Don't Break a Leg! Road Safety in the City of Toronto

STA2453 - Project II Final Report

Sunwoo (Angela) Kang, Sergio E. Betancourt, Jing Li, and Jiahui (Eddy) Du

2019-03-04

Contents

1 Introduction

Road traffic safety is a crucial component of urban planning and development. Nowadays governments (and sometimes the private sector) dedicate significant resources to providing ample and sufficient infrastructure to accommodate diverse modes of transportation, thereby increasing the productivity of any given urban area. In this project we examine road safety in the City of Toronto from 2007 to 2017 and explore the areas with highest risk of a traffic incident, controlling for different factors.

2 Methods

We define the City of Toronto as per the these guidelines (https://www.toronto.ca/city-government/data-research-maps/neighbourhoods-communities/neighbourhood-profiles/). Below are the neighborhood limits and the official 2016 population census estimates:

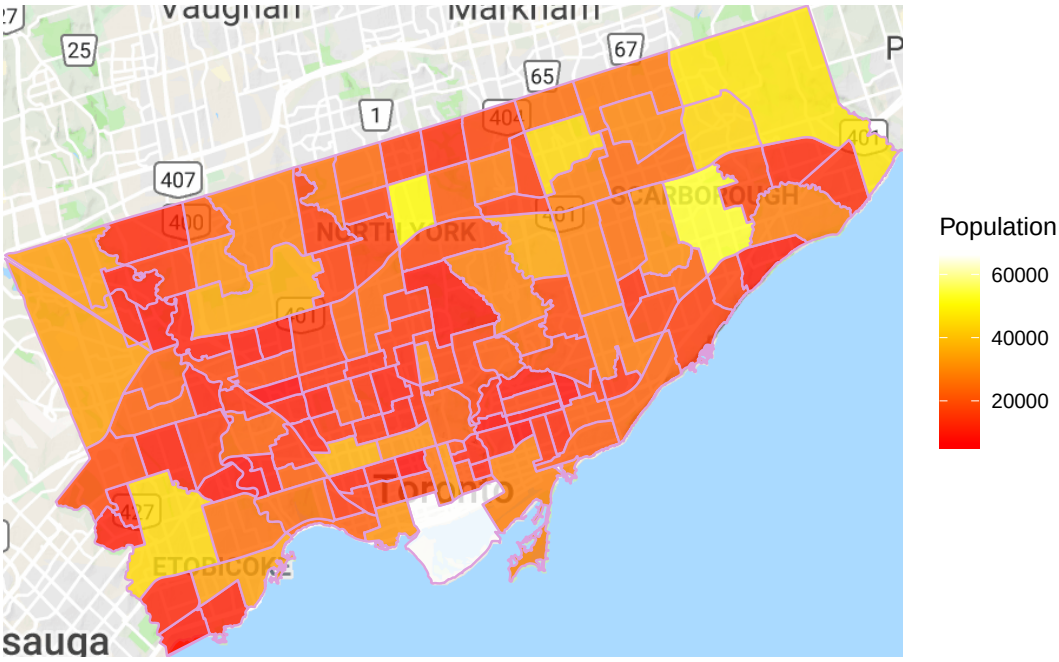


Figure 1: 'Population by neighborhood of the City of Toronto in the census year 2016'

2.1 Primary Questions

The analysis focuses on answering two main questions:

1. Given a collision occurred which areas in Toronto are the most deadly, controlling for other factors?
2. Which factors are related to the collision safety of neighbourhoods?

2.2 Data Collection

For our analysis we employed data from the Toronto Police Service, the City of Toronto, and Environment Canada. Each of these datasets contains different levels of granularity and information, and were therefore combined to obtain the following variables of interest outlined in **Appendix: Dataset Variables and Definitions**.

2.3 Data Preparation

The following table provides an overview of the merged data:

Accident.Key	Fatal	Date	Neighborhood	Population	Max.Temp
5002235651	1	2015-12-30	Greenwood-Coxwell	7072	4.7
5000995174	1	2015-06-13	Annex	26703	22.3
5000995174	1	2015-06-13	Annex	26703	22.3
1249781	0	2011-08-04	Bay Street Corridor	19348	26.4

Traffic incident information provided by Toronto Police served as a base for the data used for this analysis. There are 3,902 unique accidents in this dataset. Each of the 11,360 entries represent a party involved in a traffic collision event. Other features such as the location of the collision (intersection, neighborhood, ward), road condition (visibility, road precipitation), driver action (e.g. speeding, involved alcohol), and types of vehicles (e.g. automobile, pedestrian, cyclist) involved were also used.

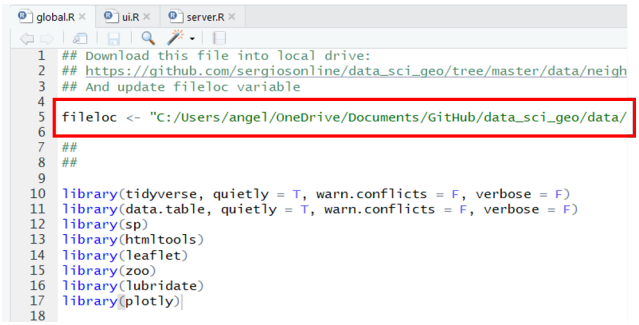
Population counts for 2011 and 2016 are available through the national census for each neighborhood. The populations for the dates not provided by the census were extrapolated using a linear growth model.

Historical weather data collected from the station in University of Toronto was also merged based on the day the accident occurred.

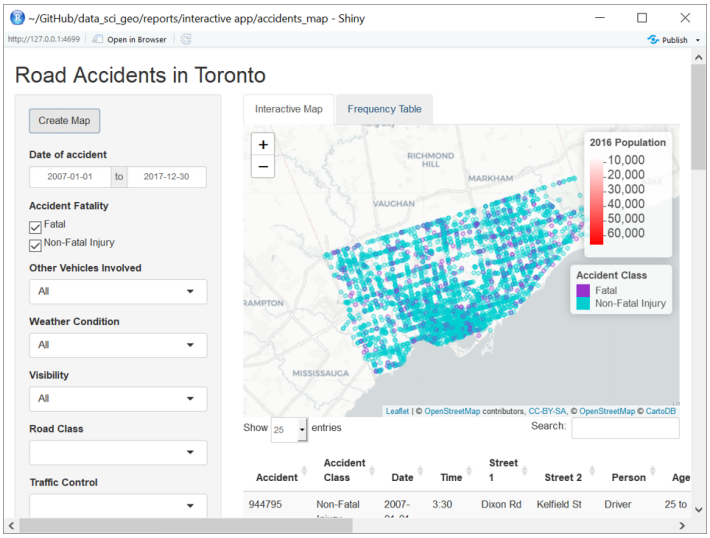
2.4 Exploratory Analysis

Since the data is spatial in nature, it was of foremost interest to be able to plot the accidents on a map. In order to interact with the visualization and filter data by features of interest a Shiny application was created. The following are instructions on downloading and using the application. Because the function `rgdal::readOGR` requires a file path, calibrations must be made on your device to use the application.

1. Download `global.R`, `ui.R`, and `server.R` from here
2. Download the contents of the file here and save to a local directory.
3. Update the variable `file_loc` in the `global.R` script as seen below.



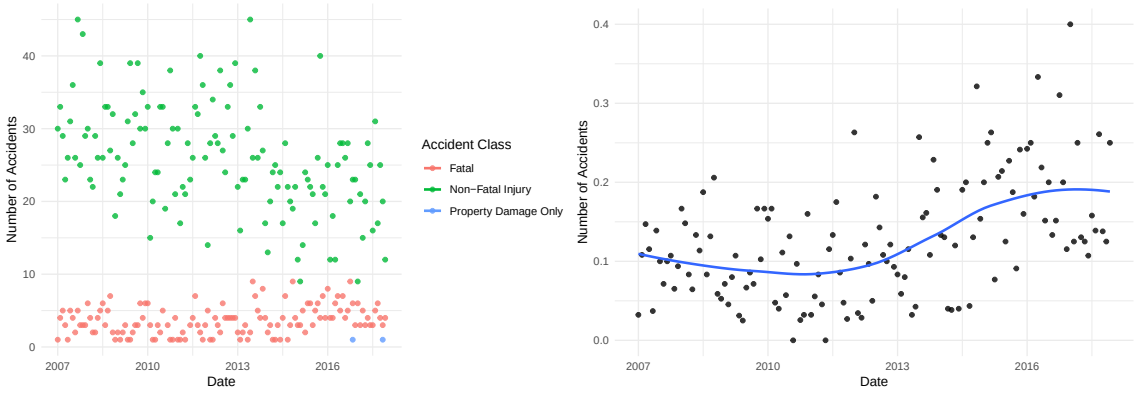
4. Run the application from within RStudio



Using this application for our exploratory analysis allowed us to uncover some interesting trends quite quickly. Another advantage was that the application was one way to check whether a feature was reasonable to include into our models. Moreover, because it is simple to add inputs and functionality we could customize it to meet our needs dynamically.

