- LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN -

INSTITUT FÜR STATISTIK

Ein Regressionsmodell zur Untersuchung des Effektes von Datensatzcharakteristiken auf die relative Güte von Prädiktionsalgorithmen mit Anwendung auf 50 Microarray-Studien

BACHELORARBEIT

ZUR ERLANGUNG DES AKADEMISCHEN GRADES
BACHELOR OF SCIENCE (B.Sc.)

Autorin: Vera Völkl

Betreuerin und Gutachterin: Prof. Anne-Laure Boulesteix

München, den 26. Juli 2013

Abstract

Für die Diagnose und Behandlung von Krebs ist eine exakte Klassifikation von Tumoren essentiell. Die verhältnismäßig neue Technologie der Microarrays ermöglicht die simultane Messung tausender Genexpressionen. Mit ihrer Hilfe ist es möglich, komplexe Fragestellungen präziser zu beantworten. Microarrays führen zu umfangreichen Datensätzen mit $p >> n$. Klassische statistische Methoden sind für die Analyse solcher Daten meist ungeeignet. Es wurde eine Vielzahl an Methoden zur Klassifikation von Microarrays entwickelt, ein eindeutiger Favorit konnte allerdings noch nicht ausgemacht werden. Ziel dieser Arbeit ist es, in einem geeigneten Framework ein lineares Regressionsmodell zu formulieren, mit dem der Einfluss verschiedener Datensatzcharakteristiken auf die relative Güte von Prädiktionsalgorithmen untersucht werden kann. Nach der theoretischen Formulierung folgt eine praktische Anwendung des formulierten Regressionsmodells. Dafür liegen die Daten von 50 Microarray-Studien vor. Als Klassifikationsverfahren werden die lineare, die diagonale lineare sowie die quadratische Diskriminanzanalyse betrachtet.

II Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1 Einleitung

Die verhältnismäßig neue Technologie der Microarrays ermöglicht die simultane Messung tausender Genexpressionen. Durch diese Technik gewonnene, hochdimensionale Daten erlangen eine immer größere Bedeutung in der medizinischen Forschung. Mit ihrer Hilfe ist es möglich, komplexe Fragestellungen präziser zu beantworten. Microarray-Daten führen allerdings zu statistischen Herausforderungen. Die Datensätze sind mit einer immer größer werdenden Anzahl an Variablen sehr umfangreich. Aufgrund des hohen finanziellen Aufwandes stehen dieser nur relativ wenige Beobachtungen gegenüber. Dieses Problem wird mit $p >> n$ bezeichnet, wobei p für die Anzahl der Variablen und n für die Anzahl an Beobachtungen steht. Klassische statistische Methoden sind für die Analyse solcher Daten meist ungeeignet. Somit sind Microarrays nicht nur in der medizinischen, sondern auch in der statistischen Forschung ein aktuelles Thema.

Insbesondere in der Analyse von Tumordaten werden Microarrays verwendet. Anhand des Genexpressionsniveaus eines Patienten lassen sich Rückschlüsse auf einen Krankheitsbefall oder bösartige Veränderungen von Zellen ziehen. Dafür wurden verschiedene Prädiktionsalgorithmen entwickelt. Die Aufgabe von Prädiktionsalgorithmen besteht darin, das Genexpressionsniveau von Patienten, für die bereits eine Diagnose vorliegt, zu untersuchen und Entscheidungskriterien für die Klassifikation zukünftiger Patienten zu ermitteln.

Für die Diagnose und Behandlung von Krebs ist eine präzise Klassifikation von Tumoren essentiell. Aus diesem Grund ist es wichtig, verlässliche Prädiktionsalgorithmen zu identifizieren. Es wurde eine Vielzahl an Methoden zur Klassifikation von Microarrays entwickelt. In der heutigen Forschung hat sich allerdings noch kein eindeutiger Favorit für alle Arten von Datensätzen herauskristallisiert. Manche Methoden schneiden bei einer bestimmten Art von Datensätzen besser ab, manche bei einer anderen. Nun stellt sich die Frage, welche Datensatzcharakteristiken einen Einfluss auf die relative Güte von Prädiktionsalgorithmen haben.

Zur Beantwortung dieser Frage werden in der vorliegenden Arbeit 50 Microarray-Studien untersucht. Zunächst werden alle Individuen mit Hilfe des R Package CMA klassifiziert. Als Klassifikationsmethoden werden die lineare, die diagonale lineare sowie die quadratische Diskriminanzanalyse verwendet. In jeder Studie existieren zwei mögliche Gruppen. Diese Gruppen sind je nach Studie unterschiedlich definiert. Einige Studien beschäftigen sich mit dem momentanen Zustand des Patienten, andere mit längerfristigen Prognosen. Die wahre Klassenzugehörigkeit aller Patienten

ist bereits bekannt. Somit kann anschließend die Prädiktionsgenauigkeit der linearen, der diagonalen linearen und der quadratischen Diskriminanzanalyse für die 50 Microarray-Studien untersucht werden. Ist die Prädiktionsgenauigkeit bekannt, kann die eigentliche Fragestellung dieser Arbeit untersucht werden. Ziel ist es, in einem geeignetem Framework ein lineares Regressionsmodell zu formulieren, mit dem der Einfluss verschiedener Datensatzcharakteristiken auf die relative Güte von Prädiktionsalgorithmen untersucht werden kann. Als Response wird die jeweilige Differenz der geschätzten Prädiktionsfehler zweier Klassifikationsmethoden eingesetzt. Interessante Einflussgrößen sind beispielsweise die Anzahl an Beobachtungen und an Variablen eines Datensatzes.

Der Aufbau dieser Arbeit ist folgender: Im zweiten Kapitel wird die verwendete Methodik vorgestellt. Dazu zählen die verschiedenen Arten der Diskriminanzanalyse, Möglichkeiten der Variablenselektion sowie die Messung der relativen Güte mit Monte-Carlo-Kreuzvalidierung. Das dritte Kapitel beschäftigt sich mit dem theoretischen Hintergrund des Vergleichs zweier Methoden. Dazu werden Hypothesentests formuliert und die theoretischen Hintergründe des linearen Regressionsmodells erläutert. Im vierten Kapitel folgt die bereits beschriebene Anwendung auf 50 Microarray-Studien sowie die qualitative und quantitative Darstellung der Ergebnisse.

2 Methodik

Vor der Anwendung auf reale Daten werden alle dazu benötigten Methoden vorgestellt. Dazu gehören verschiedene Klassifikationsverfahren sowie ihre Prädiktionsfehler, Variablenselektion und Kreuzvalidierung.

2.1 Diskriminanzanalyse

Die Diskriminanzanalyse ist ein Verfahren aus der multivariaten Statistik. Hierbei geht man von einer Grundgesamtheit aus, die in k disjunkte Populationen mit Indikator $c \in \{1, ..., k\}$ zerfällt. Ziel ist es, ein Individuum i ($i \in \{1, ..., n\}$) mit unbekannter Klassenzugehörigkeit einer dieser Gruppen eindeutig zuzuordnen.

Dazu wird für jedes Individuum ein Merkmalsvektor x_i der Länge p erhoben. Seine Einträge sind die Ausprägungen von p beobachtbaren Variablen. Mit Hilfe dieses Merkmalsvektors kann auf das nicht beobachtbare c geschlossen werden. Die gezogene Stichprobe wird mit $s_0 = \{(x_1, c_1), ..., (x_n, c_n)\}$ bezeichnet. Prädiktor und Response folgen einer gemeinsamen Verteilung, die mit f bezeichnet wird. Für $k = 2$ (2-Klassen-Fall) spricht man von einer einfachen Diskriminanzanalyse, für $k > 2$ von einer multiplen. In der Praxis wird die einfache Diskriminanzanalyse am häufigsten benötigt.

Die Diskriminanzanalyse wird in unterschiedlichen Forschungsgebieten verwendet. Das klassische Anwendungsbeispiel ist die Überprüfung der Kreditwürdigkeit. Anhand der Kontodaten werden Kreditnehmer als bedenklich beziehungsweise unbedenklich eingestuft. In der Medizin wird die Diskriminanzanalyse zur frühzeitigen Diagnose und Prognose des Therapieerfolges eingesetzt. Marktforscher nutzen sie zur Einschätzung des Konsumverhaltens, Meteorologen zur Wettervorhersage.

In einem ersten Schritt wird die Entscheidungsregel $\delta(x)$ geschätzt:

$$\begin{aligned} \delta : \mathbb{R}^p &\longrightarrow \{1, ..., k\} \\ \mathrm{x} &\longmapsto \delta(x) \end{aligned}$$

Diese Entscheidungsregel klassifiziert Individuen mit unbekannter Gruppenzugehörigkeit. Sinnvoll ist es, die Daten dafür in einen Lerndatensatz $\mathcal{L}$ und einen Testdatensatz $\mathcal{T}$ aufzuteilen und $\delta(x)$ nur anhand der Lerndaten zu schätzen. Die Güte von $\delta(x)$ kann anschließend mittels der Trainingsdaten evaluiert werden (vgl. Abschnitt 2.3).

Fasst man x und c als Zufallsvariablen auf, so sind diese durch folgende relevante Größen charakterisiert:

- $p(r) = \mathbb{P}(c = r)$ a-priori-Wahrscheinlichkeit der Klasse r
- $\mathbb{P}(r \mid x) = \mathbb{P}(c = r \mid x)$ a-posteriori-Wahrscheinlichkeit der Klasse r
- $f(x \mid 1), ..., f(x \mid k)$ Verteilung der Merkmale, gegeben die Klasse
- $f(x) = f(x \mid 1)p(1) + ... + f(x \mid k)p(k)$ Mischverteilung der Population

[Fahrmeir et al. (1984), Leisch (2009), Wiesböck (1987)].

2.1.1 Bayes-Zuordnung

Eine mögliche Zuordnungsregel $\delta(x)$ ist die Bayes-Zuordnung. Sie teilt jedes Individuum in die Klasse mit der größten a-posteriori-Wahrscheinlichkeit ein und minimiert damit die Gesamtfehlerrate ε. Sie ist definiert als:

$$\delta(x) = r \Leftrightarrow \mathbb{P}(c = r|x) = \max_{j=1,...,k} \mathbb{P}(c = j|x). \tag{1}$$

Zu jeder Klasse r ist eine Diskriminanzfunktion $d_r(x)$ definiert:

$$d_r(x) = \mathbb{P}(c = r|x). \tag{2}$$

Um ein Individuum zu klassifizieren ist nicht die genaue Kenntnis von $\mathbb{P}(c = r|x)$ nötig. Es genügt zu wissen, welche Diskriminanzfunktion für x maximal ist. Somit ist jede monotone Transformation wie

$$d_r(x) = f(x|r) \cdot p(r) \tag{3}$$

oder

$$d_r(x) = log(f(x|r)) + log(p(r)) \tag{4}$$

äquivalent bezüglich der Zuordnung.

Die Klassifikation erfolgt über Differenzen. Sind $i, j \in c$ und $i \neq j$, so gilt:

$$\delta(x) = \begin{cases} i, d_i(x) - d_j(x) \geq 0 \\ j, d_i(x) - d_j(x) < 0 \end{cases}$$

Die a-posteriori-Wahrscheinlichkeiten $\mathbb{P}(c = r|x)$ können über den Satz von Bayes

bestimmt werden, falls $p(r)$ und $f(x \mid r)$ bekannt sind:

$$\mathbb{P}(c = r|x) = \frac{\mathbb{P}(x|c = r) \cdot \mathbb{P}(c = r)}{\sum_{j=1}^{k} \mathbb{P}(x|c = j) \cdot \mathbb{P}(c = j)} = \frac{f(x|r) \cdot \mathbb{P}(c = r)}{\sum_{j=1}^{k} f(x|j) \cdot \mathbb{P}(c = j)}. \tag{5}$$

In den meisten Fällen muss man allerdings von unbekanntem $p(r)$ und $f(x \mid r)$ ausgehen. Diese müssen im Voraus aus der Lernstichprobe geschätzt werden.

Für den Spezialfall $p(1) = ... = p(k)$ entspricht die Bayes-Zuordnung der Maximum-Likelihood-Zuordnung. Eine alternative Zuordnungsregel ist zum Beispiel die kostenoptimale Zuordnung. Die Diskriminanzanalyse ist ein (Bayes-)optimales Verfahren, falls die Merkmale, gegeben die Klasse, normalverteilt sind [Slawski et al. (2008), Fahrmeir et al. (1984), Völkl (2013)].

2.1.2 Quadratische Diskriminanzanalyse (QDA)

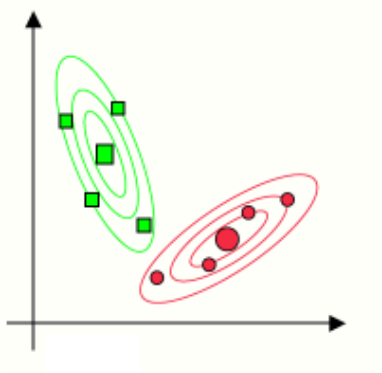

Abbildung 1: QDA
Quelle: Rahnenführer (2009)

In der quadratischen Diskriminanzanalyse (QDA) geht man von multivariat normalverteilten Klassendichten mit Erwartungswert μ_r und Kovarianzmatrix Σ_r aus.

$$x|c = r \sim N(\mu_r, \Sigma_r)$$

Für die Verteilung der Merkmale, gegeben der Klasse r, ergibt sich somit:

$$f(x|r) = \frac{1}{(2\pi)^{p/2}|\Sigma_r|^{1/2}} exp\{-\frac{1}{2}(x-\mu_r)^T\Sigma_r^{-1}(x-\mu_r)\}. \tag{6}$$

In die logarithmierte Form der Bayes-Regel (4) eingesetzt, erhält man folgende Diskriminanzfunktion:

$$d_r(x) = -\frac{1}{2}(x - \mu_r)^T\Sigma_r^{-1}(x - \mu_r) - \frac{1}{2}log(|\Sigma_r|) + log(p(r)). \tag{7}$$

Der additive Term $-\frac{p}{2}log(2\pi)$ wird hier vernachlässigt. Die Klassen werden somit von einer quadratischen Trennfunktion geteilt.

Im 2-Klassen-Fall lautet die Entscheidungsregel:

$$d_1(x) - d_2(x) > 0 \Leftrightarrow \delta(x) = 1. \tag{8}$$

Die Trennfläche zwischen den beiden Klassen ergibt sich für $d_1(x) - d_2(x) = 0$.

In der Praxis sind die Parameter der Normalverteilung meist unbekannt. Sie müssen deshalb aus der Lernstichprobe geschätzt werden. Klassischerweise werden folgende unverzerrte Schätzer verwendet:

- $\hat{p}(r) = \frac{n_r}{n}$ geschätzte a-priori-Wahrscheinlichkeit der Klasse r
- $\hat{\mu_r} = \overline{x}_r$ geschätzter Mittelpunkt der Klasse r
- $\hat{\Sigma_r} = S_k$ geschätzte Kovarianzmatrix der Klasse r.

Die quadratische Diskriminanzanalyse ist für orthogonale Matrizen A invariant gegenüber singulären Transformationen, d.h. das Klassifikationsergebnis wird nicht durch Merkmalstransformationen beeinflusst. Die Trennflächen zwischen den Klassen sind Hyperebenen und nehmen eine elliptische, parabolische oder hyperbolische Form an.

Der Vorteil der quadratischen Diskriminanzanalyse ist, dass weniger Annahmen als in der linearen oder diagonalen linearen Diskriminanzanalyse vorausgesetzt werden. Es werden keinerlei Aussagen über die Kovarianzmatrizen Σ_r getroffen. Dies führt allerdings zu einer großen Anzahl zu schätzender Parameter für die verschiedenen Kovarianzmatrizen. Somit ist diese Methode nur für Datensätze mit vielen Beobachtungen in jeder Klasse geeignet [Fahrmeir et al. (1984), Nothnagel (1971), Tutz (2013)].

2.1.3 Lineare Diskriminanzanalyse (LDA)

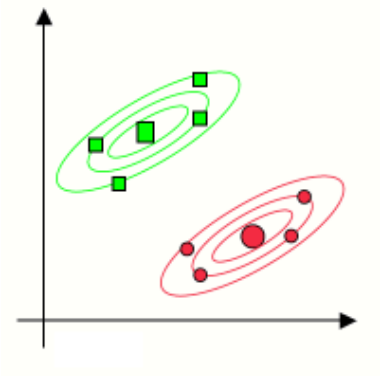

Abbildung 2: LDA
Quelle: Rahnenführer (2009)

Die lineare Diskriminanzanalyse (LDA) ergibt sich für den Spezialfall von klassenweise identischen Kovarianzmatrizen $\Sigma_r = \Sigma$ mit $r = 1, ..., k$.

$$x|c = r \sim N(\mu_r, \Sigma)$$

Daraus ergibt sich die Diskriminanzfunktion

$$d_r(x) = -\frac{1}{2} \underbrace{(x - \mu_r)^T \Sigma^{-1} (x - \mu_r)}_{\text{quadratische Mahalanobis Distanz}} + log(p(r)).$$

Da das quadratische Glied aus (7) nun nicht mehr von r abhängt, kann es vernachlässigt werden. Somit ist die Trennfunktion für klassenweise identische Kovarianzmatrizen linear. Aus diesem Grund spricht man von einer linearen Diskriminanzanalyse.

Für gleiche a-priori-Wahrscheinlichkeiten $p(1) = ... = p(k)$ wird das Individuum i derjenigen Klasse zugeordnet, deren quadratische Mahalanobis Distanz minimal ist.

