Evaluierung von Feature-Selektionsmethoden in Mammographie-Screeningdaten

Diplomarbeit

an der

Ludwig-Maximilians-Universität München

vorgelegt von

Judith Schwitulla

Erstkorrektor: Prof. Dr. Kurt Ulm

Betreuer: Dipl. Stat. Alexander Hapfelmeier

München, 03. Februar 2010

DANKSAGUNG

Mein Dank geht an Herrn Prof. Dr. Ulm für die Ermöglichung der Bearbeitung dieses interessanten und vielseitigen Themengebiets.

Bedanken möchte ich mich auch bei Dipl. Stat. Alexander Hapfelmeier, der mich von Anfang bis Ende sehr gut betreut hat und mir v.a. bei R Problemen stets schnell weiterhalf,

bei Tibor, für das Korrekturlesen,

bei Petra, die mir bei jeglichen Problemen mit Rat und Tat zur Seite stand,

und bei Lars, für die medizinischen Hintergrundinformationen.

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Einleitung

Die Statistik hat sich in den letzten Jahren zunehmend zu einem unentbehrlichen Begleiter der medizinischen Wissenschaft entwickelt.
Der Einsatz von Computern lässt hochkomplexe Analysen zu, deren Aussagewerte die Effizienz der medizinischen Forschung erheblich steigern können. Frühzeitige Diagnosen und die Prognose des Therapieerfolges sind Teile davon.
Die dadurch umfangreichen Datensätze führen jedoch zu einer immer größeren Anzahl von Variablen. Um die Durchführbarkeit der Analysen zu gewährleisten, gilt es, die Anzahl der Variablen auf die aussagekräftigsten zu beschränken.
Ein klassisches Beispiel hierfür sind Microarray Daten.
Gerade in den letzten Jahren ist es in diesem Bereich der medizinischen Statistik zu Veränderungen gekommen. Hochdimensionale Daten erlangen eine immer größere Bedeutung, da man mit ihrer Hilfe weitaus komplexere und medizinisch tiefgründigere Fragestellungen beantworten kann. Microarrays ermöglichen es, gleichzeitig mehrere tausend Gene zu messen, und sind daher weit verbreitet in der medizinischen Forschung. Sie unterstützen die Identifikation von Krankheits-Biomarkern, die u.a. wichtig sein können zur Erkennung verschiedener Krankheiten[21] bzw. zur Prognose von Krankheitsverläufen und der Prädiktion eines Therapieerfolges.

Das Problem bei hochdimensionalen Daten liegt darin, dass es deutlich mehr Variablen als Beobachtungen gibt. In der Literatur wird dies häufig als $p >> n$ Problem bezeichnet, wobei p für die Anzahl an Variablen steht und n für die Beobachtungen.

In Bezug auf die Genetik besteht ein solcher Datensatz aus mehreren tausend oder sogar zehntausenden Genexpressionsvariablen, die angeben, wie "aktiv" ein bestimmtes Gen ist bzw. wie oft es transkribiert wurde. Da die Herstellung solcher Microarray Daten sehr aufwändig und teuer ist, stehen häufig nur wenige Beobachtungen zur Verfügung (ca. 20-300)[2].

Ziel einer statistischen Analyse ist es, anhand der vorliegenden Variablenausprägungen eine möglichst sichere Entscheidung bezüglich des Vorliegens einer Krankheit zu treffen.
Das "Vorliegen einer Krankheit" kann als binäre oder nominale/ordinale Zielgröße dargestellt werden. Bei einer binären Zielgröße gilt die Fragestellung dem Auftreten bzw. Nicht-Auftreten des Zielmerkmals, während bei nominalen oder ordinalen Responsevariablen in mehrere Klassen (≥ 3) eingeteilt wird.

Wegen der hohen Anzahl an Variablen kann die Anwendung von Standardprädiktionsverfahren, wie Logistische Regression oder Diskriminanzanalyse problematisch sein, da die Modelle überbestimmt sind und letztendlich nicht eindeutige Lösungen der Schätzgleichungen resultieren.
Es gilt herauszufinden, welche der Variablen im Zusammenhang mit der Krankheit stehen bzw. welche Variablen den größten Erklärungswert haben. So werden schließlich nur die ausgewählten Variablen in die Klassifikation mit einbezogen und die effiziente Anwendung standardmäßiger Verfahren ermöglicht.
Für die Analyse von Microarray Daten wurden in den letzten Jahren verschiedene innovative Methoden entwickelt. Daraus konnten sich Standardverfahren etablieren, welche sich der "Dimensionsproblematik" annehmen.

Ziel dieser Arbeit ist es, einige dieser Verfahren aufzugreifen, zu beschreiben und einen Überblick über die Eigenschaften der jeweiligen Methoden zu geben. Dies geschieht anhand eines realen Datensatzes von Mammographie-Screeningdaten, die aus einer öffentlich zugänglichen Datenbank erhoben wurden.

Im folgenden Kapitel wird zuerst grundlegend erklärt, wie mit hochdimensionalen Daten umgegangen werden kann. Anschließend wird auf die hier verwendeten Variablenselektionsverfahren eingegangen. Kapitel 2 beschreibt die Idee des Überwachten Lernens, geht auf die Lineare Diskriminanzanalyse (LDA), die Logistische Regression (LogReg), die Support Vector Machine (SVM), sowie die Least Absolute Shrinkage and Selection Operators (LASSO) ein.
Anhand der Klassifikationsgenauigkeit wird schließlich die Güte der Variablenselektionsverfahren gemessen.
Kapitel 3 gibt eine Einführung über den medizinischen Hintergrund, einen kurzen Überblick über die Daten und erläutert schließlich die Anwendung der Methoden. Die Ergebnisse von Variablenselektion und Klassifikation sind in Kapitel 4 dargestellt. Abschließend folgt eine Zusammenfassung mit Diskussion, Fazit und Ausblick im letzten Kapitel.

Kapitel 1

Dimensionsproblematik

Wie bereits in der Einleitung erwähnt, führt eine zu hohe Anzahl an Variablen bei den Standardprädiktionsverfahren zu Problemen und ungenauen Vorhersagen. Die Aufgabe dieses Kapitels besteht darin, die Variablenmenge so zu reduzieren, dass einerseits eine einwandfreie Anwendung der nachfolgenden Klassifikation ermöglicht wird und andererseits die Prädiktionsgenauigkeit nicht negativ beeinflusst wird.

1.1 Umgang mit hochdimensionalen Daten

Die Auswahl von aussagekräftigen Variablen hat in Verbindung zu hochdimensionale Daten verschiedene Ziele. Zum einen kann diese als vorangehender Schritt zur Klassifikation gesehen werden, da die gewählte Klassifikationsmethode nur mit einer geringen Anzahl an Variablen umgehen kann. Zum anderen kann die Variablenselektion dazu dienen, die Variablen zu identifizieren, die mit der Krankheit assoziiert sind.
Die Klassifikation bei hochdimensionalen Daten kann man grob in drei Gruppen einteilen[2]:

- Ansätze, die auf vorheriger Variablenselektion beruhen
- Ansätze, die auf Dimensionsreduktion beruhen und
- Ansätze mit integrierter Variablenselektion

In dieser Arbeit werden verschiedene Ansätze zur Variablenselektion mit anschließender Klassifikation miteinander verglichen. Auch LASSO, welches ein Verfahren mit integrierter Variablenselektion darstellt, wird in den Vergleich mit einbezogen.

1.2 Vorangehende Variablenselektion

Bei der Variablenselektion gilt es aus einer großen Anzahl an Informationen die wichtigsten auszuwählen, d.h. diejenigen Variablen zu selektieren, die mit dem Auftreten der Krankheit assoziiert werden können.
Die Literatur unterscheidet hier univariate und multivariate Ansätze, die im Folgenden kurz vorgestellt werden sollen.

Univariate Ansätze

Beim univariaten Ansatz wird jede Variable für sich betrachtet. Daher ist er im Allgemeinen schnell und einfach durchzuführen.
Die Variablen werden nach einem bestimmten (univariaten) Kriterium sortiert, z.B. nach den Werten einer Teststatistik, um schließlich die besten dieses Rankings auszuwählen. Übliche Statistiken für ein solches Ranking sind der t-Test, der nicht parametrische Wilcoxon Rang Summen Test oder der AUC-Wert.
Der Nachteil liegt darin, dass bei diesem univariaten Ansatz weder Korrelationen noch Interaktionen zwischen den Variablen beachtet werden. So ist es möglich, dass die ersten ausgewählten Variablen so stark miteinander korrelieren, dass nur einige davon brauchbare Information enthalten. In einem solchen Fall wäre es also vorteilhafter, Variablen auszuwählen, die zwar eine schlechtere Wertung im Ranking haben, dafür aber nicht redundante Informationen liefern.

"Semi-multivariater" Ansatz

Als "Semi-multivariat" wird der von Jaeger et al.[20] vorgestellte Korrelationsansatz gesehen. Dieser basiert zum einen auf dem univariaten Ranking nach einer bestimmten Statistik, zum anderen wird jedoch auch die paarweise Korrelation zwischen den Variablen betrachtet, die jeweils unter einem bestimmten Wert liegen soll. (s. 2.3.2)

Multivariate Ansätze

In multivariaten Ansätzen werden die Variablen nun nicht mehr für sich betrachtet, sondern Variablenkombinationen miteinander verglichen. Man spricht hier von "Wrapper" und "Filter Criteria". Das erste Kriterium basiert auf der Prädiktionsgenauigkeit und somit auf der Prädiktionsregel. Das zweite misst die Stärke der Abgrenzung (z.B. mittels der Mahalanobis Distanz) verschiedener Variablenkombinationen und ist damit unabhängig von einer Prädiktionsregel[1].
Dies führt zu rechentechnisch teils sehr aufwändigen Verfahren.
Es sei noch darauf verwiesen, dass multivariate Ansätze lediglich Korrelationen zwischen den einzelnen Variablen betrachten, jedoch keine Interaktionen. Diaz-Uriarte und de Andrés[6] stellen eine der wenigen Methoden vor, die dies, basierend auf Random Forests, beachten.

Variablenranking: Vorbereitung für die Selektion

Beim Variablenranking, einer Vorstufe zur Variablenselektion, werden die Variablen zuerst nach einem bestimmten Kriterium sortiert.
Verwendet wird hier der AUC-Wert (Area Under the ROC Curve). Eine ROC (Receiver Operating Characteristic) Analyse beschreibt das Verhältnis zwischen Sensitivität und 1-Spezifität, also zwischen der richtig positiven und falsch positiven Rate. Trägt man beide Werte in ein Koordinatensystem(Sensitivität/1-Spezifität) ein, so ergibt sich die ROC Kurve, bei der die Fläche zwischen der Winkelhalbierenden des ersten Quadranten und der Kurve berechnet wird[21]. Die ROC Methode zeigt, wie gut ein System zwei Verteilungen, mindestens ordinalskalierter Merkmale, unterscheiden kann.

Anschaulich gesprochen ergibt sich dann ein hoher AUC-Wert, wenn sich die Werte in den beiden Gruppen (gesund oder krank) besonders stark voneinander unterscheiden, d.h. wenn die Beobachtungen ohne Krankheit andere Variablenausprägungen haben, als die Beobachtungen mit Krankheit.
Hat man für jede Variable den AUC-Wert berechnet, werden diese sortiert. An erster Stelle des Rankings steht die Variable mit dem höchsten AUC-Wert, an zweiter Stelle die mit dem zweithöchsten AUC-Wert, usw..
Vorgestellt werden nun zwei Methoden, die auf diesem AUC Ranking beruhen, zuerst ohne Beachtung der Korrelationen zwischen den Variablen und schließlich mit deren Berücksichtigung.

1.3 Variablenselektionsmethoden

1.3.1 Top Ranking Variablen

Eine Möglichkeit der Variablenselektion besteht darin, die ersten Variablen dieses Rankings zu betrachten.
Es gibt keine Richtlinie bei der Bestimmung einer genauen Anzahl an Variablen für die Klassifikation[23]. Daher werden hier einige Möglichkeiten ausprobiert, um eine eventuelle Tendenz bzgl. der Anzahl an Variablen zu beobachten. Um ein breites Spektrum abzudecken, werden zehn verschiedene Werte von 1% bis 100% der Ranking Liste betrachtet: 1%(entspricht den ersten fünf Variablen), 2%(9), 5%(23), 8%(36), 10%(45), 15%(68), 22%(99), 55%(249), 70%(316) und 100%(452).
Ein Problem dieses univariaten Ansatzes ist, wie schon oben beschrieben, dass die obersten Variablen stark korrelieren und somit redundante Information geben könnten.

1.3.2 Korrelationen

Um dieses Problem zu vermeiden wird folgender Ansatz vorgestellt, in den die Korrelationen mit einbezogen werden[20]. Diese sollen bei allen ausgewählten Variablen unter einer bestimmten Grenze liegen um gänzlich redundante

Variablen auszuschließen. Als Grenzwert werden auch hier zehn verschiedene Werte zwischen 0 und 1 betrachtet: 0.02, 0.05, 0.2, 0.35, 0.45, 0.6, 0.75, 0.95, 0.98 und 1.
Hierfür wird eine Korrelationsmatrix berechnet, welche auf dem AUC-Ranking beruht. D.h. die Variable mit dem größten AUC-Wert steht an erster Stelle, die mit dem kleinsten an letzter. Es wird die Korrelation nach Pearson berechnet. Begonnen wird, die Korrelation mit der stärksten Variable des Rankings zu betrachten, also die Korrelation zwischen erster und zweiter Variable. Liegt die Korrelation unter dem vorgegebenen Grenzwert, so wird die Variable aufgenommen, ansonsten aus der Variablenmenge entfernt. Spaltenweise werden die Korrelationen mit der ersten Variablen betrachtet und bei Überschreiten des Grenzwerts aussortiert. Anschließend wiederholt sich der Vorgang für die übrig gebliebenen Variablen, so dass schrittweise die Korrelation aller Variablen untereinander mit dem Grenzwert abgeglichen werden. Es resultiert schließlich eine Variablenmenge, deren Korrelationen jeweils den Grenzwert nicht überschreiten.

Der Ansatz kann leicht verändert werden, indem die stärkste Variable weggelassen und eine Korrelationsmatrix, die mit der zweitstärksten Variablen beginnt, für die Selektion genutzt wird.
Interessant hierbei ist, inwiefern sich das Wegnehmen der stärksten Variablen des Rankings auf die resultierende Variablenmenge und die Klassifikationsgenauigkeit auswirkt.
Ein weiterer Vergleich kommt zustande mit der drittstärksten Variablen an erster Stelle der Korrelationsmatrix und dem Weglassen der beiden AUC stärksten Variablen.
Auch für die Ansätze mit der zweit- und drittstärksten Variablen des Rankings gelten dieselben Korrelationsgrenzen wie oben aufgelistet.

Um das Ranking und die Korrelationsansätze untereinander vergleichbar zu machen, wurde darauf geachtet, dass bei beiden jeweils etwa gleiche Variablenanzahlen resultieren. Gleichzeitig wurde das Gitter für

Variablenanzahlen, die kleiner als 100 sind, feiner gestaltet, weil es von Interesse ist, zu sehen, wie die Variablenselektionsmethoden bei kleinen Variablenmengen abschneiden. Ziel ist es, mit möglichst wenigen Variablen eine möglichst gute Prognose zu erreichen.

1.3.3 Einfache Vorwärtsselektion

Zum Vergleich dient auch die einfache Vorwärtsselektion, beruhend auf dem BIC Kriterium (Bayes Informationskriterium).
Hier wird ein logistisches Modell an die Daten angepasst, in das schrittweise weitere Variablen aufgenommen werden, nämlich immer die, mit der das neue Modell den geringsten BIC-Wert aufweist. Ergeben sich bei Hinzunahme einer weiteren Variablen keine Verbesserungen mehr, so wird die Vorwärtsselektion an dieser Stelle abgebrochen und die Variablenmenge als die optimalste gesehen[10].

Kapitel 2

Supervised Learning

Überwachtes Lernen (Supervised Learning) spielt eine wichtige Rolle in der Statistik und wird in der Literatur ausführlich behandelt. Nachfolgend wird ein kurzer Einblick gegeben über die Definition des Supervised Learnings im Allgemeinen und über einzelne Methoden im Speziellen.

Eine Beobachtungseinheit $\omega \in \Omega$, von der nicht bekannt ist, welcher Klasse sie angehört, soll mit Hilfe des an ihr beobachteten Merkmalsvektors x_ω in eindeutiger Weise genau einer der Klassen $y = 1, ..., g$ zugeordnet werden.
Typische Fragestellungen in der Medizin diesbezüglich lauten z.B. "Hat der Patient mit diesen Merkmalen Krebs oder nicht?" oder "Welchen Schweregrad hat der Tumor des Patienten bei diesen Laborwerten?".

Um solche Problemstellungen behandeln zu können werden in diesem Zusammenhang der so genannte Lerndatensatz und der Testdatensatz eingeführt. Diese dienen dazu, zuerst aus den Daten zu "lernen" um schließlich das Gelernte an neuen Daten zu "testen".
Der Lerndatensatz, für den die Zielvariable bereits bekannt ist, entspricht einer Zufallsauswahl an Beobachtungseinheiten aus der Grundgesamtheit, anhand derer eine Regel aufgestellt wird. Mithilfe dieser konstruierten Regel wird künftig, allein durch Übergabe der verschiedenen Merkmalsausprägungen (z.B. klinische Messwerte, Alter, Geschlecht, etc.), eine Entscheidung

bzgl. Krankheit oder nicht Krankheit getroffen(→ Klassifikation). Auf den Testdatensatz wird die gelernte Regel anschließend angewendet.

Bewertet wird die Güte einer Entscheidungsregel schließlich anhand der Genauigkeit ihrer Zuordnung, d.h. der Übereinstimmung vorhergesagter und tatsächlich beobachteter Werte[16].

Mit dem gegebenen Prädiktorraum $X \in \mathbb{R}^p$ und der abhängigen Variablen $Y \in \{1, ..., g\}$ liegt eine Stichprobe von Paaren aus Prädiktor- und Klassenvariablen vor: $S = \{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_n, y_n)\}$
Es soll nun eine Entscheidungsregel aufgestellt werden, die jeder Beobachtung $\mathbf{x}$ aus dem Stichprobenraum S einen geschätzten Klassifikationsindex $\hat{y} \in \{1, ..., g\}$ zuordnet und dabei nur möglichst geringe Fehlentscheidungen zulässt.
Prinzipiell kann zwischen der Maximum-Likelihood- und der Bayes-Entscheidungsregel unterschieden werden, wobei die Bayes-Entscheidungsregel (siehe Kapitel 3.1) für alle $\mathbf{x}$ die kleinste bedingte Fehlerrate
$\epsilon(x) = P(\delta(x) \neq y|x))$ und damit auch die kleinste Gesamtfehlerrate
$\epsilon = \int \epsilon(x) f(x) dx = \int P(\delta(x) \neq y|x)) f(x) dx$ besitzt.
Solche Entscheidungsfunktionen

$$\begin{aligned} \hat{\delta} : X &\to Y \\ x &\to \hat{\delta}(x) \end{aligned}$$

werden gebildet, indem durch Merkmalsausprägungen und deren bereits bekannter Klassenzugehörigkeit eine Zuordnungsregel aufgestellt wird[28].
Ein mögliches Vorgehen ist den Datensatz in einen so genannten Lern- und Testdatensatz aufzuspalten, wobei sowohl Lern- als auch Testdatensatz bekannte Responses haben.
Anhand des Lerndatensatzes $\mathcal{L} = \{(\mathbf{x}_{1\mathcal{L}}, y_{1\mathcal{L}}), ..., (\mathbf{x}_{n\mathcal{L}}, y_{n\mathcal{L}})\}$ wird die Entscheidungsregel aufgestellt und diese mit Hilfe des Testdatensatzes
$\mathcal{T} = \{\mathbf{x}_{1\mathcal{T}}, ..., \mathbf{x}_{n\mathcal{T}}\}$ evaluiert. Dies geschieht indem die resultierenden Klassen mit der wahren Ausprägung verglichen werden.

