Spring 5-2019

Traffic Fatalities in Illinois

Cobi Illian

Traffic Fatalities in Illinois

Cobi Illian

Introduction

Abstract

The goal of this project was to gain insight into the time series structure of fatal car accidents in Illinois. This study is of interest as knowing the structure of fatal accidents can help prevent them or allow responders to react to them earlier. Two large datasets were available for use: FARS (Fatality Analysis Reporting) and HSIS (Highway Safety Information System). The final model describes the number of fatal accidents by incorporating spatial information through the county and temporal information through the time when the accident occurred.

Background

The idea to investigate traffic fatalities in Illinois is inspired by a challenge from Summer 2018 by the U.S. Department of Transportation [1]. The goal of the challenge was to develop analytical visuals to help them make insights based on data from the Federal Railroad Administration, the Federal Highway Administration, and the National Highway Traffic Safety Administration. Although this project is not part of the challenge, the data and goals are similar. The goal of this project is to expand upon basic time series analysis from classes by fitting more complex models to a large dataset. This differs from the DOT challenge where the goal was merely data visualization.

Data

This project investigates two datasets. The first is FARS (Fatality Analysis Reporting System) dataset [2] which has case-by-case information on fatal accidents in the U.S. The FARS dataset contains information regarding when the fatal accident occured, the number of fatalities, and the location of the accident. The second dataset was the HSIS (Highway Safety Information System) data[3], which contains information about accidents in general (not necessarily fatal), with variables describing the location, the date, and the condition of the road. This analysis is limited to Illinois since it was one of the states for which has HSIS data available, and my home state.

There are many different challenges when working with these datasets. First, they both have case-by-case data which is not equally spaced over time and finer detail than of interest here. In addition, both datasets are collected and maintained by separate agencies, so the variable names and definitions must be aligned in order to merge for analysis. Each of these data sets are collected and stored annually, so the record keeping from year to year differs. This adds another hurdle to merging the datasets even if they are collected by the same agency.

Plan for Analysis

The time series structure of this data will be analyzed with and ARIMA model. The nature of this model and its assumptions will be discussed in the ARIMA models section.

Discussion of Previous Work

The first article that examined was an analysis of road traffic injuries in Valledupar, Colombia [4]. Their data appears to have similar variables to the data I will be using, however their data appeared to have more information about the people involved in the accidents. They analyzed the data by fitting an ARIMA model. Their ARIMA model was a (5,1,2)X(1,1,0,12) as described in their paper [4].

ARIMA models are not the only way to analyze this type of data. The second paper investigated the idea of detecting unsafe roadways using information about crashes (both fatal and non-fatal) in the state of New Hampshire. They used a Poisson process to model the data, and were able to use this to perform statistical estimates of crash rates for each area of the state [5]. They made use of density graphics with both fatal and non-fatal accidents to display the data. By combining their graphics and model, they wanted to be able to evaluate crash risk in small areas.

The third paper discussed detection of "spatial-temporal dependencies of crash occurances" [6]. It had data containing the total number of crashes per day over a four year period for Mashhad, Iran. Plots were made with the density of crashes over two-hundred and fifty zones which the city was divided into. The methods used to analyze the data were Moran's I and Lisa which were used to detect spatial-temporal autocorrelation and determine if the pattern of crashes was non-random. The results of the study found that the pattern of the crashes were non-random with spatial-temporal autocorrelation present.

This last paper had similar goals to the previous paper as it aimed to identify crash patterns through the use of a discrete response model [7]. The goal of the model was to forecast the likelihood of accidents based on time and location. The model used weather, traffic flow, and geometric characteristics as the variables. The geometric data was gathered from aerial photos and contained information such as the number of on and off ramps, and the degree and length of horizontal curves. The results of this study found traffic flow, weather conditions, and geometric characteristics all to be signifcant in forecasting the likelihood of accidents.

Using the previous results, we also will account for the spatial temporal nature of the accident rates in Illinois. In addition, similar to the first paper, we will to fit ARIMA models by county which will likely differ over the different counties in Illinois due to varying populations. We will also accompany the results with a visual to explain why certain counties were found to have a time series structure, and why others were not.