Unter Verwendung der obigen Parameterschätzer ergibt sich folgende geschätzte Diskriminanzfunktion:

$$\hat{d}_r(x) = \overline{x}_r^T S^{-1} x - \frac{1}{2}\overline{x}_r^T S^{-1}\overline{x}_r + log(p(r)) \tag{9}$$

mit

$$S = \frac{1}{n-k}\sum_{c=1}^{k}\sum_{i=1}^{n_c}(x_{ci} - \overline{x}_c)(x_{ci} - \overline{x}_c)^T. \tag{10}$$

Die Entscheidungsregel im 2-Klassen-Fall (8) kann man somit umformen zu:

$$(x - \frac{1}{2}(\overline{x}_1 + \overline{x}_2))^T S^{-1}(\overline{x}_1 - \overline{x}_2) > log(\frac{p(2)}{p(1)}) \Leftrightarrow \delta(x) = 1.$$

Die Klassengrenzen der LDA bestehen abschnittsweise aus Hyperebenen. Sie ist invariant gegenüber singulären Transformationen. Ihr Vorteil liegt in der einfachen Struktur und Interpretierbarkeit. Im Gegensatz zur QDA müssen nur wechselseitige Differenzen zwischen den Diskriminanzfunktionen der Klassen geschätzt werden. Das heißt, die Anzahl der zu schätzenden Parameter ist deutlich niedriger [Fahrmeir et al. (1984), Nothnagel (1971), Tutz (2013)].

2.1.4 Diagonale Lineare Diskriminanzanalyse (DLDA)

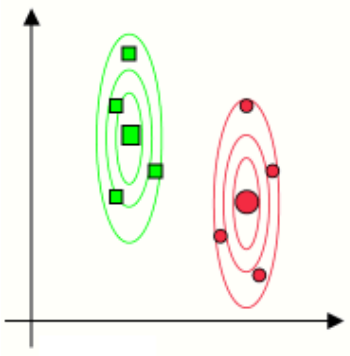

Abbildung 3: DLDA
Quelle: Rahnenführer (2009)

Bei der zusätzlichen Annahme von unkorrelierten Kovariablen wird die diagonale lineare Diskriminanzanalyse (DLDA) verwendet. Die gemeinsame Kovarianzmatrix der Klassen ist diagonal.

$$x|c = r \sim N(\mu_r, \sigma^2 I)$$

Die Klassen werden von einer linearen Funktion getrennt:

$$d_r(x) = -\frac{1}{2\sigma^2}(x - \mu_r)^T(x - \mu_r) + log(p(r)). \tag{11}$$

Im Gegensatz zur LDA müssen die Variablen in der DLDA nicht dieselbe Varianz haben. Die Einträge auf der diagonalen Kovarianzmatrix sind unterschiedlich.

Die DLDA ist in der Umsetzung am simpelsten und kann viele Variablen aufnehmen.

Man kann in praktischen Anwendungen mit guten Ergebnissen rechnen, solange die Variablen nicht zu stark korreliert sind. In der Analyse von Microarray-Studien, in denen typischerweise $p >> n$ gilt, findet sie eine häufige Anwendung. QDA und LDA haben Probleme bei Datensätzen mit mehr Beobachtungen als Variablen. Möchte man diese Methoden verwenden, ist eine vorherige Variablenselektion nötig (vgl. Abschnitt 2.2) [Pang et al. (2009), Tutz (2013)].

2.2 Dimensionsproblematik

Microarray-Studien führen zu sehr umfangreichen Datenmengen. Bei den meisten Klassifikationsmethoden ist eine zu hohe Anzahl an Variablen allerdings problematisch. Manche Prädiktionsregeln lassen sich gar nicht berechnen, wenn alle Variablen aufgenommen werden. Und selbst wenn es möglich ist, so führt die Aufnahme von Variablen mit niedrigem oder gar keinem Beitrag zur Klassifikation zu einer Verschlechterung der Performance. Die statistische Schwierigkeit besteht darin, die Menge an Informationen auf die wichtigsten zu reduzieren. Dabei können drei verschiedene Ansätze unterschieden werden:

- (explizite) Variablenselektion
- Dimensionsreduktion
- integrierte Variablenselektion.

Auch Kombinationen dieser Ansätze sind möglich. Beispielsweise könnte zunächst eine Variablenselektion und im Anschluss eine Dimensionsreduktion durchgeführt werden. Da die Variablenselektion Teil der Konstruktion der Prädiktionsregel ist, sollte sie nur auf Basis des Lerndatensatzes durchgeführt werden.

2.2.1 (Explizite) Variablenselektion

Ziel der expliziten Variablenselektion ist es, im Voraus eine Auswahl der aussagekräftigsten Variablen zu treffen. Basierend auf dieser Vorauswahl, kann dann ein traditionelles Klassifikationsverfahren (z.B. QDA, LDA, k-Nearest-Neighbors) durchgeführt werden. Hierbei unterscheidet man zwischen univariaten und multivariaten Ansätzen.

Im univariaten Verfahren werden die einzelnen Variablen getrennt voneinander betrachtet. Es wird, beispielsweise mit Hilfe einer Teststatistik, ein Ranking erstellt. Der Rang einer Variablen hängt von ihrem Nutzen zur Ermittlung der Klassenzugehörigkeit ab. Anhand dieses Rankings können nun die relevanten Einflussgrößen ausgewählt und zur Klassifikation verwendet werden. Für die Ermittlung der

Rangfolge sind verschiedene Kriterien wie der t-Test, der AUC-Wert oder der nicht parametrische Wilcoxon Rangsummentest denkbar.

Ein großer Vorteil des univariaten Ansatzes ist die schnelle und einfache Durchführung. Allerdings werden weder Korrelationen noch Interaktionen zwischen den Variablen beachtet. Sind die laut Ranking besten Variablen stark korreliert, so ist der Informationsgehalt gering.

Im multivariaten Ansatz hingegen werden nicht die einzelnen Variablen, sondern Variablenkombinationen betrachtet. Er wird durch das Kriterium zum Ranking der Variablenkombinationen sowie durch den Algorithmus, der eine Auswahl aus allen 2^{p-1} möglichen Kombinationen trifft, charakterisiert. Das Ranking kann anhand von „Wrapper-“ oder von „Filter-“ Kriterien erstellt werden. Das erste basiert auf der Prädiktionsgenauigkeit und damit auf der Prädiktionsregel. Das zweite misst die Stärke der Abgrenzung der Variablenkombination (zum Beispiel mit der Mahalanobis Distanz) und ist somit unabhängig von der Prädiktionsregel.

Nachteile dieses Ansatzes sind der rechnerisch hohe Aufwand, die Anfälligkeit gegenüber kleinen Änderungen in den Daten und die Tendenz zum Overfitting. Hinzu kommt, dass zwar meist die Korrelationen zwischen den Variablen beachtet werden, nicht aber die Interaktionen. Eine Ausnahme ist die auf Random Forest basierende Methode von Diaz-Uriarte und de Andrés.

Einen Mittelweg stellen die „semi-multivariaten“ Methoden dar. Hierbei wird zunächst ein univariates Ranking durchgeführt. Aus der Gruppe der univariat höchstrangigsten Variablen werden die paarweise niedrig korrelierten ausgewählt [Ambroise and McLachlan (2002), Boulesteix et al. (2008)].

2.2.2 Dimensionsreduktion

Ein Nachteil der Variablenselektion besteht in dem relativ starken Informationsverlust aufgrund der Auswahl einiger weniger Einflussgrößen. Die Dimensionsreduktion verfolgt deshalb einen anderen Ansatz: Eine große Menge an Variablen wird zu wenigen neuen Variablen zusammengefasst. Dies geschieht oftmals mit Hilfe von Linearkombinationen. Daraufhin können klassische Klassifikationsverfahren mit den neuen Einflussgrößen durchgeführt werden. Allerdings sind die einzelnen Komponenten nun nicht mehr interpretierbar. Methoden zur Dimensionsreduktion sind unter anderem Principal Component Analysis oder Partial Least Squares [Boulesteix et al. (2008)].

2.2.3 Integrierte Variablenselektion

Die dritte mögliche Lösung des Dimensionsproblems ist die Anwendung einer Klassifikationsmethode, die mit einer großen Anzahl an Variablen umgehen kann. Dies kann als integrierte Variablenselektion angesehen werden, da direkt zwischen relevanten und irrelevanten Variablen unterschieden wird. Dafür gibt es zum einen statistische Modelle, die auf Penalisierung oder Shrinkage basieren (z.B. Penalized Logistic Regression). Diese beinhalten normalerweise einen oder mehrere Penalty- beziehungsweise Shrinkage-Parameter, je nach dem Grad der Regularisierung. Zum anderen existieren Methoden aus dem Bereich des Machine Learnings (z.B. Random Forests). Diese Methoden können problemlos für Daten mit $n < p$ angewandt werden. Microarrays könnten sie allerdings überfordern, weshalb eine Kombination mit vorhergehender Variablenselektion oder Dimensionsreduktion oft sinnvoll ist [Boulesteix et al. (2008)].

In dieser Arbeit wird eine explizite Variablenselektion durchgeführt. Im Voraus werden mit einem klassischem t-Test die Mittelwerte der beiden Gruppen auf Gleichheit getestet. Ausgewählt werden die Variablen mit den kleinsten p-Werten.

2.3 Messung der relativen Güte

Eine genaue Schätzung der Fehlklassifikationsrate ist ein wichtiger Bestandteil der Diskriminanzanalyse. Dabei wird der Anteil der falsch klassifizierten Individuen für die gegebenen Daten ermittelt. Der Prädiktionsfehler ist auch für den Vergleich verschiedener Prädiktionsalgorithmen ein wichtiges Hilfsmittel.

2.3.1 Prädiktionsfehler

Die Wahrscheinlichkeit einer Fehlklassifikation, gegeben der feste Merkmalsvektor x, ist definiert als:

$$\varepsilon(x) = \mathbb{P}(\delta(x) \neq c \mid x) = 1 - \mathbb{P}(\delta(x) = c \mid x). \tag{12}$$

Die Gesamtfehlerrate ist der Anteil der falsch klassifizierten Individuen an der Grundgesamtheit. Sie ist definiert als:

$$\varepsilon = \mathbb{P}(\delta(x) \neq c) = \mathbb{E}(L(\delta(x) \neq c)), \tag{13}$$

wobei L eine Verlustfunktion, zum Beispiel die Indikatorfunktion

$$L = \begin{cases} 0, \text{Individuum wurde der richtigen Klasse zugeordnet} \\ 1, \text{Individuum wurde der falschen Klasse zugeordnet} \end{cases}$$

darstellt.

Es gilt:

$$\varepsilon = \mathbb{P}(\delta(x) \neq c) = \int \mathbb{P}(\delta(x) \neq c \mid x) \cdot f(x)dx = \int \varepsilon(x) \cdot f(x)dx.$$

Diese Definitionen der Fehlerrate gelten allerdings nur für ungeordnete Klassen. Für ordinal skalierte c wäre es sinnvoller eine Verlustfunktion zu wählen, die Fehlklassifikationen in weiter entfernte Klassen stärker bestraft als Fehlklassifikationen in benachbarte Klassen.

Die Bayes-Zuordnung - basierend auf der wahren Verteilung $f(x \mid r)$ - minimiert die theoretische Gesamtfehlerrate ε und ist somit eine optimale Zuordnung für bekannte Verteilungen in den Klassen. In Kapitel 3 wird die Fehlerrate für unbekannte Verteilungen näher betrachtet [Boulesteix et al. (2008), Fahrmeir et al. (1984), Leisch (2009)].

2.3.2 Monte-Carlo-Kreuzvalidierung (MCCV)

Evaluiert man die relative Güte eines Modells mit denselben Daten, die zur Aufstellung der Klassifikationsregel benutzt wurden, erhält man einen verzerrten Schätzer. Um dieses Problem zu umgehen, wird typischerweise eine Form der Kreuzvalidierung verwendet. Die Kreuzvalidierung ist eine Methode zur Evaluierung der Performance eines Modells. Hier wird die Monte-Carlo-Kreuzvalidierung betrachtet. Dabei werden die Daten mehrmals gesplittet, um den Prädiktionsfehler eines Klassifikationsmodells unverzerrt zu schätzen. Somit kann das Modell mit der besten Anpassung an die Daten identifiziert werden.

Man geht von einem Datensatz S mit n Beobachtungen aus. Um für diesen Datensatz ein Prädiktionsmodell aufzustellen, wird er in einen Lerndatensatz $\mathcal{L}$ und einen Testdatensatz $\mathcal{T}$ aufgeteilt. Der Lerndatensatz mit n_l Beobachtungen wird verwendet, um ein Modell zu fitten. Die Prädiktionsgenauigkeit dieses Modells wird am Testdatensatz mit $n_v = n - n_l$ Beobachtungen evaluiert.

Für die MCCV werden n_v Beobachtungen zufällig und ohne Zurücklegen aus dem

Datensatz gezogen. Dieser Vorgang wird Hunderte oder sogar Tausende Mal wiederholt, bis b Test- und Lerndatensätze entstehen. Die Anzahl der Iterationen kann vom Anwender beliebig hoch gewählt werden, solange die Leistung des Computers ausreicht. Je mehr Iterationen, desto robuster ist die Schätzung des Prädiktionsfehlers.

Das Verhältnis von $\mathcal{L}$ zum gesamten Datensatz β ist laut Smyth (1996) mit Werten von 0,5 und höher üblicherweise relativ groß. Shao (1993) zeigte, dass ein relativ großes β die Varianz in den Testdaten im Vergleich zur CV(n_v) reduziert. Allerdings gibt es keine weit verbreiteten Richtlinien dafür, welcher genaue Wert für β gewählt werden sollte. Nach Boulesteix et al. (2008) sind typische Verhältnisse von Lern- zu Testdaten 2:1, 4:1 oder 9:1. Dies hängt auch vom Ziel der Studie ab. Geht es nur um den Vergleich zweier Methoden, ist ein relativ kleiner Lerndatensatz angemessen. Ist auch der genaue Prädiktionsfehler von Interesse, sollte der Lerndatensatz größer gewählt werden.

Nach dem wiederholten Splitting wird für jedes Modell der durchschnittliche Prädiktionsfehler berechnet. Anhand dieses unverzerrten Schätzers für den wahren Prädiktionsfehler kann das am besten angepasste Modell ausgewählt werden [Dudoit et al. (2002), Shao (1993), Slawski et al. (2008), Smyth (1996)].

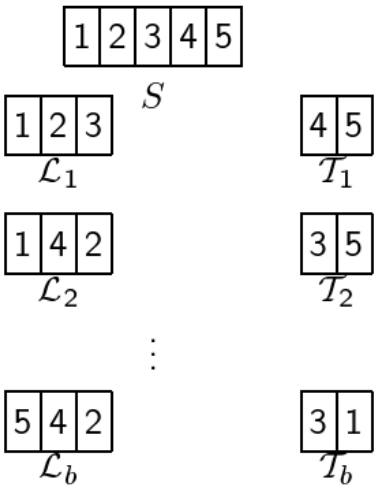

Abbildung 4: MCCV Quelle: Boulesteix et al. (2008)

Abbildung 4 zeigt eine schematische Darstellung der Monte-Carlo-Kreuzvalidierung für einen Datensatz S mit $n = 5$ und $\beta = 2/5$.

3 Benchmarking

Im Benchmarking ist in erster Linie nicht die Beurteilung der Performance verschiedener Algorithmen das Ziel, sondern die Identifikation des besten unter ihnen. Ein Vergleich der Leistung verschiedener Methoden - meist mit Hilfe des Prädiktionsfehlers - findet sich in vielen Artikeln über Machine Learning oder computationale Statistik. Dabei werden Unterschiede zwischen neuen oder bereits bekannten Methoden anhand von realen Daten ermittelt. Im Folgenden wird der statistische Hintergrund zu der Frage, welcher Algorithmus den besseren Klassifikator produziert, erläutert.

3.1 Hypothesentests

Um die Güte verschiedener Verfahren zu vergleichen, werden normalerweise Hypothesentests verwendet. Hypothesentests dienen der Überprüfung von Annahmen über einen Parameter oder auch eine Verteilung in der Grundgesamtheit. Zur Beantwortung solcher Fragestellungen muss das statistische Testproblem formuliert werden. Im Benchmarking ist der Anteil der falsch klassifizierten Individuen das interessierende Merkmal. Um Alternativ- und Nullhypothese aufzustellen, muss man also zunächst die Prädiktionsfehler der zu prüfenden Methoden kennen (vgl. Abschnitt 2.3.1).

Die Entscheidungsregel δ, basierend auf der Stichprobe s_0, ist definiert als:

$$\begin{aligned} \delta : \mathbb{R}^p &\longrightarrow c \\ \mathrm{x} &\longmapsto \delta^{s_0}(x). \end{aligned}$$

Die möglichen Methoden zur Aufstellung der Entscheidungsregel werden als M_k ($k \in \{1, ..., K\}$) bezeichnet. Der wahre Fehler ε der Methode M_k, basierend auf der Stichprobe s_0, wird mit $\varepsilon(\delta^{s_0}_{M_k}, f)$ bezeichnet, wobei f nun als unbekannt angesehen wird.

$$\varepsilon(\delta^{s_0}_{M_k}, f) = \mathbb{E}_f[L(\delta^{s_0}_{M_k}(x) \neq c)]$$

$\mathbb{E}_f$ steht für den Erwartungswert der gemeinsamen Verteilung f und L für eine Verlustfunktion (vgl. (13)). Die Notation betont die Abhängigkeit des Fehlers von der Verteilung f, der Methode M_k sowie von der Stichprobe s_0, die zur Aufstellung der Klassifikationsregel verwendet wurde. $\varepsilon(\delta^{s_0}_{M_k}, f)$ wird als abhängiger Fehler bezeichnet, da er auf der Wahl der Stichprobe s_0 basiert.

