	230.28531	2.46578	0.19659	0.06519
10	209.59778	2.49392	0.19841	0.06235

Tabelle A.16: Mittlere interne Validierungsindizes für den Trainingsdatensatz

# Cluster	Calinski-Harabasz	Davies-Bouldin	Dunn	Silhouette Width
2	887.80434	1.21876	0.13200	0.30620
3	594.53385	1.48516	0.15099	0.22127
4	468.03219	1.96509	0.16085	0.14980
5	387.67074	2.32439	0.16700	0.11464
6	329.48149	2.43352	0.17262	0.08214
7	287.58599	2.51028	0.17369	0.07239
8	257.89951	2.54172	0.17738	0.07384
9	238.04138	2.61787	0.18075	0.06472
10	216.37693	2.54670	0.18640	0.06105

Tabelle A.17: Mittlere interne Validierungsindizes für den Testdatensatz

# Cluster	Jaccard	ARI
2	0.73681	0.53331
3	0.69358	0.59511
4	0.63864	0.57820
5	0.48116	0.43935
6	0.41757	0.38564
7	0.37207	0.36102
8	0.38268	0.39423
9	0.38480	0.42026
10	0.38317	0.42318

Tabelle A.18: Mittlere externe Validierungsindizes

A.2.3 Spektrales Clustering

# Cluster	Calinski-Harabasz	Davies-Bouldin	Dunn	Silhouette Width
2	508.13972	1.64333	0.12352	0.22902
3	512.32429	2.00433	0.07814	0.15940
4	446.94697	2.32642	0.10821	0.12204
5	394.99255	2.49378	0.07798	0.09532
6	310.84047	6.03936	0.09904	0.07740
7	278.96445	2.80574	0.10830	0.05423
8	251.72023	3.19877	0.10799	0.04706
9	230.97833	2.91545	0.10311	0.04276
10	215.89558	2.86539	0.09819	0.04723

Tabelle A.19: Mittlere interne Validierungsindizes für den Trainingsdatensatz

# Cluster	Calinski-Harabasz	Davies-Bouldin	Dunn	Silhouette Width
2	762.39729	1.45765	0.10943	0.25997
3	523.54508	1.96441	0.09059	0.16134
4	455.49628	2.28639	0.11044	0.12333
5	409.21816	2.39410	0.09680	0.09865
6	320.08564	4.99986	0.08707	0.07873
7	275.38058	4.33186	0.09736	0.05847
8	260.60936	2.93759	0.10612	0.04648
9	238.99934	2.82799	0.10233	0.04173
10	220.74084	3.08206	0.10074	0.05028

Tabelle A.20: Mittlere interne Validierungsindizes für den Testdatensatz

# Cluster	Jaccard	ARI
2	0.77964	0.65874
3	0.89700	0.89775
4	0.77873	0.81783
5	0.72376	0.78842
6	0.70785	0.77609
7	0.41578	0.49741
8	0.39638	0.48654
9	0.47014	0.57810
10	0.51013	0.62658

Tabelle A.21: Mittlere externe Validierungsindizes

A.2.4 ARI-Werte für Kombinationen mit besten Ergebnissen von internen Validierungsindizes

Split	Algorithmus	# Cluster	Silhouette (train)	ARI
1	Complete Link	2	0.33828	0.32679
2	Complete Link	2	0.32512	0.20077
3	Complete Link	2	0.34064	0.29725
4	Complete Link	2	0.32977	0.66244
5	K-means	2	0.30307	0.86533
6	K-means	2	0.29970	0.89380
7	K-means	2	0.29892	0.91489
8	Complete Link	2	0.33942	0.70866
9	Complete Link	2	0.31024	0.72979
10	Complete Link	2	0.32899	0.87800
				Mittlerer ARI = 0.64777

Tabelle A.22: Clustering mit bestem Average Silhouette Width für jede Aufteilung der Daten

Split	Algorithmus	# Cluster	Calinski-Harabasz (train)	ARI
1	K-means	2	1065.01087	0.90712
2	K-means	2	1062.76153	0.82641
3	K-means	2	1071.14714	0.92844
4	K-means	2	1067.05470	0.92069
5	K-means	2	1089.03708	0.86533
6	K-means	2	1063.07367	0.89380
7	K-means	2	1068.62174	0.91489
8	K-means	2	1080.30571	0.87308
9	K-means	2	1084.15166	0.92655
10	K-means	2	1068.35134	0.92071
				Mittlerer ARI = 0.89770

Tabelle A.23: Clustering mit bestem Calinski-Harabasz Index für jede Aufteilung der Daten

B. Implementierungen

In diesem Anhang sind Informationen über Implementierungen der Analysen sowie die dafür verwendeten Pakete enthalten. Der Code wurde mit Hilfe der freien statistischen Software **R** erstellt.

B.1 R-Pakete

Es wurden für die Analyse des im Kapitel 2 beschriebenen Datensatzes folgende Pakete benutzt:

- `cluster` (Maechler et al., 2019)
- `kknn` (Schliep and Hechenbichler, 2016)
- `clusterCrit` (Desgraupes, 2018)
- `mclust` (Scrucca et al., 2016)
- `clusteval` (Ramey, 2012)
- `reshape2` (Wickham, 2007)
- `ggplot2` (Wickham, 2016)

Somit sind alle Funktionen, die im Code verwendet wurden, in den oben aufgelisteten Paketen sowie im Standard-Paket `stats` (R Core Team, 2019) zu finden.

B.2 R-Code

Der **R**-Code ist in dem dieser Arbeit beigefügten elektronischen Anhang als Datei `code_BA.R` enthalten.

