Crime Data Mining Final Product

Brandon Sturgis
Department of Computer Science
Virginia Tech
Blacksburg, Virginia, USA
bdsturgis@vt.edu

Na Le
Department of Computer Science
Virginia Tech
Falls Church, Virginia, USA
nale@vt.edu

Uditi Goyal
Department of Computer Science
Virginia Tech
Blacksburg, Virginia, USA
ud2607@vt.edu

Atsushi Ikeda
Department of Computer Science
Virginia Tech
Falls Church, Virginia, USA
ash208@vt.edu

Product Description

With millions of crime offenses recorded each year in the United States, safety and security are our top concern when deciding where to live. To help increase the community's confidence in public safety, our team proposed a product that can help people view the crime history as well as predict future times and places for crimes in the Washington D.C. area. This product is designed for a typical worker, a real estate agent, a regular student, and a researcher or an analyst that are interested in exploring the likelihood of crime occurring in the area to ensure safety. The product provides users with analysis on the number of records in yearly cases or in each ward in the metropolitan area, analysis on how different machine learning models can effectively classify and predict the offense group, as well as a time series analysis in each census tract and thus, can predict the crime trend in the same census tract in the upcoming month.

Product Functionalities

A link to a demo of our product can be found on VT GitLab.

Yearly Case Count Analysis

In this analysis, we provide a visualization of the yearly crime trends. We allow users to check the number of crimes per year in the D.C. area. Users can also check the number of crimes by type of crime in each year by clicking on the bar representing the year. Additionally, by activating the animation, users can instantly understand the trend of the number of crimes in each year.

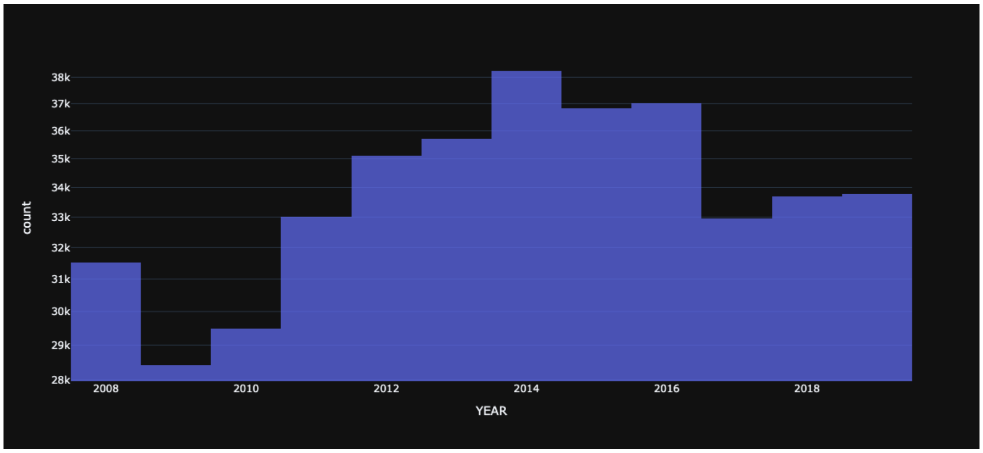


Figure 1: Yearly Case Count

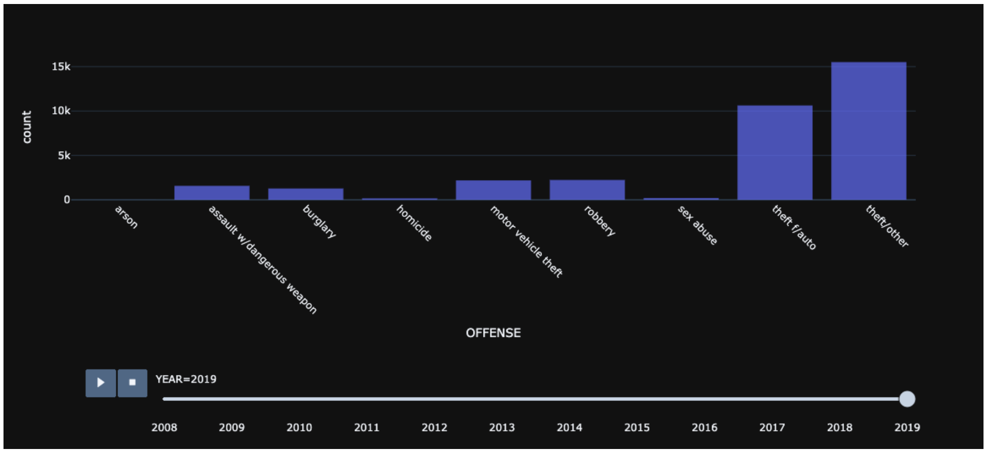


Figure 2: Each Year Case Count

Ward Case Count Analysis

In this analysis, we provide a visualization of the number of crimes in each ward. We allow users to investigate the number of crimes in each ward by hovering their mouse. Users can also see the number of crimes by type of crime in each ward. Like the yearly case count analysis, users can investigate the number of crimes in each ward interactively by clicking on the bar representing the ward.

