

APPENDIX A

SAMPLE SCREEN SHOTS

To start with, first we need to login into the Google Collaborator platform. The Google Collaborator platform is an open source platform and can be accessed by any user. The user can login with their Gmail account. Figure A1 shows the login page of Google Collaborator where the user can login to the website using “sign in” option.

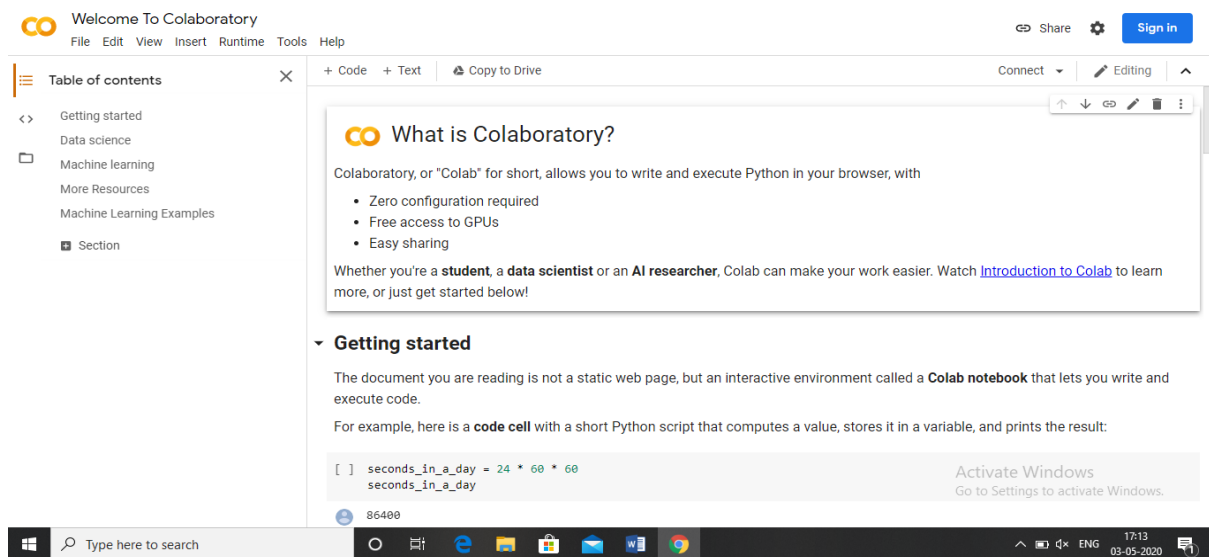


Figure A1 Google Collaborator login page

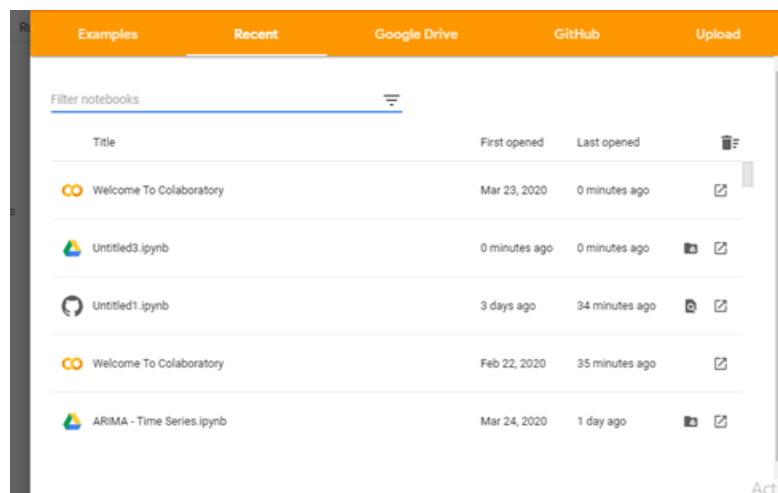


Figure A2 Google Collaborator Files

The Figure A2 shows the files available after the signing process. The user can create a new file by clicking on “new notebook” option.