Der Fehler $\varepsilon(\delta^{s}_{M_k}, f)$ kann als Zufallsvariable angesehen werden, wobei s für eine

zufällige i.i.d. Stichprobe, die der Verteilung f^n folgt, steht.

$$\varepsilon(n, M_k, f) = \mathbb{E}_{fn}[\varepsilon(\delta^s_{M_k}, f)]$$

wird als unabhängiger Fehler der Methode M_k bezeichnet, da er von M_k, dem Stichprobenumfang n und der gemeinsamen Verteilung f, nicht aber von einer bestimmten Stichprobe s_o abhängt.

Im Benchmarking geht es um die Frage: Hat die mit der Methode M_2 gefittete Entscheidungsregel δ^{s0}_{M2} auf zukünftige Datensätze angewendet eine niedrigere Fehlerrate als die mit M_1 gefittete Entscheidungsregel δ^{s0}_{M1}? Dazu lassen sich folgende Null- und Alternativhypothese aufstellen:

$$\begin{aligned} H_0^{cond} &: \varepsilon(\delta^{s0}_{M_2}, f) - \varepsilon(\delta^{s0}_{M_1}, f) \geq 0 \\ \text{vs. } H_1^{cond} &: \varepsilon(\delta^{s0}_{M_2}, f) - \varepsilon(\delta^{s0}_{M_1}, f) < 0. \end{aligned}$$

Der Exponent „cond“ steht für *conditional*. Er soll die Abhängigkeit der Hypothesen von der Stichprobe s_0 verdeutlichen.

Anwender sind allerdings in erster Linie nicht an der Anpassung der Methoden an s_0 interessiert, sondern an der mittleren Güte des Klassifikators über verschiedene Stichproben. Dafür stehen folgende Hypothesen:

$$\begin{aligned} H_0^{uncond} &: \varepsilon(n, M_2, f) - \varepsilon(n, M_1, f) \geq 0 \\ \text{vs. } H_1^{uncond} &: \varepsilon(n, M_2, f) - \varepsilon(n, M_1, f) < 0. \end{aligned}$$

Ist der unabhängige Fehler für M_2 kleiner als für M_1, so kann die Nullhypothese verworfen werden. Somit kann man sagen, dass die Methode M_2 besser als die Methode M_1 ist.

Problematisch dabei ist, dass in realen Daten die Verteilung f meist unbekannt ist. Somit ist es schwierig H_0^{uncond} zu testen. Man benötigt einen Schätzer für $\varepsilon(n, M_2, f) - \varepsilon(n, M_1, f)$. Dafür kann man mehrere Stichproben ziehen und die Differenzen zwischen den Fehlern mitteln. Eine mögliche Methode ist die Kreuzvalidierung (vgl. Abschnitt 2.3.2). Allerdings bleibt die wahre unabhängige Varianz der Differenz unter H^{uncond} unbekannt. Schätzer basieren immer auf der Stichprobe s_0, sollen aber die Varianz über verschiedene Stichproben schätzen. Somit sind diese Schätzer wiederum abhängig [Boulesteix et al. (2013)].

3.2 Framework zum Testen realer Daten

Das „no free lunch“-Theorem besagt, dass für die Klasse aller Probleme alle Algorithmen durchschnittlich gleich gut sind. Man kann also nicht erwarten, dass eine neue Methode M_2 für alle Stichprobengrößen und Verteilungen besser als die Standardmethode M_1 ist. Laut Webb (2000) ist es fraglich, ob das Messen von Fehlerraten zwischen unterschiedlichen Gebieten (hier im Sinne von Verteilungen) überhaupt sinnvoll ist. Durchschnittlich niedrige Fehlerraten weisen aber auf eine Tendenz zu niedrigen Fehlerraten auf diesem bestimmten Gebiet hin.

Aus diesem Grund werden im von Boulesteix et al. (2013) vorgestellten Framework zum Testen der Differenzen der Fehler von M_1 und M_2 mehrere Datensätze eines bestimmten Forschungsfeldes betrachtet. Es werden J Datensätze unabhängig und zufällig gezogen. In der Praxis ist dies eher ungewiss. Forscher könnten sich auf Daten beschränken, die in Bezug auf Größe oder Verteilung nicht repräsentativ für das ganze Gebiet sind. Dennoch wird der Einfachheit halber eine zufällige Auswahl angenommen.

Die Datensätze $D_1, ..., D_J$ haben unterschiedliche Verteilungen f_j und einen Umfang n_j, $(j = 1, ..., J)$. f_j kann als das Outcome einer Zufallsvariable $\Phi_j : \Omega \to V$ angesehen werden, wobei V die Menge aller möglichen Verteilungen auf diesem Forschungsfeld ist. n_j ist das Outcome der Zufallsvariable $N_j : \Omega \to N$. $(\Phi_1, N_1), ..., (\Phi_J, N_J)$ sind unabhängig und identisch verteilt, dabei ist nur $N_j = n_j$ beobachtbar. N und Φ sind nicht zwangsläufig voneinander unabhängig.

Es können folgende Hypothesen aufgestellt werden:

$$\begin{aligned} H_0^{real} &: \mathbb{E}(\varepsilon(N, M_2, \Phi)) - \mathbb{E}(\varepsilon(N, M_1, \Phi)) \geq 0 \\ \text{vs. } H_1^{real} &: \mathbb{E}(\varepsilon(N, M_2, \Phi)) - \mathbb{E}(\varepsilon(N, M_1, \Phi)) < 0. \end{aligned}$$

$\varepsilon(N, M_k, \Phi)$ steht für eine Zufallsvariable mit Realisationen $\varepsilon(n, M_k, f)$, $k \in \{1, 2\}$. Der unbekannte Fehler für jeden Datensatz wird normalerweise durch ein Resampling-Verfahren geschätzt, zum Beispiel wiederholtes Splitting in Test- und Trainingsdaten. Der geschätzte Fehler wird mit $e(n, M_k, D)$ bezeichnet. Er kann als Schätzer für den unbekannten Parameter $\varepsilon(n, M_k, f)$ angesehen werden. Da der Lerndatensatz immer weniger Beobachtungen als der gesamte Datensatz enthält, ist $e(n, M_k, D)$ im Schnitt größer als $\varepsilon(n, M_k, f)$. Geht man von einem gleichen Bias für die Methoden M_1 und M_2 aus, führt das zu folgenden Hypothesen:

$$\begin{aligned} H_0^{real} &: \mathbb{E}(e(N, M_2, D)) - \mathbb{E}(e(N, M_1, D)) \geq 0 \\ \text{vs. } H_1^{real} &: \mathbb{E}(e(N, M_2, D)) - \mathbb{E}(e(N, M_1, D)) < 0. \end{aligned}$$

Da man Zugang zu Realisationen von $e(N, M_2, D) - e(N, M_1, D)$ hat, ist diese Formulierung verglichen mit der vorherigen von Vorteil. Seien nun $\Delta e(n_j, D_j) = e(n_j, M_2, D_j) - e(n_j, M_1, D_j)$ unabhängig und identisch verteilte Realisationen von $e(N, M_2, D) - e(N, M_1, D)$. Unter der Normalverteilungsannahme oder für sehr großes J kann man H_0^{real} mit einem t-Test für verbundene Stichproben testen. Die Teststatistik T lautet:

$$T = \frac{\overline{\Delta e}}{\sqrt{\frac{1}{J}\frac{1}{J-1}\sum_{j=1}^{J}(\Delta e(n_j, D_j) - \overline{\Delta e})^2}}.$$

Sie folgt einer Student-Verteilung mit $J-1$ Freiheitsgraden. Gilt $T < t_{\alpha,J-1}$, so kann die Nullhypothese auf einem Signifikanzniveau α abgelehnt werden. $t_{\alpha,J-1}$ bezeichnet das α-Quantil der Student-Verteilung mit $J-1$ Freiheitsgraden. Neben dem t-Test werden auch häufig nonparametrische Tests wie der Wilcoxon-Rangsummentest durchgeführt [Boulesteix et al. (2013), Fahrmeir et al. (2011), Kruse and Moewes (2011)].

3.3 Lineare Regression

Ist für einen bestimmten Datensatz Methode 1 besser als Methode 2, so trifft das auf einen anderen Datensatz nicht unbedingt zu, selbst wenn beide aus demselben Forschungsgebiet stammen. Interessant ist nun, ob bestimmte Datensatzcharakteristiken einen Einfluss auf die Performance verschiedener Methoden haben. Um einen solchen Einfluss qualitativ und quantitativ zu untersuchen, ist es möglich ein Regressionsmodell aufzustellen.

Ziel eines Regressionsmodells ist es, Eigenschaften einer Zielvariable y in Abhängigkeit von p Kovariablen $x_1, ..., x_p$ zu erklären. Dieser Zusammenhang lässt sich nicht exakt als Funktion $f(x_1, ..., x_p)$ angeben, da er von zufälligen Störungen überlagert wird. Demnach ist die Zielgröße y eine Zufallsvariable. Ihre Verteilung hängt von der Verteilung der Kovariablen ab. y kann nicht exakt bestimmt werden. Stattdessen wird der durchschnittliche Wert für y, gegeben die unabhängigen Variablen, ermittelt.

$$\mathbb{E}(y \mid x_1, ..., x_p) = f(x_1, ..., x_p)$$

Für die Zielgröße gilt:

$$y = \mathbb{E}(y \mid x_1, ..., x_p) + \epsilon = f(x_1, ..., x_p) + \epsilon.$$

ϵ bezeichnet die zufällige, nicht durch Kovariablen erklärte Abweichung vom Erwartungswert. Sie wird als stochastische Komponente oder Fehlerterm bezeichnet. $f(x_1, ..., x_p)$ wird als systematische Komponente charakterisiert. Diese systematische Komponente wird aus den Daten geschätzt. Ist f eine lineare Funktion, so spricht man von einem linearen Regressionsmodell der Form

$$y = \beta_0 + \beta_1 x_1 + ... + \beta_p x_p + \epsilon. \tag{14}$$

Das lineare Regressionsmodell wird für eine stetige und wenn möglich approximativ normalverteilte Zielgröße sowie lineare Effekte der Kovariablen eingesetzt. Für den Spezialfall $p = 1$ spricht man von einem einfachen linearen Regressionsmodell. Die Regressionskoeffizienten $\beta_0, ..., \beta_p$ sind unbekannt und müssen aus den Daten geschätzt werden. Dafür wird üblicherweise die Methode der kleinsten Quadrate verwendet. Die geschätzten Parameter werden mit $\hat{\beta}_0, ..., \hat{\beta}_p$ bezeichnet, um sie von den wahren Regressionskoeffizienten zu unterscheiden.

Setzt man die Daten jeder Beobachtung i ein, so erhält man n Gleichungen

$$y_i = \beta_0 + \beta_1 x_{i1} + ... + \beta_p x_{ip} + \epsilon_i. \tag{15}$$

Diese n Gleichungen lassen sich kompakt in Matrixnotation schreiben:

$$y = X\beta + \epsilon.$$

Dabei sind y, β und ϵ als Vektoren

$$y = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}, \beta = \begin{pmatrix} \beta_0 \\ \vdots \\ \beta_p \end{pmatrix} \text{ und } \epsilon = \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_n \end{pmatrix}$$

sowie X als Designmatrix

$$X = \begin{pmatrix} 1 & x_{11} & \cdots & x_{1p} \\ \vdots & \vdots & & \vdots \\ 1 & x_{n1} & \cdots & x_{np} \end{pmatrix} = \begin{pmatrix} x_1^T \\ \vdots \\ x_n^T \end{pmatrix}$$

definiert. Die Spalten von X müssen linear unabhängig sein, das heißt keine Kovariable darf eine lineare Transformation einer anderen sein. Ist zum Beispiel x_1 das Gewicht einer Person in kg und x_2 das Gewicht in g, so ergeben sich durch x_2 keinerlei neue Informationen.

Im linearen Modell werden zwei grundsätzliche Annahmen getroffen. Zum einen ist die Funktion $f(x_1, ..., x_p)$ eine Linearkombination der Kovariablen, d.h.

$$f(x_1, ..., x_p) = \beta_0 + \beta_1 x_1 + ... + \beta_p x_p = x'\beta,$$

zum anderen die Additivität des Fehlerterms ϵ. Diese Annahme ist in den vielen praktischen Anwendungen zumindest annähernd erfüllt.

Im klassischen linearen Modell werden zusätzliche Anforderungen an die stochastische Komponente gestellt:

1. Die Störgrößen sind im Mittel Null : $\mathbb{E}(\epsilon_i) = 0$.
2. Die Störgrößen sind homoskedastisch: $Var(\epsilon_i) = \sigma^2$.
3. Die Störungen sind unkorreliert: $Cov(\epsilon_i, \epsilon_j) = 0$, für $i \neq j$.

Aus den Annahmen 2. und 3. ergibt sich die Kovarianzmatrix $Cov(\epsilon) = \mathbb{E}(\epsilon\epsilon^T) = \sigma^2 I$. Gilt außerdem $\epsilon \sim N(0, \sigma^2 I)$, so spricht man von einer klassischen linearen Normalregression.

Ein Maß für die Anpassung des Modells an die Daten ist das Bestimmtheitsmaß. Es wird mit R^2 bezeichnet und nimmt Werte zwischen 0 und 1 an. Das Bestimmtheitsmaß gibt den Anteil der Streuung an, der durch das Modell erklärt wird. $R^2 = 1$ bedeutet also eine perfekte Anpassung an die Daten. Dies kann in realen Daten allerdings nie erreicht werden [Fahrmeir et al. (2009)].

3.4 Formulierung eines Regressionsmodells

Nach der allgemeinen Einführung in die lineare Regression soll in diesem Abschnitt ein eigenes Modell formuliert werden, um den Einfluss von Datensatzcharakteristiken auf die Performance verschiedener Diskriminanzanalysen zu untersuchen. Die interessierende Variable ist in diesem Fall der Vergleich zweier Methoden. Es geht also nicht um den Einfluss verschiedener Variablen auf die Güte einer einzelnen Methoden, sondern darum, wie sich das Verhältnis der Güte zweier Methoden ändert. Schneidet zum Beispiel die LDA mit wachsender Anzahl an Beobachtungen besser ab als die DLDA? Oder kommt eine Methode besser mit Datensätzen zu bestimmten Krebsarten zurecht als eine andere? Aus diesem Grund wird als Zielgröße die Differenz der gemittelten Prädiktionsfehler zweier Methoden $\Delta e(N, D) = e(N, M_1, D) - e(N, M_2, D)$ verwendet. Diese Differenzen sowie p Einflussgrößen werden für jeden Datensatz D ermittelt. In das Regressionsmodell fließen also J Beobachtungen ein. Im Matrixnotation lässt sich das Regressionsmodell folgendermaßen

formulieren:

$$\begin{pmatrix} \Delta e(n_1, D_1) \\ \vdots \\ \Delta e(n_J, D_J) \end{pmatrix} = \begin{pmatrix} 1 & x_{11} & \cdots & x_{1p} \\ \vdots & \vdots & & \vdots \\ 1 & x_{J1} & \cdots & x_{Jp} \end{pmatrix} \begin{pmatrix} \beta_0 \\ \vdots \\ \beta_p \end{pmatrix} + \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_J \end{pmatrix}.$$

Insgesamt wurden drei verschiedene Einflussgrößen ausgewählt, die an allen Datensätzen erhoben wurden. Es handelt sich dabei um die Anzahl an Beobachtungen, die Anzahl der gemessenen Variablen (in diesem Fall die Anzahl der beobachteten Gene) sowie den Fokus der Studie (hier: Art der untersuchten Krebserkankungen).

In diesem Modell sind zwei Punkte zu beachten. Zum einen handelt es sich bei der Zielgröße um eine Differenz zweier Klassifikationsfehler. Normalerweise wird eine Variable als Response gewählt. Hier handelt es sich um eine Verknüpfung zweier, eventuell korrelierter, Variablen. Zum anderen geht man normalerweise von Daten **eines** Datensatzes aus. Hier wird aus J verschiedenen Datensätzen ein neuer gebildet. Ob infolgedessen Modellannahmen verletzt werden, muss überprüft werden. Dafür sind - neben formalen Tests - graphische Modelldiagnosen, die auf Residuen basieren, nützlich. Zur Überprüfung der Homoskedastizität ist zum Beispiel ein Residualplot hilfreich. In einem Residualplot werden die standardisierten oder studentisierten Residuen gegen die geschätzten Werte $\hat{y}_i$ abgetragen. Idealerweise, das heißt bei Erfüllung der Modellannahmen, sollten die Residuen unsystematisch und mit konstanter Variabilität um Null streuen [Fahrmeir et al. (2009)].

4 Anwendung auf 50 Microarray-Studien

In den Kapiteln 2 und 3 wurden der theoretische Hintergrund verschiedener Klassifikationsmethoden sowie deren Vergleich betrachtet. Mit diesem Wissen folgt nun die praktische Anwendung anhand von 50 Microarray-Studien. Im ersten Abschnitt wird die Technologie der Microarrays kurz erläutert. Es folgt der Vergleich der relativen Güte von linearer, diagonaler linearer und quadratischer Diskriminanzanalyse anhand von 50 Datensätzen. Im letzten Teil wird mit Regressionsmodellen der Einfluss verschiedener Datensatzcharakteristiken auf die relative Güte der Prädiktionsalgorithmen untersucht.

4.1 Microarrays

Für die Diagnose und Behandlung von Krebs ist eine verlässliche und präzise Klassifikation von Tumoren essentiell. Die Technologie der Microarrays wurde in den späten 90er Jahren entwickelt. Es handelt sich dabei um die simultane Messung des Expressionsniveaus tausender oder sogar zehntausender Gene. Anhand des Expressionsniveaus können Rückschlüsse auf einen Krankheitsbefall oder bösartige Veränderungen von Zellen gezogen werden.

Ein Microarray, auch Gen-Chip genannt, bezeichnet einen Glas-Objektträger, auf dem tausende kurze Gen-Abschnitte angeordnet sind. Die Anzahl entspricht der Anzahl der zu untersuchenden Gene. Diese Gen-Abschnitte dienen als „Andockstellen“.