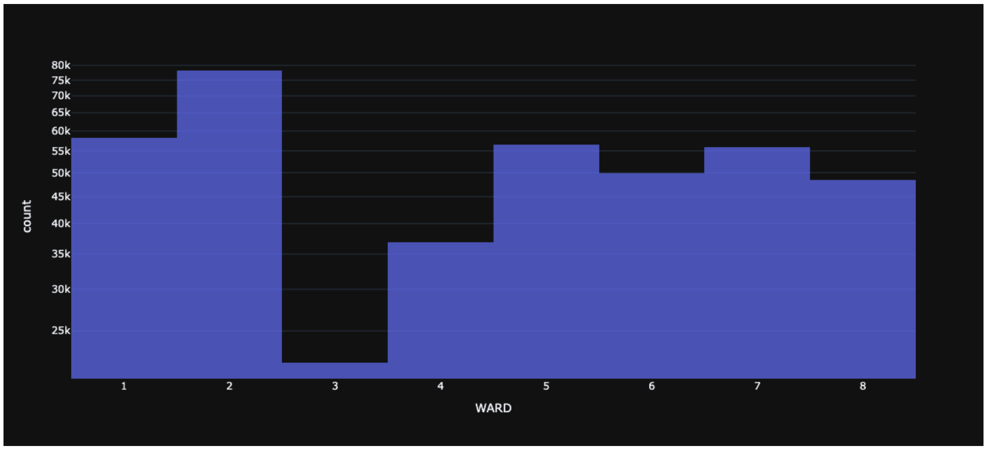


Figure 3: Ward Case Count

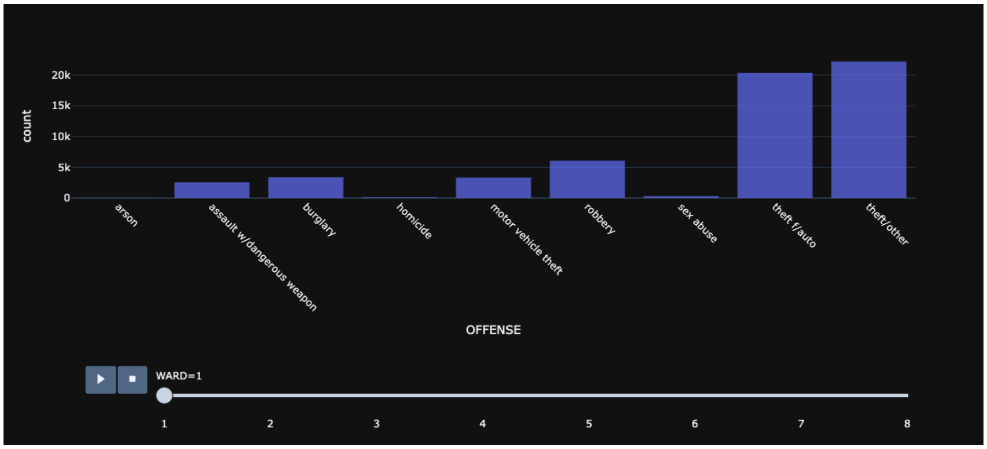


Figure 4: Each Ward Case Count

Hourly Case Proportion Analysis

In this analysis, we display the hourly historic crime rate in the ward that users want to investigate. We provide the hourly incidence of all crimes committed in the overall D.C. area. Additionally, it is possible for the user to choose the ward and hour when they want to check which crimes occur frequently in a specific ward and hour. For example, from this graph, users can understand that they need to be careful about burglaries from 4 am to 8 am in ward 1, because the rate of burglary occurrence is higher from 4 am to 8 am than in other time zones in ward 1.

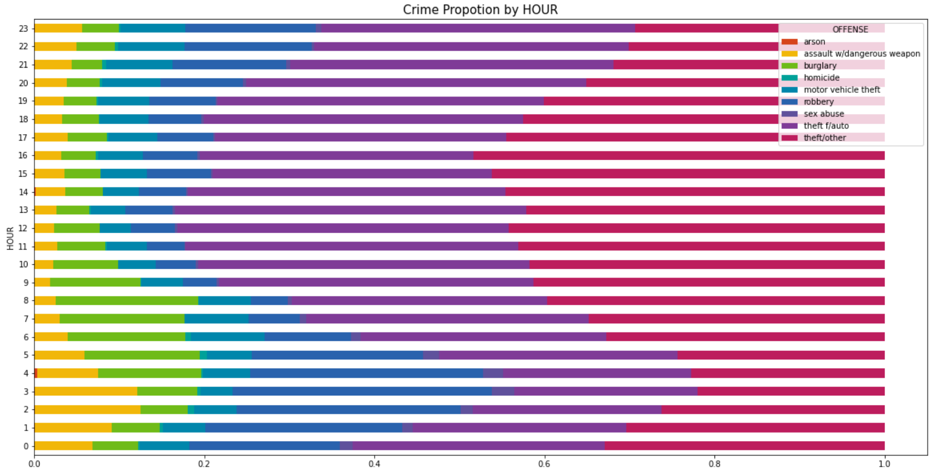


Figure 5: Hourly Case Proportion

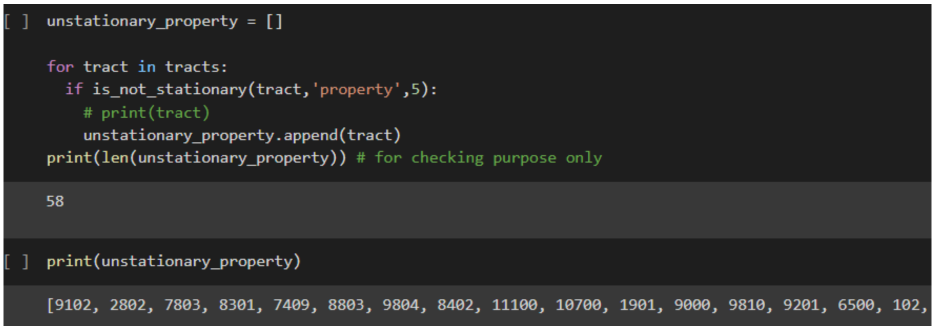
Machine Learning

Our team also trained and tested the crime dataset on 5 machine learning models using 5-fold cross validation. The models we used to classify and predict the property and violent group include: Decision Tree, K-Nearest Neighbors, Random Forest, AdaBoost and Multi-layer Perceptron. In each model, we recorded various metrics used for evaluating the performance. Of all the models, we found that the random forest classifier had the best accuracy in predicting crime groups. Our reason for using machine learning was to understand which features have the most significant impact on types of crime. While we can get a basic understanding by viewing graphs like the crime proportion by hour, the reality is the type of crime depends on multiple features, and this pattern can best be learned by machine learning.

