Los Angeles Crime Prediction Using Spatio-Bayesian Modeling and Bayesian Multinomial Logistic Regression

Matthew Tsang, Reinhard Wilmer, Veronica Zhao

December 7, 2020

Abstract

The purpose of this project is to help predict the top crimes and to indicate where most of the crimes occur in the city of Los Angeles. In order to better allocate police resources to crime scenes, the police should know the distribution of crime over the city areas and be able to predict the type of crime that is going to occur. A Bayesian Multinomial Logistic Regression model was developed with the brms package to predict the type of crime for the top 5 crimes that occur. This model is evaluated through calculating the test prediction accuracy, comparing it with a Naive-Bayes model as a benchmark, and comparing it to a baseline accuracy of 45.63% [1]. The Bayesian Multinomial Logistic Regression model and Naive-Bayes model achieved very similar test prediction accuracy, with both being around 65%, which is much higher than the baseline accuracy. This study also uses a hierarchical model to model the distribution of crime over police areas in the city. The model for crime rate considers the Bernoulli or Poisson distribution of crime rate for each area where each area follows the same hyperpriors. The model with the lowest reliable eploo value is chosen to model the crime rate. After fitting the two models, the Bernoulli model contains a lower reliable eploo of -7909 compared to -14745 for the Poisson model thus the exact crime rates for each region are modelled with the Bernoulli model. Finally, we generated heatmaps for the posterior draws of the Poisson and Bernoulli hierarchical models which shows the high correlation between the posterior responses of both models and region 12 as the area with the highest number of crimes and probability of crime across all regions based on the posterior draws.

I Introduction

I.I Background

Los Angeles is known to have one of the highest crime rates across all communities in the nation, especially for violent crimes [5]. It seems that most crimes are patterned, meaning similar crimes tend to repeat in the same areas. Thus, it would be beneficial for the police force to know where to best allocate their resources and what type of resource to allocate to a given location [3]. This, in turn, would hopefully reduce inefficient police resource allocation and response time.

I.II Research Problem

In order to improve police resource allocation and response time, we aim to investigate and find where the highest concentration of crimes (hotspots) are in the city according to the districts that each police station is responsible for. Furthermore, given information such as the date of crime, time of crime, area of crime, victim age, victim sex, victim descent, weapons used, and status of the case, a prediction model is made to predict the type of crime for the top 5 crimes/crime code that occur in Los Angeles, which are robbery/210, intimate partner-simple assault/626, battery/624, criminal threats/930 and assault with deadly weapon/230.

II Related Work

There are various papers that focus on crime prediction in their respective subject of research. For instance, Bharati et al. compared the accuracy of various classifier models such as K-Nearest Neighbors, Random Forest, etc. with Bayesian methods such as Multinomial Naive-Bayes classifier. Although this paper does not give a concrete conclusion on which model is most accurate, we decided to use the accuracy of the Multinomial Naive-Bayes model in this paper as our baseline accuracy of 45.63% since it is most similar to the model we used [1]. In another paper, Hu et al. used spatio-temporal Bayesian analysis to predict urban crime in Wuhan, China in 2013. They built a spatio-temporal Bayesian model to analyze spatio-temporal patterns and analyze the associated covariates. While they were able to find significant correlation between burglary crime rate and average resident population, they found that there are many zero crime communities that are not captured by the spatio-temporal model [3]. In addition, a 2010 research by Liao et al. used Bayesian Learning Theory to generate hotspot maps for crime prediction. The results managed to predict the crimes quite accurately; however, it only considered geographical variables due to limitation in data [7]. Similarly, Law et al. used a Bayesian spatial shared component approach in identifying crime hotspots but focuses on cluster analysis and modeling. This paper used a shared component approach to account for spatial and non-spatial correlation structures; however, it does not account for temporal aspects of crime prediction [4]. Other papers like Rajadevi et al. employed a Multinomial Logistic Regression model to predict crime occurrence with a distinct advantage in the target response of crime occurrence which is relatively more simple than crime prediction with more categories such that it is suitable for large datasets in terms of computing time [6]. Similarly, other Bayesian methods such as Naive-Bayes are also quite popular on large datasets as shown in a 2017 paper by Vural et al. by building a Naive-Bayes network model and comparing it with decision tree, where they found that Naive-Bayes outperformed decision tree while retaining about 80% of suspect prediction [8].

While Bayesian inference on crime data using spatial modeling and prediction using Multinomial Logistic Regression is not a new thing, our paper will compare two methods by sampling crime rate from two distribution models for our spatial model priors. Furthermore, while other papers have used Multinomial Logistic Regression to predict crime occurrence, our paper will use Bayesian Multinomial Logistic Regression to predict type of crime based on a given covariate and analyzing the covariates in our posterior draws.

