Stand: 3/2018

Ökonometrie mit R

Reinhold Kosfeld

Inhaltsverzeichnis

1. Einführung

1.1 Vorbemerkungen
1.2 Rechnen mit Zahlen und Funktionen
1.3 Variablen, Vektoren und Matrizen
1.4 Zeitreihenobjekt und -diagramm
1.5 Einlesen von Zeitreihen aus Datendateien

2. Regressionsmodell und Kleinst-Quadrate-Methode

2.1 Einfache Regressionsanalyse und lm-Funktion
2.2 Multiple Regressionsanalyse und lm-Funktion
2.3 Regressionsanalyse und Matrizenrechnung

3. Multikollinearität

3.1 Regression mit drei erklärenden Variablen
3.2 Einfache und erweiterte Korrelationsanalyse
3.3 Determinante und Konditionszahl
3.4 Verfahren der Hilfsregressionen (auxiliary regressions)

4. Heteroskedastizität

4.1 Tests auf Heteroskedastizität
4.1.1 Goldfeld-Quandt-Test
4.1.2 Breusch-Pagan-Test
4.2 Gewichtete Kleinst-Quadrate-Schätzung (WLS)
4.3 Heteroskedastizitäts-konsistente (HC) Standardfehler

5. Autokorrelation

5.1 Autokorrelationskoeffizient und Korrelogramm
5.2 Tests auf Autokorrelation
5.2.1 Durbin-Watson-Test
5.2.2 Breusch-Godfrey-Test
5.3 Das Cochran-Orcutt-Verfahren
5.4 Heteroskedastizitäts- und Autokorrelations-konsistente (HAC) Standardfehler

6. Modelle mit qualitativen Variablen

6.1 Modelle mit qualitativen Regressoren

6.2 Strukturbruchtest

6.3 Lineares Wahrscheinlichkeitsmodell

6.3 Logit-Modell

1. Einführung
1.1 Vorbemerkungen

Die Statistiksoftware und Programmierumgebung R wird von einer internationalen Entwicklergemeinschaft als Open Source Software entwickelt und kann von jedermann entsprechend der GNU General Public License Version 2 (Juni 1991) frei genutzt und auch weiter verbreitet werden. Die Software wird auf der Homepage des R - Projektes (http://www.r-project.org) als Download bereitgestellt. Quellcode und Binärdateien für diverse Betriebssysteme werden durch CRAN bereitgestellt (http://CRAN.R-project.org).

R ist eine hochflexible, interpretierte Programmiersprache und –umgebung zur statistischen und grafischen Datenanalyse. Es handelt sich hierbei um keine vollständige, grafische Benutzeroberfläche (GUI), jedoch sind Werkzeuge zu ihrer Entwicklung vorhanden. In dem R-Commander lassen sich aktuell bereits einige Methoden der Datenanalyse menügesteuert ausführen. Bei der Ausführung von R-Funktionen erfolgt in der Regel keine Ausgabe aller errechneten Werte. Vielmehr werden einige Ergebnisse erst einmal in Objekten zwischengespeichert. Sie können später jederzeit abgerufen werden. Andererseits speichert R keine Ergebnisse ab, wenn kein Objekt hierzu angegeben wird.

R kann sowohl interaktiv oder im Batch-Modus genutzt werden. Der Batch-Modus sollte bei größeren Befehlsfolgen verwendet werden, in dem eine Skript-Datei ausgeführt wird. Das Programm ist bereit, wenn das Zeichen > erscheint. Befehle können durch ein Semikolon (;) oder durch den Beginn einer neuen Zeile voneinander getrennt werden. Ist ein Befehl nicht vollständig, erscheint das Pluszeichen (+). Bereits verwendete Befehle können mit den Pfeiltasten (↑ und ↓) abgerufen werden. Mit dem Zeichen # wird ein Kommentar eingeleitet. Als Dezimalzeichen wird der Punkt (.), nicht das Komma (,) verwendet. Die ESC-Taste unterbricht die gerade ausgeführte Rechenprozedur.

Alle in einer R-Sitzung vorhandenen Objekte können mit dem Befehl ls() angezeigt. Ein Objekt obj lässt sich mit dem remove-Befehl rm(obj) löschen. Texte wie z.B. „Ökonometrie macht Spaß“können unter Verwendung von Anführungszeichen mit einem Gleichheitszeichen (=) oder Pfeil (<-) einem Objekt obj zugewiesen werden:

```
>obj = "Ökonometrie macht Spaß"   oder   >obj <- "Ökonometrie macht Spaß"
>obj
[1] "Ökonometrie macht Spaß ".
```

Hier wird bei Zuweisungen allein von dem Gleichheitszeichen (=) Gebrauch gemacht. In diesem Fall ist das Objekt obj vom Typ character (String-Objekt (Textzeichenfolge)).

Die verfügbaren Objekte in der aktuellen R-Umgebung werden mit dem Befehl ls() in der R-Konsole angezeigt:

```
>ls()
 [1] "obj"
```

Das aktuelle Arbeitsverzeichnis (working directory) kann mit dem getwd()-Befehl überprüft werden:

```
>getwd()
[1] "C:/Users/Kosfeld/Documents/Ökonometrie/Ökonometrie mit R/Daten"
```

Es kann über das R-Menü File – Change Dir… oder mit dem setwd()-Befehl geändert werden:

```
> setwd("C:/Users/Kosfeld/Documents/Ökonometrie/Ökonometrie mit R/Daten").
```

Verlassen werden kann das Programm R durch den quit-Befehl q() oder durch Anklicken des Symbols x des Programmfensters RGui (64-bit).

1.2 Rechnen mit Zahlen und Funktionen

Das Programm R kann als Taschenrechner benutzt werden, indem Rechenoperationen wie Addition (+), Subtraktionen (-), Multiplikationen (*), Divisionen (/)und Potenzierung (^) direkt ausgeführt werden:

```
> 3+4
[1] 7

> 7-12
[1] -5

> 5*6
[1] 30

> 9/4
[1] 2.25
```

Die Exponentiation mit einer beliebigen Basis erfolgt mit dem Zeichen ^,

```
> 2^4
[1] 16,
```

während sie zur Basis e (Eulersche Zahl: e = 2.718282) mit der exp-Funktion ausgeführt wird:

```
>exp(3)
[1] 20.08554
```

Die Quadratwurzel wird mit der sqrt-Funktion gezogen:

```
>sqrt(64)
[1] 8
```

Der Logarithmus zur Basis e wird mit der log-Funktion gebildet,

```
> log(8)
[1] 2.079442
```

und der Logarithmus zur Basis 10 mit der log10-Funktion:

```
> log10(1000)
[1] 3
```

Neben den arithmetischen Funktionen exp, sqrt und log führen wir an dieser Stelle die trigonometrischen Funktionen sin, cos und tan sowie ihre Umkehrfunktionen asin, acos und atan ein. Als Argument der Sinusfunktion muss der Winkel in Bogenmaß z.B. 2 (= 2*180/pi = 114.5916°) angegeben werden:

```
> sin(2)
[1] 0.9092974
```

Die Kreiskonstante π ist in R unter pi gespeichert:

```
>pi
[1] 3.141593
```

Mit der Konstanten pi lässt sich der mit der Arcussinus-Funktion asin in Bogenmaß ermittelte Winkel,

```
>asin(0.9)
[1] 1.119770,
```

in einen Winkel in Grad transformieren:

```
>asin(0.9)*180/pi
[1] 64.15807
```

Mit der Funktion round werden Zahlen auf ganze Stellen,

```
>round(1.8786)
[1] 2
```

oder auf die durch das Argument digit=2 angegebenen Dezimalstellen gerundet:

```
>round(1.8786, digit=2)
[1] 1.88
```

Übungsaufgaben

1.2.1 Berechnen Sie den Ausdruck $4 \cdot (6\frac{1}{2} - \frac{1}{2}) / 2$!

1.2.2 Berechnen Sie den Ausdruck $(8\frac{1}{3} + 2\frac{1}{2})^2 / \sqrt{2}$ und runden Sie ihn auf drei Dezimalstellen!

1.2.3 Berechnen Sie den Ausdruck $\log_{10} 1000 + \sqrt[3]{27}$!

1.2.4 Berechnen Sie den Ausdruck $x^2 = \log_2 16$!

1.2.5 Berechnen Sie den Ausdruck $\ln x = \sqrt{8}$!

1.3 Variablen, Vektoren und Matrizen

In den vorherigen Beispielen geht nach Auswertung das Ergebnis nach seiner Ausgabe am Bildschirm verloren. Um die Ergebnisse der Rechenoperationen in einer Sitzung verfügbar zu haben, müssen sie unter einem Variablennamen gespeichert werden.

Das Ergebnis wird nicht mit der Zuweisung zu einer Variablen ausgegeben, sondern erst mit einem weiteren Befehl durch Nennung des Variablennamens:

```
> a = log10(1000)
> a
[1] 3
```

Das Objekt a ist hier der Spezialfall eines numerischen Vektors mit nur einem Element. Ein Vektor ist in R die einfachste Datenstruktur oder der einfachste Objekttyp.

Allgemein lässt sich ein Vektor durch Zuweisung von Zahlen mit Hilfe der Funktion c() definieren:

```
> x = c(3, 6, 7, 1, 9, 4)
> x
[1] 3 6 7 1 9 4
```

Ein Vektor mit einer Folge aufeinanderfolgender natürlicher Zahlen wird durch

```
> y = 1:6
> y
[1] 1 2 3 4 5 6
```

kreiert. Eine allgemeinere Sequenz von Zahlen kann mit dem seq()-Befehl durch Angabe des Inkrements z.B. by = 0.5

```
> a = seq(1, 4, by = 0.5)
```

```
>a
[1] 1.0 1.5 2.0 2.5 3.0 3.5 4.0
```

oder der Längez.B. length = 7

```
> b = seq(1, 4, length=7)
> b
[1] 1.0 1.5 2.0 2.5 3.0 3.5 4.0
```

einem Vektor zugewiesen werden. Das erste Argument gibt dabei jeweils den Startwert (from = 1), das zweite den Endwert (to = 4) an.

Unter Verwendung der einfachen Operatoren +, -, *, / und ^ werden Operationen mit Vektoren elementweise ausgeführt. In den folgenden Beispielen werden die Vektoren x,

```
> x
[1] 3 6 7 1 9 4,
```

und y,

```
> y
[1] 1 2 3 4 5 6,
```

verwendet:

```
> d = x + 3
> d
[1]  6  9 10  4 12  7

> e = 2*x
> e
[1]  6 12 14  2 18  8

> f = x + y
> f
[1]  4  8 10  5 14 10

> g = x * y
> g
[1]  3 12 21  4 45 24
```

Um das Skalarprodukt der beiden Vektoren x und y zu bilden, muss der Operator %*% verwendet werden:

```
> t(x)%*%y
    [,1]
[1,]  109
```

Hierbei wird der Zeilenvektor t(x) mit dem Spaltenvektor y multipliziert. Die Transponation eines Spaltenvektors in einen Zeilenvektor oder eines Zeilenvektors in einen Spaltenvektor erfolgt mit der Funktion t().

Das äußere Produkt der beiden Vektoren x und y ergibt sich durch Multiplikation des Spaltenvektors x mit dem Zeilenvektor t(y) unter Verwendung des Multiplikationsoperators%*%:

```
> x%*%t(y)
  [,1] [,2] [,3] [,4] [,5] [,6]
[1,]    3    6    9   12   15   18
[2,]    6   12   18   24   30   36
[3,]    7   14   21   28   35   42
[4,]    1    2    3    4    5    6
[5,]    9   18   27   36   45   54
[6,]    4    8   12   16   20   24
```

Das Ergebnis ist in diesem Fall eine 6x6-Matrix.

Aus der Sicht der Statistik lassen sich in einem Vektor die Werte eines Merkmals (Variablen) darstellen, die an n Untersuchungseinheiten erhoben worden sind. Allgemein werden bei einer Erhebung jedoch die Daten von m Merkmalen erfasst. Diese Daten lassen sich zweckmäßig in einer nxm-Matrix anordnen.

Eine Matrix lässt sich z.B aus den Werten eines Vektors durch Angabe der Zeilen- und Spaltenzahl (nrow und ncol) mit der Funktion matrix erzeugen:

```
> A = matrix(c(3,5,6,2,8,4), nrow=3, ncol=2)
> A
   [,1] [,2]
[1,]    3    2
[2,]    5    8
[3,]    6    4
```

Das erste Argument data ist hierbei mit der Funktion c() spezifiziert. Standardmäßig wird die Matrix spaltenweise ausgefüllt. Sollen die Werte zeilenweise eingetragen werden, muss das Argument byrow gleich TRUE gesetzt werden:

```
> B = matrix(1:6, nrow=3, ncol=2, byrow=TRUE)
> B
   [,1] [,2]
[1,]    1    2
[2,]    3    4
[3,]    5    6
```

Mit colnames() können den Spalten einer Matrix Namen zugeordnet werden:

```
>colnames(B) = c("b1", "b2")
```

```
> B
  b1 b2
[1,]  1  2
[2,]  3  4
[3,]  5  6
```

Entsprechend lassen sich mit rownames() den Zeilen einer Matrix Namen zuordnen. Beides gleichzeitig wird mit dem Befehl dimnames() bewerkstelligen:

```
>dimnames(B) = list(c("B1", "B2", "B3"),c("b1", "b2"))
> B
  b1 b2
B1  1  2
B2  3  4
B3  5  6
```

Die Dimensionen von Matrizen können mit der dim-Funktion abgefragt werden:

```
>dim(B)
[1] 3 2
```

Der erste Wert gibt die Zeilenzahl, der zweite die Spaltenzahl an.

Für manche Berechnungen ist es zweckmäßig, die Zeilen- und Spalten einer Matrix als Vektoren zu definieren. Enthalten die beiden Spalten der Matrix B die Werte zweier Variablen, können diese extrahiert und z.B. den Variablenvektoren b1 und b2zugeordnet werden:

```
> b1 = B[,1]
> b1
B1 B2 B3
 1  3  5
> b2 = B[,2]
> b2
B1 B2 B3
 2  4  6.
```

Mit den Funktionen rbind und cbind lassen sich umgekehrt Vektoren durch zeilen- bzw. spaltenweises Zusammenfügen zu einer Matrix verbinden. So erhält man durch eine zeilenweise Verknüpfung der Vektoren b1 und b2 die Matrix B1,

```
> B1 = rbind(b1,b2)
> B1
  B1 B2 B3
b1  1  3  5
b2  2  4  6,
```

und durch spaltenweise Verknüpfung die Matrix B2:

```
> B2 = cbind(b1,b2)
> B2
  b1 b2
B1  1  2
B2  3  4
B3  5  6.
```

Übungsaufgaben

1.3.1 Weisen Sie der Variablen x den Wert $4^3 - 6 \cdot 3$ und der Variablen y den Aus-druck $3 \cdot x + 5$ zu. Berechnen Sie den Ausdruck $z = \left(\frac{1}{3}x^2 - x \cdot \sqrt{y}\right)$!

1.3.2 Berechnen Sie $s = \frac{\sqrt[3]{(y^3 - 3 \cdot x + 4)}}{y - 2}$ unter Verwendung der in Aufgabe 1.3.1 angegebenen Zuweisungen für x und y!

1.3.3 Definieren Sie einen Vektor x aus den Werten 3, 5, 8, 2, 4, 1!

1.3.4 Erzeugen Sie einen Vektor y mit den Komponenten 4, 5, 6, 7, 8, 9 auf drei unterschiedliche Arten!

1.3.5 Addieren und multiplizieren Sie die in den Aufgaben 1.3.3 und 1.3.4 gebildeten Vektoren x und y elementweise und speichern Sie die Ergebnisse unter den Objektnamen s bzw. m!

1.3.6 Bilden Sie das Skalarprodukt der beiden Vektoren x und y (s. Aufg. 1.3.3 und 1.3.4)!

1.3.7 Bilden Sie aus den beiden Vektoren x und y (s. Aufg. 1.3.3 und 1.3.4) 2x3-Matrizen A und B!

1.3.8 Multiplizieren Sie die Matrix A mit der transponierten Matrix B (s. Aufg. 1.3.7) und speichern Sie das Matrixprodukt unter dem Objektnamen AB!

1.3.9 Füllen Sie eine 3x3-Matrix C zeilenweise mit den Werten 1 bis 9!

1.3.10 Weisen Sie den Spalten der in Aufg. 1.3.9 gebildeten Matrix C die Namen c1, c2 und c3 zu!

1.3.11 Verbinden Sie die in den Aufgaben 1.3.3 und 1.3.4 gebildeten Vektoren x und y zu einer 6x2-Matrix D!

1.4 Zeitreihenobjekte und -diagramm

In der Ökonometrie werden Zeitreihen-, Querschnitts- und Paneldaten verwendet. Während sich mikroökonomische Analysen in der Regel auf einen Querschnitt oder ein Panel von Unternehmen, Haushalten oder Erwerbstätige beziehen, liegen makroökonomische Daten wie z.B. das Bruttoinlandsprodukt, der Private Konsum und die Investitionen häufig als Zeitreihen vor. Die ökonometrischen Methoden werden hier vor allem anhand von Zeitreihendaten illustriert.

Zeitreihen können mit der Funktion ts durch Angabe eines Vektors oder einer Matrix mit quantitativen Daten, der Start- und Endperiode sowie der Periodizität (frequency) als univariate – oder allgemeiner multivariate – Zeitreihen definiert werden. Beispiel-haft seien die Jahresdaten des Verbraucherpreisindex (VPI) für den Zeitraum 2000 bis 2011 als Werte der Zeitreihe vpi definiert:

```
>vpi = ts(c(92.7, 94.5, 95.9, 96.9, 98.5, 100, 101.6, 103.9, 106.6, 107, 108.2, 110.7), start=2000,
end=2011,frequency=1)
```

Die Anwendung der Funktion str auf das Objekt vpi weist seine interne Struktur aus,

```
>str(vpi)
 Time-Series [1:12] from 2000 to 2011: 92.7 94.5 95.9 96.9 98.5 ...
```

während mit der Funktion class die Objektklasse angegeben wird:

```
>class(vpi)
[1] "ts".
```

Definition und Inhalt des ts-Objekts vpi werden durch Eingabe des Objektnamens nach Bestätigung mit der Return-Taste aufgelistet:

```
>vpi
Time Series:
Start = 2000
End = 2011
Frequency = 1
 [1]  92.7  94.5  95.9  96.9  98.5 100.0 101.6 103.9 106.6 107.0 108.2 110.7
```

Eines der beiden Argumente end oder frequency ist entbehrlich. Letzteres gibt die Anzahl der Zeitreihenwerte pro Jahr an:

- Jahresdaten: frequency=1
- Halbjahresdaten: frequency=2
- Quartalsdaten: frequency=4
- Monatsdaten:frequency=12

Anstelle der Periodizität (frequency) kann auch die äquidistante Differenz (deltat) Δx_t zwischen zwei aufeinanderfolgenden Zeitreihenwerten x_t und x_{t-1}als Argument der ts-Funktion angegeben werden.

Eine deskriptive Beschreibung einer Zeitreihe erhält man mit der summary-Funktion:

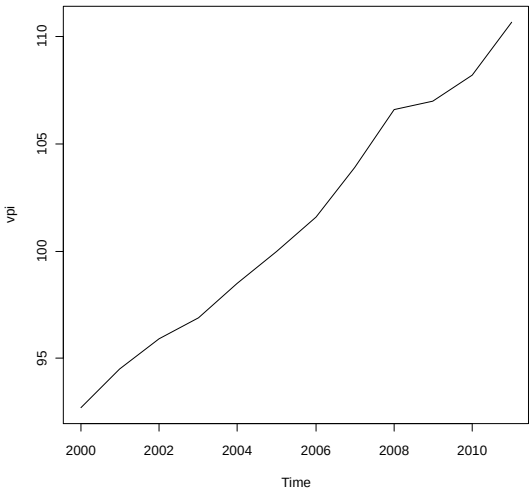
```
>summary(vpi)
Min.1st Qu.Median    Mean 3rd Qu.Max.
92.70   96.65  100.80  101.40  106.70  110.70
```

Sie besteht aus der Angabe des kleinsten und größten Zeitreihenwerts, der Quartile (2. Quartil = Median) und des arithmetischen Mittels.

Ein Zeitreihendiagramm lässt sich bei einer einzelnen Zeitreihe mit der plot-Funktion erstellen.[1] Hierbei wird die Zeit t auf der Abzisse abgetragen, während die Ordinate die Werte der Zeitreihe x_t wiedergibt. Wird die Funktion plot auf ein Zeitreihenobjekt angewandt, hat R Kenntnis über die Zeit, so dass nur noch die Zeitreihe selbst genannt werden muss. Im einfachsten Fall erhält man ein Zeitreihendiagramm z.B. für den Verbraucherpreisindex, indem man allein den Namen des Zeitreihenobjekts (vpi) in der plot-Funktion angibt:

```
>plot(vpi)
```

Abbildung 1.1: Elementares Zeitreihendiagramm des Verbraucherpreisindex (VPI)



Das in Abbildung 1.1 wiedergegebene elementare Zeitreihendiagramm lässt sich durch die Aufnahme optionaler Argumente und Verwendung von Low-level-Befehlen verbessern, die díe aktive Grafik verändern:

```
>plot(vpi, main="Entwicklung des Verbraucherpreisindex (VPI)", xlab="", ylab="VPI",
lwd=3,cex.main=1.25, font.main=1)
>axis(side=1, at=c(2001,2003,2005,2007,2009,2011))
>text(2006, 103, "VPI", adj=0.4, cex=1.2)
>box(which="figure")
```

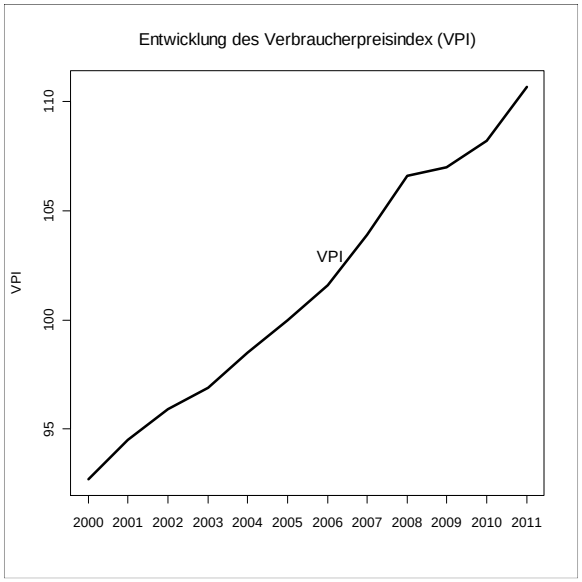
In der plot-Funktion (High-level-Befehl) legt das Argument main die Überschrift des Zeitreihendiagramms fest. Mit ylab wird die Beschriftung der Ordinate spezifiziert; xlab="" unterdrückt die Beschriftung der Abzisse mit „Time“. Während der Parameter lwd die Linienstärke des Zeitreihenpolygons (Kurvenzugs) kontrolliert, bestimmt der Parameter cex.main die Größe der Überschrift (Standardeinstellungen: lwd=1 und cex.main=1).Der Parameter font gibt den Schrifttyp (1 = einfach, 2 = fett. 3 = kursiv. 4 = fett kursiv) an.

[1] Die Verwendung der ts.plot-Funktion bringt erst bei der grafischen Darstellung multipler Zeitreihen Vorteile mit sich.

Wie aus Abbildung 1.1 hervorgeht, erstellt das Programm R für die Zeitreihe vpi automatisch die Abzisse mit Angabe der geraden Jahreszahlen. Die Abzisse mit allen Jahreszahlen werden der aktiven Grafik mit dem axis-Befehl durch das Argument at hinzugefügt. Das Argument side=1 gibt dabei an, dass die Achse unten angefügt werden soll.

Mit dem text-Befehl erfolgt eine Beschriftung des Zeitreihenpolygons mit VPI. Die Adjustierung des Textes wird über den Parameter adj(0: Ausrichtung links, 0.5: Ausrichtung mittig, 1: Ausrichtung rechts) gesteuert. Die Schriftgröße wird hierbei mit dem Parameter cex auf 1.2 festgelegt. Der box-Befehl mit dem Argument which="figure" oder which="outer" zieht einen äußeren Rahmen um die Grafik.

Abbildung 1.2: Zeitreihendiagramm des Verbraucherpreisindex (VPI)



Wenn Vektoren und Matrizen zeitbezogene quantitative Daten enthalten, können sie mit dem Befehl as.ts oder ts in univariate bzw. multivariate Zeitreihen umgewandelt werden. So wird der Vektor vpi, der die Beobachtungswerte des Verbraucherpreisindex enthält,

```
>vpi = c(92.7, 94.5, 95.9, 96.9, 98.5, 100, 101.6, 103.9, 106.6, 107, 108.2, 110.7)
```

durch den Befehl

```
>vpi = as.ts(vpi)  oder   >vpi = ts(vpi)
```

in ein Zeitreihenobjekt umgewandelt, dessen Werte auf die Zeiteinheiten 1, 2, …, 12 bezogen werden:

```
>vpi
Time Series:
Start = 1
End = 12
Frequency = 1
 [1]  92.7  94.5  95.9  96.9  98.5 100.0 101.6 103.9 106.6 107.0 108.2 110.7
```

Bei Spezifizierung der Argumente start, end und frequency in dem as.ts-Befehl wie in der ts-Funktion lässt sich die Zeitreihe auf den Beobachtungszeitraum 2001 – 2009 datieren. Diese Datierung kann auch im Nachhinein mit dem ts-Befehl vorgenommen werden:

```
>vpi =ts(vpi, start=2000, frequency=1)
>vpi
Time Series:
Start = 2000
End = 2011
Frequency = 1
 [1]  92.7  94.5  95.9  96.9  98.5 100.0 101.6 103.9 106.6 107.0 108.2 110.7
```

Ein Ausschnitt einer Zeitreihe wird mit der Funktion window ausgewählt:

```
>vpi0308 = window(vpi, start=2003, end=2008)
> vpi0308
Time Series:
Start = 2003
End = 2008
Frequency = 1
[1]  96.9  98.5 100.0 101.6 103.9 106.6
```

Hier wird der Verbraucherpreisindex für den Ausschnitt 2003-2008 des Beobachtungszeitraums in dem ts-Objekt vpi0308 gespeichert.

Übungsaufgaben

1.4.1 Erzeugen Sie einen Vektor GASVk aus den Werten 1, 0.8, 0.7, 0.9, 1.2, 1.1, 1.4 des Erdgasverbrauchs!

1.4.2 Definieren Sie den in Aufgabe 1.4.1 erstellten Vektor GASVk als Zeitreihe und erstellen Sie ein elementares Zeitreihendiagramm!

1.4.3 Datieren Sie die in Aufgabe 1.4.2 definierte Zeitreihe GASVk auf den Zeitraum 2005 bis 2011!

1.4.4 Plotten Sie die in Aufgabe 1.4.3 datierte Zeitreihe GASVk mit einer Linienstärke von 3.5. Beschriften Sie die Ordinate und den Kurvenzug mit GASVk. Überschreiben Sie das Zeitreihendiagramm mit „Gasverbrauch 2005-2011“ bei einer Schriftgröße von 1.15.

1.5 Einlesen von Zeitreihen aus Datendateien

Kurze Zeitreihen wie z.B. die Zeitreihe des Verbraucherpreisindex vpi, die hier zur Illustration der plot-Funktion verwendet worden ist, lassen sich in R unaufwändig über die Tastatur eingeben. Lange oder multiple Zeitreihen werden dagegen üblicher-weise von einer Datendatei eingelesen. Sind die Daten in einer .xls- oder .xlsx-Datei gespeichert, können sie mit einem Tabellenkalkulationsprogramm wie z.B. Excel in eine csv-Textdatei (Comma Separated Values) umgewandelt werden. Sie lassen sich dann mit dem read.table- oder read.csv-Befehl in R einlesen.

