Count Modelle - diskrete Änderungen darstellen in R

Benjamin Schlegel

17. Oktober 2016

In diesem Artikel wird erklärt, wie man diskrete Änderungen (discrete changes) für Count Modelle (Poisson, Quasi Poisson Negativ Binomial) berechnet und diese anschliessend darstellen kann.

Im Artikel Poisson, Qusi Poisson oder Negativ Binomial? wird erklärt, wie man die drei Modelle schätzt und anschliessend entscheiden kann, welchem dieser drei Modelle den Vorzug gegeben werden soll. Dieser Artikel baut auf diesem auf und verwendet deshalb das Quasi Poisson Modell.

Zuerst werden wieder die Daten eingelesen und das Quasi Poisson Modell gerechnet. Das Modell regressiert die Anzahl Affären auf Kinder haben und die Religiosität.

```
library(COUNT)
data("affairs")
affairs$kids = factor(affairs$kids)
relig.matrix = as.matrix(affairs[,8:12])
affairs$religion = factor(relig.matrix%*%(1:ncol(relig.matrix)),
    labels = c("anti religious","not religious","slightly religious",
               "somewhat religious","very religious"))
model.quasi = glm(naffairs ~ kids + religion, data=affairs, family=quasipoisson)
summary(model.quasi)
```

```
Call:
glm(formula = naffairs ~ kids + religion, family = quasipoisson,
    data = affairs)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.5060  -1.7709  -1.3952  -0.9859   7.3434

Coefficients:
                            Estimate Std. Error t value Pr(>|t|)
(Intercept)                   0.4498     0.3074   1.463 0.143863
kids1                         0.6944     0.2401   2.892 0.003970 **
religionnot religious        -0.4397     0.2931  -1.500 0.134123
religionslightly religious   -0.3168     0.2963  -1.069 0.285403
religion somewhat religious  -1.1713     0.3227  -3.629 0.000309 ***
religionvery religious       -1.1601     0.4212  -2.754 0.006066 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for quasipoisson family taken to be 7.308601)

    Null deviance: 2925.5  on 600  degrees of freedom
Residual deviance: 2722.6  on 595  degrees of freedom
```

```
AIC: NA

Number of Fisher Scoring iterations: 7
```

