analysing the crimes in india

sharang dev srivastava

MSc in Data Analytics

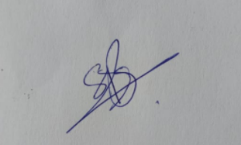
Athlone Institute of Technology



##### Declaration

I hereby certify that the material, which is submitted in this thesis towards the award of MSc. in Data Analytics, is entirely my own work and has not been submitted for any academic assessment other than part fulfillment of the above-named award.

Future students may use the material contained in this thesis provided that the source is acknowledged in full.



Signed\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date 19th August 2020

##### Abstract

In the article in the Business Insider, suggests that the crimes in India have cost economic growth. According to an international think tank, the Institute for Economics and Peace, crimes have cost the country, 9% of its GDP in 2017. This equates to roughly $600 per person in India(Dhillon, 2018). Analysis of crime and taking prevention to ensure the safety of the public is the necessary step for the government and law enforcement. This research will mainly focus on the top crimes that are being committed in India. Top crimes in India will be identified on the chart and the worst affected states for each crime will be plotted on the map of India. The trend of total IPC crimes in every state for the next 6 years will be identified in this research using a machine learning technique. The seasonal trend graph will analyze the observed and trend crime of every state in India. This forecasting technique will help law enforcement to examine the trend of crimes every state in India and so that they can take minimize the crime as much as possible. This forecasting model will be dependent on the count of past crimes that are committed. Cities of the top worst state affected will be clustered according to the crimes committed this will help the government to identify the highly vulnerable cities. Cities that are highly affected in top states will be analyzed.

##### Acknowledgements

First, I would like to thank my supervisor Barry O’Loughlin who thoroughly helped me in my writing and conducting my research from start to end. Miss Emer Conolly was always helpful in my writing part and corrected all my grammatical issues and made sure that my dissertation was unique.

I want to thank my family members and friends who had always supported me. I especially want to thank my mother who has supported me since my childhood, she made sure that I complete all my work on time.

--

Regards,

Sharang Dev Srivastava.

Table of Contents

[Chapter 1 - INTRODUCTION 1](#_Toc48746492)

[1.1 INTRODUCTION 1](#_Toc48746493)

[1.2 RESEARCH AIMS AND OBJECTIVES 3](#_Toc48746494)

[1.2.1 RESEARCH QUESTION 3](#_Toc48746495)

[1.2.2 OBJECTIVE 3](#_Toc48746496)

[Chapter 2 – LITERATURE REVIEW 4](#_Toc48746497)

[2.1 INTRODUCTION 4](#_Toc48746498)

[2.2 CRIMES 4](#_Toc48746499)

[2.3 FORECASTING 11](#_Toc48746500)

[2.4 CLUSTERING 12](#_Toc48746501)

[2.5 CRISP-DM 14](#_Toc48746502)

[Chapter 3 – RESEARCH METHODOLOGY 16](#_Toc48746503)

[3.1 INTRODUCTION 16](#_Toc48746504)

[3.2 RESEARCH STRATEGY 17](#_Toc48746505)

[3.3 RESEARCH METHOD 17](#_Toc48746506)

[3.4 METHOD OF DATA COLLECTION 19](#_Toc48746507)

[3.4.1 OVERVIEW OF THE DATASET 20](#_Toc48746508)

[3.4.2 VISUALISING tHE DATA 24](#_Toc48746509)

[3.5 DESIGNING 24](#_Toc48746510)

[3.5.1 SELECTION OF MODEL 24](#_Toc48746511)

[3.5.2 TOOLS USED FOR ANALYSIS 25](#_Toc48746512)

[3.5.3 MEASURING ACCURACY 26](#_Toc48746513)

[3.6 IMPLEMENTATION 27](#_Toc48746514)

[3.6.1 STEPS INVOLVED IN DATA EXPLORATION 27](#_Toc48746515)

[3.6.2 MODELLING APPROACH 27](#_Toc48746516)

[3.6.3 ARIMA MODEL 28](#_Toc48746517)

[3.6.4 K-MEANS CLUSTERING MODEL 30](#_Toc48746518)

[3.6.5 LIBRARIES 30](#_Toc48746519)

[3.7 LIMITATION 32](#_Toc48746520)

[Chapter 4 – FINDINGS AND ANALYSIS 33](#_Toc48746521)

[4.1 INTRODUCTION 33](#_Toc48746522)

[4.2 CRIMES IN INDIA 33](#_Toc48746523)

[4.3 TOP 10 CRIMES IN INDIA 34](#_Toc48746524)

[4.3.1.1 THEFT IN INDIA 36](#_Toc48746525)

[4.3.1.2 HURT IN INDIA 37](#_Toc48746526)

[4.3.1.3 KIDNAPPING IN INDIA 38](#_Toc48746527)

[4.3.1.4 CHEATING IN INDIA 39](#_Toc48746528)

[4.3.1.5 BURGLARY IN INDIA 40](#_Toc48746529)

[4.3.1.6 RIOTS IN INDIA 41](#_Toc48746530)

[4.3.1.7 MURDER IN INDIA 42](#_Toc48746531)

[4.3.1.8 ASSAULT ON WOMEN MODEST IN INDIA 43](#_Toc48746532)

[4.3.1.9 CRUELTY BY HUSBAND IN INDIA 44](#_Toc48746533)

[4.3.1.10 DEATH BY NEGLIGENCE 45](#_Toc48746534)

[4.3.1.11 OTHER IPC CRIMES 46](#_Toc48746535)

[4.3.1.12 TOTAL IPC CRIMES IN INIDA 47](#_Toc48746536)

[4.4 FORECASTING 48](#_Toc48746537)

[4.4.1 ANDAMAN AND NICOBAR ISLAND 48](#_Toc48746538)

[4.4.2 ANDHRA PRADESH 50](#_Toc48746539)

[4.4.3 ARUNACHAL PRADESH 51](#_Toc48746540)

[4.4.4 ASSAM 52](#_Toc48746541)

[4.4.5 GOA 53](#_Toc48746542)

[4.4.6 GUJARAT 54](#_Toc48746543)

[4.4.7 HARYANA 55](#_Toc48746544)

[4.4.8 HIMACHAL PRADESH 56](#_Toc48746545)

[4.4.9 JAMMU AND KASHMIR 57](#_Toc48746546)

[4.4.10 JHARKHAND 58](#_Toc48746547)

[4.4.11 KARNATAKA 59](#_Toc48746548)

[4.4.12 KERALA 60](#_Toc48746549)

[4.4.13 MADHYA PRADESH 61](#_Toc48746550)

[4.4.14 MAHARASTRA 62](#_Toc48746551)

[4.4.15 MANIPUR 63](#_Toc48746552)

[4.4.16 MEGHALAYA 64](#_Toc48746553)

[4.4.17 MIZORAM 65](#_Toc48746554)

[4.4.18 NAGALAND 66](#_Toc48746555)

[4.4.19 ODISHA 67](#_Toc48746556)

[4.4.20 PUNJAB 68](#_Toc48746557)

[4.4.21 RAJASTHAN 69](#_Toc48746558)

[4.4.22 SIKKIM 70](#_Toc48746559)

[4.4.23 TAMIL NADU 71](#_Toc48746560)

[4.4.24 TRIPURA 72](#_Toc48746561)

[4.4.25 UTTARAKHAND 73](#_Toc48746562)

[4.4.26 WEST BENGAL 74](#_Toc48746563)

[4.4.27 UTTAR PRADESH 75](#_Toc48746564)

[4.4.28 CHANDIGARH 76](#_Toc48746565)

[4.4.29 DADRA AND NAGAR 77](#_Toc48746566)

[4.4.30 DAMAN AND DIU 78](#_Toc48746567)

[4.4.31 DELHI 79](#_Toc48746568)

[4.4.32 LAKSHADWEEP 80](#_Toc48746569)

[4.4.33 PUDUCHERRY 81](#_Toc48746570)

[4.4.34 BIHAR 82](#_Toc48746571)

[4.4.35 CHHATTISGARH 83](#_Toc48746572)

[4.5 CLUSTERING 84](#_Toc48746573)

[4.5.1 UTTAR PRADESH 84](#_Toc48746574)

[4.5.2 MAHARASTRA 86](#_Toc48746575)

[4.5.3 TAMIL NADU 88](#_Toc48746576)

[4.5.3 ANDHRA PRADESH 90](#_Toc48746577)

[4.5.5 MADHYA PRADESH 92](#_Toc48746578)

[Chapter 5 – DISCUSSION AND CONCLUSIONS 94](#_Toc48746579)

[5.1 INTRODUCTION 94](#_Toc48746580)

[5.2 DISCUSSION 94](#_Toc48746581)

[5.3 CONCLUSION AND REFLECTION 95](#_Toc48746582)

[REFERENCES 97](#_Toc48746583)

[GLOSSARY OF TERMS 103](#_Toc48746584)

[LIST OF ABBREVIATIONS 104](#_Toc48746585)

[APPENDIX 105](#_Toc48746586)

##### List of Tables

[Table 3.4.1.1: Overview of the Dataset 23](#_Toc48746257)

[Table 3.6.3.1: Description Parameters used 29](#_Toc48746258)

[Table 3.6.5.1: Libraries used 32](#_Toc48746259)

##### List of Figures

[Figure 2.2.1:Age of Women (Patel Bhai, 2020) 6](#_Toc48746278)

[Figure 2.2.2:Nature of Crime(Patel Bhai, 2020) 7](#_Toc48746279)

[Figure 2.2.3:(Victim-offender relationship)(Patel Bhai, 2020) 7](#_Toc48746280)

[Figure 2.2.4:(Caste system)(Rao, 2010) 8](#_Toc48746281)

[Figure 2.2.5:Number of hate crimes (Statista, 2019) 9](#_Toc48746282)

[Figure 2.5.1:(CRISP-DM)(Vorhies, 2016) 15](#_Toc48746283)

[Figure 3.1.1:The research ‘onion’ (Saunders, Lewis and Thornhill, 2009) 16](#_Toc48746284)

[Figure 3.5.2.1: (Common Programming language)(Mackenzie, 2019) 25](#_Toc48746285)

[Figure 4.2.1:Distribution of crimes 33](#_Toc48746286)

[Figure 4.3.1:Top 10 crimes India 34](#_Toc48746287)

[Figure 4.3.1.1:Theft In India 36](#_Toc48746288)

[Figure 4.3.1.2:Hurt In India 37](#_Toc48746289)

[Figure 4.3.1.3:Kidnapping In India 38](#_Toc48746290)

[Figure 4.3.1.4:Cheating In India 39](#_Toc48746291)

[Figure 4.3.1.5: Burglary In India 40](#_Toc48746292)

[Figure 4.3.1.6: Riots In India 41](#_Toc48746293)

[Figure 4.3.1.7: Murder In India 42](#_Toc48746294)

[Figure 4.3.1.8: Assault On Women With Intent To Outrage Her Modesty 43](#_Toc48746295)

[Figure 4.3.1.9: Cruelty By Husband 44](#_Toc48746296)

[Figure 4.3.1.10: Death By Negligence 45](#_Toc48746297)

[Figure 4.3.1.11: Other IPC Crimes 46](#_Toc48746298)

[Figure 4.3.1.12: Total IPC Crimes 47](#_Toc48746299)

[Figure 4.4.2.1: Trend of Andaman and Nicobar 48](#_Toc48746300)

[Figure 4.4.2.2: Forcast in Andaman and Nicobar 48](#_Toc48746301)

[Figure 4.4.3.1: Trend Chart In Andhra Pradesh 50](#_Toc48746302)

[Figure 4.4.3.2: Forcast of Andaman and Nicobar 50](#_Toc48746303)

[Figure 4.4.4.1: Trend of Arunachal Pradesh 51](#_Toc48746304)

[Figure 4.4.4.2: Forecast In Arunachal Pradesh 51](#_Toc48746305)

[Figure 4.4.5.1: Trend of Assam 52](#_Toc48746306)

[Figure 4.4.5.2: Forecast In Assam 52](#_Toc48746307)

[Figure 4.4.6.1: Trend of Goa 53](#_Toc48746308)

[Figure 4.4.6.2: Forecast In Goa 53](#_Toc48746309)

[Figure 4.4.7.1: Trend In Gujarat 54](#_Toc48746310)

[Figure 4.4.7.2: Forecast in Gujarat 54](#_Toc48746311)

[Figure 4.4.8.1: Trend of Haryana 55](#_Toc48746312)

[Figure 4.4.8.2: Forecast In Haryana 55](#_Toc48746313)

[Figure 4.4.9.1: Trend of Himachal Pradesh 56](#_Toc48746314)

[Figure 4.4.9.2: Forecast In Himachal Pradesh 56](#_Toc48746315)

[Figure 4.4.10.1: Trend of Jammu and Kashmir 57](#_Toc48746316)

[Figure 4.4.10.2: Forecast In Jammu and Kashmir 57](#_Toc48746317)

[Figure 4.4.11.1: Trend of Jharkhand 58](#_Toc48746318)

[Figure 4.4.11.2: Forecast In Jharkhand 58](#_Toc48746319)

[Figure 4.4.12.1: Trend of Karnataka 59](#_Toc48746320)

[Figure 4.4.12.2: Forecast In Karnataka 59](#_Toc48746321)

[Figure 4.4.13.1: Trend of Kerala 60](#_Toc48746322)

[Figure 4.4.13.2 : Forecast In Kerela 60](#_Toc48746323)

[Figure 4.4.13.1 : Trend of Madhya Pradesh 61](#_Toc48746324)

[Figure 4.4.13.2 : Forecast In Madhya Pradesh 61](#_Toc48746325)

[Figure 4.4.14.1 : Trend of Maharastra 62](#_Toc48746326)

[Figure 4.4.14.2 : Forecast In Maharastra 62](#_Toc48746327)

[Figure 4.4.15.1 : Trend In Manipur 63](#_Toc48746328)

[Figure 4.4.15.2 : Forecast In Manipur 63](#_Toc48746329)

[Figure 4.4.16.1 : Trend of Meghalaya 64](#_Toc48746330)

[Figure 4.4.16.2 : Forecast In Meghalaya 64](#_Toc48746331)

[Figure 4.4.17.1 : Trend of Mizoram 65](#_Toc48746332)

[Figure 4.4.17.2 : Forecast In Mizoram 65](#_Toc48746333)

[Figure 4.4.18.1 : Trend of Nagaland 66](#_Toc48746334)

[Figure 4.4.18.2: Forecast In Nagaland 66](#_Toc48746335)

[Figure 4.4.19.1 : Trend of Odisha 67](#_Toc48746336)

[Figure 4.4.19.2 : Forecast In Odisha 67](#_Toc48746337)

[Figure 4.4.20.1 : Trend In Punjab 68](#_Toc48746338)

[Figure 4.4.20.2 : Forecast In Punjab 68](#_Toc48746339)

[Figure 4.4.21.1 : Trend of Rajasthan 69](#_Toc48746340)

[Figure 4.4.21.2 : Forecast in Rajasthan 69](#_Toc48746341)

[Figure 4.4.22.1 : Trend of Sikkim 70](#_Toc48746342)

[Figure 4.4.22.2 : Forecast In Sikkim 70](#_Toc48746343)

[Figure 4.4.23.1 : Trend of Tamil Nadu 71](#_Toc48746344)

[Figure 4.4.23.2 : Forecast In Tamil Nadu 71](#_Toc48746345)

[Figure 4.4.24.1 : Trend of Tripura 72](#_Toc48746346)

[Figure 4.4.24.2 : Forecast In Tripura 72](#_Toc48746347)

[Figure 4.4.25.1 : Trend In Uttarakhand 73](#_Toc48746348)

[Figure 4.4.25.2 : Forecast In Uttarakhand 73](#_Toc48746349)

[Figure 4.4.26.1 : Trend of West Bengal 74](#_Toc48746350)

[Figure 4.4.26.2 : Forecast In West Bengal 74](#_Toc48746351)

[Figure 4.4.27.1 : Trend of Uttar Pradesh 75](#_Toc48746352)

[Figure 4.4.27.2 : Forecast In Uttar Pradesh 75](#_Toc48746353)

[Figure 4.4.28.1 : Trend of Chandigarh 76](#_Toc48746354)

[Figure 4.4.28.2 : Forecast In Chandigarh 76](#_Toc48746355)

[Figure 4.4.29.1 : Trend In Dadra 77](#_Toc48746356)

[Figure 4.4.29.2 : Forecast In Dadra and Nagar 77](#_Toc48746357)

[Figure 4.4.30.1 : Trend of Daman 78](#_Toc48746358)

[Figure 4.4.30.2 : Forecast In Daman 78](#_Toc48746359)

[Figure 4.4.31.1 : Trend In Delhi 79](#_Toc48746360)

[Figure 4.4.31.2 : Forecast In Delhi 79](#_Toc48746361)

[Figure 4.4.32.1 : Trend In Lakshadweep 80](#_Toc48746362)

[Figure 4.4.32.2 : Forecast In Lakshadweep 80](#_Toc48746363)

[Figure 4.4.33.1 : Trend of Pondicherry 81](#_Toc48746364)

[Figure 4.4.33.2 : Forecast In Puducherry 81](#_Toc48746365)

[Figure 4.4.34.1 : Trend of Bihar 82](#_Toc48746366)

[Figure 4.4.34.2 : Forecast In Bihar 82](#_Toc48746367)

[Figure 4.4.35.1 : Trend In Chhattisgarh 83](#_Toc48746368)

[Figure 4.4.35.2 : Forecast In Chhattisgarh 83](#_Toc48746369)

[Figure 4.5.1.1 : Crimes happening in Uttar Pradesh 84](#_Toc48746370)

[Figure 4.5.1.2 : Cities Affected in UP 84](#_Toc48746371)

[Figure 4.5.1.3 : Clustered cities in Uttar Pradesh 85](#_Toc48746372)

[Figure 4.5.2.1 : Crimes happening in the Maharastra 86](#_Toc48746373)

[Figure 4.5.2.2 : Cities affected In Maharastra 86](#_Toc48746374)

[Figure 4.5.2.3 : Clustered cities in Maharastra 87](#_Toc48746375)

[Figure 4.5.3.2 : Cities Affected in Tamil Nadu 88](#_Toc48746376)

[Figure 4.5.3.3 : Clustered cities in Tamil Nadu 89](#_Toc48746377)

[Figure 4.5.4.1 : Crimes In Andra Pradesh 90](#_Toc48746378)

[Figure 4.5.4.2 : Cities Affected In Andhra Pradesh 90](#_Toc48746379)

[Figure 4.5.4.3 : Clustered Cities In Andhra Pradesh 91](#_Toc48746380)

[Figure 4.5.5.1 : Crimes In Madhya Pradesh 92](#_Toc48746381)

[Figure 4.5.5.2 : Cities Affected In Madhya Pradesh 92](#_Toc48746382)

[Figure 4.5.5.3 : Clustered Cities In Madhya Pradesh 93](#_Toc48746383)

# Chapter 1 - INTRODUCTION

## 1.1 INTRODUCTION

In 1947, India got its independence from the British Colony, with the largest democracy and second most populated country in the world. Crime is still the major concern in the country until now; day by day it keeps on increasing. In 2018, according to the NCRB report (livemint, 2020), they found that crime rates in India have increased by 1.3%. There had been a major surge in relation to abduction and kidnapping across the country; this increased by 10.3% over the previous years. Therefore, it is important to analyze the major crimes (abduction, kidnapping, rape, theft, etc) happening in every major state and their cities. This can help Law and Enforcement to analyze the trend behind the crime and in the future, they can prevent or stop it from happening. The crimes in urban areas have been rising, national crime records bureau released the data where cites in the northern state's records two times the higher crimes rates than the southern urban agglomerations of India(Jha, 2019). Crimes against women are the major concern for the government, violent crimes against women including rape are steadily rising every passing year India's GINI coefficient that has increased from 0.32 to 0.38 in the last two decades(Himabindu, Arora and Prashanth, 2015). In 2012, the crimes against women reported by official statistics increased by 24.7%, compared to those reported in 2008. Out of 28 states, 10 states reported more than 10000 cases in 2011(Himabindu, Arora, and Prashanth, 2015). According to the report, 3,78,277 cases of crime against women were reported in the country, up from 3,59,849 in 2017. Uttar Pradesh topped the list with 59,445 cases, followed by Maharashtra (35,497) and West Bengal (30,394)(Singh, 2020). White-collar crime has acquired new dimensions. Political institu­tions have changed very rapidly and cultural norms have not kept pace with them. Hence, there is a ‘cultural lag’ in today’s India, Power has also become a source of crime for the privileged sections of society. There is a tendency among powerful persons to abuse their influence and authority. Several cases of rape and murder have been reported by wards of influential persons and political heavyweights in the recent past. White-collar crime is a phenomenon found among educated people engaged in trade, professions, and government services(Mondal, n.d.).

One of the oldest civilizations in the world, India is one of the most popular destinations on the planet earth, In June 2019 with over 720 thousand tourists visiting the country(Madhumitha Jaganmohan, 5AD). According to the statistics the share of foreigners spending in the year 2017 was around 87% and it is expected to increase by 1% till 2028(Madhumita Jaganmohan, 2020). Saftey of the tourist is a major concern for the government, this analysis will help them to understand that which state or city safer for the tourist, and which are unsafe, areas where the government needs to improve the situation. Top cities will be clustered according to most crimes committed in India this will help to identify the cities within the cluster. The defined cluster will divide the top cities with respect to the count of crimes committed, these subgroups will help the police and the government to take the necessary measure to ensure the safety of the public. The author will also forecast the total IPC crimes committed in India for the next 6 years. Forecasting or prediction of the crime is necessary for every state because it will help the local law enforcement to manage the crimes according to it.

The Introduction is followed by the 4 chapters. The next chapter after the introduction is the literature review. In the literature review, the author will discuss all the past work that was carried out by the other author in the field of crimes and the technique that will be used to obtain the objective of this thesis. In this section author will also discuss the use Crisp-Dm model and why it is necessary to follow while working on the project. Related works on the technique like clustering and forecasting will also be discussed in this chapter. In chapter 3 all the research methodology will be discussed to obtain the objective of the research. The main purpose of the research methodology is to discuss all the resources that was been used in this research. The use of the Crisp-DM method will be thoroughly explained with all the objectives and their functions. Dataset will be explained with all the labels and the columns, the author will also discuss all the tools that were used to achieve the objectives. The machine learning model which will be used to build the forecasting model and clustering techniques will be discussed thoroughly with all the parameters that were used to obtain the results. Another dataset will be used to plot the map of India, and that was also used to merge with the original dataset. The next chapter after research methodology is finding and analysis, in this chapter the author will discuss the results of the objective and all the findings and main points that need to be highlighted. All the trend graphs of the states will be discussed with the machine learning model. The graph made to identify the top crimes in India. The worst affected states with respect to the count of crimes will be plotted on the map of India. The author also computed the worst affected cities by considering a threshold value for the top crimes and those cities will be plotted on the graph for better understanding. The last chapter of this thesis will discuss the conclusions and reflections, in this chapter author will write all relative objectives and their points and limitation of the project with future scope to make this analysis better.

## 1.2 RESEARCH AIMS AND OBJECTIVES

### 1.2.1 RESEARCH QUESTION

Can historical crime data for India be used to forecast the trend of crimes in each state of India for the next six years?

### 1.2.2 OBJECTIVE

* To provide aggregate statistics from the dataset such as the highest crime areas.
* To represent key data and findings using a suitable visualization method and tool.
* To cluster applicable data.
* To consider follow on research based on the findings of this work.
* To forecast the trend of crimes for all the states for the next 6 years

# Chapter 2 – LITERATURE REVIEW

## 2.1 INTRODUCTION

The literature review will look through the accessible data identified with the concerned point and break down, sum up that data. Information or data regarding the concern topics has been obtained from the many sources of medium such as the internet, books, existing research papers, or websites. The main purpose of the literature review is to analyze all the previously existing work on crimes in India and gather information on it, this helped to find a better understanding of the topic. Previous research work can facilitate the use of the different techniques and methods this for that core knowledge of the area must be known, this can only be done by a literature review of all the previous work.

The collection of tools, technologies, and techniques that creates insight from data is known as analytics. In the current scenario where everyone wants to evolve with time, machine learning is the best fit to start with.

A subset of Artificial Intelligence, Machine learning(ML) has recently has made a lot of progress in developing the new solution or make the existing solution more reliable to use(Flovik, 2019). Some of the studies that have been carried out on the crimes in India are mainly on, poverty is the mother of crime, organized crimes in India, and online cybercrimes. This chapter discusses current literature related to all the major crimes committed in India, forecasting, clustering, and machine learning algorithms relevant to this research project. The Analysis of crimes allows countries to improve the law&order to maintain the safety of the people, this research will review the criminal data of India.

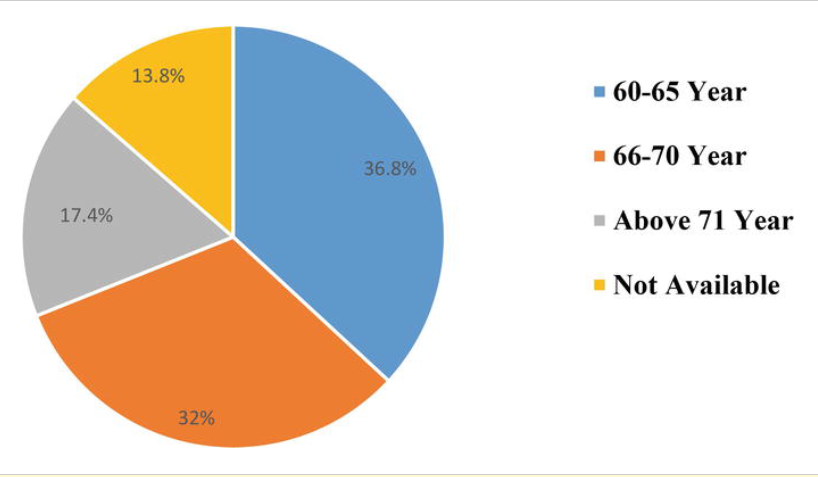
## 2.2 CRIMES

The total number of crimes and the crime rate has increased, Major reason for the increase in crime due to the population of Indian society has been increased and there is a rapid expansion in industrialization and urbanization as well over the 40 years. Cases of Burglary has been decreased by 79.84% over the 53 years, but the murder rate has been increased by 7.39%(Dubey, Agarwal and University Christ, 2015). Robbery and riots have been declined by 28.85% and 10.58% respectively, but kidnapping has been increased by 47.8%. The location has played a significant role in crimes in India, In 2006, the highest crime rate was reported in Puducherry (447.7%) for crimes under the Indian Penal Code which is 2.7 times the national crime rate of 167.7%. Kerela reported the highest crime rate at 312.5% among states. West Bengal(Kolkata) 71.0% and TamilNadu(Madurai) 206.2% were the only two major cities that reported less crime rate than their domain states West Bengal 79.0% and Tamil Nadu 227.6%. Delhi, Mumbai, and Bengaluru have accounted for 16.2%, 9.5%, and 8.1% crime rate 769.1% among the megacities in India followed by Madhya Pradesh(Bhopal) 719.5% and Rajasthan(Jaipur) 597.1%(Dubey, Agarwal and University Christ, 2015). Jammu and Kashmir 33.7%, Manipur 33.0%, Assam 30.4%, and  Daman and Diu and Puducherry 29.4%,  reported a higher violent crime rate compared to 18.4% at the national level. Uttar Pradesh reported the highest incidence of violent crimes accounting for 12.1% of total violent crimes in India (24,851 out of 2, 05,656) followed by Bihar with 11.8% (24,271 out of 2,05,6556). Among 35 megacities, Delhi reported 31.2% (533 out of 1,706) of total rape cases. Madhya Pradesh has reported the highest number of rape cases (2,900) accounting for 15.0% of total such cases reported in the country. Uttar Pradesh reported 10% (5,480 out of 32,481) of total murder cases in the country and 18.4% (4,997 out of 27,230) total attempts to murder cases. Dacoity has been decreased over the years 10627 cases were recorded in the year 1973 and 4579 in the year 2007, crime related to the property has been decreased from 1973. The recorded cases of theft in 1973 was 379421, and in the year 2007 recorded cases were 285042, statistics clearly show that theft has been decreased over the past 35 years(Dubey, Agarwal and University Christ, 2015). Factors that are affecting the increase in crime rates are social changes, According to author social disorganization, the rapid growth of urban population, increase stress on people urban people are some of the reason behind the increasing crime rates(Dubey, Agarwal and University Christ, 2015). As Indian society progressively put more prominent accentuation on material accomplishment as a method for deciding individual worth, there was a correspondingly higher pace of property wrongdoing. As India delighted in gigantic financial development, and the general size of the Indian economy quickly extended during the 1990s, the number of accessible focuses for property offenses expanded also. Dubey, et al suggested that to sustain the trend in the increase in the crime number of violations of legislation should be increased than only major offenses will decrease. Another reason in that same article author suggested that improvements in law&order and criminal justice system.

In 2019, In an article, GS Bajpai suggested that most of the crimes in India are under-registered and under-reported(Bajpai, 2019).  India’s crime rate is 379.3 per 1,00,000 persons. Cases of dacoity, attempt to murder, robbery, rape, and riot have gone down by 36.11 percent, 16.26 percent, 20.15 percent, 0.78 percent, and 54 percent respectively in 2018 as compared to the year before(Bajpai, 2019). Most of the crime data are collected and published by the National Crime Records Bureau(NCRB), data published in this article were reported by the local police station(Bajpai, 2019). The Author suggested that police station getting information is one thing and logging the FIR report is another Most of the Crimes are not being reported if the offender is from a very high society or powerful people(Bajpai, 2019).

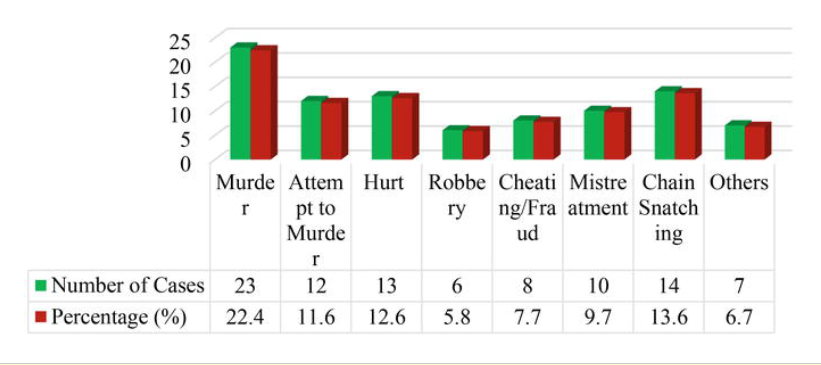
According to Prieto Curiel et al, Traditional media mostly covers the distorted version of the crime in a particular city mainly biased towards the violence or a particular caste and group(Prieto Curiel *et al.*, 2020). Prieto describes that there is a difference between what traditional media covers and what is posted on social media. Although traditional media is biased towards the violence for instance it covers major disasters more in-depth, meanwhile social media covers the crimes more accurately(Prieto Curiel *et al.*, 2020). Prieto suggests that social media could provide more information on the crime and it can be useful for detecting the fear of crime. Accordingly, social media could be an integral asset for estimating crime patterns and examples however just if it is, truth be told, identified with the crime, or social media could assist us with understanding the dread of crime and impression of security yet just on the off chance that it is, indeed, identified with genuine dread of crime. Likewise, social media could be utilized at a city level, both as far as crime and dread of crime(Prieto Curiel *et al.*, 2020).

Today in India, crimes cases against Elderly women is on a rise mainly women are facing problem such as murder, theft, hurt, bag snatching(Patel Bhai, 2020). Patel Bhai suggested two objectives in this article; the first one is to understand the nature of crimes and the second is to examine the relationship between women and their offender (Patel Bhai, 2020).



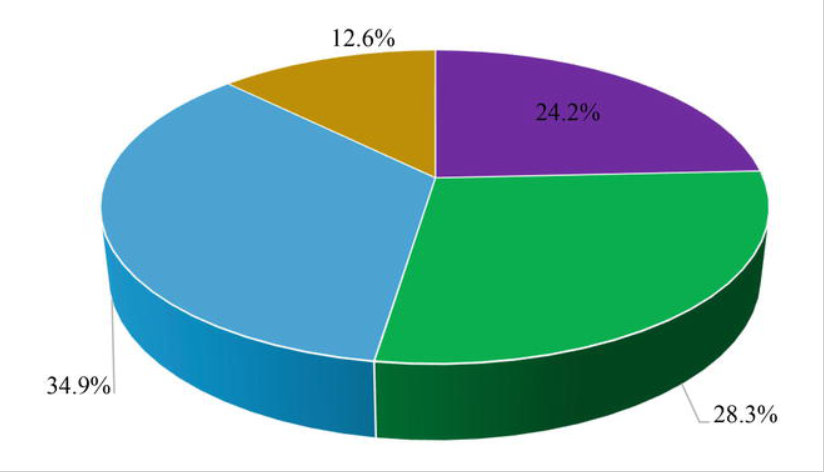
1. :Age of Women (Patel Bhai, 2020)

Women of age more 60 to 65 have been victimized the most, the chart suggests that 36.8% of women are the most affected (Patel Bhai, 2020).



1. :Nature of Crime(Patel Bhai, 2020)

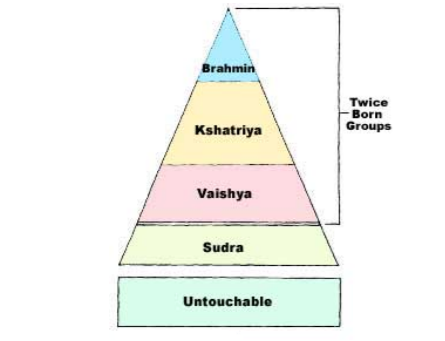
The nature of the crimes against the women is mostly murder with 22.4% followed by chain snatching and Hurt with 13.6% and 12.6% respectively(Patel Bhai, 2020).



1. :(Victim-offender relationship)(Patel Bhai, 2020)

24.2% of crimes against women are committed by a known person such as family or relatives, 34.9% of crimes offenders are neighbors to the women(Patel Bhai, 2020). Hence it can be said that elderly women are most insecure in their families and neighborhood.

The caste system is known as the system which divides Hindu culture into different subgroup based on their work and dharma(duty), The caste system divides Hindus into four main categories - Brahmins, Kshatriyas, Vaishyas, and the Shudras(BBC, 2019).



1. :(Caste system)(Rao, 2010)

Caste crime in India is one of the major concerns and the reason for the increasing number of crimes. Similar to the hate crime many caste groups are subjected to discrimination and social injustice(Sharma, 2015). These groups have been victims of crimes mostly to the upper castes. Acknowledging the severity of the problem in 2006, Indian prime minister Dr. Manmohan Singh compared the practice to that of apartheid in South Africa. Even after 60 years of constitutional support still, the majority of Dalits are facing discrimination and injustice(Rahman Maseesh, 2006).

Hate crimes in India is on a rise in recent years(Statista, 2019), Hate crimes are one of the heinous crimes in the world. Discrimination against the caste group and religion is the main reason for the hate crimes. People in society hate crimes is one of the exclusionary tactics to protect their turf and to maintain existing hierarchy in society(Sharma, 2015). The graph tells that hate crime has been increased over the year 92 number of cases we recorded by the Indian media in 2018 .

A screenshot of a cell phone

Description automatically generated

1. :Number of hate crimes (Statista, 2019)

The trend of hate crimes against the minority has been increased over the year mainly leaping from 2017 to 2018(Statista, 2019).

Each year thousands of girls and women are murdered by male relatives(Tripathi and Yadav, 2004) for the sake of family reputation across India (although mostly in the Northern part of India) in the name of family honor. ‘Honour killing’ is an ancient practice in which men kill female relatives in the name of family ‘honour’ for forced or suspected sexual activity outside marriage, even when they have been victims of rape. The reason for execution is mainly rape, infidelity, flirting or any other instance perceived as disgracing the family’s honor and not giving women any chance to defend (Tripathi and Yadav, 2004).

Meghalaya is one of the most beautiful states in India, but recently there is a surge in crimes against women. Meghalaya demonstrates a consistent ascent in gender-based viciousness over the previous decade with incidents including rape, kidnapping, molestation of women showing an increasing trend(Ropmay, 2014).

In 2014, data mining procedures can be used to solve crime faster rather than focusing on the cause of crime occurrence Instead, the main focus is to look for the crimes committed each day(Sathyadevan, Devan and Surya Gangadharan, 2014).

Crimes against women are a major worry for law enforcement right now with an increasing number of crimes. Devakuchari built a linear regression model and Devakunchari compared the Predicted values vs the Actual values for the year 2012. Results had very good accuracy, and another module was build which can help the government or law and enforcement to identify the type of crimes that happening against the women in each state(Devakunchari *et al.*, 2019).

Social distancing and an end to mass mobilization across the country have seen a sharp drop down in the crime rates. Crime rates in Delhi and Karnataka have been plunged by half since the lockdown was announced(Khanna and Das, 2020). In the given article experts fear that cases of cyberbullying and domestic violence might increase and many people will not report them. Following a nationwide lockdown, crime saw a significant decline over the past months mainly in March and February. According to the Delhi police, the total number of crimes dropped to 1,890 in 2020 from 3,416 in the previous year. Every major crime such as molestation, kidnapping, and robberies have gone down over the past few months, one of the main reasons is that sealing the Delhi state border has helped to lower the crime rates(Khanna and Das, 2020).

## 2.3 FORECASTING

Forecasting is a technique that can help one to predict the future outcome with the given historical data. Currently, the machine learning models that are used for forecasting are ARIMA, SARIMA, SARIMAX, and GARCH (Pavlyshenko, B.M., 2019). The Pavlyshenko has described sales prediction as a regression problem rather than a time-series analysis. Different regression models can give a better result as compared to forecasting models. The main assumption the author Pavlyshenko took from the regression model is that the patterns in the past data will be repeated in the future. The author describes that while performing forecasting techniques the obtained observation may be biased on the validation set concerning the original values. To minimize the biasing of the model generalization of the machine learning model must be done so that proper accuracy can be obtained from it. The generalization of the model helps developers to get a more accurate result which is more resilient to the noise.

Demand Forecasting uses data where the effect of promotions or shortages has been corrected, intending to reflect the actual market demand, and tradition forecasting directly collects the data from POS(point of Scale). For supply chain management Demand & Sales, forecasting is of the import element, to reduce the information Lag between the final client and the n-tier supplier. The most widely known statistical techniques meant to forecast are also called traditional forecasting methods(Cadavid, Lamouri and Grabot, 2018). The traditional forecasting method mainly follows the time-series, they are applied under the hypothesis that the past demand can statistically estimate future demand. Hence traditional forecasting and regression both machine learning techniques can be used for the sales analysis

Measuring accuracy is one of the important aspects of validating the machine learning model. Without checking the accuracy data cannot be deployed in the real world. Mean absolute percentage error(MAPE) is one of the best formulas to validate the forecasting models(Pradhan and Kumar, 2019). Another mathematical formula to calculate the accuracy is the root mean square error(RMSE), Lower RMSE values are good for prediction(Pradhan and Kumar, 2019).

In 2018, Author Spyros Makridakis described two measuring accuracies in the article (Makridakis, Spiliotis, and Assimakopoulos, 2018) were symmetric Mean Absolute Percentage Error (sMAPE) and the Mean Absolute Scaled Error (MASE). The problem with sMAPE is that it doesn't penalize the negative error much. MASE is mainly used with seasonal time-series analysis. Therefore MAPE and RMSE measuring techniques can also be used to calculate the accuracy with sMAPE and MASE

## 2.4 CLUSTERING

One of the most frequently used supervised machine learning algorithms is Clustering. Clustering creates a homogenous group of entities for better management of it. To derive clusters, the clustering algorithm uses distance formula to create similarity of dissimilarity between entities(Pradhan and Kumar, 2019). Types of distance metrics currently are EUCLIDEAN, MINKOWSKI, JACCARD SIMILARITY COEFFICIENT, COSINE, and GOWER’s. One of the methods authors described finding the definite number of cluster is using the Elbow curve method(Pradhan and Kumar, 2019).

