

Mass Shooting Prediction Using Time Series

Hruthik Vinnakota

Department of Data Analytics, San Jose State University

DATA 270: Data Analytics Processes

Dr. Ming-Hwa Wang

May 17, 2023

Model Development

Forecasting mass shooting gun violence is essential for law enforcement in the United States of America so that they can adequately prepare for safety measures. It is crucial to forecast mass shooting gun violence in the USA. By forecasting gun violence, law enforcement can anticipate where and when shootings are likely to occur. They can also use the data to identify potential shooters and intervene before a tragedy occurs. The time series analysis method can be used for analyzing data on gun violence in the USA over a period of time to identify any potential causes or patterns. Predicting upcoming events of imminent mass shootings is commonly employed in time series analysis. To accomplish this, need to examine historical data on the use of firearms, as well as other factors like the number of victims and event dates, in order to determine whether any trends or patterns can be detected. By examining and deciphering these patterns, it is possible to forecast fatalities and ammo shootings on specific dates. Based on historical data, the research forecasts mass shootings in the United States.

The CRISP-DM approach was used to carry out the research. The data for the research was gathered from a couple of websites, including mother jones and the gun violence archive. The gun violence archive website provided the original data, which was dated from 2014 to 2020, with a sample of 16k rows. For the purpose of this study, data from the Mother Jones website, with a sample of 15k rows, was used for the years 2021 to 2022, managing missing values, anomalies, and redundant information was all included in data cleaning. The dataset initially contained 12 features, however after preprocessing and modification, it was reduced to 10. PCA was found to provide a means of reducing the dimensions of the model. To display the main data findings, exploratory data analysis was carried out. Finally, data were standardized for Machine learning modeling. In the end, Data were normalized for machine learning modeling.

Model Proposals

Prophet

Ning et al. (2022) This study compares three machine learning techniques—Auto-Regressive Integrated Moving Averages (ARIMA), Long-Short Term Memory (LSTM) recurrent neural network, and Prophet model—for time series oil production forecasting in the petroleum industry. Machine learning models in energy resources because of their cost-effectiveness in extracting the most value from large information created in subsurface reservoirs, assessment, drilling, completion, production, and other activities. the distinctions in exploration and production between supervised and unsupervised learning. Unlike standard empirical decline curve analysis and science-based reservoir simulation models, which are frequently employed for predicting the performance of unconventional oil reservoirs. In order to forecast time series oil production in the petroleum business, the Prophet model was contrasted with two other machine-learning techniques that specialize in modeling nonlinear time series data with seasonality effects on a daily, monthly, and annual basis. It was used to forecast future oil output from 65 wells in the Denver-Julesburg Basin (DJ Basin), and it performed better than previous methods because it could manage missing data and outliers, making the matching process nearly automatic. When compared to other models, the Prophet model excelled in terms of RMSE and MAE results and was able to anticipate the general declining trend as well as the seasonal impact. Additionally, it identified prospective seasonal effects that might be useful to operators beforehand to avoid any production changes the Prophet model performs better than the (ARIMA) and (LSTM) models in predicting the oil production trend with the least amount of error for a period of two years, followed by the LSTM model. As the Prophet model may do a

better job of capturing the seasonality effect, the ideal model may depend on whether or not there is seasonality in the production data.

Yang and Li (2022) The creation of a shared bicycle traffic forecast model is important for controlling shared bicycles, maintaining the availability of shared services, and enhancing user convenience and effectiveness. The Prophet and Bidirectional Long-Short Term Memory (BiLSTM) models, which are used to create a prediction model based on pre-processed data, are coupled using the least squares approach to acquire the best weights in order to increase prediction accuracy. The Prophet-BiLSTM combination model offers the best forecasting performance, according to tests and comparisons with single-item forecasting models and three widely used time series forecasting models. The Prophet model can incorporate seasonal elements into time series and fit cyclical fluctuations with the general trend. Piecewise linear and piecewise logistic regression are two popular methods.

piecewise logistic regression:

$$g(t) = \frac{C}{1 + e^{(-k(t-b))}} \quad (1)$$

$g(t)$ is the trend function, saturation value (C), growth rate (k), and offset (b) from equation (1). The saturation value is the maximum limit that a process can reach. It cannot grow beyond this limit. The growth rate determines how quickly the process approaches the saturation value. A higher growth rate means the process reaches its limit faster. From equation (2) the offset represents the starting or ending point in time for the process's growth. When the formula is graphed, there is an initial flat line until the offset (b) is reached. Then, the function starts increasing and eventually flattens out at the saturation value (C). The steepness of the curve is determined by the growth rate (k).

