A Mini Project Report

on

ELECTION PREDICTION ANALYSIS

submitted in partial fulfillment of the requirements for the award of degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING

by

K. RAHUL(17211A05D5)

K.HIMA BINDHU(17211A05E3)

M.SREEJA(17211A05G3)

M.AKHILESH(17211A05J0)

Under the guidance of

Mrs.P.JHANSI DEVI,M.tech**,**

Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

B.V.RAJU INSTITUTE OF TECHNOLOGY

(UGC Autonomous, Accredited by NBA & NAAC)

Vishnupur, Narspur, Medak(Dist.), Telangana State,India-502313

2017 - 2021

B. V. Raju Institute of Technology
(UGC Autonomous, Accredited By NBA & NAAC)
Vishnupur, Narspur, Medak (Dist.),
Telangana State, India – 502313

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Major Project entitled **"ELECTION PREDICTION ANALYSIS"**, being submitted by

In partial fulfillment of the requirements for the award of degree of BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE AND ENGINEERING to B.V.RAJU INSTITUTE OF TECHNOLOGY is a record of bonafide work carried out during a period from May 2019 to July 2020 by them under the guidance of **P.Jhansi Devi**, Assistant Professor, CSE Department.

K. RAHUL(17211A05D5)
K.HIMA BINDHU(17211A05E3)
M.SREEJA(17211A05G3)
M.AKHILESH(17211A05J0)

This is to certify that the above statement made by the students is/are correct to the best of my knowledge.

P.JHANSI DEVI
Assistant Professor

The Project Viva-Voce Examination of this team has been held on _____.

Mr. Karthik Kovuri
Project Coordinator

Dr. Ch. Madhu Babu
Professor & HoD-CSE

EXTERNAL EXAMINER

B. V. Raju Institute of Technology
(UGC Autonomous, Accredited By NBA & NAAC)
Vishnupur, Narspur, Medak (Dist.),
Telangana State, India – 502313

CANDIDATE'S DECLARATION

We hereby certify that the work which is being presented in the project entitled **"Election Prediction Analysis"** in partial fulfillment of the requirements for the award of Degree of Bachelor of Technology and submitted in the Department of Computer Science and Engineering, B. V. Raju Institute of Technology, Narsapur is an authentic record of my own work carried out during a period from May 2020 to July 2021 under the guidance of **P.Jhansi Devi** , Assistant Professor. The work presented in this project report has not been submitted by us for the award of any other degree of this or any other Institute/University.

K. RAHUL(17211A05D5)
K.HIMA BINDHU(17211A05E3)
M.SREEJA(17211A05G3)
M.AKHILESH(17211A05J0)

ABSTRACT

In this project we were interested to see what factors can predict elections. To address this question we look at a dataset of India general election 2019 that includes information about their constituency, candidate name, party, electoral votes, Gender etc . We have considered a total of 19 columns. Our analysis depicts how these variables interact with Winner, and essentially How people voted using EDA. Using our models on the training data we are going to predict dependent variables on the test data. Later we look at our solution to extract features with the highest effect on predicting the winner. This analysis could be used to give predictions on the basis of our data to see what factors are most important to affect the winner. In order to answer these questions, we used a couple of models to predict exact values for our dependent variables (winner).

Key Words: Election , Politics, EDA,Feature engineering ,Random Forest, K-Nearest Neighbor

CONTENTS

Chapter 1

INTRODUCTION

1.1 Motivation

It is undoubtedly true that even with the presence of institutions and services of the safety personnel, healthcare workers, E commerce, Entertainment industry, etc., if good governance is lacking to ensure the proper management of these institutions and execute effective policies, their effects will not be felt by the populace. Hence the need for good and efficient governance in the society.
This is why the need to elect the right candidate who would fill that position shouldn't be underestimated.

Election is among influential factors on not only the nation's economy but also many others on a global scale. We must pay attention to political events and how they influence us. Given this significance and how it is capable of influence. There is a need to predict the outcome of an election and formulate the consequences based on that.

1.2 Problem Definition

The major challenge in predicting elections is to find the factors influencing it as they are different for each country like USA and India being largest democracies are nowhere same in many considerations for election . This makes prediction either complex or not efficient to predict especially with countries like India, no one can say what could be a deciding factor. Since we have a good amount of data in today's world, we can use various machine learning algorithms to analyze the data for hidden patterns. The

hidden patterns can be used in understanding the trends in the political arena.

1.3 Objective of Project

Elections and voting is an essential part of any democracy. The development of information and data motivated us to work on them for one's own curiosity or necessity to be able to predict the future in many areas with development in AI and ML concepts .

The main objective of developing this project are:

1.To develop machine learning model to predict future possible winner by implementing classification namely random forest classifier

2.To determine significant factors based on dataset which may influence election behaviour with EDA ploty

3.To understand how an indian voter voted .

The features of election Prediction Using Machine Learning are as follows.

- This Project will predict the winner of the Elections based on the factors and other general information using the datasets.

- This is done based on the previous datasets so after comparing it can provide up to 80% of accurate results, and the project is still developing further to get the 100% accurate results.

- With the help of election prediction, it can predict the winner and so that we can conclude various trends and perceptions of voting behaviour.

- It provides security for the system so that no one can break into that and no one can make any changes in the system.

1.4 Limitations of Project

The limitation of this project are

1.By using random forest,it is that a large number of trees can make the algorithm too slow and ineffective for real-time predictions. In general, these algorithms are fast to train, but quite slow to create predictions once they are trained. A more accurate prediction requires more trees, which results in a slower model.

2. unrepresentative sample because it puts the sample into the context of the population by considering the sample size.

3.the fact that one model is trained with particular election data hurts our ability to use it in other countries and times.

Chapter 2

LITERATURE SURVEY

2.1 Introduction

Although forecasting has been used many times in numerous fields, it has a brief history in political science. Forecasting political events started in the late 1970s when Fair (1978) investigated the effect of the economic condition in the election year as well the incumbent parties in a forecasting model. Sigelman (1979) examined the relation between the results of an election with the previous ones. Lewis-Beck and Rice (1982) developed a model using the president's job approval and an economic factor as independent variables. Abramowitz (1988) added a time-dependent variable to improve the performance of the forecasting model. The dependent variable of the model was the percentage of the incumbent party votes, and the independent variables were GDP growth, the incumbent president's job approval rating in June of the election year, and the consecutive terms that the incumbent party governs the country. He used the ordinary least-squares (OLS) method to estimate the parameters of the linear regression model. Later, Abramowitz (2016) utilized his model, which is called "Time for change forecasting model", to forecast 2016 election.

Some years later, Lewis-Beck and Rice (1992) reformed their model by adding two new variables: the result of the previous congress election and the previous presidential election. Holbrook and DeSart (1999) used the percentage of voters and the last votes of parties, as variables in their forecasting model. They employed the OLS method to estimate the parameters of their regression model.

2.2 Existing System

Generally we predict election traditionally

•Opinion polls

•Exit polls

•Survey data

•Other mathematical and Statistical tools.

Some traditional methods are:

Averaging polls

Combining poll data lowers the forecasting mistakes of a poll.[5] Political forecasting models include averaged poll results, such as the RealClearPolitics poll average.

Poll damping

Poll damping is when incorrect indicators of public opinion are not used in a forecast model. For instance, early in the campaign, polls are poor measures of the future choices of voters. The poll results closer to an election are a more accurate prediction. Campbell[6] shows the power of poll damping in political forecasting.

Nomenclature

When discussing the likelihood of a particular electoral outcome, political forecasters tend to use one of a small range of shorthand phrases.[8][9][10] These include:

Solid (e.g., "Solid Republican"), also Safe. Very unlikely that the party which currently holds the seat will change in the upcoming election.
Likely (e.g., "Likely Democratic"), also Favored. It is not thought at the moment that the seat will be particularly competitive, and hence the party is likely to remain unchanged, but there is a possibility this may alter.
Lean (e.g., "Leans Independent"). One candidate / party has a slight advantage in polling and forecasting, but other outcomes are possible.
Tilt. Used less widely than the other terms, but indicates a very small advantage to one or another party.
Toss-Up. These are the seats that are considered to be the most competitive, with more than one party having a good chance of winning.

2.3 Disadvantages of Existing System

- This is more time consuming and costly .
- Once the algorithm is created, it only needs to be run on the new data.
- Additionally, the systemic limitation around conformity that polls struggle with , data source where people mostly interact with strangers, increases the gap in conformity.
- These models are not real time.
- the voter turnout for specific regions in the country which is almost impossible to do with polls.
- Last but not least, Beauchamp's research proves that ML can be more accurate than traditional polls (Beauchamp, 2015).

2.4 Proposed System

We propose Using Machine Learning algorithms to predict the winner and to analyse the data .Machine learning applies complex mathematical algorithms to automatically recognize patterns, capture demand signals and spot complicated relationships in large datasets. Apart from analyzing huge volumes of information, smart systems continuously retrain models, adapting them to changing conditions thus addressing volatility. These capabilities enable ML-based software to produce more accurate and reliable predictions in complex
scenarios.

● aggregating historical and new data from different sources;
● cleansing data;
● determining which algorithm fits your product best;
● building predictive models to identify likely outcomes and discover relationships between various factors; and
● monitoring models to measure their results and improve prediction accuracy.

ADVANTAGES OF PROPOSED SYSTEM

- Proper use of data classification algorithms has several advantages.
- These algorithms serve to validate the data, because they are based on mathematical models.
- Once the system has been trained, within the domain that is done, the manipulation of data, interpretation of the results is often simple.
- Because, using the mathematical methods, experiments can be repeated and provide the same results for verification.
- The clustering techniques have significant advantages in terms of time.
- The cost of learning is null. No need to make any assumptions about the concepts to learn.
- You can extend the mechanism to predict continuous values

Chapter3

ANALYSIS

3.1 Introduction

3.1 MACHINE LEARNING

Tom Mitchell states machine learning as “A computer program is said to learn from experience and from some tasks and some performance on, as measured by, improves with experience”. Machine Learning is combination of correlations and relationships, most machine learning algorithms in existence are concerned with finding and/or exploiting relationship between datasets. Once Machine Learning Algorithms can pinpoint on certain correlations, the model can either use these relationships to predict future observations or generalize the data to reveal interesting patterns. In Machine Learning there are various types of algorithms such as Regression, Linear Regression, Logistic Regression, Naive Bayes Classifier, Bayes theorem, KNN (K-Nearest Neighbor Classifier), Decision Tress, Entropy, ID3, SVM (Support Vector Machines), K-means Algorithm, Random Forest and etc.,

The name machine learning was coined in 1959 by Arthur Samuel. Machine learning explores the study and construction of algorithms that can learn from and make predictions on data Machine learning is closely related to (and often overlaps with) computational statistics, which also focuses on prediction-making through the use of computers. It has strong ties to mathematical optimization, which delivers methods, theory and application domains to the field. Machine learning is sometimes conflated with data mining, where the latter subfield focuses more on exploratory data analysis and is known as unsupervised learning.

Within the field of data analytics, machine learning is a method used to devise complex models and algorithms that lend themselves to prediction; in commercial use, this is known as predictive analytics. These analytical models allow researchers, data scientists, engineers, and analysts to "produce reliable, repeatable decisions and results" and uncover "hidden insights" through learning from historical relationships and trends in the data.

Machine learning tasks Machine learning tasks are typically classified into several broad categories:

Supervised learning: The computer is presented with example inputs and their desired outputs, given by a "teacher", and the goal is to learn a general rule that maps inputs to outputs. As special cases, the input signal can be only partially available, or restricted to special feedback.

Semi-supervised learning: The computer is given only an incomplete training signal: a training set with some (often many) of the target outputs missing.

Active learning: The computer can only obtain training labels for a limited set of instances (based on a budget), and also has to optimize its choice of objects to acquire labels for. When used interactively, these can be presented to the user for labelling.

Unsupervised learning: No labels are given to the learning algorithm, leaving it on its own to find structure in its input. Unsupervised learning can be a goal in itself (discovering hidden patterns in data) or a means towards an end (feature learning).

Reinforcement learning: Data (in form of rewards and punishments) are given only as feedback to the program's actions in a dynamic environment, such as driving a vehicle or playing a game against an opponent.

3.1.1 FEATURES OF MACHINE LEARNING

• It is nothing but automating the Automation.

• Getting computers to program themselves.

• Writing Software is bottleneck.

• Machine leaning models involves machines learning from data without the help of humans or any kind of human intervention.

• Machine Learning is the science of making of making the computers learn and act like humans by feeding data and information without being explicitly programmed.

• Machine Learning is totally different from traditionally programming, here data and output is given to the computer and in return it gives us the program which provides solution to the various problems. Below is the figure

• Machine Learning is a combination of Algorithms, Datasets, andPrograms.

• There are Many Algorithms in Machine Learning through which we will provide us the exact solution in predicting the disease of the patients.

• How Does Machine Learning Works?

• Solution to the above question is Machine learning works by taking in data, finding relationships within that data and then giving the outpt

• There are various applications in which machine learning is implemented such as Web search, computing biology, finance, e-commerce, space exploration, robotics, social networks, debugging and much more.

• There are 3 types of machine learning supervised, unsupervised, and

reinforcement.

3.2 Software Requirement Specification

A software requirements specification (SRS) captures a complete description about how the system is expected to perform. Software requirements deal with defining software resource requirements and prerequisites that need to be installed on a computer to provide optimal functioning of an application.

3.2.1 User requirements

- Laptop/Phone.
- Internet connectivity.

