1.CRIME RATE PREDICTION

1.1 Introduction:

Crimes are the significant threat to the humankind. There are many crimes that happens regular interval of time. Perhaps it is increasing and spreading at a fast and vast rate. Crimes happen from small village, town to big cities. Crimes are of different type – robbery, murder, rape, assault, battery, false imprisonment, kidnapping, homicide. Since crimes are increasing there is a need to solve the cases in a much faster way.

The crime activities have been increased at a faster rate and it is the responsibility of police department to control and reduce the crime activities. Crime prediction and criminal identification are the major problems to the police department as there are tremendous amount of crime data that exist. There is a need of technology through which the case solving could be faster.

Through many documentation and cases, it came out that machine learning and data science can make the work easier and faster. The aim of this project is to make crime prediction using the features present in the dataset. The dataset is extracted from the official sites. With the help of machine learning algorithm, using python as core we can predict the type of crime which will occur in a particular area with crime_percapita.

The objective would be to train a model for prediction. The training would be done using Training data set which will be validated using the test dataset. The Multi Linear Regression(MLR)will be used for crime prediction. Visualization of dataset is done to analyze the crimes which may have occurred in a particular year and based on population and number of crimes. This work helps the law enforcement agencies to predict and detect the crime_percapita in an area and thus reduces the crime rate.

1.2 Objective Of The Research:

The main objective of the project is to predict the percapita of crime rate and analyze the crime rate to be happened in future. Based on this Information the officials can take charge and try to reduce the crime rate.

The concept of Multi Linear Regression is used for predicting the graph between the Types of Crimes (Independent Variable) and the Year (Dependent Variable)

The system will look at how to convert crime informationinto a regression problem, so that it will help detectives insolving crimes faster. Crime analysis based on available information to extract crime patterns.Using various multi linear regression techniques, frequency of occurring crime can be predicted based on territorial distribution of existing data and Crime recognition.

1.3 Problem Statement:

The main problem is that day to day the population is going to be increased and by that the crimes are also going to be Increased in different areas by this the crime rate can’t be accurately predictedby the officials. The officials as they focus on many issues may not predict the crimes to be happened in the future. The officials/police officers although they tries to reduce the crime rate they may not reduce in full-fledged manner. The crime rate prediction in future may be difficult for them.

1.4 Industry Profile:

The main significant approach, used in this paper for the predicting result is a concept of machine learning and result tested in the Crime Department index data set. To seize the best accurate output, the approach decided to be implemented is machine learning along with supervised classifier. It helps the various departments of police to make a decision and strategy to prevent the crime.

2.REVIEW OF LITERATURE

Machine Learningis the study and analysis of criminology, can be categorized into main areas, crime control andcrimesuppression. Crime control tends to use knowledge from the analyzed data to control and prevent the occurrence of crime.Crime detection and prevention techniques are applied todifferent applications ranging from cross-border security,Internet security to household crimes.Abraham et al. (2006) proposed a method to employ computer log files as history datato search some relationships by using the frequency occurrence of incidents. From the literature study, it could be concluded that crime data is increasing to very large quantities running into zotabytes (1024bytes). This in turn is increasing the need for advanced and efficient techniques for analysis. Machine learning as an analysis and knowledge discovery tool has immense potential for crime data analysis. As is the case with any other new technology, therequirement of such tool changes, which is further augmented bythe new and advanced technologies used by criminals. All thesefacts confirm that the field is not yet mature and needs furtherinvestigations.

Is crime in India rising or falling? The answer is not as simple as politicians make it out to be because of how the FBI collects crime data from the country’s more than 18,000 police agencies. National estimates can be inconsistent and out of date, as the FBI takes months or years to piece together reports from those agencies that choose to participate in the voluntary program.

