**National-Rural-Health-Mission**

A Project Report

***Submitted by***

**Ayush Jain (2022180)**

**Harshit Shah (202218040)**

**Nakul Tomar (2022180)**

**Shashva Machchar (2022180)**

IT 495 Exploratory Data Analysis

SEMESTER II

Instructor: Dr. Gopinath Pandya

Group No: 2



**DEPARTMENT OF DATA SCIENCE**

DHIRUBHAI AMBANI INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGY

# INDEX

1. Declaration 4

2. LIST OF FIGURES 3

3. INTRODUCTION 5

4. TOOLS AND TECHNOLOGY: 5

5. METHODOLOGY 5

Dataset 1

Mobile Medical Units data from September 2016 5

Dataset 2

State wise Ambulances Operational under NHM during 2018 8

Dataset 3

State-wise Established Special New-born Care Units (SNCU) 10

Dataset 4

State-wise Accredited Social Health Activists (ASHAs) selected under NHM during 2018 14

Dataset 5

Geocode Health Centre & NHM July 2022 15

6. Regression: 34

7. Conclusion: 38

LIST OF FIGURES:

[Figure 1: Top 10 State-Wise comparison of Operational MMUs in 2016 using bar chart 8](#_Toc133890830)

[Figure 2: Top 10 State wise operational Ambulance in the year 2018 using Stacked bar chart. 11](#_Toc133890831)

[Figure 3: State-wise established SNCU (2020) 14](#_Toc133890832)

[Figure 4:State wise ASHAs selected by NHRM in 2018 16](#_Toc133890833)

[Figure 5: Heatmap to show null values in dataset 17](#_Toc133890834)

[Figure 6: Null values in dataset 18](#_Toc133890835)

[Figure 7: Correlation Matrix- comparing all the columns in the dataframe 20](#_Toc133890836)

[Figure 8: Plotting pairplot using seaborn of the dataframe. 21](#_Toc133890837)

[Figure 9: Budget proposed and Release of Funds for 2015 -16 22](#_Toc133890838)

[Figure 10: Budget proposed and release of funds for 2016 -17 23](#_Toc133890839)

[Figure 11: Budget proposed and release of funds for 2017 -18 23](#_Toc133890840)

[Figure 12: Budget proposed and release of funds for 2018 -19 23](#_Toc133890841)

[Figure 13: Budget proposed and release of funds for 2019 -20 24](#_Toc133890842)

[Figure 14: Budget proposed and release of funds for 2020 -21 24](#_Toc133890843)

[Figure 15: Budget proposed and release of funds for 2021 -22 24](#_Toc133890844)

[Figure 16: Budget proposed and release of funds for 2022 -23 25](#_Toc133890845)

[Figure 17: Bar chart of Budget Proposed by State/UTs 26](#_Toc133890846)

[Figure 18: Bar chart of Budget approved for the state/uts 26](#_Toc133890847)

[Figure 19: Opening balance with the states/ UTs 27](#_Toc133890848)

[Figure 20: Release of Government of India’s Fund 27](#_Toc133890849)

[Figure 21: Total Expenditure reported (Including State’s share) 28](#_Toc133890850)

[Figure 22: Extent of Budget approved against budget proposed 28](#_Toc133890851)

[Figure 23: Extent of Funds utilised against budget approved 29](#_Toc133890852)

[Figure 24: Extent of Funds utilised against budget approved 29](#_Toc133890853)

[Figure 25: Budget Propsed by the State/UTs 30](#_Toc133890854)

[Figure 26: Budget Approved for the States/Uts 30](#_Toc133890855)

[Figure 27: Opening Balance with the states/uts 30](#_Toc133890856)

[Figure 28: Release of Government of India’s Fund 31](#_Toc133890857)

[Figure 29: Total Expenditure Reported (Including State’s share) 31](#_Toc133890858)

[Figure 30: Extent of Budget Approved Against Budget Proposed 31](#_Toc133890859)

[Figure 31:Extent of Funds utilised against Budget Approved 32](#_Toc133890860)

[Figure 32: Extent of Funds utilised against Budget Approved 32](#_Toc133890861)

[Figure 33: Budget Proposed by the States/ UTs in 2015 -16 33](#_Toc133890862)

[Figure 34: Budget Proposed by the States/ UTs in 2016 -17 33](#_Toc133890863)

[Figure 35: Budget Proposed by the States/ UTs in 2017 -18 33](#_Toc133890864)

[Figure 36: Budget Proposed by the States/ UTs in 2018 -19 34](#_Toc133890865)

[Figure 37: Budget Proposed by the States/ UTs in 2019 -20 34](#_Toc133890866)

[Figure 38: Budget Proposed by the States/ UTs in 2020 -21 34](#_Toc133890867)

[Figure 39: Budget Proposed by the States/ UTs in 2021 -22 35](#_Toc133890868)

[Figure 40: Budget Proposed by the States/ UTs in 2022 -23 35](#_Toc133890869)

[Figure 41: Linear Regression plot for the budget proposed vs the budget approved 36](#_Toc133890870)

[Figure 42: Linear Regression 37](#_Toc133890871)

[Figure 43: Linear Regression on release of Govt. Funds and total expenditure 37](#_Toc133890872)

## DECLARATION:

We, the undersigned, declare that this project titled [Project Title] is our original work and that all information contained herein has been properly cited and referenced. We affirm that we have not engaged in any form of academic dishonesty, including but not limited to plagiarism or falsification of data. This project has not been submitted, in part or in whole, for any other academic purpose and all sources of information used in this project have been identified in the references section. We understand that any breach of academic integrity may result in severe Penalties, including revocation of our degrees.

Additionally, we confirm that each member of the group contributed equally to the completion of the project. Each member participated in this project's planning, research, analysis, and writing. We also communicated and collaborated effectively throughout the project, ensuring that each member's contributions were properly integrated.

