

# 1 Mehrkategoriale Regressionsmodelle (III)

## Aufgabe 1

a) Datenaufbereitung

```
# Daten einlesen
knie <- read.table("knie.txt", header=T)

attach(knie)
str(knie)

## 'data.frame': 123 obs. of 4 variables:
## $ TH : int 0 0 0 1 1 0 1 1 1 0 ...
## $ AGE : int 28 32 41 21 34 24 28 40 24 39 ...
## $ GEN : int 0 0 0 0 0 0 0 0 1 1 ...
## $ PAIN: int 4 4 3 2 2 2 2 2 3 3 ...

# Kodierung als ordinale Variable
PAIN <- as.ordered(PAIN)

# hier nicht unbedingt nötig, da bereits 0/1-kodiert
TH <- as.factor(TH)
GEN <- as.factor(GEN)

# Einzentrieren um 30 (entspricht in etwa dem arithmetischen Mittel)
mean(AGE)

## [1] 29.98374

AGE <- AGE-30
```

```
library(MASS)

# Für das kumulative Logit-Modell mit polr:
#  $P(y \leq r/x) = \frac{\exp(\theta_{0r} - x'\gamma)}{1 + \exp(\theta_{0r} - x'\gamma)}$ 
# bzw.  $\log(P(y \leq r/x)/P(y > k/x)) = \theta_{0r} - x'\gamma$ .
# ACHTUNG:
# In der Vorlesung, bzw. bei Tutz (2000) wird statt
#  $\log(P(y \leq r/x)/P(y > k/x)) = \theta_{0r} - x'\gamma$  nämlich
#  $\log(P(y \leq r/x)/P(y > k/x)) = \theta_{0r} + x'\gamma$  verwendet!
# Die Definition von polr erleichtert die Interpretation:
# positiver Koeffizient -> "positiver" Zusammenhang
# (in dem Sinne: die Chance auf eine HÖHERE Kategorie steigt)
```

b) siehe Übungsmitschrift

c) siehe Übungsmitschrift

d) Modell mit Haupteffekten (falls im Argument  $Hess = T$ , wird die beob. Fisher-Matrix gespeichert):

```

ModelAge <- polr(PAIN ~ TH + GEN + AGE)
summary(ModelAge)

## Call:
## polr(formula = PAIN ~ TH + GEN + AGE)
##
## Coefficients:
##          Value Std. Error  t value
## TH1   -1.0335247   0.34469 -2.998455
## GEN1    0.0007772   0.37663  0.002064
## AGE   -0.0260173   0.01765 -1.474401
##
## Intercepts:
##      Value  Std. Error t value
## 1|2 -1.6393  0.3131   -5.2355
## 2|3 -0.4243  0.2782   -1.5255
## 3|4  0.5621  0.2752    2.0427
##
## Residual Deviance: 329.3384
## AIC: 341.3384

```

Kumulatives Logit-Modell: Parameterschätzer für Einflussgrößen sind über alle Response-Kategorien gleich; man kann Daten in disjunkte Untermengen  $1, \dots, r$  ( $Y_r = 1$ ) und  $r + 1, \dots, k$  ( $Y_r = 0$ ) einteilen und diese (mit zugehörigem Intercept  $\theta_{0r}$ ) wie binäre Regressionsmodelle für die abhängige Variable  $Y_r$  interpretieren.

```

exp(-ModelAge$coef[1])

##          TH1
##  2.810956

```

Interpretation: beim Medikament ist die Chance für eine niedrigere Responsekategorie, d.h. hier für weniger starke Schmerzen, um den Faktor 2.8 größer als beim Placebo, bei festgehaltenen übrigen Kovariablen.

Konfidenzintervalle für Parameterschätzer:

```

confint(ModelAge)

##          2.5 %          97.5 %
## TH1  -1.71903172 -0.364982140
## GEN1 -0.74037880  0.741033508
## AGE  -0.06090193  0.008485807

```

Man erkennt: Der Therapie-Effekt ist signifikant von 0 verschieden (negativ), der Geschlechts-Effekt sowie der Alters-Effekt (linear) offenbar nicht. (Das gilt wegen Dualität von Konfidenzintervallen und Hypothesentests: die im KI enthaltenen Werte sind genau diejenigen, die beim Test nicht signifikant verworfen werden würden.)

- e) siehe Übungsmitschrift
- f) Alter wird quadratisch mit aufgenommen

```

AGE2 <- AGE^2

ModelAge2 <- update(ModelAge, . ~ . + AGE2)
summary(ModelAge2)

```

```
## Call:
## polr(formula = PAIN ~ TH + GEN + AGE + AGE2)
##
## Coefficients:
##           Value Std. Error t value
## TH1  -1.012266   0.347869 -2.9099
## GEN1   0.084941   0.381525  0.2226
## AGE  -0.007900   0.019249 -0.4104
## AGE2 -0.005233   0.002191 -2.3887
##
## Intercepts:
##           Value Std. Error t value
## 1|2 -2.1264   0.3808   -5.5841
## 2|3 -0.8614   0.3363   -2.5613
## 3|4  0.1576   0.3240    0.4864
##
## Residual Deviance: 323.4087
## AIC: 337.4087
```

*#Konfidenzintervalle:*

```
confint(ModelAge2)
```

```
##           2.5 %          97.5 %
## TH1  -1.704034892 -0.337353237
## GEN1 -0.664725567  0.835995008
## AGE  -0.045791576  0.029878840
## AGE2 -0.009645029 -0.001015357
```

Der quadratische Effekt des Alters ist signifikant von 0 verschieden; offenbar hat das Alter doch einen Einfluss. Er ist nur nicht linear.

