

Stats 111 Homework 5

Owen Lin

1. a. The model doesn't need an offset term. $\ln(Admissions) = \beta_0 + \beta_1 * Temperature$.
 b. $\ln(Admissions) = 1.971 + 0.025 * Temperature$.
 1 Fahrenheit increase in temperature leads to a relative change in expected count by $e^{0.025}$.
 15 Fahrenheit increase in temperature leads to a relative change in expected count by $e^{0.025*15}$.

```
ERtemp = read.csv("D:\\Coding\\Stats111\\Data\\ERtemp.csv", header=TRUE)
ER.model = glm(Admissions~Temperature, family=poisson, data=ERtemp)
summary(ER.model)
```

```
##
## Call:
## glm(formula = Admissions ~ Temperature, family = poisson, data = ERtemp)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -3.08968  -0.69711   0.07021   0.68360   2.41351
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.9713619   0.0597449   33.00  <2e-16 ***
## Temperature  0.0254139   0.0007212   35.24  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##      Null deviance: 1461.60  on 199  degrees of freedom
## Residual deviance:  184.97  on 198  degrees of freedom
## AIC: 1341.3
##
## Number of Fisher Scoring iterations: 4
```

- c. The expected count of events when the temperature is 85 degrees is 62.274. The 95% CI for the true expected count is (61.078, 63.493), that we are 95% confident that the true expected count when temperature is 85 degree is between 61.078 and 63.493.

```
linContr.glm(c("(Intercept)", "Temperature"), c(1, 85), model = ER.model)
```

```
##
## Test of H_0: exp( 1*(Intercept) + 85*Temperature ) = 1 :
```

```
##      exp( Est )      se.est      zStat pVal  ci95.lo  ci95.hi
## 1      62.27378  0.00989536  417.5228    0  61.07765  63.49334
```

d. $H_0: \beta_1 = 0$

$H_a: \beta_1 \neq 0$

p-value: ~ 0

conclusion: We reject the null and conclude that the coefficient on temperature is not 0 (it's significant).

e. Null Deviance: 1461.60

Residual Deviance: 184.97

The difference is huge, which support the same conclusion in part d) where temp does gave us information about the expected count in ER Admissions.

2. a. Always using a condom leads to a relative change in expected count by $e^{-0.373}$ to those who don't. The expected number of reinfections for someone who always wears a condom and is followed for 5 years is $5 * e^{-0.373}$

```
std = read.csv("D:\\Coding\\Stats111\\Data\\stdgrp.csv")
std.model = glm(n.reinfect~condom.always+offset(log(yrsfu)), family=poisson, data=std)
summary(std.model)
```

```
##
## Call:
## glm(formula = n.reinfect ~ condom.always + offset(log(yrsfu)),
##      family = poisson, data = std)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.2044  -1.0130  -0.2516   0.7187   3.1648
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.79801    0.06579 -12.129  < 2e-16 ***
## condom.always -0.37318    0.11380  -3.279  0.00104 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##      Null deviance: 364.88  on 257  degrees of freedom
## Residual deviance: 353.73  on 256  degrees of freedom
## AIC: 705.29
##
## Number of Fisher Scoring iterations: 5
```

```
# model = glm(n.reinfect~condom.always, family = poisson, data = std, offset = log(yrsfu))
```

- b. Always using a condom leads to a relative change in expected count by $e^{-0.362}$ to those who don't, holding education level and ethnicity the same.

```
std.model2 = glm(n.reinfect~white+condom.always+edugrp+offset(log(yrsfu)), family=poisson, data=std)
coef(std.model2)
```

```
##      (Intercept)          white  condom.always  edugrp(12.9,18]  edugrp[6,11.9]
##      -0.7604249      -0.3256030      -0.3624320      -0.3926074      0.2104841
```

c. $H_0: \beta_1 = \beta_3 = \beta_4 = 0$

H_a : H_0 is not true

test statistic: 22.395 ~ chi_squared(3)

p-value: ~ 0.0001

conclusion: We reject the null and conclude that race or education level (or both) is significant.

```
lrtest(std.model, std.model2)
```

```
##
## Assumption: Model 1 nested within Model 2
```

```
##      Resid. Df Resid. Dev Df Deviance pValue
## 1          256      353.727
## 2          253      331.332  3    22.395  1e-04
```

d. WLOG, we are 95% confident that going from [11.9, 12.9] edugrp to [12.9, 18] edugrp leads to a relative change in rate that is between 0.465 and 0.981.