ARIMA Models

Autoregressive integrated moving average (ARIMA) models are those that have terms for the autoregressive and moving average nature of non-stationary time series.

Autoregressive (AR) models try to determine the relationship between the current observation and previous observations, while moving average models (MA) try to determine the relationship between the current observation the the previous error. The differencing term is used to deal the with non-stationary nature of these time series In the case of the models seen in this paper, ARIMA(5,1,0), there are 5 AR terms and no ARIMA terms, with one differencing term. The structure of this model is:

$$X_t = c + \phi_{t-1} * X_{t-1} + \phi_{t-2} * X_{t-2} + \phi_{t-3} * X_{t-3} + \phi_{t-4} * X_{t-4} + \phi_{t-5} * X_{t-5} + \lambda * X_{t-1} + \epsilon_t$$

Where X_t is the number of fatal accidents at time t and the time steps are by hour.

This model suggests that to address the non-stationary nature of the data one differencing term, represented by $\lambda = 1$ is used. While the current observation is related to the previous 5 hours as seen with the ϕ terms above.

Data Cleaning

Merging the Datasets

The process to clean the data was definitely the most challenging aspect of this work. The raw data consists of separate case-by-case datasets for each year from 2006 to 2016. The main challenge of merging these was renaming the variables each year, as they sometimes changed. We updated all the names to match allowing them to be merged. Below is an example of the code to do this.

```
data_6 <- rename(data_6, state = istatenum, date= saccdate,person = ipnumber,
city=icity, county= icounty, day_of_month=iaccday,
       hour=iacchr, month=iaccmon,year=iaccyr, day_of_week=dayofweek,
person_type=iptype, body=ibody, weight=igvwrating)
```

After renaming the variables, all the datasets were vertically merged using the code below.

```
#merging the data from each year
full_data <- bind_rows(data_list)
```

Below is the top of the merged dataset. Note: the variables new_data and county_name were added in later. The merged file contains 27330 observations.

```
##   state person city county    date day_of_month hour month year
## 2    17      1 2610    163 1012006            1    4     1 2006
## 3    17      1 2610    163 1012006            1    4     1 2006
## 4    17      1    0    111 1012006            1   21     1 2006
## 5    17      1 1670     31 1032006            3   12     1 2006
## 6    17      1 1670     31 1032006            3   12     1 2006
## 7    17      1 9997    163 1042006            4    0     1 2006
##   day_of_week numfatal person_type   acc_date county_name
## 2           1        1           1 2006-01-01   St. Clair
## 3           1        1           1 2006-01-01   St. Clair
## 4           1        1           1 2006-01-01     McHenry
## 5           3        1           1 2006-01-03        Cook
```

```
## 6           3        1           1 2006-01-03        Cook
## 7           4        1           1 2006-01-04   St. Clair
```

Fixing the Dates

The next challenge was to setup the dates for each observation. The following code performs this task.

```
#fixing date
full_data$date <- as.character(full_data$date) #turn date into character

#add zeros to the front on dates with 7 characters then make them dates and
make everything a date
for(k in 1:length(full_data$date)){
  if(str_length(full_data$date[k])==7){
    full_data$new_date[k] <- as.character(as.Date(paste0("0",
full_data$date[k]), format = "%m%d%Y"))
  }else{
    full_data$new_date[k] <- as.character(as.Date(full_data$date[k], format =
"%m%d%Y"))
  }
}
```

This was done to solve the problem of dates not containing a leading 0 for months without a second digit. For example Jan 1st, 2016 was written as 1012006. A leading 0 was added to each date that was like this, so they could be converted to a date format.