Time Series Analysis

We allow the users to check the stationary of the crime rate in a census tract. If no stationary is found, it means that the crime rate in this area has a tendency to either increase or decrease. Otherwise, there is no potential trend in the crime rate in the area.

For example, in the figure below, we check the stationary of property-group crime rate in all census tracts to determine which tract has a potential trend in crime rate increasing/decreasing. The output shows that 58 tracts were found in this criteria.



```
[ ]  unstationary_property = []

     for tract in tracts:
       if is_not_stationary(tract,'property',5):
         # print(tract)
         unstationary_property.append(tract)
     print(len(unstationary_property)) # for checking purpose only

     58

[ ]  print(unstationary_property)

     [9102, 2802, 7803, 8301, 7409, 8803, 9804, 8402, 11100, 10700, 1901, 9000, 9810, 9201, 6500, 102,
```

Figure 6: Checking non-stationary census tract with property-related offense

Similarly, we find that 24 census tracts were reported as non-stationary.

In the product, we also provide the users with offense prediction in both property and violent types with the plot of the original data in the visualization. The users can input a census tract number and a desired date that they want to predict the crime trend in the area. In future development, we hope to extend the choice of the user by letting he/she specify a full address instead of a census tract so he/she can narrow down the area and learn the crime trend in this specific neighborhood.

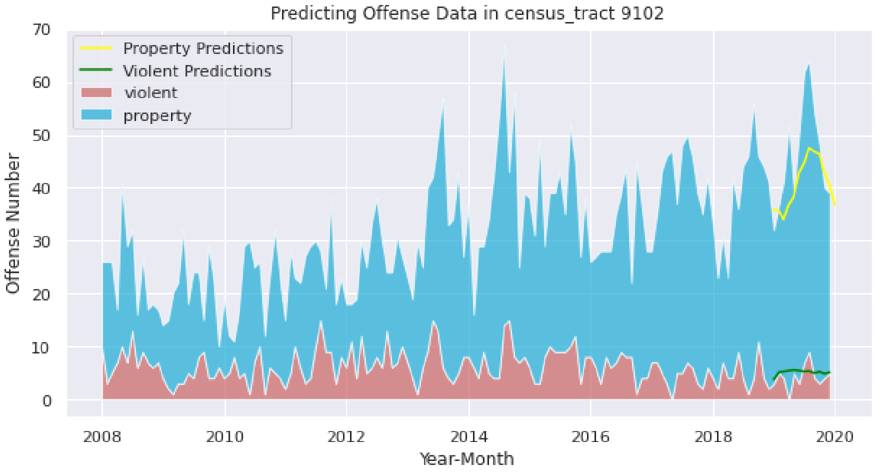


Figure 7: Crime rate prediction for different offense group in census tract 9102

For example, this is the output of when a user explores census tract number 9102 and wants to predict the crime trend up to the end of 2020. As can be seen in Figure [fig number], we can see violent-related crime as the red area whereas property-related crime is the blue area. The plot provides users a sense of how the number of records in each offense group and how the proportion of each offense group in total changes throughout years. The yellow and green lines represent the crime trend prediction in property offense and violent offense accordingly. This output suggests that it is safer for the users to visit this tract or stay less alert in the later 2020 since there is no trend of violence increasing and property crime is potentially decreasing.

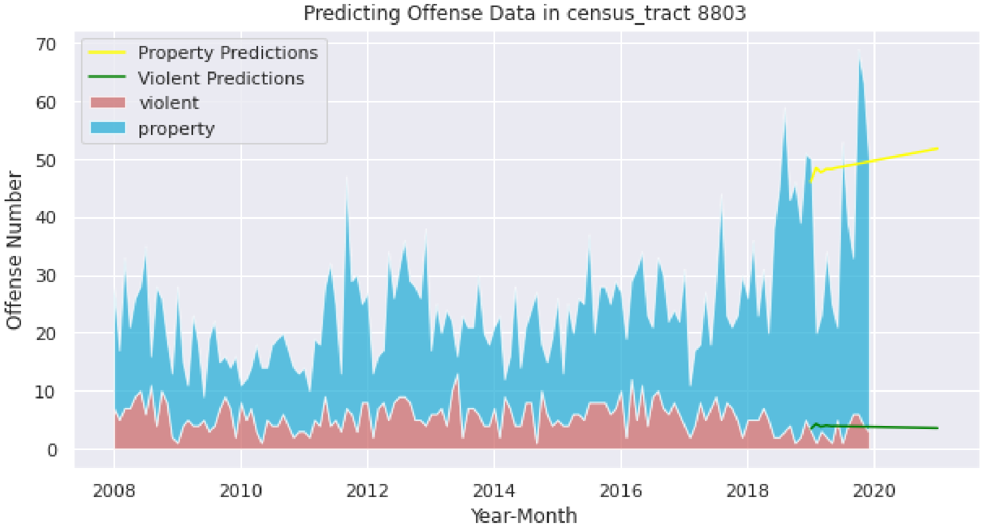


Figure 8: Crime rate prediction for different offense group in census tract 8803

In another example, a user chooses to explore the crime trend in census tract number 8803 and wants to know how the crime rate will be by 2021-01-01. Based on the data of previous years, our product predicts that property crime is likely to increase after 2019 and continue the trend until 2021-01-01 while the violence crime remains stable.