Mit einem Likelihood-Ratio Test vergleicht man, ob das Modell mit quadratischem Alter zum Haupteffekt-Modell reduziert werden kann.

```
anova(ModelAge,ModelAge2)
```

```
## Likelihood ratio tests of ordinal regression models
```

```
##
```

```
## Response: PAIN
```

| ##   | Model                 | Resid. df | Resid. Dev | Test   | Df | LR stat. | Pr(Chi)    |
|------|-----------------------|-----------|------------|--------|----|----------|------------|
| ## 1 | TH + GEN + AGE        | 117       | 329.3384   |        |    |          |            |
| ## 2 | TH + GEN + AGE + AGE2 | 116       | 323.4087   | 1 vs 2 | 1  | 5.929702 | 0.01488776 |

Signifikanter Unterschied ( $p - Wert = 0.0149$ ), keine Reduktion möglich. Auch ein Vergleich des AICs spricht für das Modell mit quadratischem Alter.

```
AIC(ModelAge)
```

```
## [1] 341.3384
```

```
AIC(ModelAge2)
```

```
## [1] 337.4087
```

Interpretation des Effekts von Alter auf Basis der Koeffizienten schwierig, da quadratisch. Daher Zeichnung der entsprechenden Funktion

```

alter <- function(a,z,b1,b2)
{
  b1*(a-z) + b2*(a-z)^2
}

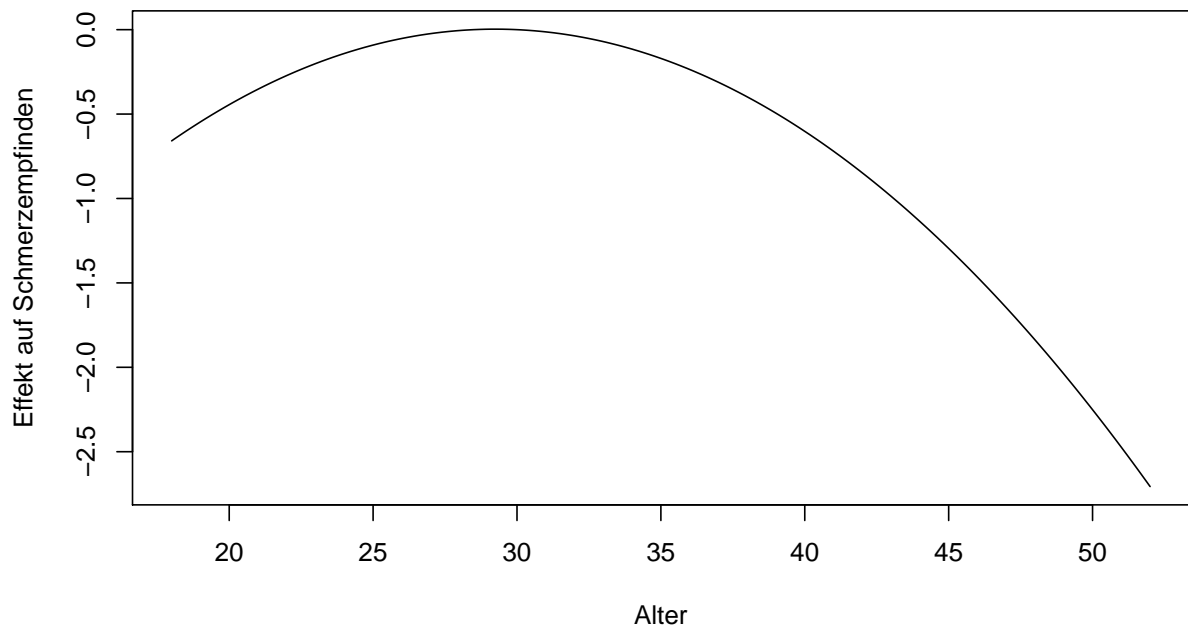
# Beobachtete Werte des (wahren) Alters
summary(knie$AGE)

##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  18.00  21.00  28.00  29.98  38.00  52.00

a <- seq(18,52,by=0.1)

# Darstellung auf Basis des nicht zentrierten Alters
plot(a,alter(a,30,ModelAge2$coef[3],ModelAge2$coef[4]),type="l",
      xlab = "Alter", ylab = "Effekt auf Schmerzempfinden")

```



Nach Aussage des Modells sind die (subjektiven) Schmerzen für eine Person knapp unter 30 am schlimmsten.

