CSE158 Assignment 2: Gun Violence Prediction

Wenyi Li
wel164@ucsd.edu
PID: A92126727

Bingjie Zhou
biz003@ucsd.edu
PID: A13711806

Yang Liu
yal318@ucsd.edu
PID: A15342355

Abstract

Gun violence is and has always been a serious issue in the United States. Enforcing more gun control does not alleviate this growing problem effectively. In this report, we take on a statistical approach help local crime prevention by analyzing over 200,000 gun violence incidents in the data set of Gun Violence database in the U.S., unraveling the trends and the increasing frequency of gun violence. With this large data set, we select and design models to forecast related crime activities given place and time. Model concepts that we experiment with include random forest classifier, support vector machine, k-nearest neighbor. We aim to find a predictor with high ($> 95\%$) accuracy judging whether there is going to be at least a gun violence incidence at the location and time.

1 Dataset

The dataset we use for prediction task is from Kaggle: Gun violence database which archives U.S. gun violence incidents collected from over 2,000 media sources, containing incident data, state, city or country, address, number of killed and number of injured. The size of dataset is consisted of 23,0399 samples (having removed the duplicates and Nan rwos). From exploratory analyses of the dataset, we have diagramed a number of injuries and deaths due to the incidents happening since 2013.

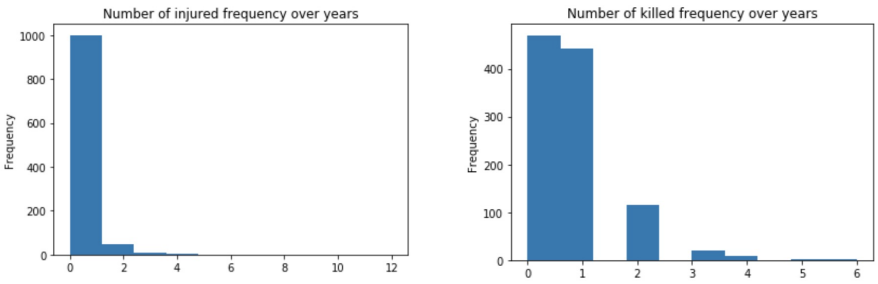


Figure 1: number of victims since 2013

This dataset also contains geographical data and exact latitude and longitude where the incidents have happened. Combining them with the date of each datapoint, we are able to explore the representation of this issue on the map of the United States.

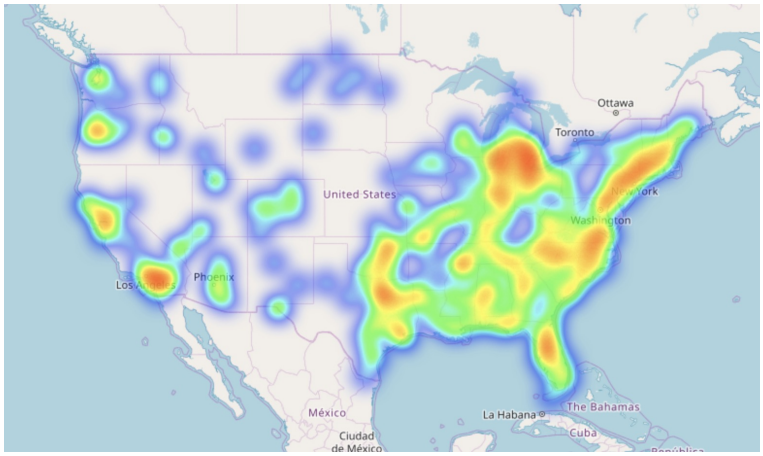


Figure 2: Heat map of gun violence incidents

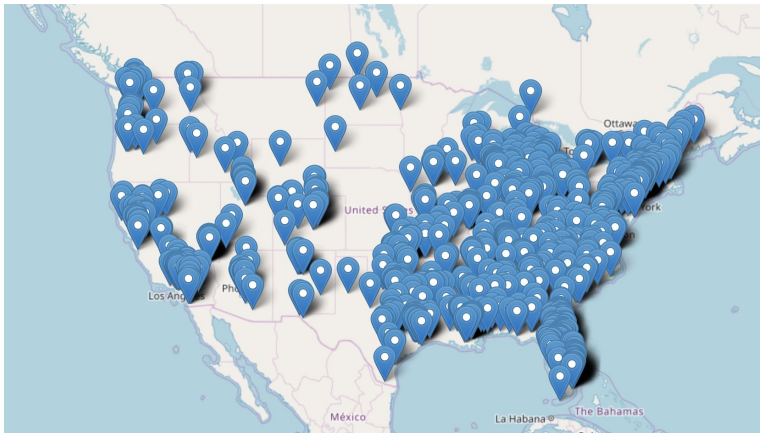
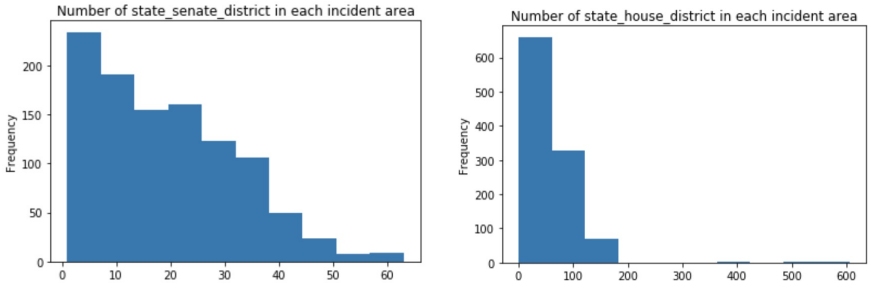


Figure 3: Pin-point map of gun violence incidents



The credibility of the data is sound, given most records have corresponding proof and source in this dataset. At the same time, we found that the number of senate district in the area of the incidents is correlated with the frequency of gun violation instances. Since our dataset contains a number of senates with a number of incidents in a specific area, as well as the number of state houses, we thus

can visualize using histograms how severe violent situation is controlled in depending on the location of the area.