Piecewise linear regression:

$$g(t) = (k + a(t)^T \delta) * (t + (m + a(t)^T \gamma)) \quad (2)$$

The Linear Additive Model (LAM) is used by piecewise linear. The piecewise linear model divides the data into parts and draws straight lines for each part. It assumes that the data changes at different speeds in each part. It assumes that the relationship between the data is a straight line in each part. This model is used to analyze data that is not a straight line. The model was initialized with the holiday item set for significant holidays affecting bicycle traffic, changepoints set to 0.15, and changepoint_prior_scale set to 10. To get the best fitting effect, the predictions were assessed using evaluation measures like RMSE, which measures the difference between the observed value and the real value. Eventually, it was discovered that the model, which was created to forecast shared bicycle traffic, was more successful when it extracted periodicity from the data, factoring in vacation effects, as well as other influencing elements like weather and temperature. Bidirectional long-short-term memory (BiLSTM) was utilized to identify the temporal dependencies in the data, and the Prophet model was used to match the bicycle rental data to daily, weekly, and holiday periodicity components. For each model, the mean absolute error (MAE) and root mean square error (RMSE) are utilized as evaluation metrics.

Samal et al. (2019) Analyzed the efficacy of several analytics models in predicting air pollution levels, particularly in Bhubaneswar City, India, utilizing time series forecasting approaches due to the inadequacy of conventional linear regression methods. It emphasizes the serious effects of pollutants like RSPM, SO₂, NO₂, and SPM on human health, causing both acute and chronic effects. Various techniques, including Box-Jenkins, ARIMA, SARIMA, and Prophet model, emphasize the effectiveness of the SARIMA and Prophet models used for subsequent execution. Using Mean Squared Error (MSE) and Root Mean Square Error (RMSE)

incorporates the impacts of seasonal changes and holidays on air quality levels and assesses the model accuracy. With a wide confidence interval, the Prophet model was utilized to predict future pollution levels. The Prophet model was used to build an early warning system for public safety and can be extended to analyze healthcare data to establish health correlations with pollution levels in the future. The findings demonstrate that the Prophet model on log transformation is the most effective method for predicting future pollution levels. However, both the SARIMA and Prophet models offer good accuracy in this regard. To create a public safety early warning system that can be improved further by deep learning algorithms.

Pindiga (2022) focused on time-series forecasting to anticipate stock index using ARIMA and Facebook Prophet models, notably forecasting the Dow Jones Industrial Average (DJIA) to assist investors of the 30 businesses featured in it in making informed judgments examining the responses of the ARIMA and Facebook Prophet models to the forecasted DJIA index value and gathers data from 2009 to 2019 to forecast stock prices, verify forecasts, and create price visualizations. Due to the volatile nature of the stock market, close observation of the predictions is required. However, ARIMA forecasting produces fewer errors than Facebook Prophet offers insights on predicting stock market trends and can be used as a reference by investors and economic professionals. The Facebook Prophet is a tool designed to easily predict time series data at a large scale. The model is designed to easily predict time series data at a large scale, using a single variable of time to forecast results. According to equation (3) the prophet model has three main components includes three key components: trend, seasonality, and holiday effects. $y(t)$ represents a dependent variable that is the sum of four components: $g(t)$, $s(t)$, $h(t)$, and e .

$$y_t = g(t) + s(t) + h(t) + \varepsilon_t \quad (3)$$

The trend function is denoted by $g(t)$, the periodic function by $s(t)$, the holiday function by $h(t)$, and the error or noise by ϵ_t . T is the error term that accounts for any unusual changes that the model does not account for the ARIMA model performed better than the Facebook Prophet model in forecasting the Dow Jones Industrial Average index. Therefore, ARIMA is a better model to use for forecasting stock market trends, especially for short-term predictions. However, they also caution that the stock market is critical and requires careful observation of predictions, regardless of whether the model used gives information on predicting stock market trends and can be used as a reference by investors and economic analysts. Prophet employs several kinds of characteristics to regulate the model's behavior and the accuracy of the forecasts. Prophet uses a number of significant factors, including:

