```r
set.seed(42)
load('awards.RData')
load('DebTrivedi.RData')
```

Aufgabe 1

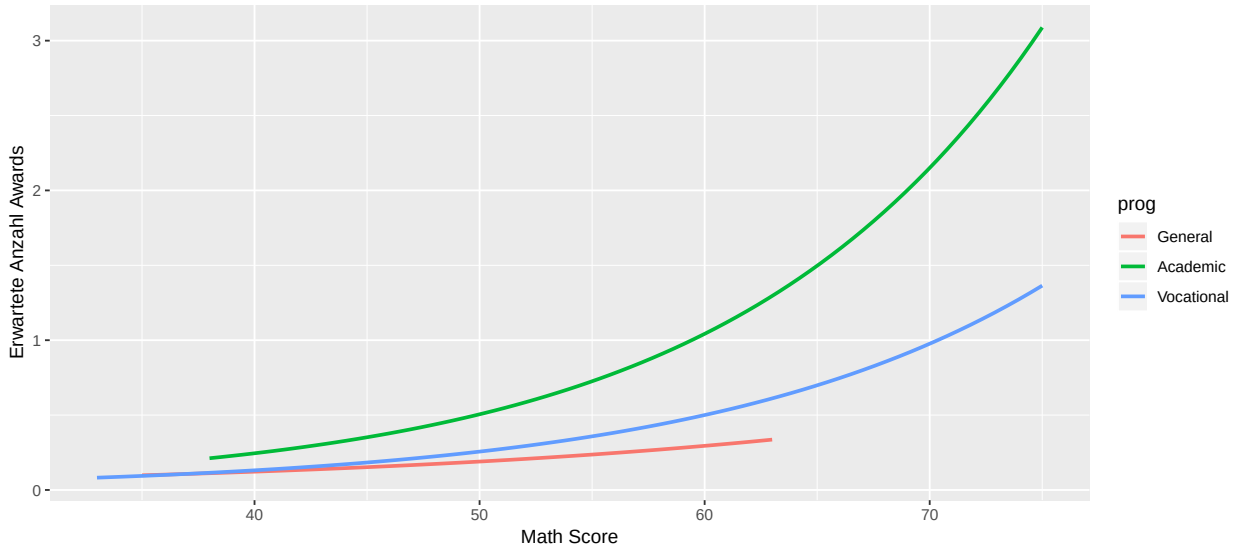
a)

```r
data = awards

# poisson regression
plm = glm(
  num_awards ~ prog + math,
  data = data,
  family= poisson
)
summary(plm)
#>
#> Call:
#> glm(formula = num_awards ~ prog + math, family = poisson, data = data)
#>
#> Deviance Residuals:
#>     Min       1Q   Median       3Q      Max
#> -2.2043  -0.8436  -0.5106   0.2558   2.6796
#>
#> Coefficients:
#>                Estimate Std. Error z value Pr(>|z|)
#> (Intercept)    -5.24712    0.65845  -7.969 1.60e-15 ***
#> progAcademic    1.08386    0.35825   3.025  0.00248 **
#> progVocational  0.36981    0.44107   0.838  0.40179
#> math            0.07015    0.01060   6.619 3.63e-11 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> (Dispersion parameter for poisson family taken to be 1)
#>
#>     Null deviance: 287.67  on 199  degrees of freedom
#> Residual deviance: 189.45  on 196  degrees of freedom
#> AIC: 373.5
#>
#> Number of Fisher Scoring iterations: 6
```

Die Summary zeigt die Poisson-Regressionskoeffizienten für jede der Variablen sowie die Standardfehler, Z-Scores, p-Werte und 95% Konfidenzintervalle für die Koeffizienten. Der Koeffizient für `math` liegt bei 0.07. Die Variable `progAcademic` vergleicht zwischen `prog = Äcademic"` und `prog = "General"` mit einem Koeffizienten von 1.08. Die Variable `prog.Vocational` zeigt die erwartete Differenz in der Anzahl zwischen `prog = "Vocational"` und der Referenzgruppe `(prog = "General")` mit einem Koeffizienten von 0.37.