In addition to the overall quantity of gun violence incidents on the map, we find there is a change of geographical distributions every year. Within each year, the number of victims varies quarterly. We explore the meaning of such incidents by year and plot number v.s. time graph that each contains the number of total incidents, the number of deaths, and the number of injuries each year. The plots demonstrate an ideal trend of increasing number of report on gun violence and the data distribution change.

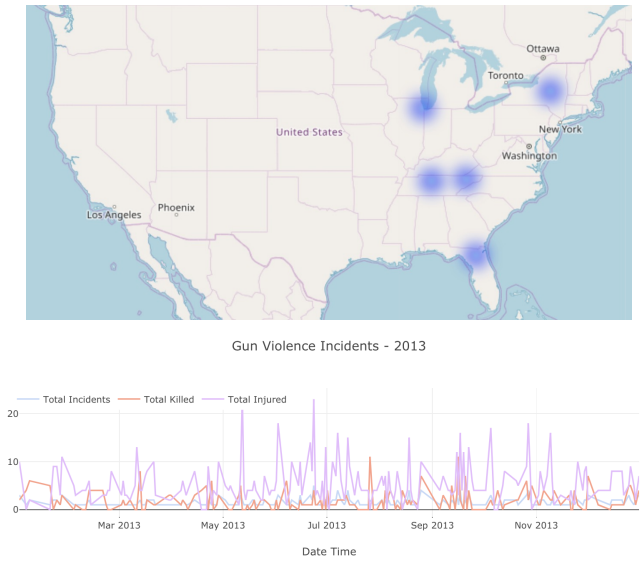


Figure 4: 2013 Gun Violence Incidents Heatmap and Tendency chart

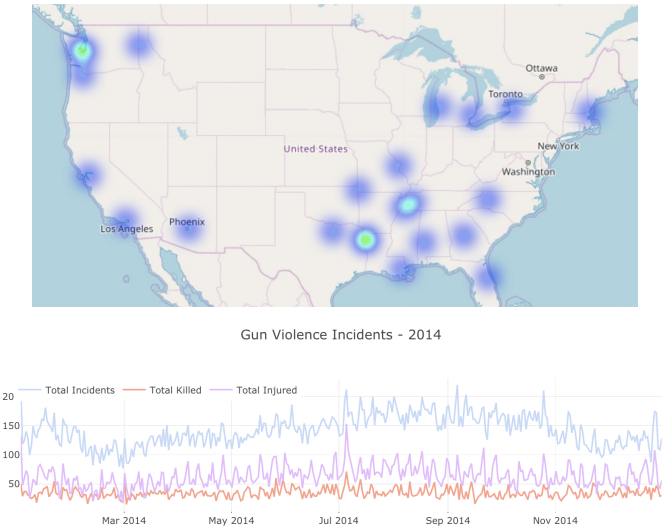


Figure 5: 2014 Gun Violence Incidents Heatmap and Tendency chart

The gun violence incidents in 2013 and 2014 are sparsely distributed in the United States but clustering on particular spots. And in 2014, there are more gun violence incidents and more concentrated on particular locations.

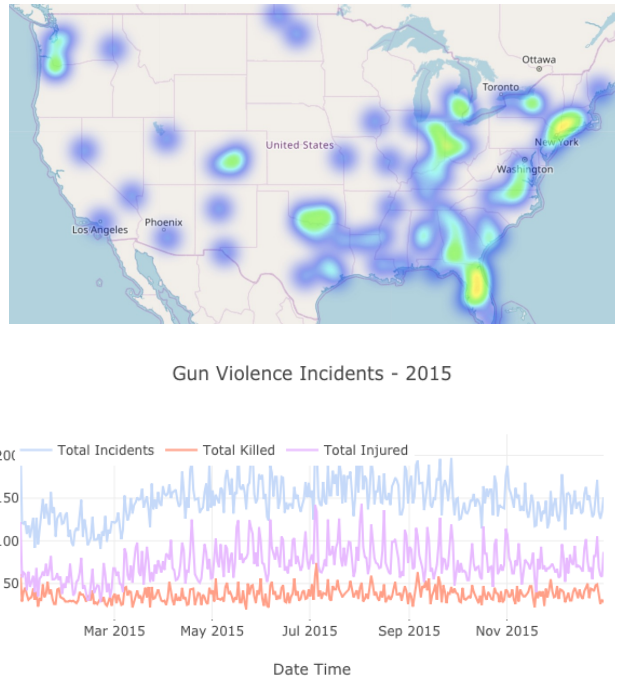


Figure 6: 2015 Gun Violence Incidents Heatmap and Tendency chart

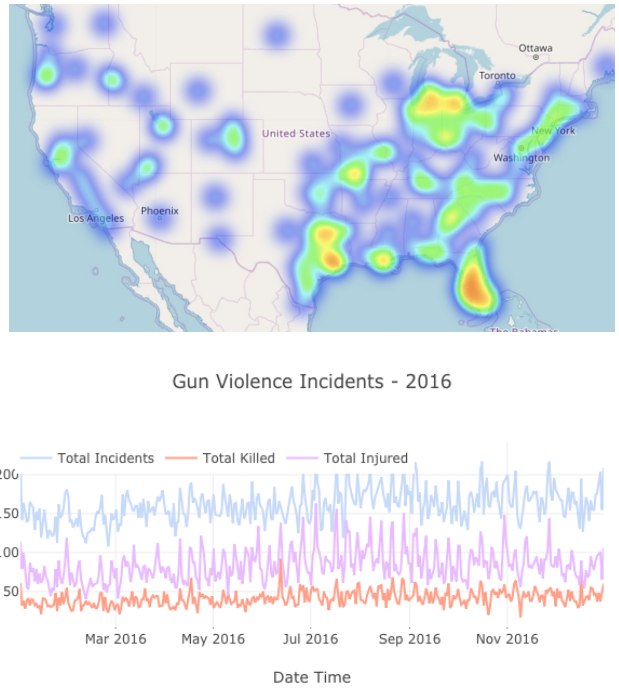


Figure 7: 2016 Gun Violence Incidents Heatmap and Tendency chart

We noticed the increasing fluctuation of amplitudes in the plots and the average number of injuries, including the Las Vegas gun shooting incident in October 2017. They imply a increasing seriousness of the in this year and the following.