Wichtig ist, dass $\mathcal{L}$ und $\mathcal{T}$ disjunkte Teilmengen sind. Jede der Beobachtungen aus der Grundgesamtheit $\mathcal{L} \cup \mathcal{T}$ darf demnach entweder nur in $\mathcal{L}$ oder nur in $\mathcal{T}$ vorkommen.

Die im Folgenden beschriebenen Klassifikationsverfahren sind, mit Ausnahme von LASSO, Techniken, bei denen die Klassengrenzen linear in den Prädiktorvariablen x sind. Der Variablenraum wird in seine Klassen zerlegt, getrennt durch Hyperebenen[1] als Klassengrenzen.

2.1 Lineare Diskriminanzanalyse

Mit der Diskriminanzanalyse wird eine Methodenklasse vorgestellt, die Diskriminanzfunktionen $\delta_y(x)$ für jede Klasse modellieren und den Variablenvektor $\mathbf{x}$ schließlich in die Klasse einordnet, die den größten Wert für seine a posteriori Wahrscheinlichkeit bzw. Diskriminanzfunktion besitzt[16]. Die Bayes Entscheidungsregel lautet $\delta(x) = y \iff P(y|x) = \max\limits_{i=1,\ldots,g} P(i|x)$. Für eine optimale Klassifikation werden die a posteriori Wahrscheinlichkeiten $P(Y = y|X = x)$ oder deren monotone Transformationen benötigt, die die Wahrscheinlichkeit angeben, dass eine Beobachtungseinheit mit Merkmalsvektor $\mathbf{x}$ der Klasse y angehört. Diese kann man über das Theorem von Bayes bestimmen:

$$P(y|x) = \frac{f(x|y)f(y)}{f(x)} = \frac{f_y(x)\pi_y}{\sum_{l=1}^{y} f_l(x)\pi_l}$$

mit der bedingten Dichte $f_y(x)$ von x in Klasse y und der a priori Wahrscheinlichkeit π_y für Klasse y mit $\sum \pi_y = 1$.
Mögliche Diskriminanzfunktionen sind [24]:

$$\delta_y(x) = P(y|x) \text{ oder } \delta_y(x) = f(x|y)\pi(y)$$

Nimmt man nun multivariat normalverteilte Klassendichten, $\mathbf{x} \sim N_p(\mu, \Sigma_y)$, an, so ergibt sich für die Verteilung der Merkmale, gegeben der Klasse:

[1] Hyperebenen sind: in $\mathbb{R}^2$ Geraden, in $\mathbb{R}^3$ Ebenen, in $\mathbb{R}^4$ dreidimensionale Ebenen, etc.

$$f(\mathbf{x}|y) = \frac{1}{(2\pi)^{\frac{p}{2}} \det(\mathbf{\Sigma}_y)^{\frac{1}{2}}} \exp\{-\frac{1}{2}(\mathbf{x} - \mu_y)^T \mathbf{\Sigma}_y^{-1}(\mathbf{x} - \mu_y)\} \tag{2.1}$$

mit

Σ_y als Kovarianzmatrix in Klasse $y = 1, ..., g$

x als Merkmalsvektor und

μ_y als Mittelwert der Klasse y

Gleichung (2.1) eingesetzt in die logarithmierte Form der Bayes Regel führt zu den Diskriminanzfunktionen

$$\delta_y(x) = -\frac{1}{2}(\mathbf{x} - \mu_y)^T \mathbf{\Sigma}_y^{-1}(\mathbf{x} - \mu_y) - \frac{1}{2}\ln(\det \Sigma_y) + \ln \pi(y)$$

Sowohl der Term $(2\pi)^{\frac{p}{2}}$ als auch $f(x)$ fallen raus und kommen nicht mehr in $\delta_y(x)$ vor, da beide konstant sind. Gut erkennbar ist, dass die a posteriori Wahrscheinlichkeit nicht mit der Diskriminanzfunktion übereinstimmt.
Die Lineare Diskriminanzanalyse ergibt sich für den Spezialfall von klassenweise identischen Kovarianzmatrizen $\Sigma_y = \Sigma$ mit $y = 1, ...g$.
Bei der Linearen Diskriminanzanalyse handelt es sich um einen entscheidungstheoretischen Ansatz (im Gegensatz zur historisch älteren Fischerschen Diskriminanzanalyse[11]). Ein Vorteil der LDA besteht in ihrer Invarianz gegenüber nichtsingulären Transformationen ($x \rightarrow Ax + b$). Das Klassifikationsergebnis bleibt also selbst bei Merkmalstransformationen gleich[9].

Mit Hilfe der Bayes Regel $P(y|x)\,\alpha\, f(x|y)\pi(y)$ in logarithmierter Form ergibt sich für die Diskriminanzfunktion bei Normalverteilung mit gleichen Kovarianzen $\Sigma_y = \Sigma$:

$$\begin{aligned} \delta_y(x) &= -\frac{1}{2}\overbrace{(x - \mu_y)^T \Sigma^{-1}(x - \mu_y)}^{quadr. Mahalanobis Distanz} + \ln \pi(y) \\ &\alpha \quad \mu_y^T \Sigma^{-1} x - \frac{1}{2}\mu_y^T \Sigma^{-1} \mu_y + \ln \pi(y) \end{aligned}$$

Bei gleichen a priori Wahrscheinlichkeiten($\pi_1 = ... = \pi_g$) wird x der Klasse zugeordnet, deren quadratische Mahalanobis Distanz minimal ist.
Da in der Praxis die Parameter der Normalverteilung meist nicht bekannt sind, müssen diese aus der Lernstichprobe geschätzt werden:

- $\hat{\pi}(y) = \frac{n_y}{n}$ die geschätze a priori Wahrscheinlichkeit für Klasse y
- $\hat{\mu}_y = \overline{x}_y = \sum_{y=1}^{g} \frac{x_y}{n_y}$ der geschätzte Klassenmittelpunkt
- $\hat{\Sigma} = \frac{1}{N-g} \sum_{y=1}^{g} \sum_{i=1}^{n_y} (y_{iy} - \overline{x}_y)(x_{iy} - \overline{x}_y)^T$

mit n_y: Anzahl der Beobachtungen in Klasse y
N: Anzahl der Beobachtungen insgesamt und
g: Anzahl der Klassen

Diese Schätzer ergeben den Schätzer für die Diskriminanzfunktionen:

$$\hat{\delta}_y(x) = \hat{\mu}_y^T \hat{\Sigma}^{-1} x - \tfrac{1}{2} \hat{\mu}_y^T \hat{\Sigma}^{-1} \hat{\mu}_y + \ln \hat{\pi}(y)$$

Speziell für den Zwei-Klassen-Fall ergibt sich folgende Zuordnungsregel:

$$\delta(x) = \delta_0(x) - \delta_1(x) = [x - \frac{1}{2}(\mu_0 - \mu_1)]^T \Sigma^{-1} (\mu_0 - \mu_1) - \ln \frac{\pi(1)}{\pi(0)}$$

x wird Klasse 1 zugeordnet, wenn $\delta(x) > 0$.
Die Klassengrenzen bestehen abschnittsweise aus Hyperebenen, die sich aus $\delta_0(x) = \delta_1(x)$ für zwei benachbarte Gebiete ergeben.
Betrachtet man die log-odds der a posteriori Wahrscheinlichkeiten bei identischen Kovarianzmatrizen,

$$\begin{aligned} \log \frac{P(k|x)}{P(l|x)} &= \log \frac{f_k(x)}{f_l(x)} + \log \frac{\pi_k}{\pi_l} \\ &= \log \frac{\pi_k}{\pi_l} - \frac{1}{2}(\mu_k + \mu_l)^T \Sigma^{-1} (\mu_k + \mu_l) + x^T \Sigma^{-1} (\mu_k + \mu_l) \end{aligned}$$

so erkennt man, dass die Grenze zwischen den Klassen k und l linear in x ist, wie für jedes andere Klassenpaar auch.

Wird der gesamte Stichprobenraum $X \in \mathbb{R}^p$, $p \geq 3$ in seine Klassen aufgeteilt, so sind diese durch Hyperebenen getrennt.

2.2 Logistische Regression

Die Logistische Regression kann als Methode für Regression oder für Klassifikation angewendet werden. In Bezug auf Regression geht es darum, den Einfluss von bestimmten Variablen auf das Zielmerkmal zu bestimmen. In der Klassifikation geht es, wie bereits oben beschrieben, darum Objekte mit bestimmten Merkmalsausprägungen einer Klasse zuzuordnen. Der Zwei-Klassen-Fall des Logit Modells ist weit verbreitet in der medizinischen Statistik, da hier häufig binäre Fragestellungen behandelt werden, wie z.B. Patient überlebt/stirbt[16].

Die Logistische Regression ist im Vergleich zur LDA allgemeiner, indem weniger Annahmen getroffen werden. Im Vergleich zur LDA wird im logistischen Modell keine Annahme über die Verteilung von $x|y$ getroffen, genauso wie auch die a priori Wahrscheinlichkeit zur Zugehörigkeit zu einer Klasse nicht spezifiziert wird[24].

Das Logit-Modell gibt die Wahrscheinlichkeit an, dass das Objekt mit Merkmalsausprägung x in Klasse $y = 1$ kommt[9]:

$$P(y = 1|x) = F(x'\beta) = \frac{\exp(x'\beta)}{1 + \exp(x'\beta)}$$

Auch hier sind die Parameter von $P(y = 1|x)$ i.d.R. unbekannt und werden geschätzt, indem die bedingte Likelihood maximiert wird. Im Fall der LDA geschieht dies durch Maximierung der vollen log Likelihood.
Da mit der Berechnung des Modells lediglich eine Wahrscheinlichkeit resultiert, aber noch keine Zuordnung zu einer Klasse, wird ein Schwellenwert bestimmt, ab dem ein Objekt in Klasse 1 angenommen wird. Üblich ist, diesen Grenzwert auf 0.5 zu setzen (aber auch jeder andere Wert ist möglich),

so dass die Entscheidungsregel

$$\delta(x) = 1 \iff P(y = 1|x) \geq \frac{1}{2}$$

lautet und x Klasse 1 zugeordnet wird, falls $\delta(x) = \beta_0 + \beta^T x \geq 0$, ansonsten Klasse 0.
Resultiert dem Modell nach eine Wahrscheinlichkeit, die größer als 0.5 ist, so wird die Beobachtung zu Klasse 1 zugeordnet. Liegt die Wahrscheinlichkeit unter 0.5, so erfolgt die Zuordnung zu Klasse 0.
Auch die Logistische Regression liefert lineare Klassengrenzen, anschaulich gezeigt durch die log-posterior odds zwischen Klasse 0 und 1:

$$\log \frac{P(y = 1|x)}{P(y = 0|x)} = \log \frac{\exp(\eta)/1 + \exp(\eta)}{1/1 + \exp(\eta)} = \log \exp(\eta) = \beta_0 + \beta_1 x_{i1} + ... + \beta_p x_{ip}$$

2.3 Support Vector Machines

Die Grundidee der Support Vector Machines (SVM) liegt in der Klassifikation durch trennende Hyperebenen. Hierbei soll eine Hyperebene so durch die Daten gelegt werden, dass die Punkte ihrer Klasse nach getrennt werden. Da es für diesen Fall möglicherweise unendlich viele trennende Hyperebenen gibt, stellt sich die Frage nach einer eindeutig und optimal trennenden Hyperbene, also eine Hyperebene, die den Rand zwischen beiden Klassen maximiert[13]. Die Idee hierbei ist, dass je größer der Rand bei den Trainingsdaten ist, desto besser die beiden Klassen voneinander getrennt werden können. Umso genauer ist dann auch die Klassifikation bei den Testdaten[16]. Als Rand bezeichnet man die beiden Hyperebenen, parallel und beidseitig der trennenden, zwischen denen keine Datenpunkte liegen. Die Punkte direkt auf dem Rand, also die Punkte, die der trennenden Hyperebene am nächsten liegen, heißen Support Vectors. In die Identifikation der Support Vectors gehen also *alle* Punkte mit ein. Für die optimale Hyperebene sind schließlich nur die Support Vectors von Bedeutung[29].

In der Praxis treten zwei verschiedene Datenstrukturen auf: Klassen, die linear trennbar sind (Abb.2.1) und Klassen, die nicht linear trennbar sind. Betrachtet wird ein binäres Klassifikationsproblem mit n Beobachtungspaaren $\{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}$ mit $x_i \in \mathbb{R}^p$ und $y_i \in \{-1, 1\}$. Im Folgenden werden Möglichkeiten beschrieben, linear und nicht linear trennbare Klassen zu klassifizieren[16].

2.3.1 Optimal trennende Hyperebenen

Optimal trennbare Hyperebenen können nur bei linear trennbaren Problemen bestimmt werden. Da sie die Voraussetzung für die Support Vector Machines bilden, werden sie hier genauer beschrieben:
Die Hyperebene ist defniert als $x_i^T\beta + \beta_0 = 0$
Nun soll der Rand M um diese Hyperebene maximiert werden, also

$$max_{\beta,\beta_0} M \tag{2.2}$$

unter den Bedingungen

$$||\beta|| = const. \text{ und } y_i(x_i^T\beta + \beta_0) \geq M \;\; \forall i = 1, ..., n \tag{2.3}$$

Für $||\beta|| = 1/M$ sind (2.2) und (2.3) äquivalent zu

$$min_{\beta,\beta_0} \frac{1}{2}||\beta||^2 \text{ mit}$$

$$y_i(x_i^T\beta + \beta_0) \geq 1 \;\; \forall i = 1, ..., n$$

wobei die Bedingungen besagen, dass der Rand um die Hyperebene leer sein soll und eine Breite von $1/||\beta||$ besitzt.

Hierbei handelt es sich um ein quadratisches Optimierungsproblem mit linearen Ungleichungsbedingungen.
Die Lösung dieses quadratischen Optimierungproblems erfolgt mit Hilfe der Lagrange Funktion.

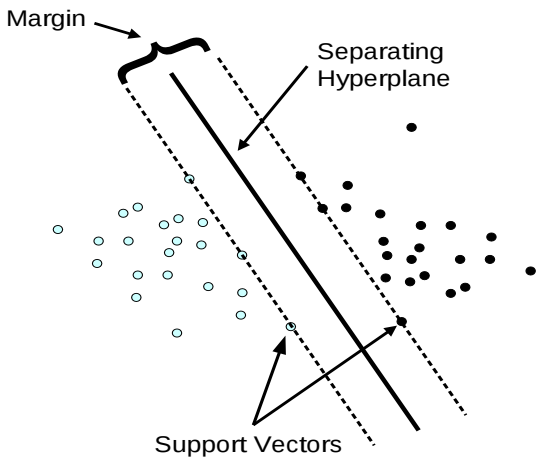

Abbildung 2.1: *Optimal trennende Hyperebene[26]*

Primal Lagrange Funktion:

$$L_P = \frac{1}{2}||\beta||^2 - \sum_{i=1}^{n} \alpha_i[y_i(x_i^T\beta + \beta_0) - 1] \tag{2.4}$$

mit den Lagrange Multiplikatoren α_i[24].
Die partiellen Ableitungen von (2.4) nach β und β_0 liefern:

$$\frac{\partial L_P}{\partial \beta} : \hat{\beta} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i$$

$$\frac{\partial L_P}{\partial \beta_0} = \sum_{i=1}^{n} \alpha_i y_i = 0$$

Die so genannte Wolfe dual Funktion ergibt sich durch das Einsetzen von $\hat{\beta}$ in (2.4):

$$L_D = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j$$

Die Multiplikatoren α_i erfüllen zugleich die Kuhn-Tucker Bedingungen[16]:

- $\beta = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i$
- $0 = \sum_{i=1}^{n} \alpha_i y_i$

- $\alpha_i \geq 0$

- $\alpha_i[y_i(x_i^T\beta + \beta_0) - 1] = 0 \ \ \forall i$

Ist $\alpha_i > 0$ (und somit $y_i(x_i^T\beta + \beta_0) = 1$), so liegt x_i genau auf dem Rand und wird als Support Vector bezeichnet.
Die Funktion der optimal trennenden Hyperebene lautet schließlich:

$$\hat{f}(x) = x_i^T\hat{\beta} + \hat{\beta}_0$$

Um eine neue Beobachtung zu klassifizieren wird

$$\hat{G}(x) = \text{sgn}\,\hat{f}(x)$$

verwendet.

2.3.2 Support Vector Classifier

Bei dem Support Vector Classifier handelt es sich um eine Möglichkeit, auch bei sich überlappenden Klassen eine linear trennende Hyperebene zu definieren. Bei diesem Vorgehen wird wieder eine Hyperebene mit maximalem Rand gesucht und die Überlappungen durch so genannte "slack" Variablen bestraft. "Slack" Variablen $\epsilon = (\epsilon_1, \epsilon_2, ..., \epsilon_l)$ werden definiert zur Bestrafung der Punkte, die auf der falschen Seite der trennenden Hyperebene liegen.
Die Bedingungen für das Optimierungsproblem $min||\beta||$ lauten hier[16]:

$$y_i(x_i^T\beta + \beta_0) \geq 1 - \epsilon_i \ \ \forall i$$

$$\epsilon_i \geq 0, \quad \sum_{i=1}^{l} \epsilon \leq const.$$

2.3.3 Support Vector Machine

Support Vector Machine ist eine zweite Möglichkeit, nicht linear trennbare Daten zu klassifizieren. Mit Hilfe von Kernfunktionen (→ Kernel Trick) werden die Daten aus ihrer originalen Dimension("Input Space") in eine

höhere Dimension ("Feature Space") projiziert, wo das ursprünglich nicht linear trennbare Problem als ein lineares Problem aufgefasst und gelöst werden kann. Anschließend werden die Daten wieder in die Ausgangsdimension zurück transformiert.
Um die Daten in eine höhere Dimension zu projizieren werden diese mit Hilfe einer Funktion h transformiert. Die duale Lagrange Funktion ist dann folgendermaßen definiert:

$$L_D = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j < h(x_i), h(x_j) > \tag{2.5}$$

Die transformierten Daten gehen hier lediglich über das innere Produkt in (2.5) ein[33].
Der so genannte "Kernel Trick" verwendet Projektionen, bei denen das innere Produkt einer bereits bekannten Kernfunktion entspricht und die Projektion h nie explizit angegeben werden muss.

$$< h(x_i), h(x_j) >= K(x_i, x_j)$$

Die Entscheidungsfunktion lautet also:

$$f(x) = \sum_{i=1}^{n} \alpha_i y_i K(x_i, x) + \beta_0$$

Mit der Klassifikationsregel bzgl. $\text{sgn}(f(x))$ wird in die Klassen +1 und -1 sortiert.
Bei der Kernfunktion sollte es sich um eine symmetrisch, (semi) positiv definite Funktion handeln. In der Literatur übliche Kernfunktionen sind[16]:

- Polynom d-ten Grades: $K(x_i, x_j) = (1 + < x_i, x_j >)^d$
- Radial Basis Funktion: $K(x_i, x_j) = \exp(-||x_i - x_j||^2/c)$
- Neural Network: $K(x_i, x_j) = \tanh(\kappa_1 < x_i, x_j > +\kappa_2)$

2.4 LASSO

Mit den Least Absolute Shrinkage and Selection Operators (LASSO) gibt es eine Methode, die die Eigenschaften von Ridge Regression und Variablenselektion vereint. Die Ridge Regression dient zur Modellregularisierung, speziell zur Schrumpfung von Parametern, während eine Variablenselektion durchgeführt wird, um zum einen die Anzahl an Prädiktoren zu verringern und zum anderen nur die wichtigen einzuschließen.
Der LASSO Schätzer ist so konstruiert, dass er die Koeffizienten soweit schrumpft, dass einige von ihnen exakt 0 ergeben können und die zugehörige Variable somit aus dem Modell fällt[27]. Er kann als eine Art Kompromiss von Shrinkage-Methode und Variablenselektion angesehen werden.