Geospatial Plot

We equip the product with interactive maps so the users can freely see the location of all offenses in a census tract throughout 13 years.

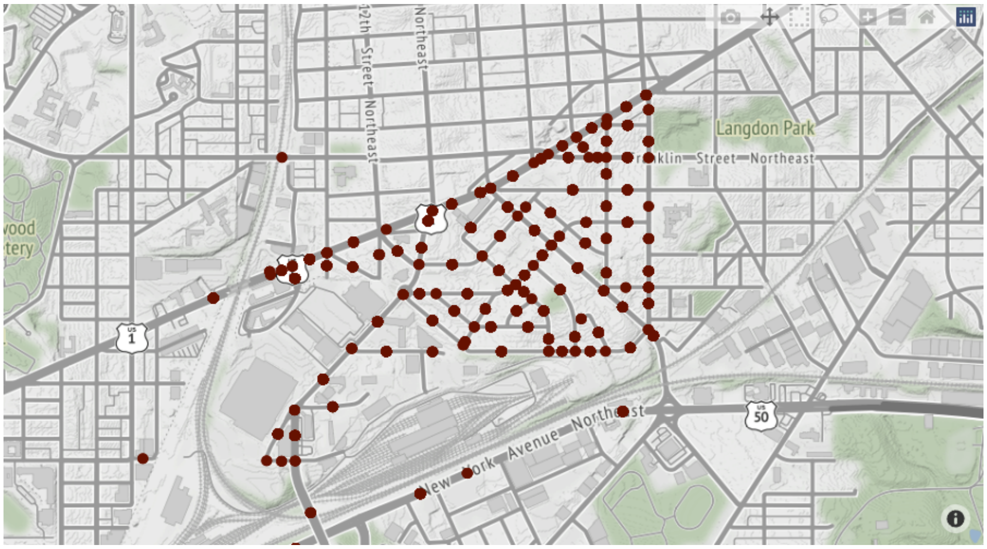


Figure 9: Crime data in census tract 9102 from 2008 - 2020

In Figure 9, the user chooses to explore the crime data in tract 9102 through 13 years and can visualize all the records on map. He/she can zoom out the map to see the exact location where a crime occurs like the figure below suggests:

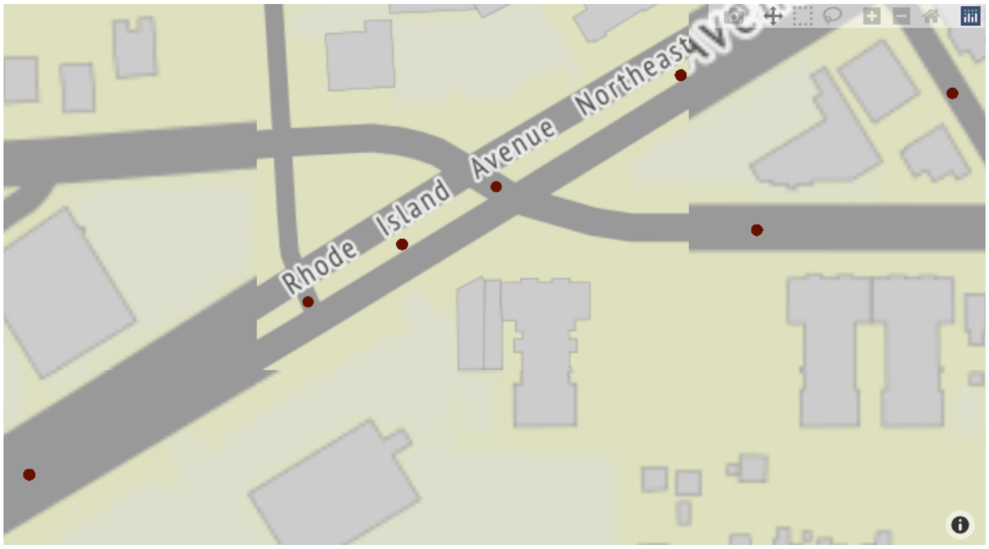


Figure 10: Zoom in and zoom out on the map

The users can also determine offense locations in a ward and on a date as they specify.

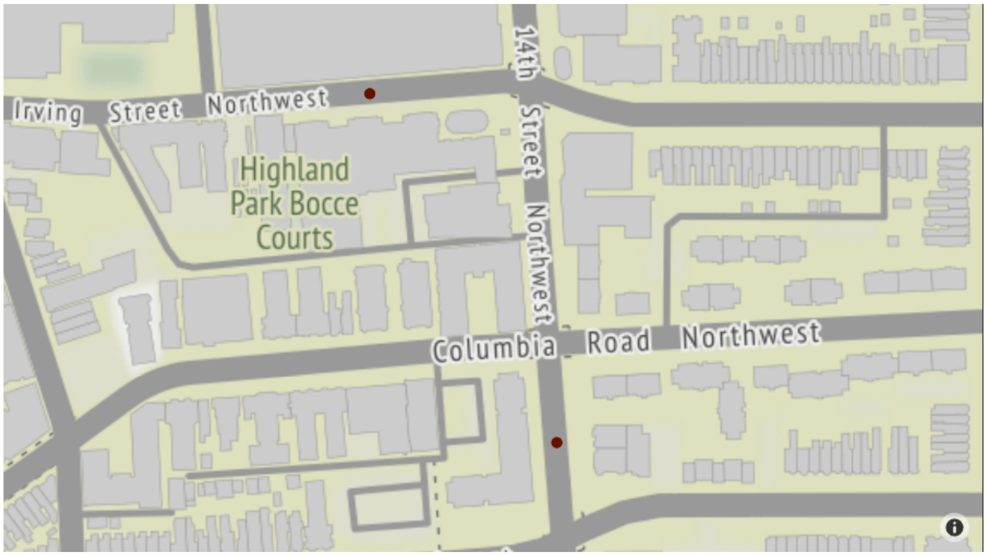


Figure 11: Crime records of ward 1 on 2019-10-22.