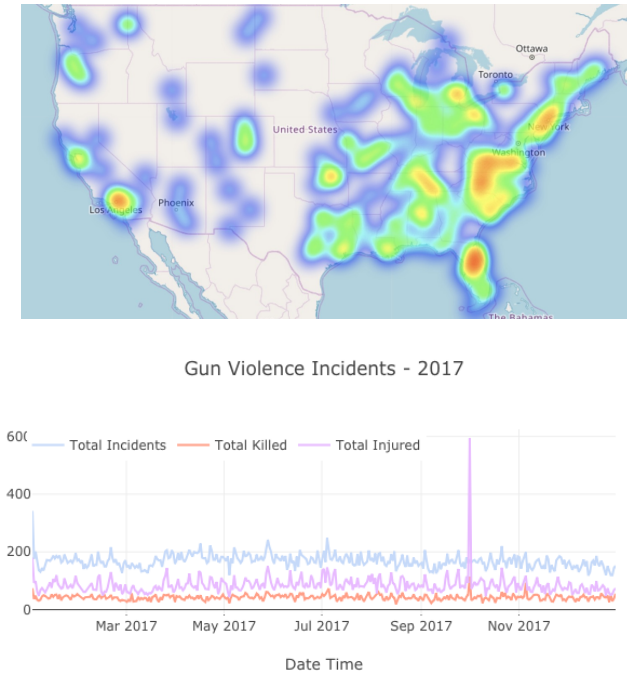


Figure 8: 2017 Gun Violence Incidents Heatmap and Tendency chart

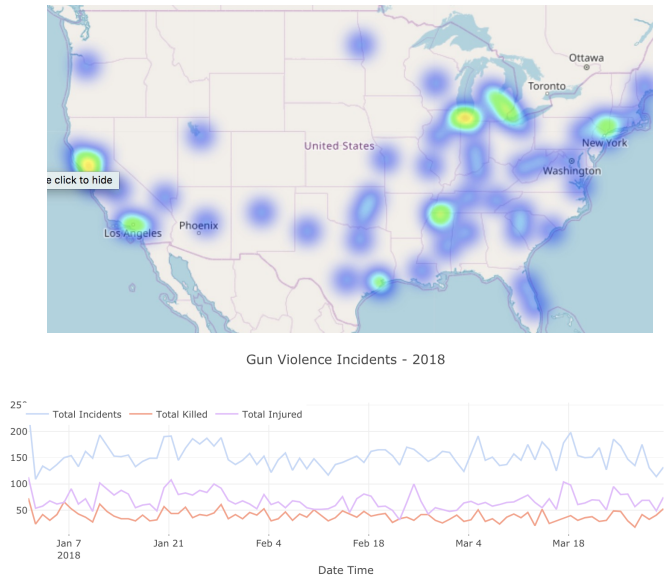


Figure 9: 2018 Partial Gun Violence Incidents Heatmap and Tendency chart (Jan.-Aug.)

The visualizations also suggest that the incidents happen uniformly in seven days of a week, roughly distributed in each month, but an increasing trend yearly. It is also suggestive that violent incidents happen more likely than suburban area.

At the same time, we combine another dataset named Google local review[9] which contains ratings, price ranges, opening hours, etc of local restaurants, which we try to analyze if downtown business district would have higher rate of gun violence incidents.

Example (business)


```
{
  'name': u'Diamond Valley Lake Marina',
  'price': None,
  'address': [u'2615 Angler Ave', u'Hemet, CA 92545'],
  'hours': [[u'Monday', [[u'6:30 am--4:15 pm']]], [u'Tuesday',
[[u'6:30 am--4:15 pm']]], [u'Wednesday', [[u'6:30 am--4:15 pm']], 1],
[u'Thursday', [[u'6:30 am--4:15 pm']]], [u'Friday', [[u'6:30 am--4:15
pm']]], [u'Saturday', [[u'6:30 am--4:15 pm']]], [u'Sunday', [[u'6:30
am--4:15 pm']]]],
  'phone': u'(951) 926-7201',
  'closed': False,
  'gPlusPlaceId': '104699454385822125632',
  'gps': [33.703804, -117.003209]
}
```


Figure 10: Google Local Data Example

Many features from the Gun Violation dataset can be used by our models such as id, date, state, city or country, address, number of killed, number of injured, latitude, and longitude. By exploring the correlations between existing data and the other data we acquire, we can design the input feature vector from the uses of more than one sources to train a model with more confidence in the task of prediction. Some simple features that can be put into vectors for the model can be, for example, the number of Senates and number of state houses close by, since we have seen a correlation from charts above: with more senate in the area, the number of violent incidents decreases and vice versa. At the same time, we can also feature whether the location is close to suburban areas or not, which seems to be correlated to the chance to be involved with gun violation crimes.

