Mid Semester Project Progress Report

on

(Prediction of Population of India)

Submitted in partial fulfillment for award of

Bachelor of Technology

Degree

in

Computer Science & Engineering

By

Sonali Tiwari 1822210157
Arpit Kumar 1822210032
Kunal Ahlawat 1822210073
Sagar Kumar Verma 1822210125

Under the Guidance of:

(Ms. Akansha Sharma)
(Assistant Professor)

I.T.S ENGINEERING COLLEGE
GREATER NOIDA

MARCH 2021

Project Progress Report

1. Course : Bachelor of Technology
2. Semester : VIIth
3. Branch : Computer Science & Engineering
4. Project Title : Prediction of Population of India
5. Details of Students:

S. No.	Roll No.	Name	Role as	Signature
1	1822210157	Sonali Tiwari	Team leader, Coder	
2	1822210032	Arpit Kumar	Coder	
3	1822210073	Kunal Ahlawat	Coder	
4	1822210125	Sagar Kr. Verma	Documentation	

6. SUPERVISOR:

(Name of Supervisor)

Remarks from Project Supervisor:

……

……

……

……

SYNOPSIS

With its rapidly growing population India is on the verge of touching 1 billion mark. Lockdown due to COVID had its own effects on India. Out of which the most affected ones were economy, usage of resources, population, and mental health of people. All of these are interrelated. And should be analyzed properly so that further precautions and safety measures could be taken.

Population and sex ratio are one of the very old challenges that India is facing. India ranks second in terms of population growth (17.70%)after China with 18.47%. whereas talking about sex ratio according to SRS report in 2018 sex ratio in India was 899 females per 1000 males.

These alarming statistics make it more obvious and important to analyze population and gender ratio in India and its trends. Conversations about overpopulation have become controversial because they beg the question Who exactly is the cause of problem and what if anything should be done about it? Discussing the problems of global overpopulation can never be an excuse or in any way provide a platform for having that type of conversation. Each human being is a legitimate claim on Earths resources. But with a population reaching over 8 billion, even if everyone adopted a relatively low material standard of living, it will still push earth to its ecological breaking point. Unfortunately, the main the problem is that even people are aware about the overpopulation on earth an average person till consumes at a rate of 50% above a sustainable level The aim of the project is to analyze population trend in India and sex ratio trends in largest and smallest (as per population) state if India and build a machine learning model to predict the future statistics.

The project includes usage of various python libraries for extracting useful information from raw data, method of data preprocessing to filter the raw data and training testing data for further predictions and visualizations

ABOUT THE PROJECT

1) The project is about population prediction and analysis of population of India

2) Prediction of population of India in 2022 with the help of past population data

3) Analysis will be done with the help of 2011 census data. Following points will be analyzed-

- A bar chart to show from which states, how many cities are taken for examination.
- A table to show top 10 cities with most population
- State that has most of its population in urban areas
- Top 10 cities with high male population
- Top 10 cities with high female population

- Which state has most of the kids in urban population
- Top 10 cities with high kid's population
- Analyzing literacy rates
- Top 10 cities with most literate female literates live
- Analyzing female literacy rates of states
- Analyzing effective literacy rate
- Analyzing graduates
- Analyzing sex ratio
- Analyzing sex ratio for children below the age of 6

4) After the analysis bar graph for each analysis will be plotted and also the results will be plotted on the map of India. And the final analysis results will be documented.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

The word population comes from the Latin word populous, meaning “the people”. It is used to refer a group of people living in a particular area, such as city, country, continent, or the world. Few aspects of human societies are as fundamental as size, composition and rate of change their populations. Such factors affect economic well-being, health, education, family structure, crime patterns, language and culture. The study of human population is called demography. On the other hand, talking about the emerging developments in technologies, there are various tech fields like Artificial Intelligence and Machine learning which are emerging as potential technologies for machines or computer to imitate human intelligence and make their work easier. Nowadays machine learning is broadly used to predict data and improvise further consequences in advance. Since the resources are limited there is vital need to study population growth in a trustworthy manner in order to necessary steps towards control of overpopulation and hence prioritizing sustainable development.

This project uses machine learning algorithm to predict population growth and study population trends in India for a better understanding population composition which enables future leaders of the country to prepare for natural resources allocation and work upon Indian economy more efficiently.

1.1 OVERVIEW OF THE POPULATION GROWTH

Most of the growth in the world population has taken place in the modern era. The time required for the world population to reach its first billion stretched through all of the human prehistory into the early 1800s. The second billion was added in 1930, and the 3 billion mark was reached by 1960. The fourth, fifth, sixth billion marks were added in 1974,1990,1999. The seven billion milestone was reached in 2011.

Population growth is the increase in the number of people in a population. In very basic words we can say that population growth is change in size of population. It can be positive or negative. It is affected by number of births and deaths. Comparatively high number of deaths can result in very slow or declined population growth.

Stabilized population is very important because we cannot have a sustainable planet without stabilized population. A very high population can result in higher usage of resources like plants

and animal (for food), water resources, energy, land, etc. which may further lead them to be endangered or even extinct

Population growth can be more efficiently depicted by population growth rate which is given by formula

Rate = $(P_2-P_1)*100 / P_1$

P_2 :- population(time2),

P_1 :- population(time1)

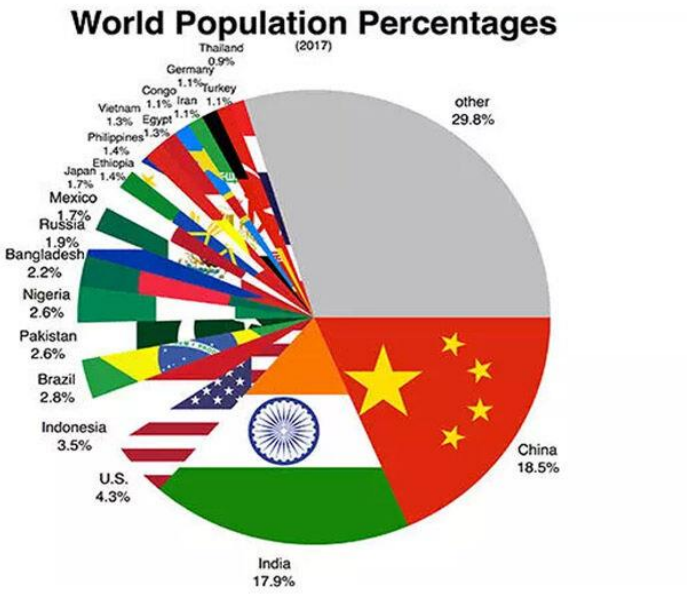

Fig1.1.1: World Population Percentages

1.2 POPULATION GROWTH IN INDIA

India ranks 2nd in terms of population growth with a score 17.70% of world's population after China which has a score of 18.47%. India's current population is estimated as 1.3 billion with growth rate of 1% which means India will contribute the addition of 13 million more in the world's population. The reason for such high population growth are (i) high birth rate and (ii) low death rate. The factor that mostly contribute to this is lack of family planning. Family planning is not practiced sincerely on a large scale, especially in rural areas. This situation has resulted into large proportion of youth (15 to 24 years, 2%) along with aged (32% in 2011) who are dependent on a relatively small workforce of the population. It is estimated that due to 25 million people are homeless and 171 million people have no access to safe drinking water. This

increase in population growth at increasing rate has caused major problems like malnutrition, poverty**.** In a country where 50 million people live on less than USD 2 a day and nearly 200 million people are undernourished, the growing population will only make the food security situation worse. Our ambition to transform into a world power will remain only an aspiration if we have such a large population living in poverty. Any photograph of India is full of teeming crowds tightly packed together in a small space or a narrow street or a railway station. This is the image of India in the world.

To change this image analysis and working upon required field is very important and necessary

Fig 1.2.1 Increasing Growth Rate of Indian Population

1.3 Objective of Studying Population Trends

Since population has a very large impact on the economy and development of any country therefore it is very important to study and understand population trends in order to take correct decision. Study of population can be in the form of studying gender composition, educational analysis, employment details and many more The very first thing to determine while analyzing population is determining which states and cities make the most of it. Which further indicates that we have to classify or arrange the states and cities in order of most populated to least populated. This which the data will be arranged from most to least in terms of gender division, educational area etc.

These process helps students to engage in understanding of resources and its uses. Multiple studies have helped to facilitate the quality of life. The results when shown to people may help in educating them about population growth.

These arguments prove that study related to population is useful in following ways

- To enable students to understand that family size is controlled.

- That population limitation can facilitate the development of a higher quality of life in the nation.
- That a small family size can contribute materially to the quality of living for the individual family
- To enable students to appreciate the fact that for preserving the health and welfare of the members of the family and to ensure good prospects for the younger generation, the Indian family of today and tomorrow should be small and compact
- To give accurate information to the students about the effect of family size and in national population on the individual.

CHAPTER 2

DATA DOWNLOAD

2.1 Data Download

The first step towards analyzing data is downloading dataset from a reliable site.We downloaded data from kaggle.com. the following steps were involved for the same.

1) Go to kaggle.com and type the dataset you want to find in search bar

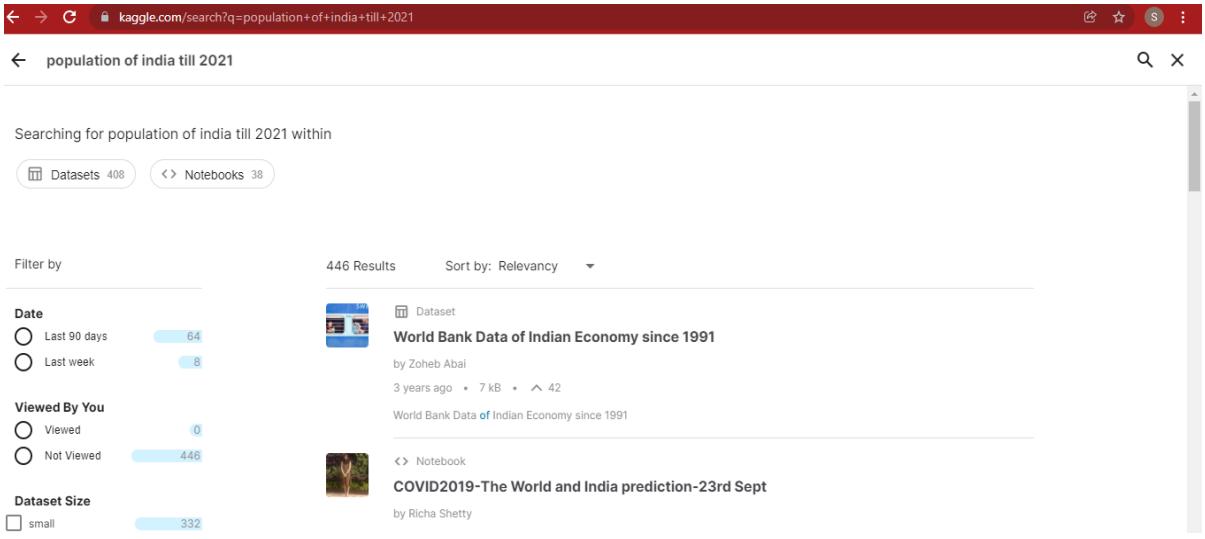

Fig 2.1.1: Search bar in kaggle.com

2) Select the most relevant result from the list.

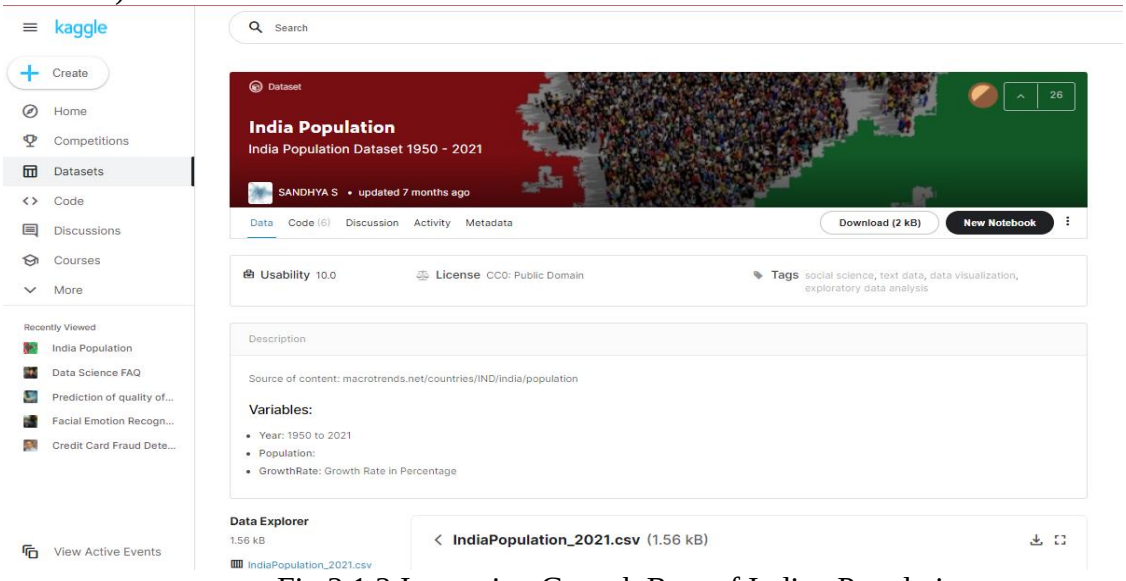

Fig 2.1.2 Increasing Growth Rate of Indian Population

3) Download the dataset

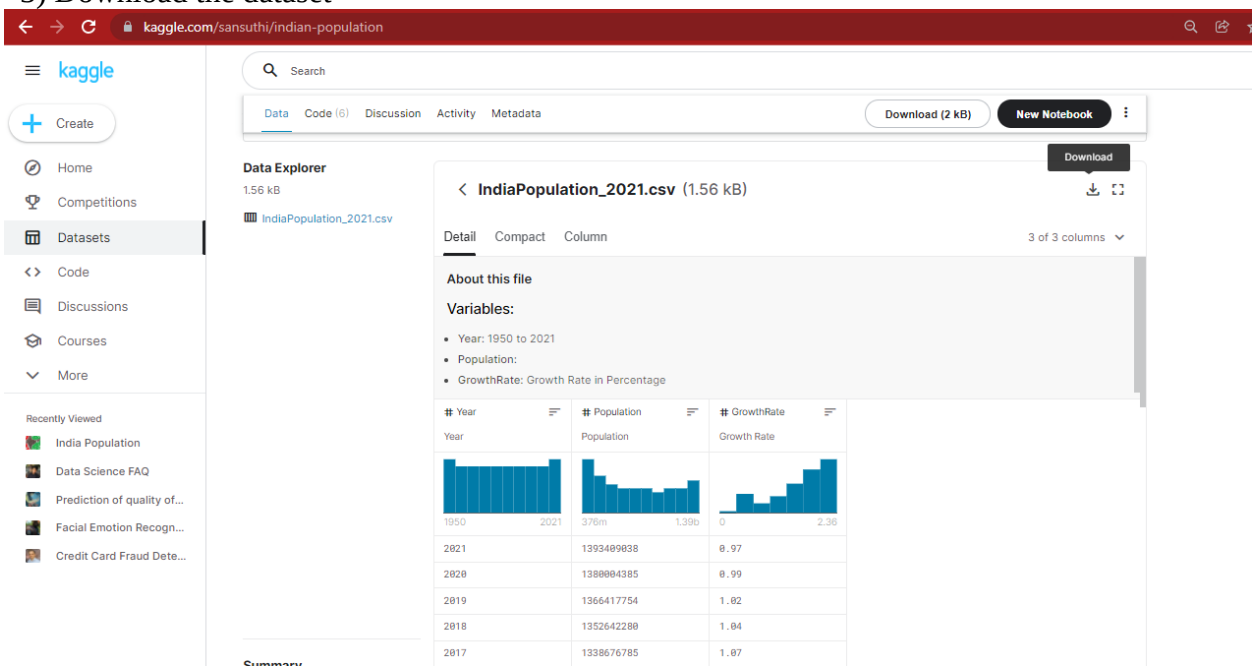

Fig 2.1.3: Downloading Data

2.2 Initial Data Analysis

Initial data analysis consists of variety of steps. These include Data Collection, Data cleaning, Data processing etc. First step in data analysis is importing all the necessary libraries that we will be using in future for preprocessing, predictions, calculations, training and testing, visualization etc. The libraries will be imported in jupyter environment using python language. We will be using following libraries

i) PANDAS: It is a software library written for the Python programming language. It is basically used for data manipulation and analysis.

ii) MATPLOTLIB PYPLOT: It is a comprehensive library for creating static and interactive visualization in python. And all the functions in pyplot makes some changes to a figure like create a figure, create plotting area in figure etc.

```
In [1]: import pandas as pd
        import matplotlib.pyplot as plt
```

Fig 2.2.1: Code for importing libraries

CHAPTER 3
DATA CLEANING AND PREPROCESSING

3.1 Importing data

The next step is to start working upon data to make it clean so that the further results of predictions are correct and significant. We will import data in our jupyter notebook from where the data was originally downloaded. The library used for this particular step is pandas which was called as “pd” when we imported all the libraries in first step.

After this the data is copied into another variable called “df”. This step was taken because while data preprocessing several editing steps will be taken. Copying data in another variable makes sure we have original data with us for future references. The functions used in importing data and copying data frame in another variable are read () and copy (). In the project population in the year 2022 will be predicted with the help of population of India from 1960 to 2021.Following is the short overview of the dataframe

1.Year: - It consists of entries of each and every year consecutively from 1950 to 2021.