3.2.2 Software Requirements

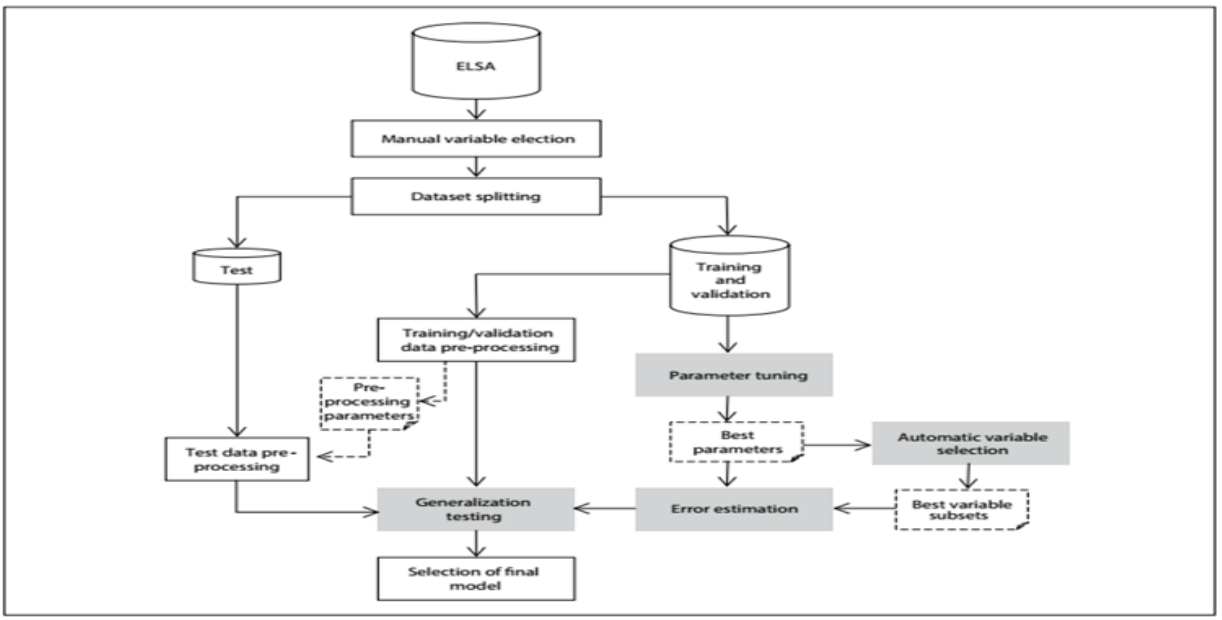
- ❖ Operating System : Windows 7, 10 or Higher Versions
- ❖ Platform : Jupiter Notebook
- ❖ Programming Lang : Python

3.2.3 Hardware Requirements

- ❖ System : Pentium 4, Intel Core i3, i5, i7 and 2 GHz Minimum
- ❖ RAM : 512Mb or above
- ❖ Hard Disk : 10 GB or above
- ❖ Input Device : Keyboard and Mouse
- ❖ Output Device : Monitor or PC

3.3 Content Diagrams of Project

Election prediction using machine learning predicts the winner based on various factors and the information such as gender, voters number and many more such general information through the data. The architecture of the system election prediction using machine learning consist of various datasets through which we will compare and predicts it, then the datasets are transformed into the smaller sets and from there it gets classified based on the classification algorithms later on the classified data is then processed into the machine learning technologies through which the data gets processed and goes in to the prediction model using all the inputs from the user that is mentioned above. Then entering the above information and overall processed data combines and compares in the prediction model of the system and finally predicts winner. An architecture diagram is a graphical representation of a set of concepts, that are part of an architecture, including their principles, elements and components. The diagram explains about the system software in perception of overview of the system.



3.4 Algorithms and Flowcharts

RANDOM FOREST ALGORITHM

• It is an ensemble classifier using many decision trees models; it can be used for regression as well as classification.

• Accuracy and variable importance information can be provided with the results.

• A random forest is the classifier consisting of a collection of tree structured classifiers k, where the k is independently, identically distributed random trees and each random tree consists of the unit of vote for classification of input.

• Random forest uses the Gini index for the classification and determining the final class in each tree.

• The final class of each tree is aggregated and voted by the weighted values to construct the final classifier.

• The working of random forest is, A random seed is chosen which pulls out at a random, a collection of samples from the training datasets while maintaining the class distribution

Chapter 4

DESIGN

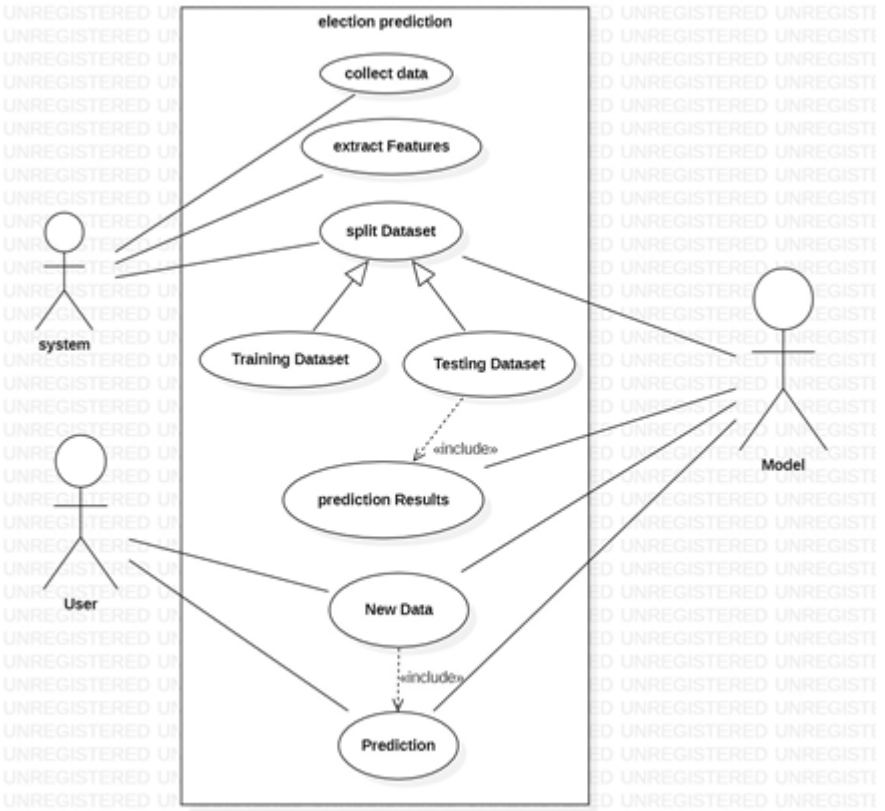
4.1 INTRODUCTION

The Design goals consist of various design which we have implemented in our system disease prediction using machine learning. This system has built with various designs such as data flow diagram, sequence diagram, class diagram, use case diagram, component diagram, activity diagram, state chart diagram, deployment diagram. After doing these various diagrams and based on these diagrams we have done our project.

4.1 UML DIAGRAM

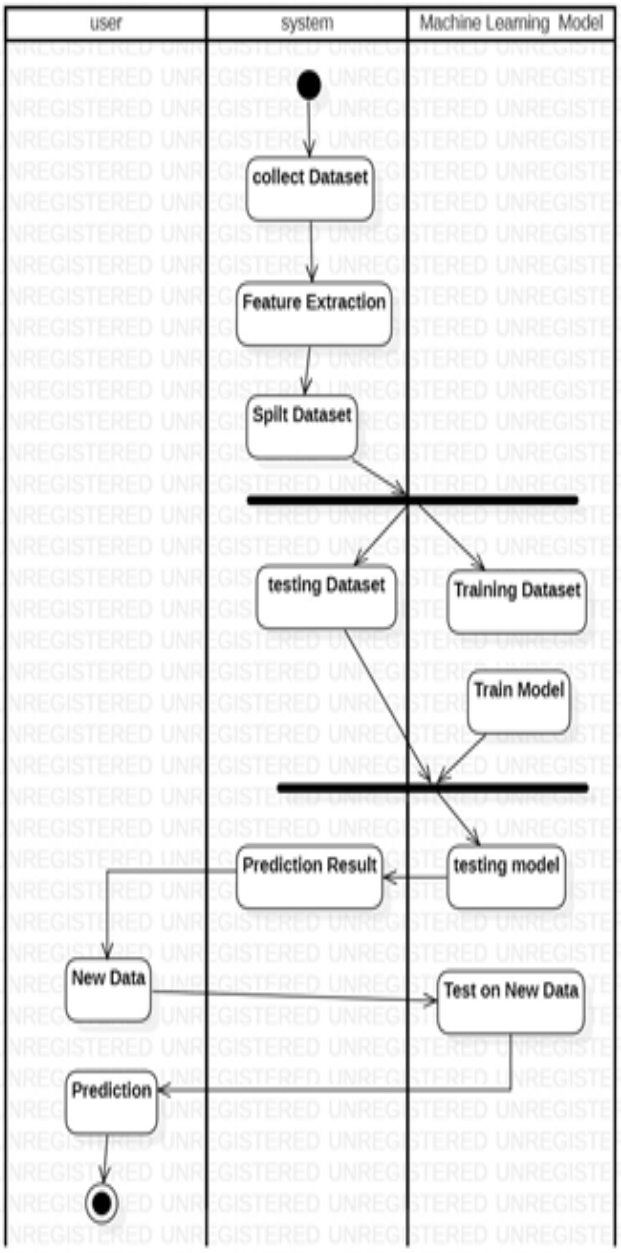
Use Case diagram

The Use Case diagram of the project election prediction using machine learning consist of all the various aspects a normal use case diagram requires. This use case diagram shows how from starting the model flows from one step to another, like he enter into the system then enters all the information's and all other general information along with the factors that goes into the system, compares with the prediction model and if true is predicts the appropriate results otherwise it shows the details where the user if gone wrong while entering the information's and it also shows the appropriate precautionary measure for the user to follow. Here the use case diagram of all the entities are linked to each other where the user gets started with the system.



Activity Diagram

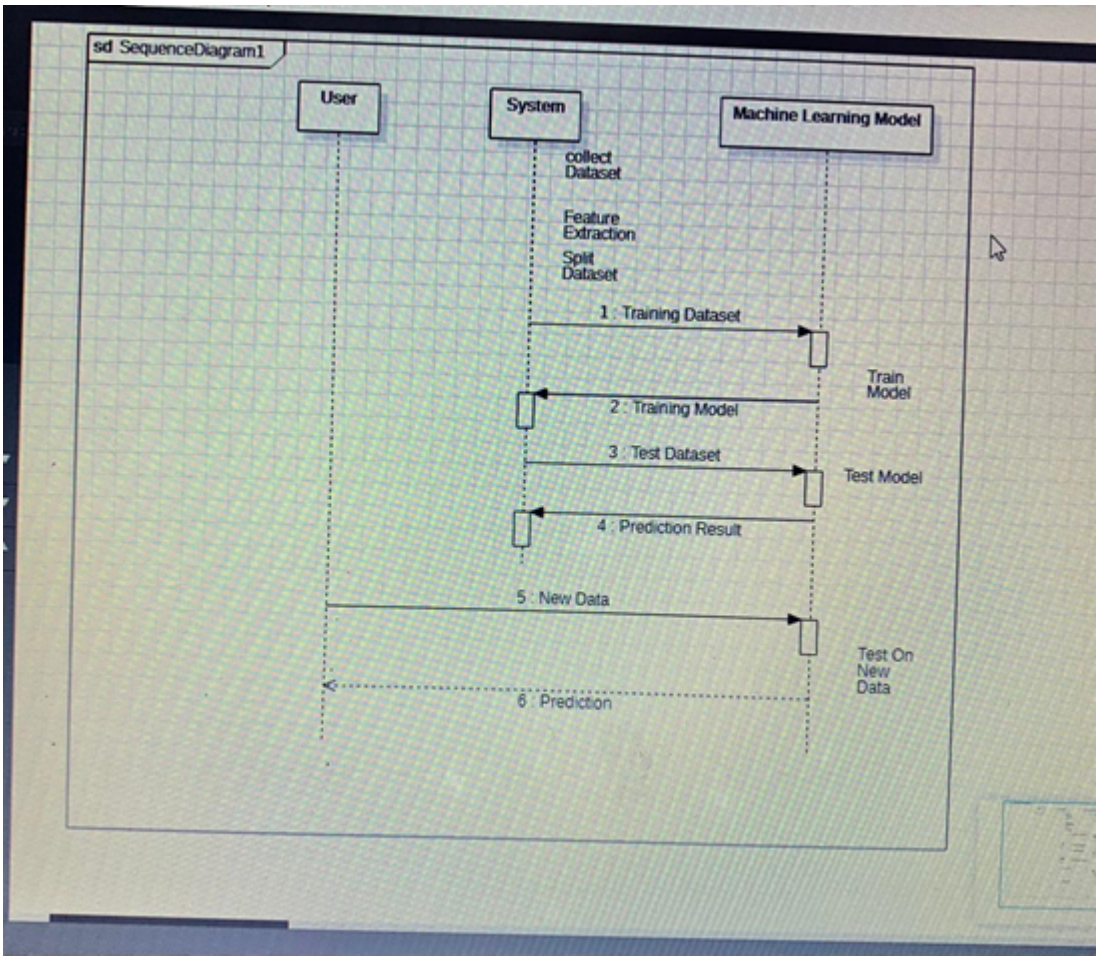
Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another. Here in this diagram the activity starts from system where the user registers into the system then login using the credentials and then the credentials are matched in the system and if its true, then the user proceeds to the prediction phase where the prediction happens. Then finally after processing the data from datasets the analysis will happen then the correct result will be displayed that is nothing but the Output.



