Crimes in Boston Prediction with Big Data

Wei Jiang*
Department of Computer Science
Boston University
Boston, MA 02215
weijiang@bu.edu

Ci Chu
Department of Computer Science
Boston University
Boston, MA 02215
chuci@bu.edu

Kaikang Zhu
Department of Computer Science
Boston University
Boston, MA 02215
zhukk@bu.edu

Abstract

Crime, as one of the most significant and serious problems in United States, has great influences on public personal safety, economic development and children growth. Therefore, understanding what are key factors for crimes and how to alleviate are critical both to law enforcement authorities and policy makers. In our paper, we pay great attention to the crime data in Boston from 2015 to 2018 and hope to make predictions based on various features and prevent specific crime cases to some extent. Different from traditional approaches, which use demographics and geographical effects to estimate, we utilize various machine learning methods to make predictions and measure model performance based upon testing datasets. From our result analysis, we observe that the accuracy and loss for our model has a significant improvement and could become reference for further researches.

1 Introduction

Crimes, which are a part of common social problems, have already affected society safety and economic growth seriously for centuries [1]. Crime rate has been considered as an essential factor that determines whether certain city or region is suitable to live or should be avoided. Considering the continuity of crime cases, law enforcement authorities are demanding reliable and effective geographical information system and new data analysis methods to better prevent certain crimes and protect the communities [2].

Although crimes could occur everywhere, it is common that criminals tend to choose places or areas that are familiar to them, therefore, it is reasonable for us to analyze crime pattern and determine the criminal hotspots by using big data and data mining approaches [3]. In addition, by analyzing the crime time, location and type, we hope to raise public awareness to avoid certain dangerous regions and specific time slots. Besides, through data analysis and measurement, people could have better options when facing new living place choices. For police departments, they could better utilize this solution to increase the level of crime prediction and prevention and distribute police force at certain time and place more reasonably.

In this research, we implemented Regression, Classification and Neural Network algorithms on features set which contained crime type, time, location, etc. Here, 80% of data will be trained

*Github: https://github.com/yizheshexin/cs542_final_project.

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

according to the given algorithms and tested on the rest of 20% data. The aim is to predict top most crime cases with the accurate predictive model and help police or law enforcement makers take necessary actions.

The rest of the paper is organized as follows: Section 2 gives an overview of data cleaning and pre-processing procedure, it also uses several visualization figures to help analyze key features and understand geographical distribution. Section 3 introduces information about certain algorithm and give detailed analysis on neural network researches. Section 4 provides common measurement metrics and compare models' performance, while Section 5 concludes with the findings and provides future work consideration.

2 Data Preparation

2.1 Data Processing

We pick several features as our dataset. Considering some of the features are categorical data, we decide to map them into integer so that we can train them. For example, we can map A to 0, B to 1 and so on. Then we apply one-hot encoding to mapped data. One hot encoding is a process by which categorical variables are converted into a form that could be provided to ML algorithms to do a better job in prediction. It can make data sparser and raise the training accuracy. Finally, to reduce the difference among different features, we apply normalization to the longitude and latitude.

2.2 Data Analysis

Based on processed dataset, we pick up several key features to make corresponding charts and analyze crime category, trend of time and geographical distribution. From different figures, we could better understand the head components and crime tendency.

2.2.1 Crime Category

Offenses were grouped into eight categories: person, sex, weapons, property, drug, noncompliance, status, and "other". Data from detention and correctional centers use their own hierarchical system to determine the most serious offense at the time of admission, and identify offense category only by one offense. Arrest data contain information about multiple charges. The Authority has developed a method of classifying and organizing arrest incidents to determine and classify arrests by the most serious charge in an incident should there be multiple charges. To maintain consistency, the same offense category classification system was used for arrest, detention, and corrections data.

For our dataset, we sort the crime group in descending order and make histograms in Figure 1 and 2.