2. Population: - It means total number of people who reside in geographical area of India legally according to the counting done by government of India. In this Andaman and Nicobar Islands are also included.

```
In [12]: df
```

Out[12]:

	Year	Population
0	2021	1393409038
1	2020	1380004385
2	2019	1366417754
3	2018	1352642280
4	2017	1338676785
...	...	...
67	1954	402578596
68	1953	395544369
69	1952	388799073
70	1951	382376948
71	1950	376325200

72 rows × 2 columns

Fig 3.1.1 – Overview of dataframe

3.2 Data Cleaning

Data cleaning refers to the process of deleting unnecessary elements present in the data in order to reduce number of errors during our predictions. Data cleaning may include deletion of unnecessary columns like serial numbers, dropping the column with null values, dropping rows and columns within significant or misleading data. To get started with data cleaning the very first thing to check is if there are null values in our data frame. Null value refers to the missing values that exist in data frames. Missing values can be referred to as NA i.e. not available values in Pandas whereas sometimes we may get datasets or data frames with missing values either it was not collected or it never existed.

In Pandas missing data is represented by two values:

None – None is a python singleton object that is often used for missing data in Python code

Nan – Nan is an acronym for Not a Number. It is a special floating point value recognized all systems that use the standard IEEE floating point representation.

When for a large amount of data if there is a small percentage of missing values then the result may remain unaffected but if the data is in small amount or there is a large percentage of missing values in comparatively large dataset then missing data creates imbalanced observations, cause biased estimates, and in extreme cases, can even lead to invalid conclusions.

We will check null values in data frame with the help of isna () function of panda library. Isna function helps to detect the null values that may be in the form of nan, none etc. It takes array like object and indicates whether there are missing values or not. Further if these values existed then to count these values we used sum () function. According to the online definition sum function sums up the numbers in the list.

```
In [13]: df.isna().sum()

Out[13]: Year          0
         Population    0
         dtype: int64
```

Fig 3.2.1- Output for checking null values

3.3 Splitting data

After cleaning the data, we will prepare the data to fit perfectly in our machine learning algorithm. The data that we are provided for prediction has two columns first is year and second is population so we will split the data in two variables vertically. For this we will use iloc function of pandas. It helps us to retrieve a particular value belonging to a row or column using index value assigned to it. The syntax of the iloc function is as follows

dataframe. iloc[:,start_col: end_col]

After implying the function, the data will be splitted into X and Y vertically. Year will go into variable X and population will go in Y variable. These further will be used in machine learning algorithm. Splitting the data helps us to fit the data into train and test model more accurately. This will lead the prediction to be more correct and with less error.

CHAPTER-4

MACHINE LEARNING MODEL

4.1 Train and Test Split

After the initial data analysis and data preprocessing we will imply the machine learning model on our data. To imply our machine learning model first, we will perform train and test split. The train-test split procedure is used to estimate the performance of machine learning algorithms when they are used to make predictions on data not used to train the model. The data that was previously splitted into X and Y variables will be distributed into train and test data frames separately. So after which there will be four data sets namely x_train, x_test, y_train and y_test. For this step we will import train_test_split from sklearn.model_selection

Train_test_split - It split arrays or matrices into random train and test subsets.

Sklearn.model_selection – sklearn stands for scikit library in python. Scikit learn is a collection of classes which helps to generate list of train test arrays.

After importing the model from the library we will split X and Y into train and test the syntax for which will be

X_train, X_test, Y_train, Y_test= (X, Y, test_size=__, random_state=__)

- **X_train** – The subset that will be trained by algorithm
- **X_test** – The subset that will be provided as input when we will test the algorithm
- **Y_train** – these will be used in training as expected output while training X_train.
- **Y_test** – the output received from testing X_test will be compared to these values to evaluate correctness of our predictions.
- **Test_size** – It refers to the percentage between 0 to 1 of data we wish to keep as test size. The value used in the project is 0.33 which means data will be divided as 67% train and 33% test
- **Random_state** – This ensuring that we get the same data in train and test otherwise if we run the code multiple times every time the data will differ.

4.2. Implying the Machine Learning Model

The most important part of making predictions using a data frame is applying the machine learning model to our training set. After the analysis of data, we found out that there are two variables in the data frame first is year which is independent variable and second is population which is dependent variable. Population is considered as dependent variable because we have to predict the population and the variable for which we have to make predictions is considered as dependent variable. According to this we will use **Linear Regression model** and since there are only two variables we will particularly use Simple Linear Regression Model.

Simple linear regression is a statistical **method** that allows us to summarize and study relationships between two continuous (quantitative) variables. It works on the line equation y=mx+c. where y is dependent variable, x is independent variable, m is slope of line and c is y intercept of line.

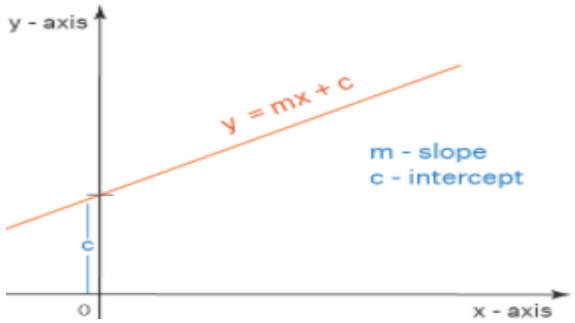

Fig 4.2.1- Graph for y=mx+c

For implying the model, we will first import Linear Regression class from scikit learn library. Linear Regression is a machine learning algorithm based on supervised learning. It performs a regression task. Regression models a target prediction value based on independent variables. After importing linear regression, we will create an instance of the class which can also be termed as regressor.

Regressor - In statistics, a regressor is the name given to any variable in a regression model that is used to predict a response variable.

Now to fit the training data frame in our model we will use fit () function and to predict values we will use predict () function.

Fit () function- Fit () method will fit the model to the input training instances

Predict () function - Perform predictions on the testing instances, based on the learned parameters during fit.

4.3. Coefficient of Determination

Coefficient of determination or R squared is a statistical measurement which tells us how much of variation of Y is described by the variation of X. Which may further mean that it tells us how good our fit is. This measure is represented as a value between 0.0 and 1.0, where a value of 1.0 indicates a perfect fit, and is thus a highly reliable model for future forecasts, while a value of 0.0 would indicate that the model fails to accurately model the data at all. In the project we calculated the coefficient of determination with the help of score () function. Score function of scikit library helps us to calculate R squared or coefficient of determination. It takes x_test and y_test as argument and returns r^2 as output. The output came out as 0.98730 which means our fit is 98% accurate.

```
In [17]: slr_score=std_reg.score(x_test,y_test)
         slr_score

Out[17]: 0.9873805044598396
```

Fig 4.3.1- Output for Coefficient of Determination

CHAPTER – 5
FINAL PREDICTION AND VISUALIZATION

5.1. Equation of Line

Final prediction which means to predict the population in 2022 will be done with the help of the equation of line. First of all, we will calculate the slope of line, then we will calculate the y intercept of line and we will substitute the value in the equation y=mx+c. For this we will use following functions-

Coef_ ()- It returns a two dimensional array with a single element m which is the slope of line or coefficient of features of our dataset.

Intercept_ - It returns a single dimensional array with an element b_0 which is y intercept or we can say constant of the equation. Intercept means the point on y axis where the line intersects the y axis.

```
In [18]: #coefficient of line
         slr_coefficient = std_reg.coef_
         #intercept of line
         slr_intercept=std_reg.intercept_

In [19]: slr_coefficient

Out[19]: array([15290166.77274535])

In [21]: slr_intercept

Out[21]: -29528419953.984055
```

Fig 5.1.1- Output for intercept and coefficient

Equation of line y= 15290166*x + (-29528419953)

Fig 5.1.2- Equation of line

After forming the equation of line we will put value of x as 2022 because initially data was divided as X (year) and Y(population). After calculation we will get value of Y which will be population in the year 2022.

```
In [23]: -29528419953 + 15290166*2022

Out[23]: 1388295699
```

Fig 5.1.3- Final Prediction

According to data provided by www.worldmeter.info the current population of India is 1,401,309,721 as of 28 January 2022.

5.2. Visualization

Data visualization is the process of understanding data and its inferences by placing it in a visual context so that the patterns or trends that were if ignored or left are then understood by person analyzing it. The visual context may include graphs like bar graph, charts like pie chart or plot like box plot. The library used for visualization is matplotlib.pyplot. This library helps the user to visualize the data into various graphs. for example, scatterplots, bar graphs, histograms etc.

```
In [24]: plt.scatter(x_test,y_predict)
         plt.plot(x_test,y_predict)
         plt.ylim(ymin=0)
         plt.show
```

Fig 5.2.1 Code for Plotting Predicted Values

Plt- matplotlib was imported as plt in earlier stages of code.

Scatter ()- it is a function used to specify that the user wants to plot a scatterplot. It takes the values to be plotted on the graph as argument

Ylim () – It gives the minimum value for y axis**.**

Giving x_test and y_predict as arguments shows that we want to plot x_test that is the values in the year that were not trained and y_predict that is the values (predicted population) that were predicted by our machine learning model with the help of our test values.

```
Out[24]: <function matplotlib.pyplot.show(close=None, block=None)>
```

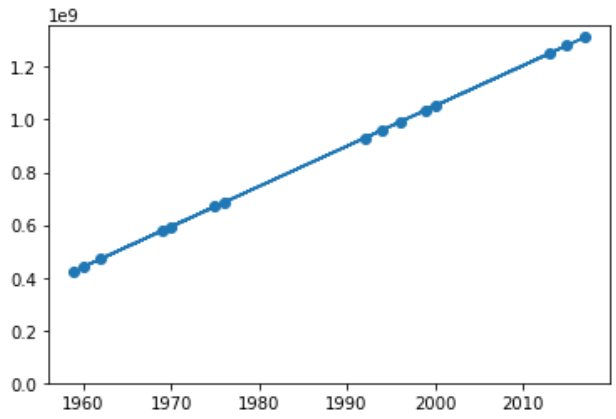

Fig 5.2.2 Scatterplot for Predicted Values

```
In [25]: plt.scatter(x_test,y_test)
         plt.plot(x_test,y_test)
         plt.ylim(ymin=0)
         plt.show
```

```
Out[25]: <function matplotlib.pyplot.show(close=None, block=None)>
```

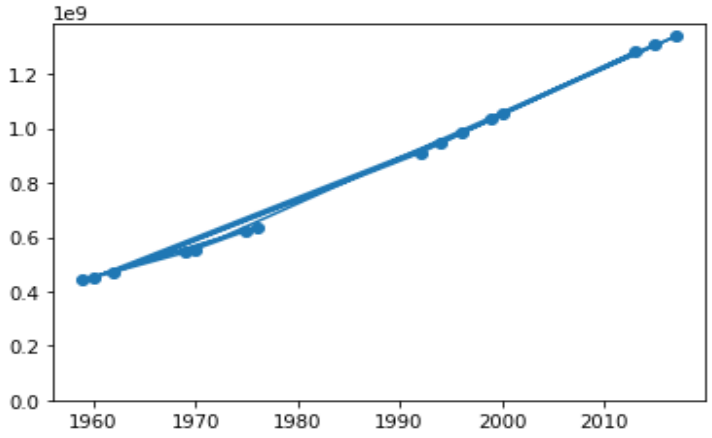

Fig 5.2.3 Code and Scatterplot for Actual Test Values

These visualizations help us to compare the predicted values and actual values in order to view the accuracy of predictions. However, the visualizations may not be the best way to check accuracy but it is great to have a direct look onto to similarities of prediction and actual values.

To check the accuracy, methods like r squared or coefficient of determination are the best choices.

CHAPTER 6
DATA ANALYSIS(PART2)

The part 2 of the analysis project consists of exploratory data analysis of top 500 cities with population above one lakh according to 2011 census of India. Various factors that will be analyzed will be education, gender distribution, top ten most populated cities. These will be then arranged in proper order from most to least or vice versa. This part will also contain plotting the analyzed result on the map of India with the help of python language.

6.1. Data preview

The data taken for exploratory data analysis is that of 500 hundred cities with population of over one lakh according to 2011 census of India. It has over 500 rows and 22 columns. Description of the data as follows

nameofcity - Name of the city

statecode - State code of the city

statename - State name of the city

distcode - District code where the city belongs

populationtotal - Total population of the city

populationmale - Male population

populationfemale – Female population

0-6populationtotal – 0 to 6 age Total population

0-6populationmale – 0 to 6 age Total Male population

0-6femalepopulation – 0 to 6 age Total Female population

literatestotal – Total literates

literatesmale – Male Literates

literatesfemale – Female Literates

sexratio – Sex Ratio

childsexratio – Sex ratio in 0 to 6

effectivelitercayratetotal – Female rate over age 7

effectiveliteracyratemale – Male literacy rate over age 7

effectiveliteracyratefemale – Female literacy rate over age 7

location – Latitude, Longitude

totalgraduates – Total graduates in the city

femalegraduates – Total female graduates in the city

malegraduates- Total male graduates

6.2. Importing Libraries

There are various modules that are pre-defined in python and are stored in various libraries. In order to use the functions in these modules we need to import the libraries. Importing libraries in our code means we can use the pre-defined functions. Following libraries will be used in the code-

PANDAS- It is a software library written for the Python programming language. It is basically used for data manipulation and analysis

NUMPY-numpy is a python library used for working with arrays. It also has functions for working in domain of linear algebra, Fourier transform, and matrices. To work specifically with arrays, we have imported ARRAY module from numpy library.

MATPLOTLIB PYPLOT- It is a comprehensive library for creating static and interactive visualization in python. .

COLORMAPS- It is a matrix of values that define the colors for graphic objects such as surface, image, and patch objects. This will be used to color differentiate different cities in Indian maps

DATE2NUM- This module is used to convert date time objects into matplotlib dates. It will be used in time series graphs plotting.

SEABORN- It is used to plot statistical graph.

```
In [1]: # importing packages
        import pandas as pd
        import numpy as np
        from numpy import array
        import matplotlib as mpl

        # for plots
        import matplotlib.pyplot as plt
        from matplotlib import cm
        from matplotlib.dates import date2num
        from mpl_toolkits.basemap import Basemap

        # for date and time processing
        import datetime

        # for statistical graphs
        import seaborn as sns
```

Fig 6.2.1- Code for Importing Libraries

6.3. Data Cleaning

The very first step in data analysis is reading the data file. The data file is in the form of comma separated file. We will read it using pandas library which was imported as pd. The data will be read in a variable called cities, which means all the data in the data file will copied into the variable cities. The function used for reading the file is **read ()** of pandas library. The argument passed in the read function is the name by which the data file is saved or stored in our folder within double quotes. Further the analysis will be done on the data present in the variable itself.

```
In [67]: cities = pd.read_csv ("cities_r2.csv")
         cities
```

Out[67]:

	name_of_city	state_code	state_name	dist_code	population_total	population_male	population_female	0-6_population_total	0-6_population_male	0-6_popul
0	Abohar	3	PUNJAB	9	145238	76840	68398	15870	8587	
1	Achalpur	27	MAHARASHTRA	7	112293	58256	54037	11810	6186	
2	Adilabad	28	ANDHRA PRADESH	1	117388	59232	58156	13103	6731	
3	Adityapur	20	JHARKHAND	24	173988	91495	82493	23042	12063	
4	Adoni	28	ANDHRA PRADESH	21	166537	82743	83794	18406	9355	
...	...	...	...	...	...	...	...	...	...	
488	Vizianagaram	28	ANDHRA PRADESH	12	227533	111596	115937	20487	10495	
489	Warangal	28	ANDHRA PRADESH	9	620116	310400	309716	55392	28434	
490	Wardha	27	MAHARASHTRA	8	105543	53241	52302	9754	5139	
491	Yamunanagar	6	HARYANA	3	216628	115404	101224	22905	12556	
492	Yavatmal	27	MAHARASHTRA	14	116714	58717	57997	11081	5894	

493 rows × 22 columns

Fig 6.3.1 – Overview of Data frame and read function

After finalizing the data to be worked upon we will check for the null values. By null values we mean the missing values in the dataset. These missing values affect the analysis and hence if present we either drop the corresponding row or column. The null values will be checked by the function **info ()**. The info function gives basic information about the dataset. The basic

information includes number of rows, number of columns, datatype of all the attributes, number of missing values. The syntax followed will be "name of dataset.info ()

```
In [68]: cities.info ()
         # there is no null values anywhere in the dataset

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 493 entries, 0 to 492
Data columns (total 22 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   name_of_city                    493 non-null    object
 1   state_code                      493 non-null    int64
 2   state_name                      493 non-null    object
 3   dist_code                       493 non-null    int64
 4   population_total                493 non-null    int64
 5   population_male                 493 non-null    int64
 6   population_female               493 non-null    int64
 7   0-6_population_total            493 non-null    int64
 8   0-6_population_male             493 non-null    int64
 9   0-6_population_female           493 non-null    int64
 10  literates_total                 493 non-null    int64
 11  literates_male                  493 non-null    int64
 12  literates_female                493 non-null    int64
 13  sex_ratio                       493 non-null    int64
 14  child_sex_ratio                 493 non-null    int64
 15  effective_literacy_rate_total   493 non-null    float64
 16  effective_literacy_rate_male    493 non-null    float64
 17  effective_literacy_rate_female  493 non-null    float64
 18  location                        493 non-null    object
 19  total_graduates                 493 non-null    int64
 20  male_graduates                  493 non-null    int64
 21  female_graduates                493 non-null    int64
dtypes: float64(3), int64(16), object(3)
memory usage: 84.9+ KB
```

Fig 6.3.2 – Basic Information of Data

The final step to complete the initial data analysis is to check all the statistical components of the data that we are working upon. Statistical components include mean, median, quartile range, standard deviation, count of the data. It helps us to understand correlation among data. And complete statistical visualization and obtain various inferences about data before detailed analysis of data.

```
In [4]: cities.describe ()
```

Out[4]:

	state_code	dist_code	population_total	population_male	population_female	0-6_population_total	0-6_population_male	0-6_population_female	literates_total
count	493.000000	493.000000	4.930000e+02	4.930000e+02	4.930000e+02	4.930000e+02	493.000000	493.000000	4.930000e+02
mean	18.643002	16.782961	4.481124e+05	2.343468e+05	2.137656e+05	4.709285e+04	24849.527383	22243.320487	3.461527e+05
std	9.297168	15.566131	1.033228e+06	5.487786e+05	4.848622e+05	1.050279e+05	55535.310272	49523.241379	8.220952e+05
min	1.000000	1.000000	1.000360e+05	5.020100e+04	4.512600e+04	6.547000e+03	3406.000000	3107.000000	5.699800e+04
25%	9.000000	7.000000	1.261420e+05	6.638400e+04	6.041100e+04	1.363900e+04	7221.000000	6457.000000	9.768700e+04
50%	19.000000	13.000000	1.841330e+05	9.665500e+04	8.776800e+04	1.944000e+04	10342.000000	9172.000000	1.413290e+05
75%	27.000000	21.000000	3.490330e+05	1.750550e+05	1.700260e+05	3.794500e+04	19982.000000	17954.000000	2.679000e+05
max	35.000000	99.000000	1.247845e+07	6.736815e+06	5.741632e+06	1.209275e+06	647938.000000	561337.000000	1.023759e+07

Fig 6.3.3- Statistical Component of Data

CHAPTER 7
VISUALIZATION

Visualization refers to the process of converting numerical or categorical data into graphs or plots so that it is easier for user to understand hidden trends of data. It also helps in easy understanding of data and also check for outliers. Outliers are the values that are totally different from trends that the values are following. Since this project is just a detailed exploratory analysis of different populous cities of different cities in India, we will plot various bar graphs. Also with these graphs we will be plotting various cities on map of India using libraries like seaborn, matplotlib and basemap.