Die csv-Datei Konsumfkt(Daten) enthält multiple Zeitreihen zur Schätzung der makroökonomischen Konsumfunktion für Deutschland für den Zeitraum 1991 – 2011. Die Jahresdaten können gleichwertig mit read.table- oder read.csv-Befehl in R eingelesen und als R-Objekt gespeichert werden:

```
>Konsumfkt.data = read.csv(file="Konsumfkt(Daten).csv", dec=".", sep=";", header=TRUE)
>Konsumfkt.data
   Jahr   PIKPrivKonsVerfEinkPrivKonsrVerfEinkr
1  1991  79.42   882.55  1005.13   1111.24   1265.59
2  1992  82.84   949.79  1077.97   1146.54   1301.27
3  1993  85.86   989.11  1119.76   1152.00   1304.17
4  1994  88.03  1032.75  1159.39   1173.18   1317.04
5  1995  89.27  1066.47  1190.61   1194.66   1333.72
6  1996  90.11  1088.64  1212.71   1208.12   1345.81
7  1997  91.26  1110.82  1231.47   1217.20   1349.41
8  1998  91.75  1130.14  1251.21   1231.76   1363.72
9  1999  92.11  1161.86  1278.66   1261.38   1388.19
10 2000  92.84  1195.04  1300.70   1287.20   1401.01
11 2001  94.56  1233.43  1347.78   1304.39   1425.32
12 2002  95.68  1240.58  1363.26   1296.59   1424.81
13 2003  97.24  1264.51  1394.62   1300.40   1434.20
14 2004  98.36  1283.61  1419.00   1305.01   1442.66
15 2005 100.00  1306.98  1448.13   1306.98   1448.13
16 2006 100.99  1339.54  1480.52   1326.41   1466.01
17 2007 102.49  1356.73  1502.37   1323.77   1465.87
18 2008 104.19  1387.70  1542.58   1331.89   1480.55
19 2009 104.25  1387.43  1531.09   1330.87   1468.67
20 2010 106.28  1423.02  1575.85   1338.93   1482.73
21 2011 108.51  1474.42  1626.67   1358.79   1499.10
```

Neben dem Dateinamen (file) sind in dem read.csv-Befehl einige optionale Parameter angegeben. Das Argument dec gibt an, dass den Punkt in der Datendatei Konsumfkt(Daten).csv als Dezimalzeichen verwendet worden ist. Die Werte selbst sind – nicht sichtbar – in dieser Datei durch Semikola voneinander getrennt (sep=“;“). Der logische Wert TRUE des Arguments headerzeigt an, dass die erste Zeile der Datei Konsumfkt(Daten).csv die Variablennamen enthält.

Der Inhalt der Datei Konsumfkt(Daten).csv wird dem R-Objekt Konsumfkt.data zugewiesen. Die erste Spalte enthält als Zeitangaben die Jahre der Beobachtungswerte unter dem Variablennamen Jahr. In der zweiten Spalte steht der Preisindex des Privaten Konsums (PIK). Es folgen die nominalen Werte des Privaten Konsums (PrivKons) und verfügbaren Einkommens (VerfEink) in den Spalten 3 und 4. Schließlich sind die realen Werte des Privaten Konsums (PrivKonsr) und verfügbaren Einkommens (VerfEinkr) in den Spalten 5 und 6 gespeichert.

Mit dem Befehl

```
>class(Konsumfkt.data)
[1] "data.frame"
```

wird der Typ des Objekts Konsumfkt.data ausgegeben. Konsumfkt.data ist noch kein Zeitreihenobjekt, sondern ein Objekt vom Typ data.frame, d.h. ein Datensatz, der wie ein Matrixobjekt aus Zeilen und Spalten besteht. In den Zeilen stehen die Beobachtungswerte der statistischen Einheiten, in den Spalten die Variablen. Im Unterschied zu einem matrix-Objekt kann ein data.frame-Objekt jedoch auch nicht-nummerische Werte z.B. für qualitative Variablen enthalten.

Der str-Befehl

```
>str(Konsumfkt.data)
'data.frame':   21 obs. of  6 variables:
$ Jahr    : int  1991 1992 1993 1994 1995 1996 1997 1998 1999 2000 ...
 $ PIK     : num  79.4 82.8 85.9 88 89.3 ...
 $ PrivKons : num  883 950 989 1033 1066 ...
 $ VerfEink : num  1005 1078 1120 1159 1191 ...
 $ PrivKonsr: num  1111 1147 1152 1173 1195 ...
 $ VerfEinkr: num  1266 1301 1304 1317 1334 ...
```

gibt Aufschluss über die Struktur des data.frame-Objekts Konsumfkt.data.

Da die makroökonomische Konsumfunktion unter Verwendung der realen Werte des Privaten Konsums und verfügbaren Einkommens in R mit Hilfe der Regressionsanalyse geschätzt werden soll, definieren wir die fünfte Spalte des data.frame-Objekts Konsumfkt.data als Zeitreihe PKr,

```
>PKr = ts(Konsumfkt.data$PrivKonsr, start=1991, frequency=1)
>PKr
Time Series:
Start = 1991
End = 2011
Frequency = 1
 [1] 1111.24 1146.54 1152.00 1173.18 1194.66 1208.12 1217.20 1231.76 1261.38 1287.20 1304.39 1296.59
1300.40 1305.01 1306.98 1326.41 1323.77 1331.89 1330.87 1338.93 1358.79,
```

und die sechste Spalte als Zeitreihe VEr:

```
>VEr = ts(Konsumfkt.data$VerfEinkr, start=1991, frequency=1)
>VEr
Time Series:
Start = 1991
End = 2011
Frequency = 1
 [1] 1265.59 1301.27 1304.17 1317.04 1333.72 1345.81 1349.41 1363.72 1388.19 1401.01 1425.32 1424.81
1434.20 1442.66 1448.13 1466.01 1465.87 1480.55 1468.67 1482.73 1499.10.
```

Das Argument start der Funktion ts benennt das Anfangsjahr 1991 der Zeitreihen. Der Wert 1 des Arguments frequency definiert Jahresdaten.

Mit den Anweisungen

```
>class(PKr)
[1] "ts"
```

und

```
>class(VEr)
[1] "ts"
```

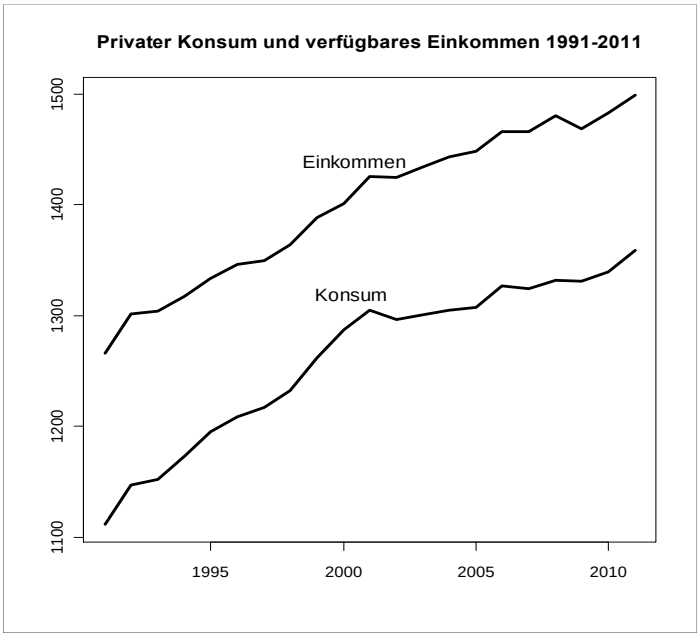
erhält man die Bestätigung, dass die Objekte PKr und VEr vom Typ ts, d.h. Zeitreihen-objekte, sind.

Multiple Zeitreihen lassen sich in R am besten mit der ts.plot-Funktion darstellen. Das Diagramm der Zeitreihen des realen Privaten Konsums(PKr) und realen verfügbaren Einkommens (VEr) ist für den Zeitraum 1991 - 2011 in Abbildung 1.3 wiedergegeben:

```
>ts.plot(PKr, VEr, main="Privater Konsum und verfügbares Einkommen 1991-2011",gpars=list(xlab="",
lwd=3))
>text(2000, 1440, "Einkommen", adj=0.4, cex=1.2)
>text(2000, 1320, "Konsum", adj=0.4, cex=1.2)
> box(which="figure")
```

Um eine Beschriftung der beiden Zeitreihen vornehmen zu können, sind die Grafik-parameter in dem Argument gpars der ts.plot-Funktion gelistet.

Abbildung 1.3: Zeitreihendiagramm des Privaten Konsums und verfügbaren Einkommens 1991 - 2011



Die Zeitreihe des Privaten Konsums weist bis zum Jahr 2001 eine größere Steigung auf als die Zeitreihe des verfügbaren Einkommens. In der ersten Rezession nach der Jahrtausendwende bricht der reale Konsum jedoch stärker ein als das Realeinkommen. Gleichzeitig flacht die Konsumkurve flacht merklich ab. Am aktuellen Rand steigen beide Aggregate nach der Rezession 2009 wieder stärker an.

Übungsaufgaben

1.5.1 Erstellen Sie eine CSV-Datei Energie(lang) aus den Daten für Energienachfragemodelle (Dregery/Kosfeld/Eckey, 2014, Ökonometrie, Tabelle 7)!

1.5.2 Lesen Sie die CSV-Datei Energie(lang) als `data.frame`-Objekt `Energie1` in R ein!

1.5.3 Definieren Sie die Variablen `GASPR` und `FERNWPR` des `data.frame`-Objekts `Energie1` als Zeitreihen!

1.5.4 Plotten Sie die in Aufgabe 1.5.3 erstellte Zeitreihe `GASPR`mit dem Titel „Relativer Gaspreis 1980 – 1995“!

1.5.5 Erstellen Sie ein Zeitreihendiagramm mit den Zeitreihen `GASPR` und `FERNWPR` mit dem Titel „Relative Gas- und Fernwärmepreise 1980-1995“!

1.5.6 Definieren Sie die Variablen GASV und VEINKR des `data.frame`-Objekts Energie als Zeitreihen! Plotten Sie die beiden Zeitreihen separat mit den Überschriften „Erdgasverbrauch 1980-1995“ bzw. „Reales verfügbares Einkommen 1980-1995“!

2. Regressionsmodell und Kleinst-Quadrate-Schätzung

2.1 Einfache Regressionsanalyse und lm-Funktion

In dem einfachen Regressionsmodell wird eine abhängige (endogene) Variable y durch eine unabhängige (exogene) Variable x erklärt. Im Falle einer lineare Beziehung zwischen den beiden Variablen lautet das Regressionsmodell

(2.1) $y_t = \beta_1 + \beta_2 \cdot x_t + u_t$, t=1.2….n.

Der Index t kann ein Zeit- oder Querschnittsindex sein. Letzterer kann sich z.B. auf Unternehmen, Verbraucher oder Länder beziehen. Für die Störvariable u werden als Standardannahmen ein Erwartungswert von null, eine konstante Varianz (Homoskedastizität) und fehlende Autokorrelation angenommen.

Bei Gültigkeit der Standardannahmen können die Regressionskoeffizienten β_1 und β_2 mit der gewöhnlichen Methode der kleinsten Quadrate (ordinary least-squares (OLS)) geschätzt werden. β_1 ist das absolute Glied und β_2 das Steigungsmaß.

Zur Kleinst-Quadrate-Schätzung eines Regressionsmodells ist in R die lm-Funktion verfügbar. Beispielhaft wird hier die einfache Regressionsanalyse anhand der makroökonomischen Konsumfunktion für Deutschland aufgezeigt. Für die Keynessche Konsumfunktion werden die Zeitreihen des Privaten Konsums (PKr) und des verfügbaren Einkommens (VEr) benötigt, die aus der im Abschnitt 1.5 eingelesenen csv-Datei Konsumfkt(Daten) verfügbar sind. Unter Außerachtlassung der Anpassungsphase nach der Vereinigung der beiden deutschen Staaten wird die Keynessche Konsumfunktion für den Zeitraum 1997 – 2011 unter Verwendung der lm-Funktion geschätzt:

```
> PKr9711 = ts(PKr[-c(1:6)], start=1997, freq=1)
```

oder

```
> PKr9711 = window(PKr, start=1997, end=2011)
> PKr9711
Time Series:
Start = 1997
End = 2011
Frequency = 1
 [1] 1217.20 1231.76 1261.38 1287.20 1304.39 1296.59 1300.40 1305.01 1306.98 1326.41 1323.77 1331.89
1330.87 1338.93 1358.79
```

```
> VEr9711 = ts(VEr[-c(1:6)], start=1997, freq=1)
```

oder

```
> VEr9711 = window(VEr, start=1997, end=2011)
> VEr9711
Time Series:
Start = 1997
```

```
End = 2011
Frequency = 1
 [1] 1349.41 1363.72 1388.19 1401.01 1425.32 1424.81 1434.20 1442.66 1448.13 1466.01 1465.87 1480.55
1468.67 1482.73 1499.10
>konsfkt.lm = lm(PKr9711~VEr9711).
```

Innerhalb der runden Klammer der lm-Funktion erfolgt die Modellspezifikation. Vor der Tilde (~) steht die zu erklärende Variable PKr9711, nach der Tilde ist die erklärende Variable VEr9711 benannt. Das absolute Glied wird standardmäßig mitgeschätzt. Das Schätzergebnis ist in dem lm-Objekt konsfkt.lm gespeichert:

```
>class(konsfkt.lm)
[1] "lm"
```

Eine Kurzfassung des Schätzergebnisses erhält man durch Angabe des Namens des lm-Objekts,

```
>konsfkt.lm

Call:
lm(formula = PKr9711 ~ VEr9711)

Coefficients:
(Intercept)     VEr9711
   57.9858      0.8659  ,
```

wodurch allein die Spezifikation des Regressionsmodells und die OLS-Schätzer der Regressionskoeffizienten ausgegeben werden. Der autonome Konsum nimmt den Wert 57.9858 an. Eine marginale Konsumquote von 0.8659 bedeutet, dass im Stützzeitraum 1997 – 2011 im Schnitt 86,6% des zusätzlichen Einkommens konsumiert worden sind.

Einen detaillierteren Output erhält man unter Verwendung der summary-Funktion:

```
>summary(konsfkt.lm)

Call:
lm(formula = PKr9711 ~ VEr9711)

Residuals:
   Min     1Q  Median     3Q    Max
-9.2379 -4.2249 -0.9917  2.0491 16.0817

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  57.98585   63.12937   0.919    0.375
VEr9711       0.86590    0.04394  19.706   4.58e-11 ***
---
```

```
Signif.codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.31 on 13 degrees of freedom
Multiple R-squared: 0.9676,    Adjusted R-squared: 0.9651
F-statistic: 388.3 on 1 and 13 DF,  p-value: 4.579e-11.
```

Er stellt zusätzlich einen Signifikanztest der geschätzten Regressionskoeffizienten bereit. Anhand der p-Werte ist erkennbar, dass die marginale Konsumquote hochsignifikant ist (p<0,01), während der autonome Konsum bei einem p-Wert von 0.375 nicht signifikant von null abweicht.

Der Determinationskoeffizient beträgt 0.9676, so dass das OLS-geschätzte ökonometrische Modell 96,76% der Varianz des Privaten Verbrauchs erklärt: Der F-Test ist bei einem p-Wert, der kleiner als 0,01 ist, hochsignifikant. Die vorgenommene Spezifikation der makroökonomischen Konsumfunktion lässt sich daher als substanziell betrachten.

Zusätzlich enthält der Output noch einige Kenngrößen der Residuen. In einer Fünf-Zahlen-Darstellung werden der Median, das erste und dritte Quartil sowie der minimale und maximale Wert der Residuen ausgewiesen. Der Residual standard error von 7.31 gibt die erwartungstreue Standardabweichung der Residuen wieder. Diese Kenngröße wird auch als Standardfehler der Regression (standard error of regression (SER)) bezeichnet.

Das lm-Objekt konsfkt.lm enthält neben den geschätzten Regressionskoeffizienten,

```
>konsfkt.coeff = konsfkt.lm$coeff
>konsfkt.coeff
 (Intercept) VEr9711
  57.9858465   0.8658985
> class(konsfkt.coeff)
[1] „numeric“,
```

diverse andere Komponenten. Für die Regressionsdiagnostik werden die Regressionswerte benötigt, die als Vektor unter dem Namen fitted.values verfügbar sind:

```
> PKr9711f = konsfkt.lm$fitted.values
```

oder

```
 PKr9711f = fitted(konsfkt.lm)
> PKr9711f
1       2       3       4       5       6       7       8       9      10      11      12      13      14      15
1226.438 1238.829 1260.017 1271.118 1292.168 1291.727 1299.857 1307.183 1311.919 1327.402 1327.280
1339.992 1329.705 1341.879 1356.054
>class(PKr9711f)
[1] "numeric".
```

Der Vektor PKr9711f wird mit dem ts-Befehl als Zeitreihe definiert:

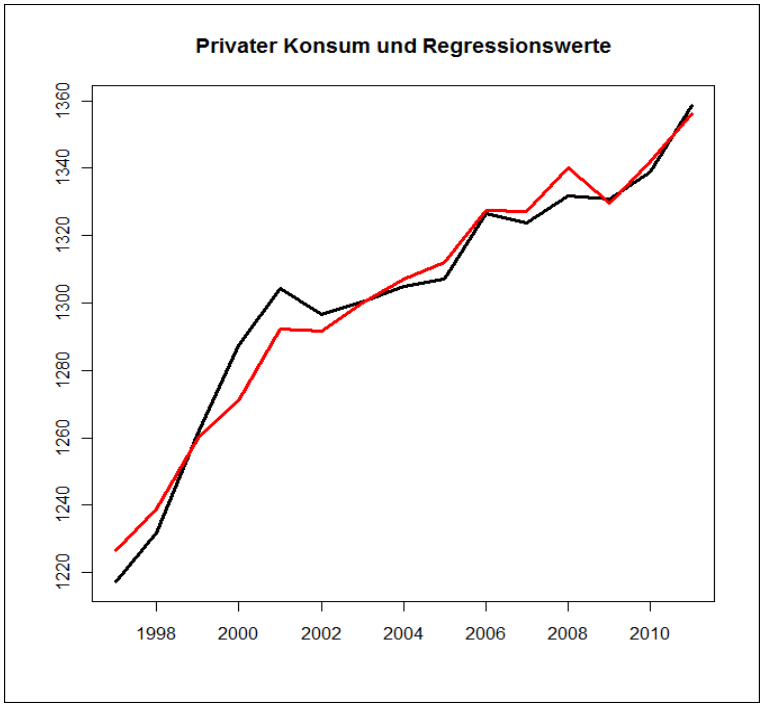
```
> PKr9711f = ts(PKr9711f, start=1997, freq=1)
> PKr9711f
Time Series:
Start = 1997
End = 2011
Frequency = 1
     1       2       3       4       5       6       7       8       9      10      11      12      13      14      15
1226.438 1238.829 1260.017 1271.118 1292.168 1291.727 1299.857 1307.183 1311.919 1327.402 1327.280
1339.992 1329.705 1341.879 1356.054.
```

Der grafische Vergleich der Beobachtungswerte des Privaten Konsums und der Regressionswerte wird mit der ts.plot-Funktion ermöglicht:

```
>ts.plot(PKr9711, PKr9711f, main="PrivaterKonsum und Regressionswerte",gpars=list(xlab="", lwd=3,
col=c("black","red")))
> box(which="figure").
```

Abbildung 2.1 zeigt zwar grundsätzlich den engen Verbund der beobachteten und angepassten Zeitreihe auf. In der Börsenhausse um die Jahrtausendwende und zu Beginn des anschließenden Crashs lag der Konsum jedoch deutlich über den aufgrund der Regression zu erwartenden Werten. Umgekehrt übten die Konsumenten zu Beginn der jüngsten Finanzkrise 2007 – 2008 eine deutliche Kaufzurückhaltung.

Abbildung 2.1: Privater Konsum und Regressionswerte 1997 - 2011



Für die grafische Diagnostik werden die in dem lm-Objekt konsfkt.lm verfügbaren Residuen der Konsumfunktion als Zeitreihe definiert:

```
> konsfkt.res = ts(konsfkt.lm$residuals, start=1997, freq=1)
```

oder

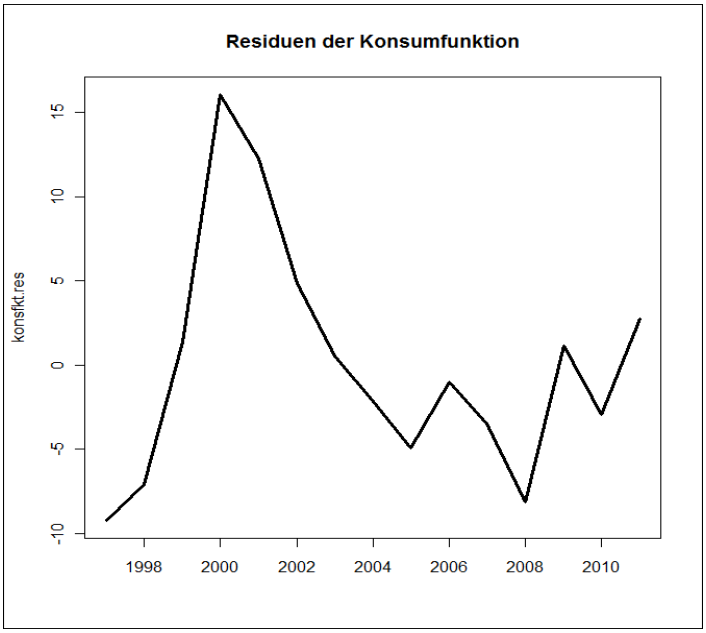
```
> konsfkt.res = ts(resid(konsfkt.lm), start=1997, freq=1)
> konsfkt.res
Time Series:
Start = 1997
End = 2011
Frequency = 1
1          2          3          4          5          6          7          8          9         10         11         12         13         14         15
-9.2379145 -7.0689218  1.3625424 16.0817239 12.2217319  4.8633401  0.5425534 -2.1729478 -4.9394125 -
0.9916773 -3.5104515 -8.1018412  1.1650328 -2.9494999  2.7357420.
```

Der Residuenplot wird mit der plot-Funktion erstellt:

```
>plot(konsfkt.res, main="Residuen der Konsumfunktion",xlab="", lwd=3)
> box(which="figure").
```

Aus Abbildung 2.2 gehen absolut große Residuen vor allem in der ersten Hälfte des Beobachtungszeitraums deutlich hervor. Hierbei wird das Jahr 2000, in dem das Residuum mit einem Wert von 16,08 außerhalb des 2·SER-Bandes (SER=7,31) liegt, als potenzieller Ausreißer identifiziert.

Abbildung 2.2: Residuen der Konsumfunktion



Übungsaufgaben

2.1.1 Regressieren Sie die Erdgasnachfrage (GASV) auf den Erdgaspreis (GASPR) (Daten s. data.frame-Objekt Energiel) unter Verwendung der lm-Funktion und speichern Sie das Ergebnis unter dem Namen energiefkt1.lm(=Energienachfragemodell 1)!

2.1.2 Geben Sie den verkürzten Regressionsoutput des lm-Objekts energiefkt1.lm in der R-Konsole aus und interpretieren Sie ihn!

2.1.3 Geben Sie den detaillierten Regressionsoutput des lm-Objekts energiefkt1.lm in der R-Konsole aus und interpretieren Sie ihn!

2.1.4 Definieren Sie die Regressionswerte des Energienachfragemodel 1 als Zeitreihe GASVf1 und stellen Sie sie zusammen mit der Zeitreihe GASV unter dem Titel „Erdgasverbrauch und Regressionswerte 1980-1995“ in einem Zeitreihendiagramm dar!

2.1.5 Definieren Sie die Zeitreihe der Residuen energiefkt1.res des Erdgasnachfrage-modells 1 und plotten Sie sie unter dem Titel „Residuen der Erdgasnachfragefunktion 1“!

2.2 Multiple Regressionsanalyse und lm-Funktion

Als Beispiel für die multiple Regressionsanalyse wird die Geldnachfragefunktion betrachtet. Die Daten werden aus der csv-Datei Geldnachfragefkt(Daten) eingelesen und in dem data.frame-Objekt Geldnachfragefkt.data gespeichert:

```
>Geldnachfragefkt.data = read.csv(file="Geldnachfragefkt(Daten).csv", dec=".", sep=";", header=TRUE)
>class(Geldnachfragefkt.data)
[1] "data.frame"
>Geldnachfragefkt.data
   Jahr   M1EU  M1GER  M1GERr EURIBOR Rbond   VPI   Irate   BIP  PIBIP    BIPr
1  1997 1626.9  465.0  440.45    3.33  5.10  90.0 0.019 1912.6  94.72 1811.61
2  1998 1785.4  513.4  489.17    3.54  4.50  90.8 0.009 1959.7  95.28 1867.20
3  1999 1972.0  557.6  532.34    2.97  4.32  91.3 0.006 2000.2  95.47 1909.59
4  2000 2084.6  574.6  544.89    4.39  5.41  92.7 0.015 2047.5  94.83 1941.64
5  2001 2279.0  601.6  576.87    4.26  4.79  94.5 0.019 2101.9  95.89 2015.51
6  2002 2499.4  583.5  567.57    3.32  4.66  95.8 0.014 2132.2  97.27 2073.99
7  2003 2727.1  631.9  621.35    2.33  3.74  96.8 0.010 2147.5  98.33 2111.64
8  2004 2948.9  655.4  651.40    2.11  3.68  98.5 0.018 2195.7  99.39 2182.31
9  2005 3482.1  725.8  725.80    2.19  3.14 100.0 0.015 2224.4 100.00 2224.40
10 2006 3758.6  760.0  762.36    3.08  3.76 101.6 0.016 2313.9 100.31 2321.07
11 2007 3901.2  789.7  805.10    4.28  4.30 103.9 0.023 2428.5 101.95 2475.86
12 2008 4035.7  832.8  855.54    4.63  4.19 106.6 0.026 2473.8 102.73 2541.33
13 2009 4556.2 1015.9 1055.93    1.23  3.22 107.0 0.004 2374.5 103.94 2468.06
14 2010 4747.0 1110.2 1160.83    0.81  2.52 108.2 0.011 2476.8 104.56 2589.74
```

Die Variablen sind wie folgt definiert:
M1EU: Geldmenge M1 (nominal) in der Europäischen Union,
M1GER: Geldmenge M1 (nominal) in Deutschland,
M1GERr: Geldmenge M1 (real) in Deutschland,
EURIBOR: Euro InterBankOffered Rate (Zinssatz für Termingelder in Euro im Interbankengeschäft),
Rbond: Umlaufrendite festverzinslicher Wertpapiere
VPI: Verbraucherpreisindex
Irate: Inflationsrate
BIP: Bruttoinlandsprodukt (nominal) in Deutschland,

PIBIP:Preisindex des Bruttoinlandsprodukts in Deutschland
BIPr: Bruttoinlandsprodukt (real) in Deutschland

Der EUROBOR hat den FIBOR (Frankfurt Interbank Offered Rate) als Referenzzinssatz bei Krediten und Anlageprodukten abgelöst.

Das ökonometrische Modell der Geldnachfragefunktion ist durch ein multiples Regressionsmodell,

(2.2) $y_t = \beta_1 + \beta_2 \cdot x_{2t} + \beta_3 \cdot x_{3t} + u_t$, t=1.2….n,

gegeben, in dem die logarithmierte reale Geldmenge M1 (y) durch die Logarithmen des realen Bruttoinlandsprodukts (x_1) und des Referenzzinssatzes EURIBOR (x_2) erklärt wird. Für die Störgröße u werden wiederum die Standardannahmen vorausgesetzt.

Vor der ökonometrischen Schätzung der Geldnachfragefunktion werden die Logarithmen der Variablen gebildet:

```
>attach(Geldnachfragefkt.data)
> lnM1GERr = log(M1GERr)
>class(lnM1GERr)
[1] "numeric"
> lnM1GERr
 [1] 6.087797 6.192710 6.277282 6.300584 6.357617 6.341364 6.431895 6.479124 6.587274 6.636419
6.690966 6.751733 6.962177 7.056891
>lnBIPr = log(BIPr)
>class(lnBIPr)
[1] "numeric"
>lnBIPr
 [1] 7.501971 7.532195 7.554644 7.571288 7.608628 7.637230 7.655220 7.688139 7.707242 7.749784
7.814343 7.840443 7.811188 7.859313
>lnEURIBOR = log(EURIBOR)
>class(EURIBOR)
[1] "numeric"
>lnEURIBOR
 [1] 1.2029723 1.2641267 1.0885620 1.4793292 1.4492692 1.1999648 0.8458683 0.7466879 0.7839015
1.1249296 1.4539530 1.5325569 0.2070142 -0.2107210
```

Die gewöhnliche Kleinst-Quadrate-Schätzung erfolgt mit der lm-Funktion, in der die additive Verknüpfung der erklärenden Variablen durch das Pluszeichen (+) gekennzeichnet wird:

```
>geldfkt.lm = lm(lnM1GERr ~ lnBIPr+lnEURIBOR)
>summary(geldfkt.lm)

Call:
lm(formula = lnM1GERr ~ lnBIPr + lnEURIBOR)
```

```
Residuals:
    Min       1Q   Median       3Q      Max
-0.085319 -0.034164  0.001443  0.025786  0.076636

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -8.66721    1.05730  -8.198 5.18e-06 ***
lnBIPr       1.99542    0.13571  14.704 1.41e-08 ***
lnEURIBOR   -0.14650    0.03223  -4.545 0.000837 ***
---
Signif.codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.05269 on 11 degrees of freedom
Multiple R-squared: 0.971,     Adjusted R-squared: 0.9657
F-statistic:   184 on 2 and 11 DF,  p-value: 3.504e-09
```

Aufgrund der Verwendung der Variablen in logarithmierter Form können die geschätzten Regressionskoeffizienten als Elastizitäten interpretiert werden. Danach hat eine Erhöhung des realen BIP um 1% eine Steigerung der realen Geldnachfrage um 2,00% zur Folge. Erhöht sich der EURIBOR dagegen um 1%, sinkt die reale Geldnachfrage um 0,15%. Das absolute Glied hat keine spezielle ökonomische Bedeutung, sondern es ist eine technische Konstante, die die Lage der Regressionsebene in dem von den drei Variablen aufgespanntem Raum determiniert.