Growth. Prophet's preference for the trend model depends on this setting. Depending on the type of trend you wish to model, it can be set to linear, logistic, or custom. The linear model is a straight line, meaning that the rate of change is constant over time. The logistic model is an S-shaped curve, meaning that the rate of change increases or decreases over time. The custom model allows you to specify a trend function, allowing you to more accurately model complex trends.

Seasonality Mode. This setting regulates the kind of seasonality model that the Prophet employs. Depending on whether the seasonality is additive or multiplicative, it can be set to either an additive or multiplicative calculation. In an additive seasonality model, the seasonality is added on top of the trend to model the time series. Whereas in a multiplicative seasonality model, the seasonality affects the trend by multiplying it. Thus, the trend is adjusted depending on the seasonality of the data.

Holidays. The time series is predicted to be affected by the holidays listed in this parameter. The names of the holidays and the dates they fall on can be used to identify them. Prophet will automatically modify the forecasts to take these holidays into consideration. It is important to note that holidays must be specified in advance for Prophet to adjust for their effects on the forecast, as Prophet does not have the ability to detect them automatically.

Changepoints. The dates when the trend, seasonality, or holiday effects are anticipated to alter in the time series are specified by this option. Prophet will automatically modify the model to take these changes into consideration and generate more precise forecasts.

Uncertainty samples. This parameter defines the number of samples to be used when computing uncertainty intervals for the forecasts. Higher values will bring more accurate uncertainty intervals, but they will also increase the computation time significantly.

In general, Prophet employs a number of parameters to regulate the model's behavior and the accuracy of the forecasts. These variables can be changed to suit the unique properties of the time series data and the user's forecasting requirements. By adjusting the parameters, users are able to create models that can better capture the underlying patterns of the data and produce more accurate forecasts. Furthermore, this flexibility allows Prophet to be used for a wide range of forecasting problems, such as short-term and long-term forecasting, as well as forecasting at different levels of granularity.

Model Supports

The approach for analyzing time series data from mass shootings was trained on a local machine MacBook Pro. The processor used was an Intel Core i7 with 16 GB of RAM. The GPU used was an NVIDIA GeForce GPU with 8 GB of dedicated video memory. The operating system macOS High Sierra makes it suitable for tasks consisting of machine learning models. As

a result, The MacBook Air M1 is equipped with a powerful M1 chip, making it capable of handling complex computing tasks with ease. It was necessary for the demand forecasting model to be trained on a MacBook Air M1 running macOS Big Sur, the most recent version of Apple's operating system, because it features Core ML and Create ML, among other machine learning-focused applications and libraries. These frameworks allow developers to quickly and efficiently develop and optimize machine learning models for a variety of tools, including demand forecasting.

Google Colab is a free cloud-based platform that enables users to easily create, edit, and share notebooks containing live code, equations, visualizations, and narrative text and to develop machine learning models with the help of powerful computing resources such as GPUs and TPUs. It is also used to host Jupyter notebooks with zero setup/configuration. Moreover, supports collaboration with other users, allowing for easier sharing of research results. Furthermore, it provides a variety of data sets for use in this research project and makes running code in the cloud quick and easy. That can be used to build powerful, intelligent applications. Additionally, it provides access to a wide range of cloud services and computing resources, allowing specialists to manage the project efficiently and securely. Its user-friendly interface makes it an appealing platform for machine learning experts and makes it simple to build and execute code.

Table 2

The following is a list of the procedures and tools used in this study

	Library	Procedure	Operation
Pandas	Panda	Dataframe	Data manipulation,
	Panda.plotting	operations, data cleaning	analysis and visualization
Numpy	numpy	Mathematical	Used for
		approaches	computations in mathematics and numerical calculations.
Sklern	sklearn.linear_model	StratifiedKfold,	Machine learning
	Sklearn.model_selection	train, test, StandardScaler	algorithms, model selection, data preprocessing
Matplotlib	matplotlib.pyplot	pyplot	used for tracking the plots and visualizing data.
Seaborn	Seaborn	seaborn	Data visualization, statistical plotting.
Dateutil	Dateutil.parser	parser	Date parsing and manipulation.