Sequence diagram

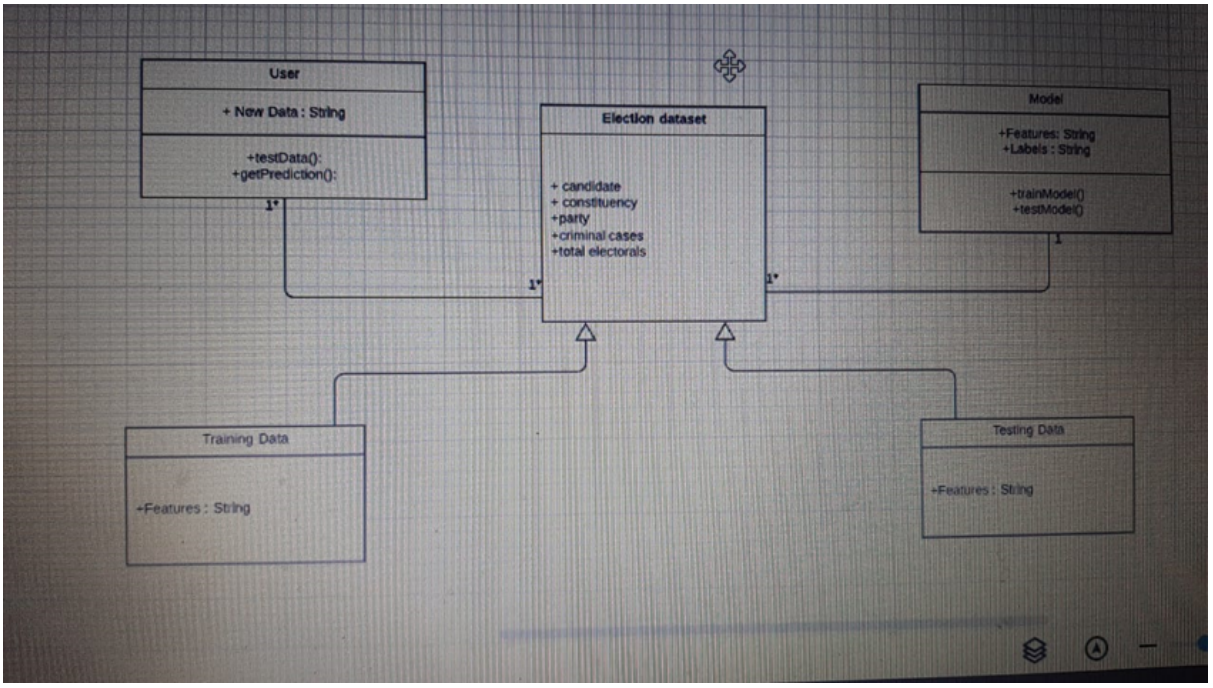
The Sequence diagram of the project election prediction using machine learning consist of all the various aspects a normal sequence diagram requires. This sequence diagram shows how from starting the model flows from one step to another, like he enter into the system then enters all the information's and all other general information along with the factors that goes

into the system, compares with the prediction model and if true is predicts the appropriate results otherwise it shows the details where the user if gone wrong while entering the information's and it also shows the appropriate precautionary measure for the user to follow. Here the sequence of all the entities are linked to each other where the user gets started with the system



Class diagram

Election prediction using machine learning consist of class diagram that all the other application that consists the basic class diagram, here the class diagram is the basic entity that is required in order to carry on with the project. Class diagram consist information about all the classes that is used and all the related datasets, and all the other necessary attributes and their relationships with other entities, all these information is necessary in order to use the concept of the prediction,



4.3 MODULE DESIGN AND ORGANISATION:

The main modules of the system are:

- collection of Data
- cleaning the raw data
- analysing the data
- identifying significant variable
- building model
- finding the best fit model for prediction

Chapter 5

IMPLEMENTATION

5.1 Introduction

The exit polls are feverish now and curious about their party's fate in the elections and nobody is the exception of this band. All exit polls are the predictions from the people's opinion and a thought stuck in what would be Artificial Intelligence's opinion. Machine learning struck the chord and worked on the data then identified some feature lists to predict the Election 2019 winner using Machine learning.

Election commission of India

Have collected the data from the election commission of India for the past 5 terms of data spanning from 1998 to 2014 results. Below is the dataset that I have captured with various features includes State, constituency, Gender, Caste, Votes polled, Votes majority and finally Party Won. Below is the dataset that is obtained from the https://eco.Gov.in.

5.2 Explanation of Key functions

Importing the dataset

```
# This Python 3 environment comes with many helpful analytics libraries installed
# It is defined by the kaggle/python Docker image: https://github.com/kaggle/docker-python
# For example, here's several helpful packages to load

import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)

# Input data files are available in the read-only "../input/" directory
# For example, running this (by clicking run or pressing Shift+Enter) will list all files under the input directory

import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))
import os, sys
from collections import defaultdict
from urllib.request import urlopen
import json
import plotly.graph_objects as go
from plotly.subplots import make_subplots
from ipywidgets import widgets
import geopandas as gpd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
import random
from plotly.offline import download_plotlyjs, init_notebook_mode, plot, iplot
```

```
import plotly.express as px
import plotly.graph_objects as go
import plotly.figure_factory as ff
from plotly.colors import n_colors
from plotly.subplots import make_subplots
init_notebook_mode(connected=True)
import cufflinks as cf
cf.go_offline()
from wordcloud import WordCloud , ImageColorGenerator
from PIL import Image
from sklearn.utils import resample
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import cross_val_score
# You can write up to 5GB to the current directory (/kaggle/working/) that gets preserved as output when you create a version using "Save & Run All"
# You can also write temporary files to /kaggle/temp/, but they won't be saved outside of the current session
```

Check out the data

```
vote=pd.read_csv('/kaggle/input/indian-candidates-for-general-election-2019/LS_2.0.csv')
vote.head()
```

Out[2]:

	STATE	CONSTITUENCY	NAME	WINNER	PARTY	SYMBOL	GENDER	CRIMINAL\nCAS
0	Telangana	ADILABAD	SOYAM BAPU RAO	1	BJP	Lotus	MALE	52
1	Telangana	ADILABAD	Godam Nagesh	0	TRS	Car	MALE	0
2	Telangana	ADILABAD	RATHOD RAMESH	0	INC	Hand	MALE	3
3	Telangana	ADILABAD	NOTA	0	NOTA	NaN	NaN	NaN
4	Uttar Pradesh	AGRA	Satyapal Singh Baghel	1	BJP	Lotus	MALE	5

Identifying the Null values in the columns

```
vote.isnull().sum()
```

Out[3]:

```
STATE                                          0
CONSTITUENCY                                   0
NAME                                           0
WINNER                                         0
PARTY                                          0
SYMBOL                                       245
GENDER                                       245
CRIMINAL\nCASES                              245
AGE                                          245
CATEGORY                                     245
EDUCATION                                    245
ASSETS                                       245
LIABILITIES                                  245
GENERAL\nVOTES                                 0
POSTAL\nVOTES                                  0
TOTAL\nVOTES                                   0
OVER TOTAL ELECTORS \nIN CONSTITUENCY          0
OVER TOTAL VOTES POLLED \nIN CONSTITUENCY      0
TOTAL ELECTORS                                 0
dtype: int64
```

```
TOTAL ELECTORS                                 0
```

Identifying the null entries in the data

```
vote[vote.SYMBOL.isnull()==True]['NAME'].unique()
```

Out[4]:

```
array(['NOTA'], dtype=object)
```

Cleaning up the Assets and Liabilities columns

```
def value_cleaner(x):
```

```
    try:
        str_temp = (x.split('Rs')[1].split('\n')[0].strip())
        str_temp_2 = ''
        for i in str_temp.split(","):
            str_temp_2 = str_temp_2+i
        return str_temp_2
    except:
        x = 0
        return x
vote['ASSETS'] = vote['ASSETS'].apply((value_cleaner))
vote['LIABILITIES'] = vote['LIABILITIES'].apply((value_cleaner))
vote.head()
```

Out[5]:

	STATE	CONSTITUENCY	NAME	WINNER	PARTY	SYMBOL	GENDER	CRIMINAL\nCAS
0	Telangana	ADILABAD	SOYAM BAPU RAO	1	BJP	Lotus	MALE	52
1	Telangana	ADILABAD	Godam Nagesh	0	TRS	Car	MALE	0
2	Telangana	ADILABAD	RATHOD RAMESH	0	INC	Hand	MALE	3
3	Telangana	ADILABAD	NOTA	0	NOTA	NaN	NaN	NaN
4	Uttar Pradesh	AGRA	Satyapal Singh Baghel	1	BJP	Lotus	MALE	5

Remaining columns

```
vote.rename(columns={"CRIMINAL\nCASES": "CRIMINAL CASES",
"GENERAL\nVOTES": "GENERAL VOTES", "POSTAL\nVOTES": "POSTAL
VOTES","TOTAL\nVOTES": "TOTAL VOTES","OVER TOTAL ELECTORS \nIN
CONSTITUENCY": "OVER TOTAL ELECTORS IN CONSTITUENCY","OVER TOTAL VOTES
POLLED \nIN CONSTITUENCY": "OVER TOTAL VOTES POLLED IN CONSTITUENCY"},
inplace=True)
```

```
vote.head()
```

Out[6]:

	STATE	CONSTITUENCY	NAME	WINNER	PARTY	SYMBOL	GENDER	CRIMINAL CASES	AGE
0	Telangana	ADILABAD	SOYAM BAPU RAO	1	BJP	Lotus	MALE	52	52.
1	Telangana	ADILABAD	Godam Nagesh	0	TRS	Car	MALE	0	54.
2	Telangana	ADILABAD	RATHOD RAMESH	0	INC	Hand	MALE	3	52.
3	Telangana	ADILABAD	NOTA	0	NOTA	NaN	NaN	NaN	Nal
4	Uttar Pradesh	AGRA	Satyapal Singh Baghel	1	BJP	Lotus	MALE	5	58.

Cleaning up the Educational Qualification of the election contestants

```
In [7]:
vote.EDUCATION.unique()
```

In [7]:
```
vote.EDUCATION.unique()
```

Out[7]:
```
array(['12th Pass', 'Post Graduate', nan, 'Doctorate', 'Graduate',
       'Others', '10th Pass', '8th Pass', 'Graduate Professional',
       'Literate', 'Illiterate', '5th Pass', 'Not Available',
       'Post Graduate\n'], dtype=object)
```

```
array(['12th Pass', 'Post Graduate', nan, 'Doctorate', 'Graduate',
       'Others', '10th Pass', '8th Pass', 'Graduate Professional',
```

```
In [8]:
vote.EDUCATION.replace({'Post Graduate\n':'Post Graduate'},inplace=True)
vote.EDUCATION.unique()
```

Out[8]:

```
array(['12th Pass', 'Post Graduate', nan, 'Doctorate', 'Graduate',
       'Others', '10th Pass', '8th Pass', 'Graduate Professional',
       'Literate', 'Illiterate', '5th Pass', 'Not Available'],
      dtype=object)
```

Identifying the Data Type of the columns

In [9]:

```
vote.dtypes
```

Out[9]:

```
STATE                                  object
CONSTITUENCY                           object
NAME                                   object
WINNER                                  int64
PARTY                                  object
SYMBOL                                 object
GENDER                                 object
CRIMINAL CASES                         object
AGE                                   float64
CATEGORY                               object
EDUCATION                              object
ASSETS                                 object
LIABILITIES                            object
GENERAL VOTES                           int64
POSTAL VOTES                            int64
TOTAL VOTES                             int64
OVER TOTAL ELECTORS IN CONSTITUENCY    float64
OVER TOTAL VOTES POLLED IN CONSTITUENCY float64
TOTAL ELECTORS                          int64
dtype: object
```

Identifying Discrepancy entries in the columns

In [10]:

```
vote[vote['CRIMINAL CASES']=='Not Available'].head()
```

Out[10]:

Out[10]:

	STATE	CONSTITUENCY	NAME	WINNER	PARTY	SYMBOL	GENDER	CRIMINAL CASES
468	Bihar	BUXAR	Ramchandra Singh Yadav	0	IND	Almirah	MALE	Not Available
532	Tamil Nadu	CHIDAMBARAM	SIVAJOTHI M	0	NTK	Ganna Kisan	MALE	Not Available
612	Uttar Pradesh	DEORIA	BINOD KUMAR JAISWAL	0	BSP	Elephant	MALE	Not Available
613	Uttar Pradesh	DEORIA	NIYAZ AHMED	0	INC	Hand	MALE	Not Available
654	Tamil Nadu	DINDIGUL	JOTHIMUTHU, K.	0	PMK	Mango	MALE	Not Available

In [11]:

```
vote['ASSETS']=pd.to_numeric(vote['ASSETS'])
vote['LIABILITIES']=pd.to_numeric(vote['LIABILITIES'])
vote['CRIMINAL CASES'].replace({np.NaN:0})
vote['CRIMINAL CASES'] = pd.to_numeric(vote['CRIMINAL CASES'],
errors='coerce').fillna(0).astype(np.int64)
```

Out[10]:

	STATE	CONSTITUENCY	NAME	WINNER	PARTY	SYMBOL	GENDER	CRIMINAL CASES
468	Bihar	BUXAR	Ramchandra Singh Yadav	0	IND	Almirah	MALE	Not Available
532	Tamil Nadu	CHIDAMBARAM	SIVAJOTHI M	0	NTK	Ganna Kisan	MALE	Not Available
612	Uttar Pradesh	DEORIA	BINOD KUMAR JAISWAL	0	BSP	Elephant	MALE	Not Available
613	Uttar Pradesh	DEORIA	NIYAZ AHMED	0	INC	Hand	MALE	Not Available
654	Tamil Nadu	DINDIGUL	JOTHIMUTHU, K.	0	PMK	Mango	MALE	Not Available

