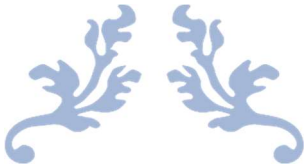


AN ANALYSIS OF BIKE CROSSING COUNTS USING A GENERALIZED POISSON DISTRIBUTION

Analyst: Sameer Patel

JUNE 3, 2020

COMP 4441 – INTRODUCTION TO PROBABILITY AND STATISTICS

University of Denver

Project Plan

Team Members

I have elected to work alone on this project.

Statistical Method

For this project, I analyzed a set of count data using Poisson regression in R. My analysis delves into the necessary assumptions for applying a Poisson regression, as well as addresses alternative methods of statistical analysis in the conclusion should some assumptions be violated (as would be expected when using real-world data).

Data Source and Description

The data I have chosen to analyze is of a daily record of the number of bicycles crossing into or out of Manhattan via the Brooklyn Bridge for a stretch of 9 months. The data ranges from April, 2017 through October, 2017.

The data was obtained from GitHub (https://gist.github.com/sachinsdate/c17931a3f000492c1c42cf78bf4ce9fe) and includes 5 columns of data:

- Date – Character representation of dates within the time period
- High T – (F) Maximum temperature recorded per day
- Low T – (F) Minimum temperature recorded per day
- Precipitation – (in) Presence of rainfall recorded in inches
- Count – Count of bicycles crossing the Brooklyn Bridge

Project Paper

Data Source Definitions and Analysis

The dataset chosen exhibits a random process that most-closely adheres to the guidelines for a Poisson regression analysis. A walkthrough of the Poisson process is summarized below.

Properties

The process is comprised of a sequence of random variables (bike crossings) during some interval of time (per day).

It is a stochastic and discrete process. Stochasticity is ensured due to the fact that a different sequence of random outcomes will occur per the probability distribution we will calculate for this set of data, every time it is run. Discreteness is obviously ensured due to the dataset being comprised of integer counts.

In addition, domain increments of the process are independent of one another, as the number of predicted events in any interval is independent of the number predicted in any other disjoint interval.

The random variables of the Poisson process taken from this dataset all have identical Poisson distributions given by the probability mass function (PMF)

$$P_X(k) = \frac{e^{-\lambda} * \lambda^k}{k!}$$

under the constraint that the sum of the probabilities for all possible values of k is 1.0.

In an ideal situation, we can assume that there is a certain rate of occurrence of events λ (lambda) that drives the generation of the data. In this analysis, we assume a non-constant lambda that is influenced by a vector of explanatory variables (High T, Low T, and Precipitation). We will henceforth refer to these variables as X.

Research Question

For this analysis, we will explore whether precipitation has an effect on the expected number of bike crossings on a given day. We will explore this relationship by applying a generalized Poisson model to our dataset, calculating quantities of interest, plotting the calculated data, and analyzing the results statistically.

Process

The following pages contain the knitted R Markdown file from the analysis performed on this dataset.

Note: I used the Zelig package for much of the analysis. Zelig converts the free-ranging syntax of thousands of statistical methods into a short list of easy functions. More about Zelig can be learned from its website (http://docs.zeligproject.org/index.html).

I installed version 4.2-1 for the purposes of this project.

In addition, I used a Zelig tutorial (https://methods.sagepub.com/dataset/howtoguide/poisson-in-brfss-2013) centered around Poisson models as a general guide for the following analysis.

Project Paper

Data Preparation

We will begin by loading the data and using the summary() function to get a quick glance of it. We will also plot the raw data of bike crossings along the time period (1-Apr-2017 through 31-Oct-2017).

```
dat<-read.csv("nyc_bb_bicyclist_counts.csv",stringsAsFactors = FALSE)
str(dat)
```

```
## 'data.frame':    214 obs. of  5 variables:
##  $ Date    : chr  "4/1/2017" "4/2/2017" "4/3/2017" "4/4/2017" ...
##  $ HIGH_T  : num  46 62.1 63 51.1 63 48.9 48 55.9 66 73.9 ...
##  $ LOW_T   : num  37 41 50 46 46 41 43 39.9 45 55 ...
##  $ PRECIP  : num  0 0 0.03 1.18 0 0.73 0.01 0 0 0 ...
##  $ BB_COUNT: int  606 2021 2470 723 2807 461 1222 1674 2375 3324 ...
```

```
summary(dat)
```