Firstly, the number of accidents did not appear to increase with year even though the population of Toronto grew by a considerable amount from 2007 to 2017. While it's great news for Torontonians, this could be a result of many factors. It would be favorable to assume that the number of accidents decreased due to the efforts of the City to improve road safety, it is equally likely for it to have been because the Toronto Police were less rigorous with their data keeping, or that people involved in accidents were less likely to report it and get the police involved. In addition, while the total number of accidents have been going down, the number of fatal accidents have remained stable, causing the proportion of fatal accidents to actually increase by year.

Any plots with smooth lines used Loess - the default in ggplot to avoid distracting the reader if the data was very noisy.

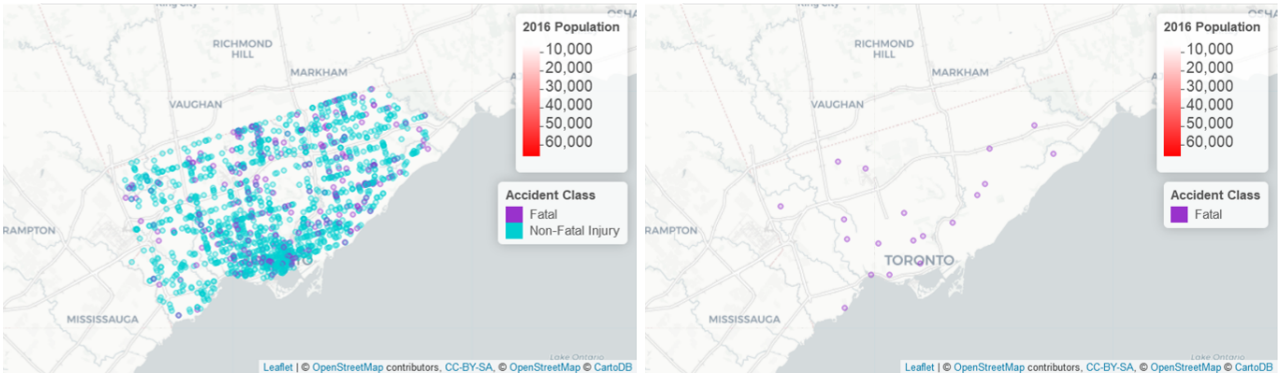


(a) Accidents by Class (b) Total Number of Accidents

Figure 2: Loess-Smoothed plots of Total Accidents and Accidents by Class in the City of Toronto in the years 2007 - 2017

The accidents also appear to be concentrated in downtown core which is likely due to the high population density. However, it is worthy to note that fatal accidents do not appear to be concentrated in downtown. This could be due to the lower speed limits, and shorter intersections which do not allow cars to accelerate as much as freeways or even arterial roads.

The types of vehicles that were involved in an accident also displayed differences in the proportion of fatalities recorded. All accidents in the dataset represent those that involved automobiles and therefore, as expected, when pedestrians or cyclists were involved as well, the proportion of fatalities was much higher. Accidents involving bicycles also revealed interesting spatial patterns.



(a) Pedestrians (b) Cyclists

Figure 3: Pedestrian and Cyclist Deaths in the City of Toronto

As expected, most of them were concentrated in downtown since there is more infrastructure in place for cyclists in this area of the city and because the shorter distances and traffic levels make it a more popular transportion method. We find,

however, that the fatal accidents are more uniformly distributed around the city perhaps making the case that bicycle lanes are effective at preventing lethal accidents for cyclists.

The road conditions, namely the visibility of the road and whether it had precipitated also affected the probability of an accident occurring. The following is a map of where the accidents occurred for the different road conditions.

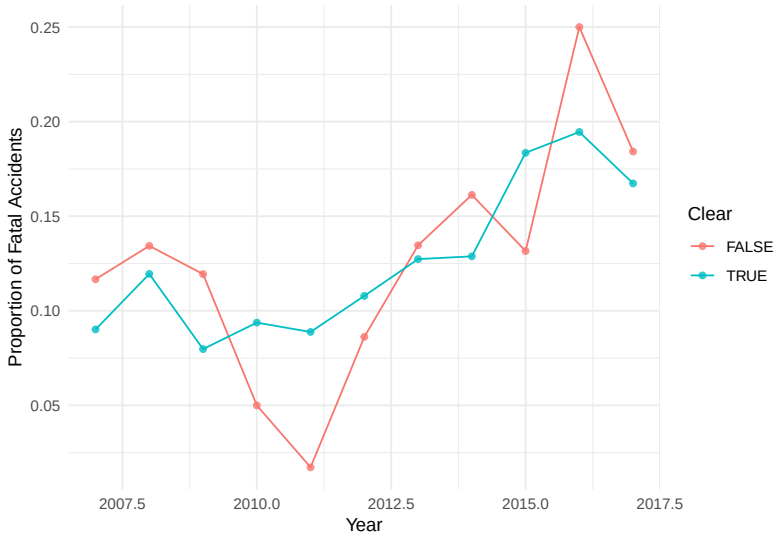
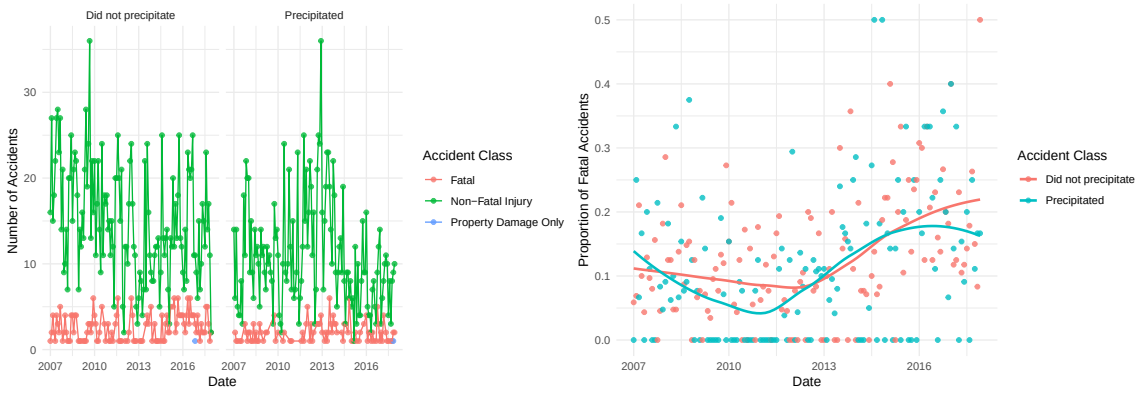


Figure 4: Proportion of Fatal Accidents by Visibility of Road

Surprisingly, if it had precipitated the day of the accident, there were periods of time where it was less likely we were to find a fatal accident. This may be because bad weather deters people from going outside and driving. Note that due to data limitations, the type of precipitation was not distinguished.



(a) Count (b) Proportion

Figure 5: Loess-Smoothed plots of Accidents by Precipitation in the City of Toronto in the years 2007 - 2017

By summing up counts from 2007 to 2017, West Humber-Clairville appeared to be the deadliest intersection followed by South Parkdale, then Wexford/Maryvale. Thankfully, the fatalities appeared to be quite low compared to the total number of collisions reported by the Toronto Police.

hood_name	Total Fatalities	Total Collisions
Don Valley Village	7	31
South Parkdale	6	56
West Humber-Clairville	6	124
Downsview-Roding-CFB	4	59
Morningside	4	12
Clairlea-Birchmount	3	56

West Humber-Clairville, and Wexford/Maryvale appear again as a dangerous neighborhood even when focussing on pedestrian or cyclist fatalities.

hood_name	Total Pedestrian Fatalities	Total Collisions with Pedestrians
Birchcliffe-Cliffside	2	14
Caledonia-Fairbank	2	10
Clairlea-Birchmount	2	30
Danforth East York	2	10
Don Valley Village	2	16
Kennedy Park	2	22

hood_name	Total Cyclist Fatalities	Total Collisions with Cyclists
Annex	1	10
Dovercourt-Wallace Emerson-Junction	1	9
Downsview-Roding-CFB	1	4
Rockcliffe-Smythe	1	4
Rosedale-Moore Park	1	3
South Parkdale	1	8

3 Modeling

We fit and assess 3 models in our project:

- **Meaningful temporal and fixed effects in accident fatality:** Bayesian Mixed-Effects Semi-parametric Logit Model
- **Seasonality and Prediction:** Semi-Parametric Poisson Regression Model
- **Spatial Intensity:** Log-Gaussian Cox Process Model (Work in Progress)

3.1 Bayesian Mixed-Effects Semi-parametric Logit Model

Mixed effects logistic regression is used to model binary outcome variables, in which the odds of the outcomes are modeled as a linear combination of the predictor variables when data are clustered (random effects). This mixed effect model is used to describe the binomial probability of an auto accident resulting to fatality, taking into account not just unobserved differences between neighborhoods, but also the evolution of these odds through time with the inclusion of semi-parametric terms.

$$Y_{ijt} \sim \text{bernoulli}(\pi_{ijt}) \tag{1}$$

$$\text{logit}(\pi_{ijt}) = X_{ijt}\beta + U_i + f(W_t) \tag{2}$$

$$U_i \sim N(0, \sigma_U^2) \quad \text{(Residual Time Component)} \tag{3}$$

$$W_{t+1} - W_t \sim N(0, \sigma_W^2) \quad \text{(RW1 - Time Trend Component)} \tag{4}$$

The fixed effects of this model contains are *visibility*, *types of road*, *traffic control* and *Precipitation*. Those covariates used in the model are unrelated to the personel involved in the accidents, so factors such as condition of the drivers are not included.