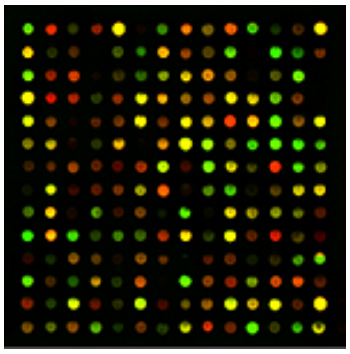

Abbildung 5: Microarray
Quelle: Poirazi (n.d.)

Um ein Genexpressionsprofil zu erstellen, muss zunächst die mRNA aus dem zu untersuchenden Gewebe isoliert werden. Aus der mRNA wird mit dem Enzym Reverse Transkriptase die dazu komplementäre cDNA synthetisiert. Anschließend wird die cDNA mit einem roten, fluoreszierendem Farbstoff markiert und auf den Gen-Chip aufgetragen. Eine grün markierte Vergleichsstichprobe wird hinzugefügt. Es findet eine Hybridisierung statt, das heißt die cDNA-Sequenzen lagern sich am jeweiligen komplementären Gegenpart auf dem Microarray an (vgl. Abb. 5). Nun kann die Intensität der roten und grünen fluoreszierenden Lösung gemessen werden. Sie entspricht der Menge an hybridisierter cDNA an jedem Punkt des Microarrays. Diese Intensitäten werden in Expressionsniveaus umgerechnet. So-

mit ist ein Genexpressionsprofil entstanden, welches die Aktivität von tausenden Genen enthält [Amaratunga and Cabrera (2001), Efron et al. (2001)].

Bei der Auswertung der Microarrays ergeben sich mathematische Hindernisse. Da diese Technologie mit hohen Kosten verbunden ist, werden meist nur wenige Genexpressionsprofile angefertigt. Dies führt zu Datensätzen mit sehr großem p - üblicherweise zwischen 5.000 und 50.000 - und vergleichsweise kleinem n ($<$300) . Klassische statistische Verfahren sind für diese Art von Datensätzen meist ungeeignet. Im folgenden Abschnitt wird die Anpassung von linearer, diagonal linearer und quadratischer Diskriminanzanalyse an 50 Microarray-Studien verglichen [Boulesteix et al. (2008)].

4.2 Beschreibung der Daten

Bei den vorliegenden Microarray-Datensätzen handelt es sich um Studien zu verschiedenen Krebsarten, wie Brustkrebs oder Leukämie, mit bereits bekannter Klassenzugehörigkeit $c \in \{0, 1\}$. Einige Studien beschäftigen sich mit dem momentanen Zustand des Patienten andere mit längerfristigen Prognosen. Die Patienten werden in Klassen wie „Metastasen ja / nein “, „ Gute / schlechte Prognose “ oder „erneutes Auftreten des Tumors ja / nein “ unterteilt. Die Anzahl an Beobachtungen n liegt zwischen 23 und 286. Je nach Studie wurden an den untersuchten Personen zwischen 1099 und 54676 Variablen p gemessen. Die Daten einer Studie sind jeweils in einer $n \times p + 1$-Matrix zusammengefasst. Die Zeileneinträge $x_i = (x_{i1}, ..., x_{ip})$ bezeichnen das Genexpressionsprofil eines Individuums i. Der $(p + 1)$-te Zeileneintrag c_i bezeichnet die Klassenzugehörigkeit des Individuums i.

Die Anzahl an Datensätzen ist mit $J = 50$ vergleichsweise hoch gewählt. Hintergrund dazu sind Powerbetrachtungen. Boulesteix et al. (2013) empfehlen, dies bei der Planung von Benchmark-Experimenten zu berücksichtigen. Ist die Varianz der Differenzen der Fehlerraten zwischen den unterschiedlichen Datensätzen hoch, so ist eine große Anzahl an Datensätzen nötig, um eine angemessene Power zu erreichen. Somit ist in diesem Vergleich eine ausreichende Power gewährleistet, um auch niedrige Differenzen Δ zu entdecken.

4.3 Methodenvergleich

Im Folgenden werden die Prädiktionsfehler verschiedener Klassifikationsverfahren berechnet und auf signifikante Unterschiede untersucht. Als Klassifikationsverfahren werden lineare, diagonale lineare und quadratische Diskriminanzanalyse verwendet.

Dabei wird eine unterschiedlich hohe Anzahl an Kovariablen in die Modelle aufgenommen. In der DLDA ist es möglich alle Variablen aufzunehmen, die LDA verlangt $n \leq p$. In der QDA ist eine noch rigorosere Auswahl notwendig. Aus diesem Grund muss im Voraus eine univariate Variablenselektion durchgeführt werden. Dazu werden mit einem klassischem t-Test die Mittelwerte der beiden Gruppen auf Gleichheit getestet. Ausgewählt werden die Variablen mit den kleinsten p-Werten.

Insgesamt wurden neun verschiedene Methoden berechnet:

- DLDA mit allen, 500, 20, 10 und 5 Variablen
- LDA mit 20, 10 und 5 Variablen
- QDA mit 5 Variablen

Für alle Analysen wurde das R Paket CMA ("**C**lassification for **M**icro**A**rrays") von Slawski et al. (2008) verwendet.

4.3.1 Prädiktionsfehler der Methoden

Zur Schätzung des Prädiktionsfehlers wird die Monte-Carlo-Kreuzvalidierung verwendet (vgl. Abschnitt 2.3.2). Jeder Datensatz wird in einem Verhältnis von 4:1 in Trainings- und Testdaten aufgeteilt. Dieses Splitting wird 300 Mal durchgeführt. Anhand der Testdatensätze wird der gemittelte Prädiktionsfehler berechnet.

Methode	alle Variablen	500 Variablen	20 Variablen	10 Variablen	5 Variablen
DLDA	0.269	0.232	0.225	0.234	0.247
LDA	-	-	0.266	0.240	0.239
QDA*	-	-	-	-	0.257

Tabelle 1: Prädiktionsfehlerraten der Methoden DLDA, LDA, QDA für eine unterschiedliche Anzahl an aufgenommenen Variablen.
(*basierend auf 42 Datensätzen)

Tabelle 1 zeigt die geschätzten Prädiktionsfehler der drei Methoden mit unterschiedlich starker Variablenselektion. Die DLDA mit allen verfügbaren Variablen liefert durchschnittlich die schlechteste Performance. Dies bestätigt die These, dass die Aufnahme nicht relevanter Variablen zu einer Verschlechterung der Modells führt. In der LDA führt die niedrigste Variablenanzahl zu den besten Ergebnissen. Die quadratische Diskriminanzanalyse kann selbst mit fünf Variablen nur für 42 Datensätze durchgeführt werden, da in acht Studien zu wenig Beobachtungen in einer oder in beiden Gruppen vorhanden sind. Aus diesem Grund wird die QDA in die weiteren Analysen nicht mehr aufgenommen. Das insgesamt beste Ergebnis erhält man mit

der linearen diagonalen Diskriminanzanalyse bei einer Aufnahme von 20 Variablen. Eine Fehlerrate von 0,225 bedeutet, dass durchschnittlich 22,5 % der Individuen einer falschen Klasse zugeordnet werden. Generell liegen die Fehlerraten mit Werten von 0,225 bis 0,269 relativ hoch. Demnach werden in jedem Modell circa ein Viertel der Patienten falsch klassifiziert.

4.3.2 Vergleich unterschiedlich starker Variablenselektion

Mit Hypothesentests werden die Unterschiede zwischen den Modellen auf Signifikanz überprüft.

$$H_0^{real} : \mathbb{E}(e(N, M_2, D)) - \mathbb{E}(e(N, M_1, D)) \geq 0$$
$$\text{vs. } H_1^{real} : \mathbb{E}(e(N, M_2, D)) - \mathbb{E}(e(N, M_1, D)) < 0.$$

Die Realisationen von $e(N, M_k, D)$ geben den mit MCCV geschätzten Fehler der Methode k für den Datensatz D_j (vgl. Abschnitt 3.2) an. Die jeweiligen Differenzen werden mit einem einseitigen t-Test für verbundene Stichproben sowie dem nonparametrischen Wilcoxon-Rangsummentest auf Signifikanz überprüft.

Zunächst werden nur die Differenzen innerhalb einer Methode mit unterschiedlicher Variablenanzahl betrachtet.

		t-Test		**Wilcoxon-Test**	
Vergleich	Differenz	t	p-Wert	W	p-Wert
DLDA-all vs. DLDA-500	0.038	-4.220	5e-05	210.0	2e-02
DLDA-all vs. DLDA-20	0.045	-3.252	0.00104	349.0	0.00272
DLDA-all vs. DLDA-10	0.036	-2.466	0.0086	409.0	0.01387
DLDA-500 vs. DLDA-20	0.007	-0.875	0.19298	550.0	0.2005
DLDA-10 vs. DLDA-500	0.002	-0.198	0.4221	603.0	0.46433
DLDA-10 vs. DLDA-20	0.009	-3.823	0.00019	276.0	0.00025

Tabelle 2: Ergebnisse des t-Tests und des Wilcoxon-Rangsummentests für die Hypothesen H_0^{real} vs. H_1^{real}. Es werden die paarweisen Differenzen der DLDA mit unterschiedlicher Variablenanzahl über alle Datensätze getestet.

In Tabelle 2 sind die Ergebnisse des t-Tests und des Wilcoxon-Rangsummentests für die Unterschiede innerhalb der diagonalen linearen Diskriminanzanalyse dargestellt. Es werden die paarweisen Unterschiede bei unterschiedlicher Variablenanzahl über alle Datensätze getestet. Die erstgenannte Methode steht jeweils für M_1, die zweite für M_2. Die über alle Datensätze gemittelten Differenzen liegen zwischen 0,002 und 0,045. Beide Tests zeigen für vier Differenzen signifikante Unterschiede an (auf einem

Signifikanzniveau $\alpha = 0,05$). Lediglich die Unterschiede von 500 zu 20 bzw. 10 aufgenommenen Variablen sind nicht statistisch signifikant.

		t-Test		**Wilcoxon-Test**	
Vergleich	Differenz	t	p-Wert	W	p-Wert
LDA-20 vs. LDA-10	0.026	-5.563	0.00000	123.0	0.00000
LDA-20 vs. LDA-5	0.027	-4.689	1e-05	244.5	8e-05
LDA-10 vs. LDA-5	0.002	-0.646	0.26064	556.0	0.21713

Tabelle 3: Ergebnisse des t-Tests und des Wilcoxon-Rangsummentests für die Hypothesen H_0^{real} vs. H_1^{real}. Es werden die paarweisen Differenzen der LDA mit unterschiedlicher Variablenanzahl über alle Datensätze getestet.

Tabelle 3 zeigt die Ergebnisse der gleichen Tests wie in Tabelle 2, diesmal allerdings für die lineare Diskriminanzanalyse. Wieder stimmen die beiden Tests bezüglich der Signifikanz überein. Es gibt signifikante Unterschiede zwischen der Aufnahme von 20 Variablen und der von 10 bzw. 5. Die Differenz der Modelle mit 5 und mit 10 Variablen hingegen ist nicht signifikant.

4.3.3 Vergleich LDA, DLDA

Nun interessieren nicht nur die Unterschiede innerhalb einer Methode, sondern auch die zwischen DLDA und LDA. Welches Modell besser ist, hängt von der Verteilung f ab. Ist die Kovarianzmatrix Σ tatsächlich diagonal, so kann man von der DLDA gute Ergebnisse erwarten. Existieren Korrelationen zwischen den Variablen und ist n groß genug, so ist die LDA eine gute Wahl. Generell gilt: je näher das Modell an die echte Verteilung f kommt, desto bessere Ergebnisse liefert es. Um einen Vergleich der Performance zu erhalten, werden aus den DLDA- und den LDA-Modellen diejenigen mit der niedrigsten Fehlerrate ausgewählt. Es werden also die Differenzen von LDA mit 5 und von DLDA mit 20 Variablen untersucht.

		t-Test		**Wilcoxon-Test**	
Vergleich	Differenz	t	p-Wert	W	p-Wert
LDA-5 vs. DLDA-20	0.014	-2.894	0.00284	359.5	0.00369

Tabelle 4: Ergebnisse des t-Tests und des Wilcoxon-Rangsummentests für die Hypothesen H_0^{real} vs. H_1^{real}. Es werden die paarweisen Differenzen von LDA (5 Variablen) und DLDA (20 Variablen) getestet.

Tabelle 4 zeigt, dass sich die Prädiktionsfehler der Methoden LDA und DLDA signifikant voneinander unterschieden. Im Schnitt schneidet die DLDA mit 20 aufge-

nommenen Variablen besser ab. Die Differenz liegt bei 0,014. Dies gilt allerdings nur über alle 50 Datensätze gemittelt. Betrachtet man die Datensätze getrennt voneinander, kann durchaus die LDA die bessere Wahl sein. Insgesamt liefert die DLDA für 32 Studien die niedrigere Fehlerrate, die LDA für 18. Um den Grund für diese Unterschiede zu untersuchen, werden im nächsten Abschnitt Regressionsmodelle gerechnet. Dabei werden verschiedene Datensatzcharakteristiken als Einflussgrößen aufgenommen, die Differenzen der Prädiktionsfehler dienen als Response.

4.4 Regressionsmodelle

Beispielhaft werden für drei verschiedene Differenzen des Prädiktionsfehlers $\Delta e(N, D)$ lineare Modelle aufgestellt. Als Kovariablen werden für jeden Datensatz die beiden stetigen Variablen „Anzahl an Beobachtungen“ und „Anzahl an Genexpressionen“ erhoben. Zusätzlich wird die dummy-kodierte Variable „Art der Krebserkrankung“ aufgenommen. Es existieren hunderte verschiedene maligne Tumorerkrankungen. Diese lassen sich je nach Gewebe- und Zellart, in der sie ihren Ursprung nehmen, in drei Hauptgruppen untergliedern: Karzinome, Sarkome und Lymphone / Leukämien. Karzinome machen etwa 80% aller bösartigen Tumore aus. Sie haben ihren Ursprung in Ephitelgeweben. Sarkome entstehen im Stützgewebe wie Muskeln oder Fettgewebe, Leukämie oder Lymphone in blutbildenden Organen [Das Lebenshaus e.V. (2011)]. Da zu Sarkomen nur ein Datensatz existiert, wird dieser als NA kodiert. Demnach wird nur in Karzinome und Lymphone / Leukämien unterteilt, wobei die Gruppe der Karzinome als Referenzkategorie dient.

In das Regressionsmodell werden also drei Einflussgrößen aufgenommen. Allgemein kann die Modellformel folgendermaßen formuliert werden:

$$\begin{pmatrix} \Delta e(n_1, D_1) \\ \vdots \\ \Delta e(n_{50}, D_{50}) \end{pmatrix} = \begin{pmatrix} 1 & x_{1,gen} & x_{1,beob} & x_{1,tumor} \\ \vdots & \vdots & & \vdots \\ 1 & x_{50,gen} & x_{50,beob} & x_{50,tumor} \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_{gen} \\ \beta_{beob} \\ \beta_{tumor} \end{pmatrix} + \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_{50} \end{pmatrix},$$

mit $\Delta e(n_j, D_j) = e(n_j, M_2, D_j) - e(n_j, M_1, D_j)$.

4.4.1 DLDA-20 vs. DLDA-10

Zum Vergleich unterschiedlich starker Variablenselektion in der DLDA wird beispielhaft die Differenz von 10 und 20 aufgenommenen Variablen betrachtet. Der Einfluss der Kovariablen wird zunächst anhand von Scatterplots graphisch dargestellt. Auf der y-Achse wird die Zielvariable - hier die Differenz der Fehlerraten von DLDA-10

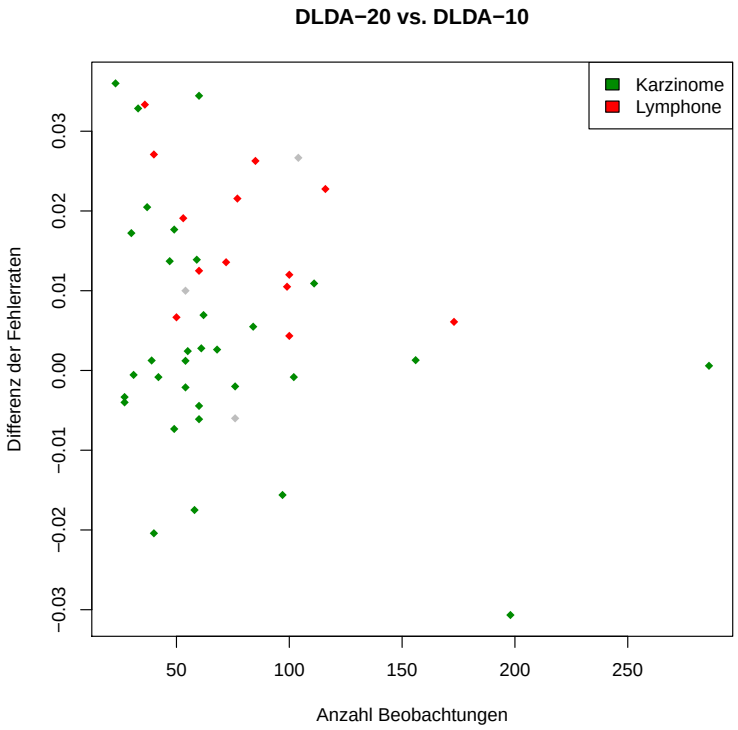

Abbildung 6: Scatterplot zu DLDA mit Kovariable „Anzahl Beobachtungen“

und DLDA-20 - angetragen. Auf der x-Achse die jeweils interessierende Kovariable. Die dummy-kodierte Einflussgröße „Art der Krebserkrankung“ wird zusätzlich durch Färbung der einzelnen Punkte dargestellt. Die grün gefärbten Punkte markieren Datensätze zu Karzinomen, die roten Punkte Datensätze zu Lymphonen oder Leukämien. Die drei grauen Punkte stehen für den Datensatz zu Sarkomen sowie für zwei Datensätze ohne weitere Angabe.