	Library	Procedure	Operation
Prophet	Performance_metrics	Time series forecasting.	Accurate time-series predictions.

Model Architecture and Data Flow

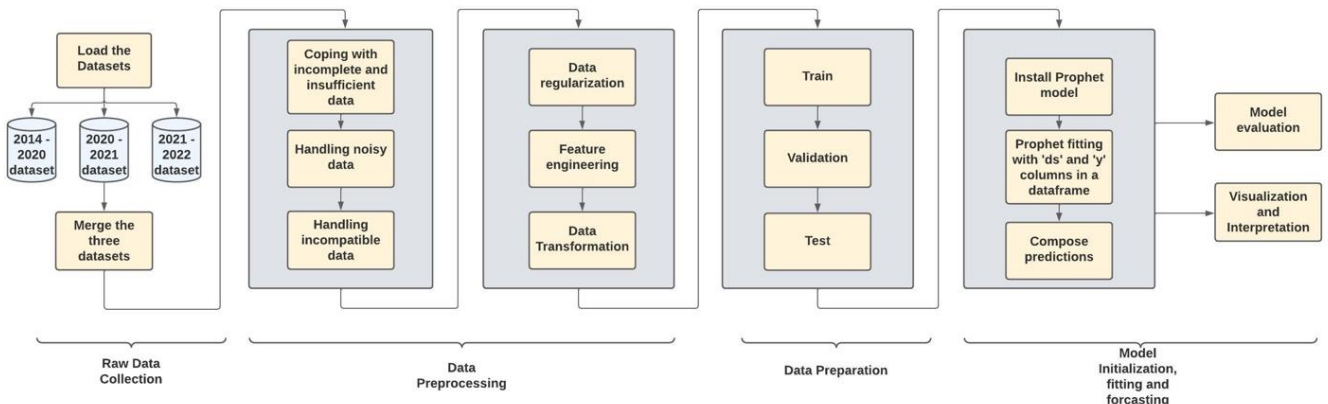
The datasets utilized in this study were obtained from the Gun Violence Archive website for the years 2014 through 2022, while the Mother Jones websites were used for the years 2021 and 2022. It was combined after loading the three datasets by using a union statement. The EDA process included identifying patterns and trends in the data using tools such as Pandas and data visualization to identify and remove anomalies, inconsistencies, and noisy data. The process also included identifying outliers, and data points that do not fit the general pattern of the data, and discarding them. Finally, EDA allowed for data cleaning by removing noise from the data set, which improved the accuracy of the analysis. Data preprocessing is an important step in this study. It helped to improve the accuracy and performance of the model by transforming raw data into a more suitable format, which created new features from existing data and reduced the number of features. This helped to reduce complexity and improved the generalization capability of the model. The Data preparation involved the preprocessed datasets were segregated into three sets for analysis, 60% allocated for training, and 20% allocated for both validation and testing.

The datasets were used to demonstrate the value of data quality and labeling for the prophet model, as well as the steps involved in creating reliable and precise predictions. The Prophet Library is used to forecast time series by feeding the data into the model using the fit method. Future time points were represented by "ds" and "y". expected values were represented by "yhat," and their corresponding lower and higher bounds were represented by "yhat_lower"

and "yhat_upper." The actual numbers of casualties over time were represented by values for metric evaluations. The plot was labeled with the relevant axis labels, a title, and a legend. Each category was plotted using a distinct color virtually in Figure 1 which was elaborated in a detailed structure

Figure 1

Representation of Data flow diagram



Model Comparison and Justification

Prophet is a machine learning technique used for forecasting future trends and patterns in time series data. It is particularly well-suited for forecasting future mass shootings. Prophet is a forecasting library that uses an additive model and works best with time series that have a consistent pattern. It combines multiple trends into a single model and can detect anomalies, trend changes, and seasonality. It also has built-in components for dealing with holiday effects, allowing it to identify and predict the impacts of holidays on time series data. However, in this study, the holiday effect was not performed due to the data which was used. Firstly, compared to other time series models called Long Short-Term Memory (LSTM), Prophet is designed to be simpler and more intuitive. It also has the ability to automatically detect the trend, seasonality, and holidays of a time series, making it an ideal choice for those who don't have much

experience with time series analysis. Secondly, Prophet models are linear models that make use of time series data. It was created to deal with time-series data, one of the data types the models were managed and cannot be used with other data kinds like tabular, image, audio, or text.