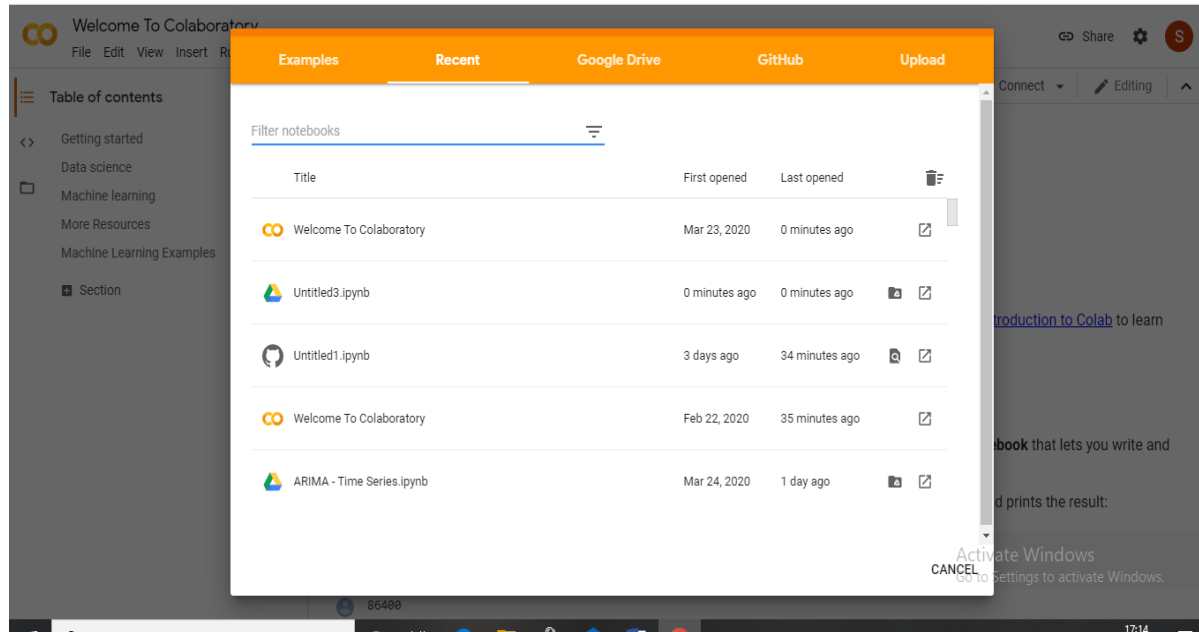


Figure A3 Google Collaborator Files

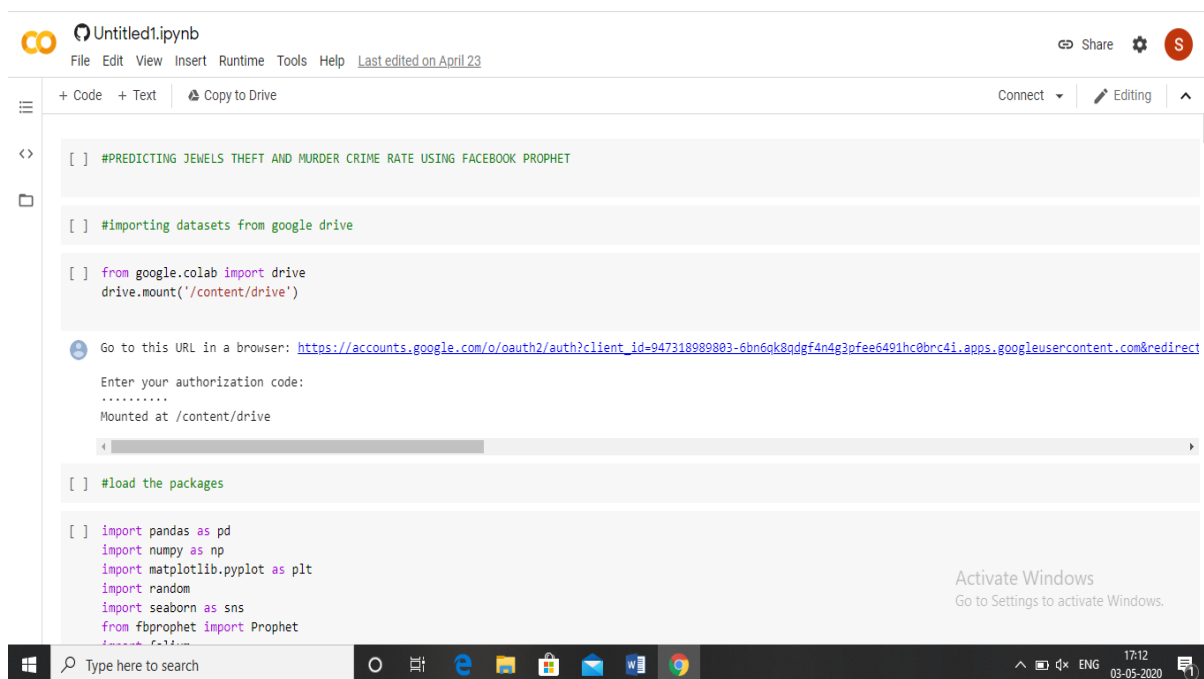


Figure A4 Google drive mount

Figure A4 represents the Google drive mount. The crime datasets are stored in Google drive which contains nearly 10 L crime records. Therefore, to access these records we need to mount them using authentication code.

```

[ ] #load the packages

[ ] import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import random
import seaborn as sns
from fbprophet import Prophet
import folium
import math
from folium.plugins import HeatMap
import folium
plt.style.use('seaborn')
%matplotlib inline
import warnings
warnings.filterwarnings('ignore')

/usr/local/lib/python3.6/dist-packages/statsmodels/tools/_testing.py:19: FutureWarning: pandas.util.testing is deprecated. Use the functions in the public API at
import pandas.util.testing as tm