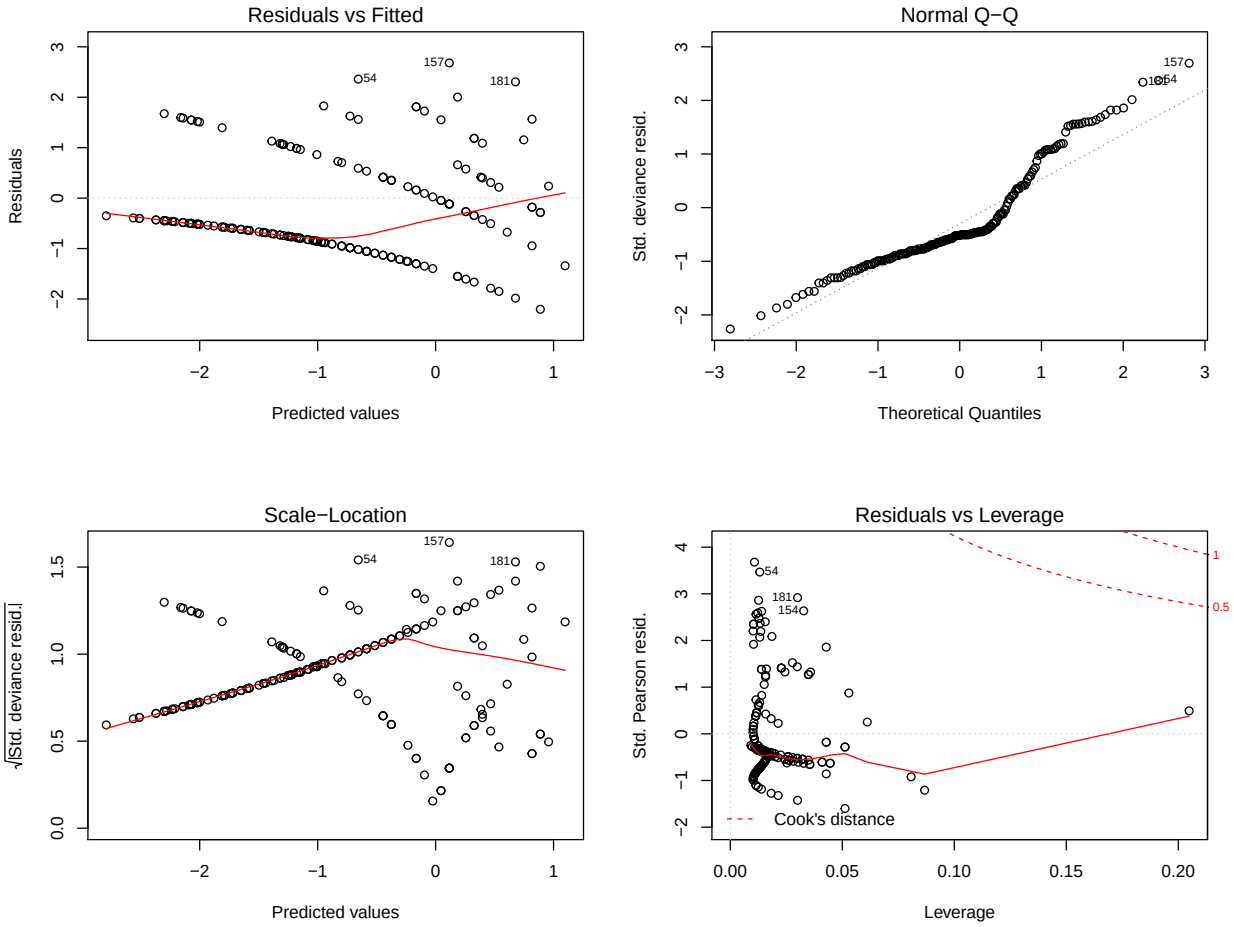
b)

```
library(ggplot2)
#> Warning: package 'ggplot2' was built under R version 3.5.3

pred = predict.glm(
  plm,
  type='response'
)

gg = ggplot(
  data = data,
  mapping = aes(
    x = math,
    y = num_awards,
    color = prog
  )
)
# gg = gg + geom_point()
gg = gg + geom_smooth(
  method = "glm",
  method.args = list(
    family = "poisson"
  ),
  se = FALSE
)
gg = gg + labs(
  x = "Math Score",
  y = "Erwartete Anzahl Awards"
)
gg
#> Warning in plyr::split_indices(scale_id, n): '.Random.seed[1]' ist keine
#> zulässige ganze Zahl, wird also ignoriert
```



