ST 518 - Homework 8

Nora Quick

R Question:

1. Consider the crashi data in the VGAM library that we examined in Lab. Sometimes people will fit multiple linear regression models to data like this, not recognizing that the responses are counts.

a. Using the Time.Cat and Day.Cat variables that we created in Lab, fit a multiple linear regression model to the counts—be sure to include the interaction between Time.Cat and Day.Cat just as we did in the lab. How does the model fit compare to the negative binomial regression model that you fit in lab? Please explain.

```r
data(crashi)

hour <- rownames(crashi) ## grab the hours
crashi2 <- stack(crashi) ## combine 7 columns of crashes into 1
names(crashi2) <- c("Count","Day")
crashi2$Day <- factor(crashi2$Day,levels(crashi2$Day)[c(2,6,7,5,1,3,4)])  # make sure the days are orde
crashi2$Hour <- as.numeric(rep(hour, ncol(crashi)))

crashi2 %<>% mutate(.,Day.Cat = ifelse((Day != "Fri" & Day != "Sat" & Day != "Sun"),"Weekday",as.charac
    breaks = c(-1, 5.5, 11.5, 18.5, 25),
    labels = c("Early.Morn", "Morn", "Afternoon", "Evening")))
head(crashi2)
```

```
##   Count Day Hour Day.Cat   Time.Cat
## 1    16 Mon    0 Weekday Early.Morn
## 2    13 Mon    1 Weekday Early.Morn
## 3     5 Mon    2 Weekday Early.Morn
## 4     6 Mon    3 Weekday Early.Morn
## 5     7 Mon    4 Weekday Early.Morn
## 6    12 Mon    5 Weekday Early.Morn
```

```r
mod <- glm(Count ~ Day.Cat * Time.Cat, data = crashi2)
summary(mod)
```

```
##
## Call:
## glm(formula = Count ~ Day.Cat * Time.Cat, data = crashi2)
##
## Deviance Residuals:
```

```
##     Min       1Q   Median       3Q      Max
## -44.833  -10.808   -1.000    8.042   61.167
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                        20.833      8.528   2.443 0.015719 *
## Day.CatSat                         17.000     12.061   1.410 0.160729
## Day.CatSun                         28.833     12.061   2.391 0.018044 *
## Day.CatWeekday                     -8.667      9.535  -0.909 0.364827
## Time.CatMorn                       53.333     12.061   4.422 1.85e-05 ***
## Time.CatAfternoon                 106.167     11.622   9.135 3.87e-16 ***
## Time.CatEvening                    48.767     12.650   3.855 0.000170 ***
## Day.CatSat:Time.CatMorn           -28.333     17.057  -1.661 0.098751 .
## Day.CatSun:Time.CatMorn           -54.167     17.057  -3.176 0.001810 **
## Day.CatWeekday:Time.CatMorn        11.333     13.485   0.840 0.401967
## Day.CatSat:Time.CatAfternoon      -40.857     16.436  -2.486 0.014009 *
## Day.CatSun:Time.CatAfternoon      -66.833     16.436  -4.066 7.65e-05 ***
## Day.CatWeekday:Time.CatAfternoon  -14.119     12.994  -1.087 0.278944
## Day.CatSat:Time.CatEvening        -21.800     17.889  -1.219 0.224883
## Day.CatSun:Time.CatEvening        -63.233     17.889  -3.535 0.000541 ***
## Day.CatWeekday:Time.CatEvening    -18.783     14.143  -1.328 0.186128
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for gaussian family taken to be 436.4019)
##
##     Null deviance: 255316  on 167  degrees of freedom
## Residual deviance:  66333  on 152  degrees of freedom
## AIC: 1515.1
##
## Number of Fisher Scoring iterations: 2
```

```
mod2 <- glm.nb(Count ~ Day.Cat * Time.Cat, data = crashi2)
summary(mod2)
```

```
##
## Call:
## glm.nb(formula = Count ~ Day.Cat * Time.Cat, data = crashi2,
##     init.theta = 13.71581333, link = log)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.97855  -0.68334  -0.06206   0.52714   2.25245
##
## Coefficients:
##                                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)                       3.03655    0.14196  21.391  < 2e-16 ***
## Day.CatSat                        0.59664    0.19159   3.114 0.001845 **
## Day.CatSun                        0.86878    0.18883   4.601 4.21e-06 ***
## Day.CatWeekday                   -0.53785    0.16314  -3.297 0.000977 ***
## Time.CatMorn                      1.26976    0.18588   6.831 8.42e-12 ***
## Time.CatAfternoon                 1.80763    0.17802  10.154  < 2e-16 ***
## Time.CatEvening                   1.20621    0.19392   6.220 4.97e-10 ***
## Day.CatSat:Time.CatMorn          -0.76247    0.25673  -2.970 0.002979 **
## Day.CatSun:Time.CatMorn          -1.28668    0.25617  -5.023 5.09e-07 ***
## Day.CatWeekday:Time.CatMorn       0.57318    0.21117   2.714 0.006642 **
## Day.CatSat:Time.CatAfternoon     -0.80471    0.24505  -3.284 0.001024 **
## Day.CatSun:Time.CatAfternoon     -1.22433    0.24335  -5.031 4.88e-07 ***
## Day.CatWeekday:Time.CatAfternoon  0.34012    0.20273   1.678 0.093415 .
## Day.CatSat:Time.CatEvening       -0.66810    0.26801  -2.493 0.012675 *
## Day.CatSun:Time.CatEvening       -1.55050    0.27088  -5.724 1.04e-08 ***
## Day.CatWeekday:Time.CatEvening    0.03632    0.22114   0.164 0.869524
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for Negative Binomial(13.7158) family taken to be 1)
##
##     Null deviance: 880.82  on 167  degrees of freedom
## Residual deviance: 172.14  on 152  degrees of freedom
## AIC: 1439.5
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  13.72
##           Std. Err.:  1.93
##
##  2 x log-likelihood:  -1405.533
```

```
#cbind(mod$coefficients, mod2$coefficients)
```