Im folgenden überprüfen wir mit Hilfe der Devianzanalyse, ob das Modell auf andere Art reduziert werden kann.

```

# ModelAge2 ohne Geschlecht
ModelAge2_gen <- update(ModelAge2, . ~ . - GEN)
anova(ModelAge2,ModelAge2_gen)

## Likelihood ratio tests of ordinal regression models
##
## Response: PAIN
##
##      Model Resid. df Resid. Dev  Test   Df  LR stat.  Pr(Chi)
## 1      TH + AGE + AGE2      117  323.4583
## 2 TH + GEN + AGE + AGE2      116  323.4087 1 vs 2    1 0.04957645 0.8238019

# keine Signifikanz (p-Wert = 0.824), also kann GEN weggelassen werden.

```

```

# ModelAge2 ohne Therapie:
ModelAge2_th <- update(ModelAge2, . ~ . - TH)
anova(ModelAge2, ModelAge2_th)

## Likelihood ratio tests of ordinal regression models
##
## Response: PAIN
##
##           Model Resid. df Resid. Dev   Test      Df LR stat.    Pr(Chi)
## 1      GEN + AGE + AGE2      117   332.1104
## 2 TH + GEN + AGE + AGE2      116   323.4087 1 vs 2      1 8.701625 0.003179266

# keine Reduktion möglich (p-Wert = 0.0032).

# ohne AGE:
Model_age <- update(ModelAge2, . ~ . - AGE - AGE2)
anova(ModelAge2, Model_age)

## Likelihood ratio tests of ordinal regression models
##
## Response: PAIN
##
##           Model Resid. df Resid. Dev   Test      Df LR stat.    Pr(Chi)
## 1      TH + GEN      118   331.5227
## 2 TH + GEN + AGE + AGE2      116   323.4087 1 vs 2      2 8.113912 0.01730161

# Wie zu erwarten war (vgl. oben): auch hier keine Reduktion möglich
# (p-Wert = 0.0173)

```

Demnach entscheidet man sich für ein Modell mit Kovariablen TH, AGE und  $AGE^2$  (eine Entscheidung auf Basis des AIC würde hier zum selben Modell führen).

#### g) Vorhersage

```

# Verwenden des Sprays
newdataTh <- data.frame(TH=factor(1),
                        GEN=factor(0),
                        AGE=22-30,
                        AGE2=(22-30)^2)

# Placebo
newdataPl <- data.frame(TH=factor(0),
                        GEN=factor(0),
                        AGE=22-30,
                        AGE2=(22-30)^2)

# Vorhersage für ModelAge2 ohne GEN (wurde in (c) gewählt):
predict(ModelAge2_gen, newdata=newdataTh, type="p")

##           1           2           3           4
## 0.2969697 0.3025794 0.2059070 0.1945439

predict(ModelAge2_gen, newdata=newdataPl, type="p")

##           1           2           3           4
## 0.1336878 0.2198826 0.2484241 0.3980054

# Die Wahrscheinlichkeit für geringeres Schmerzempfinden ist bei Verwendung des
# Sprays deutlich höher! als bei Verabreichung des Placebos.

```

```

# Zum Vergleich: Vorhersage für ModelAge2 (mit GEN):
predict(ModelAge2,newdata=newdataTh,type="p")

##           1           2           3           4
## 0.3010318 0.3030643 0.2046032 0.1913007

predict(ModelAge2,newdata=newdataPl,type="p")

##           1           2           3           4
## 0.1353273 0.2213756 0.2490072 0.3942900

# Ignorieren des Geschlechts verändert die vorhergesagten Wahrscheinlichkeiten
# kaum.

```

Die Funktion `lrm()` aus der library `Design` kann ebenfalls zum Fit eines kumulativen Logit-Modells verwendet werden. Achtung: dort ist die Modellgleichung anders definiert:  $\log(P(y \geq r|x)/P(y < r|x)) = \theta_{0r} + x'\gamma$

## Aufgabe 2

### a) Überblick

```

# Wenn installiert, laden wie gewohnt:
library(VGAM)

# Einen Überblick verschafft man sich am besten mit der Dokumentation als pdf.

```

- b) Zum Fitten eines Proportional Odds Modell verwendet man die Funktion `vglm`, wobei man sich das geeignete family-Objekt mit der Funktion `cumulative` erzeugen kann. Damit man nicht nur eine kumulative sondern auch ein Proportional Odds Modell erhält, muss dort `parallel=TRUE` gewählt werden.

```

modelvglm1a <- vglm(PAIN ~ TH + AGE + AGE2, family=cumulative(parallel=T))
summary(modelvglm1a)

##
## Call:
## vglm(formula = PAIN ~ TH + AGE + AGE2, family = cumulative(parallel = T))
##
## Pearson residuals:
##           Min           1Q   Median           3Q          Max
## logit(P[Y<=1]) -1.933 -0.8242 -0.2573 0.7619 2.825
## logit(P[Y<=2]) -2.961 -0.5013 0.2934 0.8858 1.813
## logit(P[Y<=3]) -2.429 -1.0076 0.2705 0.3895 1.393
##
## Coefficients:
##           Estimate Std. Error z value Pr(>|z|)
## (Intercept):1 -2.140711   0.365604  -5.855 4.76e-09 ***
## (Intercept):2 -0.875355   0.318713  -2.747 0.00602 **
## (Intercept):3 0.141804   0.311846   0.455 0.64931
## TH1           1.006955   0.338947   2.971 0.00297 **
## AGE           0.007508   0.019149   0.392 0.69501
## AGE2          0.005188   0.002155   2.408 0.01606 *
## ---