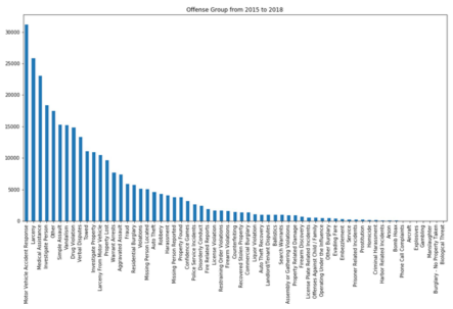

Figure 1: Offense group bar chart

Motor Vehicle Accident Response	31186
Larceny	25808
Medical Assistance	23041
Investigate Person	18379
Other	17508
Simple Assault	15264
Vandalism	15234
Drug Violation	14829
Verbal Disputes	13321
Towed	11061
Investigate Property	10893
Larceny From Motor Vehicle	10490
Property Lost	9629
Warrant Arrests	7682
Aggravated Assault	7406
Fraud	5881
Residential Burglary	5696
Violations	5103
Missing Person Located	5041
Auto Theft	4598

Figure 2: Offense group head category

2.2.2 Trend of time

For the different time criminals took, we analyze its distribution from the aspect of hour, month and year. In addition, we hope to find the pattern for different time slots and determine whether specific time should be avoided or not. In Figure 3 and 4, we provide the hour and month distribution for crime dataset.

Figure 3: Happening hour bar chart

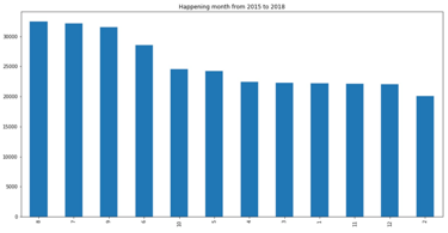

Figure 4: Happening month histogram

2.2.3 Geographical distribution

Most existing work adopts a place-centric paradigm, where the research question is to predict the location of crime incidents [4]. And the predicted location is usually referred by the term hotspot, which has various geographical size. In order to analyze street and region division, we use K Means Clustering technique and folium to display clusters situation dynamically in Figure 5 and 6.

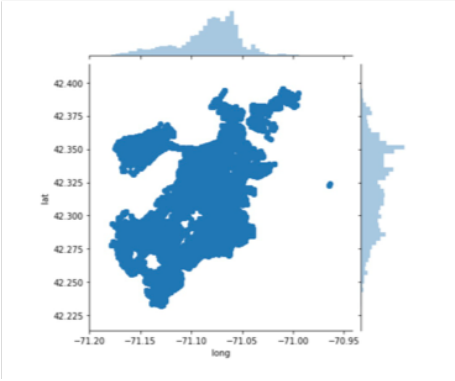

Figure 5: Crime spotting area geographical distribution

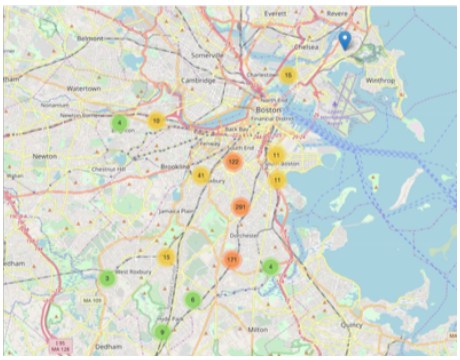

Figure 6: Shooting area interaction page

3 Models

3.1 Regression Model

3.1.1 Linear Regression

Our algorithm uses linear regression for prediction and use weighted instances to balance the difference between models. This method of regression is simple and easy to provide an interpretable and complete description of how input data will affect output. It models a variable Y as a linear function of vector variable X; Given n samples of data points in multiple dimensions, linear regression could be expressed as $Y = \alpha + \beta X$, where α and β are regression coefficients. We assume variance of Y is a constant, therefore, the coefficients could be solved using the least squares method and gradient descent approach, which will minimize the error between actual data point and regression line.

3.1.2 Logistic Regression

We also apply logistics regression to predict the crime type. Though logistic regression is more suitable to model a binary dependent variable, we want to use this to model a more complex variable which has 20 classes. Logistic regression uses a different loss function from linear regression. We decide to split the train and test dataset into a 9:1 ratio. And we apply normalization to the data and it proves that it increased accuracy by approximately 1%. And finally, logistics regression achieves a reasonable result around 20% which improves a lot comparing to guess a crime type randomly (5%).