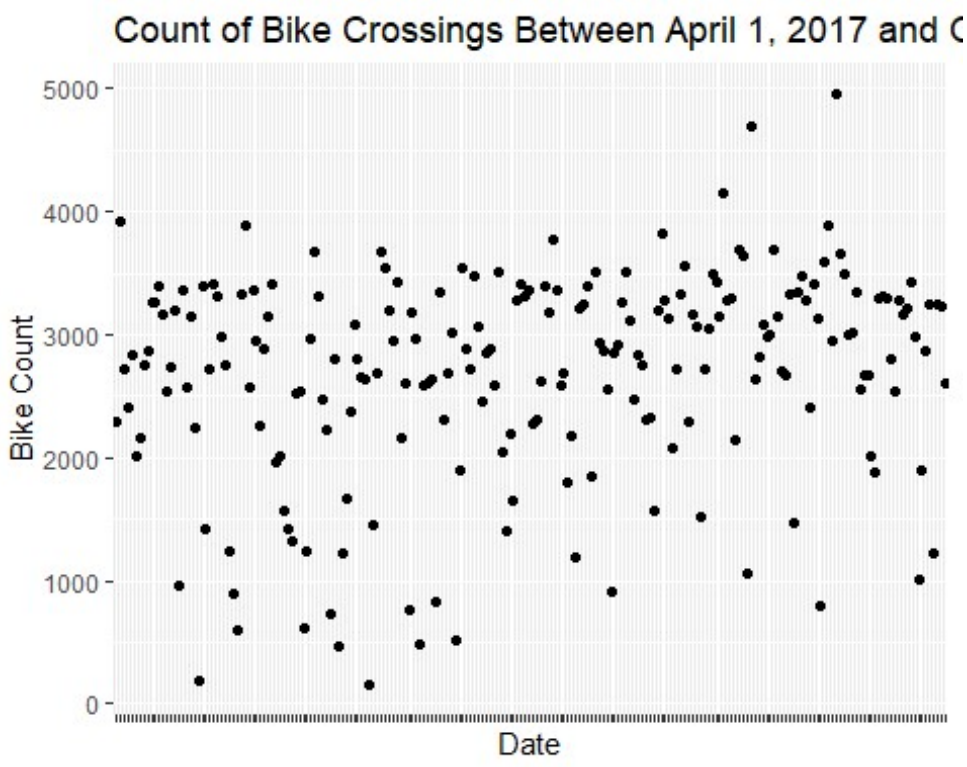
```
##      Date              HIGH_T         LOW_T           PRECIP       
##  Length:214         Min.   :46.0   Min.   :37.00   Min.   :0.0000  
##  Class :character   1st Qu.:66.9   1st Qu.:55.23   1st Qu.:0.0000  
##  Mode  :character   Median :75.9   Median :64.00   Median :0.0000  
##                     Mean   :74.2   Mean   :62.03   Mean   :0.1324  
##                     3rd Qu.:82.0   3rd Qu.:70.00   3rd Qu.:0.0375  
##                     Max.   :93.9   Max.   :78.10   Max.   :3.0300  
##     BB_COUNT    
##  Min.   : 151  
##  1st Qu.:2298  
##  Median :2857  
##  Mean   :2680  
##  3rd Qu.:3285  
##  Max.   :4960
```

```
print(head(dat))
```

