Explorative Regressionsanalyse epidemologischer Daten auf Basis von R

Wissenschaftliche Projektarbeit

Otto-von-Guericke-Universität Magdeburg

Verfasser: Daniel Schneider
Studiengang: Statistik (Master)
Martrikelnr.: 214240

Betreuer: Prof. Dr. Bernhard Preim

10. März 2017

Inhaltsverzeichnis

1 Einleitung

Im Zusammenhang mit epidemologischen Studien wurde am Lehrstuhl für Visualisierung der Otto-von-Guericke Universität Magdeburg eine Arbeitsgruppe ins Leben gerufen. Diese Arbeitsgruppe unter Prof. Dr. Bernhard Preim beschäftigt sich damit, hochdimensionale Datensätze durch Clustering-, Regressions und Visualisierungsverfahren aufzuarbeiten. Die Methoden zielen nicht direkt darauf ab, Erklärungen für medizinische Befunde zu finden, können aber dabei helfen Hypothesen für weiterführende Analysen aufzubauen. Als wichtige Grundlage dienen dabei Kohortenstudien wie die *Study of Health in Pomerania* (SHIP). Diese Studie zur Identifizierung von Risikofaktoren und Häufigkeiten von Krankheiten besteht aus zwei Kohorten von anfänglich über 4300 Untersuchungspersonen. Von 1997 an wurden von den Versuchpersonen über mehrere Jahre hinweg widerholt medizinische, demographische und sozioökonomische Daten ermittelt [1].
"Testing features for associations with diseases using regression models is one of the most important epidemiological tools."[2] Davon abgeleitet war das Ziel dieser Arbeit, die explorative Ermittlung von Regressionsmodellen auf Basis des SHIP-Datensatzes, um die Risikofaktoren für die Tumorentstehung und mögliche Interaktionseffekte zwischen ihnen zu erklären. Auf Grund des Umfangs des Datensatzes ergeben sich daraus für die multiple Regression sehr viele Möglichkeiten, wie Modelle gebildet werden können. Die Anzahl möglicher Modelle ist auf Grund der hohen Dimensionalität des Datensatzes zu groß, als dass alle Modelle einzeln berechnet werden könnten. Deshalb wurde ein Tool entwickelt, dass alle möglichen Modelle automatisch erstellt und testet. Dieses Tool soll in diesem Text näher beschrieben werden. Weitere Bemühungen sind dahingehend gerichtet, dass die Rechendauer gering gehalten wird und besonders gute Modelle automatisch hervorgehoben werden. Die Berechnungen und Erstellung der im Weiteren beschriebenen Funktionen im Tool beruhen auf der Software R.
Da für die explizite Auswahl von Modellen, die für die Epidemologie sinnvoll erscheinen, tiefer gehendes Fachwissen aus der Medizin notwendig ist, kann diese Eingrenzung nicht automatisiert werden. Das Tool soll zunächst dahingehend helfen, Modelle hervorzuheben, die rechnerisch interessant sind (zB. durch Signifikanz) und diese dann nach selbstgewählten Kriterien zu ordnen. Für die medizinische Forschung sollen damit aus tausenden denkbaren Modellen diejenigen herausgefiltert werden, die Erklärungskraft besitzen. Im weiteren Rahmen dieser Froschungsarbeit soll das Tool so weiterentwickelt werden, dass es möglichst leicht zu handhaben ist und tiefergehende Analysen auch auf visueller Basis ermöglicht. (Siehe hierzu u.a. *P. Klemm, S. Oeltze-Jafra, K. Lawonn, K. Hegenscheid, H. Völzke, B. Preim; Interactive Visual Analysis of Image-Centric Cohort Study Data; IEEE Transactions on Visualization and Computer Graphics (TVCG); 2014; S. 1673-1682*)
Dieser Text soll als Überblick und Einführung einer weiterführenden Arbeit dienen. Zunächst werden die Regressionsmodelle beschrieben, die im später Anwendung finden. Ab Seite 5 wird

[1] Bundesgesundheitsblatt 2012 · 55:790–794 · DOI 10.1007/s00103-012-1483-6; Springer; 2012
[2] P. Klemm, K. Lawonn, S. Glaßer, U. Niemann, K. Hegenscheid, H. Völzke, B. Preim; 3D Regression Heat Map Analysis of Population Study Data; 2015; S. 81-90

behandelt, wie das Tool überprüft, ob die für die Regression notwendigen Anforderungen von den Modellen erüllt werden. In Kapitel 4 wird der zugrunde liegende Code erklärt und aufgearbeitet, sowie dessen Anwendung besprochen. Im Schluss wird ein kurzer Forschungsausblick vorgestellt, an den weiterführende Arbeiten anknüpfen könnten

2 Überblick über lineare Regressionsverfahren

Da die abhängige Variable Y (Tumorgröße) als metrische Variable vorliegt und auf Basis der vorliegenden Daten erklärt werden sollte, beruht die Funktion auf einer multiplen Regressionsanalyse.

Für das allgemeine lineare Regresionsmodell wird die Form

$$y = X\beta + \varepsilon$$

angenommen[3]. X spiegelt dabei die Datenmatrix der unabhängigen Variablen wieder, β den zu ermittelnden Vektor der den Einfluss der einzelnen unabhängigen Variablen auf die abhängige angibt und ε unerklärten Fehlerterm.
Der Schätzer für β, $\hat{\beta} = (\hat{\beta}_1, ..., \hat{\beta}_k)$ wird mittels Kleinste-Quadrate (KQ)-Methode ermittelt und unter den zuvor angegebenen Bedingungen gilt demzufolge:

$$\hat{\beta} = (X'X)^{-1}X'y$$

mit $X = (x_{j1}, ...x_{jn})'$, $y = (y_1, ..., y_n)$, j=(1,...,k), k=Anzahl Parameter, (i=(1,...,n), n=Stichprobengröße).
Anders ausgedrückt erfüllt $\hat{\beta}$ das KQ-Krierium und liefert somit diejenigen Steigungsvektoren, welche die Gerade/Ebene beschreiben, die den kleinsten quadrierten Abstand zu den Messwerten hat

$$KQ(\beta) = \Sigma_{i=1}^{n}(y_i - x_i'\beta)^2.$$

Um eine lineare Regression durchführen zu können, müssen einige Voraussetzungen[4] erfüllt sein:

- abhängige Variable: metrisch, unabhängig, normalverteilt mit N(E(y),Var(y))
- unabhängige Variablen: (quasi-)metrisch / binär
- Gauß-Markov-Annahmen:
    1. $\forall i : E(\varepsilon_i) = 0)$ & ε_i unabhängig identisch verteilt
    2. Homoskedastizität und Fehlen von Autokorrelation: $\forall i : Var(\varepsilon_i) = \sigma^2 I_n$
    3. Stochastische Unabhängigkeit der Residuen

[3] L. Fahrmeir,T. Kneib,S. Lang; Regression: Modelle, Methoden und Anwendungen; Springer; 2. Auflage; 2009; S. 61

[4] L. Fahrmeir,T. Kneib,S. Lang; 2009; S. 61f.

Homoskedastizität liegt vor, wenn die Streuung der Datenmessung homogon ist und nicht mit steigender Ausprägung steigt oder sinkt. Von Autokorrelation spricht man, wenn die Messungen voneinander abhängig sind, vorherige Messungen also spätere beeinflussen. Ebenso würde es den Schätzer verzerren, wenn die Residuen der einzelnen Messungen voneinander abhängig sind, was zum Beispiel durch ein fehlerhaftes Messinstrument geschehen könnte. Auf die genaue Bedeutung dieser Annahmen wird im späteren Verlauf dieses Textes eingegangen.
Für das geschätzte multiple lineare Regressionsmodell gilt folglich

$$\hat{f}(x_1, ..., x_k) = \hat{\beta}_0 + \hat{\beta}_1 x_{i1} + ... + \hat{\beta}_k x_{ki} + \varepsilon_i$$

wobei β_j den geschätzten Einfluss des jeweiligen Regressionskoeffizienten angibt[5]. x_j kann hierbei sowohl einen einzelnen Regressor darstellen als auch einen Interaktionseffekt von mehreren Regressoren. Gerade in der Medizin sind solche Interaktionseffekte oft relevant, wenn man beispielsweise an die Wechselwirkungen mehrerer Medikamente denkt.