[Group member name and signature]

[Date submitted]

# INTRODUCTION:

NRHM seeks to provide equitable, affordable, and quality health care to the rural population, especially the vulnerable groups. Under the NRHM, the Empowered Action Group (EAG) States, as well as the North Eastern States, Jammu and Kashmir and Himachal Pradesh, have been given special focus. The thrust of the mission is on establishing a fully functional, community-owned, decentralized health delivery system with inter-sectoral convergence at all levels, to ensure simultaneous action on a wide range of determinants of health such as water, sanitation, education, nutrition, social and gender equality. Institutional integration within the fragmented health sector was expected to provide a focus on outcomes, measured against Indian Public Health Standards for all health facilities.

In this we have taken datasets from the “DATA.GOV.ORG” and perform EDA on various sectors of health sector which are part of NHRM

# TOOLS AND TECHNOLOGY:

* Pandas
* Matplotlib
* Seaborn
* NumPy
* Plotly Express
* Ipywidgets
* Scikit Learn
* JSON

# METHODOLOGY:

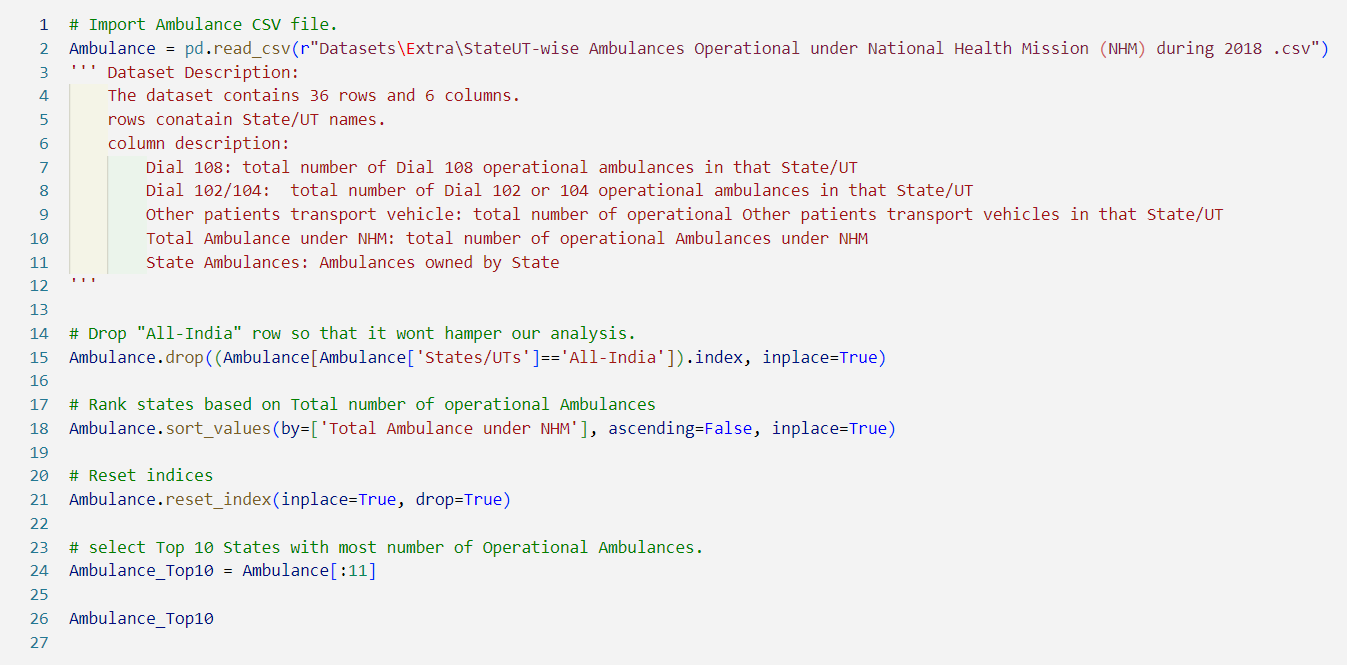
We have conducted Exploratory Data Analysis (EDA) on several datasets we found on the DATA.GOV website. Then we followed a process step by step.

1. Defined the problem and questions.
2. Collect and load data.
3. Clean and pre-process data.
4. Explore variables.
5. Visualize data.
6. Performed statistical analysis.
7. Summarized our findings

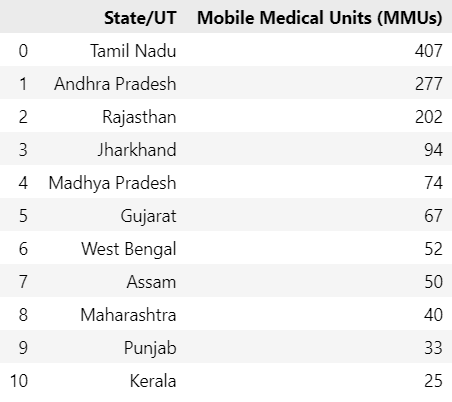
## Dataset 1: Mobile Medical Units data from September 2016

Code:

The code imports a CSV file containing information about the number of operational ambulances in each state/UT in India under the National Health Mission (NHM) during 2018. It drops the row containing "All-India" data, ranks the states based on the total number of operational ambulances, resets the indices, and selects the top 10 states with the most number of operational ambulances. The resulting dataframe is stored in the variable `Ambulance\_Top10`.