Alle geschätzten Regressionskoeffizienten sind hochsignifikant ($p<0.01$). Außerdem ist der Gesamtzusammenhang bei einem Bestimmtheitsmaß von 0,971 statistisch gesichert ($F=182$, $p=0.0000$). Der Standardfehler der Regression (SER) beträgt auf der logarithmischen Skala 0.053.

Statt auf der logarithmischen Skala soll die reale Geldmenge M1 auf dem originären Niveau mit den angepassten Werten verglichen werden. Hierzu wird zunächst der Spaltenvektor M1GERr des data.frame-Objekts Geldnachfragefkt.data als Zeitreihe definiert:

```
> M1GERr = ts(M1GERr, start=1997, freq=1)
> M1GERr
Time Series:
Start = 1997
End = 2010
Frequency = 1
[1] 440.45 489.17 532.34 544.89 576.87 567.57 621.35 651.40 725.80 762.36 805.10 855.54 1055.93
1160.83
```

Außerdem werden die im lm-Objekt geldfkt.lm enthaltenen logarithmierten Regressionswerte geldfkt.lm$fitted.values unter Verwendung der Exponentialfunktion expent-logarithmiert und dabei ebenfalls als Zeitreihe definiert:

```
> M1GERrf = ts(exp(geldfkt.lm$fitted.values), start=1997, freq=1)
```

oder

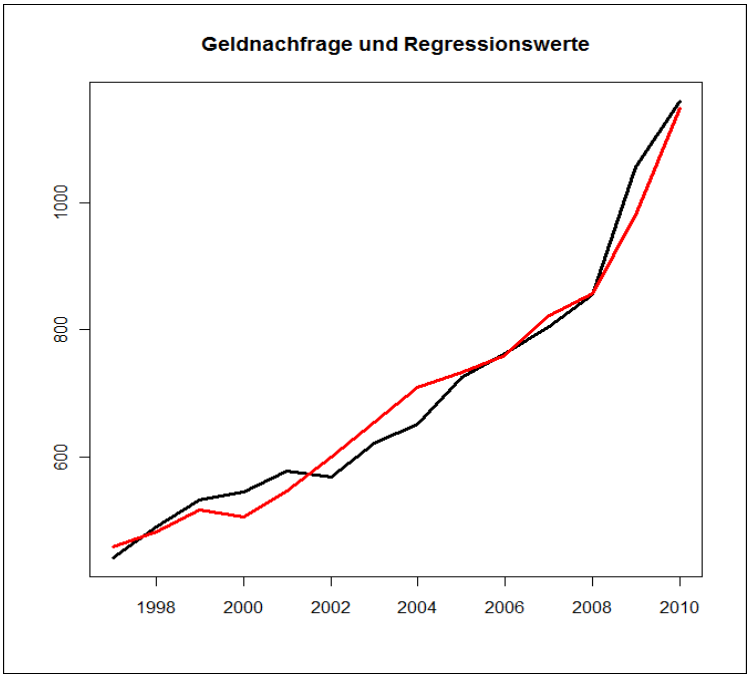
```
> M1GERrf = ts(exp(fitted(geldfkt.lm)), start=1997, freq=1)
> M1GERf
Time Series:
Start = 1997
End = 2010
Frequency = 1
     1       2       3       4       5       6       7       8       9      10      11      12      13      14
 457.6557  481.7705  516.9682  504.6920  546.1313  599.7149  654.7324  709.4168  732.9743  759.0281
822.7575  856.8187  981.4547 1148.5570
```

Der Plot der multiplen Zeitreihe erfolgt wiederum mit der ts.plot-Funktion:

```
>ts.plot(M1GERr,M1GERrf, main="Geldnachfrage und Regressionswerte", gpars=list(xlab="", lwd=3,
col=c("black","red")))
>box(which="figure")
```

Abbildung 2.3 zeigt eine Unterschätzung der Geldnachfrage im ersten und letzten Drittel und eine Überschätzung der Geldnachfrage im mittleren Drittel des Beobachtungszeitraums auf.

Abbildung 2.3: Geldnachfrage und Regressionswerte 1997 - 2010



Unter Verwendung der Zeitreihe der logarithmierten Residuen,

```
> geldfkt.res = ts(geldfkt.lm$residuals, start=1997, freq=1)
```

oder

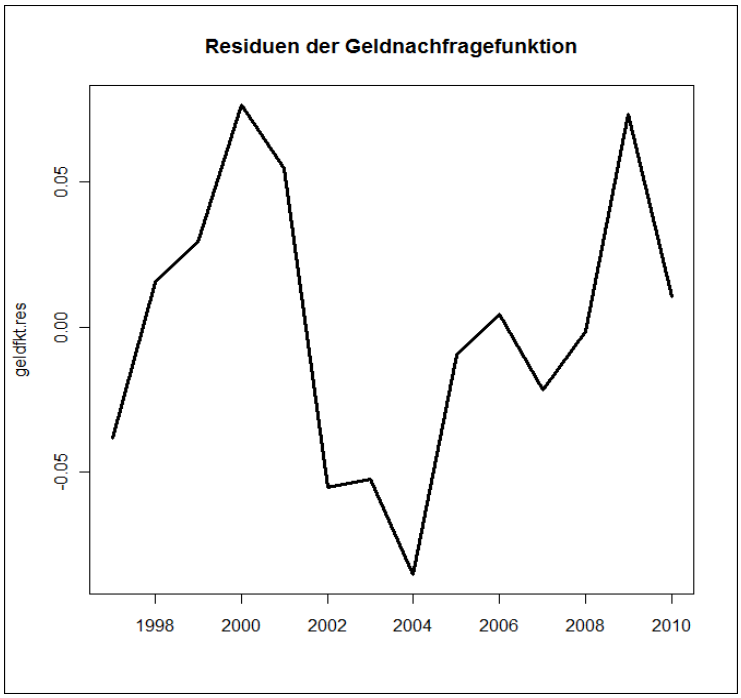
```
> geldfkt.res = ts(resid(geldfkt.lm), start=1997, freq=1)
> geldfkt.res
```

```
Time Series:
Start = 1997
End = 2010
Frequency = 1
1          2          3          4          5          6          7          8          9          10         11         12         13
14
-0.038320129  0.015242321  0.029300955  0.076635526  0.054757433 -0.055090280 -0.052332065 -
0.085319372 -0.009836130  0.004380105 -0.021695053 -0.001493488  0.073141298  0.010628878,
```

wird der Residualplot erstellt:

```
>plot(geldfkt.res, main="Residuen der Geldnachfragefunktion",xlab="", lwd=3)
> box(which="figure").
```

Abbildung 2.4: Residuen der Geldnachfragefunktion



Trotz des ausgeprägt zyklischen Verlaufs lassen sich keine Residuen außerhalb des 2·SER-Bandes identifizieren (SER=0,053).

Übungsaufgaben

2.2.1 Regressieren Sie die Ergasnachfrage (GASV) auf den Erdgaspreis (GASPR) und den Fernwärmepreis FERNWPR (Daten s. data.frame-Objekt Energiel) unter Verwendung der lm-Funktion und speichern Sie das Ergebnis unter dem Namen energiefkt2.lm (=Energienachfragemodell 2)!

2.2.2 Geben Sie den verkürzten Regressionsoutput des lm-Objekts energiefkt2.lm in der R-Konsole aus und interpretieren Sie ihn!

2.2.3 Geben Sie den detaillierten Regressionsoutput des lm-Objekts energiefkt2.lm in der R-Konsole aus und interpretieren Sie ihn!

2.2.4 Definieren Sie die Regressionswerte des Energienachfragemodel 2 als Zeitreihe GASVf2 und stellen Sie sie zusammen mit der Zeitreihe GASV unter dem Titel „Erdgasverbrauch und Regressionswerte 1980-1995“ in einem Zeitreihendiagramm dar!

2.2.5 Definieren Sie die Zeitreihe der Residuen energiefkt2.res des Erdgasnachfrage-modells 2 und plotten Sie sie unter dem Titel „Residuen der Erdgasnachfragefunktion 2“!

2.3 Multiple Regressionsanalyse und Matrizenrechnung

Um den gewöhnlichen Kleinst-Quadrate-Schätzer (OLS-Schätzer)

$$(2.3)\ \hat{\boldsymbol{\beta}} = \begin{bmatrix} \hat{\beta}_1 \\ \hat{\beta}_2 \\ \hat{\beta}_3 \end{bmatrix} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

für den Vektor der Regressionskoeffizienten $\boldsymbol{\beta}' = (\beta_1\ \beta_2\ \beta_3)$ der Geldnachfragefunktion mit Hilfe der Matrizenrechnung zu bestimmen, werden zunächst die 14x3-Beobachtungsmatrix **X** und der 14x1-Vektor **y** der abhängigen Variablen definiert:

```
> n = length(M1GERr)
>n
[1] 14

>one = rep(1,n)
>one
 [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1

> X = cbind(one, lnBIPr, lnEURIBOR)
> X
one   lnBIPrlnEURIBOR
 [1,]   1 7.501971  1.2029723
 [2,]   1 7.532195  1.2641267
 [3,]   1 7.554644  1.0885620
 [4,]   1 7.571288  1.4793292
 [5,]   1 7.608628  1.4492692
 [6,]   1 7.637230  1.1999648
 [7,]   1 7.655220  0.8458683
 [8,]   1 7.688139  0.7466879
 [9,]   1 7.707242  0.7839015
[10,]   1 7.749784  1.1249296
[11,]   1 7.814343  1.4539530
[12,]   1 7.840443  1.5325569
[13,]   1 7.811188  0.2070142
[14,]   1 7.859313 -0.2107210
```

```
> y = lnM1GERr
>y
[1] 6.087797 6.192710 6.277282 6.300584 6.357617 6.341364 6.431895 6.479124 6.587274 6.636419
6.690966 6.751733 6.962177 7.056891.
```

Mit der transponierten Beobachtungsmatrix**X'**,

```
>Xt = t(X)


>Xt
          [,1]     [,2]     [,3]     [,4]     [,5]     [,6]      [,7]     [,8]      [,9]   [,10]   [,11]   [,12]     [,13]      [,14]
one      1.000000 1.000000 1.000000 1.000000 1.000000 1.000000 1.0000000 1.000000 1.0000000 1.000000
1.000000 1.000000 1.0000000  1.0000000
lnBIPr    7.501971 7.532195 7.554644 7.571288 7.608628 7.637230 7.6552202 7.688139 7.7072425 7.749784
7.814343 7.840443 7.8111877  7.8593128
lnEURIBOR  1.202972  1.264127  1.088562  1.479329  1.449269  1.199965  0.8458683  0.746688  0.7839015
1.124930 1.453953 1.532557 0.2070142 -0.2107210,
```

lässt sich unter Verwendung des Operators %*% der Matrizenmultiplikation das Matrixprodukt **X'X** bilden:

```
>XtX = Xt %*% X
>XtX
one   lnBIPrlnEURIBOR
one       14.00000 107.5316  14.16841
lnBIPr    107.53163 826.1196 108.47594
lnEURIBOR  14.16841 108.4759  17.66181.
```

Schneller ist das Matrixprodukt **X'X** mit der crossprod-Funktion zu berechnen:

```
>XtX = crossprod(X)
>XtX
one   lnBIPrlnEURIBOR
one       14.00000 107.5316  14.16841
lnBIPr    107.53163 826.1196 108.47594
lnEURIBOR  14.16841 108.4759  17.66181
```

Die inverse Matrix $(\mathbf{X'X})^{-1}$ wird mit der solve-Funktion gebildet:

```
>XtXi = solve(XtX)
>XtXi
one     lnBIPrlnEURIBOR
one      402.705276 -51.6650087 -5.7346850
lnBIPr   -51.665009   6.6346088  0.6973148
lnEURIBOR  -5.734685   0.6973148  0.3742266.
```

Nach Multiplikation der Inversen $(\mathbf{X'X})^{-1}$ von rechts mit dem Vektor $\mathbf{X'y}$,

```
>Xty = Xt %*% y
>Xty
[,1]
one       91.15383
lnBIPr    700.56231
lnEURIBOR 91.06668
```

oder

```
>Xty = crossprod(X,y)
>Xty
[,1]
one       91.15383
lnBIPr    700.56231
lnEURIBOR 91.06668,
```

ergibt sich der gesuchte Kleinst-Quadrate-Schätzer $\hat{\boldsymbol{\beta}}$:

```
> b = XtXi %*% Xty
>b
[,1]
one          -8.6672107
lnBIPr        1.9954165
lnEURIBOR    -0.1464953.
```

Die geschätzten Regressionskoeffizienten, die hier mit einer größeren Genauigkeit ausgegeben werden, stimmen mit denen der lm-Funktion überein.

Die geschätzte Varianz-Kovarianz-Matrix der Regressionskoeffzienten,

$$(2.4)\quad \hat{\mathrm{Cov}}(\hat{\boldsymbol{\beta}}) = \begin{bmatrix} \hat{\mathrm{V}}(\hat{\beta}_1) & \hat{\mathrm{Cov}}(\hat{\beta}_1,\hat{\beta}_2) & \hat{\mathrm{Cov}}(\hat{\beta}_1,\hat{\beta}_3) \\ \hat{\mathrm{Cov}}(\hat{\beta}_2,\hat{\beta}_1) & \hat{\mathrm{V}}(\hat{\beta}_2) & \hat{\mathrm{Cov}}(\hat{\beta}_2,\hat{\beta}_3) \\ \hat{\mathrm{Cov}}(\hat{\beta}_3,\hat{\beta}_1) & \hat{\mathrm{Cov}}(\hat{\beta}_3,\hat{\beta}_2) & \hat{\mathrm{V}}(\hat{\beta}_3) \end{bmatrix} = \hat{\sigma}^2 \cdot (\mathbf{X'X})^{-1},$$

enthält in der Hauptdiagonale ihre geschätzten Varianzen, die für Konfidenzintervalle und Signifikanztests benötigt werden. $\hat{\sigma}^2$ ist hierin der erwartungstreue Schätzer der Varianz der Störgröße, σ^2,

$$(2.5)\quad \hat{\sigma}^2 = \frac{1}{n-k}\sum_{t=1}^{n}\hat{u}_t^2 ,$$

der durch Division der Summe der quadrierten Residuenquadrate $\hat{u}_t^2$ durch die Anzahl der Freiheitsgrade, n-k, gegeben ist:

```
>dim(X)
[1] 14  3
> k = dim(X)[2]
> k
[1] 3
> s2 = sum(geldfkt.res^2)/(n-k)
> s2
[1] 0.002775921.
```

Multiplikation der Elemente der Inversen $(\mathbf{X'X})^{-1}$mit der Residualvarianz $\hat{\sigma}^2$ ergibt die geschätzte Varianz-Kovarianz-Matrix:

```
>covb = s2*XtXi
>covb
                one          lnBIPr        lnEURIBOR
one          1.11787798  -0.143417975  -0.015919032
lnBIPr      -0.14341798   0.018417149   0.001935691
lnEURIBOR -0.01591903   0.001935691   0.001038823.
```

Direkt lässt sich $\widehat{\mathrm{Cov}}(\hat{\boldsymbol{\beta}})$ in R mit der vcov-Funktion bestimmen:

```
> covb = vcov(geldfkt.lm)
> covb
                (Intercept)        lnBIPr        lnEURIBOR
(Intercept)    1.11787798  -0.143417975  -0.015919032
lnBIPr        -0.14341798   0.018417149   0.001935691
lnEURIBOR    -0.01591903   0.001935691   0.001038823.
```

Nach Extraktion der geschätzten Varianzen der Regressionskoeffizienten aus der Varianz-Kovarianz-Matrix $\widehat{\mathrm{Cov}}(\hat{\boldsymbol{\beta}})$,

```
>vb = diag(covb)
>vb
    one          lnBIPr       lnEURIBOR
1.117877977 0.018417149 0.001038823,
```

erhält man die Standardfehler der Regressionskoeffizienten mit der Wurzelfunktion sqrt:

```
>seb = sqrt(vb)
>seb
1.05729749   0.13570980   0.03223078.
```

Durch Division der Regressionskoeffizienten (b) durch ihre Standardfehler (seb) ergeben sich die für die Signifikanztests der Nullhypothesen

(2.6)H_0: $ß_j = 0$, j=1,2,…,k

benötigten t-Werte:

```
>tb = b/seb
>tb
                [,1]
one          -8.197514
lnBIPr       14.703556
lnEURIBOR   -4.545198.
```

Sie sind mit einem kritischen Wert zu vergleichen, der aus der t-Verteilung mit n-k=14-3=11 Freiheitsgraden zu ermitteln ist. Bei einem Signifikanzniveau α von 5% ist der kritische Wert bei einem zweiseitigen Test durch das 0,975-Quantil $t_{11;0,975}$ gegeben, der in R mit der Quantil-Funktion qt bestimmt werden kann:

```
> tkrit5 = qt(0.975, 11)
> tkrit5
[1] 2.200985
```

Da alle t-Werte der geschätzten Regressionskoeffizienten absolut größer als der kritische Wert $t_{11;0,975}$ sind, müssen die Nullhypothesen (2.6) auf dem Signifikanzniveau von 5% abgelehnt werden. Somit haben das reale BIP und der EURIBOR einen signifikanten Einfluss auf die reale Geldnachfrage. Außerdem ist die technische Konstante signifikant.

Alternativ kann die Signifikanz der Einflussgrößen auf der Basis der p-Werte beurteilt werden, wozu in R die Verteilungsfunktion pt verfügbar ist. Die p-Werte

```
> p.tb = (1 - pt(abs(tb), 11))*2
> p.tb
                [,1]
one         5.176329e-06
lnBIPr      1.405882e-08
lnEURIBOR   8.366998e-04
```

geben die tatsächlichen Signifikanzniveaus der geschätzten Regressionskoeffizientender Geldnachfragefunktion wieder.

Abschließend soll der F-Test über den mit der Geldnachfragefunktion spezifizierten Gesamtzusammenhang unter Verwendung des DeterminationskoeffizientenR^2 durchgeführt werden. Hierzu wird zunächst der Determinationskoeffizient alsVerhältnis der Varianzen der logarithmierten Regressions- und Beobachtungswerte der realen Geldnachfrage mit der var-Funktion berechnet:

```
> lnM1GERrf = geldfkt.lm$fitted.values
```

```
> lnM1GERrf
     1        2        3        4        5        6        7        8        9       10       11       12       13       14
6.126117 6.177468 6.247981 6.223948 6.302860 6.396454 6.484227 6.564443 6.597111 6.632039 6.712662
6.753226 6.889036 7.046262
> R2 = var(lnM1GERrf)/var(lnM1GERr)
> R2
[1] 0.9709833.
```

Unter Verwendung des R^2-Werts nimmt die Prüfgröße des F-Test,

$$(2.6)\ F = \frac{R^2/(k-1)}{(1-R^2)/(n-k)},$$

den Wert

```
> F = (n-k)*R2/((1-R2)*(k-1))
> F
[1] 184.0461.
```

an.

Bei einem Signifikanzniveau von 5% ist der kritische Wert gleich dem 95%-Quantil der F-Verteilung mit k-1=2 und n-k=11 Freiheitsgrade:

```
> Fkrit5 = qf(0.95,k-1,n-k)
> Fkrit5
[1] 3.982298.
```

Wegen F>Fkrit5 ist der Gesamtzusammenhang bei einem Signifikanzniveau von 5% statistisch gesichert.

Zu diesem Schluss kommt man ebenfalls auf der Basis des p-Werts,

```
>p.F = 1 - pf(F,k-1,n-k)
> p.F
[1] 3.503993e-09,
```

da dieser kleiner als das vorgegebene Signifikanzniveau α von 5% ist.

Übungsaufgaben

2.3.1 Definieren Sie die Beobachtungsmatrix XE2 und den Vektor ye2 der abhängigen Variablen des Energienachfragemodells 2 (Daten s. data.frame-Objekt Energiel)!

2.3.2 Bestimmen Sie den Kleinst-Quadrate-Schätzer be2 für den Vektor der Regressionskoeffizienten des Energienachfragemodells 2 mit Hilfe der Matrizenrechnung!

2.3.3 Berechnen Sie den erwartungstreuen Schätzer s2e3 für die Störvarianz σ^2 des Energienachfragemodells 2 mit der sum-Funktion!

2.3.4 Geben Sie die geschätzte Varianz-Kovarianz-Matrix covbe2 des Kleinst-Quadrate-Schätzers be2 an!

2.3.5 Testen Sie die geschätzten Regressionskoeffizienten des Energienachfragemodells 2 unter Verwendung der Hauptdiagonalelemente der Varianz-Kovarianz-Matrix covbe2 unter Verwendung

- des kritischen Werts.
- des p-Werts

Auf Signifikanz (α=0.05)!

2.3.6 Überprüfen Sie die Signifikanz (α=0.05) des durch das Energienachfragemodell 2 postulierten Gesamtzusammenhangs unter Verwendung

- des kritischen Werts.
- des p-Werts!

3. Multikollinearität

3.1 Regression mit drei erklärenden Variablen

Um die Verfahren zur Aufdeckung von Multikollinearität exemplarisch aufzuzeigen, wird zusätzlich die logarithmierte Inflationsrate als erklärende Variable in der Geldnachfragefunktion verwendet:

```
>lnIrate = log(Geldnachfragefkt.data$Irate)
>lnIrate
 [1] -3.963316 -4.710531 -5.115996 -4.199705 -3.963316 -4.268698 -4.605170 -4.017384 -4.199705 -4.135167
-3.772261 -3.649659 -5.521461 -4.509860
```

Die reale Geldnachfrage wird jetzt durch das reale BIP, den Referenzzinssatz EURIBOR und die Inflationsrate erklärt, wobei alle Variablen auf der logarithmischen Skala gemessen werden. Die gewöhnliche Kleinst-Quadrate-Schätzung der erweiterten Geldnachfragefunktion ergibt folgendes Ergebnis:

```
>  geldfkt1.lm = lm(lnM1GERr ~ lnBIPr+lnEURIBOR+lnIrate)
>summary(geldfkt1.lm)

Call:
lm(formula = lnM1GERr ~ lnBIPr + lnEURIBOR + lnIrate)

Residuals:
     Min       1Q   Median       3Q      Max
-0.059256 -0.019728 -0.001689  0.015969  0.074864

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -9.90849    1.12439  -8.812 5.00e-06 ***
lnBIPr       2.11293    0.13391  15.779 2.15e-08 ***
lnEURIBOR   -0.09649    0.03798  -2.541   0.0293 *
```

```
lnIrate    -0.06651    0.03327  -1.999   0.0735 .
---
Signif.codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.04671 on 10 degrees of freedom
Multiple R-squared: 0.9793,     Adjusted R-squared: 0.973
F-statistic: 157.4 on 3 and 10 DF,  p-value: 1.028e-08.
```

Bei unveränderter Einflussrichtung der erklärenden Variablen lnBIPr und lnEURIBOR wirkt sich eine Erhöhung der Inflationsrate (lnIrate) negativ auf die reale Geldnachfrage aus. Dieser Effekt, der auf dem 10%-Niveau signifikant ist, kann wie beim Zinssatz als Ökonomisierung der Geldhaltung mit zunehmender Inflationsrate interpretiert werden.

Von einer Multikollinearität bleibt der Gesamteffekt der Einflussgrößen zwar unberührt. Jedoch sind die Einzeleffekte im Falle einer ausgeprägten Multikollinearität mit einer großen Unsicherheit behaftet, da die Zurechnung auf die einzelnen Regressoren erschwert wird.

3.2 Einfache und erweiterte Korrelationsanalyse

Ob die Multikollinearität im Falle der Geldnachfragefunktion als gravierend eingestuft werden muss, lässt sich mittels verschiedener Diagnoseverfahren überprüfen. In der der einfachen Korrelationsanalyse werden die Korrelationen zwischen den exogenen Variablen bestimmt. Insbesondere bei einer geringen Anzahl von Regressoren kann dies in R durch paarweise Berechnung der Korrelationskoeffizienten unter Verwendung de cor-Funktion erfolgen:

```
>r23 = cor(lnBIPr,lnEURIBOR)
> r23
[1] -0.4425414
>r24 = cor(lnBIPr,lnIrate)
> r24
[1] 0.08459758
>r34 = cor(lnEURIBOR,lnIrate)
> r34
[1] 0.5511654.
```

Bei einer größeren Anzahl zu erklärender Variablen wird eine übersichtliche Darstellung durch Ausgabe der Korrelationsmatrix erreicht. Hierzu wird die originäre Beobachtungsmatrix X um die logarithmierte Inflationsrate lnIrate erweitert:

```
> X1 = cbind(X, lnIrate)
> X1
     One   lnBIPr   lnEURIBOR    lnIrate
[1,]   1 7.501971  1.2029723 -3.963316
[2,]   1 7.532195  1.2641267 -4.710531
 [3,]  1 7.554644  1.0885620 -5.115996
```

```
 [4,]   1 7.571288  1.4793292 -4.199705
 [5,]   1 7.608628  1.4492692 -3.963316
 [6,]   1 7.637230  1.1999648 -4.268698
 [7,]   1 7.655220  0.8458683 -4.605170
 [8,]   1 7.688139  0.7466879 -4.017384
 [9,]   1 7.707242  0.7839015 -4.199705
[10,]   17.749784  1.1249296 -4.135167
[11,]   1 7.814343  1.4539530 -3.772261
[12,]   1 7.840443  1.5325569 -3.649659
[13,]   1 7.811188 0.2070142 -5.521461
[14,]   1 7.859313-0.2107210 -4.509860
```

Die Korrelationsmatrix der exogenen Variablen RX1 kann mit der cor-Funktion mit dem Argument X1 ohne die in der ersten Spalte enthaltenen Scheinvariablen erzeugt werden:

```
> RX1 = cor(X1[,-1])
> RX1
                 lnBIPr      lnEURIBOR      lnIrate
lnBIPr        1.00000000   -0.4425414   0.08459758
lnEURIBOR   -0.44254137    1.0000000   0.55116538
lnIrate       0.08459758    0.5511654   1.00000000
```

Nach einer Standardisierung der Variablen mit der scale-Funktion,

```
> Z1 = scale(X1[,-1])        # Standardisierte Beobachtungsmatrix (ohne Scheinvariable)
> Z1
         lnBIPr     lnEURIBOR    lnIrate
 [1,] -1.48957145  0.3776700  0.7077189
 [2,] -1.23786047  0.4986288 -0.7310173
 [3,] -1.05090481  0.1513751 -1.5117267
 [4,] -0.91228721  0.9242828  0.2525603
 [5,] -0.60131912  0.8648263  0.7077189
 [6,] -0.36311649  0.3717214  0.1197168
 [7,] -0.21328757 -0.3286544 -0.5281491
 [8,]  0.06086803 -0.5248255  0.6036143
 [9,]  0.21996336 -0.4512198  0.2525603
[10,]  0.57425280  0.2233075  0.3768270
[11,]  1.11191575  0.8740906  1.0755893
[12,]  1.32927906  1.0295630  1.3116560
[13,]  1.08563709 -1.5922588 -2.2924361
[14,]  1.48643103 -2.4185071 -0.3446326
attr(,"scaled:center")
  lnBIPr   lnEURIBOR    lnIrate
7.680831  1.012030 -4.330873
attr(,"scaled:scale")
  lnBIPr   lnEURIBOR    lnIrate
0.1200743 0.5055808 0.5193547
```

```
> class(Z1)
[1] "matrix",
```

lässt sich die Korrelationsmatrix RX1 durch Matrixmultiplikation unter Verwendung des Operators %*%,

```
> RX1 = t(Z1)%*%Z1/(n-1)
> RX1
              lnBIPr    lnEURIBOR    lnIrate
lnBIPr      1.00000000 -0.4425414 0.08459758
lnEURIBOR -0.44254137  1.0000000 0.55116538
lnIrate     0.08459758  0.5511654 1.00000000
```

oder mit Hilfe der crossprod-Funktion,

```
> RX1 = crossprod(Z1)/(n-1)
> RX1
              lnBIPr     lnEURIBOR    lnIrate
lnBIPr      1.00000000 -0.4425414 0.08459758
lnEURIBOR -0.44254137  1.0000000 0.55116538
lnIrate     0.08459758  0.5511654 1.00000000,
```

berechnen:

Nach der Faustregel der einfachen Korrelationsanalyse wird Multikollinearität durch Korrelationskoeffizienten angezeigt, die absolut größer als 0,8 sind. Da alle Korrelationskoeffizienten absolut kleiner als 0,8 sind, ist die Multikollinearität in der Geldnachfragefunktion nicht gravierend.