```

```

## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Number of linear predictors:  3
##
## Names of linear predictors: logit(P[Y<=1]), logit(P[Y<=2]), logit(P[Y<=3])
##
## Dispersion Parameter for cumulative family:  1
##
## Residual deviance: 323.4583 on 363 degrees of freedom
##
## Log-likelihood: -161.7292 on 363 degrees of freedom
##
## Number of iterations: 6

# Vergleich mit Aufgabe 24
summary(ModelAge2_gen)

## Call:
## polr(formula = PAIN ~ TH + AGE + AGE2)
##
## Coefficients:
##           Value Std. Error t value
## TH1    -1.006968   0.347069 -2.9013
## AGE     -0.007508   0.019176 -0.3915
## AGE2   -0.005188   0.002179 -2.3806
##
## Intercepts:
##           Value  Std. Error t value
## 1|2  -2.1407   0.3753   -5.7045
## 2|3  -0.8754   0.3303   -2.6501
## 3|4   0.1418   0.3161    0.4486
##
## Residual Deviance: 323.4583
## AIC: 335.4583

```

Die Vorzeichen der  $\gamma$  sind umgekehrt. Hier wird die Darstellung aus der Vorlesung verwendet.

- c) Verwendet man nicht den Logit-Link sondern den komplementären loglog-Link erhält man das Proportional Hazards Modell (vgl. Vorlesung / Blatt 9)

```

modelvglm1b <- vglm(PAIN ~ TH + AGE + AGE2,
family=cumulative(link="cloglog",parallel=T))
summary(modelvglm1b)

##
## Call:
## vglm(formula = PAIN ~ TH + AGE + AGE2, family = cumulative(link = "cloglog",
##           parallel = T))
##
## Pearson residuals:
##           Min       1Q   Median       3Q      Max
## cloglog(P[Y<=1]) -1.552 -0.8994 -0.3047  0.8887  2.499
## cloglog(P[Y<=2]) -3.063 -0.5763  0.2447  0.8725  1.845
## cloglog(P[Y<=3]) -2.930 -0.9202  0.2133  0.3588  1.624
##
## Coefficients:
##           Estimate Std. Error z value Pr(>|z|)
## (Intercept):1 -1.974543   0.275276  -7.173 7.34e-13 ***
## (Intercept):2 -1.004108   0.233233  -4.305 1.67e-05 ***
## (Intercept):3 -0.359446   0.211590  -1.699 0.089359 .

```

```

## TH1          0.754508    0.220125    3.428 0.000609 ***
## AGE          0.000857    0.012260    0.070 0.944274
## AGE2         0.003648    0.001320    2.764 0.005707 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Number of linear predictors:  3
##
## Names of linear predictors:
## cloglog(P[Y<=1]), cloglog(P[Y<=2]), cloglog(P[Y<=3])
##
## Dispersion Parameter for cumulative family:  1
##
## Residual deviance: 320.1936 on 363 degrees of freedom
##
## Log-likelihood: -160.0968 on 363 degrees of freedom
##
## Number of iterations: 6

# Beispiel-Interpretation
exp(coef(modelvglm1b)[4])

##          TH1
## 2.126565

```

Die Wahrscheinlichkeit, eine höhere Kategorie als r zu beobachten, d.h. hier für größere Schmerzen, verändert sich (unabhängig von r) bei Gabe des Medikaments im Vergleich zum Placebo (bei festgehaltenem Geschlecht und Alter) um die Potenz 2.13, d.h. die Wkt. für hohe Kat. /starke Schmerzen sinkt.

Grundsätzlich lassen sich auch noch andere Link-Funktionen verwenden, wie z.B. probit

```

modelvglm1c <- vglm(PAIN ~ TH + AGE + AGE2,
family=cumulative(link="probit",parallel=T))
summary(modelvglm1c)

##
## Call:
## vglm(formula = PAIN ~ TH + AGE + AGE2, family = cumulative(link = "probit",
##      parallel = T))
##
## Pearson residuals:
##          Min       1Q   Median       3Q      Max
## probit(P[Y<=1]) -1.818 -0.8390 -0.2564  0.7818  2.769
## probit(P[Y<=2]) -2.746 -0.5219  0.2912  0.9310  1.681
## probit(P[Y<=3]) -2.412 -1.0452  0.2718  0.3813  1.375
##
## Coefficients:
##          Estimate Std. Error z value Pr(>|z|)
## (Intercept):1 -1.237874    0.207336  -5.970 2.37e-09 ***
## (Intercept):2 -0.480287    0.190777  -2.518  0.01182 *
## (Intercept):3  0.129685    0.188957   0.686  0.49251
## TH1           0.562382    0.200469   2.805  0.00503 **
## AGE           0.003195    0.011507   0.278  0.78125
## AGE2          0.003083    0.001276   2.416  0.01571 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Number of linear predictors:  3
##

```



```
## Names of linear predictors: probit(P[Y<=1]), probit(P[Y<=2]), probit(P[Y<=3])
##
## Dispersion Parameter for cumulative family: 1
##
## Residual deviance: 324.5491 on 363 degrees of freedom
##
## Log-likelihood: -162.2745 on 363 degrees of freedom
##
## Number of iterations: 5
```

Dann lassen sich die über die Kategorien konstanten Koeffizienten  $\gamma$  aber nicht mehr so schön interpretieren.