III Dataset

III.I Description

This dataset is provided by the Los Angeles Police Department. It reflects crime incidents in the City of Los Angeles dating back to 2010. There may be some inaccuracies within the data because it was transcribed from original crime reports that were typed on paper. There are also some missing data for location fields and are thus denoted as (0°, 0°). There are 26 columns in the dataset which are: DR Number, Date Reported, Date Occurred, Time Occurred, Area ID, Area Name, Reporting District, Crime Code, Crime Code Description, MO Codes, Victim Age, Victim Sex, Victim Descent, Premise Code, Premise Description, Weapon Used Code, Weapon Description, Status Code, Status Description, Crime Code 1, Crime Code 2, Crime Code 3, Crime Code 4, Address, Cross Street, and Location.

III.II Collection/Processing

This dataset is transcribed from original crime reports dating back from 2010 in the City of Los Angeles by the Los Angeles Police Department (LAPD). This dataset is updated weekly and distributed through their website or through kaggle.com. There are 2.12 million rows and 26 columns, with each row being a crime incident. The data has a Location column and the geographic unit of the data is Latitude/Longitude. There are also around 11% of missing values for most variables.

To visualize some important variables, we used a visualization tool [2] to analyze their frequency, types of values, and relationship with other variables.

The stacked plot below shows the types and number of crimes each area has. The x-axis is the area code; the y-axis is the number of crimes; the colors correspond to the different crimes. Based on this plot, we see that area code 12, which corresponds to the area: 77th Street, has the most number of crimes.

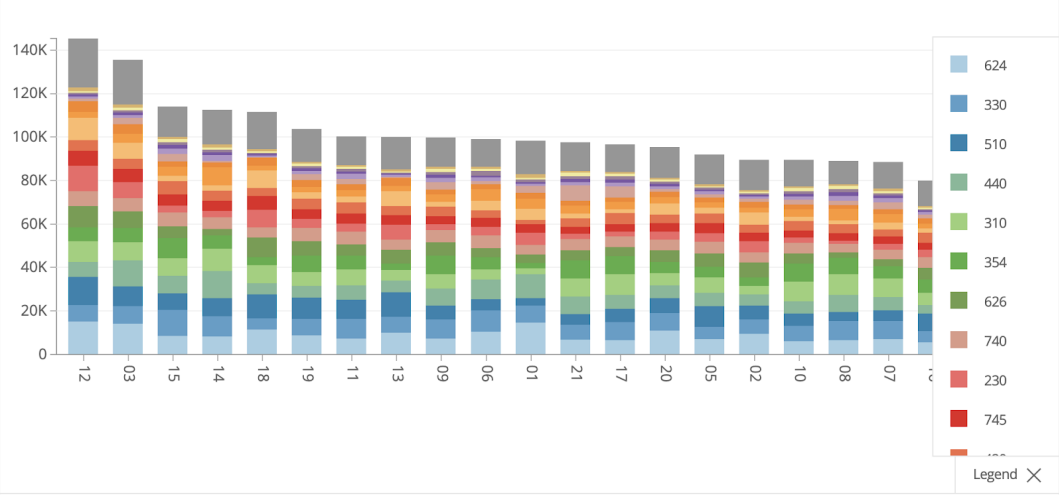

Figure 1: Plot of Crimes by Area

The time series plot below shows the number of crimes from 2010-2018. The plot extends to 2020, however only these years are shown in the graph. We see that the number of crimes is relatively similar but seems to have a slight increase over the years.

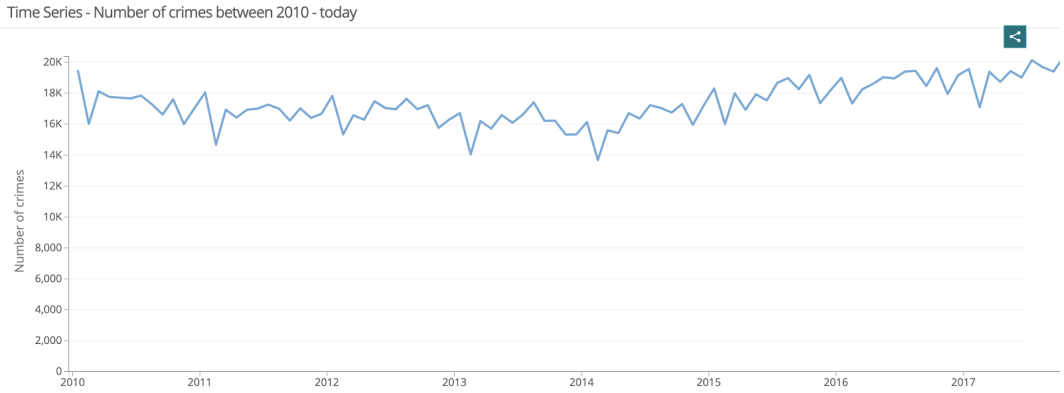

Figure 2: Time Series Plot of Number of Crimes from 2010-2018

To prepare our data for model fitting, we first cleaned our data by removing all NA values. We dropped variables that seemed to correlate with one another, such as description columns. We converted the Location column into two separate columns: Latitude and Longitude. We converted Date.Occurred (date when crime occurred) into days of the week. We also changed Time.Occurred (time when crime occurred) to just the hour. Lastly, we assigned numerical factors to each categorical variable so we can use them to fit our models. Then depending on the models we wanted to fit our data with, we further transformed our dataset.