Die Faustregel der erweiterten Korrelationsanalyse stellt dagegen auf den multiplen Korrelationskoeffizienten $R_{yx2x3x4}$ ab. Mit dem im Regressionsoutput ausgewiesenen Determinationskoeffizienten von **0.9793** erhält man $\sqrt{0{,}9793} = 0{,}9896$, so dass hiernach ebenfalls keine Multikollinearität entdeckt wird.

3.3 Determinante und Konditionszahl

Bei exakter Multikollinearität ist die Produktmatrix **X'X** singulär und damit nicht invertierbar. Gleiches gilt für die Korrelationsmatrix RX1 der exogenen Variablen. Die Determinante beider Matrizen ist in diesem Fall gleich null. Merkliche Multikolli-nearität wird daher anhand von Determinanten der beiden Matrizen nahe bei null identifiziert.

Die Produktmatrix **X'X**, X1tX1, lässt sich mit dem Operator %*% für die Matrizenmultiplikation,

```
> X1tX1 = t(X1) %*% X1
```

```
> X1tX1
                one       lnBIPr  lnEURIBOR    lnIrate
one          14.00000  107.5316   14.16841  -60.63223
lnBIPr      107.53163  826.1196 108.47594 -465.63729
lnEURIBOR    14.16841  108.4759   17.66181  -59.48022
lnIrate     -60.63223 -465.6373  -59.48022  266.09699,
```

oder mit der crossprod-Funktion,

```
> X1tX1 = crossprod(X1)
> X1tX1
                One       lnBIPr  lnEURIBOR    lnIrate
one          14.00000  107.5316   14.16841  -60.63223
lnBIPr      107.53163  826.1196 108.47594 -465.63729
lnEURIBOR    14.16841  108.4759   17.66181  -59.48022
lnIrate     -60.63223 -465.6373  -59.48022  266.09699
```

bilden. Da der Wert ihrer Determinante,

```
> X1tX1.det = det(X1tX1)
> X1tX1.det
[1] 13.81835,
```

erheblich größer als null ist, indiziert er kein gravierendes Multikollinearitätsproblem.

Die Determinante der Korrelationsmatrix **R** ist bei perfekter Multikollinearität ebenfalls gleich null und erreicht bei unkorrelierten Variablen den maximalen Wert von eins. Unter Verwendung der in Abschn. 3.2 berechneten Korrelationsmatrix RX1 erhält man

```
> RX1.det = det(RX1)
> RX1.det
[1] 0.4519481.
```

Erneut weicht der Wert der Determinante deutlich von null ab, so dass hierdurch keine merkliche Multikollinearität indiziert wird.

Für die Konditionszahl der Korrelationsmatrix RX1 werden ihre Eigenwerte benötigt:

```
> ev.RX1 = eigen(RX1)$values
> ev.RX1
[1] 1.6670172 1.0825430 0.2504398.
```

Sie ergibt sich als Quadratwurzel des Verhältnisses aus dem größten und kleinsten Eigenwert:

```
> cond.RX1 = sqrt(ev.RX1[1]/ev.RX1[3])
> cond.RX1
[1] 2.579992.
```

Nach der Faustregel liegt gravierende Multikollinearität im Falle einer Konditionszahl größer als 30 vor, was hier nicht der Fall ist.

3.4 Verfahren der Hilfsregressionen (auxiliary regressions)

Beim Verfahren der Hilfsregressionen (auxiliary regressions) wird jeweils eine der k-1 erklärenden Variablen, x_j, j=2,3,…,k, auf die übrigen k-2 “echten“ exogenen Variablen und einer Scheinvariablen regressiert. Mit dem F-Test wird anschließend überprüft, ob ein signifikanter statistischer Zusammenhang zwischen x_j und den übrigen Variablen vorliegt. Die Prüfgrößen der F-Tests werden hier unter Verwendung der Bestimmtheitsmaße der Hilfsregressionen berechnet.

Für die F-Tests der um die logarithmierte Inflationsrate erweiterten Geldnachfragefunktion auf Multikollinearität benötigt man die Anzahl der Beobachtungen n und die Anzahl der Regressoren k:

```
> dim(X1)
[1] 144
> n = dim(X1)[1]
> n
[1] 14
> k = dim(X1)[2]
> k
[1] 4.
```

Bei der erweiterten Geldnachfragefunktion sind drei Hilfsregressionen durchzuführen:

```
>reg2.lm = lm(lnBIPr~lnEURIBOR+lnIrate)
>lnBIPrf = reg2.lm$fitted.values
> R2.2 = var(lnBIPrf)/var(lnBIPr)
> R2.2
[1] 0.3508514
> F.2 = (n-k+1)*R2.2/((1-R2.2)*(k-2))
> F.2
[1] 2.972636

> reg3.lm = lm(lnEURIBOR~lnBIPr+lnIrate)
>lnEURIBORf = reg3.lm$fitted.values
> R2.3 = var(lnEURIBORf)/var(lnEURIBOR)
> R2.3
[1] 0.5447941
> F.3 = (n-k+1)*R2.3/((1-R2.3)*(k-2))
> F.3
[1] 6.582444

>reg4.lm = lm(lnIrate~lnBIPr+lnEURIBOR)
>lnIratef = reg4.lm$fitted.values
```

```
> R2.4 = var(lnIratef)/var(lnIrate)
> R2.4
[1] 0.4379853
> F.4 = (n-k+1)*R2.4/((1-R2.4)*(k-2))
> F.4
[1] 4.286221.
```

Wie sich zeigt, ist bei einem Signifikanzniveau von 5% nur der Wert der Prüfgröße F.2 nicht signifikant,

```
> Fkrit5 = qf(0.95,k-2,n-k+1)
> Fkrit5
[1] 3.982298,
```

während auf dem 1%-Niveau keine signifikanten F-Werte mehr auftreten:

```
> Fkrit1 = qf(0.99,k-2,n-k+1)
> Fkrit1
[1] 7.205713.
```

Unter den strikten Annahmen der F-Tests ist eine potenzielle Multikollinearität der erweiterten Geldnachfragefunktion somit nur dann vernachlässigbar, wenn man bereit ist, einen höheren Fehler 2. Art in Kauf zu nehmen.

Um die strikten Annahmen zu vermeiden, stützt man sich bei empirisch-ökonometrischen Analysen jedoch häufig auf die Tolerenzkoeffizienten und Varianzinflationsfaktoren als explorative Instrumente. Da die Toleranzkoeffizienten tol2, tol3 und tol4 den Schwellenwert von 0,10 nicht unterschreiten,

```
> tol2 = 1 - R2.2
> tol2
[1] 0.6491486

> tol3 = 1 - R2.3
> tol3
[1] 0.4552059

> tol4 = 1 - R2.4
> tol4
[1] 0.562014,
```

lässt sich nicht auf eine gravierende Multikollinearität der erweiterten Geldnachfragefunktion schließen. Dass Multikollinearität jedoch nicht vollständig abwesend ist, wird durch die Varianzinflationsfaktoren VIF2, VIF3 und VIF4 angezeigt. Im Vergleich zu einer Situation unkorrelierter Einflussgrößen erhöhen sich die Varianzen der geschätzten Regressionskoeffizienten um das 1,5- bis 2,2-fache:

```
> VIF2 = 1/tol2
```

```
> VIF2
[1] 1.540479

> VIF3 = 1/tol3
> VIF3
[1] 2.196808

> VIF4 = 1/tol4
> VIF4
[1] 1.779313.
```

Übungsaufgaben

3.1 Regressieren Sie die Erdgasnachfrage (GASV) auf den Erdgaspreis (GASPR), den Fernwärmepreis (FERNWPR) und das verfügbare Einkommen (VEINKR) (Daten s. data.frame-Objekt Energiel) unter Verwendung der lm-Funktion und speichern Sie das Ergebnis unter dem Namen energiefkt3.lm (= Energienachfragemodell 3)!

3.2 Geben Sie den detaillierten Regressionsoutput des lm-Objekts energiefkt3.lm in der R-Konsole aus und interpretieren Sie ihn!

3.3 Beurteilen Sie das Ausmaß der Multikollinearität im Energienachfragemodell 3 anhand einer einfachen und erweiterten Korrelationsanalyse!

3.4 Lässt sich aufgrund des auf die Produktmatrix **X'X** und die Korrelationsmatrix **R** angewandten Determinantenkriteriums auf gravierende Multikollinearität im Energienachfragemodell 3 schließen?

3.5 Ist Ihre in Aufgabe 3.4 vorgenommene Bewertung der Multikollinearität aufgrund der Konditionszahl der Korrelationsmatrix **R** zu revidieren?

3.5 Beurteilen Sie das Ausmaß der Multikollinearität im Energienachfragemodell 3 mit dem Verfahren der Hilfsregressionen auf der Basis

- der F-Tests (α=0.05),
- der Toleranzkoeffizienten!

3.6 Bestimmen Sie die Varianzinflationskoeffizienten für das Energienachfragemodell 3 und interpretieren Sie sie!

4. Heteroskedastizität

4.1 Tests auf Heteroskedastizität

Bei der Kleinst-Quadrate-Schätzung des ökonometrischen Eingleichungsmodells ist von einer Konstanz der Varianz der Störvariablen u (Homoskedastizität) ausgegangen worden. Trifft diese Voraussetzung nicht zu, liegt Heteroskedastizität vor. Die OLS-geschätzten Regressionskoeffizienten sind in diesem Fall nicht mehr effizient. Überhöhte Varianzschätzer gehen mit einem Verlust an Präzision der geschätzten Regressionskoeffizienten einher. Als Folge sind die Signifikanztests der Regressionskoeffizienten nicht mehr valide. Außerdem sind die mit ihnen gebildeten Konfidenzintervalle verzerrt.

4.1.1 Goldfeld-Quandt-Test

Ob tatsächlich die Annahme der Homoskedastizität bei einem ökonometrischen Modell verletzt ist, lässt sich mit Tests auf Heteroskedastizität prüfen. Beim Goldfeld-Quandt-Test wird geprüft, ob gruppenweise Heteroskedastizität vorliegt:

H_0: $\sigma_1^2 = \sigma_2^2$ und H_1: $\sigma_1^2 \neq \sigma_2^2$.

Um dies für die Geldnachfragefunktion zu testen, werden die Beobachtungen der Variablen nach den Werten der exogenen Variablen lnBIPr in aufsteigender Reihenfolge geordnet. Hierzu wird zunächst die Beobachtungsmatrix **X** um den Vektor **y** der abhängigen Variablen erweitert,

```
>yX = cbind(y,X)
>yX
         y      one  lnBIPr    lnEURIBOR
[1,] 6.087797   1 7.501971  1.2029723
[2,] 6.192710   1 7.532195  1.2641267
[3,] 6.277282   1 7.554644  1.0885620
[4,] 6.300584   1 7.571288  1.4793292
[5,] 6.357617   1 7.608628  1.4492692
[6,] 6.341364   1 7.637230  1.1999648
[7,] 6.431895   1 7.655220  0.8458683
[8,] 6.479124   1 7.688139  0.7466879
[9,] 6.587274   1 7.707242  0.7839015
[10,] 6.636419  1 7.749784  1.1249296
[11,] 6.690966  1 7.814343  1.4539530
[12,] 6.751733  1 7.840443  1.5325569
[13,] 6.962177  1 7.811188  0.2070142
```

```
[14,] 7.056891   1 7.859313 -0.2107210,
```

und für die abhängige Variable die allgemeine Bezeichnung y durch den Variablennamen lnM1GERr ersetzt:

```
>colnames(yX)
[1] "y"       "one"      "lnBIPr"   "lnEURIBOR"
>colnames(yX)[[1]] = "lnM1GERr"
>colnames(yX)
[1] "lnM1GERr"  "one"      "lnBIPr"   "lnEURIBOR".
```

Die Sortierung der in der Matrix yX enthaltenen Variablen erfolgt mit der order-Funk-tion:

```
>yX.order = yX[order(lnBIPr),]
>yX.order
     lnM1GERr one  lnBIPr    lnEURIBOR
 [1,] 6.087797   1 7.501971  1.2029723
 [2,] 6.192710   1 7.532195  1.2641267
 [3,] 6.277282   1 7.554644 1.0885620
 [4,] 6.300584   1 7.571288  1.4793292
 [5,] 6.357617   1 7.608628  1.4492692
 [6,] 6.341364   1 7.637230  1.1999648
 [7,] 6.431895   1 7.655220  0.8458683
 [8,] 6.479124   1 7.688139  0.7466879
 [9,] 6.587274   1 7.707242  0.7839015
[10,] 6.636419   1 7.749784  1.1249296
[11,] 6.962177   1 7.811188  0.2070142
[12,] 6.690966   1 7.814343  1.4539530
[13,] 6.751733   17.840443  1.5325569
[14,] 7.056891   1 7.859313 -0.2107210
```

Unter Herausnahme der beiden mittleren Beobachtungen mit den Ordnungsnummern7 und 8 (c=2) werden die beiden Stichproben mit den kleinsten und größten lnBIPr-Werten definiert:

```
> y1 = yX.order[1:6,1]
> y1
[1] 6.087797 6.192710 6.277282 6.300584 6.357617 6.341364
> X1 = yX.order[1:6,3:4]
> X1
     lnBIPr    lnEURIBOR
[1,] 7.501971  1.202972
```

```
[2,] 7.532195  1.264127
[3,] 7.554644  1.088562
[4,] 7.571288  1.479329
[5,] 7.608628  1.449269
[6,] 7.637230  1.199965

> y2 = yX.order[9:14,1]
> y2
[1] 6.587274 6.636419 6.962177 6.690966 6.751733 7.056891
> X2 = yX.order[9:14,3:4]
> X2
       lnBIPr    lnEURIBOR
[1,] 7.707242  0.7839015
[2,] 7.749784  1.1249296
[3,] 7.811188  0.2070142
[4,] 7.814343  1.4539530
[5,] 7.840443  1.5325569
[6,] 7.859313 -0.2107210
```

Der Vektor y1 (y2) enthält die Werte der endogenen Variablen der kleinsten (größten) lnBIPr-Werte (1. Stichprobe bzw. 2. Stichprobe). Entsprechend enthält die Matrix X1 (X2) die Werte der exogenen Variablen (ohne absolutes Glied) der kleinsten (größ-ten) lnBIPr-Werte (1. Stichprobe bzw. 2. Stichprobe).

Für die 1. Stichprobe lauten die Schätzergebnisse

```
> geldfkt1.lm = lm(y1 ~ X1)
>summary(geldfkt1.lm)

Call:
lm(formula = y1 ~ X1)

Residuals:
       1        2        3        4        5        6
-0.0450976 -0.0007379  0.0586455  0.0166274  0.0083077 -0.0377451

Coefficients:
                Estimate   Std. Error t value Pr(>|t|)
(Intercept)   -7.64581    3.43841  -2.224   0.1127
X1lnBIPr       1.82232    0.46046   3.958   0.0288 *
X1lnEURIBOR  0.08954    0.14874   0.602   0.5897
---
```

```
Signif.codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.04914 on 3 degrees of freedom
Multiple R-squared: 0.8614,     Adjusted R-squared: 0.7691
F-statistic: 9.325 on 2 and 3 DF,  p-value: 0.05158
```

und für die 2. Stichprobe

```
> geldfkt2.lm = lm(y2 ~ X2)
>summary(geldfkt2.lm)

Call:
lm(formula = y2 ~ X2)

Residuals:
      1        2        3        4        5        6
-0.015639  0.005879  0.046680 -0.014262  0.006853 -0.029509


Coefficients:
                Estimate  Std. Error t value Pr(>|t|)
(Intercept)   -9.00394    2.14449  -4.199  0.02465 *
X2lnBIPr       2.04264    0.27453   7.441  0.00502 **
X2lnEURIBOR -0.17380    0.02261  -7.688  0.00457 **
---
Signif.codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.03454 on 3 degrees of freedom
Multiple R-squared: 0.9797,     Adjusted R-squared: 0.9662
F-statistic:  72.4 on 2 and 3 DF,  p-value: 0.002892.
```

Unter Verwendung der Residualquadratsummen der beiden Stichproben,

```
> sq1 = sum(geldfkt1.lm$resid^2)
> sq1
[1] 0.007243815
> sq2 = sum(geldfkt2.lm$resid^2)
> sq2
[1] 0.003579361,
```

errechnet sich für die Prüfgröße GQ des Goldfeld-Quandt-Tests der Wert

```
> GQ = sq1/sq2
> GQ
[1] 2.023773.
```

Mit den Größen n, k und c,

```
> n = dim(yX)[1]
> n
[1] 14
> k = dim(yX)[2]-1
> k
[1] 3
> c = 2
> c
[1] 2
```

lässt sich die Zahl der Freiheitsgrade (n-c)/2-k berechnen, die für den kritischen Wert der F-Verteilung benötigt wird. Bei einem Signifikanzniveau von 5% erhält man den kritischen F-Wert

```
> Fkrit5 = qf(0.95,(n-c)/2-k,(n-c)/2-k)
> Fkrit5
[1] 9.276628.
```

Wegen GQ < Fkrit5 lässt sich die Nullhypothese der Homoskedastizität auf dem 5%-Signifikanzniveau mit dem Goldfeld-Quandt-Test nicht verwerfen. Alternativ kann die Testentscheidung durch einen Vergleich des p-Werts,

```
> p.GQ = 1 - pf(GQ,(n-c)/2-k,(n-c)/2-k)
> p.GQ
[1] 0.2886486,
```

mit dem nominalen Signifikanzniveau α=0,05 erfolgen. Aus p.GQ >α folgt für die Geldnachfragefunktion die Beibehaltung der Nullhypothese der Homoskedastizität.

Der hier detailliert aufgezeigte Goldfeld-Quandt-Test lässt sich direkt mit der R-Funktion gqtest nach Aufruf der Programmbibliothek lmtest des Pakets zoo,

```
>library(lmtest),
```

durchführen, die hierzu wie folgt zu spezifizieren ist:

```
> geldfkt.gq = gqtest(geldfkt.lm, fraction=2, alternative = "less", order.by=~ lnBIPr).
```

Das Testergebnis ist in dem htest-Objekt geldfkt.gq gespeichert,

```
>class(geldfkt.gq)
[1] "htest",
```

das folgende Struktur besitzt:

```
> str(geldfkt.gq)
List of 5
 $ statistic: Named num 0.494
  ..- attr(*, "names")= chr "GQ"
 $ parameter: Named num [1:2] 3 3
  ..- attr(*, "names")= chr [1:2] "df1" "df2"
 $ method   : chr "Goldfeld-Quandt test"
 $ p.value  :num 0.289
 $ data.name: chr "geldfkt.lm"
 - attr(*, "class")= chr "htest".
```

Hieraus gehen bereits der Wert der Prüfgröße und der p-Wert hervor. Der Aufruf des htest-Objekts geldfkt.gq liefert das vollständige Testergebnis unter Angabe der Anzahl der Freiheitsgrade:

```
>geldfkt.gq

      Goldfeld-Quandt test

data:  geldfkt.lm
GQ = 0.4941, df1 = 3, df2 = 3, p-value = 0.2886.
```

Der ausgewiesene p-Wert von 0.2886 bei jeweils 3 Freiheitsgraden im Zähler- und Nenner stimmt mit demjenigen der ausführlichen Berechnung überein. Die Prüfgröße GQ von 0.4941 ist als Verhältnis der Residualvarianzen der zweiten und ersten Stichprobe, sq2/sq1, berechnet worden. Sie entspricht daher genau dem Kehrwert der aus der ausführlichen Berechnung hervorgehenden aufgerundeten Prüfgröße von 2.0238 (=1/0.4941).[2]

[2] In Lehrbüchern zur Ökonometrie wird die Prüfgröße des Goldfeld-Quandt-Tests in der Regel als Verhältnis der größeren zur kleineren Residualvarianz der beiden Stichproben definiert. Vgl. Dreger/Kosfeld/Eckey (2014, S. 87f.).

4.1.2 Breusch-Pagan-Test

Mit dem Breusch-Pagan-Test wird getestet, ob die Störvarianz eine Funktion beobachtbarer Variablen z_2, z_3,…, z_p ist:

$$(4.1)\quad \sigma_t^2 = h\left(\alpha_1 + \alpha_2 \cdot z_{2t} + \ldots + \alpha_p \cdot z_{pt}\right).$$

Die Nullhypothese der Homoskedastizität,

(4.2) H_0: $\alpha_2 = \alpha_3 = \ldots = \alpha_p = 0$,

wird abgelehnt, wenn mindestens ein α_j ungleich 0 ist:,

H_1: Mindestens ein $\alpha_j \neq 0$, j=2,3,…,p.

Bei der Geldnachfragefunktion wird die Störvarianz unter der Alternativhypothese als lineare Funktion des logarithmierten realen BIP, lnBIPr, betrachtet:

$$(4.3)\quad \sigma_t^2 = \alpha_1 + \alpha_2 \cdot \ln BIPr_t\,.$$

Mit dem Maximum-Likelihood-Schätzer (ML-Schätzer)

$$\hat{\sigma}_{ML}^2 = \frac{1}{n}\sum_{t=1}^{n}\hat{u}_t^2$$

für die Störvarianz σ^2 der Geldnachfragefunktion,

```
> s2ML = sum(geldfkt.resl^2)/n
> s2ML
[1] 0.002181081,
```

lässt sich die abhängige Variable $\hat{u}_t^2/\hat{\sigma}_{ML}^2$ der Hilfsregression (auxiliary regression) operationalisieren:

```
> u2s2ML = geldfkt.res^2/s2ML
> u2s2ML
Time Series:
Start = 1997
End = 2010
Frequency = 1
```

```
        1          2          3          4          5          6          7          8          9         10         11         12         13
14
0.673259049 0.106519840 0.393633287 2.692703651 1.374720572 1.391484020 1.255636746 3.337517679
0.044358498 0.008796244 0.215799134 0.001022661 2.452751793 0.051796826.
```

Die Hilfsregression der abhängigen Variablen $\hat{u}_t^2/\hat{\sigma}_{ML}^2$, u2s2ML, auf die unabhängige Variable lnBIPr,

```
>auxreg.bp = lm(u2s2ML ~ lnBIPr),
```

erbringt das Schätzergebnis

```
>summary(auxreg.bp)

Call:
lm(formula = u2s2ML ~ lnBIPr)

Residuals:
   Min     1Q  Median     3Q    Max
-1.1225 -0.7889 -0.5904  0.3091  2.3488

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  12.837     20.510   0.626    0.543
lnBIPr       -1.541      2.670  -0.577    0.574

Residual standard error: 1.156 on 12 degrees of freedom
Multiple R-squared: 0.02701,    Adjusted R-squared: -0.05407
F-statistic: 0.3331 on 1 and 12 DF,  p-value: 0.5745.
```

Als Differenz zwischen der totalen Abweichungsquadratsumme SST der abhängigen Variablen u2s2ML der Hilfsregression (4.3),

```
>SST.bp = sum((u2s2ML-mean(u2s2ML))^2)
>SST.bp
[1] 16.47925,
```

und der Residualabweichungsquadratsumme SSR,

```
>SSR.bp = sum(auxreg.bp$res^2)
>SSR.bp
```

```
[1] 16.03410,
```

erhält man die Abweichungsquadratsumme SSE:

```
>SSE.bp = SST.bp - SSR.bp
>SSE.bp
[1] 0.4451416.
```

Damit nimmt die Prüfgröße des Breusch-Pagan-Tests,

(4.4) $\mathrm{BP}_1 = \frac{1}{2} \cdot \mathrm{SSE}$,

```
> BP1 = SSE.bp/2
> BP1
[1] 0.2225708,
```

die asymptotisch χ^2-verteilt ist mit 1 (=p-1) Freiheitsgrad. Wegen

```
> chikrit5 = qchisq(0.95, 1)

> chikrit5
[1] 3.841459
```

gilt $BP_1 < \chi^2_{1;\,0.95}$, so dass die Nullhypothese (4.2) auf dem 5%-Signifikanzniveau nicht verworfen werden kann. Zu dieser Testentscheidung kommt man gleichermaßen mit dem p-Wert,

```
> p.BP1 = 1 - pchisq(BP1, 1)
> p.BP1
[1] 0.637088,
```

wegen p=0.637>α=0.05. Wie mit dem Goldfeld-Quandt-Test lässt sich somit auch mit dem Breusch-Pagan-Test keine Heteroskedastizität der Geldnachfragefunktion aufdecken.

Der Breusch-Pagan-Test ist ebenfalls in der Programmbibliothek lmtest des Pakets zoo direkt verfügbar. Mit der Spezifikation

```
>geldfkt.bp = bptest(geldfkt.lm, varformula= ~ lnBIPr, studentize=FALSE)
```

wird durch Eingabe des htest-Objektnamens geldfkt.bp nach Bestätigung mit der Return-Taste das Testergebnis in der R-Konsole ausgegeben,

```
>geldfkt.bp

Breusch-Pagan test

data:  geldfkt.lm
BP = 0.2226, df = 1, p-value = 0.6371,
```

das unter Berücksichtigung der Rundung mit demjenigen der detaillierten Berechnung übereinstimmt.

Übungsaufgaben

4.1.1 Berechnen Sie detailliert die Prüfgröße des Goldfeld-Quandt-Tests für die Nachfrage nach Erdgas (GASV) in Abhängigkeit vom Erdgaspreis (GASPR) und des verfügbaren Einkommens (VEINKR) (=Energienachfragemodell 2a)! Testen Sie ihre Signifikanz (α=0,05) unter Verwendung

- des kritischen Werts,
- des p-Werts!

4.1.2 Führen Sie den Goldfeld-Quandt-Test in dem Energienachfragemodell 2a unter Verwendung der in der Programmbibliothek lmtest verfügbaren Funktionendurch und interpretieren Sie das Testergebnis!

4.1.3 Berechnen Sie detailliert die Prüfgröße des Breusch-Pagan-Tests für die Nachfrage nach Erdgas (GASV) in Abhängigkeit vom Erdgaspreis (GASPR) und des verfügbaren Einkommens (VEINKR) (=Energienachfragemodell 2a)! Testen Sie ihre Signifikanz (α=0,05) unter Verwendung

- des kritischen Werts,
- des p-Werts!

4.1.4 Führen Sie den Breusch-Pagan-Test in dem Energienachfragemodell 2a unter Verwendung der in der Programmbibliothek lmtest verfügbaren Funktionen durch und interpretieren Sie das Testergebnis!

4.2 Gewichtete Kleinst-Quadrate-Schätzung (WLS)

Im Falle von Heteroskedastizität können Parameter des multiplen Regressionsmodells mit der gewichteten Methode der kleinsten Quadrate (WLS-Methode) geschätzt werden. Hierzu wird der Ansatz einer Abhängigkeit der Varianz σ_t^2 der Störvariablen u von den quadrierten Werten einer exogenen Variablen z gewählt:

(4.4) $\sigma_t^2 = \sigma^2 \cdot z_t^2, \quad t = 1,2,...,n$.

Bei der WLS-Methode werden die Variablen eines Regressionsmodells mit den inversen Beobachtungswerten $1/z_t$ multipliziert bevor das transformierte Modell OLS-geschätzt wird. Wird z.B. die Heteroskedastizität der Geldnachfragefunktion auf das reale Bruttoinlandsprodukt x_2 zurückgeführt, ist statt (2.2) das transformierte Modell

(4.5a) $y_t/x_{2t} = \beta_1 \cdot 1/x_{2t} + \beta_2 \cdot x_{2t}/x_{2t} + \beta_3 \cdot x_{3t}/x_{2t} + u_t/x_{2t}$

oder

(4.5b) $y_t^* = \beta_1 \cdot x_{1t}^* + \beta_2 + \beta_3 \cdot x_{3t}^* + u_t^*$

mit $y_t^* = y_t/x_{2t}$, $x_{1t}^* = 1/x_{2t}$, $x_{3t}^* = x_{3t}/x_{2t}$, $u_t^* = u_t/x_{2t}$.

mit der gewöhnlichen Kleinst-Quadrate-Methode zu schätzen.