c)

```
par(mfrow=c(2,2))
plot(plm)
```



d)

```
lm = lm(
  num_awards ~ math + prog,
  data = data
)

pAIC <- AIC(plm)
lAIC <- AIC(lm)
```

Poisson AIC	Linear AIC
373.5	532.25

Es zeigt sich ein niedrigerer AIC Wert bei der Poisson Regression als bei der linearen Regression. Ein niedriger AIC deutet auf ein besseres Modell hin als ein hoher AIC. Folglich ist die Poisson Regression als ein besseres Modell zu betrachten.

Aufgabe 2

a)

```
summary(poissreg2 <- glm(ofp~health+numchron+hosp+married+medicaid, data=DebTrivedi))
#>
#> Call:
#> glm(formula = ofp ~ health + numchron + hosp + married + medicaid,
#>     data = DebTrivedi)
#>
#> Deviance Residuals:
#>     Min       1Q   Median       3Q      Max
#> -12.044   -3.867   -1.474    1.993   77.729
#>
#> Coefficients:
#>                 Estimate Std. Error t value Pr(>|t|)
#> (Intercept)      4.29698    0.20018  21.465  < 2e-16 ***
#> healthpoor       2.48736    0.30365   8.191 3.35e-16 ***
#> healthexcellent -1.49890    0.36915  -4.060 4.98e-05 ***
#> numchron         0.56995    0.09885   5.766 8.69e-09 ***
#> hosp             3.46726    0.25065  13.833  < 2e-16 ***
#> marriedyes      -0.12063    0.19976  -0.604    0.546
#> medicaidyes      0.20350    0.34913   0.583    0.560
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> (Dispersion parameter for gaussian family taken to be 41.76134)
#>
#>     Null deviance: 201252  on 4405  degrees of freedom
#> Residual deviance: 183708  on 4399  degrees of freedom
#> AIC: 28956
#>
#> Number of Fisher Scoring iterations: 2
```

Die Summary zeigt die Poisson-Regressionskoeffizienten f??r jede der Variablen sowie die Standardfehler, Z-Scores, p-Werte und 95% Konfidenzintervalle f??r die Koeffizienten. Es zeigt sich dass “poor health”ëinen positiven Einfluss auf die Anzahl Arztbesuche im Verlgeich zu einer “average health” hat, w??hrend “excellent health” einen negativen Einfluss hat. Auch die Variable der chronischen Erkrankungen zeigt einen positiven Koeffizienten zur Anzahl Arztbesuche, genau wie die Notwendigkeit eines Krankenhausaufenthalts und staatliche Unterst??tzung. Nur die positive Variable des verheiratet seins zeigt einen negativen Einfluss auf die Anzahl der Arztbesuche.

b)

```r
mean(predict(poissreg2, data.frame(married='yes',medicaid='no',health='average',numchron=2, hosp=0)))
#> [1] 5.316249

Y = poissreg2$coefficients[1]+2*poissreg2$coefficients[3]+poissreg2$coefficients[5]+poissreg2$coefficien

Y
#> (Intercept)
#>    7.253795
```