7.1 State wise Selection of Cities

We will start the visualization process by plotting a bar graph about number of cities taken from each state. This will help us to know that maximum cities are taken from which states. This may further help us to understand whether or not the basis of selection of cities and state is population wise, area wise or any other else parameter. For plotting this bar graph we will use **matplotlib** library which was imported as plt earlier. Figure function is called and the dimension of 15x15 is passed as an argument. Next line of the code will be declaration of the variable that will be plotted on y axis that is state. The state variable is declared and a function **groupby ()** is used where the arguments name_of_city and state_name is passed. This signifies that all the cities are grouped according to state that they belong. And the number of cities for each state will be the respective value of the state which will further be plotted on y axis and will be counted using **count ()** function. Also these values are presorted in ascending order using **sort ()** function with ascending parameter as True. Further **plot ()** function is used to specify details of the type of graph we are plotting so the type of the graph is barh which means bar graph. We have also used grid function which used to display grid lines in the graph. After which the x and y label are defined which includes the title of x and y axis and variables which will be plotted on respective axis which are states on y axis and number of cities on x axis.

```
# A bar chart to show from which states, how many cities are taken for examination.
fig = plt.figure(figsize=(15,15))
states = cities.groupby('state_name')['name_of_city'].count().sort_values(ascending=True)
states.plot(kind="barh", fontsize = 15)
plt.grid()
plt.xlabel('No of cities taken for analysis', fontsize = 15)
plt.ylabel('State name', fontsize = 20)
plt.show ()
```

Fig 7.1.1 – Code for plotting bar graph (state wise selection)

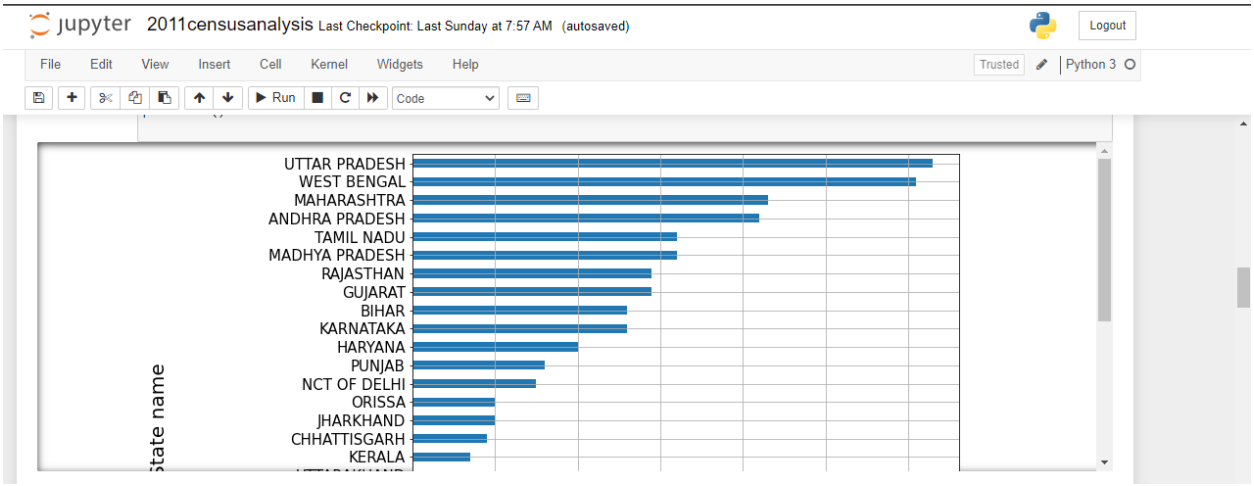

Fig 7.1.2 – Bar graph of state wise selection of cities

The graph shows highest number of cities taken for analysis vs state name in which the highest bar goes for Uttar Pradesh accompanied by West Bengal. This means these two states were given high priority by taking most of the number of cities. Although this does not mean that the city with highest population may be from any of these states.

7.2 Top 10 populated cities

Next we will analyze top 10 populated cities. This means we will extract 10 cities that are most populated starting from highly populated which means in descending order. This will be done with the help of sort values function in python. Sort values function helps to sort values in ascending or descending order. In the code two arguments were given. The first is by value. This specifies how to data will be sorted. That is on which basis. The value given by argument is population _ total which is a column in the cities dataset which represents total population of that respective city taken for analysis. After the by value a Boolean value is given. The value is false

given to the variable called Ascending. This means the data sorted will be in descending order. After which the result is printed which is as follows.

The Top 10 Cities sorted according to the Total Population (Descending Order)

	name_of_city	state_code	state_name	dist_code	population_total	population_male	population_female	0-6_population_total	0-6_population_male
185	Greater Mumbai	27	MAHARASHTRA	99	12478447	6736815	5741632	1139146	599007
141	Delhi	7	NCT OF DELHI	99	11007835	5871362	5136473	1209275	647938
72	Bengaluru	29	KARNATAKA	18	8425970	4401299	4024671	862493	444639
184	Greater Hyderabad	28	ANDHRA PRADESH	99	6809970	3500802	3309168	725816	373794
7	Ahmadabad	24	GUJARAT	7	5570585	2935869	2634716	589076	317917
119	Chennai	33	TAMIL NADU	2	4681087	2357633	2323454	418541	213084
274	Kolkata	19	WEST BENGAL	16	4486679	2362662	2124017	300052	155475
449	Surat	24	GUJARAT	25	4462002	2538243	1923759	531522	293208
380	Pune	27	MAHARASHTRA	25	3115431	1602137	1513294	324572	171152
225	Jaipur	8	RAJASTHAN	12	3073350	1619280	1454070	378788	204320

Fig 7.2.1- Top 10 populated cities list

After extracting the top 10 populated cities we will plot them on the map of India. For plotting these cities, we will use matplot library and base map. To start with we will begin with creating a subplot of size 20 x 15. After which we will create an instance of base map with basic features of the plot. To plot the map of India we will give width value which gives width of desired map domain in projection coordinates in meters, height which gives height of desired map domain in projection coordinates (meters), the Lambert conformal conic projection as type of projection, and latitude and longitude details which will be given in 4 sections Upper left corner, Upper right corner, Lower left corner, Lower right corner and at last center of map domain in form of lon_o and lat_o. This will help to create the outer boundary of India.

Now to plot the top 10 populated cities we will use scatterplot with base as map of India we will first pass the longitude and latitude values in the instance of base map which will further be assigned to variable x and y. After which these values will be passed in the scatter function of matplot library along with other specification like the size will be on the basis of population size the marker shape will be circle, color of the marker will be on the basis of population sizes which means that different sizes will have different color.

Now after plotting these cities we will name the respective cities. For this a for loop is used iterating upon name of the city and longitude and latitude of plotted cities. For this for loop we will use all these three arrays in a zip function as the iteration will be parallel. And within this for loop the text will be printed. However, some features have been specified for the text. These include the position of the text which is some centimeters away from actual plotted city and the font that will be bold. At last the title of the figure is given. Following is the resultant map-

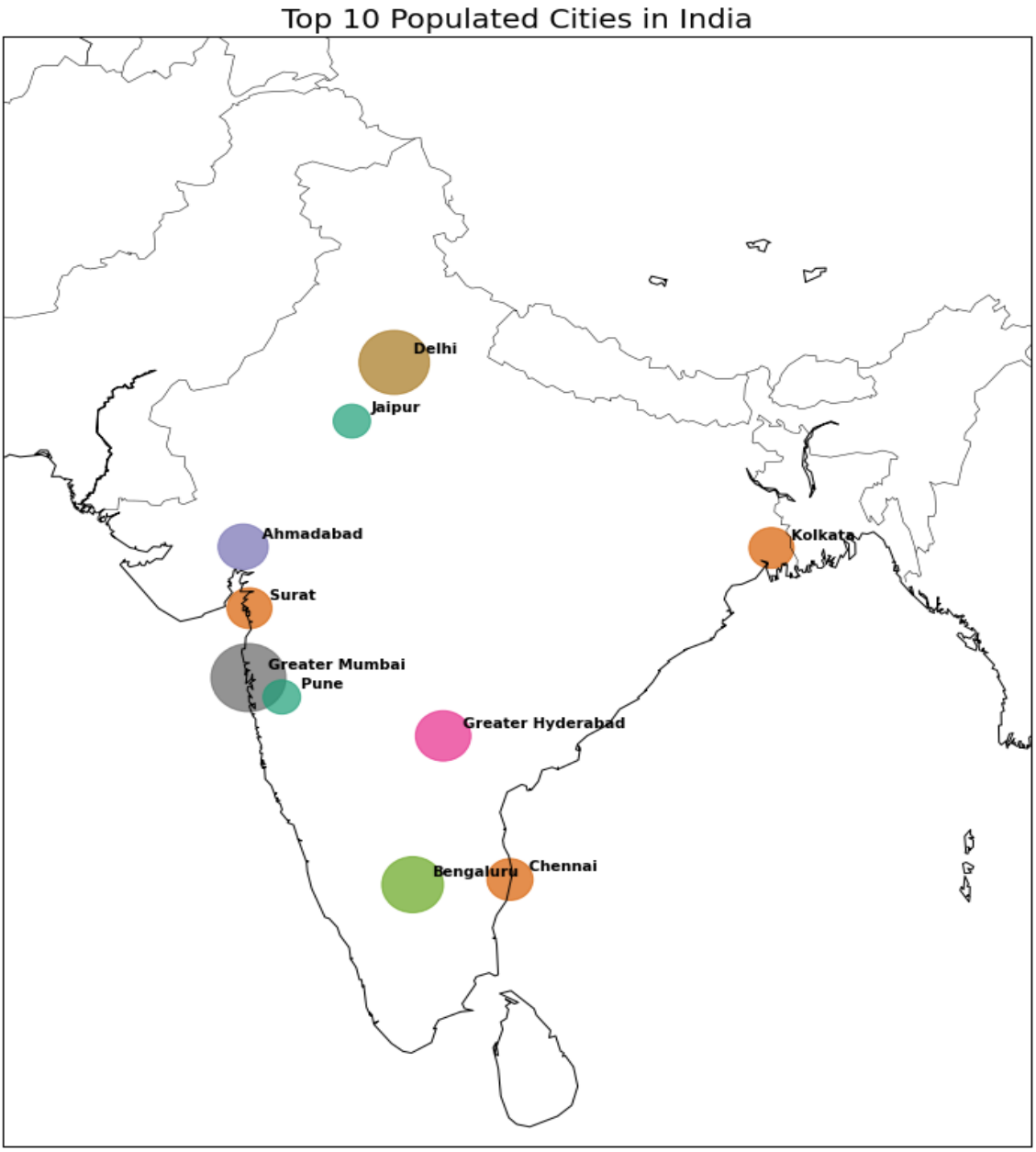

Fig 7.2.2- Top 10 populated cities Map

7.3 State wise Urban Population

In this section we will plot a bar graph which will tell us about how much population is residing in urban areas briefly called as urban population of any state. The data set taken has only the urban population in cities of India. For which we will check which state has maximum urban population. And which state has minimum urban population. Studying urban population tells us about cities, processes of urbanization and urban lifestyle. This can directly be related to educational and career development. Also can be helpful in allocating resources effectively to the much needed state for an equality in development. This results in growth of economy and hence happiness score of a particular country.

For this purpose, we will plot bar graph using matplot library. The very first step is to give the size of the figure using figure function of matplot. The argument given in parenthesis will be fig size which is size of the figure. The size is given as 20*20. Next the cities dataframe is grouped into state of the basis of population total. And the results are stored in a variable called states. Then the variable states are plotted into bar graph using plot function. Plot function is a function in mat plot library which helps to plot different kinds of graph like scatterplot, bar graph, boxplots etc. In this section we pave specifies the type as barh which means bar graph and another parameter is font whose value is given as 20. This is for labeling of x and y axis. At last title of the plot is given.

Plotting Statewise cities to check which state have most population living in urban areas

```
In [57]: # A bar chart to show the population of the states
         fig = plt.figure(figsize=(20,20))
         states = cities.groupby('state_name')['population_total'].sum().sort_values(ascending=True)
         states.plot(kind="barh", fontsize = 20)
         plt.xlabel('No of cities taken for analysis', fontsize = 20)
         plt.show ()
```

Fig 7.3.1- Code for sate wise urban population

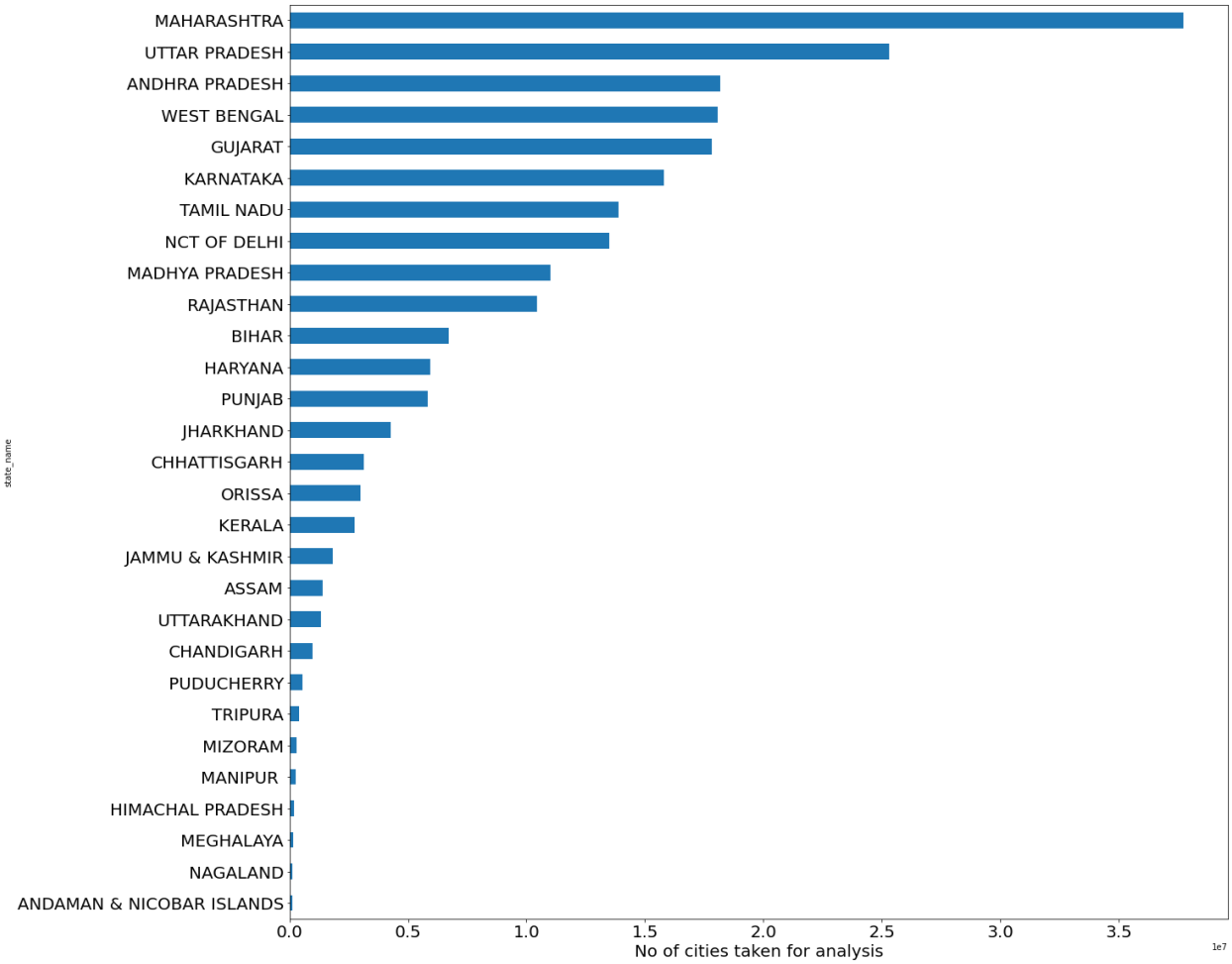

Fig 7.3.2- Bar Graph for sate wise urban population

From above bar graph we can see that states like Maharashtra and Uttar Pradesh have maximum population living in urban areas.