Die transformierten Variablen werden durch den Suffix s gekennzeichnet:

```
> lnM1GERrs = lnM1GERr/lnBIPr
>Intercepts = 1/lnBIPr
>lnEURIBORs = lnEURIBOR/lnBIPr.
```

Die WLS-Schätzung der Geldnachfragefunktion ergibt sich durch ihre Spezifikation unter Verwendung der transformierten Variablen lnM1GERrs, Intercepts und lnEURIBORs in der lm-Funktion:

```
>geldfkt.wls = lm(lnM1GERrs ~ Intercepts+lnEURIBORs)
>summary(geldfkt.wls)

Call:
lm(formula = lnM1GERrs ~ Intercepts + lnEURIBORs)

Residuals:
     Min       1Q    Median       3Q       Max
-0.0110534 -0.0045686  0.0001382  0.0033927  0.0100421

Coefficients:
              Estimate   Std. Error t value  Pr(>|t|)
(Intercept)    1.99529   0.13652  14.615   1.50e-08 ***
Intercepts    -8.66753   1.06358  -8.149   5.48e-06 ***
lnEURIBORs    -0.14523   0.03285  -4.422   0.00103 **
---
Signif.codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.006879 on 11 degrees of freedom
Multiple R-squared: 0.9334,     Adjusted R-squared: 0.9213
F-statistic: 77.06 on 2 and 11 DF,  p-value: 3.387e-07.
```

Die WLS-Schätzung unterscheidet sich nur marginal von der OLS-Schätzung, da der Goldfeld-Quandt-Test und Breusch-Pagan-Test keine Heteroskedastizität in der Geldnachfragefunktion aufgedeckt haben, so dass sie hier allein zu Illustrationszwecken verwendet worden ist.

Bei der Identifikation der geschätzten Regressionskoeffizienten ist zu beachten, dass das absolute Glied (Intercept) von 1.99529 des WLS-Modells die Einkommenselastizität angibt, da die Modellvariablen mit den Kehrwerten des logarithmierten realen BIP multipliziert worden sind. Der geschätzte Regressionskoeffizient der Variablen lnBIPRs, -8.66753, gibt dagegen das neue absolute Glied wieder. Die Identifikation der Zinselastizität von -0.14523 als geschätzter Regressionskoeffizient der Variablen lnEURIBORs ist dagegen unproblematisch.

Übungsaufgaben

4.2.1 Transformieren die Variablen der Regression der Erdgasnachfrage (GASV) auf den Erdgaspreis (GASPR) und das verfügbare Einkommen (VEINKR) (=Energienachfragemodells 2a, Daten s. data.frame-Objekt Energiel) unter Verwendung des Ansatzes (4.4) mit z_t=VEINKR$_t$!

4.2.2 Schätzen Sie das dasEnergienachfragemodell 2a unter Verwendung des Ansatzes (4.4) mit z_t=VEINKR$_t$ mit der gewichteten Methode der kleinsten Quadrate (WLS-Methode) und speichern Sie das Ergebnis unter dem Namen energiefkt2a.wls!

4.2.3 Vergleichen Sie die OLS- und WLS-Schätzer des Energienachfragemodells 2a! Wie lässt sich der Befund interpretieren?

4.2.4 Berechnen Sie den Determinationskoeffizienten für die Güte der WLS-Schätzung des Energienachfragemodells 2a für die originären Modellvariablen!

4.3 Heteroskedastizitäts-konsistente (HC) Standardfehler

Da bei Heteroskedastizität nicht die OLS-Schätzer der Regressionskoeffizienten, sondern ihre Standardfehler verzerrt sind, könnte bei der Behebung des Modelldefekts unmittelbar hier angesetzt werden. White (1980) hat hierzu vorgeschlagen, die Varianz-Kovarianz-Matrix der geschätzten Regressionskoeffizienten (2.4) durch

$$(4.6) \quad \widehat{Cov}(\hat{\boldsymbol{\beta}}) = (\mathbf{X'X})^{-1} \cdot \mathbf{X'} \cdot \mathrm{diag}(\hat{u}_1^2, \hat{u}_2^2, ..., \hat{u}_n^2) \cdot \mathbf{X} \cdot (\mathbf{X'X})^{-1}$$

zu ersetzen, in der anstelle der einheitlichen Residualvarianz $\hat{\sigma}^2$ die quadrierten Residuen $\hat{u}_1^2, \hat{u}_2^2, ..., \hat{u}_n^2$ verwendet werden. Auf diese Weise wird der veränderlichen Streuung der Störgröße u bei unterschiedlichen Querschnitts- oder Zeiteinheiten t Rechnung getragen. Die Wurzeln der Diagonalelemente der geschätzten Varianz-Kovarianz-Matrix (4.6) heißen Heteroskedastizitäts-konsistente (HC) Standardfehler.

Whites originärer HC-Schätzer der Kovarianzmatrix $\mathrm{Cov}(\hat{\boldsymbol{\beta}})$ lässt sich für die geschätzten Regressionskoeffizienten der Geldnachfragefunktion mit dem Matrixkalkül berechnen, wenn man die Zeitreihe der quadrierten Residuen,

```
> u2 = geldfkt.res^2
> u2
Time Series:
Start = 1997
End = 2010
Frequency = 1
        1            2            3            4            5            6
1.468432e-03 2.323284e-04 8.585460e-04 5.873004e-03 2.998376e-03 3.034939e-03
        7            8            9           10           11           12
2.738645e-03 7.279395e-03 9.674946e-05 1.918532e-05 4.706753e-04 2.230506e-06
       13           14
5.349650e-03 1.129731e-04,
```

in eine Diagonalmatrix transformiert:

```
> U2 = diag(u2):

> covb.White = XtXi%*%Xt%*%U2%*%X%*%XtXi
> covb.White
                one          lnBIPr       lnEURIBOR
one       0.295919018 -0.0382850964  0.0012679068
lnBIPr   -0.038285096  0.0049663856 -0.0002368461
lnEURIBOR 0.001267907 -0.0002368461  0.0004711442.
```

Die Hauptdiagonalelemente der Matrix covb.White enthalten Whites originäre HC-Schätzer der Varianzen $\mathrm{Var}(\hat{\beta}_{j})$, j=1, 2, 3, die bei vorhandener Heteroskedastizität anstelle der üblichen unter der Annahme der Homoskedastizität ermittelten Varianzschätzer für Signifikanztests und Konfidenzintervalle verwendet werden können.

In R sind jedoch in den Paketen car und sandwich Funktionen zur Berechnung diverser HC-Schätzer der Varianz-Kovarianz-Matrix $\mathrm{Cov}(\hat{\boldsymbol{\beta}})$ verfügbar. Im Paket car,

```
>library(car)
Lade nötigesPaket: MASS
Lade nötiges Paket: nnet
Lade nötiges Paket: survival
Lade nötiges Paket: splines,
```

ist dies die Funktion hccm. Der originäre White-Schätzer (4.6) lässt sich hiermit über die Spezifikation type="hc0" erzeugen:

```
> covb.hc0 = hccm(geldfkt.lm, type="hc0")
> covb.hc0
              (Intercept)        lnBIPr       lnEURIBOR
(Intercept)  0.295919018  -0.0382850964  0.0012679068
lnBIPr      -0.038285096   0.0049663856 -0.0002368461
lnEURIBOR    0.001267907  -0.0002368461  0.0004711442.
```

Der originäre HC-Schätzer von White (HC0-Schätzer) besitzt die asymptotische Eigenschaft der Konsistenz. In modifizierter Form lassen sich seine Eigenschaften bei kleinen Stichproben jedoch verbessern. Beim HC-Schätzer vom Typ "hc1" erfolgt eine Anpassung an den endlichen Stichprobenumfang durch eine Multiplikation der rechten Seite von (4.6) mit dem Faktor n/(n-k):

```
> covb.hc1 = hccm(geldfkt.lm, type="hc1")

> covb.hc1
              (Intercept)        lnBIPr       lnEURIBOR
(Intercept)  0.37662420  -0.0487264863  0.0016136996
lnBIPr      -0.04872649   0.0063208544 -0.0003014405
lnEURIBOR    0.00161370  -0.0003014405  0.0005996381.
```

Per Voreinstellung wird mit der hccm-Funktion der HC-Schätzer vom Typ "hc3" ge-wählt, so dass die Befehle

```
> covb.hc3 = hccm(geldfkt.lm)
```

und

```
> covb.hc3 = hccm(geldfkt.lm, type="hc3")
```

zu demselben Ergebnis führen:

```
> covb.hc3
              (Intercept)        lnBIPr       lnEURIBOR
(Intercept)  0.50161377  -0.0651046342  0.0021542700
lnBIPr      -0.06510463   0.0084731370 -0.0004095327
lnEURIBOR    0.00215427  -0.0004095327  0.0008405830.
```

Der HC3-Schätzer ergibt sich aus (4.6), in dem man die Residuenquadrate $\hat{u}_t^2$ durch die Größen $\hat{u}_t^2/(1-h_{tt})^2$ ersetzt. Hierin sind die Größen h_{tt} die Hauptdiagonalelemente der sog. Hat-Matrix $\mathbf{H} = \mathbf{X}(\mathbf{X'X})^{-1}\mathbf{X'}$, die den Vektor der beobachteten Werte der abhängigen Variablen, **y**, in den Vektor der Regressionswerte, $\hat{\mathbf{y}}$, transformiert. Die Diagonalelemente h_{tt} geben den Einfluss der t-ten Beobachtung der unabhängigen Variablen (leverage) auf den jeweiligen Regressionswert an.

Heteroskedastizitäts-konsistente (HC) Schätzer der Varianz-Kovarianz-Matrix der OLS-geschätzten Regressionskoeffizienten können alternativ mit der Funktion vcovHC erzeugt werden, die in R nach Laden des Pakets sandwich verfügbar ist:

```
> library(sandwich)
Lade nötiges Paket: zoo.
```

Bei Angabe des lm-Objekts geldfkt.lm lassen sich mit der Funktion vcovHC die diversen HC-Schätzer durch Spezifikation des Arguments type wie z.B. type="HC0" berechnen. Voreingestellt ist erneut der HC3-Schätzer. Durch die Spezifikation type="const" wird der unkorrigierte Schätzer angefordert.

Für die Signifikanztests der Regressionskoeffizienten der Geldnachfragefunktion unter Berücksichtigung potenziell vorhandener Heteroskedastizität wird hier auf den HC3-Schätzer der Varianz-Kovarianz-Matrix $\mathrm{Cov}(\hat{\boldsymbol{\beta}})$ zurückgegriffen. Hierzu werden aus dem HC-Schätzer covb.hc3 die Hauptdiagonalelemente (=geschätzte Varianzen von $\hat{\beta}_1, \hat{\beta}_2, \ldots, \hat{\beta}_k$) extrahiert:

```
>vb.hc3 = diag(covb.hc3)
>vb.hc3
  (Intercept)      lnBIPr       lnEURIBOR
0.501613769 0.008473137  0.000840583.
```

Die sich hieraus ergebenden Heteroskedastizitäts-konsistente (HC) Standardfehler der Regressionskoeffizienten,

```
>seb.hc3 = sqrt(vb.hc3)
>seb.hc3
  (Intercept)      lnBIPr      lnEURIBOR
0.70824697  0.09204964  0.02899281,
```

unterscheiden sich von den unter der Annahme der Homoskedastizität (= gleiche Störvarianz für alle Beobachtungen) berechneten Werte:

```
>seb = sqrt(vb)
>seb
 (Intercept)       lnBIPr       lnEURIBOR
1.05729749   0.13570980   0.03223078.
```

Während sich der geschätzte Standardfehler des Regressionskoeffizienten der Zinsvariablen um den Faktor 1,11 verringert, nimmt der HC3-Schätzer des Standardfehlers gegenüber dem unkorrigierten Schätzer bei der Einkommensvariablen um den Faktor 1,47 ab.

Ein Vergleich der mit den HC-Standardfehlern berechneten t-Werte

```
> tb.hc3 = b/seb.hc3
> tb.hc3
                [,1]
one         -12.237554
lnBIPr       21.677613
lnEURIBOR   -5.052814
```

mit dem 0,975-Quantil der t-Verteilung mit 11 (=n-k=14-3) Freiheitsgraden,

```
> tkrit5 = qt(0.975, 11)
> tkrit5
[1] 2.200985
```

zeigt, dass alle Regressionskoeffizienten wie bei der Verwendung der Standardfehler des OLS-Modells auf dem 5%-Niveau signifikant sind. Ihr tatsächliches Signifikanzniveau geht aus den p-Werten hervor:

```
>  p.tbhc = (1 - pt(abs(tb.hc3), 11))*2
>  p.tbhc
                [,1]
one         9.507521e-08
lnBIPr      2.248521e-10
lnEURIBOR   3.704356e-04.
```

Zur Durchführung der Signifikanztests über die Regressionskoeffizienten mit einem HC-Schätzer der Varianz-Kovarianz-Matrix $\mathrm{Cov}(\hat{\boldsymbol{\beta}})$ ist die schrittweise Berechnung jedoch nicht erforderlich. Mit der Funktion `coeftest`, die in dem R-Paket `lmtest` enthalten ist,

```
> library(lmtest),
```

lässt sich ein zur originären OLS-Schätzung vergleichbarer Testoutput erzeugen. Da der den Signifikanztests zugrunde liegende HC-Schätzer über das Argument vcov mit der Funktion vcovHC auszuwählen und berechnen ist, muss – sofern noch nicht erfolgt – zusätzlich das R-Paket sandwich geladen werden:

```
> library(sandwich)
Lade nötiges Paket: zoo.
```

Mit dem Argument df lassen sich unter Angabe der Anzahl der Freiheitsgrade t-Tests der geschätzten Regressionskoeffizienten anfordern. Bei der Spezifikation von df=Inf werden die Signifikanztests dagegen mit der Normalapproximation (z-Tests) durchgeführt.

Signifikanztests in Form von t-Tests der Regressionskoeffizienten der Geldnachfrage funktion werden unter Verwendung des Standardfehlers vom Typ HC3 durch die Spezifikation der Funktion coeftest (n=14, k=3)

```
> coeftest.HC3 = coeftest(geldfkt.lm, df=n-k, vcov=vcovHC(geldfkt.lm, type="HC3"))
```

oder aufgrund der Voreinstellung des HC3-Schätzers in der Funktion vcovHC

```
> coeftest.HC3 = coeftest(geldfkt.lm, vcov=vcovHC(geldfkt.lm))
```

durchgeführt:

```
> coeftest.HC3

t test of coefficients:

              Estimate  Std. Error  t value   Pr(>|t|)
(Intercept)  -8.667211  0.708247  -12.2376  9.508e-08 ***
lnBIPr        1.995417  0.092050   21.6776  2.249e-10 ***
lnEURIBOR    -0.146495  0.028993   -5.0528  0.0003704 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1.
```

Die Testergebnisse stimmen mit denjenigen der schrittweisen Tests bei Verwendung des HC3-Schätzers der Varianz-Kovarianz-Matrix der geschätzten Regressionskoeffizienten überein.

Übungsaufgaben

4.3.1 Berechnen Sie die Heteroskedastizitäts-konsistente Varianz-Kovarianz-Matrix der Regressionskoeffizienten des Energienachfragemodells 2a unter Verwendung der Formel (4.6)! Überprüfen Sie das Ergebnis mit der hccm-Funktion!

4.3.2 Berechnen Sie den HC1-Schätzer für $\mathrm{Cov}(\hat{\boldsymbol{\beta}})$ des Energienachfragemodells 2a mit der hccm- und vcovHC-Funktion!

4.3.3 Vergleichen Sie die HC3-Standardfehler der Regressionskoeffizienten des Energienachfragemodells 2a mit den OLS-Standardfehlern! Kennzeichnen Sie die Unterschiede!

4.3.4 Testen Sie die Signifikanz der Regressionskoeffizienten des Energienachfrage-modells 2a (α=0.05)

- schrittweise mit der Funktion vcovHC,
- mit der Funktion coeftest

unter Verwendung der HC3-Standardfehler!

5. Autokorrelation

5.1 Autokorrelationskoeffizient und Korrelogramm

Unter Autokorrelation ist die Korrelation von Werten einer Zeitreihe (x_t) zu verstehen. Folgen die Zeitreihenwerte unmittelbar aufeinander, spricht man von einer Autokorrelation 1. Ordnung. In diesem Fall wird die Korrelation zwischen den n-1 Wertepaaren (x_2, x_1), (x_3, x_2), …, (x_n, x_{n-1}) vergleichbar zum Pearsonschen Korrelationskoeffizienten gemessen:

$$(5.1) \qquad r_1 = \frac{\sum_{t=2}^{n}(x_t - \bar{x})(x_{t-1} - \bar{x})}{\sum_{t=1}^{n}(x_t - \bar{x})^2}.$$

Die Kovarianz zwischen den aufeinander folgenden Werten wird auf die Varianz der Zeitreihe bezogen, wobei der Faktor 1/n in Gleichung (5.1) vernachlässigt ist. Zur Berechnung des Autokorrelationskoeffizienten 1. Ordnung, r_1, wird außerdem in (5.1) der Mittelwert $\bar{x}$ der gesamten Zeitreihe (x_t) anstelle der Mittelwerte der beiden Teilreihen verwendet.

In der Ökonometrie wird untersucht, ob die Residuen $\hat{u}_t$, t=1,2,…,n, einer Regres-sion autokorreliert sind. Der empirische Autokorrelationskoeffizient 1. Ordnung der Residuen, $\hat{\rho}_1$, lässt sich aus den n-1 Wertepaaren $(\hat{u}_2, \hat{u}_{1)}, (\hat{u}_3, \hat{u}_{2)}, ..., (\hat{u}_n, \hat{u}_{n-1})$ berechnen. Wegen $\bar{\hat{u}} = 0$ ist er durch

$$(5.2) \quad \hat{\rho}_1 = \frac{\sum_{t=2}^{n} \hat{u}_t \cdot \hat{u}_{t-1}}{\sum_{t=1}^{n} \hat{u}_t^2}$$

gegeben. Die verwendete Bezeichnung $\hat{\rho}_1$ kennzeichnet eine Schätzung des unbekannten theoretischen Autokorrelationskoeffizienten 1. Ordnung, ρ_1, der Störgröße u_t.

Um den empirischen Autokorrelationskoeffizienten 2. Ordnung der Residuen, $\hat{\rho}_2$,zu bestimmen, werden n-2 geordnete Paare $(\hat{u}_3, \hat{u}_{1)}, (\hat{u}_4, \hat{u}_{2)}, ..., (\hat{u}_n, \hat{u}_{n-2})$ miteinander verglichen:

$$(5.3) \quad \hat{\rho}_2 = \frac{\sum_{t=3}^{n} \hat{u}_t \cdot \hat{u}_{t-2}}{\sum_{t=1}^{n} \hat{u}_t^2} .$$

Allgemein ist der Autokorrelationskoeffizient k-ter Ordnung der Residuen, $\hat{\rho}_k$, als Korrelation zwischen den n-k Wertepaaren $(\hat{u}_{k+1}, \hat{u}_{1)}, (\hat{u}_{k+2}, \hat{u}_{2)}, ..., (\hat{u}_n, \hat{u}_{n-k})$ durch

$$(5.4) \quad \hat{\rho}_k = \frac{\sum_{t=k+1}^{n} \hat{u}_t \cdot \hat{u}_{t-k}}{\sum_{t=1}^{n} \hat{u}_t^2}$$

gegeben.

Beispielhaft sollen die Autokorrelationskoeffizienten erster bis vierter Ordnung der Residuen der Geldnachfragefunktion berechnet werden. Hierzu wird zunächst die Zeitreihe der Residuen u definiert:

```
>u =ts(geldfk.lm$residuals, start=1997, freq=1)
Time Series:
Start = 1997
End = 2010
Frequency = 1
1          2          3          4          5          6          7          8          9          10         11         12
-0.038320129  0.015242321  0.029300955  0.076635526  0.054757433 -0.055090280 -0.052332065 -
0.085319372 -0.009836130  0.004380105 -0.021695053 -0.001493488
       13         14
0.073141298  0.010628878.
```

Die um eine Periode verzögerte Zeitreihe der Residuen, u_1, ist durch

```
> u_1 = lag(u, -1)
```

```
> u_1
Time Series:
Start = 1998
End = 2011
Frequency = 1
        1          2          3          4          5          6          7          8          9         10         11         12
13         14
-0.038320129  0.015242321  0.029300955  0.076635526  0.054757433 -0.055090280 -0.052332065 -
0.085319372 -0.009836130  0.004380105 -0.021695053 -0.001493488  0.073141298  0.010628878
```

gegeben. Nach Gleichung (5.2) ergibt sich für den Zähler des Autokorrelationskoeffizienten 1. Ordnung,

```
> r1.Z = sum(u*u_1, na.rm=TRUE)    oder    r1.Z = sum(u*u_1)
> r1.Z
[1] 0.01203739
```

und für den Nenner

```
> r.N = sum(u^2)
> r.N
[1] 0.03053513.
```

Damit nimmt $\hat{\rho}_1$ den Wert

```
> r1 = r1.Z/r.N
> r1
[1] 0.3942144
```

an[3].

Unter Verwendung der um zwei Perioden verzögerten Residualreihe u_2,

```
> u_2 = lag(u, -2)
> u_2
Time Series:
Start = 1999
End = 2012
Frequency = 1
        1          2          3          4          5          6          7          8          9         10         11         12
13         14
-0.038320129  0.015242321  0.029300955  0.076635526  0.054757433 -0.055090280 -0.052332065 -
0.085319372 -0.009836130  0.004380105 -0.021695053 -0.001493488  0.073141298  0.010628878,
```

[3] Berechnet man Autokorrelationskoeffizienten der Residuen direkt mit der cor-Funktion, ergeben sich geringfügig differierende Ergebnisse, da in diesem Fall nicht für beide Zeitreihen der einheitliche Mittelwert von null der gesamten Residualreihe verwendet wird, sondern die jeweiligen Mittelwerte der Lagreihen in die Berechnung eingehen.

erhält man den Autokorrelationskoeffizienten 2. Ordnung $\hat{\rho}_2$:

```
> r2.Z = sum(u*u_2, na.rm=TRUE)    oder    r1.Z = sum(u*u_2)
> r2.Z
[1] -0.001992230
> r2 = r2.Z/r.N
> r2
[1] -0.06524385.
```

Entsprechend lässt sich der Autokorrelationskoeffizienten 3. Ordnung $\hat{\rho}_3$,

```
> u_3 = lag(u, -3)
> u_3
Time Series:
Start = 2000
End = 2013
Frequency = 1
1          2          3          4          5          6          7          8          9          10         11         12         13
14
-0.038320129  0.015242321  0.029300955  0.076635526  0.054757433 -0.055090280 -0.052332065 -
0.085319372 -0.009836130  0.004380105 -0.021695053 -0.001493488  0.073141298  0.010628878
> r3.Z = sum(u*u_3, na.rm=TRUE)    oder    r1.Z = sum(u*u_3)
> r3.Z
[1] -0.01013049
> r3 = r3.Z/r.N
> r3
[1] -0.3317651,
```

und der Autokorrelationskoeffizienten 4. Ordnung $\hat{\rho}_4$,

```
> u_4 = lag(u, -4)
> u_4
Time Series:
Start = 2001
End = 2014
Frequency = 1
         1          2          3          4          5          6          7          8          9          10         11         12
13         14
-0.038320129  0.015242321  0.029300955  0.076635526  0.054757433 -0.055090280 -0.052332065 -
0.085319372 -0.009836130  0.004380105 -0.021695053 -0.001493488  0.073141298  0.010628878
> r4.Z = sum(u*u_4, na.rm=TRUE) )    oder    r1.Z = sum(u*u_4)
> r4.Z
[1] -0.01119989
> r4 = r4.Z/r.N
> r4
[1] -0.3667872
```

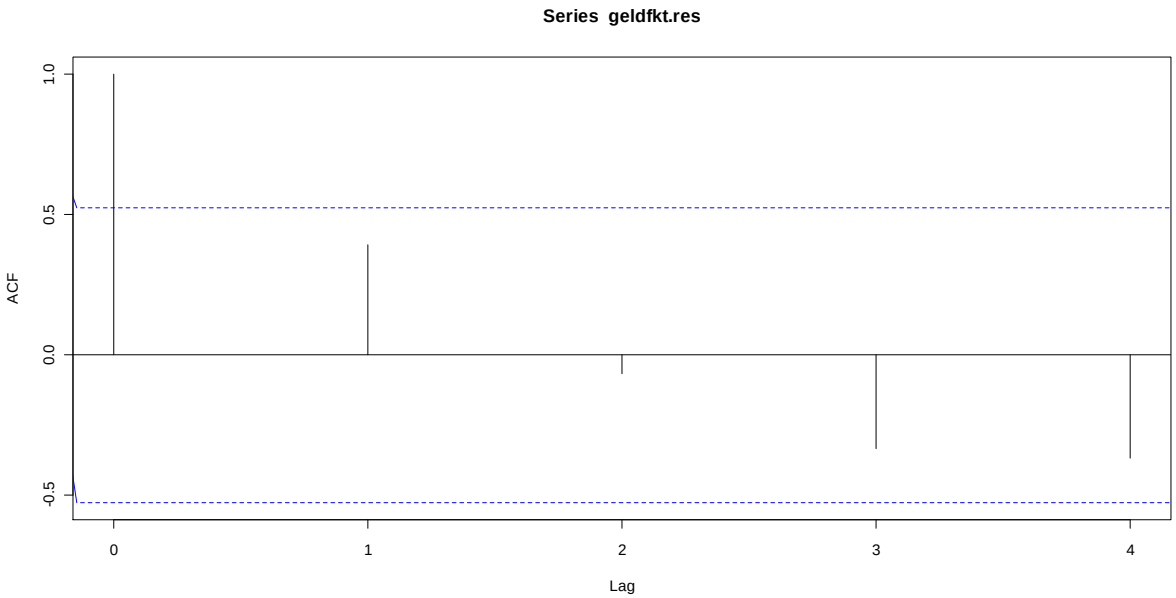
berechnen.

Mit der acf-Funktion lassen sich die Autokorrelationskoeffizienten der Geldnachfragefunktion geldfkt.lm bis zu einem vorgegebenen Lag grafisch darstellen:

```
>res.acf = acf(geldfkt.res, lag.max=4).
```

Die grafische Darstellung der Autokorrelationskoeffizienten in Abbildung 5.1 heißt Korrelogramm.

Abbildung 5.1: Korrelogramm der Residuen der Geldnachfragefunktion



Die beiden blau gepunkteten Linien kennzeichnen die Grenzen des 95%-Konfidenzintervalls für die Autokorrelationskoeffizienten. Abgesehen von der Eigenkorrelation der Residualreihe zum Lag 0 ($\hat{\rho}_0 = 1$) liegt kein Autokorrelationskoeffizient außerhalb der Konfidenzbänder, so dass für die Lags 1 bis 4 keine Signifikanz auf dem 5%-Niveau vorliegt.

Das acf-Objekt res.acf,

```
>class(res.acf)
[1] "acf",
```

enthält das Attribut acfmit den berechneten Autokorrelationskoeffizienten:

```
>str(res.acf)
List of 6
 $ acf  : num [1:5, 1, 1] 1 0.3942 -0.0652 -0.3318 -0.3668
 $ type  :chr "correlation"
 $ n.used: int 14
$ lag  : num [1:5, 1, 1] 0 1 2 3 4
```

```
$ series: chr "geldfkt.res"
$ snames: NULL
- attr(*, "class")= chr "acf".
```

Die Autokorrelationskoeffizienten können in dem Vektor r,

```
> r = res.acf$acf[-1]
>r
[1]  0.39421443 -0.06524385 -0.33176508 -0.36678718
```

gespeichert werden. Sie basieren ebenfalls auf der allgemeinen Formel (5.4). so dass sie mit den detailliert berechneten Autokorrelationskoeffizienten identisch sind.

Übungsaufgaben

5.1.1 Berechnen Sie detailliert die Autokorrelationskoeffizienten der Residuen des Energienachfragemodells 1 (energiefkt1.lm) für einen maximalen Lag von 4!

5.1.1 Erstellen Sie das Korrelogramm der Residuen des Energienachfragemodells 1 (energiefkt1.lm) für einen maximalen Lag von 6 und interpretieren Sie es!

5.1.2 Speichern Sie die in Aufg. 5.1.2 ermittelten Autokorrelationen 1. bis 6. Ordnung in einen Vektor r und stellen Sie sie unter Verwendung der barplot-Funktion grafisch dar!

5.2 Tests auf Autokorrelation

5.2.1 Durbin-Watson-Test

Die stochastische Abhängigkeit der Residuen mit einem Abstand von ein, zwei und mehr Perioden lässt sich mit dem Autokorrelationskoeffizienten messen. In der Ökonometrie wird die Signifikanz der Autokorrelation 1. Ordnung mit dem Durbin-Watson-Test oder Breusch-Godfrey-Test überprüft. Mit letzterem Test wird oft auch geprüft, ob die Störgröße eines ökonometrischen Modells einem autoregressiven Prozess höherer Ordnung folgt.

In R kann die Durbin-Watson-Statistik DW,

$$\text{(5.5)} \quad DW = \frac{\sum_{t=2}^{n}\left(\hat{u}_t - \hat{u}_{t-1}\right)^2}{\sum_{t=1}^{n}\hat{u}_t^2}$$

zwar direkt unter Verwendung der lag- und sum-Funktionen berechnet werden:

```
> DW.Z = sum((u-u_1)^2, na.rm=TRUE)
> DW.Z
[1] 0.03541408
> DW.N = sum(u^2)
```

```
> DW.N
[1] 0.03053513
> DW = DW.Z/DW.N
> DW
[1] 1.159781.
```

Doch sind die kritischen Werte hierzu nicht unmittelbar verfügbar.