For instance, in Figure 11, the user specifies to look up crime data in ward 1 on 10/22/2019. He/she can learn that there were two offenses committed on this date. One was near Highland Park Bocce Courts and the other was near the intersection of Columbia Road NW and 14th Street NW.

To get a better understanding of what certain type of offense happened in the area, the user can also choose the ward or census tract along with the date or time of the year that they want to explore.

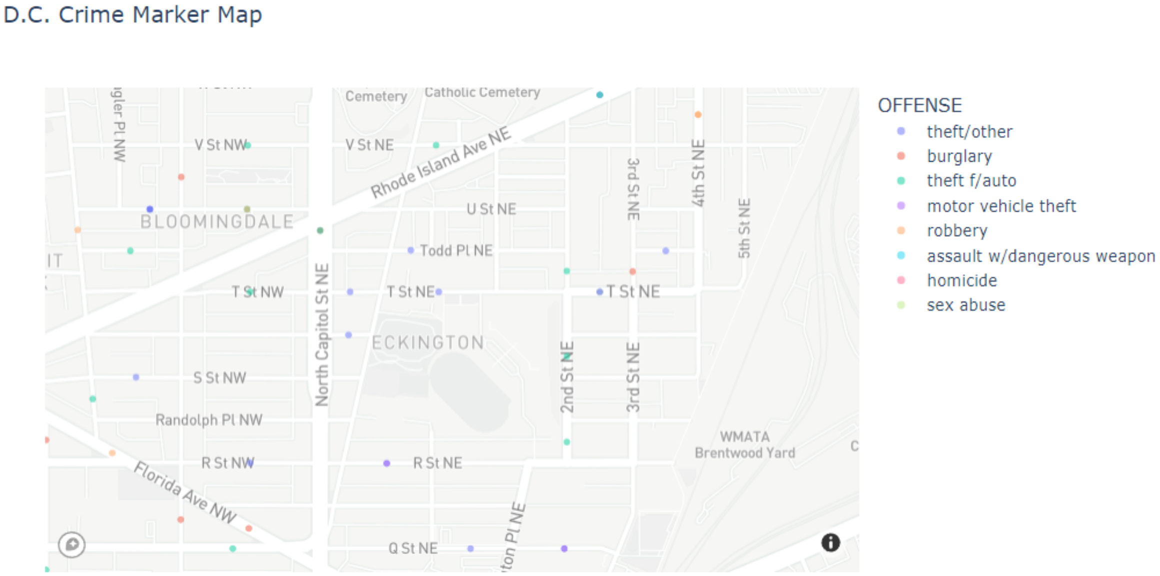


Figure 12: Offense map in ward 5 in Dec 2019

The figure above shows the zoom-out map of different offenses committed in ward 5 during December 2019.

Last but not least, the user can track the hotspot of an area by searching on the heatmap that our product provides.

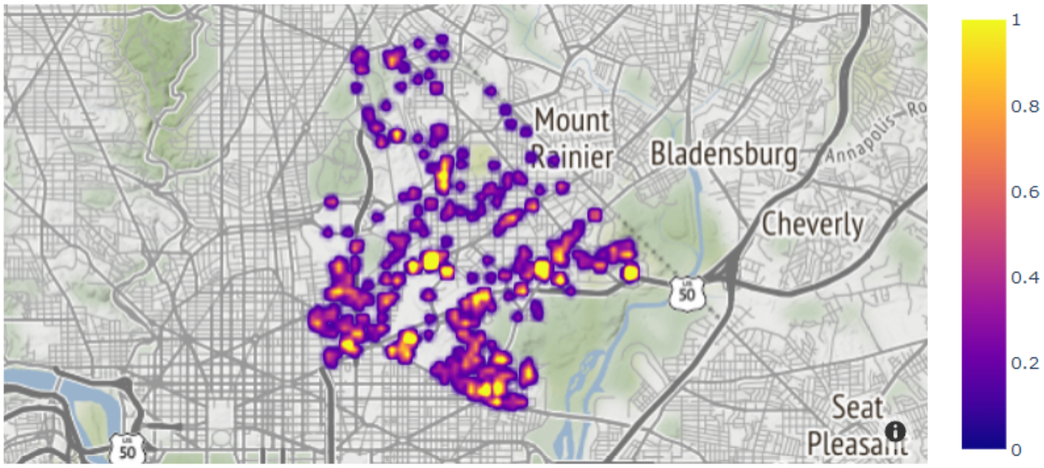
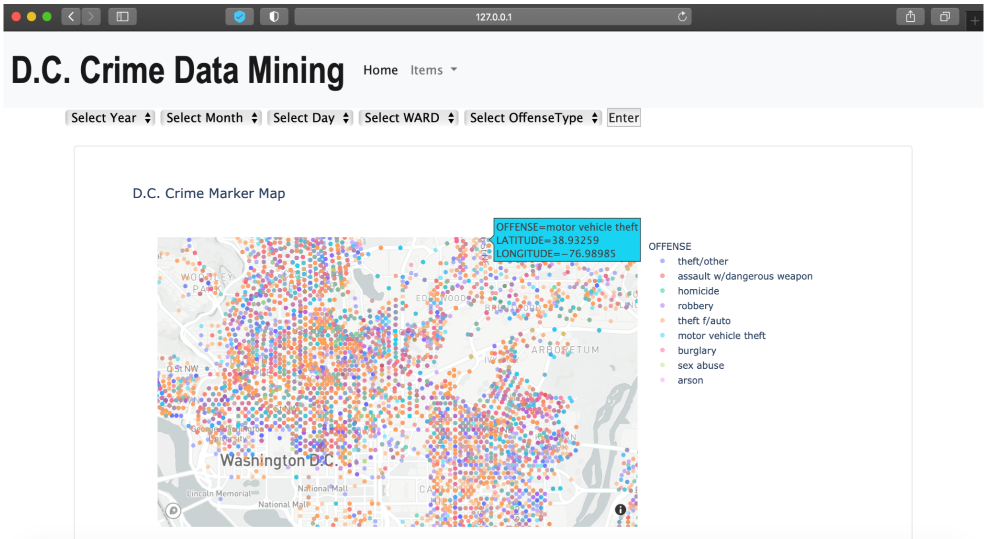