2 Predictive task

2.a Model output

We take a straightforward definition for the task of the models: to predict if a gun violence incident would happen in some areas at a specific time. For example, we want to know if gun violence would happen in San Diego tomorrow or not. Given the location of San Diego and the date, the prediction result is simply ”yes” or ”no”. The entire data set is divided into half one for validation and one for training. When N is the number of data points in validatoin set, validating the model requires using all the data points form the validation set and calculate the accuracy by dividing the number of correct prediction by the number of all predictions.
Since the balanced dataset is 400k large, 200k of validations gives significance to the accuracy we find. However, to achieve effective learning of the models, certain un-useful parts of the data need to be pre-processed.

2.b Data processing

Before vectorizing these features directly, we need to ”pad” the data with the same number of data points that are not gun violence incidents, namely the negative targets. This way we can achieve balanced outputs, so the models will not learn a the skewed distribution of all positive targets. At the same time, shuffling all existing data helps the training become less biased as well. By doing so, the inputs of the model can be sampled from an approximately uniform distribution of incidents that involve violence (as ”1”) and that do not (as ”0”).

2.c Features

Here are the multiple categories we choose as features for every model's training for better performance in the selection shown below. We also use one-hot encoded date data to increase model's

GPS	If in one of the most dangerous city	if it's in one of the most dangerous state	Date

complexity. January 1st does not necessarily have correlation with Janary 2nd. Therefore we manage not to use ordinal representation for the month and the day of the month. However, we learned from our visualizations that the gun violence incidents grow yearly, so it is safe to leave the year feature not one-hot encoded because, for example, 2014 does mean more gun violation incidents than 2013's amount. Meanwhile, we combine some existing feature we consider to be highly correlated into new features to reduce model over-fitting. For example, we sum "number of injured" and "number of killed" of a data point and mark it as the "number of loss." While there are other features we have considered such whether it is a state senate district, we think latitude and longitude of the GPS geoencoding can act as two independent principle components of the inputs that are well defined.

2.d Baseline

Our baseline is built on the fact that some areas have more incidents happened compared with other cities. Therefore, cities with larger number of incidents happened before are predicted to have higher possibility of happening another gun violence.
Following this design, we map cities to a corresponding weighted number of incidents happened in this city. The weighted number is a sum of whether a violation has happened or not (1 or 0) + number of people injured + d number of people killed, from 2013 to 2018. Then we sort this mapping by the weight and extract the most dangerous cities by extracting 50 percentile of the sorted list. That is if test city is in the list of "most dangerous cities", our baseline model outputs 1, otherwise 0. The accuracy for the baseline model is 0.5217, which is better than a coin flip, but it leads us to try different percentile values to optimize the model. Using a 70 percentile to extract most dangerous cities, we are able to improve the accuracy by a small amount; however, this model has an upper limit of performance due to the fact that it is only judging the data from the average of all time. The dataset, also in the visualization, has a time-variant property that makes the distribution of crimes change through time. By taking the average of all data from 2013 to 2018, we can only get an average accuracy due to the noises and fluctuating geographical distribution.

2.e Proposed Models

1. Jaccard similarity model: It evaluates the similarity between different cities, and if we test "San Diego", we can find lists of the most similar cities with "San Diego." We check if those cities have some specific properties which can determine if violence would happen at San Diego. The potential drawback of this model is that we cannot simulate city with correct enough features, thus leads to calculated similarity scores having high uncertainty deviation.

2. Linear SVC(Support Vector Machine): We can train a classifier that focuses on the difficult examples by minimizing the misclassification error,[6] which is mainly used on binary classification that emphasize accuracy of not mislabelling. While the training time is short, this model may not have enough power to fit complex data.

3. Random forest: It is a model that applies the general technique of bootstrap aggregating and leads to better model performance because it decreases the variance of the model without increasing the bias.[8] This model can self-select some good features among all features we feed in at the disadvantage of extremely long training time.

4. K-nearest neighbor (KNN): This clustering is popular for its unsupervised fashion of learning the inner data structure. For geographical data we get, KNN can be a better model to find the

center of the crime network. Compared to support vector classfiers, it does not linearly interpret the data, so there is a better classification of the points. However, since we do know the local distribution of points, we do not know if the distribution is a good fit for the clustering: when the places of gun violence incidents can be clustered, can the place without those incidents be clustered as well?

2.f Evaluation

After pre-processing data, we divide the test dataset and validation dataset and make sure there is no duplicated datapoint with each other. We want to test the model with unseen data not in the training set. Moreover, since we balance the data by adding cases that no violent incident, the model can learn and formulate patterns close to real life situation. Due to a large set of data, we try different splits between test and validate data and use "mini-batch" fashion to train and evaluate the model firstly on a small set of data such as 10,000-20,000 and compare which model performs better with same hyperparameters setting. We consider precisions and recall may be a better evaluation method in real life situations. When validation set only contains negative data, a model that only outputs negative can score 1.0 accuracy. However, since our test set distribution is tightly controlled, using accuracy is a decent evaluation method in our case. With uniform distribution and set data randomly shuffled, with multiple evaluations, we also achieve the effect of k-fold cross-validation. After we decide the final model, we train it on a larger set of size in an attempt to increase its accuracy.

3 Models

3.a Jaccard/Cosine Similarity

In order to get the probability of criminal incident occurrence on given date and place, one idea to improve the baseline is to calculate the similarity between two dates based on the geographical features of their criminal incidents location history. The main idea is based on the intuition that places with similar geographical characters with areas that have a criminal record would have a larger chance to become the occasion criminals choose for violence events. For example, during large festivals, criminals would choose prosperous commercial districts as gun violence event location if the criminal motivation is based on their anti-social personality. Therefore, this added feature of similarity would be able to boost the accuracy of baseline prediction.