- The covariate **visibility** was binarized to either "Clear" or "Not Clear", "Clear" was used as reference.
- For covariate **types of road**, "Major Arterial", "Major Arterial Ramp" and "Minor Arterial" were grouped into "Arterial"; "Expressway", "Expressway Ramp" were grouped into "expressway"; "Local", "Laneway" were grouped into "Local", where "Local" was used as reference.
- For covariate **traffic control**, "School Guard", "Police Control", "Traffic Controller" were grouped into "Human Control", and since there is not fatal accident in "Human Control", all records under "Human Control" were removed to avoid spiked estimate."Stop Sign", "Yield Sign", "Traffic Gate" were grouped into "Traffic Sign" and "Pedestrian Crossover", "Streetcar (Stop for)" were grouped into "Pedestrian Crossing". "No Traffic Control" was used as reference.

3.2 Semi-Parametric Poisson Regression (GAM) Model

A semi-parametric Poisson temporal model is used to fit the total accident counts with months as factors, number of days from 2007 to 2017 as a non-parametric time term and neighbourhoods as random effects. The population of the City of Toronto is estimated with a linear growth function extrapolating the yearly growth rate in population from 2006 (est. 2'503,281) to 2016 (est. 2'731,571). This model includes an offset term corresponding to the logarithm of population, to account for observed growth in population.

$$Y_i \sim Poisson(O_i\lambda_i) \tag{5}$$

$$log(\lambda_i) = X_i\beta + f(day) + f(\mu_i) \tag{6}$$

3.3 Log Gaussian-Cox Process Model

A spatial model Log Gaussian-Cox Process (LGCP) is used to fit the accident counts in 2017 with intercept only.

$$Y_{ij} \sim Poisson(O_i\lambda(s_i)) \tag{7}$$

$$\lambda(s_i) = U(s) \tag{8}$$

$$cov[U(s+h), U(s)] = \sigma^2 \rho(h/\phi; v) \tag{9}$$

4 Results

4.1 Bayesian Mixed-Effects Semi-parametric Logit Model

Below are the resulting estimates of this model:

Table 1: Posterior mean and 2.5 and 97.5 percentiles for the odds ratio of deadly accident by model coefficients

	mean	0.025quant	0.975quant
(Intercept)	0.115	0.078	0.168
visibilitybNot Clear	1.182	0.940	1.481
roadclassArterial	1.067	0.788	1.459
roadclassCollector	0.987	0.661	1.474
roadclassExpressway	1.737	1.023	2.934
trafficctrlPedestrian Crossing	0.871	0.447	1.619
trafficctrlTraffic Sign	0.557	0.422	0.730
trafficctrlTraffic Signal	0.538	0.462	0.628
persontypePedestrian not involved	0.638	0.542	0.752
totprecipmm	0.980	0.965	0.995
SD for weeknum	0.046	0.020	0.096
SD for weekiid	1.491	1.349	1.622
SD for hoodid	0.917	0.772	1.069

- The odds of having fatality are higher when driving on highway.
- Having traffic signage and traffic light leads to a lower odds, compared to no control.
- Accidents without pedestrian (vehicle to vehicle) involved has lower odds of having fatality
- The odds of fatality are slightly lower when there is more precipitation. -2% odds of fatality with 1mm of precipitation increasing. It may be due to drivers slow down their speed when they have difficulty seeing clear ahead or knowing road is slippery.

The time trend graph below shows the odds of fatality rising until mid-2016 and then decreasing. This is consistent with press reports deeming 2015-2017 as bad years for Toronto in terms of fatality. The decrease post-2017 could be attributed to the Vision Zero municipal plan to address road fatalities.

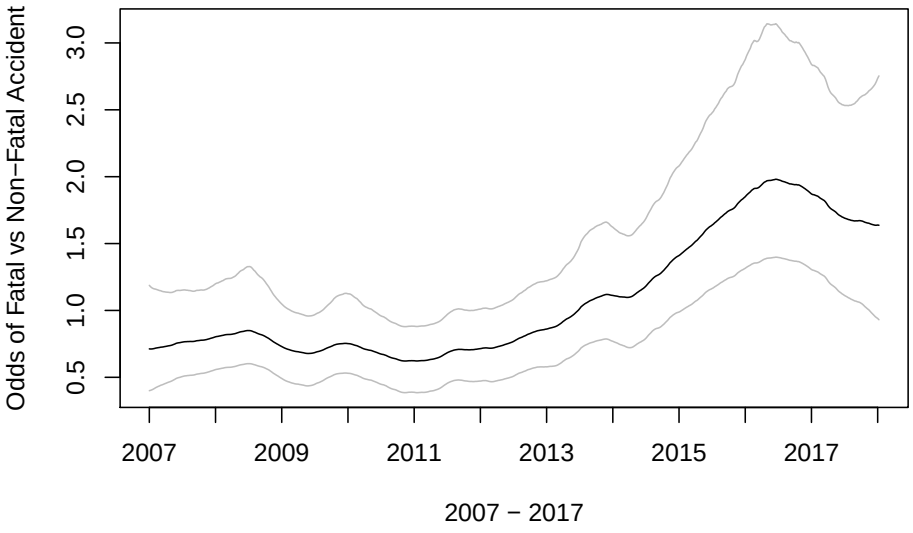
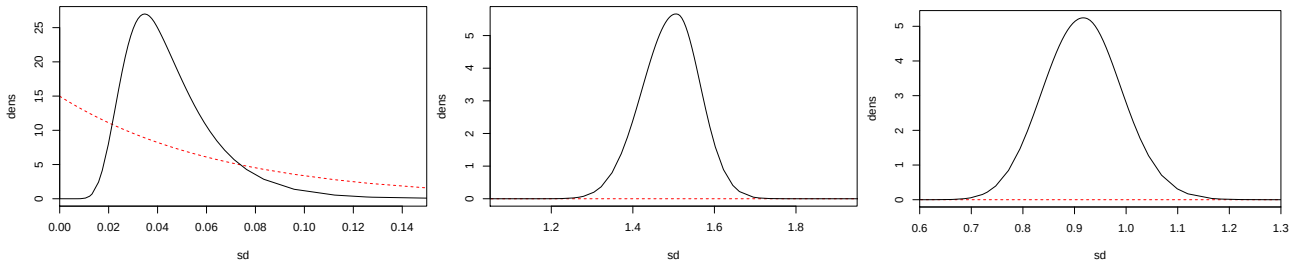


Figure 6: Plot of time trend effect of Odds of Fatality for the City of Toronto

The below plots of posterior distributions for the parameters of this model indicate, namely plot (b) on the residual time variation, indicate that there are still temporal factors affecting our estimation which were not controlled for in the model. This makes sense given the arguably little information about road structure and policy that we added as model covariates. It is also important to note that this model does not model correlation in space, which we hypothesize is very important for the task at hand.



(a) SD of time trend (b) SD of residual time variation (c) SD of neighborhood random effect

Figure 7: Plot of posterior distributions on random intercept (neighborhood) and random time components, showing different degrees of influence on the data

4.2 Semi-Parametric Poisson Regression (GAM) Model

Table 2: Estimated coefficients and their standard errors. Here the baseline month is January.

	Estimate	Std. Error
(Intercept)	-13.638	0.037
monthf02	-0.034	0.052
monthf03	-0.009	0.050
monthf04	0.028	0.049
monthf05	0.117	0.048
monthf06	0.076	0.046
monthf07	0.080	0.047
monthf08	0.104	0.047
monthf09	0.032	0.047
monthf10	0.060	0.047
monthf11	0.016	0.048
monthf12	-0.003	0.050

From the table above we may notice summer has higher amount of accidents in general. It may because there are more drivers and more racing cars in the summer. Also drivers are more cautious when driving in snow days. A rate ratio plot below show this same pattern.