To try to fill this gap, The Marshall Project collected and analyzed more than 40 years of data on the four major crimes the FBI (Federal Bureau of Investigation) classifies as violent — homicide, rape, robbery and assault — in 68 police jurisdictions with populations of 250,000 or greater. We obtained 2015 reports, which have yet to be released by the FBI, directly from 61 of them. We calculated the rate of crime in each category and for all violent crime, per 100,000 residents in the jurisdiction, based on the FBI’s estimated population for that year. We used the 2014 estimated population to calculate 2015 crime rates per capita.

The crime data was acquired from the FBI Uniform Crime Reporting program's "Offenses Known and Clearances by Arrest" database for the year in question, held at the National Archives of Criminal Justice Data. The data was compiled and analyzed by Gabriel Dance, Tom Meagher, and Emily Hopkins of The Marshall Project; the analysis was published as par

3.DATA COLLECTION

Dataset:

A **data set** (or **dataset**) is collection of data. Most commonly a data set corresponds to the contents of a single database table, or a single statistical data matrix, where every column of the table represents a particular variable, and each row corresponds to a given member of the data set in question. The data set lists values for each of the variables, such as height and weight of an object, for each member of the data set. Each value is known as a datum. The data set may comprise data for one or more members, corresponding to the number of rows.

Data collection is critical for the accurate prediction of crime occurrences. In this section, we present collection methods for data from India. We employed data from seven domains: crime occurrence reports, demographic, housing, economic, education, weather, and image data. Data were collected from India because it has both a large population (approximately 2.7 million) and a high crime level (a total of 274,064 cases in 2014). The report containing crime occurrence data was collected from Indian Data Portal. We used the report from 2014, which contains the date, crime type, and Years when crimes occurred as coordinates of incidents involving crime. The report lists a total of 2830 Rows of Years, Population, Crimes and per capita of Crimes and Crime dataset from kaggle is used in CSV format.

The Attributes that are used in our dataset are:-

- Report_Year:-It specifies the years in which Crimes happened and it is a dependent attribute.
- Population:-It specifies the population of specific years in which Crimes happened and it is a dependent attribute.
- Robberies:-It specifies the total number of Robberies that has occurred in different Years and it is a dependent attribute.
- Robberies_percaptia:- It specifies the percapita of the Robberies that occurred in future.

Prediction in Multiple Linear Regression algorithm:

Just as in simple linear regression, we may be interested to produce prediction intervals for specific or for general new observations. For a specific set of values of the predictor.

$$\boxed{x'0 = [1, x01, x02,...,x0k]}$$

a point estimate for a future observation y at x0 is

$$\boxed{y\hat{}0 = x'\ 0\beta\hat{}}$$

a 100(1 − α)% prediction interval for this future observation is

$$\boxed{y\hat{}0 \pm t\ \alpha/2,n-k-1\ /\sigma\hat{}2\ (1 + x'0(X'\ X)-1x0)}$$

4. METHODOLOGY

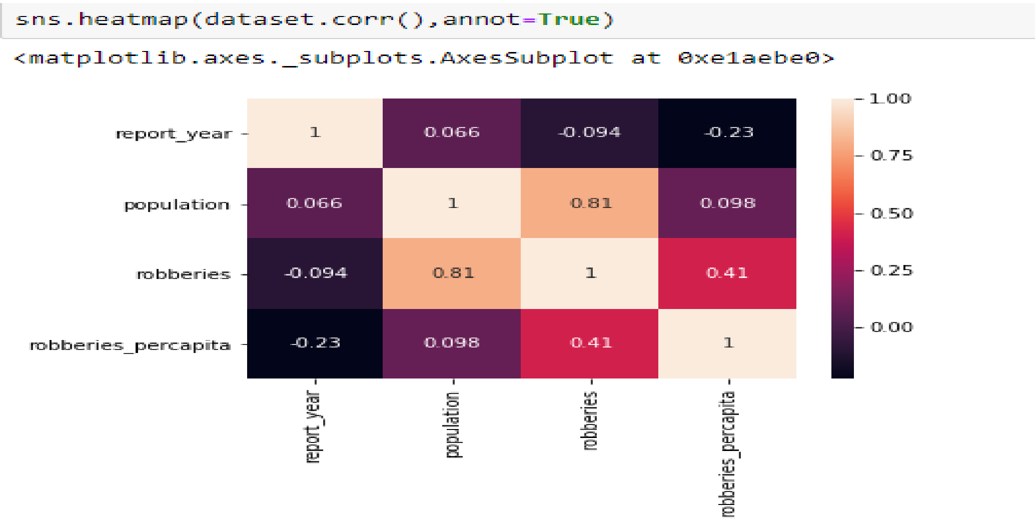
4.1 Exploratory Data Analysis:

Exploratory data analysis (EDA) is an approach to analyzing data sets to summarize their main characteristics, often with visual methods. A statistical model can be used or not, but primarily EDA is for seeing what the data can tell us beyond the formal modeling or hypothesis testing task. Exploratory data analysis was promoted by John Tukey to encourage statisticians to explore the data, and possibly formulate hypotheses that could lead to new data collection and experiments. EDA is different from initial data analysis (IDA),[1] which focuses more narrowly on checking assumptions required for model fitting and hypothesis testing, and handling missing values and making transformations of variables as needed. EDA encompasses IDA.

The objectives of EDA are to:

- Suggest hypotheses about the causes of observed phenomena
- Assess assumptions on which statistical inference will be based
- Support the selection of appropriate statistical tools and techniques
- Provide a basis for further data collection through surveys or experiments

4.1.1 Figures And Tables:

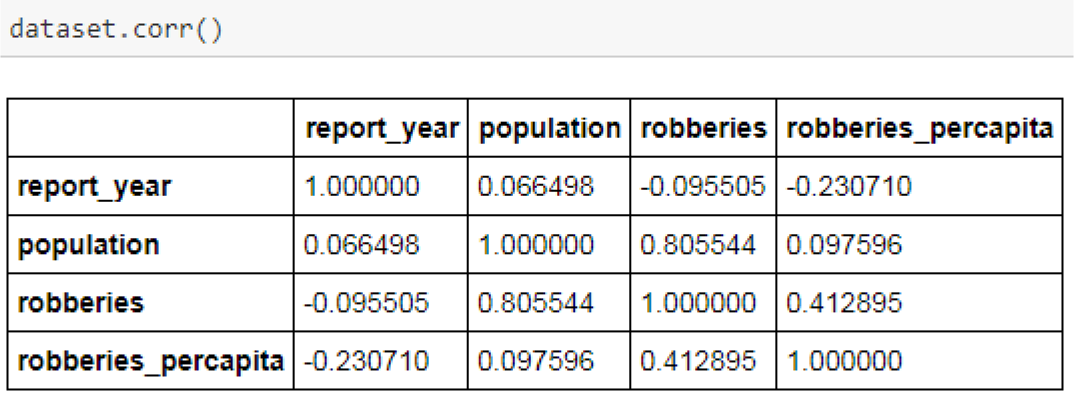


```
sns.heatmap(dataset.corr(),annot=True)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0xe1aebe0>
```

	report_year	population	robberies	robberies_percapita
report_year	1	0.066	-0.094	-0.23
population	0.066	1	0.81	0.098
robberies	-0.094	0.81	1	0.41
robberies_percapita	-0.23	0.098	0.41	1