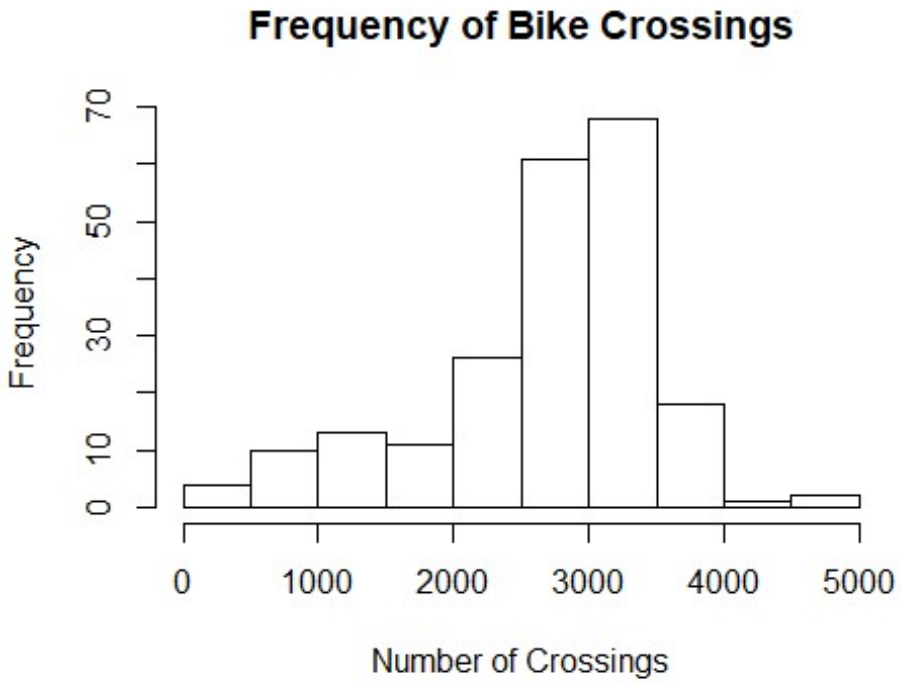
```
##       Date HIGH_T LOW_T PRECIP BB_COUNT
## 1 4/1/2017   46.0    37   0.00      606
## 2 4/2/2017   62.1    41   0.00     2021
## 3 4/3/2017   63.0    50   0.03     2470
## 4 4/4/2017   51.1    46   1.18      723
## 5 4/5/2017   63.0    46   0.00     2807
## 6 4/6/2017   48.9    41   0.73      461
```

```
g<-ggplot(dat,aes(x=Date,y=BB_COUNT)) +
  geom_point() +
  theme(axis.text.x=element_blank()) +
  ggtitle("Count of Bike Crossings Between April 1, 2017 and October 31, 2017
") +
  ylab("Bike Count")
g
```

Project Paper

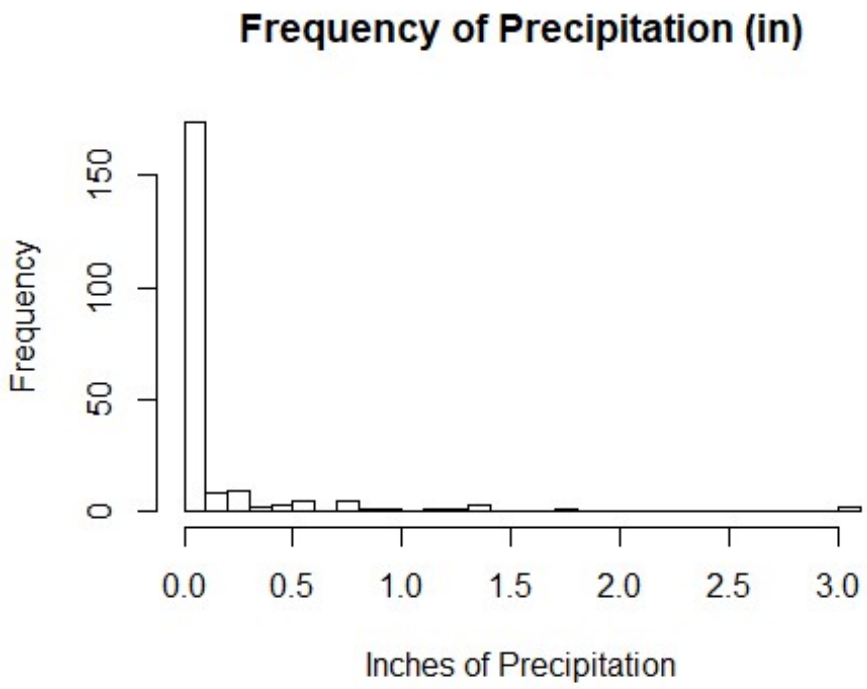



```r
hist(dat$BB_COUNT,main="Frequency of Bike Crossings",xlab="Number of Crossing
s")
```