Limiting the People

This data provides information regarding the type of people an accident using the number codes below:

1: Driver of motor vehicle in transport

2: Passenger of motor vehicle in transport

3: Occupant of a motor vehicle not in transport

4: Occupant of a non-motor vehicle transport device

5: Pedestrian

6: Bicyclist

7: Other Cyclist

8: Other persons on personal Conveyances/ in buildings

9: Unknown occupant type

10: Persons in/on buildings

19: Unknown type of non-motorist

From above we can see accidents included people of many different types, however not all of these types of people were represented in the datasets. After making some tables displaying the count of people types by year from 2010 to 2016, only people of type 1,2, or 9 were included in the datasets. The dataset observations were limited to people of only 1 and 2, since these were the only ones included throughout each year. People of type 9 were also excluded since they made up only a few cases and they represent unknown occupants therefore these observations may not be reliable. Below is the code used to keep only type 1 and 2 people.

```
#limiting to type 1 and 2 people
full_data <- full_data[which(full_data$person_type == 1 |
full_data$person_type == 2 ), ]
save(full_data, file="full_data.RData")
```

Finally county names were added instead of in place of identification numbers. This was important since counties are much easier to identify by name. To do this, a dataset with county codes provided by Illinios was joined using the method below.

```
#merging in county names based on codes
full_data <- left_join(full_data, county_data, by="county")
```

Each county was matched to its number while keeping all the observations in the full dataset.

Merging FARS and HSIS Datasets

The next stage was to merge the two large datasets that had been cleaned. All the variables to be kept were setup, then both datasets were aggregated before the merge as shown below.

```
#aggregating HSIS
x <- (aggregate(.~acc_date + hour + county_name, data = hsis_full, FUN =
sum))
#aggregating FARS
fars_agg <- aggregate(.~acc_date + hour + county_name, data =
fars_2006_2010.df, FUN = sum)
```

This created an hourly temporal resolution of crashes within each county. After this, the following code merged the two datasets, while sorting the datasets by date.

```
fars_hsis.df <- left_join(x, fars_agg)
#sort the data
fars_hsis.df <- fars_hsis.df[order(acc_date, hour),]
```

Below is the top of the final dataset used the the subsequent analysis.

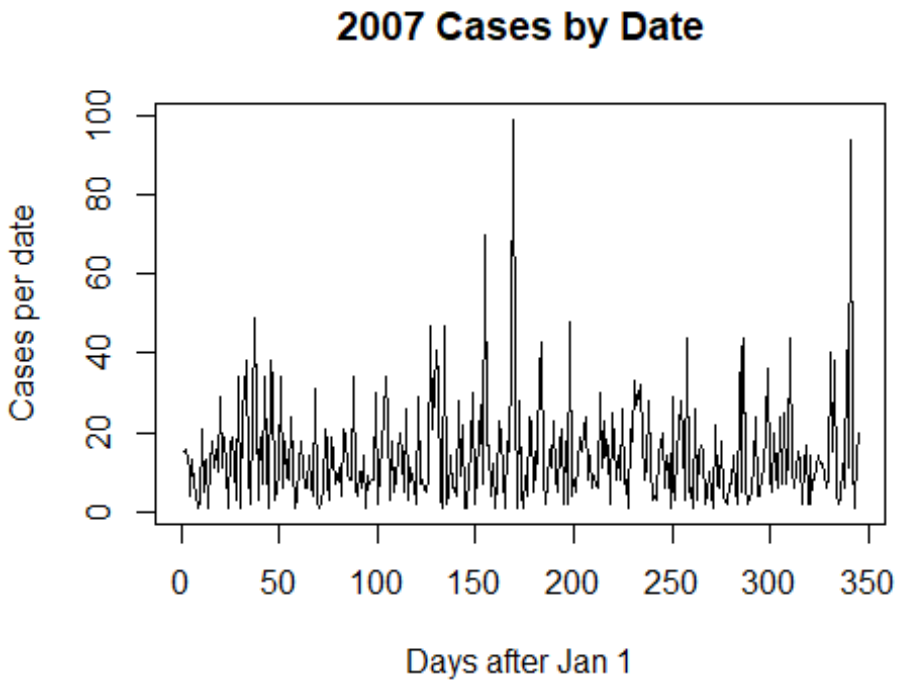
```
head(fars_hsis.df)

##          acc_date hour county_name rd_def acc_count numfatal
## 27743  2006-01-01    0        Cook    100         2        0
## 78160  2006-01-01    0      Dupage      1         1        0
```