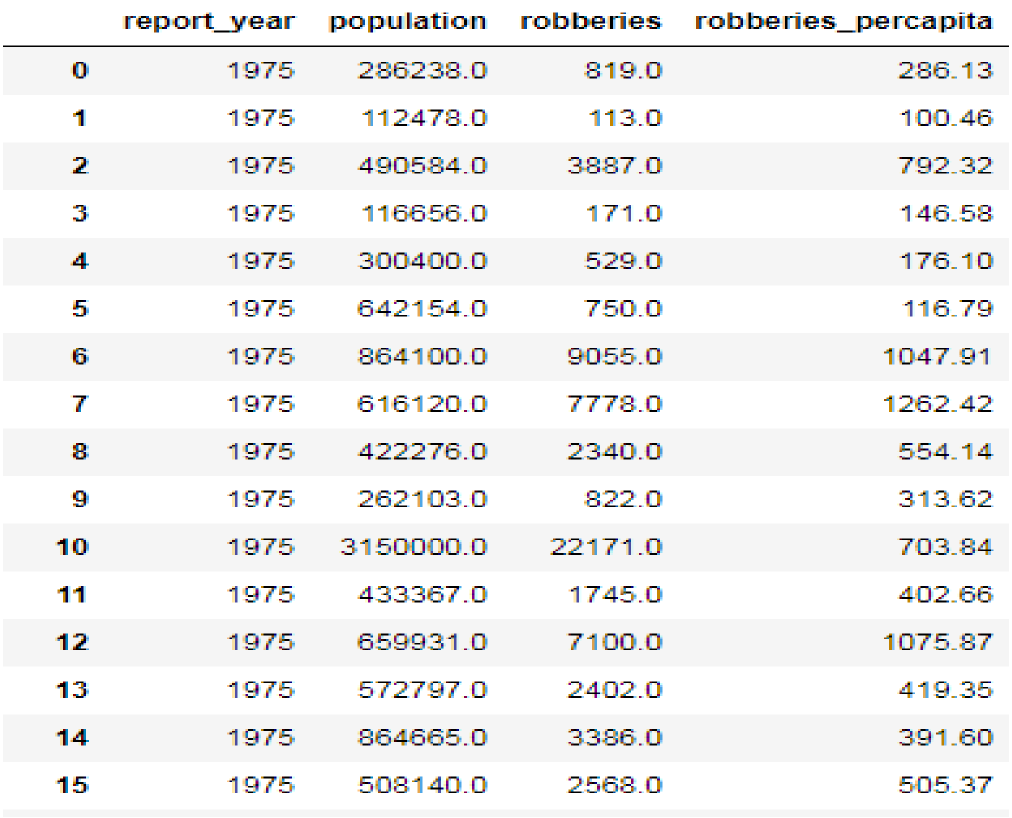
Figure:1 Seaborn



```
dataset.corr()
```

	report_year	population	robberies	robberies_percapita
report_year	1.000000	0.066498	-0.095505	-0.230710
population	0.066498	1.000000	0.805544	0.097596
robberies	-0.095505	0.805544	1.000000	0.412895
robberies_percapita	-0.230710	0.097596	0.412895	1.000000

Table :2 Co-Relation Analysis



	report_year	population	robberies	robberies_percapita
0	1975	286238.0	819.0	286.13
1	1975	112478.0	113.0	100.46
2	1975	490584.0	3887.0	792.32
3	1975	116656.0	171.0	146.58
4	1975	300400.0	529.0	176.10
5	1975	642154.0	750.0	116.79
6	1975	864100.0	9055.0	1047.91
7	1975	616120.0	7778.0	1262.42
8	1975	422276.0	2340.0	554.14
9	1975	262103.0	822.0	313.62
10	1975	3150000.0	22171.0	703.84
11	1975	433367.0	1745.0	402.66
12	1975	659931.0	7100.0	1075.87
13	1975	572797.0	2402.0	419.35
14	1975	864665.0	3386.0	391.60
15	1975	508140.0	2568.0	505.37

Table :3 Sample Dataset

4.2 Statistical Techniques And Visualization:

Statistics is a collection of tools that you can use to get answers to important questions about data. You can use descriptive statistical methods to transform raw observations into information that you can understand and share. You can use inferential statistical methods to reason from small samples of data to whole domains. Statistics is a pillar of machine learning. You cannot develop a deep understanding and application of machine learning without it.