Assigning centroid is the most important step to obtain the equal distance between the entities. K-Means clustering can also be used pattern recognition, data mining, and cloud computing(Hamdan Ali and Emad Kadhum, 2015).

One of the clustering techniques discussed in the article(Pierre and Fredrik, n.d.) was DBSCAN (Density-Based Spatial Clustering of Applications with Noise)(Pierre and Fredrik, n.d.). DBSCAN is a density-based unsupervised machine learning algorithm given by Hans-Peter Kriegel, Martin Ester, Xiaowei Xu, and Jörg Sander in the year 1996(Pierre and Fredrik, no date). The density-based algorithm captures the insight from the point if it belongs to the cluster and looks for another cluster if it is close to another cluster point, One of the advantages of using DBSCAN is that it can find arbitrary shapes of clusters(Pierre and Fredrik, n.d.). It depends on two parameters, a positive number epsilon, and the minimum number called minPoints. Pierre also described the K-Means technique which is a very common unsupervised machine learning algorithm(Pierre and Fredrik, n.d.). The K-means is a centroid-based clustering algorithm. K-means calculation works in 2 stages, in the initial step, all information focuses are doled out to the cluster with the closest centroid. In the subsequent advance, all clusters recalculate and refresh the centroids area location on the mean of all information directs doled out toward their clusters(Pierre and Fredrik, n.d.).

The author describes Clustering is a powerful machine learning technique for detecting structures in datasets(Alashwal et al., 2019). Alashwal Mainly focusing on the techniques which can be applied to the dataset of neurological diseases(Alashwal et al., 2019). Techniques that were discussed by Alashwal are K-MEANS, K-Means-Mode, Multilayer Clustering, and Hiereacrical Agglomerative clustering(Alashwal et al., 2019). K-Means-Mode is a type of algorithms that can deal with both numeric and categorical data, K-Means-Mode works on the distance formula where each point is moved to the minimum distance to the cluster(Alashwal et al., 2019).In the study, K-Means-Mode was used to predict the likelihood of disease. In the Multilayer clustering process similarities between each pair is determined. Artificial binary classification is done on the records where the algorithm distinguishes between the defined classes and group them accordingly(Alashwal et al., 2019). The article also discussed the Hiereacrical Agglomerative clustering, it is a bottom-up approach with the end goal that every information point starts in a different cluster, and matches of clusters at the bottom are consolidated as it goes up the hierarchy(Alashwal et al., 2019). Many distance formulas can be applied such as Euclidean and Manhattan distance.

In 2014, The author Mythill and Madhiya described how the clustering algorithm can be used in data mining(Mythili and Madhiya, 2014). Algorithms suggested are Partitional Algorithms, Nearest Neighbor Clustering, Fuzzy Clustering. Fuzzy clustering is a type of clustering in which each data point can belong to more than one cluster(Mythili and Madhiya, 2014).

The author Oyelade and Oladipup et al.. described that monitoring the progress of the students is a critical issue for higher learning this can be done by one of the machine learning techniques that is clustering(Oyelade, Oladipupo and Obagbuwa, 2010). Clustering is divided into two segments hierarchical clustering and non-hierarchical clustering techniques. Means, single linkage and complex linkage are some examples of hierarchical clustering, some examples of Non-hierarchical techniques include k-means, adaptive k-medoids, k-means, and fuzzy clustering(Oyelade, Oladipupo and Obagbuwa, 2010). The author suggested that clustering will discover key characteristics from the student's performance can be used for future prediction, with the help of Euclidean distance similar students score can be chosen for analysis.

Machine learning is growing in every sector rapidly. The current article discusses the use of unsupervised machine learning methods for phenotyping and discovery(Khalid and Alhambra Prieto, 2019). The authors Khalid and Alhambra performed a subgroup or cluster within the population. Encoding is one of the features of unsupervised machine learning used for labeling the data. Further in the article, Sara Khalid discussed the two more clustering analysis apart from K-Means and Hierarchical clustering those are internal cluster evaluation and external cluster Evaluation(Khalid and Alhambra Prieto, 2019). The purpose of internal cluster evaluation is to judge, how well the subgroups data are trained within the cluster. It can be validated by checking if the individual clusters are homogenous. For evaluating clusters, Internal Evaluation is not sufficient for that External evaluation can be performed using information about the population that was not used as a feature in the cluster model.

In one article author, Pierre and Fredrik suggested that the K-Means clustering method can use for phenotyping and discovery(Khalid and Alhambra Prieto, 2019), and in another article, the author(Pierre and Fredrik, no date) used DBSCAN (Density-Based Spatial Clustering of Applications with Noise). The density-based algorithm captures the insight from the point if it belongs to the cluster and looks for another cluster if it is close to another point. Hence DBSCAN can also be used for phenotyping. It can also be used for feature selection to perform cluster within the population.

## 2.5 CRISP-DM

CRISP-DM is a freely available model that has become the leading methodology in data mining(Wowczko, 2015). CRISP\_DM follows six-steps.

Business Understanding: Focuses on understanding the project objectives and requirements from a business perspective, and then converting this knowledge into a data mining problem definition and a preliminary plan(Vorhies, 2016)

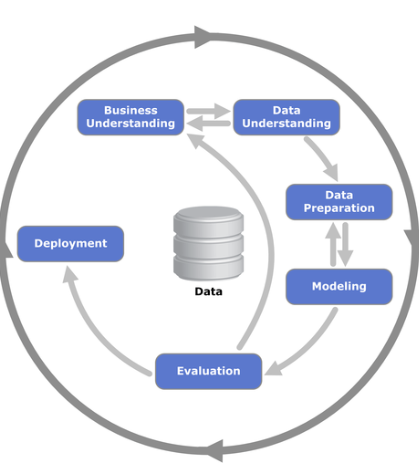
Data Understanding: Data understanding is the process of collecting data from a different source and to understand the data(Wowczko, 2015). To understand the CRISP-DM model data must be present, without the data model can build or used. Data understanding gives the author a clear view of what data is telling about. what can be obtained from it?

Data Preparation: It is one of the most important steps where the raw data is transformed into the final dataset(Vorhies, 2016). To prepare the final dataset for the machine learning model data preparation method is used for it. All the Null and NAN values are handled in the data preparation method, if there is an outlier in the dataset it is handled accordingly.

Modeling: Modeling techniques are selected and applied.  In this section one of the machine learning algorithms related to the data is applied and checked for accuracy(Vorhies, 2016).

Evaluation: Once one or more models have been built that appear to have high quality based on whichever loss functions have been selected, these need to be tested to ensure they generalize against unseen data and that all key business issues have been sufficiently considered.  The result is the selection of the champion model(s)(Vorhies, 2016).

Deployment: Deployment plan, deployment execution, and monitoring, final report, and review of the project(Wowczko, 2015).



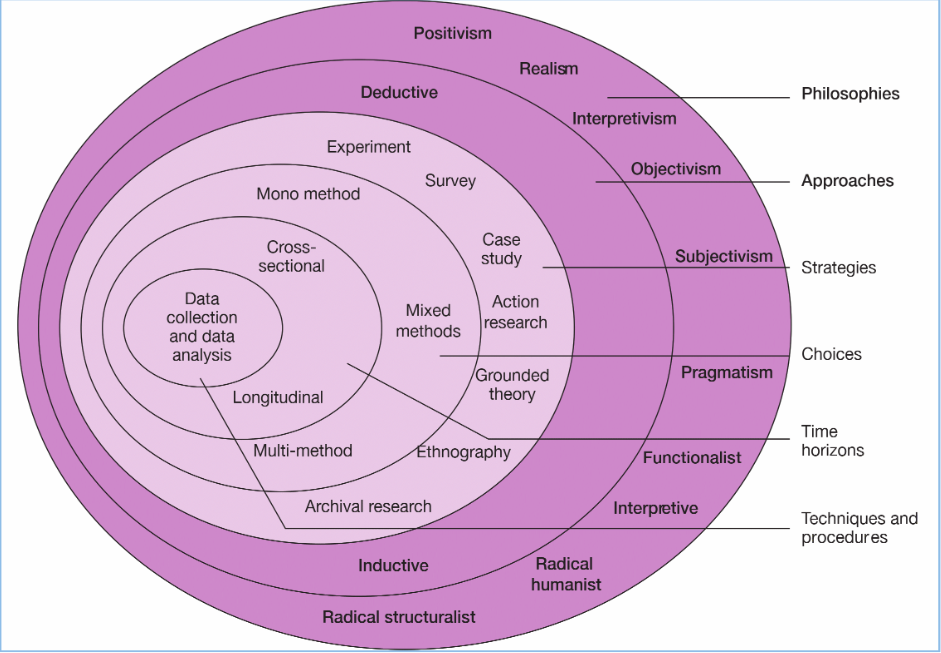
1. :(CRISP-DM)(Vorhies, 2016)

# Chapter 3 – RESEARCH METHODOLOGY

## 3.1 INTRODUCTION

As it is shown in the title, this section incorporates the research methodology of the thesis. In this section author mainly discuss the research strategy, the research method, the research approach, the methods of data collection, the selection of the sample, the research process, the type of data analysis, and the limitation(Langos, 2018).

The figure below explains the research onion. The main purpose of research onions is to obtain the goal of the research project, onion is divided into the 6 layers, these layers guided the author to obtain the goal of the project. The main objective of this research was to find the types of crimes that are being carried out in India and forecast the past crime for the next 6 years(Dissertation writers, 2019).



1. :The research ‘onion’ (Saunders, Lewis and Thornhill, 2009)

The first step of the research onion is philosophies, in this author looks for the objectives for the research purpose and the nature of reality to be investigated. The primary purpose of the philosophies is to collect the data first, in this thesis, the historical crime data was collected from the website data.gov.in. The goal of the second layer of research onion describes the approach in terms of being deductive or inductive. The deductive approach tends to flow from generic to specific. The purpose of using deductive reasoning is that it would start with theory and move on to the primary question which is tested through data collection afterward. The Inductive approach is often used or helpful when there is little research available on a topic(Dissertation writers, 2019). This research adopted the deductive approach and looked for the trend of crimes committed in India. The third layer explains the process of collecting data. The data collection method includes an experiment, survey, case study, action research, grounded theory, ethnography, and archival research. In this research, the approach author adopted was an archival research strategy as the existing dataset was available. The fourth layer of the research onion is concerned with qualitative and quantitative methodologies. In this research, the methodology that the author adopted was quantitative as the dataset contains the count of all the crimes that are being committed in India. Layer 5 of the research onion is to do with the timeframe for the research. There are two options for a researcher to choose from, i.e. cross-sectional and longitudinal. This project followed the cross-sectional approach as it had the time constraint. The final layer of the research onion looks at data collection and analysis, for this research data was collected from the website data.gov.in(Ministry of Home Affairs India, 2016)

## 3.2 RESEARCH STRATEGY

The research held with this thesis was not particularly new, but the applied one. The author studied numerous existing work on crimes in India and gather information on it. Research papers that were gathered by the author were on statistics of crimes in India, the reason behind crimes in India, and the rise of hate crimes and caste crimes in India, and overall crime rates in India. The main objective of the author was to forecast the trend of crimes in every state in India for the next 6 years

## 3.3 RESEARCH METHOD

To achieve the objectives of the research exploratory study was done by the author. An exploratory study is done to obtain the new insight from the given data and to conduct a descriptive analysis to understand the past data. This research also follows the CRISP-DM model. CRISP-DM is a standard methodology to ensure a better outcome(Vorhies, 2016) which was discussed in chapter 2. CRISP-DM here covers every step that is involved from business understanding to the deployment. CRISP-DM provides the guidelines for an organized and transparent execution of any project(Wowczko, 2015).

**Business understanding** gives the author a clear understanding of the total IPC crimes are that being committed in India and it helps the author to define the strategy and convert it into data mining. The tasks that are required to obtain are the main focus of business understanding in this thesis. The main objective earlier of the thesis is to forecast the trend of crimes for the next 6 years for every state in India and besides the author is trying to find the majority of crimes that are being committed in India, and trying to identify the worst states with their respective district.

**Data understanding** gives the author an overview from where the dataset can be obtained. In data understanding, the author can look for the required columns that are necessary to obtain the given task, A brief view of a dataset can be known from the data understanding. All the information about the columns whether it is textual data or numerical data can be known. All these pieces of information can be obtained from the data understanding such as if the values in the columns are missing or not, and the range of data in each column. Data verification and data exploration are two main subparts of the data understanding. In data, verification obtained data is verified and checked if it is fit for the model validation, before creating any machine learning algorithm. In the data exploration section, all the possible insight is obtained, in this thesis, the crimes which are affecting the parts of India are analyzed. The author obtained the dataset for analyzing district wise crimes from the website data gov.in(Ministry of Home Affairs India, 2016).

**Data preparation** is one of the most important methods while preparing the final dataset. The author thoroughly checked all the NULL and NAN values and dealt accordingly, the author also looked for the outliers in the dataset, outliers can cause a problem while developing a machine learning model. Some columns were renamed according to the crimes and the states of India. The geographical dataset was also used for analyzing the total IPC crimes in India, it was obtained from the GisDataCollection(GISMAP, n.d.). The geographical map of India was used with crimes dataset so that top crimes in India can be plotted on every state of India.

**Modeling:** Modeling is done to get the desired output.In this research, the main objective of the author was to forecast the trend of crimes in every state in India for the next 6 years. In the dataset, column total IPC crime was used to build the ARIMA model. The forecasting model was built on every state in India, and it was plotted on the graph. Another objective for the author was to cluster the district of the top 15 states in India. While building the clustering model, the K-Means algorithm was used. K-means machine learning algorithm was briefly discussed in the literature review. All the selected columns while building the clustering algorithm were normalized. Normalization is a technique used to scale down all the columns within the range of 0 to 1. K-Means algorithms make the cluster within the dataset, In this dataset, the elbow curve method was used to define the clusters. In this project, clustering will help the author to determine the worst affected district. The district will be clustered together and they will be divided into the cluster\_ids.

**Evaluation** of the model is one of the most important aspects of the CRISP-DM method. Evaluation is mostly done after the machine learning model has been built. The dataset available from 2001 to 2013 helps to verify the accuracy of the training dataset with the testing dataset. Dataset was divided into subparts. In one of the parts, the ARIMA model was build which was discussed in chapter 2. The machine learning model was built from 2001 to 2010 and further compared the prediction of training dataset with the testing dataset from 2011 to 2013. Validating the model with the testing dataset is very important as only then can any assumptions be made if it is working properly or not. To check the accuracy of the working model many derived methods are available to verify it. One of the methods which were used to verify the accuracy was the Mean absolute percentage error(MAPE). MAPE is one of the best formulas to validate forecasting models(Pradhan and Kumar, 2019). Another way to validate the forecasting model is the root mean square error(RMSE) which was discussed in Chapter 2.

## 3.4 METHOD OF DATA COLLECTION

Understanding the Business need is very important and for this task forecasting total IPC crimes committed in every state for the next 6 years is very important, the data set used for this thesis is “**District-wise crimes under various sections of Indian Penal Code (IPC) crimes**”. This dataset was obtained from the website data gov.in(Ministry of Home Affairs India, 2016). The short description of the variables used in the data set is in Table.

This dataset contains the count of all major crimes that were reported in India from 2001 to 2013. Currently, in India, there are 28 states and 8 union territories and 718 districts as per record(Government of India, n.d.). In this dataset, the total number of states that were present were 28 and 7 union territories, 2019 the status of Jammu and Kashmir was downgraded to union territory from state. All the crimes that are recorded are for every state and their respective districts.

### 3.4.1 OVERVIEW OF THE DATASET

|  |  |
| --- | --- |
| **Feature name** | **Feature description** |
| **STATE/UT** | Text Value which denotes all the Name of the state and Union Territories |
| **DISTRICT** | Text Value which covers all the urban and rural cities of every state in India |
| **YEAR** | Numerical Value denoting the year's when the crimes have occurred from 2001 to 2013 |
| **THEFT** | Numerical Value that denotes the count of theft that happened in India |
| **ATTEMPT TO MURDER** | Numerical Value that denotes the count of attempt to murder that happened in India |
| **CULPABLE HOMICIDE NOT AMOUNTING TO MURDER** | Numerical Value that denotes the count of culpable homicide not amounting to murder |
| **RAPE**: | Numerical Value that denotes the count of Rape that happened in India |
| **CUSTODIAL RAPE** | Numerical Value that denotes the count of Custodial Rape that happened in India |
| **OTHER RAPE:** | Numerical Value that denotes the count of Other Rape that happened in India |
| **KIDNAPPING & ABDUCTION** | Numerical Value that denotes the count of Kidnapping and Abduction that happened in India |
| **KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS** | Numerical Value that denotes the count of Kidnapping and Abduction of Women and Girls that happened in India |
| **KIDNAPPING AND ABDUCTION OF OTHERS:** | Numerical Value that denotes the count of Kidnapping and Abduction of Others that happened in India |
| **DACOITY** | Numerical Value that denotes the count of Dacoity that happened in India |
| **PREPARATION AND ASSEMBLY FOR DACOITY** | Numerical Value that denotes the count of PREPARATION AND ASSEMBLY FOR DACOITY that happened in India |
| **ROBBERY** | Numerical Value that denotes the count of Robbery that happened in India |
| **BURGLARY** | Numerical Value that denotes the count of Burglary that happened in India |
| **THEFT** | Numerical Value that denotes the count of Theft that happened in India |
| **AUTO THEFT** | Numerical Value that denotes the count of Auto theft that happened in India |
| **OTHER THEFT** | Numerical Value that denotes the count of Other theft that happened in India |
| **RIOTS** | Numerical Value that denotes the count of Riots that happened in India |
| **CRIMINAL BREACH OF TRUST** | Numerical Value that denotes the count of Criminal breach of trust that happened in India |
| **CHEATING** | Numerical Value that denotes the count of Cheating that happened in India |
| **COUNTERFEITING** | Numerical Value that denotes the count of Counterthefting that happened in India |
| **ARSON** | Numerical Value that denotes the count of Arson that happened in India |
| **HURT/GRIEVOUS HURT** | Numerical Value that denotes the count of Hurt that happened in India |
| **DOWRY DEATHS** | Numerical Value that denotes the count of Dowry deaths that happened in India |
| **ASSAULT ON WOMEN WITH INTENT TO OUTRAGE HER MODESTY** | Numerical Value that denotes the count of Assault on women with intent to outrage her modesty that happened in India |
| **INSULT TO MODESTY OF WOMEN** | Numerical Value that denotes the count of Insult to the modesty of women that happened in India |
| **CRUELTY BY HUSBAND OR HIS RELATIVES** | Numerical Value that denotes the count of cruelty by husband or his relatives that happened in India |
| **IMPORTATION OF GIRLS FROM FOREIGN COUNTRIES** | Numerical Value that denotes the count of Importation of girls from foreign countries that happened in India |
| **CAUSING DEATH BY NEGLIGENCE** | Numerical Value denotes the count of causing death by negligence that happened in India. |
| **OTHER IPC CRIMES** | Numerical Value that denotes the count of other IPC crimes reported that happened in India. |
| **TOTAL IPC CRIMES** | Numerical Value that denotes the count of all the total crimes combined that happened in India. |

Table 3.4.1.1: Overview of the Dataset

### 3.4.2 VISUALISING tHE DATA

One of the core aspects to get insight from the data is visualization. Data visualization is the demonstration of taking (data) and setting it into a visual setting, for example, a guide or chart. Data visualizations make of all shapes and size data simpler for the human mind to understand, and visualization additionally makes it simpler to detect patterns, trends, and outliers in gatherings of data. Great data visualizations should put significance into confounded datasets with the goal that their message is clear and compact(Heitzman, 2019).

This thesis analyzes the past crimes that happened in India from 2001 to 2013. Data visualization techniques will help the author to find in-depth insights from the data such as the worst affected state in India, the count of crimes that are being committed in India by criminals, and which is the most frequent crime, etc.

A pie chart is a method of showing information where a circle is partitioned into sections (or cuts") that mirror the relative size or recurrence of the classifications(Salkind J, n.d.). In this thesis, the author was trying to identify the top crimes that are being committed in India by criminals. This helped the author to classify the count or frequency of each crime so that it can be further analyzed which district or which state is worst affected concerning the top crimes in India. Line-chart was also used by the author to see the trend of top crimes India, Line chart was mainly used to depicts the changes in values over the period(Yi, 2019). In this dataset author, was trying to demonstrate the changes in values of crimes committed from 2001 to 2013. The bar graph was created by the author to get more in-depth detail of the worst affected district. Plotting the bar chart with respect to the count of crimes and districts of particular states.

## 3.5 DESIGNING

### 3.5.1 SELECTION OF MODEL

The output from model training might be utilized for the deduction, which means making expectations on new data. A model is a refined portrayal of what a machine learning framework has learned. Machine learning models are likened to mathematical capacities they take permission in the form of input data, make a forecast or prediction on that input data, and then serve a reaction. In supervised and unsupervised machine learning, the model depicts the sign in the noise or the example distinguished from the training data(AI Wiki, n.d.).

A machine learning model that was used to forecast the crimes in India for the next 6 years for every state was ARIMA. The auto-regressive integrated moving average (ARIMA) is a time series based model, the ARIMA model uses its variable on itself to predict the outcome. In the research, the author used total IPC crimes committed and years to forecast the crime rates for the future. Another method that was used by the author in this project was clustering and the machine learning model that was build was K-Means. The clustering technique helped the author detect the patterns in the dataset.

### 3.5.2 TOOLS USED FOR ANALYSIS

The most common programming languages are that are used for building the machine learning model and for data analysis are R-programming language, C++, and python programming language.



Figure 3.5.2.1: (Common Programming language)(Mackenzie, 2019)

Python programming language is the most used language in the world for data science with over 83% of users(Mackenzie, 2019). Python programming language has some excellent machine learning libraries and visualization libraries that help in building a machine learning model. Python has an array of machine learning libraries most common libraries are Tensor flow, sci-kit learn, PyTorch, and OpenCV. The author used the ScikitLearn library to build the clustering model, and another library that was used for building the ARIMA was the stats-model library. Both ScikitLearn and stats-model are an excellent library and used in the analysis which provides various features for data manipulation and analysis tasks to do analysis.

The IDE(Integrated Development Environments) that was used by the author to build the forecasting model was Jupyter Notebook. It can be used with the help of Anaconda Command Prompt, it was very handy to use. Jupyter is a free, open-source, interactive web tool known as a computational notebook, which researchers can use to combine software code, computational output, explanatory text, and multimedia resources in a single document(Perkel M, 2018). Jupyter notebook also provides the markdown file function where the online report can be generated within the instance and all the given code can be explained in that with the help of comments on it. Other IDE such as spyder, pycharm, etc are comparatively slower than the Jupyter in terms of .processing and speed, Techniques like data-cleaning, data exploration, and plotting and model building are quite easy as compared with others.

### 3.5.3 MEASURING ACCURACY

The purpose of the machine learning models is to predict the outcome with the best accuracy score. In this thesis, the author was trying to forecast the total IPC crimes committed in India for the next 6 years. It was assumed that the prediction might vary little, but finding the accuracy of the model with a good score was an important step. The two most popular techniques that are used to calculate the forecast accuracy are

1) Root Mean Square Error(RMSE)

2) Mean Absolute Percentage Error(MAPE).

The technique that the author used in this thesis was the Mean Absolute Percentage Error(MAPE). To use this technique the author created the python function that takes the actual and predicted value and returns the value of mean absolute percentage error. Since MAPE is dimensionless, it can be used for comparing different models with varying scales(Pradhan and Kumar, 2019).

The author also used clustering techniques to identify the patterns within the districts of India. In clustering, there is no measuring accuracy technique like the forecasting method. The clustering approach uses the K-Means model to identify the patterns with the help of centroids. The following techniques to identify the centroids are.

1) Using Dendrogram

2) Elbow Curve method

The technique that was identified by the author to find the centroid was the Elbow curve method. In the K-Means clustering algorithm, the interia\_ parameter was used. This provides the total variance for a particular number of clusters(Pradhan and Kumar, 2019).

## 3.6 IMPLEMENTATION

### 3.6.1 STEPS INVOLVED IN DATA EXPLORATION

* Importing the data, from data.goi website, as a CSV file.
* Importing all the necessary libraries for the thesis
* Checking for all the information such as the number of columns and rows present in this dataset there are 33 columns present.
* Types of variables: 2 categorical and 30 numerical.
* Checking the count of all the crimes and states
* Checking for the NULL or NAN value in the dataset
* Merging the crime data with geographical data that was collected from the website GisDataCollection.

### 3.6.2 MODELLING APPROACH

The model that was selected for this research to predict the total IPC crimes is ARIMA. As there was no independent variable present in the dataset, the author used the time-series analysis to approach this objective. For prediction, there are many algorithms such as linear regression, logistic regression, and tree-based approach, these all algorithms require to depend on the variable to predict the outcome. In time series analysis the outcome variable depends on itself to predict the future. In this thesis, the column total IPC crimes is dependent on the number of years to predict the outcome. The second model that the author build was K-Means clustering, it was used to detect the pattern. The worst affected city of clustered in one group similarly least affected city was clustered in another group.

### 3.6.3 ARIMA MODEL

An [ARIMA model](https://en.wikipedia.org/wiki/Autoregressive_integrated_moving_average) is a class of statistical models for analyzing and forecasting time series data.

It explicitly caters to a suite of standard structures in time-series data, and as such provides a simple yet powerful method for making skillful time-series forecasts(Brownlee, 2017). ARIMA is an acronym that stands for AutoRegressive Integrated Moving Average. It is a generalization of the simpler AutoRegressive Moving Average and adds the notion of integration. This acronym is descriptive, capturing the key aspects of the model itself. Briefly, they are:

* **AR**: Autoregression. A model that utilizes the dependent relationship between perception and some number of lagged perceptions (Brownlee, 2017).
* **I**: Integrated. The utilization of differencing of crude perceptions (for example deducting an observation from the previous perception at the past time step) to make the time series stationary (Brownlee, 2017).
* **MA**: Moving Average. A model that utilizes the reliance between a perception and a remaining mistake from a moving normal model applied to slack perceptions (Brownlee, 2017).

Standard notation is used of ARIMA(p,d,q) where the parameters are substituted with integer values to quickly implies the specific ARIMA machine learning model being used. Each of these parts of the model are explicitly specified in the model as a parameter.

* AR is a component of p lags
* Integration component(d)
* MA with q lags, MA(q)

The parameters of the ARIMA model are defined as follows:

|  |  |  |
| --- | --- | --- |
| **Parameter Name** | **Default value** | **Parameter Description** |
| P | 1 | The number of lag observations included in the model also called the lag order. |
| d | 0 | The number of times that the raw observations are differenced also called the degree of difference |
| q | 0 | The size of the moving average window also called the order of the moving average |

Table 3.6.3.1: Description Parameters used

A linear regression model is constructed including the specified number and type of terms, and the data is prepared by a degree of differencing to make it stationary, i.e. to remove trend and seasonal structures that negatively affect the regression model. A value of 0 can be used for a parameter, which indicates to not use that element of the model. This way, the ARIMA model can be configured to perform the function of an ARMA model, and even a simple AR, I, or MA model(Brownlee, 2017). In this thesis, the author divided the dataset into two parts training and testing. The training data set was trained from the year 2001 to 2010 and was further validated from 2011 to 2013.

### 3.6.4 K-MEANS CLUSTERING MODEL

For better management and to create the homogenous groups of entities most frequently is used application us Clustering. The clustering is a divide and conquer strategy which divides the dataset into a homogenous group(Pradhan and Kumar, 2019). Clustering algorithms are unsupervised learning algorithms in which labels classes are not defined, unlike supervised machine learning.

The function author used to define the clustering technique was K-Means. The K-Means uses the euclidean distance to find the distance between the two or more observation. It takes the mean(average) of every sample in the dataset and fit them accordingly.

#### 3.6.4.1 STEPS INVOLVED IN K-MEANS

1) Decide the Value of K

2) Choose the value of K, this can be done by the elbow cure method which was discussed in section 3.4.3.

3) The K observations selected in step 2 are the centroid of the cluster(Pradhan and Kumar, 2019)

4) For remaining observation find the cluster closest to the centroid.

5) Repeat step 4 until all observations are assigned to the cluster(Pradhan and Kumar, 2019).

In this research, the author first transformed the data into one scale so that outliers do not affect the model. Normalization of the data can be done with the help of the StandardScaler library then the value of K was identified using the elbow curve method for every district the model was build and the value of K was taken according to it, and ranom\_state was taken 42. A new column was defined in the data frame so that cluster\_Id can be stored in it.

### 3.6.5 LIBRARIES

Importing python libraries is one of the core aspects of completing any project, without the library not a single work can be done one the dataset. Even to read the dataset, the author must import a library. Many libraries were used while building the machine learning model and for data preparation. The libraries that were used by the author were given below with the description

|  |  |
| --- | --- |
| **LIBRARY** | **DESCRIPTION** |
| **Pandas** | The Pandas is one of the most important libraries while dealing with data, all the data analysis, and data manipulation can be done with the help of pandas |
| **Numpy** | All the mathematical functions can be computed through the NumPy library. |
| **Seaborn** | Seaborn library is used for visualization. |
| **Seasonal\_decompose** | This library was used to find the trend of  crimes in every state. |
| **Statsmodels.api** | Statsmodels.api is used for modeling purpose, to use the forecasting machine learning algorithms, statsmodels.api provides all the necessary libraries in it. |
| **Sklearn** | TheSckitlearning(Sklearn) module contains all the machine learning algorithms eg Linear Regression Classification and clustering, in this project K-Means library was used to derive the cluster |
| **Plotly** | Plotly library provides an in-depth visualization of any data |
| **Matplotlib** | Matplotlib was used for visualization |
| **Geopandas** | Geopandas was used to draw the map of India. All the worst affected states in India were plotted on the map |
| **ARIMA** | ARIMA machine learning model was used for forecasting, this model was explained in the above section |
| **Plot\_acf, plot\_pacf** | Thelags of every state were derived from the ACF. ACF stands for autocorrelation function. |

Table 3.6.5.1: Libraries used

## 3.7 LIMITATION

The limitation of this research was while creating the machine learning model. The forecasting model needs at least 20 years of data to build the model. The amount of data available was only for 13 years, the accuracy that was obtained from this model for every state was good but it could have been better if the more data was available. Another limitation was that in this data there was no label class given. There was no reason behind the crimes committed by the criminals. Machine learning models such as regression and classification cannot be applied to this data as there was no output variable present.

# Chapter 4 – FINDINGS AND ANALYSIS

## 4.1 INTRODUCTION

In this section, the author discussed all the analysis and modeling that was done. Top crimes that were committed in India from 2001 to 2013 and to forecast the trend of crimes in every state in India.

## 4.2 CRIMES IN INDIA

A screenshot of a cell phone

Description automatically generated

1. :Distribution of crimes

The above figure shows the distribution of crimes that were committed in India from 2001 to 2013. Crimes that were committed in India were

MURDER, ATTEMPT TO MURDER,

CULPABLE HOMICIDE NOT AMOUNTING TO MURDER, RAPE, KIDNAPPING & ABDUCTION, DACOITY,

PREPARATION AND ASSEMBLY FOR DACOITY, ROBBERY, BURGLARY,

RIOTS, CRIMINAL BREACH OF TRUST,

CHEATING, COUNTERFEITING, ARSON, HURT/GRIEVOUS HURT,

DOWRY DEATHS, ASSAULT ON WOMEN WITH INTENT TO OUTRAGE HER MODESTY,

INSULT TO MODESTY OF WOMEN, CRUELTY BY HUSBAND OR HIS RELATIVES,

IMPORTATION OF GIRLS FROM FOREIGN COUNTRIES,

CAUSING DEATH BY NEGLIGENCE, THEFTTOTAL

This pie chart helped the author to identify the top 10 crimes that were committed in India. All the analysis further in the section was done using top 10 crimes in India

## 4.3 TOP 10 CRIMES IN INDIA

A screenshot of a cell phone

Description automatically generated

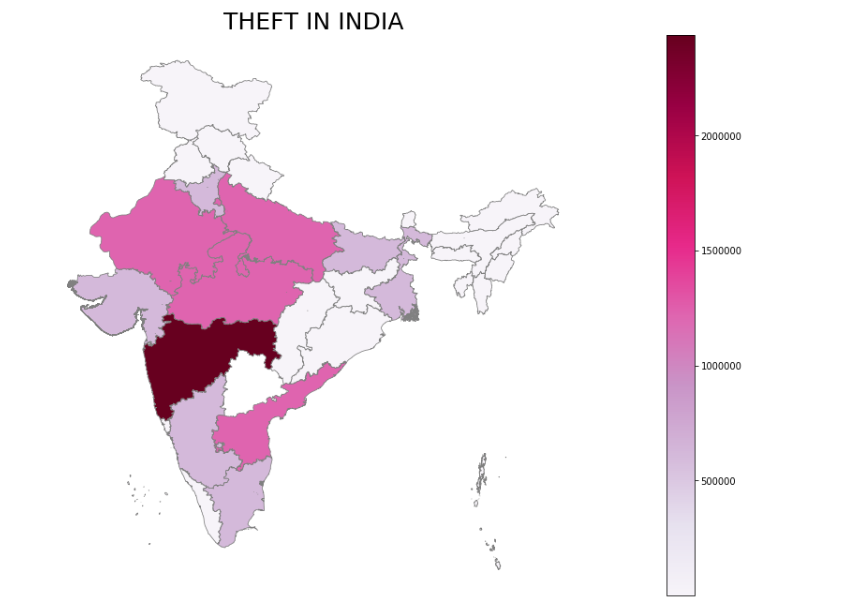
1. :Top 10 crimes India

Identifying crimes plays an important role in keeping the country safe. In India the top 10 crimes that were committed between 2001 and 2013 were:

MURDER, KIDNAPPING & ABDUCTION, BURGLARY, RIOTS, CHEATING, HURT/GRIEVOUS HURT, ASSAULT ON WOMEN WITH INTENT TO OUTRAGE HER MODESTY, CAUSING DEATH BY NEGLIGENCE, THEFTTOTAL, CRUELTY BY HUSBAND OR HIS RELATIVES.

The author combined theft, auto theft, and other theft as the Theft total. It can be observed from the pie chart that the most committed crime in India was **THEFT** with 38.4% followed by **HURT** with 22.5%. The ratio of these two crimes was the most from 2001 to 2013. Crimes whose percentage was less than 2% were **MURDER, KIDNAPPING & ABDUCTION** with 2.65% and 2.57% respectively. **BURGLARY** with 7.43% and **CAUSING DEATH BY NEGLIGENCE** with 6.77%, and were third and fourth respectively. At fifth position recoded crimes was **CRUELTY BY HUSBAND OR HIS RELATIVES** with 6.04%. Crimes that were below 6% was **CHEATING, RIOTS**, and **ASSAULT ON WOMEN WITH INTENT TO OUTRAGE HER MODESTY** with 5.32%, 5.15%, and 3.18% respectively.

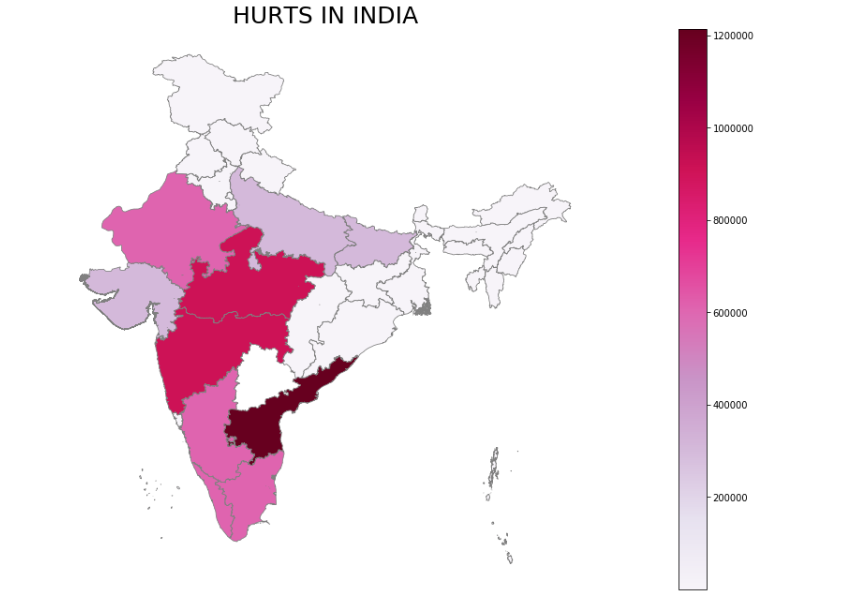
### 4.3.1.1 THEFT IN INDIA



1. :Theft In India

The theft was the most committed crime in India with a 38.4% rate. The worst affected state in India was Maharastra in terms of theft with a count of more than 2000000 was recorded. States between 1500000 and 2000000 were Madhya Pradesh, Rajasthan, Uttar Pradesh, and Andra Pradesh.

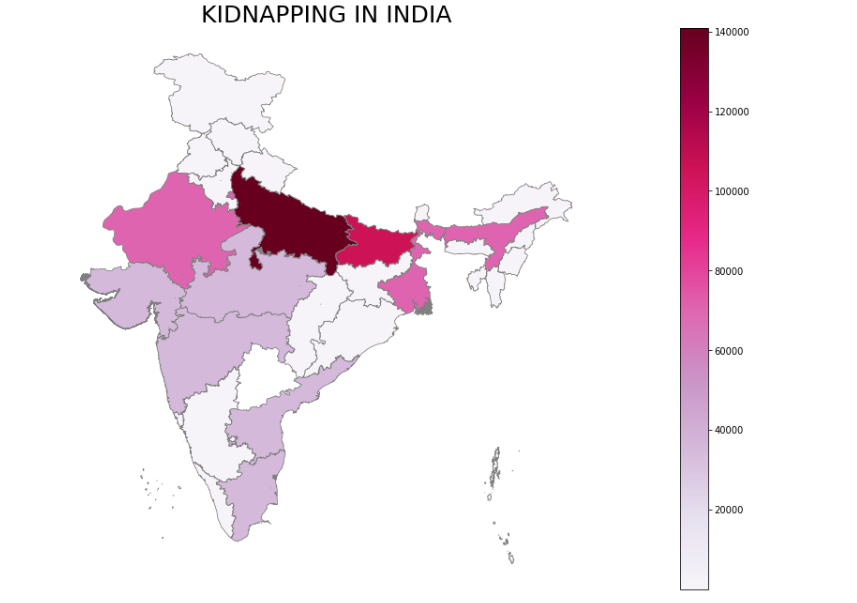
### 4.3.1.2 HURT IN INDIA



1. :Hurt In India

The Hur was the second most committed crime in India with a 22.5% rate. The worst affected state in India was Andra Pradesh in terms of Hurt crime with a count around 1200000 was recorded. States between 1000000 and 1200000 were Madhya Pradesh and Maharashtra. Other less affected states were Rajasthan, Karnataka, Tamil Nadu, and Kerela with count between 600000 and 100000.

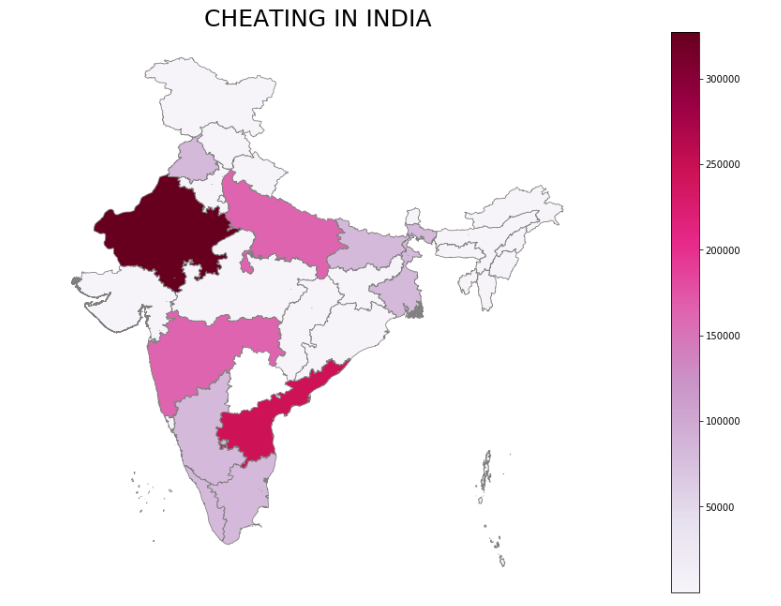
### 4.3.1.3 KIDNAPPING IN INDIA



1. :Kidnapping In India

The worst affected states were Uttar Pradesh and Bihar, the count of both of the states were more than 120000. States between 800000 and 1200000 were Assam, West Bengal, and Rajasthan. The frequency of kidnapping was as low as 2.5% in all over India. The kidnapping was the second least top crimes committed in India.