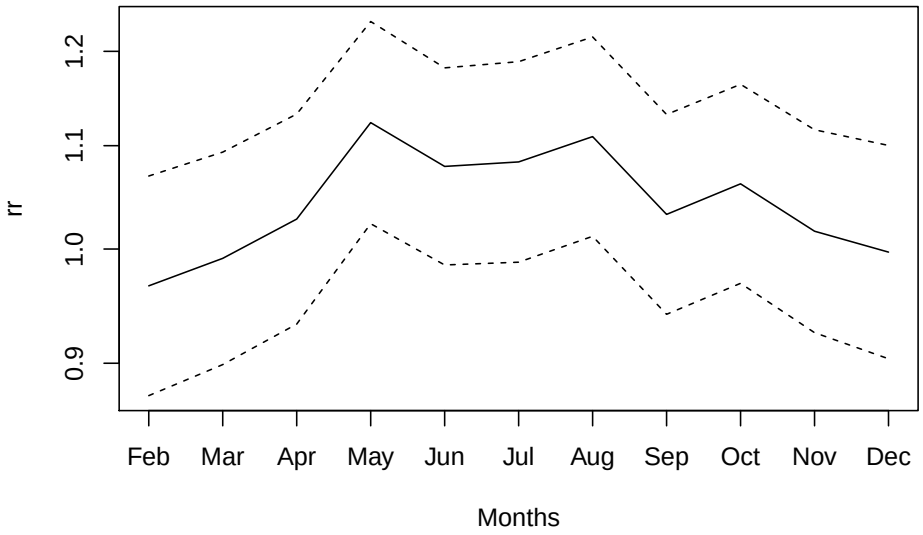
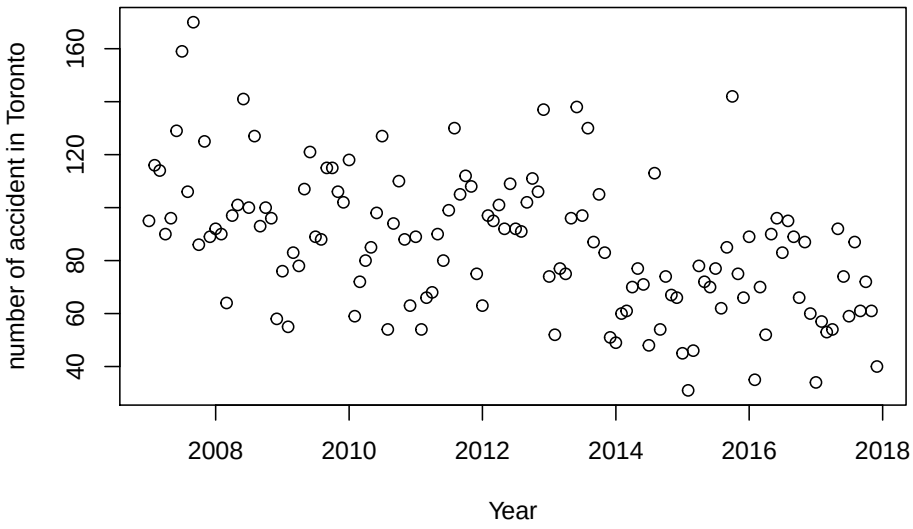
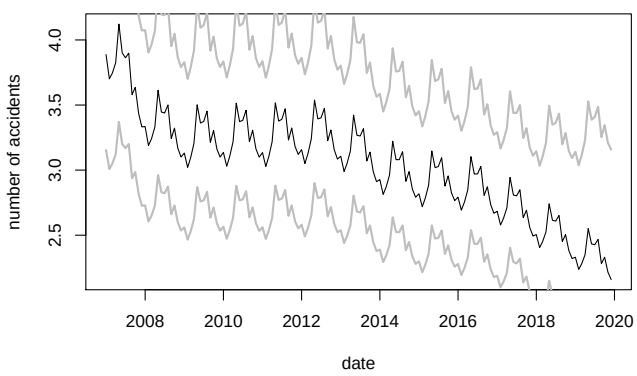


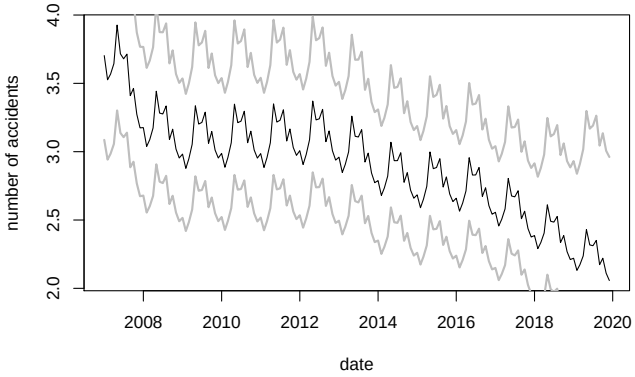
Figure 8: Estimated seasonality of accidents

Below are plots of number of accidents happend per month in Toronto and the predictions of accidents in neighbour 5 and 122 from GAM. In gereral they capture a trend that the monthly number of accidents are descreasing over years.





(a) Elms-Old Rexdale

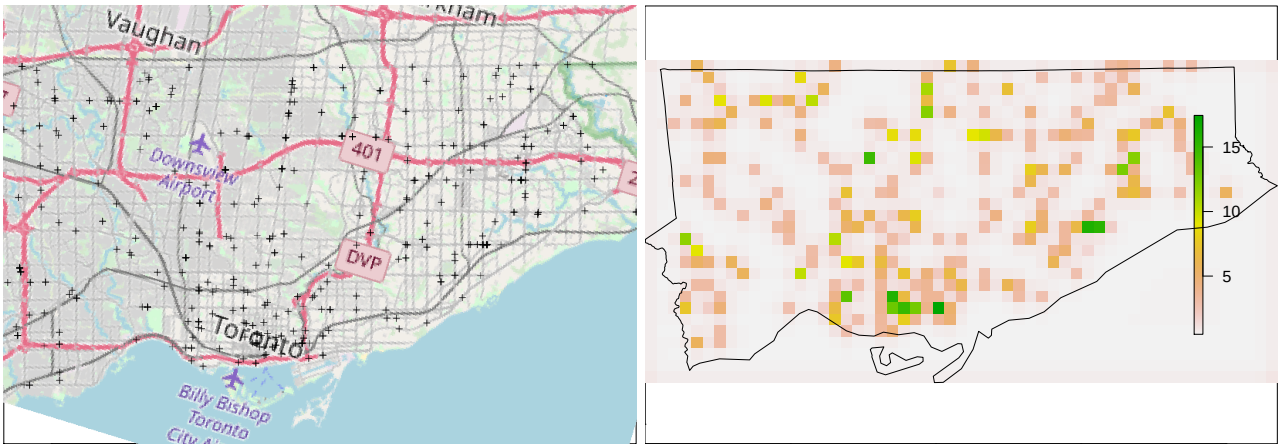


(b) Birchcliffe-Cliffside

Figure 9: Estimated accident counts per neighborhood through the years

4.3 Log Gaussian-Cox Process Spatial Model

The plots below correspond to the observed and expected accident counts (λ) in Toronto in the year 2017, as estimated by our model. We could see that many accidents are expected in downtown area and along the Yonge street. More traffic control may be required, perhaps in the shape of better signage or human control.



(a) Observed (b) Estimated

Figure 10: Observed and estimated accident counts in Toronto in the year 2017

5 Conclusions and Discussion

In this project we sought to explore and predict various components of road safety in the City of Toronto, namely, vehicular collisions. We adopted two different frameworks for our analysis: A spatial and a longitudinal one, with greater success in our longitudinal exploration.

One of the biggest limitations in our project has been data quality and granularity. The data made available by Geotab does not include large areas of the City of Toronto. Moreover, there are plenty missing observations. We also acknowledge the fact that the collision information we procured from the Toronto Police Service may not describe perfectly the actual number of incidents, as there are many of these that are non-fatal or go unreported.

With regards to our spatial approach, we realized rather quickly that accessible and up-to-date spatial datasets are difficult to obtain, perhaps due to the cost and privacy concerns tied to it. We first worked with Geotab's datasets thanks to their extensive information, some even covering the entire City of Toronto. However, we failed to convert them into a proper and usable spatial data format. We can definitely expand the scope of our spatial analysis with the appropriate information/covariates, estimating spatial intensities even for regions where we may not observe the outcome of interest. A solid spatial model would provide great insight for the betterment of traffic control policies, road infrastructure, and perhaps even better insurance quotes.