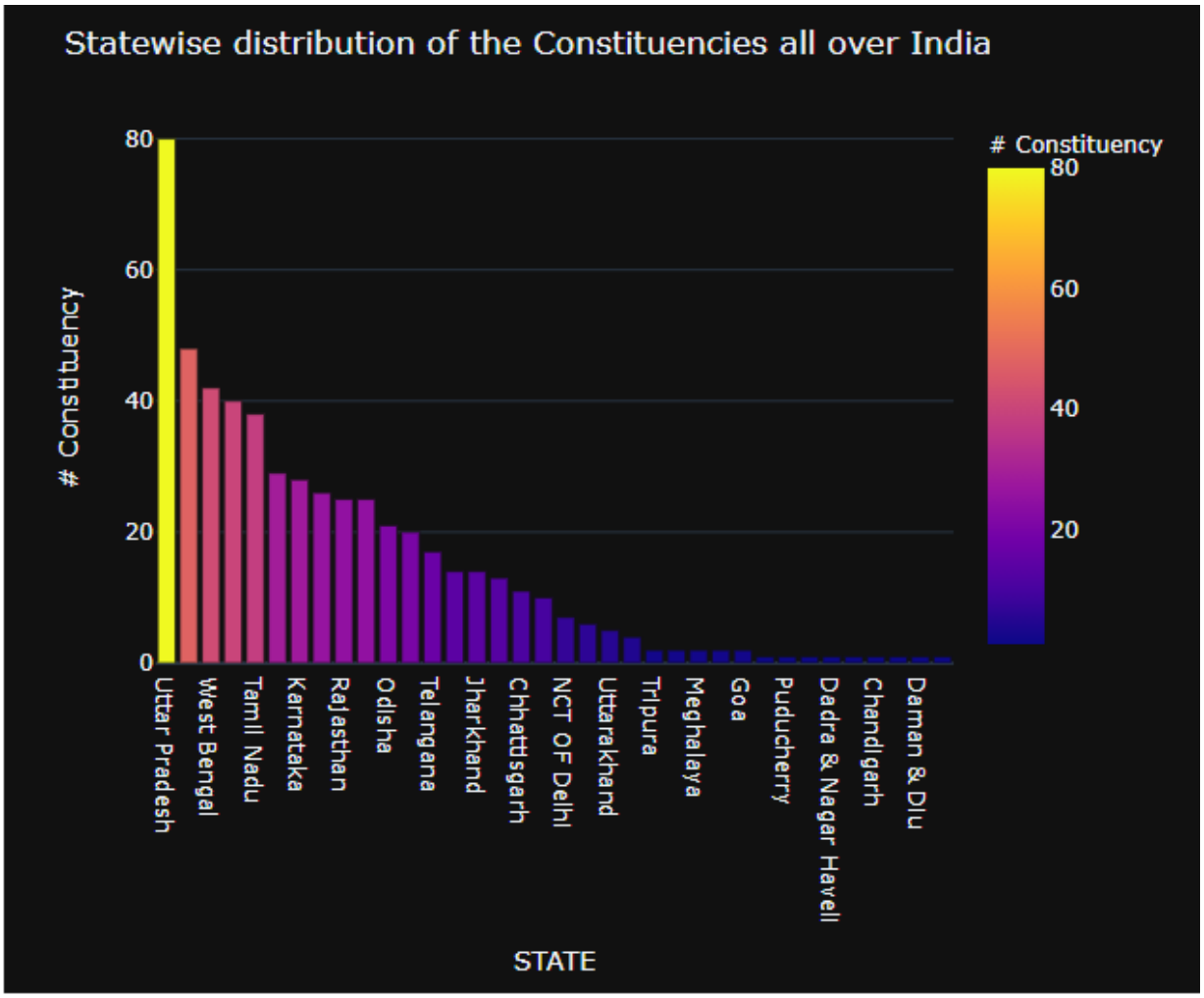
The Analysis

State and Constituency Level Analysis

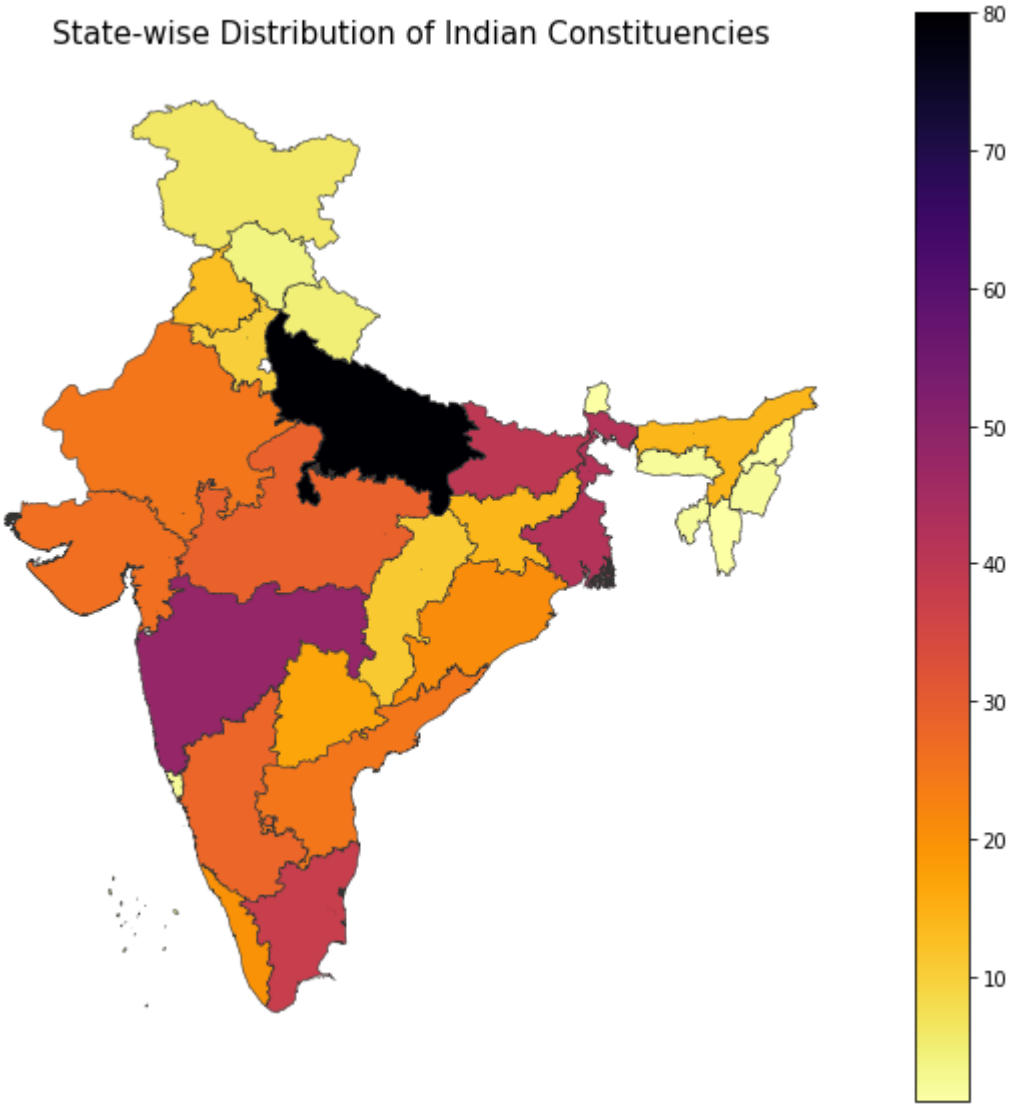
the distribution of Constituencies over all the states

In [12]:

```
st_con=vote.groupby('STATE').apply(lambda
x:x['CONSTITUENCY'].nunique()).reset_index(name='# Constituency')
shp_gdf =
gpd.read_file('/kaggle/input/india-states/Igismap/Indian_States.shp')
merged = shp_gdf.set_index('st_nm').join(st_con.set_index('STATE'))
fig, ax = plt.subplots(1, figsize=(10, 10))
ax.axis('off')
ax.set_title('State-wise Distribution of Indian Constituencies',
             fontdict={'fontsize': '15', 'fontweight' : '3'})
fig = merged.plot(column='# Constituency',
cmap='inferno_r',linewidth=0.5, ax=ax, edgecolor='0.2',legend=True)
st_con.sort_values(by='# Constituency',ascending=False,inplace=True)
fig2 = px.bar(st_con, x='STATE', y='# Constituency',
                      color='# Constituency',
             labels={'pop':'Constituencies of India'})
fig2.update_layout(title_text='Statewise distribution of the
Constituencies all over India',template='plotly_dark')
fig2.show()
```



Statewise distribution of the Constituencies all over IndiaSTATE# Constituency

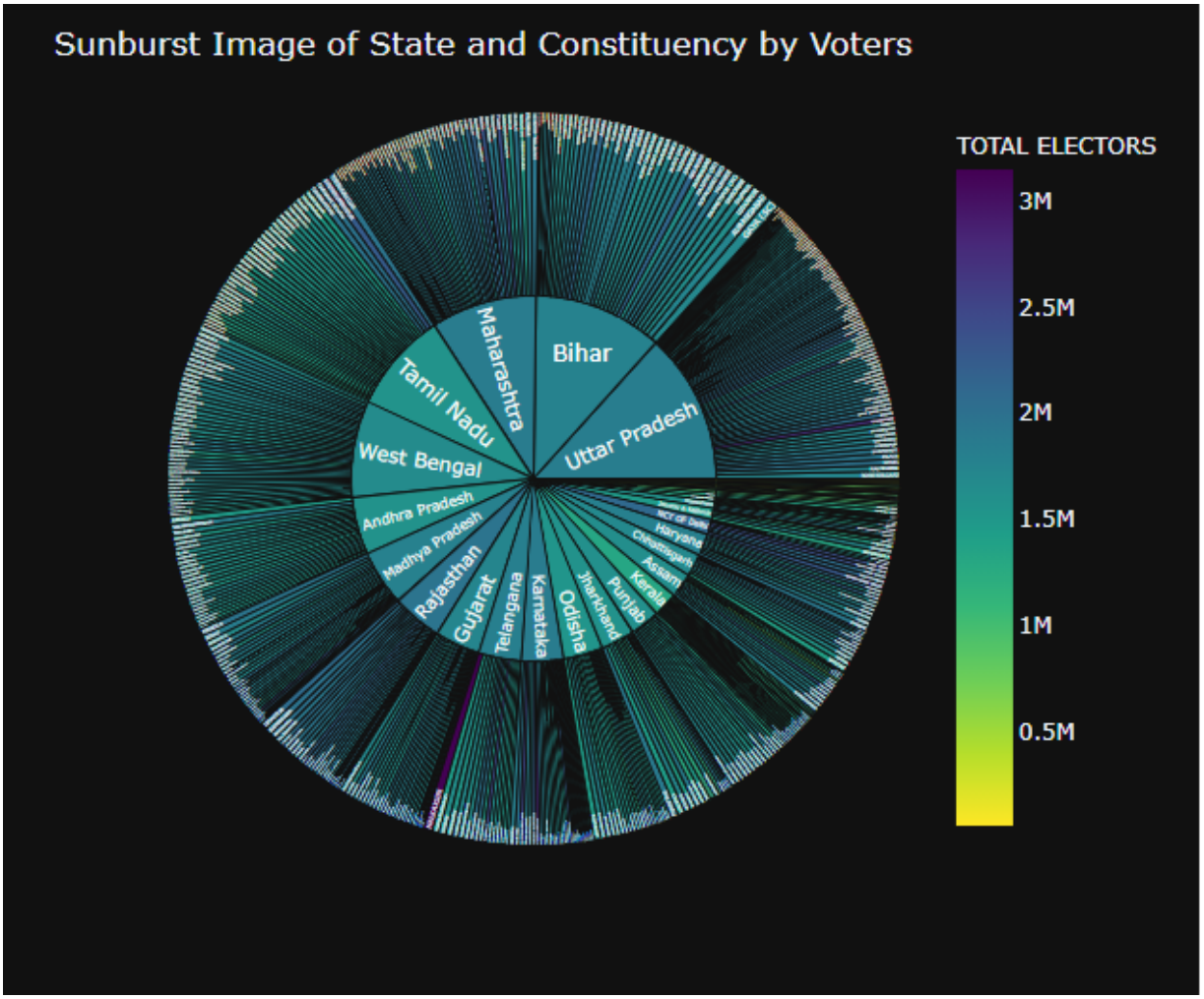


Observation Uttar Pradesh, Maharashtra and West Bengal- The sates have the most number of constituencies. There exists a direct relationship of count of constituencies and population- The constituencies are divided based on the population of 1971- and this shall remain till the year 2026. Although currently Bihar has a higher population, West Bengal has the 3rd highest constituency count based on the above fact.