3 Modellauswahl

Erstellte Modelle können auf zweierlei Weise vom wahren Modell abweichen:

1. Fehlende Variablen: Das korrekte Modell sei $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \varepsilon_i$, das geschätzte Modell ist aber ein Untermodell davon mit $y = \beta_0^* + \beta_1^* x_1 + \varepsilon_i^*$. Der Regressor x_2 wurde also nicht mit einbezogen.

2. Überflüssige Variablen: Das korrekte Modell sei $y = \beta_0 + \beta_1 x_1 + \varepsilon_i$, das geschätzte Modell aber ein Obermodell davon mit $y = \beta_0^* + \beta_1^* x_1 + \beta_2^* x_2 + \varepsilon_i^*$. Also ist x_2 überflüssig.

Im ersten Fall ist der Schätzer $\hat{\beta}^*$ verzerrt, schätzt also nicht den Erwartungswert (außer es besteht keinerlei korrelation zwischen x_1 und x_2). Außerdem lässt sich zeigen, dass $Var(\hat{\beta}_j^*) \leq Var(\hat{\beta}_j)$ sowie $MSE(\hat{\beta}_j^*) \leq MSE(\hat{\beta}_j)$, d.h. das sparsamere Modell besitzt sogar bessere statistische Eigenschaften als das korrekte Modell.
Im zweiten Fall ist $\hat{\beta}$ zwar unverzerrt aber es gilt $Var(\hat{\beta}_j^*) \geq Var(\hat{\beta}_j)$.
Daraus ergibt sich, dass sparsamere Modelle anzustreben sind, um den mittleren quadratischen Fehler (MSE) möglichst gering zu halten. Das gängigste Maß zur Beurteilung der Anpassungsgüte eines Modells ist das Bestimmtheitsmaß

$$R^2 = \frac{\Sigma_{i=1}^n (\hat{y}_i - \bar{y})}{\Sigma_{i=1}^n (y_i - \bar{y})}$$

welches den Anteil der erklärten Varianz an der Gesamtvarianz angibt. Je höher der Wert also ist, desto weniger Streuung in den Daten bleibt unerklärt. Gerade bei komplexeren Modellen wird

[5]L. Fahrmeir,T. Kneib,S. Lang; 2009; S. 60 u. 101f.

meist zusätzlich das adjustierte Bestimmtheitsmaß

$$R^2_{adj.} = 1 - \frac{n-1}{n-p}(1-R^2)$$

angegeben, da R^2 den rechnerischen Nachteil hat, dass es bei steigender Zahl von Kovarialen gegen 1 konvergiert (Überanpassung)[6].

3.1 Qualitätsmaße

Im Folgenden sollen zwei weitere Kriterien zur Modellauswahl vorgestellt werden. Diese sind darauf ausgelegt, Modelle unterschiedlicher Komplexität miteinander zu vergleichen.

3.1.1 Akaikes Informationskriterium

Bei der Entscheidung zwischen mehreren Modellen mit unterschiedlichen Parametern tritt das Entscheidungsproblem auf, dass Modelle einer seits so sparsam wie möglich gehalten werden sollen und andererseits jedoch die Daten so gut wie möglich wiedergeben sollen. Ersteres ist wünschenswert, um die Modelle verständlicher zu gestalten und Überanpassung an die Daten zu vermeiden. Zweiteres führt zu steigender Komplexität der Modelle und wird im Rahmen der Likelihood-Inferenz durch Maximum-Likelihood-Schätzung (ML-Schätzer) gelöst[7]. Das am häufigsten genutzte Modellwahlkriterium, welches die Methode der ML-Schätzung nutzt, nennt sich *Akaikes Informationskriterium* (AIC). Im linearen Modell ergibt sich unter der Normalverteilungsannahme:

$$AIC = n \cdot log(\hat{\sigma}^2) + 2|M+1|$$

wobei $M+1$ die Gesamtzahl der Parameter ist und $n\hat{\sigma}^2 = n^{-1}\hat{\varepsilon}'\hat{\varepsilon}$ der ML-Schätzer. Es gilt, dass kleinere Werte auf ausgewogenere Modelle hindeuten. Eine steigende Zahl von Modellparametern fließt strafend durch $2|M+1|$ in die Formel mit ein, so das es ein Ausgleich zwischen Datenannähreung und Sparsamkeit gegeben ist.

3.1.2 Bayesianisches Informationskriterium

Ein Kritikpunkt am AIC ist, dass die Stichprobengröße n nicht mit einbezogen wird. Da es aber bei steigendem n oft zu einer Verringerung der Varianz kommt, werden Modelle mit vielen Parametern bei großen Stichproben durch das AIC bevorzugt. Das Bayesianische Informationskriterium (BIC) ist unter der Normalverteilungsannahme gegeben durch:

$$BIC = n \cdot log(\hat{\sigma}^2) + log(n)|M|$$

[6]L. Fahrmeir,T. Kneib,S. Lang; 2009; S. 99ff.
[7]L. Fahrmeir,T. Kneib,S. Lang; 2009; S. 477

Im Unterschied zum AIC steigt der Strafterm beim BIC deutlich schneller mit M und ist logarithmisch von n abhängig, so dass bei der Anwendung des BIC meist sparsameren Modellen der Vorzug gegeben wird[8]. Beim Vergleich sehr unterschiedlicher Modelle sollten also beide Informationskriterien betrachtet werden.

3.2 Kollinearitätsanalyse

Von Kollinearität spricht man, wenn zwei Variablen so stark miteinander zusammenhängen, dass davon auszugehen ist, dass die Unterschiede nur auf Messungenauigkeiten beruhen und die beiden Variablen eigentlich dasselbe wiedergeben. Aus der Formel für die Varianz der Regressionskoeffizienten $\hat{\beta}_j$ (siehe Wooldridge: 2006)

$$Var(\hat{\beta}_j) = \frac{\sigma^2}{(1 - R_j^2)\Sigma_{i=1}^{n}(x_{ij} - \bar{x}_j)^2}$$

ergibt sich, dass bei steigender linearer Abhängigkeit zwischen zwei Regressoren und damit steigendem Bestimmtheitsmaß R^2 die Varianz des Schätzers gegen ∞ konvergiert. Der Varianzinflationsfaktor[9]

$$VIF_j = \frac{1}{1 - R_j^2}$$

gibt an, um welchen Faktor die Varianz $Var(\hat{\beta}_j)$ durch die Kollinearität ansteigt. Ab einem $VIF > 10$ (andere Quellen nennen 5 oder 20) wird von einem Kollinearitätsproblem ausgegangen. Es gibt mehrere Möglichkeiten, mit Multikollinearität umzugehen. Im hier vorgestellten Tool wird die simpelste Variante gewählt und Modelle mit hohem VIF schlicht ignoriert. Das entspricht im Grunde dem oft genutzten Vorgehen, dass alle bis auf eine der hoch korrelierten Variablen weggelassen werden: Restriktivere Modelle bleiben in der Liste enthalten, während Modelle mit zusätzlichen hochkorrelierten Merkmalen entfernt werden. Die betroffen Kovariablen messen exakt dasselbe, somit reicht es aus, eine davon in das Modell mit einzubeziehen.
Eine andere Variante ist, aus den korrelierenden Variablen eine neue (gut zu interpretierende) Variable zu formen. Beispielsweise können Temperaturmessungen in Grad Celsius und Fahrenheit in Kelvin umgewandelt und jeweils der Mittelwert der beiden Messungen als neue Variable herangezogen werden. Hierfür bedarf es aber Wissen über die Variablen, welches nicht in die explorative Funktion eingebaut werden kann. Für dieses Vorgehen muss der Anwender also zuvor den Datensatz auf solche Variablen hin untersucht und ihn dann neu modelliert haben.
Eine komplexere Herangehensweise, die mit Hinblick auf die Rechendauer der Modellzusammensetzung interessant ist, stellt die Dimensionsreduktion bei Regression dar. Ähnlich wie bei der Hauptkomponentenanalyse[10] wird hierbei das Modell nicht aus den Kovariablen selber, sondern