C. Elektronischer Anhang

Dieser Arbeit ist eine CD beigefügt, die unter anderem den für die Analyse verwendeten Datensatz sowie den **R**-Code enthält. Der Inhalt ist wie folgt strukturiert:

- `data_original`
  Der für die Analyse verwendete Datensatz (*log2-CPM* Werte wurden berechnet)
- `data_processed`
  Teildatensätze, die via `code_BA.R` erzeugt wurden
- `plots`
  Im Rahmen der Analyse mit Hilfe des **R**-Codes erzeugte Grafiken
- `code_BA.R`
  R-Code, der für die Analyse des Datensatzes verwendet wurde und sämtliche Ergebnisse reproduziert
- `BA_Holovchak.pdf`
  PDF-Version der Arbeit

Selbständigkeitserklärung

Hiermit bestätige ich, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Hilfsmittel benutzt habe. Die Stellen der Arbeit, die dem Wortlaut oder dem Sinn nach anderen Werken entnommen sind, wurden unter Angabe der Quelle kenntlich gemacht.

.....	.....
Ort, Datum	Unterschrift

Literaturverzeichnis

O. Arbelaitz, I. Gurrutxaga, J. Muguerza, J. Pérez, and I. Perona. An extensive comparative study of cluster validity indices. *Pattern Recognition*, 46:243–256, 01 2013. doi: 10.1016/j.patcog.2012.07.021.

G. Brock, V. Pihur, S. Datta, and S. Datta. clValid: An R package for cluster validation. *Journal of Statistical Software*, 25(4):1–22, 2008. URL `http://www.jstatsoft.org/v25/i04/`.

S. Datta and S. Datta. Comparisons and validation of statistical clustering techniques for microarray gene expression data. *Bioinformatics*, 19(4):459–466, 03 2003. ISSN 1367-4803. doi: 10.1093/bioinformatics/btg025. URL `https://doi.org/10.1093/bioinformatics/btg025`.

B. Desgraupes. *clusterCrit: Clustering Indices*, 2018. URL `https://CRAN.R-project.org/package=clusterCrit`. R package version 1.2.8.

L. Fahrmeir, A. Hamerle, and G. Tutz. *Multivariate statistische Verfahren* -. Walter de Gruyter, Berlin, 2. überarb. edition, 1996. ISBN 978-3-110-13806-1.

L. Fahrmeir, C. Heumann, R. Künstler, I. Pigeot, and G. Tutz. *Statistik: Der Weg zur Datenanalyse*. Springer-Lehrbuch. Springer Berlin Heidelberg, 2016. ISBN 9783662503720. URL `https://books.google.de/books?id=rKveDAAAQBAJ`.

J. Handl, J. Knowles, and D. B. Kell. Computational cluster validation in post-genomic data analysis. *Bioinformatics*, 21(15):3201–3212, 08 2005. ISSN 1367-4803. doi: 10.1093/bioinformatics/bti517. URL `https://doi.org/10.1093/bioinformatics/bti517`.

C. Hennig, M. Meila, F. Murtagh, and R. Rocci. *Handbook of Cluster Analysis*. Chapman and Hall/CRC, 2015.

W. Lingle, B. J. Erickson, M. L. Zuley, R. Jarosz, E. Bonaccio, J. Filippini, ..., and N. Gruszauskas. *Radiology Data from The Cancer Genome Atlas Breast Invasive Carcinoma [TCGA-BRCA] collection. The Cancer Imaging Archive*, 2016.

D. Liu and J. Graham. Simple measures of individual cluster-membership certainty for hard partitional clustering. *The American Statistician*, 73(1):70–79, 2019. doi: 10.1080/00031305.2018.1459315. URL `https://doi.org/10.1080/00031305.2018.1459315`.

M. Maechler, P. Rousseeuw, A. Struyf, M. Hubert, and K. Hornik. *cluster: Cluster Analysis Basics and Extensions*, 2019. R package version 2.0.8 — For new features, see the 'Changelog' file (in the package source).

S. Mohanty, K. Das, D. Mishra, and R. Ranjan. Cluster Validity Indices for Gene Expression Data. *International Journal ofComputer Trends and Technology*, 4(5):1465–1470, 05 2013. ISSN 2231-2803. URL `http://ijcttjournal.org/archives/ijctt-v4i5p95`.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2019. URL https://www.R-project.org/.

J. A. Ramey. *clusteval: Evaluation of Clustering Algorithms*, 2012. URL https://CRAN.R-project.org/package=clusteval. R package version 0.1.

K. Schliep and K. Hechenbichler. *kknn: Weighted k-Nearest Neighbors*, 2016. URL https://CRAN.R-project.org/package=kknn. R package version 1.3.1.

L. Scrucca, M. Fop, T. B. Murphy, and A. E. Raftery. mclust 5: clustering, classification and density estimation using Gaussian finite mixture models. *The R Journal*, 8(1):205–233, 2016. URL https://journal.r-project.org/archive/2016-1/scrucca-fop-murphy-etal.pdf.

H. Wickham. Reshaping data with the reshape package. *Journal of Statistical Software*, 21(12):1–20, 2007. URL http://www.jstatsoft.org/v21/i12/.

H. Wickham. *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York, 2016. ISBN 978-3-319-24277-4. URL https://ggplot2.tidyverse.org.

K. Y. Yeung, D. R. Haynor, and W. L. Ruzzo. Validating clustering for gene expression data. *Bioinformatics*, 17(4):309–318, 04 2001. ISSN 1367-4803. doi: 10.1093/bioinformatics/17.4.309. URL https://doi.org/10.1093/bioinformatics/17.4.309.