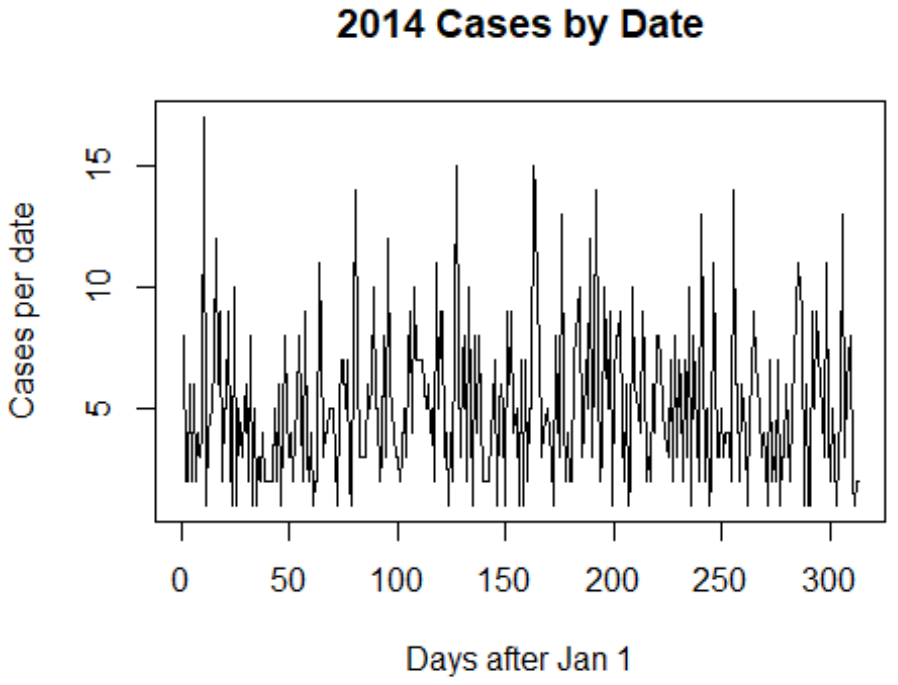
```
## 198436 2006-01-01    0         Lee     99         1        0
## 202767 2006-01-01    0       Logan      1         1        0
## 226141 2006-01-01    0      Marion      1         1        0
## 243857 2006-01-01    0      Mclean      1         1        0
##        fatal_acc_count adj_hour
## 27743                0       24
## 78160                0       24
## 198436               0       24
## 202767               0       24
## 226141               0       24
## 243857               0       24
```

Subsets of this dataset by county will be used to fit ARIMA models using the number of fatalities (numfatal), and the hour as a regressor.

Visuals

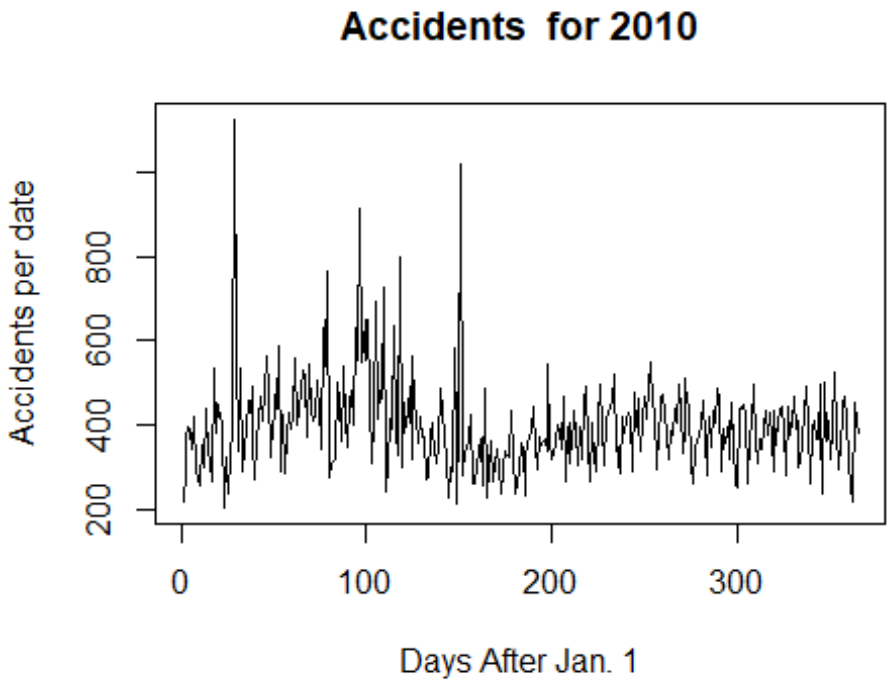
The first visual to be discussed is a time series plot of the cases per data for 2007 compared to the cases per date of 2014.