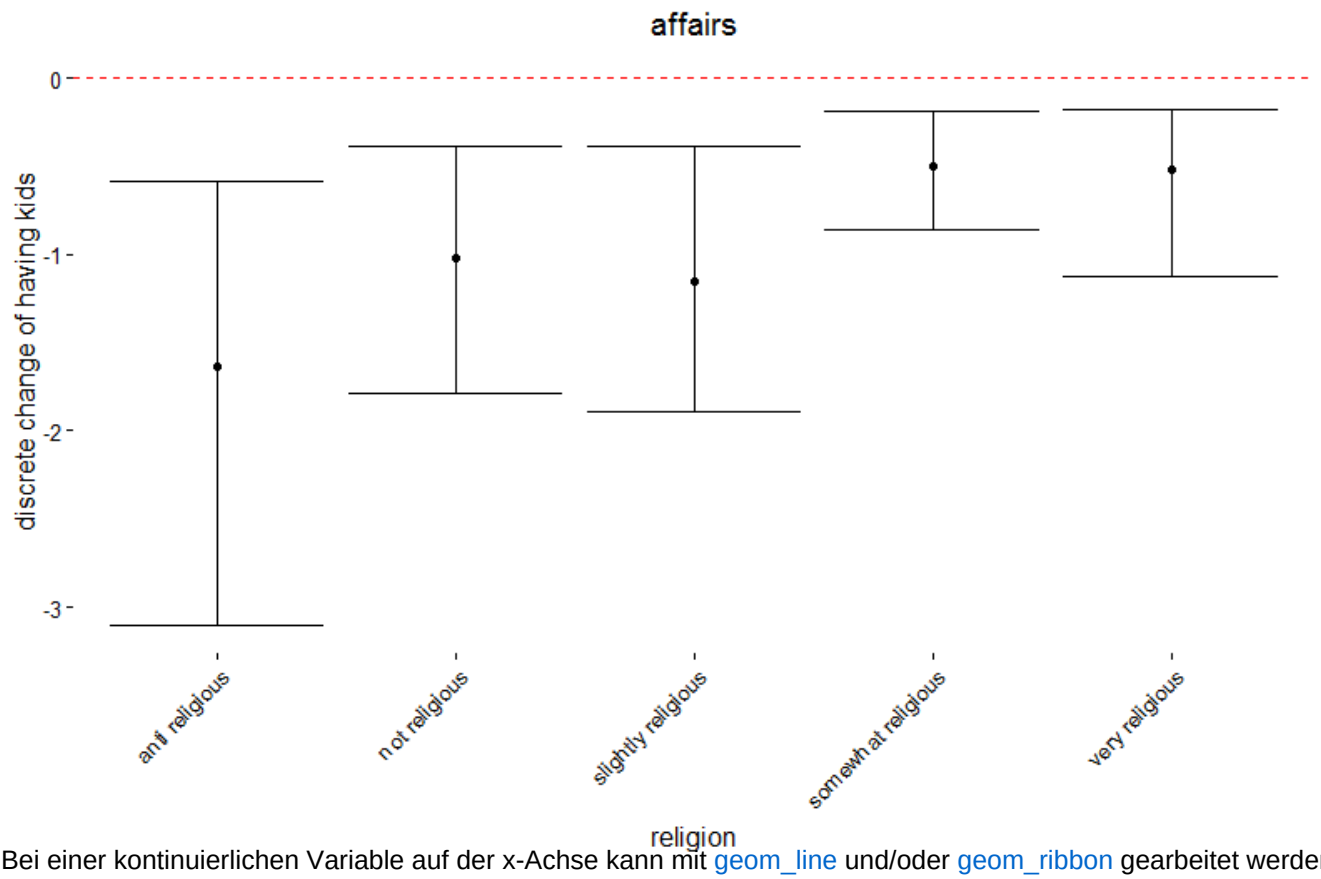
Als nächstes werden die diskreten Änderungen für Kinder haben berechnen. Dazu kann die Funktion predicts aus der Bibliothek glm.predict verwendet werden.

```
library(glm.predict)
output = predicts(model.quasi,"F;F", position = 1)
output
```

```
  val1_mean val1_lower val1_upper val2_mean val2_lower val2_upper    dc_mean   dc_lower
 dc_upper kids_val1 kids_val2           religion
1 1.6259891  0.8574781  2.7504034 3.1959447  1.9148637   4.898056 -1.5699555 -2.9399766
-0.4202833         0         1     anti religious
2 1.0303588  0.6204725  1.6102396 2.0545780  1.4089705   2.814969 -1.0242191 -1.7245441
-0.4118740         0         1      not religious
3 1.1926255  0.6947076  1.9796240 2.3341549  1.6143378   3.172421 -1.1415294 -1.9180069
-0.4215088         0         1 slightly religious
4 0.5059256  0.2728331  0.8524498 0.9930625  0.6331944   1.472394 -0.4871368 -0.8701596
-0.1693827         0         1 somewhat religious
5 0.5321323  0.2261750  1.0911512 1.0396026  0.5242662   1.851278 -0.5074703 -1.0262726
-0.1627877         0         1     very religious
```

Wir sehen, dass der Unterschied zwischen Kinder haben und keine Kinder haben bei allen Ausprägungen von Religiosität signifikant ist. Nun wollen wir die Resultate grafisch darstellen mit Hilfe von ggplot2. Damit die Religiosität in der richten Reihenfolge dargestellt wird, definieren wir die Variable als Faktor mit der korrekten Reihenfolge. Mit geom_errorbar können die Fehlerbalken gezeichnet werden. Mit geom_hline kann die Null-Linie gezeichnet werden, welche für die Signifikanz relevant ist.

```
output$religion = factor(output$religion,levels =
   c("anti religious","not religious","slightly religious",
   "somewhat religious","very religious"))
library(ggplot2)
ggplot(output,aes(x=religion)) + geom_point(aes(y=dc_mean)) +
  geom_errorbar(aes(ymax=dc_upper,ymin=dc_lower)) +
  geom_hline(yintercept = 0,linetype="dashed",col="red") +
  theme_classic() + ylab("discrete change of having kids") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  ggtitle("affairs")
```



Bei einer kontinuierlichen Variable auf der x-Achse kann mit geom_line und/oder geom_ribbon gearbeitet werden.