

3 Generalisierte lineare Modelle

Lösung zu Aufgabe 5

- (a) siehe Übungsmitschrift
- (b) siehe Übungsmitschrift
- (c) siehe Übungsmitschrift
- (d) siehe Übungsmitschrift
- (e) Datensatz `fakesoep` von Uebungsblatt 4

```
# Einlesen der Daten
soep <- read.table("fakesoep.dat",header=T)

# Sichtbarmachen der Variablen
attach(soep)
```

Schaetzung der Modelle

```
# Volles Modell
gammaLog <- glm(beink ~ groesse + alter + dauer + verh + geschl + deutsch + abitur,
family=Gamma(link=log))

# Modell mit \beta_{alter} = \beta_{2} = 0
gammaLogredai <- glm(beink ~ groesse + dauer + verh + geschl + deutsch + abitur,
family=Gamma(link=log))
```

- Likelihood-Ratio

```
# Funktion zur Berechnung der Devianz
deviance_gamma <- function(y,X,beta){

  n <- length(y)
  X <- as.matrix(X)
  dev <- 0
  for(i in 1:n){
    x <- X[i,]
    dev <- dev+(-log(y[i]/exp(x%%beta)))+(y[i]-exp(x%%beta))/exp(x%%beta))
  }
  dev <- 2*dev
  return(dev)
}

y <- soep$beink
X <- cbind(1,soep[,-1])
beta_tilde <- coef(gammaLogredai)
beta_hat <- coef(gammaLog)
```

```

dev_tilde <- deviance_gamma(y,X[,-3],beta_tilde)
dev_hat   <- deviance_gamma(y,X,beta_hat)

# LR - Statistik
lambda <- 1/summary(gammaLog)$dispersion*(dev_tilde-dev_hat)
lambda

##           [,1]
## [1,] 0.03521757

# Vergleich mit Chi^2 (p-Wert)
1 - pchisq(lambda,df=1)

##           [,1]
## [1,] 0.8511405

# => H_0 beibehalten

# einfacher
lambda <- 1/summary(gammaLog)$dispersion*(gammaLogredai$deviance - gammaLog$deviance)
lambda

## [1] 0.03521757

# oder noch einfacher mit anova
anova(gammaLogredai,gammaLog,test="Chi")

## Analysis of Deviance Table
##
## Model 1: beink ~ groesse + dauer + verh + geschl + deutsch + abitur
## Model 2: beink ~ groesse + alter + dauer + verh + geschl + deutsch + abitur
##   Resid. Df Resid. Dev Df   Deviance Pr(>Chi)
## 1      2993      502.5
## 2      2992      502.5  1 0.0061545  0.8511

```

- Wald

```

X <- as.matrix(X)
InvF_hat <- solve(1/summary(gammaLog)$dispersion*t(X)%*%X)

w <- coef(gammaLog)[3]^2/InvF_hat[3,3]
w

##      alter
## 0.03651594

t <- coef(gammaLog)[3]/sqrt(InvF_hat[3,3])
t

##      alter
## 0.1910914

# Vergleich mit Chi^2 (p-Wert)
1 - pchisq(w,df=1)

##      alter
## 0.848454

2*pt(-abs(t),df=gammaLog$df.residual)

##      alter
## 0.8484669

# => H_0 beibehalten

```

```

# einfacher
varbeta <- diag(summary(gammaLog)$cov.scaled)
w <- gammaLog$coef[3]^2/varbeta[3]
w

##      alter
## 0.03651594

# oder einfach ablesen aus
summary(gammaLog)

##
## Call:
## glm(formula = beink ~ groesse + alter + dauer + verh + geschl +
##      deutsch + abitur, family = Gamma(link = log))
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.47399 -0.31490 -0.05961  0.18374  1.94176
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.2202325  0.2182771  28.497 < 2e-16 ***
## groesse      0.0082530  0.0011930   6.918 5.58e-12 ***
## alter        0.0001786  0.0009345   0.191  0.848
## dauer        0.0119425  0.0009816  12.166 < 2e-16 ***
## verh        -0.0178813  0.0173453  -1.031  0.303
## geschl      -0.3179608  0.0216168 -14.709 < 2e-16 ***
## deutsch      0.0050755  0.0279452   0.182  0.856
## abitur       0.3466434  0.0167621  20.680 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Gamma family taken to be 0.1747557)
##
##      Null deviance: 757.46  on 2999  degrees of freedom
## Residual deviance: 502.50  on 2992  degrees of freedom
## AIC: 49184
##
## Number of Fisher Scoring iterations: 5

```

- Score

```

h_tilde <- c(exp(X[,-3]**coef(gammaLogredai)))
D_tilde <- diag(h_tilde)
sigma_tilde <- summary(gammaLogredai)$dispersion*D_tilde^2

score_tilde <- t(X)**D_tilde**solve(sigma_tilde)**(y-h_tilde)
InvF_tilde <- solve(1/summary(gammaLogredai)$dispersion*t(X)**X)

u <- t(score_tilde)**InvF_tilde**score_tilde
u

##              [,1]
## [1,] 0.03397168

# Vergleich mit Chi^2 (p-Wert)
1 - pchisq(u,df=1)

##              [,1]
## [1,] 0.8537671

# => H_0 beibehalten

```

```

# oder einfacher mit anova
anova(gammaLogredai, gammaLog, test="Rao")

## Analysis of Deviance Table
##
## Model 1: beink ~ groesse + dauer + verh + geschl + deutsch + abitur
## Model 2: beink ~ groesse + alter + dauer + verh + geschl + deutsch + abitur
##   Resid. Df Resid. Dev Df Deviance      Rao Pr(>Chi)
## 1      2993      502.5
## 2      2992      502.5  1 0.0061545 0.0059367  0.8538

rao <- anova(gammaLogredai, gammaLog, test="Rao")[2,5]
rao/summary(gammaLogredai)$dispersion

## [1] 0.03397168

```

(f) Hypothesen (a)(ii) und (a)(iii)

```

# Modell mit \beta_2 = \beta_4 = \beta_6 = 0
gammaLogredaii <- glm(beink ~ groesse + dauer + geschl + abitur,
                      family=Gamma(link=log))

lambda <- 1/summary(gammaLog)$dispersion*(gammaLogredaii$deviance - gammaLog$deviance)
lambda

## [1] 1.109219

# Vergleich mit Chi^2 (p-Wert)
1 - pchisq(lambda, df=3)

## [1] 0.774849

# => H_0 beibehalten

```

```

# Modell mit \beta_1 = \beta_3
gammaLogredaiii <- glm(beink ~ I(groesse + dauer) + alter + verh + geschl + deutsch + abitur,
                      family=Gamma(link=log))

lambda <- 1/summary(gammaLog)$dispersion*(gammaLogredaiii$deviance - gammaLog$deviance)
lambda

## [1] 5.663328

# Vergleich mit Chi^2 (p-Wert)
1 - pchisq(lambda, df=1)

## [1] 0.01732323

# => H_0 ablehnen

```