6 Appendix: Dataset Variables and Definitions

Feature	Description	Source
speeding	Binary indicator on whether the vehicle had speeding	Automobile (Toronto Police)
ag_driving	Binary indicator on whether the driver had angry driving	Automobile (Toronto Police)
at_redlight	Binary indicator on whether at redlight	Automobile (Toronto Police)
alcohol	Binary indicator on whetherthe driver had alcohol	Automobile (Toronto Police)
disability	Binary indicator on whetherthe driver had disability	Automobile (Toronto Police)
key	Unique identifier of personale involved in the accident	Automobile (Toronto Police)
accident_key	Unique identifier for the accident	Automobile (Toronto Police)
date	Date in range(2007-01-01 to 2017-12-31) inclusive	Automobile (Toronto Police)
time	Time in range(00:00 to 23:59) inclusive	Automobile (Toronto Police)
hour	Hour in range (0 to 23) inclusive	Automobile (Toronto Police)
minute	Minute in range(0 to 59) inclusive	Automobile (Toronto Police)
long	longtitude of the accident location	Automobile (Toronto Police)
lat	latitude of the accident location	Automobile (Toronto Police)
street1	street name 1 of the accident location	Automobile (Toronto Police)
street2	street name 2 of the accident location	Automobile (Toronto Police)
offset	offset details of the accident location	Automobile (Toronto Police)
road_class	Class of the road (e.g. Collector)	Automobile (Toronto Police)
district	District name of the accident location	Automobile (Toronto Police)
located	Road type of the accident location (e.g. Intersection)	Automobile (Toronto Police)
accident_loc	Road type of the accident location (e.g. At Intersection)	Automobile (Toronto Police)
traffic_ctrl	Whether the accident location was under traffic control (e.g. traffic signal)	Automobile (Toronto Police)
visibility	Condition of visibility when accident happened (e.g. clear)	Automobile (Toronto Police)
road_condition	Condition of road when accident happened (e.g. wet)	Automobile (Toronto Police)
acc_class	Class of the accidents (e.g. Fatal)	Automobile (Toronto Police)
impact_type	Type of accidents (e.g. Pedestrain collisions)	Automobile (Toronto Police)
person_type	Type of personale for this record (e.g. driver)	Automobile (Toronto Police)
person_age	Age of the personale for this record	Automobile (Toronto Police)
Injury	Accident injury result to the personale (e.g. Major)	Automobile (Toronto Police)
fatality_no	Number of fatality in the accident	Automobile (Toronto Police)
init_dir	Direction of vehicle	Automobile (Toronto Police)
veh_type	type of vehicle involved	Automobile (Toronto Police)
manoeuver	driving condition when the accident happened	Automobile (Toronto Police)
driver_act	Driver's action when the accident happened	Automobile (Toronto Police)
ped_act	Pedestrain's action when the accident happened	Automobile (Toronto Police)
cyc_type	type of cyclist involved in the accident	Automobile (Toronto Police)
cyc_act	Cyclist's action when the accident happened	Automobile (Toronto Police)
cyc_cond	Condition of the cyclist	Automobile (Toronto Police)
inv_ped	number of pedestrian involved in the accident	Automobile (Toronto Police)
inv_cyc	number of cyclist involved in the accident	Automobile (Toronto Police)
inv_moto	number of motocycle involved in the accident	Automobile (Toronto Police)
inv_truck	number of truck involved in the accident	Automobile (Toronto Police)
inv_emergveh	number of emergency vehicle in the accident	Automobile (Toronto Police)
had_passenger	Binary indicator on whether the vehicle had passenger	Automobile (Toronto Police)
division	division number of the accident location	Automobile (Toronto Police)
ward_name	name of the ward	Automobile (Toronto Police)
hood_num	Neighbourhood ID in range (1-140) inclusive	Automobile (Toronto Police)
hood_name	Name of the Neighbourhood	Automobile (Toronto Police)
max_temp	Monthly max on highest daily Weather degree (celsus)	Government of Toronto (UofT
min_temp	Monthly min on highest daily Weather degree (celsus)	Government of Toronto (UofT
tot_rain_mm	Daily total rain volumn in mm	Government of Toronto (UofT
tot_snow_cm	Daily total snow volumn in cm	Government of Toronto (UofT
tot_precip_mm	Daily total percip volumn in mm	Government of Toronto (UofT
ground_snow_cm	Volumn of snow on group in cm	Government of Toronto (UofT
population	number of population in the neighbourhood	Statistic Canada

7 Appendix: Neighborhoods of Toronto

These are the neighborhoods of the City of Toronto as defined by the municipal government in the year 2016:

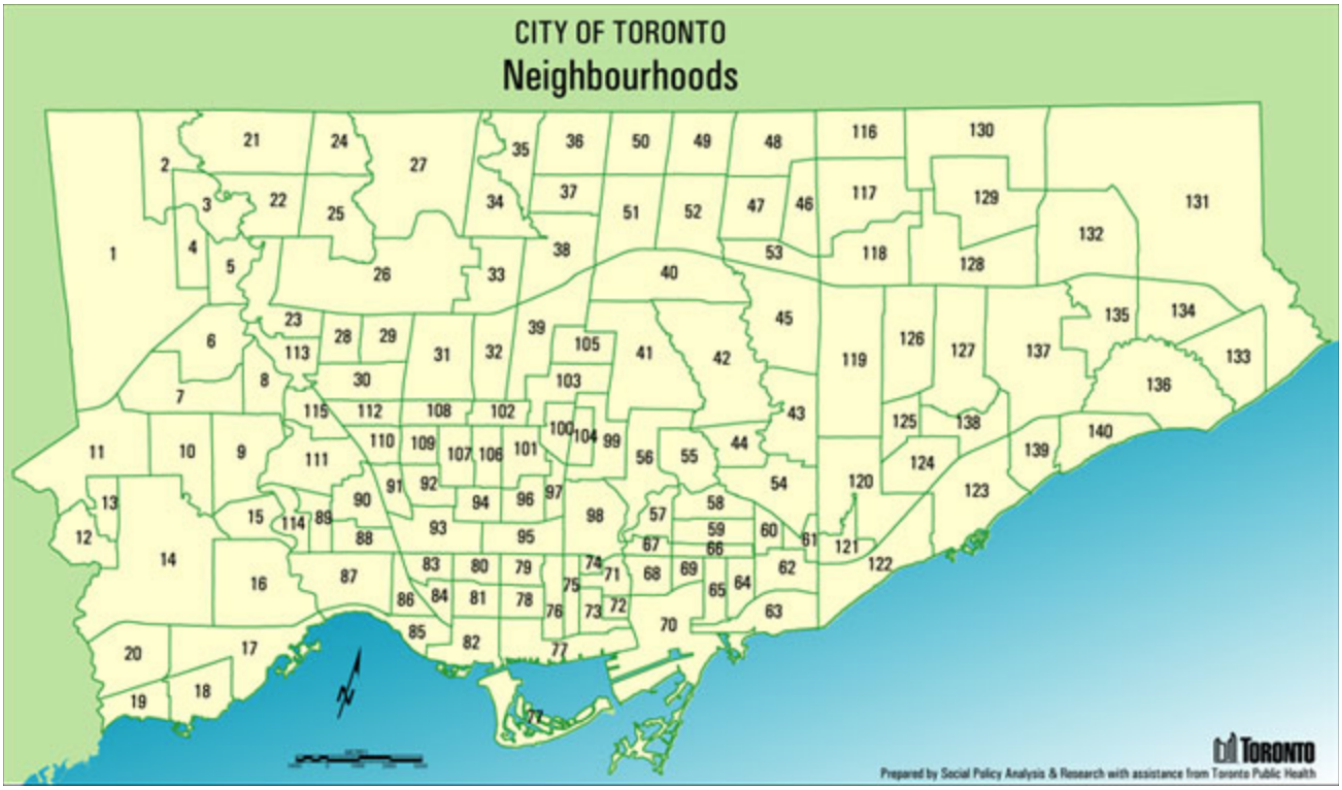
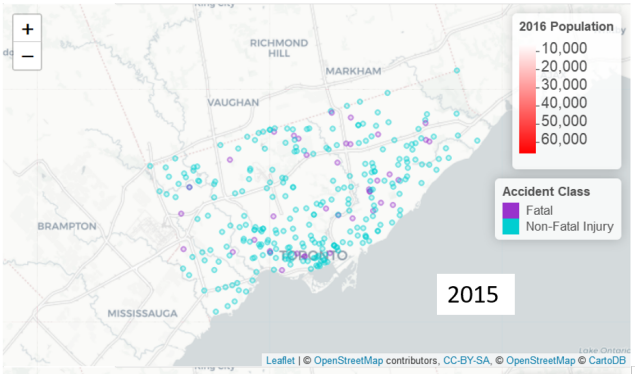


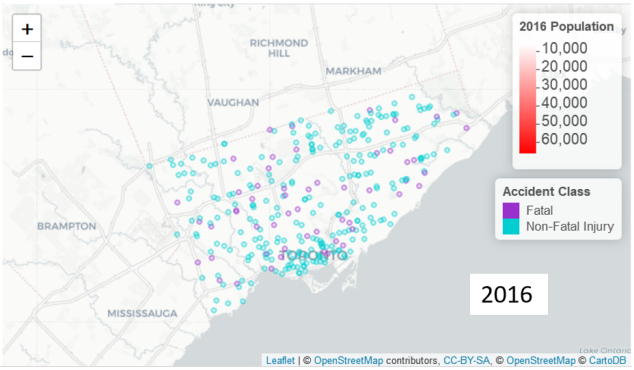
Figure 11: Official City of Toronto Neighborhoods

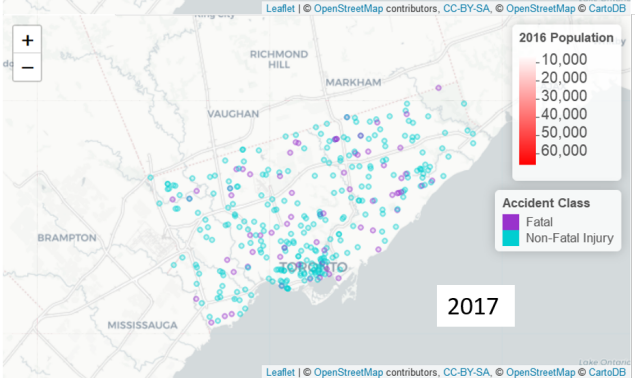
Refer to the **City of Toronto** for the neighborhood names matching the indeces above.