```

Figure A5 Loading packages

```

df_0 = pd.read_csv('/content/drive/My Drive/df_0.csv', error_bad_lines=False)

[ ] #Concat all the dataset into one

[ ] df = pd.concat([df_1, df_2, df_3, df_4, df_5, df_6], ignore_index=False, axis=0)
df.head()

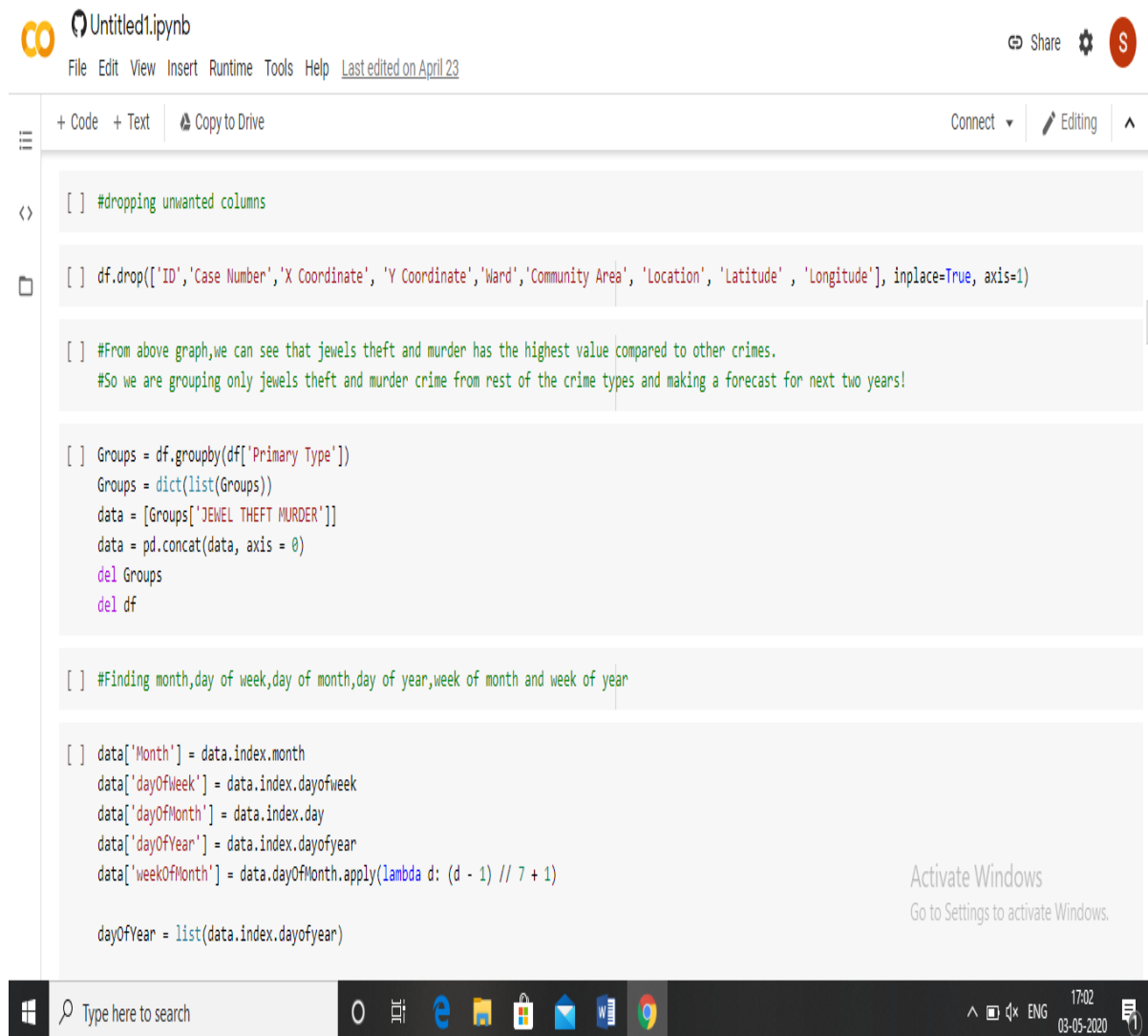
```

| | ID | Case Number | Date | Primary Type | Location Description | Arrest | District | Ward | Community Area | X Coordinate | Y Coordinate | Year | Latitude | Longitude | Location |
|---|----------|-------------|------------------------|----------------------------|----------------------|--------|----------|------|----------------|--------------|--------------|------|-------------|--------------|-------------------------------|
| 0 | 11149297 | JA506752 | 01-01-2014 00:01 | CRIMINAL SEXUAL ASSAULT | RESIDENCE | False | 25.0 | 26.0 | 23.0 | NaN | NaN | 2014 | NaN | NaN | NaN |
| 1 | 10394673 | HX561124 | 07-05-2014 00:00 | OFFENSE INVOLVING CHILDREN | RESIDENCE | False | 25.0 | 31.0 | 22.0 | NaN | NaN | 2014 | NaN | NaN | NaN |
| 2 | 12032628 | JD211529 | 09-01-2014 14:25 | DECEPTIVE PRACTICE | RESIDENCE | False | 7.0 | 16.0 | 67.0 | NaN | NaN | 2014 | NaN | NaN | NaN |
| 3 | 11895409 | JC515153 | 11/14/2014 08:00:00 AM | CRIMINAL SEXUAL ASSAULT | RESIDENCE | False | 16.0 | 31.0 | 15.0 | NaN | NaN | 2014 | NaN | NaN | NaN |
| 4 | 9863337 | HX513133 | 11/19/2014 09:20:00 PM | BATTERY | ALLEY | False | 24.0 | 50.0 | 2.0 | 1156959.0 | 1945011.0 | 2014 | 42.00488511 | -87.69788912 | (42.004884982, -87.697889126) |

Figure A6 Loading of datasets

Figure A5 represents the loading of packages required to perform this project. The necessary packages are imported such as numpy, pandas, Prophet, seaborn for system implementation.