We calculated the Jaccard and cosine similarity based on the following two formulas. With the input of date and location, we search through all dates with incident happened on the same location and compare the similarity between them. If any of the similarities is over a threshold, the result of the given date and location is true; i.e. there would be gun violence on the given date and location. The threshold for Jaccard is 0.5 and 0.9 for cosine. The formulat uses algebraic terms: J = Jaccard, D = Date, P = Place, B = Business, Ap = Average Price, Bc = Business count, Bp = Business price.

$$J(D_1, D_2) = \frac{P_{1 \in D_1} \cap P_{2 \in D_2}}{P_{1 \in D_1} \cup P_{2 \in D_2}}$$

$$Sim_{Bc}(D_1, D_2) = \frac{\mathrm{B}_{\in P1 \in D_1} \cdot \mathrm{B}_{\in P2 \in D_2}}{\| \mathrm{B}_{\in P1 \in D_1} \| \cdot \| \mathrm{B}_{\in P2 \in D_2} \|}$$

$$Sim_{Bp}(D_1, D_2) = \frac{\mathrm{Ap}_{\in P1 \in D_1} \cdot \mathrm{Ap}_{\in P2 \in D_2}}{\| \mathrm{Ap}_{\in P1 \in D_1} \| \cdot \| \mathrm{Ap}_{\in P2 \in D_2} \|}$$

In order to get the geographical features of places, we use data from Google Local Reviews.[9] We utilize the places data in this dataset which includes longitude and latitude GPS information and approximate price level of businesses all over the world. In order to match the gun violence dataset

which only includes the gun violence in the US, we filter the Google Local Business dataset and restrict to US business. We use the two features of samples in this dataset which are GPS location and price level of businesses. Therefore, we can calculate how many businesses around each gun violence incident location there are and what the average price level or consumption level is near the location based on the GPS information. However, the cosine similarity calculated based on the nearby geographical information does not have state-of-the-art performance on our prediction task. This may due to the extremely limited range of the price distribution which is majorly within [2,3]. Also, the number of businesses around gun violence locations does not have a large effect on the prediction result.

3.b SVM

Compared to logistic regressor, a support vector classifier may be more qualified to achieve the goal of finding fairly exact location's gun violence incidents for crime prediction. We want to clearly separate yes from no across the cities and counties, and make clear decisions with boundary between decimal units of latitude and longitude. A linear kernel in SVC separates the data with lines; a polynomial or a RBF kernel separates data with high demensional, complex curved lines, which is less preferable than linear kernel. There are seldom geographical organization in the United States that has curved patterns, and it does not make sense geographically:

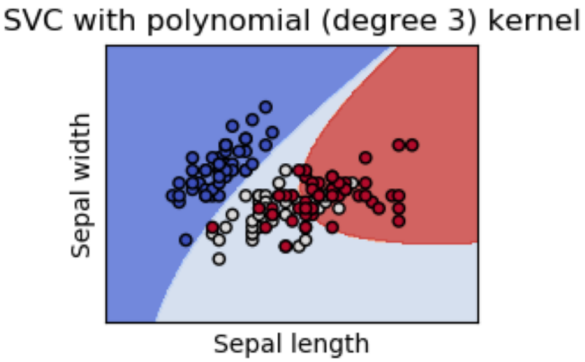


Figure 11: Nonlinear SVM Classification Visualization

The plot above[7] illustrate an example polynomial kernels and how it separates three different classes. It is not preferred in our case. If the red dots represent the gun violation incidents, the white dots in the middle should not be classified as other "safe" places. A counterexample to this argument can be if it is in the mountains and they do have this separation (e.g. by different altitude). However, since the most gun violation incidents are in urban or populated regions, the model needs to learn a better generalization than over-fitting to accommodate very few special cases. SCV with a linear kernel is also chosen for the faster run-time efficiency compared to the other kernels. It is also an ideal model to use when working on a binary classification problem.
We tried different regularization constant and types of features and find that one-hot encoded date increases the model accuracy by 16%. However, there are drawbacks. Even if the data provided, the feature vector representation we have designed may not be enough to let the model best describe the trend or the inner structure in the actual data. Therefore there is a limit of what a support vector classifier can learn due to the design of the feature. The dataset also poses another limit to the accuracy of the model. From the heat map visualizations, we can tell that there are noises in the map. The more random noises in the graph, the less confidence that the model draws the boundary support vectors.

3.c Random Forest

When we are not 100% sure about the internal relationships between the data points, A random forest classifier is highly beneficial to eliminate incorrect hypotheses of the data structures we propose. In the training process, we feed in different combinations of features in random forest such as:

	Input features
1	Date, GPS
2	Date, GPS, Popular crime cities
3	Date(one hot encoding), GPS
4	Date(one hot encoding), GPS, Popular crime cities

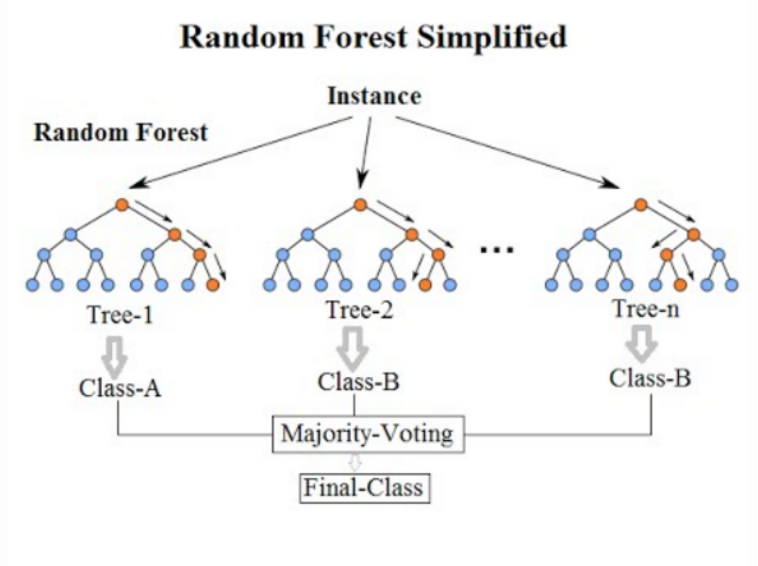


Figure 12: Random Forest Architecture

Moreover, we train the model based on different numbers of estimators for max depth. Random forest is good for dealing with unbalanced and missing data. However, the runtime is relatively long and it can not be used to predict beyond the training data. Meanwhile, a random forest is highly possible to over-fit if training data is noisy.[8]