Figure 13: Heatmap of ward 5 in Dec 2019

The map above tracks the crime rate committed in all neighborhoods in ward 5 in Dec 2019. Purple denotes locations with the lowest crime rate while the lighter color denotes locations with higher rates.

Web Dashboard

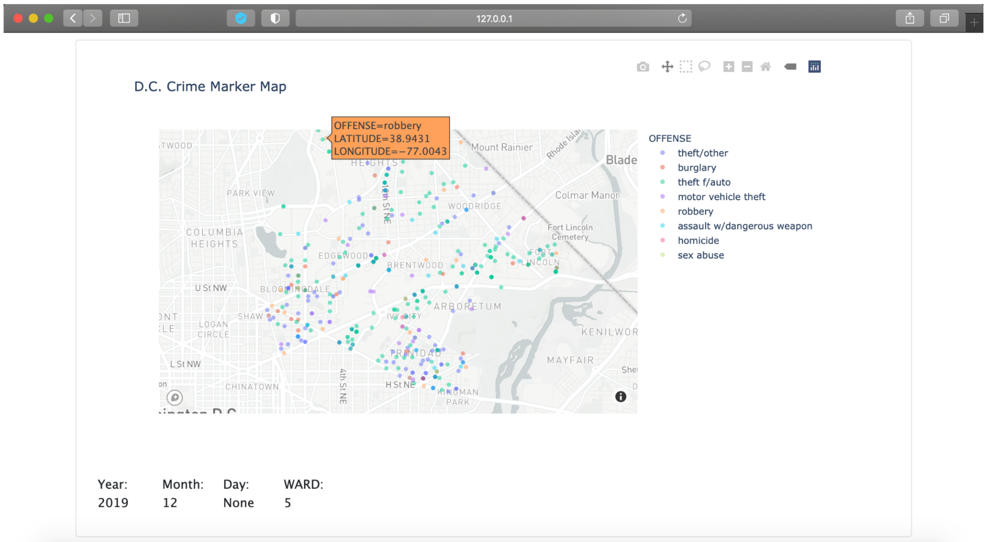
We provide users with visualizations of analysis in the form of a web dashboard. The dashboard allows for quick navigation to the analysis that users are interested in with the top of the drop-down menu “Items”. When selecting in the drop-down menu, users can investigate the analysis immediately. Users can specify the values of the selection boxes located at the top of this dashboard. If the user wants to investigate the crime data in ward5 in December 2019, specifying the value of the selection boxes changes the dashboard to what the user is interested in.



OFFENSE=motor vehicle theft
LATITUDE=38.93259
LONGITUDE=−76.98985

Figure 14: Web Dashboard Top Screen

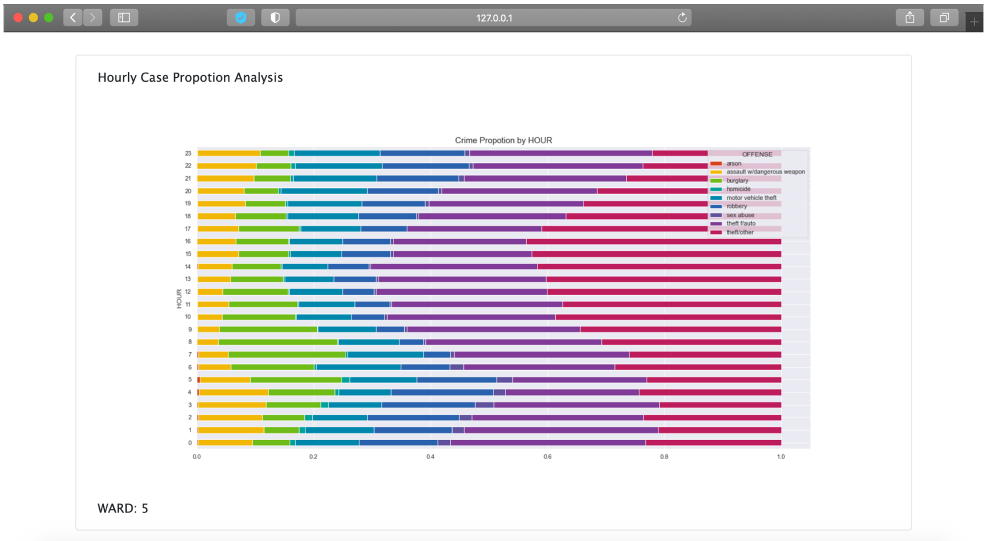
The following graph is a marker plot of crime data in ward 5 in December 2019. When users hover the mouse over a marker on the map, users can investigate the crime that occurred there. Additionally, as shown in figure 16, it is also possible to check each crime rate for each hour, linked to the ward value of the selection box selected by users.