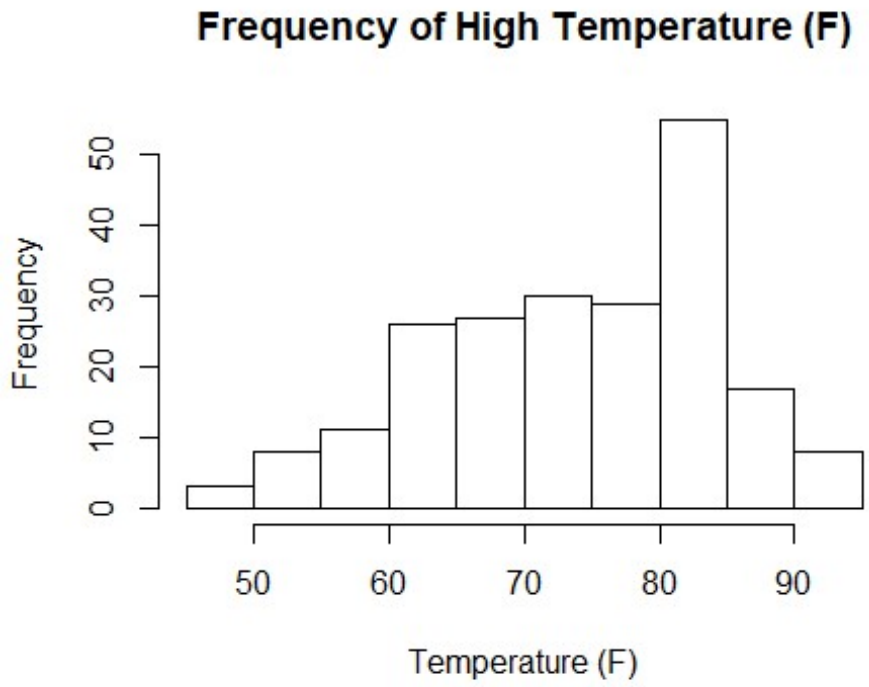
Project Paper

```
hist(dat$PRECIP,main="Frequency of Precipitation (in)",xlab="Inches of Precip
itation",breaks=35)
```

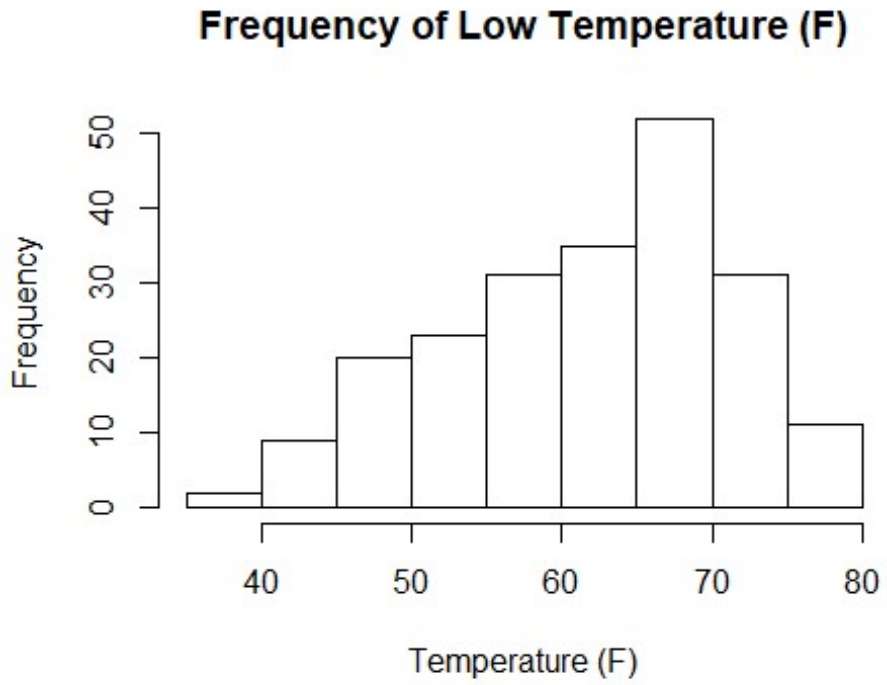



```
hist(dat$HIGH_T,main="Frequency of High Temperature (F)",xlab="Temperature (F
)")
```

Project Paper




```
hist(dat$LOW_T,main="Frequency of Low Temperature (F)",xlab="Temperature (F)"
)
```