- Problem Framing: Requires the use of exploratory data analysis and data mining.
- Data Understanding: Requires the use of summary statistics and data visualization.
- Data Cleaning. Requires the use of outlier detection, imputation and more.
- Data Selection. Requires the use of data sampling and feature selection methods.
- Data Preparation. Requires the use of data transforms, scaling, encoding and much more. Model Evaluation. Requires experimental design and resampling methods.
- Model Configuration. Requires the use of statistical hypothesis tests and estimation statistics. Model Selection. Requires the use of statistical hypothesis tests and estimation statistics. Model Presentation. Requires the use of estimation statistics such as confidence intervals. Model Predictions. Requires the use of estimation statistics such as prediction intervals.

NumPy is a commonly used Python data analysis package. By using NumPy, you can speed up your workflow, and interface with other packages in the Python ecosystem, like scikit-learn, that use NumPy under the hood. NumPy was originally developed in the mid 2000s, and arose from an even older package called Numeric. This longevity means that almost every data analysis or machine learning package for Python leverages NumPy in some way.

we’ll walk through using NumPy to analyze data on wine quality. The data contains information on various attributes of wines, such as pH and fixed acidity, along with a quality score between 0 and 10 for each wine. The quality score is the average of at least 3 human taste testers. As we learn how to work with NumPy, we’ll try to figure out more about the perceived quality of wine.

Numpy 2-Dimensional Arrays With NumPy, we work with multidimensional arrays. We’ll dive into all of the possible types of multidimensional arrays later on, but for now, we’ll focus on 2-dimensional arrays. A 2-dimensional array is also known as a matrix, and is something you should be familiar with.

Creating A NumPy Array : We can create a NumPy array using the numpy.array function. If we pass in a list of lists, it will automatically create a NumPy array with the same number of rows and columns. Because we want all of the elements in the array to be float elements for easy computation, we’ll leave off the header row, which contains strings. One of the limitations of NumPy is that all the elements in an array have to be of the same type, so if we include the header row, all the elements in the array will be read in as strings. Because we want to be able to do computations like find the average quality of the wines, we need the elements to all be floats. In the below code:

- Import the numpy package.
- Pass the list of lists wines into the array function, which converts it into a NumPy array.
- Exclude the header row with list slicing.
- Specify the keyword argument dtype to make sure each element is converted to a float.

Pandas - is an open source python library that is built on top of NumPy. It allows you do fast analysis as well as data cleaning and preparation. Pandas is hands down one of the best libraries of python. It supports reading and writing excelspreadsheets, CVS's and a whole lot of manipulation. It is more like a mandatory library you need to know if you’re dealing with datasets from excel files and CSV files. i.e for Machine learning and data science. This is part one of Pandas tutorial. I’m not going to cover everything possible with pandas, however, I want to give you a taste of what it is and how you can get started with it. This tutorial is going to be super short just introducing you to Series object of pandas.

As other libraries, you’d import pandas and reference it as pd.

import pandas as pd

pd.Series()

pd.Series() is a method that creates a series object from data passed. The data must be defined as a parameter. what is a “Series” object in Pandas? It is a data structure defined by Pandas. Basically it looks like a table having rows and columns. Notice that these numbers on the first column were added automatically by pandas. They serve as index. These variables are known as categorical variables and in terms of pandas, these are called ‘object’.

To retrieve information using the categorical variables, we need to convert them into ‘dummy’ variables so that they can be used for modelling.We do that using pandas.get_dummies feature.

First we create a list of the categorical variables. Then we convert these variables into dummy variables. We have created dummy variables for each categorical variables and printing out the head of the new data-frame. You can understand, how the categorical variables are converted to dummy variables which are ready to be used in the modelling of this data-set. But, we have a slight problem here. The actual categorical variables still exist and they need to be removed to make the data-frame ready for machine learning.