Figure A6 represents the loading of datasets. The crime datasets contain a summary report of crimes from 2014 to 2019. All these records are concatenated into one csv file for better use.



```
[ ] #dropping unwanted columns

[ ] df.drop(['ID','Case Number','X Coordinate','Y Coordinate','Ward','Community Area','Location','Latitude','Longitude'], inplace=True, axis=1)

[ ] #From above graph,we can see that jewels theft and murder has the highest value compared to other crimes.
    #So we are grouping only jewels theft and murder crime from rest of the crime types and making a forecast for next two years!

[ ] Groups = df.groupby(df['Primary Type'])
    Groups = dict(list(Groups))
    data = [Groups['JEWEL THEFT MURDER']]
    data = pd.concat(data, axis = 0)
    del Groups
    del df

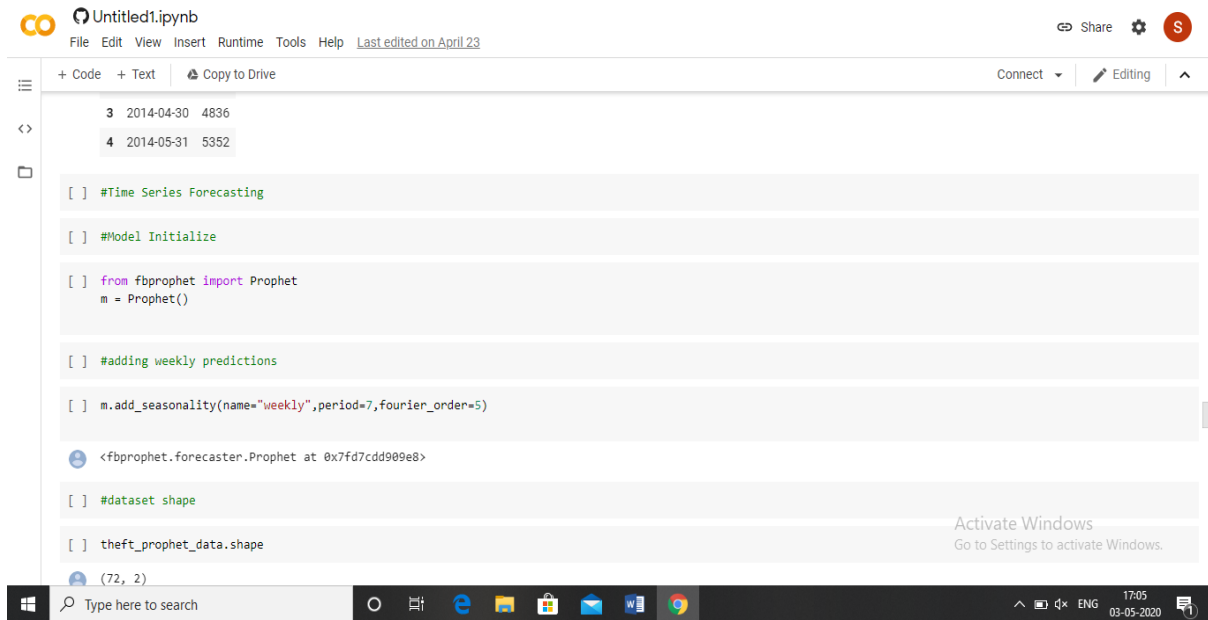
[ ] #Finding month,day of week,day of month,day of year,week of month and week of year

[ ] data['Month'] = data.index.month
    data['dayOfWeek'] = data.index.dayofweek
    data['dayOfMonth'] = data.index.day
    data['dayOfYear'] = data.index.dayofyear
    data['weekOfMonth'] = data.dayOfMonth.apply(lambda d: (d - 1) // 7 + 1)

    dayOfYear = list(data.index.dayofyear)
```

Figure A7 Jewel theft murder crimes

Figure A7 represents the grouping of jewel theft murder crimes. The crime datasets contain many crime types but since our project focuses only on jewel theft murder crimes, we are grouping them.



The screenshot shows a Jupyter Notebook titled 'Untitled1.ipynb' with the following code cells:

```
[ ] #Time Series Forecasting
```

```
[ ] #Model Initialize
```

```
[ ] from fbprophet import Prophet
    m = Prophet()
```

```
[ ] #adding weekly predictions
```

```
[ ] m.add_seasonality(name="weekly",period=7,fourier_order=5)
```

```
<fbprophet.forecaster.Prophet at 0x7fd7cdd909e8>
```

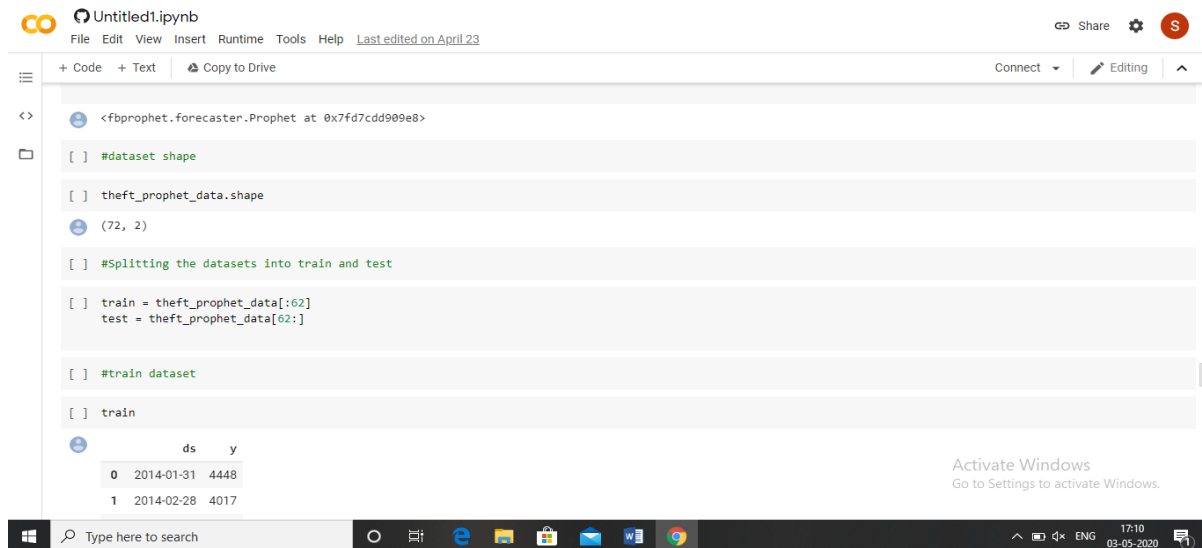
```
[ ] #dataset shape
```

```
[ ] theft_prophet_data.shape
```

```
(72, 2)
```

The output of the last cell shows the shape of the dataset as (72, 2). The Windows taskbar at the bottom shows the time as 17:05 on 03-05-2020.