c)

```r
library('dplyr')
#> Warning: package 'dplyr' was built under R version 3.5.3
cBeob <- matrix(NA,nrow=3,ncol=2)
colnames(cBeob) <- c("hosp 0","hosp 1")
rownames(cBeob) <- c("poor","average","excellent")
cGesch <- matrix(NA,nrow=3,ncol=2)
colnames(cGesch) <- c("hosp 0","hosp 1")
rownames(cGesch) <- c("poor","average","excellent")

cBeob[2,1] <- sum(filter(DebTrivedi,health=='average'&hosp==0)$ofp)
cBeob[1,1] <- sum(filter(DebTrivedi,health=='poor'&hosp==0)$ofp)
cBeob[3,1] <- sum(filter(DebTrivedi,health=='excellent'&hosp==0)$ofp)
cBeob[2,2] <- sum(filter(DebTrivedi,health=='average'&hosp==1)$ofp)
cBeob[1,2] <- sum(filter(DebTrivedi,health=='poor'&hosp==1)$ofp)
cBeob[3,2] <- sum(filter(DebTrivedi,health=='excellent'&hosp==1)$ofp)
cBeob
#>           hosp 0 hosp 1
#> poor        2578   2351
#> average    14014   5323
#> excellent   1018    158

glmCGesch <- glm(data = DebTrivedi, ofp~health+hosp, family="poisson")
cGesch[2,1] <- predict.glm(glmCGesch, data.frame(health='average',hosp=0),type='response')
nk21 <- dim(filter(DebTrivedi,health=='average'&hosp==0))[1]
cGesch[1,1] <- predict.glm(glmCGesch, data.frame(health='poor',hosp=0),type='response')
nk11 <- dim(filter(DebTrivedi,health=='poor'&hosp==0))[1]
cGesch[3,1] <- predict.glm(glmCGesch, data.frame(health='excellent',hosp=0),type='response')
nk31 <- dim(filter(DebTrivedi,health=='excellent'&hosp==0))[1]
cGesch[2,2] <- predict.glm(glmCGesch, data.frame(health='average',hosp=1),type='response')
nk22 <- dim(filter(DebTrivedi,health=='average'&hosp==1))[1]
cGesch[1,2] <- predict.glm(glmCGesch, data.frame(health='poor',hosp=1),type='response')
nk12 <- dim(filter(DebTrivedi,health=='poor'&hosp==1))[1]
cGesch[3,2] <- predict.glm(glmCGesch, data.frame(health='excellent',hosp=1),type='response')
nk32 <- dim(filter(DebTrivedi,health=='excellent'&hosp==1))[1]
cGesch
#>             hosp 0    hosp 1
#> poor      7.032564 11.814785
#> average   4.920448  8.266407
#> excellent 3.236100  5.436685
```

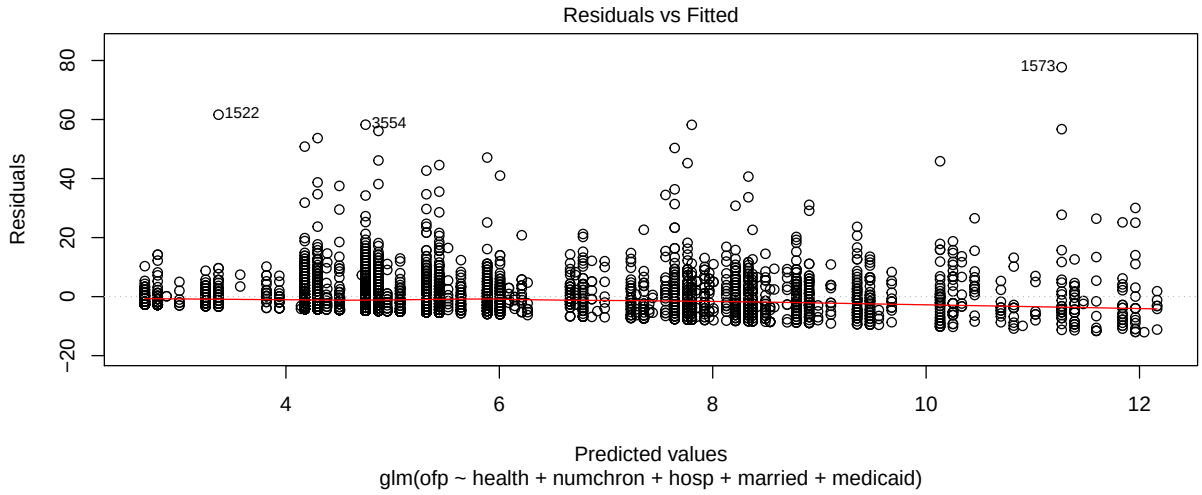
```
chisq <- ((cBeob[1,1]-nk11*cGesch[1,1])^2)/(nk11*cGesch[1,1])+
  ((cBeob[2,1]-nk21*cGesch[2,1])^2)/(nk21*cGesch[2,1])+
  ((cBeob[3,1]-nk31*cGesch[3,1])^2)/(nk31*cGesch[3,1])+
  ((cBeob[1,2]-nk12*cGesch[1,2])^2)/(nk12*cGesch[1,2])+
  ((cBeob[2,2]-nk22*cGesch[2,2])^2)/(nk22*cGesch[2,2])+
  ((cBeob[3,2]-nk32*cGesch[3,2])^2)/(nk32*cGesch[3,2])

chisq
#> [1] 44.29864
```