8 Appendix: Visualizations of Collision Locations through the Years

The below visualizations are screenshots taken from our Shiny app, available here:







9 Appendix: Code

```
library(MASS); library(lmtest); library(knitr); library(kableExtra); library(nleqslv);
library(Pmisc); library(extrafont); library(VGAM); library(INLA); library(MEMSS);
library(nlme); library(ciTools); library(sf); library(tibble); library(sp); library(dplyr);
 library(lme4);  library(mgcv); library(data.table);
library(geostatsp, quietly = TRUE);library(mapmisc, quietly = TRUE);library(maptools);
library(raster);library(ggmap); library(rgdal); library(ggplot2);library(plyr);
library(zoo);library(tidyverse, quietly = T, warn.conflicts = F, verbose = F)
library(htmltools);library(zoo);library(lubridate);library(plotly);


knitr::opts_chunk$set(fig.pos = 'H');
options(tinytex.verbose = TRUE)
# Loading polygon and population data from the City of Toronto
population <- read.csv("https://raw.githubusercontent.com/sergiosonline/data_sci_geo/master/data/neighbourhoo

#require(sf)
shape <- read_sf(dsn = "~/Documents/Github/data_sci_geo/data/neighbourhoods_planning_areas_wgs84_SEB/", layer

neighborhoods <- shape

# Adding populaation info to neighborhood polygon
neighborhoods <- add_column(neighborhoods, '2016pop'=NA, 'x_coords' = NA, 'y_coords' = NA)

# Separating X and Y coordinates from polygon
for (hood in neighborhoods$AREA_NAME) {
  ## Adding population
  pop = as.numeric(neighborhoods[neighborhoods$AREA_NAME == hood,][["AREA_S_CD"]])
  neighborhoods[neighborhoods$AREA_NAME == hood,]$'2016pop' =
    population[population$HoodID == pop,]$Pop2016
  ## Adding x-y
  temp = unlist(subset(neighborhoods,AREA_NAME == hood)$geometry[[1]])
  ll = length(temp)
  x_coord = list(temp[1:(ll/2)])
  y_coord = list(temp[((ll/2)+1):ll])
  neighborhoods[neighborhoods$AREA_NAME == hood,]$x_coords = x_coord
  neighborhoods[neighborhoods$AREA_NAME == hood,]$y_coords = y_coord
}

st_write(neighborhoods,"~/Documents/Github/data_sci_geo/data/neighbourhoods_planning_areas_wgs84_SEB/NEIGHBOR
         , delete_layer = TRUE)

neighborhoods <- read_sf(dsn = "~/Documents/Github/data_sci_geo/data/neighbourhoods_planning_areas_wgs84_SEB/

###ALTERNATIVE VISUALIZATION
neighborhoods = rgdal::readOGR(dsn = "~/Documents/Github/data_sci_geo/data/neighbourhoods_planning_areas_wgs8
accidents <- read.csv("https://raw.githubusercontent.com/sergiosonline/data_sci_geo/master/data/accidents.csv

# Set up df
neighborhoods@data$id = rownames(neighborhoods@data)
neighborhoods.points = fortify(neighborhoods, region="id")
neighborhoods.df = join(neighborhoods.points, neighborhoods@data, by = "id")

# Plotting command - basic

#ggplot(neighborhoods.df) + aes(long,lat,group=group,fill=X2016pop)+ geom_polygon() +
#+   geom_path(color="black") + coord_equal()
```

```r
# Adding points

#sum_accidents <- accidents %>%
#  group_by(Neighbourhood, YEAR) %>%
#  summarize(`Total Fatalities` = sum(INJURY == "Fatal", na.rm = T),
 #           `Total Collisions` = n()) %>%
#  arrange(desc(`Total Fatalities`))

cbPalette <- c("#999999", "#E69F00", "#56B4E9", "#009E73", "#F0E442", "#0072B2", "#D55E00", "#CC79A7")

#To use for fills, add
#scale_fill_manual(values=cbPalette)

# To use for line and point colors, add
#scale_colour_manual(values=cbPalette)


ggmap::register_google(key = "AIzaSyB13QyZy3PLnR5BYGtwezYWFaSq_pjrNjA")


#####
p0 <- ggmap(get_googlemap(center = c(lon = -79.384293, lat = 43.71),
                    zoom = 10, scale = 2,
                    maptype ='terrain',
                    color = 'color'), maprange=T,extent = "normal") +
    labs(x = "", y = "") +
    scale_x_continuous(limits = c(-79.63926, -79.11524), expand = c(0, 0)) +
scale_y_continuous(limits = c(43.581, 43.85546), expand = c(0, 0)) +
  theme(legend.position = "right",
        panel.background = element_blank(),
        axis.line = element_blank(),
        axis.text = element_blank(),
        axis.ticks = element_blank(),
        plot.margin = unit(c(0, 0, -1, -1), 'lines')) +
  xlab('') +
  ylab('')

#p2 <- p0 + geom_polygon(aes(long,lat,group=group,fill=NA,color="white"),color="plum",fill=NA,data=neighborho
#                        breaks=c("Fatal", "Non-Fatal Injury"),
#                        labels=c("Fatal", "Non-Fatal"))

p1 <- p0 + geom_polygon(data=neighborhoods.df, aes(long,lat,group=group, fill=X2016pop),alpha = 0.8,color="pl

p1

#p2
data.frame(`Accident Key` = c(5002235651, 5000995174, 5000995174, 1249781),
           Fatal = c(1, 1, 1, 0),
           Date = c("2015-12-30", "2015-06-13", "2015-06-13", "2011-08-04"),
           Neighborhood = c("Greenwood-Coxwell", "Annex", "Annex", "Bay Street Corridor"),
           Population = c(7072, 26703, 26703, 19348),
           `Max Temp` = c(4.7, 22.3, 22.3, 26.4)) %>%
  kable() %>%
  kable_styling(bootstrap_options = c("striped"))
## Visualizing neighborhoods of Toronto for reference
url9 <- "https://raw.githubusercontent.com/sergiosonline/data_sci_geo/master/reports/final/images/screenshot%
download.file(url = url9,
          destfile = "app-instructions.png",
```

```
          mode = 'wb')

knitr::include_graphics(path="app-instructions.png")
## Visualizing neighborhoods of Toronto for reference
url10 <- "https://raw.githubusercontent.com/sergiosonline/data_sci_geo/master/reports/final/images/screenshot
download.file(url = url10,
          destfile = "general-app.png",
          mode = 'wb')

knitr::include_graphics(path="general-app.png")
accidents <- read.csv("https://raw.githubusercontent.com/sergiosonline/data_sci_geo/master/data/final/acciden

p <- accidents %>%
  group_by(accident_key) %>%
  filter(row_number() == 1) %>%
  ungroup() %>%
  mutate(monthyear = as.yearmon(date),
         month = month(date),
         year = year(date),
         numdays = as.numeric(days_in_month(as.Date(date))))

p %>% group_by(monthyear, acc_class) %>%
      dplyr::summarize(num = n()) %>%
  ggplot(., aes(x = monthyear, y = num, col = acc_class)) +
  geom_point(alpha = 0.8) +
  stat_smooth(se = F) + ylab("Number of Accidents") + xlab("Date") +
  labs(color = "Accident Class") + theme_minimal()

p %>% group_by(monthyear) %>%
  dplyr::summarize(perc_fat = sum(acc_class == "Fatal")/n()) %>%
  ggplot(., aes(x = monthyear, y = perc_fat)) +
  geom_point(alpha = 0.8) +
  stat_smooth(se = F) + ylab("Number of Accidents") + xlab("Date") +
  labs(color = "Accident Class") + theme_minimal()
## Visualizing neighborhoods of Toronto for reference
url13 <- "https://raw.githubusercontent.com/sergiosonline/data_sci_geo/master/reports/final/images/all%20pede
download.file(url = url13,
          destfile = "pedestrians.png",
          mode = 'wb')

url14 <- "https://raw.githubusercontent.com/sergiosonline/data_sci_geo/master/reports/final/images/fatal%20cy

download.file(url = url14,
          destfile = "cyclists.png",
          mode = 'wb')

knitr::include_graphics(path=c("pedestrians.png","cyclists.png"))
p %>% mutate(clear = visibility == "Clear") %>%
  group_by(clear, year) %>%
  dplyr::summarize(prop_fat = sum(acc_class == "Fatal")/n()) %>%
  ggplot(., aes(x = year, y = prop_fat, col = clear)) +
  geom_point(alpha = 0.8) + geom_line() + ylab("Proportion of Fatal Accidents") + xlab("Year") +
  labs(color = "Clear") + theme_minimal()
p %>%
  mutate(precipitated = if_else(tot_precip_mm > 0, "Precipitated", "Did not precipitate")) %>%
  group_by(monthyear, acc_class, precipitated) %>%
  dplyr::summarize(num = n()) %>%
  ggplot(., aes(x = monthyear, y = num, col = acc_class)) +
```

```
  geom_point(alpha = 0.8) + geom_line() + ylab("Number of Accidents") + xlab("Date") +
  labs(color = "Accident Class") + facet_wrap(~precipitated) + theme_minimal()

p %>%