3.2 Classification Model

3.2.1 Naïve Bayesian Classification

Naïve Bayesian Classification is a supervised learning algorithm, which has been widely used for prediction. It is a statistical model which could predict probabilities based on Bayes Theorem that assumes independency between attributes. As our crime features have the independent property between each other, this classification would be an ideal choice for us.

We constructed this model using Scikit–Learn that provides a set of open source data-mining tools for Python. We applied Gaussian Naïve Bayes and Bernoulli Naïve Bayes Classification, which conforms to the categorical features in our datasets. The crime features contain reporting area, shooting, time and location of the crime while we selected the crime type to represent the class label. We randomly divided the dataset into 80% of data as a training set and 20% of data as a testing set. We trained two classifiers on the training data for our datasets to obtain two different models ready for crime type prediction.

3.2.2 Decision Tree Classification

Another classification algorithm is decision tree. Decision tree is the most powerful and popular tool for classification and prediction. A Decision tree is a flowchart like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label. It's a good choice for us to model the crime type. We apply the same data process technique to the data and we get the final accuracy 18.0% which is a little lower than logistic regression.

3.2.3 Random Forest Classification

Random forest algorithm is a supervised learning algorithm. Random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction. It can be used for both classification and regression problem. We divide the dataset into 90% training set and the other 10% set for testing. We get a 17.9% accuracy as a result which is no better than decision tree classification. We think it may be related to that random forest classifier randomly choose features as a set to build a decision tree and the random features isn't a suitable combination.

3.3 Neural Network

Apart from the above models we have already used, we put our focus on neural network research and improvement. We hope that neural network's non-linear transformation will help us to find the relation among Boston's crime. In this part, we mainly try two types of neural network: multilayer perceptron and recurrent neural network.

3.3.1 Multilayer Perceptron

Multilayer perceptron (MLP) belongs to the class of feedforward artificial neural network. An MLP consists of at least three layers of nodes: an input layer, a hidden layer and an output layer. Except for the input nodes, each node is a neuron that uses a nonlinear activation function. MLP utilizes a supervised learning technique called backpropagation for training. Its multiple layers and non-linear activation distinguish MLP from a linear perceptron, while it can distinguish data that is not linearly separable. We will introduce the input data, hidden layer and output data of our MLP separately.

We choose these data as our input dimension: 'Year', 'Month', 'Day', 'Hour', 'District': 'Year', 'Month', 'Day' and 'Hour' are time information. 'District' is location information. One thing that is worthy to discuss is that in the original data, we have many columns that is related to location information: 'District', 'Reporting Area', 'Street', 'Latitude', 'Longitude'. Obviously, 'Latitude' and 'Longitude' are the most accurate location that we can get since each pair of 'Latitude' and 'Longitude' represents a precise location on the earth. Compared to them, other features only represent an area of location. By first intuition, we may use 'Latitude' and 'Longitude' as the input feature since we would like the data to be as accurate as possible. But after thinking twice, we finally decide to use 'District' as location input feature. The reason is that it's ideal to use the most accurate data as input only if our neural network can do a very precise prediction. In such cases, a subtle change in 'Latitude' and 'Longitude' may have a great influence on the result. But if our neural network is not accurate enough, it's better for us to use an ambiguous input to do a more general prediction rather than a specific prediction. And to use 'District', since it is a non-continuous value, we have to transfer it to one-hot code. Thus, our final input data is 16 dimensions: 'YEAR', 'MONTH', 'DAY', 'HOUR', 'A1', 'A15', 'A7', 'B2', 'B3', 'C11', 'C6', 'D14', 'D4', 'E13', 'E18', 'E5'.

We use keras.layers.Dense() to build our hidden layer. To find a best performance network structure, we try lots of combinations of various number of dense layers and different units in each dense layer. For number of dense layers, we use: [2, 3, 4, 5, 6, 7, 8, 9]. For number of units in dense layer, we try: [64, 128, 256, 512, 1000, 1024, 1250, 1500]. And for activation function, we choose Rectified Linear Unit.