OFFENSE=robbery
LATITUDE=38.9431
LONGITUDE=−77.0043

Year:	Month:	Day:	WARD:
2019	12	None	5

Figure 15: Marker plot of crime data in ward5 in December 2019



Hourly Case Propotion Analysis

WARD: 5

Figure 16: Hourly Case Proportion Analysis linked with the user's selection

A link to a demo of this web dashboard can be found on VT GitLab.

Design

Our product followed a standard data science procedure in its design. We started with finding a suitable dataset that contained enough spatiotemporal data for our analysis. The features we were looking for were time features like year, month, day, and hour. We also needed spatial features, like longitude, latitude, or different location groupings. At this point, we weren't sure about how specific we wanted our predictions to be, so we preferred all the data to be as specific as possible. The dataset we used was generated from the D.C. Metropolitan Police Department, and the website allowed us to select the type of data and specify the range of time over which to select from.

We proceeded with exploratory data analysis to learn more about some of the spatial features and the data distributions. One of the main things we wanted to gain from this analysis was how many values did each spatial feature have. The initial dataset contained the longitude and latitude values, but also had various location features that we didn't recognize. We spent time thoroughly researching the different location groups and finally decided that census tract and ward were the ones to keep. Census tracts were groupings of the city made for the U.S. Census Bureau. While we don't know why the lines were drawn the way they were, we do know that our dataset contained a little over 200 different census tracts. This allowed us to group our data by a reasonable amount for the time series analysis to use while also keeping it specific enough for relevant predictions. One of the groupings we did consider was the police districts, which would intuitively make it easier for police to respond to crime within their districts. However, during this step we discovered that there were only 7 police districts in the entire city, making it difficult to provide specific predictions that we saw as necessary from our user stories. Ward was also chosen because of an idea we had to improve generalizability of our findings along with improving the performance of our models. This idea was to merge the crime dataset we had found with various demographic features about the city. To accomplish this, we set out to find a dataset containing information such as average income, average age, percentage minors, etc. We wanted this data to be as specific as possible. Unfortunately, we were only able to find this data for each ward of the city, of which there are eight. Even worse was that the demographic information was not gathered frequently. We were forced to utilize the demographic information gathered in 2020 for our entire dataset, which spanned from 2008 to 2020. Regardless, we did merge the datasets and attempt to utilize them for our machine learning models, although it is likely that the demographic features were filtered out during principal component analysis due to the lack of variation.

Data preparation mainly involved removing the incomplete and unnecessary data. To simplify things, we removed any data sample that had a null value. Additionally, any features found to not be relevant during the exploration phase were also removed. Of the remaining features, we standardized the continuous ones as it is proven to increase model performance. For the categorical features, we performed one hot encoding to convert them into binary features, which significantly increased the number of features in our dataset due to the census tract having over 200 unique values. To reduce this amount that we knew would put a lot of strain on the models, we performed principal component analysis (PCA). Using PCA, we found that we could capture the same amount of data variance using only 10 features as opposed to the 300+ we had originally. Following this, we split the dataset into the features and the labels, which is necessary for the next step. The label we chose to predict was the offense group, which had two values, property and violent. This was necessary because machine learning forced us to presuppose a crime had already been committed, while the time series analysis didn’t. Thus, we decided with machine learning we should try to find correlations between the crime groups and the spatiotemporal variables. We felt these groups over the specific types were enough for our user stories and would also result in better performance. There was a significant skew in the amount of samples that were property crime vs violent crime. Because we wanted to know how the features affected the type of crime, we needed to correct this imbalance. To do this, we selected an equal sample of data by label and recombined them to make the dataset we used for our analysis.

We then used the dataset to train and evaluate various machine learning algorithms along with our time series models. For machine learning we needed to pick which models we wanted to test with our datasets. We would then perform 5-fold cross validation using the features and labels from before, which would result in a predicted label for each sample of the features. These predictions along with the actual labels were used with various metrics, such as precision, recall, and accuracy, to produce scores we would use to evaluate the model's performance. In addition to this, a confusion matrix was generated that allowed us to visualize the distribution of predictions and actual labels. For selecting the models to use, the decision tree classifier was the clear first choice. This method is very simple and intuitive to understand. The accuracy generated from testing the decision tree classifier with our data was around 63.5%. For a binary output problem, the random guessing accuracy would be 50%. From this, it is clear that the decision tree classifier was able to recognize some patterns between the location, time, and the type of crime. The next model we chose was the K-nearest neighbors classifier. This is another basic model that we thought could better utilize the spatial features in its predictions. The accuracy for this model was 68.5%, which was noticeably better than the decision tree. After that, we tested the random forest classifier, which is an ensemble method that utilizes multiple decision trees to make better predictions. This method outperformed both of the previous models, obtaining a 71% accuracy score. The next model we used was an AdaBoost classifier. The AdaBoost classifier is another ensemble method but while the random forest uses the bagging technique, AdaBoost uses boosting. We wanted to evaluate if the type of ensemble method had an effect on the performance of the models. After evaluating the AdaBoost classifier, we noticed that the accuracy was slightly lower, with only a 68% accuracy. The final model we used for our machine learning analysis was a multilayer perceptron with 3 hidden layers, ReLU activation, and the Adam optimizer. As a neural network, this model is able to use non-linear decision boundaries in it's predictions, which we thought would result in the best performance. Contrary to our hypothesis, the model only had a 68.5% accuracy. We suspected that this could be because of the lack of features that contributed to the overall data variance. Following the machine learning analysis, we utilized time series analysis to predict how much crime would occur in the future for each part of D.C.