7.4 Plotting every city

In this section we will roughly plot every city on the map of India on the basis of population. For this first we will create a function for plotting India map. The function will named as plot_map and will take size and color bar value as arguments. Color bar value will help in setting the position of color bar. Color bar converts scalar value into colors. The chart which tells which color corresponds to which value limit. Size indicates the data which needs to be plotted and color bar indicates color specifications. The function includes almost every detail required to plot the map of India. For giving these details we will create an instance of basemap function

which will be named as map. Basemap is a tool to create map using python in a very simple way. It is matplotlib extension which has features like data visualization, adding geographical projections, drawing coastlines of various country. It will include the figure size, setting up of base map with longitude and latitude corner and center values, resolution, projection, height and width. After the setting the base map the required coordinates will be plotted and in the last color values are given. Now we will plot every city taken in the data set on the map of India. The basis of this projection will be population. This means the size of the circle marker of the scatterplot will be on the basis of the population of that particular city. After this we will add colorbar. This will define different color values for a range of population. Colorbar function is applied on map and the parameters in this functions are given as position of the colorbar that is either on right or on left and the padding which means how much distance away it will be away from our plot. After this we will use set_yticklabel function in which the text frames that appear with major ticks on the axes of the colorbar. The text will be given with colorbarvalue. It will be defined when we will call the function.

For the code part we will call the plot map function and will give the respective arguments. For sizes population sizes column from cities data frame is given. For color bar value linspace function of numpy library is used. Linspace function gives scalar vector values between two values. It takes various parameters but in this section we will need only three, the starting point which is the minimum value in population sizes, the end point which is the maximum value and num which is 10. This means that all the value between the start and end point will be divided evenly into 10 parts. These ten parts will signify different colors for different population sizes. But set label function gives values in string data type so we will convert the values into integer data type using as type data function. Astype is a function in pandas library. It is used to convert from one datatype into another datatype.

Following this we will a map of India in which almost all the cities are plotted according to their respective population. The marker is of circle shape; whose size depends on population. And the color of the circle will help us to put different cities in a particular population range.

Following are the output.

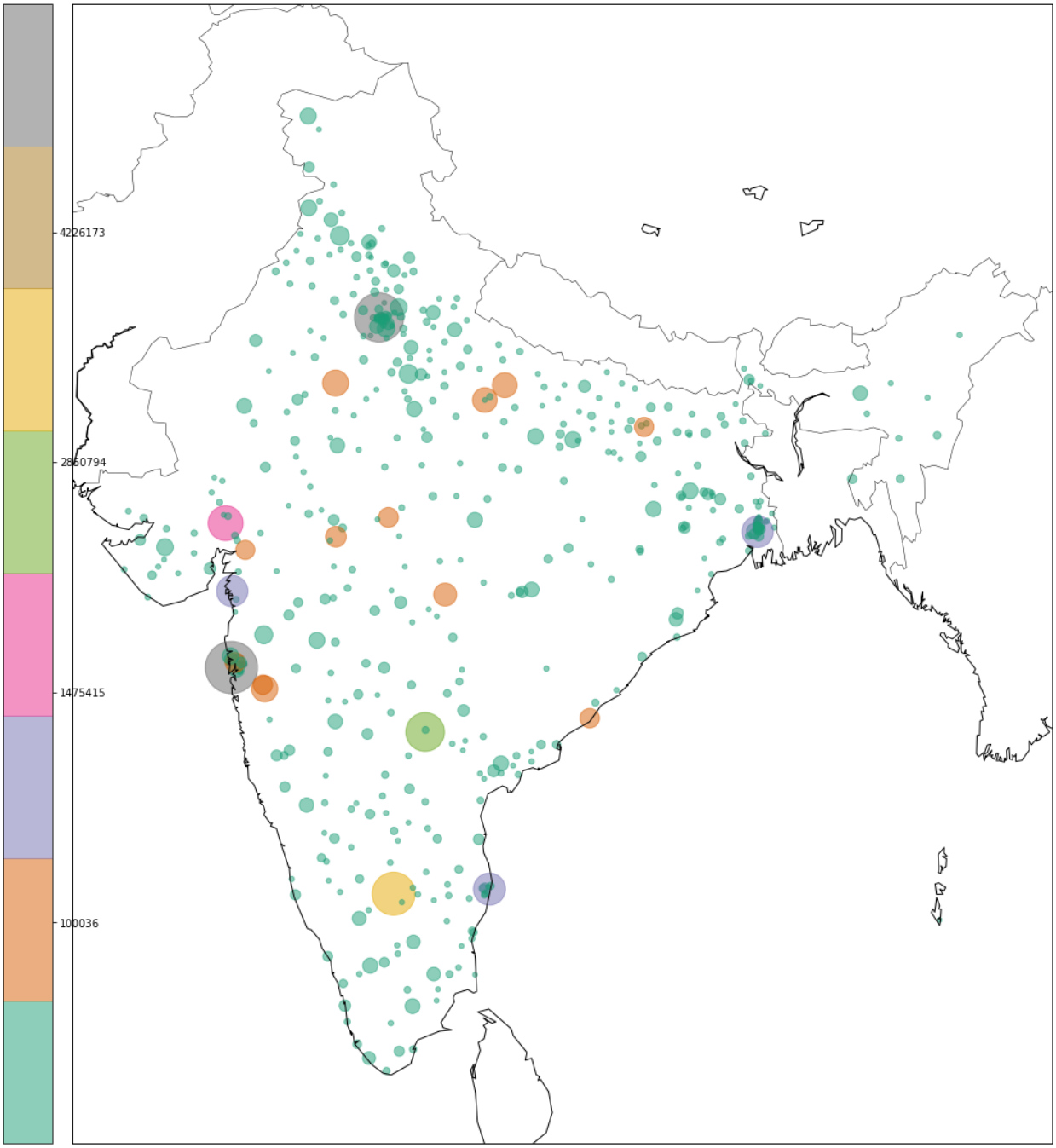

Fig 7.4.1- All the cities on the basis of population

7.5 Male population of all the cities

In this section we will elaborate on male population of different states and cities. Studying of male population may help in improvisation of sex ratio in India as it has been a major problem for decades. The stories of discrimination have been known by all. Analyzing and presenting this information may have a significant impact on people's mind. The analysis will be on the basis of state with highest male population, top ten cities with high male population using bar graph and then plotting the outcomes. Male population will include adult and minor above 6 years of age. For kids of 0 to 6 age we will make different observations. First we will plot a bar graph for all the states using group by function. The basis of the projection will be the population. First of all, we will give the size of the figure using matplot library. In the next line we will create a variable named as state. It will contain list of all the states which will be grouped by male population column in cities dataset. This means that all male population values in different cities of a particular state will be added and then will be arranged in ascending order using sort function. Next we will plot a bar graph using plot function. The parameters given in this function are type which tells which type of graph we want to plot and font size of x and y labels.

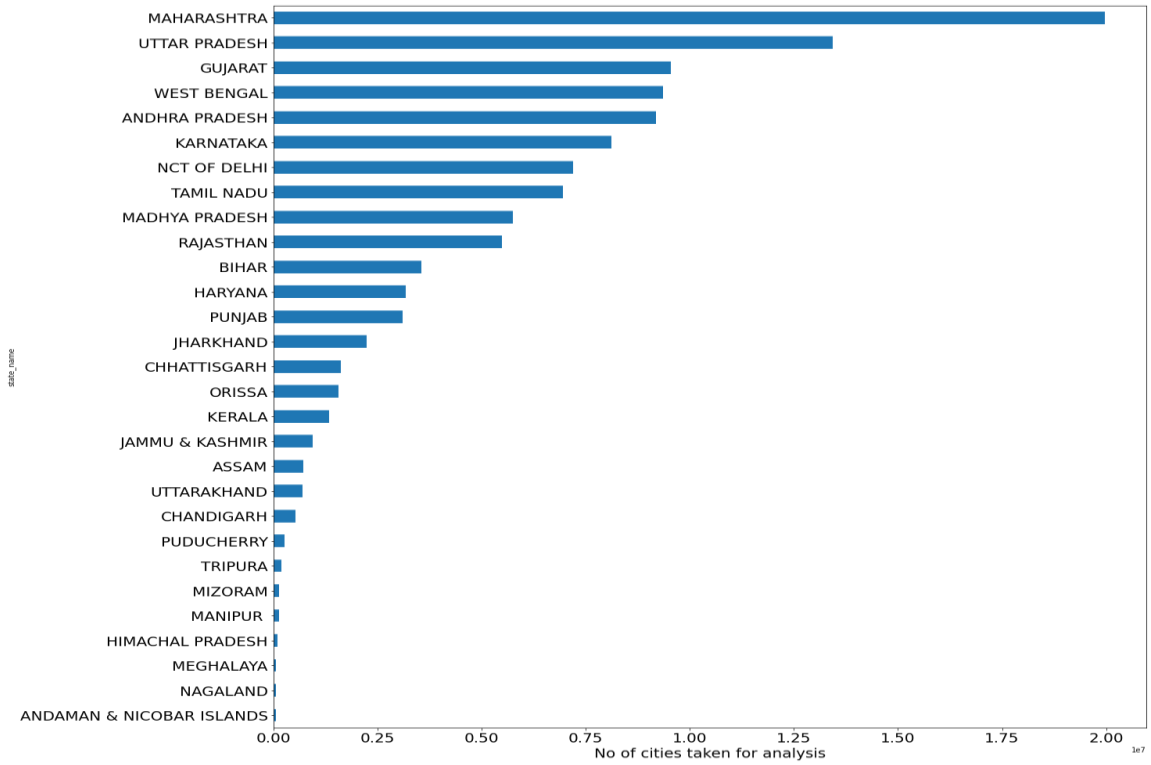

Fig 7.5.1- Bar graph for state wise population of men

Now we will plot all the cities on the basis of male population of each and every city. We will use the plot_map function for plotting this map. The arguments however will be different. For sizes we will give total male population of the city. For the color bar the start and end point will be minimum of male population and maximum of the male population. After which we will change the datatype with the help of astype function.

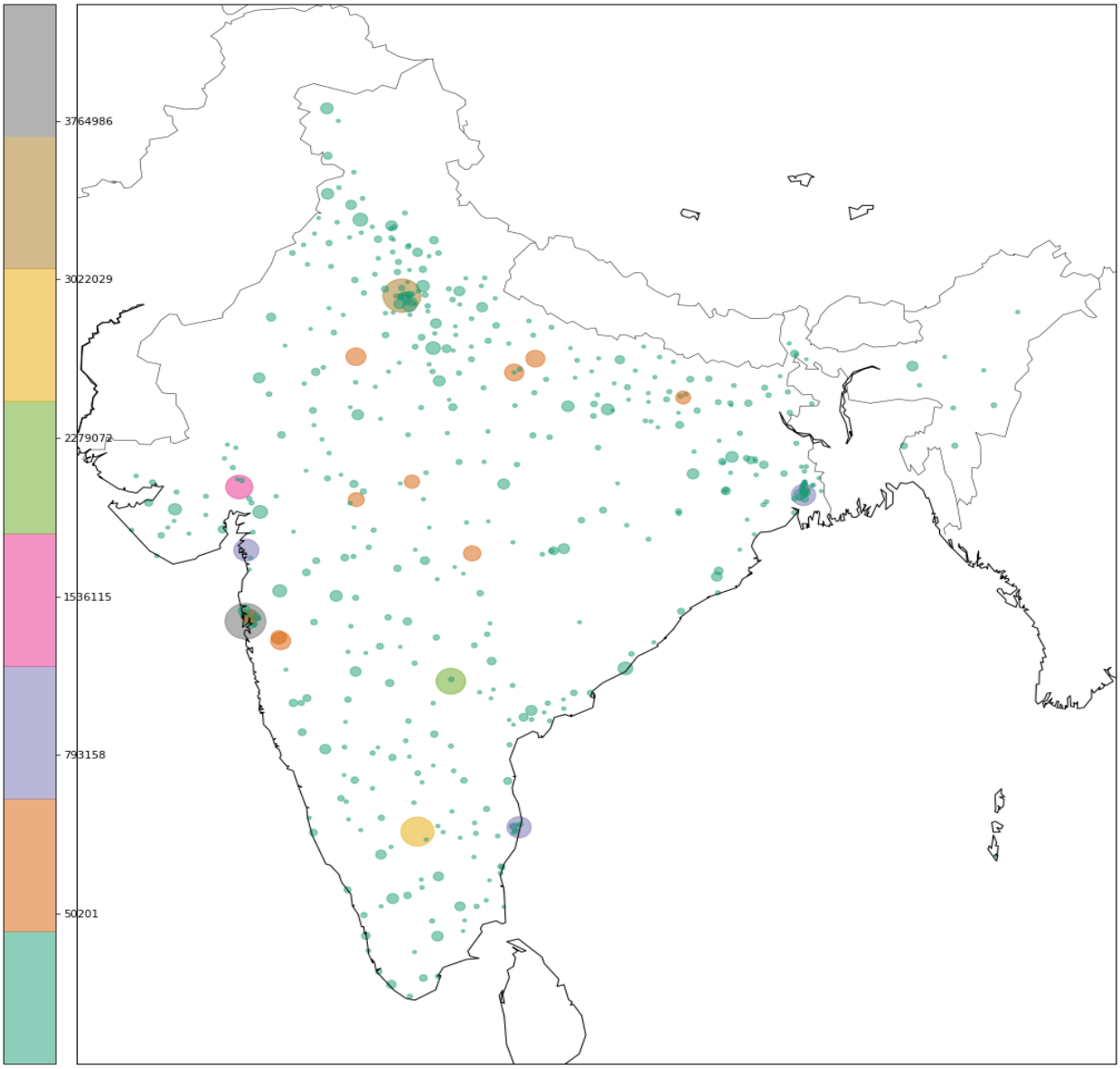

Fig 7.5.2- All the cities on the basis of male population

7.6 Top 10 male populated cities

In this section we will see top 10 cities with highest male population. First of all, we will sort the male population column from dataset in descending order which is highest to lowest and then will extract first ten cities data. These will be top 10 cities with highest male population. After this we will plot these cities on map of India. For extracting the name of cities we will define a variable top_male_cities in which there will be the list of cities and their data which will be sorted on the basis of male population with the help of sort_value function and to ensure that the data is sorted in descending order we will give Boolean value False to ascending. After this we will select first 10 values in different variable called top top10_male_pop_cities. After extracting the names of the cities we will plot these cities on the map of the India using the basemap extension of matplot library. The coordinates given will be longitude and latitude of the respective cities.

```
The Top 10 Cities sorted according to the Total Population (Descending Order)
```

	name_of_city	state_code	state_name	dist_code	population_total	population_male	population_female	0-6_population_total	0-6_population_male
185	Greater Mumbai	27	MAHARASHTRA	99	12478447	6736815	5741632	1139146	599007
141	Delhi	7	NCT OF DELHI	99	11007835	5871362	5136473	1209275	647938
72	Bengaluru	29	KARNATAKA	18	8425970	4401299	4024671	862493	444639
184	Greater Hyderabad	28	ANDHRA PRADESH	99	6809970	3500802	3309168	725816	373794
7	Ahmadabad	24	GUJARAT	7	5570585	2935869	2634716	589076	317917
119	Chennai	33	TAMIL NADU	2	4681087	2357633	2323454	418541	213084
274	Kolkata	19	WEST BENGAL	16	4486679	2362662	2124017	300052	155475
449	Surat	24	GUJARAT	25	4462002	2538243	1923759	531522	293208
380	Pune	27	MAHARASHTRA	25	3115431	1602137	1513294	324572	171152
225	Jaipur	8	RAJASTHAN	12	3073350	1619280	1454070	378788	204320