3.d K-Nearest Neighborhood

From the data visualization part, we can see that the gun violence events happen in a distribution that can be clustered. Therefore, we come up with the idea to use K-Nearest Neighborhood(KNN) classifier for our prediction task. Because KNN is used by separating classes into different classes to predict the result based on nearby sample points. In our prediction task, from the baseline, it shows that the places with more criminal historical records would have a larger chance to involve in gun violence tasks. So clustering the sample points with their locations and predicting based on nearby gun violence history is an intuitively feasible approach.

In our model, we firstly pad our data with negative data points and randomly shuffle the dataset. Then we divide the dataset with half and half ratio into training and validation dataset. Then we empirically train three KNN classifiers with three different k of 2, 5, 10 which represents the number of nearby samples the classifier uses to predict each input. For each classifier, we use four different feature vectors to train the model and evaluate the result of each model based on the accuracy score on the validation set. In order to prevent overfitting, our validation set is completely separated from the

training set. We will discuss the result of the model in the result section and have more explanation of the details.

3.e Different approaches of data balancing

The original data set we have contains all reports of gun violence incidents, which means that all data points are targeted positive. Only learning the positive target will mislead the model to learn to only generate positive outputs. Therefore we want to balance the data as an evenly distributed and more generalized input. The original data are labeled with violent = True, and we pad the data with randomly generated data points with violent = False.

Our first approach is to shuffle all features value we select and recombine them to different location point, date time, city and state. However, we realize that the thoroughly random generated location can be too diverse so that the geological locations do not match to the corresponding city name, which are unreal data that may negatively affect model's learning

Our second approach is to shuffle all features value in the range of each state. For example, we shuffle all data points of longitude and latitude in the range of California to make sure the newly generated location would not go outside California. There are still possible mismatches since not all states are square-like; location of some generated points would be located in the ocean or in some places outside the US.

Both approaches present the possibility of generating duplicates in conflicts of the original data. For example, there may be a generated data indicating no gun violence but it could be a point from the original dataset at the same time with the same location, indicating an actual gun violence event. We check the date by string comparison, and for geological coordinates, we introduce a spacial different parameter lambda,

$$\lambda = 0.1$$

We want to make sure the Manhattan distance is less than lambda.

$$|randLat - lat| + |randLong - long| < \lambda$$

This methods approximate Manhattan distance to be less than 7 miles or 1 km, this ensures the affected area of gun violence.

3.f Unsuccessful attempts

We have generated a list of the most dangerous cities (cities with a higher number of gun violent incidents). In the beginning, we sort cities by a number of incidents happened in that area before and select top ordered cities based on threshold without considering a different number of incidents happened in each city. For example, if we have some data like this:

City	Number of incidents
New York	60
San Diego	20
Los Angeles	35
Atlanta	77
Detroit	140

We generate the order like Detroit, Atlanta, New York, LA and San Diego. If the threshold value is 40%, we will only get list of cities: Detroit and Atlanta. However, this way of generating is not accurate since we treat some cities have much more incidents happened the same as other cities

without putting any weight on them to emphasize. Therefore, we change to generate list by weighted number of incidents happened in cities. From this method, we will generate the list of cities: Detroit. The latter list would have more precisely generating result which is better to use in model.

In addition, we implement the cosine similarity calculation based on the nearby number of business and average price range of the businesses. However, this model does not have obvious performance improvement. This may due to the small range of the average prices of business. After some research, we find that the cosine similarity cannot be effected obviously through magnitude of the vectors which means the difference of prices and number of business does not have large effect on cosine similarity difference. Therefore, the result of cosine similarity we calculate out from the model is differentiated to extreme values of 0 or 1. And also the insufficient or mismatch of two dataset can also generate inaccurate and irrelevant results.

3.g Scalability

Our dataset is already reasonably large enough and we get pretty high accuracy result on our model. However, if we extend the size of the dataset to 10 times, the result and computing time would relatively have some changes.

For baseline model, the change of result would not be much because the model is just linearly checking if the city is among top criminal cities based on history gun violence record and the computing time would be larger but not change too much.

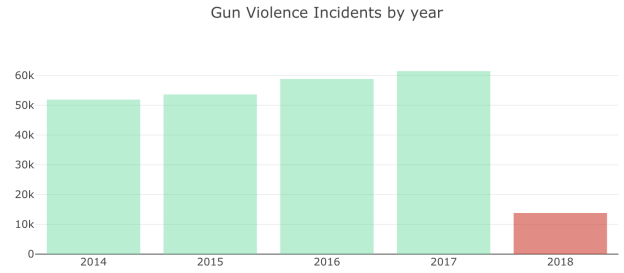
For the Jaccard similarity model, the most challenging thing is the computation cost. Because we need to get the nearby business counts and average price. The runtime for getting this information is O(mn) where m is the size of Google Local data and n is the number of different locations. Therefore, if we extend the dataset to 10 times, the computation would be very slow. And with 10 times large dataset, the validation set would also be 10 times as before. So the prediction time would be increased to a great extent.

For the random forest model, if we extend the data set to 10 times, the data set may become noisier and affect our model negatively. However, if we preprocess with larger data set again before approaching to train model, larger data would boost up our accuracy. For time complexity, random forest has O(v * n log(n)), where n is the number of records and v is the number of variables/attributes.[10] Therefore, with an unchanging value of v and a ten times increase of the value of n, we would have 10n*log(10n) time complexity increase, which the speed is still acceptable.