In time series analysis, we conduct the Augmented Dickey-Fuller (ADF) test to check the stationary of the offense rate in a census tract. We decide to use this test because it tests the null hypothesis that a unit root is available in the current time series sample. The alternative hypothesis in our test is the trend-stationarity of the time series. The ADF statistic suggests that the more negative it is, the stronger the rejection of the hypothesis we are testing. In the implementation of the ADF test, we compare the test statistic with the critical value at 99%, 95%, and 90% level of confidence. If the test statistic is larger than any of these critical values, we conclude that the time series is not stationary and there is a potential trend in crime rate increasing/decreasing. Otherwise, the time series is stationary and no trend is found.

We also provide the users with 2 different metrics to assess the effectiveness of the time series analysis: mean absolute scaled error (MASE) and adjusted symmetric mean absolute percentage error (adjusted-SMAPE). A MASE value under 1 shows that the prediction is good whereas any value above 1 shows that the prediction was bad, suggesting that either the model needs to be fine-tuned or the time series analysis model is not appropriate for the current dataset. The adjusted SMAPE evaluates how close our predictions are to the actual values. It determines the error rate and its values range from 0 - 100%. The benefits of these metrics are that they are scale-independency and easy to interpret, which fit well with our current data.

We chose the Seasonal Auto-Regressive Integrated Moving Average with eXogenous factors (SARIMAX) as the prediction model in our product because it is a popular model in time series analysis. Both SARIMAX and ADF test are imported from the statmodel library. Using the SARIMAX model and combining the 2 above metrics with the stationary test mentioned, we could test the time series model with different hyperparameters to figure out which hyperparameters work best for our current dataset. The best model we have achieved so far has a MASE score of 0.41, followed by the 2nd best model with MASE score 0.44.

After fine-tuning the model, we let the user enter the census tract number and the desired future date for prediction he/she wants to use. We continue to extract the crime records of this census tract from the database and split it into training and testing dataset with a ratio of approximately 90-10. The model will be then trained on this training dataset and will provide the predicted trend on either the time range in the testing set or the time specified by the user.

In the geospatial plot, we pre-process our data by filtering and extracting the dataset according to the user's choice of ward, census tract, and date. We provide the users with interactive maps using Mapbox API. This was initially set up with Google Map API, but we then changed to using Mapbox API because it was more convenient to generate interactive maps with cheaper cost of service. In addition, Mapbox is equipped with attractive visual and base maps with very fast loading. Considering the scope of our project and we also want to learn a new API, we decided to use this service as an alternative to Google Map API. We also need to obtain a token to get access to Mapbox API like Google Map. The map plotting was straightforward as we can combine the API service with plotly library to generate the visualization of the crime data on a map.

In the development of the web dashboard, we use Bootstrap, a front-end web application framework, and Flask, a Python web application framework. Thus, this web dashboard allows users to interactively use and analyze the visualized data. Additionally, this dashboard draws a lot of graphs with Plotly Express, a Python visualization library that enables interactive graph drawing. Furthermore, the Mapbox

API is incorporated into this web dashboard to enable interactive operations such as zoom in and out on the crime marker map. Those tools provide users with a more interactive experience and makes our dashboard more effective.

Retrospection

Throughout this project our group experienced multiple things that went well and a few things that went wrong.

One example of something that went well was our use of Google Colab. Through using Google Colab, our entire team was able to collaboratively develop our product in real-time and share code dependencies. For our project, which followed a very linear design pattern due to its unique nature, this was a huge bonus over using version control through GitLab. Another thing that went well was our use of Trello to keep track of tasks that needed to be completed. Trello allowed us to assign tasks to individuals and set deadlines for when they should be completed. This helped us plan out our expectations and monitor our progress.

One thing that caused us a little bit of trouble was finding meeting times that worked for everyone. It was a lot harder to meet at times where everyone was free because we had a teammate that lives abroad so we could only meet at certain times after 7 p.m. Another difficulty that our team had to overcome was finding the data to use for our analysis. There is not a lot of accessible information on the types of crime in D.C. that contains as much information as we were looking for for our analysis. Even the data we did gather was not as much as we would prefer. Our last major difficulty was performing the analysis itself. Our goals for the project were to predict the time and place of a crime, but we were forced to determine how specific we wanted to predict these variables. To solve this, we decided to group the crimes by one of the spatial features of the dataset, census tract, and also by month, and perform a time series analysis in each of those. While it isn’t predicting the crime down to the exact time and location, it is the best we can do to get results.

Recommendations for Future

Some of the recommendations we have for the future to enhance this project would be getting access to better data. We believe this could be possible by getting more datasets and merging them. We would also want to provide the users more flexible choices of location, that is, they can specify an address and analyze the crime data in its neighborhood instead of a broader area. Additionally, one improvement that could be made is accessing datasets from other places such as New York City. Furthermore, meeting more often and communicating with each other would also allow the project to run smoother and provide more input on what the rest of the group is working on. Finally, adding more in-depth analysis and adding advanced web development which allows us to display the analysis in an interactive dashboard would create a better product for users to interact with. We would also like to use the machine learning models to identify which features have the most significant impacts on the types of crime as well as the amounts of crime. With more time, we believe all of these goals could be achieved.