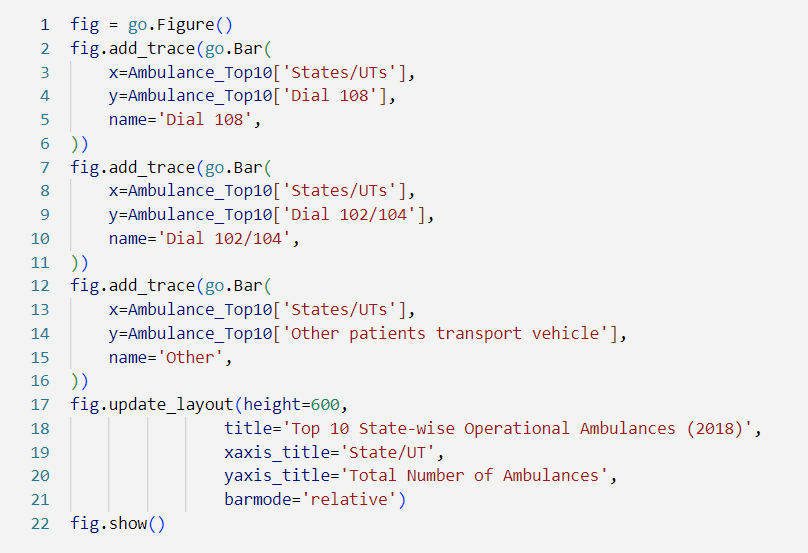


The dataset consist of 3 columns with State/ UT names and followed by MMUs in a particular state.



Code:

This code is creating a bar chart using the Plotly library. It is adding three traces to the chart, each representing a different type of ambulance service (Dial 108, Dial 102/104, and Other patients transport vehicle) for the top 10 states/UTs with the highest number of operational ambulances in 2018. The chart is then updated with a title, axis labels, and a relative barmode, and finally displayed using the `fig.show()` function.



Using this data we generated bar graph to represent to compare MMUs count for top 10 states in INDIA.

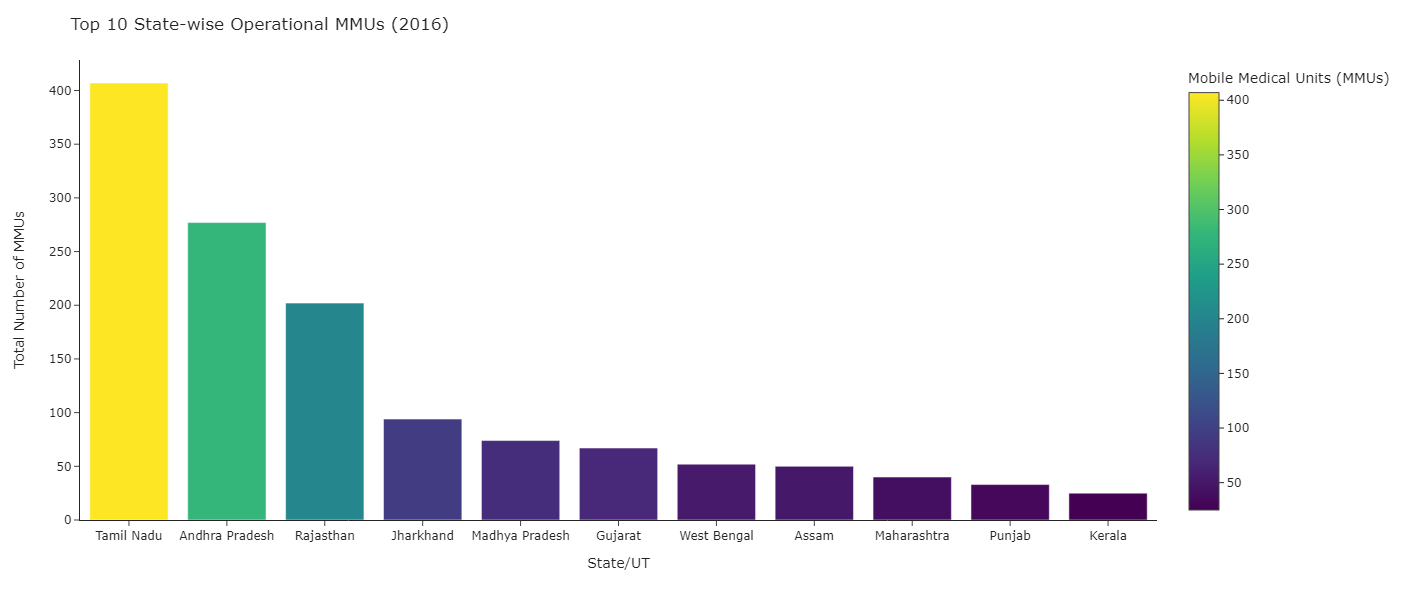


Figure 1: Top 10 State-Wise comparison of Operational MMUs in 2016 using bar chart

Findings: Visualising the graph it can be clearly seen that Tamil Nadu has the maximum number of operational MMUs (407) according to the data which then followed by Andhra Pradesh and Rajasthan which have 277 and 202 operational units.

## Dataset 2: State wise Ambulances Operational under NHM during 2018

The dataset contains 36 rows and 6 columns.