Sunburst image of all the States and Constituencies

```
In [13]:
st_con_vt=vote[['STATE','CONSTITUENCY','TOTAL ELECTORS']]
fig = px.sunburst(st_con_vt, path=['STATE','CONSTITUENCY'],
values='TOTAL ELECTORS',
                  color='TOTAL ELECTORS',
                  color_continuous_scale='viridis_r')
```

```
fig.update_layout(title_text='Sunburst Image of State and Constituency
by Voters',template='plotly_dark')
fig.show()
```

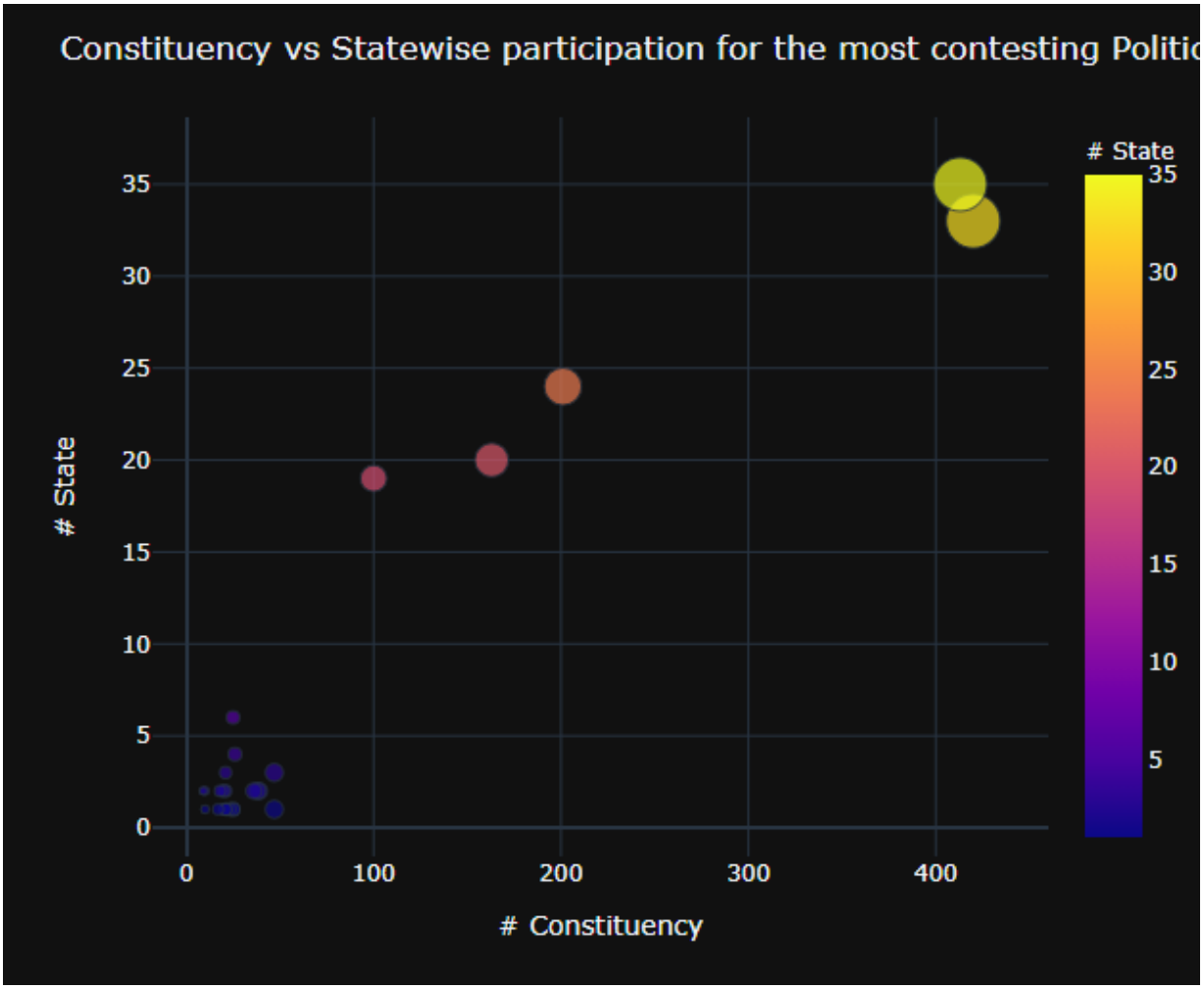


Party Level Analysis

Which Parties have been present in most constituencies and States?

```
In [14]:
vote_prty=vote[vote['PARTY']!='NOTA']
prty_cnt=vote_prty.groupby('PARTY').apply(lambda
x:x['CONSTITUENCY'].count()).reset_index(name='# Constituency')
```

```
prty_st=vote_prty.groupby('PARTY').apply(lambda
x:x['STATE'].nunique()).reset_index(name='# State')
prty_cnt.sort_values(by='# Constituency',ascending=False,inplace=True)
prty_top_cn=prty_cnt[:25]
prty_top_all=pd.merge(prty_top_cn,prty_st,how='inner',left_on='PARTY',ri
ght_on='PARTY')
fig = px.scatter(prty_top_all, x='# Constituency', y='# State', color='#
State',
                 size='# Constituency', hover_data=['PARTY'])
fig.update_layout(title_text='Constituency vs Statewise participation
for the most contesting Political Parties',template='plotly_dark')
fig.show()
```



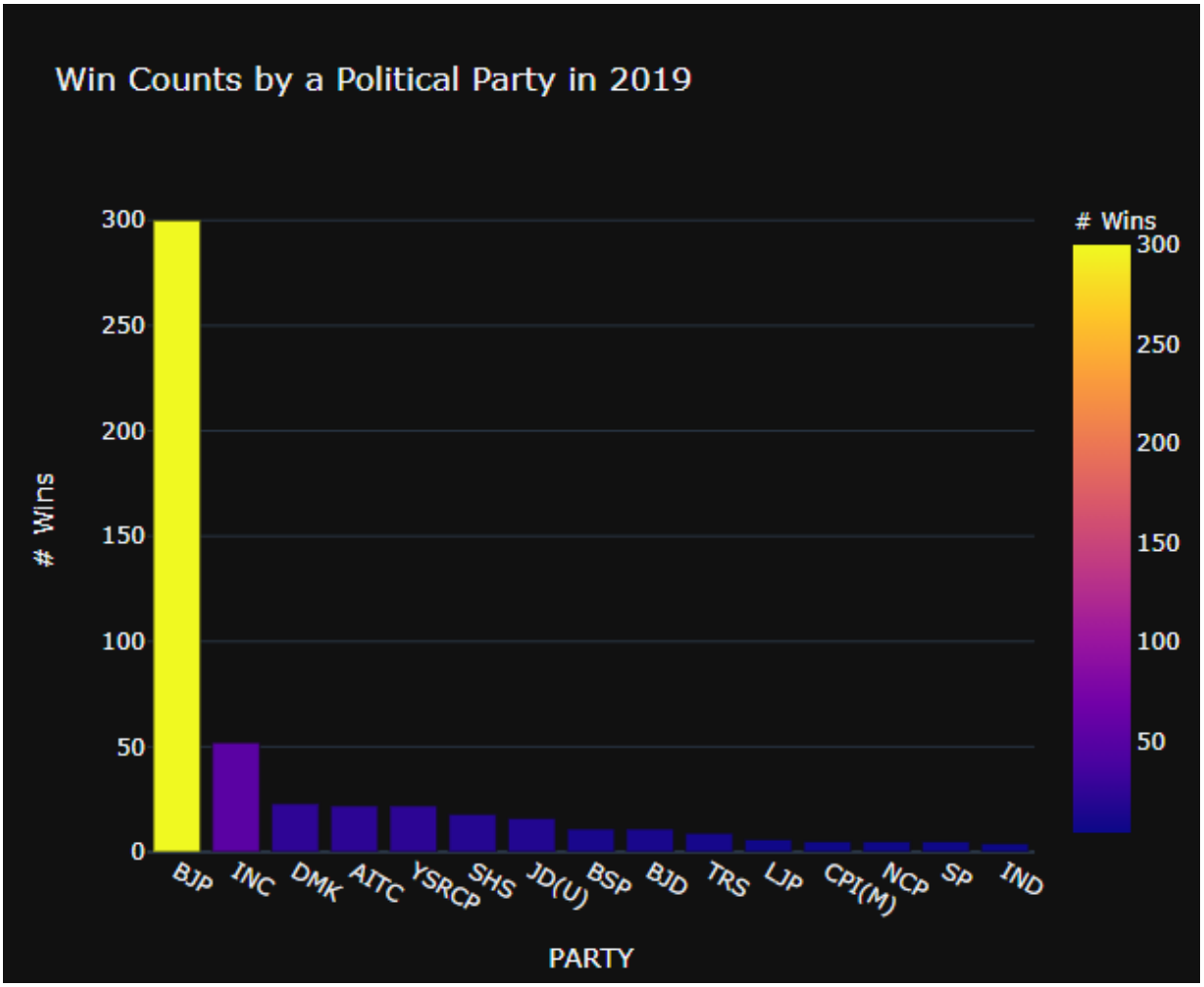
Observation The Bharatiya Janata Party (BJP) and Indian National Congress (INC) have participated in the most number of constituencies all over India. While BJP leads in the number of constituency contested, INC wins in terms of the number of States. While these

are the major parties to contest almost all over India, we see the rest of the parties have restricted themselves to a handfull of states.

Which party has won the most constituencies?¶

In [16]:

```
part_win=vote.groupby('PARTY').apply(lambda
x:x['WINNER'].sum()).reset_index(name='# Wins')
part_win.sort_values(by='# Wins',ascending=False,inplace=True)
top_part_win=part_win[0:15]
fig = px.bar(top_part_win, x='PARTY', y='# Wins',
                    color='# Wins',title='Win Counts by a Political
Party in 2019')
fig.update_layout(title_text='Win Counts by a Political Party in
2019',template='plotly_dark')
fig.show()
```

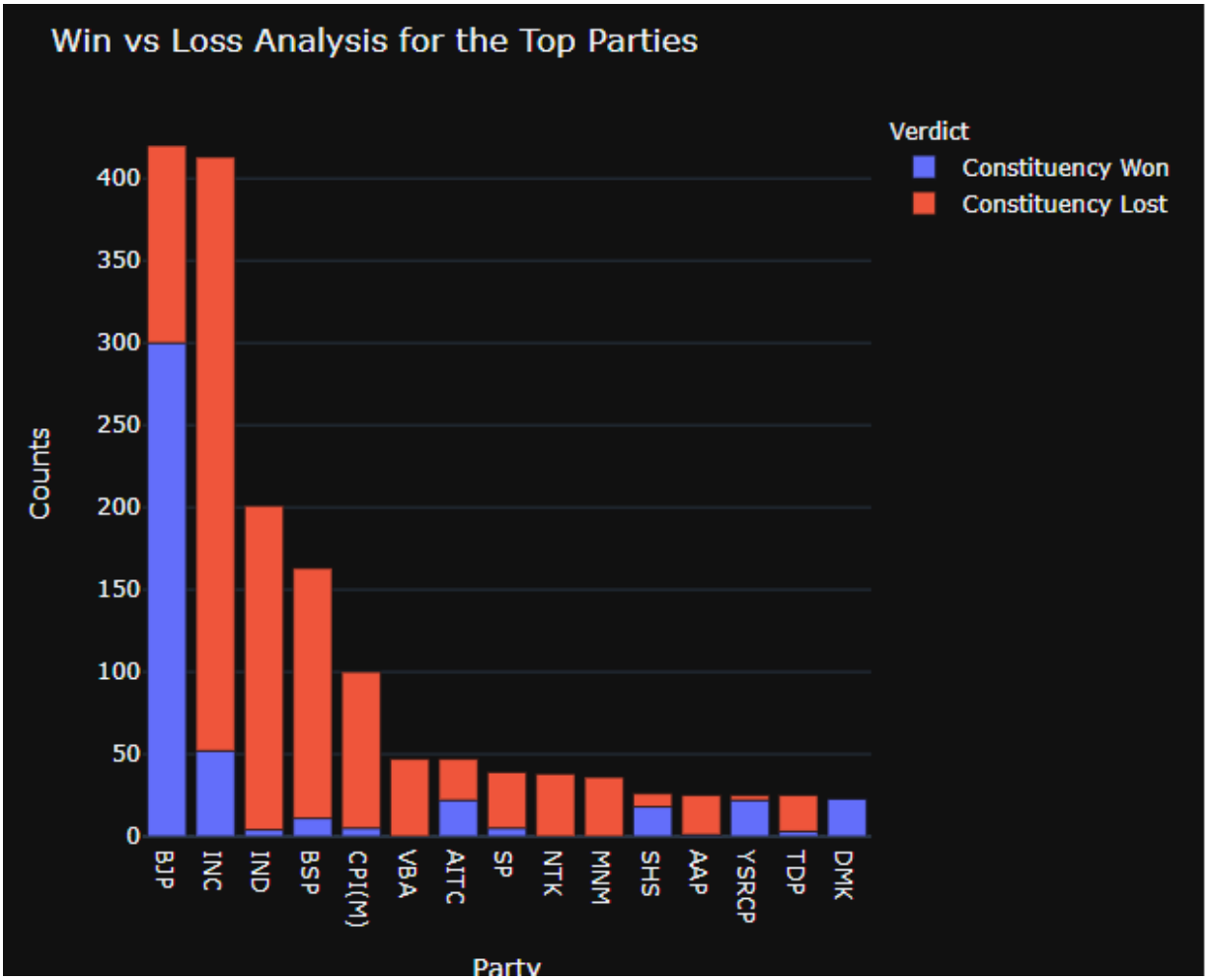


Observation As seen from the data, In 2019, BJP has won the maximum constituencies all over India. The Image below the introduction also suggests the same. The distribution of all the parties is presented below. INC, who stood 2nd in the number of victories had only 52, which is practically 1/6th of the constituencies won by BJP

What has been the general Win vs Loss relationship for the Parties in 2019?

```
In [17]:
prty_cnt_win=pd.merge(prty_cnt,part_win,how='inner',left_on='PARTY',righ
t_on='PARTY')
prty_cnt_win['Lost']=prty_cnt_win['# Constituency']-prty_cnt_win['#
Wins']
prty_wins_cnt=prty_cnt_win[['PARTY','# Wins']]
prty_wins_cnt['Verdict']='Constituency Won'
prty_loss_cnt=prty_cnt_win[['PARTY','Lost']]
```

```
prty_loss_cnt['Verdict']='Constituency Lost'
prty_wins_cnt.columns=['Party','Counts','Verdict']
prty_loss_cnt.columns=['Party','Counts','Verdict']
top_prty_wins_cnt=prty_wins_cnt[:15]
prty_loss_cnt_cnt=prty_loss_cnt[:15]
prt_win_loss=pd.concat([top_prty_wins_cnt,prty_loss_cnt_cnt])
fig = px.bar(prt_win_loss, x='Party', y='Counts', color='Verdict')
fig.update_layout(title_text='Win vs Loss Analysis for the Top
Parties',template='plotly_dark')
fig.show()
```