[8] K. P. Burnham, D. R. Anderson; Multimodel Inference Understanding AIC and BIC in Model Selection; 2004

[9] K. Backhaus, B. Erichson, W. Plinke, R. Weiber; Multivariate Analysemethoden – Eine anwendungsorientierte Einführung; Springer; 14. Auflage; 2015; S. 108-110

[10] R. Schlittgen; Lehr- und Handbücher der Statistik - Multivariate Statistik; De Gruyter Oldenbourg; 2. Auflage; 2016, S. 455

aus Linearkombinationen (mittels Hauptachsentransformation) gebildet, die so gestaltet sind, dass sie möglichst viel der Streuung des ursprünglichen Modells erklären. Die ursprünglichen Dimensionen werden durch neue und weniger Hauptachsen "ersetzen".

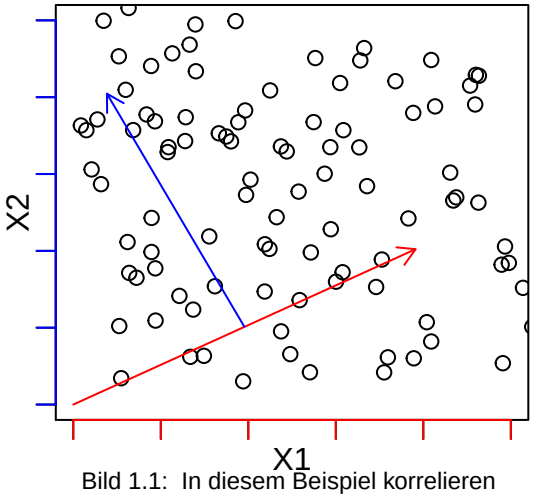

Bild 1.1: In diesem Beispiel korrelieren die beiden Merkmale kaum miteinander, so dass auch nach der Hauptachsen−transformation noch eine zweidimensionale Datenstruktur besteht

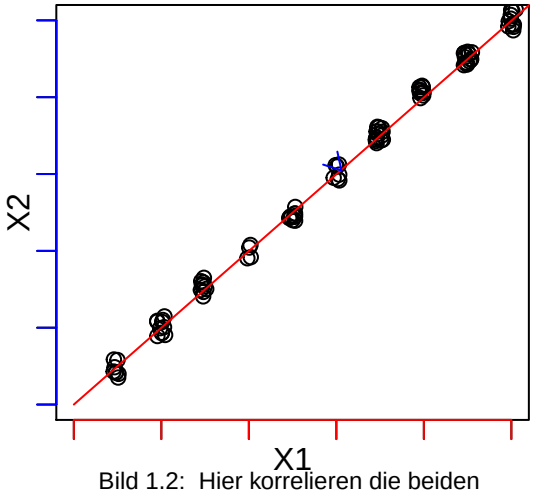

Bild 1.2: Hier korrelieren die beiden Merkmale sehr stark miteinander, so dass nach der Transformation eine Dimension ignoriert werden kann, da sie kaum zur Varianzaufklärung beiträgt

Erstellt man aus dem SHIP-Datensatz mit seinen ungefähr 200 Variablen beispielsweise Modelle mit 3 Kovariaten, so ergeben sich $\binom{200}{3} = 1.313.400$ mögliche Modelle. Gelingt es aber, durch Dimensionsreduktion den Datensatz so verkleinern, dass er beispielsweise nur noch die Dimension $150 \times n$ hat, dann sind es $\binom{150}{3} = 551.300$ Modelle, was die Rechendauer auf 42% reduziert. Oft lassen sich Datenstrukturen sogar viel weiter zusammenfassen. Um die Einsparung von Rechenaufwand durch Dimensionsreduktion zu gewährleisten, müsste die Generierung des reduzierten Datensatzes natürlich vor der Erstellung der Modelle geschehen, was wiederum einmalig sehr rechenaufwändig ist, aber im Vergleich zur komplexen Erstellung sehr vieler "unnötiger" Modelle bei hochdimensionalen Datensätzen dennoch vorzuziehen ist. Ein großer Nachteil bei diesem Vorgehen ist allerdings, die Konjunktivität von Variablen aus dem ursprünglichen Datensatz, da die Interpretation der Modelle dadurch erschwert wird. Es ist fraglich, ob der geringere Rechenaufwand die deutlich schwierigere Interpretierbarkeit der Variablen rechtfertigt, vor allem für mit dieser Methode nicht vertrauten Anwender,

3.3 Analyse von Heteroskedastizität

In den Modellannahmen wird davon ausgegangen, dass die Varianz der Störgrößen ε_i für alle Beobachtungen konstant ist, also Homoskedastizität vorliegt. Wenn ε aber in Abhängikeit des Regressors oder der Regressoren systematisch größer oder kleiner wird, spricht man von Heteroskedastizität.

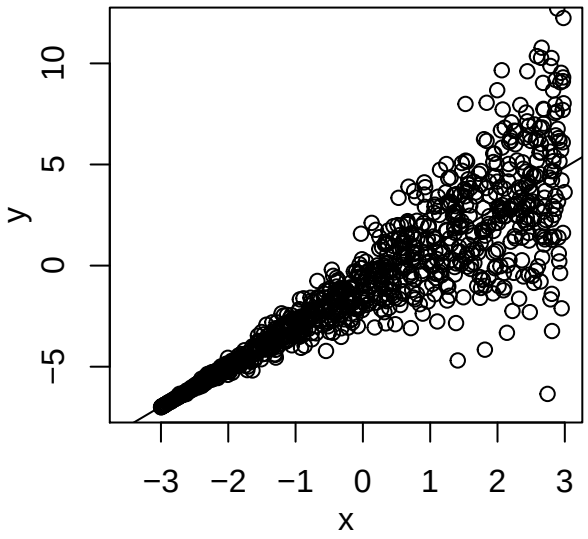

Bild 2.1: Verteilung bei welcher die Fehlerterme mit steigendem x zunehmen

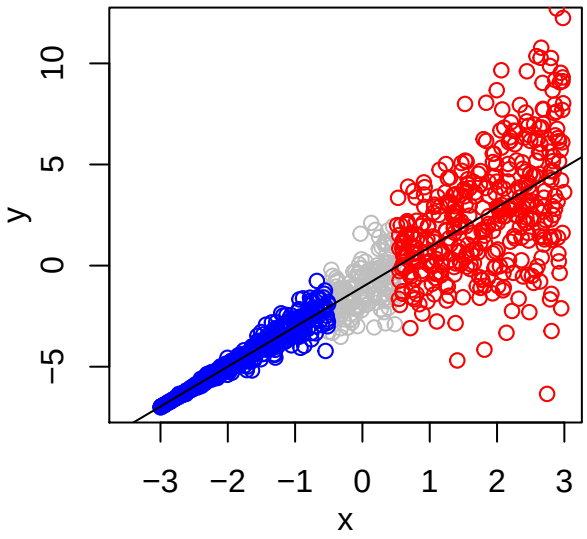

Bild 2.2: Der Goldfeld−Quandt−Test vergleicht die Varianz zweier Teilstichproben

Am besten ist Heteroskedastizität graphisch zu erkennen[11], da dabei Unzufälligkeiten der Varianzstrukturen in den ε schnell ins Auge fallen. In diesem Projekt der Versuch unternommen wird, möglichst große Teile der Analyse zu automatisieren, wurde der Goldfeld-Quandt-Test als ein Test auf Heteroskedastizität implementiert.
Dabei wird die Stichprobe hinsichtlich einer unabhängigen Variable in zwei disjunkte Teilmengen geteilt. Die Varianzen (s_1^2, s_2^2) der Teilstichproben werden dann miteinander ins Verhältnis gesetzt und mit einer F-Statistik verglichen[12]:

$$GQ - Test : \frac{s_1^2}{s_2^2} > F_{n_1-k;n_2-k;1-\alpha}$$

Wenn das Verhältnis der beiden Varianzen größer als der kritische Wert der F-Verteilung ist, muss die Nullhypothese ($s_1^2 = s_2^2$: Homoskedastizität) abgelehnt werden. Da der Software nicht ersichtlich ist, ob auf steigende ($s_1^2 < s_2^2$) oder sinkende ($s_2^2 < s_1^2$) Varianz getestet werden soll, wird ein zweiseitiger Test vorgenommen. Der Test muss außerdem für alle Regressoren durchgeführt werden, da jeder einzelne Homoskedastizität aufweisen muss, um mit der abhängigen Variable mittels linearer Regression in Verbindung gesetzt werden zu können. Er wird deshalb vor der Berechnung der Modelle durchgeführt und der Datensatz um all jene Variablen bereinigt, die den Test nicht bestehen. Natürlich unterlaufen hierbei immer wieder $\alpha-$ und $\beta-$Fehler, was an der hohen Anzahl getesteter Variablen liegt (siehe S. 11) .

[11] L. Fahrmeir,T. Kneib,S. Lang; 2009; S. 129
[12] S. M. Goldfeld, R. E. Quandt; Some Tests for Homoscedasticity; Journal of the American Statistical Association. 60, Vol. 310, Juni 1965, S. 539–547

3.4 Analyse von Normalverteilung der Residuen

Wie die Homoskedastizität sollte auch die geforderte Normalverteilung der Residuen ε_i am besten graphisch untersucht werden. Mit dieser Verteilungsannahme wird gefordert, dass die Abstände der Messwerte zur Regressionsgerade normalverteilt sind (Mittelwert=Regressionsgerade). Dies geschieht meist mittels eines QQ-Plots (Quantile-Quantile-Plot). Dafür werden bei der Regressions die Fehlerterme der Messwerte zunächst geordnet und anschließend die daraus resultierenden Quantile mit denen der theoretischen Normalverteilung verglichen. Wenn die Residuen normalverteilt sind, liegen alle Punkte des QQ-Plots auf einer Diagonalen. Wenn die Punkte zu sehr von dieser Linie abweichen, kann man nicht mehr davon ausgehen, dass eine Normalverteilung vorliegt[13].
Als Abstandsmaß wird dabei die *Mahalanobis-Distanz* verwendet.

$$d_{Mahalanobis}(x, y) = \sqrt{(x-y)^T \Sigma^{-1} (x-y)}$$

Sie berechnet den Abstand eines Punktes zu einer Verteilung im multidimensionalen Raum. Sie ist translationsinvariant und dadurch, dass sie die Kovarianzmatrix Σ der Daten mit einbezieht, ist sie im Gegensatz zur euklidischen Distanz auch skaleninvariant[14].

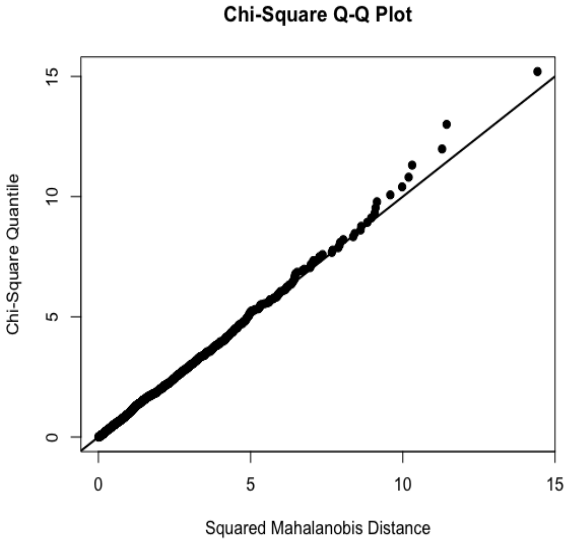

Bild 3.1: Die Punkte liegen auf der Diagonalen, deshalb kann von Normalverteilung ausgegangen werden

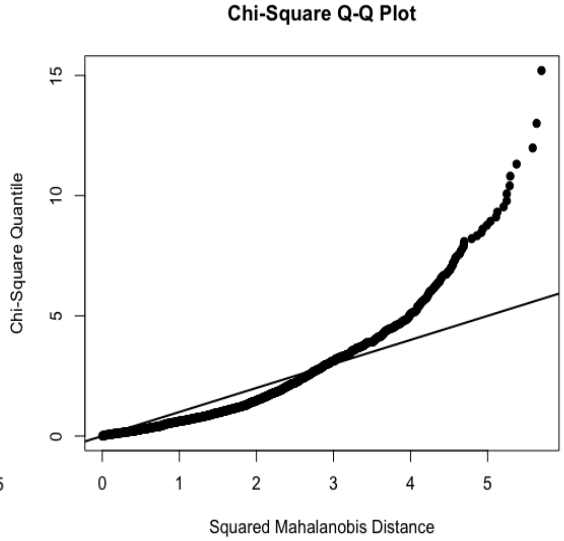

Bild 3.2: Die Quantile der Residuen decken sich nicht mit denen der Normalverteilung. Die Annahme ist also verletzt

[13] L. Fahrmeir,T. Kneib,S. Lang; 2009; S. 129f.
[14] P. C. Mahalanobis; On the generalised distance in statistics; Proceedings of the National Institute of Science of India; Vol. 2; Nr. 1; 1936; S. 49–55

Neben der graphischen Analyse gibt es einige statistische Tests auf Normalverteilung. Meist wird die graphische Analyse allerdings als sinnvoller eingeschätzt, da Tests meist auf eine Ja/Nein Entscheidung hinauslaufen. Der Grad und die Art der Abweichung von der Verteilung müssen vom Tester selbst bewertet werden, wobei zu berücksichtigen ist, dass die Differenzierbarkeit beim reinen Ablesen aus Tests leidet.
Dennnoch wurde im Tool die Möglichkeit eingebaut, einen solchen Test durchzuführen. Dabei handelt es sich um den *Royston-Test*[15], der auf dem *Shapiro-Wilk-Test* beruht und diesen für größere Stichproben (bis $n = 5000$) erweitert. Bei diesem Testverfahren wird die theoretische mit der tatsächlichen Varianz verglichen. Die Nullhypothese ist dabei, dass die beiden Varianzen unterschiedlich sind, die Residuen der Stichprobe also nicht normalverteilt sind. Das gewünschte Signifikanzniveau für diesen Test kann manuell bestimmt werden.