2.4.1 LASSO im Linearen Modell

Die Idee des LASSO Schätzers von Tibshirani[32] basiert auf den Annahmen des Linearen Modells.
Betrachtet wird die hierfür übliche Datensituation $(\mathbf{x}_i, y_i)$,$i = 1, 2, ..., n$ mit den Einflussgrößen $\mathbf{x}_i = (x_{i1}, ..., x_{ip})^T$ und der Response Variablen y_i.
Es wird davon ausgegangen, dass die Kovariablen standardisiert sind, also dass gilt:

$$\frac{1}{n}\sum_{i=1}^{n} x_{ij} = 0 \ , \ \frac{1}{n}\sum_{i=1}^{n} x_{ij}^2 = 1$$

und dass die Responsevariable um 0 zentriert ist: $\frac{1}{n}\sum_{i=1}^{n} y_i = \overline{y} = 0$
Diese Standardisierung ist notwendig auf Grund der Abhängigkeit der LASSO Koeffizienten von der Skalierung der Kovariablen und der Wahl des Ursprungs für die Response Variable.
Der LASSO Schätzer minimiert die Residuenquadratsumme unter der Bedingung, dass der Absolutbetrag der aufsummierten Koeffizienten eine vorgegebene Grenze t (mehr dazu unten) nicht überschreitet.

$$\min_{\beta_1,...,\beta_p} \frac{1}{2}\sum_{i=1}^{n}(y_i - \sum_{j=1}^{p} x_{ij}\beta_j)^2 \tag{2.6}$$

unter der Nebenbedingung

$$\sum_{j=1}^{p} |\beta_j| \leq t,\ t \geq 0 \tag{2.7}$$

Ohne (2.7) handelt es sich um den gewöhnlichen KQ-Schätzer. Wenn also der KQ-Schätzer die Bedingung (2.7) erfüllt, entspricht dieser genau dem LASSO Schätzer.
Äquivalent zu (2.6) ist das damit eng zusammenhängende Optimierungsproblem, nämlich die penalisierte Formulierung des LASSO Schätzers[27]

$$\min_{\beta_1,\ldots,\beta_p} \frac{1}{2} \sum_{i=1}^{n} (y_i - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \tag{2.8}$$

bei dem $\lambda \sum_{j=1}^{p} |\beta_j|$ den Strafterm darstellt.
Der Zusammenhang zwischen (2.6) und (2.8) besteht darin, dass für gegebenes λ ($0 \leq \lambda < \infty$) ein $t \geq 0$ existiert, so dass beide Optimierungsprobleme zur selben Lösung führen.
Die Stärke der Schrumpfung der Koeffizienten β hängt von der Wahl des so genannten Tuning- bzw. Lassoparameters t ab. Je kleiner t, desto stärker werden die Koeffizienten geschrumpft. In Bezug auf (2.8) und den Penalisierungsparameter λ ergibt sich ein umgekehrter Zusammenhang: Je größer λ, desto stärker die Schrumpfung der LASSO Schätzungen.
Auf Grund der oben beschriebenen Abhängigkeit zwischen der Anzahl der selektierten Kovariablen und dem Tuningparameter t gilt es, eine optimale Schätzung für t zu finden. Tibshirani[32] stellt hier drei Möglichkeiten zur Schätzung vor: Das Kreuzvalidierungsverfahren, die generalisierte Kreuzvalidierung(GCV) und eine analytische, unverzerrte Risikoschätzung.

Wir halten $t \geq 0$ fest. Bei dem Optimierungsproblem (2.6) handelt es sich um ein quadratisches Optimierungsproblem mit 2^p Ungleichungsbedingungen[15]. Dieses numerisch zu lösen, stellt keine triviale Aufgabe dar und den Algorithmus zu lösen, wie er in [32] vorgestellt wird, zeichnet sich als ein computerintensives Vorgehen ab.

2.4.2 Modifizierte Algorithmen

Neben der geringen Rechengeschwindigkeit treten im ursprünglichen LASSO-Algorithmus auch Probleme auf, sobald die Anzahl p der Variablen der Anzahl n der Beobachtungen nahe kommt bzw. diese übersteigt. Tritt die, typisch für Microarray Daten, $p >> n$ Situation auf, so können mit LASSO maximal n Variablen ausgewählt werden[35]. Ein weiterer Nachteil bei der ursprünglichen LASSO Schätzung liegt zudem darin, dass bei einer Gruppe von stark korrelierten Prädiktoren dazu tendiert wird, lediglich eine Einflussgröße dieser Gruppe auszuwählen statt mehrere[18].
Den oben genannten Problemen wird mit zwei neuen Algorithmen begegnet: Osborne et al.[27] stellen eine neue Möglichkeit vor, die auch für die Datensituation $p >> n$ geeignet ist.
Efron et al.[8] haben mit LARS(Least Angle Regression, wobei das "S" für "Lasso" und "Stagewise" steht) eine Methode entwickelt, die einerseits deutlich weniger Iterationen benötigt und andererseits für hochdimensionale Daten problemlos verwendet werden kann. Eine einfache Veränderung dieses Algorithmus führt zu LASSO Schätzungen, die aber deutlich weniger Rechenzeit benötigen als der originale Algorithmus[8].
Die Least Angle Regression beginnt mit der Wahl des Startwerts
$\hat{\mu} = X\hat{\beta} = 0$, d.h. alle Koeffizienten werden zu Beginn gleich 0 angenommen. Gesucht ist die Kovariable, die am stärksten mit dem Response korreliert. Nun geht man soweit in Richtung dieses Prädiktors, bis ein weiterer Prädiktor dieselbe Korrelation mit dem aktuellen Residuenvektor hat. Anschließend geht LARS in die Richtung weiter, die den Winkel zwischen den beiden ausgewählten Prädiktoren halbiert, bis eine dritte Variable mit der höchsten Korrelation hinzukommt. Die Menge aller ausgewählten Variablen nennt man "Active Set".
Die Veränderung, die beim oben beschriebenen LARS Algorithmus durchgeführt wird, um exakte LASSO Schätzer zu erhalten, führt dazu, dass hinzugefügte Variablen wieder aus dem Active Set entfernt werden können[8]. Somit durchläuft der LARS Algorithmus weniger Iterationen als die LASSO Modifikation, welche im Gegensatz zum orignalen Algorithmus[32] trotzdem

wesentlich weniger rechenintensiv verläuft.

2.4.3 LASSO in GLMs

Im Fall der in Kapitel 4 vorgestellten Daten handelt es sich um einen binären Response. Das Lineare Modell, welches die Voraussetzung für die vorangehenden Definitionen war, kann also nicht mehr als Grundlage angesehen werden. Benötigt wird ein Generalisiertes Lineares Modell.
Analog zu LASSO (2.8) wird nun die penalisierte log Likelihood
(log Likelihood an Stelle der Residuenquadratsumme) maximiert:

$$\max_{\beta_0,\beta}[\frac{1}{N}\sum_{i=1}^{N}\{I(g_i=1)\log p(x_i)+I(g_i=2)\log(1-p(x_i))-\lambda P_\alpha(\beta)\}] \quad (2.9)$$

Der Bestrafungsterm aus (2.9) hat die Form:

$$\begin{aligned} P_\alpha(\beta) &= (1-\alpha)\frac{1}{2}||\beta||_{l_2}^2+\alpha||\beta||_{l_1} \\ &= \sum_{j=1}^{p}[\frac{1}{2}(1-\alpha)\beta_j^2+\alpha|\beta_j|] \end{aligned} \quad (2.10)$$

mit α als "Elastic Net mixing Parameter", $0<\alpha\leq 1$.
Wählt man $\alpha=1$ so ergibt sich $\lambda|\beta_j|$ als Strafterm und man erhält LASSO Schätzungen.
Friedman, Hastie und Tibshirani gehen weiter so vor, dass eine quadratische Approximation bzgl. der log Likelihood gebildet wird[12]:

$$l_Q(\beta_0,\beta)=-\frac{1}{2N}\sum_{i=1}^{N}w_i(z_i-\beta_0-x_i^T\beta)^2+C(\tilde{\beta}_0,\tilde{\beta})^2 \quad (2.11)$$

mit
$z_i=\tilde{\beta}_0+x_i^T\tilde{\beta}+\frac{y_i-\tilde{p}(x_i)}{\tilde{p}(x_i)(1-\tilde{p}(x_i))}$ als "working response" und
$w_i=\tilde{p}(x_i)(1-\tilde{p}(x_i))$ als Gewichte
$\tilde{p}(x_i)$ wird jeweils bei den aktuellen Schätzern der Parameter $(\tilde{\beta}_0,\tilde{\beta})$ berechnet.

Neue Parameterschätzungen (β_0, β) erhält man durch Minimieren von (2.11). Um schließlich das penalisierte, gewichtete KQ Problem

$$\min\{-l_Q(\beta_0, \beta) + \lambda P_1(\beta)\}$$

zu lösen, wendet man "coordinate descent" an.

Kapitel 3

Anwendung an Mammmographie - Screeningdaten

3.1 Das Mammakarzinom

Krebs gehört weltweit zu den 10 häufigsten Todesursachen. Unter Frauen ist das Mammakarzinom (Brustkrebs) mit einem Anteil von 16% die am häufigsten tödlich verlaufende Krebsart[34].
Derzeit anerkannte Risikofaktoren für das Auftreten von Brustkrebs sind u.a. das Alter, familiäre Belastung/genetische Disposition, ionisierende Strahlung (z.B. Röntgenstrahlung), aber auch die Art der Lebensführung, wie Alkoholkonsum oder postmenopausales Übergewicht[1][30].

Man unterscheidet zwischen benignen und malignen Brusterkrankungen. Benigne bedeutet gutartig, d.h. der Tumor verdrängt durch sein Wachstum Gewebe, dringt aber nicht in dieses ein und streut nicht im Körper. Ist der Tumor maligne, so infiltriert dieser das Gewebe, kann es zerstören und über Blut- bzw. Lymphwege zu Lymphknoten oder Organen gelangen (Metastasenbildung).

[1] Zeit nach der Menopause

Bösartige Tumore werden, basierend auf ihrer TNM-Klassifikation in ein Stadium eingeteilt was die Grundlage für die Therapie der Wahl und die Beurteilung des Therapieerfolges bildet. T(Tumor) beschreibt die Ausdehnung des Primärtumors, N(Nodulus), das Fehlen oder Vorhandensein und in diesem Fall die Ausdehnung von Lymphknotenmetastasen und M(Metastase) das Fehlen oder Vorhandensein von Fernmetastasen.
Bezüglich der Ausdehnung des Primärtumors unterscheidet man grob in situ[2] und invasive[3] Karzinome.

Die Größe eines Karzinoms und das Ausmaß der Lymphknotenmetastasierung gehören zu den wichtigsten prognostischen Kriterien des Mammakarzinoms[30]. Sowohl die Tumorgröße, als auch der Lymphknotenstatus haben unabhängig voneinander einen negativen Einfluss auf die Überlebensrate. Auch die Anzahl der befallenen Lymphknoten korreliert direkt mit der Größe des Primärtumors, der Rezidiv(Rückfall)- und der Überlebensrate. Leider ist die klinische Einschätzung des Befalls extrem unzuverlässig und sowohl falsch positive als auch falsch negative Befunde treten in hohem Maße auf[5].
Die Metastasierung[4] des Mammakarzinoms tritt in der Regel schon frühzeitig auf. Fernmetastasen korrelieren mit der Tumorgröße, dem Malignitätsgrad und der lymphogenen Ausbreitung. Der Großteil der an Brustkrebs sterbenden Frauen hat weit gestreute Metastasen. Am häufigsten werden diese in Knochen, Lunge oder Leber lokalisiert.
Auch der Nachweis von Östrogen- oder Progesteronrezeptoren im Tumorgewebe hat Auswirkungen auf die Prognose[30].
Wie bei jedem Tumor des menschlichen Körpers, ist die Früherkennung eines Mammakarzinoms ein entscheidender Faktor hinsichtlich des Behandlungserfolges.

[2]Carcinoma in situ: lokal begrenzter Krebsherd=frühestes Krebsstadium
[3]Man spricht von einem invasiven Tumor, wenn dieser in das umliegende Gewebe hineinwächst
[4]Absiedlungen eines Tumors in entferntem Gewebe

3.2 Mammographie

Eine Mammographie ist eine Röntgenuntersuchung der Brust, die die Mamma in zwei Ebenen darstellt(kraniokaudaler[5] und lateraler[6] Strahlengang). Wegen der weichen Strahlung erhält man eine ausgeprägte Feinstrukturenzeichnung. Die Mammographie ist besonders von Vorteil in der Erkennung kleiner, nicht tastbarer Karzinome und ist Goldstandard im Bereich der bildgebenden Verfahren zur Früherkennung [30].
Mit Hilfe dieser Röntgenaufnahmen können Verkalkungen im Gewebe erkannt werden, welche Hinweise auf einen gut- oder bösartigen Befund geben können. Fettgewebe zeigt sich als relativ dunkler Bereich, während Zysten, Verkalkungen oder Karzinome röntgendichte Strukturen darstellen und auf der Aufnahme zu einer Verschattung[7] führen. Im Hinblick auf Größe, Form und Muster des so genannten Mikrokalks kann ein Urteil über Malignität oder Benignität erfolgen. Maligne Tumoren sind meist unscharf begrenzt und zeigen häufig sternförmige Ausläufer, während benigne Tumoren (z.B. Zysten/Firbome) homogen dicht und glatt begrenzt sind.

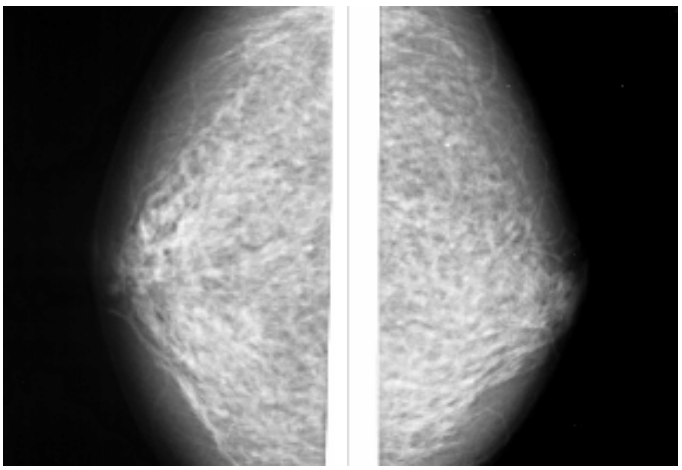

Abbildung 3.1: *Mammographie der linken und rechten Brust*

[5]kraniokaudal: Vom Kopf ausgehend in Richtung Füße
[6]seitlich (im 45° Winkel zum kraniokaudalen Strahlengang)
[7]Im Röntgenbild bezeichnet man schwarz als Aufhellung und weiß als Verschattung

Die Erstellung einer Mammographie ist Teil des Screenings, welches eine routinemäßige Reihenuntersuchung an Frauen ohne Symptome bezeichnet. Es dient zusammen mit weiteren Untersuchungen zur frühzeitigen Erkennung von Brustkrebs.
Die übliche Vorgehensweise zur Beurteilung eines Mammograms ist eine Doppelbefundung durch zwei Ärzte, die unabhängig voneinander die Röntgenaufnahme begutachten. Die Betrachtung durch zwei Ärzte ("double reading") führt zu einer höheren Krebserkennungsrate als die Begutachtung durch lediglich einen Arzt ("single reading")[14].
Eine neue Möglichkeit bieten CAD-Systeme (Computer-Aided Detection), die den Radiologen bei den Auswertungen unterstützen. Hierbei werden die digitalisierten Mammographien mit Hilfe einer Computersoftware analysiert, verdächtige Stellen identifiziert und markiert[25].

3.3 Die Daten

Bei den dieser Arbeit zugrundeliegenden Daten handelt es sich um Mammographie-Screeningdaten, die auf Basis der DDSM (Digital Database for Screening Mammography) erhoben wurden. Bei der DDSM[8] handelt es sich um eine öffentlich zugängliche Datenbank der Mammographie-Bildanalyse-Forschungsgemeinschaft der Universität Süd Florida. Ziel dieser Homepage ist die Vereinfachung der Forschung in Bezug auf CAD-Systeme anhand einer einheitlichen und frei zugänglichen Quelle an digitalisierten Mammographien.
Aus vier verschiedenen Krankenhäusern in den USA wurden die Röntgenaufnahmen bereits in den frühen 90er Jahren bezogen und bis 1999 vervollständigt. Dieser Datenbestand setzt sich zusammen aus insgesamt 2620 Beobachtungen unterteilt in "normal", "Krebs", "gutartig" und "gutartig ohne Rückruf". "Normal" bezeichnet Mammographien, die als normal befunden wurden und vier Jahre später ebenfalls wieder in die Kategorie normal eingestuft werden konnten. "Krebs" bezeichnet Fälle, bei denen ein Tumor

[8] http://marathon.csee.usf.edu/Mammography/Database.html

histologisch nachgewiesen wurde. "Gutartig" beinhaltet Aufnahmen, in denen Auffälligkeiten entdeckt wurden, die nach einer erneuten Untersuchung bestätigt werden konnten. "Gutartig ohne Rückruf" sind Beobachtungen mit einer nennenswerten Abnormalität, die jedoch keine erneute Untersuchung erforderten.
Jede Beobachtung besteht aus vier Aufnahmen, je zwei von jeder Brust aus kraniokaudaler und lateraler Perspektive. Zusätzliche Informationen zu jeder Mammographie sind: Datum der Studie, Alter der Patientin zu diesem Zeitpunkt, die Brustdichte nach ACR[9] Richtlinie, Datum an dem die Mammographie digitalisiert wurde und Informationen über die Auflösung des Bildes[17]. Zusätzlich wird zu jedem nicht normalen Fall noch die Klassifikation gemäß der BI-RADS Kodierung (Tab. 3.1) angegeben. Sowohl die Brustdichte als auch die BI-RADS Kodierung werden von einem Radiologie-Experten bestimmt.