Thirdly, the data were made more understandable for the model by the preprocessing stage by reducing the dataset size helped to decrease the complexity of the model and made it easier to train in which data wrangling and transformation were also used to improve the accuracy of the model by ensuring the data is consistent and in the correct format. Additionally, Prophet was a flexible model capable of adjusting to changes in the data and generated precise forecasts even in the presence of missing values or observations that do not fit the pattern. So, was found to be less susceptible to underfitting and overfitting than LSTM models, and it was found that the Prophet model could function effectively with sparse data. Furthermore, Prophet was more accurate than traditional time series models and can be used to forecast multiple time series at once. Eventually, able to capture seasonality and non-linear trends in data more accurately than other models. This makes it a powerful and useful tool for forecasting.

The LSTM models were linear models created to handle time series data and required metric evaluation to operate at their best. While the prophet model was considered non-seasonal factors whereas the seasonal component of the LSTM model was able to capture the cyclical nature of data, such as monthly mass shooting patterns or daily. LSTM had a smaller chance of erroneous predictions because of a valid generalization, and it required less time and processing power to perform. However, LSTM dealt with underfitting and signs of overfitting. In addition, LSTM was specifically designed for time series data and cannot handle other types of data such as tabular, image, audio, or text where data points gathered at regular intervals of time, such as daily, monthly, or yearly assessments, were referred to as time series data. In addition, the LSTM

model performed well for small to medium-sized data sets, but they could need assistance to be effective when dealing with missing or sparse data. Furthermore, the LSTM model performed well for medium-sized to big datasets and have higher computational complexity. Likewise, it required a moderate amount of time for preprocessing and training, although LSTM models took longer to train than the prophet model because of the difficulty of accounting for seasonality and whereas both models have a high space complexity since they need to store the complete time series as well as the model parameters. Moreover, the amount of preprocessing needed and the quantity of the dataset both had an impact on how sophisticated the models were. Therefore, different tasks like data exploration, data wrangling, and data transformation were carried out prior to fitting the models in this study effort. In the end, LSTM performed well compared to prophet models due to taking seasonality into account.

Table 2

Evaluating and contrasting the models

Characteristic	Prophet	LSTM
Fundamental	Time series	Time series
Architecture		
Types of data	Linear and time series	Is appropriate for time series data for seasonal
Data size	Any size	Any size
Computational complexity	High	High

Characteristic	Prophet	LSTM
Space complexity	Varies depending on the dataset, but is often low	Varies depending on the dataset, but is often low
Preprocessing time	Reasonable	Reasonable
Training period	Faster	Faster
Rate of learning	N/A No gradient descent	N/A No gradient descent
Limitations	Requires domain knowledge	domain knowledge not required
Strengths	easy-to-use	Complicated

Model Evaluation methods

The problem at hand involves applying time series forecasting techniques to accurately classify mass shootings based on historical data from previous years. In the following, you will find a comprehensive analysis and examination of the metrics that were employed to assess the performance and effectiveness of the models. The discussion delves into the various evaluation criteria that were utilized to evaluate the models' Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and overall quality in forecasting. The number of data points in the dataset, "n" as well as the real values and predicted values for each data point, "actual_value" and "predicted_value," respectively, are all taken into account by these measures.