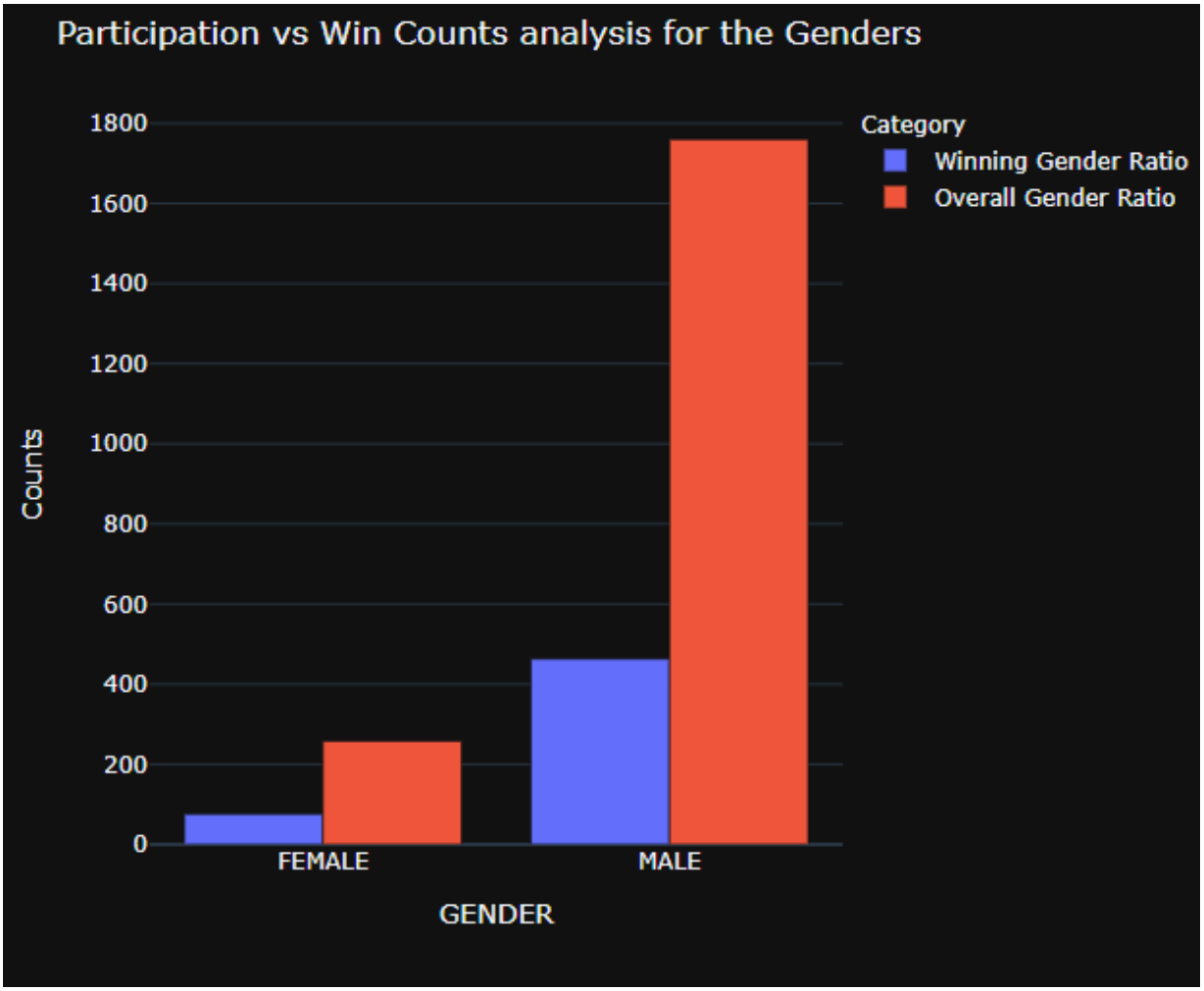
Observation As seen in the above chart, the 2019 elections have been extremely lucky for parties like BJP,SHS or DMK. But it has been a major failure for the rest of the parties, where they have lost more than they won.

Politician Level Analytics

What is the Gender Ratio of the Contestants? Also the Gender Ratio of the Winners?

In [18]:

```
vote_gndr=vote[vote['PARTY']!='NOTA']
gndr_overall=vote_gndr.groupby('GENDER').apply(lambda
x:x['NAME'].count()).reset_index(name='Counts')
gndr_overall['Category']='Overall Gender Ratio'
winners=vote_gndr[vote_gndr['WINNER']==1]
gndr_winner=winners.groupby('GENDER').apply(lambda
x:x['NAME'].count()).reset_index(name='Counts')
gndr_winner['Category']='Winning Gender Ratio'
gndr_overl_win=pd.concat([gndr_winner,gndr_overall])
fig = px.bar(gndr_overl_win, x='GENDER', y='Counts',
             color='Category', barmode='group')
fig.update_layout(title_text='Participation vs Win Counts analysis for
the Genders',template='plotly_dark')
fig.show()
```

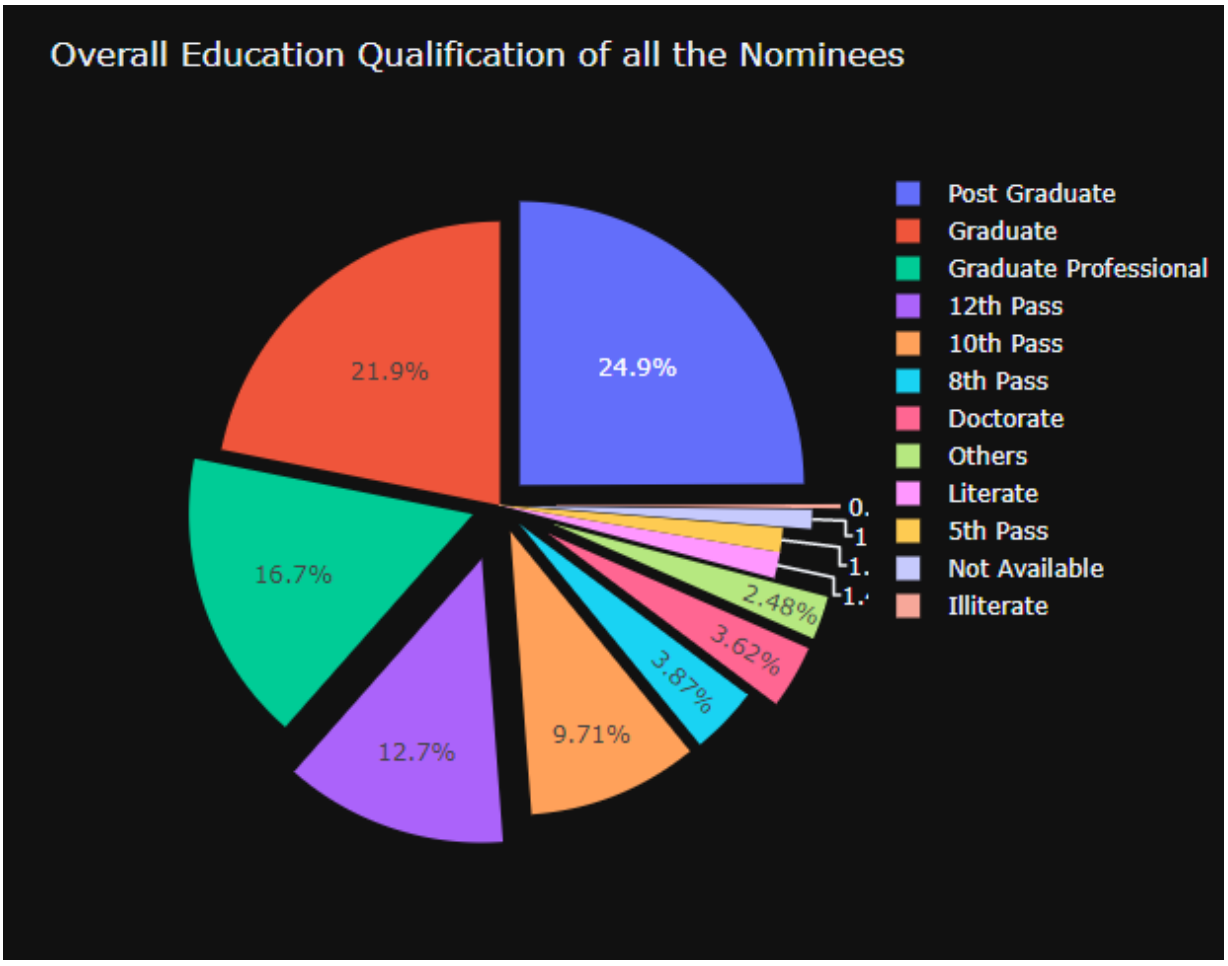


Observation Out of the total list of participants only 12.78% (258 out of 2018) are female politicians, while 87.21% (1760 out of 2018) are male. Upon considering the winners, 14.1% (76 out of 463) are female politicians, while 85.9% are male politicians. The Gender ratio is not very well distributed as can be seen from the above presentation.

What is the Educational Qualification of our politicians?

```
In [19]:
ed_valid=vote[vote['PARTY']!="NOTA"]
ed_cnt=ed_valid.groupby('EDUCATION').apply(lambda
x:x['PARTY'].count()).reset_index(name='Counts')
fig = go.Figure(data=[go.Pie(labels=ed_cnt['EDUCATION'],
values=ed_cnt['Counts'], pull=[0.1, 0.2, 0, 0.1, 0.2, 0,0.1, 0.2, 0,0.1,
0.2, 0.1])])
fig.update_layout(title_text='Overall Education Qualification of all the
Nominees',template='plotly_dark')
```

```
fig.show()
ed_won=ed_valid[ed_valid['WINNER']==1]
ed_win_cnt=ed_won.groupby('EDUCATION').apply(lambda
x:x['PARTY'].count()).reset_index(name='Counts')
fig2 = go.Figure(data=[go.Pie(labels=ed_win_cnt['EDUCATION'],
values=ed_win_cnt['Counts'], pull=[0.1, 0.2, 0, 0.1, 0.2, 0,0.1, 0.1,
0.2,0, 0.1, 0.2],title='Education Qualification of the Winners')])
fig2.update_layout(title_text='Education Qualification of the
Winners',template='plotly_dark')
fig2.show()
```

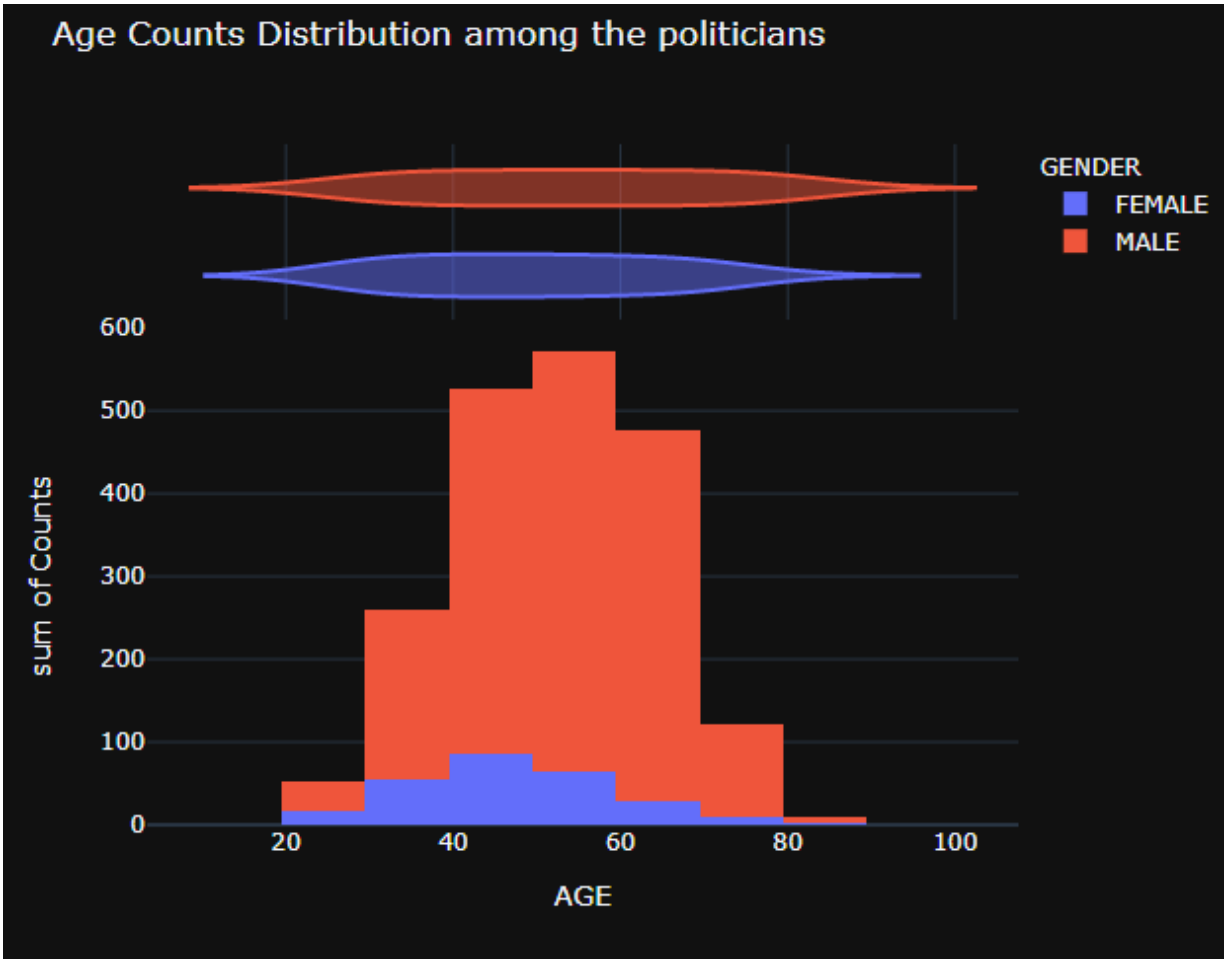


Observation The total percentage of Graduate+ educated people contesting in the election is 67.12%, which has increased to 72.17% of the winners. This is actually a positive sign, as educated politicians are a very big factor towards a country's development. But still around 28% of the politicians have received no professional degree. Hope with passing time, we

improve upon this factor, and consider the educational qualification as a primary requirement while voting!