3.5 α-Inflation

Bei der sogenannten α-Inflation handelt es sich um ein Problem des multiplen Testens.
Es gibt zwei Fehlerarten die auftreten können:

- α-Fehler / Fehler 1. Art: Der Test erlaubt es, die Nullhypothese abzulehnen obwohl sie wahr ist.
- β-Fehler / Fehler 2. Art: Der Test erlaubt es nicht, die Nullhypothese abzulehnen obwohl die Alternativhypothese wahr ist

Der β-Fehler sorgt also dafür, dass Modelle ausgelassen werden, die Erklärungskraft hätten, während der α-Fehler Modellen Erklärungskraft zuweist, die keine haben. Deshalb wird der α-Fehler als problematischer angesehen. Die Wahrscheinlichkeit, einen Fehler 1. Art zu begehen beträgt α, also im Normalfall 5%. Werden nun mehrere aufeinanderfolgende Tests durchgeführt, wie in diesem Fall ein Goldfeld-Quandt-Test, ein Royston-Test und anschließend eine lineare Regression, multipliziert sich die Wahrscheinlichkeit, einen Fehler 1. Art zu begehen. Wenn drei aufeinanderfolgende Tests vorgenommen werden, so ergibt sich eine Fehlerwahrscheinlichkeit von $(1 - (1 - 0.05)^3) \approx 14.3\%$.
Um dieses Problem in den Griff zu bekommen, wird meistens die *Bonferroni-Holm-Prozedur* angewandt. Hierbei wird berechnet, wie klein α gewählt werden muss, damit sich nach Kumulierung die gewünschte Fehlerwahrscheinlichkeit ergibt
Eine Idee welche aus testtheoretischer Sicht zwar problematisch, wegen des explorativen Charakters der Untersuchung aber denkbar ist, wäre, die jeweiligen p-Werte, also die "kleinstmöglichen" α-Werte aller Tests für ein Modell miteinander zu verrechnen. So kann ermittelt werden, wie groß die kumulierte Fehlerwahrscheinlichkeit für die einzelnen Modelle ist. Bisher wurde diese Idee

[15] J. P. Royston; An Extension of Shapiro and Wilk's W Test for Normality to Large Samples; Journal of the Royal Statistical Society. Series C (Applied Statistics); Vol. 31; Vol.2; 1982; S. 115-124

aber noch nicht in den Code implementiert, so dass das Problem bisher nur über die*Bonferroni-Holm-Prozedur*, also die manuelle Anpassung der drei α-Werte (alpha, alpha.royston, alpha.gq), behandelt werden kann[16].
Ein weiteres Problem ergibt sich schließlich aus der schieren Menge der möglichen Modelle. Unabhängig von vorherigen Tests werden bei der explorativen Untersuchung möglicherweise tausende Modelle getestet. Dieses Problem ist vom Gedanken her ähnlich der α-Inflation. Mit jedem weiteren Modell steigt die Wahrscheinlichkeit, dass bei einer Teilmenge der Modelle ein α-Fehler vorliegt. Der Fehler bei den betroffenen Modellen würde aber selbstverständlich auch dann auftreten, wenn die Modelle alleine getestet würden. Hier hilft nur, ein kleineres α zu wählen.

4 Explorative Modellbildung bei Regressionsverfahren

Anders als üblicherweise in der Testtheorie wurde keine spzeifische, zu testende Hypothese aufgestellt. Es besteht lediglich Interesse daran, ein Merkmal zu erklären, ohne vorher allzu viele Annahmen zu treffen, welche anderen Variablen dieses Merkmal erklären können. Die Anforderung ist, alle möglichen Modelle zu testen und jene herauszufiltern, die eine statistische Erklärungskraft besitzen. Da dies per Hand für eine solche Menge an Modellen nicht mehr möglich ist, wurde der Versuch unternommen, die Arbeitsschritte weitestgehend mittels einer Funktion in R zu automatisieren.

4.1 Anforderung an den Datensatz

Im Folgenden wird der Aufbau der Funktion dargestellt, die in R durchgeführt wird. Damit die Daten bearbeitet werden können, müssen sie das Format eines Datensatzes (*data.frame()*) oder einer Liste (*list()*) haben. Im Falle eines Datensatzes muss es sich um einen strukturierten Datensatz handeln, in welchem die Variablen spaltenweise und die Testobjekte zeilenweise angeordnet sind.
Die Funktion wurde aufgrund ihrer explorativen Arbeitsweise mit “explore“ bezeichnet. In der Anwendung soll sie durch mehrere Argumente je nach Wunsch veränderbar sein. Die beiden wichtigsten Argumente sind hierbei y und d. Dem y muss als Objekt die abhängige Variable zugewiesen werden. In d sind die Regressoren der späteren Analyse enthalten und diese Matrix muss als Liste oder Datensatz vorliegen.

4.2 Aufbau der *explore*-Funktion

Die Argumente *alpha, Rsquared* und *VIF* dienen dazu, von vorneherein Modelle auszuschließen. Standardmäßig sind sie so gesetzt, dass alle möglichen Modelle ausgegeben werden. Entschließt sich ein Anwender beispielsweise dazu, nur Modelle mit einem Variationskoeffizienten $R^2 > 0.3$

[16]S. Holm; A simple sequentially rejective multiple test procedure; Scandinavian Journal of Statistics; Vol. 6; 1979; S. 65–70.

zu untersuchen, so kann er *Rsquared* diesen Wert zuweisen, wodurch Modelle mit niedrigerem R^2 nicht mit ausgegeben werden. *alpha* schließt Modelle aus, unter welchen keiner der Regressoren auf einem $(1-\alpha)$-Niveau Signifikanz aufweist, hiermit lässt sich also die Fehlerwahrscheinlichkeit abwägen (siehe S. 11). Der Varianzinflationsfaktor (*VIF*) dient der Filterung von Modellen mit hoher Multikollinearität (siehe S.7). Je größer der VIF, desto stärker sind die Hinweise auf Multikollinearität.
Im Weiteren wurde die Funktion so aufgebaut, dass eine spätere Implementierung von Tests für andere Skalenniveaus leichter geschehen kann. Bisher sind aber nur multiple Regressionsmodelle aufbauend auf (quasi-)metrischen Variablen möglich (*intervall=T*).
Außerdem kann spezifiziert werden, ob die Daten für die Berechnung der Regressionsmodelle zentriert werden (*center*). Da dies für die Interpretation des Y-Achsenabschnitts hilfreich sein kann und keinen rechnerischen Nachteil (bis auf minimalen Zeitaufwand) mit sich bringt, wird es standardmäßig durchgeführt.
Mittels des Arguments *model* kann die Komplexität der erstellten Modelle variiert werden. Momentan sind allerdings erst lineare univariate Modelle ($y_i = \hat{\beta}_0 + \hat{\beta}_1 x_{i1}$), lineare bivariate Modelle ($y_i = \hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \hat{\beta}_2 x_{i2}$), sowie quadratische univariate Modelle ($y_i = \hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \hat{\beta}_2 x_{i1}^2$) möglich und wählbar (*"unilin", "unisq", "bilin"*).
Als letztes Argument kann eingegeben werden, nach welchem Kriterium die zum Schluss ausgebene Liste der Modelle geordnet wird. Dabei stehen folgende Möglichkeiten zur Verfügung: *Number* (Laufnummer), *Modell* (alphabetisch nach Variablenbezeichnungen der Modelle), R^2, adjustiertes R^2, VIF, BIC, AIC. Die Möglichkeiten der Nutzung der Funktion sind also vielseitig und individuell manipulierbar. Im Folgenden wird der weitere Ablauf der Funktion beschrieben, der für den Anwender nicht direkt sichtbar ist.
Zunächst wird sichergestellt, dass die Struktur des unter *d* angegebenen Objekts in einen Datensatz umgewandelt wird, der als *data* bezeichnet ist. Für den Fall, dass im Datensatz Spalten enthalten sind, welche nur aus fehlenden Werten bestehen, werden diese gelöscht. Die Positionen aller numerischen Spalten werden in *numcounta* abgespeichert und bilden die Menge der für die Regressionsmodelle ausgewählten Variablen. Die abhängige Variable wird zwischengespeichert, da sie später aus dem Datensatz gelöscht wird, um nicht mit sich selber in Beziehung gesetzt zu werden. Im nächsten Schritt wird, falls nicht anders gewünscht, die Zentrierung durchgeführt. Es wird dabei von jedem metrischen Merkmal der Mittelwert errechnet und von den einzelnen Ausprägungen subtrahiert, so dass der neue Mittelwert der zentrierten Merkmale bei Null liegt ($x_{ji} - \bar{x}_j$). Im letzten Schritt dieser Funktion wird schließlich die Unterfunktion *bothnum()* durchgeführt, welche die Regressionsmodelle für metrische abhängige und unabhängige Variablen zusammensetzt.

```
explore <- function(y, d,alpha=1,Rsquared=-1, VIF=100, intervall=T, nominal=F,
                    logical=F, center=T,model, order=c("AIC"), alpha.gq=1,
                    alpha.royston=0)
{if(!any(model=="unilin" | model=="bilin" | model=="unisq"))
```

```
{stop("model has to be at least one of the following: 'unilin', 'bilin',
      'unisq'") }
data <- data.frame(d);
Filter(function(x)!all(is.na(x)), data);
numcounta <- c(as.numeric(which(sapply(data, is.numeric))));
av <- y;
if(center==T)
{for(i in numcounta)
  {data[,i] <-c(data[,i]-mean(data[,i], na.rm=T))}
  av <- av-mean(av,na.rm=T)}
datlength <- ncol(data);
dataold <- data;t
for(i in 1:datlength)
{if(identical(av,dataold[,i])==T)
  {data[,i] <- NULL}}
numcount <- c(as.numeric(which(sapply(data, is.numeric))));
if(is.numeric(av)==T & intervall==T)
 {c5 <- 1
  homosk <- c()
  for(k in numcount){
    if(gqtest(av~data[,k], alternative="two.sided",
              order.by = ~data[,k])[[4]]>alpha){
      homosk[c5] <- k
      c5 <- c5+1}}
  bothnum(av,data,homosk, model=model, alpha=alpha, Rsquared=Rsquared,
          VIF=VIF, order,
          alpha.gq=alpha.gq, alpha.royston=alpha.royston)}else{}}
```