Der Durbin-Watson-Test (DW-Test) lässt sich jedoch nach Laden des Paketes car,

```
>library(car)
Lade nötigesPaket: MASS
Lade nötiges Paket: nnet
Lade nötiges Paket: survival
Lade nötiges Paket: splines,
```

unter Verwendung der Funktion durbinWatsonTestdurchführen:

```
>geldfkt.DW = durbinWatsonTest(geldfkt.lm)
>geldfkt.DW
lag Autocorrelation D-W Statistic p-value
1      0.3942144     1.159781   0.026
 Alternative hypothesis: rho != 0.
```

Angesichts des p-Wertes von 0,026 wird die Nullhypothese fehlender Autokorrelation bei einem zweiseitigen Test (Voreinstellung: alternative="two.sided") auf einem Signifikanzniveau von 5% abgelehnt. Die Klasse und Struktur des Objekts geldfkt.DW wird durch Aufruf der R-Funktionen class und str angezeigt:

```
>class(geldfkt.DW)
[1] "durbinWatsonTest"
>str(geldfkt.DW)
List of 4
$ r         : num 0.394
 $ dw        : num 1.16
$ p         : num 0.026
$ alternative: chr "two.sided"
- attr(*, "class")= chr "durbinWatsonTest".
```

Übungsaufgaben

5.2.1.1 Berechnen Sie die Durbin-Watson-Statistik aus den Residuen des Energienachfragemodells 1 (energiefkt1.lm) und interpretieren Sie sie deskriptiv!

5.2.1.2 Testen Sie die Residuen des Energienachfragemodells 1 (energiefkt1.lm) mit dem Durbin-Watson-Test auf Autokorrelation (α=0,05)!

5.2.1.3 Welches Ergebnis des Durbin-Watson-Tests ergibt sich für die Residuen des Energienachfragemodells 3 (energiefkt3.lm) bei einem Signifikanzniveau von 5%?

5.2.2 Breusch-Godfrey-Test

Der Breusch-Godfrey-Test (BP-Test) ist ein Test auf Autokorrelation beliebiger Ordnung. Zur Prüfung der Residuen $\hat{u}_t$ der Geldnachfragefunktion,

```
> u
        1          2          3          4          5          6          7          8          9         10         11         12
-0.038320129  0.015242321  0.029300955  0.076635526  0.054757433 -0.055090280 -0.052332065 -
0.085319372 -0.009836130  0.004380105 -0.021695053 -0.001493488
       13         14
 0.073141298  0.010628878,
```

auf Autokorrelation 1. Ordnung, ist die Hilfsregression

(5.6) $\hat{u}_t = \alpha_0 + \alpha_1 \cdot \hat{u}_{t-1} + \mathbf{x}_t'\boldsymbol{\delta} + w_t$

mit $\mathbf{x}_t' = (x_{2t}\ x_{3t} \ldots x_{kt})$ und $\boldsymbol{\delta}_t' = (\delta_2\ \delta_3 \ldots \delta_k)$ durchzuführen. Bei Gültigkeit der Nullhypothese fehlender Autokorrelation 1. Ordnung,

(5.7) H_0: $\alpha_1 = 0$,

ist zu erwarten, dass der geschätzte Regressionskoeffizient für α_1 nicht signifikant von null verschieden ist. Speziell wird die Nullhypothese (5.7) verworfen, wenn die Prüfgröße

(5.8) $BG = n \cdot R^2$,

das (1-α)-Quantil der χ^2-Verteilung mit einem Freiheitsgrad überschreitet oder der p-Wert kleiner als das vorgegebene Signifkanzniveau α ist. R^2 ist hierin das Bestimmtheitsmaß der Hilfsregression.

Bei der Ausführung der Hilfsregression (5.6) ist darauf zu achten, dass in der zeitlich verzögerte Residuenreihe $\hat{u}_{t-1}$ der erste Wert auf null gesetzt wird ($\hat{u}_1 = 0$):

```
> u0 = 0
> u0_1 = ts(c(u0, u_1), start=1997, end=2010, freq=1)
> u0_1
Time Series:
Start = 1997
End = 2010
Frequency = 1
                1          2          3          4          5          6          7          8          9         10         11
 0.000000000  -0.038320129  0.015242321  0.029300955  0.076635526  0.054757433 -0.055090280 -
0.052332065 -0.085319372 -0.009836130  0.004380105 -0.021695053
       12         13
-0.001493488  0.073141298.
```

Mit den erklärenden Variablen lnBIPr (= x_2) und lnEURIBOR (= x_3) führt die OLS-Schätzung der Hilfsregression (5.6) zur Geldnachfragefunktion zu folgendem Ergebnis:

```
> auxreg.bg1 = lm(u ~ u0_1 + lnBIPr + lnEURIBOR)
>summary(auxreg.bg1)

Call:
lm(formula = u ~ u0_1 + lnBIPr + lnEURIBOR)

Residuals:
    Min      1Q   Median      3Q     Max
-0.07723 -0.02903  0.00710  0.02382  0.07412

Coefficients:
              Estimate   Std. Error   t value Pr(>|t|)
(Intercept)  -1.309e-03  1.019e+00  -0.001   0.999
u0_1          3.960e-01  2.912e-01   1.360   0.204
lnBIPr        9.516e-05  1.308e-01   0.001   0.999
lnEURIBOR     8.678e-04  3.106e-02   0.028   0.978

Residual standard error: 0.05076 on 10 degrees of freedom
Multiple R-squared: 0.1561,    Adjusted R-squared: -0.09707
F-statistic: 0.6166 on 3 and 10 DF,  p-value: 0.6198.
```

Die Prüfgröße BG nach Gleichung (5.8) nimmt den Wert

```
> R2 = summary(auxreg.bg1)$r.squared
> R2
[1] 0.1561036
> n
[1] 14
> BG1 = n*R2
> BG1
[1] 2.18545
```

an, der das 95%-Quantil der χ^2-Verteilung mit einem Freiheitsgrad unterschreitet:

```
> chikrit5 = qchisq(0.95, 1)
> chikrit5
[1] 3.841459.
```

Dementsprechend ist der p-Wert

```
> p.BG1 = 1 - pchisq(BG1, 1)
> p.BG1
[1] 0.1393203
```

größer als das vorgegebene Signifikanzniveau α von 5%, so dass die Nullhypothese fehlender Autokorrelation im Unterschied zum Durbin-Watson-Test mit dem BG(1)-Test nicht verworfen werden kann.

Man erhält dieses Ergebnis auch direkt mit der Funktion bgtest, die in dem Paket lmtest verfügbar ist:

```
>library(lmtest)
Lade nötiges Paket: zoo.
```

In der Funktion bgtest braucht bei einem Test auf Autokorrelation erster Ordnung nur das originäre Regressionsmodell in der R-Notation oder als lm-Objekt spezifiziert zu werden. Voreingestellt ist die Lag-Ordnung 1 und die Verwendung der χ^2-Statistik:

```
> geldfkt.BG1 = bgtest(geldfkt.lm)
> geldfkt.BG1

Breusch-Godfrey test for serial correlation of order 1

data:  geldfkt.lm
LM test = 2.1855, df = 1, p-value = 0.1393.
```

Das R-Objekt geldfkt.BG1 gehört zur Klasse htest,

```
>class(geldfkt.BG1)
[1] "htest",
```

und weist die folgende Struktur auf:

```
>str(geldfkt.BG1)
List of 5
 $ statistic: Named num 2.19
  ..- attr(*, "names")= chr "LM test"
 $ parameter: Named int 1
  ..- attr(*, "names")= chr "df"
 $ method   : chr "Breusch-Godfrey test for serial correlation of order 1"
$ p.value  :num 0.139
 $ data.name: chr "geldfkt.lm"
 - attr(*, "class")= chr "htest".
```

Mit den BG(2)- und BG(4)-Tests kann Autokorrelation höherer Ordnung der Residuen der Geldnachfragefunktion aufgedeckt werden:

```
> geldfkt.BG2 = bgtest(geldfkt.lm, order=2)
> geldfkt.BG2

Breusch-Godfrey test for serial correlation of order 2

data:  geldfkt.lm
```

```
LM test = 3.2049, df = 2, p-value = 0.2014

> geldfkt.BG4 = bgtest(geldfkt.lm, order=4)
> geldfkt.BG4

Breusch-Godfrey test for serial correlation of order 4

data:  geldfkt.lm
LM test = 6.1846, df = 4, p-value = 0.1858.
```

Aufgrund der Testergebnisse des Breusch-Godfrey-Tests ergeben sich keine Hinweise auf Autokorrelation zweiter und vierter Ordnung der Störgröße der Geldnachfragefunktion.

Übungsaufgaben

5.2.2.1 Führen Sie die Hilfsregression des Breusch-Godfrey-Tests für das Energienachfragemodell 1 (energiefkt1.lm) als Basis für einen

- Test auf Autokorrelation 1. Ordnung,
- Test auf Autokorrelation 2. Ordnung

durch!

5.2.2.2 Berechnen Sie auf der Grundlage der Schätzergebnisse der in Aufg. 5.2.2.1 durchgeführten Hilfsregressionen die Prüfgröße des

- BG(1)-Tests,
- BG(2)-Tests!

Testen Sie die BG(1)-Statistik (BG(2)-Statistik) unter Verwendung des kritischen Werts (p-Werts) (α=0,05)!

5.2.2.3 Vergleichen Sie die in Aufg. 5.2.2.2 ermittelten Testergebnisse mit den Resultaten der Funktion bgtest!

5.3 Das Cochran-Orcutt-Verfahren

Bei autokorrelierten Störvariablen verlieren die gewöhnlichen Kleinst-Quadrate-Schätzer (OLS-Schätzer) der Regressionskoeffizienten ihre Effizienzeigenschaft. Das bedeutet, dass ihre geschätzten Standardfehler verzerrt sind, so dass die Signifikanztests nicht mehr valide durchgeführt werden können. Bei Anwendung der verallgemeinerten Methode der kleinsten Quadrate (generalised least-squares (GLS)) lassen sich die geschätzten Regressionskoeffizienten wieder valide testen.

Hier soll beispielhaft die GLS-Schätzung in Form des Cochran-Orcutt-Verfahrens für die Geldnachfragefunktion bei Verwendung des logarithmierten Bruttoinlandsprodukts lnBIPr und des logarithmierten Zinssatzes lnEURIBOR als erklärende Variablen erfolgen.

Die GLS-Schätzung nach dem Cochran-Orcutt-Verfahren entspricht einer OLS-Schätzung des transformierten Regressionsmodells

(5.9) $y_t^* = \mathbf{x}_t^{*'} \boldsymbol{\beta} + v_t$

mit

(5.10) $$\begin{aligned} y_t^* &= y_t - \hat{\phi} \cdot y_{t-1}, \quad t = 2,3,\ldots,n \\ \mathbf{x}_t^{*'} &= \mathbf{x}_t' - \hat{\phi} \cdot \mathbf{x}_{t-1}' \end{aligned}$$

Um die transformierten Variablenwerte zu berechnen, wird der geschätzte autoregressive Parameter $\hat{\phi}$ benötigt, den man aus der OLS-Schätzung des Markov-Prozesses

(5.11) $u_t = \phi \cdot u_{t-1} + v_t$

erhält. Hierzu wird von den beim Durbin-Watson-Test (Abschn. 5.2.1) definierten Zeitreihen der Residuen u und u_1 Gebrauch gemacht:

```
>u
Time Series:
Start = 1997
End = 2010
Frequency = 1
        1          2          3          4          5          6          7          8          9         10         11         12
-0.038320129  0.015242321  0.029300955  0.076635526  0.054757433 -0.055090280 -0.052332065 -
0.085319372 -0.009836130  0.004380105 -0.021695053 -0.001493488
       13         14
0.073141298  0.010628878
> u_1
Time Series:
Start = 1998
End = 2011
Frequency = 1
        1          2          3          4          5          6          7          8          9         10         11         12
-0.038320129  0.015242321  0.029300955  0.076635526  0.054757433 -0.055090280 -0.052332065 -
0.085319372 -0.009836130  0.004380105 -0.021695053 -0.001493488
       13         14
 0.073141298  0.010628878.
```

Die OLS-Schätzung des Markov-Prozesses (5.11),

```
> u.AR1 = lm(u[-1] ~ 0+u_1[-14])
> u.AR1

Call:
lm(formula = u[-1] ~ 0 + u_1[-14])
```

```
Coefficients:
u.lag1[-14]
   0.3957,
```

ergibt $\hat{\phi} = 0.3957$. Ohne die unterschiedliche Datierung der Zeitreihenobjekte u und u_1 explizit berücksichtigen zu müssen, lässt sich die Kleinst-Quadrate-Schätzung des autoregressiven Prozesses 1. Ordnung nach ihrer Verknüpfung zu einem Matrix-objekt durchführen:

```
> U = cbind(u, u_1)
> U
Time Series:
Start = 1997
End = 2011
Frequency = 1
             u            u_1
1997 -0.038320129        NA
1998  0.015242321 -0.038320129
1999  0.029300955  0.015242321
2000  0.076635526  0.029300955
2001  0.054757433  0.076635526
2002 -0.055090280  0.054757433
2003 -0.052332065 -0.055090280
2004 -0.085319372 -0.052332065
2005 -0.009836130 -0.085319372
2006  0.004380105 -0.009836130
2007 -0.021695053  0.004380105
2008 -0.001493488 -0.021695053
2009  0.073141298 -0.001493488
2010  0.010628878  0.073141298
2011        NA     0.010628878
>u.AR1 = lm(U[,1] ~ 0+U[,2])
> u.AR1

Call:
lm(formula = U[, 1] ~ 0 + U[, 2])

Coefficients:
U[, 2]
0.3957  .
```

Mit diesem Schätzwert von 0,3957 für den autoregressiven Parameter ϕ lassen sich die Transformationen (5.10) für die Modellvariablen lnM1GERr, lnBIPr und lnEURIBOR durchführen:

```
>phi = u.AR1$coeff
```

```
>phi
u.lag1[-14]
  0.3956783

> lnM1GERr = ts(lnM1GERr, start=1997, freq=1)
> lnM1GERr
Time Series:
Start = 1997
End = 2010
Frequency = 1
 [1] 6.087797 6.192710 6.277282 6.300584 6.357617 6.341364 6.431895 6.479124 6.587274 6.636419
6.690966 6.751733 6.962177 7.056891
> lnM1GERrs98_10 = lnM1GERr - phi*lag(lnM1GERr, k=-1)
> lnM1GERrs98_10
Time Series:
Start = 1998
End = 2010
Frequency = 1
 [1] 3.783901 3.826961 3.816799 3.864612 3.825793 3.922754 3.934162 4.023625 4.029977 4.065079
4.104262 4.290663 4.302108

>lnBIPr = ts(lnBIPr, start=1997, freq=1)
>lnBIPr

Time Series:
Start = 1997
End = 2010
Frequency = 1
 [1] 7.501971 7.532195 7.554644 7.571288 7.608628 7.637230 7.655220 7.688139 7.707242 7.749784
7.814343 7.840443 7.811188 7.859313
> lnBIPrs98_10 = lnBIPr - phi*lag(lnBIPr, k=-1)
> lnBIPrs98_10
Time Series:
Start = 1998
End = 2010
Frequency = 1
 [1] 4.563828 4.574317 4.582079 4.612833 4.626660 4.633334 4.659134 4.665212 4.700195 4.747922
4.748476 4.708894 4.768595

>lnEURIBOR = ts(lnEURIBOR, start=1997, freq=1)
>lnEURIBOR
Time Series:
Start = 1997
End = 2010
Frequency = 1
 [1] 1.2029723 1.2641267 1.0885620 1.4793292 1.4492692 1.1999648 0.8458683 0.7466879 0.7839015
1.1249296 1.4539530 1.5325569 0.2070142
```

```
[14] -0.2107210
> lnEURIBORs98_10 = lnEURIBOR - phi*lag(lnEURIBOR, k=-1)
> lnEURIBORs98_10
Time Series:
Start = 1998
End = 2010
Frequency = 1
 [1] 0.7881366 0.5883744 1.0486088 0.8639306 0.6265204 0.3710682 0.4119962 0.4884533 0.8147567
1.0088427 0.9572591 -0.3993854 -0.2926321.
```

Die Kleinst-Quadrate-Schätzung der Geldnachfragefunktion mit den transformierten Variablen (Cochran-Orcutt-Methode) führt zu folgendem Ergebnis:

```
> geldfkt.co = lm(lnM1GERrs98_10 ~ lnBIPrs98_10+lnEURIBORs98_10)
>summary(geldfkt.co)

Call:
lm(formula = lnM1GERrs98_10 ~ lnBIPrs98_10 + lnEURIBORs98_10)

Residuals:
    Min      1Q  Median      3Q     Max
-0.08430 -0.01421  0.01305  0.02013  0.06827

Coefficients:
                   Estimate  Std. Error t value  Pr(>|t|)
(Intercept)        -4.56129    0.99328  -4.592  0.000992 ***
lnBIPrs98_10        1.85211    0.21176   8.746  5.35e-06 ***
lnEURIBORs98_10 -0.15595    0.03213  -4.853  0.000668 ***
---
Signif.codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.04806 on 10 degrees of freedom
Multiple R-squared: 0.9355,    Adjusted R-squared: 0.9226
F-statistic: 72.53 on 2 and 10 DF,  p-value: 1.115e-06
```

Während sich die geschätzte Einkommenselastizität nur geringfügig von rd. 2 auf 1,85 verringert hat, wird der Einfluss des Zinssatzes auf die Geldnachfrage bei einem Wert von -0,156 statt -0,147 geringfügig stärker eingeschätzt. Das absolute Glied in Bezug auf die originäre Regressionsgleichung der logarithmierten Variablen fällt mit -7,548 (=-4,56129/(1-0,39568)) etwas niedriger aus als bei der OLS-Schätzung (-8,667). Alle geschätzten Regressionskoeffizienten sind nach wie vor auf dem 1%-Niveau statistisch gesichert.

Wie der Durbin-Watson-Test zeigt,

```
>library(car)
Lade nötigesPaket: MASS
Lade nötiges Paket: nnet
```

```
Lade nötiges Paket: survival
Lade nötiges Paket: splines
>geldfkt.DWco
lag Autocorrelation D-W Statistic p-value
1      0.2184553     1.544104    0.14
 Alternative hypothesis: rho != 0,
```

ist für die Residuen der transformierten Geldnachfragefunktion keine Autokorrelation mehr statistisch nachweisbar.

Übungsaufgaben

5.3.1 Bereinigen Sie die Zeitreihen GASV und GASPR des Energienachfragemodells 1 von Autokorrelation durch Anwendung der Transformation des Cochran-Orcutt-Verfahrens!

5.3.2 Ermitteln Sie die GLS-Schätzer der Regressionskoeffizienten des Geldnachfragemodells 1 mit dem Cochran-Orcutt-Verfahren! Vergleichen Sie die GLS-Schätzer mit dem gewöhnlichen Kleinst-Quadrate-Schätzer!

5.3.3 Zu wie viel Prozent wird die Varianz des Erdgasverbrauchs (GASV) bei der in Aufg. 5.3.2 durchgeführten GLS-Schätzung durch den Erdgaspreis (GASPR) erklärt?

5.3.4 Sind die GLS-Residuen des Energienachfragemodells 1 (s. Aufg. 5.3.2) auf der Basis des Durbin-Watson-Tests frei von Autokorrelation (α=0,05)?

5.4 Heteroskedastizitäts- und Autokorrelations-konsistente (HAC) Standardfehler

Wie im Falle der Heteroskedastizität sind die OLS-Schätzer der Regressionskoeffizienten des ökonometrischen Standardmodells bei autokorrelierten Störvariablen weiterhin erwartungstreu, während ihre Standardfehler verzerrt sind. Daher bietet sich auch hier als Alternative zu einer GLS-Schätzung z.B. durch Anwendung des Cochran-Orcutt-Verfahrens eine Korrektur der gängigen Standardfehler zur Behebung des Modelldefekts an. Newey und West (1987) haben hierzu einen Heteroske-dastizitäts- und Autokorrelations-konsistenten (HAC) Schätzer der Varianz-Kovarianz-Matrix $\mathrm{Cov}(\hat{\boldsymbol{\beta}})$ entwickelt, der ohne Berücksichtigung von Autokorrelation (maximaler Lag L=0) mit Whites Heteroskedastizitäts-konsistentem (HC) Schätzer (4.6) zusammenfällt.

In R ist die Funktion NeweyWest zur Berechnung des Newey/West HAC-Schätzers für $\mathrm{Cov}(\hat{\boldsymbol{\beta}})$ im Paket sandwich verfügbar:

```
> library(sandwich)
Lade nötiges Paket: zoo
```

Will man die bei der Geldnachfragefunktion potenziell vorhandene Autokorrelation ohne eine vorherige Ausschaltung direkt bei der Schätzung der Varianz-Kovarianz-Matrix $\mathrm{Cov}(\hat{\boldsymbol{\beta}})$ berücksichtigen, ist das Argument prewhite beim Aufruf der Funktion NeweyWest gleich FALSE zu setzen. Unter Verwendung der automatischen Selektionsprozedur der maximalen Laglänge p von Newey und West (1994) ergibt sich der HAC-Schätzer

```
> covb.nw = NeweyWest(geldfkt.lm, prewhite=FALSE)
> covb.nw
              (Intercept)        lnBIPr        lnEURIBOR
(Intercept)   0.358638890  -0.0463005546   0.0020541166
lnBIPr       -0.046300555   0.0059945391  -0.0003583977
lnEURIBOR     0.002054117  -0.0003583977   0.0005830118,
```

der die für große Stichproben relevante Eigenschaft der Konsistenz besitzt. Wie sich zeigt, liegt dem HAC-Schätzer covb.nw eine maximale Laglänge von 1 zugrunde:

```
> NeweyWest(geldfkt.lm, lag=1, prewhite=FALSE)
              (Intercept)        lnBIPr        lnEURIBOR
(Intercept)   0.358638890  -0.0463005546   0.0020541166
lnBIPr       -0.046300555   0.0059945391  -0.0003583977
lnEURIBOR     0.002054117  -0.0003583977   0.0005830118.
```

Bei Anpassung des Newey/West HAC-Schätzers für $\mathrm{Cov}(\hat{\boldsymbol{\beta}})$ an den geringen Um-fang der vorliegende Stichprobe durch Multiplikation mit dem Faktor n/(n-k) (adjust =TRUE), erhält man den adjustierten HAC-Schätzer

```
> covb.nwa = NeweyWest(geldfkt.lm, prewhite=FALSE, adjust=TRUE)
> covb.nwa
              (Intercept)        lnBIPr        lnEURIBOR
(Intercept)   0.45644950  -0.0589279786   0.0026143302
lnBIPr       -0.05892798   0.0076294134  -0.0004561425
lnEURIBOR     0.00261433  -0.0004561425   0.0007420150,
```

bei dem das automatische Selektionsverfahren ebenfalls eine maximale Laglänge von 1 ermittelt hat.

Die Signifikanz der Regressionskoeffizienten könnte hier wie im Falle der Heteroskedastizität nach Extraktion der Varianzschätzer der geschätzten Regressionskoeffizienten robust getestet werden. An dieser Stelle greifen wir hierzu jedoch direkt auf die Funktion coeftest des R-Pakets lmtest zurück. Nach Laden dieses Pakets,

```
> library(lmtest),
```

ist die Funktion coeftest, die auf die Newey/West-Prozedur zugreifen kann, verfügbar. Bei Vernachlässigung des Anpassungsfaktors n/(n-k) für kleine Stichproben ergeben sich unter Verwendung der Normalapproximation (z-Test) die Testergebnisse

```
> coeftest.NWz = coeftest(geldfkt.lm, df=Inf, vcov=NeweyWest(geldfkt.lm, prewhite=FALSE))
> coeftest.NWz

z test of coefficients:

              Estimate   Std. Error   z value   Pr(>|z|)
(Intercept)  -8.667211   0.598865   -14.4727   < 2.2e-16 ***
lnBIPr        1.995417   0.077424    25.7724   < 2.2e-16 ***
lnEURIBOR    -0.146495   0.024146    -6.0672    1.302e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1.
```

Unter Berücksichtigung des Anpassungsfaktors n/(n-k) ergeben sich dagegen bei Spezifikation der Anzahl der Freiheitsgrade, n-k=14-3=11, folgende Ergebnisse der t-Tests:

```
> coeftest.NWt = coeftest(geldfkt.lm, df=n-k, vcov=NeweyWest(geldfkt.lm, prewhite=FALSE, adjust=
TRUE))
> coeftest.NWt

t test of coefficients:

              Estimate   Std. Error   t value   Pr(>|t|)
(Intercept)  -8.667211   0.675610   -12.829   5.839e-08 ***
lnBIPr        1.995417   0.087347    22.845   1.278e-10 ***
lnEURIBOR    -0.146495   0.027240    -5.378   0.0002241 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1.
```

Die bei Gültigkeit der Standardannahmen erzielten hochsignifikanten Testergebnisse bleiben bei Verwendung der Heteroskedastizitäts- und Autokorrelations-konsistenten Standardfehler unabhängig von einer Anpassung an den Umfang der Stichprobe bestehen. Doch verringert sich der HAC-Standardfehler des geschätzten Regressionskoeffizienten des Einkommens bei Berücksichtigung Stichprobenanpassung im Vergleich zu der unkorrigierten Standardabweichung um den Faktor 1,55. Im Falle des Zinssatzes tritt dagegen eine Reduktion um den Faktor 1,18 ein.

Übungsaufgaben

5.4.1 Geben Sie den Heteroskedastizitäts- und Autokorrelations-konsistenten (HAC) Schätzer der Varianz-Kovarianz-Matrix $\mathrm{Cov}(\hat{\boldsymbol{\beta}})$ nach Newey und West für das OLS-geschätzte Energienachfragemodell 1 (energiefkt1.lm) mit und ohne die Anpassung für endliche Stichproben unter Einsatz der automatischen Selektionsprozedur für die maximale Laglänge an!

5.4.2. Welche maximalen Laglänge hat die automatische Selektionsprozedur nach Newey/West in Aufg. 5.4.1 ermittelt?

5.4.3 Bestimmen Sie unter Verwendung der in Aufg. 5.4.1 des HAC-Schätzers der Varianz-Kovarianz-Matrix $\mathrm{Cov}(\hat{\boldsymbol{\beta}})$ mit der Anpassung an endliche Stichproben die Standardfehler der geschätzten Regressionskoeffizienten! Vergleichen Sie sie mit den unkorrigierten Standardfehlern der OLS-Schätzung!

5.4.4 Prüfen Sie die Signifikanz der geschätzten Regressionskoeffizienten des Energienachfragemodells 1 unter Verwendung der in Aufg. 5.4.3 ermittelten HAC-Standardfehler mit Hilfe des t-Tests (α=0,05)!

5.4.5 Testen Sie die Signifikanz der Regressionskoeffizienten des Energienachfragemodells 1 (α=0,05) mit der Funktion coeftest mit Hilfe des

- z-Tests ohne Anpassung an endliche Stichproben,
- t-Tests mit Anpassung an endliche Stichproben!

6. Modelle mit qualitativen Variablen

6.1 Modelle mit qualitativen Regressoren

In den Standardmodellen der Regressionsanalyse werden Beziehungen zwischen metrisch skalierten Variablen untersucht. Jedoch gibt es Anwendungen, bei denen auch qualitative Variablen von Bedeutung sind. So beobachtet man z.B. bei bestimmten Produkten ein unterschiedliches Nachfrageverhalten von Männern und Frauen. Auch kann es sich bei spezifischen Studien als erforderlich erweisen, ein unterschiedliches Konsum- oder Sparverhalten der sozialen Schichten der Bevölkerung zu berücksichtigen. In Regionalstudien sind in der Regel räumliche Effekte zu berücksichtigen und bei Zeitreihenanalyse temporale Effekte.

All diese Beispiele beinhalten qualitative Einflussgrößen, die in geeigneter Form in ein Regressionsmodell zu integrieren sind. Im einfachsten Fall bestehen die qualitativen Variablen aus zwei Zuständen wie z.B. beim Geschlecht „männlich“ und „weiblich“ der Erwerbstätigkeit „erwerbstätig“ und „nicht erwerbstätig“. Unter dieser Voraussetzung lassen sich die qualitativen Regressoren in einfacher Form durch eine Dummy-Variable (0,1-Variable) in einem ökonometrischen Modell berücksichtigen. Welcher Wert „0“ oder „1“ für eine Kategorie wie z.B. „männlich“ oder „weiblich“ verwendet wird, ist dabei unerheblich. Doch wird man eine besonders interessierende Kategorie z.B.