- d) Um ein sequentielles Modell der Form  $P(y = r | Y \geq r, x)$  zu fitten, verwende man als family sratio.

Wir verwenden zunächst den komplementären loglog Link

```
modelvglm2b <- vglm(PAIN ~ TH + AGE + AGE2,
family=sratio(link="cloglog",parallel=T))
summary(modelvglm2b)

##
## Call:
## vglm(formula = PAIN ~ TH + AGE + AGE2, family = sratio(link = "cloglog",
## parallel = T))
##
## Pearson residuals:
##           Min          1Q          Median          3Q          Max
## cloglog(P[Y=1|Y>=1]) -1.437 -0.6523 -4.683e-01 0.9087638 2.585
## cloglog(P[Y=2|Y>=2]) -2.703 -0.6883 -6.533e-05 1.1101240 2.089
## cloglog(P[Y=3|Y>=3]) -2.623 -0.7608 4.924e-07 0.0000431 1.905
##
## Coefficients:
##           Estimate Std. Error z value Pr(>|z|)
## (Intercept):1 -1.9745434 0.2752763 -7.173 7.34e-13 ***
## (Intercept):2 -1.4804000 0.2556277 -5.791 6.99e-09 ***
## (Intercept):3 -1.1035506 0.2556045 -4.317 1.58e-05 ***
## TH1           0.7545087 0.2201250 3.428 0.000609 ***
## AGE           0.0008569 0.0122599 0.070 0.944275
## AGE2          0.0036485 0.0013199 2.764 0.005707 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Number of linear predictors: 3
##
## Names of linear predictors:
## cloglog(P[Y=1|Y>=1]), cloglog(P[Y=2|Y>=2]), cloglog(P[Y=3|Y>=3])
##
## Dispersion Parameter for sratio family: 1
##
## Residual deviance: 320.1936 on 363 degrees of freedom
##
## Log-likelihood: -160.0968 on 363 degrees of freedom
##
## Number of iterations: 6

# Vergleich mit kumulativem Modell
summary(modelvglm1b)
```

```

##
## Call:
## vglm(formula = PAIN ~ TH + AGE + AGE2, family = cumulative(link = "cloglog",
##     parallel = T))
##
## Pearson residuals:
##           Min         1Q   Median       3Q      Max
## cloglog(P[Y<=1]) -1.552 -0.8994 -0.3047  0.8887  2.499
## cloglog(P[Y<=2]) -3.063 -0.5763  0.2447  0.8725  1.845
## cloglog(P[Y<=3]) -2.930 -0.9202  0.2133  0.3588  1.624
##
## Coefficients:
##           Estimate Std. Error z value Pr(>|z|)
## (Intercept):1 -1.974543   0.275276  -7.173 7.34e-13 ***
## (Intercept):2 -1.004108   0.233233  -4.305 1.67e-05 ***
## (Intercept):3 -0.359446   0.211590  -1.699 0.089359 .
## TH1           0.754508   0.220125   3.428 0.000609 ***
## AGE           0.000857   0.012260   0.070 0.944274
## AGE2          0.003648   0.001320   2.764 0.005707 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Number of linear predictors: 3
##
## Names of linear predictors:
## cloglog(P[Y<=1]), cloglog(P[Y<=2]), cloglog(P[Y<=3])
##
## Dispersion Parameter for cumulative family: 1
##
## Residual deviance: 320.1936 on 363 degrees of freedom
##
## Log-likelihood: -160.0968 on 363 degrees of freedom
##
## Number of iterations: 6

# Die \gamma Koeffizienten sind gleich (vgl. Blatt 9)

# Zugriff auf Koeffizienten mit der Function
coef(modelvglm2b)

## (Intercept):1 (Intercept):2 (Intercept):3          TH1          AGE
## -1.9745434355 -1.4803999703 -1.1035506304  0.7545087066  0.0008569391
##           AGE2
## 0.0036484906

# oder auf den entsprechenden Slot mittels @
modelvglm2b@coefficients

## (Intercept):1 (Intercept):2 (Intercept):3          TH1          AGE
## -1.9745434355 -1.4803999703 -1.1035506304  0.7545087066  0.0008569391
##           AGE2
## 0.0036484906

# Die entsprechenden \theta des kumulativen Modells sind also
theta <- modelvglm1b@coefficients[1:3]

# Wir berechnen
log(exp(theta)[2:3] - exp(theta)[1:2])

## (Intercept):2 (Intercept):3
## -1.480400 -1.103551

```

```

# und vergleichen mit
modelvglm2b@coefficients[2:3]

## (Intercept):2 (Intercept):3
##      -1.480400      -1.103551

# Wie auf Blatt 9 gezeigt sind diese (bis auf numerische Ungenauigkeiten) gleich