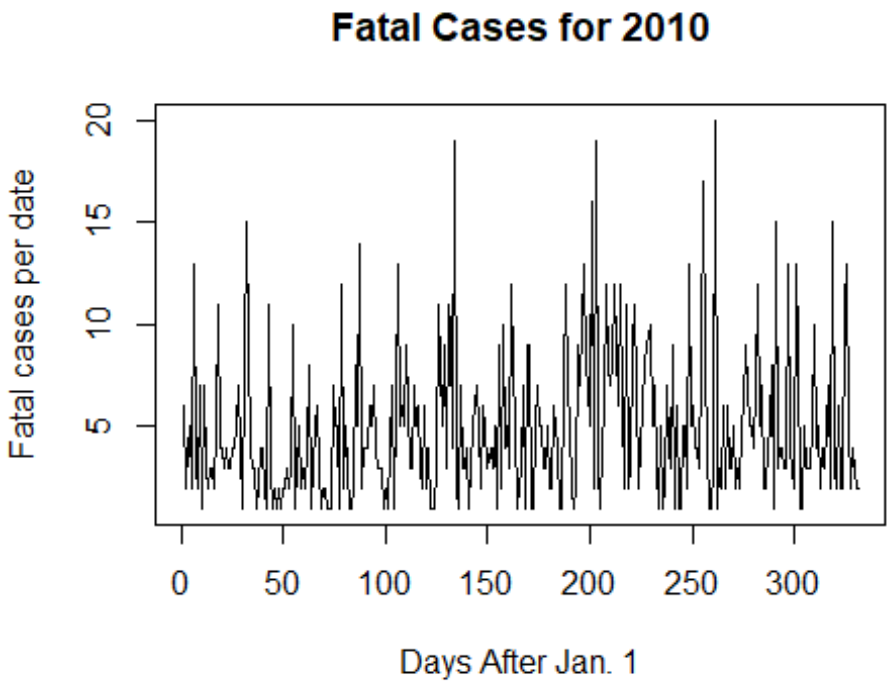




Notice that the max number of cases for 2007 is significantly higher than the max number of cases for 2014. This is the case for 2006-2009 vs. 2010-2016, with the older datasets containing significantly more cases. We are unsure of why such an artifact occurs, but it was interesting to note. Beyond that, both time series appear to be stationary with constant variance. These two plots are similar to the plots from their groups, i.e. 2007 is similar to the plot from 2006,2008,2009, and 2014 is similar to the others. The number of fatalities also seems to follow the trend of being higher in the earlier years.

The next thing of interest was comparing the number of accidents (HSIS data), to the number of fatal accidents (FARS data).





From the plots above we can see that there are difference between fatal accidents and accidents in general. The first thing to note is the vast difference in the scale, since obviously there are more accidents than fatal accidents. Beyond this though, the time series for just accidents does not appear to be stationary (and this appears the same for the other

years), as there appears to be some type of changing mean over time at some wave-like frequency. Based on the plot, the variance may also not be constant, as it appears to decrease as time goes on.