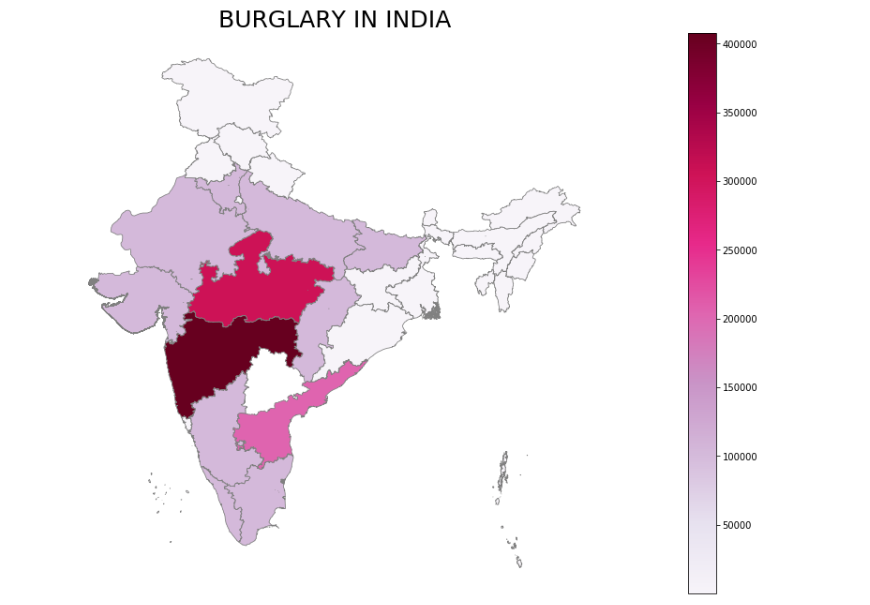
### 4.3.1.4 CHEATING IN INDIA



1. :Cheating In India

Percentage of Cheating In India is 5.32 which comes the sixth position in top crimes in India. The worst affected states in terms of Cheating is Rajasthan whose count more than 300000. States whose count is between 100000 and 300000 are Maharashtra, Andra Pradesh, and Uttar Pradesh. States like karnataka, kerela and Tamil Nadu had cases between 100000 and 150000. All other states in India have less than 100000 counts.

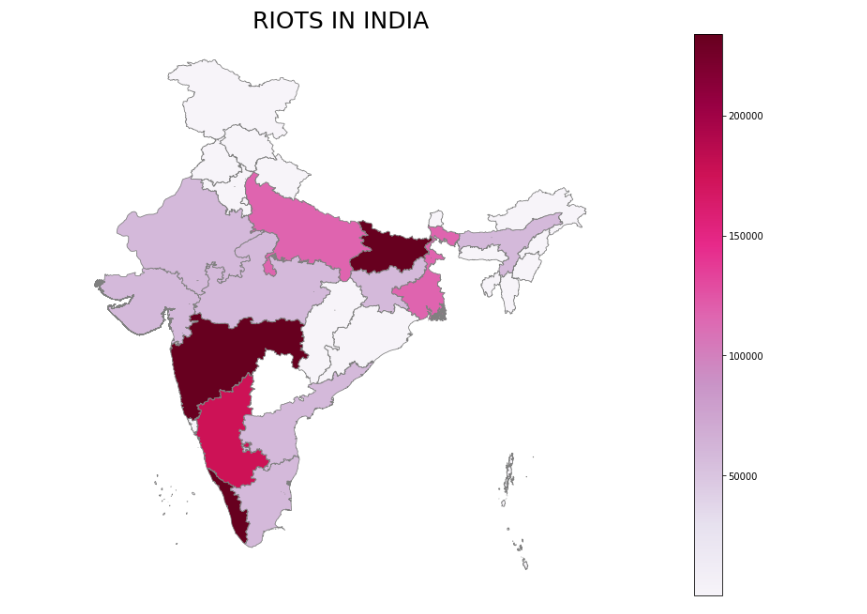
### 4.3.1.5 BURGLARY IN INDIA



1. : Burglary In India

Burglary is the third most committed crimes in India after hurt/Grevious the overall percentage of burglary committed in India was 7.43%. The worst affected states in India are Maharastra and Madhya Pradesh with a count of more than 300000. The state between 200000 and 250000 was Andra Pradesh.

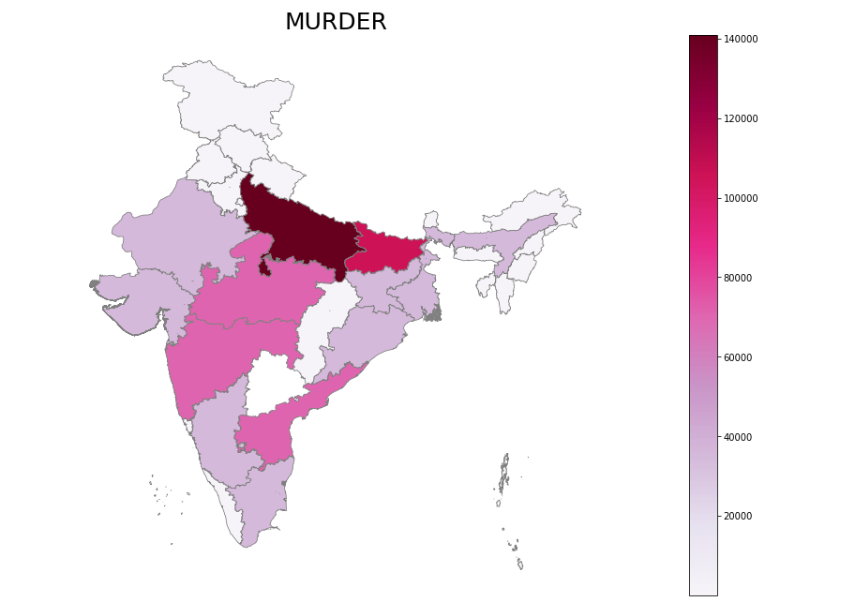
### 4.3.1.6 RIOTS IN INDIA



1. : Riots In India

The count of Riots has spread all over India, the percentage of riots was less as compared to the other top crimes in India. The frequency of riots that happened in India was around 5.15%. The states that were worst affected by the count of more than 200000 were Maharashtra, Bihar, and Kerela. The states whose count was between 150000 and 200000 were Karnataka, Uttar Pradesh, and West Bengal.

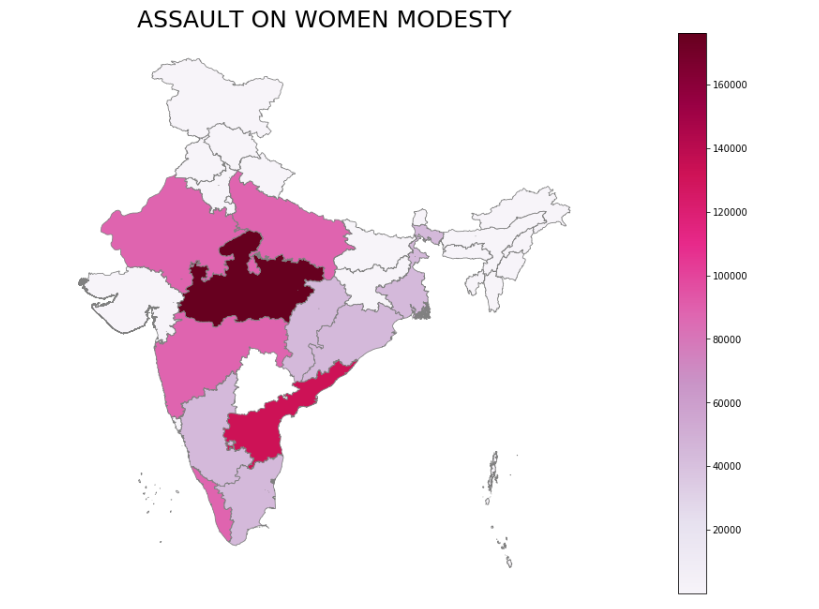
### 4.3.1.7 MURDER IN INDIA



1. : Murder In India

One of the least committed top crimes in India. Most of the Murder happens in the northern part of India, Uttar Pradesh and Bihar were the most affected states in India. The states whose count was between 60000 and 100000 were Madhya Pradesh, Maharashtra, and Andhra Pradesh. The count of all other states in India was less than 60000.

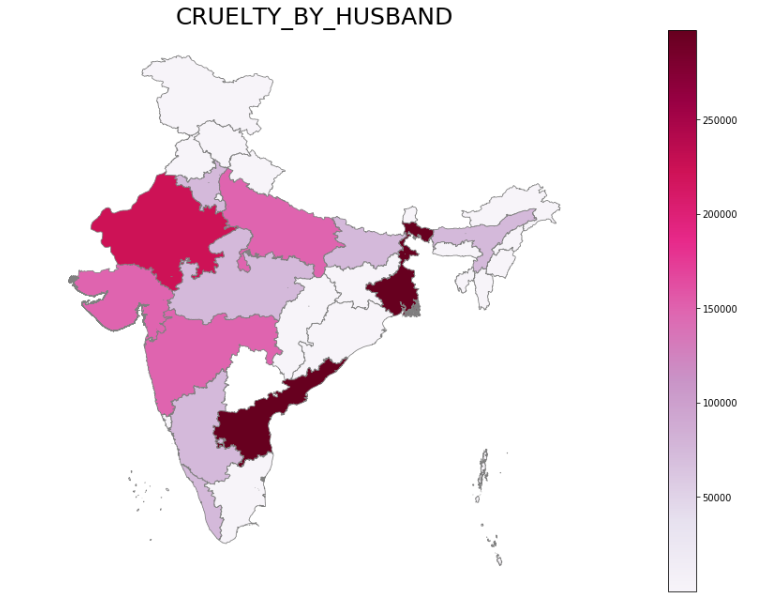
### 4.3.1.8 ASSAULT ON WOMEN MODEST IN INDIA



1. : Assault On Women With Intent To Outrage Her Modesty

Frequency of the crime Assault on women with intent to outrage her modesty is 3.18%. The most affected state was Madhya Pradesh which recorded the crime more than 160000, next most affected state was Andra Pradesh followed by Maharastra, Kerela, Rajasthan, and Uttar Pradesh. The count of all of these states recoded more than 100000.

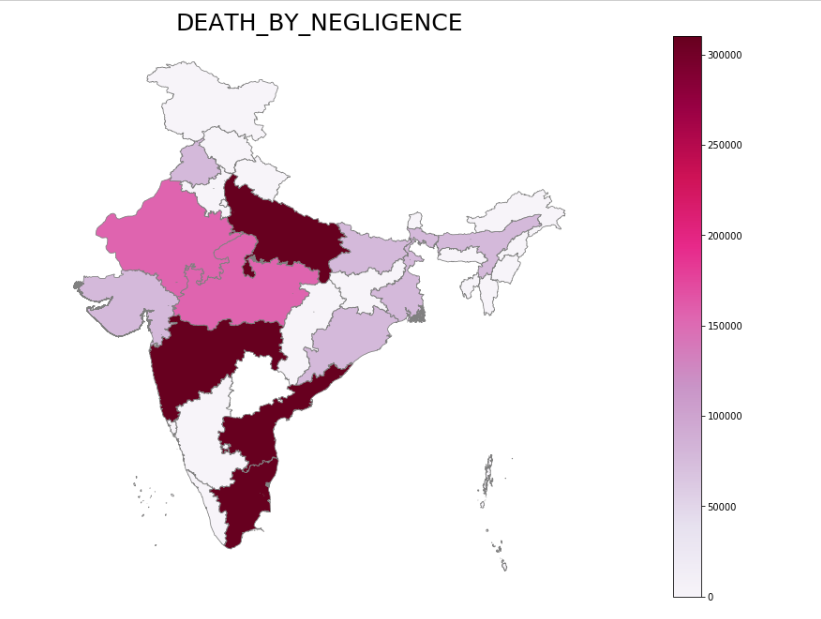
### 4.3.1.9 CRUELTY BY HUSBAND IN INDIA



1. : Cruelty By Husband

The percentage of this crime was 6.04. The most affected state by this crime was West Bengal, Andhra Pradesh which recorded more than 250000 cases, followed by Gujarat, Maharastra, Rajasthan, and Uttar Pradesh which counted more than 150000 cases but less than 250000.

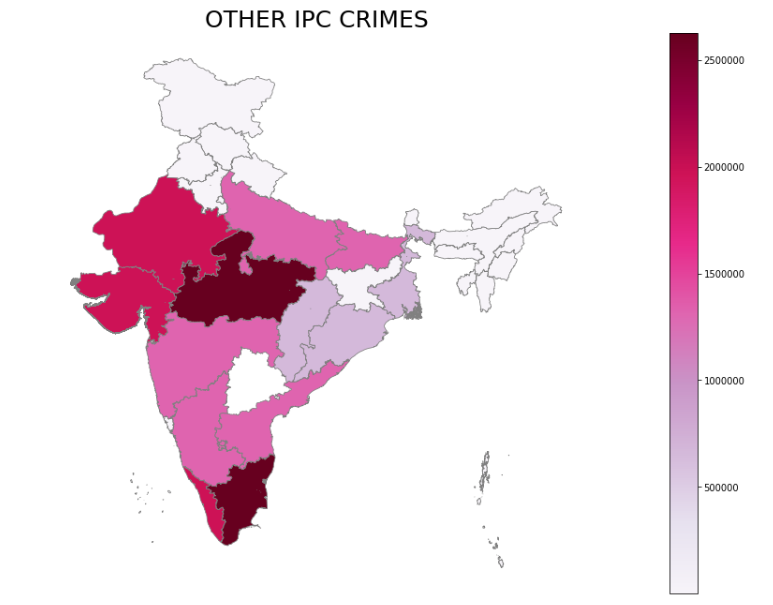
### 4.3.1.10 DEATH BY NEGLIGENCE



1. : Death By Negligence

Death by negligence was the fourth most committed crime in India with 6.7%. From this figure, it can be observed that this crime was spread almost all over India. The worst states that recorded more than 300000 cases were Uttar Pradesh, Maharastra, Tamil Nadu, and Andra Pradesh followed by Madhya Pradesh and Rajasthan with over 150000 cases. Gujarat, West Bengal, and Bihar recorded less than 100000 cases.

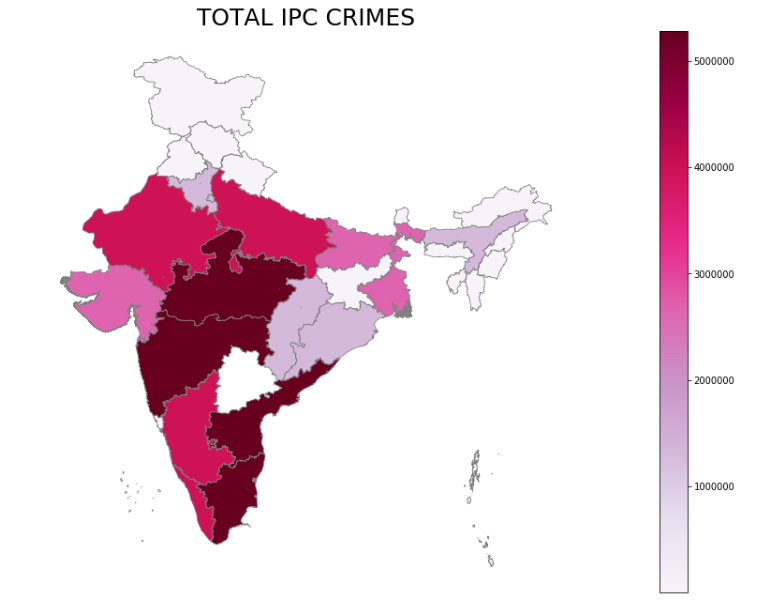
### 4.3.1.11 OTHER IPC CRIMES



1. : Other IPC Crimes

The above figure illustrates the count of other IPC crimes that happened in India. Other IPC crimes can be of any type like Cybercrime. The worst affected state in terms of other IPC crime was Madhya Pradesh and Tamil Nadu which recoded more than 2500000 cases between 2001 and 2013. Maharashtra, Karnataka, Andra Pradesh, and Uttar Pradesh recorded less than 200000 cases. States like Gujarat, Rajasthan, and Kerela recorded more than 150000 cases but were less than Madhya Pradesh and Tamil Nadu between 2001 and 2013. All other states in were below 100000 cases.

### 4.3.1.12 TOTAL IPC CRIMES IN INIDA



1. : Total IPC Crimes

The total IPC crimes are the sum of all the crimes present. All the crimes that were present in the data set were MURDER, ATTEMPT TO MURDER, CULPABLE HOMICIDE NOT AMOUNTING TO MURDER, RAPE, KIDNAPPING & ABDUCTION, DACOITY, PREPARATION AND ASSEMBLY FOR DACOITY, ROBBERY, BURGLARY, RIOTS, CRIMINAL BREACH OF TRUST, CHEATING, COUNTERFEITING, ARSON, HURT/GRIEVOUS HURT, DOWRY DEATHS, ASSAULT ON WOMEN WITH INTENT TO OUTRAGE HER MODESTY, INSULT TO MODESTY OF WOMEN, CRUELTY BY HUSBAND OR HIS RELATIVES, IMPORTATION OF GIRLS FROM FOREIGN COUNTRIES, CAUSING DEATH BY NEGLIGENCE, THEFTTOTAL. The worst affected states in India are Maharastra, Madhya Pradesh, Andra Pradesh, and Tamil Nadu which recoded more than 500000 cases India. States like Rajasthan, Uttar Pradesh, Karnataka, and Kerela recoded around 400000 cases in India. All the other states in India recorded less than 300000 cases.

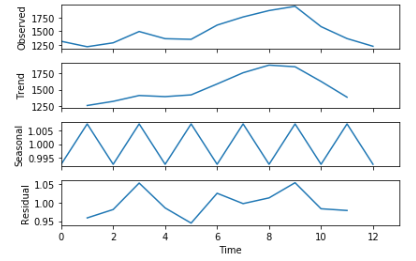
## 4.4 FORECASTING

In this section, the author builds the machine learning model that was used to forecast the trend in every state for the next six years. The author used the ARIMA model to forecast the trend.

### 4.4.1 ANDAMAN AND NICOBAR ISLAND

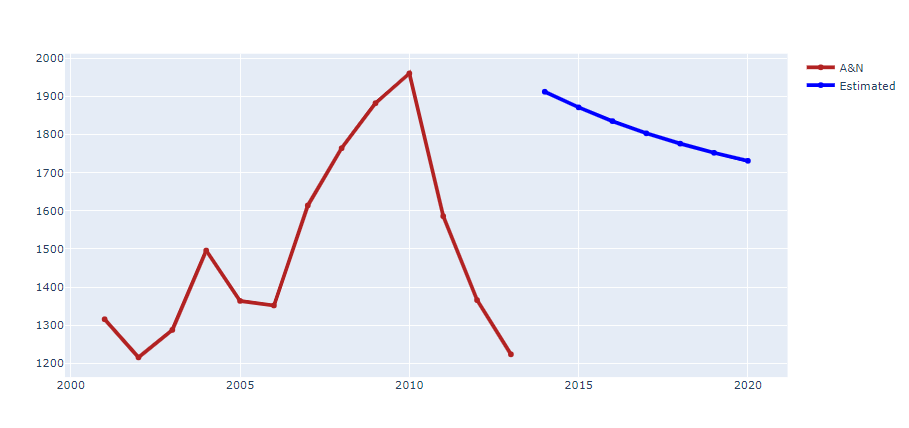
Andaman and Nicobar Island are located in the Bay of Bengal, these are the group of small 3000 islands. The capital of Andaman and Nicobar Island is Port Blair(Tourmyindia, no date). In 2019 the recoded population of Andaman and Nicobar Island was 417036(Statstics Times, 2020).

The figure below shows the trend of Andaman and Nicobar Island from 2001 to 2013.



1. : Trend of Andaman and Nicobar

From the graph, it can be observed that the trend of crimes will decrease in this state. The observed crime was around 1900 cases and in 2010 the state recorded the highest number of cases, between 2001 and 2002 the least amount of cases were recorded.



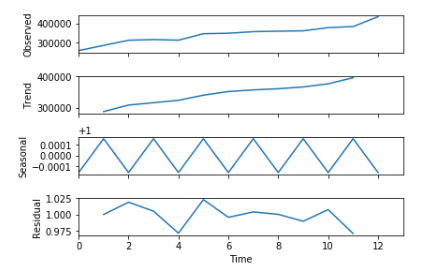
1. : Forcast in Andaman and Nicobar

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will increase in 2014 and slowly it will decrease till 2020. The model predicted that the cases in 2014 will around 1900 and until 2020 it will go to 1750 cases. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 35.84%.

### 4.4.2 ANDHRA PRADESH

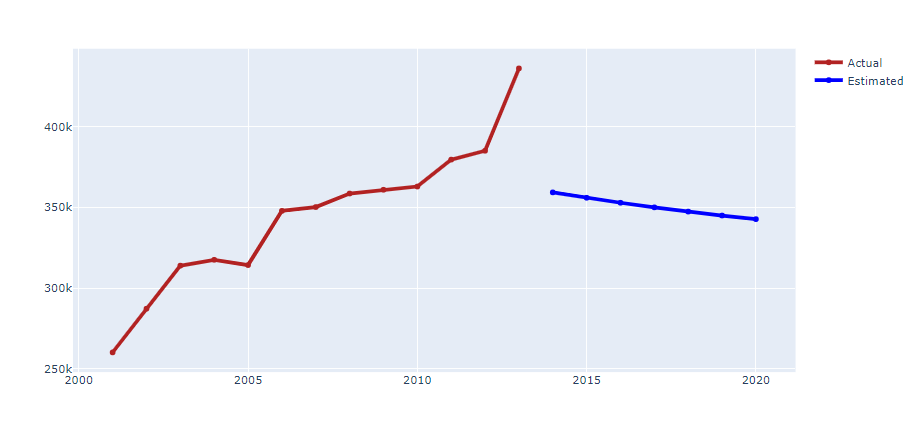
Andhra Pradesh is recognized widely for its legendary dynasties, the revered temple Tirupati, the beautiful language Telugu(Tourmyindia, n.d.). The capital Hyderabad is of the major cites in India. In 2019 the recoded population of the state was 53,903,393(Statstics Times, 2020).

The figure below shows the trend of Andhra Pradesh Island from 2001 to 2013.



1. : Trend Chart In Andhra Pradesh

From the graph, it can be observed that the trend of crimes will follow the observed trend in this state. The observed crime was more than 40000 cases and from 2002 the number of cases has been increased over the years, between 2001 and 2002 the least amount of cases were recorded.



1. : Forcast of Andaman and Nicobar

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 till 2020. The model predicted that the cases in 2014 will more than 350k and until it reaches 2020 it will go below 350k cases. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 10.66%. In 2013 the highest number of cases was recorded 415000.

### 4.4.3 ARUNACHAL PRADESH

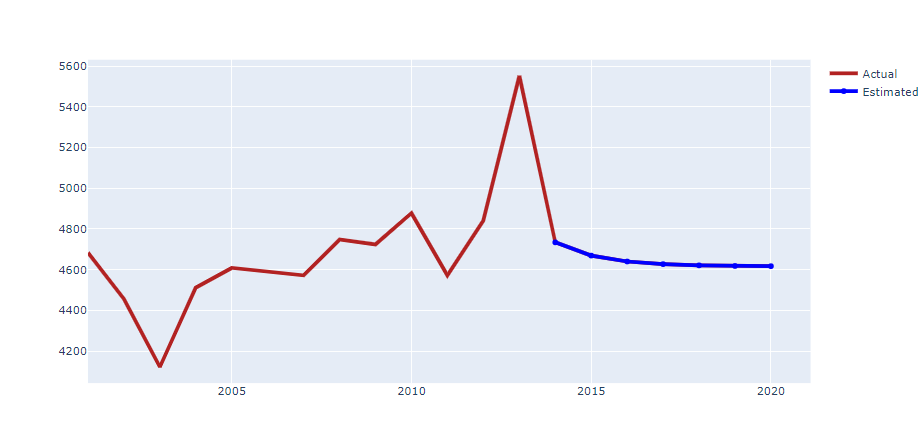
Adorned with beautiful and unspoiled rivers, lakes, hills and valleys along with opulent wildlife and culture. Arunachal Pradesh shares its southern boundary with Assam, and on its west is Bhutan, whereas, China makes the north(Tourmyindia, n.d.). The population in 2019for this state was 1,570,458(Statstics Times, 2020)

The figure below shows the trend of Arunachal Pradesh and from 2001 to 2013.



1. : Trend of Arunachal Pradesh

From the graph, it can be observed that the trend of crimes will follow the observed trend in this state. The observed crime was more than 5000 cases and from 2002 the number of cases has been increased over the years, between 2001 and 2002 the least amount of cases were recorded.



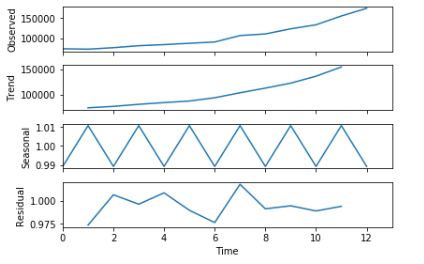
1. : Forecast In Arunachal Pradesh

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 till 2020. The model predicted that the cases in 2014 will around 4700 and until it reaches 2020 cases will remain almost constant. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 98.3%. In 2011 and 2012 the state recorded the highest number of cases around 5500.

### 4.4.4 ASSAM

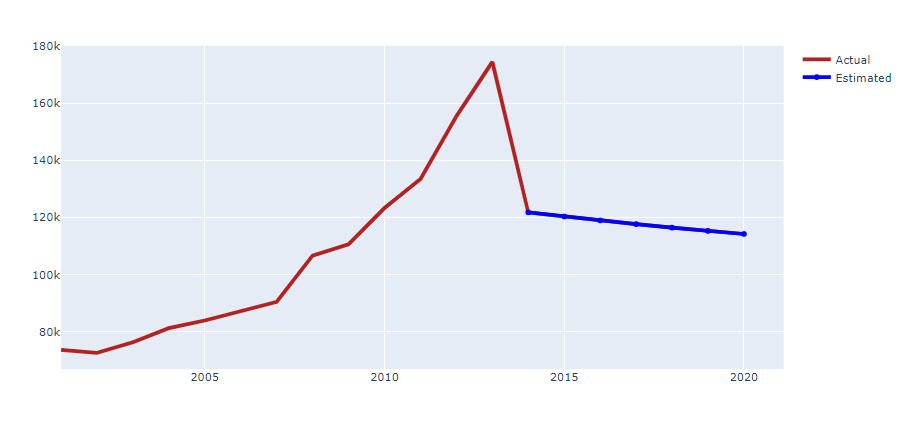
Soothing beauty, offbeat tourist attractions, spiritual ambiance, the sacred river, humble people, and zealous culture, Assam boasts plenty of fascinating secrets(Tourmyindia, n.d.). Capital of Assam is Dispur. In 2019 the recoded population of the state was 35,607,039(Statstics Times, 2020).

The figure below shows the trend of Arunachal Pradesh and from 2001 to 2013.



1. : Trend of Assam

From the graph, it can be observed that the trend of crimes will follow the observed trend in this state. The observed crime was more than 150000 cases and from 2002 the number of cases has been increased over the years, between 2001 and 2002 the least amount of cases were recorded.



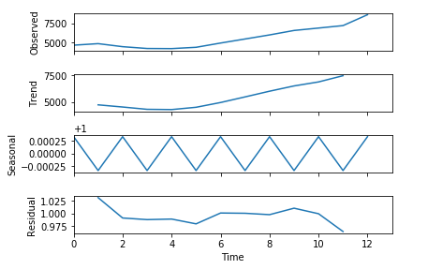
1. : Forecast In Assam

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 till 2020. The model predicted that the cases in 2014 will around 120000 and until it reaches 2020 cases will keep on decreasing. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 21.3%. In 2011 and 2012 the state recorded the highest number of cases.

### 4.4.5 GOA

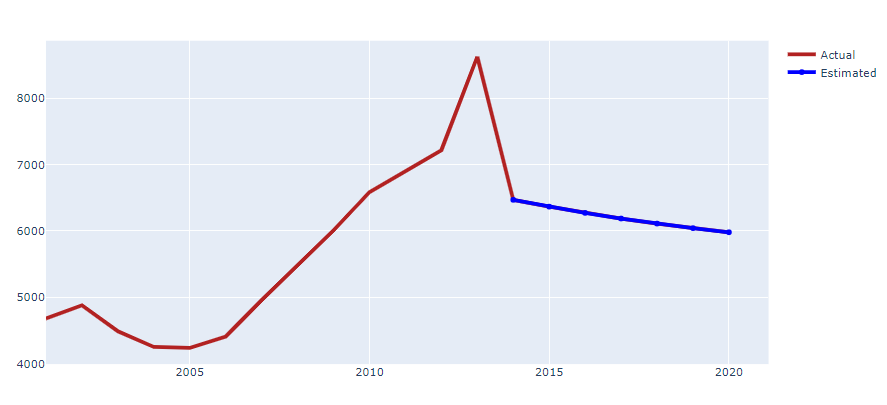
Having a tranquil atmosphere with the essence of western culture in the traditional land of India, this unparalleled state has become one of the most popular and preferred tourism destinations for travelers of all genres (Tourmyindia, n.d.). Capital of GOA is Panji, the population of goa is around 1,586,250(Statstics Times, 2020)

The figure below shows the trend of GOA from 2001 to 2013.



1. : Trend of Goa

From the graph, it can be observed that the trend of crimes will follow the observed trend in this state. The observed crime was more than 7500 cases and from 2005 the number of cases has been increased over the years, between 2003 and 2004 the least amount of cases were recorded.



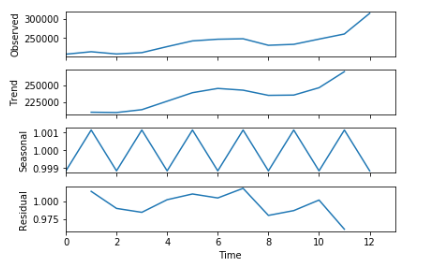
1. Forecast In Goa

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 onwards. The model predicted that the cases in 2014 will around 6500 and until it reaches 2020 cases will keep on decreasing. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 15.3%. In 2003 and 2004 the state recorded the least number of cases 4100.

### 4.4.6 GUJARAT

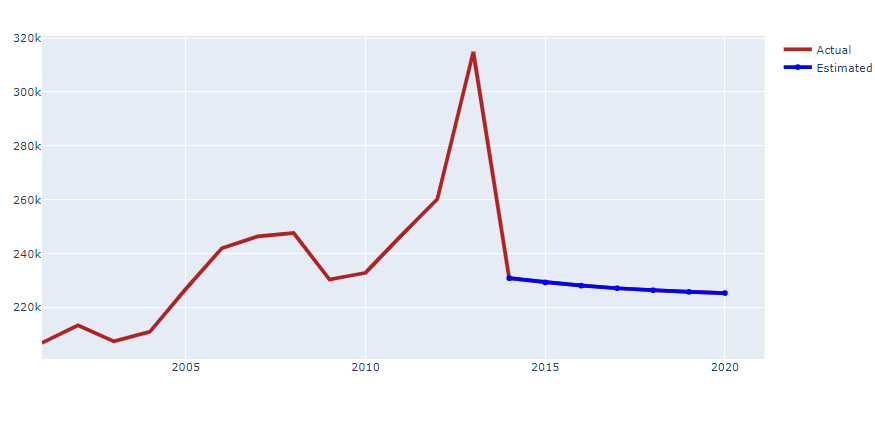
The capital of this state is Gandhinagar. This state is the birthplace of the great Mahatma Gandhi. World's largest white desert, India's longest coastline, shore temples, the habitat of Asiatic lions, countries first marine national park, ancient Buddhist caves, Harappan citadels more than 4,000 years old and handicrafts and handlooms trail(Tourmyindia, n.d.). The population of this city is 63,872,399(Statstics Times, 2020).

The figure below shows the trend of Gujarat from 2001 to 2013.



1. : Trend In Gujarat

From the graph, it can be observed that the trend of crimes will follow the observed trend in this state. The observed crime was more than 300000 cases and from 2004 the number of cases has been increased over the years, There was a dip in cases from the year 2008 to 2010



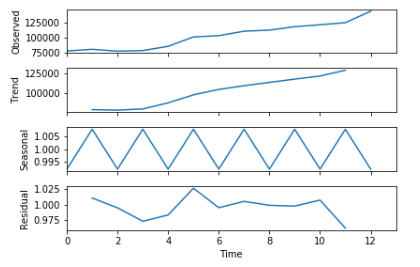
1. : Forecast in Gujarat

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 onwards. The model predicted that the cases in 2014 will be around 230000 and until it reaches 2020 cases will keep on decreasing gradually. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 15.3%. In 2011 and 2012 there was a sudden spike in several cases 315000.

### 4.4.7 HARYANA

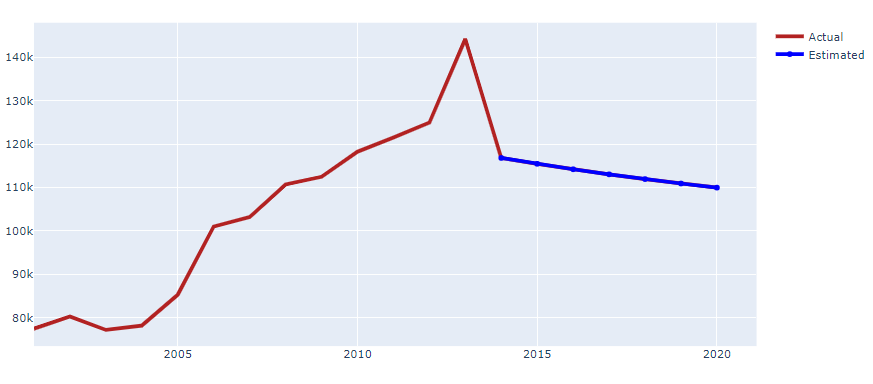
Haryana prides itself on being counted among the most economically developed states in India, and why not. Ever since Independence, this prosperous state has achieved excellence in various fields, from industry and agriculture to sports(Tourmyindia, n.d.). (Statstics Times, 2020)The population of Haryana is around 28,204,692

The figure below shows the trend of Haryana from 2001 to 2013.



1. : Trend of Haryana

Crimes in Haryana followed the trend as shown in the above figure. The observed crime was more than 125000 cases and from 2004 the number of cases has been increased over the years, The trend is showing that over the two years cases will remain constant



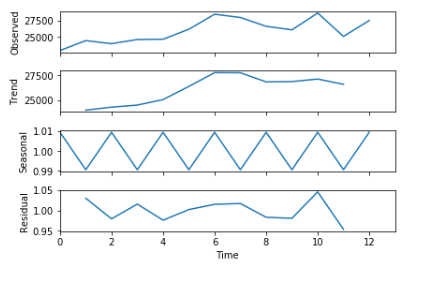
1. : Forecast In Haryana

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 onwards. The model predicted that the cases in 2014 will be around 115000 and until it reaches 2020 cases will keep on decreasing gradually. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 10.76%. In 2004 and 2003 the number of cases was recorded less compared to other years.

### 4.4.8 HIMACHAL PRADESH

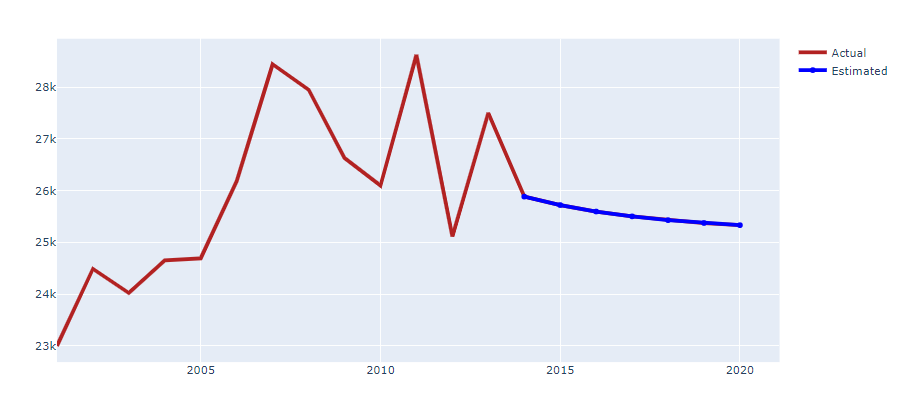
A land offering solace to the mind and soul, Himachal Pradesh is one of the much-admired and praised tourist destinations in India(Tourmyindia, n.d.). With the population around 7,451,955, Himachal Pradesh is one of the most beautiful cities in India(Statstics Times, 2020)

The figure below shows the trend of Himachal Pradesh from 2001 to 2013.



1. : Trend of Himachal Pradesh

Crime in Himachal Pradesh followed the trend as shown in the above figure The observed crime was more than 27500 cases and from 2004 the number of cases has been increased, but the trend shows that the cases will decrease.



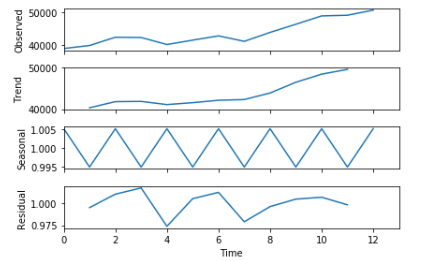
1. : Forecast In Himachal Pradesh

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 onwards. The model predicted that the cases in 2014 will be around 26000 and until it reaches 2020 cases will keep on decreasing gradually. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 10.76%. Cases were increasing and dressing from 2010 to 2013.

### 4.4.9 JAMMU AND KASHMIR

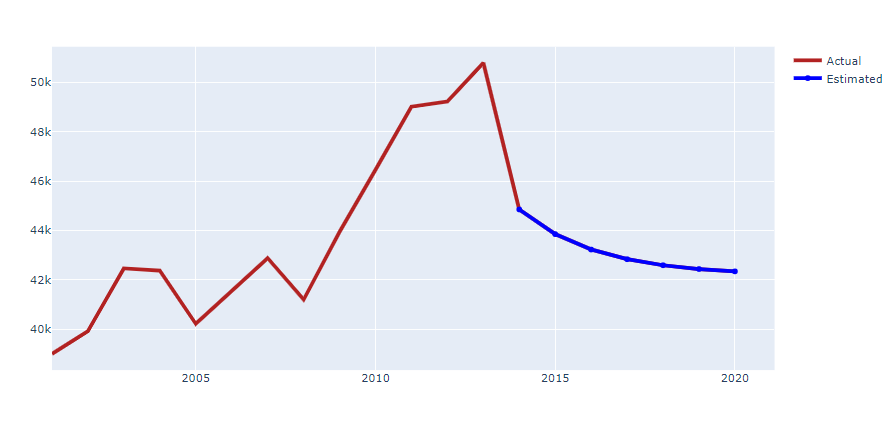
Known as Heaven on the Earth, Jammu & Kashmir is where Mother Nature has bestowed all her love. J&K has two capital Jammu is the winter capital and Kashmir is the summer capital(Tourmyindia, no date). The population of this state is 13,606,320(Statstics Times, 2020)

The figure below shows the trend of Jammu and Kashmir from 2001 to 2013.



1. : Trend of Jammu and Kashmir

Crime in Jammu&Kashmir followed the trend as shown in the above figure The observed crime was more than 45000 cases and from 2007 the number of cases has been increased over the years, The trend is showing that over the two years cases will increase.