Figure A8Model Initialize



The screenshot shows a Jupyter Notebook titled 'Untitled1.ipynb' with the following code cells:

```
<fbprophet.forecaster.Prophet at 0x7fd7cdd909e8>
```

```
[ ] #dataset shape
```

```
[ ] theft_prophet_data.shape
```

```
(72, 2)
```

```
[ ] #Splitting the datasets into train and test
```

```
[ ] train = theft_prophet_data[:62]
    test = theft_prophet_data[62:]
```

```
[ ] #train dataset
```

```
[ ] train
```

```
ds      y
0  2014-01-31  4448
1  2014-02-28  4017
```

The output of the last cell shows the first two rows of the training dataset. The Windows taskbar at the bottom shows the time as 17:10 on 03-05-2020.

Figure A9 Train test data

Figure A9 represents the splitting of train test data. 90 % of the data is used for training and 10% of data is used for testing.

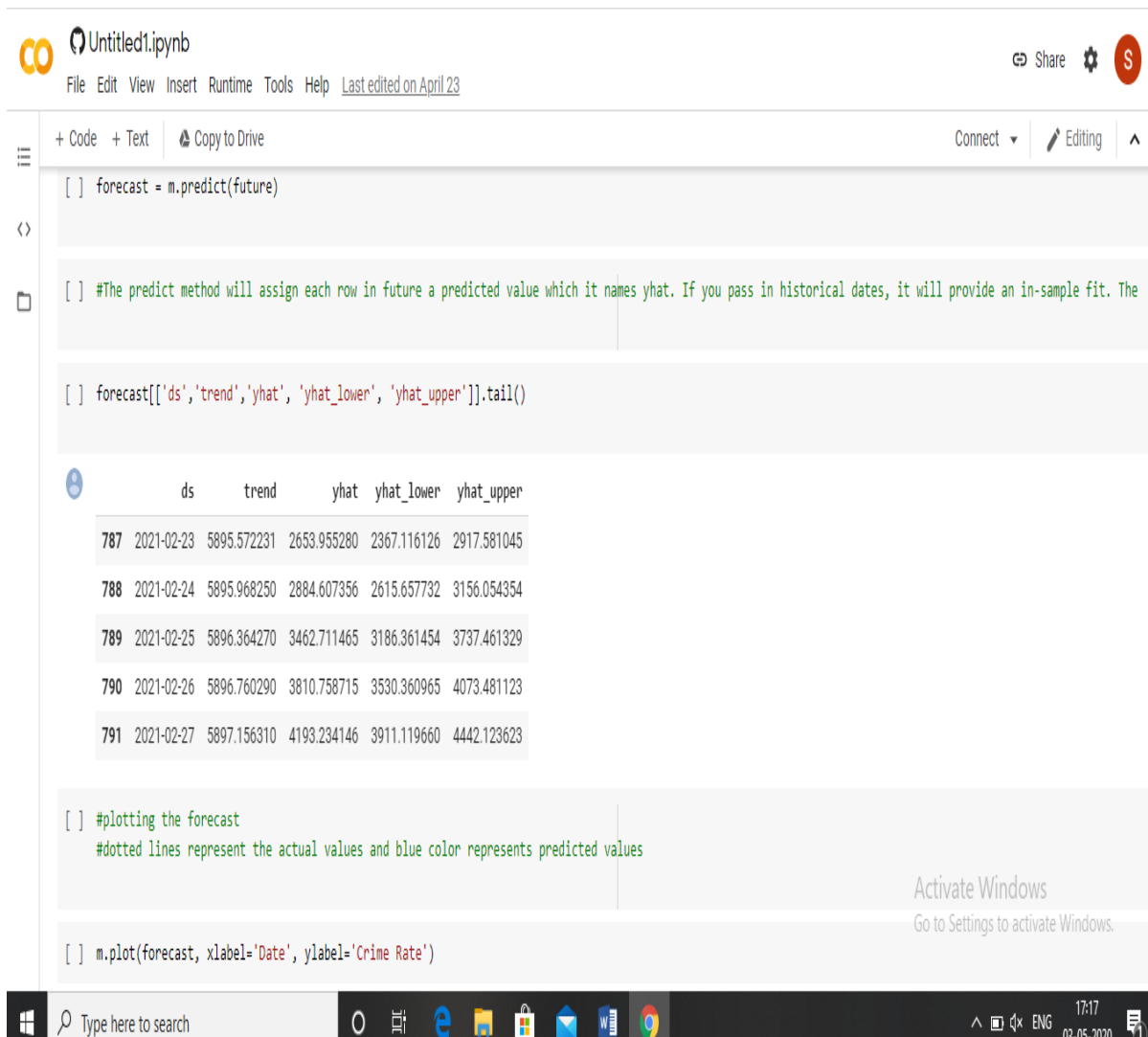


Figure A10 Forecasted values

Figure A10 represents the predicted values (yhat) using time series forecasting. The forecasting is done for training data and it produces forecasted values for next two years of future.