BI-RADS	Befund	Karzinomrisiko
1	"nothing to comment on"	0%
2	gutartig	0%
3	wahrscheinlich gutartig, kontrollbedürftig	$< 2\%$
4	suspekt, abklärungsbedürftig	2-90%
5	karzinomverdächtig	$> 90\%$

Tabelle 3.1: *BI-RADS (Breast Imaging Reporting and Data System)[30]*

In den im Folgenden verwendeten Daten werden lediglich Fälle mit malignen oder benignen Verkalkungen einbezogen. Der daraus resultierende Datensatz umfasst insgesamt 1347 Beobachtungen mit jeweils 453 Variablen. Als Response dient die binäre Variable "Severity", die angibt, ob es sich um eine maligne oder benigne Verkalkung handelt. Insgesamt enthält der Datensatz 610 Beobachtungen mit malignen und 737 mit benignen Verkalkungen. Die weiteren 452 Variablen sind metrisch und definieren z.B. einfache Gruppierungsvariablen wie die Anzahl der Partikel, die Form wie Größe oder

[9] American College of Radiology

Kreisform, die Erscheinung wie ihre Dichte oder auch die Verteilung der Teilchen anhand ihrer Entfernung vom Mittelpunkt des Herdes.

3.4 Anwendung der Methoden

Die in Kapitel 1 und 2 beschriebenen Methoden werden nun auf die beschriebenen Mammographie-Screeningdaten angewendet.
Alle Berechnungen wurden mit der frei zugänglichen Software `R` ausgeführt[10], wobei u.a. die Pakete `WilcoxCV` [3], `ROC`, `e1071` und `glmnet` verwendet wurden.
Bei der in Kapitel 3 erwähnten Trennung des Datensatzes in einen Lern- und Testdatensatz kann es zu einer Verzerrung der Ergebnisse kommen (Selektionsbias). Diese Verfälschung der Ergebnisse kann entstehen, da die Klassifikationsregel auf dem Lerndatensatz aufgestellt wird, also lediglich einem kleinen Teil des Gesamtdatensatzes, der nicht alle Informationen enthält[22]. Entgegenwirken kann man dem durch Kreuzvalidierung, d.h. der gesamte Datensatz wird in verschiedene Kombinationen getrennt und die Anwendung auf jeder dieser Kombinationen ausgeführt[31]. Hier wird wegen seiner geringen Varianz der Fehlerrate die Monte-Carlo Kreuzvalidierung (MCCV) verwendet. Bei Anwendung der MCCV entsteht der Lerndatensatz $\mathcal{L}$ aus zufällig aus der Gesamtstichprobe ohne Zurücklegen gezogenen Beobachtungen. Der Testdatensatz $\mathcal{T}$ besteht aus den übrigen Beobachtungen. Insgesamt werden 500 Iterationen durchgeführt, d.h. 500 verschiedene Lern- und Testdatensätze generiert.
Hinsichtlich der Größe von Lern- und Testdatensatz gibt es keine Richtlinien. Gängige Varianten sind eine Trennung im Verhältnis $2:1$, $4:1$ oder $9:1$ [2]. Wichtig ist, dass der Testdatensatz letztendlich groß genug ist, um eine adäquate Trennung zwischen den Klassen zu ermöglichen[7]. Demnach eignet sich eine Trennung im Verhältnis $9:1$ lediglich für entsprechend große Datensätze, während ein $2:1$ Schema auch für relativ wenige Beobachtungen noch zu sinnvollen Ergebnissen führt. Vor allem hängt das gewählte

[10] http://www.r-project.org

Verhältnis $n_L : n_T$ aber vom jeweiligen Ziel der Studie ab. Handelt es sich lediglich um einen Methodenvergleich, so würde schon eine Trennung von $2 : 1$ genügen, während ein größeres Verhältnis gewählt werden sollte, sobald es um die Vorhersagegenauigkeit an sich geht[2]. Da hier ein Vergleich von Feature Selektionsmethoden behandelt wird, wird der Datensatz in einem Verhältnis von $2 : 1$ getrennt.

Für die Aufstellung der Klassifikationsregel darf ausschließlich der Lerndatensatz verwendet werden. Die Prädiktion beruht schließlich auf dem Testdatensatz.

Im Folgenden werden die vier in Kapitel 2 beschriebenen Variablenselektionsverfahren miteinander verglichen. Die Effizienz der verschiedenen Methoden wird anhand der Klassifiaktionsgenauigkeit beurteilt, gemessen an der AUC (Area Under the ROC Curve).
Zur besseren Übersicht wird die Selektion nach dem Ranking der Variablen lediglich mit "Ranking" bezeichnet, der Korrelationsansatz, in dem die Korrelation mit der stärksten Variablen betrachtet wird mit "Korrelation I", der mit der zweitstärksten "Korrelation II" und der mit der drittstärksten "Korrelation III".
Lineare Diskriminanzanalyse und Support Vector Machine arbeiten mit den resultierenden Variablen aus den Selektionsverfahren, wobei für die SVM der Radial Basis Funktions Kern ohne Tuning der Hyperparameter verwendet wird. Die Logistische Regression wird nach einer Vorwärtsselektion durchgeführt. Als ein Ansatz mit integrierter Variablenselektion wird bei LASSO der ganze Lerndatensatz übergeben. Die Variablen werden standardisiert und eine 10-fache Kreuzvalidierung zur optimalen λ Bestimmung durchgeführt.

Kapitel 4

Ergebnisse

In den folgenden Abschnitten werden die Ergebnisse der Selektions- und Klassifikationsmethoden zusammenfassend dargestellt. Als Maß zur Bestimmung der Güte des Selektionsverfahrens wird die Area under the ROC curve (AUC) angegeben. Der Mean Squared Error (MSE) wurde ebenfalls berechnet und ist im Anhang B aufgeführt, ebenso die Tabellen der ungerundeten Ergebnisse und die Darstellung der Verteilung von AUC - und MSE - Werten durch Boxplots.
Zuerst werden die Resultate der Variablenselektionsverfahren aufgeführt. Anschließend erfolgt die Betrachtung der Klassifikationsergebnisse ohne vorherige Variablenselektion. Die darauffolgenden Abschnitte erläutern die Klassifikationsergebnisse der einzelnen Methoden sowie einen Vergleich der Methoden.

4.1 Variablenselektion

Aus den vier Variablenselektionsverfahren Ranking, Korrelation I, Korrelation II und Korrelation III resultieren je 10 verschiedene Variablenkombinationen.
Beim Ranking ergeben sich 5 bis 452 Variablen, welche den Anteilen von 1%, 2%, 5%, 8%, 10%, 15%, 22%, 55%, 70% und 100% der nach der AUC sortierten Variablenmenge entsprechen.
Bei den Korrelationsansätzen sind es durchschnittlich 4.5 bis 451.7 Variablen, was den Korrelationsstärken von 0.02, 0.05, 0.20, 0.35, 0.45, 0.60, 0.75, 0.95, 0.98 und 1 entspricht.
Der Zusammenhang von Korrelationsstärke und resultierender Variablenanzahl wird in Abb. 4.1 verdeutlicht. Die Abbildung gibt an, wieviele Variablen jeweils aus der vorgegebenen Korrelationsgrenze resultieren, basierend auf den nach AUC-Wert geordneten Variablen.

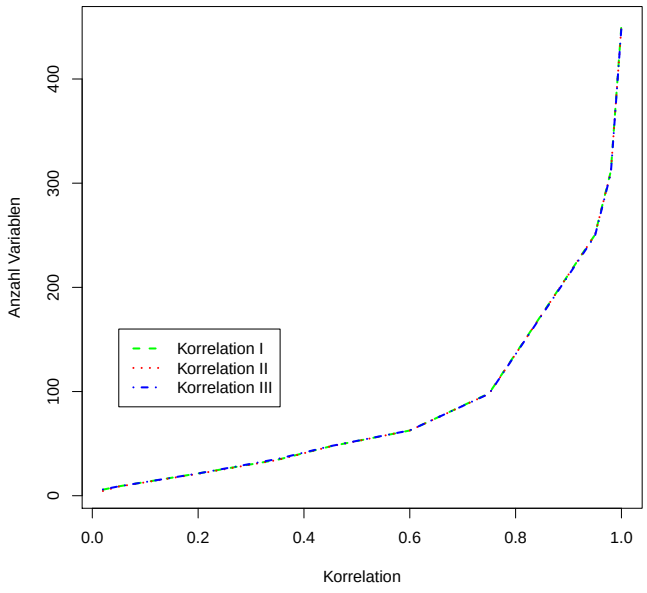

Abbildung 4.1: *Variablenanzahl in Abhängigkeit der Korrelation*

Erkennbar ist, dass sich die resultierenden Variablenmengen der drei Korrelationsansätze kaum unterscheiden. Unabhängig von der Betrachtung der Korrelation mit der ersten, zweiten oder dritten Variablen des Rankings, lässt sich also keine Auswirkung auf die Variablenanzahl feststellen. Die vermutete hohe Korrelation der Variablen untereinander wird bestätigt durch den exponentiellen Verlauf der Variablenanzahl, der in der Abbildung erkennbar ist.
Etwa ein Viertel der Variablen weist untereinander eine Korrelation auf, die zwischen 0.98 und 1 liegt. Zwischen 0.95 und 1 sind es knapp die Hälfte und für eine Korrelation über 0.75 drei Viertel aller Variablen.

4.2 Klassifiaktion ohne vorherige Selektion

Um einen möglichen Effekt der Variablenselektion sichtbar zu machen, dient Abb. 4.2. Hier wird die Verteilung der AUC-Werte für die gesamte Variablenanzahl dargestellt.

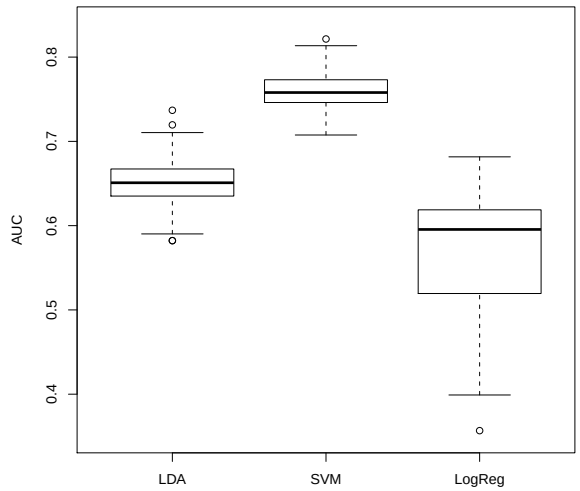

Abbildung 4.2: *AUC bei voller Variablenanzahl*

Bei Übergabe der kompletten Variablenmenge erreicht die SVM mit einer durchschnittlichen AUC von 0.760 das beste Resultat, die LDA mit 0.651 das zweitbeste und die LogReg mit 0.573 das schlechteste. Wie anhand der folgenden Ergebnisse zu sehen ist, verbessert die Variablenselektion die durchschnittliche AUC im Vergleich zum vollen Modell. Eine Ausnahme stellt die SVM dar.

4.3 Betrachtung der Klassifikationsergebnisse

4.3.1 Lineare Diskriminanzanalyse

In jedem der vier Variablenselektionsansätze liefern die Modelle der Linearen Diskriminanzanalyse mit den jeweils meisten Variablen die niedrigsten AUC-Werte. Beim Ranking ist dies das Modell mit voller Variablenanzahl, also 100% der Rankingvariablen. Bei den Korrelationsansätzen I, II und III handelt es sich jeweils um jene Variablenmengen, die untereinander eine Korrelation von bis zu 1 aufweisen.

	Lineare **D**iskriminanz **A**nalyse									
Prozent	1	2	5	8	10	15	22	55	70	100
Anzahl	5	9	23	36	45	68	99	249	316	452
Korrelation	0.02	0.05	0.20	0.35	0.45	0.60	0.75	0.95	0.98	1
Anzahl (ϕ)	5.4	8.9	21.3	34.7	47.4	62.6	98.0	250.1	310.7	450.7
Ranking	0.754	0.750	0.756	**0.760**	0.760	0.750	0.741	0.704	0.687	0.651
K I	0.695	0.666	0.715	0.762	**0.765**	0.761	0.747	0.688	0.679	0.651
K II	0.754	0.752	0.751	0.762	**0.764**	0.762	0.746	0.688	0.678	0.648
K III	0.672	0.652	0.718	0.742	0.750	**0.753**	0.740	0.684	0.677	0.648

Tabelle 4.1: *Gerundete AUC Mittelwerte der LDA Messergebnisse für die verschiedenen Korrelationen und prozentualen Anteile. Die jeweils maximalen AUC-Werte sind fettgedruckt*

Die AUC erreicht ihr Maximum in allen vier Ansätzen bei 36 bis 63 Variablen und sinkt anschließend kontinuierlich ab. Innerhalb des Rankings liegt es bei einem Anteil von 8%. Für Korrelation I und II liegt es bei einer Korrelationsstärke von 0.45 und bei Korrelation III bei 0.60.
Eine "Parallelität" des Verlaufes der AUC Kurven ist beim Vergleich der Korrelationsansätze erkennbar (Abb.4.3). Während allerdings Korrelation I und III nur sehr geringe AUC-Werte für die ersten drei Variablenkombinationen liefern, startet Korrelation II mit einer guten Anpassung an die Daten auch schon bei geringer Variablenanzahl. Gemeinsam haben die drei Korrelationsansätze, dass sie jeweils für eine Korrelation von 0.02 eine bessere Vorhersagegenauigkeit besitzen, als für eine Korrelation von 0.05. Auch der Ranking Ansatz zeigt für einen Anteil von 1% noch einen leicht besseren AUC-Wert als für 2%. Anschließend, bei einem Anteil von 5% bzw. einer Korrelation von 0.20 kommt es bei allen vier Methoden zu einer höheren bzw. bei Korrelation II zu einer gleichbleibenden AUC.
Nachdem jeder der vier Ansätze sein Maximum bei einer Variablenanzahl unter 70 erreicht, fallen die AUC-Werte anschließend linear im Ranking und näherungsweise linear in den Korrelationsansätzen ab. Dies muss nicht an den Selektionsverfahren an sich liegen. Wahrscheinlicher ist ein Zusammenhang mit der LDA, welche für den Umgang mit größeren Variablenmengen nicht geeignet ist.

Ein Urteil über den "besten" Selektionsansatz in Bezug auf die LDA zu fällen ist nicht möglich. Deutlich ist allerdings, dass der Korrelation III-Ansatz, in welchem die beiden AUC-stärksten Variablen fehlen, am schlechtesten ausfällt.
Wenn Korrelation I auch teilweise bessere AUC-Werte erreicht, so bestimmt Korrelation II innerhalb der Korrelationsansätze die zur Klassifiaktion optimaleren Variablenkombinationen, was sich dadurch zeigt, dass aus Korrelation II für geringere Variablenanzahlen deutlich höhere AUCs resultieren. Korrelation I schneidet trotz des Vorhandenseins der AUC-stärksten Variablen für geringe Korrelationen deutlich schlechter ab als Korrelation II, zeigt aber dann ab einer Korrelation von 0.35 einen fast identischen Verlauf.

Die AUC-Werte des Rankings werden bis zu einer Korrelation von 0.75 größtenteils von AUCs aus Korrelation I und II überboten, liegen danach aber über diesen. Die über das Ranking resultierende Variablenmenge führt bei größerer Variablenanzahl zu besseren Klassifikationsresultaten, als die Korrelationsansätze.

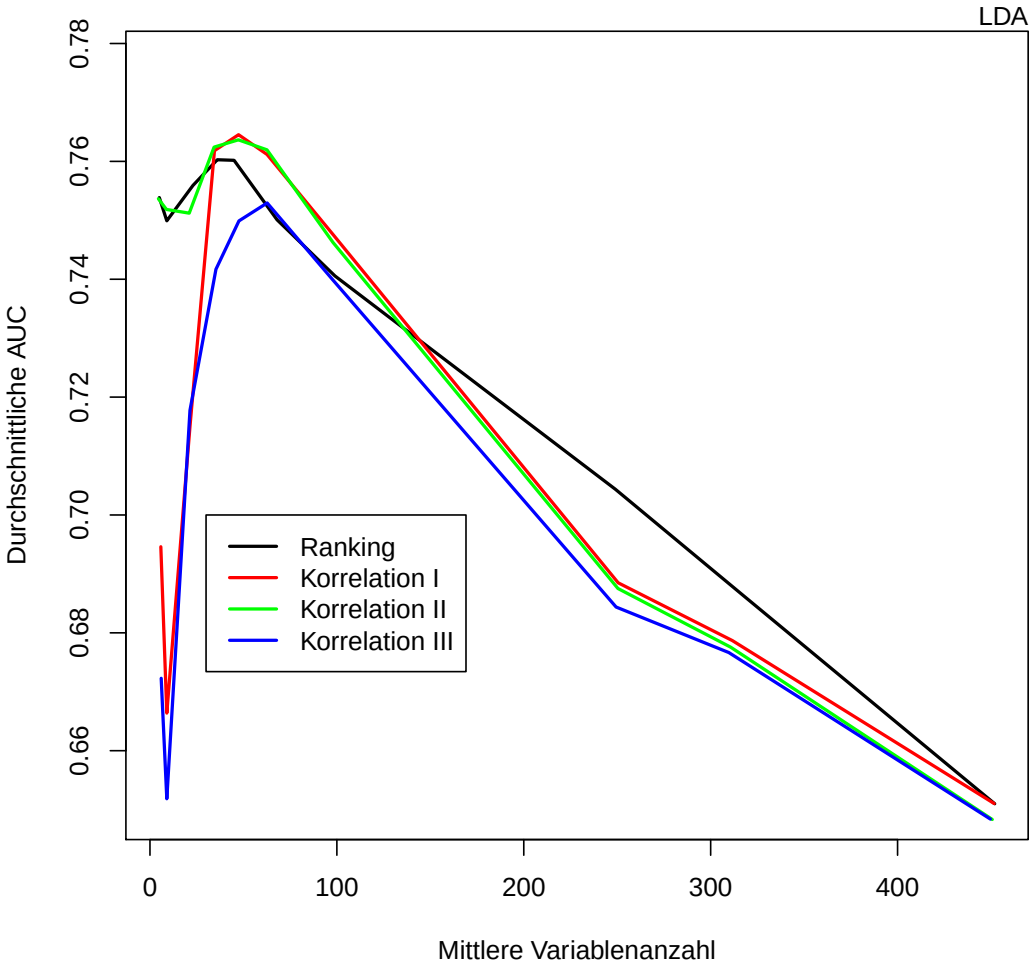

Abbildung 4.3: *Selektionsmethoden anhand von LDA*

Hastie, Tibshirani und Friedman[16] sprechen im Kontext der Klassifikationsgenauigkeit, die man an der Fehlerrate misst, von dem so genannten "Badewanneneffekt", wenn die Fehlerrate für eine geringe Variablenanzahl hoch ist, für steigende Variablenanzahl abfällt und schließlich bei immer größer werdender Anzahl wieder ansteigt. Mit der AUC als Kriterium, kommt es hier zu einem ähnlichen Effekt: Für geringe Anzahlen zeigt sich keine gute Anpassung bzw. eine niedrige AUC. Diese steigt mit Hinzunahme weiterer Variablen und fällt wieder wenn zu viele Variablen aufgenommen werden.
Der typische "Badewanneneffekt" ist gut erkennbar bei Betrachtung der MSE-Werte anstelle der AUC. Siehe hierzu Anhang B, Abb.B.3.