Matplotlib - is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython. There is also a procedural "pylab" interface based on a state machine (like OpenGL), designed to closely resemble that of MATLAB, though its use is discouraged.SciPy makes use of Matplotlib. Matplotlib was originally written by John D. Hunter, has an active development community, and is distributed under a BSD-style license. Michael Droettboom was nominated as matplotlib's lead developer shortly before John Hunter's death in August 2012, and further joined by Thomas Caswell. As of 23 June 2017, matplotlib 2.0.x supports Python versions 2.7 through 3.6. Matplotlib 1.2 is the first version of matplotlib to support Python 3.x. Matplotlib 1.4 is the last version of Matplotlib to support Python 2.6. Matplotlib has pledged to not support Python 2 past 2020 by signing the Python 3 Statement. Pyplot is a Matplotlib module which provides a MATLAB-like interface.

Matplotlib is designed to be as usable as MATLAB, with the ability to use Python, and the advantage of being free and open-source. Several toolkits are available which extend Matplotlib functionality. Some are separate downloads, others ship with the Matplotlib source code but have external dependencies.

- Basemap: map plotting with various map projections, coastlines, and political boundaries.
- Cartopy: a mapping library featuring object-oriented map projection definitions, and arbitrary point, line, polygon and image transformation capabilities. (Matplotlib v1.2 and above)
- Excel tools: utilities for exchanging data with Microsoft Excel • GTK tools: interface to the GTK+ library.

Visualization with Matplotlib- One of Matplotlib's most important features is its ability to play well with many operating systems and graphics backends. Matplotlib supports dozens of backends and output types, which means you can count on it to work regardless of which operating system you are using or which output format you wish. This cross-platform, everything-to-everyone approach has been one of the great strengths of Matplotlib. It has led to a large user base, which in turn has led to an active developer base and Matplotlib's powerful tools and ubiquity within the scientific Python world.

In recent years, however, the interface and style of Matplotlib have begun to show their age. Newer tools like ggplot and ggvis in the R language, along with web visualization toolkits based on D3js and HTML5 canvas, often make Matplotlib feel clunky and old-fashioned. Still, I'm of the opinion that we cannot ignore Matplotlib's strength as a well-tested, cross-platform graphics engine. Recent Matplotlib versions make it relatively easy to set new global plotting styles (see Customizing Matplotlib: Configurations and Style Sheets), and people have been developing new packages that build on its powerful internals to drive Matplotlib via cleaner, more modern APIs—for example, Seaborn (discussed in Visualization With Seaborn), ggpy, HoloViews, Altair, and even Pandas itself can be used as wrappers around Matplotlib's API. Importing Matplotlib - Just as we use the np shorthand for NumPy and the pd shorthand for Pandas, we will use some standard shorthands for Matplotlib imports.

Plotting from a script - If you are using Matplotlib from within a script, the function plt.show() is your friend. plt.show() starts an event loop, looks for all currently active figure objects, and opens one or more interactive windows that display your figure or figures.

The plt.show() command does a lot under the hood, as it must interact with your system's interactive graphical backend. The details of this operation can vary greatly from system to system and even installation to installation, but matplotlib does its best to hide all these details from you.

4.3 Data Modelling and visualization:

Node-RED is a flow-based development tool for visual programming developed originally by IBM for wiring together hardware devices, APIs and online services as part of the Internet of Things. Node-RED provides a web browser-based flow editor, which can be used to create JavaScript functions. Elements of applications can be saved or shared for re-use. The runtime is built on Node.js. The flows created in Node-RED are stored using JSON. Since version 0.14 MQTT nodes can make properly configured TLS connections.

In 2016, IBM contributed Node-RED as an open source JS Foundation project.