  mutate(precipitated = if_else(tot_precip_mm > 0, "Precipitated", "Did not precipitate")) %>%
  group_by(monthyear, precipitated) %>%
  dplyr::summarize(prop_fat = sum(acc_class == "Fatal")/n()) %>%
  ggplot(., aes(x = monthyear, y = prop_fat, col = precipitated)) +
  geom_point(alpha = 0.8) + geom_smooth(se = F) + ylab("Proportion of Fatal Accidents") + xlab("Date") +
  labs(color = "Accident Class") + theme_minimal()
accidents <- read.csv("https://raw.githubusercontent.com/sergiosonline/data_sci_geo/master/data/final/acciden

accidents %>% group_by(accident_key) %>%
  filter(row_number() == 1) %>%
  ungroup() %>%
  group_by(hood_name) %>%
  dplyr::summarize(`Total Fatalities` = sum(injury == "Fatal", na.rm = T),
            `Total Collisions` = n()) %>%
  arrange(desc(`Total Fatalities`)) %>%
  head() %>%
  kable()%>%
  kable_styling(bootstrap_options = c("striped"))
accidents %>%
  filter(inv_ped == 1) %>%
  group_by(accident_key) %>%
  filter(row_number() == 1) %>%
  ungroup() %>%
  group_by(hood_name) %>%
  dplyr::summarize(`Total Pedestrian Fatalities` = sum(injury == "Fatal", na.rm = T),
            `Total Collisions with Pedestrians` = n()) %>%
  arrange(desc(`Total Pedestrian Fatalities`)) %>%
  head() %>%
  kable()%>%
  kable_styling(bootstrap_options = c("striped"))

accidents %>%
  filter(inv_cyc == 1) %>%
  group_by(accident_key) %>%
  filter(row_number() == 1) %>%
  ungroup() %>%
  group_by(hood_name) %>%
  dplyr::summarize(`Total Cyclist Fatalities` = sum(injury == "Fatal", na.rm = T),
            `Total Collisions with Cyclists` = n()) %>%
  arrange(desc(`Total Cyclist Fatalities`)) %>%
  head() %>%
  kable()%>%
  kable_styling(bootstrap_options = c("striped"))
# Loading final monthly incident data, by neighborhood
incidentdata <- read.csv("https://raw.githubusercontent.com/sergiosonline/data_sci_geo/master/data/accidents.

#incidentdata$Population2 <- incidentdata$Population/1000
#incidentdata$Days_since_start2 <- incidentdata$Days_since_start/100
#incidentdata <- filter(incidentdata, ACCLASS != "Property Damage Only")

#population <- read.csv("https://raw.githubusercontent.com/sergiosonline/data_sci_geo/master/data/toronto_hoo

#Adding neighborhood area
#incidentdata_test <- incidentdata %>%
```

```r
#  left_join(dplyr::select(population, HoodID, area_sqkm), by = c("Hood_ID" = "HoodID")) #%>% mutate(density

#write.csv(incidentdata_test, "~/Desktop/Grad_School/COURSEWORK/Spring 2019/Data Science/rough work/accidents

freqmod1 <- glmer(as.factor(ACCLASS) ~ Days_since_start2 + Tot_precip + Min_temp + (1 + Days_since_start2 |Ne
                  control=glmerControl(optimizer= "Nelder_Mead"))
accidents <- read.csv(file="https://raw.githubusercontent.com/sergiosonline/data_sci_geo/master/data/final/ac
accidents4 = accidents

accidents4$year = substr(as.character(accidents4$date),1,4)
accidents4$month = substr(as.character(accidents4$date),6,7)
accidents4$day = substr(as.character(accidents4$date),9,10)
accidents4$longitude = accidents4$long
accidents4$latitude = accidents4$lat
accidents4$hood_id = as.factor(accidents4$hood_num)


accidents4$date = paste(accidents4$year, accidents4$month, accidents4$day, sep = "-")

timeOrigin = ISOdate(2007,1,1,0,0,0, tz='UTC')
accidents4$daynum = as.integer(as.numeric(difftime(accidents4$date, timeOrigin, units='days')))
accidents4$weeknum = as.integer(as.numeric(difftime(accidents4$date, timeOrigin, units='weeks')))

accidents4 <- filter(accidents4, acc_class!="Property Damage Only")
accidents4$accclass <- ifelse(accidents4$acc_class=="Fatal",1,0)

accidents3 = accidents4
accidents3$visibilityb = as.character(accidents3$visibility)
accidents3$visibilityb = as.factor(ifelse(accidents3$visibilityb =="Clear", "Clear", "Not Clear"))

#factorize hood_id
accidents3$hoodid = as.factor(accidents3$hood_num)

#group road class
accidents3$roadclass = as.character(accidents3$road_class)
accidents3$roadclass = ifelse(accidents3$road_class %in% c("Major Arterial", "Major Arterial Ramp", "Minor Ar

accidents3$roadclass = as.factor(accidents3$roadclass)
accidents3$roadclass = relevel(accidents3$roadclass,ref='Local')

#traffic control class
accidents3$trafficctrl = as.character(accidents3$traffic_ctrl)
accidents3$trafficctrl = ifelse(accidents3$trafficctrl %in% c("", "No Control"), "No Control", ifelse(acciden

accidents3 =  subset(accidents3, trafficctrl != "Human Control")
accidents3$totprecipmm <- accidents3$tot_precip_mm

accidents3$trafficctrl = as.factor(accidents3$trafficctrl)
accidents3$trafficctrl = relevel(accidents3$trafficctrl,ref='No Control')


#group invaded type - may be correlated to road class
accidents3$persontype = as.character(accidents3$person_type)
accidents3$persontype = as.factor(ifelse(accidents3$persontype %in% c("Pedestrian", "Pedestrian - Not Hit"),

accidents3$weekiid = accidents3$weeknum

fitS <- inla(accclass ~ visibilityb + roadclass + trafficctrl + persontype + totprecipmm +
```

```
                f(weeknum, model='rw1' , hyper = list(prec=list(prior='pc.prec', param=c(0.2, 0.05)))
) + f(weekiid, model='iid' , hyper = list(prec=list(prior='pc.prec', param=c(0.2, 0.05)))
)
  + f(hoodid, model='iid', hyper = list(prec=list(prior='pc.prec', param=c(0.25, 0.01)))
), data=accidents3, family='binomial',
control.mode = list(theta = c(2.2, 7.2, 5), restart=TRUE)
)

fitS$priorPost = Pmisc::priorPost(fitS)

resTable1 <- exp(fitS$summary.fixed[, c("mean", "0.025quant",
"0.975quant")]);
resTable2 <- Pmisc::priorPostSd(fitS)$summary[,
c("mean", "0.025quant", "0.975quant")]
restable <- rbind(resTable1,resTable2)
#GAM

# library(Hmisc)

accidents <- read.csv(file="https://raw.githubusercontent.com/sergiosonline/data_sci_geo/master/data/final/ac
accidents4 = accidents

accidents4$year = substr(as.character(accidents4$date),1,4)
accidents4$month = substr(as.character(accidents4$date),6,7)
accidents4$day = substr(as.character(accidents4$date),9,10)
accidents4$longitude = accidents4$long
accidents4$latitude = accidents4$lat
accidents4$hood_id = as.factor(accidents4$hood_num)


accidents_time <- accidents4 %>% group_by(hood_id, year,month) %>% dplyr::summarize(value_perd=n())
accidents_time_weather <- accidents4 %>% group_by(hood_id, year,month) %>% summarise(avg_snow = mean(ground_s

accidents_ts <-  merge(accidents_time, accidents_time_weather, by=c("hood_id","year", "month"), all.x=TRUE)
accidents_ts$date = paste(accidents_ts$year, accidents_ts$month,'01', sep = "-")

accidents_ts$monthf = as.factor(accidents_ts$month)

timeOrigin = ISOdate(2007,1,1,0,0,0, tz='UTC')
accidents_ts$day_num = as.numeric(difftime(accidents_ts$date, timeOrigin, units='days'))
#offset pop
  # pop = accidents4 %>%
  #   select(hood_id, year, Population) %>%
  #   group_by(hood_id, year) %>%
  #   arrange(hood_id, year) %>%
  #   slice(n())
  #
  # pop2 = pop %>%
  #   select(year,Population)%>%
  #   group_by(year) %>%
  #   summarise(Population_sum=sum(Population))
  #
  # accidents_ts <-  merge(accidents_ts, pop2, by=c("year"), all.x=TRUE)
  #estimate population
  A = (2731571-2503281)/10; B = 2503281 - 2006*A
  year = seq(2007, 2017, by=1)

  est_pop = as.data.frame(cbind(year, year*A + B))
```

```r
  names(est_pop)[2] = "population_est"

  accidents_ts <-  merge(accidents_ts, est_pop, by=c("year"), all.x=TRUE)
  accidents_ts$log_pop = log(accidents_ts$population_est)
#plot of time trend without model fitting
  newX = data.frame(date = seq(from = timeOrigin, by = "months", length.out = 12 * 11))
  # plot(newX$date, accidents_ts$value_perd,xlab = "Year", ylab = "number of accident in Toronto")
# accidents_ts$value = cumsum(accidents_ts$value_perd)
# accidents_ts2 = c()
# for (i in 1:length(levels(accidents_ts$hood_id)))
# { temp = accidents_ts
#   temp$hood_num = as.numeric(accidents_ts$hood_id)
#
#   current = subset(temp,temp$hood_num == i)
#   current$value = cumsum(current$value_perd)
#
#   accidents_ts2 = rbind(accidents_ts2, current) }
#####change back not to be cummulative count