For the first part of our project, we wanted to predict the top 5 crimes using the predictor variables: Date.Occurred, Time.Occurred, Area.ID, Victim.Age, Victim.Sex, Victim.Descent, Weapons.Used.Code, and Status.Code. We filtered our dataset to only include the top 5 crimes/crime code: robbery/210, intimate partner-simple assault/626, battery/624, criminal threats/930 and assault with deadly weapon/230. Then, we filtered each predictor variable such that values that are low in frequency will be removed (i.e frequency less than 1000), resulting in a dataset of 418455 rows and 9 columns. We sampled 10000 rows from the filtered dataset to use for our model. We ensured that this is a representative sample by making sure it contains the same levels or values for each variable as the dataset.

	Date.Occurred	Time.Occurred	Area.ID	Crime.Code	Victim.Age	Victim.Sex	Victim.Descent	Weapon.Used.Code	Status.Code
291669	6	18	15	624	64	2	19	400	3
120855	4	8	15	624	58	1	19	400	5
62135	1	7	8	624	22	2	9	400	5
274191	2	2	6	210	27	2	4	400	5
339877	2	17	18	230	61	2	4	200	2
114659	4	15	12	624	54	1	4	400	5
368045	7	20	8	626	38	2	19	400	2
316237	4	18	5	624	59	1	9	400	3
133179	3	22	1	626	36	1	19	400	3
267157	6	2	3	624	46	2	3	400	5

Figure 3: Top 10 Rows of Dataframe Used for Multinomial Logistic Regression

For our spatial hierarchical Bernoulli model, the dataset was processed such that there are n rows for n crime reports and p columns for the p area id's. Each element is either a 1 or a 0 to indicate if the crime occurred in that region (1 meaning occurred and 0 meaning not occurred). This dataset contains 667506 rows and 21 columns.

	Area.ID_1	Area.ID_2	Area.ID_3	Area.ID_4	Area.ID_5	Area.ID_6	Area.ID_7	Area.ID_8	Area.ID_9	Area.ID_10	Area.ID_11	Area.ID_12
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	1	0	0
8	0	0	0	0	0	0	0	0	0	1	0	0
9	0	0	0	0	0	0	0	0	0	1	0	0
10	0	0	0	0	0	0	0	0	0	0	1	0

Figure 4: Top 10 Rows of Dataframe Used for Hierarchical Bernoulli Model

As for the spatial Poisson model, the data was processed similarly as the Bernoulli model but each row indicates a week, each column indicates the Area ID, and each element indicates the number of crimes committed in that week in the area. This data contains 53 rows and 22 columns.

	week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	1	728	821	1069	645	618	661	522	372	519	451	518	1314	849	523	595	580	493	1065	774	700	499
2	2	692	634	925	488	484	591	491	313	434	411	415	1168	654	472	460	382	359	903	494	634	435
3	3	689	724	1008	453	484	602	487	294	454	399	484	1246	715	493	499	434	366	1063	547	670	436
4	4	721	720	942	518	518	607	504	308	483	383	482	1210	774	588	487	440	370	1061	536	662	484
5	5	711	765	1017	506	627	605	471	343	479	431	507	1223	821	596	522	463	407	985	602	677	459
6	6	716	677	932	554	468	586	487	309	426	412	486	1175	741	536	479	430	358	973	567	700	487
7	7	757	762	914	550	480	599	457	345	500	430	507	1267	729	517	517	433	386	989	583	629	469
8	8	701	665	901	522	477	632	422	310	499	427	450	1201	742	554	538	428	378	971	527	679	436
9	9	721	714	1011	579	536	612	492	378	528	444	491	1207	732	562	514	492	414	1024	603	673	465
10	10	716	733	977	525	535	593	469	306	491	387	493	1217	825	529	510	490	382	1011	604	641	456

Figure 5: Top 10 Rows of Dataframe Used for Hierarchical Poisson Model

IV Methods

We used the brm function from the brms package to fit the Bayesian Multinomial Logistic Regression model since it is used to fit Bayesian generalized nonlinear multivariate multilevel models. This function allows us to predict a categorical variable with multiple classes using categorical and quantitative predictors. We fit our model on our training set and used it to calculate the training and test accuracy. As a baseline for our model comparison, we also fit a Naive-Bayes model on the training set and obtained the training and test accuracy. Then, we compared both models' accuracy to our baseline accuracy of 45.63% [1].