In Abbildung 6 ist die Anzahl der Beobachtungen gegen die Differenzen abgetragen. Es fällt auf, dass der Betrag der Differenzen mit steigender Anzahl an Beobachtungen niedriger wird. Das heißt, der Unterschied zwischen den beiden Methoden ist für größere Datensätze geringer als für kleine. Achtet man auf die Farbgebung, sieht man, dass alle roten Punkte über einer gedachten horizontalen Linie bei $y = 0$ liegen. Für Lymphone und Leukämien ist die Aufnahme von 10 Variablen also mit einer höheren Fehlerrate verbunden als die Aufnahme von 20 Variablen.

Abbildung 7 stellt $\Delta e(N, D)$ der Anzahl an Genexpressionen gegenüber. Anhand des Scatterplot lässt sich kein klarer Trend entdecken. Man sieht, dass der Großteil der Datensätze höchstens 25.000 Variablen enthält. Drei Datenpunkte liegen mit einer Anzahl von etwa 55.000 Genexpressionen weit abseits.

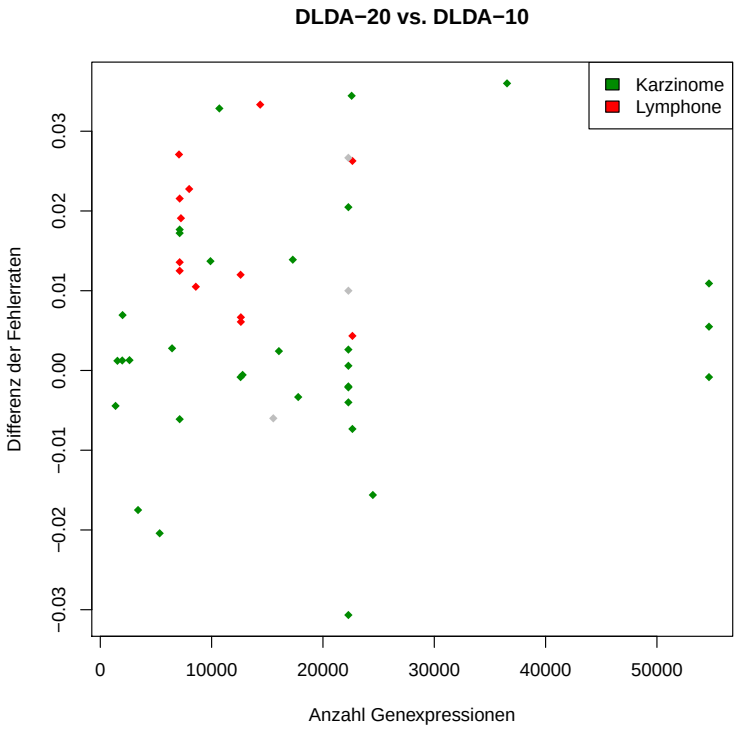

Abbildung 7: Scatterplot zu DLDA mit Kovariable „Anzahl Genexpressionen“

Nach dieser qualitativen Interpretation der Einflussgrößen folgt nun eine quantitative. Die Modellformel dazu lautet:

$$e(N, DLDA_{10}, D) - e(N, DLDA_{20}, D) = \beta_0 + \beta_{gen} x_{gen} + \beta_{beob} x_{beob} + \beta_{tumor} x_{tumor} + \epsilon$$

Variable	Koeffizient	Standardabweichung	t	p-Wert	
Intercept	9.054e-03	4.547e-03	1.991	0.052810	.
Beobachtungen	-1.191e-04	4.156e-05	-2.866	0.006413	**
Genexpressionen	1.448e-07	1.613e-07	0.897	0.374511	
Tumor	1.908e-02	4.469e-03	4.269	0.000106	***

Tabelle 5: Ergebnisse des Regressionsmodells mit den Differenzen der Fehlerraten von DLDA-20 und DLDA-10 als Zielvariable

Die Ergebnisse des Regressionsmodells sind in Tabelle 5 dargestellt. Sowohl die Anzahl der Beobachtungen als auch die Art des Tumors haben einen signifikanten Einfluss auf die Zielvariable. Der geschätzte Regressionskoeffizient der Kovariable „Anzahl an Beobachtungen“ ist negativ, mit steigender Anzahl an Beobachtungen werden die Werte der Zielgröße also niedriger. Enthält ein Datensatz j 150 Beobach-

tungen, so nimmt $\Delta e(n_j, D_j)$ durchschnittlich um $100 * (-1.191e-04) = -0.01191$ niedrigere Werte an als für einen Datensatz i mit 50 Beobachtungen. Für kleinere Datensätze liefert die DLDA mit 20 Variablen die besseren Ergebnisse. Ab einer Größe von $n = (9.054e-03)/(1.191e-04) \approx 77$ ist die DLDA mit 10 Variablen durchschnittlich besser. Der Schätzer $\hat{\beta}_{tumor}$ sagt aus, dass die Werte der Zielvariable in der Kategorie „Lymphone / Leukämie“ durchschnittlich um 0.01908 höher liegen als in der Referenzkategorie „Karzinome“. Die Anzahl an Genexpressionen hat keinen statistisch signifikanten Einfluss. Die Anpassung an die Daten ist für dieses Regressionsmodell mit $R^2 = 0,368$ vergleichsweise gut.

Betrachtet man alle Differenzen innerhalb der DLDA, so fällt auf, dass eine hohe Anzahl an Beobachtungen tendenziell für eine stärkere Variablenselektion spricht. Ein möglicher Grund dafür ist, dass für sehr kleine Datensätze die Rangfolge der Variablen nicht präzise berechenbar ist. So ist die Aufnahme einer größeren Menge an Variablen sinnvoll. Generell führen zu viele Variablen aber zu einer schlechteren Performance. Ist für größere Datensätze also die Erstellung einer exakten Rangfolge möglich, so führt eine konsequentere Auswahl zu besseren Ergebnissen. Über die Anzahl der Genexpressionen lässt sich keine klare Aussage treffen. Datensätze zu Leukämien oder Lymphonen schneiden im Verhältnis zu Karzinom-Studien tendenziell mit einer höheren Anzahl an aufgenommenen Variablen besser ab. Zu beachten ist, dass der Großteil der Ergebnisse keinen signifikanten Einfluss hat. Die Anzahl der Beobachtungen ist nur in einem Modell signifikant, die Art des Tumors für zwei Modelle. Die Ergebnisse sollten also nicht überinterpretiert werden.

Nach der Schätzung müssen die Modellannahmen Homoskedastizität, Unkorreliertheit der Störgrößen sowie die Linearität des Prädiktors überprüft werden. Diese Annahmen sollten zumindest approximativ erfüllt sein, um Fehlschlüsse zu vermeiden. Eine Korreliertheit zwischen den Residuen ist in erster Linie bei Daten mit zeitlicher Struktur, wie Zeitreihen oder Longitudinaldaten, ein Problem. Da dies hier nicht der Fall ist, wird die Überprüfung dieser Annahme vernachlässigt. Ob eine Linearität der Einflussgrößen gegeben ist, sieht man an den Scatterplots (vgl. Abb. 6 / 7). Zumindest für Abbildung 6 scheint ein linearer Einfluss plausibel.

Die Homoskedastizität wird anhand von Residuenplots überprüft. In Abbildung 8 wurden die standardisierten Residuen gegen die geschätzten Werte $\hat{y}_i$ abgetragen. Die Residuen streuen mit konstanter Variabilität um die Null. Es lassen sich keine Regelmäßigkeiten entdecken. Man kann also von homoskedastischen Fehlern ausgehen. Insgesamt sind zumindest keine schweren Verletzungen der Modellannahmen

zu erkennen [Fahrmeir et al. (2009)].

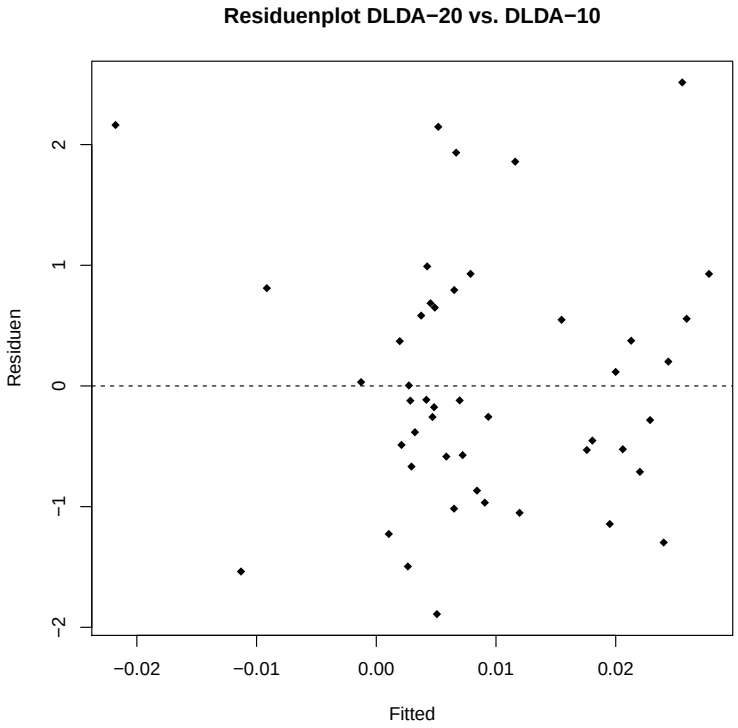

Abbildung 8: DLDA: Standardisierte Residuen gegen die geschätzten Werte $\hat{y}_i$

4.4.2 LDA-20 vs. LDA-10

In diesem Regressionsmodell wird die Differenz der Fehlerraten der LDA mit 10 und mit 20 aufgenommenen Variablen untersucht. Dabei werden dieselben Datensatzcharakteristiken wie im obigen Modell - zunächst mit Scatterplots - auf ihren Einfluss untersucht.

In Abbildung 9 sind die Differenzen von LDA-10 und LDA-20 gegen die Anzahl der Beobachtungen abgetragen. Es ist ein deutlich positiver Trend zu erkennen. Für eine steigende Anzahl an Beobachtungen streben die Differenzen zwischen den Fehlerraten gegen Null. Insgesamt liegt $\Delta e(N, D)$ für die meisten Datensätze sehr nahe an der Null. Das bedeutet, es existieren keine großen Unterschiede in der Performance von LDA mit 20 oder 10 aufgenommenen Variablen. Die meisten Datenpunkte liegen unter der Horizontalen durch $y = 0$. Die Differenzen von LDA-20 und LDA-10 sind also häufig kleiner Null. Tendenziell scheint die Aufnahme von nur 10 Variablen also etwas bessere Ergebnisse zu liefern. Im Unterschied zu Abbildung 6 und 7 ist hier kein Trend für verschiedene Krebsarten zu erkennen. Anhand des Scatterplots ist

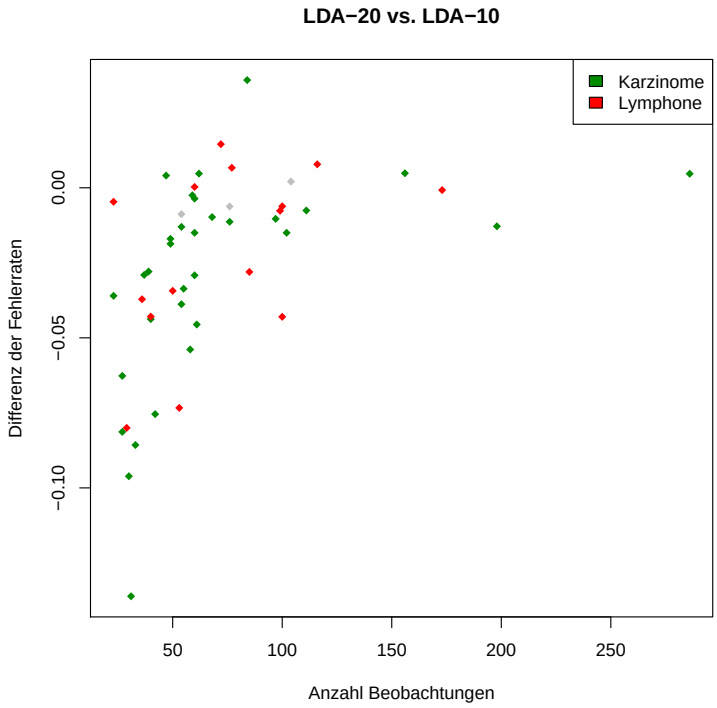

Abbildung 9: Scatterplot zu LDA mit Kovariable „Anzahl Beobachtungen“

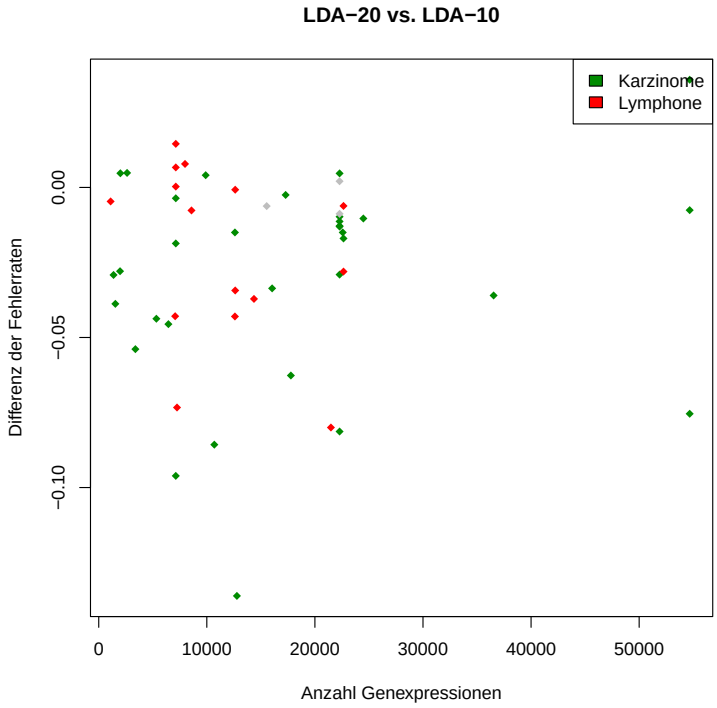

Abbildung 10: Scatterplot zu LDA mit Kovariable „Anzahl Genexpressionen“

nicht klar zu erkennen, ob es sich hier um einen linearen Einfluss handelt. Die Anzahl der Beobachtungen könnte auch einen quadratischen Einfluss haben. Anhand von Abbildung 10 lässt sich weder ein negativer noch ein positiver Einfluss der Anzahl der Gene feststellen.

Geht man von einem linearen Einfluss der Beobachtungen aus, so lautet die Modellgleichung für die beschriebene Regression:

$$e(N, LDA_{10}, D) - e(N, LDA_{20}, D) = \beta_0 + \beta_{gen} x_{gen} + \beta_{beob} x_{beob} + \beta_{tumor} x_{tumor} + \epsilon.$$

Die zugehörigen Ergebnisse sind in Tabelle 6 dargestellt. Die Anzahl der Beobachtungen hat einen signifikanten Einfluss auf die Differenzen. Dieser Einfluss ist positiv, das heißt mit steigender Anzahl an Beobachtungen steigt $\Delta e(N, D)$. Für den Großteil der Datensätze ist die Aufnahme von nur 10 Kovariablen sinnvoller. Erst ab 170 Beobachtungen liefert die LDA mit 20 Kovariablen im Schnitt die besseren Werte. Diese Beobachtung steht im Gegensatz zur DLDA. Dort schnitt für viele Beobachtungen die Auswahl von 10 Variablen besser ab, für wenige die Wahl von nur 20 Variablen. Das Bestimmtheitsmaß liegt bei 0,229. Das Modell erklärt also nur 22,9% der Streuung. Die Anpassung an die Daten ist demnach weniger gut als im vorherigen Modell. Eine Transformation der Variable „Anzahl an Beobachtungen“, um ihren quadratischen Einfluss zu überprüfen, bringt keine Verbesserung.

Variable	Koeffizient	Standardabweichung	t	p-Wert	
Intercept	-5.304e-02	9.991e-03	5.309	3.66e-06	***
Beobachtungen	3.120e-04	9.132e-05	-3.417	0.00139	**
Genexpressionen	7.433e-08	3.545e-07	-0.210	0.83489	
Tumor	7.118e-03	9.821e-03	-0.725	0.47254	

Tabelle 6: Ergebnisse des Regressionsmodells mit den Differenzen der Fehlerraten von LDA-20 und LDA-10 als Zielvariable

Für alle drei LDA-Vergleiche ergeben sich zwei Trends. Zum einen sprechen Datensätze mit vielen Beobachtungen für eine Berechnung anhand einer größeren Anzahl an Variablen. Dies ist in zwei von drei Modellen signifikant und entgegengesetzt zu den Ergebnissen aus dem DLDA-Vergleich. Allerdings muss man dazu bemerken, dass die mögliche Aufnahme von Variablen in der LDA ohnehin auf 20 begrenzt ist. Bei der Analyse von Leukämie- / Lymphon-Studien wirkt sich wie in der DLDA die Aufnahme mehrerer Variablen positiv aus. Dieser Einfluss ist allerdings in keinem Modell signifikant. Für die Anzahl der Gene lässt sich keine eindeutige Tendenz

erkennen.

Zur Überprüfung der Modellannahmen wurde wieder ein Residuenplot erstellt (Abbildung 11). In diesem Residuenplot lässt sich eine klare Verletzung der Modellannahmen feststellen. Der trichterförmige Verlauf ist typisch für heteroskedastische Varianzen. Die Störgrößen schwanken zwar um Null, doch die Varianz ist offensichtlich nicht gleichbleibend.