What is the relationship of Age and Politics?

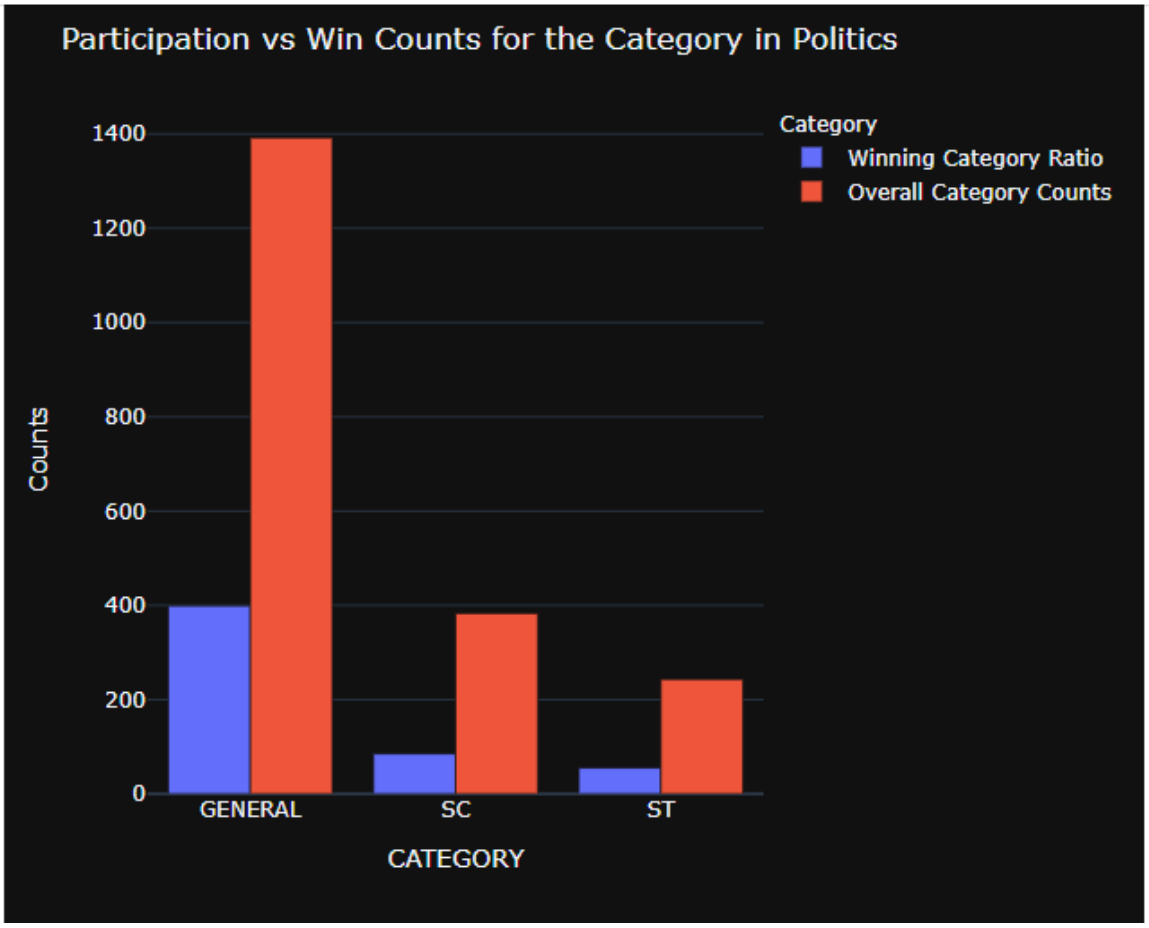
```
In [20]:
age_cnt=ed_valid.groupby(['AGE','GENDER']).apply(lambda
x:x['NAME'].count()).reset_index(name='Counts')
fig = px.histogram(age_cnt,
x="AGE",y='Counts',color='GENDER',marginal='violin',title='Age Counts
Distribution among the politicians')
fig.update_layout(title_text='Age Counts Distribution among the
politicians',template='plotly_dark')
fig.show()
```



Observation Most Number of female politicians have their average age between 45-50, while for male politician, it ranges from 50-60 range. The average age of male politians is more as compared to female politicians contesting for the Lok Sabha elections.

What relation does the Politician category have with the election results?

```
In [21]:
vote_cat=vote[vote['PARTY']!='NOTA']
cat_overall=vote_cat.groupby('CATEGORY').apply(lambda
x:x['NAME'].count()).reset_index(name='Counts')
cat_overall['Category']='Overall Category Counts'
winners_cat=vote_gndr[vote_gndr['WINNER']==1]
cat_winner=winners_cat.groupby('CATEGORY').apply(lambda
x:x['NAME'].count()).reset_index(name='Counts')
cat_winner['Category']='Winning Category Ratio'
cat_overl_win=pd.concat([cat_winner,cat_overall])
fig = px.bar(cat_overl_win, x='CATEGORY', y='Counts',
             color='Category', barmode='group')
fig.update_layout(title_text='Participation vs Win Counts for the
Category in Politics',template='plotly_dark')
fig.show()
```

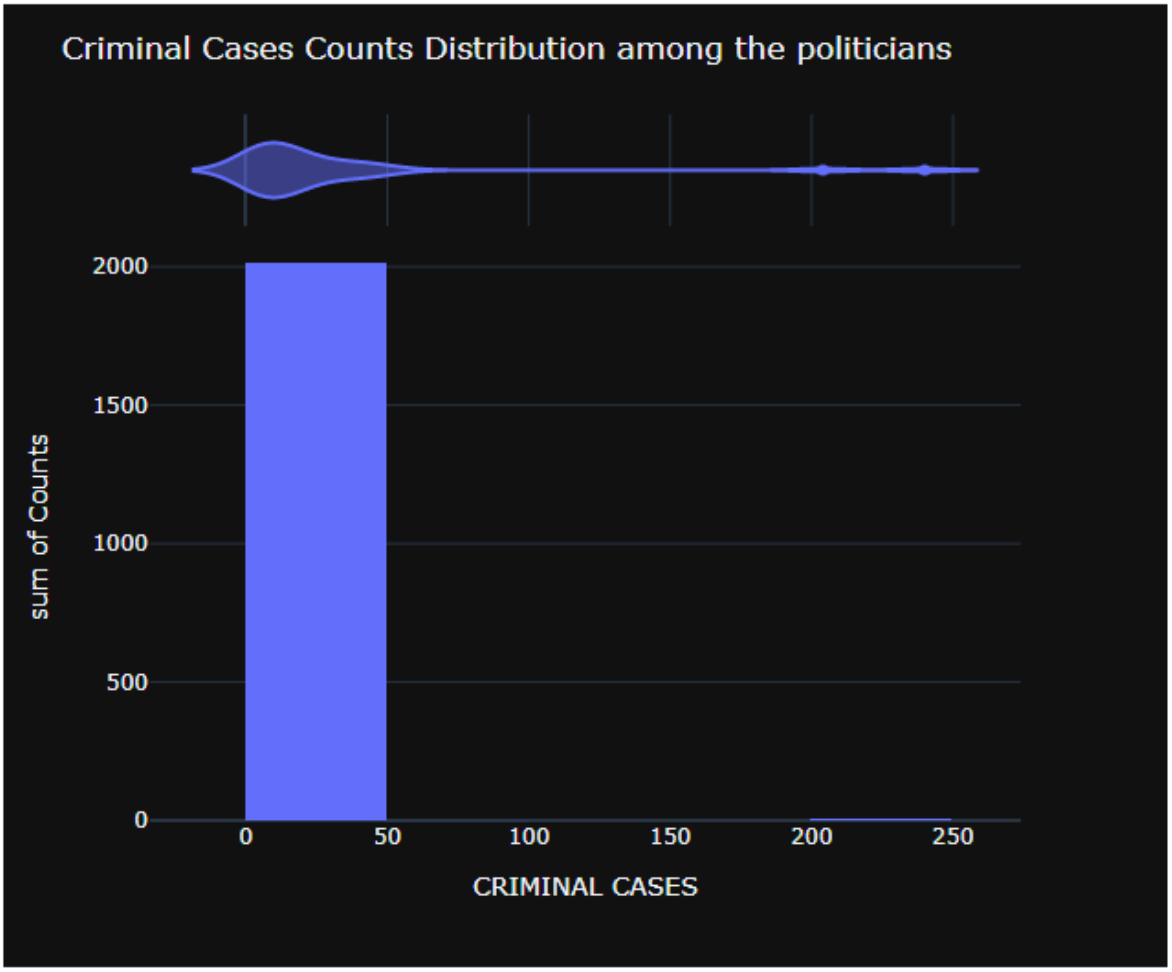


Observation The Category participation of General-SC-ST have been in the ratio of 68.97:18.97:12.04- while as of the winners, the ratios have been modified to 74.02:15.76:10:20

Have the politicians been involved with criminal activities?

In [22]:

```
crim_cnt=ed_valid.groupby('CRIMINAL CASES').apply(lambda
x:x['NAME'].count()).reset_index(name='Counts')
fig = px.histogram(crim_cnt, x='CRIMINAL
CASES',y='Counts',marginal='violin')
fig.update_layout(title_text='Criminal Cases Counts Distribution among
the politicians',template='plotly_dark')
fig.show()
```

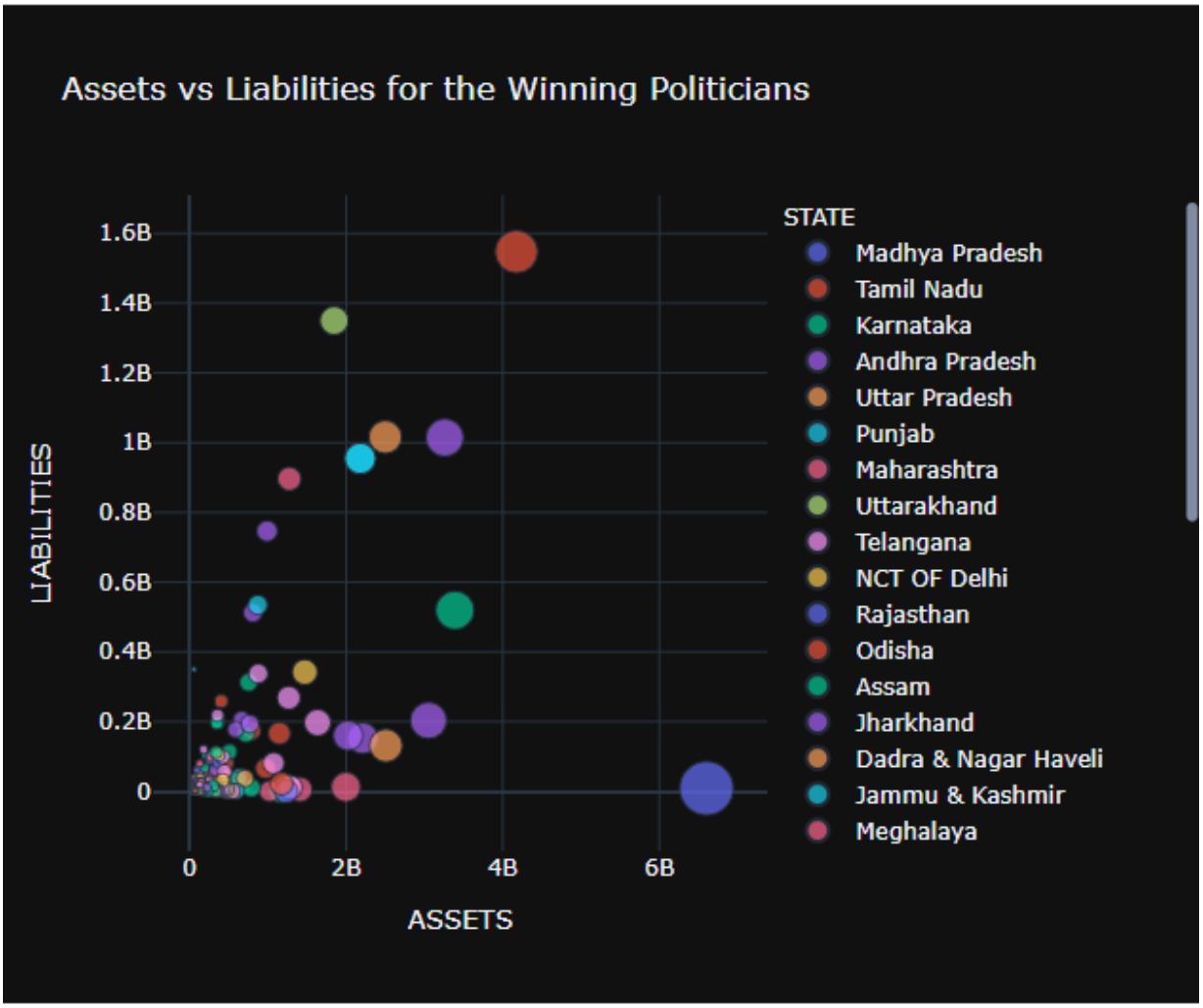


Observations Many politicians have been associated with criminal activities. Always these cases pressed need not be genuine, but obviously, when its multiple- this is a serious issue. We must take the responsibility while voting, as its our duty to choose the right person- as a duty towards the nation.