For the Bayesian Multinomial Logistic Regression model, we used a flat prior for all of the coefficients. The crime codes are modeled by

$$y_1 \dots y_j \dots y_k | \theta_1, \dots \theta_j, \dots, \theta_k \sim multinomial(\theta_1, \dots \theta_j, \dots \theta_k)$$

where y represents the crime code and θ_j where theta is the probability that the crime occurring is the j'th crime. For each θ_j,

$$logit(\theta_j) = \alpha + \sum_{k=1}^{p} \beta_k x_k$$

models θ_j where x are the predictor variables for all p predictors. Therefore the likelihood of each observation is

$$p(y_1, \dots y_j, \dots y_k | \alpha, \beta, x) = \frac{n!}{y_1! \dots y_j! \dots y_k!} \sum_{j=1}^{p} p_j^{y_j}.$$

For the second part of our project, we did spatial modeling to estimate the posterior distribution of crimes over the map of Los Angeles. Using the dataset, we created two hierarchical models. One model will use the Bernoulli distribution to model crime rate in each region and the other will be a Poisson distribution to model the mean number of crimes in each region per week. The two models will be then compared with PSIS-LOO values in order to evaluate the fit. The LOO values will be generated with Stan's Loo package with Pareto smoothed importance sampling, leave one out cross-validation and the expected log predictive density (ELPD) will be compared. The value with a lower, reliable expected log predictive density suggests a better fit thus will be used. With each region, we will estimate the posterior distribution of crime rate, δ. To estimate δ, we compute crime rate p_i, where i represents each region, which is represented by either a Bernoulli model $p_i \sim Bernoulli(p_i)$ or a Poisson model $p_i \sim Poisson(p_i)$. For the Bernoulli model, each region will use a beta prior. The parameters for beta will be drawn from a Gamma distribution $Gamma(1, 1)$ for both priors. For the Poisson model, each region will use a Normal prior where each mu and sigma will be drawn from a $N(0, 1)$ hyperprior distribution and $Inv - \chi^2(10)$ distribution respectively. Each model was created with Stan in R which uses Hamiltonian MCMC to generate the posterior distribution of the Bernoulli and Poisson models.

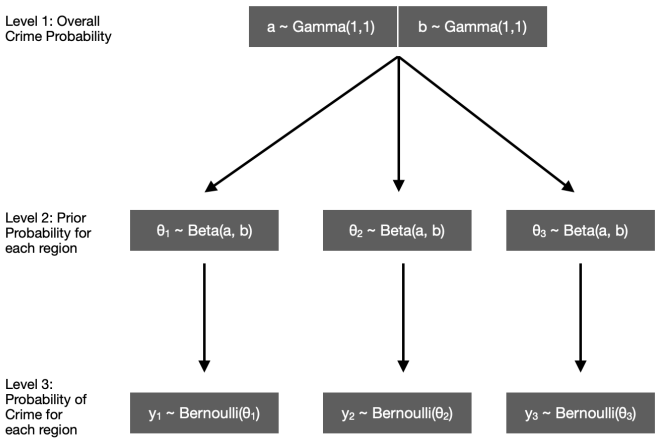

Figure 6: Bernoulli Hierarchical Model

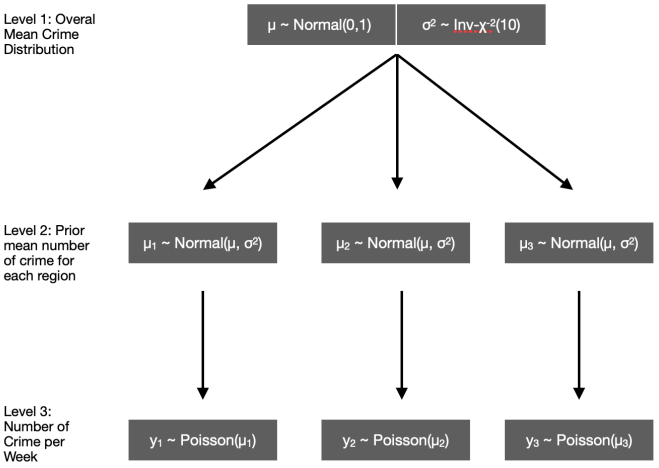

Figure 7: Poisson Hierarchical Model

V Experiments/Results

We split our sample of 10000 into 70% training and 30% testing to assess the prediction accuracy of our models. We fit a Naive-Bayes model to predict Crime.Code using our 8 predictors: Date.Occurred, Time.Occurred, Area.ID, Victim.Age, Victim.Sex, Victim.Descent, Weapons.Used.Code, Status.Code. Then, we calculated its training and test accuracy by comparing its predictions to the actual crime codes. We also fit a Bayesian Multinomial logistic regression model using brm on the same variables, using 1000 iterations, 2 chains, and 250 warmup. After, we calculated its training and test accuracy in the same way as for the Naive-Bayes model. The table below shows the training and test accuracy of each model.