Root-Mean-Squared Error (RMSE)

Determines the average amount of forecasting errors in a group of predictions. It measures the differences between the estimated values and the actual values and then takes the square root of the average of the squares of these differences. It is typically calculated as the

mean absolute error (MAE), which is the average of the absolute values of the errors. This metric indicates the forecasts' level of accuracy. RMSE is frequently used since it has the same units as the observed and predicted values and remains simple to read. When comparing predictive models, the RMSE is a helpful metric since it measures the average size of mistakes in a group of predictions. It provides a better depiction of the performance of the model by penalizing significant errors more severely than small ones. It can be challenging to use in some mathematical applications because it is sensitive to outliers and uses the square root approach.

$$RMSE = \sqrt{\frac{1}{n} * \sum (y - y_i)^2} \quad (4)$$

The equation (4) for RMSE is the square root of the sum (\sum) of the squared difference between the predicted values “ y_i ” and the actual values” y ”, divided by the “ n ” total number of observations. Comparing the evaluation metrics in order to identify whether the model is overfitting or underfitting. By analyzing the performance measures and contrasting them to determine if the model is capturing too much noise and irrelevant patterns (overfitting) or failing to capture the underlying patterns and relationships in the data (underfitting).

Mean Absolute Error (MAE)

measures the average deviation from the forecasts in a set. the sum of the absolute difference between the predicted and observed values and dividing it by the number of observations. This statistic can be used to assess a forecast's accuracy and the effectiveness of a forecasting model. Due to the fact that MAE is a measurement of the average magnitude of the errors in a series of predictions, without taking into account their trajectory. Measures the accuracy of predictions regardless of the size of the true values, making it a useful metric for forecasting values regardless of their numerical scale. However, the errors might grow immeasurable large with extremely small actual values, which can be troublesome.

$$MSE = \frac{1}{n} * \sum (y - y_i) \quad (5)$$

Equation (5) is derived by the average of the absolute differences between the actual values and the predictions. The total number of data points is represented by n, and the sum (\sum) is calculated across all of the data points. The MAE measure is useful for evaluating the prediction ability of models since it can be used with perpetual quantities.

Model Validation and Evaluation

Prophet

In Prophet, predictive models were rapidly and readily created using the sklearn model API. These models may then be used to anticipate future trends and behaviors. To determine the data's prophet model, trials were conducted with various algorithms and parameters. The fit method is used to fit the model to the data, while the predict method is used to generate predictions based on the fitted model. The "ds" column should contain the date, of the data points and the "y" column contained the corresponding values of the victim's column. The target metric in the y column must be a numerical value such as an integer, float, or decimal. This is because the metric was used to measure the success of a particular outcome or action. The data frame also contained a column labeled "y" that contains the values of the target variable for the corresponding dates in the "ds" column. This column is used to train a predictive model, which is then used to generate predictions for the future values of the target variable. A data frame is created, including the dates for the predictions to be made in the "ds" column. This will enable us to assess the accuracy of the model by comparing the predictions to actual historical data. Additionally, the inclusion of historical dates will also help us to identify any potential issues or discrepancies in the model. The original forecasted data represents the expected values, while the lower and upper intervals represent the range of values within which the true value is likely to

fall. This range was calculated by taking into account the uncertainty associated with the forecast. The prophet model was initialized with the seasonality mode set to 'multiplicative'. Deseasonalized victims' data (column: 0) up to '2019-12-29' is selected, and the data frame is prepared for training. The model is fitted to the deseasonalized victims' data. For future data preparation DataFrame is created using the `make_future_dataframe()` function with a period of 26 weeks and a frequency of weekly. Moreover, for forecasting and reconstruction, the model is used to predict the values for the future time points specified in the future DataFrame where the forecasted values are multiplied by the corresponding seasonal factors from '2020-01-05' to '2020-06-28' to reconstruct the original scale of the victim's data and at last separate columns ('yhat', 'yhat_lower', 'yhat_upper') were created in the forecast DataFrame to store the reconstructed values. Eventually, The `prophet_pred_df` DataFrame was displayed, by showcasing the forecasted values ('yhat', 'yhat_lower', 'yhat_upper') and the corresponding actual values in Figure 2. Yhat_lower values represented the accuracy of the model, with lower values indicating higher accuracy this means that the current model was more accurate.