The estimated rate for the lowest education group (less than 12), that has white=0 and condom.always=0 is $e^{-0.76+0.21} = 0.577$.

e. The 95% confidence interval for the relative change in rate for condom.always is (0.556, 0.872) holding education and race status constant.

We are 95% confident that always using condom will result in a relative change in rate that is between 0.556 and 0.872.

```
glmCI(std.model2)
```

```
##      exp( Est ) ci95.lo ci95.hi z value Pr(>|z|)
## (Intercept)      0.4675  0.3808  0.5738 -7.2715  0.0000
## white            0.7221  0.5632  0.9258 -2.5677  0.0102
## condom.always    0.6960  0.5557  0.8717 -3.1548  0.0016
## edugrp(12.9,18]  0.6753  0.4645  0.9818 -2.0562  0.0398
## edugrp[6,11.9]   1.2343  0.9837  1.5487  1.8177  0.0691
```

3. a. See output

```
abortion = read.table("D:\\Coding\\Stats111\\Data\\abortion.txt", col.names=c("year", "rel", "edu", "att", "count"))
mfit.abort = multinom( att ~ edu+rel, data=abortion, weights=count )
```

```
## # weights: 18 (10 variable)
## initial value 3556.207978
## iter 10 value 2358.708889
## iter 20 value 2030.003018
## final value 2029.986903
## converged
```

```
summary(mfit.abort)
```

```
## Call:
## multinom(formula = att ~ edu + rel, data = abortion, weights = count)
##
## Coefficients:
##      (Intercept)      eduLow      eduMed      relProt      relSProt
## Neg  -0.7056055   0.02674898 -0.1918151 -0.2124176 -0.5669957
## Pos   1.6887222  -1.13707761 -0.4767824   0.7295730   0.3828038
##
## Std. Errors:
##      (Intercept)      eduLow      eduMed      relProt      relSProt
## Neg   0.2054948  0.2453480  0.2185189  0.1947135  0.2277385
## Pos   0.1221815  0.1472414  0.1238369  0.1152013  0.1247378
##
## Residual Deviance: 4059.974
## AIC: 4079.974
```

$$b. \ln \frac{P(Y_i=Neg|X_i)}{P(Y_i=Mixed|X_i)} = \beta_{0,Neg} + \beta_{1,Neg} * Edu + \beta_{2,Neg} * religion$$

$$\ln \frac{P(Y_i=Pos|X_i)}{P(Y_i=Mixed|X_i)} = \beta_{0,Pos} + \beta_{1,Pos} * Edu + \beta_{2,Pos} * religion$$

c. For someone with low education and religion being protestant:

the probabilities of having Positive attitude is 0.719

the probabilities of having Negative attitude is 0.082

the probabilities of having Mixed attitude is 0.199

For someone with high education and religion being protestant:

the probabilities of having Positive attitude is 0.889

the probabilities of having Negative attitude is 0.032

the probabilities of having Mixed attitude is 0.079

The probability of having Positive attitude as a Protestant increased by $0.889 - 0.719 = 0.17$ percent when comparing low education to high education.

```
# Predict probabilities using multinomial distribution
newdata = data.frame(edu="Low", rel="Prot")
predict(mfit.abort, type="probs", newdata=newdata)
```

```
##           Mix           Neg           Pos
## 0.19955481 0.08184398 0.71860121
```

```
# Another prediction  
newdata = data.frame(edu="High", rel="Prot")  
predict(mfit.abort, type="probs", newdata=newdata)
```

```
##           Mix           Neg           Pos  
## 0.07920157 0.03162579 0.88917263
```