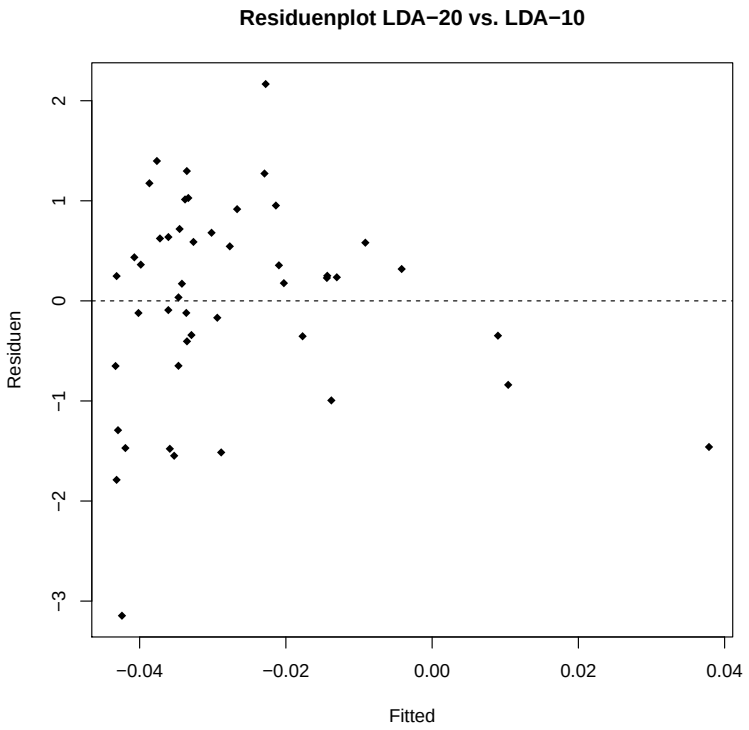

Abbildung 11: LDA: Standardisierte Residuen gegen die geschätzten Werte $\hat{y}_i$

Heteroskedastische Varianzen der Störgrößen wirken sich insbesondere auf die Schätzung der Varianz der Regressionskoeffizienten $\hat{\beta}_i$ aus. Sind diese Varianzen falsch geschätzt, hat das auch eine Einfluss auf Hypothesentests über Regressionsparameter sowie deren Konfidenzintervalle. Um dies zu vermeiden, wäre es möglich ein allgemeines lineares Regressionsmodell aufzustellen. Im allgemeinen Modell sind homoskedastische Störgrößen keine Voraussetzung. Die Regressionsparameter werden dabei mit der gewichteten Methode der kleinsten Quadrate geschätzt [Fahrmeir et al. (2009)].

4.4.3 DLDA-20 vs. LDA-5

Das dritte Regressionsmodell hat einen Vergleich von LDA und DLDA als Zielgröße. Dafür wurde für beide Methoden jeweils die Variablenanzahl gewählt, die die besten Ergebnisse lieferte. Es werden also die Differenzen der Fehlerraten zwischen LDA mit 5 Variablen und DLDA mit 20 Variablen als Response verwendet.

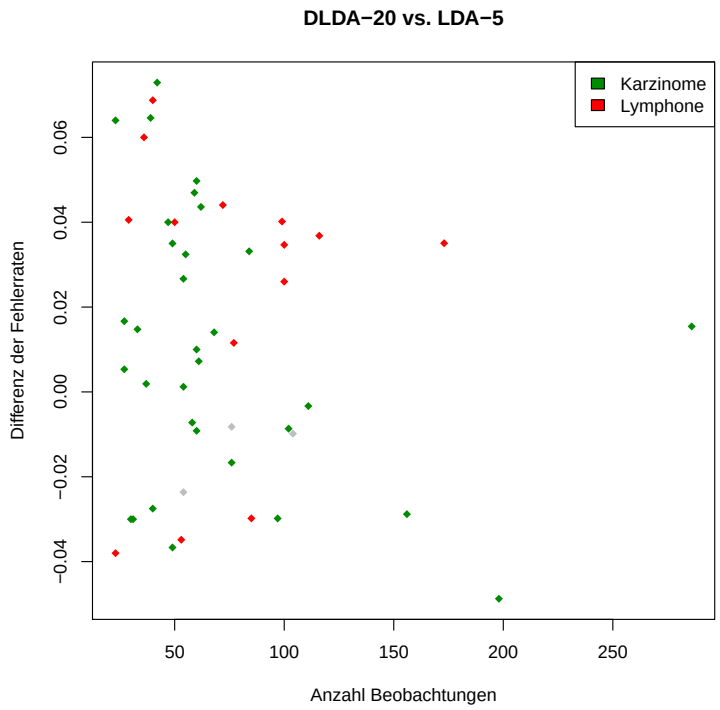

Abbildung 12: Scatterplot zu LDA-DLDA mit Kovariable „Anzahl Beobachtungen“

Abbildung 12 zeigt einen Scatterplot mit der Anzahl an Beobachtungen auf der x-Achse. Mit steigender Anzahl an Beobachtungen scheinen die Differenzen gegen Null zu streben. Es liegen mehr Werte über der Horizontalen $y = 0$. Die Ausdehnung ist von $y = 0$ aus betrachtet nach oben weiter als nach unten. Insgesamt schneidet die DLDA-20 also besser ab. Die Leukämie- / Lymphon-Daten liegen tendenziell höher als die der Karzinome.

Im Scatterplot zur Anzahl der Genexpressionen (Abb. 13) lassen sich keine Trends ausmachen. Die Datenpunkte sind relativ gleichmäßig verteilt.

Im Regressionsmodell

$$e(N, LDA_5, D) - e(N, DLDA_{20}, D) = \beta_0 + \beta_{gen} x_{gen} + \beta_{beob} x_{beob} + \beta_{tumor} x_{tumor} + \epsilon.$$

werden diese Vermutungen bestätigt (vgl. Tabelle 7).

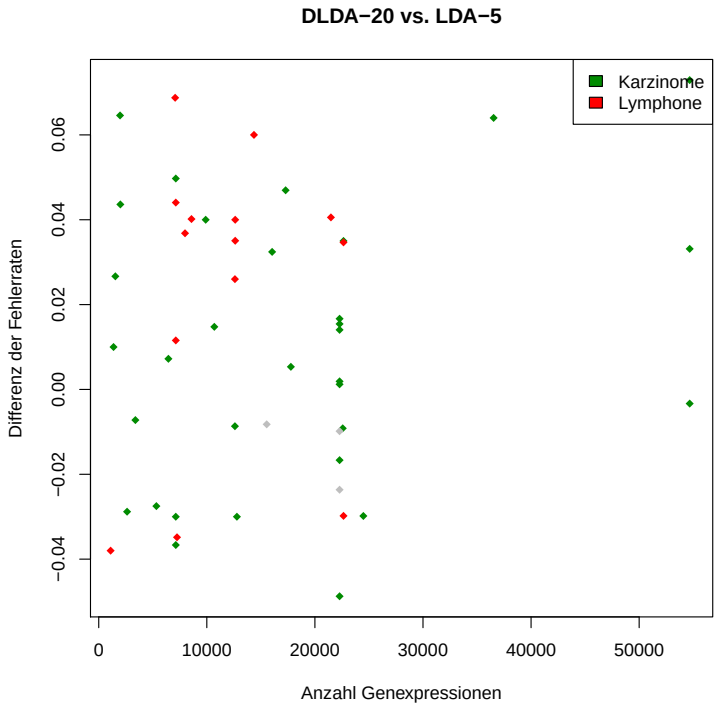

Abbildung 13: Scatterplot zu LDA-DLDA mit Kovariable „Anzahl Genexpressionen“

Variable	Koeffizient	Standardabweichung	t	p-Wert	
Intercept	1.079e-02	1.121e-02	0.962	0.3414	
Beobachtungen	-1.381e-04	1.025e-04	-1.348	0.1848	
Genexpressionen	4.962e-07	3.979e-07	1.247	0.2191	
Tumor	2.320e-02	1.102e-02	2.105	0.0412	*

Tabelle 7: Ergebnisse des Regressionsmodells mit den Differenzen der Fehlerraten von LDA-5 und DLDA-20 als Zielvariable

Den einzigen signifikanten Einfluss hat die Art der Krebserkrankung mit $\hat{\beta}_{tumor} = 2.320e{-}02$. Die Differenzen liegen also für Datensätze zu Lymphonen und Leukämien durchschnittlich um 2.320e-02 höher als für Studien zu Karzinomen. Im Verhältnis zur DLDA-20 schneidet die LDA-5 demnach für Lymphone und Leukämien schlechter ab. Allerdings hat dieses Regressionmodell mit $R^2 = 0,128$ die mit Abstand schlechteste Anpassung an die Daten.

Die Residuen im Plot 14 streuen gleichmäßig und regellos um die Null. Man kann in diesem Modell also von homoskedastischen Varianzen der Störgrößen ausgehen.

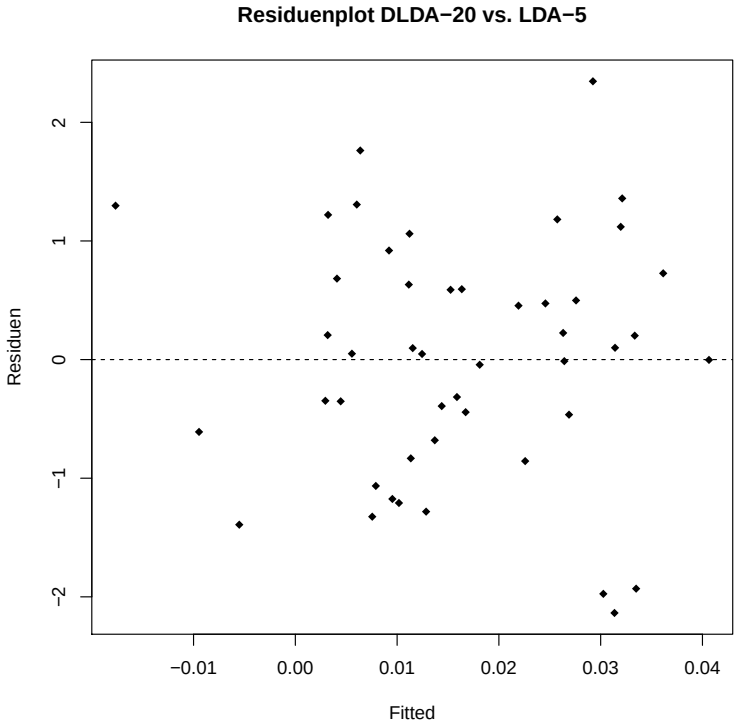

Abbildung 14: LDA-DLDA: Standardisierte Residuen gegen die geschätzten Werte $\hat{y}_i$

5 Fazit

In einem ersten Schritt wurde in dieser Arbeit ein Benchmarking durchgeführt. Dabei wurden verschiedene Formen der Diskriminanzanalyse anhand von 50 Microarray-Studien miteinander verglichen. Für die jeweiligen Methoden war im Voraus eine unterschiedlich starke Variablenselektion notwendig. Die diagonale lineare Diskriminanzanalyse ist in der Lage alle Variablen zur Schätzung einer Prädiktionsregel aufzunehmen. Die Aufnahme aller Variablen brachte allerdings die insgesamt schlechtesten Ergebnisse. Mit 20 aufgenommen Variablen ergaben sich für die DLDA der kleinste durchschnittliche Prädiktionsfehler. Die lineare und die quadratische Diskriminanzanalyse ließen sich ohne vorherige Variablenselektion nicht durchführen. Für die LDA durften bis zu 20 Variablen aufgenommen werden. Die besten Ergebnisse lieferte sie für 5 Variablen. Die QDA ließ sich mit 5 Variablen nur für 42 Datensätze berechnen. Selbst mit 3 Variablen war sie nicht auf alle Daten anwendbar. Deshalb wurde sie von weiteren Analysen ausgeschlossen. Insgesamt war die DLDA mit 20 Variablen die Methode mit der höchsten Prädiktionsgenauigkeit. Welche Methode für einen Datensatz wirklich die beste ist, hängt jedoch vom Einzelfall ab. Es lassen sich also keine pauschalen Aussagen für Krebsstudien treffen. Insgesamt lag der durchschnittliche Prädiktionsfehler mit Werten zwischen 0,225 und 0,269 für alle Methoden relativ hoch.

Nach der individuellen Betrachtung wurden die Differenzen der Prädiktionsfehler zweier Methoden analysiert. Diese wurden mit einem einseitigen t-Test für verbundene Stichproben sowie dem nonparametrischen Wilcoxon-Rangsummentest auf Signifikanz überprüft. Die zugehörigen Hypothesentests wurden in Kapitel 3 ausführlich dargestellt. Es erwiesen sich übereinstimmend 7 von 10 Differenzen als signifikant von Null verschieden.

Im zweiten Schritt wurde die eigentliche Fragestellung der Arbeit untersucht: Inwiefern beeinflussen Datensatzcharakteristiken die relative Güte verschiedener Prädiktionsalgorithmen? Um dies zu ermitteln, wurde ein lineares Regressionsmodell formuliert. Dabei dient die Differenz der Prädiktionsfehler zweier Methoden als Response. Als Einflussgrößen wurden die Anzahl der Beobachtungen, die Anzahl der gemessenen Genexpressionen sowie die Art der in der jeweiligen Studie untersuchten Krebserkrankung gewählt. Basierend auf dieser Formulierung wurden insgesamt zehn verschiedene Regressionsmodelle gerechnet.

Anhand dieser verschiedenen Regressionsmodelle lassen sich unterschiedliche Schlüsse ziehen. Die Art der Krebserkrankung zeigte in fast allen Modellen eine ähnliche

Tendenz. Bei der Analyse von Leukämie- / Lymphon-Studien wirkt sich im Vergleich zu Karzinom-Studien die Aufnahme mehrerer Variablen positiv aus. Dies gilt sowohl für die LDA als auch für die DLDA. Die Anzahl an Beobachtungen erbrachte unterschiedliche Ergebnisse. In der DLDA sollte bei einer hohen Anzahl an Beobachtungen tendenziell eine stärkere Variablenselektion vorgenommen werden. Für die LDA gilt das Gegenteil. Datensätze mit vielen Beobachtungen sprechen für eine Berechnung anhand einer größeren Anzahl an Variablen. Die Anzahl der Genexpressionen zeigte keine klaren Trends. Bei diesen Aussagen sollte allerdings beachtet werden, dass nur relativ wenige Ergebnisse auf einem Niveau von 0,05 signifikant waren. Von den zehn gerechneten Modellen hatte die Anzahl der Beobachtungen in vier Fällen einen signifikanten Einfluss. Die Art der Krebserkrankung war zweimal signifikant, die Anzahl der Genexpressionen nie.

Die Modellannahmen der linearen Regression wurden ebenfalls überprüft. Der Erwartungswert der Residuen liegt für alle Modelle sehr nahe an Null. Die Homoskedastizität ist aber nicht immer gewährleistet. Betrachtet man die Residuenplots, so sind für zwei Modelle Verletzungen zu befürchten. Auch die Linearität des Einflusses der Kovariablen ist nicht immer unbedingt gegeben. Diese Annahmen sollten also im Einzelfall überprüft werden.

Ebenfalls anzumerken ist die zum Großteil schlechte Anpassung der Modelle an die Daten. Das R^2 lag zwischen 0,01 und 0,37. Die untersuchten Kovariablen erklären die signifikanten Unterschiede zwischen den Differenzen also nur unzureichend.

Auf die bisherigen Ergebnisse aufbauend wäre es möglich, weitere Einflussgrößen sowie eventuelle Interaktionen aufzunehmen. Denkbar wären zum Beispiel das Verhältnis von Beobachtungen zu Genexpressionen, die Balance zwischen den Klassen, die mittlere Fehlerrate oder die bereits vorgegebenen Klassen der Studien. Auch eine präzisere Einteilung in verschiedene Krebserkrankungen könnte sinnvoll sein. Mit verschiedenen Methoden, wie zum Beispiel der Vorwärts- oder Rückwärtsselektion, wäre es möglich, aus einer größeren Anzahl an Kovariablen die relevanten ausfindig zu machen. Dies könnte zu besseren Modellen und damit zu aussagekräftigeren Ergebnissen führen.

Literatur

Amaratunga, D. and Cabrera, J. (2001). Analysis of Data From Viral DNA Microchips, *Journal of the American Statistical Association* **96**(456): 1161–1170.

Ambroise, C. and McLachlan, G. (2002). Selection bias in gene extraction on the basis of microarray gene-expression data, *Proceedings of the National Academy of Sciences of the United States of America* **99**(10): 6562–6566.

Boulesteix, A.-L., Hable, R. and Lauer, S. (2013). A Statistical Framework for Hypothesis Testing in Real Data Comparison Studies, *Technical Report, Department of Statistics, LMU* **136**: 1–26.

Boulesteix, A.-L., Strobl, C., Augustin, T. and Daumer, M. (2008). Evaluating Microarray-based Classifiers: An Overview, *Cancer Informatics* **6**: 77–97.

Das Lebenshaus e.V. (2011). Wissen // Sarkome, Retrieved July 5, 2013.
URL: *http://www.lh-gist.org/d/2791*

Dudoit, S., Fridlyand, J. and Speed, T. (2002). Comparison of Discrimination Methods for the Classification of Tumors Using Gene Expression Data, *Journal of the American Statistical Association* **97**(457): 77–87.

Efron, B., Tibshirani, R., Storey, J. D. and Tusher, V. (2001). Empirical Bayes Analysis of a Microarray Experiment, *Journal of the American Statistical Association* **96**(456): 1151–1160.

Fahrmeir, L., Häuß ler, W. and Tutz, G. (1984). Diskriminanzanalyse, *Multivariate statistische Verfahren*, 2 edn, Walter de Gruyter & Co., Berlin, chapter 8, pp. 357–435.