Project Paper

Fitting the data to a Poisson Model

We will now incorporate the zelig() function from the Zelig package to fit the counts to the explanatory variables. Note that the specified family="poisson" uses a log link function by default.

```
fit<-zelig(BB_COUNT~PRECIP+HIGH_T+LOW_T,model="poisson",data=dat,cite=FALSE)
summary(fit)
```

```
##
## Call:
## glm(formula = formula, weights = weights, family = poisson(),
##     model = F, data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -37.319   -7.468    0.382    7.530   35.259
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  7.0320668  0.0104034  675.94   <2e-16 ***
## PRECIP      -0.7841734  0.0068052 -115.23   <2e-16 ***
## HIGH_T       0.0232082  0.0002968   78.18   <2e-16 ***
## LOW_T       -0.0129832  0.0003192  -40.67   <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: 70021  on 213  degrees of freedom
## Residual deviance: 30113  on 210  degrees of freedom
## AIC: 32185
##
## Number of Fisher Scoring iterations: 4
```

The Zelig function returns the same results as would be expected from the built-in function glm().

We can see that the Deviance Residuals are approximately normally distributed, but exhibit a little right-skewness since the median is not quite 0.
Looking at the results, specifically the coefficients of the independent variables and their statistical significance, we can see that each coefficent estimate (for PRECIP, HIGH_T, and LOW_T) is statistically significantly different from 0, with PRECIP being the most influential explanatory variable. The coefficient for PRECIP being negative indicates an inverse relationship with bike crossings. The proceeding analysis will focus on precipitation, with the explanatory variables for LOW_T and HIGH_T fixed at their means.

A common way to interpret the results of a Poisson regression model is to compute the expected value (predicted mean) and predicted counts for the dependent variable based on the

results of the analysis. Computing expected values and predicted counts requires setting accessory independent variables to some fixed value. Doing so allows us to evaluate how the independent variable of interest impacts changes in the results after recomputing quantities of interest.

The Zelig setx() function allows us to set independent variables to specific values in order to create profiles of interest. For exploring the effect of precipitation as aforementioned, we will establish two profiles: rain and shine (max and min precipitation, respectively). The other explanatory variables (HIGH_T and LOW_T) will be set equal to their means in both profiles. Though they still exhibit some influence on the predicted data, they are close enough to 0 and insignificant compared to the coefficient for PRECIP.

```
rain<-setx(fit,PRECIP=max(PRECIP),
           HIGH_T=mean(HIGH_T),
           LOW_T=mean(LOW_T))
shine<-setx(fit,PRECIP=min(PRECIP),
           HIGH_T=mean(HIGH_T),
           LOW_T=mean(LOW_T))
```

The setx() function uses the variables identified in the formula generated by zelig() and sets the values of the explanatory variables to the selected values. The output returns a model matrix (with default unconditional prediction) based on the specified values for the explanatory variables.

The setx() function is used in conjunction with the sim() function (within the Zelig package).

The sim() function is used estimate the expected values and predicted counts of bike crossings, along with confidence intervals, for the profiles we defined. These quantities of interest are estimated using post-estimation (a posteriori) simulation. The process computes 1000 sets of expected values and predicted counts by simulating values for the model coefficients based on their estimated values, variances, and covariances.

```
set.seed(2357)
counts.rain <- sim(fit, x=rain)
counts.shine <- sim(fit, x=shine)
summary(counts.rain)

##
## Model:  poisson
## Number of simulations:  1000
##
## Values of X
##   (Intercept) PRECIP   HIGH_T   LOW_T
## 1           1   3.03 74.20187 62.0271
## attr(,"assign")
## [1] 0 1 2 3
##
## Expected Values: E(Y|X)
##     mean    sd     50%    2.5%   97.5%
```

```
##  263.287 5.309 263.067 253.248 273.688
##
## Predicted Values: Y|X
##    mean     sd 50% 2.5% 97.5%
##  263.61 17.236 263  230   297

summary(counts.shine)

##
## Model:  poisson
## Number of simulations:  1000
##
## Values of X
##   (Intercept) PRECIP   HIGH_T   LOW_T
## 1           1      0 74.20187 62.0271
## attr(,"assign")
## [1] 0 1 2 3
##
## Expected Values: E(Y|X)
##      mean    sd      50%     2.5%    97.5%
##  2832.277 4.164 2832.039 2824.385 2840.487
##
## Predicted Values: Y|X
##     mean     sd  50% 2.5%   97.5%
##  2832.38 51.903 2832 2730 2935.05
```