4.3.2 Support Vector Machine

Bei einer Klassifikation mittels Support Vector Machine zeigen die vier Variablen-Selektionsverfahren bei geringen Variablenanzahlen ihre jeweils schlechteste Anpassung, d.h. den niedrigsten AUC-Wert .

	Support Vector Machine									
Prozent	1	2	5	8	10	15	22	55	70	100
Anzahl	5	9	23	36	45	68	99	249	316	452
Korrelation	0.02	0.05	0.20	0.35	0.45	0.60	0.75	0.95	0.98	1
Anzahl (ϕ)	5.4	8.9	21.3	34.7	47.4	62.6	98.0	250.1	310.7	450.7
Ranking	0.743	0.751	0.757	0.760	0.761	0.760	0.760	**0.773**	0.771	0.760
K I	0.758	0.745	0.733	0.764	0.771	**0.777**	0.772	0.766	0.764	0.760
K II	0.757	0.756	0.752	0.764	0.770	**0.777**	0.772	0.765	0.764	0.759
K III	0.727	0.716	0.720	0.747	0.760	**0.766**	0.763	0.757	0.756	0.752

Tabelle 4.2: *Gerundete AUC Mittelwerte der SVM Messergebnisse für die verschiedenen Korrelationen und prozentualen Anteile. Die jeweils maximalen AUC-Werte sind fettgedruckt*

Die minimale AUC liegt beim Ranking bereits bei 1% der Variablen. Korrelation I und Korrelation II erreichen ihre geringste AUC bei einer Korrelationsstärke von 0.20, während Korrelation III den niedrigsten AUC-Wert der vier Selektionsverfahren bereits bei einer Korrelationsstärke von 0.05 zeigt.

Basierend auf dem Ranking erreicht die SVM ihre maximale AUC bei Verwendung von 55% der Variablen. Alle Korrelationsansätze besitzen ihr Maximum einheitlich bei einer Korrelationsstärke von 0.60. Keine der AUC Kurven fällt nach Erreichen des Maximums rapide ab, insbesondere die drei Korrelationsansätze haben einen sehr ähnlichen Verlauf und sinken nur langsam.

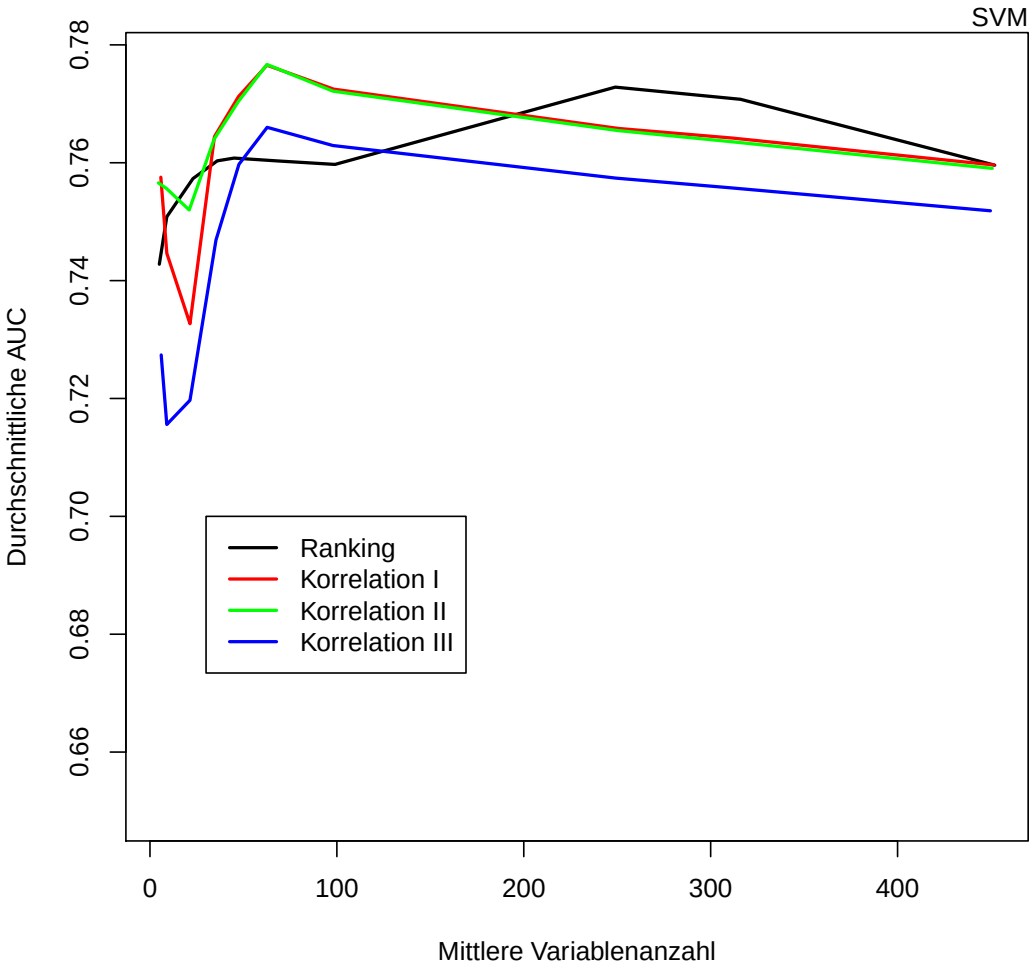

Abbildung 4.4: *Selektionsmethoden anhand von SVM*

Auch vor Erreichen ihres jeweiligen Maximums verlaufen die Korrelationsansätze I, II und III ähnlich, ganz im Gegensatz zur Rankingkurve.

Die AUC-Werte sinken bereits bei der Korrelationsgrenze von 0.05. Während Korrelation III unmittelbar danach wieder ansteigt, erhöhen sich die AUC-Werte im Korrelation I- und Korrelation II-Ansatz erst ab einer Korrelation von 0.35.

Die Korrelationskurven verlaufen annähernd parallel. Ab einer Korrelation von 0.35 verlaufen Korrelation I und II sogar nahezu identisch, Korrelation I weist dabei minimal höhere AUC-Werte auf. Korrelation III zeigt bei allen Korrelationsgrenzen schwächere AUC-Werte auf, als Korrelation I und II.

Die AUC-Werte des Ranking-Ansatzes verlaufen bis zu einem Anteil von 22% der gesamten Variablenmenge unterhalb der Korrelationskurven, dies ändert sich ab einem Anteil von 55%. Ab dieser Variablenmenge liegt das Ranking über den Korrelationsansätzen und sinkt erst bei voller Variablenanzahl wieder auf dieselbe AUC wie Korrelation I.

Auch in Bezug zur SVM kristallisiert sich kein global bester Selektionsansatz heraus. Für Korrelationsstärken bis 0.35 dominiert Korrelation II die Korrelationsansätze und wird anschließend, wenn auch nur minimal, von Korrelation I übertroffen. Ab einer Variablenanzahl von etwa 250 liegen die AUC-Werte des Rankings über denen der Korrelationsansätze.

4.3.3 Logistische Regression und LASSO

Sowohl bei Anwendung der Logistischen Regression, als auch bei LASSO, wird keine der vorherigen Variablenselektionsmethoden Ranking, Korrelation I, II oder III durchgeführt.
Die Logistische Regression beruht auf vorheriger Vorwärtsselektion. Die 500 resultierenden Variablenkombinationen besitzen 3 bis 19 Variablen mit einem Mittelwert von 6.8.

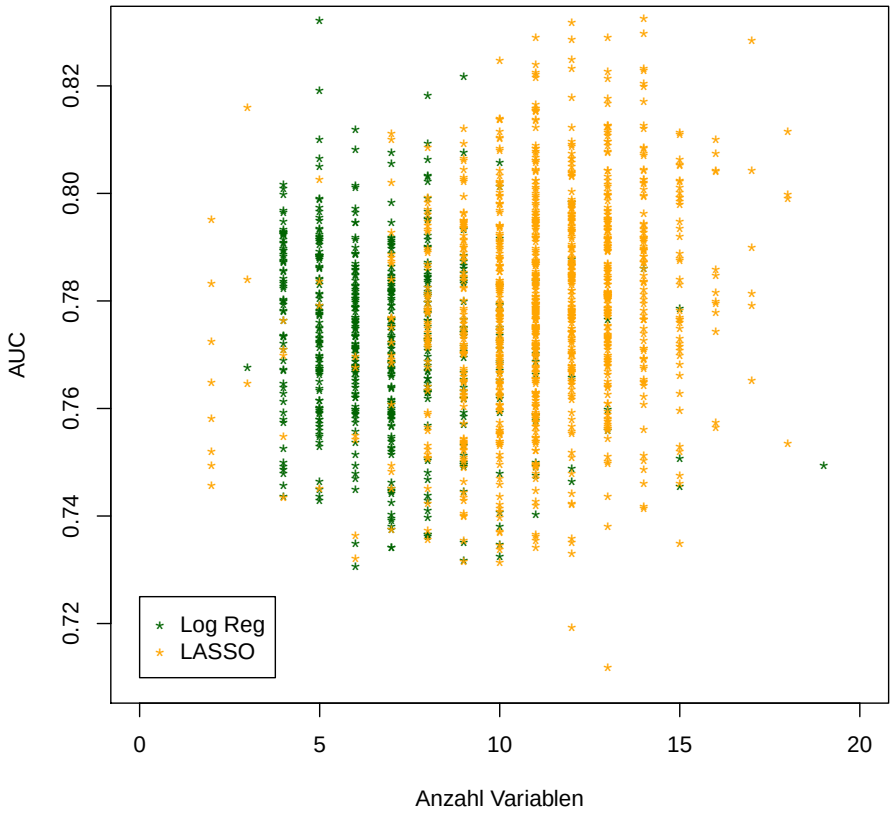

Abbildung 4.5: *Variablenmenge LogReg und LASSO*

Ein leichter negativer Zusammenhang zwischen der Anzahl der Variablen und der AUC ist hier erkennbar (Abb.4.5).
Die AUC-Werte erstrecken sich von 0.731 bis 0.832 und erreichen einen Mittelwert von 0.773.

Die integrierte Variablenselektion des LASSO-Verfahrens führt zu 2 bis 18 Variablen bei einem Mittelwert von 11.2. Hier zeigt sich ein leichter positiver Zusammenhang zwischen Variablenanzahl und AUC.
Die Spanne der AUC-Werte reicht von 0.712 bis 0.833 bei einem Mittelwert von 0.780.

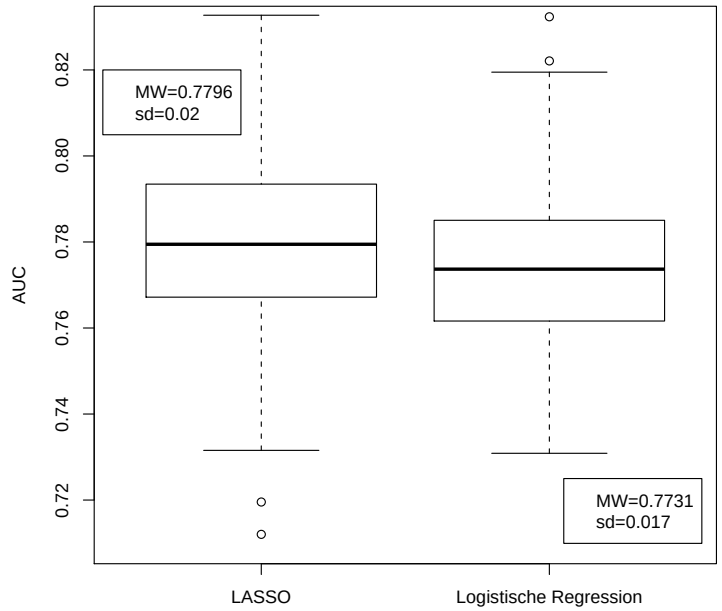

Abbildung 4.6: *Verteilung der AUC-Werte bei LogReg und LASSO*

LASSO weist auf Grund zweier Ausreißer (Abb. 4.6) eine größere Bandbreite an AUC-Werten auf und wählt im Durchschnitt mehr Variablen aus. Im Mittel liegen die AUC-Werte des LASSO über denen der Logistischen Regression, was sowohl in Abb. 4.5 als auch in Abb. 4.6 veranschaulicht wird.

4.4 Vergleich der Methoden

4.4.1 LDA und SVM

Vergleicht man nun die Ergebnisse der Linearen Diskriminanzanalyse und der Support Vector Machine und somit auch die jeweilige Effizienz der Selektionsverfahren, so erkennt man Gemeinsamkeiten bei den Korrelationsansätzen und auch Unterschiede, insbesondere im Rankingansatz.

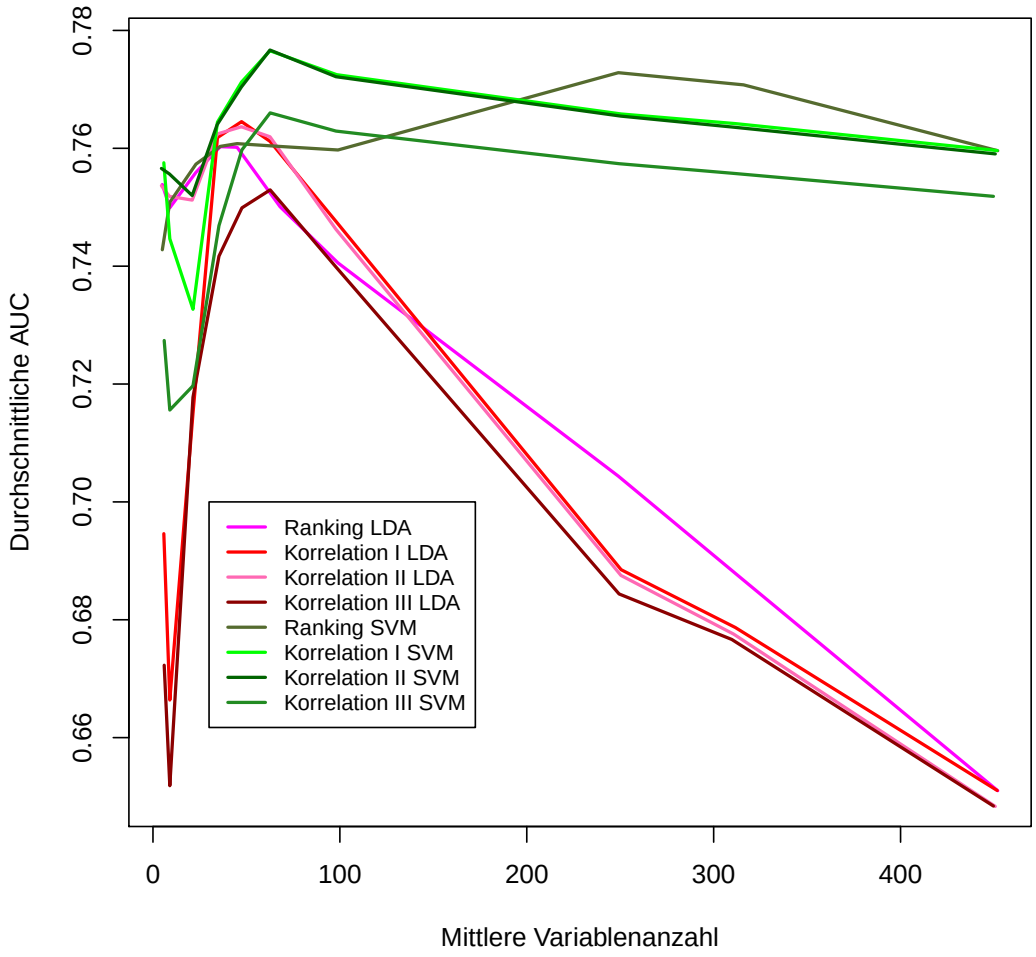

Abbildung 4.7: *Selektionsmethoden anhand von LDA und SVM*

Alle Korrelationsansätze, Korrelation I, II und III, weisen sowohl bei der LDA als auch bei der SVM im zweiten Schritt (was einer Korrelation von 0.05 entspricht) eine geringere AUC auf, als bei einer Korrelation von 0.02. Während die AUC-Werte der LDA bei den Korrelationsansätzen unmittelbar danach wieder ansteigen (Korrelation I und III) bzw. gleichbleiben (Korrelation II), sinken die AUCs der SVM im zweiten Schritt weiter. Korrelation III erreicht hier bereits sein Minimum, Korrelation I und III haben ihre niedrigste AUC im darauffolgenden Schritt, bei einer Korrelation von 0.20.
Gemeinsam ist beiden Klassifiaktionsmethoden, dass sowohl bei Anwendung der LDA als auch der SVM, die maximalen AUC-Werte jeweils bei geringen Variablenanzahlen erreicht werden. Bei der LDA sind dies durchschnittlich 47.4 Variablen, bei der SVM im Durchschnitt 62.6. Somit benötigt die LDA weniger Variablen um ihr Maximum zu erreichen als die SVM. Diese erreicht aber insgesamt eine höhere AUC. Allerdings folgt den maximalen AUC-Werten der LDA ein umso rapiderer Abfall, desto größer die verwendete Variablenmenge wird. Im Gegensatz zur SVM werden die Minima hier erst bei der kompletten Variablenanzahl erreicht. Die AUC-Werte der SVM haben bei größer werdender Variablenmenge lediglich eine geringe Verminderung.
Betrachtet man in beiden Methoden den Rankingansatz, so liegt hier die einzige Gemeinsamkeit darin, dass in beiden Fällen jeweils die AUC-Werte ab einem Anteil von 55% höher liegen als die der Korrelationsansätze. Vorher werden sie insbesondere von Korrelation I und II dominiert.
Während in Bezug zur SVM die AUC-Werte des Rankings kontinuierlich bis zum Erreichen des Maximums ansteigen und sich danach leicht verringern, zeigen die AUC-Werte der LDA einen ähnlichen Verlauf wie die Korrelationsansätze. Bei einem Variablenanteil von 2% liegen die AUC-Werte unter denen von 1%, steigen danach bis zum Erreichen des Maximums an und fallen schließlich bis zu einem Rankinganteil von 100% rapide ab. Auch wird die maximale AUC im Fall der LDA schon bei einem Anteil von 8% der Rankingvariablen erreicht, während dies im Fall der SVM erst bei einem Anteil von 55% eintritt.

4.4.2 LDA, SVM, LogReg und LASSO

Vergleicht man schließlich die AUC-Werte aller Klassifikationsverfahren miteinander, so erreicht LASSO mit einer mittleren AUC von 0.780 das absolute Maximum. Darunter folgen mit einer AUC von 0.777 die Korrelationsansätze I und II der SVM bei einer Korrelationsstärke von 0.60.