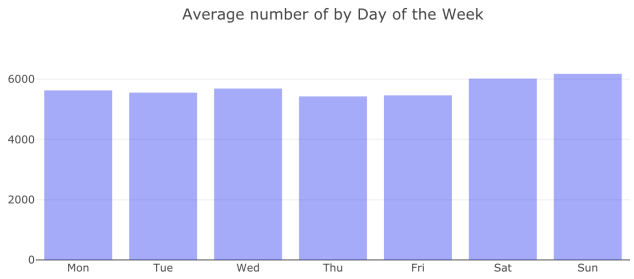
For SVM, the training time complexity is about quadratic. With 10 times large dataset, the training time would be quadratic slower than before. The runtime for the prediction process is linear with respect to the number of support vectors. Therefore, there will be a linear increase of runtime on prediction.

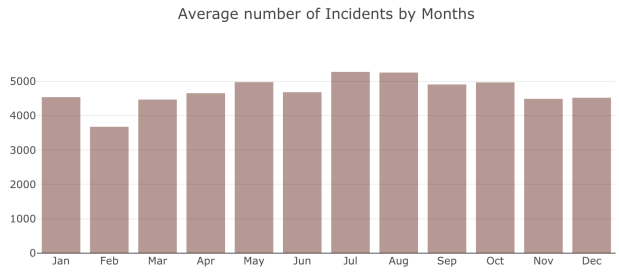
4 Literature

Our dataset comes from Kaggle: Gun Violence Data, updated by James Ko. It was being used in different ways such as analyzing data and generating new predictions.
From "Deep Exploration of Gun Violence in the U.S." written by Shivam Bansal, we learn the distribution of each category in the dataset and how they relate to each other.[1]For example, we have an increasing number of gun violence incidents each year by increasing growth rate.

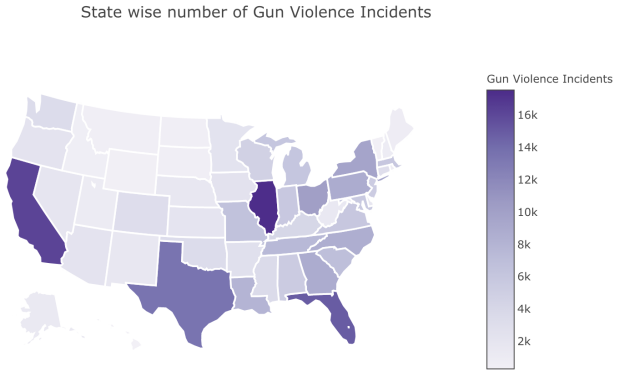


And we can conclude that we have a fairly even distributed model for date's happening rate from the following two diagrams. Since the data set has been studied in the past, we can find that training date and year only is not enough and relatively meaningless because they are evenly distributed.

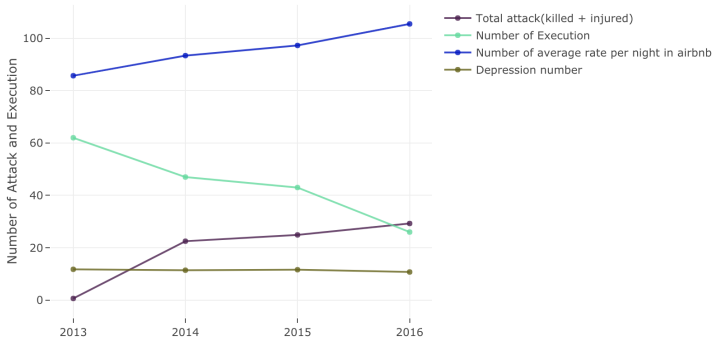




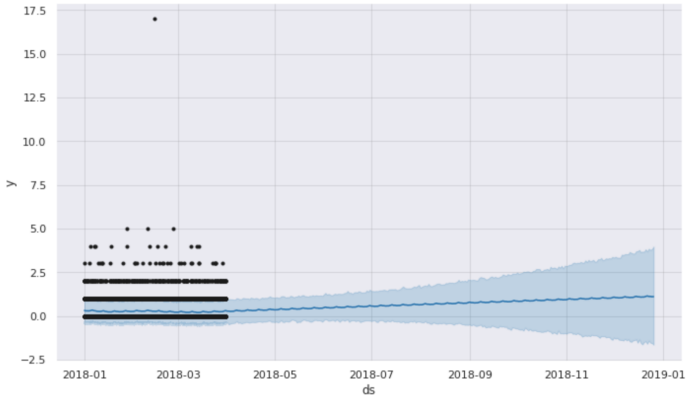
Even if we have parts of data that are evenly distributed, we can find some more data which has a distinctive distribution such as a number of incidents happened in different states. When training our models, we can put more weights on different states to classify less vague boundary between our train cases.



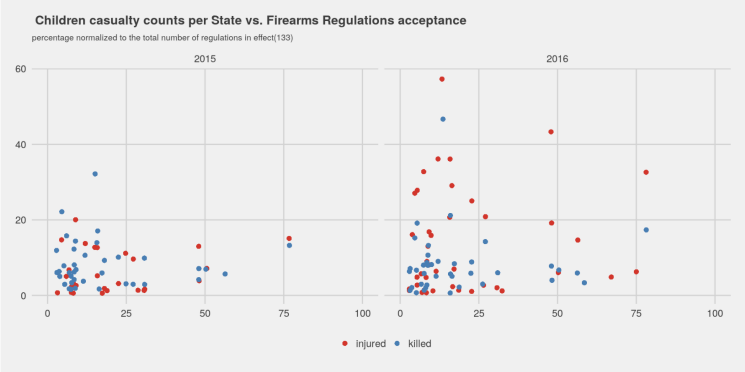
From ”Why gun violence increases in Texas” written by DATAI, we learn some innovative ways of generalization and prediction, that is to input some outside correlated data and combine it with our existing features. For example, DATAI utilizes Airbnb’s price per night in Texas and depression number to generalize the diagrams of potential reasons causing gun violence increases in Texas.[2]