	Training.Accuracy	Test.Accuracy
Bayesian Logistic Regression	0.6468571	0.6450000
Naive Bayes	0.6475714	0.6536667

Figure 8: Table of Training and Test Accuracy

We can see from the results that the Naive-Bayes achieved a training accuracy of 64.78% and test accuracy of 65.34%. Meanwhile, the Logistic model achieved a training accuracy of 64.69% and test accuracy of 64.50%. Both results are very similar to one another, with Naive-Bayes being slightly higher but not significantly. We adjusted the priors to see if they will improve the accuracy of the Bayesian Logistic Regression model; however, even after changing them, they still give approximately the same results. For example, when we used a $N(0, 10)$ prior, we got a training accuracy of 64.63% and test accuracy of 64.70%. We see that both models performed a lot better than the baseline accuracy 45.63% [1], confirming that our models did decently well at predicting the top 5 crime codes.

For our spatial model, after fitting the Bernoulli model, we got the probabilities of crime in each region. From the model, area 12 and 18 had the highest probability of crime of 10% and 8%, respectively. The regions with the lowest probability of crime are 8 and 16 with probabilities of about 3%. Similarly, the Poisson model shows that areas 12 and 18 also have the highest mean number of crimes per week with 1215 and 1029 crimes per week, respectively. The regions 8 and 17 contain the lowest mean number of crimes of 308 and 371, respectively, which differs slightly from the Bernoulli model. In order to compare the two models, we used PSIS LOO under the Loo package in R to compare eploo values.

	Estimate <S3: AsIs>	SE <S3: AsIs>
elpd_loo	-7909.5	129.2
p_loo	21.2	0.5
looic	15819.0	258.3

Figure 9: elpd_loo Values of the Bernoulli Model

	Estimate <S3: AsIs>	SE <S3: AsIs>
elpd_loo	-14745.9	1701.1
p_loo	273.8	38.5
looic	29491.8	3402.2

Figure 10: elpd_loo Values of the Poisson Model

The elpd_loo estimate for the Bernoulli model is -7909.5 and -14745.9 for the Poisson model. The estimates are reliable since the Bernoulli model had all k diagnostic values under 0.5, while the Poisson model had 98.1% of the k diagnostic values under 0.5. Therefore when comparing the eploo values, since the Bernoulli model had the smaller LOO-IC value and a lower standard error, the Bernoulli model better models the crime rate across the regions.

To generate the hotspot map, using the processed data from the Naive-Bayes prediction model, we select the median of the latitude and longitude of the whole dataset that was grouped by Area.ID and Area.Name, bind this dataframe with the output dataframe from our spatial hierarchical Bernoulli and Poisson models and map each area code/region with the column means of the posterior draws. Then, we piped in the dataframe to the ggmap function from the ggmap library to generate the roadmap and added features using stat summary 2d, geom_text and scale gradient fill to provide labels and gradient. The heatmaps of both models' results are shown below.