accidents_ts2 = accidents_ts
accidents_ts2$value = accidents_ts2$value_perd
#######
accidents_ts2$monthf = relevel(as.factor(accidents_ts2$monthf), ref="01")

accident_ts_gam = gam(value ~ monthf + offset(log_pop) + s(day_num) + s(hood_id,bs="re"), data=accidents_ts2,
# accident_ts_gam = gam(value ~ month_f + s(day_num,bs="re", by = hood_id), data=accidents_ts2, family='poiss

# rownames(accident_ts_gam) = c("Intercept", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "
knitr::kable(restable, digits=3, escape=F, format="latex", booktab=T,linesep = "", caption="Posterior mean an
  kable_styling(latex_options = "hold_position")
# plotting
matplot(
as.numeric(fitS$summary.random$weeknum$ID),
exp(fitS$summary.random$weeknum[,
c('0.025quant','0.975quant', '0.5quant')]), xlab='2007 - 2017', lty=1, col=c('grey','grey','black'), type='l'
axis(1, at=seq(0,600,52), labels=c("2007","2008","2009","2010","2011","2012","2013","2014","2015","2016","201
par(mar = c(4,4,4,2) + 0.1);
#par(mgp=c(2,1,0));

for (Dparam in fitS$priorPost$parameters[2:4]) {
  do.call(matplot, fitS$priorPost[[Dparam]]$matplot)
}
fitS$priorPost$legend$x = "topleft"
#do.call(legend, fitS$priorPost$legend)

# summary(accident_ts_gam)
knitr::kable(summary(accident_ts_gam)$p.table[,1:2],digits=3, escape=F, format="latex", booktab=T,linesep = "
  kable_styling(latex_options = "hold_position")

# check 1 hood
# hood_id=77
plot_predict_hoodid = function(hood_id){
  i = hood_id
  newX = data.frame(date = seq(from = timeOrigin, by = "months", length.out = 12 * 13))
  newX$day_num = as.numeric(difftime(newX$date, timeOrigin, units = "days"))
  newX$monthf = as.factor(substr(as.character(newX$date),6,7))
  newX$year = substr(as.character(newX$date),1,4)
  newX$hood_id = i
  newX_all = newX
```

```
  year = seq(min(newX$year), max(newX$year), by=1)
  est_pop = as.data.frame(cbind(year, year*A + B))
  names(est_pop)[2] = "population_est"

  newX_all <-  merge(newX_all, est_pop, by=c("year"), all.x=TRUE)
  newX_all$log_pop = log(newX_all$population_est)

  newX_all$hood_id = as.factor(newX_all$hood_id)

  accident_ts_gam_pred = predict(accident_ts_gam, newX_all, se.fit = TRUE)
  accident_ts_gam_pred = cbind(newX, accident_ts_gam_pred)

  accident_ts_gam_pred$lower = accident_ts_gam_pred$fit - 2 * accident_ts_gam_pred$se.fit
  accident_ts_gam_pred$upper = accident_ts_gam_pred$fit + 2 * accident_ts_gam_pred$se.fit
  for (D in c("fit", "lower", "upper")) {
  accident_ts_gam_pred[[paste(D, "exp", sep = "")]] = exp(accident_ts_gam_pred[[D]])
  ####################plot rr#################
 # accident_ts_gam_pred_rr = as.matrix(as.data.frame(predict.gam(accident_ts_gam, newX_all, type = "terms", t
 # accident_ts_gam_pred_rr = exp(accident_ts_gam_pred_rr[,c(1,4)] %*% Pmisc::ciMat())
 #
 # matplot(newX_all$year, accident_ts_gam_pred_rr, log = "y", xaxt = "n", xlab = "date", type = "l", lty = c(
 # axis(1, at = difftime(newX_all$year, timeOrigin, units = "days"), labels = format(dSeq, "%Y"))

}


pred_hood = accident_ts_gam_pred
plot(pred_hood$date, pred_hood[, "fitexp"], type = "n", xlab = "date", ylab = "number of accidents")
matlines(pred_hood$date, pred_hood[, c("lowerexp", "upperexp", "fitexp")], lty = 1, col = c("grey","grey", "b
}

# plot_predict_hoodid(5)
# plot_predict_hoodid(122)


#plot rr by month
# accident_ts_gam_pred_rr = exp(summary(accident_ts_gam)$p.table[2:12,1:2] %*% Pmisc::ciMat())
#
# matplot( accident_ts_gam_pred_rr, log = "y", xaxt = "n", xlab = "Months", type = "l", lty = c(1, 2, 2), col
# axis(1, at = 1:11, labels = c("Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"))
#plot rr by month
accident_ts_gam_pred_rr = exp(summary(accident_ts_gam)$p.table[2:12,1:2] %*% Pmisc::ciMat())

matplot( accident_ts_gam_pred_rr, log = "y", xaxt = "n", xlab = "Months", type = "l", lty = c(1, 2, 2), col =
axis(1, at = 1:11, labels = c("Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"))
accidents_ts_y = accidents4 %>% group_by( year,month) %>% dplyr::summarize(value_perd=n())
accidents_ts_y$date = paste(accidents_ts_y$year, accidents_ts_y$month,'01', sep = "-")


accidents_ts_y$monthf = as.factor(accidents_ts_y$month)

timeOrigin = ISOdate(2007,1,1,0,0,0, tz='UTC')
accidents_ts_y$day_num = as.numeric(difftime(accidents_ts_y$date, timeOrigin, units='days'))

newX_all = data.frame(date = seq(from = timeOrigin, by = "months", length.out = 12 * 11))

plot(newX_all$date, accidents_ts_y$value_perd,xlab = "Year", ylab = "number of accident in Toronto")
plot_predict_hoodid(5)
```

```
plot_predict_hoodid(122)
# neighborhoods = rgdal::readOGR(dsn = "C:/Users/ThinkPad/Desktop/Eddy/DS", layer = "NEIGHBORHOODS_WGS84")
# neighborhoods = rgdal::readOGR("C:/Users/ThinkPad/Desktop/Eddy/DS/NEIGHBORHOODS_WGS84.shp",layer="NEIGHBORH

# zoning = rgdal::readOGR("C:/Users/EDDY/Documents/UNIVERSITY/STA2453/Proj2/zoning/ZONING_ZONE_CATAGORIES_WGS
# traffic_signals <- read.csv(file="C:/Users/EDDY/Documents/UNIVERSITY/STA2453/Proj2/traffic_signals.csv", he

accidents <- read.csv(file="https://raw.githubusercontent.com/sergiosonline/data_sci_geo/master/data/final/ac


accidents$YEAR = substr(as.character(accidents$date),1,4)
accidents$longitude = accidents$long
accidents$latitude = accidents$lat


#add new features
#day/night; logdensity
# accidents$day_night = as.factor(ifelse(accidents$Hour >21 | accidents$Hour <6, "Night", "Day")) #day time i
# accidents$logdensity = log(accidents$density) #day time is 1

#subset to 2017 for now
accidents = subset(accidents, accidents$YEAR==2017)
#####

accidents_lonlat = as.matrix(cbind(accidents$longitude, accidents$latitude),nrow=nrow(accidents))

accidents_spatial = SpatialPointsDataFrame(coords= accidents_lonlat, data = accidents, coords.nrs = numeric(0

# spRbind(accidents_spatial, zoning)

accidents2 = spTransform(accidents_spatial, mapmisc::omerc(accidents_spatial, angle=-17))
theMap = mapmisc::openmap(accidents2, maxTiles=4, fact=3)
mapmisc::map.new(accidents2)
plot(theMap, add=TRUE, maxpixels=10^7)
plot(accidents2, col=mapmisc::col2html("black", 0.4), cex=0.6, add=TRUE)


canada <- getData(name="GADM", country="CAN", level=2)
trt_border = subset(canada, NAME_2=="Toronto")
accidents_spatial_border = spTransform(trt_border, projection(accidents2))
# plot(accidents_spatial)

accidents_fit = lgcp(formula = ~ 1, data = accidents2, grid = 55, shape = 1, buffer = 2000,
                     prior = list(range = 6000, sd =0.5), border=accidents_spatial_border,
                                  control.inla = list(strategy='gaussian'), verbose=FALSE)

 mapmisc::map.new(accidents_spatial_border)
 plot(accidents_fit$raster[['predict.exp']]*10^6, add=TRUE)
 plot(accidents_spatial_border, add=TRUE)
var_def <- read.csv("https://raw.githubusercontent.com/sergiosonline/data_sci_geo/master/reports/draft/variab

knitr::kable(var_def, format="latex", booktab=T, linesep = "")%>%
#escape=F,
kable_styling(bootstrap_options = c("striped"))
## Visualizing neighborhoods of Toronto for reference
url7 <- "https://raw.githubusercontent.com/sergiosonline/data_sci_geo/master/reports/draft/toronto-hoods.png"
download.file(url = url7,
          destfile = "toronto-hoods.png",
```

```
          mode = 'wb')

knitr::include_graphics(path="toronto-hoods.png")
## Visualizing neighborhoods of Toronto for reference
url13 <- "https://raw.githubusercontent.com/sergiosonline/data_sci_geo/master/reports/final/images/all%20year
download.file(url = url13,
          destfile = "all-years3.png",
          mode = 'wb')

knitr::include_graphics(path="all-years3.png")
```