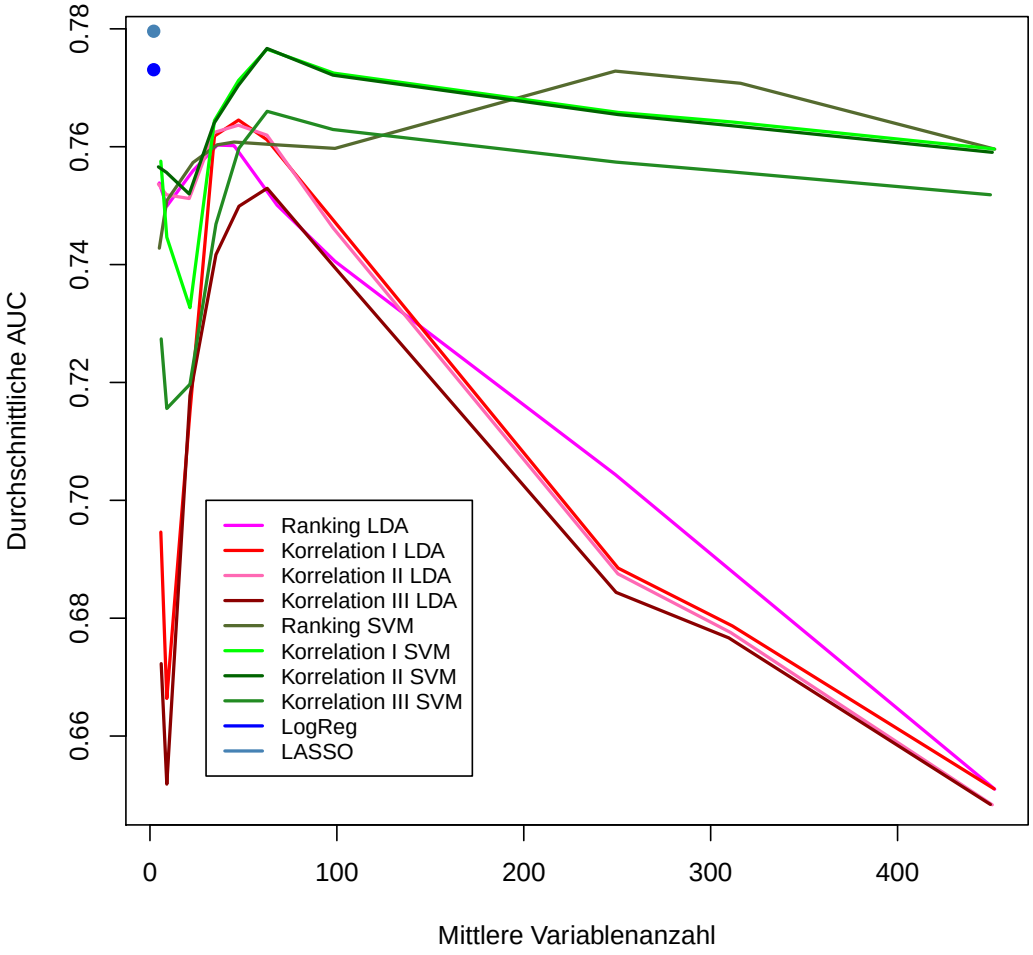

Abbildung 4.8: *AUCs im Vergleich*

Die AUC der Logistischen Regression mit 0.773 und die maximale AUC der LDA mit 0.765 (bei Korrelation I und einer Korrelationsstärke von 0.45)

bilden die Maxima der anderen Klassifikationsmethoden.

Die niedrigsten AUC-Werte innerhalb der Korrelationsansätze erreichen sowohl bei der LDA als auch der SVM jeweils die Korrelation III-Ansätze.
Bei Betrachtung der jeweils ausgewählten durchschnittlichen Variabelnanzahl ergeben sich teils große Unterschiede zwischen den einzelnen Klassifikationsmethoden. LASSO benötigt durchschnittlich nur 11.2 Variablen, um sein Maximum zu erreichen, während es bei der SVM trotz geringerem AUC-Wert im Mittel 62.6 sind. Die durchschnittlich resultierende Variablenanzahl der Vorwärtsselektion liegt bei 6.8, während die maximale AUC der LDA auf durchschnittlich 47.4 Variablen beruht.

Klassifikations-methode	Mittlere AUC	Selektions-methode	Korrelations-stärke	Durchschnittl. Variablenanzahl
LDA	0.765	Korrelation I	0.45	47.4
LogReg	0.773	Vorwärts	-	6.8
SVM	0.777	Korrelation II	0.60	62.6
LASSO	0.780	integriert	-	11.2

Tabelle 4.3: *Maximale AUC-Werte der jeweiligen Klassifikationsmethode*

Kapitel 5

Diskussion

Zusammenfassung und Diskussion

Ziel dieser Arbeit war es, anhand der Klassifiaktionsgenauigkeit von Linearer Diskriminanzanalyse und Support Vector Machine die Variablenselektionsverfahren Ranking, Korrelation I, II und III miteinander zu vergleichen und zu bewerten. Die Logistische Regression mit vorheriger klassischer Vorwärtsselektion und das LASSO-Verfahren als multivariater Ansatz mit integrierter Variablenselektion dienen hier als Vergleich zu den jeweiligen Messergebnissen.
Die Variablenselektion, einmal als Reduzierung großer Variablenmengen und zum anderen als Auswahl der aussagekräftigsten Variablen ist auf Grund von immer komplexer werdenden Datenmengen zu einem Hauptbestandteil der medizinischen Statistik herangewachsen. Besonders das Auftreten von Microarray Daten hat die Forschung auf dem Gebiet der Klassifiaktion und Variablenreduzierung in den letzten Jahren vorangebracht. Einige Ideen und Verfahren wurden in dieser Arbeit aufgegriffen und angewandt.
Bei den hier verwendeten Mammographie-Screeningdaten handelt es sich nicht um sogenannte Genexpressionsdaten, die tausende bis zehntausende Gene beinhalten. Auch bei den gegebenen 452 Variablen kommt es, wie hier gezeigt wurde, bei der Logistischen Regression oder der Linearen Diskriminanzanalyse nicht mehr zu aussagekräftigen Ergebnissen, sodass eine

Vorauswahl notwendig wird.
Diese Vorauswahl wird für die Lineare Diskriminanzanalyse und Support Vector Machine mit Hilfe der Variablenselektionsmethoden Ranking, Korrelation I, II und III getroffen. Das Hauptinteresse dieser Arbeit liegt in dem Abschneiden der Korrelationsansätze und der Frage, ob dieser als "semimultivariat" bezeichnete Ansatz ([20]) durch die zusätzliche Betrachtung der Korrelation der Variablen untereinander die üblichen univariaten Ansätze ablösen kann bzw. bessere Ergebnisse liefert. Es stellt sich auch die Frage, ob diese univariaten Ansätze als eine leicht durchführbare Alternative zu den sehr komplexen multivariaten Variablenselektionsmethoden gesehen werden können. Deren großer Vorteil liegt in der Beachtung von Korrelationen und Interaktionen. Auf Grund ihrer Komplexität werden diese Ansätze selten angewendet.
Der Selektionsansatz mit der geringsten AUC ist Korrelation III der LDA und dieser ist auch gleichzeitig der mit der allgemein schlechtesten Klassifikationsgenauigkeit. Auch in Bezug zur SVM resultieren nach Korrelation III die niedrigsten AUC-Werte, was sich dadurch erklären lässt, dass die beiden AUC-stärksten Variablen, also die mit dem größten univariaten Erklärungswert, hier fehlen. Besonders für Variablenanzahlen kleiner als 36 erhält man mit Korrelation III besonders niedrige AUC-Werte. Auffallend ist jedoch, dass sowohl bei der LDA als auch der SVM der Korrelation II-Ansatz für Variablenanzahlen kleiner 36 besser abschneidet als Korrelation I, obwohl in Korrelation II die AUC-stärkste Variable nicht mit eingeht. Da die Methoden lediglich an einem Datensatz evaluiert wurden, können keine Rückschlüsse auf ein generelles Verhalten dieser Korrelationsansätze gezogen werden. Möglicherweise tragen einzelne Variablen zu diesen Ausprägungen bei und es handelt sich um ein rein datenspezifisches Verhalten. Eine Betrachtung der ausgewählten Variablen könnte diese Frage klären.
Eine weitere Auffälligkeit zeigt sich darin, dass der Ranking-Ansatz, sowohl bei der LDA als auch der SVM bei einem Anteil von 55% die Korrelationsansätze übertrifft, also höhere AUC-Werte resultieren. Zwei Erklärungen scheinen hier plausibel:
Zum einen weisen die ersten Variablen des AUC-Rankings eventuell eine

hohe Korrelation untereinander auf, was dazu führen kann, dass sie redundante Informationen beinhalten und so zu einer schlechten Prädiktion führen.
Zum anderen entspricht die Variablenmenge bei einem Rankinganteil von 55% bereits einer Korrelationsstärke von 0.95 in den Korrelationsansätzen. Die durchschnittlich 250 ausgewählten Variablen besitzen also untereinander bereits eine Korrelation von bis zu 0.95. Unbestritten besteht ein starker Zusammenhang zwischen den ausgewählten Variablen. Die Aussage, dass mit den Korrelationsansätzen nur Variablen selektiert werden, die nicht miteinander korrelieren, trifft hier nicht mehr zu.
Im Vergleich mit der Klassifikationsgenauigkeit der Logistischen Regression und des LASSO Schätzers findet man in LASSO das Verfahren mit der höchsten AUC, gefolgt von der SVM im Korrelation I- und II-Ansatz, jeweils bei einer Korrelationsstärke von 0.60. Es folgen die Logistische Regression und schliesslich die LDA.
Allein unter den Korrelationsansätzen gibt es keinen global besten Ansatz. Korrelation II schneidet zunächst für geringere Variablenanzahlen besser ab als Korrelation I. Dies ändert sich aber einer Korrelationsstärke von 0.45 Die AUC-stärkste Variable auszulassen, wie es in Korrelation II der Fall ist, hat dem Ansatz mit eben dieser Variable gegenüber keinen Vorteil.

Fazit

Im direkten Vergleich der Methoden finden die Korrelationsansätze bei einer Variablenanzahl von bis zu 200 günstige Variablenkombinationen, welche sich in höheren AUC-Werten spiegeln. Dies gilt sowohl für die LDA als auch die SVM. Ab einer Variablenanzahl größer als 200 dominiert jeweils das Ranking.
Ich persönlich bevorzuge bei der Wahl der Selektions- und Klassfikationsmethode LASSO, welches auf Grund seiner integrierten Variablenselektion und dem dadurch multivariaten Ansatz in kurzer Zeit eine Variablenmenge selektiert, die auch mit ihrer geringen Größe eine sehr gute Prädiktion erzielt.

Implementierung

Wichtig im Zusammenhang mit der höchsten AUC ist die Betrachtung des Aufwands der Implementierung, der Rechenzeit und des resultierenden Nutzens. Ein Vielfaches an Rechenzeit steht in keinem Verhältnis zu einer minimal höheren AUC. Auch die Handhabung der verwendeten Funktionen spielt eine wichtige Rolle.
Auf Grund der vorherigen Variablenselektion und der daraus resultierenden 10 Variablenkombinationen benötigen die LDA und SVM die meiste Rechenzeit: Etwa 3.5 Tage für die SVM bzw. 2.5 für die LDA. Dagegen erwies sich die Implementierung für beide Methoden als gleich aufwändig.
Im Vergleich hierzu beträgt die Rechenzeit der LogReg nur 1.5 Tage. Aus der Vorwärtsselektion resultiert jedoch jeweils nur eine Variablenkombination, mit der die Logistische Regression durchgeführt wird.
LASSO ist in seiner Implementierung das aufwändigste Verfahren, zumal zusätzlich eine Kreuzvalidierung zur optimalen λ-Bestimmung durchgeführt werden muss und erst anschließend der LASSO-Schätzer bestimmt werden kann. LASSO ist aber auch das mit Abstand schnellste Verfahren mit einer Rechenzeit von etwa 4 Stunden. Im Hinblick auf die daraus folgende Klassifikationsgenauigkeit und der schnellen Durchführung hält sich der Aufwand der Implementierung in Grenzen.

Ausblick

Um weitere Aufschlüsse über die Güte von univariaten und multivariaten Variablenselektionsmethoden zu erhalten, ließen sich die hier vorgestellten Verfahren ausweiten.
Je nach Datenbeschaffenheit bietet sich an Stelle der Monte Carlo Cross Validation z.B. für Datensätze mit extrem kleiner Beobachtungszahl Bootstrap an[4].
Um eine noch bessere Robustheit der Schätzer zu garantieren, ist eine Aufstockung der 500 Iterationen auf eine Anzahl von 1000 möglich. In dieser Analyse waren ursrpünglich 1000 Iterationen vorgesehen, diese wurden aber

wegen zu langer Rechenzeit auf die verwendeten 500 herabgesetzt.
Auch auf das Parametertuning für die Support Vector Machine wurde wegen einer erwarteten Rechenzeit von mehreren Wochen verzichtet. Durch ein Tuning des Parameters λ und des Kostenparameters C können weitere Verbesserungen in der Prädiktionsgenauigkeit erzielt werden[2]. Ebenso verhält es sich beim Vergleich der einzelnen Kernfunktionen.
Die Verwendung weiterer univariater Selektionsmethoden bietet ein breiteres Spektrum an Vergleichsmöglichkeiten. Auch führt ein feineres Gitter der Korrelationsstärken zu einem detaillierteren Verständnis der Auswirkung der Korrelation auf die Variablenanzahl und die anschließende Prädiktion.
Für geringe Variablenanzahlen kann es auch von Interesse sein, zu sehen, welche Variablen jeweils ausgewählt werden und wie sich die jeweilige Variablenkombination auf die Klassifikationsgenauigkeit auswirkt.
Um ein präzises Ergebnis präsentieren zu können, sind mehrere Datensätze zwingend erforderlich. Nur wenn sich das Verhalten der angewandten Methoden auch bei der Durchführung an anderen Datensätzen wiederholt, kann eine allgemeingültige Aussage erfolgen.

Anhang A

Exakte Messergebnisse

Selektions-methode	An-teil	**LDA**	**SVM**	Variablen-anzahl (ϕ)
Ranking	1%	0.7538540	0.7427812	5
	2%	0.7499174	0.7508410	9
	5%	0.7559156	0.7573101	23
	8%	0.7602774	0.7603246	36
	10%	0.7601812	0.7607917	45
	15%	0.7500876	0.7603130	68
	22%	0.7405552	0.7597164	99
	55%	0.7043656	0.7728306	249
	70%	0.6867101	0.7707627	316
	100%	0.6509923	0.7595886	452
Korrelation	0.02	0.6946100	0.7575468	5.8
mit der	0.05	0.6663810	0.7446452	9.0
stärksten	0.20	0.7150180	0.7326827	21.4
Variablen	0.35	0.7618143	0.7644380	34.5
des	0.45	0.7645254	0.7712430	47.4
Rankings	0.60	0.7611848	0.7765495	62.6
	0.75	0.7474516	0.7724933	98.3
	0.95	0.6884925	0.7658260	250.5
	0.98	0.6786873	0.7641890	311.6
	1	0.6510343	0.7596187	451.7
Korrelation	0.02	0.7536577	0.7565851	4.5
mit der	0.05	0.7518452	0.7556453	8.8
zweitstärksten	0.20	0.7512139	0.7519868	21.0
Variablen	0.35	0.7624200	0.7639910	34.3
des	0.45	0.7636546	0.7704033	47.3
Rankings	0.60	0.7619605	0.7766769	62.6
	0.75	0.7462553	0.7721498	97.9
	0.95	0.6875220	0.7654593	250.4
	0.98	0.6775654	0.7635834	310.7
	1	0.6483317	0.7590487	450.7
Korrelation	0.02	0.6722893	0.7273905	6.0
mit der	0.05	0.6518427	0.7155845	9.0
drittstärksten	0.20	0.7177546	0.7196986	21.4
Variablen	0.35	0.7416954	0.7468673	35.3
des	0.45	0.7498963	0.7597387	47.6
Rankings	0.60	0.7529452	0.7660082	62.7
	0.75	0.7398636	0.7629229	97.9
	0.95	0.6843780	0.7573881	249.4
	0.98	0.6766561	0.7557706	309.7
	1	0.6483897	0.7518457	449.7

Tabelle A.1: *AUC Mittelwerte der LDA und SVM(ungerundet)*

Selektions-methode	An-teil	**LDA**	**SVM**	Variablen-anzahl (ϕ)
Ranking	1%	0.3032472	0.3055991	5
	2%	0.3072428	0.3068241	9
	5%	0.3061782	0.3111492	23
	8%	0.3044365	0.3108953	36
	10%	0.3048330	0.3123875	45
	15%	0.3088998	0.3180134	68
	22%	0.3059020	0.3219287	99
	55%	0.3208062	0.2972383	249
	70%	0.3337506	0.2981871	316
	100%	0.3682584	0.3058040	452
Korrelation	0.02	0.3898753	0.3371314	5.8
mit der	0.05	0.3752517	0.3366904	9.0
stärksten	0.20	0.3438797	0.3267973	21.4
Variablen	0.35	0.3022895	0.2968151	34.5
des	0.45	0.2994298	0.2898530	47.4
Rankings	0.60	0.2994610	0.2866414	62.6
	0.75	0.3077149	0.2906013	98.3
	0.95	0.3461559	0.2994833	250.5
	0.98	0.3491403	0.3009310	311.6
	1	0.3680935	0.3058263	451.7
Korrelation	0.02	0.3145212	0.3029889	4.5
mit der	0.05	0.3155367	0.3044410	8.8
zweitstärksten	0.20	0.3131136	0.3076481	21.0
Variablen	0.35	0.3017595	0.2978486	34.3
des	0.45	0.3004944	0.2906013	47.3
Rankings	0.60	0.2976615	0.2854566	62.6
	0.75	0.3090646	0.2912249	97.9
	0.95	0.3467884	0.2996971	250.4
	0.98	0.3499020	0.3013185	310.7
	1	0.3698218	0.3063252	450.7
Korrelation	0.02	0.4025657	0.3700356	6.0
mit der	0.05	0.3847439	0.3580980	9.0
drittstärksten	0.20	0.3391180	0.3374699	21.4
Variablen	0.35	0.3218797	0.3147394	35.3
des	0.45	0.3141247	0.3014967	47.6
Rankings	0.60	0.3088374	0.2951136	62.7
	0.75	0.3152160	0.2985791	97.9
	0.95	0.3503296	0.3059911	249.4
	0.98	0.3506370	0.3074477	309.7
	1	0.3695011	0.3113363	449.7

Tabelle A.2: *MSE Mittelwerte der LDA und SVM(ungerundet)*

	LogReg	LASSO
AUC	0.7730696 [0.731; 0.832]	0.7796021 [0.712; 0.833]
MSE	0.2895457 [0.238; 0.341]	0.2889555 [0.229; 0.358]
Variablenanzahl	6.77 [3; 19]	11.22 [2; 18]

Tabelle A.3: *AUC/MSE Mittelwerte der LogReg und LASSO(ungerundet), sowie die durchschnittliche Variablenanzahl mit Angabe des jeweiligen Minimum und Maximum*