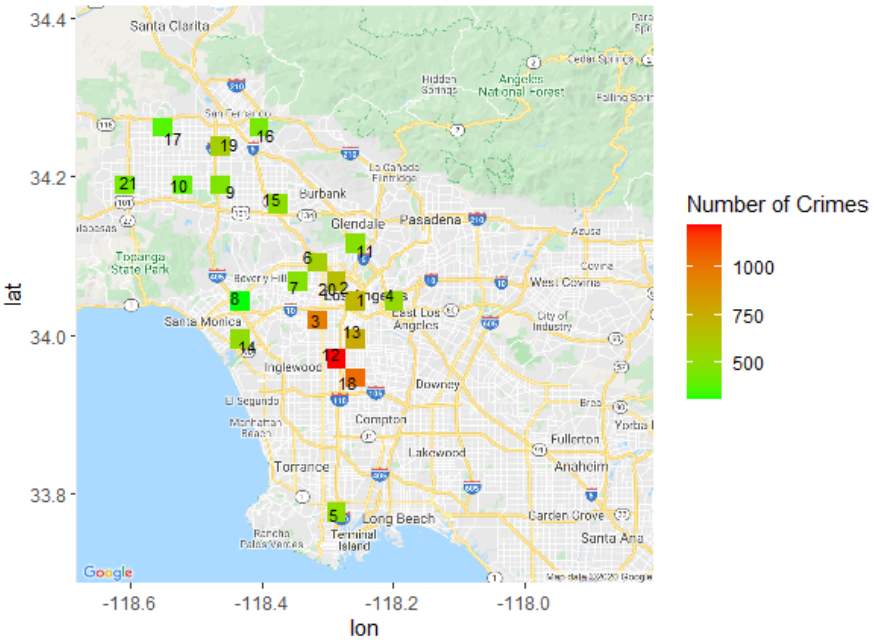

Figure 11: Poisson Hierarchical Model Heatmap

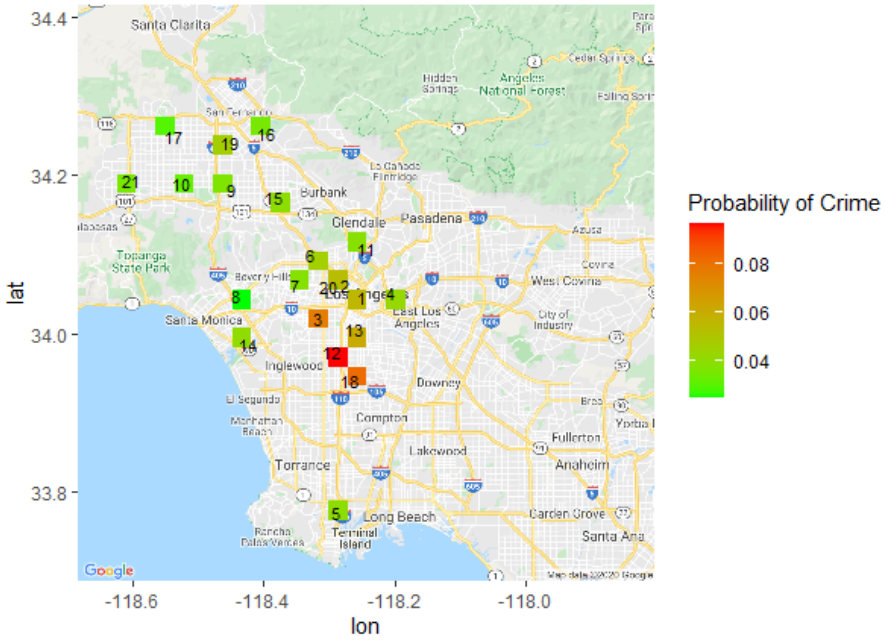

Figure 12: Bernoulli Hierarchical Model Heatmap

As we can see, both the Poisson and Bernoulli heatmaps produce exactly the same plots. The Poisson posterior samples are the number of crimes committed in each region, while the posterior samples of the Bernoulli model is the probability of crime in each region. Thus, the posterior responses of both models are highly correlated. Furthermore, region 12 has the highest number of crimes and probability of crime across all regions based on the posterior draws.

VI Summary/Discussion

In this project, we built a Bayesian Multinomial Logistic Regression model to predict the top 5 crimes (robbery, intimate partner-simple assault, battery, criminal threats and assault with deadly weapon) given the date of crime, time of crime, area of crime, victim age, victim sex, victim descent, weapons used, status of the case. We also fit a Naive-Bayes model as our baseline model to compare the test prediction accuracy of the two models. Our models performed very well, with both test accuracy being above the baseline accuracy of 45.63% [1]. We believe that the results are significantly more accurate due to having less crime codes to predict and only keeping features with higher frequencies in the dataset.

The heat maps generated from the posterior samples of the Poisson and Bernoulli models are exactly the same due to the high correlation between the posterior responses. The plots show that region 12 has the highest number of crimes and probability of crime across all regions based on the posterior draws. While the Poisson and Bernoulli models contain very similar results, they differ slightly in predicting the exact crime rate of the regions with the lowest amount of crime. Since the Bernoulli model has lower elpd_loo values, the Bernoulli model is better fitted to find the exact value of each region's crime rate.

VII Contributions

All group members worked together to design and implement the project and write the report. While everyone contributed to every part, group members focused on different parts. Wilmer took the lead in researching methods to implement in the project, finding relevant research papers to reference, and visualizing the spatial model hotspots. Veronica worked on cleaning and processing the dataset and building the prediction models. Matthew focused on coding and executing the spatial modeling part of the project.

References

[1] Dr Sarvanaguru RA Alkesh Bharati. "Crime Prediction and Analysis Using Machine Learning". In: 5.9 (Sept. 2018), pp. 1037–1042. URL: https://irjet.net/archives/V5/i9/IRJET-V5I9192.pdf.

[2] *Configure Visualization*. URL: https://data.lacity.org/d/63jg-8b9z/visualization. (accessed: 11.23.2020).

[3] Tao Hu et al. "Urban crime prediction based on spatio-temporal Bayesian model". In: *PLOS ONE* 13.10 (Oct. 2018), pp. 1–18. DOI: 10.1371/journal.pone.0206215. URL: https://doi.org/10.1371/journal.pone.0206215.

[4] Matthew Quick Jane Law and Afraaz Jadavji. "A Bayesian spatial shared component model for identifying crime-general and crime-specific hotspots". In: *Annals of GIS* 26.1 (2020), pp. 65–78. DOI: 10.1080/19475683.2020.1720290. eprint: https://doi.org/10.1080/19475683.2020.1720290. URL: https://doi.org/10.1080/19475683.2020.1720290.

[5] City of Los Angeles. *Los Angeles Crime & Arrest Data*. 2019. URL: https://www.kaggle.com/cityofLA/los-angeles-crime-arrest-data.