```

Als nächstes berechnen wir ein sequentielles Logit- und Probit-Modell

```

modelvglm2a <- vglm(PAIN ~ TH + AGE + AGE2,
family=sratio(link="logit",parallel=T))
summary(modelvglm2a)

##
## Call:
## vglm(formula = PAIN ~ TH + AGE + AGE2, family = sratio(link = "logit",
##      parallel = T))
##
## Pearson residuals:
##           Min           1Q           Median           3Q          Max
## logit(P[Y=1|Y>=1]) -1.405 -0.6658 -4.571e-01 8.816e-01 2.669
## logit(P[Y=2|Y>=2]) -2.441 -0.6823 -1.036e-04 1.151e+00 2.140
## logit(P[Y=3|Y>=3]) -2.577 -0.7666 -1.756e-05 4.239e-05 1.852
##
## Coefficients:
##           Estimate Std. Error z value Pr(>|z|)
## (Intercept):1 -1.972883   0.331040  -5.960 2.53e-09 ***
## (Intercept):2 -1.415051   0.313332  -4.516 6.30e-06 ***
## (Intercept):3 -0.891371   0.324442  -2.747 0.006007 **
## TH1           0.937492   0.278234   3.369 0.000753 ***
## AGE           0.004472   0.015689   0.285 0.775589
## AGE2          0.004519   0.001779   2.540 0.011093 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Number of linear predictors:  3
##
## Names of linear predictors:
## logit(P[Y=1|Y>=1]), logit(P[Y=2|Y>=2]), logit(P[Y=3|Y>=3])
##
## Dispersion Parameter for sratio family:  1
##
## Residual deviance: 320.687 on 363 degrees of freedom
##
## Log-likelihood: -160.3435 on 363 degrees of freedom
##
## Number of iterations: 5

modelvglm2c <- vglm(PAIN ~ TH + AGE + AGE2,
family=sratio(link="probit",parallel=T))
summary(modelvglm2c)

##
## Call:
## vglm(formula = PAIN ~ TH + AGE + AGE2, family = sratio(link = "probit",
##      parallel = T))
##
## Pearson residuals:

```

```

##              Min      1Q      Median      3Q      Max
## probit(P[Y=1|Y>=1]) -1.383 -0.6762 -4.625e-01 8.870e-01 2.686
## probit(P[Y=2|Y>=2]) -2.347 -0.6859 -8.886e-05 1.181e+00 2.139
## probit(P[Y=3|Y>=3]) -2.509 -0.7758 -1.245e-05 3.851e-05 1.827
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept):1 -1.171687   0.189277  -6.190 6.00e-10 ***
## (Intercept):2 -0.859150   0.185365  -4.635 3.57e-06 ***
## (Intercept):3 -0.535333   0.197304  -2.713 0.006663 **
## TH1            0.554832   0.165954   3.343 0.000828 ***
## AGE            0.002690   0.009490   0.283 0.776827
## AGE2           0.002686   0.001069   2.512 0.012021 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Number of linear predictors:  3
##
## Names of linear predictors:
## probit(P[Y=1|Y>=1]), probit(P[Y=2|Y>=2]), probit(P[Y=3|Y>=3])
##
## Dispersion Parameter for sratio family:  1
##
## Residual deviance: 321.1369 on 363 degrees of freedom
##
## Log-likelihood: -160.5684 on 363 degrees of freedom
##
## Number of iterations: 5

```

Nach der Devianz zu urteilen bieten das komplementäre loglog-Modell und das sequentielle Logit-Modell die beste Anpassung an die Daten.

- e) Höhere Flexibilität ergibt sich, wenn man auch über Kategorien variierende  $\gamma$  Vektoren zulässt

```

# Wir betrachten zum Beispiel ein sequentielles Logit-Modell mit veränderlichen
# Koeffizienten für Therapie
modelvglm3a <- vglm(PAIN ~ TH + AGE + AGE2,
family=sratio(link="logit",parallel=F ~ TH))
summary(modelvglm3a)

##
## Call:
## vglm(formula = PAIN ~ TH + AGE + AGE2, family = sratio(link = "logit",
##   parallel = F ~ TH))
##
## Pearson residuals:
##              Min      1Q      Median      3Q      Max
## logit(P[Y=1|Y>=1]) -1.168 -0.6057 -5.206e-01 0.8403771 2.105
## logit(P[Y=2|Y>=2]) -2.758 -0.5128 -7.040e-05 0.8578864 3.104
## logit(P[Y=3|Y>=3]) -2.735 -0.7601 -1.662e-05 0.0000283 1.815
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept):1 -1.496431   0.350278  -4.272 1.94e-05 ***
## (Intercept):2 -2.083964   0.449676  -4.634 3.58e-06 ***
## (Intercept):3 -0.878360   0.368750  -2.382  0.0172 *
## TH1:1          0.095838   0.417536   0.230  0.8185
## TH1:2          2.052499   0.517988   3.962 7.42e-05 ***
## TH1:3          0.908919   0.580481   1.566  0.1174
## AGE            0.004989   0.015990   0.312  0.7550
## AGE2           0.004477   0.001811   2.473  0.0134 *