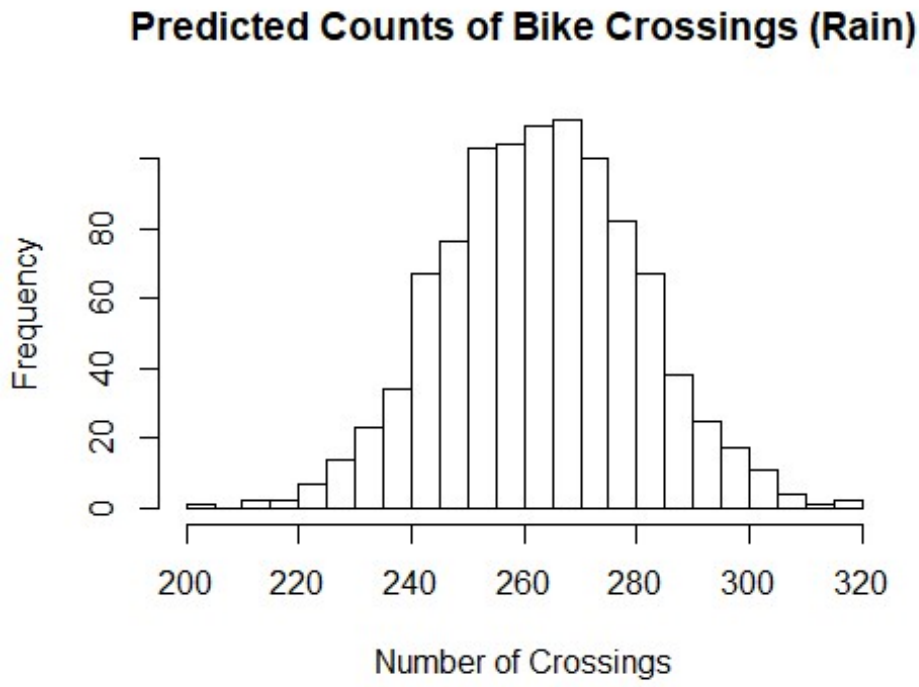
The results display estimated means, standard deviations, medians (50%), and the 2.5 and 97.5 percentiles for the 1000 simulated expected values of bike crossings. Also presented are the same statistics calculated based on the predicted counts of bike crossings. Looking at the mean values for expected values of bike crossings between the rain and shine profiles, we can see that precipitation has a statistically significant effect on the dependent variable. The effect is quite pronounced: likely because I opted to use the maximum and minimum values of PRECIP for rain and shine, respectively. For continued analysis, more logical values, such as 75th quantile and 25th quantile, could be used to better estimate the data.

To visualize the data in a better way, we can create histograms of 1000 simulated counts for each profile. To do this, we use the simulation.matrix() function from the Zelig package to extract the simulated counts from the rain and shine profiles.