Fahrmeir, L., Kneib, T. and Lang, S. (2009). *Regression*, 2 edn, Springer-Verlag, Berlin Heidelberg.

Fahrmeir, L., Künstler, R., Pigeot, I. and Tutz, G. (2011). Testen von Hypothesen, *Statistik - Der Weg zur Datenanalyse*, 7 edn, Springer-Verlag, Berlin Heidelberg, chapter 10, pp. 397–430.

Kruse, R. and Moewes, C. (2011). Evolutionäre Algorithmen - No-Free-Lunch-Theorem, Parallelisierung, Zufallszahlen, Retrieved June 15, 2013.
URL: *http://fuzzy.cs.uni-magdeburg.de/ci/ea/ea2011_v09_nofreelunch.pdf*

Leisch, F. (2009). Klassifikation: Allgemeine Theorie, Skript zur Vorlesung 'Multivariate Verfahren' von Prof. F. Leisch im SS2009, Retrieved May 20, 2013.

URL: *http://www.stat.uni-muenchen.de/institut/ag/leisch/teaching/mv09/folien/mv-folien-4-4.pdf%257D*

Nothnagel, M. (1971). Klassifikationsverfahren der Diskriminanzanalyse - Eine vergleichende und integrierende Übersicht, *Diplomarbeit, Humboldt Universität Berlin* pp. 29–37.
URL: *http://portal.ccg.uni-koeln.de/ccg/docs/stat-gen/publ/diploma/mndalink.pdf*

Pang, H., Tong, T. and Zhao, H. (2009). Shrinkage-based Diagonal Discriminant Analysis and Its Applications in High-Dimensional Data, *Biometrics* **65**(4): 1–9.

Poirazi, Y. (n.d.). Research in Gene Expression Data Processing, Retrieved June 25, 2013.
URL: *http://www.imbb.forth.gr/people/poirazi/drupal/?q=node/4*

Rahnenführer, J. (2009). Expressionsdaten: Diskriminanzanalyse, Skript zur Vorlesung 'Statistik in der Bioinformatik' von Prof. J. Rahnenführer im SS09, Retrieved May 30, 2013.
URL: *http://www.statistik.tu-dortmund.de/fileadmin/user_upload/Lehrstuehle/Genetik/BI09/Vorlesung20090623_4x.pdf*

Shao, J. (1993). Linear Model Selection by Cross-Validation, *Journal of the American Statistical Association* **88**(422): 486–494.

Slawski, M., Daumer, M. and Boulesteix, A. (2008). CMA - A comprehensive Bioconductor package for super- vised classification with high dimensional data, *BMC Bioinformatics* **9:439**: 1–34.

Smyth, P. (1996). Clustering using Monte Carlo Cross-Validation, *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96)*, pp. 126–133.

Tutz, G. (2013). Diskriminanzanalyse, 2. Tutorium zur Vorlesung 'Multivariate Verfahren' von Prof. G. Tutz im SS2013, Retrieved May 13, 2013.
URL: *http://www.statistik.lmu.de/institut/lehrstuhl/semsto/Lehre/Multivariate2013/Tutorium/Tutorium2.pdf*

Völkl, V. (2013). Mitschrift der Vorlesung 'Multivariate Verfahren' von Prof. G. Tutz im SS2013.

Webb, G. (2000). Multiboosting : A Technique for Combining Boosting and Wagging, *Machine Learning* **40**: 170–172.

Wiesböck, K. (1987). Lineare Diskriminanzanalyse - Theoretische Ansatzpunkte und computergestützte Realisierung, *PhD Thesis, Universität Passau* pp. 1–23.