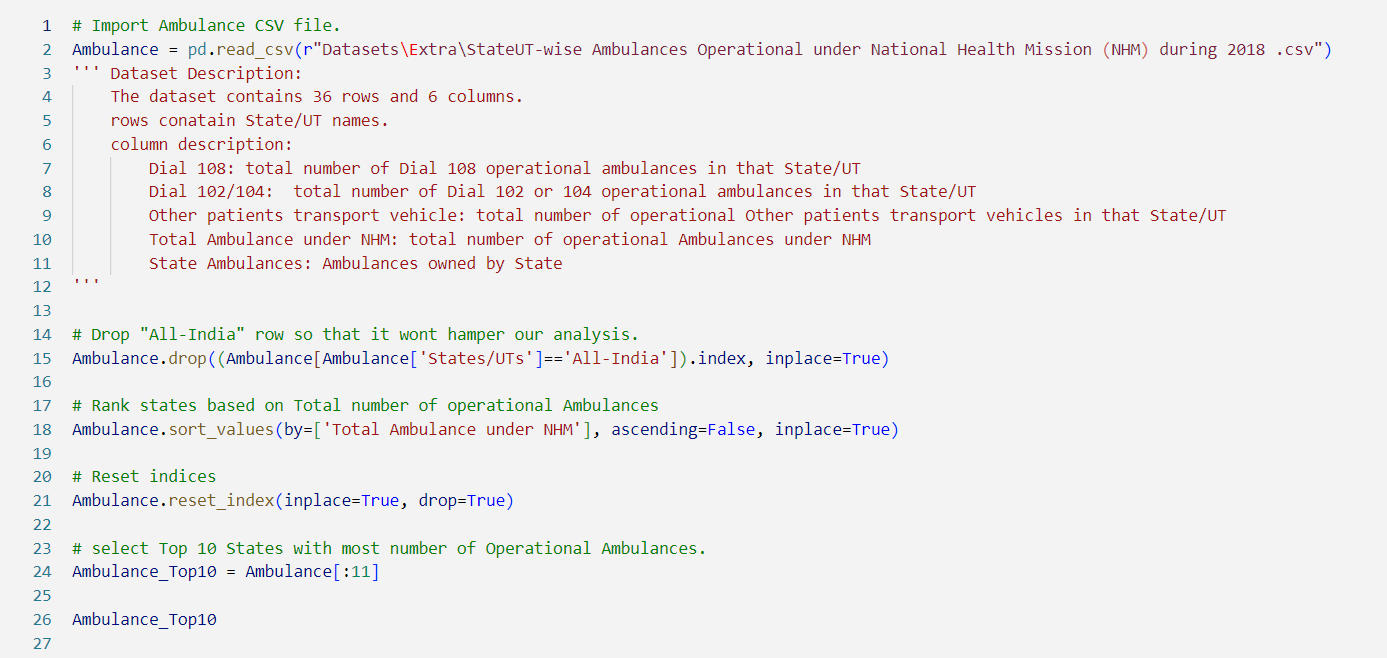
Rows contains: State/UT names.

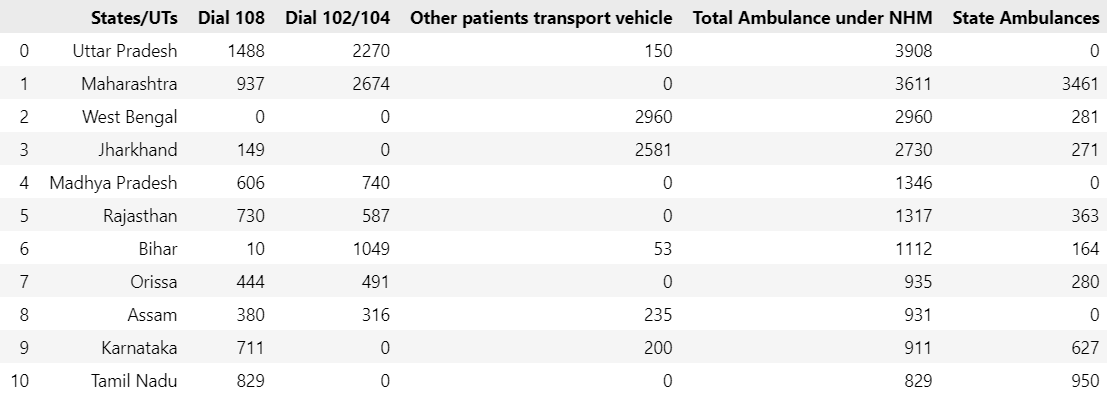
Column description:

1. Dial 108: total number of Dial 108 operational ambulances in that State/UT
2. Dial 102/104:  total number of Dial 102 or 104 operational ambulances in that State/UT
3. Other patients transport vehicle: total number of operational Other patients transport vehicles in that State/UT
4. Total Ambulance under NHM: total number of operational Ambulances under NHM
5. State Ambulances: Ambulances owned by State

Code:

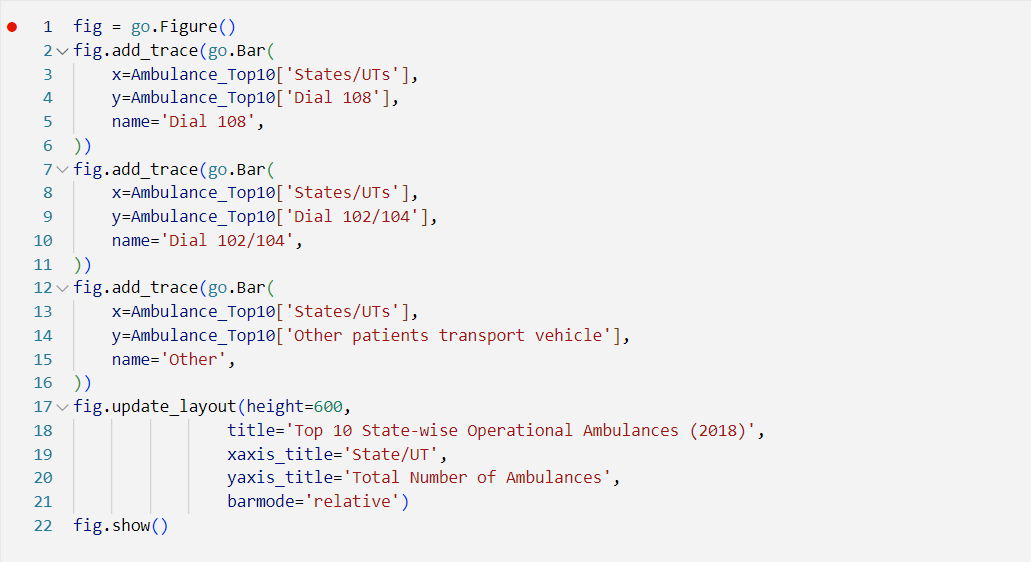
The code imports a CSV file containing information about the number of operational ambulances in each state/UT in India under the National Health Mission (NHM) during 2018. It drops the row containing "All-India" data, ranks the states based on the total number of operational ambulances, resets the indices, and selects the top 10 states with the most number of operational ambulances. The resulting dataframe is stored in the variable `Ambulance\_Top10`.