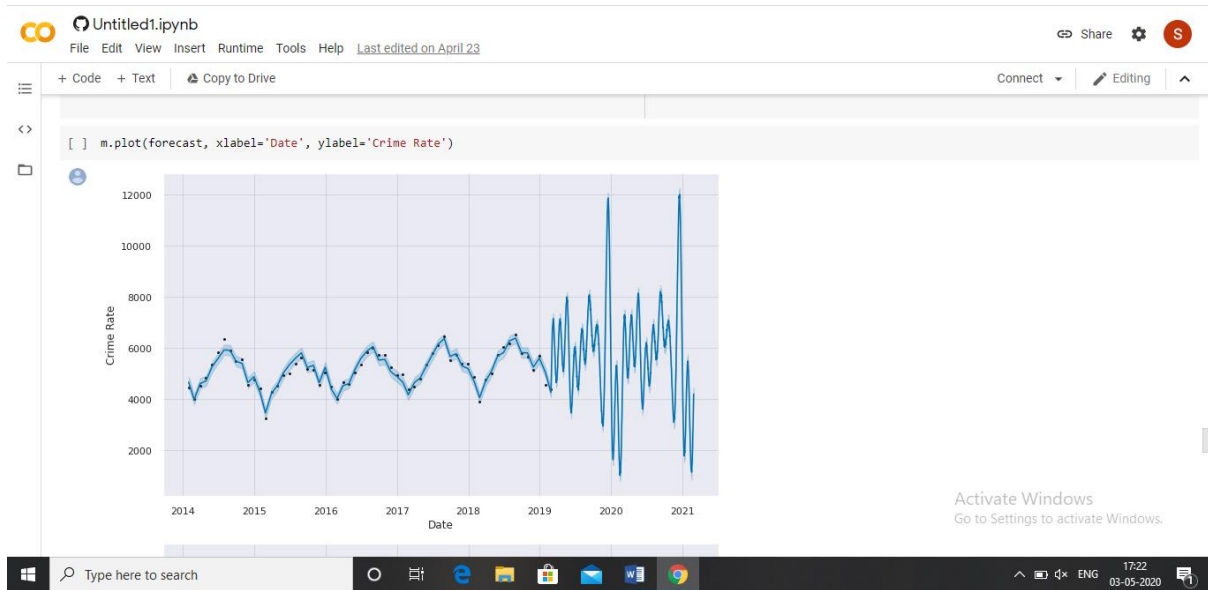


Figure A11 Visualization of forecasted values

Figure A11 represents the visualization of forecasted values. The black dotted lines represent the actual values and blue lines represent the predicted values. We can see that the actual values quite matches with the forecasted values and it also foreseeing into the future producing crime rates.

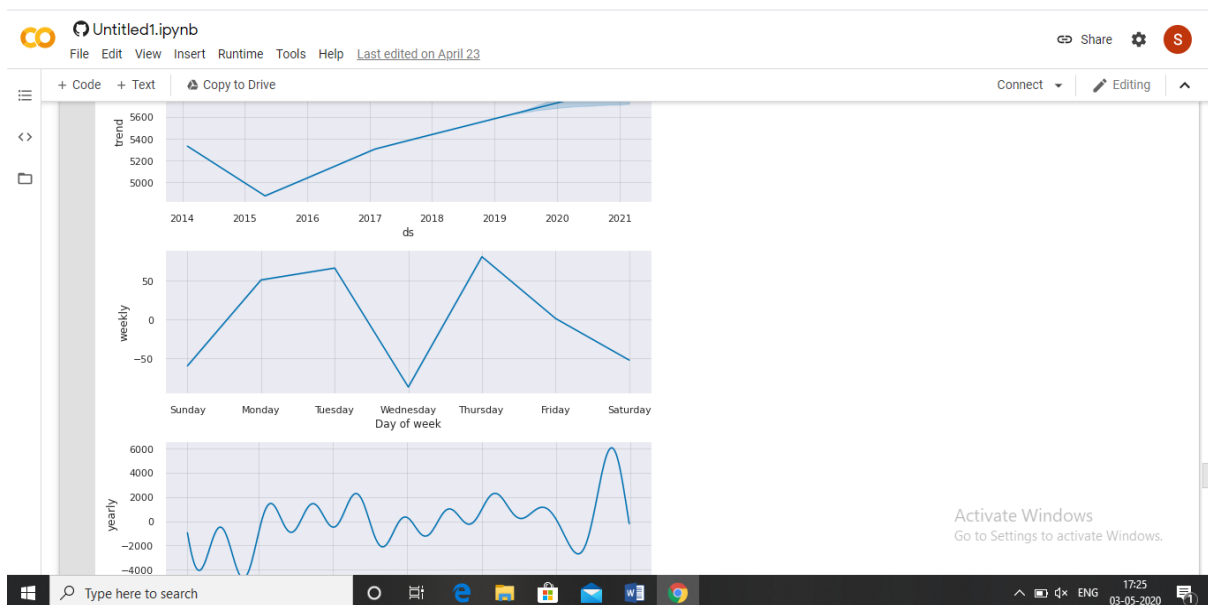


Figure A12 Forecasting components

Figure A12 represents the visualization of forecasting components. It gives an insight analysis about the trend, weekly seasonality and monthly seasonality.