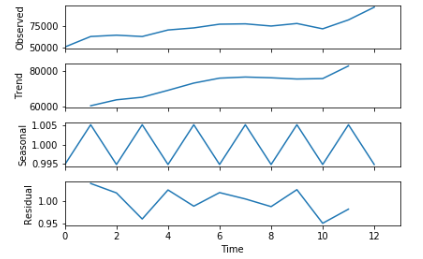
1. : Forecast In Jammu and Kashmir

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 onwards and it will keep on decreasing over the years. The model predicted that the cases in 2014 will be around 46000 and until it reaches 2020 cases will keep on decreasing gradually. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 11.41%. Cases were increasing and dressing from 2008 to 2013.

### 4.4.10 JHARKHAND

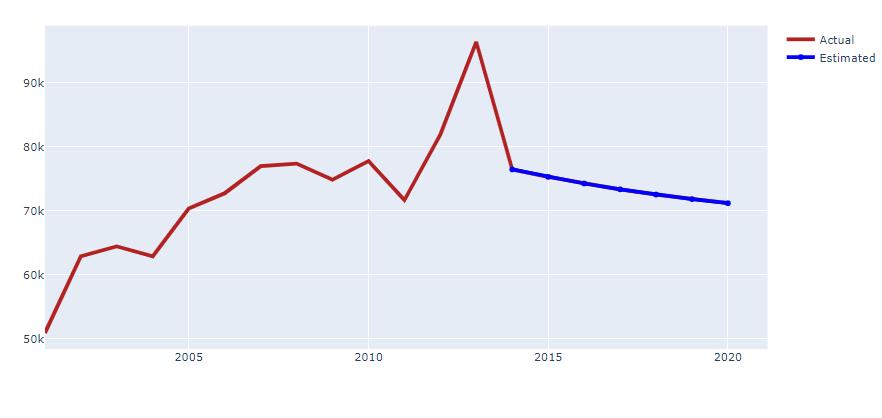
The eastern state of India, Jharkhand is a lesser explored destination that truly deserves a visit due to its rich natural beauty, culture, and tradition, heritage, wildlife, and more(Tourmyindia, n.d.). The population of the state is 38,593,948(Statstics Times, 2020)

The figure below shows the trend of Jharkhand from 2001 to 2013.



1. : Trend of Jharkhand

Crime in Jharkhand followed the trend as shown in the above figure. The observed crime was more than 75000 cases and from 2003 the number of cases has been increased over the years, There was a spike in the number of cases from 2010.



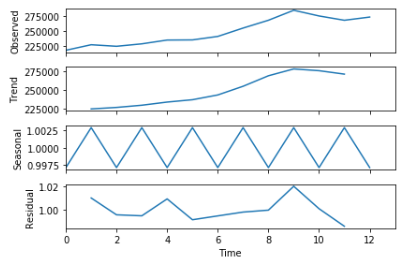
1. : Forecast In Jharkhand

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 onwards and it will keep on decreasing over the years. The model predicted that the cases in 2014 will be around 46000 and until it reaches 2020 cases will keep on decreasing gradually. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 11.41%. Cases were increasing and dressing from 2008 to 2013.

### 4.4.11 KARNATAKA

Karnataka with its glorious history, rich culture, and enriching tradition is located in the South-Western region of India(Tourmyindia, n.d.). The population of this state is around 67,562,686(Statstics Times, 2020)

The figure below shows the trend of Karnataka from 2001 to 2013.



1. : Trend of Karnataka

Crime in Karnataka followed the trend as shown in the above figure. The observed crime was more than 275000 cases and from 2003 the number of cases has been increased over the years, In 2009 cases were dropped the little but after 2010 again it started rising.



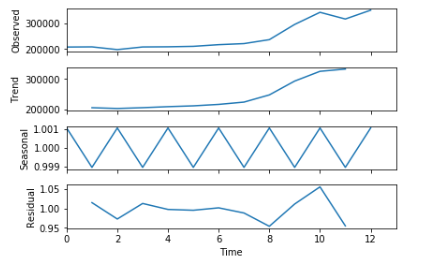
1. : Forecast In Karnataka

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 onwards and it will keep on decreasing over the years. The model predicted that the cases in 2014 will be around 280k and until it reaches 2020 cases will keep on decreasing to 260000 cases. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 3.41%. In 2010 the highest number of cases was reported around 290000.

### 4.4.12 KERALA

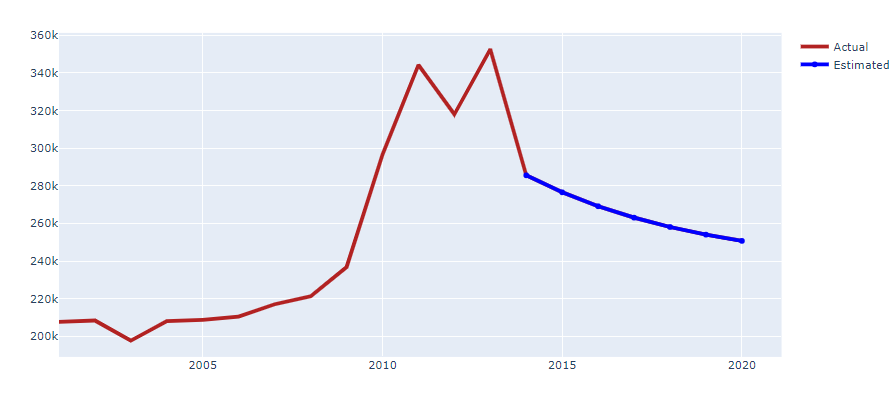
Encompassing serenity, Kerala is a charming South Indian destination. The population Kerala is around 35,699,443(Statstics Times, 2020)

The figure below shows the trend of Kerala from 2001 to 2013.



1. : Trend of Kerala

Crime in Kerela followed the trend as shown in the above figure. The observed crime was more than 300000, in 2010 the highest number of cases were recorded in the state. The trend shows that cases will remain constant over the years.



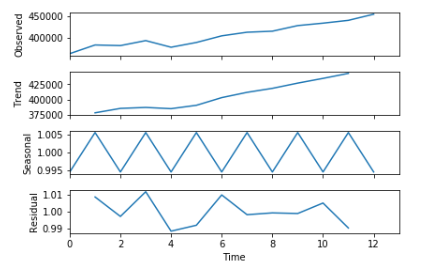
1. : Forecast In Kerela

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 onwards and it will keep on decreasing over the years. The model predicted that the cases in 2014 will be around 280k and until it reaches 2020 cases will keep on decreasing to 250000 cases. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 17.42%. In 2012 and 2013 the highest number of cases was reported 350000.

### 4.4.13 MADHYA PRADESH

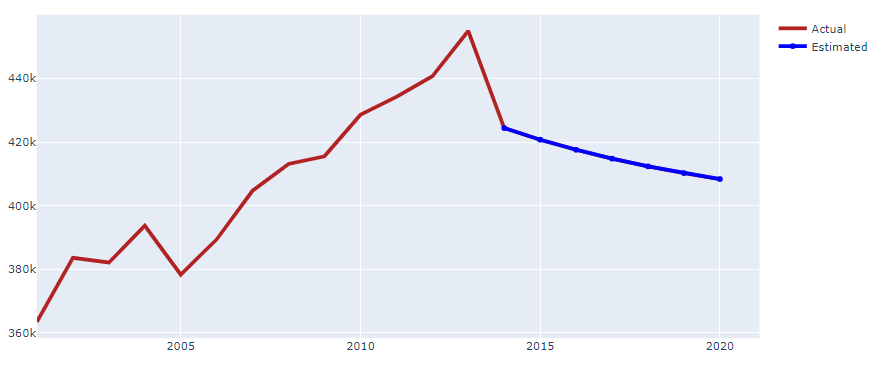
Welcome to the heart of India, Madhya Pradesh. The exotic land is an intoxicating mixture of rich history, vibrant sights, awe-spring art, and pious shrines(Tourmyindia, no date). The population of the state is around 85,358,965(Statstics Times, 2020).

The figure below shows the trend of Madhya Pradesh from 2001 to 2013.



1. : Trend of Madhya Pradesh

Crime in Madhya Pradesh followed the trend as shown in the above figure. The observed crime was more than 400000, the cases were increasing over the years



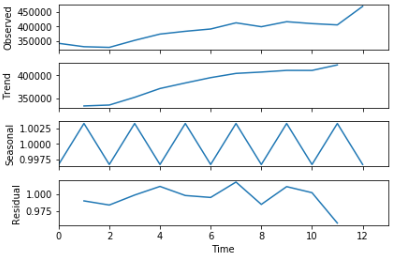
1. : Forecast In Madhya Pradesh

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 onwards and it will keep on decreasing over the years. The model predicted that the cases in 2014 will be around 430k and until it reaches 2020 cases will keep on decreasing to 400000 cases. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 5%. In 2013 the highest number of cases was reported 450000.

### 4.4.14 MAHARASTRA

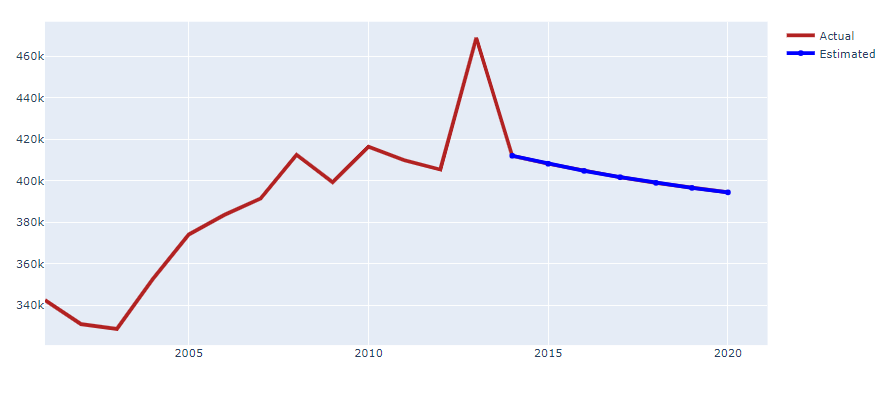
The third-largest state and the second most populous state in India, with a population of more than 123,144,223(Statstics Times, 2020)

The figure below shows the trend of Maharastra from 2001 to 2013.



* + - * 1. : Trend of Maharastra

Crime in Maharastra followed the trend as shown in the above figure. The observed crime was more than 400000, the cases were increasing over the years and in 2002 the lowest number of crimes were recorded.



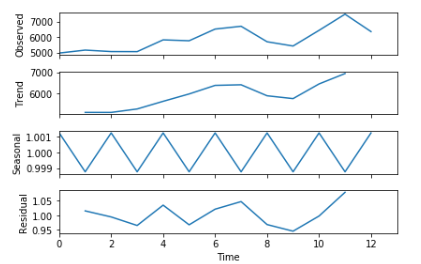
* + - * 1. : Forecast In Maharastra

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 onwards and it will keep on decreasing over the years. The model predicted that the cases in 2014 will be around 430k and until it reaches 2020 cases will keep on decreasing to 350000 cases. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 4.96%. In 2013 the highest number of cases was a reporter 470000.

### 4.4.15 MANIPUR

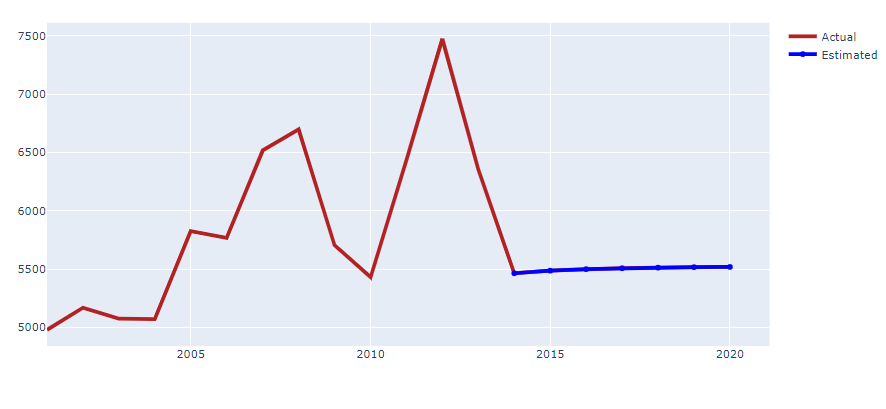
Nestled in the slopes of the south-flowing sub-Himalayan ranges in the northeast of India(Tourmyindia, n.d.). The population of Manipur in 2019 was around3,091,545(Statstics Times, 2020)

The figure below shows the trend of Manipur from 2001 to 2013.



* + - * 1. : Trend In Manipur

Crime in Manipur followed the trend as shown in the above figure. The observed crime was more than 7000 in 2011, the cases started decreasing from 2012. The year 2009 recorded the lowest number of crimes. The trend line of Manipur shows that crime will increase over the next two years.



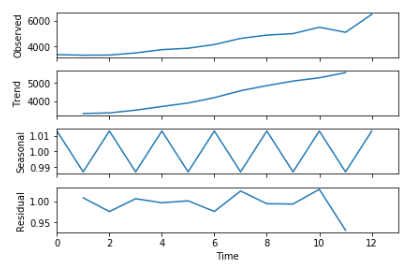
* + - * 1. : Forecast In Manipur

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease in the year 2014, but the cases will increase from 2015. The model predicted that the cases in 2014 will be around 4900 and until it reaches 2020 cases will keep on increasing to 5100 cases. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 18.38%. In 2012 the highest number of cases was reported 7500.

### 4.4.16 MEGHALAYA

Nestled in the northeastern Himalayas, Meghalaya takes the distinction of being the wettest place in the world (Tourmyindia, n.d.) The population of Meghalaya in 2019 was around3,366,710(Statstics Times, 2020).

The figure below shows the trend of Meghalaya from 2001 to 2013.



* + - * 1. : Trend of Meghalaya

Crime in Meghalaya followed the trend as shown in the above figure. The observed crime was more than 6000 in 2011, the cases started to increase from 2012. The year 2009 recorded the lowest number of crimes as compared to 2010 and 2008.



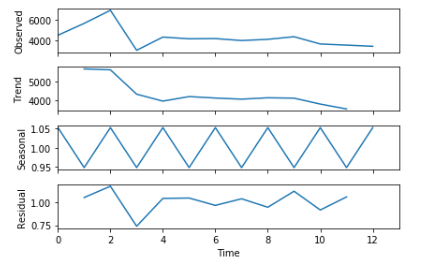
* + - * 1. : Forecast In Meghalaya

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 onwards and it will keep on decreasing over the years. The model predicted that the cases in 2014 will be around 5000 and until it reaches 2020 cases will keep on decreasing to 4600 cases. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 12.76%. In 2013 the highest number of cases was reported 6500.

### 4.4.17 MIZORAM

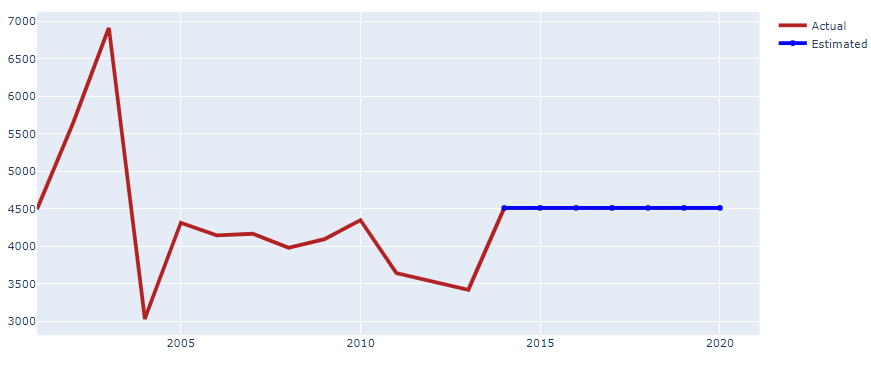
Tucked away in the thickets of green on the southern tip of Northeast India, Mizoram is a nature lover's paradise(Tourmyindia, n.d.). The population of Mizoram is around 1,239,244(Statstics Times, 2020).

The figure below shows the trend of Mizoram from 2001 to 2013.



1. : Trend of Mizoram

Crime in Mizoram followed the trend as shown in the above figure. The observed crime in this state was decreasing from 2009 and the trend line also shows the same. In 2002 the state reported the maximum number of cases that were 6000.



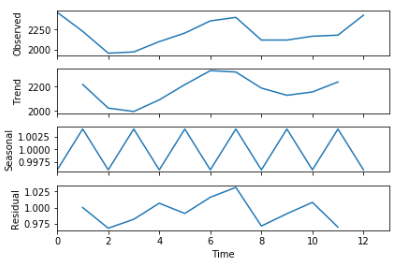
1. : Forecast In Mizoram

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will increase from 2014 onwards and will remain constant over the years. The model predicted that the cases in 2014 will be around 4500 and until it reaches 2020 the number of cases will remain constant. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 27.86%. In 2002 the highest number of cases was reported 7900.

### 4.4.18 NAGALAND

Nagaland offers unending exploration possibilities from heritage villages to wildlife sanctuaries(Tourmyindia, n.d.). The population of Nagaland in 2019 was around 2,249,695(Statstics Times, 2020).

The figure below shows the trend of Nagaland from 2001 to 2013.



* + - * 1. : Trend of Nagaland

Crime in Nagaland followed the trend as shown in the above figure. The observed crime in this state was increasing from 2012. The trend line of Nagaland shows cyclic nature. In 2002 the state recorded the least amount of cases.

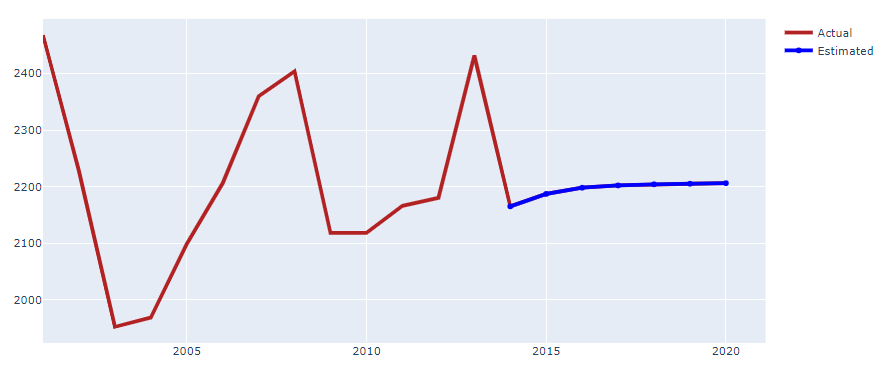


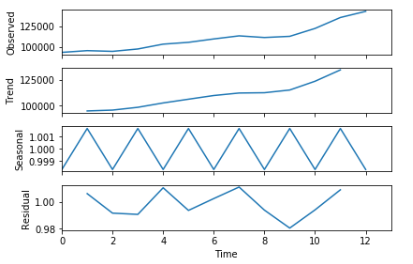
Figure 4.4.18.2: Forecast In Nagaland

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will increase from 2014 onwards and will remain constant over the years. The model predicted that the cases in 2014 will be around 2150 and until it reaches 2020 the number of cases will increase to more than 2200. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 3.33%.

### 4.4.19 ODISHA

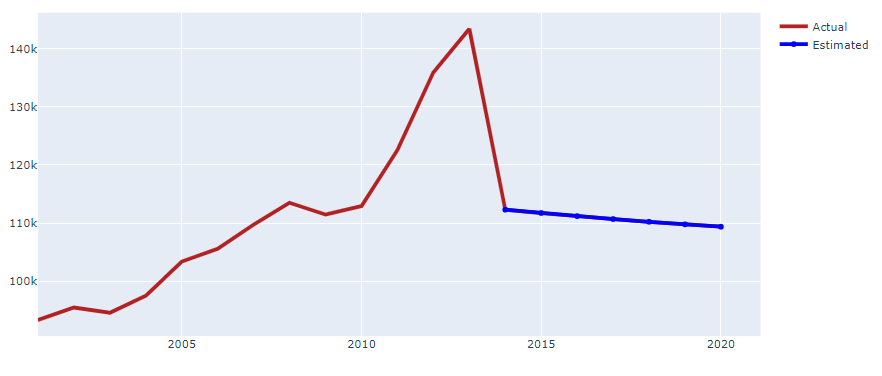
One of the most beautiful cities in India, situated on the east coast fringed by the Bay of Bengal(Tourmyindia, n.d.). The population of this state was around 46,356,334 in 2019(Statstics Times, 2020).

The figure below shows the trend of Odisha from 2001 to 2013.



* + - * 1. : Trend of Odisha

Crime in Odisha followed the trend as shown in the above figure. The observed crime in this state was increasing from 2012. The trend line also shows that crime will increase over the year.



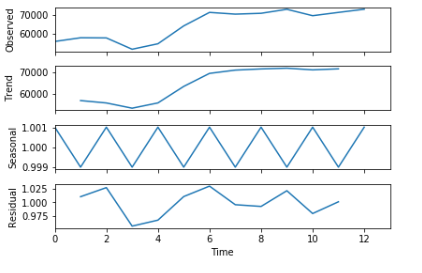
* + - * 1. : Forecast In Odisha

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 onwards and will remain constant over the years. The model predicted that the cases in 2014 will be around 110k and until it reaches 2020 the number of cases will decrease to less than 110000. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 16.23%. In 2012 the highest number of cases was reported 150000.

### 4.4.20 PUNJAB

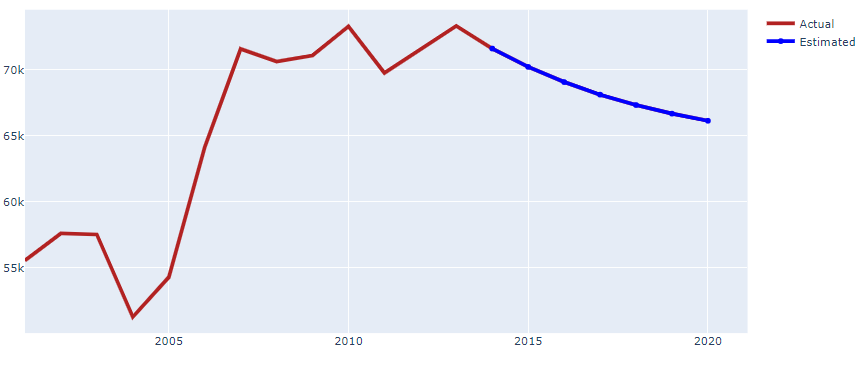
Bounded by the states of Jammu and Kashmir, Haryana, Himachal Pradesh, and Rajasthan, Punjab is gifted with a rich culture, heritage, and history(Tourmyindia, n.d.). The population of in 2019 was 30,141,373(Statstics Times, 2020)

The figure below shows the trend of Punjab from 2001 to 2013.



1. : Trend In Punjab

Crime in Punjab followed the trend as shown in the above figure. The observed crime in this state was increasing from 2012. The trend line of Punjab shows crimes will remain constant over the year. In 2003 the state recorded the least amount of cases.



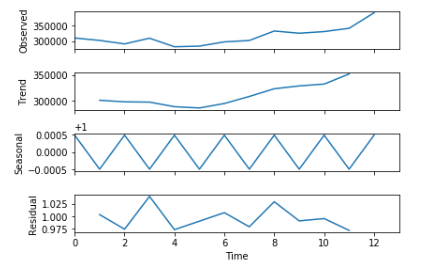
1. : Forecast In Punjab

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 onwards and over the years crime will decrease. The model predicted that the cases in 2014 will be around 73k and until it reaches 2020 the number of cases will decrease to less than 68000. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 3.45%. In 2004 the least number of cases was reported 50000.

### 4.4.21 RAJASTHAN

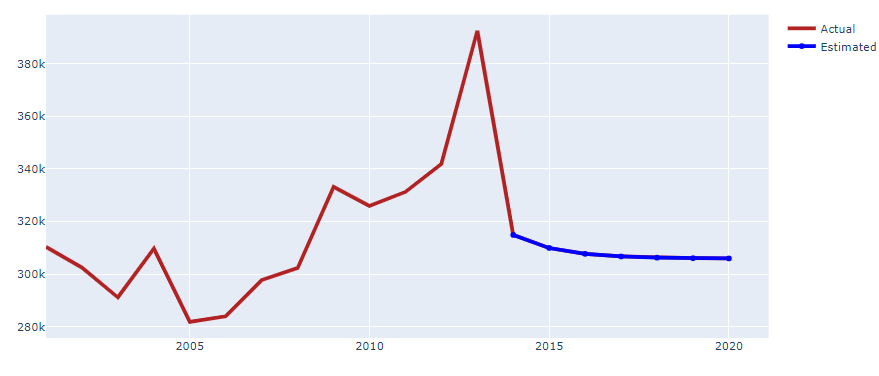
One of the most beautiful state in India sharing its border with Pakistan and having the Indias largest desert(Tourmyindia, n.d.), the recorded population of this state around 81,032,689(Statstics Times, 2020)

The figure below shows the trend of Rajasthan from 2001 to 2013.



: Trend of Rajasthan

Crime in Rajasthan followed the trend as shown in the above figure. The observed crime in this state was increasing from 2012. The trend line of Rajasthan shows crimes will increase over the years. In 2005 the state recorded the least amount of cases.



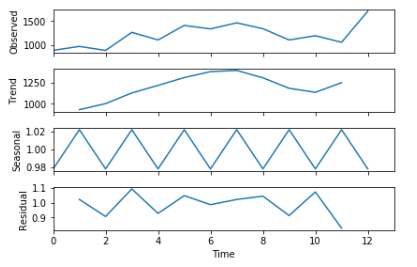
: Forecast in Rajasthan

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 onwards and over the years crime will decrease. The model predicted that the cases in 2014 will be around 325000 and until it reaches 2020 the number of cases will decrease to near 300k. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 11.46%. In 2005 the least number of cases was reported 390000.

### 4.4.22 SIKKIM

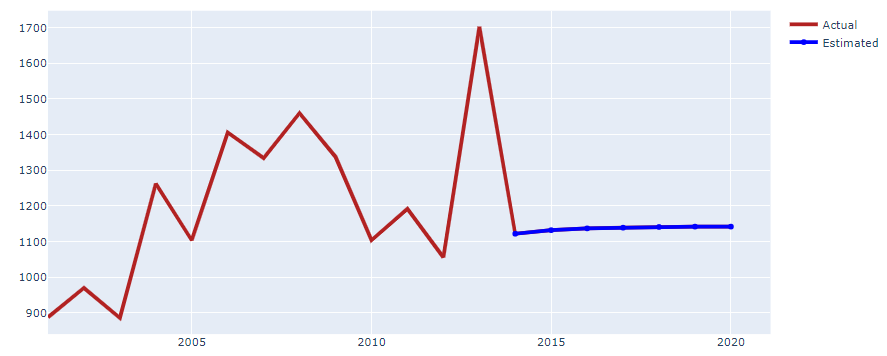
A fairytale land indeed, Sikkim is adorned with glistening high altitude lakes, rolling green mountains, pretty rhododendron groves, azure rivers, milky waterfalls, and vivid monasteries(Tourmyindia, n.d.). The population of this state was 690,251(Statstics Times, 2020).

The figure below shows the trend of Sikkim from 2001 to 2013.



1. : Trend of Sikkim

Crime in Sikkim followed the trend as shown in the above figure. The observed crime in this state was increasing from 2011. The trend line of Sikkim shows crimes will increase over the coming years. In 2007 the state recorded the highest amount of cases.



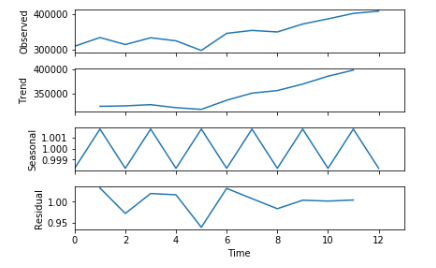
1. : Forecast In Sikkim

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 onwards and over the years crime will increase. The model predicted that the cases in 2014 will be around 1150 cases and until it reaches 2020 the number of cases will increase to near 1200. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 15.41%. In 2003 the least number of cases was reported 1700.

### 4.4.23 TAMIL NADU

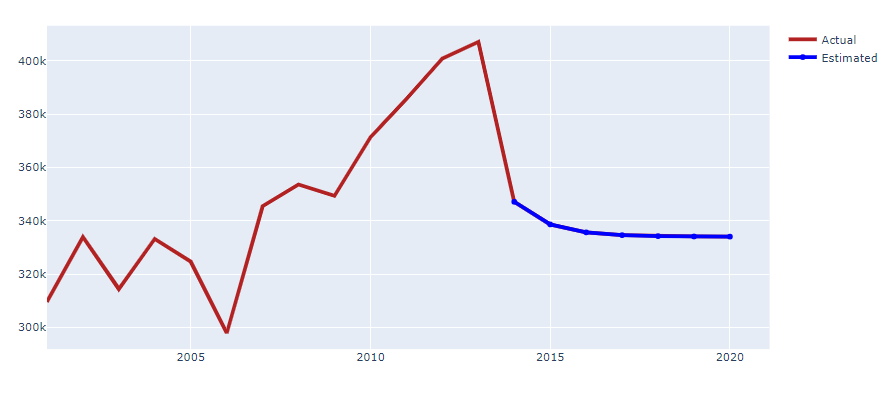
The southernmost part of India with a population of almost 77,841,267(Statstics Times, 2020).

The figure below shows the trend of Tamil Nadu from 2001 to 2013.



1. : Trend of Tamil Nadu

Crime in Tamil Nadu followed the trend as shown in the above figure. The observed crime in this state was increasing from 2009. The trend line of Tamil Nadu shows crimes will increase over the coming years. In 2005 the state recorded the least amount of cases.



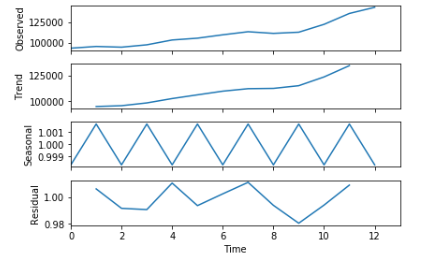
1. : Forecast In Tamil Nadu

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 onwards and over the years crime will remain constant over the years. The model predicted that the cases in 2014 will be around 350k cases and until it reaches 2020 the number of cases will less than 340000. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 14.37%. In 2006 the least number of cases was reported 290000.

### 4.4.24 TRIPURA

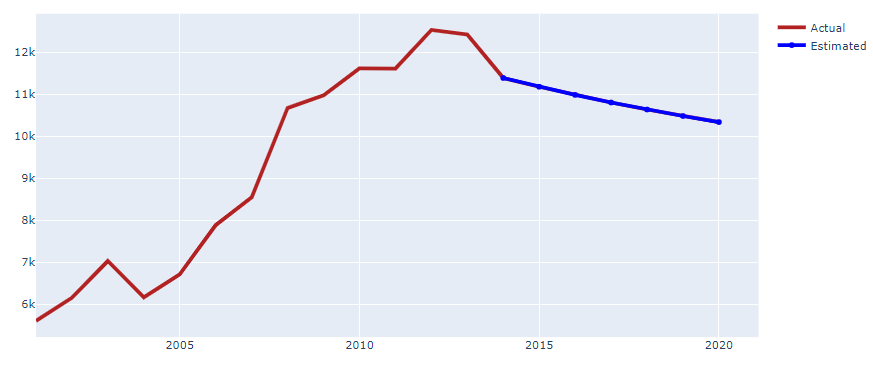
Tripura is a cultural reservoir among all the northeastern states of India(Tourmyindia, n.d.), with the population around 4,169,794(Statstics Times, 2020).

The figure below shows the trend of Tripura from 2001 to 2013.



1. : Trend of Tripura

Crime in Tripura followed the trend as shown in the above figure. The observed crime in this state was increasing from 2003. The trend line of Tripur shows crimes will increase over the coming years. In 2002 the state recorded the least amount of cases.



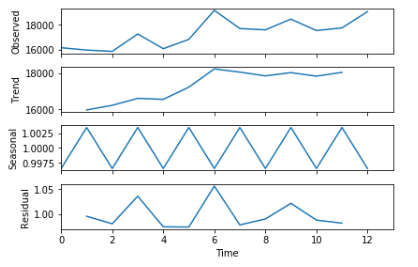
1. : Forecast In Tripura

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 onwards and over the years crime will keep on decreasing. The model predicted that the cases in 2014 will be around 11k cases and until it reaches 2020 the number of cases will less than 10000. To evaluate this forecasting technique, the author used the mean absolute percentage error(MAPE). The mean error for this state was 8.1%. In 2004 the least number of cases was reported 5000.

### 4.4.25 UTTARAKHAND

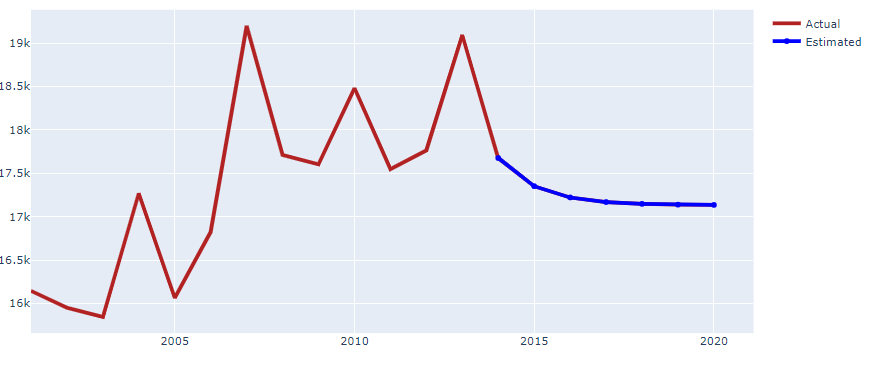
Uttarakhand is like the finest Chivalric Romance poetry when narrated, it melts the heart of the listener(Tourmyindia, n.d.). The population of Uttarakhand in 2019 was around 11,250,858(Statstics Times, 2020).

The figure below shows the trend of Uttarakhand from 2001 to 2013.



1. : Trend In Uttarakhand

Crime in Uttarakhand followed the trend as shown in the above figure. The observed crime in this state was increasing from 2011. The trend line of Uttarakhand shows crimes will remain constant or might increase. In 2006 the state recorded the highest amount of cases.



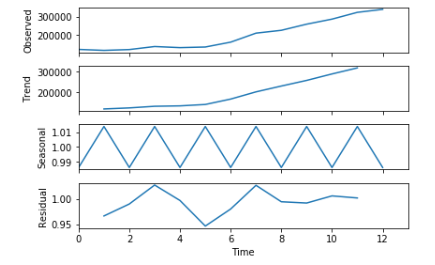
1. : Forecast In Uttarakhand

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 onwards and over the years crime will keep on decreasing. The model predicted that the cases in 2014 will be around 18k cases and until it reaches 2020 the number of cases will less than 180000. To evaluate this forecasting technique, the author used the mean absolute percentage error. The mean error for this state was 4.28%. In 2006 the highest number of cases was reported 20000.

### 4.4.26 WEST BENGAL

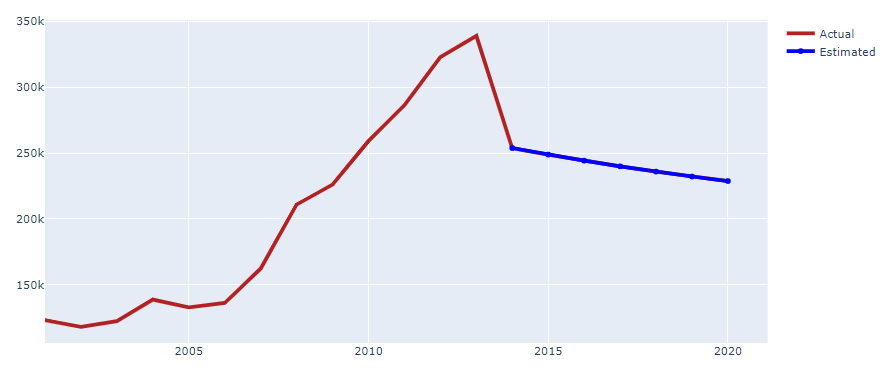
West Bengal offers a colorful variety of experiences to the tourist. Its capital, Kolkata(Tourmyindia, n.d.). The population of West Bengal was around 99,609,303(Statstics Times, 2020)

The figure below shows the trend of West Bengal from 2001 to 2013.



1. : Trend of West Bengal

Crime in West Bengal followed the trend as shown in the above figure. The observed crime in this state was increasing from 2003. The trend line of West Bengal shows crimes will increase over the coming years.



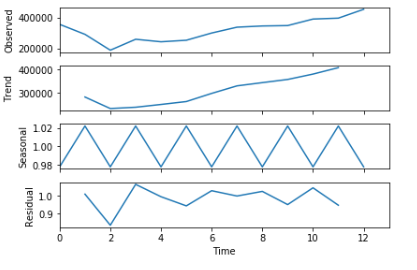
1. : Forecast In West Bengal

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 onwards and over the years crime will keep on decreasing. The model predicted that the cases in 2014 will be around 260k cases and until it reaches 2020 the number of cases will less than 240000. To evaluate this forecasting technique, the author used the mean absolute percentage error. The mean error for this state was 20.74%. In 2012 the highest number of cases was reported 340000.

### 4.4.27 UTTAR PRADESH

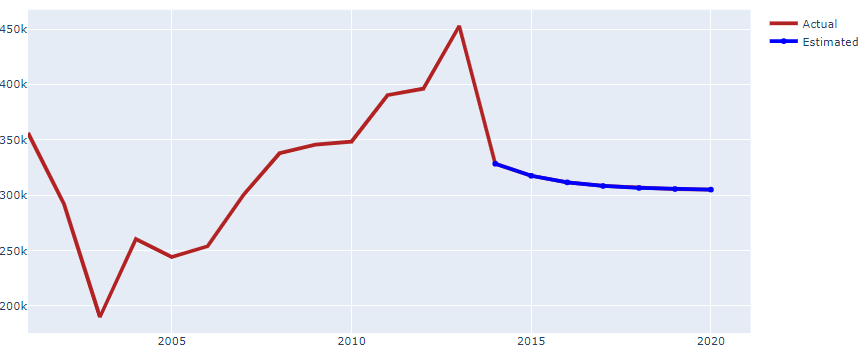
 This 4th largest state in India that spreads in an area of 243,290 sq km(Tourmyindia, n.d.) with the population around 237,882,725(Statstics Times, 2020)

The figure below shows the trend of Uttar Pradesh from 2001 to 2013.



1. : Trend of Uttar Pradesh

Crime in Uttar Pradesh followed the trend as shown in the above figure. The observed crime in this state was increasing from 2004. The trend line of Uttar Pradesh shows crimes will increase over the coming years. In 2003 the state recorded the least amount of cases.



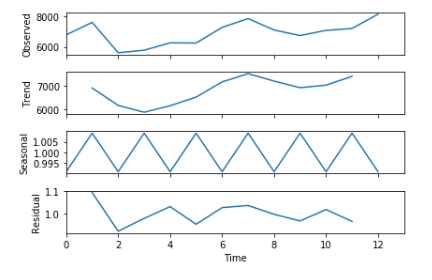
1. : Forecast In Uttar Pradesh

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 onwards and over the years crime will keep on decreasing. The model predicted that the cases in 2014 will be around 325k cases and until it reaches 2020 the number of cases will around 300000. To evaluate this forecasting technique, the author used the mean absolute percentage error. The mean error for this state was 22.29%. In 2003 the least number of cases was reported 450000.

### 4.4.28 CHANDIGARH

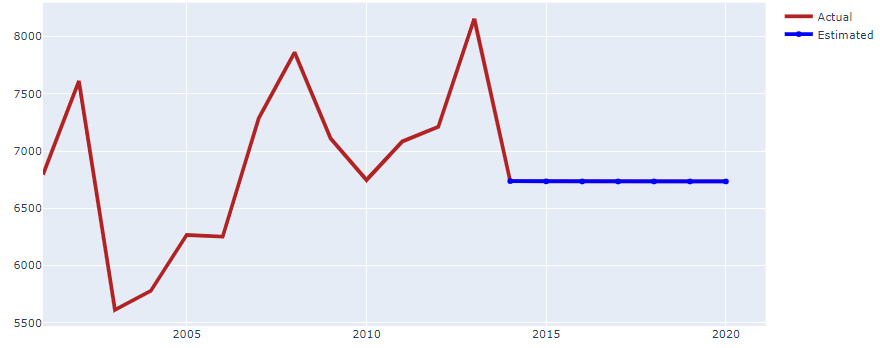
Acclaimed to be the first planned city of India, Chandigarh is renowned for its urban design and architecture(Tourmyindia, n.d.). The population of this state in 2019 was 1,158,473(Statstics Times, 2020)

The figure below shows the trend of Chandigarh from 2001 to 2013.