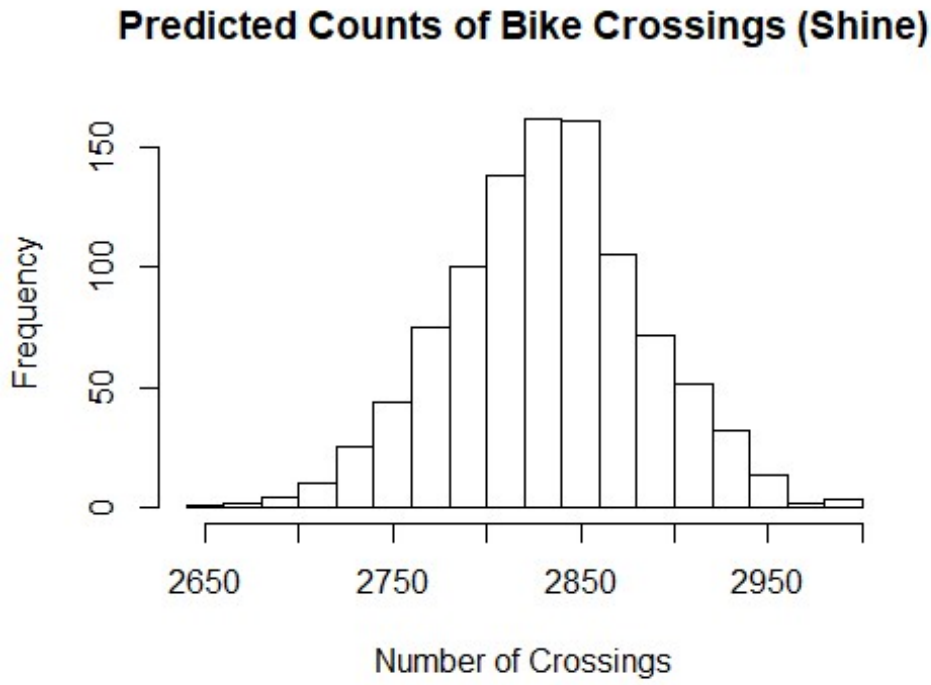
```
sim.counts.rain<-simulation.matrix(obj=counts.rain,which=summary(counts.rain)
$titles[3])[,1]
sim.counts.shine<-simulation.matrix(obj=counts.shine,which=summary(counts.shi
ne)$titles[3])[,1]

hist(sim.counts.rain,main="Predicted Counts of Bike Crossings (Rain)",xlab="N
umber of Crossings",breaks=20)
```

Project Paper




```
hist(sim.counts.shine,main="Predicted Counts of Bike Crossings (Shine)",xlab=
"Number of Crossings",breaks=20)
```



Project Paper

To continue visualizing the impact of precipitation on expected number of bike crossings, we can compute the expected value of bike crossing count repeatedly based on values of precipitation and present the results graphically. Below we show the expected value of the count of bike crossings, along with a 95% confidence interval, as a function of precipitation, whilst keeping HIGH_T and LOW_T at the means specified in the profiles we created.

To generate this data, we can use a for loop along with the aforementioned setx() and sim() functions for data points within the range of observed PRECIP data. For help in data visualization, the "zoom" package was added to the library, which allows panning/zooming on plotted figures in R.

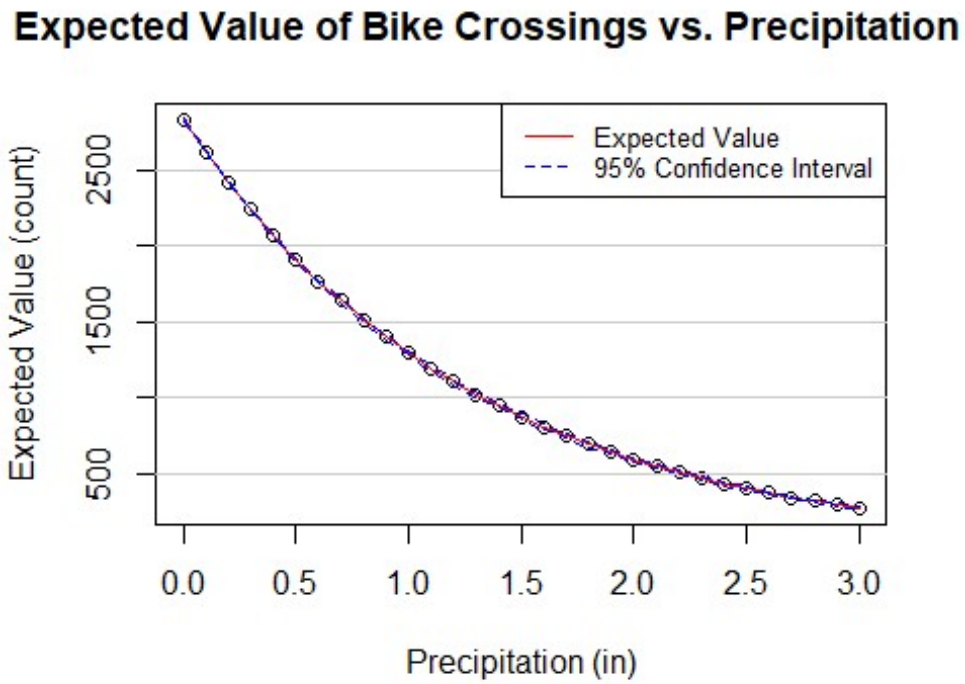
```
set.seed(2357)
prec_pts<-seq(min(dat$PRECIP),max(dat$PRECIP),by=0.1)
results<-matrix(data=NA,length(prec_pts),4)

for(i in 1:length(prec_pts)){
  prec<-setx(fit,PRECIP=prec_pts[i],
             HIGH_T=mean(HIGH_T),
             LOW_T=mean(LOW_T))
  precvals<-sim(fit,x=prec)

  results[i,1]<-precvals$stats$'Expected Values: E(Y|X)'[,3]
  results[i,2]<-precvals$stats$'Expected Values: E(Y|X)'[,4]
  results[i,3]<-precvals$stats$'Expected Values: E(Y|X)'[,5]
  results[i,4]<-prec_pts[i]
}

plot(results[,4],
     results[,1],
     xlab="Precipitation (in)",
     ylab="Expected Value (count)",
     main="Expected Value of Bike Crossings vs. Precipitation (in)"
)
abline(h=seq(500,2500,500),col="gray80")
lines(results[,4],results[,1],col="red",lwd=1)
lines(results[,4],results[,2],col="blue",lwd=1,lty=2)
lines(results[,4],results[,3],col="blue",lwd=1,lty=2)
legend("topright",
       legend=c("Expected Value","95% Confidence Interval"),
       col=c("red","blue"),
       lty=1:2,
       cex=0.8
)
```