Code:

This code is creating a bar chart using the Plotly library. It is creating a figure object using `go.Figure()`, and then adding three traces to the figure using `fig.add\_trace()`. Each trace represents a different type of ambulance service (Dial 108, Dial 102/104, and Other patients transport vehicle) for the top 10 states/UTs in India. The x-axis represents the states/UTs, and the y-axis represents the total number of ambulances for each type of service. The chart is then customized using `fig.update\_layout()` to set the title, axis labels, and barmode. Finally, the chart is displayed using `fig.show()`.



Findings:

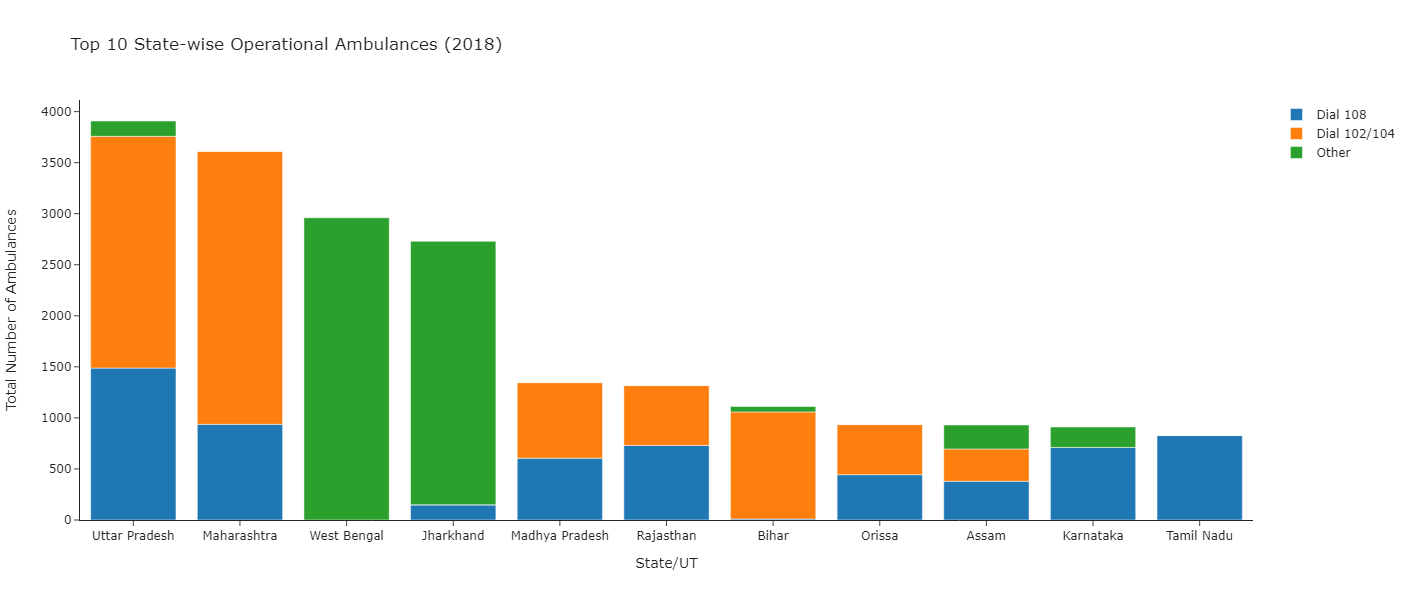


Figure 2: Top 10 State wise operational Ambulance in the year 2018 using Stacked bar chart.

From the above plot, we can deduce that Uttar Pradesh has the greatest number of Operational Ambulances.

Also UP has most number of Dial 108 ambulances whereas Maharashtra and West Bengal has most number of Dial 102/104 and other ambulances respectively

## Dataset 3: State-wise Established Special New-born Care Units (SNCU)

Data Description:

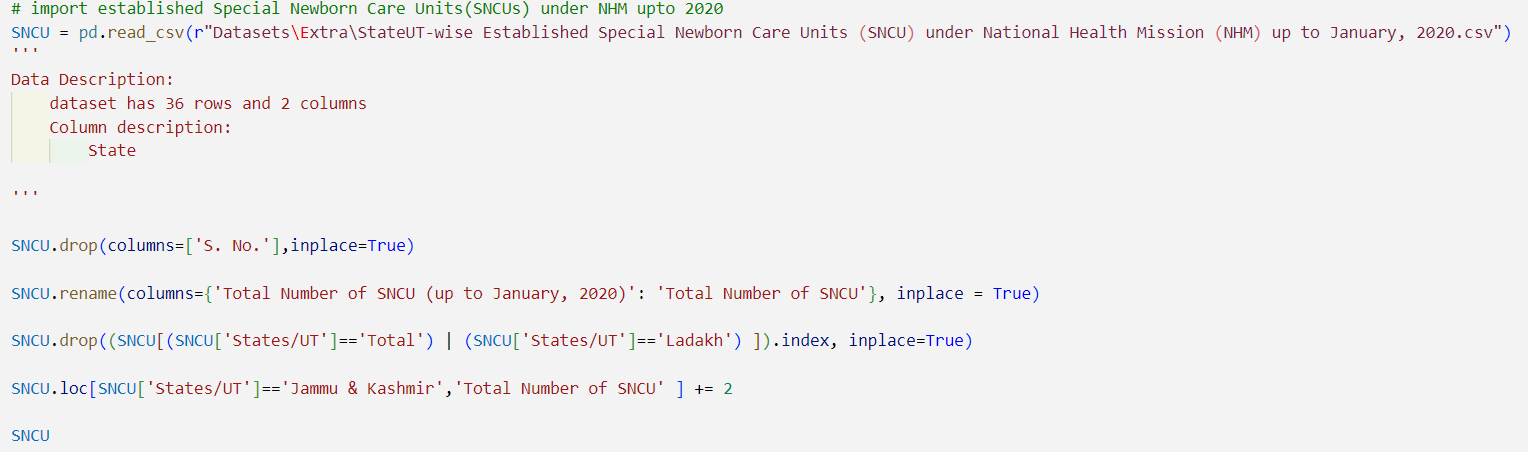
    dataset has 36 rows and 2 columns

    Column description: State

In this dataset we need to rename the column names for the better visualisation and analysis of the data.