For the output layer, since we want the result to be a probability distribution, we choose to use Softmax as activation function. Besides, there are 67 units in the output layers, which is the number of Boston crime's total categories.

3.3.2 Recurrent Neural Network

Recurrent neural network (RNN) is a kind of artificial neural network where connections between nodes form a directed graph along temporal sequences, which allows it to exhibit temporal dynamic behavior. Unlike feedforward neural networks, RNNs can use their internal memory to process input sequences, which makes it suitable to use previous input data to predict future's data.

Long short-term memory (LSTM) is a kind of recurrent neural network. A common LSTM unit is composed of a cell, an input gate, an output gate and a forget gate. The cell remembers values over arbitrary time intervals and the three gates regulate the flow of information into and out of the cell, which makes it well-suited to do classifying, processing and making predictions based on time series data. The reason what we want to try RNN is that our dataset can provide time information for each crime, which means that we can get a time sequence, and in each time point, we can know how many crimes happened.

Before building the RNN, we need to do some data processing at first. First of all, we decide to filter the data. In addition, there are several ways to filter the data: 1. Use the whole dataset. 2. Use data where 'DISTRICT' column == 'B2'. 3. Use data where 'DISTRICT' column == 'B2' and 'OFFENSE_CODE_GROUP' == 'Motor Vehicle Accident Response'. The reason that we care about 'B2' and 'Motor Vehicle Accident Response' is that B2 district is the district that has the highest number of crimes over the years and 'Motor Vehicle Accident Response' is the crimes that happen most frequently over the years in B2 district. Obviously, what we are trying to do will influence the size of our training and testing dataset. Our whole dataset has 327820 records of crimes, B2 district has 51288 records of crimes, and there are 6590 records of 'Motor Vehicle Accident Response' crimes in B2 district. By intuition, with more data, it's more likely to get a higher accuracy; with less data, but a more accurate filter, it's more likely to get a specific prediction.

In fact, we tried above 3 ways to filter data. The third way get an accuracy of more than 80%. At first, we were glad to see such high accuracy, but when we check the data again, we realized that since the dataset size is so small that after we converted the data to time sequence format, more than 80% of the data is 0, which means that even if the RNN's output is all 0, it can still get an accuracy of more than 80%, and obviously such result is useless. The first way with the largest dataset doesn't have such problem, but the highest accuracy it can get is still around 14%. And only the second way make some progress on accuracy, so we will describe the second way's result in detail.

After filtering the data, we need to do some transformation. We need to convert the 'OCCURRED_ON_DATE' column, which is in date format, to Unix timestamp. We would firstly make the minute and second to 0 and then do the transformation. This would make each crime record belongs to a time period which is an hour long. For example, to convert '2018-10-03 15:51:00', we firstly set it to '2018-10-03 15:00:00', then convert it to '1538578800', which represents that in the time period from '2018-10-03 15:00:00' to '2018-10-03 16:00:00', there is a crime happened. Then we try to generate the complete time sequence. The first crime in our dataset is happened at 1434326400 (Unix timestamp), and the last crime is happened at 1538596800. We can set step to be 3600 (seconds) to generate all the time point between 1434326400 and 1538596800. Finally, we combine above two part's result together, getting a dictionary which tells us in each time period that how many crimes happened. In Figure 7, it means than at the time period from 1434326400 to 1434330000, there were 14 crimes happened, at the time period from 1434330000 to 1434333600, there were 10 crimes happened.

Our final transformation is to decide the input data's dimension, which is in fact to decide the sequence length of input data. In Figure 8, We set sequence length to be 72, which means that we want to use the previous 72 hours (3 days) crime number to predict next hour's crime number.

```
Counter({1434326400: 14,
         1434330000: 10,
         1434333600: 7,
         1434337200: 3,
         1434340800: 1,
         1434344400: 1,
         1434348000: 2,
         1434351600: 8,
```