From ”Gun Violence in USA Insights Forecast” written by Debadri Dutta, we learn that we can try to simply forecast how gun violence can increase or decrease in the future by generating plot.prophet graph. Therefore, when we try to predict possible gun violence rate in the future, we can take the forecasting graph as a reference and compare with our predicting results.[3]



Another similar data set is Gun violence database which archives U.S. gun violence incidents collected from over 2,000 sources[4]One of research using this set is called "Firearm regulations in the U.S." written by Jonathan Bouchet. This research mainly introduces about rules generated from past data and generalization of teens and children's casualty count per state vs firearm regularization acceptance rate.[5]



For this type of data, the state-of-the-art methods on trend are utilizing graph and diagram to describe and predict how data cluster. However, for our model, we focus on predicting if gun violence incident will happen at this area in a specific day, both to warn people of how safe living environment in different areas is and how imminent gun control situation is.
Moreover, the conclusions from existing work are similar to our own finding and we both give consent that the possibility of violence is proportional to year and increases within "the most dangerous cities".

5 Results

5.a Baseline

Our baseline is the intuitive idea that places with more gun violence history records are more likely to have such events. The following table is the top 5 cities with most gun violence history record in 2013-2018.

City	#Gun Violence
Chicago	8620
Baltimore	2994
New Orleans	2626
Philadelphia	2501
Washington	2335

Our baseline model reached an accuracy of 0.5217 with the padded dataset and separated training and validation split of ratio 1:1. It turns out that the gun violence incidents have relevance to the location and the criminal records are not sparsely distributed but clustering around most popular criminal places. However, though the location is a significant feature to our prediction task, making decisions only based on counting the number of history gun violence events is not sufficient. Therefore, we optimize our model by adding in the feature of event dates and add the feature of Jaccard similarity based on the intersection of gun violence locations on the different dates to our baseline model. In this way, we get our following Jaccard similarity model.

5.b Jaccard Similarity

In this model, we add the jaccard similarity between two dates into our baseline model and boost the accuracy to 0.9226 which is already within state of art. Here are some similarity results we get from our dataset: 1.0, 0.3333, 0.5556, 1.0, 0.25, 0.5, 0.1667 .

However, as we discussed above, the cosine similarity based on nearby business number and average price is not satisfied which only have accuracy of 0.4876 and 0.4872 which even cannot beat the baseline model. This is our failed attempt and we have discussed the possible reasons for this above.

5.c SVM and KNN and Random Forest Comparisons

Model		Date+GPS	Date+GPS+ PopularCrimeCities	Date(one-hot) +GPS	Date(one-hot)+GPS +PopularCrimeCities
KNN	K=1	0.5285	0.6355	0.9687	0.9848
	K=3	0.5310	0.6449	0.9753	0.9873
	K=10	0.5288	0.6737	0.9762	0.9833
SVM	C=0.1	0.4993	0.4993	0.8599	0.8961
	C=1	0.5006	0.6999	0.8612	0.9005
	C=10	0.4993	0.6999	0.8536	0.8976
RF	max depth = 2 n estimator=100	0.5496	0.6994	0.9529	0.9532
	max depth = 2 n estimator=200	0.5496	0.6994	0.9529	0.9603
	max depth = 10 n estimator=100	0.5569	0.6994	0.9781	0.9856

Table 1: Result Table for Each Model

From the table, we can see that KNN has the best performance on our dataset with an accuracy of up to 0.9873. Random forest also has relatively accurate scores of over 0.95 averagely on one hot encoded feature vector. SVM has lower performance especially on not encoded feature vector. From the table, we can also see that with one hot encoding on dates, the accuracy of each model boost to a great extent compared to not encoded feature vectors. And adding the check that if the city is among the most popular crime cities is also boost accuracy to some point but not as large as the one hot encoded features.

For each model, we experiment on a different crucial parameter. For KNN, with larger k value, the accuracy would also increase. This indicates the data distributions including the random padded data have overlaps across the two classes. This indicates the larger clustering class is, the better knn predicts on our prediction task. For SVM, the best C parameter which is the penalty is 1. For SVM, larger C value would give model more tolerance for the optimizer. For Random Forest, the parameter of max depth has a larger effect on the prediction result and larger max depth would increase the accuracy of the result. However, the number of estimators also increase accuracy on the last feature input but the improvement is tiny and limited.

References

[1] https://www.kaggle.com/shivamb/deep-exploration-of-gun-violence-in-us/notebook

[2] https://www.kaggle.com/kanncaa1/why-gun-violance-increase-in-texas/notebook

[3] https://www.kaggle.com/duttadebadri/gun-violence-in-usa-insights-forecast

[4] https://www.kaggle.com/gunviolencearchive/gun-violence-database/kernels

[5] https://www.kaggle.com/jonathanbouchet/firearm-regulations-in-the-u-s

[6] http://cseweb.ucsd.edu/classes/fa18/cse158-a/slides/lecture3.pdf

[7] https://scikit-learn.org/stable/modules/svm.html

[8] https://en.wikipedia.org/wiki/Random_forestAlgorithm

[9] https://cseweb.ucsd.edu/ jmcauley/datasets.html#clothing_fit

[10] https://www.quora.com/What-is-the-time-complexity-of-a-Random-Forest-both-building-the-model-and-classification

Acknowledgement

Thanks for professor Julian McAuley and TAs in the class who helped us on this project. We also appreciate the related literature and data analysis from shivamb, kanncaa1, duttadebadri, gunviolencearchive, jonathanbouchet on Kaggle.