4.3 Erstellung der Modelle über *bothnum*

Für die Unterfunktion *bothnum()* werden zunächst einige Argumente aus *explore()* übertragen. Die Funktion dient wie bereits erwähnt dazu, Regressionsmodelle aus den vorher ausgewählten metrischen Variablen des Datensatzes zusammenzustellen. Je nach Einstellung des *model*-Arguments werden Modelle von unterschiedlicher Komplexität erstellt. Der Ablauf ist von der Struktur her aber immer der selbe. Über eine oder mehrere Schleifen werden zunächst die Variablen für ein Modell zusammengesellt (Anmerkung: Die Funktion *Poly* wird durchgeführt, um Probleme mit fehlenden Werten bei einigen Tests zu umgehen). Diese Konstellationen werden dann auf Homoskedastizität (*Goldfeld-Quandt-Test* siehe S.8) und Normalverteilung der Residuen (*Royston-Test*

siehe S. 10) überprüft. Standardmäßig sind α-Werte für diese Tests allerdings so gewählt, dass die Tests keine Variablen ausschließen. Die Gründe für dieses Vorgehen sind erstens, dass gerade die Normalverteilungsannahme der Residuen in der Realität oft nur approximativ gegeben ist und deshalb in der Praxis, kritischerweise, oft nicht überprüft oder die Nichterfüllung sogar ignoriert wird. Sie ist aber im Zusammenhang mit der Modellbewertung über das BIC wichtig (siehe S. 6). Zweitens sollen die Parameter selber eingestellt werden können, um α-Inflation zu kontrollieren (siehe S.11). Danach wird untersucht, ob das Modell die Anforderungen der Argumente (α, R^2, VIF) erfüllt und wenn dies der Fall ist, wird das Modell schließlich in einer Liste (*lelist*) gespeichert. Die einzelnen Listenelemente bestehen hierbei aus mehreren Unterpunkten, in welche die folgenden Merkmale des Modells eingetragen werden:

- Identifikationsnummer des Modells
- Modellbezeichnung. Beispielsweise: $Tumorgörße = Alkohol.pro.Woche + Gewicht + Gewicht^2$
- Die mittels des *summary(lm())- / summary(glm())-* Befehls ermittelten Kennzahlen der Regression
- Weitere Kennzahlen: R^2, *adjustiertes* R^2, *VIF, AIC, BIC*

Anzumerken ist des Weiteren, dass die linearen univariaten Modelle immer erstellt werden, da sie leicht zu berechnen sind und sowohl der Goldfeld-Quandt- als auch der Royston-Test in R nur aufbauend auf einem zuvor berechneten Regressionsmodell durchgeführt werden können. Erfüllt eine der Variablen diese Bedingungen nicht, wird sie nicht in die Liste eingetragen und wird vor allem für die komplexeren Modelle nicht mehr als Kandidat herangezogen, was die Rechendauer verringert.

```
bothnum <- function(y,d,n, model, alpha, Rsquared, VIF, order, alpha.gq,
                    alpha.royston){
  lelist <- list()
  ordervec <- c()
  goodies <- c()
  c3 <- 1
  c4 <- 1
  Poly<- function(x, degree = 1, coefs = NULL, raw = FALSE, ...){
    notNA<-!is.na(x)
    answer<-poly(x[notNA], degree=degree, coefs=coefs, raw=raw, ...)
    THEMATRIX<-matrix(NA, nrow=length(x), ncol=degree)
    THEMATRIX[notNA,]<-answer
    attributes(THEMATRIX)[c('degree', 'coefs', 'class')]<- attributes(answer)[c(
      'degree', 'coefs', 'class')]
```

```
  THEMATRIX}
############################ univariat linear  ############################
for(i in n)
{roydat <- data.frame(y,d[,i])
if(sum(table(y*d[,i]))<5)
{}else{
  zlm <- lm(y~d[,i], na.action=na.omit);
  if(AIC(zlm)>0 & all(as.matrix(summary(zlm)[4])[[1]][,4]<alpha) &
     length(as.matrix(summary(zlm)[4])[[1]][,4])>1 & summary(zlm)[9]>Rsquared &
     (1/(1-as.numeric(summary(zlm)[8])))<VIF)
  {if(gqtest(y~d[,i], alternative="two.sided",order.by= ~d[,i])[[4]]<alpha.gq &
      royston.test(roydat[complete.cases(roydat),])$p.value>alpha.royston)
  {if(any(model=="unilin"))
  { lelist[[c4]] <- list()
  lelist[[c4]]$Number <- c4
  lelist[[c4]]$Modell <- paste("y","=",names(d[i]),sep="")
  lelist[[c4]]$Regression <- summary(zlm)[4]
  lelist[[c4]]$R_squared <- as.numeric(summary(zlm)[8][[1]])
  lelist[[c4]]$adj.R_squared <- as.numeric(summary(zlm)[9][[1]])
  lelist[[c4]]$VIF <- (1/(1-as.numeric(summary(zlm)[8][[1]])))
  lelist[[c4]]$AIC <- AIC(zlm)
  lelist[[c4]]$BIC <- BIC(zlm)
  aicvec[c4] <- lelist[[c4]][which(names(lelist[[c4]])==order)][[1]];
  c4 <- c4+1}}}
  goodies[c3] <- i
  c3 <- c3+1}}
n <- goodies
############################ bivariat linear  ############################
if(any(model=="bilin"))
{for(k in goodies)
{for(l in goodies)
{if(sum(table(y*d[,k]*d[,l]))<5 | k>l){}else
{ zlm <- lm(y~d[,k]+d[,l], na.action=na.omit)
if(AIC(zlm)>0 & all(as.matrix(summary(zlm)[4])[[1]][,4]<alpha) &
   length(as.matrix(summary(zlm)[4])[[1]][,4])>2 & summary(zlm)[9]>Rsquared &
   (1/(1-as.numeric(summary(zlm)[8])))<VIF)
{ lelist[[c4]] <- list()
lelist[[c4]]$Number <- c4
```

```
lelist[[c4]]$Modell <- paste("y","=",names(d[k]),"+",names(d[l]),sep="")
lelist[[c4]]$Regression <- summary(zlm)[4]
lelist[[c4]]$R_squared <- as.numeric(summary(zlm)[8][[1]])
lelist[[c4]]$adj.R_squared <- as.numeric(summary(zlm)[9][[1]])
lelist[[c4]]$VIF <- (1/(1-as.numeric(summary(zlm)[8][[1]])))
lelist[[c4]]$AIC <- AIC(zlm);
lelist[[c4]]$BIC <- BIC(zlm)
aicvec[c4] <- lelist[[c4]][which(names(lelist[[c4]])==order)][[1]];
c4 <- c4+1 }}}}}
############################ univariat quadratisch ############################
if(any(model=="unisq")){
  for(i in goodies)
  {if(sum(table(y*d[,i]))<5)
  {}else
  { zlm <- lm(y~Poly(d[,i],2,raw=T), na.action=na.omit)
  if(AIC(zlm)>0 & all(as.matrix(summary(zlm)[4])[[1]][,4]<alpha) &
     length(as.matrix(summary(zlm)[4])[[1]][,4])>1 & summary(zlm)[9]>Rsquared &
     (1/(1-as.numeric(summary(zlm)[8])))<VIF)
  { lelist[[c4]] <- list()
  lelist[[c4]]$Number <- c4
  lelist[[c4]]$Modell <- paste("y","=",names(d[i]),"+",names(d[i]),"^2",sep="")
  lelist[[c4]]$Regression <- summary(zlm)[4]
  lelist[[c4]]$R_squared <- as.numeric(summary(zlm)[8][[1]])
  lelist[[c4]]$adj.R_squared <- as.numeric(summary(zlm)[9][[1]])
  lelist[[c4]]$VIF <- (1/(1-as.numeric(summary(zlm)[8][[1]])))
  lelist[[c4]]$AIC <- AIC(zlm)
  lelist[[c4]]$BIC <- BIC(zlm)
  aicvec[c4] <- lelist[[c4]][which(names(lelist[[c4]])==order)][[1]];
  c4 <- c4+1}}}}
aiccountvec <- c(seq(1:length(aicvec)))
orderfun(lelist=lelist,aiccountvec=aiccountvec,ordervec=aicvec,
         orderlist=orderlist,d=d)}
```