Code:

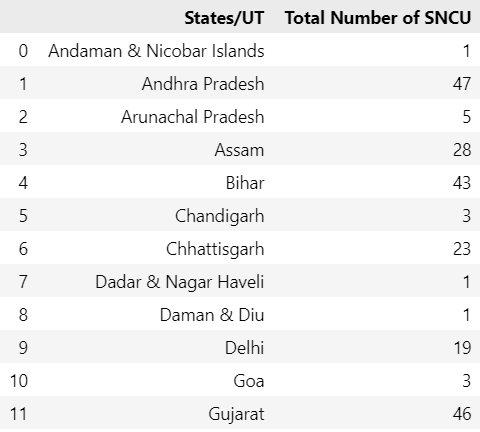
This code is importing a dataset named "StateUT-wise Established Special Newborn Care Units (SNCU) under National Health Mission (NHM) up to January, 2020.csv" and storing it in a pandas dataframe named "SNCU". It then performs some data cleaning operations such as dropping unnecessary columns, renaming a column, dropping rows with specific values, and modifying a specific value in a row.



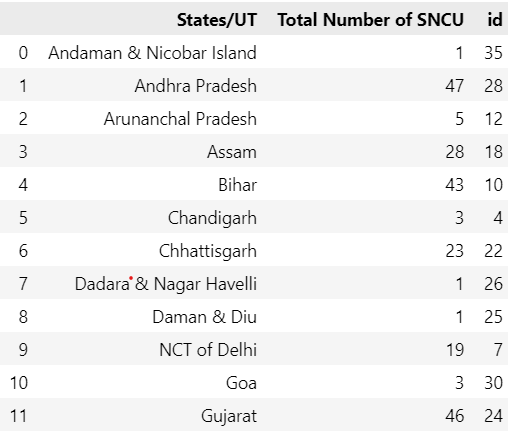
This code is loading a GeoJSON file containing the boundaries of Indian states and Union Territories, and creating a dictionary where the state/UT name is the key and the state/UT code is the value. The state/UT code is obtained from the "state\_code" property in the GeoJSON file. This dictionary can be used later for mapping data to the corresponding state/UT code.

The second prompt of the code is updating the names of certain states/union territories in the 'States/UT' column of the SNCU dataframe to match the names used in a state\_id\_map dictionary. It also creates a new column 'id' in the SNCU dataframe by applying the state\_id\_map dictionary to the 'States/UT' column.



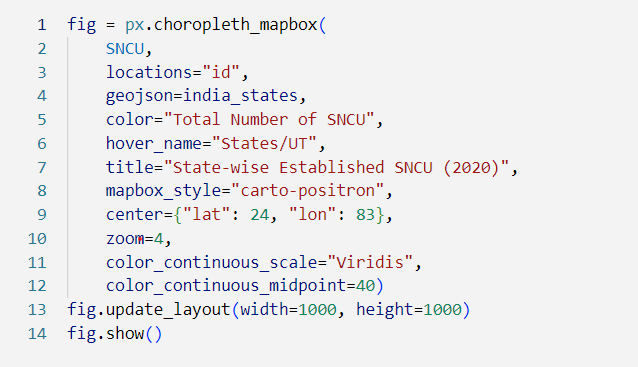


We used GeoJSON file for plotting the boundary of State/UT. We created a dictionary where state Id is key and state name is Value.



Code:

This code is creating a choropleth map using Plotly Express library in Python. The map shows the total number of SNCU (Special Newborn Care Units) established in each state/UT of India in 2020. The `SNCU` dataframe contains the data for each state/UT, `india\_states` is a geojson file containing the boundaries of each state/UT, and `id` is the column in `SNCU` that matches the state/UT names in `india\_states`. The `color` parameter specifies the column in `SNCU` that will be used to determine the color of each state/UT on the map. The `hover\_name` parameter specifies the column in `SNCU` that will be displayed when hovering over each state/UT on the map. The `title` parameter sets the title of the map. The `mapbox\_style` parameter sets the style of the map. The `center` parameter sets the center of the map. The `zoom` parameter sets the initial zoom level of the map. The `color\_continuous\_scale` parameter sets the color scale used for the map.



Findings:

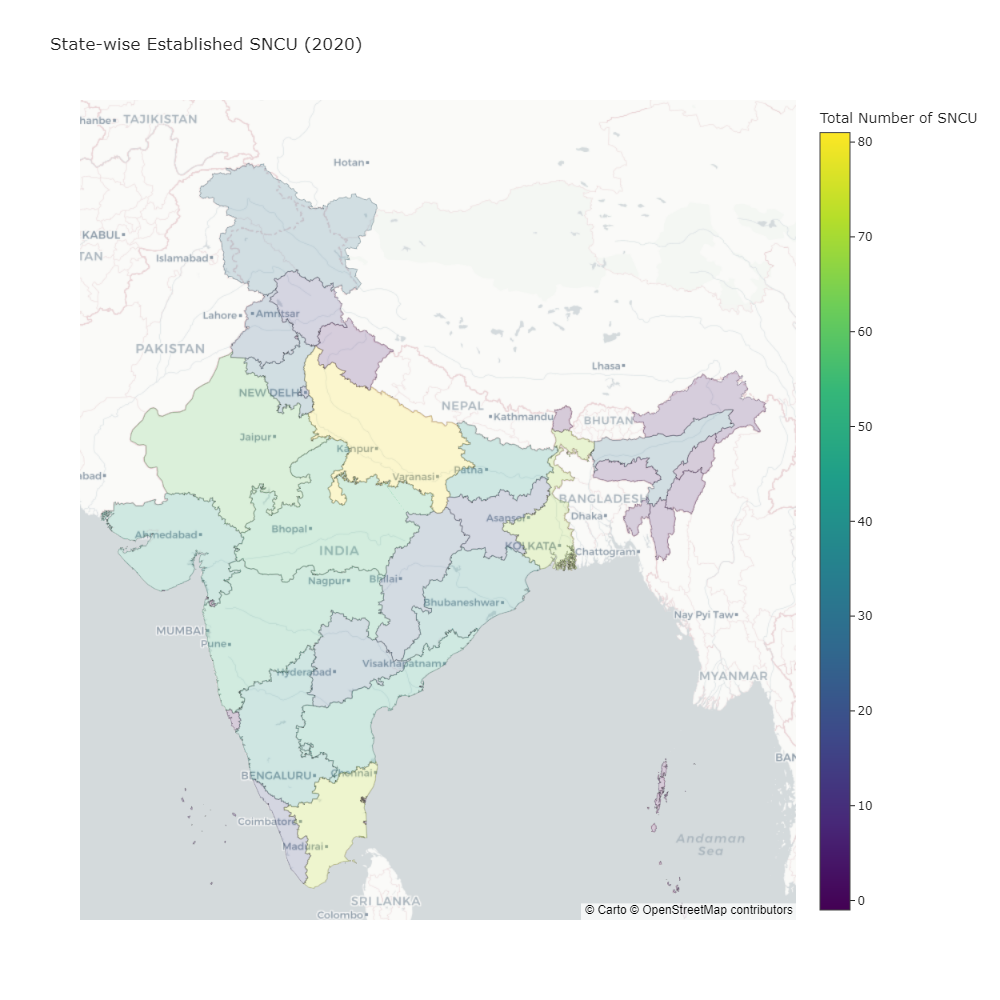
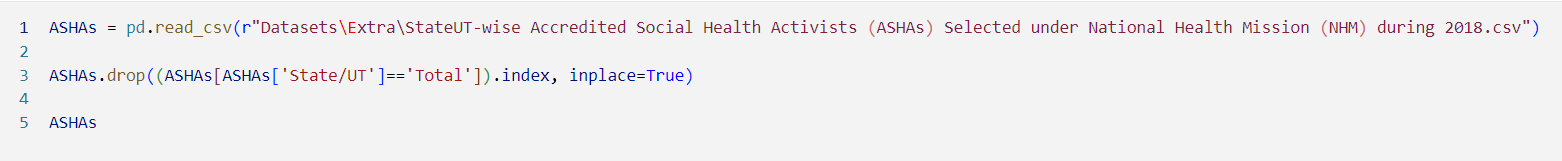


Figure 3: State-wise established SNCU (2020)

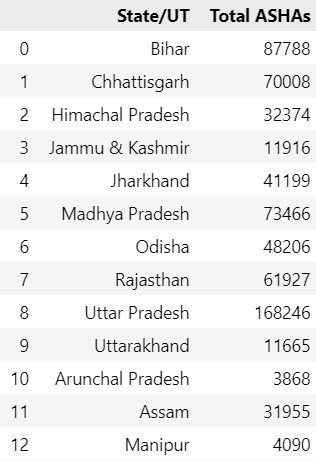
## Dataset 4: State-wise Accredited Social Health Activists (ASHAs) selected under NHM during 2018

Code:

This code is reading a CSV file named "StateUT-wise Accredited Social Health Activists (ASHAs) Selected under National Health Mission (NHM) during 2018.csv" and storing it in a pandas DataFrame called `ASHAs`.

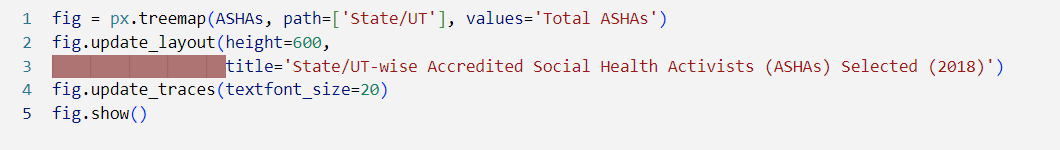


The data:



Code:

This code is creating a treemap visualization using the `px.treemap()` function from the Plotly Express library. The data used for the visualization is stored in the `ASHAs` variable. The `path` parameter specifies the hierarchical levels of the treemap, which in this case is the State/UT. The `values` parameter specifies the variable to be used for the size of the treemap tiles, which in this case is the Total ASHAs.



Findings: With the help of tree map the data has been visualized.