A R Code

```
### R Code zu den Berechnungen der Bachelorarbeit basierend auf
### dem R Code von Boulesteix et al. (2013)

### Mit setwd() Speicherpfad zu Ordner mit Datensaetzen setzen

library(CMA)

### 1. Einlesen der 50 verwendeten Datensaetze
datasetnames<-c("adrenal_dahia","bladder_blaveri","breast_desmedt",
        "breast_gruvberger","breast_kreike","breast_ma_2",
        "breast_minn","breast_sharma","breast_veer",
        "breast_wang","breast_west","cervical_wong",
        "cns_pomeroy_2","colon_alon","colon_laiho","colon_lin_1",
        "colon_watanabe","gastric_hippo","glioma_freije",
        "glioma_nutt","glioma_phillips","glioma_rickman",
        "head_neck_chung","headneck_pyeon_2",
        "leukemia_bullinger_2","leukemia_golub",
        "leukemia_haslinger","leukemia_wei","leukemia_yagi",
        "liver_chen","liver_iizuka","lung_barret","lung_bild",
        "lung_wigle","lymphoma_alizadeh","lymphoma_booman",
        "lymphoma_shipp","medulloblastoma_macdonald",
        "mixed_chowdary","mixed_ramaswamy","myeloma_tian",
        "oral_odonnell","ovarian_gilks","ovarian_jazaeri_3",
        "ovarian_li_and_campbell","pancreas_ishikawa",
        "prostate_singh","prostate_true_2","renal_williams",
        "sarcoma_detwiller")

for (i in 1:length(datasetnames))
 {
 print(i)
 datasetname<-datasetnames[i]
 dataset<-read.table(file=paste("data_txt/dataset_",datasetname,
 ".txt",sep=""),skip=1,header=FALSE)
 dataset<-t(dataset)
 dataset<-list(X=dataset[,-1],Y=as.factor(dataset[,1]))
```

```
 save(dataset, file=paste("data_R/",datasetname,".RData",sep=""))
 }

###
### 2. Berechnung der Fehlklassifikationsrate fuer DLDA mit
###      unterschiedlichen Anzahlen an Variablen
## Funktion zur Berechnung der Fehlklassifikationsrate
MCCV<-function(ratio,niter,datasetnames,methodnames)
{
  MCCV<-matrix(NA,length(datasetnames),length(methodnames))
   for (i in 1:length(datasetnames))
   {
     print(i)
     datasetname<-datasetnames[i]
     load(paste("data_R/",datasetname,".RData",sep=""))
     X<-dataset$X
     Y<-dataset$Y
     if (nlevels(Y)==2)
     { set.seed(1011)
       learn<- GenerateLearningsets(y=Y,method="MCCV",niter=niter,
       ntrain=round(length(Y)*ratio))
       varsel<-GeneSelection(X=X,y=Y,learningsets=learn,
       method="t.test")

       dlda<-evaluation(classification(X=X,y=Y,learningsets=learn,
       classifier=dldaCMA))
       dldanbgene500<-evaluation(classification(X=X,y=Y,
       learningsets=learn,genesel=varsel,nbgene=500,
       classifier=dldaCMA))
       dldanbgene20<-evaluation(classification(X=X,y=Y,
       learningsets=learn,genesel=varsel,nbgene=20,
       classifier=dldaCMA))
       dldanbgene10<-evaluation(classification(X=X,y=Y,
       learningsets=learn,genesel=varsel,nbgene=10,
       classifier=dldaCMA))
       dldanbgene5<-evaluation(classification(X=X,y=Y,
```

```
        learningsets=learn, genesel=varsel, nbgene=5,
        classifier=dldaCMA))

        MCCV[i,]<-c(mean(dlda@score),mean(dldanbgene500@score),
          mean(dldanbgene20@score),mean(dldanbgene10@score),
          mean(dldanbgene5@score))
    }
  }

  MCCV.matrix <- data.frame(MCCV)
  row.names(MCCV.matrix) <- datasetnames
  colnames(MCCV.matrix) <- methodnames
  MCCV_data <- stack(MCCV.matrix)
  MCCV_data <- data.frame(datasetnames,MCCV_data)
  colnames(MCCV_data) <- c("data","mc","algo")
  save(MCCV, file="MCCV_matrix.RData")
  save(MCCV_data, file = "MCCV_data.RData")
}

methodnames<-c("dlda","dldanbgene500","dldanbgene20",
                "dldanbgene10","dldanbgene5")

## Berechnung der Fehlklassifikationsrate mit MCCV
## (300 Iterationen, Verhaeltnis von Test- zu Trainingsdaten: 4/5)

MCCV(niter=300,ratio=4/5,datasetnames,methodnames)

load("MCCV_matrix.RData")
MCCV_matrix <- MCCV
resultmat <-MCCV_matrix
load("MCCV_data.RData")
colnames(MCCV_data) <- c("data","mc","algo")

colnames(resultmat) <- unique(MCCV_data[,3])
rownames(resultmat) <- unique(MCCV_data[,1])
```

```
## Berechnung der Differenzen
delta2_DLDA_DLDA10<-resultmat[,1]-resultmat[,4]
delta2_DLDA_DLDA20<-resultmat[,1]-resultmat[,3]
delta2_DLDA_DLDA500<-resultmat[,1]-resultmat[,2]
delta2_DLDA10_DLDA20<-resultmat[,4]-resultmat[,3]
delta2_DLDA500_DLDA10<-resultmat[,2]-resultmat[,4]
delta2_DLDA500_DLDA20<-resultmat[,2]-resultmat[,3]
delta2 <- data.frame(delta2_DLDA_DLDA500, delta2_DLDA_DLDA20,
          delta2_DLDA_DLDA10, delta2_DLDA500_DLDA20,
          delta2_DLDA500_DLDA10, delta2_DLDA10_DLDA20)

## Standardabweichungen der paarweisen Differenzen
## zwischen DLDA-Methoden
sd(delta2_DLDA_DLDA10)
sd(delta2_DLDA_DLDA20)
sd(delta2_DLDA_DLDA500)
sd(delta2_DLDA10_DLDA20)
sd(delta2_DLDA500_DLDA10)
sd(delta2_DLDA500_DLDA20)

### 3. Berechnung der Fehlklassifikationsrate fuer LDA mit
###      unterschiedlichen Anzahlen an Variablen
## Funktion zur Berechnung der Fehlklassifikationsrate
MCCVlda<-function(ratio,niter,datasetnames,methodnames)
{
  MCCVlda<-matrix(NA,length(datasetnames),length(methodnames))
  for (i in 1:length(datasetnames))
  {
    print(i)
    datasetname<-datasetnames[i]
    load(paste("data_R/",datasetname,".RData",sep=""))
    X<-dataset$X
    Y<-dataset$Y
    if (nlevels(Y)==2)
    { set.seed(1011)
      learn<- GenerateLearningsets(y=Y,method="MCCV",niter=niter,
```

```
        ntrain=round(length(Y)*ratio))
        varsel<-GeneSelection(X=X,y=Y,learningsets=learn,
        method="t.test")

        a <- classification(X=X,y=Y,learningsets=learn,
          genesel=varsel,nbgene=20,classifier=ldaCMA)
        b <- classification(X=X,y=Y,learningsets=learn,
          genesel=varsel,nbgene=10,classifier=ldaCMA)
        c <- classification(X=X,y=Y,learningsets=learn,
          genesel=varsel,nbgene=5,classifier=ldaCMA)
        ldanbgene20<-evaluation(a)
        ldanbgene10<-evaluation(b)
        ldanbgene5<-evaluation(c)

        MCCVlda[i,]<-c(mean(ldanbgene20@score),
          mean(ldanbgene10@score),mean(ldanbgene5@score))
      }
    }

    MCCVlda.matrix <- data.frame(MCCVlda)
    row.names(MCCVlda.matrix) <- datasetnames
    colnames(MCCVlda.matrix) <- methodnames
    MCCVlda_data <- stack(MCCVlda.matrix)
    MCCVlda_data <- data.frame(datasetnames,MCCVlda_data)
    colnames(MCCVlda_data) <- c("data","mc","algo")
    save(MCCVlda, file="MCCVlda_matrix.RData")
    save(MCCVlda_data, file = "MCCVlda_data.RData")
}

methodnameslda<-c("ldanbgene20","ldanbgene10","ldanbgene5")

## Berechnung der Fehlklassifikationsrate mit MCCV
## (300 Iterationen, Verhaeltnis von Test- zu Trainingsdaten: 4/5)
MCCVlda(niter=300,ratio=4/5,datasetnames,methodnameslda)

load("MCCVlda_matrix.RData")
```

```
MCCVlda_matrix <- MCCVlda
resultmatlda <-MCCVlda_matrix
load("MCCVlda_data.RData")

colnames(resultmatlda)<- unique(MCCVlda_data[,3])
rownames(resultmatlda)<- unique(MCCVlda_data[,1])

## Differenzen der Fehlerraten der verschiedenen Methoden
deltalda_LDA10_LDA20<-resultmatlda[,2]-resultmatlda[,1]
deltalda_LDA10_LDA5<-resultmatlda[,1]-resultmatlda[,3]
deltalda_LDA20_LDA5<-resultmatlda[,2]-resultmatlda[,3]
deltalda <- data.frame(deltalda_LDA10_LDA20,
        deltalda_LDA10_LDA5, deltalda_LDA20_LDA5)

## Standardabweichungen der paarweisen Differenzen
## zwischen LDA-Methoden
sd(deltalda_LDA10_LDA20)
sd(deltalda_LDA10_LDA5)
sd(deltalda_LDA20_LDA5)

## Standardabweichungen der paarweisen Differenzen
## zwischen bester LDA- und DLDA-Methode
sd(resultmatlda[,3]-resultmat[,3])

### 4. Berechnung der Fehlklassifikationsrate fuer
### QDA mit 5 Variablen
## Funktion zur Berechnung der Fehlklassifikationsrate
## mit 42 Datensaetzen
datasetnamesqda <- datasetnames[-c(12,15,18,32,36,38,42,43)]
MCCVqda<-function(ratio,niter,datasetnamesqda,methodnames)
{
  MCCVqda<-matrix(NA,length(datasetnamesqda),length(methodnames))
  for (i in 1:length(datasetnamesqda))
  {
    print(i)
    datasetname<-datasetnamesqda[i]
```

```
    load(paste("data_R/",datasetname,".RData",sep=""))
    X<-dataset$X
    Y<-dataset$Y
    if (nlevels(Y)==2)
    { set.seed(1011)
      learn<- GenerateLearningsets(y=Y,method="MCCV",niter=niter,
        ntrain=round(length(Y)*ratio))
      varsel<-GeneSelection(X=X,y=Y,learningsets=learn,
        method="t.test")

      a <- classification(X=X,y=Y,learningsets=learn,
        genesel=varsel,nbgene=5,classifier=qdaCMA)
      qdanbgene5<-evaluation(a)

      MCCVqda[i,]<-c(mean(qdanbgene5@score))
    }
  }

  MCCVqda.matrix <- data.frame(MCCVqda)
  row.names(MCCVqda.matrix) <- datasetnamesqda
  colnames(MCCVqda.matrix) <- methodnames
  MCCVqda_data <- stack(MCCVqda.matrix)
  MCCVqda_data <- data.frame(datasetnamesqda,MCCVqda_data)
  colnames(MCCVqda_data) <- c("data","mc","algo")
  save(MCCVqda, file="MCCVqda_matrix.RData")
  save(MCCVqda_data, file = "MCCVqda_data.RData")
}

methodnamesqda <- c("qdanbgene5")

## Berechnung der Fehlklassifikationsrate mit MCCV
## (300 Iterationen, Verhaeltnis von Test- zu Trainingsdaten: 4/5)
MCCVqda(niter=300,ratio=4/5,datasetnamesqda,methodnamesqda)

load("MCCVqda_matrix.RData")
MCCVqda_matrix <- MCCVqda
```

```
resultmatqda <-MCCVqda_matrix
load("MCCVqda_data.RData")

colnames(resultmatqda)<- unique(MCCVqda_data[,3])
rownames(resultmatqda)<- unique(MCCVqda_data[,1])

#########################################################################
### 5. Verschiedene Tabellen und Tests
## Tabelle zu mittleren Fehlerraten aller Methoden
## mit unterschiedlich vielen Variablen

dlda1_all <- mean(resultmat[,1])
dlda1_500 <- mean(resultmat[,2])
dlda1_20 <- mean(resultmat[,3])
dlda1_10 <- mean(resultmat[,4])
dlda1_5 <- mean(resultmat[,5])
dlda1 <- c(dlda1_all, dlda1_500, dlda1_20, dlda1_10, dlda1_5)

lda1_20 <- mean(resultmatlda[,1])
lda1_10 <- mean(resultmatlda[,2])
lda1_5 <- mean(resultmatlda[,3])
lda1 <- c("-", "-", lda1_20, lda1_10, lda1_5)

qda1_5 <- mean(resultmatqda[,1])
qda1 <- c("-", "-","-", "-",qda1_5)

variables1 <- c("all", "500", "20", "10", "5")
table1 <- data.frame(row.names=variables1, dlda1, lda1,
        qda1, stringsAsFactors = FALSE)

## t-Tests und Wilcoxon-Tests f r DLDA
tt_1 <- t.test(MCCV_data$mc[MCCV_data$algo=="dldanbgene500"],
        MCCV_data$mc[MCCV_data$algo=="dlda"],
        paired=TRUE, alternative="less")
tt_2 <- t.test(MCCV_data$mc[MCCV_data$algo=="dldanbgene20"],
        MCCV_data$mc[MCCV_data$algo=="dlda"],
```

```
          paired=TRUE, alternative="less")
tt_3 <- t.test(MCCV_data$mc[MCCV_data$algo=="dldanbgene10"],
          MCCV_data$mc[MCCV_data$algo=="dlda"],
          paired=TRUE, alternative="less")
tt_4 <- t.test(MCCV_data$mc[MCCV_data$algo=="dldanbgene20"],
          MCCV_data$mc[MCCV_data$algo=="dldanbgene500"],
          paired=TRUE, alternative="less")
tt_5 <- t.test(MCCV_data$mc[MCCV_data$algo=="dldanbgene500"],
          MCCV_data$mc[MCCV_data$algo=="dldanbgene10"],
          paired=TRUE, alternative="less")
tt_6 <- t.test(MCCV_data$mc[MCCV_data$algo=="dldanbgene20"],
          MCCV_data$mc[MCCV_data$algo=="dldanbgene10"],
          paired=TRUE, alternative="less")

wt_1 <- wilcox.test(MCCV_data$mc[MCCV_data$algo=="dldanbgene500"],
          MCCV_data$mc[MCCV_data$algo=="dlda"],
          paired=TRUE, alternative="less")
wt_2 <- wilcox.test(MCCV_data$mc[MCCV_data$algo=="dldanbgene20"],
          MCCV_data$mc[MCCV_data$algo=="dlda"],
          paired=TRUE, alternative="less")
wt_3 <- wilcox.test(MCCV_data$mc[MCCV_data$algo=="dldanbgene10"],
          MCCV_data$mc[MCCV_data$algo=="dlda"],
          paired=TRUE, alternative="less")
wt_4 <- wilcox.test(MCCV_data$mc[MCCV_data$algo=="dldanbgene20"],
          MCCV_data$mc[MCCV_data$algo=="dldanbgene500"],
          paired=TRUE, alternative="less")
wt_5 <- wilcox.test(MCCV_data$mc[MCCV_data$algo=="dldanbgene500"],
          MCCV_data$mc[MCCV_data$algo=="dldanbgene10"],
          paired=TRUE, alternative="less")
wt_6 <- wilcox.test(MCCV_data$mc[MCCV_data$algo=="dldanbgene20"],
          MCCV_data$mc[MCCV_data$algo=="dldanbgene10"],
          paired=TRUE, alternative="less")


tab.test <- data.frame(
 Comparison = c("DLDA-all vs. DLDA-500","DLDA-all vs. DLDA-20",
```

```
        "DLDA-all vs. DLDA-10", "DLDA-500 vs. DLDA-20",
        "DLDA-10 vs. DLDA-500","DLDA-10 vs. DLDA-20"),
 Difference=c(round(tt_1$estimate,3),round(tt_2$estimate,3),
        round(tt_3$estimate,3),round(tt_4$estimate,3),
        round(tt_5$estimate,3),round(tt_6$estimate,3)),
 t = c(round(tt_1$statistic,3),round(tt_2$statistic,3),
        round(tt_3$statistic,3),round(tt_4$statistic,3),
        round(tt_5$statistic,3),round(tt_6$statistic,3)),
 'p-value' = c(round(tt_1$p.value,5), round(tt_2$p.value,5),
        round(tt_3$p.value,5),round(tt_4$p.value,5),
        round(tt_5$p.value,5),round(tt_6$p.value,5)),
 W = c(wt_1$statistic, wt_2$statistic, wt_3$statistic,
        wt_4$statistic,wt_5$statistic,wt_6$statistic),
 'p-value' = c(round(wt_1$p.value,5), round(wt_2$p.value,5),
        round(wt_3$p.value,5),round(wt_4$p.value,5),
        round(wt_5$p.value,5),round(wt_6$p.value,5)))

## t-Tests und Wilcoxon-Tests fuer LDA
t_1 <- t.test(MCCVlda_data$mc[MCCVlda_data$algo=="ldanbgene10"],
        MCCVlda_data$mc[MCCVlda_data$algo=="ldanbgene20"],
        paired=TRUE, alternative="less")
t_2 <- t.test(MCCVlda_data$mc[MCCVlda_data$algo=="ldanbgene5"],
        MCCVlda_data$mc[MCCVlda_data$algo=="ldanbgene20"],
        paired=TRUE, alternative="less")
t_3 <- t.test(MCCVlda_data$mc[MCCVlda_data$algo=="ldanbgene5"],
        MCCVlda_data$mc[MCCVlda_data$algo=="ldanbgene10"],
        paired=TRUE, alternative="less")

w_1 <- wilcox.test(MCCVlda_data$mc[MCCVlda_data$algo=="ldanbgene10"],
        MCCVlda_data$mc[MCCVlda_data$algo=="ldanbgene20"],
        paired=TRUE, alternative="less")
w_2 <- wilcox.test(MCCVlda_data$mc[MCCVlda_data$algo=="ldanbgene5"],
        MCCVlda_data$mc[MCCVlda_data$algo=="ldanbgene20"],
        paired=TRUE, alternative="less")
w_3 <- wilcox.test(MCCVlda_data$mc[MCCVlda_data$algo=="ldanbgene5"],
        MCCVlda_data$mc[MCCVlda_data$algo=="ldanbgene10"],
```

```
        paired=TRUE, alternative="less")


tab.testlda <- data.frame(
 Comparison = c("DLDA-20 vs. DLDA-10",
        "DLDA-20 vs. DLDA-5", "DLDA-10 vs. DLDA-5"),
 Difference = c(round(t_1$estimate,3),round(t_2$estimate,3),
        round(t_3$estimate,3)),
 t = c(round(t_1$statistic,3), round(t_2$statistic,3),
        round(t_3$statistic,3)),
 'p-value' = c(round(t_1$p.value,5), round(t_2$p.value,5),
        round(t_3$p.value,5)),
 W = c(w_1$statistic, w_2$statistic, w_3$statistic),
 'p-value' = c(round(w_1$p.value,5), round(w_2$p.value,5),
        round(w_3$p.value,5)))

## t-Tests und Wilcoxon-Tests fuer die jeweils besten Methoden
## von DLDA und LDA
tt_dlda_lda <-t.test(MCCV_data$mc[MCCV_data$algo=="dldanbgene20"],
        MCCVlda_data$mc[MCCVlda_data$algo=="ldanbgene5"],
        paired=TRUE, alternative="less")
wt_dlda_lda <-wilcox.test(MCCV_data$mc[MCCV_data$algo==
        "dldanbgene20"], MCCVlda_data$mc[MCCVlda_data$algo==
        "ldanbgene5"], paired=TRUE, alternative="less")


tab.testall <- data.frame(
 Comparison = c("LDA-5 vs. DLDA-20"),
 Difference = c(round(tt_dlda_lda$estimate,3)),
 t = c(round(tt_dlda_lda$statistic,3)),
 'p-value' = c(round(tt_dlda_lda$p.value,5)),
 W = c(wt_dlda_lda$statistic),
 'p-value' = c(round(wt_dlda_lda$p.value,5)))

##################################################
### Eigener R Code
```

```
### 6. Regressionsmodelle
## Einflussgroessen

beobachtungen <- numeric(length(datasetnames))
for (i in 1:length(datasetnames))
 {
  print(i)
  datasetname <- datasetnames[i]
  datasetname <- read.table(file=paste("data_txt/dataset_",
                 datasetname,".txt",sep=""),skip=1,header=FALSE)
  datasetname <- t(datasetname)
  beobachtungen[i] <- nrow(datasetname*i)
 }

## Vektor mit Anzahl Variablen je Datensatz

variablen <- numeric(length(datasetnames))
for (i in 1:length(datasetnames))
 {
  print(i)
  datasetname <- datasetnames[i]
  datasetname <- read.table(file=paste("data_txt/dataset_",
                 datasetname,".txt",sep=""),skip=1,header=FALSE)
  datasetname <- t(datasetname)
  variablen[i] <- ncol(datasetname*i)
 }

## Spalte fuer Krebsart: Karzinom=0, Leukaemie&Lymphone=1,
## Sarkom=NA (nur 1 Fall)
tumor <- c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0,
        0, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1,
        1, 1, 1, NA, NA, 1, 0, 0, 0, 0, 0, 0, 0, 0, NA)

delta2 <- cbind(delta2, beobachtungen, variablen, tumor)
deltalda <- cbind(deltalda, beobachtungen, variablen, tumor)
```

```
lda5_dlda20 <- resultmatlda[,3]-resultmat[,3]
dlda_lda_10 <- resultmat[,4]-resultmatlda[,2]
dlda_lda_20 <- resultmat[,3]-resultmatlda[,1]
dlda_lda <- data.frame(lda5_dlda20, dlda_lda_10,
        dlda_lda_20, beobachtungen, variablen, tumor)

### Differenzen innerhalb der Methoden mit unterschiedlicher
### Anzahl Variablen
## a) DLDA
lmdlda1 <- lm(delta2$delta2_DLDA_DLDA500~
        delta2$beobachtungen+delta2$variablen+delta2$tumor)
summary(lmdlda1)

lmdlda2 <- lm(delta2$delta2_DLDA_DLDA20~
        delta2$beobachtungen+delta2$variablen+delta2$tumor)
summary(lmdlda2)

lmdlda3 <- lm(delta2$delta2_DLDA_DLDA10~
        delta2$beobachtungen+delta2$variablen+delta2$tumor)
summary(lmdlda3)

lmdlda4 <- lm(delta2$delta2_DLDA500_DLDA20~
        delta2$beobachtungen+delta2$variablen+delta2$tumor)
summary(lmdlda4)

lmdlda5 <- lm(delta2$delta2_DLDA500_DLDA10~
        delta2$beobachtungen+delta2$variablen+delta2$tumor)
summary(lmdlda5)

lmdlda6 <- lm(delta2$delta2_DLDA10_DLDA20~
        delta2$beobachtungen+delta2$variablen+delta2$tumor)
summary(lmdlda6)

## b) LDA
lmlda1 <- lm(deltalda$deltalda_LDA10_LDA20~
        deltalda$beobachtungen+deltalda$variablen+deltalda$tumor)
```

```
summary(lmlda1)

lmlda2 <- lm(deltalda$deltalda_LDA10_LDA5~
        deltalda$beobachtungen+deltalda$variablen+deltalda$tumor)
summary(lmlda2)

lmlda3 <- lm(deltalda$deltalda_LDA20_LDA5~
        deltalda$beobachtungen+deltalda$variablen+deltalda$tumor)
summary(lmlda3)

### Vergleich DLDA / LDA
## a) Auswahl der Variablenanzahl mit niedrigster Fehlerrate
lm_dlda_lda <- lm(dlda_lda$lda5_dlda20 ~
        dlda_lda$beobachtungen+dlda_lda$variablen+dlda_lda$tumor)
summary(lm_dlda_lda)

## b) Vergleich LDA/DLDA mit je 10/20 Variablen
lm_10 <- lm(dlda_lda$dlda_lda_10 ~
        dlda_lda$beobachtungen+dlda_lda$variablen+dlda_lda$tumor)
summary(lm_10)

lm_20 <- lm(dlda_lda$dlda_lda_20 ~
        dlda_lda$beobachtungen+dlda_lda$variablen+dlda_lda$tumor)
summary(lm_20)

### naehere Untersuchung ausgewaehlter Modelle
## zu DLDA20 vs. DLDA10

karzinom <- subset(delta2, delta2$tumor==0)
lymphon <- subset(delta2, delta2$tumor==1)
rest <- subset(delta2, is.na(delta2$tumor))

## Scatterplots Response gegen Kovariablen (Tumor einfaerben)
plot(karzinom$delta2_DLDA10_DLDA20~karzinom$beobachtungen,
        col="green4", pch=18, xlab="Anzahl Beobachtungen",
        ylab="Differenz der Fehlerraten",
```

```
        main="DLDA-20 vs. DLDA-10")
points(lymphon$delta2_DLDA10_DLDA20 ~
        lymphon$beobachtungen, col="red", pch=18)
points(rest$delta2_DLDA10_DLDA20 ~
        rest$beobachtungen, col="grey", pch=18)
legend("topright",c("Karzinome", "Lymphone"),
        fill=c("green4", "red"))

plot(karzinom$delta2_DLDA10_DLDA20~karzinom$variablen,
col="green4", pch=18, xlab="Anzahl Genexpressionen",
        ylab="Differenz der Fehlerraten",
        main="DLDA-20 vs. DLDA-10")
points(lymphon$delta2_DLDA10_DLDA20 ~
        lymphon$variablen, col="red", pch=18)
points(rest$delta2_DLDA10_DLDA20 ~
        rest$variablen, col="grey", pch=18)
legend("topright",c("Karzinome", "Lymphone"),
        fill=c("green4", "red"))

## Residuenplots; gefittete gegen standardisierte Residuen
plot(fitted(lmdlda6), rstandard(lmdlda6),
        main="Residuenplot DLDA-20 vs. DLDA-10",
        pch=18, xlab="Fitted", ylab="Residuen")
abline(h=0,lwd=1,lty="dashed")
mean(residuals(lmdlda6))

## zu LDA20 vs. LDA10

karzinomlda <- subset(deltalda, deltalda$tumor==0)
lymphonlda <- subset(deltalda, deltalda$tumor==1)
restlda <- subset(deltalda, is.na(deltalda$tumor))

## Scatterplots Response gegen Kovariablen (Tumor einfaerben)
plot(karzinomlda$deltalda_LDA10_LDA20~karzinomlda$beobachtungen,
        col="green4", pch=18, xlab="Anzahl Beobachtungen",
        ylab="Differenz der Fehlerraten",
```

```
        main="LDA-20 vs. LDA-10")
points(lymphonlda$deltalda_LDA10_LDA20 ~
        lymphonlda$beobachtungen, col="red", pch=18)
points(restlda$deltalda_LDA10_LDA20 ~
        restlda$beobachtungen, col="grey", pch=18)
legend("topright",c("Karzinome", "Lymphone"),
        fill=c("green4", "red"))

plot(karzinomlda$deltalda_LDA10_LDA20~karzinomlda$variablen,
        col="green4", pch=18, xlab="Anzahl Genexpressionen",
        ylab="Differenz der Fehlerraten",
        main="LDA-20 vs. LDA-10")
points(lymphonlda$deltalda_LDA10_LDA20 ~
        lymphonlda$variablen, col="red", pch=18)
points(restlda$deltalda_LDA10_LDA20 ~
        restlda$variablen, col="grey", pch=18)
legend("topright",c("Karzinome", "Lymphone"),
        fill=c("green4", "red"))

## Residuenplots; gefittete gegen standardisierte Residuen
plot(fitted(lmlda1), rstandard(lmlda1),
        main="Residuenplot LDA-20 vs. LDA-10",
        pch=18, xlab="Fitted", ylab="Residuen")
abline(h=0,lwd=1,lty="dashed")
mean(residuals(lmlda1))

## zu DLDA20 vs. LDA5

karzinommix <- subset(dlda_lda, dlda_lda$tumor==0)
lymphonmix <- subset(dlda_lda, dlda_lda$tumor==1)
restmix <- subset(dlda_lda, is.na(dlda_lda$tumor))

## Scatterplots Response gegen Kovariablen (Tumor einfaerben)
plot(karzinommix$lda5_dlda20~karzinommix$beobachtungen,
        col="green4", pch=18, xlab="Anzahl Beobachtungen",
        ylab="Differenz der Fehlerraten",
```

```
        main="DLDA-20 vs. LDA-5")
points(lymphonmix$lda5_dlda20 ~
        lymphonmix$beobachtungen, col="red", pch=18)
points(restmix$lda5_dlda20 ~
        restmix$beobachtungen, col="grey", pch=18)
legend("topright",c("Karzinome", "Lymphone"),
        fill=c("green4", "red"))

plot(karzinommix$lda5_dlda20~karzinommix$variablen,
        col="green4", pch=18, xlab="Anzahl Genexpressionen",
        ylab="Differenz der Fehlerraten",
        main="DLDA-20 vs. LDA-5")
points(lymphonmix$lda5_dlda20 ~
        lymphonmix$variablen, col="red", pch=18)
points(restmix$lda5_dlda20 ~
        restmix$variablen, col="grey", pch=18)
legend("topright",c("Karzinome", "Lymphone"),
        fill=c("green4", "red"))

## Residuenplots; gefittete gegen standardisierte Residuen
plot(fitted(lm_dlda_lda), rstandard(lm_dlda_lda),
        main="Residuenplot DLDA-20 vs. LDA-5",
        pch=18, xlab="Fitted", ylab="Residuen")
abline(h=0,lwd=1,lty="dashed")
mean(residuals(lm_dlda_lda))

## Residuenplots fuer restliche Regressionsmodelle

plot(fitted(lmdlda1), rstandard(lmdlda1),
        main="Residuenplot DLDA-500 vs. DLDA-all",
        pch=18, xlab="Fitted", ylab="Residuen")
abline(h=0,lwd=1,lty="dashed")
mean(residuals(lmdlda1))

plot(fitted(lmdlda2), rstandard(lmdlda2),
        main="Residuenplot DLDA-20 vs. DLDA-all",
```

```
        pch=18, xlab="Fitted", ylab="Residuen")
abline(h=0,lwd=1,lty="dashed")
mean(residuals(lmdlda2))

plot(fitted(lmdlda3), rstandard(lmdlda3),
        main="Residuenplot DLDA-10 vs. DLDA-all",
        pch=18, xlab="Fitted", ylab="Residuen")
abline(h=0,lwd=1,lty="dashed")
mean(residuals(lmdlda3))

plot(fitted(lmdlda4), rstandard(lmdlda4),
        main="Residuenplot DLDA-20 vs. DLDA-500",
        pch=18, xlab="Fitted", ylab="Residuen")
abline(h=0,lwd=1,lty="dashed")
mean(residuals(lmdlda4))

plot(fitted(lmdlda5), rstandard(lmdlda5),
        main="Residuenplot DLDA-10 vs. DLDA-500",
        pch=18, xlab="Fitted", ylab="Residuen")
abline(h=0,lwd=1,lty="dashed")
mean(residuals(lmdlda5))

plot(fitted(lmlda2), rstandard(lmlda2),
        main="Residuenplot LDA-5 vs. LDA-10",
        pch=18, xlab="Fitted", ylab="Residuen")
abline(h=0,lwd=1,lty="dashed")
mean(residuals(lmlda2))

plot(fitted(lmlda3), rstandard(lmlda3),
        main="Residuenplot LDA-5 vs. LDA-20",
        pch=18, xlab="Fitted", ylab="Residuen")
abline(h=0,lwd=1,lty="dashed")
mean(residuals(lmlda3))
```

Erklärung

Hiermit versichere ich, dass ich meine Abschlussarbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Datum: .....

(Unterschrift)