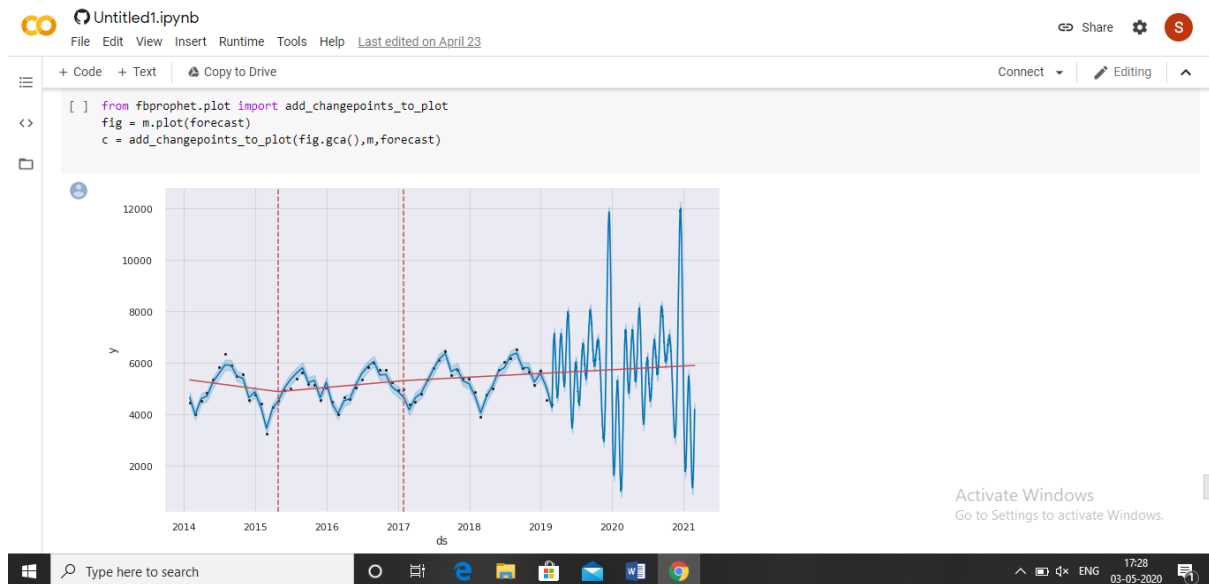


Figure A13 Trend change points

Figure A13 represents the visualization of trend change points. The red line shows the trend change point at which the rate is allowed to change.

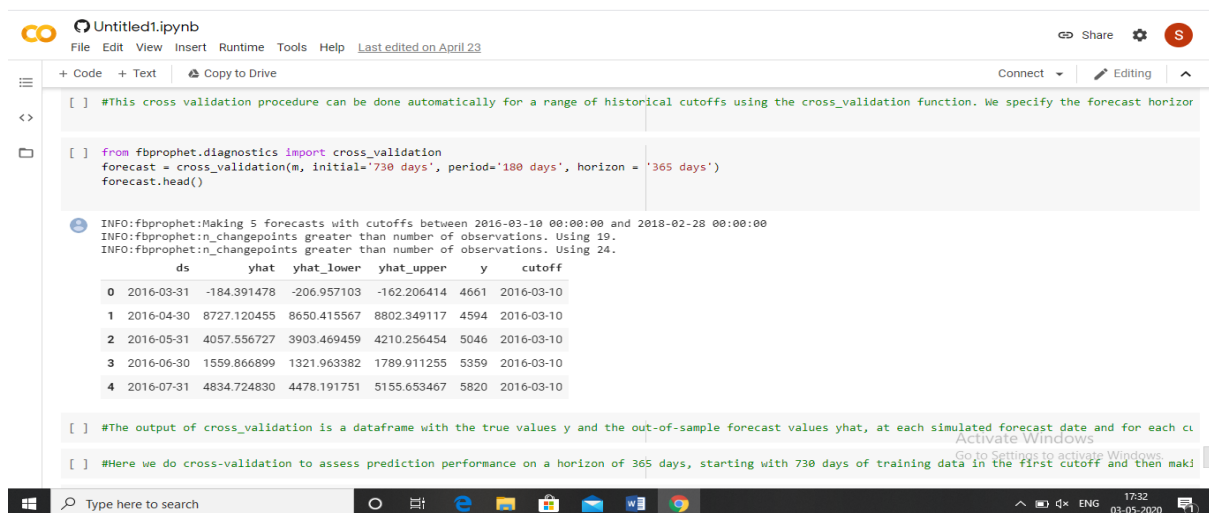


Figure A14 Model evaluation


```
[ ] #This cross validation procedure can be done automatically for a range of historical cutoffs using the cross_validation function. We specify the forecast horizon
```

```
[ ] from fbprophet.diagnostics import cross_validation
forecast = cross_validation(m, initial='730 days', period='180 days', horizon = '365 days')
forecast.head()
```

INFO:fbprophet:Making 5 forecasts with cutoffs between 2016-03-10 00:00:00 and 2018-02-28 00:00:00
INFO:fbprophet:n_changepoints greater than number of observations. Using 19.
INFO:fbprophet:n_changepoints greater than number of observations. Using 24.

| | ds | yhat | yhat_lower | yhat_upper | y | cutoff |
|---|------------|-------------|-------------|-------------|------|------------|
| 0 | 2016-03-31 | -184.391478 | -206.957103 | -162.206414 | 4661 | 2016-03-10 |
| 1 | 2016-04-30 | 8727.120455 | 8650.415567 | 8802.349117 | 4594 | 2016-03-10 |
| 2 | 2016-05-31 | 4057.556727 | 3903.469459 | 4210.256454 | 5046 | 2016-03-10 |
| 3 | 2016-06-30 | 1559.866899 | 1321.963382 | 1789.911255 | 5359 | 2016-03-10 |
| 4 | 2016-07-31 | 4834.724830 | 4478.191751 | 5155.653467 | 5820 | 2016-03-10 |

Figure A15 Model evaluation values

Figure A14 and A15 represents the model evaluation. But A15 gives the clear output of the cross validation with the values of the forecasted data. The model built is tested for errors using cross validation performance metrics which proves efficiency and nature of the model.

APPENDIX B

SAMPLE CODING

In this appendix the sample coding which are used for our implementation process and analysis process for the forecasting the crime scene in jewel theft murder are documented. First starts with the initialization of the model which is the forecasting model, face book prophet tool used. Face book prophet model is initialized code is mentioned below. Followed by the splitting the testing and training data for the model evaluation, model fit method, future data, forecasted values, visualization of the forecasting crime rate in jewel theft murder, testing of data and the trend change points, last the forecasting model evaluation code is mentioned.