Anhang B

Verteilung der AUC-/MSE-Werte von LDA und SVM

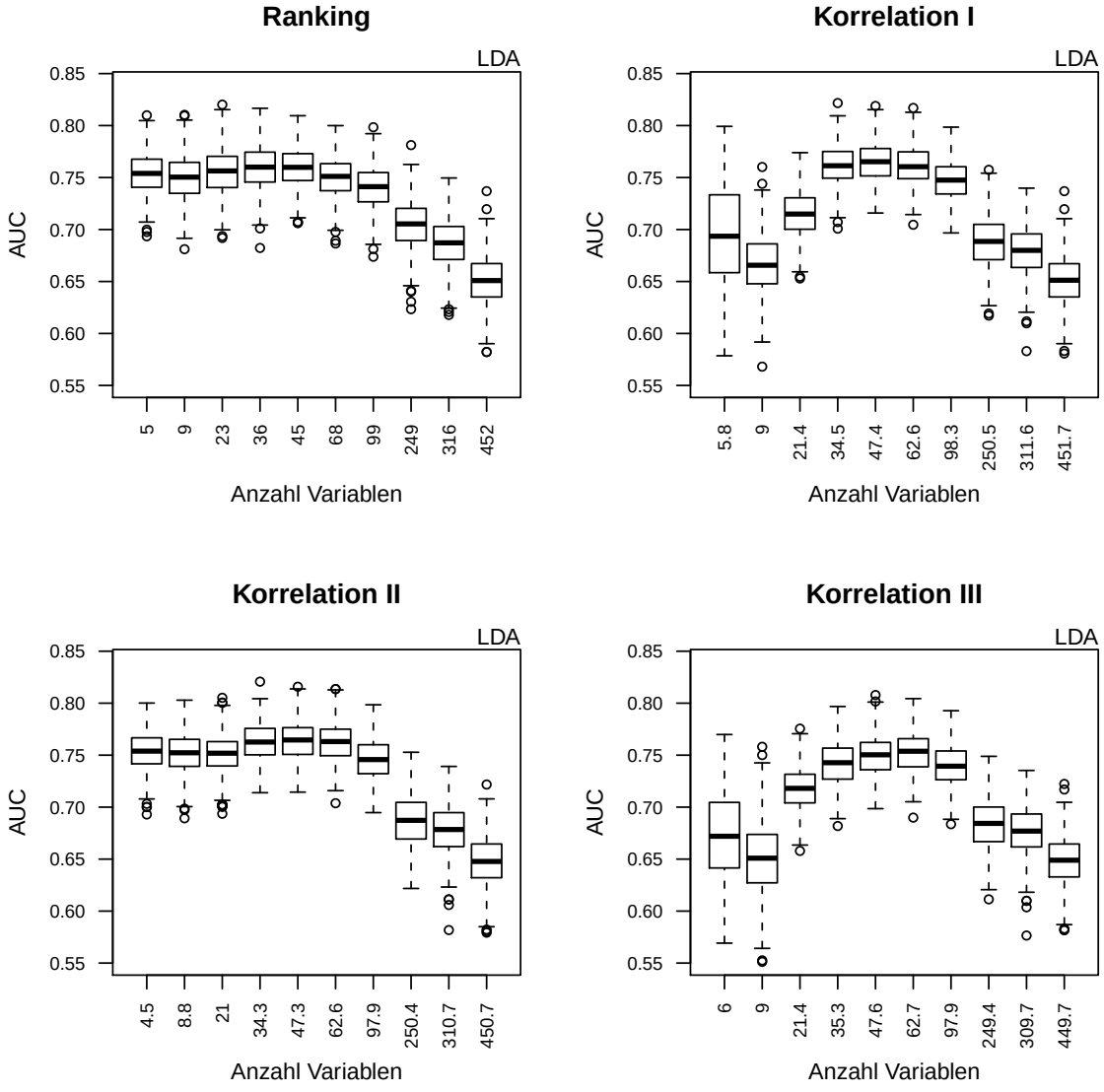

Abbildung B.1: *AUC Verteilung bei LDA*

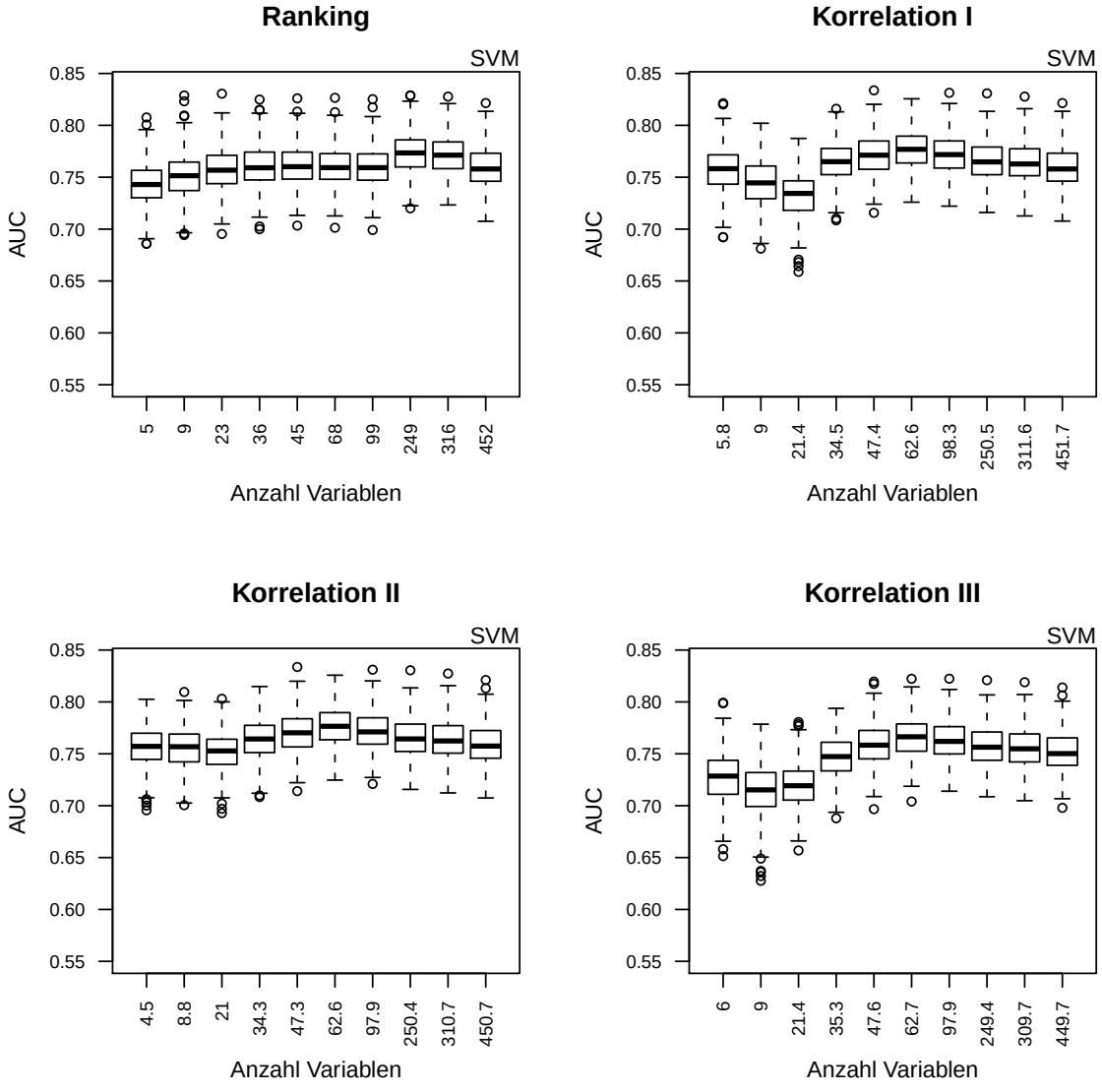

Abbildung B.2: *AUC Verteilung bei SVM*

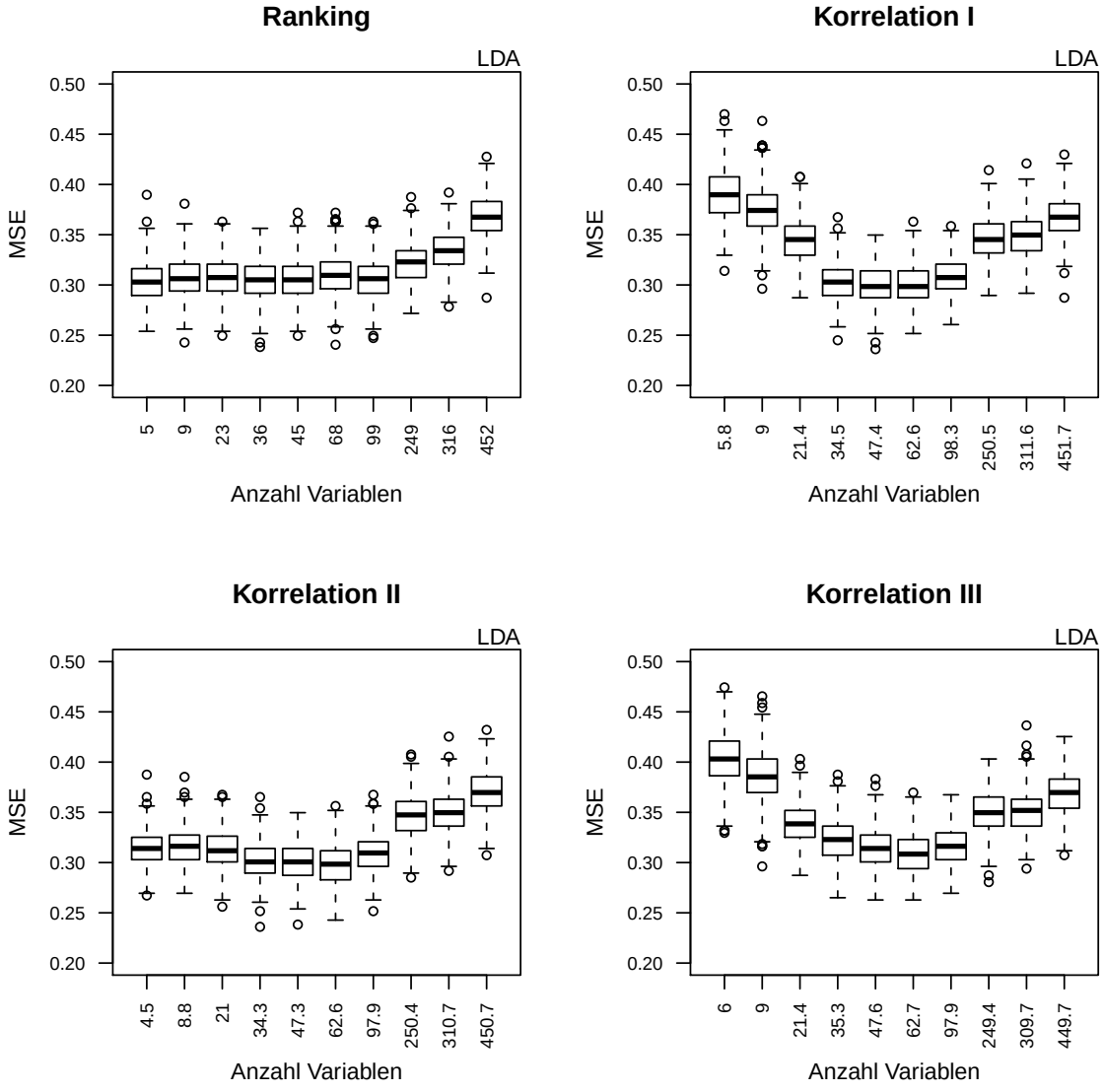

Abbildung B.3: *MSE Verteilung bei LDA*

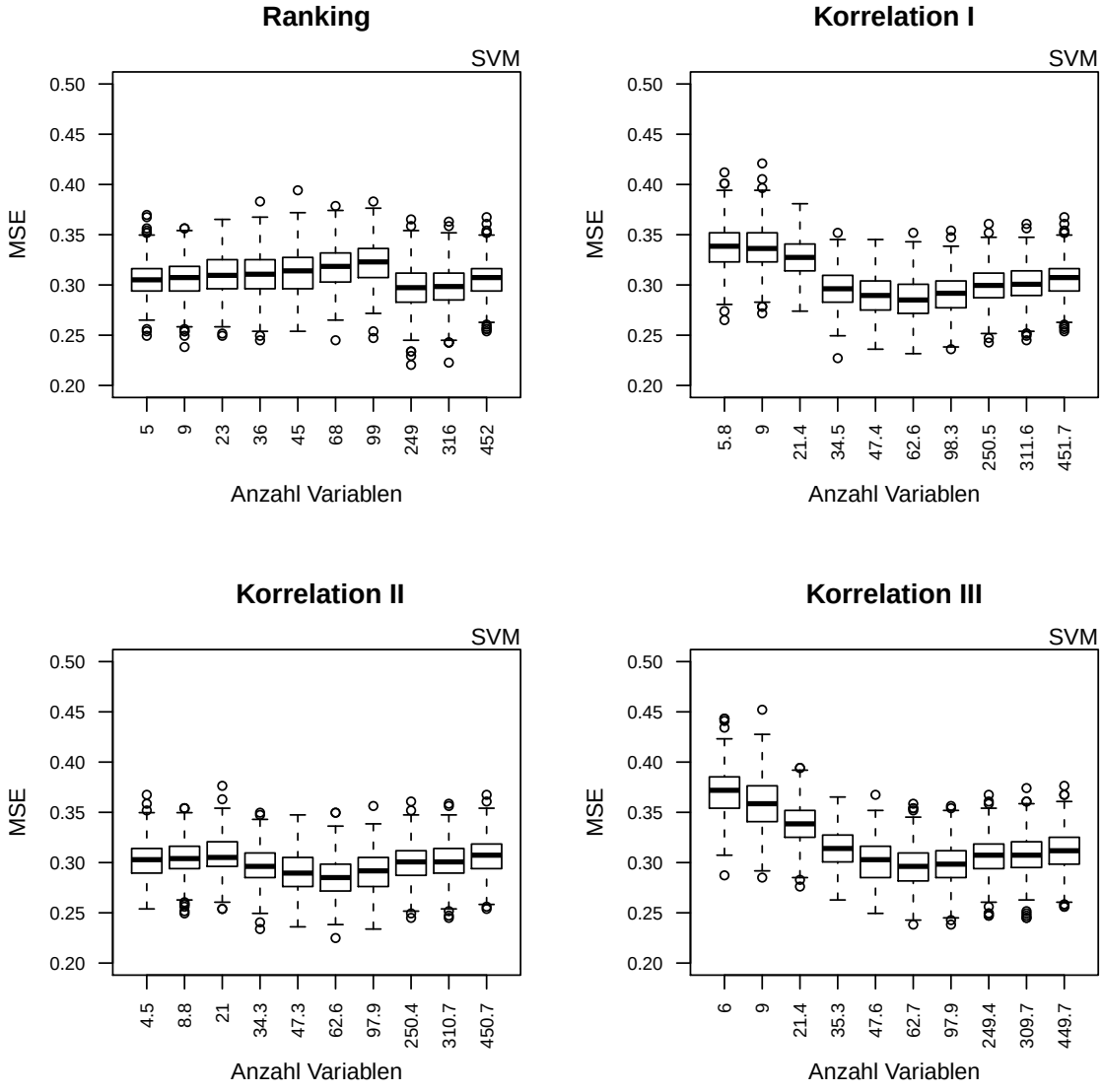

Abbildung B.4: *MSE Verteilung bei SVM*

Anhang C

Verwendeter R-Code

Variablenselektion: Ranking, Korrelation I, Korrelation II, Korrelation III

```
#   Packages laden
library(WilcoxCV)
library(ROC)
library(MASS)

dat.gesamt <- read.table("File")
N    <- 500          # Anzahl der Iterationen
verh <- 1347/3       # Für ein Verhältnis 2:1

set.seed(1334)
mccv.ind <- generate.split(niter=N,n=dim(dat.gesamt)[1],ntest=verh)
                     # MCCV
for(i in 1:N)  {
train.allg     <- dat.gesamt[-mccv.ind[i,],]
train.allg.x   <- train.allg[,-1]
test.allg      <- dat.gesamt[mccv.ind[i,],]
                    # [i,] Die i-te der 500 Iterationen wird verwendet
test.allg.x    <- test.allg[,-1]
test.allg.y    <- test.allg[,1]

#  Variablenranking  #
######################

Namen <- names(dat.gesamt)
                     # Trainingsdatensatz nach AUC-Werten ordnen
auc <- data.frame()
 for (k in 1:(dim(train.allg)[2]-1))  {
  auc[k,1] <-  k
  auc[k,2] <-  AUC(rocdemo.sca(train.allg$Severity,
                   train.allg[,k+1], rule = dxrule.sca))
```

```
  auc[k,3] <-  Namen[k+1]
                                        }
names(auc) <- c("Beob","aucwert","Name")
ordnen <- auc[order(auc$aucwert,decreasing=TRUE),]
index.auc.rank <- ordnen$Beob
aucorder <- train.allg.x[,index.auc.rank]
aucorder.test <- test.allg.x[,index.auc.rank]

anteil.var <- vector()
p <- c(0.01,0.02,0.05,0.08,0.1,0.15,0.22,0.55,0.75,1)
anteil.var <- round(p*(length(dat.gesamt)-1))

sel.anteil.01   <- names(aucorder[,1:anteil.var[1]])
sel.anteil.02   <- names(aucorder[,1:anteil.var[2]])
sel.anteil.05   <- names(aucorder[,1:anteil.var[3]])
sel.anteil.08   <- names(aucorder[,1:anteil.var[4]])
sel.anteil.10   <- names(aucorder[,1:anteil.var[5]])
sel.anteil.15   <- names(aucorder[,1:anteil.var[6]])
sel.anteil.22   <- names(aucorder[,1:anteil.var[7]])
sel.anteil.55   <- names(aucorder[,1:anteil.var[8]])
sel.anteil.70   <- names(aucorder[,1:anteil.var[9]])
sel.anteil.100  <- names(aucorder[,1:anteil.var[10]])

### Korrelation
###############
            # Diese Funktion vergleicht die Korrelation der
            # Variablen untereinander und gibt schliesslich die
            # aus, die unter der vorgegebenen Schranke liegen.
omit.cor.var <- function(cor.mat, threshold)
                {
                indexmenge <- vector()

                for (i in 2:dim(cor.mat)[1])
```

```
                    {
                    values <- cor.mat[!is.element(row.names(cor.mat),
                              indexmenge), i]
                              [1 : (i - length(indexmenge) - 1)]
                    values <- abs(values)
                    if (is.element(TRUE, values > threshold))
                    indexmenge <- c(indexmenge, row.names(cor.mat)[i])
                    }

                  selected <- row.names(cor.mat)[!is.element
                              (row.names(cor.mat), indexmenge)]
                  return(selected)
                  }

cor.matrix1 <- cor(aucorder)
sel.cor1.02 <- omit.cor.var(cor.matrix1,0.02)
sel.cor1.05 <- omit.cor.var(cor.matrix1,0.05)
sel.cor1.20 <- omit.cor.var(cor.matrix1,0.20)
sel.cor1.35 <- omit.cor.var(cor.matrix1,0.35)
sel.cor1.45 <- omit.cor.var(cor.matrix1,0.45)
sel.cor1.60 <- omit.cor.var(cor.matrix1,0.60)
sel.cor1.75 <- omit.cor.var(cor.matrix1,0.75)
sel.cor1.95 <- omit.cor.var(cor.matrix1,0.95)
sel.cor1.98 <- omit.cor.var(cor.matrix1,0.98)
sel.cor1.1 <- omit.cor.var(cor.matrix1,1.00)

cor.matrix2 <- cor(aucorder[-1])
sel.cor2.02 <- omit.cor.var(cor.matrix2,0.02)
sel.cor2.05 <- omit.cor.var(cor.matrix2,0.05)
sel.cor2.20 <- omit.cor.var(cor.matrix2,0.20)
sel.cor2.35 <- omit.cor.var(cor.matrix2,0.35)
sel.cor2.45 <- omit.cor.var(cor.matrix2,0.45)
sel.cor2.60 <- omit.cor.var(cor.matrix2,0.60)
```

```
sel.cor2.75 <- omit.cor.var(cor.matrix2,0.75)
sel.cor2.95 <- omit.cor.var(cor.matrix2,0.95)
sel.cor2.98 <- omit.cor.var(cor.matrix2,0.98)
sel.cor2.1 <- omit.cor.var(cor.matrix2,1.00)

cor.matrix3 <- cor(aucorder[-1][-1])
sel.cor3.02 <- omit.cor.var(cor.matrix3,0.02)
sel.cor3.05 <- omit.cor.var(cor.matrix3,0.05)
sel.cor3.20 <- omit.cor.var(cor.matrix3,0.20)
sel.cor3.35 <- omit.cor.var(cor.matrix3,0.35)
sel.cor3.45 <- omit.cor.var(cor.matrix3,0.45)
sel.cor3.60 <- omit.cor.var(cor.matrix3,0.60)
sel.cor3.75 <- omit.cor.var(cor.matrix3,0.75)
sel.cor3.95 <- omit.cor.var(cor.matrix3,0.95)
sel.cor3.98 <- omit.cor.var(cor.matrix3,0.98)
sel.cor3.1 <- omit.cor.var(cor.matrix3,1.00)

            # Abspeichern der gewählten Variablen in eine Liste
VarSel <- list(sel.anteil.01, sel.anteil.02, sel.anteil.05,
               sel.anteil.08, sel.anteil.10, sel.anteil.15,
               sel.anteil.22, sel.anteil.55, sel.anteil.75,
               sel.anteil.100, sel.cor1.02, sel.cor1.05,
               sel.cor1.20, sel.cor1.35, sel.cor1.45,
               sel.cor1.60, sel.cor1.75, sel.cor1.95,
               sel.cor1.98, sel.cor1.1, sel.cor2.02,
               sel.cor2.05, sel.cor2.20, sel.cor2.35,
               sel.cor2.45, sel.cor2.60, sel.cor2.75,
               sel.cor2.95, sel.cor2.98, sel.cor2.1,
               sel.cor3.02, sel.cor3.05, sel.cor3.20,
               sel.cor3.35, sel.cor3.45, sel.cor3.60,
               sel.cor3.75, sel.cor3.95, sel.cor3.98, sel.cor3.1)
```