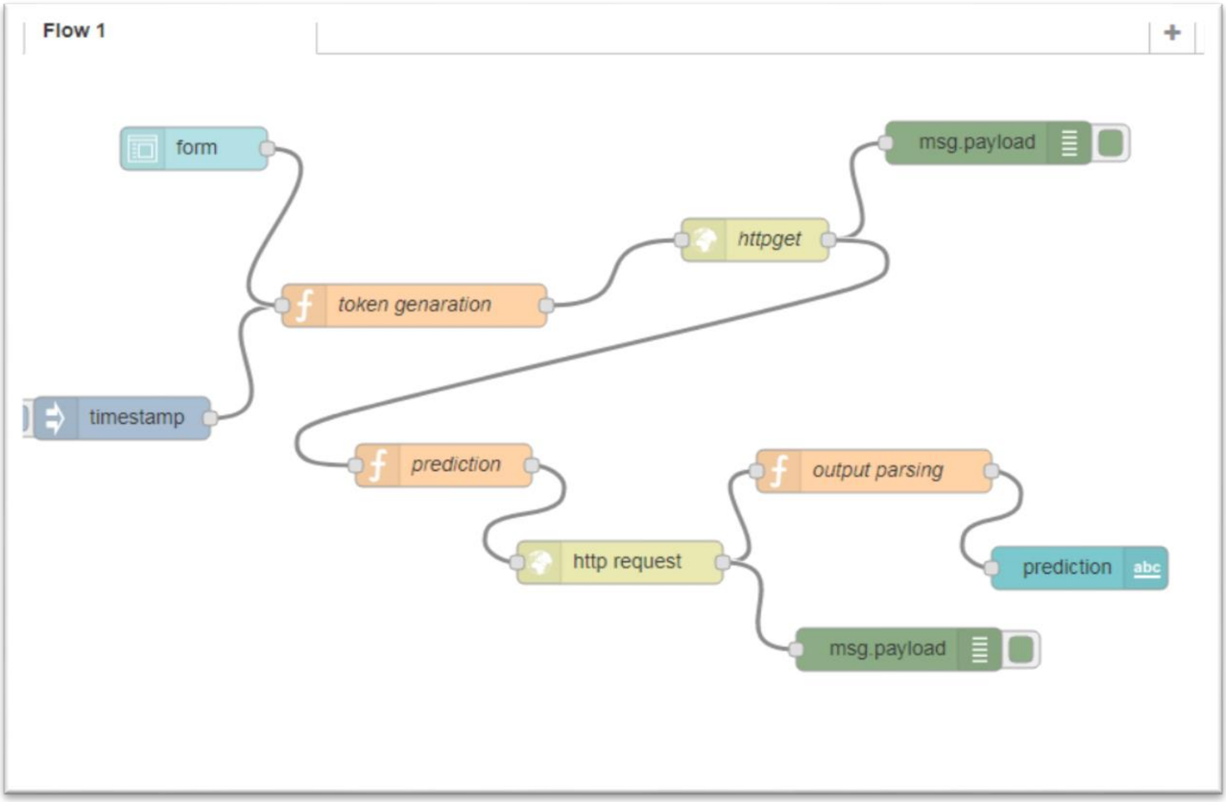
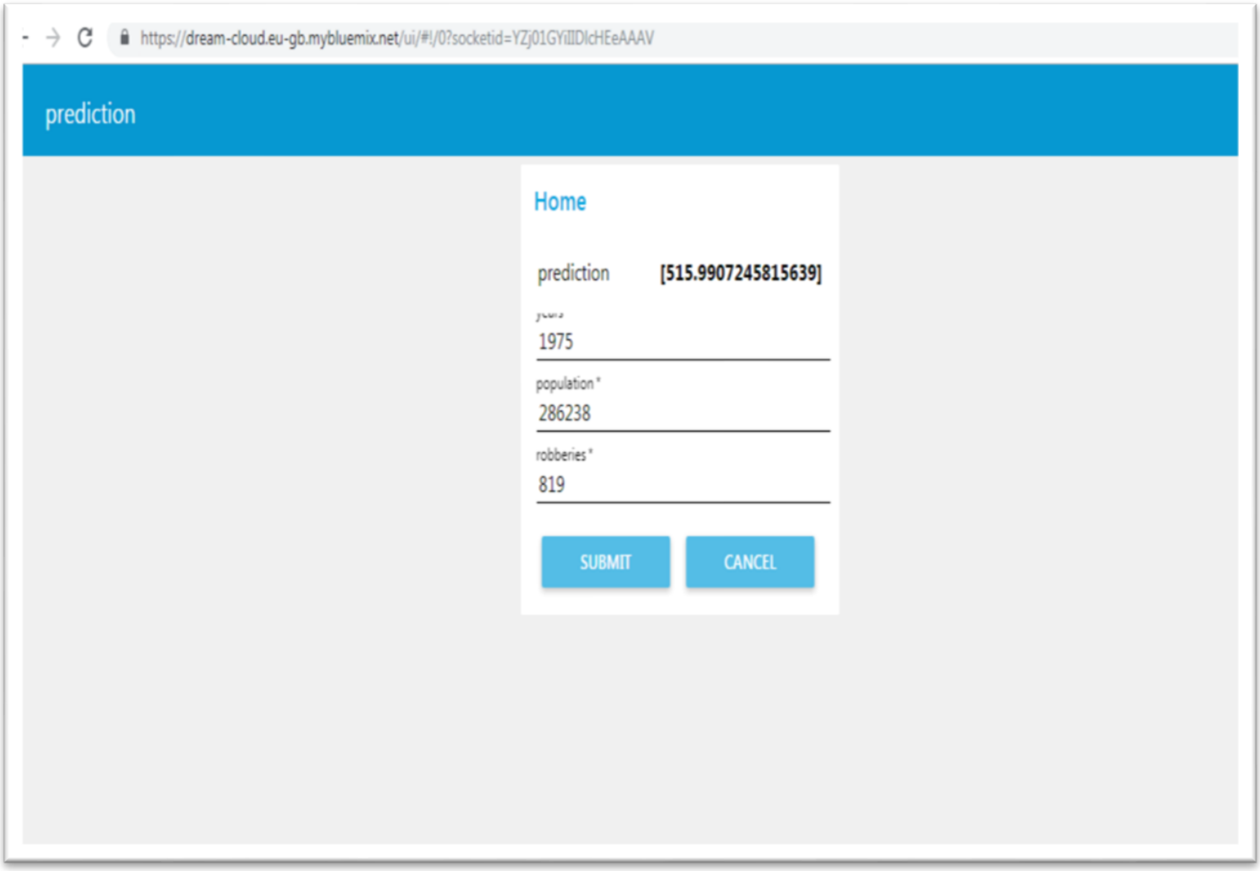