„Kauf eines Produkts A“ mit „1“ und die nicht im Vordergrund stehende Kategorie z.B. „Nichtkauf des Produkts A“ mit „0“ kodieren. Der Regressionskoeffizient der Dummy-Variablen misst den Aufschlag (+) oder Abschlag (-), der im Falle des Vorhandenseins der interessierende Eigenschaft („1“) bei der Ermittlung des Regressionswerte er betreffenden statistischen Einheiten zu machen ist.

Als Beispiel für die Berücksichtigung eines qualitativen Regressors in einem ökonometrischen Modell sei die Nachfrage q_t nach einer Güterart in Abhängigkeit von den Gesamtausgaben x_t einzelner Verbraucher t betrachtet. Sofern die Nachfrage geschlechtsspezifisch ist, lässt sich die binäre Variable Geschlecht in Form einer Dummy-Variablen,

(6.1) $$g_t = \begin{cases} 1, & \text{falls Frau} \\ 0, & \text{falls Mann} \end{cases},$$

in der stochastischen Nachfragefunktion

(6.2) $$q_t = \beta_1 + \beta_2 \cdot x_t + \beta_3 \cdot g_t + u_t$$

einbeziehen.

Zur Schätzung des Nachfragemodells (6.2) werden die Daten von acht Käufern über ihre Nachfrage nach einem Heimwerkerartikel (Nachfrage), ihre Gesamtausgaben (GesAusg) und ihr Geschlecht (Geschl) herangezogen. Sie liegen zusammen mit Angaben über ihre soziale Stellung (SozSt: 1 niedrig, 2 mittel, 3 hoch) in der Datei Kaeufer.csv vor.

Nach Einlesen der Daten aus der CSV-Datei Kaeufer.csv in das data-frame-Objekt Kauefer.df,

```
> Kaeufer.df = read.csv("Kaeufer.csv")
> Kaeufer.df
  Kaeufer Nachfrage GesAusg Geschl SozSt
1       1         9      48      0     2
2       2         1      30      1     1
3       3         4      23      0     1
4       4         8      56      0     3
5       5        14      72      0     3
6       6         4      32      0     2
7       7        12      42      0     1
8       8         2      43      1     3,
```

lässt sich die Nachfragefunktion unter Verwendung der lm-Funktion in R mit der gewöhnlichen Methode der kleinsten Quadrate schätzen:

```
> Kaeufer.lm1 = lm(Nachfrage~GesAusg+Geschl, data=Kaeufer.df).
```

Aus dem Regressionsoutput

```
> summary(Kaeufer.lm1)

Call:
lm(formula = Nachfrage ~ GesAusg + Geschl, data = Kaeufer.df)

Residuals:
     1       2       3       4       5       6       7       8
 0.03554 0.70761 -0.31983 -2.45075 0.57668 -1.99190 4.15025 -0.70761

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.04676   2.88903   0.016   0.9877
GesAusg     0.18579   0.05975   3.109   0.0266 *
Geschl     -5.32793   2.02666  -2.629   0.0466 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.393 on 5 degrees of freedom
Multiple R-squared:  0.8182,    Adjusted R-squared:  0.7455
F-statistic: 11.25 on 2 and 5 DF,  p-value: 0.0141
```

geht hervor, dass das Geschlecht einen signifikanten Einfluss auf die Nachfrage nach dem Heimwerkerartikel ausübt. Der Effekt des qualitativen Regressors Geschl ist bei einem p-Wert von 0,0466 auf dem 5%-Niveau signifikant. Gleiches trifft auf die Gesamtausgaben (GesAusg) zu, deren p-Wert 0,0266 beträgt. Beide Regressoren erklären 81,82% der Varianz der Nachfrage nach dem Heimwerkerartikel (Nachfrage). Bei einem F-Wert von 11,25 ist der durch die Nachfragefunktion gegebene Gesamtzusammenhang statistisch gesichert (p = 0,0141).

Während der geschätzte Regressionskoeffizient der Variablen GesAusg den erwarteten Anstieg der Nachfrage nach Heimwerkerartikeln (Nachfrage) bei einer Steigerung der individuellen Gesamtausgaben um 1 Einheit mit 0,186 Einheiten beziffert, spiegelt der Wert des Regressionskoeffizienten der Dummy-Variablen Geschl das unterschiedliche Nachfrageniveau zwischen Männern und Frauen wider. Im Mittel ist die Nachfrage der Frauen nach dem Heimwerkerartikel um 5,328 Einheiten geringer als diejenige von Männern:

Nachfragefunktion von Männern:

$$\hat{y}_{männl} = 0{,}047 + 0{,}186 \cdot x$$

Nachfragefunktion von Frauen:

$$\hat{y}_{weibl} = 0{,}047 + 0{,}186 \cdot x - 5{,}328 \cdot g.$$

Anstelle einer Parallelverschiebung kann jedoch auch eine Drehung der Nachfragefunktion von Relevanz sein. In diesem Fall ist die Annahme einer gleichen marginalen Konsumneigung β_2 der männlichen und weiblichen Käufer aufzugeben. Der binäre Regressor Geschlecht (g_t) ist dann in Form einer Interaktion mit den Gesamtausgaben (x_t) in das Nachfragemodell einzubeziehen:

(6.3) $$q_t = \beta_1 + \beta_2 \cdot x_t + \beta_3 \cdot x_t \cdot g_t + u_t.$$

Unter Einbeziehung des Interaktionsterms aus Gesamtausgaben und Geschlecht in der Form GesAusg:Geschl lässt sich die stochastische Nachfragefunkion (6.3) wiederum mit der lm-Funktion schätzen:

```
> Kaeufer.lm2 = lm(Nachfrage~GesAusg+GesAusg:Geschl, data=Kaeufer.df).
> summary(Kaeufer.lm2)

Call:
lm(formula = Nachfrage ~ GesAusg + GesAusg:Geschl, data = Kaeufer.df)

Residuals:
     1        2        3        4        5        6        7        8
 0.02597 -0.20056 -0.13266 -2.52327  0.37825 -1.87555  4.18790  0.13992

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)       -0.32141    2.77990  -0.116   0.9125
GesAusg            0.19366    0.05797   3.341   0.0205 *
GesAusg:Geschl    -0.14292    0.05276  -2.709   0.0423 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.351 on 5 degrees of freedom
Multiple R-squared:  0.8245,    Adjusted R-squared:  0.7543
F-statistic: 11.74 on 2 and 5 DF,  p-value: 0.01291.
```

Hier zeigt sich ein signifikanter Einfluss der Gesamtausgaben (GesAusg) und der Interaktion von Gesamtausgaben und Geschlecht (GesAusg:Geschl) auf die Nachfrage nach dem Heimwerkerartikel (Nachfrage) auf dem 5%-Niveau. Der geschätzte Regressionskoeffizient des Interaktionsterms GesAusg:Geschl weist aus, dass die Grenzneigung zum Konsum des Heimwerkerartikels bei Frauen um 0,143 niedriger ist als bei Männern:

Nachfragefunktion von männlichen Käufern (Grenzneigung zum Konsum: 0,194):

$$\hat{y}_{männl} = 0{,}047 + 0{,}194 \cdot x$$

Nachfragefunktion von weiblichen Käufern (Grenzneigung zum Konsum: 0,194-0,143=0,051):

$$\hat{y}_{weibl} = 0{,}047 + 0{,}051 \cdot x\ .$$

Bei qualitativen Regressoren mit drei Kategorien müssen zwei Dummy-Variablen gebildet werden, um ihren Einfluss auf die abhängige Variable in einem ökonometrischen Modell zu quantifizieren. Ersetzt man z.B. die qualitative Variable „Geschlecht“ in dem Nachfragemodell durch die qualitative Variable „Soziale Stellung“ mit den Kategorien niedrig, mittel und hoch, könnte man z.B. für letztere beiden Kategorien die Dummy-Variablen

$$s_{2t} = \begin{cases} 1, & \text{falls mittlere soziale Stellung} \\ 0, & \text{sonst} \end{cases}$$

und

$$s_{3t} = \begin{cases} 1, & \text{falls hohe soziale Stellung} \\ 0, & \text{sonst} \end{cases}$$

bilden.

Der qualitative Regressor “Soziale Stellung” wird in dem multiplen Regressionsmodell durch die beiden Dummy-Variable s_2 und s_3 ersetzt, die als “Mittlere soziale Stellung” und “Hohe soziale Stellung” deklariert werden:

(6.4) $q_t = \beta_1 + \beta_2 \cdot x_t + \beta_3 \cdot s_{2t} + \beta_4 \cdot s_{3t} + u_t\,.$

Die Kategorie einer niedrigen sozialen Stellung fungiert dann als Referenzkategorie. Falls beide Dummy-Variablen bei einem Käufer oder Haushalt den Wert 0 annehmen, liegt eine niedrige soziale Stellung vor.

In R braucht die Bildung von Dummy-Variablen im Falle eines polytomen Regressors jedoch nicht explizit vorgenommen zu werden. Vielmehr erfolgt sie automatisch, wenn er in der lm-Funktion als Objekt vom Typ factor spezifiziert wird. So wird das Nachfragemodell (6.4) für die Käuferdaten mit den Dummy-Variablen s_2 und s_3 für eine mittlere und hohe soziale Stellung ohne ihre vorherige Definition geschätzt, wenn die polytome Variable SozSt als Faktor factor(SozSt) in die lm-Funkton aufgenommen wird:

```
> Kaeufer.lm3 = lm(Nachfrage~GesAusg+factor(SozSt), data=Kaeufer.df)
> summary(Kaeufer.lm3)

Call:
lm(formula = Nachfrage ~ GesAusg + factor(SozSt), data = Kaeufer.df)
```

```
Residuals:
     1       2       3       4       5       6       7       8
-0.7849 -3.9823  1.8920  0.4106 -0.1593  0.7849  2.0903 -0.2514

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)     -7.33621    3.27726  -2.239   0.0888 .
GesAusg          0.41062    0.09278   4.426   0.0115 *
factor(SozSt)2  -2.58848    2.42241  -1.069   0.3455
factor(SozSt)3  -8.06897    3.12104  -2.585   0.0610 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.515 on 4 degrees of freedom
Multiple R-squared:  0.8394,    Adjusted R-squared:  0.7189
F-statistic: 6.968 on 3 and 4 DF,  p-value: 0.0457.
```

Erneut erweisen sich die Gesamtausgaben (GesAusg) auf dem 5%-Niveau als signifikant. Die beiden negativen Regressionskoeffizienten der Dummy-Variablen "Mittlere soziale Stellung" (factor(SozSt)2) und "Obere soziale Stellung" (factor(SozSt)3) deuten darauf hin, dass der Heimwerkerartikel tendenziell eher von Haushalten der unteren sozialen Stellung nachgefragt wird. Da zusätzlich $\left|\hat{\beta}_3\right| < \left|\hat{\beta}_4\right|$ gilt, kann außerdem gefolgert werden, dass die Nachfrage mit einer höheren Position in der sozialen Stellung zurückzugehen tendiert.

Danach sinkt die Nachfrage nach dem Heimwerkerartikel bei Haushalten der Mittelschicht gegenüber Haushalten der unteren Schicht um 2,588 Einheiten. Die Nachfrage der Haushalte der Oberschicht sinkt gegenüber den Haushalten der mittleren Schicht um 5,481 (=8,069-2,588) Einheiten und gegenüber den Haushalten der unteren Schicht um 8,069 Einheiten.

Die schichtspezifischen Nachfragefunktionen lauten:

für untere soziale Stellung: $\hat{y}_{s1} = -7{,}336 + 0{,}411 \cdot x_t + u_t$

für mittlere soziale Stellung: $\hat{y}_{s2} = -9{,}924 + 0{,}411 \cdot x_t + u_t$

für obere soziale Stellung: $\hat{y}_{s1} = -17{,}994 + 0{,}411 \cdot x_t + u_t$.

Da $\hat{\beta}_3$ jedoch auf keinem der gängigen Niveaus signifikant ist (p=0,346), ist die Verringerung der Nachfrage der Haushalte der Mittelschicht im Vergleich zu den Haushalten der unteren Schicht nicht statisch gesichert. Bei einem p-Wert von 0,061 lässt sich jedoch der Einfluss der oberen sozialen Schicht auf die Nachfrage als schwach signifikant bezeichnen.

Übungsaufgaben

6.1.1 Gegeben sind Angaben von sechs Erwerbspersonen über ihre Einkommen (Einkommen), ihr Alter (Alter) und ihre Stellung im Beruf (Beruf: 1 selbständig, 0 nicht selbständig):
> Einkommen = c(2400, 2700, 2800, 3400, 3200, 3500)
> Alter = c(28, 43, 33, 52, 38, 58)
> Beruf = c(0, 0, 1, 0, 1, 0).
Geben Sie die arithmetischen Mittelwerte der Variablen Einkommen und Alter und die Tabellen der absoluten und relativen Häufigkeiten der Stellung im Beruf mit Hilfe der verfügbaren R-Funktionen an!

6.2.2 Regressieren Sie das Einkommen auf das Alter und die Stellung im Beruf (Daten s. Aufg. 6.1.1)! Geben Sie die Regressionsfunktionen für die beiden Gruppen der Selbständigen und Nicht-Selbständigen an und interpretieren Sie sie!

6.2.3 Regressieren Sie das Einkommen auf das Alter und die Interaktion zwischen alter und Stellung im Beruf (Daten s. Aufg. 6.1.1) und interpretieren Sie den mit der Funktion summary erzeugten Regressionsoutput!

6.2.4 Ersetzen Sie in Aufg. 6.1.1 die binäre Variable „Stellung im Beruf“ durch die polytome Variable „Beruf“ mit den beobachteten kategorialen Ausprägungen 2, 1, 3, 1, 3, 2! Regressieren Sie das Einkommen auf die Alters- und Berufsvariable und interpretieren Sie die mit der OLS-Methode geschätzte Regressionsfunktion!

6.2 Strukturbruchtest

In einer Volkswirtschaft treten von Zeit zu Zeit Ereignisse auf, die ihre Struktur in bestimmter Hinsicht dauerhaft verändern. Der Ökonometriker spricht in diesem Fall von einem Strukturbruch. Vor dem Strukturbruch in der Periode n_1+1 sind die Einflüsse der exogenen Variablen x_{jt}, j=1,2,…,k, auf die endogene Variable y_t durch die Regressionskoeffizienten β_j^1, j=1,2,…,k, gegeben,

$$(6.5) \quad y_t = \sum_{j=1}^{k} \beta_j^1 \cdot x_{jt} + u_t, \quad t = 1,2,\ldots n_1,$$

während die Abhängigkeit nach dem Strukturbruch durch die Regressionskoeffizienten β_j^2, j=1,2,…,k, gekennzeichnet ist:

$$(6.6) \quad y_t = \sum_{j=1}^{k} \beta_j^2 \cdot x_{jt} + u_t, \quad t = n_1+1, n_1+2, \ldots n.$$

Für den gesamten Beobachtungszeitraum kann das ökonometrische Modell unter Verwendung der Dummy-Variablen d_1 und d_2,

$$d_{1t} = \begin{cases} 1 & \text{für } t = 1,2,\ldots,n_1 \\ 0 & \text{für } t = n_1+1, n_1+2,\ldots,n \end{cases}$$

und

$$d_{2t} = \begin{cases} 0 & \text{für } t = 1,2,\ldots,n_1 \\ 1 & \text{für } t = n_1+1, n_1+2,\ldots,n \end{cases}$$

die dem Strukturbruch Rechnung tragen sollen, wie folgt formuliert werden:

$$(6.7) \quad y_t = \sum_{j=1}^{k} \beta_j^1 \cdot x_{jt} \cdot d_{1t} + \sum_{j=1}^{k} \beta_j^2 \cdot x_{jt} \cdot d_{2t} + u_t, \quad t = 1,2,\ldots,n.$$

Auf diese Weise wird der Tatsache Rechnung getragen, dass sich bei einem Strukturbruch alle Regressionskoeffizienten verändern können.

Mit dem Chow-Test kann geprüft werden, ob ein Ereignis tatsächlich einen Strukturbruch zur Folge gehabt hat oder nicht. Ein Strukturbruch liegt vor, wenn sich ein oder mehrere Regressionskoffizienten β_j, j=1,2,…,k, im Zeitraum n_1+1, n_1+2, …, n gegenüber dem Zeitraum 1,2,…,n_1 verändert haben. Es kann dann nicht mehr von einer Konstanz der Strukturparameter des ökonometrischen Modells über den gesamten Beobach-tungszeitraum ausgegangen werden..

Der Chow-Test vergleicht die Residualquadratsummen Q_1 und Q_2 der Regressionen vor und nach dem potenziellen Strukturbruch, (6.5) und (6.6), mit der Residualquadratsumme Q der Regression

$$(6.8) \quad y_t = \sum_{j=1}^{k} \beta_j \cdot x_{jt} + u_t, \quad t = 1,2,\ldots,n$$

über die gesamte Beobachtungsperiode. Die Nullhypothese

$$H_0 : \beta_j^1 = \beta_j^2 \quad \text{für alle } j = 1,2,\ldots,k$$

wird auf einem Signifikanzniveau α verworfen, wenn die Prüfgröße

$$(6.9) \quad F = \frac{[Q - (Q_1 + Q_2)] \ / \ k}{(Q_1 + Q_2) \ / \ (n - 2k)}$$

das (1-α)-Quantil der F-Verteilung mit k und n-2k Freiheitsgraden überschreitet.

Beispielhaft soll hier mit dem Chow-Test geprüft werden, ob die Einführung des Euros als Bargeld ab dem 1. Januar 2002 mit einem Strukturbruch in der Geldnachfragefunktion einhergegangen ist. Mit den Residuen `geldfkt$residuals` der bereits in Abschn. 2.2 im Zeitraum 1997-2010 (n=14) geschätzten Geldnachfragefunktion erhält man die Quadratsumme

```
> Q = sum(geldfkt.lm$residuals^2)
```

```
> Q
[1] 0.03053513.
```

Die Residualquadratsummen Q_1 und Q_2 vor und nach dem potenziellen Strukturbruch lassen sich aus den Regressionen für die Teilzeiträume 1997-2001 und 2002-2010 berechnen:

Teilzeitraum 1997-2001 (n_1=5)	Teilzitraum 2002-2010 (n_2=9)
> geldfkt97_01.lm = lm(log(M1GERr)~log(BIPr)+ log(EURIBOR), data=Geldnachfragefkt.data[1:5,]) > summary(geldfkt97_01.lm) Call: lm(formula = log(M1GERr) ~ log(BIPr) + log(EURIBOR), data = Geldnachfragefkt.data[1:5, ,]) Residuals: 1 2 3 4 5 -0.019795 0.006544 0.012275 0.025402 -0.024425 Coefficients: Estimate Std. Error t value Pr(>|t|) (Intercept) -14.69078 3.58389 -4.099 0.0547 . log(BIPr) 2.78725 0.48710 5.722 0.0292 * log(EURIBOR) -0.09268 0.11820 -0.784 0.5151 --- Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 Residual standard error: 0.03023 on 2 degrees of freedom Multiple R-squared: 0.9587, Adjusted R-squared: 0.9174 F-statistic: 23.21 on 2 and 2 DF, p-value: 0.0413	> geldfkt02_10.lm = lm(log(M1GERr)~log(BIPr)+ log(EURIBOR), data=Geldnachfragefkt.data[6:14,]) > summary(geldfkt02_10.lm) Call: lm(formula = log(M1GERr) ~ log(BIPr) + log(EURIBOR), data = Geldnachfragefkt.data[6:14, ,]) Residuals: Min 1Q Median 3Q Max -0.049799 -0.016867 -0.003266 0.019674 0.054725 Coefficients: Estimate Std. Error t value Pr(>|t|) (Intercept) -11.39945 1.23635 -9.220 9.18e-05 *** log(BIPr) 2.34808 0.15890 14.777 6.04e-06 *** log(EURIBOR) -0.16603 0.02326 -7.137 0.000381 *** --- Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 Residual standard error: 0.03643 on 6 degrees of freedom Multiple R-squared: 0.9823, Adjusted R-squared: 0.9764 F-statistic: 166.6 on 2 and 6 DF, p-value: 5.532e-06
> Q1 = sum(geldfkt97_01.lm$residuals^2) > Q1 [1] 0.001827181	> Q2 = sum(geldfkt02_10.lm$residuals^2) > Q2 [1] 0.007964608

Damit nimmt die Prüfgröße (6.9) des Chow-Tests den Wert

```
> F.Chow = ((Q-(Q1+Q2))/3)/((Q1+Q2)/(14-2*3))
> F.Chow
[1] 5.649179
```

an, der auf einem Signifikanzniveau von 5% mit dem 95%-Quantil einer F-Verteilung mit k=3 und n-2·k=8 Freiheitsgraden

```
> Fkrit5 = qf(0.95, 3, 8)

> Fkrit5
```

```
[1] 4.066181
```

zu vergleichen ist. Die Signifikanz wegen F.Chow=5.649179> Fkrit5=4.066181 wird gleichermaßen durch den p-Wert

```
> p.FChow = 1-pf(F.Chow, 3, 8)
> p.FChow
[1] 0.02242426
```

angezeigt, so dass der vermutete Strukturbruch statistisch erhärtet ist.

Komprimiert lässt sich der Chow-Test mit der R-Funktion chow.test durchführen, die in dem Paket gap implementiert ist:

```
> library(gap)
```

Vor dem Aufruf der Funktion chow.test müssen die Vektoren der abhängigen Variablen y1 und y2,

```
> y1 = lnM1GERr[1:5]
> y1
[1] 6.087797 6.192710 6.277282 6.300584 6.357617

> y2 = lnM1GERr[6:14]
> y2
[1] 6.341364 6.431895 6.479124 6.587274 6.636419 6.690966 6.751733 6.962177
[9] 7.056891,
```

und die Matrizen der unabhängigen Variablen,

```
> X1 = cbind(lnBIPr[1:5], lnEURIBOR[1:5])
> X1
       [,1]     [,2]
[1,] 7.501971 1.202972
[2,] 7.532195 1.264127
[3,] 7.554644 1.088562
[4,] 7.571288 1.479329
[5,] 7.608628 1.449269

> X2 = cbind(lnBIPr[6:14], lnEURIBOR[6:14])

> X2
```

```
         [,1]      [,2]
 [1,] 7.637230  1.1999648
 [2,] 7.655220  0.8458683
 [3,] 7.688139  0.7466879
 [4,] 7.707242  0.7839015
 [5,] 7.749784  1.1249296
 [6,] 7.814343  1.4539530
 [7,] 7.840443  1.5325569
 [8,] 7.811188  0.2070142,
 [9,] 7.859313 -0.2107210,
```

für die beiden Teilzeiträume vor und nach dem potenziellen Strukturbruch definiert werden:

```
> geldfkt.chow = chow.test(y1,X1,y2,X2)
> geldfkt.chow
   F value        d.f.1          d.f.2       P value
5.64917937 3.00000000 8.00000000 0.02242426.
```

Die Prüfgröße des Chow-Tests F value wird mit einer Präzision von acht Dezimalstellen ausgewiesen. In Übereinstimmung mit dem zuvor schrittweise durchgeführten Chow-Test weist der p-Wert von 0,022 eine Signifikanz auf dem 5%-Niveau aus und bestätigt damit den Strukturbruch in der gesamtdeutschen Geldnachfragefunktion nach Einführung des Euros als Bargeld.

Übungsaufgaben

6.2.1 Schätzen Sie die die Keynessche Konsumfuktion für die Teilzeiträume 1997-2001 und 2002-2011 (Daten s. Abschn. 1.5: Konsumfkt.data) und ermitteln Sie die Residualquadratsummen Q, Q_1 und Q_2 des Gesamtzeitraums (Residuen: konsfkt.lm$res) und der beiden Teilzeiträume!

6.2.2. Berechnen Sie die Prüfgröße des Chow-Tests unter Verwendung der aus Aufg. 6.2.1 gegebenen Größen und prüfen Sie anhand des

- kritischen Wertes,
- p-Wertes,

ob mit Einführung des Euros als Bargeld seit 2002 ein Strukturbruch in der Konsumfunktion einhergeht!

6.2.3 Testen das Vorhandenseins eines Strukturbruchs in der Konsumfunktion (Abschn. 2.1: konsfkt.lm) mit Hilfe der R-Funktion chow.test!

6.3 Lineares Wahrscheinlichkeitsmodell

Mit einem linearen Wahrscheinlichkeitsmodell lässt sich die Beziehung zwischen einer qualitativen abhängigen Variablen und einer Reihe von erklärenden Variablen spezifizieren. Es stellt die einfachste Form eines mikroökonometrischen Modells zur Erklärung ökonomischer Wahlhandlungen dar. So kann z.B. geprüft werden, welche Einflussgrößen den Kauf eines Produktes bestimmen oder wodurch die Wahl eines Verkehrs- oder Transportmittels erklärt werden kann.

Als Beispiel für den Einsatz des linearen Wahrscheinlichkeitsmodells sei der Kauf eines Produkts (y_t) in Abhängigkeit vom Einkommen (x_{2t}) und Geschlecht (x_{3t}) eines Konsumenten t betrachtet. Die Merkmalsprofile von 25 Konsumenten liegen in der Datendatei Konsumenten.csv aus einer Querschnittserhebung vor:

```
> Konsumenten.df = read.csv("Konsumenten.csv")
> Konsumenten.df
  Konsument Kauf Einkommen Geschlecht
1         1    0     2500          1
2         2    1     3000          0
3         3    1     3200          0
4         4    1     3500          1
5         5    0     2800          1
6         6    0     3000          1
7         7    1     4000          0
8         8    0     2200          1
9         9    1     3800          0
10       10    0     2900          1
11       11    1     2600          1
12       12    0     2400          0
13       13    1     4200          0
14       14    1     3700          0
15       15    0     3300          1
16       16    1     4100          1
17       17    0     2500          1
18       18    1     3900          0
19       19    1     4400          0
20       20    0     2000          1
21       21    0     3100          1
22       22    1     3400          0
23       23    1     3500          1
24       24    1     3800          0
25       25    0     2200          1.
```

Die dichotome abhängige Variable Kauf nimmt den Wert 1 an, falls sich ein Konsument für den Kauf des Produkts entschieden hat. Bei einem Nichtkauf nimmt diese Variable den Wert 0 an. Das Einkommen ist eine metrisch skalierte Variable (in Euro), die zur Erklärung der Wahrscheinlichkeit für den Kauf des Produkts verwendet wird. Hierzu wird auch die Dummy-Variable Geschlecht herangezogen, die gleich 1 gesetzt wird, falls der Konsument weiblich und 0 falls er männlich ist.

Bei dem linearen Wahrscheinlichkeitsmodell (linear probability (LP) model) als dem einfachsten Wahlhandlungsmodell verändert sich z.B. hier die Wahrscheinlichkeit eines Kaufs linear mit einer Veränderung der Einflussgrößen:

(6.10) $p_t = \boldsymbol{\beta}'\mathbf{x}_t$

mit

$$\mathbf{x}_t = (x_{1t} = 1 \quad x_{2t} \quad \ldots \quad x_{kt})'$$

und

$$\boldsymbol{\beta} = (\beta_1 \quad \beta_2 \quad \ldots \quad \beta_k)'.$$

Die Zufallsvariable y_t setzt sich im LP-Modell aus der durch (6.10) gegebenen Wahrscheinlichkeit p_t als ihrem systematischen Teil und der Störvariablen u_t zusammen:

(6.11) $y_t = p_t + u_t = \boldsymbol{\beta}'\mathbf{x}_t + u_t$.

Wendet man die lm-Funktion auf den Datensatz Konsumenten.df an, erhält man die OLS-Schätzung des LP-Modells (6.11):

```
> Konsumenten.lm = lm(Kauf~Einkommen+Geschlecht, data=Konsumenten.df)
> summary(Konsumenten.lm)

Call:
lm(formula = Kauf ~ Einkommen + Geschlecht, data = Konsumenten.df)

Residuals:
   Min      1Q   Median      3Q     Max
-0.47181 -0.24857  0.00583  0.18082  0.83214

Coefficients:
                 Estimate   Std. Error  t value Pr(>|t|)
(Intercept)    -0.6619899  0.4251094  -1.557 0.133688
Einkommen       0.0004342  0.0001144   3.796 0.000991 ***
Geschlecht     -0.2991227  0.1552109  -1.927 0.066966 .
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3216 on 22 degrees of freedom
Multiple R-squared:  0.6306,    Adjusted R-squared:  0.597
F-statistic: 18.77 on 2 and 22 DF,  p-value: 1.75e-05.
```

Während das Einkommen hochsignifikant ist, weist das Geschlecht nur eine schwache Signifikanz auf. Der Regressionskoeffizient des Einkommens von 0,00043 bedeutet, dass sich die Kaufwahrscheinlichkeit für das Produkt um 4,3% erhöht, wenn das Einkommen eines Konsumenten um 100 GE steigt. Der negative Regressionskoeffizient des Merkmals Geschlecht von -0,29912 gibt eine um 29,9% geringere Kaufwahrscheinlichkeit einer Frau im Vergleich zu einem Mann an. Insgesamt erklären die beiden Variablen Einkommen und Geschlecht den Kauf des Produkts zu 63,1%.