```

```

## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Number of linear predictors:  3
##
## Names of linear predictors:
## logit(P[Y=1|Y>=1]), logit(P[Y=2|Y>=2]), logit(P[Y=3|Y>=3])
##
## Dispersion Parameter for sratio family:  1
##
## Residual deviance: 311.4967 on 361 degrees of freedom
##
## Log-likelihood: -155.7484 on 361 degrees of freedom
##
## Number of iterations: 5

# Modellvergleich über die Devianz
1 - pchisq(deviance(modelvglm2a) - deviance(modelvglm3a), df=2)

## [1] 0.01010085

# Der Vergleich legt veränderliche Koeffizienten nahe.

# auch Alterseffekt nicht restringiert
modelvglm4a <- vglm(PAIN ~ TH + AGE + AGE2,
family=sratio(link="logit",parallel=F))
summary(modelvglm4a)

##
## Call:
## vglm(formula = PAIN ~ TH + AGE + AGE2, family = sratio(link = "logit",
##      parallel = F))
##
## Pearson residuals:
##           Min          1Q          Median          3Q          Max
## logit(P[Y=1|Y>=1]) -1.192 -0.6045 -5.188e-01 8.225e-01 2.120
## logit(P[Y=2|Y>=2]) -2.836 -0.5297 -1.545e-04 7.914e-01 3.229
## logit(P[Y=3|Y>=3]) -2.753 -0.8140 -1.824e-05 5.332e-05 1.885
##
## Coefficients:
##           Estimate Std. Error z value Pr(>|z|)
## (Intercept):1 -1.510744   0.391916  -3.855 0.000116 ***
## (Intercept):2 -2.231566   0.544132  -4.101 4.11e-05 ***
## (Intercept):3 -0.657054   0.460565  -1.427 0.153688
## TH1:1          0.097533   0.419668   0.232 0.816222
## TH1:2          2.035034   0.527682   3.857 0.000115 ***
## TH1:3          0.936574   0.585536   1.600 0.109706
## AGE:1          0.005404   0.023747   0.228 0.819967
## AGE:2         -0.011792   0.028902  -0.408 0.683290
## AGE:3          0.025830   0.033934   0.761 0.446551
## AGE2:1         0.004605   0.002552   1.805 0.071128 .
## AGE2:2         0.006176   0.003447   1.792 0.073143 .
## AGE2:3         0.001359   0.004197   0.324 0.746069
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Number of linear predictors:  3
##
## Names of linear predictors:
## logit(P[Y=1|Y>=1]), logit(P[Y=2|Y>=2]), logit(P[Y=3|Y>=3])

```

```

##
## Dispersion Parameter for sratio family: 1
##
## Residual deviance: 310.3534 on 357 degrees of freedom
##
## Log-likelihood: -155.1767 on 357 degrees of freedom
##
## Number of iterations: 6

# Modellvergleich
1 - pchisq(deviance(modelvglm3a) - deviance(modelvglm4a), df=4)

## [1] 0.8873394

# Das restringierte Modell sollte beibehalten werden.

# Das Ergebnis für das komplementäre loglog-Modell ist ähnlich
modelvglm3b <- vglm(PAIN ~ TH + AGE + AGE2,
family=sratio(link="cloglog",parallel=F ~ TH))
summary(modelvglm3b)

##
## Call:
## vglm(formula = PAIN ~ TH + AGE + AGE2, family = sratio(link = "cloglog",
## parallel = F ~ TH))
##
## Pearson residuals:
##           Min       1Q   Median       3Q      Max
## cloglog(P[Y=1|Y>=1]) -1.140 -0.6125 -5.244e-01 8.531e-01 2.051
## cloglog(P[Y=2|Y>=2]) -3.137 -0.5149 -2.971e-05 8.435e-01 3.007
## cloglog(P[Y=3|Y>=3]) -2.697 -0.7666 -9.662e-06 1.188e-05 1.836
##
## Coefficients:
##           Estimate Std. Error z value Pr(>|z|)
## (Intercept):1 -1.549657  0.293311  -5.283 1.27e-07 ***
## (Intercept):2 -2.077039  0.398020  -5.218 1.80e-07 ***
## (Intercept):3 -1.064672  0.295595  -3.602 0.000316 ***
## TH1:1          0.061775  0.350925   0.176 0.860267
## TH1:2          1.628822  0.428458   3.802 0.000144 ***
## TH1:3          0.682010  0.414707   1.645 0.100061
## AGE            0.001149  0.012328   0.093 0.925722
## AGE2           0.003555  0.001328   2.676 0.007445 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Number of linear predictors: 3
##
## Names of linear predictors:
## cloglog(P[Y=1|Y>=1]), cloglog(P[Y=2|Y>=2]), cloglog(P[Y=3|Y>=3])
##
## Dispersion Parameter for sratio family: 1
##
## Residual deviance: 311.6042 on 361 degrees of freedom
##
## Log-likelihood: -155.8021 on 361 degrees of freedom
##
## Number of iterations: 6

1 - pchisq(deviance(modelvglm2b) - deviance(modelvglm3b), df=2)