1. : Trend of Chandigarh

Crime in Chandigarh followed the trend as shown in the above figure. The observed crime in this state was increasing from 2003. The trend line of Chandigarh shows the cyclic nature of the crimes.



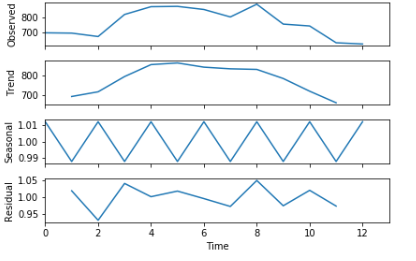
1. : Forecast In Chandigarh

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease from 2014 onwards and will remain constant over the year. The model predicted that the cases in 2014 will be around 6800 cases and until it reaches 2020 the number of cases will between 6700 to 6800. To evaluate this forecasting technique, the author used the mean absolute percentage error. The mean error for this state was 9.63%. In 2003 the least number of cases was reported 5300.

### 4.4.29 DADRA AND NAGAR

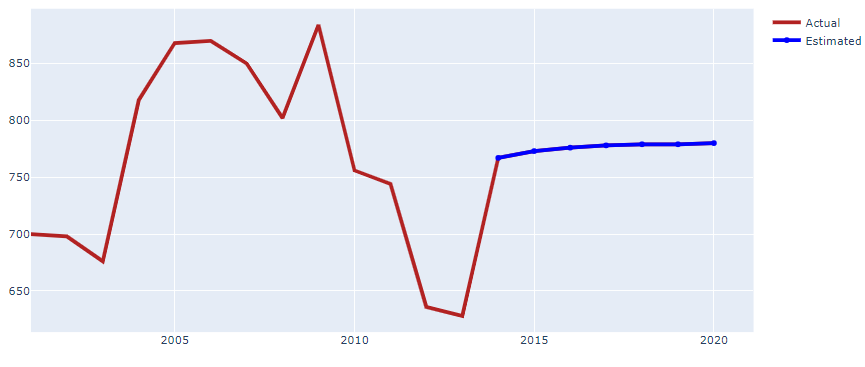
Wedged between the state of Gujarat and Maharashtra, Dadra and Nagar Haveli is a quaint Union Territory(Tourmyindia, n.d.). The population of the union territory was around 615,724(Statstics Times, 2020)

The figure below shows the trend of Dadra and Nagar from 2001 to 2013.



1. : Trend In Dadra

Crime in Dadra and Nagar followed the trend as shown in the above figure. The observed crime in this state was decreasing from 2008. The trend line of Dadra and Nagar shows that the crime will decrease.



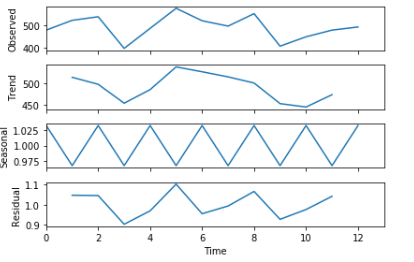
1. : Forecast In Dadra and Nagar

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will increase from 2014 onwards and over the years crime will keep on increasing till 2020. The model predicted that the cases in 2014 will be around 760 cases and until it reaches 2020 the number of cases will around 800. To evaluate this forecasting technique, the author used the mean absolute percentage error. The mean error for this state was 16.6%. In 2013 the least number of cases was reported 600.

### 4.4.30 DAMAN AND DIU

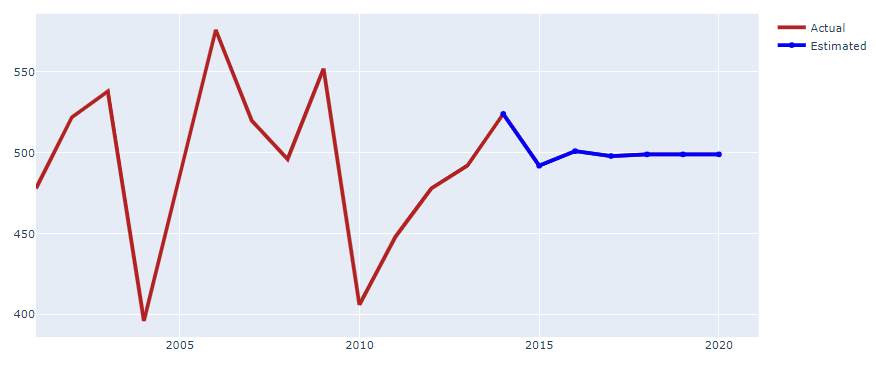
Daman and Diu is the union territory of India located in the western part of it(Tourmyindia, n.d.). The population of the union territory was around 615,724(Statstics Times, 2020)

The figure below shows the trend of Daman and Diu from 2001 to 2013.



1. : Trend of Daman

Crime in Daman and Diu followed the trend as shown in the above figure. The observed crime in this state was increasing from 2009. The trend line of Daman and Diu shows that the crime will increase.



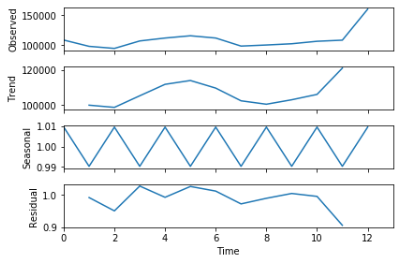
1. : Forecast In Daman

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will increase in 2014 but it will decrease 50 cases in 2015 and then again crimes will increase over the next few years. The model predicted that the cases in 2014 will be around 530 cases and until it reaches 2020 the number of cases will around 500. To evaluate this forecasting technique, the author used the mean absolute percentage error. The mean error for this state was 7.36%. In 2003 the least number of cases was reported 390.

### 4.4.31 DELHI

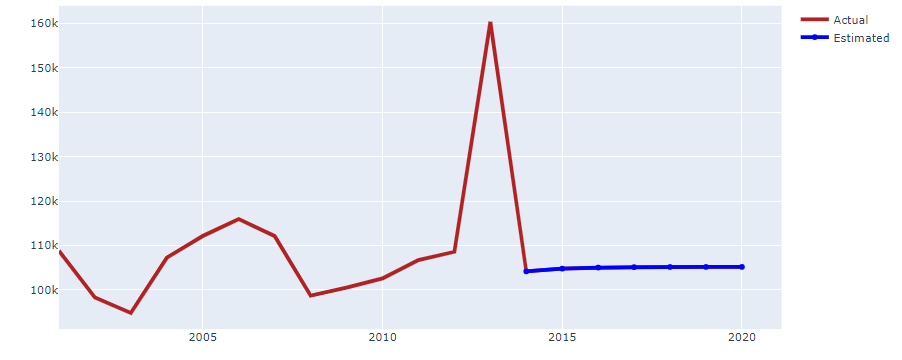
Delhi, for all the right reasons, is the national capital of incredible India(Tourmyindia, n.d.). The population of Delhi in 2019 was around 615,724(Statstics Times, 2020)

The figure below shows the trend of Delhi from 2001 to 2013.



1. : Trend In Delhi

Crime in Delhi followed the trend as shown in the above figure. The observed crime in this state was increasing from 2007. The trend line of Delhi shows that crime will increase over the next few years.



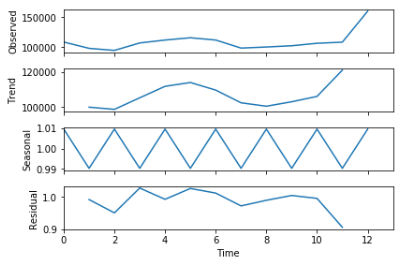
1. : Forecast In Delhi

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will decrease in 2014 but over the next few years, cases will increase slightly. The model predicted that the cases in 2014 will be around 105k cases and until it reaches 2020 the number of cases will around 108000. To evaluate this forecasting technique, the author used the mean absolute percentage error. The mean error for this state was 13.46%. In 2013 the highest number of cases was reported 160000.

### 4.4.32 LAKSHADWEEP

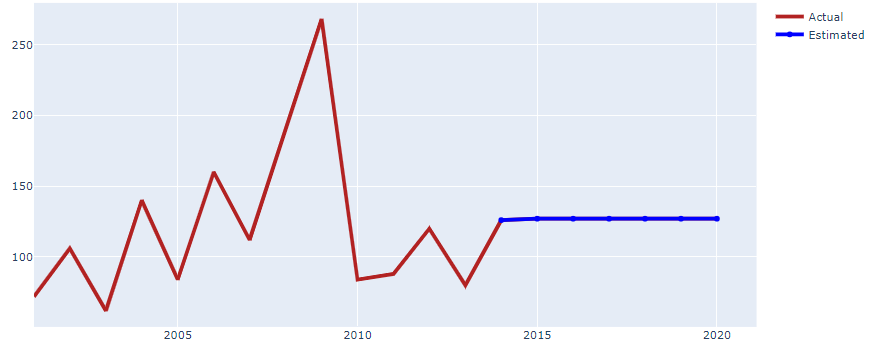
A perfect amalgamation of islands, Lakshadweep is situated off the Kerala coast in the Arabian Sea(Tourmyindia, no date). The recorded population of Lakshadweep in 2019 was around 73,183(Statstics Times, 2020).

The figure below shows the trend of Lakshadweep from 2001 to 2013.



1. : Trend In Lakshadweep

Crime in Lakshadweep followed the trend as shown in the above figure. The observed crime in this state was increasing from 2007. The trend line of Lakshadweep shows that crime will increase rapidly.



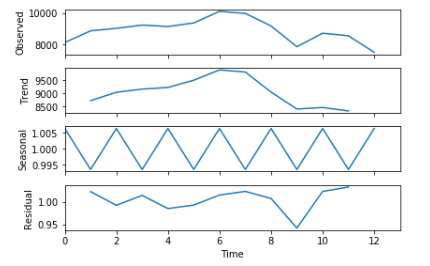
1. : Forecast In Lakshadweep

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will increase from 2014 but will remain constant. The model predicted that the cases in 2014 will be around 115 cases and until it reaches 2020 the number of cases will around 120. To evaluate this forecasting technique, the author used the mean absolute percentage error. The mean error for this state was 36.63%. In 2009 the highest number of cases was reported 280.

### 4.4.33 PUDUCHERRY

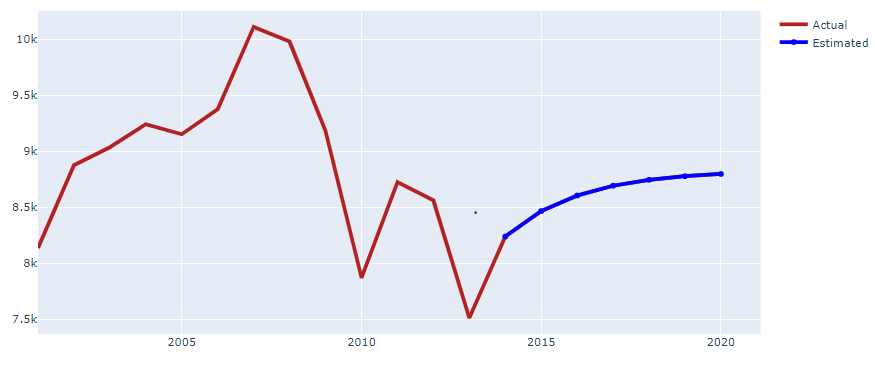
The silhouette of Pondicherry comprises the famed French Town or White Town, spectacular churches, temples, and statues with an additional charm of the sea(Tourmyindia, no date)t. The population of this union territory was around 1,413,542(Statstics Times, 2020).

The figure below shows the trend of Pondicherry from 2001 to 2013.



1. : Trend of Pondicherry

Crime in Pondicherry followed the trend as shown in the above figure. The observed crime in this state was decreasing from 2009. The trend line of Pondicherry shows that crime will decrease over the years.



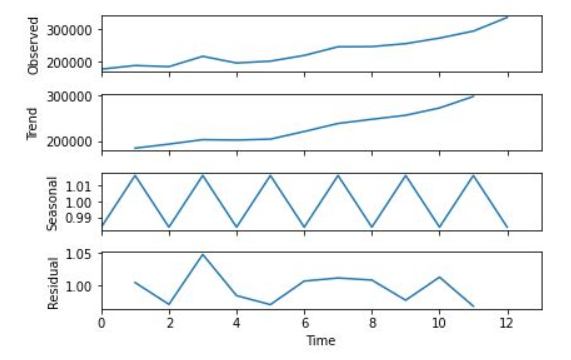
1. : Forecast In Puducherry

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will increase from 2014 and it will keep on increasing till 2020. The model predicted that the cases in 2014 will be around 8200 cases and until it reaches 2020 the number of cases will around 9000. To evaluate this forecasting technique, the author used the mean absolute percentage error. The mean error for this state was 7.08%. In 2013 the least number of cases was reported 7500.

### 4.4.34 BIHAR

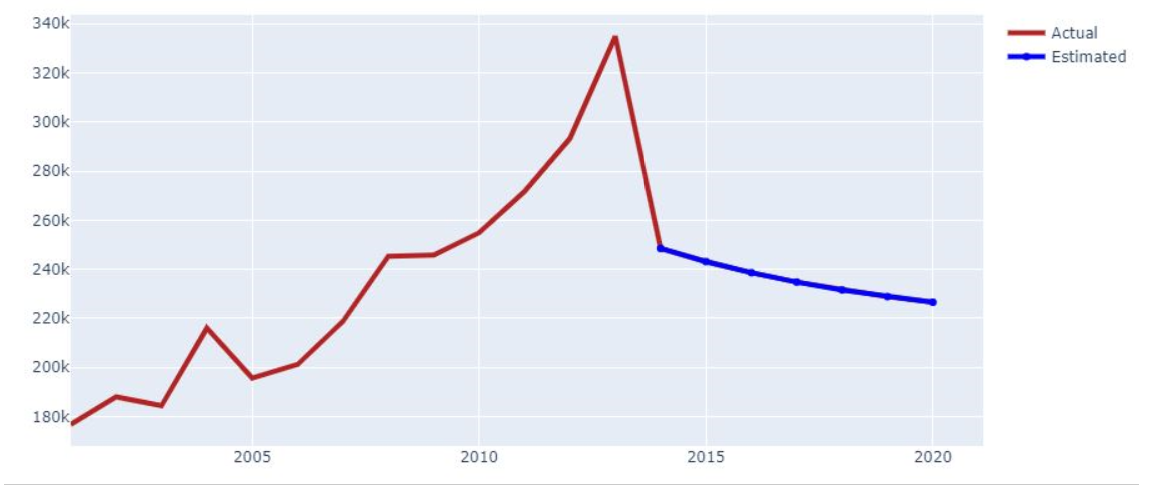
Nestled in the eastern part of India, Bihar is one such underrated destination in the country which has remained lesser-explored(Tourmyindia, n.d.). The population of this state was around 124,799,926(Statstics Times, 2020)

The figure below shows the trend of Bihar from 2001 to 2013.



1. : Trend of Bihar

Crime in Bihar followed the trend as shown in the above figure. The observed crime in this state was increasing from 2002. The trend line of Bihar shows that crime will increase over the years.



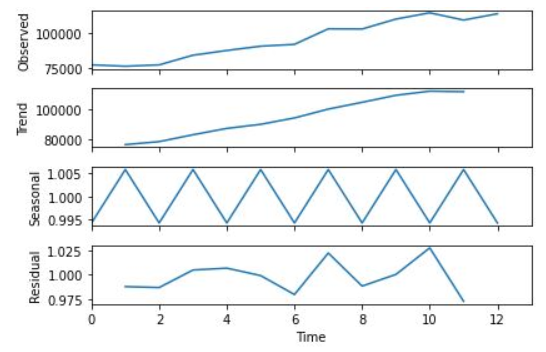
1. : Forecast In Bihar

The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will increase from 2014 and it will keep on increasing till 2020. The model predicted that the cases in 2014 will be around 250000 cases and until it reaches 2020 the number of cases will around 220. To evaluate this forecasting technique, the author used the mean absolute percentage error. The mean error for this state was 18.6%. In 2013 the highest number of cases was reported 330000.

### 4.4.35 CHHATTISGARH

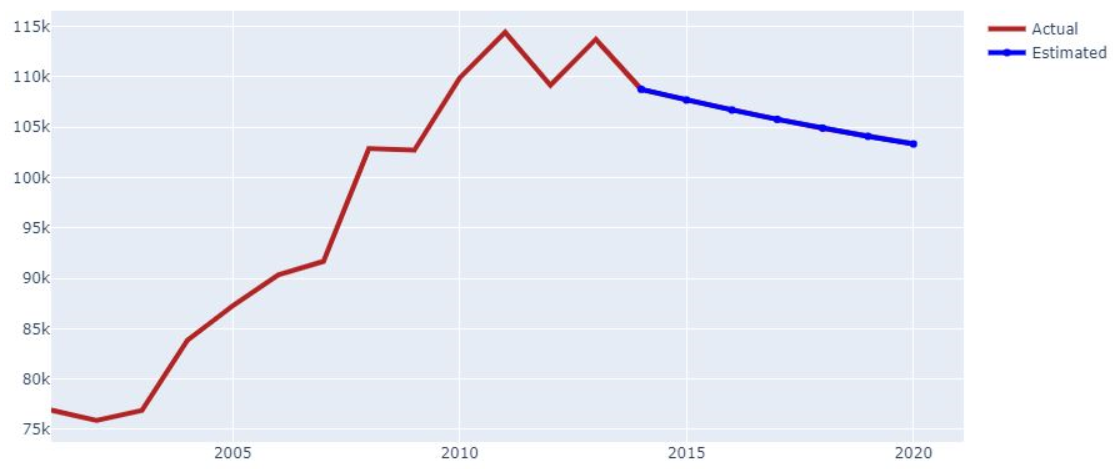
Chhattisgarh, one of the few landlocked states in central India, is a treasure house of natural beauty(Tourmyindia, n.d.). The population of Chhattisgarh was around 29,436,231(Statstics Times, 2020).

The figure below shows the trend of Chhattisgarh from 2001 to 2013.



1. : Trend In Chhattisgarh

Crime in Chhattisgarh followed the trend as shown in the above figure. The observed crime in this state was increasing from 2002. The trend line of Chhattisgarh shows that crime will remain constant over the years.



1. : Forecast In Chhattisgarh

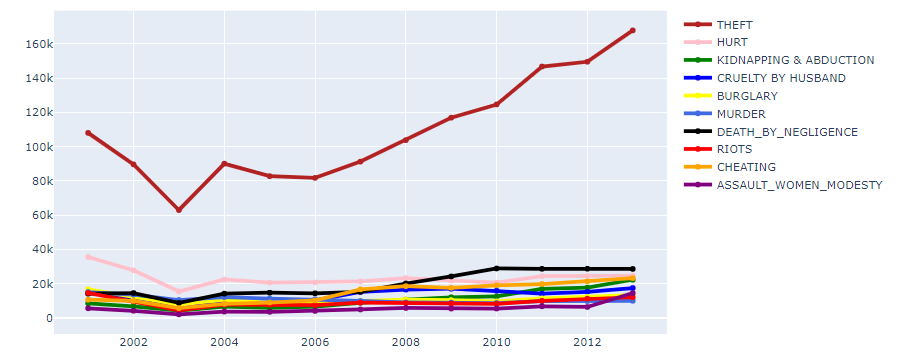
The above figure shows the forecast for the crime rate from 2014 to 2020. It can be observed from the figure that the crime in this state will increase from 2014 and it will keep on increasing till 2020. The model predicted that the cases in 2014 will be around 110000 cases and until it reaches 2020 the number of cases will drop down to 100000. To evaluate this forecasting technique, the author used the mean absolute percentage error. The mean error for this state was 4.6%. In 2013 and 2011 the highest number of cases was reported.

## 4.5 CLUSTERING

In this section, the author will try to identify the worst cities of the top 5 states in India and cluster them accordingly, and also the author will try to identify the major crimes taking place in the city.

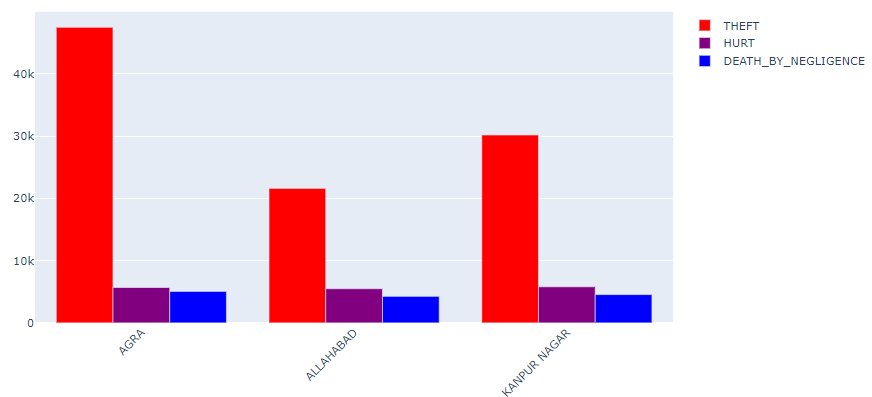
### 4.5.1 UTTAR PRADESH

From the given figure below, the author tried to find the top crimes that were happening in the state Uttar Pradesh.



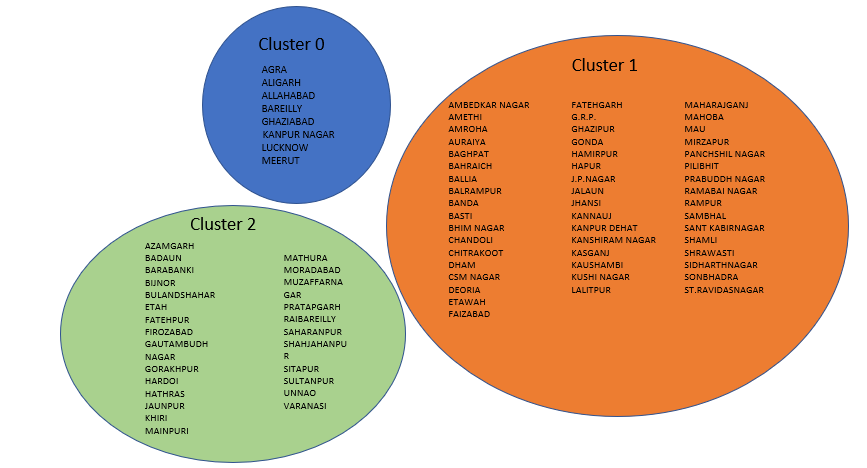
1. : Crimes happening in Uttar Pradesh

From the graph, it can be observed that top crimes in this state were Theft, Hurt, and Death by Negligence. The theft was the most committed crime in this state with over 60000 cases reported every year followed by death by negligence and hurt with more than 15000 cases.



1. : Cities Affected in UP

Top three worst-affected cities in Uttar Pradesh were Agra, Allahabad, Kanpur based on the count of crime for theft and hurt.

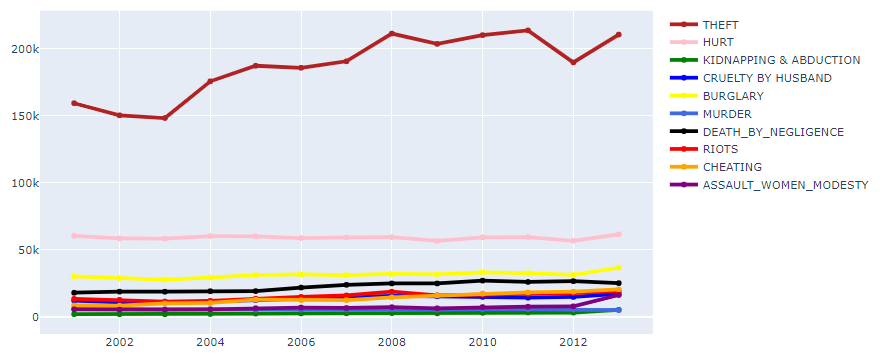


1. : Clustered cities in Uttar Pradesh

This figure shows the cities that were divided into different clusters according to the crime rates that were happening in Uttar Pradesh. Cities in Cluster 0 were the worst affected with the count of other IPC crimes ranging more than 15000 and total\_theft committed in these cites were more than 30000. Cities in Cluster 1 are less affected than cities in cluster 0. The count of crimes in these districts was less than cities in cluster 0. The least affected cities were sub-grouped in cluster 2.

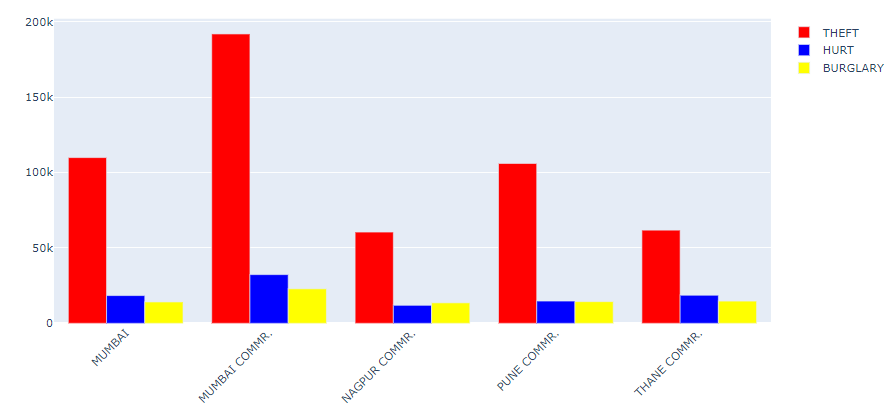
### 4.5.2 MAHARASTRA

From the given figure below, the author tried to find the top crimes that were happening in the state Maharastra.



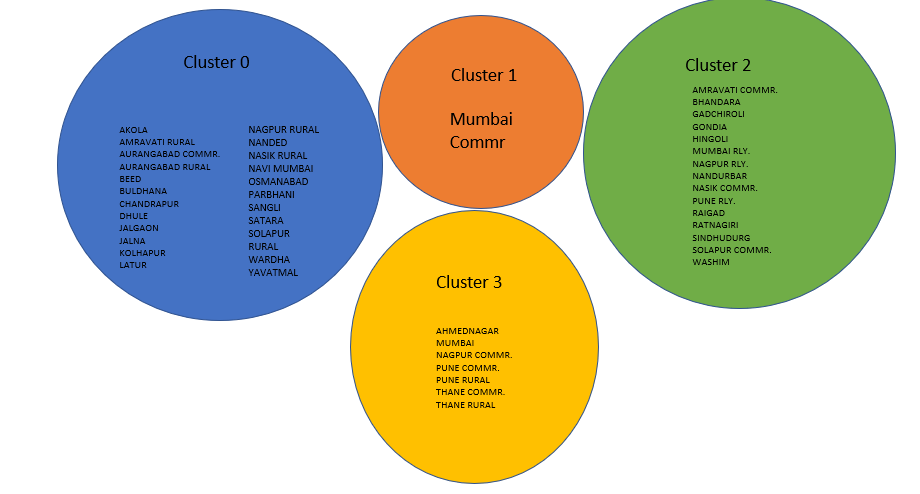
1. : Crimes happening in the Maharastra

From the graph, it can be observed that top crimes in this state were Theft, Hurt, and Burglary. The theft was the most committed crime in this state with over 150000 cases reported every year followed by burglary and hurt with more than 25000 cases and 50000 cases respectively.



1. : Cities affected In Maharastra

Worst-affected cities in Maharastra were Mumbai, Mumbai Commr, Nagpur, Pune, and Thane based on the count of crime for theft, hurt, and Burglary. The worst affected city in Maharashtra is Mumbai Commr followed by Pune.

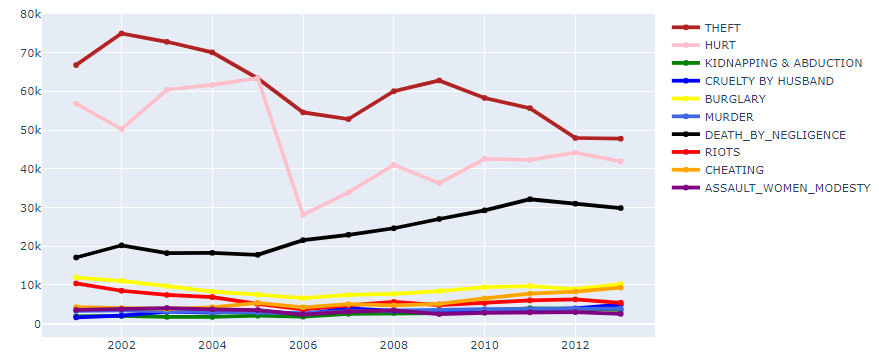


1. : Clustered cities in Maharastra

The districts in Maharastra were clustered into four groups. The worst affected city in Maharastra was Mumbai Commr with almost every crime was acceding with more than 20000 cases. The least affected cities were grouped in cluster 0. Cluster 3 was highly affected but had the fewer amount of cases than Mumbai Commr. Cities in cluster 2 didn’t have the high crime rates.

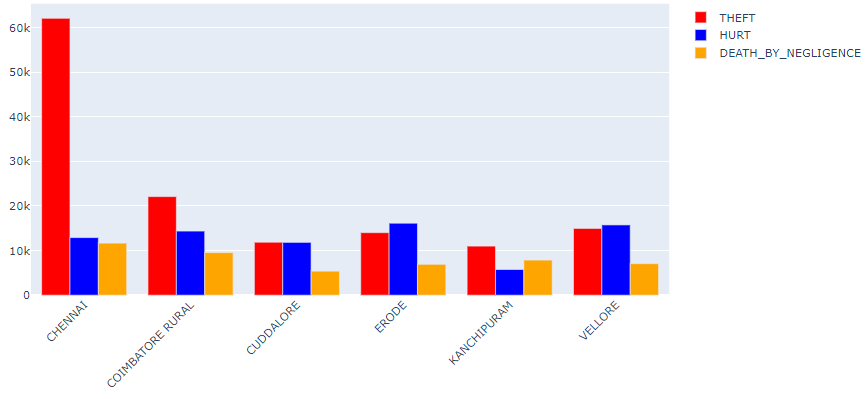
### 4.5.3 TAMIL NADU

From the given figure below, the author tried to find the top crimes that were happening in the state Maharastra.



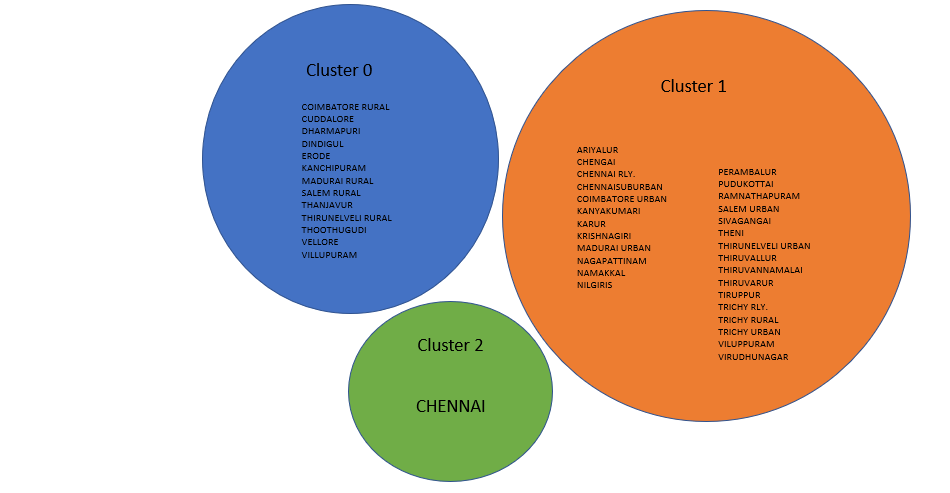
1. : Crimes in Tamil Nadu

From the graph, it can be observed that top crimes in this state were Theft, Hurt, and Death by Negligence. The theft was the most committed crime in this state with over 45000 cases reported every year followed by death by negligence and hurt with more than 15000 cases and 25000 cases respectively. The line chart shows that crimes have been decreased in this state.



1. : Cities Affected in Tamil Nadu

Worst-affected cities in Tamil Nadu were Chennai, Coimbatore rural area, Cuddalore, Erode, Kanchipuram, and Vellore based on the count of crime for theft, hurt, and Death by Negligence. From the above figure, it can be seen that Chennai and Coimbatore have reported the most number of cases.

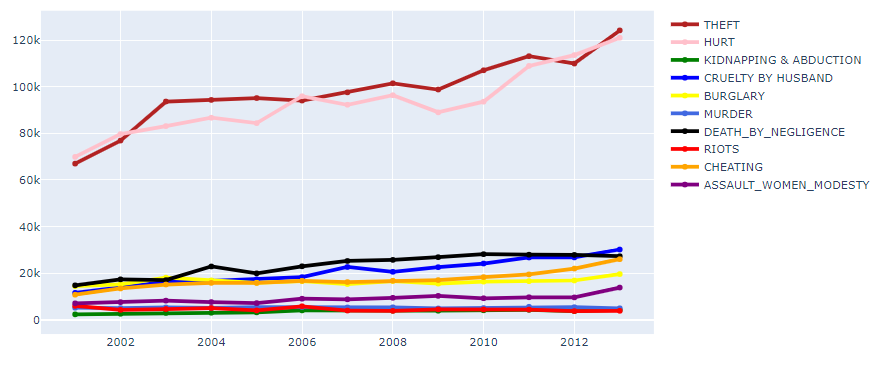


1. : Clustered cities in Tamil Nadu

The worst affected city in Tamil Nadu was Chennai with the count of total theft was more than 60000. Cities grouped in cluster 0 were more affected than the cities in Cluster\_1 but had less number of cases than Chennai. Least affected cities were grouped in cluster 1.

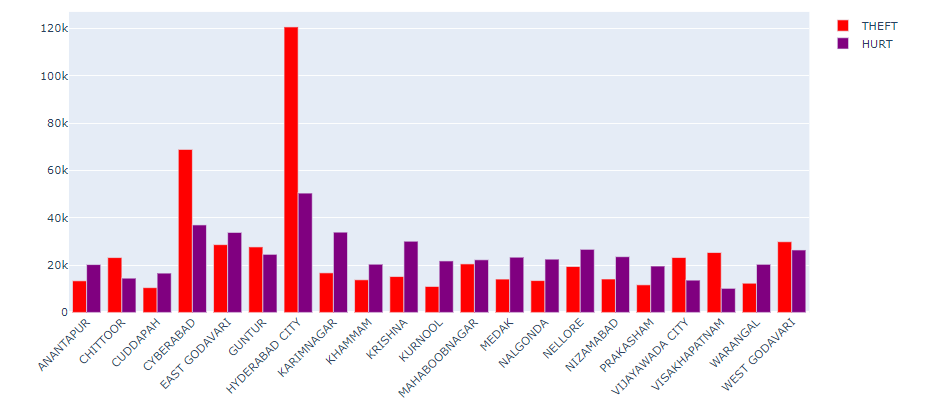
### 4.5.3 ANDHRA PRADESH

From the given figure below, the author tried to find the top crimes that were happening in the state Andhra Prades.



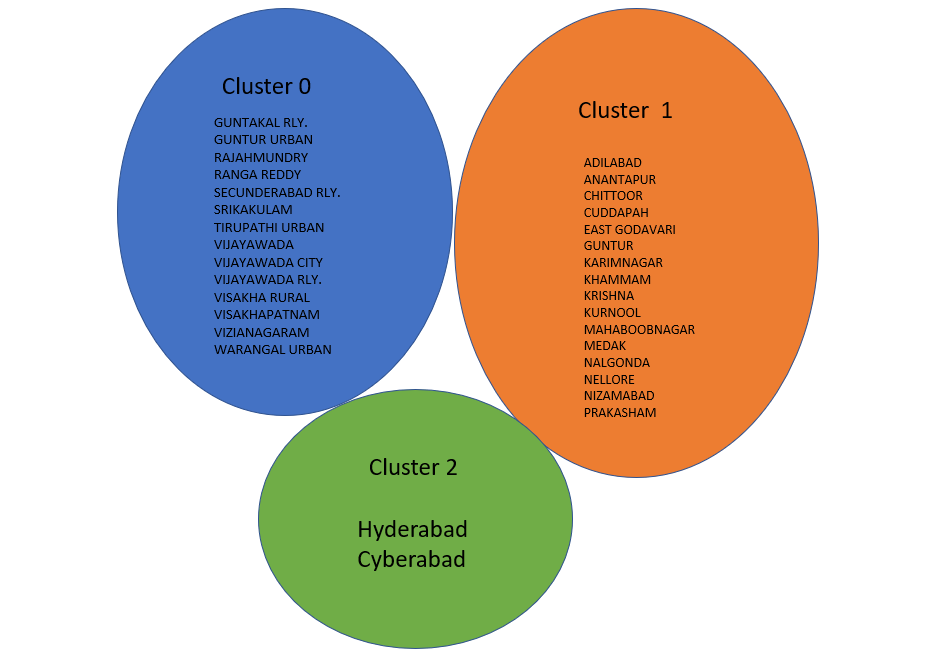
1. : Crimes In Andra Pradesh

From the graph, it can be observed that top crimes in this state were Theft, Hurt. The theft was the most committed crime in this state with over 65000 cases reported every year followed by hurt with more than 70000 cases.



1. : Cities Affected In Andhra Pradesh

Worst-affected cities in Andra Pradesh were Hyderabad, and Cyberabad based on the count of crime for theft, hurt. Other cities also have an average count above 5000 cases. From the above figure, it can be seen that Hyderabad and Cyberabad had reported the most number of cases among all the other cities in Andra.

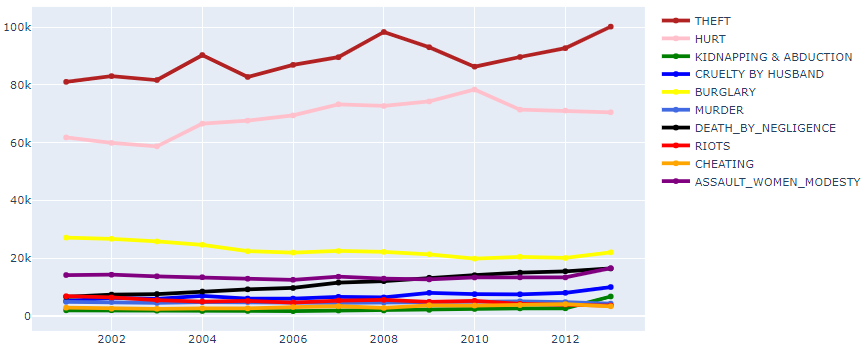


1. : Clustered Cities In Andhra Pradesh

The worst affected city in Andara Pradesh was Hyderbad and Cyberdabad with the count of total theft were more than 70000 in these cites. Cities grouped in cluster 1 were more affected than the cities in Cluster\_0 but had less number of cases than Hyderabad and Cyberabad. The least affected cities were grouped in cluster 0.

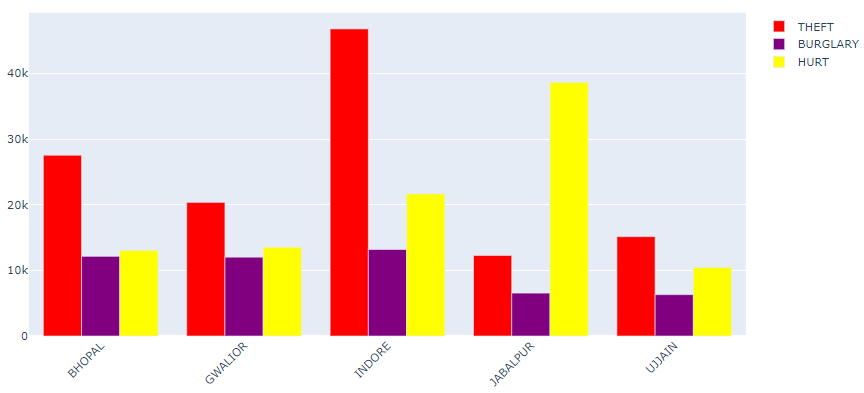
### 4.5.5 MADHYA PRADESH

From the given figure below, the author tried to find the top crimes that were happening in the state Madhya Pradesh.