Plotting the Assets vs Liabilities amount for Winning Politicians (Plotted w.r.t State)

```
In [23]:
as_liab_name=ed_valid[['NAME','PARTY','ASSETS','LIABILITIES','STATE','CO
NSTITUENCY','WINNER']]
as_liab_name.WINNER.replace({1:'Yes',0:'No'},inplace=True)
win_as_liab_name=as_liab_name[as_liab_name['WINNER']=='Yes']
win_as_liab_name.sort_values(by='ASSETS',ascending=False,inplace=True)
fig = px.scatter(win_as_liab_name, x='ASSETS', y='LIABILITIES',
                 color='STATE',size='ASSETS',
```

```
hover_data=['NAME','PARTY','CONSTITUENCY','STATE','WINNER']),
                 title='Assets vs Liabilities for the Winning
Politicians')
fig.update_layout(title_text='Assets vs Liabilities for the Winning
Politicians',template='plotly_dark')
fig.show()
```



Observations The assets and liabilities of the Winning politicians have been plotted. The parameters vary largely depending on the business/services they are associated with besides politics. No valid correlation could be inferred with respect to assets and liabilities

5.3 Method Of Implementation

We shall use Random Classifier to predict the results of the election.

```
vote_df=vote[vote['PARTY']!='NOTA']
vote_df['GENDER'].replace({'MALE':1,'FEMALE':2},inplace=True)
vote_df['CATEGORY'].replace({'GENERAL':1,'SC':2,'ST':3},inplace=True)
i=1
parties_dict={}
for j in vote_df['PARTY']:
    if j in parties_dict:
        continue
    else:
        parties_dict[j]=i
        i+=1
vote_df['PARTY'].replace(parties_dict,inplace=True)
a=1
edu_dict={}
for b in vote_df['EDUCATION']:
    if b in edu_dict:
        continue
    else:
        edu_dict[b]=a
        a+=1
vote_df['EDUCATION'].replace(edu_dict,inplace=True)
df1 = vote_df[['STATE','CONSTITUENCY','WINNER','PARTY','SYMBOL','GENDER','CRIMINAL CASES','AGE','CATEGORY','EDUCATION','TOTAL VOTES','TOTAL
```

ELECTORS','ASSETS','LIABILITIES']]

num_cols = ['PARTY','EDUCATION','CRIMINAL CASES','AGE','TOTAL VOTES','TOTAL ELECTORS','ASSETS','CATEGORY','LIABILITIES','GENDER']

dataset = pd.get_dummies(df1)

from sklearn.preprocessing import StandardScaler

standardScaler = StandardScaler()

scaling_columns = num_cols

dataset[scaling_columns] = standardScaler.fit_transform(dataset[scaling_columns])

dataset.head()

Out[24]:

	WINNER	PARTY	GENDER	CRIMINAL CASES	AGE	CATEGORY	EDUCATION	TOTAL VOTES
0	1	-0.628979	-0.382872	6.620242	-0.023051	2.251127	-1.242514	0.33295
1	0	-0.583879	-0.382872	-0.190426	0.145491	2.251127	-0.872637	0.10378
2	0	-0.538780	-0.382872	0.202498	-0.023051	2.251127	-1.242514	0.08587
4	1	-0.628979	-0.382872	0.464446	0.482577	0.816718	-0.502761	1.38765
5	0	-0.493681	-0.382872	-0.190426	-0.444408	0.816718	-0.872637	0.55976

df_not_winner = dataset[dataset.WINNER == 0]

df_winner = dataset[dataset.WINNER == 1]

df_winner_upsampled = resample(df_winner, replace = True,n_samples = 1452, random_state = 0)

df_total_upsampled = pd.concat([df_not_winner, df_winner_upsampled])

df_total_upsampled.WINNER.value_counts()

y = df_total_upsampled['WINNER']

X = df_total_upsampled.drop(['WINNER'], axis = 1)

rf_scores = []

for k in range(1,60):

randomforest_classifier= RandomForestClassifier(n_estimators=k,random_state=0)

score=cross_val_score(randomforest_classifier,X,y,cv=10)

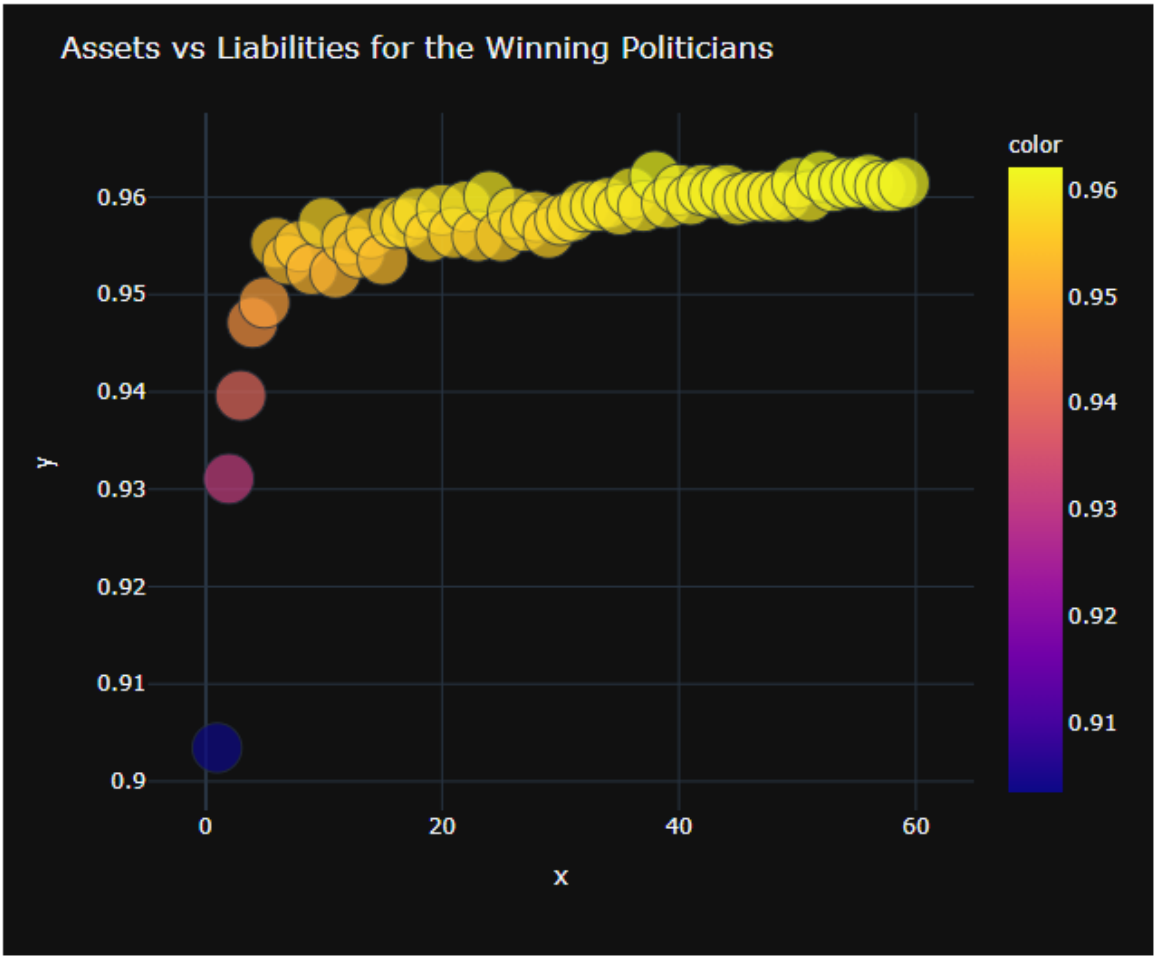
rf_scores.append(score.mean())

fig=px.scatter(x=[k for k in range(1, 60)],y=

rf_scores,color=rf_scores,size=rf_scores)

fig.update_layout(title_text='Assets vs Liabilities for the Winning Politicians',template='plotly_dark')

fig.show()



Observation As seen from the above scatter plot, for tge value of k=38, we get the highest accuracy values. We shall be using the same for our model's prediction

randomforest_classifier=
RandomForestClassifier(n_estimators=38,random_state=0)

score=cross_val_score(randomforest_classifier,X,y,cv=10)

print('% Accuracy :', round(score.mean()*100,4))

5.3.1 Output Screens:

```
% Accuracy : 96.2131
```

Log

```
Time  Line #  Log Message
9.6s       1  /kaggle/input/india-states/Igismap/Indian_States.shp
9.6s       2  /kaggle/input/india-states/Igismap/Indian_States.prj
9.6s       3  /kaggle/input/india-states/Igismap/Indian_States.dbf
9.6s       4  /kaggle/input/india-states/Igismap/Indian_States.shx
9.6s       5  /kaggle/input/indian-candidates-for-general-election-2019/LS_2.0.csv
193.8s     6  % Accuracy : 96.2131
196.5s     7  [NbConvertApp] Converting notebook __notebook__.ipynb to notebook
197.3s     8  [NbConvertApp] Writing 1159023 bytes to __notebook__.ipynb
198.0s     9  [NbConvertApp] Converting notebook __notebook__.ipynb to html
199.2s    10  [NbConvertApp] Support files will be in __results___files/
199.2s    11  [NbConvertApp] Making directory __results___files
199.2s    12  [NbConvertApp] Making directory __results___files
199.2s    13  [NbConvertApp] Writing 656453 bytes to __results__.html
199.2s    14
              Complete. Exited with code 0.
```

5.3.2 Result Analysis

After analyzing the voting pattern of Indians, we have come upon multiple conclusions. Let us look at them in the below points:

1. In 2019, the Bharatiya Janata Party (BJP) has claimed the most number of seats all over India- dominating clearly over all the national and the state specific parties. The outreach created by them has really made them claim their position in the Lok Sabha.
2. Generally, education is regarded as a significant factor while voting. Arounf 72%+ of the winners are having a graduate + degree. This is a significant factor for a developing nation like India- where we need educated politicians to lead the nation.

3. The general participation of Female politicians is much less than the male politicians. A balanced gender ratio would be good enough- as people from all sectors would be able to contribute to the progress of the society.
4. The average age of politicians is high. It might be an advantage if younger politicians contest and are able to lead the nation. Although it is never a parameter to think that the older politicians wouldn't be able to perform their tasks well, it comes with the risk of health and age related issues.
5. Criminal activities are a serious concern- as the people who are elected by us- indirectly are the face of the nation. They should be more of a role model-rather than someone who is not respected.
6. As our prediction suggests, we have reached an accuracy of 96.2% in estimating the winning participants using the Random classifier model. we shall work upon to improve this model further in the future updates

Chapter 6

TESTING AND VALIDATION

6.1 Introduction

TYPES OF TESTS

6.1 UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. It is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration.

6.2 INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

6.3 VALIDATION TESTING

An engineering validation test (EVT) is performed on first engineering prototypes, to ensure that the basic unit performs to design goals and specifications. It is important in identifying design problems, and solving them as early in the design cycle as possible, is the key to keeping projects on time and within budget. Too often, product design and performance problems are not detected until late in the

product development cycle — when the product is ready to be shipped. The old adage holds true: It costs a penny to make a change in engineering, a dime in production and a dollar after a product is in the field.

Verification is a Quality control process that is used to evaluate whether or not a product, service, or system complies with regulations, specifications, or conditions imposed at the start of a development phase. Verification can be in development, scale-up, or production. This is often an internal process.

Validation is a Quality assurance process of establishing evidence that provides a high degree of assurance that a product, service, or system accomplishes its intended requirements. This often involves acceptance of fitness for purpose with end users and other product stakeholders.

The testing process overview is as follows:

6.4 SYSTEM TESTING

System testing of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black box testing, and as such, should require no knowledge of the inner design of the code or logic.

As a rule, system testing takes, as its input, all of the "integrated" software components that have successfully passed integration testing and also the software system itself integrated with any applicable hardware system. System testing is a more limited type of testing; it seeks to detect defects both within the "inter-assemblages" and also within the system as a whole. System testing is performed on the entire system in the context of a Functional Requirement Specification (FRS) or System Requirement Specification (SRS)

Chapter – 7

CONCLUSION & FUTURE WORK

8.1 conclusion

So, Finally I conclude by saying that, this project Election prediction using machine learning is very much useful in political science and it is mainly more important for people in political Arena, because they are the one that mostly effected with this prediction. Politics always plays major role in people life so its important in a great way.In a country of billions, it is extremely difficult to predict the mood of an election if pre-poll predictions are non-existent. They help people in deciphering that their votes go to someone who might actually need it or use it rather than wasting it over a candidate that has no chance standing against the dominant parties. It prevents dividing of votes to a great extent.New party who have never been a player in an area is the underdog whose work and efforts will go unnoticed if the pre-polls did not show chances of their getting some votes and having chances of securing seats.The losing party gets a reality check, and a chance to fight harder and prove the predictions wrong. No wonder promises get bigger as pre-polls are out . A structured approach of combining forecasts from different methods that use different information provides a valuable and simple strategy to achieve that goal.

8.2 FUTURE ENHANCEMENT

- Facility for modifying user detail.
- More interactive user interface.
- Facilities for Backup creation.
- Can be done as Web page.
- Can be done as a Mobile Application.
- More Details and Latest Data

8.References

1. Abramowitz AI(1988) An improved model for predicting presidential election outcomes.PS Polit Sci Polit 21:843-847
2. Munzert S(2017) Forecasting election at the constituency level:a correction-combination procedure.Int J Forecast 33:467-48
3. https://nextbigwhat.com/Indian-election-2019-predictious-by-machine-learning/

4. Machine Learning Methods Used in prediction by www.wikipedia.com
5. https://www.researchgate.net/publication/325116774_election_prediction_usin g_machine_learning_techniques
6. https://ieeexplore.ieee.org/document/8819782/election_prediction Algorithms Details from www.dataspirant.com
7. https://www.youtube.com/election_prediction
8. https://www.slideshare.com/election_prediction

9. https:/ en.wikipedia.org /machine_learning_algorithms
10. https://en.wikipedia.org/wiki/Python_(programming_language)
11. https://wiki.python.org/TkInter
12. https://creately.com/lp/uml-diagram-tool/
13. https://app.diagrams.net/