Fig 7.6.1- Top 10 cities with highest male population

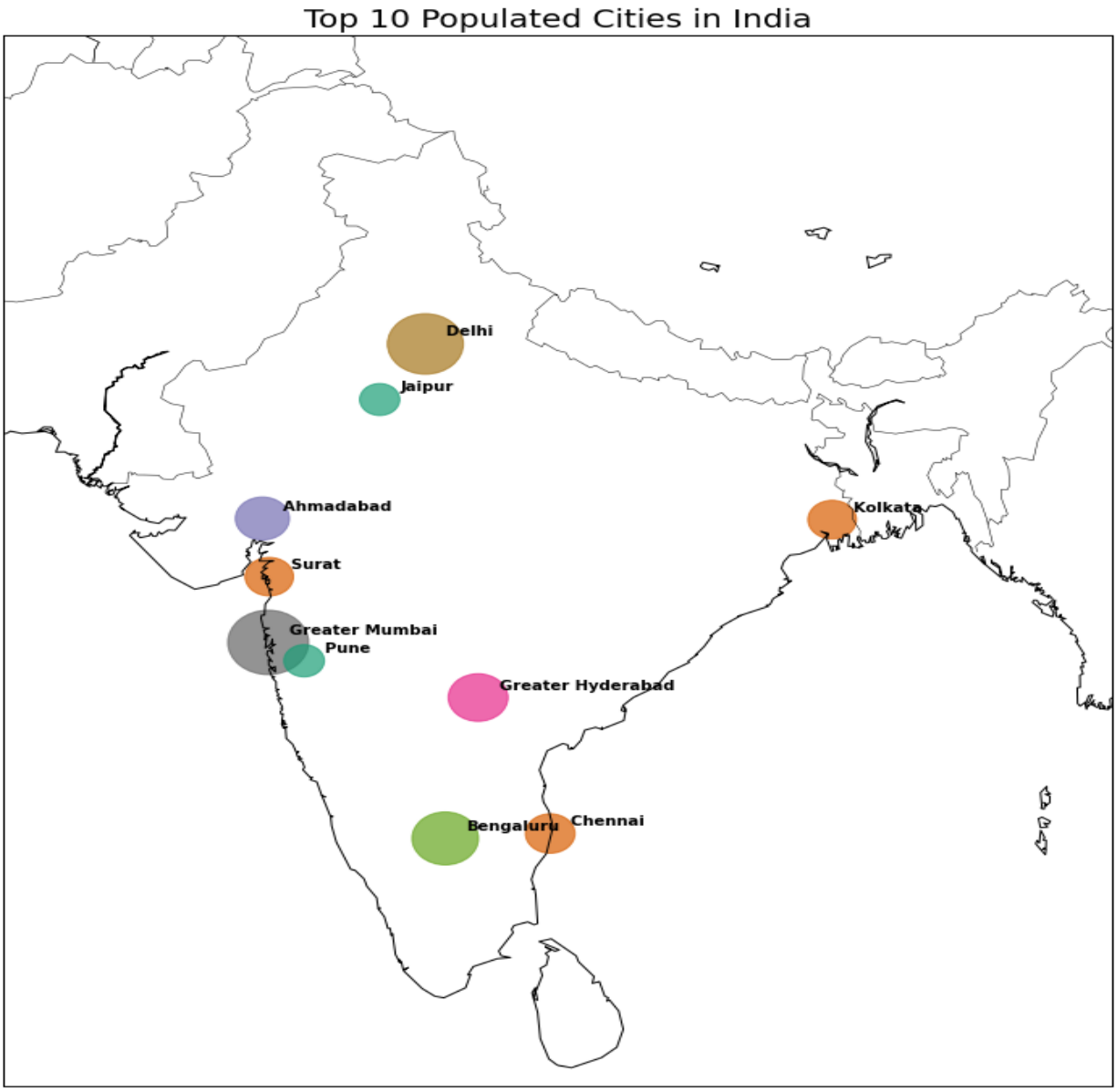

Fig 7.6.2- Top 10 cities with highest male population

7.7 Female Population

In this section we will analyze the female population both state wise and city wise. First step will be plotting bar graph for the female population which will be grouped on the basis of state name. Population_female column in cities dataset will be grouped into state name using groupby function. Size of the figure is 20*20 and font size is also 20. Following is the output for bar graph after which we will plot these cities on the map of India.

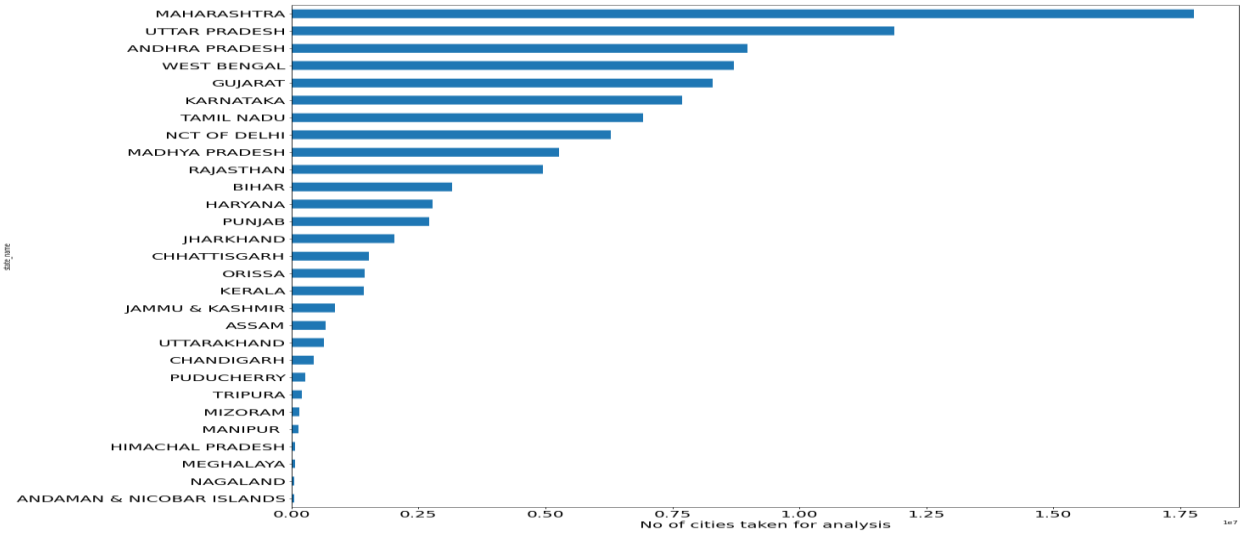

Fig 7.7.1- Bar graph for female population

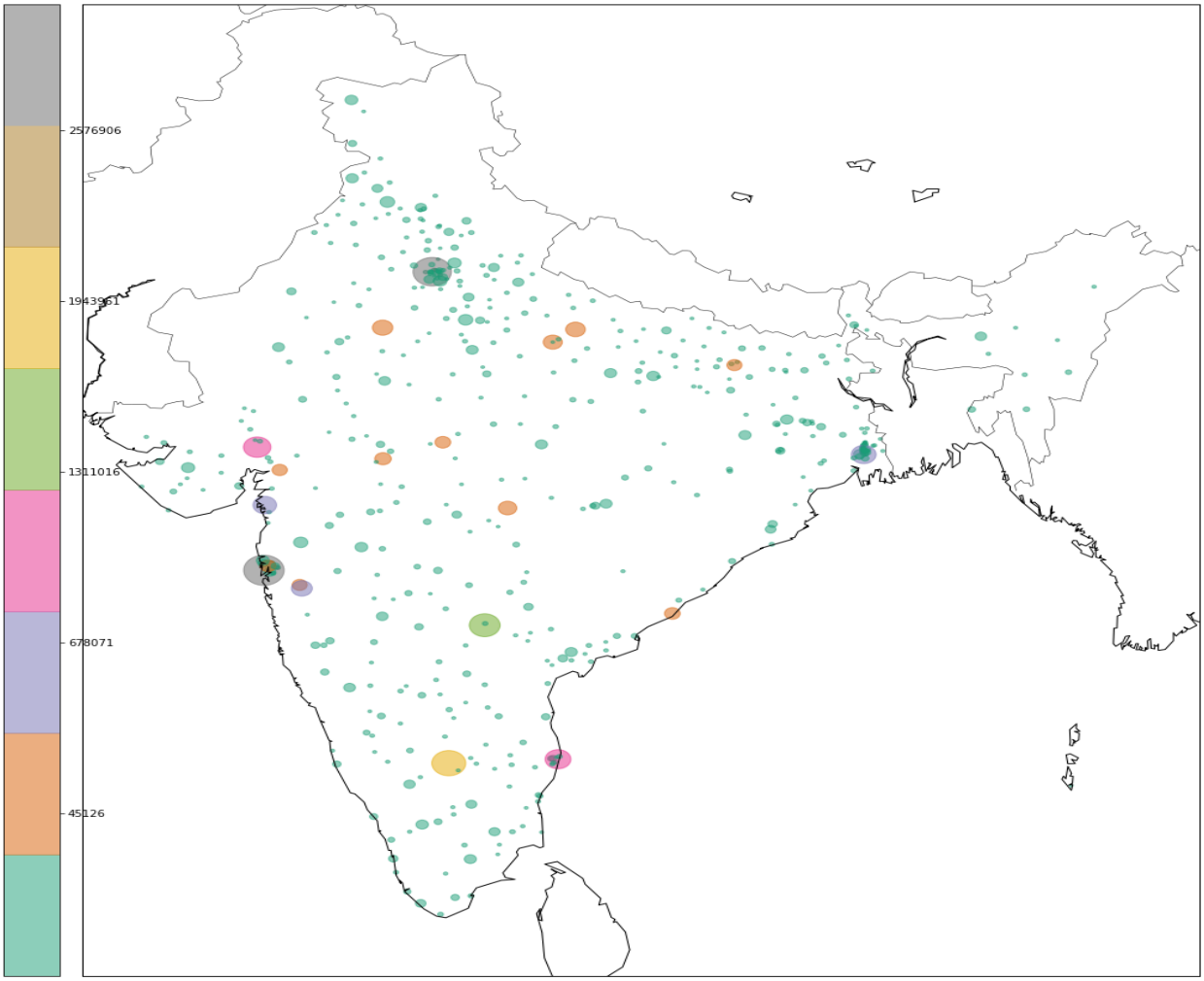

Fig 7.7.2- All the cities on the basis of female population

For plotting the cities on the map of India we will use the plot_map function. For sizes we will give female population as argument and for start and end point of colorbar we will give minimum and maximum value of female population respectively and will change data type using as type function.

7.8 Top 10 female populated cities

We will extract top 10 cities with highest population of female by first sorting all the cities in descending order on the basis of female population. And after that we will take first ten values. Following is the list of the cities.

```
The Top 10 Cities sorted according to the Total Female Population (Descending Order)
```

Out[136]:

	name_of_city	state_code	state_name	dist_code	population_total	population_male	population_female	0-6_population_total	0-6_population_male	6_popul
185	Greater Mumbai	27	MAHARASHTRA	99	12478447	6736815	5741632	1139146	599007	
141	Delhi	7	NCT OF DELHI	99	11007835	5871362	5136473	1209275	647938	
72	Bengaluru	29	KARNATAKA	18	8425970	4401299	4024671	862493	444639	
184	Greater Hyderabad	28	ANDHRA PRADESH	99	6809970	3500802	3309168	725816	373794	
7	Ahmadabad	24	GUJARAT	7	5570585	2935869	2634716	589076	317917	
119	Chennai	33	TAMIL NADU	2	4681087	2357633	2323454	418541	213084	
274	Kolkata	19	WEST BENGAL	16	4486679	2362662	2124017	300052	155475	
449	Surat	24	GUJARAT	25	4462002	2538243	1923759	531522	293208	
380	Pune	27	MAHARASHTRA	25	3115431	1602137	1513294	324572	171152	
225	Jaipur	8	RAJASTHAN	12	3073350	1619280	1454070	378788	204320	

10 rows × 24 columns

Fig 7.8.1- List of top 10 cities with high female population.

For plotting the cities on the map of India we will use basemap extension of matplot. The details for map of India will be same as used before. For longitude and latitude values we will give value corresponding to respective cities. After which x and y will contain these arrays and will be plotted as a scatterplot using scatter function. The marker will be entirely based on the female population of the cities. To put the city name in form of text we will use for loop using name of city, longitude and latitude in a zip folder. We will shift longitude and latitude at a certain distance so that the position of city and name of city is clear. Following is the output for the same.

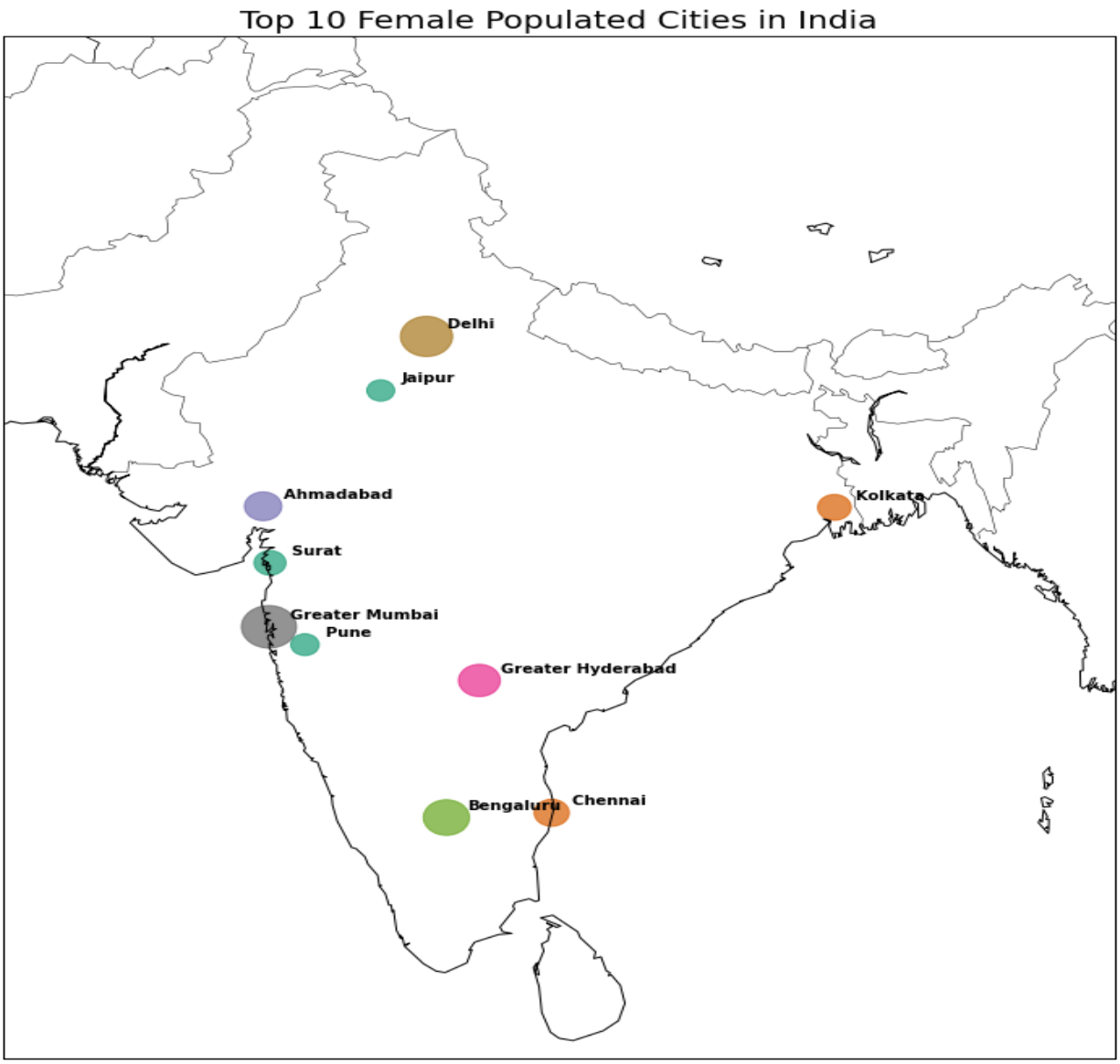

Fig 7.8.2- Top 10 female populated cities

7.9 Kids Population of states (between 0 to 6 years of age)

Now we will analyze the population of kids in all the states and cities in India. All the children who are of age in range 0 to 6 years come under this category. In this section we will see population of kids in all the states and then plot it on the map of India. For this we will plot bar graph and plot all outcomes on map of India.

First we will plot a bar graph to check which state has highest kids' population. For code we will give size of the figure. Then a variable state will be defined which will contain a list of cities which will be grouped according to the states using groupby function. After this we will use plot

function on states variable. It will take two parameters first the type of graph we want to plot which is barh and font size which is 20.

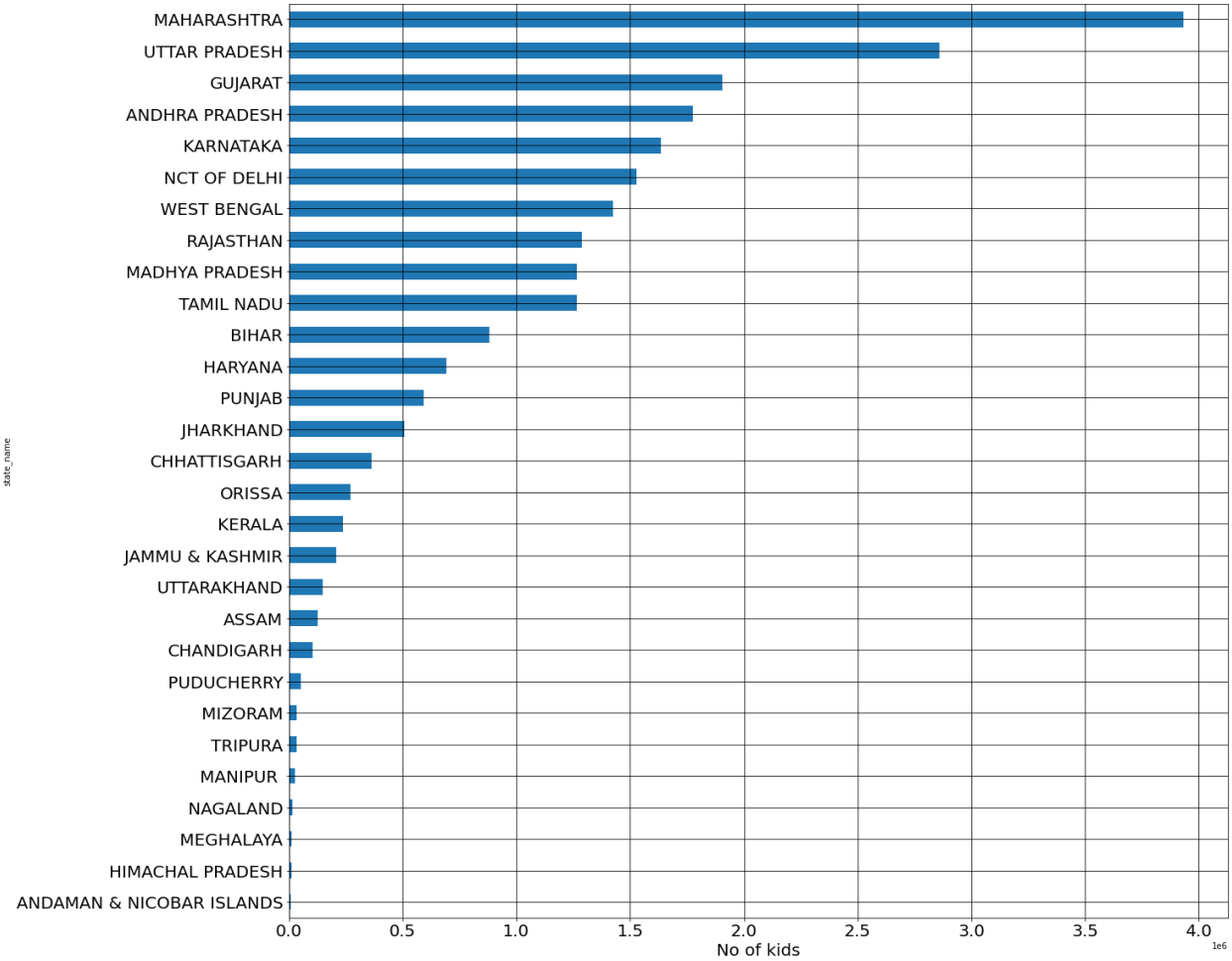

Fig 7.9.1- Kids population state wise

Now to plot all cities on the map of India on the basis of kid population of that particular city we will use plot_map function which was defined earlier. It will take two arguments. First is size which takes list of all the values of population and second colorbar which takes value calculated through linspace which again takes two values start and end point which are minimum and maximum values of kids' population of the cities.