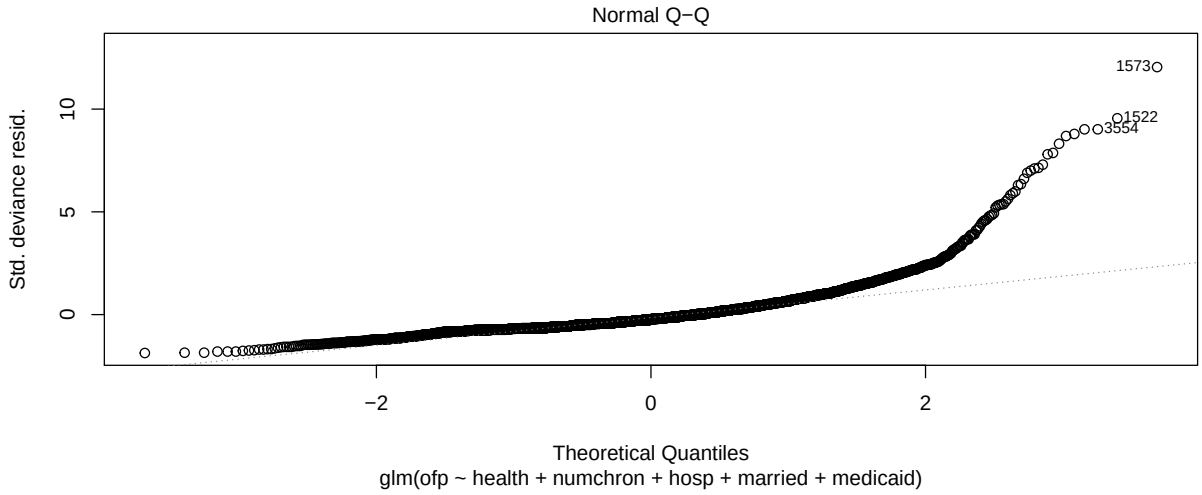
H0: Erwartete und beobachtete H??ufigkeiten sind gleichverteilt. H1: Erwartete und beobachtete H??ufigkeiten sind nicht gleichverteilt.