[6] S. Vinoth Kumar R. Rajadevi E. M. Roopa Devi. "Prediction of Crime Occurrence using Multinomial Logistic Regression". In: *International Journal of Innovative Technology and Exploring Engineering* 9 (3 2020). DOI: 10.35940/ijitee.B7663.019320.

[7] Lun Li Renjie Liao Xueyao Wang and Zengchang Qinh. "A Novel Serial Crime Prediction Model Based on Bayesian Learning Theory". In: *Proceedings of the Ninth International Conference on Machine Learning and Cybernetics,* (2010).

[8] Mehmet Sait Vural1 and Mustafa Gok. "Criminal prediction using Naive Bayes theory". In: *Neural Computing and Applications* 28.9 (2017). DOI: 10.1007/s00521-016-2205-z.

A Appendix

A.R Script Code

```
# Load libraries
library(dplyr)
library(lubridate)
library(tidyverse)
library(timeDate)
library(rstanarm)
library(brms)
library(ggplot2)
library(ggmap)
options(mc.cores = parallel::detectCores())

# Read csv file
crime = read.csv('crime-data-from-2010-to-present.csv')

# Clean/Process dataset
c = crime %>% select(-c(Crime.Code.1, Crime.Code.2, Crime.Code.3, Crime.Code.4, Cr
c = na.omit(c)
c$Area.ID <- as.factor(c$Area.ID)
c$Crime.Code <- as.factor(c$Crime.Code)
c$Weapon.Used.Code <- as.factor(c$Weapon.Used.Code)
c$Date.Occurred <- dayOfWeek(timeDate(c$Date.Occurred))
c$Date.Occurred <- as.factor(c$Date.Occurred)
c$Time.Occurred <- as.factor(trunc(c$Time.Occurred/100))

# Extract latitude and longitude
ll_lat <- function(x){
  x = toString(x)
  l = stringr::str_extract_all(x, "-*\\d*\\.\\d*")
  return(as.numeric(l[[1]][1]))
}

ll_long <- function(x){
  x = toString(x)
  l = stringr::str_extract_all(x, "-*\\d*\\.\\d*")
  return(as.numeric(l[[1]][2]))
}

c$lat = sapply(c$Location, ll_lat)
c$long = sapply(c$Location, ll_long)

# Dataset for prediction
data = c
```

```
data$Date.Occurred <- as.factor(as.numeric(factor(data$Date.Occurred)))
data$Victim.Sex <- as.factor(as.numeric(factor(data$Victim.Sex)))
data$Victim.Descent <- as.factor(as.numeric(factor(data$Victim.Descent)))
data$Status.Code <- as.factor(as.numeric(factor(data$Status.Code)))
data = data %>% select(-c(lat, long, Location))
set.seed(10)

top5 = as.data.frame(table(data$Crime.Code)) %>% filter(Freq > 1000) %>% arrange(de
top5 = as.vector(top5[,])
top_descent = as.data.frame(table(data$Victim.Descent)) %>% filter(Freq > 10000) %
top_descent = as.vector(top_descent[,])
top_weapons = as.data.frame(table(data$Weapon.Used.Code)) %>% filter(Freq > 10000)
top_weapons = as.vector(top_weapons[,])
top_status = as.data.frame(table(data$Status.Code)) %>% filter(Freq > 10) %>% selec
top_status = as.vector(top_status[,])
top5_crimes = data %>% filter(Crime.Code %in% top5, Victim.Descent %in% top_descent
top5_crimes <- droplevels(top5_crimes)

sample_size <- 10000
sample_id <- sample(1:nrow(top5_crimes), sample_size)
sample <- top5_crimes[sample_id, ]
sample <- droplevels(sample)

train_size <- floor(nrow(sample) * 0.7)
train_id <- sample(1:nrow(sample), train_size)
train <- sample[train_id, ]
train <- droplevels(train)
test <- sample[-train_id, ]
test <- droplevels(test)

# check if random sample is representative of dataset
#levels(top5_crimes$Date.Occurred) == levels(sample$Date.Occurred)
#levels(top5_crimes$Time.Occurred) == levels(sample$Time.Occurred)
#levels(top5_crimes$Area.ID) == levels(sample$Area.ID)
#levels(top5_crimes$Crime.Code) == levels(sample$Crime.Code)
#levels(top5_crimes$Victim.Sex) == levels(sample$Victim.Sex)
#levels(top5_crimes$Victim.Descent) == levels(sample$Victim.Descent)
#levels(top5_crimes$Weapon.Used.Code) == levels(sample$Weapon.Used.Code)
#levels(top5_crimes$Status.Code) == levels(sample$Status.Code)

# Naive-Bayes model
bayes_model = naiveBayes(Crime.Code ~ ., data=train)
bayes_ypreds_train = predict(bayes_model, train)
bayes_ypreds_test = predict(bayes_model, test)
```

```
bayes_train_accuracy = mean(bayes_ypreds_train == train$Crime.Code)
bayes_test_accuracy = mean(bayes_ypreds_test == test$Crime.Code)