4.4 Sortierung der Modelle über *orderfun()*

Nachdem in *bothnum()* die Modelle in die Liste eingetragen wurden, können sie nach dem in *order* beschriebenen Parameter sortiert werden, so dass jene Modelle, welche diese Kriterien am ehesten erfüllen an oberster/unterster Stelle stehen.

```
orderfun <- function(lelist,aiccountvec,ordervec,orderlist,d)
{ c1 <- 1
  orderlist <- list()
  ordermat <- matrix(c(aiccountvec,ordervec),byrow=T,nrow=2)
  orderordermat <- ordermat[, sort.list(ordermat[2,], decreasing=T) ]
  for(i in 1:ncol(ordermat))
  {orderlist[[c1]] <- lelist[[orderordermat[1,i]]]
  c1 <- c1+1}
print(orderlist)}
```

4.5 Beispielhafte Anwendung

Eine beispielhafte Eingabe und Nutzung der Funktion könnte wie folgt aussehen:

$$explore(y = Dat\$Segmentation\_ParenchymToVolume,\ d = Dat,\ alpha = .05,\ Rsquared = 0.2,\ VIF = 10,\ intervall = T,\ center = T,\ model = c(\text{"}unilin\text{"},\text{"}unisq\text{"}),\ order = c(\text{"}AIC\text{"}))$$

Hierbei ist *Dat$Segmentation_ParenchymToVolume* die Tumorgröße, also die abhängige Variable. Diese wird mittels intervallskalierten Variablen aus dem Datensatz *Dat* geschätzt. Es wird eine Liste all jener Modelle ausgegeben, die univariat-linearer (*"unillin"*) oder univariat-quadratischer (*"unisq"*) Struktur sind und folgende Bedingungen erfüllen:

- $\alpha > 5\%$
- *adjustiertes* $R^2 > 20\%$
- $VIF < 10$

Die Variablen sind außerdem zentriert worden und die Liste wird nach dem AIC geordnet ausgegeben. Von den insgesamt 13 Modellen, die die Anforderungen erfüllen, werden nur die letzten beiden Elemente der ausgegeben Liste examplarisch im Anhang gezeigt, um den Rahmen dieser Arbeit nicht zu sprengen.

5 Ausblick

Das hier vorgestellte Tool ist in seiner jetzigen Form noch lange nicht ausgereift. Aktuell ist es nur möglich, metrische Variablen in die Analyse mit einzubeziehen. Nominal und ordinal skalierte Merkmale können mittels der linearen Regression nicht getestet werden. Für binäre abhängige Variablen müsste das Tool um *generalisierte lineare* Modelle erweitert werden. Je nach Kombination von unterschiedlich skalierten Merkmalen müsste eine eigene Methode oder ein eigener Test implementiert werden. Die Funktion ist aber bereits so gestaltet, dass es möglich ist, neue

Modelle zu implementieren und einzustellen, welche Skalenniveaus berücksichtigt werden sollen. Problematisch ist aber in jedem Fall der Vergleich zwischen den Ergebnissen unterschiedlicher Testmethoden. So kann ein Chi-Quadrat Test, wie er für die Testung bei nominalen Merkmalen genutzt wird, nicht mit den Ergebnissen einer multiplen Regression verglichen werden. Vergleiche sind stets nur zwischen Ergebnissen gleicher Tests möglich.
Momentan ist außerdem die Anwendung noch relativ komplex, d.h. der Einarbeitsungsaufwand ist hoch. Für eine breitere Anwendung der Funktion muss noch auf mehreren Ebenen weitergearbeitet werden. Diese Ebenen sind: Handhabbarkeit, graphische Analyse und Verringerung des Rechenaufwands bei komplexen Modellen.

5.1 Praktische Anwendung und Visualisierung

Für die Handhabbarkeit ist beispielsweise eine andere Arbeitsumgebung notwendig, wie etwa eine selbsterklärende Website oder ein handlicheres Programm. Es sollte darauf abgezielt werden, dass kein tieferes Verständnis der Software *R* notwendig ist, um Analysen durchzuführen. Im Idealfall liegt ein auf *R* beruhendes Programm vor, in welchem die einzelnen Parameter und Tests einfach angeklickt werden können, um sie individuell zu gestalten.
Auch graphische Analysen, die im Text bereits mehrfach betont wurden, könnten auf diese Weise eingebaut werden.

> *The visualization of multivariate data using multiple connected views allows to get fast visual feedback about subject groups.*[17]

Vorstellbar ist eine interaktive Umgebung, in welcher nach Durchführung der Modellberechnung die wichtigsten Analysegraphiken für jedes Modell angezeigt werden, um damit die einzelnen Modelle besser hinterfragen und verstehen zu können, als es durch profane Zahlenwerte geschehen kann. Dies würde es möglich machen, dass sowohl der Anwender bei der Nutzung untersützt wird, als dass auch fehlerhafte Testentscheidungen erkannt werden.
In diesem Zusammhang sei noch einmal betont, dass es je nach Datensatz zu Modellen kommt, die wenig Erkenntnisgewinn liefern, vielleicht sogar tautologisch sind. Wenn beispielsweise die Körpergröße in Zentimetern als abhängige Variable gewählt wurde und im Datensatz auch die Größe in Inch als Variable enthalten ist, dann wird daraus ein hochsignifikantes Modell gebildet, welches aber keinerlei Erklärungskraft besitzt. Solche Modelle können nur vom Anwender, aber nicht immer vom Programm selber, erkannt werden. In einem interaktiv gestalteten Programm sollte dann auch die Möglichkeit bestehen, solche Modelle und irrelevanten Variablen aus der Gesamtmenge der Modelle oder Variablen auszuschließen, ohne dass tieferes Wissen über Datensatztransformation notwendig ist.