Figure 2

Date-wise results in order

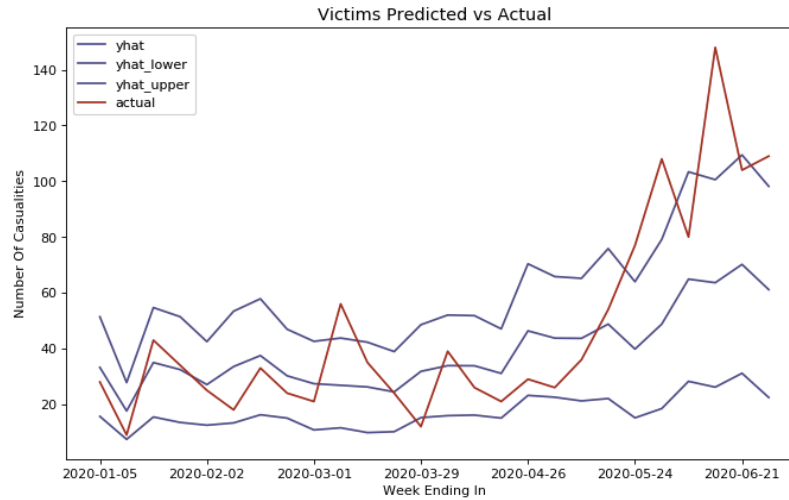
	ds	yhat	yhat_lower	yhat_upper	actual
0	2020-01-05	33.225	15.6673	51.3787	28
1	2020-01-12	17.6115	7.42128	27.7946	9
2	2020-01-19	34.9691	15.4503	54.678	43
3	2020-01-26	32.4499	13.4915	51.4128	34
4	2020-02-02	27.0656	12.5031	42.4686	25
5	2020-02-09	33.539	13.3253	53.3895	18
6	2020-02-16	37.471	16.2405	57.8449	33
7	2020-02-23	30.2083	15.037	46.9038	24

From the plot, the values compared the predicted values with the actual values over time.

However, it was observed that the predicted values exceeded the upper and lower boundaries in 5 out of the 26 predictions in Figure 3. This indicates that the model's predictions were less reliable and suggested that further refinement or evaluation was necessary

Figure 3

Comparison between predicted and actual values



In comparison to other study-wide evaluation methods. 86% accuracy is a low performance for the prophet model. While the LSTM provided the overall best accuracy.

Table 3

Comparing of various models utilized in the research paper

Model	RSME	MAE
Prophet	64.86	86.49
LSTM	25.36	34.45

Conclusion

The study aims to use time series analysis to forecast mass shooting gun violence in the United States. The research was carried out using the CRISP-DM approach, and the data was pre-processed and cleaned extensively. The study uses the Prophet model and compares its effectiveness in predicting future mass shootings with other models like the LSTM model. Moreover, the Prophet time series model is one of the simpler to use and comprehend. However, LSTM provides more accurate forecasts when compared to Prophet. Eventually, the findings can help law enforcement in the United States anticipate where and when shootings are likely to occur and create plans to increase security in those areas.

Limitations

The study is limited by the availability and quality of data. The data gathered could be incomplete, and the study relies on the accuracy of the sources used. Additionally, the Prophet model may not always produce accurate results and needs to be updated with new data for better accuracy.

Future scope

The study can be expanded to include more recent data and additional factors that can impact mass shootings, such as mental health conditions and past criminal records of potential shooters. The study can also be applied to forecasting other types of gun crimes and violence.

References

- Ning, Y., Kazemi, H., & Tahmasebi, P. (2022). A comparative machine learning study for time series oil production forecasting: ARIMA, LSTM, and Prophet. *Yanrui Ning a, *, Hossein Kazemi a, Pejman Tahmasebi, 164, 105126. <https://doi.org/10.1016/j.cageo.2022.105126>*
- Pindiga, S. N. (2022). Time-Series Forecasting: Predicting Stock Index Using Arima and Facebooks Prophet Model. *International Journal for Research in Applied Science and Engineering Technology, 10(6)*, 4832–4839. <https://doi.org/10.22214/ijraset.2022.45073>
- Samal, K. K. R., Babu, K. S., Das, S. K., & Acharaya, A. (2019). *Time Series based Air Pollution Forecasting using SARIMA and Prophet Model. <https://doi.org/10.1145/3355402.3355417>*
- Yang, G., & Li, H. (2022). *Prediction method of shared bicycle traffic based on Prophet-BiLSTM combined model. <https://doi.org/10.1145/3582197.3582239>*