Tables

We can also compare the types of people involved in fatal accidents across the years. As discussed before, type 1 people are the drivers, and type 2 are the passengers. Proportions of type 1 and 2 people for 2008 and 2015 can be seen below.

```
## [1] "Table for 2008"

##
##     1     2
## 0.664 0.336

## [1] "Table for 2015"

##
##     1     2
## 0.782 0.218
```

From the tables we can see a slight difference in the proportion of fatalities when comparing the earlier years to more recent years. Furthermore, we can see the driver appears to be the most common type of fatality in fatal accidents in both years.

Fitting a Model

Now that the data was cleaned sufficiently and merged, the next step was to begin fitting ARIMA models to the data. ARIMA models require observations to be equally spaced in time, and each county did not have an observation for each hour. So, given the desire to fit a model with location, all the missing hours by county were populated with empty observations to solve this problem. The code below was used to do this:

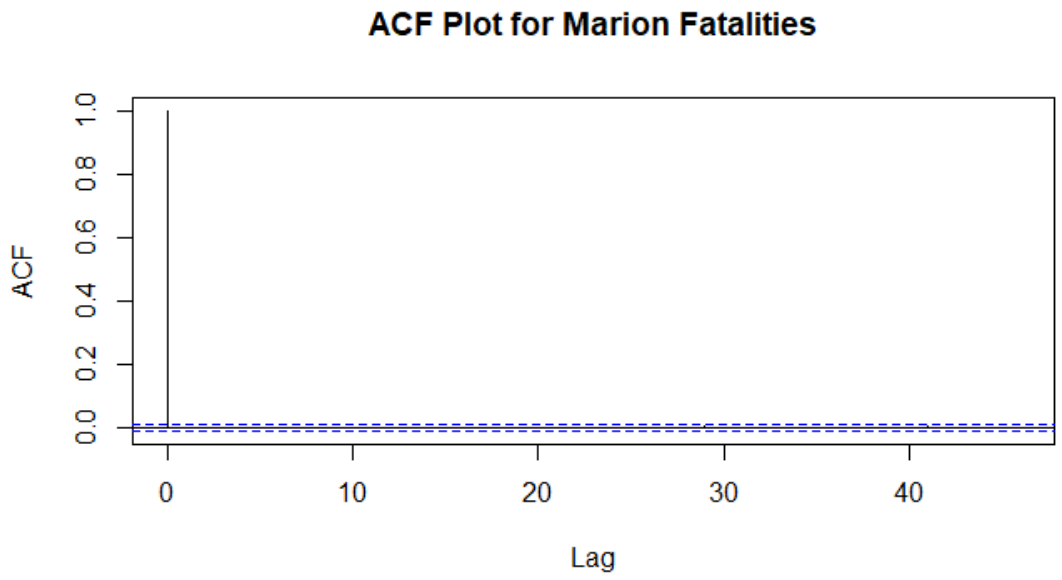
```
dates <- unique(fars_hsis.df$acc_date)
counties <- unique(fars_hsis.df$county_name)
hours <- unique(fars_hsis.df$hour)

for( i in 1:length(counties)) {
  for( j in 1:length(dates) ) {
    for( k in 1:length(hours) ) {
      if(dim(fars_hsis.df[fars_hsis.df$county_name==counties[i] &
                          fars_hsis.df$acc_date==dates[j] &
                          fars_hsis.df$hour==hours[k], ])[1]==0){
        fars_hsis.df <- rbind(fars_hsis.df, data.frame(acc_date=dates[j],
hour=hours[k],

county_name=counties[i], rd_def=99,acc_count=0,
                                                       numfatal=0,
fatal_acc_count=0, adj_hour=0) )
```

```
      }
    }
  }
  print(counties[i]); flush.console();

}
```

This code was one of the main challenges of this part of the project due to how long it took to run and the size of the dataset. This process leads to each county having 43824 observations, and since there are 102 unique counties in the dataset, this has greatly increased the size of the initial dataset (up to 4 million observations).