Figure: 4 Flow Diagram



https://dream-cloud.eu-gb.mybluemix.net/ui/#!/0?socketid=YZj01GYiIIDIcHEeAAAV

prediction

Home

prediction [515.9907245815639]

years
1975

population *
286238

robberies *
819

Figure :5 User Interface

5. FINDINGS AND SUGGESTIONS

Findings:

The Crime Rate Prediction helps the Officials to know about the crime rate in different areas. We predict the future crime rate based on the data extracted from previous datasets. The crime rate prediction will help the society for predicting different types of crimes in different areas.

Suggestions:

With predictive analytics, you can move from counting crime after it has occurred to preventing crime before it happens. In this realm, there are any number of possible applications not yet mentioned, including: • Cyber crime profiling

- Forensics analysis
- Open source intelligence analytics
- Internal and external terrorist threats
- Traffic risk profiling
- Suspect vehicle identification
- Material maintenance predictions
- Inclusion of citizen feedback.

6. CONCLUSION

With the help of machine learning technology, it has become easy to find out relation and patterns among various data's. The work in this project mainly revolves around predicting the type of crime and crime percapita which may happen in future. Using the concept of machine learning we have built a model using training data set that have undergone data cleaning and data transformation using Multi Linear Regression Algorithm. The model predicts the type of crime and Data visualization helps in analysis of data set and prediction of crimes. The graphs include bar, line and scatter graphs each having its own characteristics. We generated many graphs and found interesting statistics that helped in understanding different crime datasets that can help in capturing the factors that can help in keeping society safe.