Für den Konsumenten 1, der das Produkt nicht gekauft hat (y_1=0) und das Merkmalsprofil $\mathbf{x}_1' = (0 \quad 2500 \quad 1)$ besitzt, errechnet sich eine Kaufwahrscheinlichkeit von

$$\hat{p}_1 = -0{,}6619899 + 0{,}0004342 \cdot 2500 - 0{,}2991227 \cdot 1 = 0{,}12439.$$

Die Kaufwahrscheinlichkeit der Konsumentin 2 mit dem Merkmalsprofil $\mathbf{x}_2' = (1 \quad 3000 \quad 0)$, der das Produkt gekauft hat (y_1=1), beträgt dagegen

$$\hat{p}_2 = -0{,}6619899 + 0{,}0004342 \cdot 3000 = 0{,}64161.$$

Beide Kaufwahrscheinlichkeiten liegen zwischen 0 und 1 und sind gut interpretierbar. Der Konsument 2 besitzt aufgrund seines höheren Einkommens und Geschlechts eine deutlich höhere Kaufwahrscheinlichkeit als die Konsumentin 1. Tatsächlich hat sich der Konsument 2 für den Kauf des Produkt entschieden (y_2=1), während die Konsumentin 1 das Produkt nicht gekauft hat (y_1=0)

Bei dem linearen Wahrscheinlichkeitsmodell ist es allerdings nicht gewährleistet, dass die berechnete Wahrscheinlichkeiten als Regressionswerte im Intervall zwischen 0 und 1 liegen. So ergeben sich auch in unserem Beispiel für die Konsumenten 8, 20 und 25 aufgrund ihrer Merkmalsprofile negative Kaufwahrscheinlichkeiten. Für die Konsumenten 7, 13, 18 und 19 überschreiten dagegen die berechneten Wahrscheinlichkeiten den Wert 1. Darüber hinaus ist die Störvariable des LP-Modells (6.11) nicht homoskedastisch, sondern variiert mit den Werten der erklärenden Variablen. Schließlich wird die Anwendbarkeit des LP-Modells auch dadurch eingeschränkt, dass die Stärke der Effekte der x-Variablen auf die Wahrscheinlichkeit eines betrachteten Ereignisses als unabhängig von ihrem Niveau angenommen wird.

Übungsaufgaben

6.3.1 Schätzen Sie das lineare Wahrscheinlichkeitsmodel (LP-Modell) zur Erklärung der Stellung im Beruf in Abhängigkeit vom Einkommen und Alter mit der OLS-Methode (Daten s. Aufg. 6.1.1) und interpretieren sie die Regressionskoeffizienten!

6.3.2. Berechnen Sie die Regressionswerte des in Aufg. 6.3.1 geschätzten LP-Modells und interpretieren Sie sie unter Berücksichtigung der zu erforderlichen Eigenschaften!

6.3.3 Bei welchen Erwerbspersonen führen die durch die Regression in Aufg. 6.3.1 ermittelten Wahrscheinlichkeiten zu einer Fehlklassifikation?

6.4 Logit-Modell

In der empirischen Wirtschaftsforschung werden aus diesen Gründen anstelle des linearen Wahrscheinlichkeitsmodells das Probit- und Logit-Modell zur Erklärung ökonomischer Wahlhandlungen herangezogen. Während im Probit-Modell z.B. die Kaufwahrscheinlichkeit durch eine Standard-Normalverteilung gegeben ist, liegt dem Logit-Modell die Verteilungsfunktion der logistischen Verteilung zugrunde. Da sich die Wahrscheinlichkeitsfunktionen des Probit- und Logit-Modells in ihrer Form entsprechen, wird hier das schätztechnisch besser handhabbare Logit-Modell betrachtet.

Das Logit-Modell ist durch die Spezifikation der Verteilungsfunktion F in Form der logistischen Verteilung gegeben,

$$(6.12)\quad p_t = \frac{e^{\boldsymbol{\beta}'\mathbf{x}_t}}{1+e^{\boldsymbol{\beta}'\mathbf{x}_t}} = \frac{1}{1+e^{-\boldsymbol{\beta}'\mathbf{x}_t}} = \frac{1}{1+e^{-(\beta_1+\beta_2 \cdot x_{2t}+\ldots+\beta_k \cdot x_{kt})}},$$

woraus man nach Logarithmierung die Spezifikation

$$(6.13)\quad \ln\frac{p_t}{1-p_t} = \boldsymbol{\beta}'\mathbf{x}_t$$

erhält. Die Größen $\ln[p_t/(1-p_t)]$ sind die logarithmierten Odds $p_t/(1-p_t)$, für die sich die Bezeichnung Logits eingebürgert hat. In dem Logit-Modell (6.13) messen die Regressionskoeffizienten β_j die Impulse der exogenen Variablen x_{jt} auf die log Odds.

Die Odds

$$(6.14)\quad \frac{p_t}{1-p_t} = e^{\boldsymbol{\beta}'\mathbf{x}_t}$$

geben das Verhältnis der Wahrscheinlichkeit für eine Realisierung des Ereignisses $y_t=1$ relativ zur Wahrscheinlichkeit des Gegenereignisses $y_t=0$ wieder. Schließlich geben die Effektkoeffizienten e^{β_j} an, wie sich die Odds verändern, wenn die exogene Variable x_{jt} um eine Einheit erhöht wird.

Das Logit Modell (6.13) lässt sich unter Verwendung der R-Funktion glm mit der Maximum-Lkelhood-Methode (ML-Methode) schätzen, die zur Anpassung verallgemeinerter linearer Modelle eingesetzt werden kann. Sie ist wie die R-Funktion lm im Paket stats enthalten, verfügt jedoch über eine Reihe zusätzlicher Argumente. Zur Schätzung eines Logit-Modells mit einer binären abhängigen Variablen ist das Argument family gleich binomial(logit) zu setzen.

Ansonsten kann in unserem Konsumenten-Beispiel die beim LP-Modell verwendete Spezifikation auf die glm-Funktion übertragen werden:

```
> Konsumenten.logit = glm(Kauf~Einkommen+Geschlecht, family=binomial(logit),
data=Konsumenten.df)
> summary(Konsumenten.logit)

Call:
glm(formula = Kauf ~ Einkommen + Geschlecht, family = binomial(logit),
   data = Konsumenten.df)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.25668  -0.25625   0.04079   0.21363   2.45062

Coefficients:
                  Estimate   Std. Error  z value Pr(> | z | )
(Intercept)   -11.726436   5.960102  -1.967   0.0491 *
Einkommen      0.004480   0.002038   2.198   0.0280 *
Geschlecht     -2.874458   1.836447  -1.565   0.1175
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

   Null deviance: 34.296  on 24  degrees of freedom
Residual deviance: 12.144  on 22  degrees of freedom
AIC: 18.144

Number of Fisher Scoring iterations: 7).
```

Aus dem Regressionsoutput geht hervor, dass das Einkommen im Logit-Modell einen positiven Einfluss auf die log Odds des Produktkaufs hat und auf dem 5%-Niveau signifikant ist. Erhöht sich das Einkommen um 1 Euro, erhöhen sich die log Odds um 0,004480 Einheiten. Der Regressionskoeffizient des Geschlechts hat wie im LP-Modell

ein negatives Vorzeichen. Konkret sind die log Odds für einen Produktkauf bei einer Konsumentin um 2.874458 Einheiten geringer als bei eine Konsumenten. Der geschätzte Regressionskoeffizient des Geschlechts ist jedoch nicht signifikant.

Bei der Interpretation der Modellschätzung ist zu beachten, dass sich das Attribut fittted.values nicht auf die geschätzten log Odds $\ln[p_t/(1-p_t)]$, sondern auf die geschätzten Kaufwahrscheinlichkeiten $\hat{p}_t$ bezieht. Sie ergeben sich nach Einsatz der geschätzten Regressionskoeffizienten $\hat{\beta}_j$ in Gleichung (6.12):

$$(6.15)\quad \hat{p}_t = \frac{1}{1+e^{11,72644-0,00448\cdot x_{2t}+2,87446\cdot x_{3t}}}$$

Ein Vergleich mit den Werten der abhängigen Variablen zeigt, wie sich die berechneten Kaufwahrscheinlichkeiten in den realisierten Entscheidungen für oder gegen einen Produktkauf niederschlagen:

```
> Konsumenten.Kauf_Prob = cbind(Konsumenten.df$Kauf, Konsumenten.logit$fitted.values)
> Konsumenten.Kauf_Prob
   [,1]        [,2]
1    0   0.032299316
2    1   0.847455848
3    1   0.931557636
4    1   0.746594318
5    0   0.113472195
6    0   0.238726661
7    1   0.997964938
8    0   0.008628669
9    1   0.995028799
10   0   0.166906242
11   1   0.049649805
12   0   0.274191476
13   1   0.999168355
14   1   0.992240669
15   0   0.545981924
16   1   0.977440353
17   0   0.032299316
18   1   0.996818294
19   1   0.999660384
20   0   0.003540007
21   0   0.329238206
22   1   0.970884801
23   1   0.746594318
```

```
24   1   0.995028799
25   0   0.008628669.
```

Setzt man in (6.15) nicht nur die beobachteten Werte des Einkommens ein, sondern variiert das Einkommen über den relevanten Wertebereich von 0 bis 5000,

```
> Eink.x = 0:5000,
```

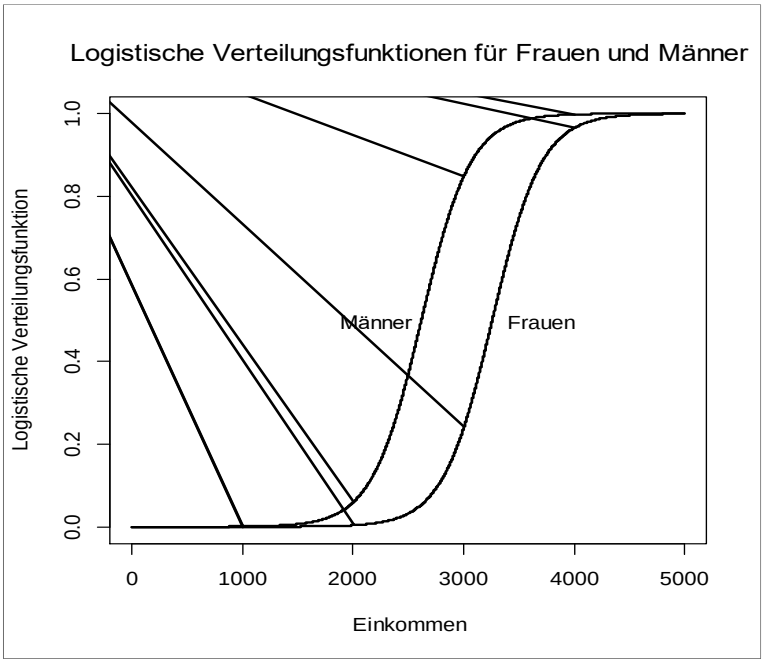
lassen sich die Funktionswerte der Verteilungsfunktionen der logistischen Verteilung für Männer und Frauen berechnen, die sich aus der Modellschätzung ergeben:

```
> pt.male = 1/(1+exp(11.72644-0.00448*Eink.x))
> pt.female = 1/(1+exp(11.72644-0.00448*Eink.x+2.87446)).
```

Sie enthalten die Wahrscheinlichkeiten für den Produktkauf bei einem erreichten Einkommensniveau. Um beide Verteilungsfunktionen in eine Grafik darzustellen, wird für letztere an Stelle der plot-Funktion die lines-Funktion verwendet:

```
> plot(Eink.x,pt.male, main="Logistische Verteilungsfunktionen für Frauen und Männer", xlab= "Einkommen", ylab="Logistische Verteilungsfunktion", type="l", lwd=2, font.main=1, cex.main=1.25)
> lines(Eink.x,pt.female, main="Logistische Verteilungsfunktionen für Frauen und Männer", xlab="Einkommen", ylab="Logistische Verteilungsfunktion", type="l", lwd=2, font.main=1, cex.main=1.25)
> text(2200,0.5, labels = "Männer")
> text(3700,0.5, labels = "Frauen")
> box(which="figure")
```

Abbildung 6.1: Logistische Verteilungsfunktionen für Männer und Frauen



Aus der Grafik wird deutlich, dass die Regressionskoeffizienten nicht genauso interpretiert werden können wie bei der linearen Regression. Wird das Einkommen im mittleren Bereich bei 3000€ um eine Einheit erhöht, dann steigt die Kaufwahrscheinlichkeit viel stärker als bei einer gleich starken Veränderung in den Randbereichen wie z.B. bei 1000 € oder bei 5000 €.

Die geschätzten Effektkoeffizienten $e^{\hat{\beta}_j}$ und die marginalen Effekte der Variablen auf die Kaufwahrscheinlichkeit werden nicht durch die glm-Funktion bereitgestellt. Jedoch lassen sie sich mit den Funktionen logitor und logitmfx des R-Pakets mfx,

```
> library(mfx),
```

ermitteln. Mit der Funktion logitor werden mit den Effektkoeffizienten zugleich auch ihre Standardfehler, z- und p-Werte ausgewiesen:

```
> Konsumenten.effcoef = logitor(Kauf~Einkommen+Geschlecht, data=Konsumenten.df)
> Konsumenten.effcoef
Call:
logitor(formula = Kauf ~ Einkommen + Geschlecht, data = Konsumenten.df)

Odds Ratio:
              OddsRatio    Std. Err.       z     P>|z|
Einkommen    1.0044905   0.0020476   2.1979   0.02796 *
Geschlecht   0.0564467   0.1036614  -1.5652   0.11753
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1.
```

Ein Effektkoeffizient des Einkommens von 1,0044905 bedeutet, dass sich die Odds für einen Produktkauf um 0,45% erhöhen, wenn das Einkommen um 1 Euro steigt. Entsprechend gibt der geschätzte Effektkoeffizient des Geschlechts von 0.0564467 an, dass das Chancenverhältnis für den Produktkauf gegenüber dem Nichtkauf bei einem Übergang von der Gruppe der männlichen Konsumenten ($x_3 = 0$) zur Gruppe der weiblichen Konsumenten ($x_3 = 1$) um 94,36% sinkt.

Der marginale Effekt einer Einflussgröße x_j auf die geschätzte Kaufwahrscheinlichkeit $\hat{p}_t$ hängt nicht nur von ihrem geschätzten Regressionskoeffizient $\hat{\beta}_j$, sondern auch von den konkreten Werten x_{jt} der Variablen ab:

$$(6.16)\quad \frac{\partial \hat{p}_t}{\partial x_{jt}} = \frac{e^{\hat{\boldsymbol{\beta}}'\mathbf{x}_t}}{\left(1+e^{\hat{\boldsymbol{\beta}}'\mathbf{x}_t}\right)^2} \cdot \hat{\beta}_j$$

Jedoch wird sich bei der Angabe marginaler Effekte häufig auf die Mittelwerte der Variablen bezogen, wozu in der Funktion logitfmx das Argument atmean auf TRUE (Voreinstellung) zu setzen ist:

```
> Konsumenten.margeff = logitmfx(Kauf~Einkommen+Geschlecht, atmean=TRUE, data=Konsumenten.df)
> Konsumenten.margeff
Call:
logitmfx(formula = Kauf ~ Einkommen + Geschlecht, data = Konsumenten.df,
   atmean = TRUE)

Marginal Effects:
                  dF/dx        Std. Err.        z     P>|z|
Einkommen    0.00088043   0.00040834   2.1561   0.03108 *
Geschlecht   -0.49707668   0.22825481  -2.1777   0.02943 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

dF/dx is for discrete change for the following variables:

[1] "Geschlecht".
```

Der marginale Effekt des Einkommens von 0.00088 ergibt sich bei einem arithmetischen Mittel von 3200€,

```
> Einkommen.mean = mean(Konsumenten.df$Einkommen)
> Einkommen.mean
[1] 3200,
```

unter der Bedingung, dass die Wahrscheinlichkeit eines weiblichen (männlichen) Konsumenten durch die relative Häufigkeit der Stichprobe (relative Häufigkeit eines weiblichen Konsumenten: 14/25=0,56) gegeben ist:

```
> n = length(Konsumenten.df$Geschlecht)
> n
[1] 25
> n1 = sum(Konsumenten.df$Geschlecht)
> n1
[1] 14
> h1  = n1/n
> h1
[1] 0.56
> b1 = Konsumenten.logit$coeff[1]
```

```
> b1
(Intercept)
  -11.72644
> b2 = Konsumenten.logit$coeff[2]
> b2
  Einkommen
0.004480407
> b3 = Konsumenten.logit$coeff[3]
> b3
Geschlecht
 -2.874458
> Einkommen.margeff = as.numeric((exp(b1+b2*3200+b3*h1)/(1+exp(b1+b2*3200+b3*h1))^2)*b2)
> Einkommen.margeff
[1] 0.0008804252.
```

Danach erhöht sich die Wahrscheinlichkeit für den Kauf des Produkts um approximativ 8,8 Prozentpunkte, wenn das mittlere Einkommen um 100€ zunimmt. Für einen männlichen Konsumenten mit durchschnittlichem Einkommen ergibt sich ein marginaler Effekt von

```
> > Einkommen.margeffm = as.numeric((exp(b1+b2*3200+b3*0)/(1+exp(b1+b2*3200+b3*0))^2)*b2)
> Einkommen.margeffm
[1] 0.0002856618
```

und für einen weiblichen Konsumenten ist der entsprechende marginale Effekt durch

```
> Einkommen.margeffw = as.numeric((exp(b1+b2*3200+b3*1)/(1+exp(b1+b2*3200+b3*1))^2)*b2)
> Einkommen.margeffw
[1] 0.001100869
```

gegeben.

Für die qualitative Variable Geschlecht wird der marginale Effekt dagegen nicht auf der Basis der Formel (6.16) berechnet, sondern unter Verwendung von (6.12) als Differenz der Kaufwahrscheinlichkeiten eines weiblichen und männlichen Konsumenten mit mittlerem Einkommen:

```
> pt.female.mean = as.numeric(1/(1+exp(-b1-b2*3200-b3)))
> pt.female.mean
[1] 0.434481
> pt.male.mean = as.numeric(1/(1+exp(-b1-b2*3200)))
```

```
> pt.male.mean
[1] 0.9315576
> Geschlecht.margeff = pt.female.mean - pt.male.mean
> Geschlecht.margeff
[1] -0.4970767.
```

Hier gibt der marginale Effekt von -0,4970767 an, dass die Wahrscheinlichkeit eines Produktkaufs bei einem Übergang von einem männlichen zu einem weiblichen Konsumenten um 49,7 Prozentpunkte sinkt.

Die Güte der Modellanpassung kann auf verschiedene Art und Weise überprüft werden. Zur Detailprüfung können verschiedene Arten von Residuen analysiert werden, die durch Anwendung der Funktion residuals auf das glm-Objekt Konsumenten.logit verfügbar sind. Durch Spezifikation des Arguments type als "response",

```
> resid.response = residuals(Konsumenten.logit, type="response")
> resid.response
          1           2           3           4           5
-0.0322993160  0.1525441520  0.0684423640  0.2534056822 -0.1134721948
          6           7           8           9          10
-0.2387266614  0.0020350623 -0.0086286691  0.0049712006 -0.1669062422
         11          12          13          14          15
 0.9503501946 -0.2741914763  0.0008316448  0.0077593308 -0.5459819239
         16          17          18          19          20
 0.0225596471 -0.0322993160  0.0031817059  0.0003396163 -0.0035400070
         21          22          23          24          25
-0.3292382064  0.0291151990  0.2534056822  0.0049712006 -0.0086286691,
```

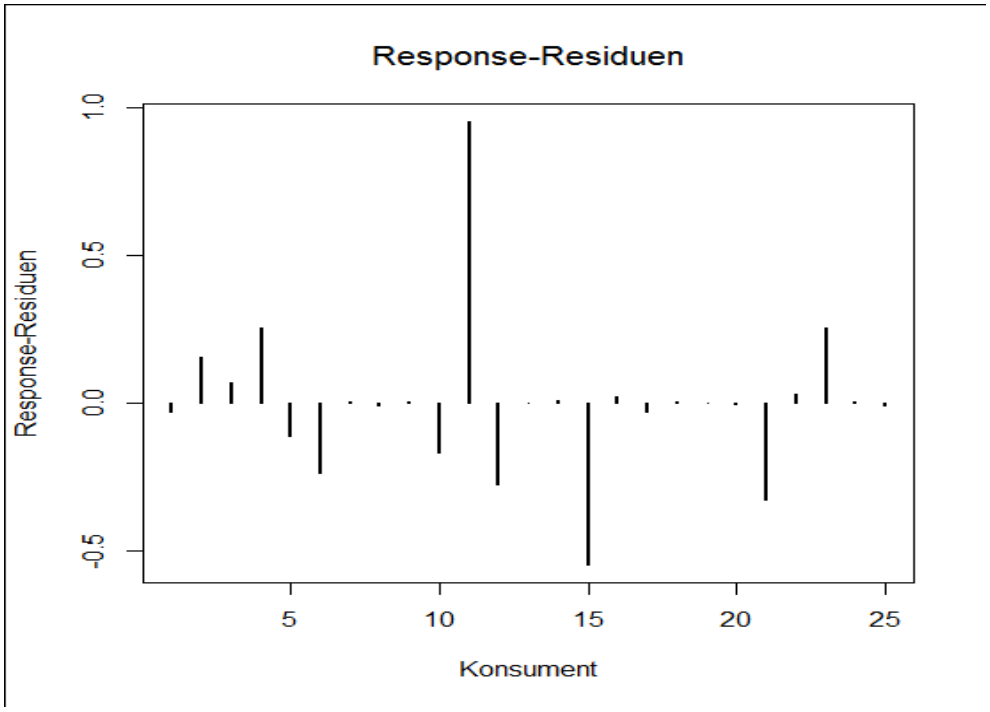
werden die Residuen als Differenz zwischen der dichotomen abhängigen Variablen (Responsevariable) Kauf und den geschätzten Kaufwahrscheinlichkeiten $\hat{p}_t$ gebildet:

```
> Konsumenten.df$Kauf-Konsumenten.logit$fitted.values
          1           2           3           4           5
-0.0322993160  0.1525441520  0.0684423640  0.2534056822 -0.1134721948
          6           7           8           9          10
-0.2387266614  0.0020350623 -0.0086286691  0.0049712006 -0.1669062422
         11          12          13          14          15
 0.9503501946 -0.2741914763  0.0008316448  0.0077593308 -0.5459819239
         16          17          18          19          20
 0.0225596471 -0.0322993160  0.0031817059  0.0003396163 -0.0035400070
         21          22          23          24          25
-0.3292382064  0.0291151990  0.2534056822  0.0049712006 -0.0086286691.
```

Ein Diagramm der Response-Residuen lässt sich mit dem plot-Befehl erstellen. Um die Residuen als senkrechte Linien abzubilden, wird hierbei das Argument type gleich "h" gesetzt:

```
> plot(resid.response, main="Response-Residuen", xlab="Konsument", ylab="Response-Residuen",
type="h", lwd=2,cex.main=1.25, font.main=1)
> box(which="figure")
```

Abbildung 6.2: Diagramm der Response-Residuen des Logit-Modells



Aus der Abb. 6.1 gehen die Abweichungen zwischen den Kaufentscheidungen der Konsumenten und den geschätzten Kaufwahrscheinlichkeiten visuell hervor. Hierbei fallen insbesondere größere Abweichungen bei den Konsumenten 11 und 15 ins Auge.

Global lässt sich die Güte der Modellanpassung mit Hilfe des Likelihood-Ratio-Testes und anhand eines dem Bestimmtheitsmaß vergleichbaren Pseudo-R^2 überprüfen. Bei Gültigkeit der Nullhypothese

$$H_0\text{: } \beta_2 = \beta_3 = \ldots = \beta_p = 0$$

ist die logarithmierte Likelihood-Funktion (restringierte Likelihood-Funktion) durch

$$(6.17) \quad L_0^* = \ln L_0 = n_1 \cdot \ln\left(\frac{n_1}{n}\right) + (n - n_1) \cdot \ln\left(\frac{n - n_1}{n}\right)$$

gegeben. Die unrestringierte Likelihood-Funktion L* basiert dagegen auf der Anpassung des Logit-Modells:

(6.18) $L^* = \ln L = \sum_{t=1}^{n} y_t \cdot \ln \hat{p}_t + (1 - y_t) \cdot \ln(1 - \hat{p}_t)$.

Unter der Nullhypothese (6.17) folgt die Likelihood-Ratio-Statistik

(6.19) $l = -2\left(L_0^* - L^*\right)$

asymptotisch einer χ^2-Verteilung mit k-1 Freiheitsgraden.

In dem Konsumenten-Beispiel besagt die Nullhypothese H_0: $\beta_2 = \beta_3 = 0$, dass die beiden Variablen Einkommen und Geschlecht keinen Einfluss auf die Wahrscheinlichkeit des Produktkaufs haben. Die Likelihood-Ratio-Statistik (6.19) lässt sich unter Verwendung der Gleichungen (6.18) und (6.19) berechnen:

```
> L0s = n1*log(n1/n)+(n-n1)*log((n-n1)/n)
> L0s
[1] -17.14825
> Ls = Konsumenten.df$Kauf%*%log(Konsumenten.logit$fitted.values) + (1-Konsumenten.df$Kauf)%*%
log(1-Konsumenten.logit$fitted.values)
> Ls
        [,1]
[1,] -6.071946
> l = -2*(L0s - Ls)
> l
      [,1]
[1,] 22.1526.
```

Das 99%-Quantil der χ^2-Verteilung mit 2 Freiheitsgraden nimmt einen Wert von

```
> chi2krit1 = qchisq(0.99, 2)
> chi2krit1
[1] 9.21034
```

an. Da der Wert der Prüfgröße von 22,1526 den kritischen Wert von 9,21034 übersteigt, ist das geschätzte Logit-Modell (6.15) auf dem 1%-Niveau statistisch gesichert. Die hohe Signifikanz wird auch durch den p-Wert

```
> p.chi = 1 - pchisq(l, 2)
> p.chi
          [,1]
[1,] 1.547477e-05
```

bestätigt, der deutlich unter dem nominalen Signifikanzniveau α von 0,01 liegt.

Das Pseudo-Bestimmtheitsmaß von McFadden, PR^2, basiert auf dem Verhältnis L_0^*/L^* der logarithmierten Likelihoodfunktion:

$$(6.20) \quad PR^2 = 1 - \frac{L^*}{L_0^*}.$$

PR^2 spiegelt den Grad der Verbesserung bei der Erklärung der Logits wieder, die sich durch Aufnahme der Variablen x_2, x_3, …, x_k gegenüber einem Modell ergibt, das nur die Konstante enthält. Obwohl das Pseudo-R^2 zwischen 0 und 1 liegt, indizieren bereits Werte zwischen 0,2 und 0,4 eine gute Anpassung.

In dem Konsumenten-Beispiel weist das Pseudo-R^2 von 0,656,

```
> PR2 = 1 - Ls/L0s
> PR2
     [,1]
[1,] 0.645914,
```

auf eine gute Modellanpassung durch die beiden Variablen Einkommen und Geschlecht hin.

Übungsaufgaben

6.3.1 Schätzen Sie das Logit-Modell zur Erklärung der Stellung im Beruf in Abhängigkeit vom Einkommen und Alter mit der Maximum-Likelihood-Methode (Daten s. Aufg. 6.1.1) und interpretieren sie die Regressionskoeffizienten!

6.3.2 Zeichnen Sie die logistischen Verteilungsfunktionen für die selbständigen und nicht-selbständigen Erwerbstätigen auf der Grundlage der in Aufg. 6.3.1 geschätzten Regressionskoeffizienten!

6.3.3 Bestimmen Sie unter Verwendung des in Aufg. 6.3.1 geschätzten Logit-Modells die

- Effektkoeffizienten,
- marginalen Effekte

und interpretieren Sie sie!

6.3.4 Stellen Sie die geschätzten Wahrscheinlichkeiten und die Response-Residuen in einem Streuungsdiagramm dar (Modellschätzung: Aufg. 6.3.1)!

6.3.5 Prüfen Sie die Nullhypothese H_0: $\beta_2 = \beta_3$ mit dem Likelihood-RatioTest für das in Aufg. 6.3.1 geschätzte Logit-Modell (α=0,05)!

6.3.6 Berechnen Sie McFaddens Pseudo-R^2 für das in Aufg. 6.3.1 geschätzte Logit-Modell uns interpretieren Sie es!