Anwendung der LDA auf die ausgewählten Variablen

```
#   Packages
library(WilcoxCV)
library(ROC)
library(MASS)

dat.gesamt <- read.table("File")
N    <- 500
verh <- 1347/3

set.seed(1334)
mccv.ind <- generate.split(niter=N,n=dim(dat.gesamt)[1],ntest=verh)

klasse.lda.list <- vector(length=N, mode="list")
auc.lda.list <- vector(length=N, mode="list")
anzahl.lda.list <- vector(length=N, mode="list")
mse.lda.list <- vector(length=N, mode="list")

for(i in 1:N)  {
train.allg     <- dat.gesamt[-mccv.ind[i,],]
train.allg.x   <- train.allg[,-1]
test.allg      <- dat.gesamt[mccv.ind[i,],]

test.allg.x    <- test.allg[,-1]
test.allg.y    <- test.allg[,1]

auc.lda <- vector(length=length(VarSel), mode="logical")
klassen.lda.matrix <- matrix(data=NA, length(test.allg.y),
                             length(VarSel))
anzahl.lda <- vector()
mse.lda <- vector()
for(w in 1:length(VarSel)){
```

```
model.lda <- lda(Severity~., data=train.allg[,c("Severity",VarSel[[w]])],
                 tol=1.0e-40)

wkeit.lda <- predict(model.lda,newdata=test.allg[,VarSel[[w]]])$
                                                    posterior[,2]
klassen.lda <- predict(model.lda,newdata=test.allg[,VarSel[[w]]])$
                                                    class
auc.lda[w]  <- AUC(rocdemo.sca(test.allg.y, wkeit.lda,
                   rule = dxrule.sca))

klassen.lda.matrix[,w] <- klassen.lda

anzahl.lda[w] <- length(VarSel[[w]])
mse.lda[w] <- 1/length(test.allg.y) *
              sum((test.allg.y - (as.numeric(klassen.lda)-1))^2)
                }

auc.lda.list[[i]] <- auc.lda
anzahl.lda.list[[i]] <- anzahl.lda
klasse.lda.list[[i]] <- klassen.lda.matrix
mse.lda.list[[i]] <- mse.lda

                }
```

Anwendung der SVM auf die ausgewählten Variablen

```
#   Packages
library(WilcoxCV)
library(ROC)
library(e1071)

dat.gesamt <- read.table("File")
N    <- 500
verh <- 1347/3
set.seed(1334)
mccv.ind <- generate.split(niter=N,n=dim(dat.gesamt)[1],ntest=verh)

auc.svm.list <- vector(length=N, mode="list")
klasse.svm.list <- vector(length=N, mode="list")
anzahl.svm.list <- vector(length=N, mode="list")
mse.svm.list <- vector(length=N, mode="list")

for(i in 1:N)  {
train.allg     <- dat.gesamt[-mccv.ind[i,],]
train.allg.x   <- train.allg[,-1]
test.allg      <- dat.gesamt[mccv.ind[i,],]
test.allg.x    <- test.allg[,-1]
test.allg.y    <- test.allg[,1]

auc.svm <- vector(length=length(VarSel), mode="logical")
klassen.svm.matrix <- matrix(data=NA, length(test.allg.y),
                             length(VarSel))
anzahl.svm <- vector()
mse.svm <- vector()

for(w in 1:length(VarSel)){
model.svm <- svm(formula=as.factor(Severity)~., type="C" ,
```

```
                data=train.allg[,c("Severity",VarSel[[w]])],
                kernel="radial", probability=T)
wkeit.svm <- attr(predict(model.svm, newdata=test.allg[,VarSel[[w]]],
                 probability = T), "probabilities")[, "1"]
klassen.svm <- predict(model.svm, newdata=test.allg[,VarSel[[w]]])
klassen.svm.matrix[,w] <- klassen.svm
auc.svm[w]  <- AUC(rocdemo.sca(test.allg.y, wkeit.svm,
                   rule = dxrule.sca))
anzahl.svm[w] <- length(VarSel[[w]])
mse.svm[w] <- 1/length(test.allg.y)*
              sum((test.allg.y -(as.numeric(klassen.svm)-1))^2)
}

auc.svm.list[[i]] <- auc.svm
klasse.svm.list[[i]] <- klassen.svm.matrix
anzahl.svm.list[[i]] <- anzahl.svm
mse.svm.list[[i]] <- mse.svm

               }
```

Logistische Regression mit Vorwärtsselektion

```
#   Packages
library(WilcoxCV)
library(ROC)
library(MASS)

dat.gesamt <- read.table("File")

N    <- 500
verh <- 1347/3

set.seed(1334)
mccv.ind <- generate.split(niter=N,n=dim(dat.gesamt)[1],ntest=verh)

# Ausgabeobjekte:
klasse.logReg.list <- vector(length=N, mode="list")
auc.logReg <- vector()
mse.logReg <- vector()
anzahl.logReg <- vector()

for(i in 1:N)  {
train.allg     <- dat.gesamt[-mccv.ind[i,],]
train.allg.x   <- train.allg[,-1]
test.allg      <- dat.gesamt[mccv.ind[i,],]
test.allg.x    <- test.allg[,-1]
test.allg.y    <- test.allg[,1]

### Einfache Vorwärtsselektion
################################

logmodel.1 <- glm(Severity~1, data=train.allg,
                  family=binomial("logit"))
logmodel.alle <- glm(Severity~., data=train.allg,
```

```
                    family=binomial("logit"))
sel.logmodel <- stepAIC (logmodel.1, scope=list(upper=logmodel.alle,
                         lower=logmodel.1), direction="forward",
                         k=log(dim(dat.gesamt)[1]))

anzahl.logReg[i] <- length(sel.logmodel$coefficients)-1

### Logistische Regression mit Variablen aus Vorwärtsselektion
#################################################################
train.logReg <- sel.logmodel
wkeit.logReg <- predict(train.logReg, newdata=test.allg.x[,namen.logReg],
                        type="response")

klasse.logReg <- vector()
for(l in 1:length(wkeit.logReg))
klasse.logReg[l] <- ifelse(wkeit.logReg[l]>=0.5,1,0)

klasse.logReg.list[[i]] <- klasse.logReg
auc.logReg[i] <- AUC(rocdemo.sca(test.allg.y, wkeit.logReg,
                     rule = dxrule.sca))
mse.logReg[i] <- 1/length(test.allg.y) *
                 sum((test.allg.y - klasse.logReg)^2)

             }
```

LASSO mit Kreuzvalidierung zur optimalen λ Bestimmung

```
#   Packages laden
library(WilcoxCV)
library(ROC)
library(glmnet)

dat.gesamt <- read.table("File")

N    <- 500
verh <- 1347/3

set.seed(1334)
mccv.ind <- generate.split(niter=N,n=dim(dat.gesamt)[1],ntest=verh)

klasse.lasso.list <- vector(length=N, mode="list")
auc.lasso <- vector()
mse.lasso <- vector()
anzahl.lasso <- vector()
namen.lasso.list <- vector(length=N, mode="list")
namen.lasso <- vector()

for(i in 1:N)  {
train.allg     <- dat.gesamt[-mccv.ind[i,],]
train.allg.x   <- train.allg[,-1]
test.allg      <- dat.gesamt[mccv.ind[i,],]

test.allg.x    <- test.allg[,-1]
test.allg.y    <- test.allg[,1]

predictors.lasso <- as.matrix(train.allg[,-1])
response.lasso <- as.factor(train.allg[,1])
lambda.poss.lasso <- glmnet(x=predictors.lasso, y=response.lasso,
```

```
                            family="binomial", alpha=1,
                            nlambda = 20, lambda.min = 0.01)$lambda
lambda.poss.lasso <- lambda.poss.lasso[1:8]

        # Kreuzvalidierung zur Bestimmung des optimalen lambda
cv.ind.lasso <- generate.cv(n=dim(train.allg)[1], m=10)
n.cv <- 10

klasse.lasso <- vector(n.cv*lambda.poss.lasso, mode="list")
aucs.lasso.cv <- matrix(NA, nrow = n.cv,
                        ncol = length(lambda.poss.lasso),
                        dimnames = list(1:n.cv, lambda.poss.lasso))

model.lasso <- vector(n.cv*length(lambda.poss.lasso), mode="list")
for(j in 1:length(lambda.poss.lasso)) {
for(m in 1:n.cv)                {
           train.pre.lasso <- predictors.lasso[-cv.ind.lasso[m,],]
           train.res.lasso <- response.lasso[-cv.ind.lasso[m,]]
           test.alle <- train.allg[,2:dim(train.allg)[2]]
           test.alle.cv <- as.matrix(test.alle[cv.ind.lasso[m,],])
           responses.orig <- train.allg[,1]
           responses.vgl <- responses.orig[cv.ind.lasso[m,]]

model.lasso[[m+(j-1)*10]] <- predict(glmnet(x=train.pre.lasso,
                             y=train.res.lasso, family="binomial",
                             alpha=1, lambda=lambda.poss.lasso[j]),
                             test.alle.cv, s=lambda.poss.lasso[j],
                             type="response")

klasse.lasso[[m+(j-1)*10]] <- ifelse(model.lasso[[m+(j-1)*10]]>=0.5,1,0)

aucs.lasso.cv[m,j] <- AUC(rocdemo.sca(responses.vgl,
                          model.lasso[[m+(j-1)*10]],
```

```
                        rule = dxrule.sca))

                              }
                              }

               # LASSO mit oben gewähltem optimalem lambda
lambda.lasso.final <- list(aucs.lasso.cv, colMeans(aucs.lasso.cv),
             names(colMeans(aucs.lasso.cv))
             [which.max(colMeans(aucs.lasso.cv))])

model.lasso.final <- glmnet(x=predictors.lasso, y=response.lasso,
                     family="binomial", alpha=1,
                     lambda=lambda.lasso.final[[3]])
wkeit.lasso  <- predict(model.lasso.final, as.matrix(test.allg.x),
                        type="response")
klasse.lasso.final <- ifelse(wkeit.lasso>=0.5,1,0)
klasse.lasso.list[[i]] <- klasse.lasso.final

auc.lasso[i] <- AUC(rocdemo.sca(test.allg.y, wkeit.lasso,
                    rule = dxrule.sca))
mse.lasso[i] <- 1/length(test.allg.y) *
                sum((test.allg.y - klasse.lasso.final)^2)
anzahl.lasso[i] <- length(which(model.lasso.final$
                   beta[1:dim(test.allg.x)[2]] !=0))
namenvar.lasso <- which(model.lasso.final$beta
                  [1:dim(test.allg.x)[2]] !=0)
namen.lasso.list[[i]] <- names(test.allg.x[,namenvar.lasso])

                  }
```

Literaturverzeichnis

[1] Ambroise, C., McLachlan, G.J., 2002, Selection Bias in gene extraction in tumor classification on basis of microarray gene expressiond data, Proceedings of the National Academy of Science, 99:6562-6

[2] Boulesteix, A.-L., Strobl, C., Augustin, T. and Daumer, M., 2008, Evaluating Microarray-based Classifiers: An Overview, Cancer Informatics, 77-97

[3] Boulesteix, A.-L., 2007,WilcoxCV: an R package for fast variable selection in cross-validation, Bioinformatics,23: 1702-1704

[4] Braga-Neto, U.M., Dougherty, E.R., 2004, Is cross-validation valid for small-sample microarray classification?, Bioinformatics, 20:374-80

[5] Carter, C.L., Allen, C., Henson, D.E., 1989, Relation of Tumor Size, Lymph Node Status, and Survival in 24740 Breast Cancer Cases, Cancer, 63:181-87

[6] Diaz-Uriarte, R. and Alvarez de Andrés, S., 2006, Gene selection and classification of microarray data using random forests, BMC Bioinformatics, 7:3

[7] Dudoit, S., Fridlyand, J., Speed, T.P., 2002, Comparison of Discrimination Methods for the Classification of Tumors Using Gene Expression Data, Journal of the American Statistical Association, Vol. 97, No. 457, 77-87

[8] Efron, B., Hastie, T., Johnstone, I., Tibshirani, R., 2004, Least Angle Regression, The Annals of Statistics, 32:407-499

[9] Fahrmeir, L., Hamerle, A., Tutz, G., 1996, Multivariate statistische Verfahren, 2. Auflage, de Gruyter

[10] Fahrmeir, L., Kneib, T., Lang, S., 2007, Regression - Modelle, Methoden und Anwendungen, Springer

[11] Fisher, R.A., 1936, The use of multiple measurements in taxonomic problems, Annals of Eugenics, 7:179-188

[12] Friedman, J., Hastie, T., Tibshirani, R., 2009, Regularization Paths for Generalized Linear Models vie Coordinate Descent, http://www-stat.stanford.edu/h̃astie/Papers/glmnet.pdf

[13] Furey, T.S., Cristianini, N., Duffy, N., Bednarski, D.W., Schummer, M., Haussler, D., 2000, Support vector machine classification and validation of cancer tissue samples using microarray expression data, Bioinformatics, 16:906-14

[14] Gilbert, F.J., Astley, S.M., Gillan, M.G.C., Agbaje, O.F., Wallis, M.G., James, J., Boggis, C.R.M., Duffy, S.W., 2008, Single Reading with Computer-Aided Detection for Screening Mammography, The New England Journal of Medicine, 359:1675-84

[15] Ghosh, D. and Chinnaiyan, A.M., 2005, Classification and Selection of Biomarkers in Genomic Data Using LASSO, Journal of Biomedicine and Biotechnology, 2:147-154

[16] Hastie, T., Tibshirani, R., Friedman, J., 2001, The Elements of Statistical Learning, Springer

[17] Heath, M., Bowyer, K., Kopans, D., Moore, R., Kegelmeyer, P., 2001, The digital database for screening mammography, Proceedings of the Fifth International Workshop on Digital Mammography, Medical Physics Publishing, 212-18

[18] Hesterberg, T., Choi, N.H., Meier, L., Fraley, C., 2008, Least angle and l1 penalized regression: A review, Statistics Survey, 2:61-93

[19] Huang, X., Pan, W., Grindle, S., Han, X., Chen, Y., Park, S.J., Miller, L.W., Hall, J., 2005, A comparative study of discriminating human heart failure etiology using gene expression profiles, BMC Bioinformatics, 6:205-220

[20] Jaeger, J., Sengupta, R., Ruzzo, W.L., 2003, Improved gene selection for classification of microarray data, Proceedings of the 2003 Pacific Symposium on Biocomputing, 53-64.

[21] Jeffrey, I.B., Higgins, D.G. and Culhane, A.C., 2006, Comparison and Evaluation of Methods for Generating Differentially Expressed Gene Lists from Microarray Data, BMC Informatics, 7:359

[22] Lachlan, G.J., Chevelu, J., Zhu, J., 2008, Correcting for selection bias via cross-validation in the classification of microarray data, IMS Collections, 1:364-76

[23] Lee, J.W., Lee, J.B., Park, M., Song, S.H., 2005, An extensive Comparison of recent Classification Tools applied to Microarray Data, Computational Statistics and Data Analysis, 48:869-885

[24] Leisch, F., 2008, Vorlesungsskript zu Multivariate Verfahren

[25] Malich, A., Fischer, D.R., Böttcher, J., 2006, CAD for mammography: the technique, results, current role and further developments, Eur Radiol, 16:1449-60

[26] Meyer, D., 2001, Support Vector Machines, R-News, Vol. 1/3: 23-26

[27] Osborne, M.R., Presnell, B., Turlach, B.A., 2000, On the LASSO and its Dual, Journal of Computational and Graphical Statistics, 9:319-337

[28] Slawski, M., Daumer, M. and Boulesteix, A.-L., 2008, CMA - A comprehensive Bioconductor package for supervised classification with high dimensional data, BMC Bioinformatics, 9:439

[29] Statnikov, A., Aliferis, C.F., Tsamardinos, I., Hardin, D., Levy, S., 2005, A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis, Bioinformatics, 21:631-43, Appendix A, Appendix B

[30] Stauber, M., Weyerstahl, T., 2005, Gynäkologie und Geburtshilfe, Thieme

[31] Surendra, K. S., Huan, L., 2006, Feature Subset Selection Bias for Classification Learning, ACM International Conference Proceeding Series, Vol. 148

[32] Tibshirani, R., 1996, Regression Shrinkage and Selection vie the Lasso, Journal of the Royal Statistical Society, 58:267-288

[33] Vapnik, V.N., 1999, The Nature of Statistical Learning Theory, Springer

[34] WHO, 2008, World Health Statistics, WHO Press

[35] Zou, H. and Hastie, T., 2005, Regularization and variable selection via the elastic net, Journal of the Royal Statistical Society, 67:301-320

Hiermit versichere ich, dass ich diese Diplomarbeit
selbständig und ohne Benutzung anderer
als der angegebenen Hilfsmittel angefertigt habe.

München, den 03. Februar 2010

.....
Judith Schwitulla