Following is the output –

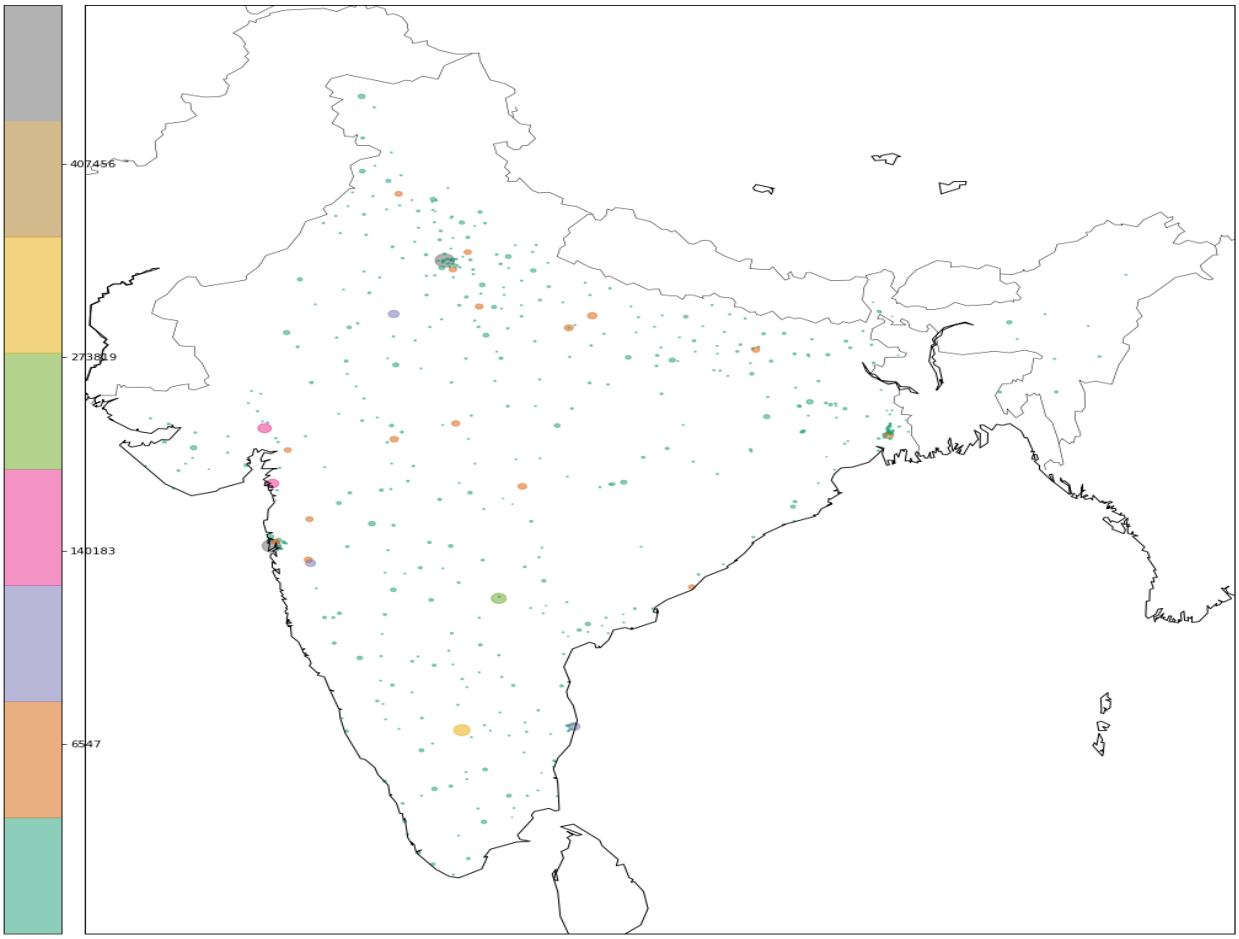

Fig 7.9.2- All the cities on the basis of kid's population

7.10 Top 10 kids populated cities

In this section we will extract top 10 cities with high kids' population. For this first the cities will be sorted on the basis of kids' population value . Following is the list of the cities-

141	Delhi	7	NCT OF DELHI	99	11007835	5871362	5136473	1209275	647938
185	Greater Mumbai	27	MAHARASHTRA	99	12478447	6736815	5741632	1139146	599007
72	Bengaluru	29	KARNATAKA	18	8425970	4401299	4024671	862493	444639
184	Greater Hyderabad	28	ANDHRA PRADESH	99	6809970	3500802	3309168	725816	373794
7	Ahmadabad	24	GUJARAT	7	5570585	2935869	2634716	589076	317917
449	Surat	24	GUJARAT	25	4462002	2538243	1923759	531522	293208
119	Chennai	33	TAMIL NADU	2	4681087	2357633	2323454	418541	213084
225	Jaipur	8	RAJASTHAN	12	3073350	1619280	1454070	378788	204320
380	Pune	27	MAHARASHTRA	25	3115431	1602137	1513294	324572	171152
274	Kolkata	19	WEST BENGAL	16	4486679	2362662	2124017	300052	155475

Fig 7.10.1- Top 10 cities in terms of kids population

For plotting the map of India we will follow the method mentioned earlier only the column for population sizes will be of 0-6 population Following is the output of the code

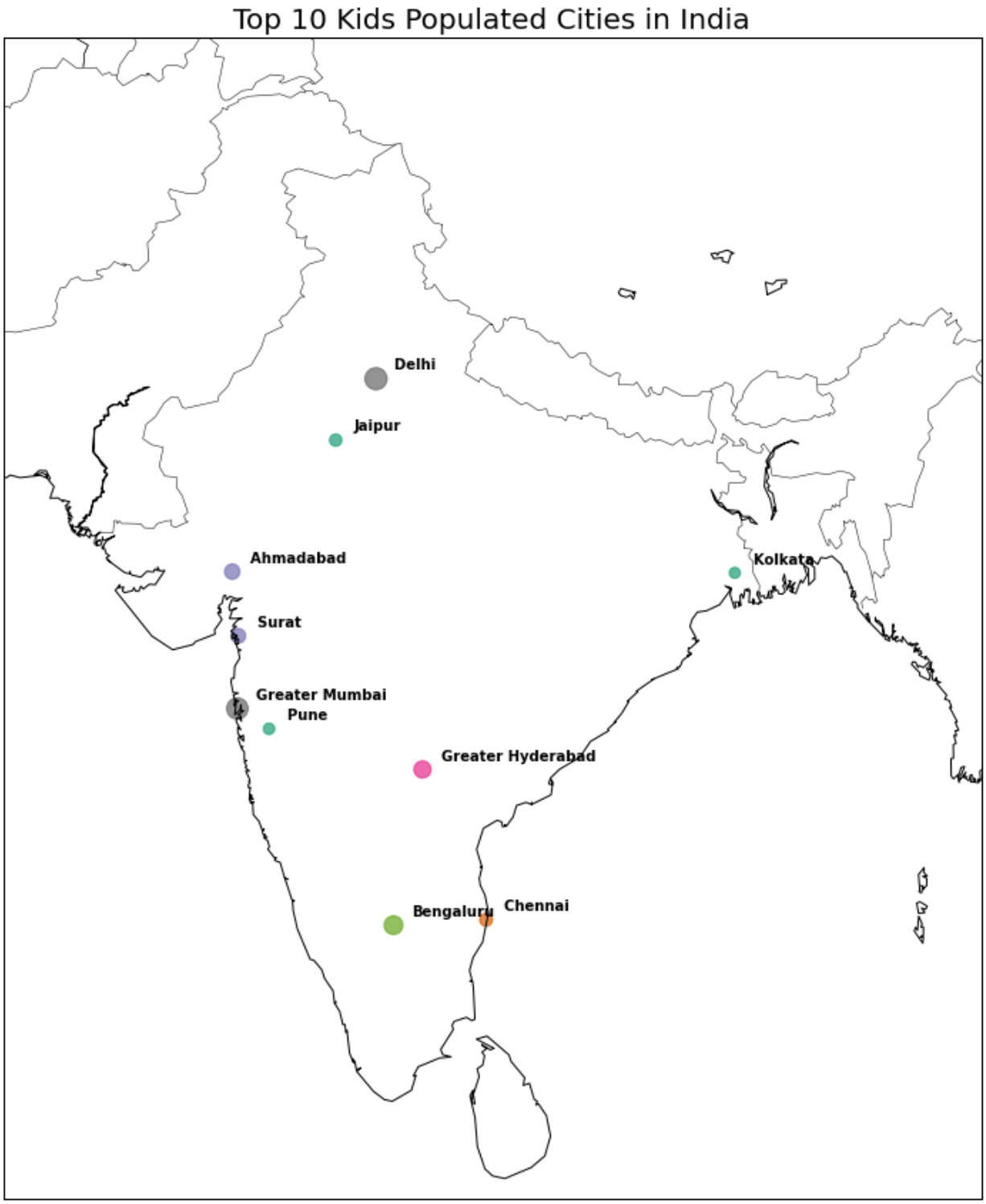

Fig 7.10.2- Top 10 cities on the basis of kid's population

In the similar way we plotted graphs for top 10 cities with highest male population of age 0 to 6 years and female kids' population. Following are the figures for same

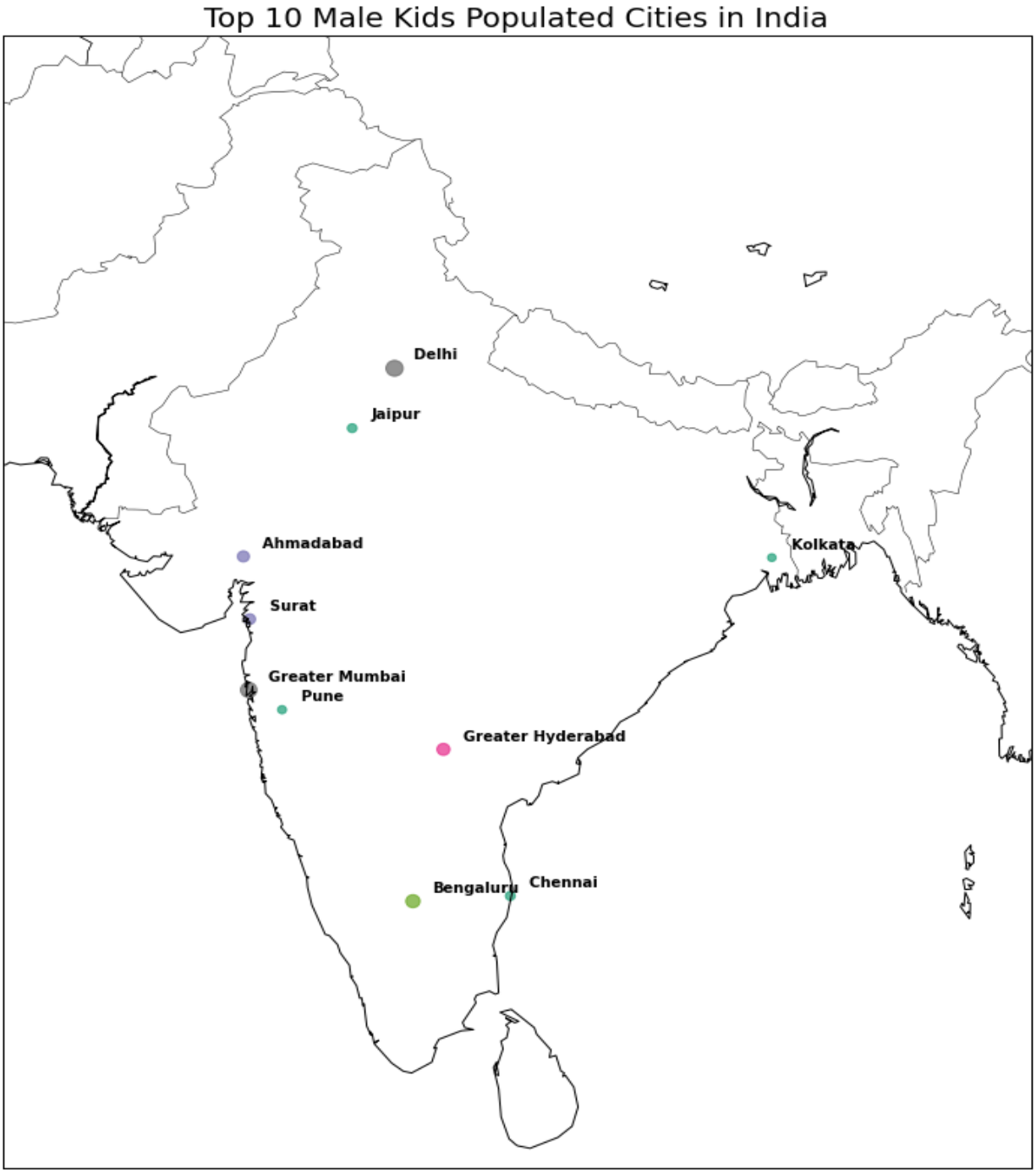

Fig 7.10.2- Top 10 cities on the basis of kid's population(male)

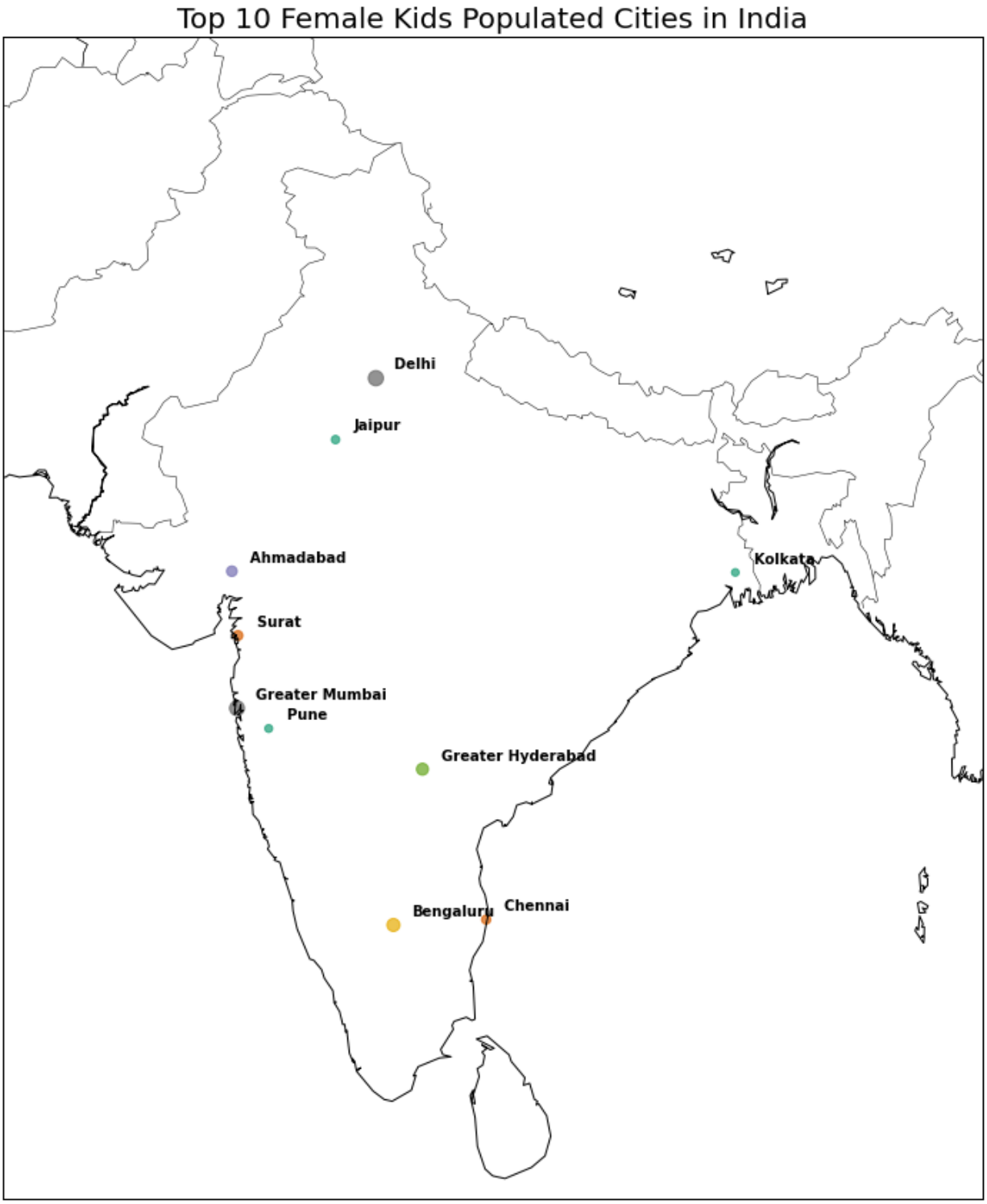

Fig 7.10.2- Top 10 cities on the basis of kid's population (female)

7.11 Literates of states

In this section we will study about literates in different states. We will be covering different aspects which includes overall number of literates in different states, to 10 cities with highest number of literates, total male literates, top10 cities with highest male literates, total female literates and top 10 cities with most female literates. Literates however is not associated with education. Being literate only means the ability to read, write and understand. For every aspect first the bar graph will be plotted and after which cities will be plotted on maps. For overall literates we will be using the plot_map function defined earlier in which we will give respective column name as input. And for the top 10 plotting we will be using basemap and code will be written differently for all three maps.