Project Paper



From the graph, we can see a relatively tight window representing the 95% confidence interval for each of the values of precipitation from our simulated profile. This indicates a good fit of the data to the Generalized Poisson Model employed in our data analysis.

Project Paper

Conclusion

From the preceding analysis, we can establish that the Generalized Poisson Model was a good choice for fitting the data. The results indicate an inverse relationship between precipitation and observed count of bicycle crossings. From the graph, the 95% confidence interval over the range of precipitation values from 0.0 to 3.0 inches indicates an excellent goodness of fit of the data to the model.

The means of the simulated profiles for both “rain” and “shine” are ~260 and ~2800, respectively. While the results do confirm real-world expectations – rainy weather leads to less people traveling by bike – the range of values is still quite significant. An explanation for this is my election to set the precipitation values for the rain and shine profiles to the maximum and the minimum of the PRECIP column from the observed data, respectively. A more robust election would have been to normalize the values in each profile to be closer to the average; perhaps 75^{th} and 25^{th} quantiles, respectively.

The standard deviations calculated for both “rain” and “shine” were ~4 and ~17 respectively, which when compared to their means (~260 and ~2800, respectively), indicates a strong central tendency of the simulated data based.

By applying a generalized Poisson model to this data, we were able to establish a clear relationship between the dependent variable, count of bicycle crossings on the Brooklyn Bridge, and the independent variable, precipitation.

Further Analysis

Possible routes of further analysis would be to explore the effect of LOW_T and HIGH_T independent variables on the expected value of bicycle crossings. The reason this was not conducted for this dataset is because of wide range of overlap of the values between both variables. This is due to the fact that this dataset contains only 7 months of data coinciding roughly with the summer season in New York. In order to full examine the effect of temperature on the dependent variable, a larger dataset would be required.

Project Paper

Works Cited

Author, Unnamed. “How-to Guide for R Software.” *Learn About Poisson Regression in R With Data From the Behavioral Risk Factor Surveillance System (2013)*, 2016, methods.sagepub.com/dataset/howtoguide/poisson-in-brfss-2013#i135.

Crawley, Michael J. “Count Data.” *Statistics: An Introduction Using R*, 2nd ed., John Wiley & Sons, Ltd, 2015, pp. 234–255.

Date, Sachin. “An Illustrated Guide to the Poisson Regression Model.” *Medium*, Towards Data Science, 8 May 2020, towardsdatascience.com/an-illustrated-guide-to-the-poisson-regression-model-50cccba15958.

Date, Sachin. “The Poisson Process: Everything You Need to Know.” *Medium*, Towards Data Science, 27 Oct. 2019, towardsdatascience.com/the-poisson-process-everything-you-need-to-know-322aa0ab9e9a.

UCLA. “POISSON REGRESSION | R DATA ANALYSIS EXAMPLES.” *IDRE Stats*, stats.idre.ucla.edu/r/dae/poisson-regression/.