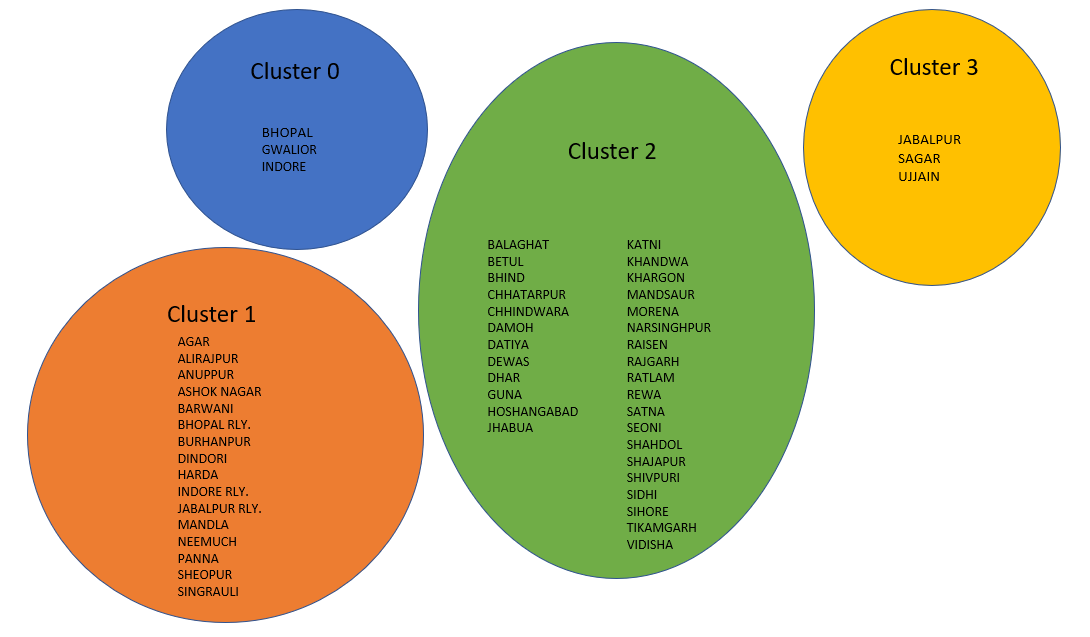
1. : Crimes In Madhya Pradesh

From the graph, it can be observed that top crimes in this state were Theft, Hurt. The theft was the most committed crime in this state with over 75000 cases reported every year followed by hurt with more than 55000 cases. The next major crime after hurt and theft was burglary.



1. : Cities Affected In Madhya Pradesh

Worst-affected cities in Madhya Pradesh were Bhopal, Gwalior, Indore, Jabalpur, Ujjain on the count of crime for theft, hurt, burglary. From the above figure, it can be seen that Indore and Jabalpur, and Bhopal had reported the most number of cases among all the other cities in Madhya. Most of the theft cases were reported in Indore and Bhopal.



1. : Clustered Cities In Madhya Pradesh

The districts in Madhya Pradesh were clustered into four groups. The worst affected city in Madhya Pradesh was Bhopal, Gwalior, and Indore with a count of total theft exceeding more than 40000 cases. The least affected cities were grouped in cluster 2. Cluster 3 was highly affected but had fewer amount of cases than Cluster 0. Cities in cluster 1 didn’t have the high crime rates.

# Chapter 5 – DISCUSSION AND CONCLUSIONS

## 5.1 INTRODUCTION

This chapter presents a discussion on the outcomes of the thesis and concludes the research.  The targets of this thesis are returned to and the commitment of embraced techniques in satisfaction of the destinations is evaluated.  The limitations of this thesis and the scope of future work are also discussed briefly in this chapter.

## 5.2 DISCUSSION

This research aimed to identify the following questions:- Can historical crime data for India be used to forecast the trend of crimes in each state of India for the next six years? The work carried out by the author did help to forecast the trend of crimes in each state of India with the error rate. Data used for the thesis was available from 2001 to 2013.

The first objective was to provide aggregate statistics from the dataset such as the highest crime areas. The author achieved this objective by plotting the pie chart. Pie chart helped the author to identify the top 10 crimes in India and plot every crime on the map of India. Frequency of crime distribution was obtained from this chart

The second objective was to represent key data and findings using a suitable visualization method and tool. In this objective, the author identified the worst top 5 states in India according to the total IPC crimes committed. It was observed that Uttar Pradesh, Maharastra, Andhra Pradesh, Tamil Nadu, and Madhya Pradesh were highly affected states. Top 10 crimes in India were murder, kidnapping & abduction, burglary, riots, cheating, hurt/grievous hurt, assault on women with intent to outrage her modesty, causing death by negligence, theft, cruelty by husband or his relatives. For the top 5 affected states, the author also computed the worst affected cities by considering a threshold value for the top crimes. Threshold values helped the author to find the count and filter the cities accordingly

The third objective was to cluster applicable data. This helped the author to divide the district of top 5 states into similar groups. These groups helped in identifying the cities from the worst affected to least affected. Elbow curve method was used to identify the centroids of the cluster of every state and using that cluster was defined. The machine learning function that was used was the K-Means technique. Clustering helped to identify the districts according to the crimes. The worst affected cities were clustered in one group and the least affected cities were in another group.

The fourth objective was to consider follow on research based on the findings of this work. The data available for this project was relatively low considering the model that was built for it. The forecasting model most of the time needs more than 20 years of data, however, the available data was from 2001 to 2013. The error rate for almost every state was pretty much low, so the project can be repeated in the future when more data is available to achieve good prediction. Clustering can be done in all states and further crimes that were not included in this research those can also be included.

The fifth objective was to forecast the trend of crimes for all the states for the next 6 years. The work carried out by the author did help to forecast the trend of crimes in each state of India with the error rate. The author built the ARIMA model which was discussed in chapter 3. The order author used to while building the machine learning model was (1,0,0). The order was calculated by the autocorrelation graph. The actual outcome and the predicted outcome both were plotted on the line chart to identify the trend of crime and the seasonal trend was also used with a frequency of 2 to observe the current and the projected trend of every state. Most of the states showed that the trend of crimes will increase over the six years and some states showed that the crimes will decrease but not rapidly.

## 5.3 CONCLUSION AND REFLECTION

In the end, the thesis was successfully able to achieve all the objectives which were mentioned by the author. Many factors affect the reason behind the crime like an increasing percentage of the unemployment rate or a steady rise in poverty. The clustering for the worst top 5 states in India was achieved successfully. ARIMA model that was built for forecasting was working well for most of the states but some states showed a high error rate. States with high crime rates were also identified and were plotted on the map of India.

The limitation of this thesis was the unavailability of the large dataset. The available dataset was from 2001 to 2013. However, the forecasting model requires a minimum of 20 years of data for more accurate predictions. Another limitation was that there unavailability of the label class. The label classes in the data can be used to determine the reasons behind the crimes and prediction can be made from the given features.

An extension for future work for this task is further analyzing the examination utilizing the information from different pieces around the world to check if the finding from this project can be applied to that. If the author starts a fresh project again, he wished to have a category or any output variable to check what are the reasons behind crimes? Clustering Algorithm can be applied to all the states and union territories to gain more information about the crimes in Indian cities. The cluster will help to identify the in-depth worst-affected cities.

In general, this undertaking was an incredible learning experience that covered all the parts of data analytics and machine learning methods. Challenges that were faced while building the working machine learning model were to draw the ACF and PACF function for every state. Some of the states didn’t show good autocorrelation function and some showed very well. Another challenge was to select the centroid point while building the clustering model. Overall every objective was achieved as expected.

# REFERENCES

AI Wiki (no date) *Machine Learning Models Explained - AI Wiki*. Available at: https://docs.paperspace.com/machine-learning/wiki/machine-learning-models-explained (Accessed: 23 July 2020).

Alashwal, H. *et al.* (2019) ‘The Application of Unsupervised Clustering Methods to Alzheimer’s Disease’, *Frontiers in Computational Neuroscience*. Frontiers Media S.A., 13, p. 31. doi: 10.3389/fncom.2019.00031.

Bajpai, G. (2019) *Crime data in India is short on information on victims and witnesses | The Indian Express*. Available at: https://indianexpress.com/article/opinion/columns/crime-data-in-india-6035032/ (Accessed: 8 July 2020).

BBC (2019) *What is India’s caste system? - BBC News*. Available at: https://www.bbc.com/news/world-asia-india-35650616 (Accessed: 11 July 2020).

Bohdan M. Pavlyshenko (2019) *(PDF) Machine-Learning Models for Sales Time Series Forecasting*. Available at: https://www.researchgate.net/publication/330484523\_Machine-Learning\_Models\_for\_Sales\_Time\_Series\_Forecasting (Accessed: 26 June 2020).

Brownlee, J. (2017) *How to Create an ARIMA Model for Time Series Forecasting in Python*. Available at: https://machinelearningmastery.com/arima-for-time-series-forecasting-with-python/ (Accessed: 24 July 2020).

Cadavid, J. P. U., Lamouri, S. and Grabot, B. (2018) *Trends in Machine Learning Applied to Demand & Sales Forecasting: A Review*. Available at: https://hal.archives-ouvertes.fr/hal-01881362/document (Accessed: 10 July 2020).

Devakunchari, R. *et al.* (2019) *Crimes Against Women in India using Regression*. Available at: https://www.ijitee.org/wp-content/uploads/papers/v8i6s4/F12970486S419.pdf (Accessed: 8 August 2020).

Dhillon, D. (2018) *Violence cost India’s economy 9% of its GDP last year - Business Insider*. Available at: https://www.businessinsider.com/violence-cost-indias-economy-9-of-its-gdp-last-year-2018-6?r=US&IR=T (Accessed: 13 August 2020).

Dissertation writers (2019) *Saunders’ Research Onion - A Step-By-Step Guide To Structuring Research Methodology Chapter For PhD And Master-Level Researchers*. Available at: https://www.ukdissertationwriters.com/saunders-research-onion-a-step-by-step-guide-to-structuring-research-methodology-chapter-for-phd-and-master-level-researchers/ (Accessed: 26 July 2020).

Dubey, S., Agarwal, P. and University Christ (2015) *Crime, Crime Rates and Control Techniques: A Statistical Analysis - Academike*. Available at: https://www.lawctopus.com/academike/crime-crime-rates/ (Accessed: 6 July 2020).

Flovik, V. (2019) *Machine Learning: From hype to real-world applications*. Available at: https://towardsdatascience.com/machine-learning-from-hype-to-real-world-applications-69de7afb56b6 (Accessed: 25 June 2020).

GISMAP (n.d.) *GisDataCollection*. Available at: https://map.igismap.com/share-map/export-layer/Indian\_States/06409663226af2f3114485aa4e0a23b4 (Accessed: 20 July 2020).

Government of India (n.d.) *States Uts - Know India: National Portal of India*. Available at: https://knowindia.gov.in/states-uts/ (Accessed: 21 July 2020).

Hamdan Ali, H. and Emad Kadhum, L. (2015) ‘K-Means Clustering Algorithm Applications in Data Mining and Pattern Recognition’, *International Journal of Science and Research (IJSR) ISSN*, 6. doi: 10.21275/ART20176024.

Heitzman, A. (2019) *Data Visualization: What It Is, Why It’s Important & How to Use It for SEO*. Available at: https://www.searchenginejournal.com/what-is-data-visualization-why-important-seo/288127/#close (Accessed: 23 July 2020).

Himabindu, B. L., Arora, R. and Prashanth, N. S. (2015) ‘Whose problem is it anyway? Crimes against women in India’, *Global Health Action*. Co-Action Publishing, 8(1). doi: 10.3402/gha.v7.23718.

Jha, R. (2019) *An analysis of crime in India’s largest urban agglomerations | ORF*. Available at: https://www.orfonline.org/expert-speak/an-analysis-of-crime-in-indias-largest-urban-agglomerations-57166/ (Accessed: 11 August 2020).

Khalid, S. and Alhambra Prieto, D. (2019) *Machine Learning for Feature Selection and Cluster Analysis in Drug Utilisation Research | SpringerLink*. Available at: https://link.springer.com/article/10.1007/s40471-019-00211-7 (Accessed: 10 July 2020).

Khanna, P. and Das, S. (2020) *Crime rate plunges amid lockdown, but domestic abuse cases may go up*. Available at: https://www.livemint.com/news/india/covid-19-stay-at-home-reduces-crime-rate-but-fear-of-domestic-violence-rises-11586746861558.html (Accessed: 8 July 2020).

Langos, S. (2018) ‘M . Sc in Marketing Management INDEPENDENT STUD Υ “ Athens as an international tourism destination : An empirical investigation to the city ’ s imagery and the role of local’, 2018(September 2014). doi: 10.13140/2.1.3023.1369.

livemint (2020) *Crime in India increased by 1.3% in 2018: NCRB*. Available at: https://www.livemint.com/news/india/crime-in-india-increased-by-1-3-in-2018-ncrb-11578562577195.html (Accessed: 24 June 2020).

Mackenzie, M. (2019) *Programming Languages For Data Scientists | by Mackenzie Mitchell | Towards Data Science*. Available at: https://towardsdatascience.com/programming-languages-for-data-scientists-afde2eaf5cc5 (Accessed: 23 July 2020).

Madhumita Jaganmohan (2020) *• India - share of tourists by expenditure 2028 | Statista*. Available at: https://www.statista.com/statistics/799499/india-share-of-tourists-by-expenditure/ (Accessed: 24 June 2020).

Madhumitha Jaganmohan (5AD) *• Monthly foreign tourist arrivals in India 2008-2018 | Statista*. Available at: https://www.statista.com/statistics/206889/number-of-foreign-tourist-arrivals-in-india-by-month/ (Accessed: 24 June 2020).

Makridakis, S., Spiliotis, E. and Assimakopoulos, V. (2018) ‘Statistical and Machine Learning forecasting methods: Concerns and ways forward’. doi: 10.1371/journal.pone.0194889.

Ministry of Home Affairs India (2016) *District-wise crimes under various sections of Indian Penal Code (IPC) crimes | Open Government Data (OGD) Platform India*. Available at: https://data.gov.in/catalog/district-wise-crimes-under-various-sections-indian-penal-code-ipc-crimes?filters%5Bfield\_catalog\_reference%5D=87615&format=json&offset=0&limit=6&sort%5Bcreated%5D=desc (Accessed: 20 July 2020).

Mondal, P. (no date) *Growing Crime Rate in India | Essay*. Available at: https://www.yourarticlelibrary.com/essay/growing-crime-rate-in-india-essay/35201 (Accessed: 11 August 2020).

Mythili, S. and Madhiya, E. (2014) *An Analysis on Clustering Algorithms in Data Mining*. Available at: https://ijcsmc.com/docs/papers/January2014/V3I1201467.pdf (Accessed: 4 July 2020).

Oyelade, O. J., Oladipupo, O. O. and Obagbuwa, I. C. (2010) ‘Application of k-Means Clustering algorithm for prediction of Students’ Academic Performance’, *IJCSIS) International Journal of Computer Science and Information Security*, 7(1). Available at: http://sites.google.com/site/ijcsis/ (Accessed: 6 July 2020).

Patel Bhai, A. (2020) *Crime against Elderly Women in India | IntechOpen*. Available at: https://www.intechopen.com/online-first/crime-against-elderly-women-in-india (Accessed: 3 July 2020).

Perkel M, J. (2018) *Why Jupyter is data scientists’ computational notebook of choice*. Available at: https://www.nature.com/articles/d41586-018-07196-1 (Accessed: 23 July 2020).

Pierre, A. G. and Fredrik, F. (n.d.) *Unsupervised Machine Learning: An Investigation of Clustering Algorithms on a Small Dataset*. Available at: https://www.diva-portal.org/smash/get/diva2:1213516/FULLTEXT01.pdf (Accessed: 4 July 2020).

Pradhan, M. and Kumar, U. D. (2019) *Machine Learning using Python*. 1st edn. Willey. Available at: https://www.wileyindia.com/machine-learning-using-python.html.

Prieto Curiel, R. *et al.* (2020) ‘Crime and its fear in social media’, *Palgrave Communications*. Palgrave Macmillan Ltd., 6(1), pp. 1–12. doi: 10.1057/s41599-020-0430-7.

Rahman Maseesh (2006) *Indian leader likens caste system to apartheid regime | World news | The Guardian*. Available at: https://www.theguardian.com/world/2006/dec/28/india.mainsection (Accessed: 30 June 2020).

Rao, J. (2010) *The Caste System: Effects on Poverty in India, Nepal and Sri Lanka*, *Global Majority E-Journal*.

Ropmay, A. D. (2014) ‘Review Research Paper Crimes against Women in Matrilineal Meghalaya A Forensic Medical Perspective’, *J Indian Acad Forensic Med*, 36(4).

Salkind J, N. (no date) *Pie Chart - SAGE Research Methods*, *2010*. Available at: https://methods.sagepub.com/reference/encyc-of-research-design/n311.xml (Accessed: 23 July 2020).

Sathyadevan, S., Devan, M. S. and Surya Gangadharan, S. (2014) ‘Crime analysis and prediction using data mining’, in *1st International Conference on Networks and Soft Computing, ICNSC 2014 - Proceedings*. Institute of Electrical and Electronics Engineers Inc., pp. 406–412. doi: 10.1109/CNSC.2014.6906719.

Saunders, M., Lewis, P. and Thornhill, A. (2009) *Research methods for business students*. Financial Times Prentice Hall. Available at: http://ezproxy.ait.ie/login?url=http://search.ebscohost.com/login.aspx?direct=true&AuthType=shib&db=cat06239a&AN=ait.16912&site=eds-live.

Sharma, S. (2015) ‘Caste-based crimes and economic status: Evidence from India’, *Journal of Comparative Economics*. Academic Press Inc., 43(1), pp. 204–226. doi: 10.1016/j.jce.2014.10.005.

Singh, V. (2020) *Uttar Pradesh tops list in crimes against women - The Hindu*. Available at: https://www.thehindu.com/news/national/uttar-pradesh-tops-in-crimes-against-women/article30526919.ece (Accessed: 11 August 2020).

Statista (2019) *• India - hate crimes 2018 | Statista*, *Statista Research Department,*. Available at: https://www.statista.com/statistics/978957/hate-crimes-india/ (Accessed: 2 July 2020).

Statstics Times (2020) *Population of Indian states 2019 - StatisticsTimes.com*. Available at: http://statisticstimes.com/demographics/population-of-indian-states.php (Accessed: 30 July 2020).

Tourmyindia (n.d.) *Indian States Information | A Travel Guide to State of India*. Available at: https://www.tourmyindia.com/states/ (Accessed: 30 July 2020).

Tripathi, A. and Yadav, S. (2004) *For the Sake of Honour: But Whose Honour? ‘Honour Crimes’ Against Women*. Available at: https://www.researchgate.net/publication/248747558\_For\_the\_Sake\_of\_Honour\_But\_Whose\_Honour\_Honour\_Crimes\_Against\_Women (Accessed: 30 June 2020).

Vorhies, W. (2016) *CRISP-DM – a Standard Methodology to Ensure a Good Outcome - Data Science Central*. Available at: https://www.datasciencecentral.com/profiles/blogs/crisp-dm-a-standard-methodology-to-ensure-a-good-outcome (Accessed: 15 July 2020).

Wowczko, I. (2015) *A Case Study of Evaluating Job Readiness with Data Mining Tools and CRISP-DM Methodology*.

Yi, M. (2019) *A Complete Guide to Line Charts | Tutorial by Chartio*. Available at: https://chartio.com/learn/charts/line-chart-complete-guide/ (Accessed: 23 July 2020).

### GLOSSARY OF TERMS

ARIMA: The technique which is used for forecasting is ARIMA. The ARIMA model predicts a future outcome based on previous data.

Clustering; Clustering is a part of the supervised machine learning algorithm which creates subgroups according to the task

### LIST OF ABBREVIATIONS

|  |  |
| --- | --- |
| ML | Machine Learning |
| ACF | Autocorrelation function |
| PACF | Partial Autocorrelation function |
| RMSE | Root Mean Square Error |
| MAPE | Mean Absolute Percentage Error |
| IDE | Integrated Development Environments |
| DBSCAN | Density-Based Spatial Clustering of Applications with Noise |

-

# APPENDIX

Project Code:

crimes\_df = pd.read\_csv("Crime\_rates\_2001-13.csv")

crimes\_main\_df = crimes\_df.copy()

crimes\_df["TOTAL\_THEFT"] = crimes\_df["THEFT"] + crimes\_df["OTHER THEFT"] + crimes\_df["AUTO THEFT"]

fp = "D:\\Igismap\\Indian\_States.shp"

map\_India = gdp.read\_file(fp)

map\_India['st\_nm'] = map\_India['st\_nm'].replace(['Andaman & Nicobar Island', "Andhra Pradesh", 'Arunanchal Pradesh', 'Assam', 'Bihar', 'Chandigarh', 'Chhattisgarh', 'Dadara & Nagar Havelli', 'Daman & Diu', 'NCT of Delhi', 'Goa', 'Gujarat', 'Haryana', 'Himachal Pradesh', 'Jammu & Kashmir', 'Jharkhand', 'Karnataka', 'Kerala', 'Lakshadweep', 'Madhya Pradesh', 'Maharashtra', 'Manipur', 'Meghalaya', 'Mizoram', 'Nagaland', 'Odisha', 'Puducherry', 'Punjab', 'Rajasthan', 'Sikkim', 'Tamil Nadu', 'Tripura', 'Uttar Pradesh', 'Uttarakhand', 'West Bengal'], ["A & N ISLANDS", "ANDHRA PRADESH", 'ARUNACHAL PRADESH', 'ASSAM', 'BIHAR', 'CHANDIGARH', 'CHHATTISGARH', 'D & N HAVELI', 'DAMAN & DIU', 'DELHI UT', 'GOA', 'GUJARAT', 'HARYANA', 'HIMACHAL PRADESH', 'JAMMU & KASHMIR', 'JHARKHAND', 'KARNATAKA', 'KERALA', 'LAKSHADWEEP', 'MADHYA PRADESH', 'MAHARASHTRA', 'MANIPUR', MEGHALAYA', 'MIZORAM', 'NAGALAND','ODISHA', 'PUDUCHERRY', 'PUNJAB', 'RAJASTHAN', 'SIKKIM', 'TAMIL NADU', 'TRIPURA', 'UTTAR PRADESH', 'UTTARAKHAND', 'WEST BENGAL'])

crimes\_df = crimes\_df.rename(columns={'STATE/UT': 'STATE'})

crimes\_df = crimes\_df.rename(columns={'TOTAL IPC CRIMES': 'TOTAL'})

crimes\_df['STATE'] = crimes\_df['STATE'].replace(['A&N Islands', "Andhra Pradesh", 'Arunachal Pradesh', 'Assam', 'Bihar', 'Chandigarh', 'Chhattisgarh', 'D&N Haveli', 'Daman & Diu', 'Delhi UT', 'Goa', 'Gujarat', 'Haryana', 'Himachal Pradesh', 'Jammu & Kashmir', 'Jharkhand', 'Karnataka', 'Kerala', 'Lakshadweep', 'Madhya Pradesh', 'Maharashtra', 'Manipur',

'Meghalaya', 'Mizoram', 'Nagaland', 'Odisha', 'Puducherry', 'Punjab', 'Rajasthan', 'Sikkim', 'Tamil Nadu', 'Tripura', 'Uttar Pradesh', 'Uttarakhand', 'West Bengal'], ["A & N ISLANDS", "ANDHRA PRADESH", 'ARUNACHAL PRADESH', 'ASSAM', 'BIHAR', 'CHANDIGARH', 'CHHATTISGARH', 'D & N HAVELI', 'DAMAN & DIU', 'DELHI UT', 'GOA', 'GUJARAT', 'HARYANA', 'HIMACHAL PRADESH', 'JAMMU & KASHMIR', 'JHARKHAND', 'KARNATAKA', 'KERALA', 'LAKSHADWEEP', 'MADHYA PRADESH', 'MAHARASHTRA', 'MANIPUR', 'MEGHALAYA', 'MIZORAM', 'NAGALAND','ODISHA', 'PUDUCHERRY', 'PUNJAB', 'RAJASTHAN', 'SIKKIM', 'TAMIL NADU', 'TRIPURA', 'UTTAR PRADESH', 'UTTARAKHAND', 'WEST BENGAL'])

crimes\_df['DISTRICT'] = crimes\_df['DISTRICT'].replace(["ZZ TOTAL"], ["TOTAL"])

crimes\_df['DISTRICT'] = crimes\_df['DISTRICT'].replace(["SOUTH-EAST", "SOUTH-WEST", "NORTH-EAST", "NORTH-WEST", “IGI AIRPORT", "GRP(RLY)", "STF", 'DELHI UT TOTAL'], ["SOUTH EAST", "SOUTH WEST", "NORTH EAST", "NORTH WEST","I.G.I. AIRPORT", "G.R.P.(RLY)", "S.T.F.", 'TOTAL'])

crimes\_df['DISTRICT'] = crimes\_df['DISTRICT'].replace(["HOWRAH CITY"], ["HOWRAH"])

crimes\_df['DISTRICT'] = crimes\_df['DISTRICT'].replace(['G.R.P. AJMER', 'G.R.P. JODHPUR'],['G.R.P.AJMER', 'G.R.P.JODHPUR'])

crimes\_df.rename(columns={'HURT/GREVIOUS HURT': 'HURT', 'KIDNAPPING & ABDUCTION': 'KIDNAPPING', 'CAUSING DEATH BY NEGLIGENCE':'DEATH\_BY\_NEGLIGENCE', 'ASSAULT ON WOMEN WITH INTENT TO OUTRAGE HER MODESTY':'ASSAULT\_WOMEN\_MODESTY','CRUELTY BY HUSBAND OR HIS RELATIVES': 'CRUELTY\_BY\_HUSBAND', 'CRIMINAL BREACH OF TRUST':'CRIMINAL\_BREACH\_OF\_TRUST'}, inplace=True)

crimes\_state = crimes\_df.groupby("STATE").agg({'TOTAL\_THEFT':sum,'HURT':sum,'KIDNAPPING':sum,'CRUELTY\_BY\_HUSBAND':sum, 'BURGLARY':sum, 'MURDER':sum,'DEATH\_BY\_NEGLIGENCE':sum, 'CHEATING':sum, 'RIOTS':sum, 'ASSAULT\_WOMEN\_MODESTY':sum ,'OTHER IPC CRIMES':sum, 'TOTAL':sum}).reset\_index()

merged\_Ind = map\_India.set\_index('st\_nm').join(crimes\_state.set\_index('STATE'))

crimes\_distribution = crimes\_df[crimes\_df.DISTRICT=='TOTAL']

crimes\_distribution = crimes\_distribution.groupby(["YEAR"]).agg({'MURDER':sum, 'ATTEMPT TO MURDER':sum,

'CULPABLE HOMICIDE NOT AMOUNTING TO MURDER':sum, 'RAPE':sum, 'CUSTODIAL RAPE':sum,

'OTHER RAPE':sum, 'KIDNAPPING':sum,

'KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS':sum,

'KIDNAPPING AND ABDUCTION OF OTHERS':sum, 'DACOITY':sum,

'PREPARATION AND ASSEMBLY FOR DACOITY':sum, 'ROBBERY':sum, 'BURGLARY':sum,

'RIOTS':sum, 'CRIMINAL\_BREACH\_OF\_TRUST':sum,

'CHEATING':sum, 'COUNTERFIETING':sum, 'ARSON':sum, 'HURT':sum,

'DOWRY DEATHS':sum, 'ASSAULT\_WOMEN\_MODESTY':sum,

'INSULT TO MODESTY OF WOMEN':sum, 'CRUELTY\_BY\_HUSBAND':sum,

'IMPORTATION OF GIRLS FROM FOREIGN COUNTRIES':sum,

'DEATH\_BY\_NEGLIGENCE':sum, 'OTHER IPC CRIMES':sum, "TOTAL\_THEFT":sum}).reset\_index()

Crimes\_distribution = crimes\_distribution.append(crimes\_distribution.sum().rename('total'))

crimes\_distribution['YEAR'].replace(26091, 'Total', inplace=True)

crimes\_distribution = crimes\_distribution[crimes\_distribution['YEAR'] == 'Total']

crimes\_sum = crimes\_distribution.T.reset\_index()

labels = ['MURDER', 'ATTEMPT TO MURDER',

'CULPABLE HOMICIDE NOT AMOUNTING TO MURDER', 'RAPE', 'KIDNAPPING & ABDUCTION', 'DACOITY',

'PREPARATION AND ASSEMBLY FOR DACOITY', 'ROBBERY', 'BURGLARY',

'RIOTS', 'CRIMINAL BREACH OF TRUST',

'CHEATING', 'COUNTERFIETING', 'ARSON', 'HURT/GREVIOUS HURT',

'DOWRY DEATHS', 'ASSAULT ON WOMEN WITH INTENT TO OUTRAGE HER MODESTY',

'INSULT TO MODESTY OF WOMEN', 'CRUELTY BY HUSBAND OR HIS RELATIVES',

'IMPORTATION OF GIRLS FROM FOREIGN COUNTRIES',

'CAUSING DEATH BY NEGLIGENCE', "THEFTTOTAL"]

values = [435744, 387394, 47994, 272844, 422318, 63948, 35100, 280194, 1221740,847053,

203648, 874927, 29067, 123061, 3706545, 99285, 523894, 136643, 994067, 923, 1113381, 6310244]

fig = go.Figure(data=[go.Pie(labels=labels, values=values ,textinfo='label+percent',

)])

fig.update\_layout(

uniformtext\_minsize= 20,

title\_text="Distribution of Crimes",

paper\_bgcolor='rgb(233,233,233)',

autosize=False,

width=1150,

height=800)

fig.show()

labels = ['MURDER', 'KIDNAPPING & ABDUCTION', 'BURGLARY',

'RIOTS',

'CHEATING', 'HURT/GREVIOUS HURT',

'ASSAULT ON WOMEN WITH INTENT TO OUTRAGE HER MODESTY',

'CRUELTY BY HUSBAND OR HIS RELATIVES',

'CAUSING DEATH BY NEGLIGENCE', "THEFTTOTAL"]

values = [435744, 422318, 1221740,847053,

874927, 3706545, 523894, 994067, 1113381, 6310244]

fig = go.Figure(data=[go.Pie(labels=labels, values=values ,textinfo='label+percent',

)])

fig.update\_layout(

uniformtext\_minsize= 20,

title\_text="Top 10 Crimes in India",

paper\_bgcolor='rgb(233,233,233)',

autosize=False,

width=1000,

height=900)

fig.show()

fig, ax = plt.subplots(1, figsize=(15, 11))

ax.axis('off')

ax.set\_title('TOTAL IPC CRIMES', fontdict={'fontsize': '25', 'fontweight' : '3'})

# plot the figure

merged\_Ind.dropna().plot(column='TOTAL', cmap='PuRd', figsize=fig, scheme='equal\_interval',linewidth=0.8, ax=ax, edgecolor='grey')

norm = Normalize(vmin=merged\_Ind['TOTAL'].min(), vmax= merged\_Ind["TOTAL"].max())

n\_cmap = cm.ScalarMappable(norm=norm, cmap='PuRd')

n\_cmap.set\_array([])

ax.get\_figure().colorbar(n\_cmap)

ax.set\_axis\_off()

plt.axis('equal')

plt.show()

fig, ax = plt.subplots(1, figsize=(15, 11))

ax.axis('off')

ax.set\_title('THEFT IN INDIA', fontdict={'fontsize': '25', 'fontweight' : '3'})

# plot the figure

merged\_Ind.dropna().plot(column='TOTAL\_THEFT', cmap='PuRd', figsize=fig, scheme='equal\_interval',linewidth=0.8, ax=ax, edgecolor='grey')

norm = Normalize(vmin=merged\_Ind['TOTAL\_THEFT'].min(), vmax= merged\_Ind["TOTAL\_THEFT"].max())

n\_cmap = cm.ScalarMappable(norm=norm, cmap='PuRd')

n\_cmap.set\_array([])

ax.get\_figure().colorbar(n\_cmap)

ax.set\_axis\_off()

plt.axis('equal')

plt.show()

fig, ax = plt.subplots(1, figsize=(15, 11))

ax.axis('off')

ax.set\_title('HURTS IN INDIA', fontdict={'fontsize': '25', 'fontweight' : '3'})

# plot the figure

merged\_Ind.dropna().plot(column='HURT', cmap='PuRd', figsize=fig, scheme='equal\_interval',linewidth=0.8, ax=ax, edgecolor='grey')

norm = Normalize(vmin=merged\_Ind['HURT'].min(), vmax= merged\_Ind["HURT"].max())

n\_cmap = cm.ScalarMappable(norm=norm, cmap='PuRd')

n\_cmap.set\_array([])

ax.get\_figure().colorbar(n\_cmap)

ax.set\_axis\_off()

plt.axis('equal')

plt.show()

fig, ax = plt.subplots(1, figsize=(15, 11))

ax.axis('off')

ax.set\_title('KIDNAPPING IN INDIA', fontdict={'fontsize': '25', 'fontweight' : '3'})

# plot the figure

merged\_Ind.dropna().plot(column='KIDNAPPING', cmap='PuRd', figsize=fig, scheme='equal\_interval',linewidth=0.8, ax=ax, edgecolor='grey')

norm = Normalize(vmin=merged\_Ind['KIDNAPPING'].min(), vmax= merged\_Ind["KIDNAPPING"].max())

n\_cmap = cm.ScalarMappable(norm=norm, cmap='PuRd')

n\_cmap.set\_array([])

ax.get\_figure().colorbar(n\_cmap)

ax.set\_axis\_off()

plt.axis('equal')

plt.show()

fig, ax = plt.subplots(1, figsize=(15, 11))

ax.axis('off')

ax.set\_title('CHEATING IN INDIA', fontdict={'fontsize': '25', 'fontweight' : '3'})

# plot the figure

merged\_Ind.dropna().plot(column='CHEATING', cmap='PuRd', figsize=fig, scheme='equal\_interval',linewidth=0.8, ax=ax, edgecolor='grey')

norm = Normalize(vmin=merged\_Ind['CHEATING'].min(), vmax= merged\_Ind["CHEATING"].max())

n\_cmap = cm.ScalarMappable(norm=norm, cmap='PuRd')

n\_cmap.set\_array([])

ax.get\_figure().colorbar(n\_cmap)

ax.set\_axis\_off()

plt.axis('equal')

plt.show()

fig, ax = plt.subplots(1, figsize=(15, 11))

ax.axis('off')

ax.set\_title('BURGLARY IN INDIA', fontdict={'fontsize': '25', 'fontweight' : '3'})

# plot the figure

merged\_Ind.dropna().plot(column='BURGLARY', cmap='PuRd', figsize=fig, scheme='equal\_interval',linewidth=0.8, ax=ax, edgecolor='grey')

norm = Normalize(vmin=merged\_Ind['BURGLARY'].min(), vmax= merged\_Ind["BURGLARY"].max())

n\_cmap = cm.ScalarMappable(norm=norm, cmap='PuRd')

n\_cmap.set\_array([])

ax.get\_figure().colorbar(n\_cmap)

ax.set\_axis\_off()

plt.axis('equal')

plt.show()

fig, ax = plt.subplots(1, figsize=(15, 11))

ax.axis('off')

ax.set\_title('RIOTS IN INDIA', fontdict={'fontsize': '25', 'fontweight' : '3'})

# plot the figure

merged\_Ind.dropna().plot(column='RIOTS', cmap='PuRd', figsize=fig, scheme='equal\_interval',linewidth=0.8, ax=ax, edgecolor='grey')

norm = Normalize(vmin=merged\_Ind['RIOTS'].min(), vmax= merged\_Ind["RIOTS"].max())

n\_cmap = cm.ScalarMappable(norm=norm, cmap='PuRd')

n\_cmap.set\_array([])

ax.get\_figure().colorbar(n\_cmap)

ax.set\_axis\_off()

plt.axis('equal')

plt.show()

fig, ax = plt.subplots(1, figsize=(15, 11))

ax.axis('off')

ax.set\_title('MURDER', fontdict={'fontsize': '25', 'fontweight' : '3'})

# plot the figure

merged\_Ind.dropna().plot(column='MURDER', cmap='PuRd', figsize=fig, scheme='equal\_interval',linewidth=0.8, ax=ax, edgecolor='grey')

norm = Normalize(vmin=merged\_Ind['MURDER'].min(), vmax= merged\_Ind["MURDER"].max())

n\_cmap = cm.ScalarMappable(norm=norm, cmap='PuRd')

n\_cmap.set\_array([])

ax.get\_figure().colorbar(n\_cmap)

ax.set\_axis\_off()

plt.axis('equal')

plt.show()

fig, ax = plt.subplots(1, figsize=(15, 11))

ax.axis('off')

ax.set\_title('DEATH\_BY\_NEGLIGENCE', fontdict={'fontsize': '25', 'fontweight' : '3'})

# plot the figure

merged\_Ind.dropna().plot(column='DEATH\_BY\_NEGLIGENCE', cmap='PuRd', figsize=fig, scheme='equal\_interval',linewidth=0.8, ax=ax, edgecolor='grey')

norm = Normalize(vmin=merged\_Ind['DEATH\_BY\_NEGLIGENCE'].min(), vmax= merged\_Ind["DEATH\_BY\_NEGLIGENCE"].max())

n\_cmap = cm.ScalarMappable(norm=norm, cmap='PuRd')

n\_cmap.set\_array([])

ax.get\_figure().colorbar(n\_cmap)

ax.set\_axis\_off()

plt.axis('equal')

plt.show()

fig, ax = plt.subplots(1, figsize=(15, 11))

ax.axis('off')

ax.set\_title('OTHER IPC CRIMES', fontdict={'fontsize': '25', 'fontweight' : '3'})

# plot the figure

merged\_Ind.dropna().plot(column='OTHER IPC CRIMES', cmap='PuRd', figsize=fig, scheme='equal\_interval',linewidth=0.8, ax=ax, edgecolor='grey')

norm = Normalize(vmin=merged\_Ind['OTHER IPC CRIMES'].min(), vmax= merged\_Ind["OTHER IPC CRIMES"].max())

n\_cmap = cm.ScalarMappable(norm=norm, cmap='PuRd')

n\_cmap.set\_array([])

ax.get\_figure().colorbar(n\_cmap)

ax.set\_axis\_off()

plt.axis('equal')

plt.show()

crimes\_total = crimes\_df.groupby(['STATE', "YEAR"]).agg({"TOTAL": sum }).reset\_index()

crime\_AN\_df = crimes\_total[crimes\_total.STATE=="A & N ISLANDS" ]

ts\_decompose = sm.tsa.seasonal\_decompose(np.array(crime\_AN\_df['TOTAL']),

model = "multiplicative", freq=2)

ts\_plot = ts\_decompose.plot()

crime\_AN\_forecast = crime\_AN\_df.copy()

crime\_AN\_forecast.drop('STATE', axis = 1, inplace = True)

acf\_plot = plot\_acf(crime\_AN\_forecast.TOTAL, lags = 10)

pcf\_1 = plot\_pacf(crime\_AN\_forecast.TOTAL, lags = 10)

an\_arim = ARIMA(crime\_AN\_forecast.TOTAL[:10].astype(np.float64).as\_matrix(), order=(1,0,0))

an\_model = an\_arim.fit()

an\_model.summary2()

def get\_mape(actual, predicted):

y\_actual = np.array(actual)

y\_pred = np.array(predicted)

return np.round(np.mean(np.abs((actual - predicted)/actual)) \* 100, 2)

forecast\_an = an\_model.predict(10,12)

forecast\_an

forecast\_an\_df, stderr, ci = an\_model.forecast(steps=7)

forecast\_an\_df

get\_mape(crime\_AN\_forecast.TOTAL[10:13],forecast\_an)

i=13

for j in range(len(forecast\_an\_df)):

zcrime\_AN\_forecast.loc[i] = crime\_AN\_forecast.YEAR[i-1]+1

crime\_AN\_forecast.TOTAL[i] = forecast\_an\_df[j]

i = i+1

fig = go.Figure()

fig.add\_trace(go.Scatter(x= crime\_AN\_df["YEAR"], y= crime\_AN\_df['TOTAL'],

name = "A&N",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= crime\_AN\_forecast["YEAR"][13:], y= crime\_AN\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

def plot(dataframe):

acf\_plot = plot\_acf(dataframe.TOTAL, lags = 10)

pcf\_1 = plot\_pacf(dataframe.TOTAL, lags = 10)

return acf\_plot, pcf\_1

crimes\_AP\_forecast = crime\_AP\_df.copy()

crimes\_AP\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_AP\_forecast = crimes\_AP\_forecast.reset\_index(drop = True)

plot(crimes\_AP\_forecast)