The model fit compared to the negative binomial fit appears to be overdispersed. In the first model we have a large residual deviance. However, looking at the AIC values the two models aren’t very different and possibly neither of them are a very good fit for the data.

b. Now compare the predicted number of crashes in the early morning on Saturdays using your model from part (a) and the negative binomial model from the Lab. Are they the same? Different?

```
newdat <- data.frame(Day.Cat = "Sat", Time.Cat = "Early.Morn")

## predict using Poisson regression and NB regression
predict(mod, newdata = newdat, type = "response")
```

```
##        1
## 37.83333
```

```
predict(mod2, newdata = newdat, type = "response")
```

```
##        1
## 37.83333
```

The predicted crashes in the early morning on Saturdays using both my model and the negative binomial model are the same.

c. Now examine the standard errors associated with the predictions from part (b). Are they the same? Different? Which model do you prefer? Please explain.

```r
predict(mod, newdata = newdat, type = "response", se.fit = TRUE)
```

```
## $fit
##        1
## 37.83333
##
## $se.fit
## [1] 8.528403
##
## $residual.scale
## [1] 20.89023
```

```r
predict(mod2, newdata = newdat, type = "response", se.fit = TRUE)
```

```
## $fit
##        1
## 37.83333
##
## $se.fit
##        1
## 4.868124
##
## $residual.scale
## [1] 1
```

The standard errors with the two predictions are different. After looking at the summary of both models and the standard errors I prefer the negative binomial model because it seems like a better fit. It has a more reasonable residual deviance/df and we want a smaller standard error which the negative binomial model provides.

Conceptual Question

2. lease find an article or web posting in which the authors discuss the statistical issues with large datasets. Write a short paragraph in which you identify at least one of the issues that the authors raise. Also please comment on whether you agree with or disagree with their assessment. Please submit either a PDF copy of the article you read, or a web link to the article or post.

The article I found had a few points about the statistical issues with large datasets. Firstly, with large sample sizes the data will produce small variances with large sample sizes which leads to large staistics and many, many statisticaly significant results (some which may not be significant at all). Next there is the big issue of bias. Large datasets are things such as websites tracking users or census's which only take in data from users or people who answer a census. This leads to a population of people that may not be representative of what we need from the large data.

To both points I agree. If we are grabbing so much data that everything is significant there is an issue because clearly everything cannot be significant. We need to find subsets so that we are looking at exactly what we want to see. Additionally, I agree that there will likely always be large bias in data because we cannot get everyone even in large datasets and there is certainly the danger of only seeing a bias of the people/data involved/included.

https://journals.sagepub.com/doi/pdf/10.1177/2053951715602495