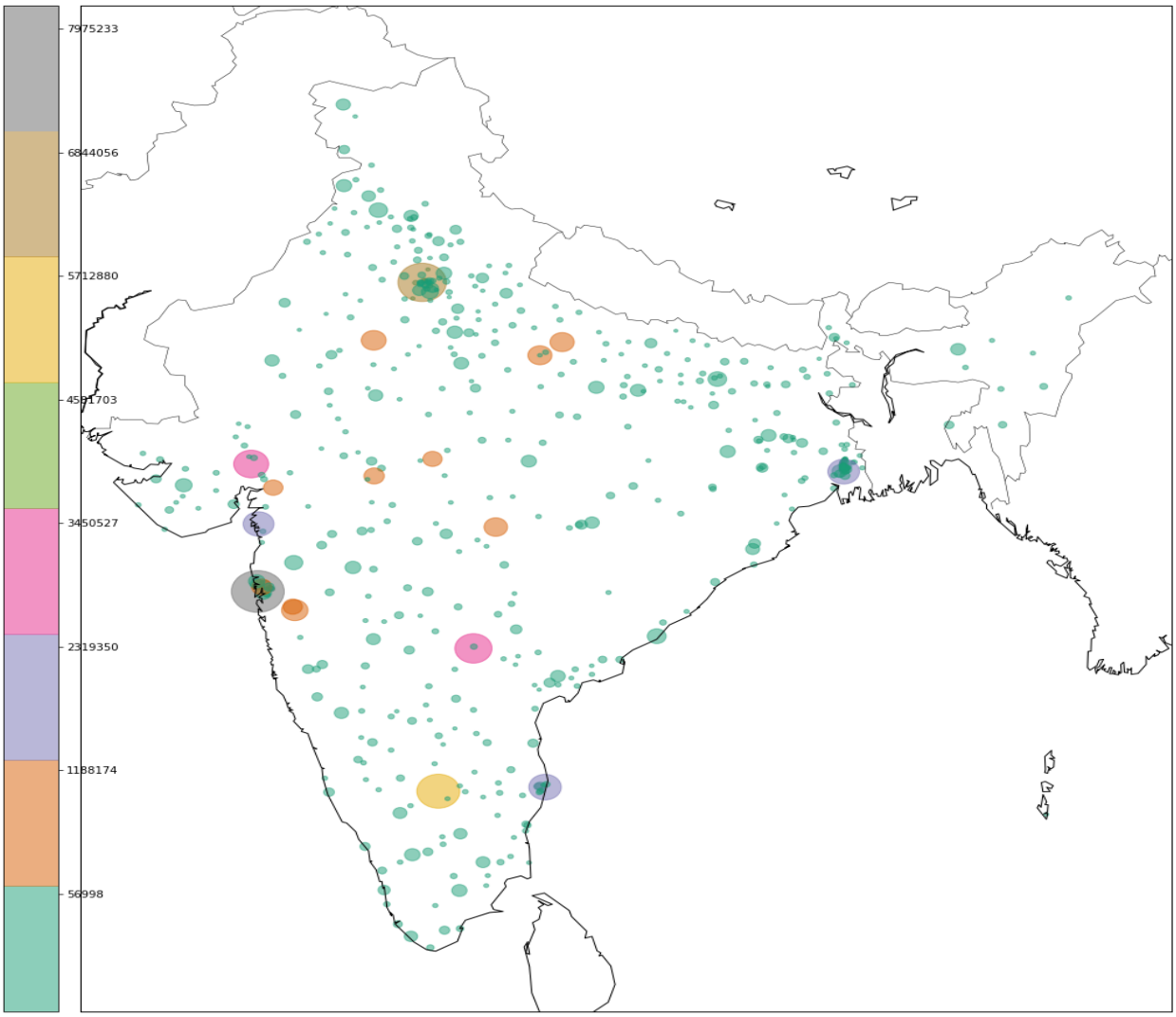

Fig 7.11.1- All the cities on the basis of number of literates

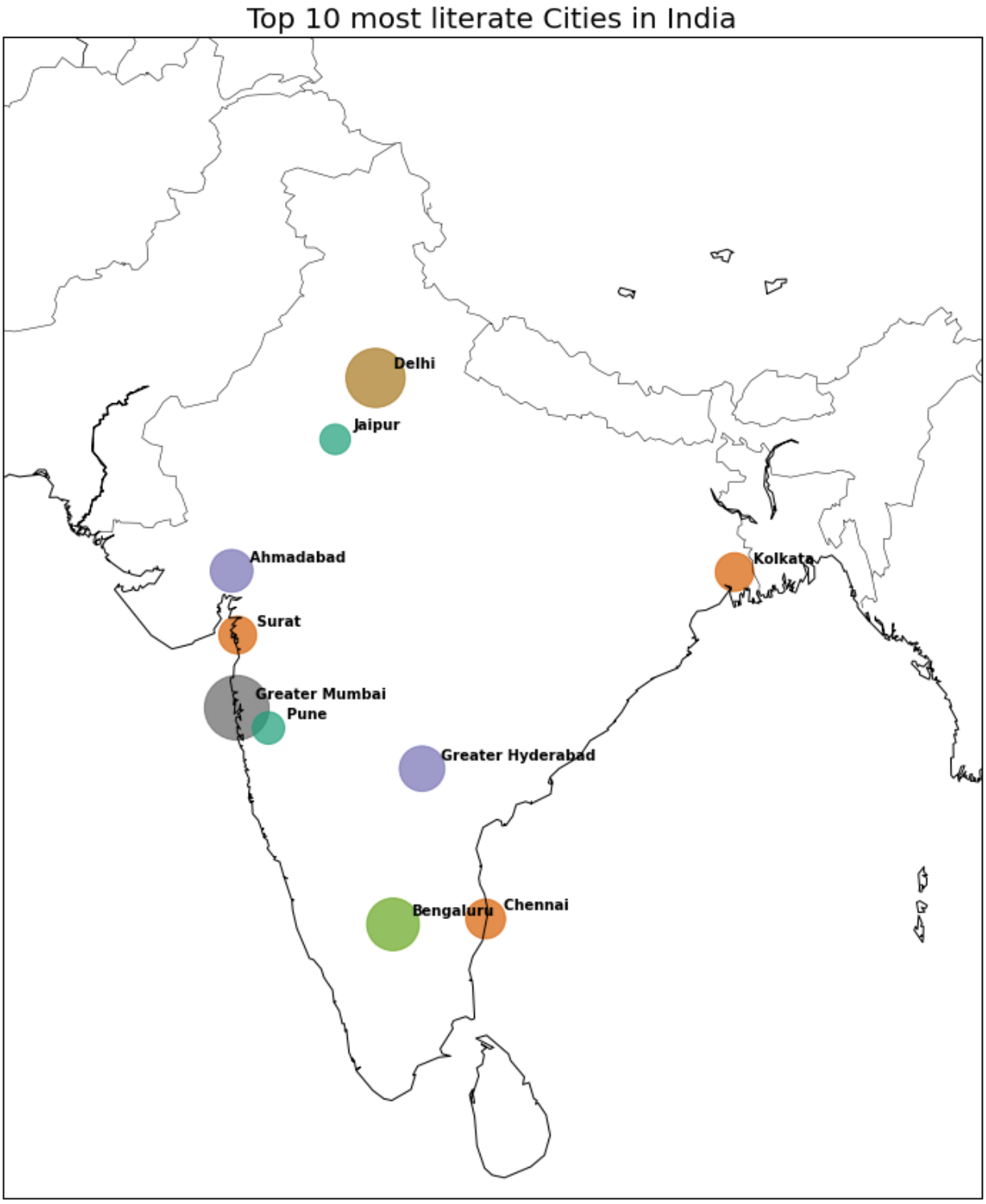

Fig 7.11.2- Top 10 cities on the basis of literates

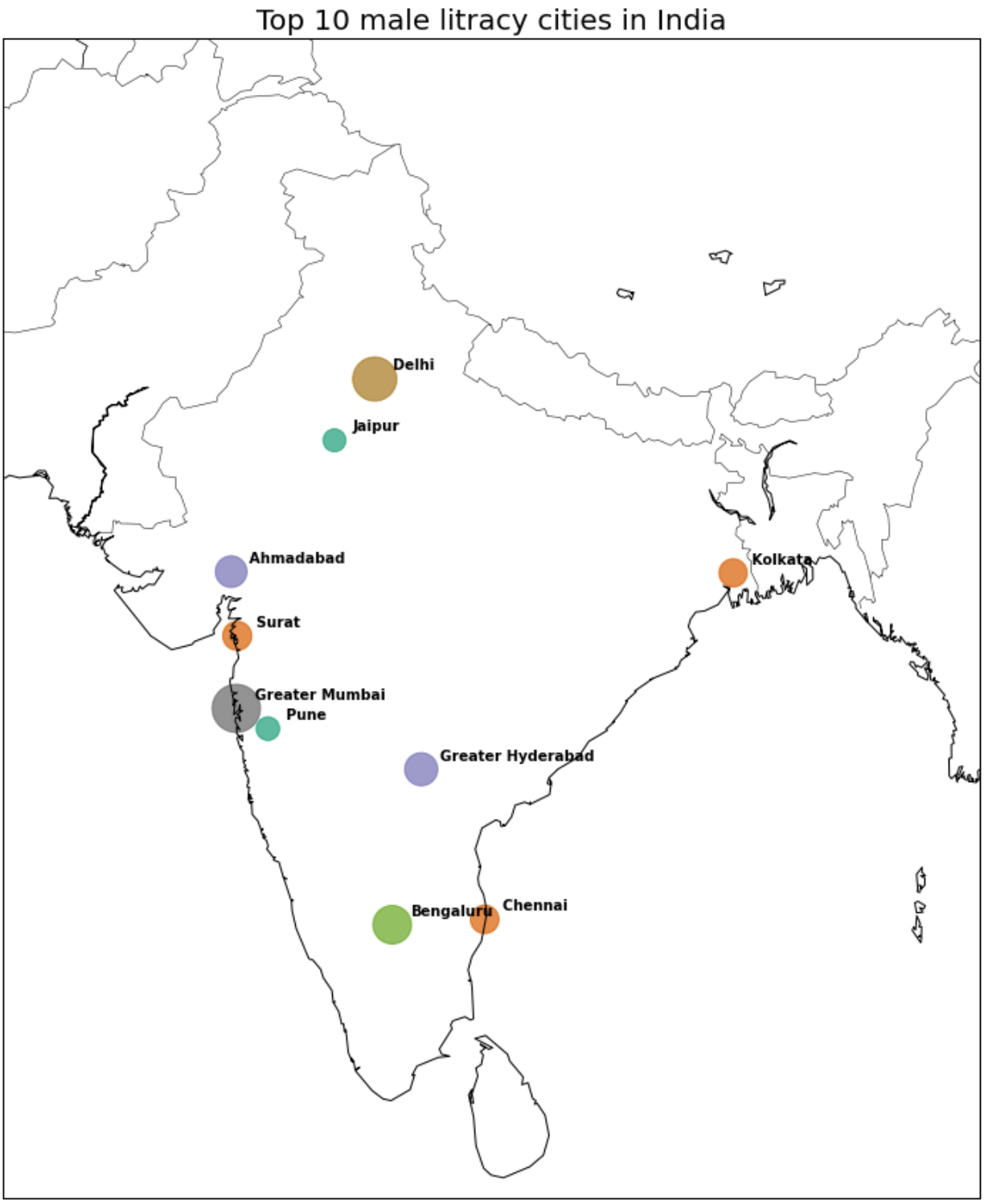

Fig 7.11.3- Top 10 cities on the basis of male literates

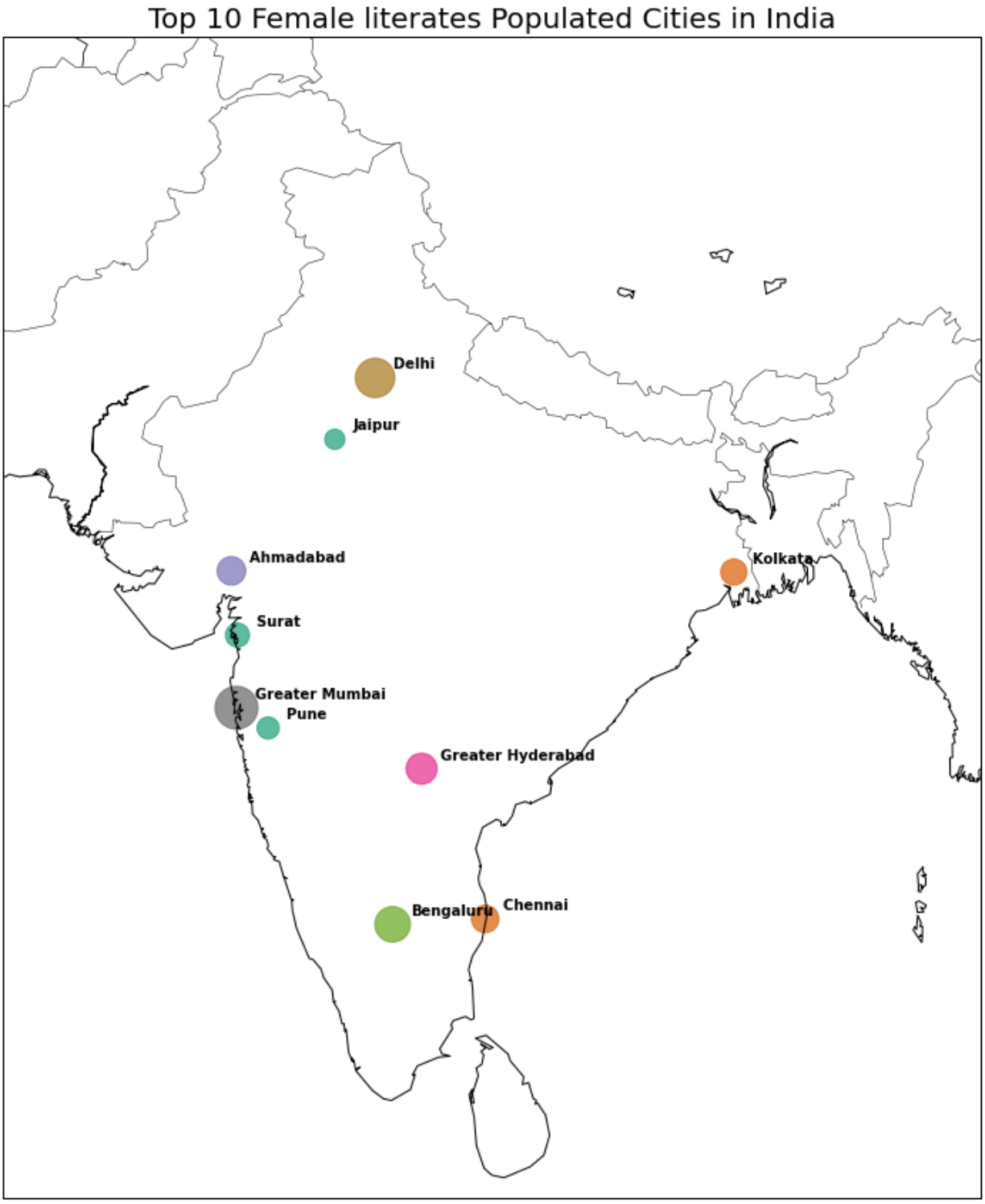

Fig 7.11.4- Top 10 cities on the basis of female literates

7.12 Effective literacy rate

Effective literacy rate can be defined as the total percentage of the population of an area at a particular time who can read and write with understanding. Since these columns are present in our data frame we will be visualizing this part also. The very first step we will create a separate dataset. This will include total effective literacy rate, male effective literacy rate and female effective literacy rate. Although unlike cities dataset this data frame will include the average of the literacy rate. We will calculate the average using numpy library.

For visualization we will be comparing all three literacy rates in a single bar graph. For plotting bar graph we will sort the state on the basis of total effective literacy rate after which in the same line we will use plot function to plot bar graph for all three rates. In the plot function some specific details are given which are as follows

i. Kind of graph which is bar graph

ii. Grid as true. This will draw gridlines in the background of the graph.

iii. Size of the figure which is 15x16

iv. Alpha whose value is 0.6. It helps to adjust the transparency of the graph.

v. Width whose value is 0.6. It adjusts the width of the bar

vi. Edge colors is black. It gives colors to the edges of the bars.

vii. Font size is 20. It is the font size of the x and y labels.