# Bayesian Multinomial Logistic Regression model
prior = get_prior(Crime.Code ~ ., data = train, family = categorical())
b2 = suppressMessages(brm(Crime.Code ~ ., data = train, family=categorical(), prior
y_preds_train_2 = predict(b2, type = "response")
y_preds_test_2 = predict(b2, newdata = test, type = "response")

get_ypreds = function(ypreds) {
  names = stringr::str_extract_all(colnames(ypreds), "\\d+")
  colnames(ypreds) = names

  max_index <- function(i){
    return(which(ypreds[i,]==max(ypreds[i,]))[[1]])
  }

  idx = vector()
  for(i in 1:nrow(ypreds)){
    idx = append(idx, max_index(i))
  }

  name <- function(i){
    return(colnames(ypreds)[i])
  }

  ypreds_vec = vector()
  for(j in idx){
    ypreds_vec = append(ypreds_vec, name(j))
  }
  return(ypreds_vec)
}

logistic_ypreds_train_2 = as.vector(as.factor(get_ypreds(y_preds_train_2)))
log_train_accuracy = mean(logistic_ypreds_train_2 == train$Crime.Code)

logistic_ypreds_test_2 = as.vector(as.factor(get_ypreds(y_preds_test_2)))
log_test_accuracy = mean(logistic_ypreds_test_2 == test$Crime.Code)

results <- data.frame("Training Accuracy"=c(log_train_accuracy, bayes_train_accurac
rownames(results) <-c("Bayesian Logistic Regression", "Naive Bayes")

# Spatial model - Poisson dataset
c$Date.Occurred = as.date(c$Date.Occurred)
pois = c %>% group_by(week = week(Date.Occurred), Area.ID) %>% count(Crime.Code) %
```

```
pois_data = pois %>% spread(Area.ID, num)
pois_data = subset(pois_data, select=c(-1)) # Remove first column which contains w

# Spatial model - Poisson
model_filename = 'stan_model(1).stan'
data <- list(N=nrow(pois_data), M=ncol(pois_data), y=pois_data)
stan_model = rstan::stan_model(model_filename)
stan_samples_pois <- rstan::sampling(stan_model, data=data, iter=3000)

# Spatial model - Poisson loo values
heir_log_lik <- extract_log_lik(stan_samples_pois, merge_chains = FALSE)
heir_r_eff <- relative_eff(exp(heir_log_lik), cores = 2)
heir_loo <- loo(heir_log_lik, r_eff = heir_r_eff, cores = 2)

# Spatial model - Bernoulli dataset
spacial_df <- fastDummies::dummy_columns(c["Area.ID"])
spacial_df <- subset(spacial_df, select=-c(Area.ID))
spacial_df <- sample_n(spacial_df, 2000)

# Spatial model - Bernoulli
model_filename = 'stan_model.stan'
data <- list(N=nrow(spacial_df), M=ncol(spacial_df), y=spacial_df)
stan_model = rstan::stan_model(model_filename)
stan_samples_bern <- rstan::sampling(stan_model, data=data, iter=3000)

# Spatial model - Poisson loo values
heir_log_lik <- extract_log_lik(stan_samples_bern, merge_chains = FALSE)
heir_r_eff <- relative_eff(exp(heir_log_lik), cores = 2)
heir_loo <- loo(heir_log_lik, r_eff = heir_r_eff, cores = 2)

# Generating Crime Hotspots based on Region
map_c = c
map_c = map_c %>% select(Area.ID, Area.Name, lat, long)
map_c = map_c %>% group_by(Area.ID, Area.Name) %>% summarize(lat = median(lat), lon
pois_df = as.data.frame(colMeans(as.data.frame(stan_samples_pois))) %>% filter(row_
pois_df = pois_df[,1]
ber_df = as.data.frame(colMeans(as.data.frame(stan_samples))) %>% filter(row_numbe
ber_df = ber_df[,1]
new_df = cbind(map_c, pois_df, ber_df) %>% mutate(pois_crimes = ...5, ber_crimes =
new_df = new_df %>% select(-Area.Name)
register_google(key = "AIzaSyDrcOYJUg9XUHUiR-Uhsdr-YGkmqD-sSuM", write = TRUE)
map <- get_map(location = "Los_Angeles", maptype = "roadmap")
#Poisson model heatmap
ggmap(map) + stat_summary_2d(data = new_df, aes(x = long, y = lat, z = pois_crimes)
  geom_text(data = new_df, aes(x = long, y = lat, label = Area.ID), size = 3) +
```

```
  scale_fill_gradient(name = "Number of Crimes", low = "green", high = "red")
##Bernoulli model heatmap
ggmap(map) + stat_summary_2d(data = new_df, aes(x = long, y = lat, z = ber_df), fun
  geom_text(data = new_df, aes(x = long, y = lat, label = Area.ID), size = 3) +
  scale_fill_gradient(name = "Probability of Crime", low = "green", high = "red")
```