#Model Initialize

```
from fbprophet import Prophet  
m = Prophet()  
m.add_seasonality(name="weekly",period=7,fourier_order=5)  
theft_prophet_data.shape
```

#Splitting train test data

```
train = theft_prophet_data[:62]  
test = theft_prophet_data[62:]  
train
```

#Model fit

```
m.fit(train)
```

#Future dates

```
future = m.make_future_dataframe(periods=730)  
future
```

#Forecasted values

```
forecast = m.predict(future)
forecast[['ds','trend','yhat', 'yhat_lower', 'yhat_upper']].tail()
```

#Visualization

```
m.plot(forecast, xlabel='Date', ylabel='Crime Rate')
fig=m.plot_components(forecast)
```

#Testing data

```
test
test = test.set_index("ds")
test = test['y']
import matplotlib.pyplot as plt
test.plot()
```

#Trend change points

```
from fbprophet.plot import add_changepoints_to_plot
fig = m.plot(forecast)
c = add_changepoints_to_plot(fig.gca(),m,forecast)
```

#Model Evaluation

```
from fbprophet.diagnostics import cross_validation
forecast = cross_validation(m, initial='730 days', period='180 days', horizon = '365
days')
forecast.head()

from fbprophet.diagnostics import performance_metrics
df_p = performance_metrics(forecast)
df_p.head()

from fbprophet.plot import plot_cross_validation_metric
fig = plot_cross_validation_metric(forecast, metric='mape')
```

APPENDIX C

SYSTEM REQUIREMENTS

In this appendix system requirements which will be most important for our work to process are mentioned. We have mentioned the hardware specifications, software specification and the browsers and the source of the crime data which we used are mentioned below.

HARDWARE SPECIFICATION

- Processor (CPU) with 8GB RAM.
- Internet Connection
- Keyboard and Mouse or some other compatible pointing device

SOFTWARE SPECIFICATION

- Windows 10 or Higher
- Google Collaboratory

BROWSERS

- Chrome
- Edge
- Mozilla Firefox
- Internet Explorer
- Safari

DATASETS

- Kaggle
- Data.gov
- Data world
- Knoema

APPENDIX D

PUBLICATIONS

- [1] Srinidhi V, Saranya P, M Ashok, presented a paper titled “An Affirmative Learning Techniques to analyse the crime Scene in Jewel Theft Murder” in the “International Conference on Research Advancements & Challenges in Engineering Sciences (ICRACE’20)” organized by Velammal Institute of Technology, Chennai on 6th and 7th of March 2020.

REFERENCES

- [1] Hsinchun Chen Wingyan Chung Jennifer Jie Xu Gang Wang Yi Qin University of Arizona Michael Chau University of Hong Kong," Crime Data Mining: A General Framework and Some Examples"(2001).
- [2] Tong Wang, Cynthia Rudin, Daniel Wagner, and Rich Sevieri," Learning to Detect Patterns of Crime"(2005).
- [3] Shiju Sathyadevan, Devan M.S.,Amrita Center for Cyber Security, Amrita Vishwa Vidyapeetham, Amritapuri,Surya Gangadharan. S, "Crime Analysis and Prediction Using Data Mining" (2007).
- [4] Malathi, Dr. S. Santhosh Baboo," An Enhanced Algorithm to Predict a Future Crime using Data Mining" International Journal of Computer Applications May 2011.
- [5] Qusay Bsoul, Juhana Salim, Lailatul Qadri Zakaria," An Intelligent Document Clustering Approach to Detect Crime Patterns" The 4th International Conference on Electrical Engineering and Informatics (ICEEI 2013) Procedia Technology 11 (2013).
- [6] Omowunmi Isafiadea, Antoine Bagulab, Sonia Bermana," A Revised Frequent Pattern Model for Crime Situation Recognition Based on Floor-Ceil Quartile Function" Information Technology and Quantitative Management March 2015.
- [7] P.A.C Duijn,"From data to Disruption" [Digital Investigation](#)[Volume 15](#), December 2015.
- [8] Vineet Pande , D. J. Sanghvi, Viraj Samant , Sindhu Nair," Crime Detection using Data Mining" International Journal of Engineering Research & Technology [IJERT] Issue 01, January-2016.

- [9] H. Benjamin Fredrick David and A. Suruliandi, "SURVEY ON CRIME ANALYSIS AND PREDICTION USING DATA MINING TECHNIQUES" ISSN: 2229-6959 (ONLINE) ICTACT JOURNAL ON SOFT COMPUTING, APRIL 2017.
- [10] Masita, Masila Abdul Jalil, Chia Pui Ling, Noor Maizura, Mohamad Noor, Fatihah Mohd., "Knowledge Representation Model for Crime Procedia Computer Science May 2017.
- [11] Prajakta Yerpude and Vaishnavi Gudur, "PREDICTIVE MODELLING OF CRIME DATASET USING DATA MINING" International Journal of Data Mining & Knowledge Management Process (IJDMP) July 2017.
- [12] Suhong Kim, Param Joshi, Parminder Singh Kalsi, and Pooya Taheri "Crime Analysis Through Machine Learning" Fraser International College, Simon Fraser University British Columbia, Canada JANUARY (2018).
- [13] Deepika Tyagi, Dr. Sanjiv Sharma, "AN APPROACH TO CRIME DATA ANALYSIS: A SYSTEMATIC REVIEW" International Journal of Engineering Technologies and Management Research Tyagi (Iss.2: SE): February, 2018].
- [14] Hitesh Kumar Reddy Toppi Reddy, Bhavna Sainia, Ginika Mahajana, "Crime Prediction & Monitoring Framework Based on Spatial Analysis" International Conference on Computational Intelligence and Data Science June 2018.
- [15] Alkesh Bharati, Dr Sarvanaguru R.A.K., "Crime Prediction and Analysis Using Machine Learning" International Research Journal of Engineering and Technology (IRJET) Sep 2018.
- [16] Varvara Ingilevich^a, Sergey Ivanov^b "Crime rate prediction in the urban environment using social factors" November 2018

- [17] Panagiotis Stalidis, Theodoros Semertzidis, Member, IEEE and Petros Daras, Senior Member, IEEE Examining Deep Learning Architectures for Crime Classification and Prediction” 3 Dec 2018