```

```

## [1] 0.01364066

modelvglm4b <- vglm(PAIN ~ TH + AGE + AGE2,
family=sratio(link="cloglog",parallel=F))
summary(modelvglm4b)

##
## Call:
## vglm(formula = PAIN ~ TH + AGE + AGE2, family = sratio(link = "cloglog",
## parallel = F))
##
## Pearson residuals:
##           Min           1Q           Median           3Q          Max
## cloglog(P[Y=1|Y>=1]) -1.278 -0.6020 -5.172e-01 0.7569329 2.111
## cloglog(P[Y=2|Y>=2]) -2.888 -0.5175 -1.890e-05 0.7719314 2.990
## cloglog(P[Y=3|Y>=3]) -2.549 -0.8359 -1.675e-05 0.0001489 1.833
##
## Coefficients:
##           Estimate Std. Error z value Pr(>|z|)
## (Intercept):1 -1.604851   0.335518  -4.783 1.73e-06 ***
## (Intercept):2 -2.119479   0.452586  -4.683 2.83e-06 ***
## (Intercept):3 -0.877648   0.358743  -2.446 0.014427 *
## TH1:1          0.068890   0.353493   0.195 0.845484
## TH1:2          1.604039   0.431221   3.720 0.000199 ***
## TH1:3          0.645505   0.416589   1.550 0.121261
## AGE:1          0.003754   0.020233   0.186 0.852818
## AGE:2         -0.010952   0.020653  -0.530 0.595925
## AGE:3          0.012160   0.024815   0.490 0.624109
## AGE2:1         0.003967   0.001974   2.009 0.044505 *
## AGE2:2         0.004095   0.002478   1.653 0.098374 .
## AGE2:3         0.001321   0.002968   0.445 0.656360
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Number of linear predictors:  3
##
## Names of linear predictors:
## cloglog(P[Y=1|Y>=1]), cloglog(P[Y=2|Y>=2]), cloglog(P[Y=3|Y>=3])
##
## Dispersion Parameter for sratio family:  1
##
## Residual deviance: 310.6111 on 357 degrees of freedom
##
## Log-likelihood: -155.3056 on 357 degrees of freedom
##
## Number of iterations: 5

1 - pchisq(deviance(modelvglm3b) - deviance(modelvglm4b), df=4)

## [1] 0.9108498

# Auch das kumulative Logit-Modell kann mit kategorie-spezifischem
# Therapie-Effekt stark verbessert werden.
modelvglm5a <- vglm(PAIN ~ TH + AGE + AGE2,
family=cumulative(link="logit",parallel=F ~ TH))
summary(modelvglm5a)

##
## Call:
## vglm(formula = PAIN ~ TH + AGE + AGE2, family = cumulative(link = "logit",

```

```

##      parallel = F ~ TH))
##
## Pearson residuals:
##           Min      1Q  Median      3Q      Max
## logit(P[Y<=1]) -2.003 -0.6890 -0.2637  0.7094  1.996
## logit(P[Y<=2]) -3.246 -0.5533  0.2622  0.7385  2.487
## logit(P[Y<=3]) -2.949 -0.9292  0.2477  0.4257  1.494
##
## Coefficients:
##           Estimate Std. Error z value Pr(>|z|)
## (Intercept):1 -1.552377   0.367791  -4.221 2.43e-05 ***
## (Intercept):2 -0.982209   0.336440  -2.919 0.003507 **
## (Intercept):3  0.003730   0.321897   0.012 0.990754
## TH1:1          0.115109   0.419890   0.274 0.783976
## TH1:2          1.403024   0.391966   3.579 0.000344 ***
## TH1:3          1.536001   0.466414   3.293 0.000990 ***
## AGE            0.009308   0.019335   0.481 0.630240
## AGE2           0.004841   0.002182   2.218 0.026533 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Number of linear predictors:  3
##
## Names of linear predictors: logit(P[Y<=1]), logit(P[Y<=2]), logit(P[Y<=3])
##
## Dispersion Parameter for cumulative family:  1
##
## Residual deviance: 311.624 on 361 degrees of freedom
##
## Log-likelihood: -155.812 on 361 degrees of freedom
##
## Number of iterations: 5

```

```
summary(modelvglm1a)
```

```

##
## Call:
## vglm(formula = PAIN ~ TH + AGE + AGE2, family = cumulative(parallel = T))
##
## Pearson residuals:
##           Min      1Q  Median      3Q      Max
## logit(P[Y<=1]) -1.933 -0.8242 -0.2573  0.7619  2.825
## logit(P[Y<=2]) -2.961 -0.5013  0.2934  0.8858  1.813
## logit(P[Y<=3]) -2.429 -1.0076  0.2705  0.3895  1.393
##
## Coefficients:
##           Estimate Std. Error z value Pr(>|z|)
## (Intercept):1 -2.140711   0.365604  -5.855 4.76e-09 ***
## (Intercept):2 -0.875355   0.318713  -2.747 0.00602 **
## (Intercept):3  0.141804   0.311846   0.455 0.64931
## TH1            1.006955   0.338947   2.971 0.00297 **
## AGE            0.007508   0.019149   0.392 0.69501
## AGE2           0.005188   0.002155   2.408 0.01606 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Number of linear predictors:  3
##
## Names of linear predictors: logit(P[Y<=1]), logit(P[Y<=2]), logit(P[Y<=3])
##
## Dispersion Parameter for cumulative family:  1
##

```



```
## Residual deviance: 323.4583 on 363 degrees of freedom
##
## Log-likelihood: -161.7292 on 363 degrees of freedom
##
## Number of iterations: 6

1 - pchisq(deviance(modelvglm1a) - deviance(modelvglm5a), df=2)

## [1] 0.002692784
```