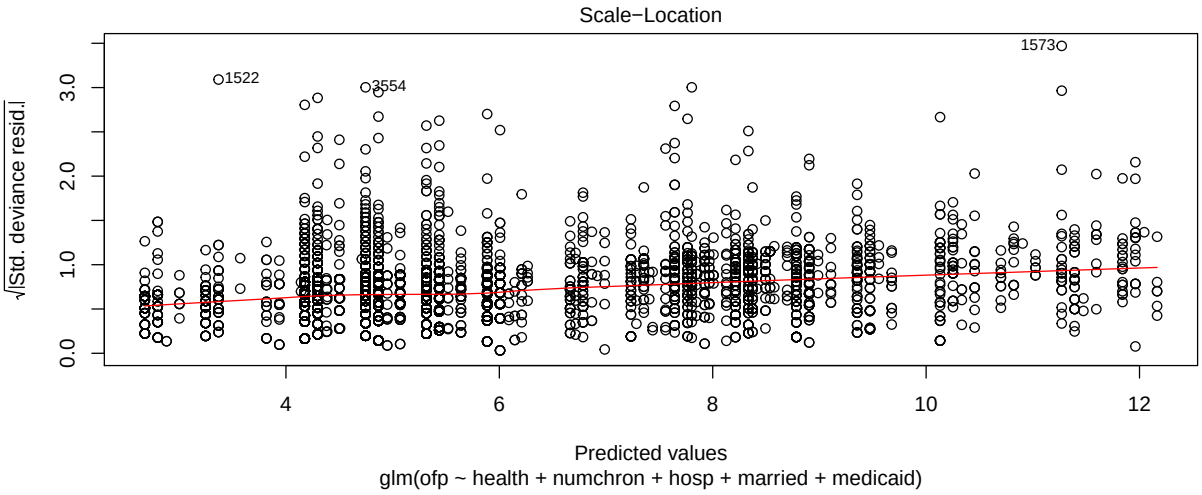
Der errechnete Wert des Chi-Quadrat Anpassungstests liegt ??ber 11,07 und damit im Ablehnungsbereich. H0 kann also verworfen und H1 angenommen werden.

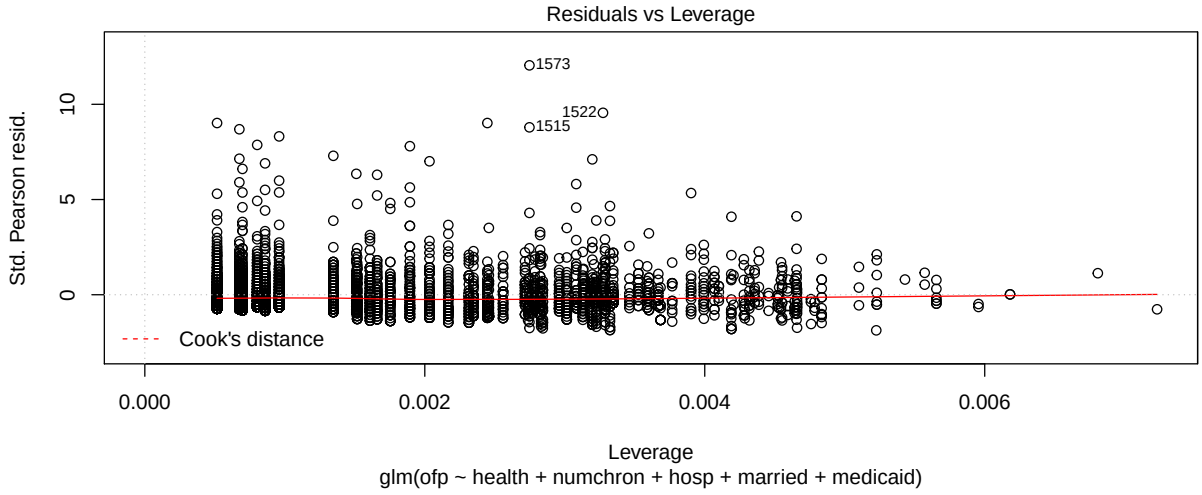
d)

```
plot(poissreg2)
```




Normal Q−Q
Std. deviance resid.
Theoretical Quantiles
glm(ofp ~ health + numchron + hosp + married + medicaid)
1573
1522
3554


Scale−Location
√|Std. deviance resid.|
Predicted values
glm(ofp ~ health + numchron + hosp + married + medicaid)
1573
1522
3554


Residuals vs Leverage
Std. Pearson resid.
Leverage
glm(ofp ~ health + numchron + hosp + married + medicaid)
Cook's distance
1573
1522
1515