Figure 7: Time sequence transformation

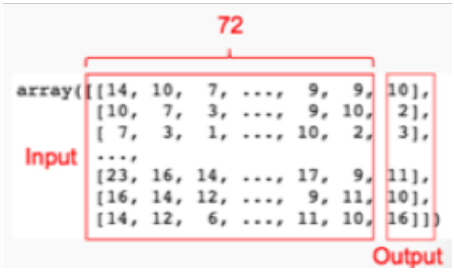

Figure 8: Input data dimension

Now, we can start to build our RNN. The input data is 72-dimension data. For hidden layer, we try [2, 3, 4] for the number of LSTM layer, [64, 128, 256] for the number of units in each LSTM layer. Each LSTM layer is followed by a dropout layer to reduce overfitting. The last LSTM layer is followed by a dense layer with 128 units and Rectified Liner Unit as activation function. Then it is followed by a dense layer with 1 unit and no activation function, which is our output data, representing the predicted number of crimes.

4 Evaluation

4.1 Evaluation of Regression model and Classification model

After implementation of above algorithms, we output several metrics that evaluate the effectiveness and efficiency of the algorithms: correlation coefficient, accuracy, precision, recall and F1 score. The results for these metrics will be used in the comparative evaluation of the crime statistics. The objective of this research is to present how effective the algorithms can be in determining patterns of criminal activities.

- Correlation Coefficient: The correlation coefficient measures the strength of association between two variables. The larger the absolute value of the correlation coefficient, the stronger is the relationship between the variables. If the correlation is positive, it means that as one variable becomes larger, the other variable tends to become larger. The result of correlation coefficient for linear regression is shown in Figure 9.
- Accuracy, Precision, Recall and F1 Score: Performance of a model is measured by reflection of well observed actual events. While training any model, a labelled data set that includes the actual values to be predicted is considered. This introduced the concepts of a confusion matrix [5], which gives a relation between an actual and predicted class. Accuracy is simple the ratio of correctly predicted observations, precision is the ratio of correctly predicted positive observations to the total predicted positive observations. In addition, recall means

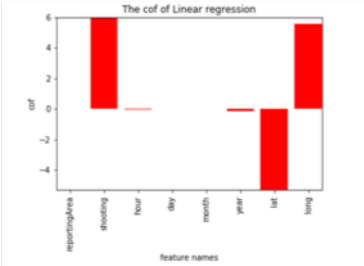

Figure 9: Correlation coefficient of linear regression

	Logistic Regression	Gaussian NB	Bernoulli NB	Decision Tree	Random Forest
Accuracy	0.199	0.118	0.121	0.180	0.179
Precision	0.188	0.084	0.035	0.167	0.169
Recall	0.197	0.118	0.121	0.173	0.181
F1 score	0.153	0.050	0.029	0.169	0.168

Table 1: Metric table for models

the ratio of correctly predicted positive observations to the all observations in actual class, and F1 score is the weighted average of precision and recall value. The final result for regression model and classification model is displayed in Table 1.

4.2 Evaluation of Neural Network model

After using neural network, we run 20 epochs for each network structure. As is shown in Figure 10 and 11, the best accuracy we can get is 14.55%, with 3 layers and 1024 units in each layer. However, 14.55% is definitely not a good accuracy for neural network. And it seems that no matter how we change the number of dense layer and units, it is hard to make a big progress using MLP.

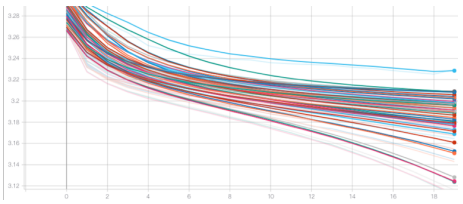

Figure 10: MLP loss curve

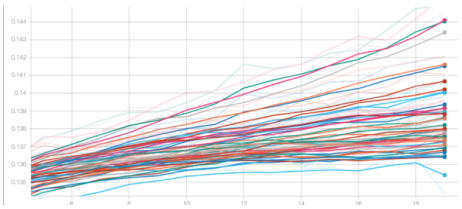

Figure 11: MLP accuracy curve

From Figure 12 and 13, the best accuracy we can get is 28.35% with 2 LSTM layers and 128 units in each layer. Although we still cannot say that this is a good accuracy for prediction, but at least we make some progress from the previous work.