After plotting graph for the effective literacy rate we will repeat the same method to plot map for graduates living in all the states. All the values for all parameters will remain same for this plot. Also after plotting graph there was a major observation that although maharashta and uttar Pradesh had huge population but there was a lot of difference in graduates, literates and sex ratio. All these are discussed in detail in next chapter. Following are the results for the above maps that were plotted.

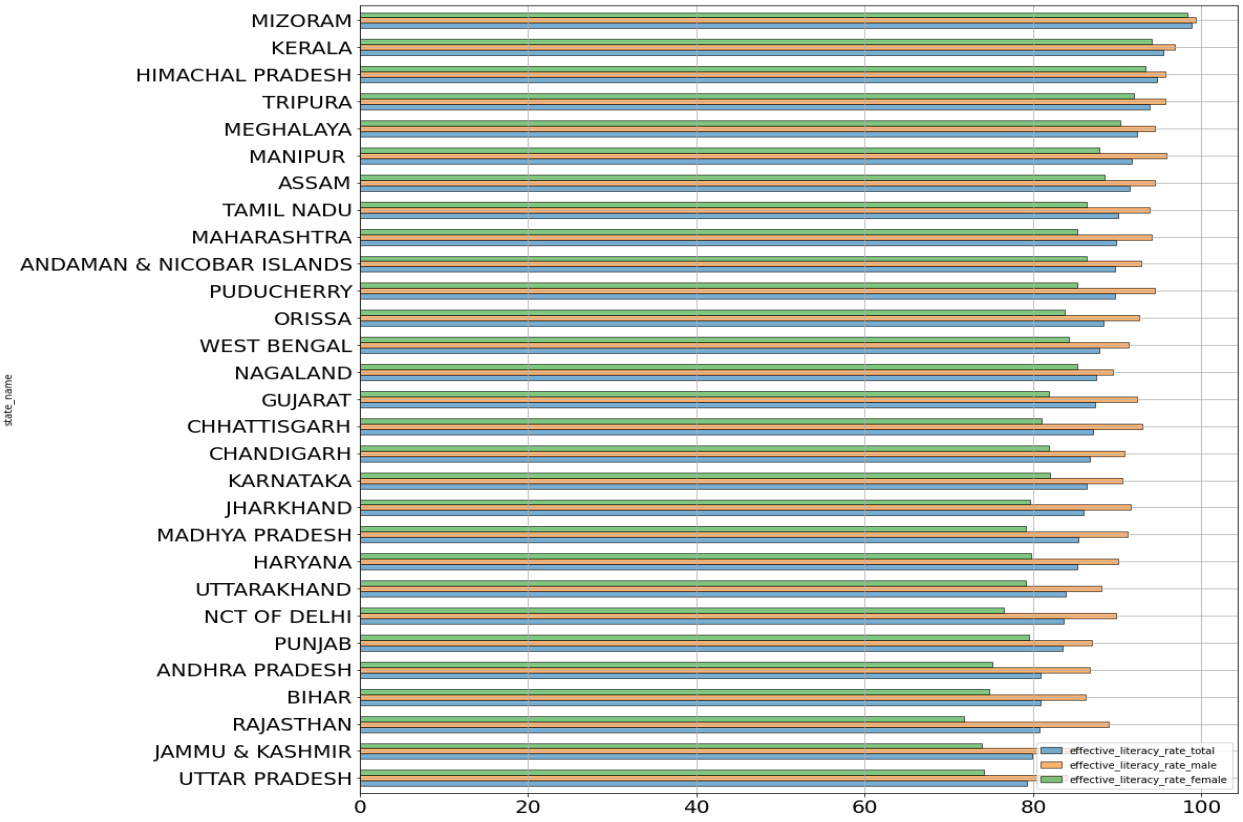

Fig 7.12.1- Effective literacy rate of all the states

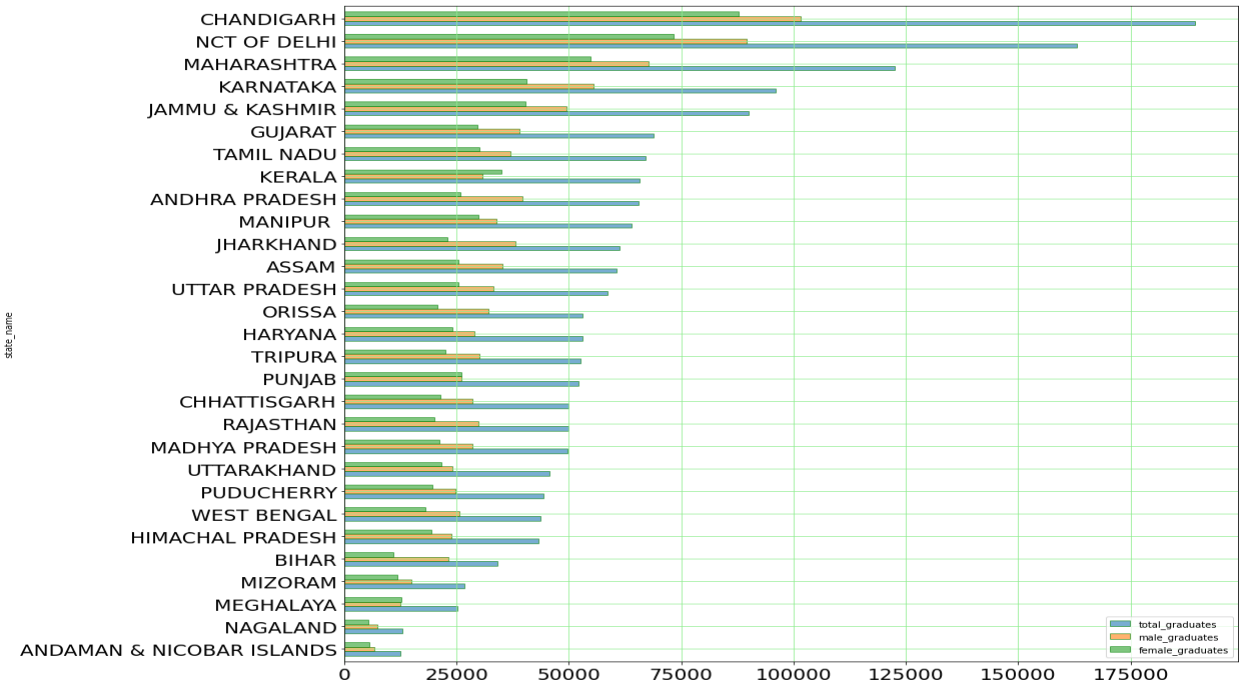

Fig 7.12.2- number of graduates

7.13 Sex ratio (Adult and child)

In this section we will analyze the sex ratio in different states. Sex ratio can be defined as number of females per 100 0 males. Since a column named sex ratio is already present in the cities data frame that why any calculation is not needed. We will just compare different states in terms of sex ratio by plotting bar graph. For plotting bar graph, the size of the figure is 20x20. For values to be plotted we will create a variable called states which contains all states which sorted in ascending order on the basis of the mean of the sex ratio. Next we will apply plot function on states variable which will take kind of the graph(bar) and font size (20) as parameters. After this we will label the x axis and then the graph will be plotted.

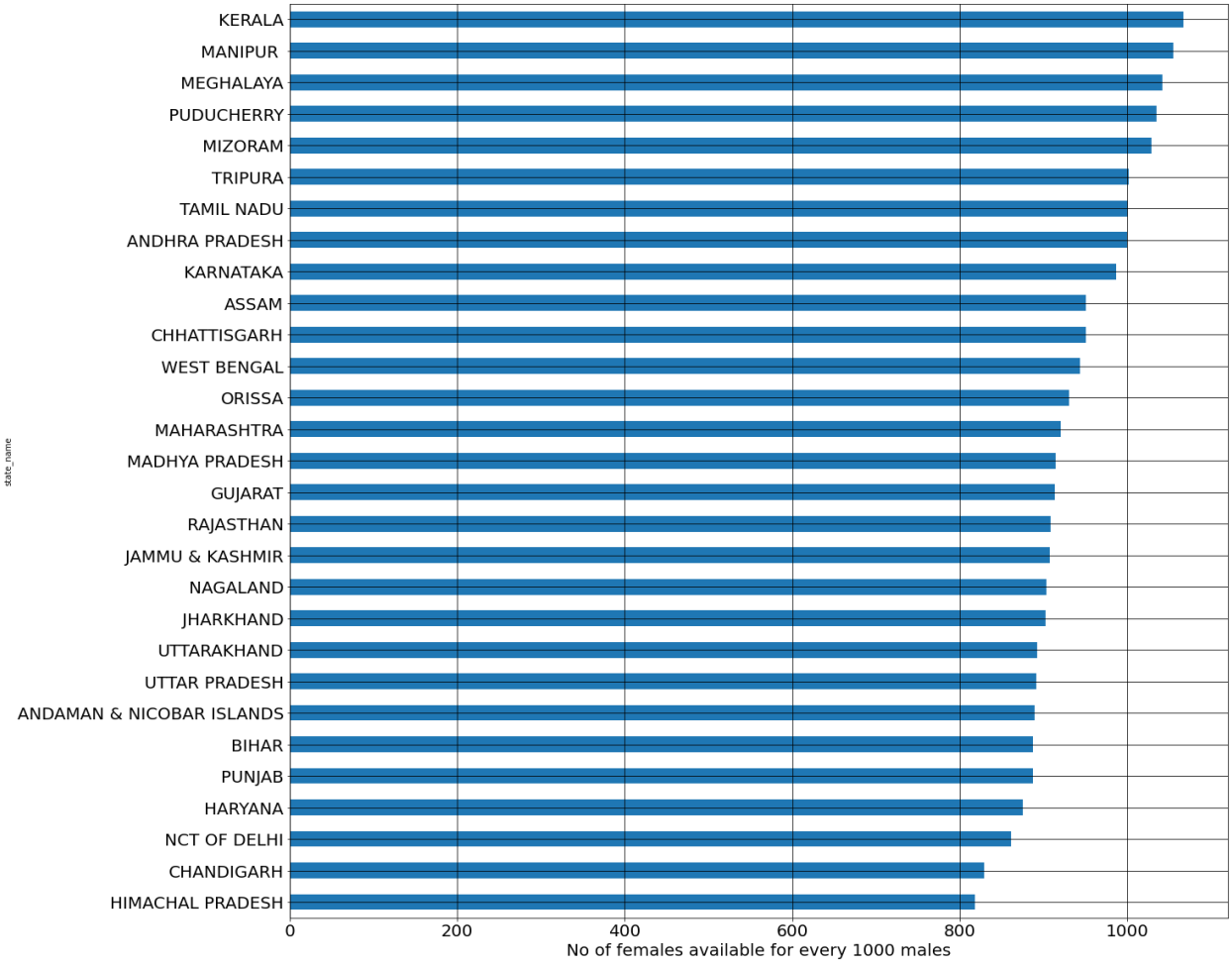

Fig 7.13.1- bar graph for adult sex ratio

We will follow the same procedure for child sex ratio using the column child_sex_ratio.

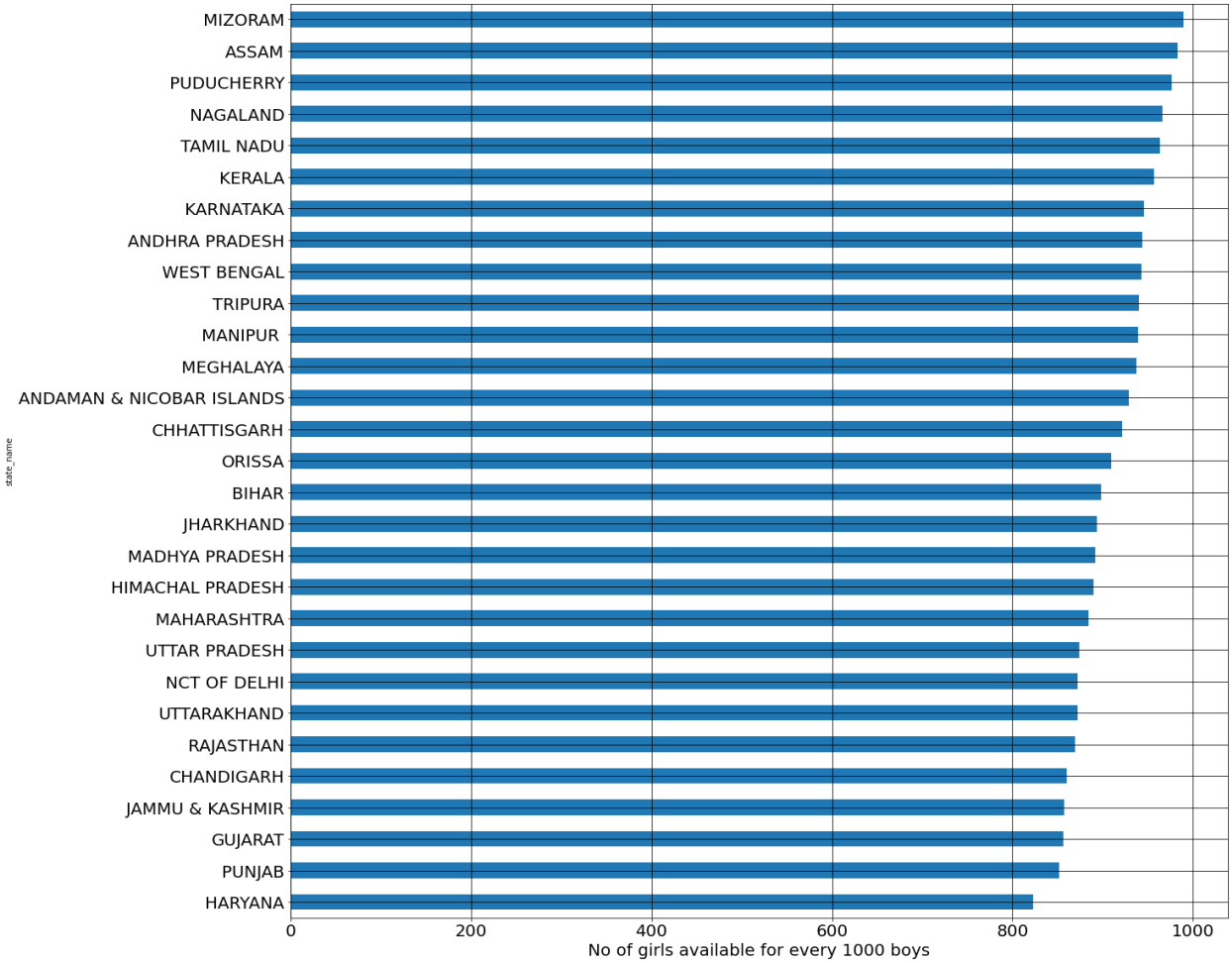

Fig 7.13.2- bar graph for child sex ratio

CHAPTER 8

OBSERVATIONS

8.1 Observations made during analysis

Following are some of the observations made during exploratory data analysis and visualization of the population of different urban cities.

1. Greater Mumbai has the highest population with a total of 12478447 which further means that the state Maharashtra has highest population living in urban areas.
2. The least populated state in terms of urban areas is Andaman and Nicobar Islands
3. Maharashtra and Uttar Pradesh have huge male population.
4. Greater Mumbai is the city with high male population.
5. Maharashtra state has high female population among all the cities with again Greater Mumbai with highest female population.
6. Kids population is highest in Maharashtra state
7. Kids population is smaller than the overall population and bigger cities like Delhi, Mumbai, Bangalore, Kolkata, Hyderabad, Chennai have vast number of kids living in cities.
8. Delhi tops the list of top ten cities in which large number of kids live.
9. Again states like Maharashtra and UP have huge male and female kids' population living in cities.
10. Maharashtra is the only state which have huge literate population living in urban areas.
11. Greater Mumbai have highest number of literates
12. Mizoram is the state that have very high effective literacy rate
13. Kerala and Himachal Pradesh. Closely follow Kerala in terms of Effective literacy rate
14. Not even a single state has more female literates than male literates. Worst case is Rajasthan, where difference between effective literacy rate of men and women is very high.
15. Almost all the states have effective literacy rate of more than 80 % (For Urban areas only)
16. Kerala and Meghalaya are the only states where more female graduates are seen than male graduates in urban areas.
17. In Bihar and Jharkhand, difference between men and women graduates is very high in urban areas itself.

18. Kerala, Manipur, Meghalaya, Puducherry, Mizoram are the states where more than 1000 females are there for every > 1000 males. It means there are more females than males. (ADULT)

19. In Chandigarh and Himachal Pradesh there are around 800 females for every 1000 males (which is clearly a bad sign)

20. When children below 6 are taken into account, not even a single state have 1000 girls for 1000 boys.

References

- For data
  https://www.kaggle.com/sansuthi
- For Understanding of various functions of pandas
  https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.html
- For understanding of matplotlib
  https://www.w3schools.com/python/matplotlib_pyplot.asp
- For understanding of Basemap
  https://matplotlib.org/basemap/
- For understanding of scikit learn
  https://scikit-learn.org/
- For understanding of linspace function
  https://www.studytonight.com/numpy/numpy-linspace-function

LINK FOR THE CODE

2022 prediction - https://github.com/Sonalitwr/Population-analysis/blob/main/2022prediction.ipynb

2011 census analysis - https://github.com/Sonalitwr/Population-analysis/blob/main/2011censusanalysis.ipynb