[17] P. Klemm, S. Glaßer, K. Lawonn, M. Rak, H. Völzke, K. Hegenscheid, B. Preim; Interactive Visual Analysis of Lumber Back Pain - What the Lumber Spine Tells About Your Life; IEEE Transactions on Visualization and Computer Graphics (TVCG); 2014; S. 1673-1682

5.2 Genetische Algorithmen zur Modellauswahl

Eine weitere Unannehmlichkeit ist folgendes Problem: Die Zahl der möglichen Modelle steigt exponentiell mit der Anzahl der gewünschten Regressoren. Das führt wie bereits erwähnt dazu, dass der Rechenaufwand bei größeren Datensätzen sehr hoch wird. Von daher liegt liegt der Gedanke nahe, die Suche nach guten Modellen über einen genetischen Algorithmus zu gestalten.
Als Bewertungskriterium würden dabei die Modellauswahlkriterien (R^2, AIC, BIC, VIF), sowie statistische Tests auf Signifikanz und Erfüllung der Regressionsovoraussetzungen dienen.
Für die Mutation wäre zunächst wichtig, welches Startmodell gewählt wird. Eine Möglichkeit ist es, ein sehr einfaches Modell zu wählen, welches sich dann durch Hinzunahme von Kovariablen solange entwickelt, bis es keine Verbesserung mehr eintritt (Vorwärts-Selektion). Es könnte aber auch mit dem "vollen Modell", also einem, welches alle Kovariablen enthält, begonnen werden und anschließend könnten schrittweise Kovariablen entfernt werden (Rückwärts-Selektion).
Unabhängig davon, welcher Startpunkt und welche Mutationsweise gewählt werden, besteht der Vortei ldarin, dass nicht alle möglichen Modelle getestet werden. Dabei ist natürlich, wie meist bei genetischen Algorithmen, nicht sichergestellt, dass dabei auch das beste Modell herauskommt, da der Algorithmus in einem lokalen Maximum verharren kann. Auch ist es in Hinblick auf die irrelevanten Modelle wohl nötig, den Algorithmus mehrmals durchlaufen zu lassen oder nur einen semi-genetischen Algorithmus zu wählen, also einen solchen, der gute Zwischenergebnisse speichert und später mit ausgibt. Ein vergleichbares Vorgehen wird in "*Regressions by Leaps and Bounds (1974)*" von M. Furnival und R. W. Wilson vorgestellt und findet Anwendung im R-Paket "*leaps*".
Die genaue Beschreibung der Vor- und Nachteile von genetischen Algorithmen bei der Anwendung auf explorative Modellgestaltung würde den Rahmen dieser Arbeit aber übersteigen, wird aber das Ziel eines umfassenderen Projektes sein.

6 Quellen

- K. Backhaus, B. Erichson, W. Plinke, R. Weiber; Multivariate Analysemethoden – Eine anwendungsorientierte Einführung; Springer; 14. Auflage; 2015
- Bundesgesundheitsblatt 2012 · 55:790–794 · DOI 10.1007/s00103-012-1483-6; Springer; 2012
- K. P. Burnham, D. R. Anderson; Multimodel Inference Understanding AIC and BIC in Model Selection; Colorado Cooperative Fish and Wildlife Research Unit (USGS-BRD); 2004
- L. Fahrmeir,T. Kneib,S. Lang; Regression: Modelle, Methoden und Anwendungen; Springer; 2. Auflage; 2009
- S. M. Goldfeld, R. E. Quandt; Some Tests for Homoscedasticity; Journal of the American Statistical Association. 60, Vol. 310, Juni 1965
- S. Holm; A simple sequentially rejective multiple test procedure; Scandinavian Journal of Statistics; Vol. 6; 1979
- P. Klemm, S. Oeltze-Jafra, K. Lawonn, K. Hegenscheid, H. Völzke, B. Preim; Interactive Visual Analysis of Image-Centric Cohort Study Data; IEEE Transactions on Visualization and Computer Graphics (TVCG); 2014; S. 1673-1682
- P. Klemm, S. Glaßer, K. Lawonn, M. Rak, H. Völzke, K. Hegenscheid, B. Preim; Interactive Visual Analysis of Lumber Back Pain - What the Lumber Spine Tells About Your Life; IEEE Transactions on Visualization and Computer Graphics (TVCG); 2014; S. 1673-1682
- P. Klemm, K. Lawonn, S. Glaßer, U. Niemann, K. Hegenscheid, H. Völzke, B. Preim; 3D Regression Heat Map Analysis of Population Study Data; IEEE Transactions on Visualization and Computer Graphics (TVCG); Vol. 22 (1); 2015; S. 81-90
- P. C. Mahalanobis; On the generalised distance in statistics; Proceedings of the National Institute of Science of India; Vol. 2; Nr. 1; 1936
- J. P. Royston; An Extension of Shapiro and Wilk's W Test for Normality to Large Samples; Journal of the Royal Statistical Society. Series C (Applied Statistics); Vol. 31; Vol.2; 1982
- R. Schlittgen; Lehr- und Handbücher der Statistik - Multivariate Statistik; De Gruyter Oldenbourg; 2. Auflage; 2016
- J. M. Wooldridge; Introdictory Exonometrics; South-Western Cengage Learning; 2006

Des Weiteren wurden folgende, frei zugängliche R-Pakete verwendet:
royston: `https://cran.r-project.org/web/packages/royston/index.html`
lmtest: `https://cran.r-project.org/web/packages/lmtest/index.html`
foreign: `https://cran.r-project.org/web/packages/foreign/index.html`
car: `https://cran.r-project.org/web/packages/car/index.html`
MASS: `https://cran.r-project.org/web/packages/MASS/index.html`
MVN: `https://cran.r-project.org/web/packages/MVN/index.html`
ggplot2: `https://cran.r-project.org/web/packages/ggplot2/index.html`

7 Anhang

```
[[12]]
[[12]]$Number
[1] 12

[[12]]$Modell
[1] "y=STRATA+STRATA^2"

[[12]]$Regression
[[12]]$Regression$coefficients
                              Estimate   Std. Error   t value     Pr(>|t|)
(Intercept)                2.288099e+01 1.103181e+00 20.740927 4.621540e-77
Poly(d[, i], 2, raw = T)1 -3.105415e-02 3.541491e-03 -8.768667 1.042213e-17
Poly(d[, i], 2, raw = T)2  1.306526e-05 2.620743e-06  4.985327 7.559037e-07


[[12]]$R_squared
[1] 0.2997613

[[12]]$adj.R_squared
[1] 0.2980429

[[12]]$VIF
[1] 1.428084

[[12]]$AIC
[1] 5214.983

[[12]]$BIC
[1] 5233.811
[[13]]
[[13]]$Number
[1] 13

[[13]]$Modell
[1] "y=Segmentation_Right_Fat+Segmentation_Right_Fat^2"

[[13]]$Regression
[[13]]$Regression$coefficients
                              Estimate   Std. Error   t value     Pr(>|t|)
(Intercept)                2.504394e+01 4.348450e+00  5.759280 0.0001828335
Poly(d[, i], 2, raw = T)1 -3.770972e-02 9.495656e-03 -3.971260 0.0026379640
Poly(d[, i], 2, raw = T)2  1.674735e-05 4.587031e-06  3.651021 0.0044547649


[[13]]$R_squared
[1] 0.640758

[[13]]$adj.R_squared
[1] 0.5689096

[[13]]$VIF
[1] 2.783639

[[13]]$AIC
[1] 64.00066

[[13]]$BIC
[1] 66.26046
```

8 Eidenstattliche Erklärung

Ich, Daniel Schneider, versichere hiermit, diese Arbeit selbstständig und lediglich unter Benutzung der angegebenen Quellen und Hilfsmittel verfasst zu haben.
Ich erkläre weiterhin, dass die vorliegende Arbeit noch nicht im Rahmen eines anderen Prüfungsverfahrens eingereicht wurde.

Daniel Schneider, 05.03.2017, Wiesloch