AP\_arim = ARIMA(crimes\_AP\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

AP\_moAP = AP\_arim.fit()

AP\_moAP.summary2()

forecast\_AP = AP\_moAP.predict(10,12)

print('prediction',forecast\_AP)

forecast\_df\_AP, stderr, ci = AP\_moAP.forecast(steps=7)

print('forecasting',forecast\_df\_AP)

get\_mape(crimes\_AP\_forecast.TOTAL[10:13],forecast\_AP)

i=13

for j in range(len(forecast\_df\_AP)):

crimes\_AP\_forecast.loc[i] = crimes\_AP\_forecast.YEAR[i-1]+1

crimes\_AP\_forecast.TOTAL[i] = forecast\_df\_AP[j]

i = i+1

fig = go.Figure()

fig.add\_trace(go.Scatter(x= crimes\_AP\_forecast["YEAR"], y= crimes\_AP\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= crimes\_AP\_forecast["YEAR"][13:], y= crimes\_AP\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_AP\_df = crimes\_total[crimes\_total.STATE=='ANDHRA PRADESH']

ts\_decompose = sm.tsa.seasonal\_decompose(np.array(crime\_AP\_df['TOTAL']),

model = "multiplicative", freq=2)

ts\_plot = ts\_decompose.plot()

crimes\_AP\_forecast = crime\_AP\_df.copy()

crimes\_AP\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_AP\_forecast = crimes\_AP\_forecast.reset\_index(drop = True)

plot(crimes\_AP\_forecast)

AP\_arim = ARIMA(crimes\_AP\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

AP\_moAP = AP\_arim.fit()

AP\_moAP.summary2()

forecast\_AP = AP\_moAP.predict(10,12)

print('prediction',forecast\_AP)

forecast\_df\_AP, stderr, ci = AP\_moAP.forecast(steps=7)

print('forecasting',forecast\_df\_AP)

get\_mape(crimes\_AP\_forecast.TOTAL[10:13],forecast\_AP)

i=13

for j in range(len(forecast\_df\_AP)):

crimes\_AP\_forecast.loc[i] = crimes\_AP\_forecast.YEAR[i-1]+1

crimes\_AP\_forecast.TOTAL[i] = forecast\_df\_AP[j]

i = i+1

fig = go.Figure()

fig.add\_trace(go.Scatter(x= crimes\_AP\_forecast["YEAR"], y= crimes\_AP\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= crimes\_AP\_forecast["YEAR"][13:], y= crimes\_AP\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_ARU\_df = crimes\_total[crimes\_total.STATE=='ANDHRA PRADESH']

ts\_decompose = sm.tsa.seasonal\_decompose(np.array(crime\_ARU\_df['TOTAL']),

model = "multiplicative", freq=2)

ts\_plot = ts\_decompose.plot()

crimes\_ARU\_forecast = crime\_ARU\_df.copy()

crimes\_ARU\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_ARU\_forecast = crimes\_ARU\_forecast.reset\_index(drop = True)

plot(crimes\_ARU\_forecast)

ARU\_arim = ARIMA(crimes\_ARU\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

ARU\_moARU = ARU\_arim.fit()

ARU\_moARU.summary2()

forecast\_ARU = ARU\_moARU.predict(10,12)

print('prediction',forecast\_ARU)

forecast\_df\_ARU, stderr, ci = ARU\_moARU.forecast(steps=7)

print('forecasting',forecast\_df\_ARU)

get\_mARUe(crimes\_ARU\_forecast.TOTAL[10:13],forecast\_ARU)

i=13

for j in range(len(forecast\_df\_ARU)):

crimes\_ARU\_forecast.loc[i] = crimes\_ARU\_forecast.YEAR[i-1]+1

crimes\_ARU\_forecast.TOTAL[i] = forecast\_df\_ARU[j]

i = i+1

fig = go.Figure()

fig.add\_trace(go.Scatter(x= crimes\_ARU\_forecast["YEAR"], y= crimes\_ARU\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= crimes\_ARU\_forecast["YEAR"][13:], y= crimes\_ARU\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_ASSAM\_df = crimes\_total[crimes\_total.STATE=='ASSAM']

ts\_decompose = sm.tsa.seasonal\_decompose(np.array(crime\_ASSAM\_df['TOTAL']),

model = "multiplicative", freq=2)

ts\_plot = ts\_decompose.plot()

crimes\_ASSAM\_forecast = crime\_ASSAM\_df.copy()

crimes\_ASSAM\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_ASSAM\_forecast = crimes\_ASSAM\_forecast.reset\_index(drop = True)

plot(crimes\_ASSAM\_forecast)

ASSAM\_arim = ARIMA(crimes\_ASSAM\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

ASSAM\_moASSAM = ASSAM\_arim.fit()

ASSAM\_moASSAM.summary2()

forecast\_ASSAM = ASSAM\_moASSAM.predict(10,12)

print('prediction',forecast\_ASSAM)

forecast\_df\_ASSAM, stderr, ci = ASSAM\_moASSAM.forecast(steps=7)

print('forecasting',forecast\_df\_ASSAM)

get\_mASSAMe(crimes\_ASSAM\_forecast.TOTAL[10:13],forecast\_ASSAM)

i=13

for j in range(len(forecast\_df\_ASSAM)):

crimes\_ASSAM\_forecast.loc[i] = crimes\_ASSAM\_forecast.YEAR[i-1]+1

crimes\_ASSAM\_forecast.TOTAL[i] = forecast\_df\_ASSAM[j]

i = i+1

fig = go.Figure()

fig.add\_trace(go.Scatter(x= crimes\_ASSAM\_forecast["YEAR"], y= crimes\_ASSAM\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= crimes\_ASSAM\_forecast["YEAR"][13:], y= crimes\_ASSAM\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_BIH\_df = crimes\_total[crimes\_total.STATE=='BIHAR']

ts\_decompose = sm.tsa.seasonal\_decompose(np.array(crime\_BIH\_df['TOTAL']),

model = "multiplicative", freq=2)

ts\_plot = ts\_decompose.plot()

crimes\_BIH\_forecast = crime\_BIH\_df.copy()

crimes\_BIH\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_BIH\_forecast = crimes\_BIH\_forecast.reset\_index(drop = True)

plot(crimes\_BIH\_forecast)

BIH\_arim = ARIMA(crimes\_BIH\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

BIH\_moBIH = BIH\_arim.fit()

BIH\_moBIH.summary2()

forecast\_BIH = BIH\_moBIH.predict(10,12)

print('prediction',forecast\_BIH)

forecast\_df\_BIH, stderr, ci = BIH\_moBIH.forecast(steps=7)

print('forecasting',forecast\_df\_BIH)

get\_mBIHe(crimes\_BIH\_forecast.TOTAL[10:13],forecast\_BIH)

i=13

for j in range(len(forecast\_df\_BIH)):

crimes\_BIH\_forecast.loc[i] = crimes\_BIH\_forecast.YEAR[i-1]+1

crimes\_BIH\_forecast.TOTAL[i] = forecast\_df\_BIH[j]

i = i+1

fig = go.Figure()

fig.add\_trace(go.Scatter(x= crimes\_BIH\_forecast["YEAR"], y= crimes\_BIH\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= crimes\_BIH\_forecast["YEAR"][13:], y= crimes\_BIH\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_CHH\_df = crimes\_total[crimes\_total.STATE=='CHHATTISGARH']

ts\_decompose = sm.tsa.seasonal\_decompose(np.array(crime\_CHH\_df['TOTAL']),

model = "multiplicative", freq=2)

ts\_plot = ts\_decompose.plot()

crimes\_CHH\_forecast = crime\_CHH\_df.copy()

crimes\_CHH\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_CHH\_forecast = crimes\_CHH\_forecast.reset\_index(drop = True)

plot(crimes\_CHH\_forecast)

CHH\_arim = ARIMA(crimes\_CHH\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

CHH\_moCHH = CHH\_arim.fit()

CHH\_moCHH.summary2()

forecast\_CHH = CHH\_moCHH.predict(10,12)

print('prediction',forecast\_CHH)

forecast\_df\_CHH, stderr, ci = CHH\_moCHH.forecast(steps=7)

print('forecasting',forecast\_df\_CHH)

get\_mCHHe(crimes\_CHH\_forecast.TOTAL[10:13],forecast\_CHH)

i=13

for j in range(len(forecast\_df\_CHH)):

crimes\_CHH\_forecast.loc[i] = crimes\_CHH\_forecast.YEAR[i-1]+1

crimes\_CHH\_forecast.TOTAL[i] = forecast\_df\_CHH[j]

i = i+1

fig = go.Figure()

fig.add\_trace(go.Scatter(x= crimes\_CHH\_forecast["YEAR"], y= crimes\_CHH\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= crimes\_CHH\_forecast["YEAR"][13:], y= crimes\_CHH\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_GOA\_df = crimes\_total[crimes\_total.STATE=='GOA']

ts\_decompose = sm.tsa.seasonal\_decompose(np.array(crime\_GOA\_df['TOTAL']),

model = "multiplicative", freq=2)

ts\_plot = ts\_decompose.plot()

crimes\_GOA\_forecast = crime\_GOA\_df.copy()

crimes\_GOA\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_GOA\_forecast = crimes\_GOA\_forecast.reset\_index(drop = True)

plot(crimes\_GOA\_forecast)

GOA\_arim = ARIMA(crimes\_GOA\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

GOA\_moGOA = GOA\_arim.fit()

GOA\_moGOA.summary2()

forecast\_GOA = GOA\_moGOA.predict(10,12)

print('prediction',forecast\_GOA)

forecast\_df\_GOA, stderr, ci = GOA\_moGOA.forecast(steps=7)

print('forecasting',forecast\_df\_GOA)

get\_mGOAe(crimes\_GOA\_forecast.TOTAL[10:13],forecast\_GOA)

i=13

for j in range(len(forecast\_df\_GOA)):

crimes\_GOA\_forecast.loc[i] = crimes\_GOA\_forecast.YEAR[i-1]+1

crimes\_GOA\_forecast.TOTAL[i] = forecast\_df\_GOA[j]

i = i+1

fig = go.Figure()

fig.add\_trace(go.Scatter(x= crimes\_GOA\_forecast["YEAR"], y= crimes\_GOA\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= crimes\_GOA\_forecast["YEAR"][13:], y= crimes\_GOA\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_GUJ\_df = crimes\_total[crimes\_total.STATE=='GUJUJARAT']

ts\_decompose = sm.tsa.seasonal\_decompose(np.array(crime\_GUJOA\_df['TOTAL']),

model = "multiplicative", freq=2)

ts\_plot = ts\_decompose.plot()

crimes\_GUJOA\_forecast = crime\_GUJOA\_df.copy()

crimes\_GUJOA\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_GUJOA\_forecast = crimes\_GUJOA\_forecast.reset\_index(drop = True)

plot(crimes\_GUJOA\_forecast)

GUJOA\_arim = ARIMA(crimes\_GUJOA\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

GUJOA\_moGUJOA = GUJOA\_arim.fit()

GUJOA\_moGUJOA.summary2()

forecast\_GUJOA = GUJOA\_moGUJOA.predict(10,12)

print('prediction',forecast\_GUJOA)

forecast\_df\_GUJOA, stderr, ci = GUJOA\_moGUJOA.forecast(steps=7)

print('forecastinGUJ',forecast\_df\_GUJOA)

GUJet\_mGUJOAe(crimes\_GUJOA\_forecast.TOTAL[10:13],forecast\_GUJOA)

i=13

for j in ranGUJe(len(forecast\_df\_GUJOA)):

crimes\_GUJOA\_forecast.loc[i] = crimes\_GUJOA\_forecast.YEAR[i-1]+1

crimes\_GUJOA\_forecast.TOTAL[i] = forecast\_df\_GUJOA[j]

i = i+1

fiGUJ = GUJo.FiGUJure()

fiGUJ.add\_trace(GUJo.Scatter(x= crimes\_GUJOA\_forecast["YEAR"], y= crimes\_GUJOA\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fiGUJ.add\_trace(GUJo.Scatter(x= crimes\_GUJOA\_forecast["YEAR"][13:], y= crimes\_GUJOA\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_HAR\_df = crimes\_total[crimes\_total.STATE=='HARYANA']

ts\_decompose = sm.tsa.seasonal\_decompose(np.array(crime\_HAROA\_df['TOTAL']),

model = "multiplicative", freq=2)

ts\_plot = ts\_decompose.plot()

crimes\_HAROA\_forecast = crime\_HAROA\_df.copy()

crimes\_HAROA\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_HAROA\_forecast = crimes\_HAROA\_forecast.reset\_index(drop = True)

plot(crimes\_HAROA\_forecast)

HAROA\_arim = ARIMA(crimes\_HAROA\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

HAROA\_moHAROA = HAROA\_arim.fit()

HAROA\_moHAROA.summary2()

forecast\_HAROA = HAROA\_moHAROA.predict(10,12)

print('prediction',forecast\_HAROA)

forecast\_df\_HAROA, stderr, ci = HAROA\_moHAROA.forecast(steps=7)

print('forecastinHAR',forecast\_df\_HAROA)

HARet\_mHAROAe(crimes\_HAROA\_forecast.TOTAL[10:13],forecast\_HAROA)

i=13

for j in ranHARe(len(forecast\_df\_HAROA)):

crimes\_HAROA\_forecast.loc[i] = crimes\_HAROA\_forecast.YEAR[i-1]+1

crimes\_HAROA\_forecast.TOTAL[i] = forecast\_df\_HAROA[j]

i = i+1

figure = HARo.Figure()

figure.add\_trace(HARo.Scatter(x= crimes\_HAROA\_forecast["YEAR"], y= crimes\_HAROA\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

figure.add\_trace(HARo.Scatter(x= crimes\_HAROA\_forecast["YEAR"][13:], y= crimes\_HAROA\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_HIM\_df = crimes\_total[crimes\_total.STATE=='HIMACHAL PRADESH']

ts\_decompose = sm.tsa.seasonal\_decompose(np.array(crime\_HIMOA\_df['TOTAL']),

model = "multiplicative", freq=2)

ts\_plot = ts\_decompose.plot()

crimes\_HIMOA\_forecast = crime\_HIMOA\_df.copy()

crimes\_HIMOA\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_HIMOA\_forecast = crimes\_HIMOA\_forecast.reset\_index(drop = True)

plot(crimes\_HIMOA\_forecast)

HIMOA\_arim = ARIMA(crimes\_HIMOA\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

HIMOA\_moHIMOA = HIMOA\_arim.fit()

HIMOA\_moHIMOA.summary2()

forecast\_HIMOA = HIMOA\_moHIMOA.predict(10,12)

print('prediction',forecast\_HIMOA)

forecast\_df\_HIMOA, stderr, ci = HIMOA\_moHIMOA.forecast(steps=7)

print('forecastinHIM',forecast\_df\_HIMOA)

HIMet\_mHIMOAe(crimes\_HIMOA\_forecast.TOTAL[10:13],forecast\_HIMOA)

i=13

for j in ranHIMe(len(forecast\_df\_HIMOA)):

crimes\_HIMOA\_forecast.loc[i] = crimes\_HIMOA\_forecast.YEAR[i-1]+1

crimes\_HIMOA\_forecast.TOTAL[i] = forecast\_df\_HIMOA[j]

i = i+1

figure = HIMo.Figure()

figure.add\_trace(HIMo.Scatter(x= crimes\_HIMOA\_forecast["YEAR"], y= crimes\_HIMOA\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

figure.add\_trace(HIMo.Scatter(x= crimes\_HIMOA\_forecast["YEAR"][13:], y= crimes\_HIMOA\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_JK\_df = crimes\_total[crimes\_total.STATE=='JAMMU KASHMIR']

ts\_decompose = sm.tsa.seasonal\_decompose(np.array(crime\_JKOA\_df['TOTAL']),

model = "multiplicative", freq=2)

ts\_plot = ts\_decompose.plot()

crimes\_JKOA\_forecast = crime\_JKOA\_df.copy()

crimes\_JKOA\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_JKOA\_forecast = crimes\_JKOA\_forecast.reset\_index(drop = True)

plot(crimes\_JKOA\_forecast)

JKOA\_arim = ARIMA(crimes\_JKOA\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

JKOA\_moJKOA = JKOA\_arim.fit()

JKOA\_moJKOA.summary2()

forecast\_JKOA = JKOA\_moJKOA.predict(10,12)

print('prediction',forecast\_JKOA)

forecast\_df\_JKOA, stderr, ci = JKOA\_moJKOA.forecast(steps=7)

print('forecastinJK',forecast\_df\_JKOA)

JKet\_mJKOAe(crimes\_JKOA\_forecast.TOTAL[10:13],forecast\_JKOA)

i=13

for j in range(len(forecast\_df\_JKOA)):

crimes\_JKOA\_forecast.loc[i] = crimes\_JKOA\_forecast.YEAR[i-1]+1

crimes\_JKOA\_forecast.TOTAL[i] = forecast\_df\_JKOA[j]

i = i+1

figure = JKo.Figure()

figure.add\_trace(JKo.Scatter(x= crimes\_JKOA\_forecast["YEAR"], y= crimes\_JKOA\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

figure.add\_trace(JKo.Scatter(x= crimes\_JKOA\_forecast["YEAR"][13:], y= crimes\_JKOA\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_KAR\_df = crimes\_total[crimes\_total.STATE=='KARNATAKA']

ts\_decompose = sm.tsa.seasonal\_decompose(np.array(crime\_KAROA\_df['TOTAL']),

model = "multiplicative", freq=2)

ts\_plot = ts\_decompose.plot()

crimes\_KAROA\_forecast = crime\_KAROA\_df.copy()

crimes\_KAROA\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_KAROA\_forecast = crimes\_KAROA\_forecast.reset\_index(drop = True)

plot(crimes\_KAROA\_forecast)

KAROA\_arim = ARIMA(crimes\_KAROA\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

KAROA\_moKAROA = KAROA\_arim.fit()

KAROA\_moKAROA.summary2()

forecast\_KAROA = KAROA\_moKAROA.predict(10,12)

print('prediction',forecast\_KAROA)

forecast\_df\_KAROA, stderr, ci = KAROA\_moKAROA.forecast(steps=7)

print('forecastinKAR',forecast\_df\_KAROA)

KARet\_mKAROAe(crimes\_KAROA\_forecast.TOTAL[10:13],forecast\_KAROA)

i=13

for j in range(len(forecast\_df\_KAROA)):

crimes\_KAROA\_forecast.loc[i] = crimes\_KAROA\_forecast.YEAR[i-1]+1

crimes\_KAROA\_forecast.TOTAL[i] = forecast\_df\_KAROA[j]

i = i+1

figure = go.Figure()

figure.add\_trace(KARo.Scatter(x= crimes\_KAROA\_forecast["YEAR"], y= crimes\_KAROA\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

figure.add\_trace(KARo.Scatter(x= crimes\_KAROA\_forecast["YEAR"][13:], y= crimes\_KAROA\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_KER\_df = crimes\_total[crimes\_total.STATE=='KERELA']

ts\_decompose = sm.tsa.seasonal\_decompose(np.array(crime\_KEROA\_df['TOTAL']),

model = "multiplicative", freq=2)

ts\_plot = ts\_decompose.plot()

crimes\_KEROA\_forecast = crime\_KEROA\_df.copy()

crimes\_KEROA\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_KEROA\_forecast = crimes\_KEROA\_forecast.reset\_index(drop = True)

plot(crimes\_KEROA\_forecast)

KEROA\_arim = ARIMA(crimes\_KEROA\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

KEROA\_moKEROA = KEROA\_arim.fit()

KEROA\_moKEROA.summary2()

forecast\_KEROA = KEROA\_moKEROA.predict(10,12)

print('prediction',forecast\_KEROA)

forecast\_df\_KEROA, stderr, ci = KEROA\_moKEROA.forecast(steps=7)

print('forecastinKER',forecast\_df\_KEROA)

KERet\_mKEROAe(crimes\_KEROA\_forecast.TOTAL[10:13],forecast\_KEROA)

i=13

for j in range(len(forecast\_df\_KEROA)):

crimes\_KEROA\_forecast.loc[i] = crimes\_KEROA\_forecast.YEAR[i-1]+1

crimes\_KEROA\_forecast.TOTAL[i] = forecast\_df\_KEROA[j]

i = i+1

figure = go.Figure()

figure.add\_trace(KERo.Scatter(x= crimes\_KEROA\_forecast["YEAR"], y= crimes\_KEROA\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

figure.add\_trace(KERo.Scatter(x= crimes\_KEROA\_forecast["YEAR"][13:], y= crimes\_KEROA\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_MP\_df = crimes\_total[crimes\_total.STATE=='MADHYA PRADESH']

ts\_decompose = sm.tsa.seasonal\_decompose(np.array(crime\_MPOA\_df['TOTAL']),

model = "multiplicative", freq=2)

ts\_plot = ts\_decompose.plot()

crimes\_MPOA\_forecast = crime\_MPOA\_df.copy()

crimes\_MPOA\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_MPOA\_forecast = crimes\_MPOA\_forecast.reset\_index(drop = True)

plot(crimes\_MPOA\_forecast)

MPOA\_arim = ARIMA(crimes\_MPOA\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

MPOA\_moMPOA = MPOA\_arim.fit()

MPOA\_moMPOA.summary2()

forecast\_MPOA = MPOA\_moMPOA.predict(10,12)

print('prediction',forecast\_MPOA)

forecast\_df\_MPOA, stderr, ci = MPOA\_moMPOA.forecast(steps=7)

print('forecastinMP',forecast\_df\_MPOA)

MPet\_mMPOAe(crimes\_MPOA\_forecast.TOTAL[10:13],forecast\_MPOA)

i=13

for j in range(len(forecast\_df\_MPOA)):

crimes\_MPOA\_forecast.loc[i] = crimes\_MPOA\_forecast.YEAR[i-1]+1

crimes\_MPOA\_forecast.TOTAL[i] = forecast\_df\_MPOA[j]

i = i+1

figure = go.Figure()

figure.add\_trace(MPo.Scatter(x= crimes\_MPOA\_forecast["YEAR"], y= crimes\_MPOA\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

figure.add\_trace(MPo.Scatter(x= crimes\_MPOA\_forecast["YEAR"][13:], y= crimes\_MPOA\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_MAHz\_df = crimes\_total[crimes\_total.STATE=='MAHARASTRA']

ts\_decoMAHose = sm.tsa.seasonal\_decoMAHose(np.array(crime\_MAHOA\_df['TOTAL']),

model = "multiplicative", freq=2)

ts\_plot = ts\_decoMAHose.plot()

crimes\_MAHOA\_forecast = crime\_MAHOA\_df.copy()

crimes\_MAHOA\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_MAHOA\_forecast = crimes\_MAHOA\_forecast.reset\_index(drop = True)

plot(crimes\_MAHOA\_forecast)

MAHOA\_arim = ARIMA(crimes\_MAHOA\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

MAHOA\_moMAHOA = MAHOA\_arim.fit()

MAHOA\_moMAHOA.summary2()

forecast\_MAHOA = MAHOA\_moMAHOA.predict(10,12)

print('prediction',forecast\_MAHOA)

forecast\_df\_MAHOA, stderr, ci = MAHOA\_moMAHOA.forecast(steps=7)

print('forecastinMAH',forecast\_df\_MAHOA)

MAHet\_mMAHOAe(crimes\_MAHOA\_forecast.TOTAL[10:13],forecast\_MAHOA)

i=13

for j in range(len(forecast\_df\_MAHOA)):

crimes\_MAHOA\_forecast.loc[i] = crimes\_MAHOA\_forecast.YEAR[i-1]+1

crimes\_MAHOA\_forecast.TOTAL[i] = forecast\_df\_MAHOA[j]

i = i+1

figure = go.Figure()

figure.add\_trace(MAHo.Scatter(x= crimes\_MAHOA\_forecast["YEAR"], y= crimes\_MAHOA\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

figure.add\_trace(MAHo.Scatter(x=crimes\_MAHOA\_forecast["YEAR"][13:], y= crimes\_MAHOA\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_MAN\_df = crimes\_total[crimes\_total.STATE=='MANIPUR']

ts\_decoMAHose = sm.tsa.seasonal\_decoMAHose(np.array(crime\_MAHOA\_df['TOTAL']),

model = "multiplicative", freq=2)

ts\_plot = ts\_decoMAHose.plot()

crimes\_MAHOA\_forecast = crime\_MAHOA\_df.copy()

crimes\_MAHOA\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_MAHOA\_forecast = crimes\_MAHOA\_forecast.reset\_index(drop = True)

plot(crimes\_MAHOA\_forecast)

MAHOA\_arim = ARIMA(crimes\_MAHOA\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

MAHOA\_moMAHOA = MAHOA\_arim.fit()

MAHOA\_moMAHOA.summary2()

forecast\_MAHOA = MAHOA\_moMAHOA.predict(10,12)

print('prediction',forecast\_MAHOA)

forecast\_df\_MAHOA, stderr, ci = MAHOA\_moMAHOA.forecast(steps=7)

print('forecastinMAH',forecast\_df\_MAHOA)

MAHet\_mMAHOAe(crimes\_MAHOA\_forecast.TOTAL[10:13],forecast\_MAHOA)

i=13

for j in range(len(forecast\_df\_MAHOA)):

crimes\_MAHOA\_forecast.loc[i] = crimes\_MAHOA\_forecast.YEAR[i-1]+1

crimes\_MAHOA\_forecast.TOTAL[i] = forecast\_df\_MAHOA[j]

i = i+1

figure = go.Figure()

figure.add\_trace(add.Scatter(x= crimes\_MAHOA\_forecast["YEAR"], y= crimes\_MAHOA\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

figure.add\_trace(MAHo.Scatter(x= crimes\_MAHOA\_forecast["YEAR"][13:], y= crimes\_MAHOA\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_MEG\_df = crimes\_total[crimes\_total.STATE=='MEGHALAYA']

ts\_decoMAHose = sm.tsa.seasonal\_decoMAHose(np.array(crime\_MAHOA\_df['TOTAL']),

model = "multiplicative", freq=2)

crimes\_MEG\_forecast = crime\_MEG\_df.copy()

crimes\_MEG\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_MEG\_forecast = crimes\_MEG\_forecast.reset\_index(drop = True)

plot(crimes\_MEG\_forecast)

MEG\_arim = ARIMA(crimes\_MEG\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

MEG\_moMEG = MEG\_arim.fit()

MEG\_moMEG.summary2()

forecast\_MEG = MEG\_moMEG.predict(10,12)

print('prediction',forecast\_MEG)

forecast\_df\_MEG, stderr, ci = MEG\_moMEG.forecast(steps=7)

print('forecasting',forecast\_df\_MEG)

get\_mape(crimes\_MEG\_forecast.TOTAL[10:13],forecast\_MEG)

i=13

for j in range(len(forecast\_df\_MEG)):

crimes\_MEG\_forecast.loc[i] = crimes\_MEG\_forecast.YEAR[i-1]+1

crimes\_MEG\_forecast.TOTAL[i] = forecast\_df\_MEG[j]

i = i+1

fig = go.Figure()

fig.add\_trace(go.Scatter(x= crimes\_MEG\_forecast["YEAR"], y= crimes\_MEG\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= crimes\_MEG\_forecast["YEAR"][13:], y= crimes\_MEG\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_MIZ\_df = crimes\_total[crimes\_total.STATE=='MIZORAM']

ts\_decoMAHose = sm.tsa.seasonal\_decoMAHose(np.array(crime\_MAHOA\_df['TOTAL']),

model = "multiplicative", freq=2)

crimes\_MIZ\_forecast = crime\_MIZ\_df.copy()

crimes\_MIZ\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_MIZ\_forecast = crimes\_MIZ\_forecast.reset\_index(drop = True)

plot(crimes\_MIZ\_forecast)

MIZ\_arim = ARIMA(crimes\_MIZ\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

MIZ\_moMIZ = MIZ\_arim.fit()

MIZ\_moMIZ.summary2()

forecast\_MIZ = MIZ\_moMIZ.predict(10,12)

print('prediction',forecast\_MIZ)

forecast\_df\_MIZ, stderr, ci = MIZ\_moMIZ.forecast(steps=7)

print('forecasting',forecast\_df\_MIZ)

get\_mMIZe(crimes\_MIZ\_forecast.TOTAL[10:13],forecast\_MIZ)

i=13

for j in range(len(forecast\_df\_MIZ)):

crimes\_MIZ\_forecast.loc[i] = crimes\_MIZ\_forecast.YEAR[i-1]+1

crimes\_MIZ\_forecast.TOTAL[i] = forecast\_df\_MIZ[j]

i = i+1

fig = go.Figure()

fig.add\_trace(go.Scatter(x= crimes\_MIZ\_forecast["YEAR"], y= crimes\_MIZ\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= crimes\_MIZ\_forecast["YEAR"][13:], y= crimes\_MIZ\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_NAG\_df = crimes\_total[crimes\_total.STATE=='NAGALAND']

ts\_decoMAHose = sm.tsa.seasonal\_decoMAHose(np.array(crime\_MAHOA\_df['TOTAL']),

model = "multiplicative", freq=2)

crimes\_NAG\_forecast = crime\_NAG\_df.copy()

crimes\_NAG\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_NAG\_forecast = crimes\_NAG\_forecast.reset\_index(drop = True)

plot(crimes\_NAG\_forecast)

NAG\_arim = ARIMA(crimes\_NAG\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

NAG\_moNAG = NAG\_arim.fit()

NAG\_moNAG.summary2()

forecast\_NAG = NAG\_moNAG.predict(10,12)

print('prediction',forecast\_NAG)

forecast\_df\_NAG, stderr, ci = NAG\_moNAG.forecast(steps=7)

print('forecasting',forecast\_df\_NAG)

get\_mNAGe(crimes\_NAG\_forecast.TOTAL[10:13],forecast\_NAG)

i=13

for j in range(len(forecast\_df\_NAG)):

crimes\_NAG\_forecast.loc[i] = crimes\_NAG\_forecast.YEAR[i-1]+1

crimes\_NAG\_forecast.TOTAL[i] = forecast\_df\_NAG[j]

i = i+1

fig = go.Figure()

fig.add\_trace(go.Scatter(x= crimes\_NAG\_forecast["YEAR"], y= crimes\_NAG\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= crimes\_NAG\_forecast["YEAR"][13:], y= crimes\_NAG\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_ODI\_df = crimes\_total[crimes\_total.STATE=='ODISHAa']

ts\_decoMAHose = sm.tsa.seasonal\_decoMAHose(np.array(crime\_MAHOA\_df['TOTAL']),

model = "multiplicative", freq=2)

crimes\_ODI\_forecast = crime\_ODI\_df.copy()

crimes\_ODI\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_ODI\_forecast = crimes\_ODI\_forecast.reset\_index(drop = True)

plot(crimes\_ODI\_forecast)

ODI\_arim = ARIMA(crimes\_ODI\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

ODI\_moODI = ODI\_arim.fit()

ODI\_moODI.summary2()

forecast\_ODI = ODI\_moODI.predict(10,12)

print('prediction',forecast\_ODI)

forecast\_df\_ODI, stderr, ci = ODI\_moODI.forecast(steps=7)

print('forecasting',forecast\_df\_ODI)

i=13

for j in range(len(forecast\_df\_ODI)):

crimes\_ODI\_forecast.loc[i] = crimes\_ODI\_forecast.YEAR[i-1]+1

crimes\_ODI\_forecast.TOTAL[i] = forecast\_df\_ODI[j]

i = i+1

fig = go.Figure()

fig.add\_trace(go.Scatter(x= crimes\_ODI\_forecast["YEAR"], y= crimes\_ODI\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= crimes\_ODI\_forecast["YEAR"][13:], y= crimes\_ODI\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_PUN\_df = crimes\_total[crimes\_total.STATE=='PUNJAB']

ts\_decoMAHose = sm.tsa.seasonal\_decoMAHose(np.array(crime\_MAHOA\_df['TOTAL']),

model = "multiplicative", freq=2)

crimes\_PUN\_forecast = crime\_PUN\_df.copy()

crimes\_PUN\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_PUN\_forecast = crimes\_PUN\_forecast.reset\_index(drop = True)

plot(crimes\_PUN\_forecast)

PUN\_arim = ARIMA(crimes\_PUN\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

PUN\_moPUN = PUN\_arim.fit()

PUN\_moPUN.summary2()

forecast\_PUN = PUN\_moPUN.predict(10,12)

print('prediction',forecast\_PUN)

forecast\_df\_PUN, stderr, ci = PUN\_moPUN.forecast(steps=7)

print('forecasting',forecast\_df\_PUN)

get\_mPUNe(crimes\_PUN\_forecast.TOTAL[10:13],forecast\_PUN)

i=13

for j in range(len(forecast\_df\_PUN)):

crimes\_PUN\_forecast.loc[i] = crimes\_PUN\_forecast.YEAR[i-1]+1

crimes\_PUN\_forecast.TOTAL[i] = forecast\_df\_PUN[j]

i = i+1

fig = go.Figure()

fig.add\_trace(go.Scatter(x= crimes\_PUN\_forecast["YEAR"], y= crimes\_PUN\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= crimes\_PUN\_forecast["YEAR"][13:], y= crimes\_PUN\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_RAJ\_df = crimes\_total[crimes\_total.STATE=='RAJASTHAN']

ts\_decoMAHose = sm.tsa.seasonal\_decoMAHose(np.array(crime\_MAHOA\_df['TOTAL']),

model = "multiplicative", freq=2)

crimes\_RAJ\_forecast = crime\_RAJ\_df.copy()

crimes\_RAJ\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_RAJ\_forecast = crimes\_RAJ\_forecast.reset\_index(drop = True)

plot(crimes\_RAJ\_forecast)

RAJ\_arim = ARIMA(crimes\_RAJ\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

RAJ\_moRAJ = RAJ\_arim.fit()

RAJ\_moRAJ.summary2()

forecast\_RAJ = RAJ\_moRAJ.predict(10,12)

print('prediction',forecast\_RAJ)

forecast\_df\_RAJ, stderr, ci = RAJ\_moRAJ.forecast(steps=7)

print('forecasting',forecast\_df\_RAJ)

get\_mRAJe(crimes\_RAJ\_forecast.TOTAL[10:13],forecast\_RAJ)

i=13

for j in range(len(forecast\_df\_RAJ)):

crimes\_RAJ\_forecast.loc[i] = crimes\_RAJ\_forecast.YEAR[i-1]+1

crimes\_RAJ\_forecast.TOTAL[i] = forecast\_df\_RAJ[j]

i = i+1

fig = go.Figure()

fig.add\_trace(go.Scatter(x= crimes\_RAJ\_forecast["YEAR"], y= crimes\_RAJ\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= crimes\_RAJ\_forecast["YEAR"][13:], y= crimes\_RAJ\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_SIK\_df = crimes\_total[crimes\_total.STATE=='SIKKIM']

ts\_decoMAHose = sm.tsa.seasonal\_decoMAHose(np.array(crime\_MAHOA\_df['TOTAL']),

model = "multiplicative", freq=2)

crimes\_SIK\_forecast = crime\_SIK\_df.copy()

crimes\_SIK\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_SIK\_forecast = crimes\_SIK\_forecast.reset\_index(drop = True)

plot(crimes\_SIK\_forecast)

SIK\_arim = ARIMA(crimes\_SIK\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

SIK\_moSIK = SIK\_arim.fit()

SIK\_moSIK.summary2()

forecast\_SIK = SIK\_moSIK.predict(10,12)

print('prediction',forecast\_SIK)

forecast\_df\_SIK, stderr, ci = SIK\_moSIK.forecast(steps=7)

print('forecasting',forecast\_df\_SIK)

get\_mSIKe(crimes\_SIK\_forecast.TOTAL[10:13],forecast\_SIK)

i=13

for j in range(len(forecast\_df\_SIK)):

crimes\_SIK\_forecast.loc[i] = crimes\_SIK\_forecast.YEAR[i-1]+1

crimes\_SIK\_forecast.TOTAL[i] = forecast\_df\_SIK[j]

i = i+1

fig = go.Figure()

fig.add\_trace(go.Scatter(x= crimes\_SIK\_forecast["YEAR"], y= crimes\_SIK\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= crimes\_SIK\_forecast["YEAR"][13:], y= crimes\_SIK\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_TN\_df = crimes\_total[crimes\_total.STATE=='TAMIL NADU']

ts\_decoMAHose = sm.tsa.seasonal\_decoMAHose(np.array(crime\_MAHOA\_df['TOTAL']),

model = "multiplicative", freq=2)

crimes\_TN\_forecast = crime\_TN\_df.copy()

crimes\_TN\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_TN\_forecast = crimes\_TN\_forecast.reset\_index(drop = True)

plot(crimes\_TN\_forecast)

TN\_arim = ARIMA(crimes\_TN\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

TN\_moTN = TN\_arim.fit()

TN\_moTN.summary2()

forecast\_TN = TN\_moTN.predict(10,12)

print('prediction',forecast\_TN)

forecast\_df\_TN, stderr, ci = TN\_moTN.forecast(steps=7)

print('forecasting',forecast\_df\_TN)

get\_mape(crimes\_TN\_forecast.TOTAL[10:13],forecast\_TN)

i=13

for j in range(len(forecast\_df\_TN)):

crimes\_TN\_forecast.loc[i] = crimes\_TN\_forecast.YEAR[i-1]+1

crimes\_TN\_forecast.TOTAL[i] = forecast\_df\_TN[j]

i = i+1

fig = go.Figure()

fig.add\_trace(go.Scatter(x= crimes\_TN\_forecast["YEAR"], y= crimes\_TN\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= crimes\_TN\_forecast["YEAR"][13:], y= crimes\_TN\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_UP\_df = crimes\_total[crimes\_total.STATE=='UTTAR PRADESH']

ts\_decoMAHose = sm.tsa.seasonal\_decoMAHose(np.array(crime\_MAHOA\_df['TOTAL']),

model = "multiplicative", freq=2)

crimes\_UP\_forecast = crime\_UP\_df.copy()

crimes\_UP\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_UP\_forecast = crimes\_UP\_forecast.reset\_index(drop = True)

plot(crimes\_UP\_forecast)

UP\_arim = ARIMA(crimes\_UP\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

UP\_moUP = UP\_arim.fit()

UP\_moUP.summary2()

forecast\_UP = UP\_moUP.predict(10,12)

print('prediction',forecast\_UP)

forecast\_df\_UP, stderr, ci = UP\_moUP.forecast(steps=7)

print('forecasting',forecast\_df\_UP)

get\_mape(crimes\_UP\_forecast.TOTAL[10:13],forecast\_UP)

i=13

for j in range(len(forecast\_df\_UP)):

crimes\_UP\_forecast.loc[i] = crimes\_UP\_forecast.YEAR[i-1]+1

crimes\_UP\_forecast.TOTAL[i] = forecast\_df\_UP[j]

i = i+1

fig = go.Figure()

fig.add\_trace(go.Scatter(x= crimes\_UP\_forecast["YEAR"], y= crimes\_UP\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= crimes\_UP\_forecast["YEAR"][13:], y= crimes\_UP\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_UK\_df = crimes\_total[crimes\_total.STATE=='UTTRAKHAND']

ts\_decoMAHose = sm.tsa.seasonal\_decoMAHose(np.array(crime\_MAHOA\_df['TOTAL']),

model = "multiplicative", freq=2)

crimes\_UK\_forecast = crime\_UK\_df.copy()

crimes\_UK\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_UK\_forecast = crimes\_UK\_forecast.reset\_index(drop = True)

plot(crimes\_UK\_forecast)