The auto.arima() function in R was used to choose the model that best fits the data. An example for Marion County(located in Southern Illinois) is below.

```
#example of model fit and output
Mariondata <- subset(fars_hsis1.df, fars_hsis1.df$county_name == "MARION")
Mariondata <- Mariondata[order(Mariondata$acc_date, Mariondata$hour),]
auto.arima(Mariondata$numfatal, xreg = c(Mariondata$hour), allowmean = TRUE,
approximation = FALSE)

## Series: Mariondata$numfatal
## Regression with ARIMA(0,0,0) errors
##
## Coefficients:
##         xreg
##       3e-04
## s.e.  1e-04
##
## sigma^2 estimated as 0.04015:  log likelihood=8267.09
## AIC=-16530.18   AICc=-16530.18   BIC=-16512.8

acf(Mariondata$numfatal, main ="ACF Plot for Marion Fatalities")
```



Above we can see an example of a fit model and the ACF and PACF plots for the model. The 'best' fit model was no ARIMA model with a regressor of hour. This model was fit to the Illinois county of Marion, and this process was used for several other counties. The ACF plot above suggest there is no dependence of the number of accidents on previous hours.

One of the challenges with this part of the process was that the auto ARIMA function did not allow for categorical variables as regressors. So I had to separate the data up by county and fit models specific to each county as a way to build a spatial aspect to my models. So far, 4 models have been fit to randomly selected counties with hour as a regressor, and all have fit an ARIMA(5,1,0) model. The use of this model will continue to be tested and investigated as the dataset is still populating missing hours by county.

After populating the data for twenty counties, the above technique was used to fit models. Only two were found to fit an ARIMA(5,1,0) model and the regression term in all the model was insignificant. The counties where more accidents were present, likely due to population, were fit with an ARIMA(5,1,0) model. Smaller counties with lower populations and less accidents were best modelled with a time-independent model like Marion. An example of a more populated county, Cook which contains the large city of Chicago, is shown below.

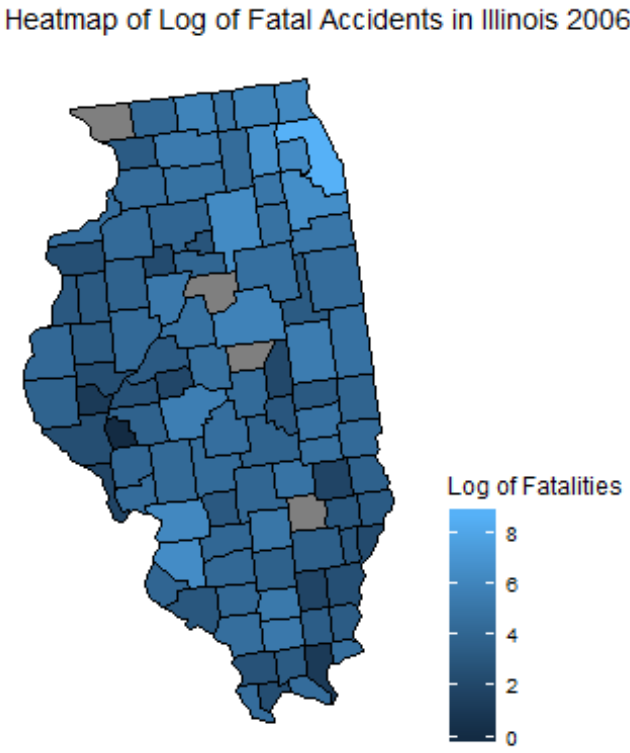
```
Cookdata <- subset(fars_hsis1.df, fars_hsis1.df$county_name == "COOK")
Cookdata <- Cookdata[order(Cookdata$acc_date, Cookdata$hour), ]
auto.arima(Cookdata$numfatal, xreg = c(Cookdata$hour), allowmean = TRUE,
approximation = FALSE)

## Series: Cookdata$numfatal
## Regression with ARIMA(5,1,0) errors
##
## Coefficients:
##           ar1      ar2      ar3      ar4      ar5     xreg
```

```
##       -0.8250  -0.6533  -0.4915  -0.3297  -0.1671  -0.0010
## s.e.   0.0047   0.0059   0.0063   0.0059   0.0047   0.0012
##
## sigma^2 estimated as 2.766:  log likelihood=-84473.3
## AIC=168960.6   AICc=168960.6   BIC=169021.4
```

Above we can see the ARIMA(5,1,0) structure, and again the regressor for hour does not appear to be significant. The ACF plot also suggest that there is a

Conclusions and Future Work

The results are not surprising given how rare fatal accidents are. Based on the heatmap below, we can see that many of the counties do not have a large amount of fatalities due to accidents. This agrees with the results of the models since many counties simply lack the frequency of fatal accidents for temporal dependence to be found.