5 Conclusions and Future Work

We generated many graphs and found interesting statistics that showed the baseline to understand Boston crimes datasets. Then, we applied several Regression, Classification and Neural Network models to help predicting future crimes in a specific location within a particular time. We achieved 28% of prediction accuracy in existing dataset. We aimed to further understand our models' findings and to capture the factors that might affect the safety of neighborhoods.

There are still lots of work we can do to improve the performance of the network. To do a more accurate prediction, we need more data sources. Now we are only using the history crime records to do prediction, however, there could be many factors that may influence the crime in Boston, such as population, economy, income and so on. We may combine these data source together to do a better job in the future. What's more, we may adjust the network structure. So far, we only try MLP and

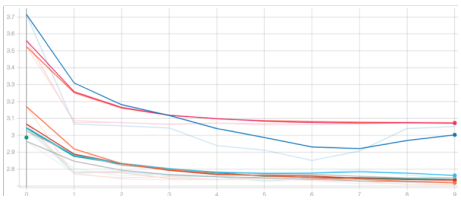

Figure 12: RNN loss curve

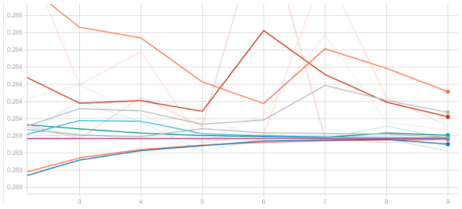

Figure 13: RNN accuracy curve

RNN structure, but there are still many different types of neural networks that may have a better performance over our dataset.

Last but not the least, we should carefully choose what we want to predict. In the above work, when we build the MLP, we want the output to be the probability of whether a kind of crime would happen. Note that this is a very precise prediction, it can tell us exactly which kind of crime is most likely to happen given a time and an area (or even a longitude and latitude), but in fact, due to many limits, the accuracy is only 14%. Then when using RNN to do prediction, we only care about how many crimes would happen and no longer care about what is the exact category of the crime. It seems that by giving up some information, we get a higher accuracy. Therefore, we may choose to compromise in some aspects to gain some benefits in other aspects to finally get a global optimization solution.

References

[1] A. Bogomolov, B. Lepri, J. Staiano, N. Oliver, F. Pianesi and A. Pentland, 'Once Upon a Crime: Towards Crime Prediction from Demographics and Mobile Data', CoRR, vol. 14092983, 2014.

[2] A. Buczak and C. Gifford, 'Fuzzy association rule mining for community crime pattern discovery', in ACM SIGKDD Workshop on Intelligence and Security Informatics, Washington, D.C., 2010, pp. 1- 10.

[3] S. Nath, 'Crime Pattern Detection Using Data Mining', in Web Intelligence and Intelligent Agent Technology Workshops, 2006. WI-IAT 2006 Workshops. 2006 IEEE/WIC/ACM International Conference on, 2006, pp. 41,44.

[4] TOOLE, J. L., EAGLE, N., AND PLOTKIN, J. B. Spatiotemporal correlations in criminal offense records. ACM Transactions on Intelligent Systems and Technology (TIST) 2, 4 (2011), 38.

[5] Pang-Ning Tan, Michael Steinbach, Vipin Kumar, "Introduction to Data Mining", Pearson Addison Wesley, 2006, Page 187,227, 296, 297, 729.

[6] Wang, Hongjian & Kifer, Daniel & Graif, Corina & Li, Zhenhui. (2016). Crime Rate Inference with Big Data. 10.1145/2939672.2939736.

[7] GB/T 7714Lin Y L, Chen T Y, Yu L C. Using Machine Learning to Assist Crime Prevention[C]// Iiai International Congress on Advanced Applied Informatics. 2017.

[8] Tariku D . DEVELOPING A PREDICTIVE MODEL FOR FERTILITY PREFERENCE OF WOMEN OF REPRODUCTIVE AGE USING DATA MINING TECHNIQUES[J]. 2013.

[9] Almanie T , Mirza R , Lor E . Crime Prediction Based On Crime Types And Using Spatial And Temporal Criminal Hotspots[J]. Computer Science, 2015, 5(4).