UK\_arim = ARIMA(crimes\_UK\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

UK\_moUK = UK\_arim.fit()

UK\_moUK.summary2()

forecast\_UK = UK\_moUK.predict(10,12)

print('prediction',forecast\_UK)

forecast\_df\_UK, stderr, ci = UK\_moUK.forecast(steps=7)

print('forecasting',forecast\_df\_UK)

get\_mape(crimes\_UK\_forecast.TOTAL[10:13],forecast\_UK)

=13

for j in range(len(forecast\_df\_UK)):

crimes\_UK\_forecast.loc[i] = crimes\_UK\_forecast.YEAR[i-1]+1

crimes\_UK\_forecast.TOTAL[i] = forecast\_df\_UK[j]

i = i+1

fig = go.Figure()

fig.add\_trace(go.Scatter(x= crimes\_UK\_forecast["YEAR"], y= crimes\_UK\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= crimes\_UK\_forecast["YEAR"][13:], y= crimes\_UK\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_WB\_df = crimes\_total[crimes\_total.STATE=='WEST BENGAL']

ts\_decoMAHose = sm.tsa.seasonal\_decoMAHose(np.array(crime\_MAHOA\_df['TOTAL']),

model = "multiplicative", freq=2)

crimes\_WB\_forecast = crime\_WB\_df.copy()

crimes\_WB\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_WB\_forecast = crimes\_WB\_forecast.reset\_index(drop = True)

plot(crimes\_WB\_forecast)

WB\_arim = ARIMA(crimes\_WB\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

WB\_moWB = WB\_arim.fit()

WB\_moWB.summary2()

forecast\_WB = WB\_moWB.predict(10,12)

print('prediction',forecast\_WB)

forecast\_df\_WB, stderr, ci = WB\_moWB.forecast(steps=7)

print('forecasting',forecast\_df\_WB)

get\_mape(crimes\_WB\_forecast.TOTAL[10:13],forecast\_WB)

i=13

for j in range(len(forecast\_df\_WB)):

crimes\_WB\_forecast.loc[i] = crimes\_WB\_forecast.YEAR[i-1]+1

crimes\_WB\_forecast.TOTAL[i] = forecast\_df\_WB[j]

i = i+1

fig = go.Figure()

fig.add\_trace(go.Scatter(x= crimes\_WB\_forecast["YEAR"], y= crimes\_WB\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= crimes\_WB\_forecast["YEAR"][13:], y= crimes\_WB\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_CHAN\_df = crimes\_total[crimes\_total.STATE=='CHANDIGARH']

ts\_decoMAHose = sm.tsa.seasonal\_decoMAHose(np.array(crime\_MAHOA\_df['TOTAL']),

model = "multiplicative", freq=2)

crimes\_CHAN\_forecast = crime\_CHAN\_df.copy()

crimes\_CHAN\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_CHAN\_forecast = crimes\_CHAN\_forecast.reset\_index(drop = True)

plot(crimes\_CHAN\_forecast)

CHAN\_arim = ARIMA(crimes\_CHAN\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

CHAN\_moCHAN = CHAN\_arim.fit()

CHAN\_moCHAN.summary2()

forecast\_CHAN = CHAN\_moCHAN.predict(10,12)

print('prediction',forecast\_CHAN)

forecast\_df\_CHAN, stderr, ci = CHAN\_moCHAN.forecast(steps=7)

print('forecasting',forecast\_df\_CHAN)

get\_mape(crimes\_CHAN\_forecast.TOTAL[10:13],forecast\_CHAN)

i=13

for j in range(len(forecast\_df\_CHAN)):

crimes\_CHAN\_forecast.loc[i] = crimes\_CHAN\_forecast.YEAR[i-1]+1

crimes\_CHAN\_forecast.TOTAL[i] = forecast\_df\_CHAN[j]

i = i+1

fig = go.Figure()

fig.add\_trace(go.Scatter(x= crimes\_CHAN\_forecast["YEAR"], y= crimes\_CHAN\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= crimes\_CHAN\_forecast["YEAR"][13:], y= crimes\_CHAN\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_DNH\_df = crimes\_total[crimes\_total.STATE=='D & N HAVELI']

ts\_decoMAHose = sm.tsa.seasonal\_decoMAHose(np.array(crime\_MAHOA\_df['TOTAL']),

model = "multiplicative", freq=2)

crimes\_DNH\_forecast = crime\_DNH\_df.copy()

crimes\_DNH\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_DNH\_forecast = crimes\_DNH\_forecast.reset\_index(drop = True)

plot(crimes\_DNH\_forecast)

DNH\_arim = ARIMA(crimes\_DNH\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

DNH\_moDNH = DNH\_arim.fit()

DNH\_moDNH.summary2()

forecast\_DNH = DNH\_moDNH.predict(10,12)

print('prediction',forecast\_DNH)

forecast\_df\_DNH, stderr, ci = DNH\_moDNH.forecast(steps=7)

print('forecasting',forecast\_df\_DNH)

get\_mape(crimes\_DNH\_forecast.TOTAL[10:13],forecast\_DNH)

i=13

for j in range(len(forecast\_df\_DNH)):

crimes\_DNH\_forecast.loc[i] = crimes\_DNH\_forecast.YEAR[i-1]+1

crimes\_DNH\_forecast.TOTAL[i] = forecast\_df\_DNH[j]

i = i+1

fig = go.Figure()

fig.add\_trace(go.Scatter(x= crimes\_DNH\_forecast["YEAR"], y= crimes\_DNH\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= crimes\_DNH\_forecast["YEAR"][13:], y= crimes\_DNH\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_DD\_df = crimes\_total[crimes\_total.STATE=='Daman&Diu']

ts\_decoMAHose = sm.tsa.seasonal\_decoMAHose(np.array(crime\_MAHOA\_df['TOTAL']),

model = "multiplicative", freq=2)

crimes\_DD\_forecast = crime\_DD\_df.copy()

crimes\_DD\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_DD\_forecast = crimes\_DD\_forecast.reset\_index(drop = True)

plot(crimes\_DD\_forecast)

DD\_arim = ARIMA(crimes\_DD\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

DD\_moDD = DD\_arim.fit()

DD\_moDD.summary2()

forecast\_DD = DD\_moDD.predict(10,12)

print('prediction',forecast\_DD)

forecast\_df\_DD, stderr, ci = DD\_moDD.forecast(steps=7)

print('forecasting',forecast\_df\_DD)

get\_mape(crimes\_DD\_forecast.TOTAL[10:13],forecast\_DD)

i=13

for j in range(len(forecast\_df\_DD)):

crimes\_DD\_forecast.loc[i] = crimes\_DD\_forecast.YEAR[i-1]+1

crimes\_DD\_forecast.TOTAL[i] = forecast\_df\_DD[j]

i = i+1

fig = go.Figure()

fig.add\_trace(go.Scatter(x= crimes\_DD\_forecast["YEAR"], y= crimes\_DD\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= crimes\_DD\_forecast["YEAR"][13:], y= crimes\_DD\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_DEL\_df = crimes\_total[crimes\_total.STATE=='DELHI UT']

ts\_decoMAHose = sm.tsa.seasonal\_decoMAHose(np.array(crime\_MAHOA\_df['TOTAL']),

model = "multiplicative", freq=2)

crimes\_DEL\_forecast = crime\_DEL\_df.copy()

crimes\_DEL\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_DEL\_forecast = crimes\_DEL\_forecast.reset\_index(drop = True)

plot(crimes\_DEL\_forecast)

DEL\_arim = ARIMA(crimes\_DEL\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

DEL\_moDEL = DEL\_arim.fit()

DEL\_moDEL.summary2()

forecast\_DEL = DEL\_moDEL.predict(10,12)

print('prediction',forecast\_DEL)

forecast\_df\_DEL, stderr, ci = DEL\_moDEL.forecast(steps=7)

print('forecasting',forecast\_df\_DEL)

get\_mape(crimes\_DEL\_forecast.TOTAL[10:13],forecast\_DEL)

i=13

for j in range(len(forecast\_df\_DEL)):

crimes\_DEL\_forecast.loc[i] = crimes\_DEL\_forecast.YEAR[i-1]+1

crimes\_DEL\_forecast.TOTAL[i] = forecast\_df\_DEL[j]

i = i+1

fig = go.Figure()

fig.aDEL\_trace(go.Scatter(x= crimes\_DEL\_forecast["YEAR"], y= crimes\_DEL\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.aDEL\_trace(go.Scatter(x= crimes\_DEL\_forecast["YEAR"][13:], y= crimes\_DEL\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_PUD\_df = crimes\_total[crimes\_total.STATE=='PUDUCHERRY']

ts\_decoMAHose = sm.tsa.seasonal\_decoMAHose(np.array(crime\_MAHOA\_df['TOTAL']),

moPUD = "multiplicative", freq=2)

crimes\_PUD\_forecast = crime\_PUD\_df.copy()

crimes\_PUD\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_PUD\_forecast = crimes\_PUD\_forecast.reset\_index(drop = True)

plot(crimes\_PUD\_forecast)

PUD\_arim = ARIMA(crimes\_PUD\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

PUD\_moPUD = PUD\_arim.fit()

PUD\_moPUD.summary2()

forecast\_PUD = PUD\_moPUD.predict(10,12)

print('prediction',forecast\_PUD)

forecast\_df\_PUD, stderr, ci = PUD\_moPUD.forecast(steps=7)

print('forecasting',forecast\_df\_PUD)

get\_mape(crimes\_PUD\_forecast.TOTAL[10:13],forecast\_PUD)

i=13

for j in range(len(forecast\_df\_PUD)):

crimes\_PUD\_forecast.loc[i] = crimes\_PUD\_forecast.YEAR[i-1]+1

crimes\_PUD\_forecast.TOTAL[i] = forecast\_df\_PUD[j]

i = i+1

fig = go.Figure()

fig.aPUD\_trace(go.Scatter(x= crimes\_PUD\_forecast["YEAR"], y= crimes\_PUD\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.aPUD\_trace(go.Scatter(x= crimes\_PUD\_forecast["YEAR"][13:], y= crimes\_PUD\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crime\_LAK\_df = crimes\_total[crimes\_total.STATE=='LAKSHWADEEP']

ts\_decoMAHose = sm.tsa.seasonal\_decoMAHose(np.array(crime\_MAHOA\_df['TOTAL']),

moLAK = "multiplicative", freq=2)

crimes\_LAK\_forecast = crime\_LAK\_df.copy()

crimes\_LAK\_forecast.drop(['STATE'], axis = 1, inplace = True)

crimes\_LAK\_forecast = crimes\_LAK\_forecast.reset\_index(drop = True)

plot(crimes\_LAK\_forecast)

LAK\_arim = ARIMA(crimes\_LAK\_forecast.TOTAL[0:10].astype(np.float64).as\_matrix(), order=(1,0,0))

LAK\_moLAK = LAK\_arim.fit()

LAK\_moLAK.summary2()

forecast\_LAK = LAK\_moLAK.predict(10,12)

print('prediction',forecast\_LAK)

forecast\_df\_LAK, stderr, ci = LAK\_moLAK.forecast(steps=7)

print('forecasting',forecast\_df\_LAK)

get\_mape(crimes\_LAK\_forecast.TOTAL[10:13],forecast\_LAK)

i=13

for j in range(len(forecast\_df\_LAK)):

crimes\_LAK\_forecast.loc[i] = crimes\_LAK\_forecast.YEAR[i-1]+1

crimes\_LAK\_forecast.TOTAL[i] = forecast\_df\_LAK[j]

i = i+1

fig = go.Figure()

fig.aLAK\_trace(go.Scatter(x= crimes\_LAK\_forecast["YEAR"], y= crimes\_LAK\_forecast['TOTAL'],

name = "Actual",line=dict(color='firebrick', width=4)))

fig.aLAK\_trace(go.Scatter(x= crimes\_LAK\_forecast["YEAR"][13:], y= crimes\_LAK\_forecast['TOTAL'][13:],

name = "Estimated",line=dict(color='blue', width=4)))

crimes\_UP= crimes\_df[crimes\_df.STATE=='UTTARPRADESH']

crimes\_UP.shape

DIST\_UP = crimes\_UP[crimes\_UP.DISAPCT=='TOTAL']

DIST\_UP= crimes\_UP.groAPby(['YEAR']).agg({'TOTAL\_THEFT':sum,'HURT':sum,'KIDNAPPING':sum,'CRUELTY\_BY\_HUSBAND':sum,

'BURGLARY':sum, 'MURDER':sum,'DEATH\_BY\_NEGLIGENCE':sum,

'CHEATING':sum, 'RIOTS':sum, 'ASSAULT\_WOMEN\_MODESTY':sum}).reset\_index()

fig = go.Figure()

fig.add\_trace(go.Scatter(x= DIST\_UP["YEAR"], y= DIST\_UP['TOTAL\_THEFT'],

name = "THEFT",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_UP["YEAR"], y= DIST\_UP['HURT'],

name='HURT',line=dict(color='pink', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_UP["YEAR"], y= DIST\_UP['KIDNAPPING'],

name='KIDNAPPING & ABDUCTION',line=dict(color='green', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_UP["YEAR"], y= DIST\_UP["CRUELTY\_BY\_HUSBAND"],

name='CRUELTY BY HUSBAND',line=dict(color='blue', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_UP["YEAR"], y= DIST\_UP['BURGLARY'],

name='BURGLARY',line=dict(color='yellow', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_UP["YEAR"], y= DIST\_UP['MURDER'],

name='MURDER',line=dict(color='royalblue', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_UP["YEAR"], y= DIST\_UP['DEATH\_BY\_NEGLIGENCE'],

name='DEATH\_BY\_NEGLIGENCE',line=dict(color='black', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_UP["YEAR"], y= DIST\_UP['RIOTS'],

name='RIOTS',line=dict(color='red', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_UP["YEAR"], y= DIST\_UP['CHEATING'],

name='CHEATING',line=dict(color='orange', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_UP["YEAR"], y= DIST\_UP['ASSAULT\_WOMEN\_MODESTY'],

name='ASSAULT\_WOMEN\_MODESTY',line=dict(color='purple', width=4)))

fig.APdate\_layout(uniformtext\_minsize= 20)

fig.show()

crimes\_UP\_df = crimes\_UP[crimes\_UP.DISAPCT!="TOTAL"]

crimes\_UP\_df = crimes\_UP\_df.groAPby('DISAPCT').agg({"TOTAL\_THEFT":sum, 'HURT':sum}).reset\_index()

crimes\_UP\_df

AP\_crimes = crimes\_UP\_df[(crimes\_UP\_df.TOTAL\_THEFT>2000) & (crimes\_UP\_df.HURT>5000)].reset\_index()

fig = go.Figure()

fig.add\_trace(go.Bar(

x=AP\_crimes["DISAPCT"],

y=AP\_crimes["TOTAL\_THEFT"],

name='THEFT',

marAP\_color='red'

))

fig.add\_trace(go.Bar(

x=AP\_crimes["DISAPCT"],

y=AP\_crimes["HURT"],

name='HURT',

marAP\_color='Purple'

))

fig.APdate\_layout(barmode='groAP', xaxis\_tickangle=-45)

fig.show()

crimes\_UP\_clus = crimes\_UP.copy()

crimes\_UP\_clus.drop(["STATE", 'TOTAL', 'YEAR', 'ATTEAPT TO MURDER',

'CULPABLE HOMICIDE NOT AMOUNTING TO MURDER', 'RAPE', 'CUSTAPAL RAPE',

'OTHER RAPE', 'KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS',

'KIDNAPPING AND ABDUCTION OF OTHERS', 'DACOITY',

'PREPARATION AND ASSEMBLY FOR DACOITY', 'ROBBERY', 'THEFT',

'AUTO THEFT', 'OTHER THEFT', 'COUNTERFIETING', 'ARSON', 'DOWRY DEATHS', 'INSULT TO MODESTY OF WOMEN', 'IAPORTATION OF GIRLS FROM FOREIGN COUNAPES'], axis=1 ,inplace = True)

crimes\_UP\_main = crimes\_UP\_clus[crimes\_UP\_clus.DISAPCT!="TOTAL"]

crimes\_KMe\_UP = crimes\_UP\_main.groAPby("DISAPCT").agg({'TOTAL\_THEFT':sum,'HURT':sum,'KIDNAPPING':sum,'CRUELTY\_BY\_HUSBAND':sum,

'BURGLARY':sum, 'MURDER':sum,'DEATH\_BY\_NEGLIGENCE':sum,

'CHEATING':sum, 'RIOTS':sum, 'ASSAULT\_WOMEN\_MODESTY':sum, 'OTHER IPC CRIMES':sum}).reset\_index()

scaled = StandardScaler()

scaled\_UP = scaled.fit\_transform(crimes\_KMe\_UP[['TOTAL\_THEFT','HURT','KIDNAPPING','CRUELTY\_BY\_HUSBAND',

'BURGLARY', 'MURDER','DEATH\_BY\_NEGLIGENCE',

'CHEATING', 'RIOTS', 'ASSAULT\_WOMEN\_MODESTY', 'OTHER IPC CRIMES']])

new\_UP = range(1,20)

rate\_UP = []

for i in new\_UP:

clustering = KMeans(i)

clustering.fit(scaled\_UP)

rate\_UP.append(clustering.inertia\_)

plt.figure(figsize=(6,4))

plt.plot(new\_UP, rate\_UP, marAP="o")

K=4

cluster\_UP = KMeans(4, random\_state = 42)

cluster\_UP.fit(scaled\_UP)

crimes\_KMe\_UP['Cluster\_id'] = cluster\_UP.labels\_

crimes\_MP= crimes\_df[crimes\_df.STATE=='MADHYA PRADESH']

crimes\_MP.shape

DIST\_MP = crimes\_MP[crimes\_MP.DISAPCT=='TOTAL']

DIST\_MP= crimes\_MP.groAPby(['YEAR']).agg({'TOTAL\_THEFT':sum,'HURT':sum,'KIDNAPPING':sum,'CRUELTY\_BY\_HUSBAND':sum,

'BURGLARY':sum, 'MURDER':sum,'DEATH\_BY\_NEGLIGENCE':sum,

'CHEATING':sum, 'RIOTS':sum, 'ASSAULT\_WOMEN\_MODESTY':sum}).reset\_index()

fig = go.Figure()

fig.add\_trace(go.Scatter(x= DIST\_MP["YEAR"], y= DIST\_MP['TOTAL\_THEFT'],

name = "THEFT",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_MP["YEAR"], y= DIST\_MP['HURT'],

name='HURT',line=dict(color='pink', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_MP["YEAR"], y= DIST\_MP['KIDNAPPING'],

name='KIDNAPPING & ABDUCTION',line=dict(color='green', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_MP["YEAR"], y= DIST\_MP["CRUELTY\_BY\_HUSBAND"],

name='CRUELTY BY HUSBAND',line=dict(color='blue', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_MP["YEAR"], y= DIST\_MP['BURGLARY'],

name='BURGLARY',line=dict(color='yellow', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_MP["YEAR"], y= DIST\_MP['MURDER'],

name='MURDER',line=dict(color='royalblue', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_MP["YEAR"], y= DIST\_MP['DEATH\_BY\_NEGLIGENCE'],

name='DEATH\_BY\_NEGLIGENCE',line=dict(color='black', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_MP["YEAR"], y= DIST\_MP['RIOTS'],

name='RIOTS',line=dict(color='red', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_MP["YEAR"], y= DIST\_MP['CHEATING'],

name='CHEATING',line=dict(color='orange', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_MP["YEAR"], y= DIST\_MP['ASSAULT\_WOMEN\_MODESTY'],

name='ASSAULT\_WOMEN\_MODESTY',line=dict(color='purple', width=4)))

fig.APdate\_layout(uniformtext\_minsize= 20)

fig.show()

crimes\_MP\_df = crimes\_MP[crimes\_MP.DISAPCT!="TOTAL"]

crimes\_MP\_df = crimes\_MP\_df.groAPby('DISAPCT').agg({"TOTAL\_THEFT":sum, 'HURT':sum}).reset\_index()

crimes\_MP\_df

AP\_crimes = crimes\_MP\_df[(crimes\_MP\_df.TOTAL\_THEFT>2000) & (crimes\_MP\_df.HURT>5000)].reset\_index()

fig = go.Figure()

fig.add\_trace(go.Bar(

x=AP\_crimes["DISAPCT"],

y=AP\_crimes["TOTAL\_THEFT"],

name='THEFT',

marAP\_color='red'

))

fig.add\_trace(go.Bar(

x=AP\_crimes["DISAPCT"],

y=AP\_crimes["HURT"],

name='HURT',

marAP\_color='Purple'

))

fig.APdate\_layout(barmode='groAP', xaxis\_tickangle=-45)

fig.show()

crimes\_MP\_clus = crimes\_MP.copy()

crimes\_MP\_clus.drop(["STATE", 'TOTAL', 'YEAR', 'ATTEAPT TO MURDER',

'CULPABLE HOMICIDE NOT AMOUNTING TO MURDER', 'RAPE', 'CUSTAPAL RAPE',

'OTHER RAPE', 'KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS',

'KIDNAPPING AND ABDUCTION OF OTHERS', 'DACOITY',

'PREPARATION AND ASSEMBLY FOR DACOITY', 'ROBBERY', 'THEFT',

'AUTO THEFT', 'OTHER THEFT', 'COUNTERFIETING', 'ARSON', 'DOWRY DEATHS', 'INSULT TO MODESTY OF WOMEN', 'IAPORTATION OF GIRLS FROM FOREIGN COUNAPES'], axis=1 ,inplace = True)

crimes\_MP\_main = crimes\_MP\_clus[crimes\_MP\_clus.DISAPCT!="TOTAL"]

crimes\_KMe\_MP = crimes\_MP\_main.groAPby("DISAPCT").agg({'TOTAL\_THEFT':sum,'HURT':sum,'KIDNAPPING':sum,'CRUELTY\_BY\_HUSBAND':sum,

'BURGLARY':sum, 'MURDER':sum,'DEATH\_BY\_NEGLIGENCE':sum,

'CHEATING':sum, 'RIOTS':sum, 'ASSAULT\_WOMEN\_MODESTY':sum, 'OTHER IPC CRIMES':sum}).reset\_index()

scaled = StandardScaler()

scaled\_MP = scaled.fit\_transform(crimes\_KMe\_MP[['TOTAL\_THEFT','HURT','KIDNAPPING','CRUELTY\_BY\_HUSBAND',

'BURGLARY', 'MURDER','DEATH\_BY\_NEGLIGENCE',

'CHEATING', 'RIOTS', 'ASSAULT\_WOMEN\_MODESTY', 'OTHER IPC CRIMES']])

new\_MP = range(1,20)

rate\_MP = []

for i in new\_MP:

clustering = KMeans(i)

clustering.fit(scaled\_MP)

rate\_MP.append(clustering.inertia\_)

plt.figure(figsize=(6,4))

plt.plot(new\_MP, rate\_MP, marAP="o")

K=4

cluster\_MP = KMeans(4, random\_state = 42)

cluster\_MP.fit(scaled\_MP)

crimes\_KMe\_MP['Cluster\_id'] = cluster\_MP.labels\_

crimes\_AP= crimes\_df[crimes\_df.STATE=='ANDRA PRADESH']

crimes\_AP.shape

DIST\_AP = crimes\_AP[crimes\_AP.DISAPCT=='TOTAL']

DIST\_AP= crimes\_AP.groAPby(['YEAR']).agg({'TOTAL\_THEFT':sum,'HURT':sum,'KIDNAPPING':sum,'CRUELTY\_BY\_HUSBAND':sum,

'BURGLARY':sum, 'MURDER':sum,'DEATH\_BY\_NEGLIGENCE':sum,

'CHEATING':sum, 'RIOTS':sum, 'ASSAULT\_WOMEN\_MODESTY':sum}).reset\_index()

fig = go.Figure()

fig.add\_trace(go.Scatter(x= DIST\_AP["YEAR"], y= DIST\_AP['TOTAL\_THEFT'],

name = "THEFT",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_AP["YEAR"], y= DIST\_AP['HURT'],

name='HURT',line=dict(color='pink', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_AP["YEAR"], y= DIST\_AP['KIDNAPPING'],

name='KIDNAPPING & ABDUCTION',line=dict(color='green', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_AP["YEAR"], y= DIST\_AP["CRUELTY\_BY\_HUSBAND"],

name='CRUELTY BY HUSBAND',line=dict(color='blue', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_AP["YEAR"], y= DIST\_AP['BURGLARY'],

name='BURGLARY',line=dict(color='yellow', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_AP["YEAR"], y= DIST\_AP['MURDER'],

name='MURDER',line=dict(color='royalblue', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_AP["YEAR"], y= DIST\_AP['DEATH\_BY\_NEGLIGENCE'],

name='DEATH\_BY\_NEGLIGENCE',line=dict(color='black', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_AP["YEAR"], y= DIST\_AP['RIOTS'],

name='RIOTS',line=dict(color='red', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_AP["YEAR"], y= DIST\_AP['CHEATING'],

name='CHEATING',line=dict(color='orange', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_AP["YEAR"], y= DIST\_AP['ASSAULT\_WOMEN\_MODESTY'],

name='ASSAULT\_WOMEN\_MODESTY',line=dict(color='purple', width=4)))

fig.APdate\_layout(uniformtext\_minsize= 20)

fig.show()

crimes\_AP\_df = crimes\_AP[crimes\_AP.DISAPCT!="TOTAL"]

crimes\_AP\_df = crimes\_AP\_df.groAPby('DISAPCT').agg({"TOTAL\_THEFT":sum, 'HURT':sum}).reset\_index()

crimes\_AP\_df

AP\_crimes = crimes\_AP\_df[(crimes\_AP\_df.TOTAL\_THEFT>2000) & (crimes\_AP\_df.HURT>5000)].reset\_index()

fig = go.Figure()

fig.add\_trace(go.Bar(

x=AP\_crimes["DISAPCT"],

y=AP\_crimes["TOTAL\_THEFT"],

name='THEFT',

marAP\_color='red'

))

fig.add\_trace(go.Bar(

x=AP\_crimes["DISAPCT"],

y=AP\_crimes["HURT"],

name='HURT',

marAP\_color='Purple'

))

fig.APdate\_layout(barmode='groAP', xaxis\_tickangle=-45)

fig.show()

crimes\_AP\_clus = crimes\_AP.copy()

crimes\_AP\_clus.drop(["STATE", 'TOTAL', 'YEAR', 'ATTEAPT TO MURDER',

'CULPABLE HOMICIDE NOT AMOUNTING TO MURDER', 'RAPE', 'CUSTAPAL RAPE',

'OTHER RAPE', 'KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS',

'KIDNAPPING AND ABDUCTION OF OTHERS', 'DACOITY',

'PREPARATION AND ASSEMBLY FOR DACOITY', 'ROBBERY', 'THEFT',

'AUTO THEFT', 'OTHER THEFT', 'COUNTERFIETING', 'ARSON', 'DOWRY DEATHS', 'INSULT TO MODESTY OF WOMEN', 'IAPORTATION OF GIRLS FROM FOREIGN COUNAPES'], axis=1 ,inplace = True)

crimes\_AP\_main = crimes\_AP\_clus[crimes\_AP\_clus.DISAPCT!="TOTAL"]

crimes\_KMe\_AP = crimes\_AP\_main.groAPby("DISAPCT").agg({'TOTAL\_THEFT':sum,'HURT':sum,'KIDNAPPING':sum,'CRUELTY\_BY\_HUSBAND':sum,

'BURGLARY':sum, 'MURDER':sum,'DEATH\_BY\_NEGLIGENCE':sum,

'CHEATING':sum, 'RIOTS':sum, 'ASSAULT\_WOMEN\_MODESTY':sum, 'OTHER IPC CRIMES':sum}).reset\_index()

scaled = StandardScaler()

scaled\_AP = scaled.fit\_transform(crimes\_KMe\_AP[['TOTAL\_THEFT','HURT','KIDNAPPING','CRUELTY\_BY\_HUSBAND',

'BURGLARY', 'MURDER','DEATH\_BY\_NEGLIGENCE',

'CHEATING', 'RIOTS', 'ASSAULT\_WOMEN\_MODESTY', 'OTHER IPC CRIMES']])

new\_AP = range(1,20)

rate\_AP = []

for i in new\_AP:

clustering = KMeans(i)

clustering.fit(scaled\_AP)

rate\_AP.append(clustering.inertia\_)

plt.figure(figsize=(6,4))

plt.plot(new\_AP, rate\_AP, marAP="o")

K=4

cluster\_AP = KMeans(4, random\_state = 42)

cluster\_AP.fit(scaled\_AP)

crimes\_KMe\_AP['Cluster\_id'] = cluster\_AP.labels\_

crimes\_MAH= crimes\_df[crimes\_df.STATE=='MAHARASTRA']

crimes\_MAH.shape

DIST\_MAH = crimes\_MAH[crimes\_MAH.DISAPCT=='TOTAL']

DIST\_MAH= crimes\_MAH.groAPby(['YEAR']).agg({'TOTAL\_THEFT':sum,'HURT':sum,'KIDNAPPING':sum,'CRUELTY\_BY\_HUSBAND':sum,

'BURGLARY':sum, 'MURDER':sum,'DEATH\_BY\_NEGLIGENCE':sum,

'CHEATING':sum, 'RIOTS':sum, 'ASSAULT\_WOMEN\_MODESTY':sum}).reset\_index()

fig = go.Figure()

fig.add\_trace(go.Scatter(x= DIST\_MAH["YEAR"], y= DIST\_MAH['TOTAL\_THEFT'],

name = "THEFT",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_MAH["YEAR"], y= DIST\_MAH['HURT'],

name='HURT',line=dict(color='pink', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_MAH["YEAR"], y= DIST\_MAH['KIDNAPPING'],

name='KIDNAPPING & ABDUCTION',line=dict(color='green', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_MAH["YEAR"], y= DIST\_MAH["CRUELTY\_BY\_HUSBAND"],

name='CRUELTY BY HUSBAND',line=dict(color='blue', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_MAH["YEAR"], y= DIST\_MAH['BURGLARY'],

name='BURGLARY',line=dict(color='yellow', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_MAH["YEAR"], y= DIST\_MAH['MURDER'],

name='MURDER',line=dict(color='royalblue', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_MAH["YEAR"], y= DIST\_MAH['DEATH\_BY\_NEGLIGENCE'],

name='DEATH\_BY\_NEGLIGENCE',line=dict(color='black', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_MAH["YEAR"], y= DIST\_MAH['RIOTS'],

name='RIOTS',line=dict(color='red', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_MAH["YEAR"], y= DIST\_MAH['CHEATING'],

name='CHEATING',line=dict(color='orange', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_MAH["YEAR"], y= DIST\_MAH['ASSAULT\_WOMEN\_MODESTY'],

name='ASSAULT\_WOMEN\_MODESTY',line=dict(color='purple', width=4)))

fig.APdate\_layout(uniformtext\_minsize= 20)

fig.show()

crimes\_MAH\_df = crimes\_MAH[crimes\_MAH.DISAPCT!="TOTAL"]

crimes\_MAH\_df = crimes\_MAH\_df.groAPby('DISAPCT').agg({"TOTAL\_THEFT":sum, 'HURT':sum}).reset\_index()

crimes\_MAH\_df

AP\_crimes = crimes\_MAH\_df[(crimes\_MAH\_df.TOTAL\_THEFT>2000) & (crimes\_MAH\_df.HURT>5000)].reset\_index()

fig = go.Figure()

fig.add\_trace(go.Bar(

x=AP\_crimes["DISAPCT"],

y=AP\_crimes["TOTAL\_THEFT"],

name='THEFT',

marAP\_color='red'

))

fig.add\_trace(go.Bar(

x=AP\_crimes["DISAPCT"],

y=AP\_crimes["HURT"],

name='HURT',

marAP\_color='Purple'

))

fig.APdate\_layout(barmode='groAP', xaxis\_tickangle=-45)

fig.show()

crimes\_MAH\_clus = crimes\_MAH.copy()

crimes\_MAH\_clus.drop(["STATE", 'TOTAL', 'YEAR', 'ATTEAPT TO MURDER',

'CULPABLE HOMICIDE NOT AMOUNTING TO MURDER', 'RAPE', 'CUSTAPAL RAPE',

'OTHER RAPE', 'KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS',

'KIDNAPPING AND ABDUCTION OF OTHERS', 'DACOITY',

'PREPARATION AND ASSEMBLY FOR DACOITY', 'ROBBERY', 'THEFT',

'AUTO THEFT', 'OTHER THEFT', 'COUNTERFIETING', 'ARSON', 'DOWRY DEATHS', 'INSULT TO MODESTY OF WOMEN', 'IAPORTATION OF GIRLS FROM FOREIGN COUNAPES'], axis=1 ,inplace = True)

crimes\_MAH\_main = crimes\_MAH\_clus[crimes\_MAH\_clus.DISAPCT!="TOTAL"]

crimes\_KMe\_MAH = crimes\_MAH\_main.groAPby("DISAPCT").agg({'TOTAL\_THEFT':sum,'HURT':sum,'KIDNAPPING':sum,'CRUELTY\_BY\_HUSBAND':sum,

'BURGLARY':sum, 'MURDER':sum,'DEATH\_BY\_NEGLIGENCE':sum,

'CHEATING':sum, 'RIOTS':sum, 'ASSAULT\_WOMEN\_MODESTY':sum, 'OTHER IPC CRIMES':sum}).reset\_index()

scaled = StandardScaler()

scaled\_MAH = scaled.fit\_transform(crimes\_KMe\_MAH[['TOTAL\_THEFT','HURT','KIDNAPPING','CRUELTY\_BY\_HUSBAND',

'BURGLARY', 'MURDER','DEATH\_BY\_NEGLIGENCE',

'CHEATING', 'RIOTS', 'ASSAULT\_WOMEN\_MODESTY', 'OTHER IPC CRIMES']])

new\_MAH = range(1,20)

rate\_MAH = []

for i in new\_MAH:

clustering = KMeans(i)

clustering.fit(scaled\_MAH)

rate\_MAH.append(clustering.inertia\_)

plt.figure(figsize=(6,4))

plt.plot(new\_MAH, rate\_MAH, marAP="o")

K=4

cluster\_MAH = KMeans(4, random\_state = 42)

cluster\_MAH.fit(scaled\_MAH)

crimes\_KMe\_MAH['Cluster\_id'] = cluster\_MAH.labels\_

crimes\_TN= crimes\_df[crimes\_df.STATE=='TAMIL NADU']

crimes\_TN.shape

DIST\_TN = crimes\_TN[crimes\_TN.DISAPCT=='TOTAL']

DIST\_TN= crimes\_TN.groAPby(['YEAR']).agg({'TOTAL\_THEFT':sum,'HURT':sum,'KIDNAPPING':sum,'CRUELTY\_BY\_HUSBAND':sum,

'BURGLARY':sum, 'MURDER':sum,'DEATH\_BY\_NEGLIGENCE':sum,

'CHEATING':sum, 'RIOTS':sum, 'ASSAULT\_WOMEN\_MODESTY':sum}).reset\_index()

fig = go.Figure()

fig.add\_trace(go.Scatter(x= DIST\_TN["YEAR"], y= DIST\_TN['TOTAL\_THEFT'],

name = "THEFT",line=dict(color='firebrick', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_TN["YEAR"], y= DIST\_TN['HURT'],

name='HURT',line=dict(color='pink', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_TN["YEAR"], y= DIST\_TN['KIDNAPPING'],

name='KIDNAPPING & ABDUCTION',line=dict(color='green', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_TN["YEAR"], y= DIST\_TN["CRUELTY\_BY\_HUSBAND"],

name='CRUELTY BY HUSBAND',line=dict(color='blue', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_TN["YEAR"], y= DIST\_TN['BURGLARY'],

name='BURGLARY',line=dict(color='yellow', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_TN["YEAR"], y= DIST\_TN['MURDER'],

name='MURDER',line=dict(color='royalblue', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_TN["YEAR"], y= DIST\_TN['DEATH\_BY\_NEGLIGENCE'],

name='DEATH\_BY\_NEGLIGENCE',line=dict(color='black', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_TN["YEAR"], y= DIST\_TN['RIOTS'],

name='RIOTS',line=dict(color='red', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_TN["YEAR"], y= DIST\_TN['CHEATING'],

name='CHEATING',line=dict(color='orange', width=4)))

fig.add\_trace(go.Scatter(x= DIST\_TN["YEAR"], y= DIST\_TN['ASSAULT\_WOMEN\_MODESTY'],

name='ASSAULT\_WOMEN\_MODESTY',line=dict(color='purple', width=4)))

fig.APdate\_layout(uniformtext\_minsize= 20)

fig.show()

crimes\_TN\_df = crimes\_TN[crimes\_TN.DISAPCT!="TOTAL"]

crimes\_TN\_df = crimes\_TN\_df.groAPby('DISAPCT').agg({"TOTAL\_THEFT":sum, 'HURT':sum}).reset\_index()

crimes\_TN\_df

AP\_crimes = crimes\_TN\_df[(crimes\_TN\_df.TOTAL\_THEFT>2000) & (crimes\_TN\_df.HURT>5000)].reset\_index()

fig = go.Figure()

fig.add\_trace(go.Bar(

x=AP\_crimes["DISAPCT"],

y=AP\_crimes["TOTAL\_THEFT"],

name='THEFT',

marAP\_color='red'

))

fig.add\_trace(go.Bar(

x=AP\_crimes["DISAPCT"],

y=AP\_crimes["HURT"],

name='HURT',

marAP\_color='Purple'

))

fig.APdate\_layout(barmode='groAP', xaxis\_tickangle=-45)

fig.show()

crimes\_TN\_clus = crimes\_TN.copy()

crimes\_TN\_clus.drop(["STATE", 'TOTAL', 'YEAR', 'ATTEAPT TO MURDER',

'CULPABLE HOMICIDE NOT AMOUNTING TO MURDER', 'RAPE', 'CUSTAPAL RAPE',

'OTHER RAPE', 'KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS',

'KIDNAPPING AND ABDUCTION OF OTHERS', 'DACOITY',

'PREPARATION AND ASSEMBLY FOR DACOITY', 'ROBBERY', 'THEFT',

'AUTO THEFT', 'OTHER THEFT', 'COUNTERFIETING', 'ARSON', 'DOWRY DEATHS', 'INSULT TO MODESTY OF WOMEN', 'IAPORTATION OF GIRLS FROM FOREIGN COUNAPES'], axis=1 ,inplace = True)

crimes\_TN\_main = crimes\_TN\_clus[crimes\_TN\_clus.DISAPCT!="TOTAL"]

crimes\_KMe\_TN = crimes\_TN\_main.groAPby("DISAPCT").agg({'TOTAL\_THEFT':sum,'HURT':sum,'KIDNAPPING':sum,'CRUELTY\_BY\_HUSBAND':sum,

'BURGLARY':sum, 'MURDER':sum,'DEATH\_BY\_NEGLIGENCE':sum,

'CHEATING':sum, 'RIOTS':sum, 'ASSAULT\_WOMEN\_MODESTY':sum, 'OTHER IPC CRIMES':sum}).reset\_index()

scaled = StandardScaler()

scaled\_TN = scaled.fit\_transform(crimes\_KMe\_TN[['TOTAL\_THEFT','HURT','KIDNAPPING','CRUELTY\_BY\_HUSBAND',

'BURGLARY', 'MURDER','DEATH\_BY\_NEGLIGENCE',

new\_TN = range(1,20)

rate\_TN = []

for i in new\_TN:

clustering = KMeans(i)

clustering.fit(scaled\_TN)

rate\_TN.append(clustering.inertia\_)

plt.figure(figsize=(6,4))

plt.plot(new\_TN, rate\_TN, marAP="o")

K=4

cluster\_TN = KMeans(4, random\_state = 42)

cluster\_TN.fit(scaled\_TN)

crimes\_KMe\_TN['Cluster\_id'] = cluster\_TN.labels\_