This does however lead to a future interest relating to the number of accidents. It would be of interest to investigate if there are different spatial-temporal ARIMA models fit using the data with accidents instead of fatalities. This interest stems from the higher frequency of accidents that are not fatal and may lead to a wider variety of models and relationships to investigate.

References and Acknowledgements

Advised by Dr. Heyman

[1] “Challenge Resources”, US Department of Transportation, 2018. [Online]. Available: https://www.transportation.gov/solve4safety/resources. [Accessed: 14- Nov- 2018]

[2]“Fatality Analysis Reporting System (FARS)”, NHTSA, 2018. [Online]. Available: https://www.nhtsa.gov/research-data/fatality-analysis-reporting-system-fars. [Accessed: 14- Nov- 2018]

[3]“HSIS - Highway Safety Information System”, Hsisinfo.org, 2018. [Online]. Available: https://www.hsisinfo.org/. [Accessed: 14- Nov- 2018]

[4]J. Rodríguez, R. Peñaloza and J. Moreno Montoya, “Road Traffic Injury Trends in the City of Valledupar, Colombia. A Time Series Study from 2008 to 2012”, PLOS ONE, vol. 10, no. 12, p. e0144002, 2015 [Online]. Available: http://libproxy.rose-hulman.edu:2048/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=aph&AN=111500313&site=ehost-live&scope=site

[5]P. Ossenbruggen, E. Linder and B. Nguyen, “Detecting Unsafe Roadways with Spatial Statistics: Point Patterns and Geostatistical Models”, Journal of Transportation Engineering, vol. 136, no. 5, pp. 457-464, 2010 [Online]. Available: http://libproxy.rose-hulman.edu:2048/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=aph&AN=49193696&site=ehost-live&scope=site

[6]A. Matkan, A. Mohaymany, M. Shahri and B. Mirbagheri, “Detecting the spatial-temporal autocorrelation among crash frequencies in urban areas”, Canadian Journal of Civil Engineering, vol. 40, no. 3, pp. 195-203, 2013 [Online]. Available: http://libproxy.rose-hulman.edu:2048/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=aph&AN=86726919&site=ehost-live&scope=site

[7]Y. Qi, B. Smith and J. Guo, “Freeway Accident Likelihood Prediction Using a Panel Data Analysis Approach”, Journal of Transportation Engineering, vol. 133, no. 3, pp. 149-156, 2007 [Online]. Available: http://libproxy.rose-hulman.edu:2119/ehost/detail/detail?vid=0&sid=970c104c-a834-41c2-81c8-5974074d9aa1%40sessionmgr104&bdata=JnNpdGU9ZWhvc3QtbGl2ZSZzY29wZT1zaXRl#AN=24064819&db=aph

R Packages

Hadley Wickham (2017). tidyverse: Easily Install and Load the ‘Tidyverse’. R package version 1.2.1. https://CRAN.R-project.org/package=tidyverse

Hadley Wickham (2018). stringr: Simple, Consistent Wrappers for Common String Operations. R package version 1.3.1. https://CRAN.R-project.org/package=stringr

H. Wickham. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York, 2016.

Hyndman R, Athanasopoulos G, Bergmeir C, Caceres G, Chhay L, O'Hara-Wild M, Petropoulos F, Razbash S, Wang E, Yasmeen F (2018). *forecast: Forecasting functions for time series and linear models*. R package version 8.4, <URL: http://pkg.robjhyndman.com/forecast>.

Paolo Di Lorenzo (2018). usmap: US Maps Including Alaska and Hawaii. R package version 0.4.0. https://CRAN.R-project.org/package=usmap