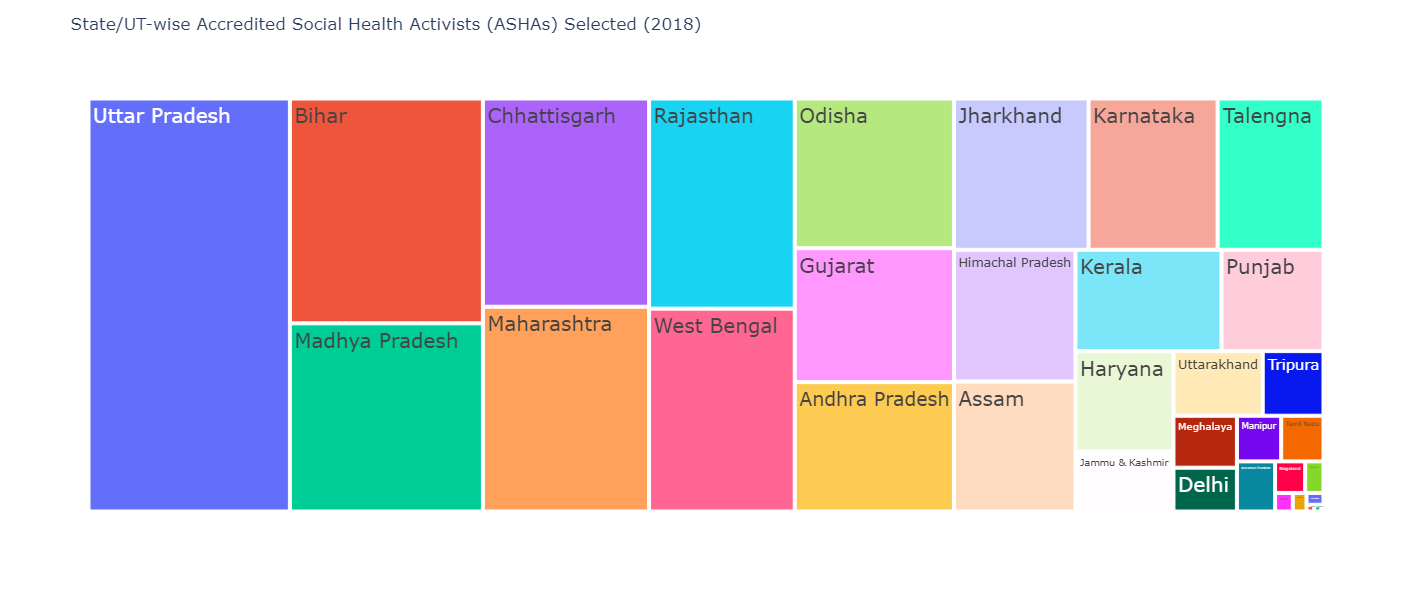


Figure 4:State wise ASHAs selected by NHRM in 2018

## Dataset 5: Geocode Health Centre & NHM July 2022

Code:

The code is reading two CSV files named 'geocode\_health\_centre.csv' and 'nhm---july-2022.csv' using pandas library's read\_csv() function. The 'low\_memory = False' parameter is used to ensure that the entire file is read into memory at once.

This code is creating a heatmap using the Seaborn library to visualize the missing values in a pandas DataFrame called `df2`. The `df2.isnull()` method returns a DataFrame with the same shape as `df2`, but with boolean values indicating whether each element is missing or not. The `sns.heatmap()` function then takes this DataFrame and creates a heatmap where missing values are represented by a blue color. The `yticklabels=False` and `cbar=False` arguments remove the y-axis tick labels and color bar respectively, while `cmap="Blues"` sets the color map to shades of blue. Finally, `plt.show()` displays the resulting plot.



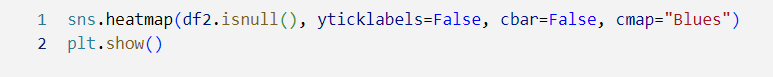
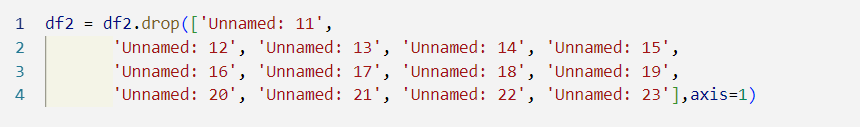




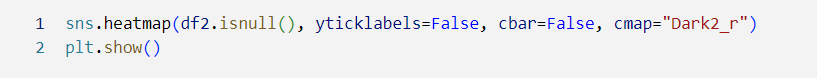
Figure 5: Heatmap to show null values in dataset

Code :

This code is dropping columns with the names ‘Unnamed: 11’ to ‘Unnamed: 23’ from the DataFrame `df2`. The `axis=1` parameter specifies that the operation should be performed on columns rather than rows. The resulting DataFrame is assigned back to the variable `df2`.



This code is creating a heatmap using the Seaborn library to visualize the missing values in a pandas DataFrame called `df2`. The `df2.isnull()` method returns a DataFrame with the same shape as `df2`, but with oolean values indicating whether each element is missing or not. The `sns.heatmap()` function then takes this DataFrame and creates a heatmap where missing values are represented by a different color (specified by the `cmap` parameter) than non-missing values. The `yticklabels` and `cbar` parameters are set to `False` to remove the y-axis tick labels and color bar respectively. Finally, `plt.show()` is used to display the resulting plot.



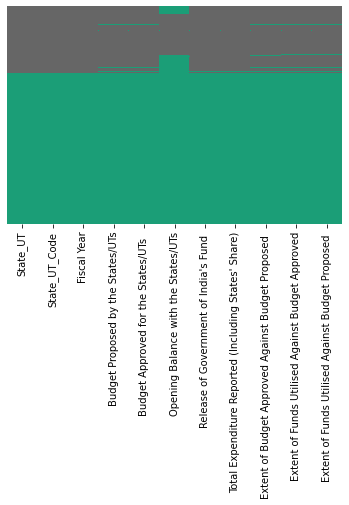
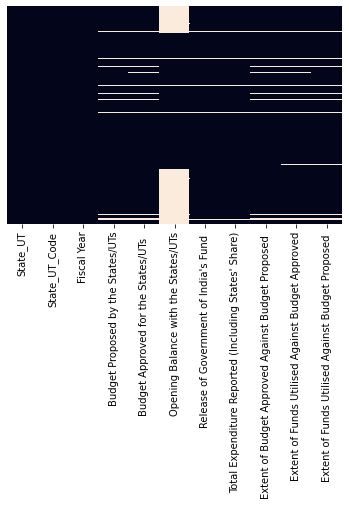


Figure 6: Null values in dataset

Since there are null values in all the columns, we can discard those values without any data loss.

Opening Balance with the States/UTs is not available for the fiscal year 2015-16, 2021-22 and 2022-23

But values for other years between these are available, thus we skip dropping this Null values



### Correlation Matrix:

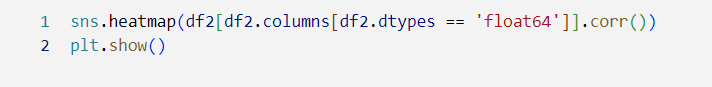
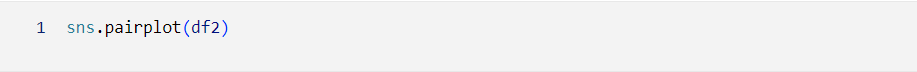




Figure 7: Correlation Matrix- comparing all the columns in the dataframe

Plotting Pairplots using seaborn



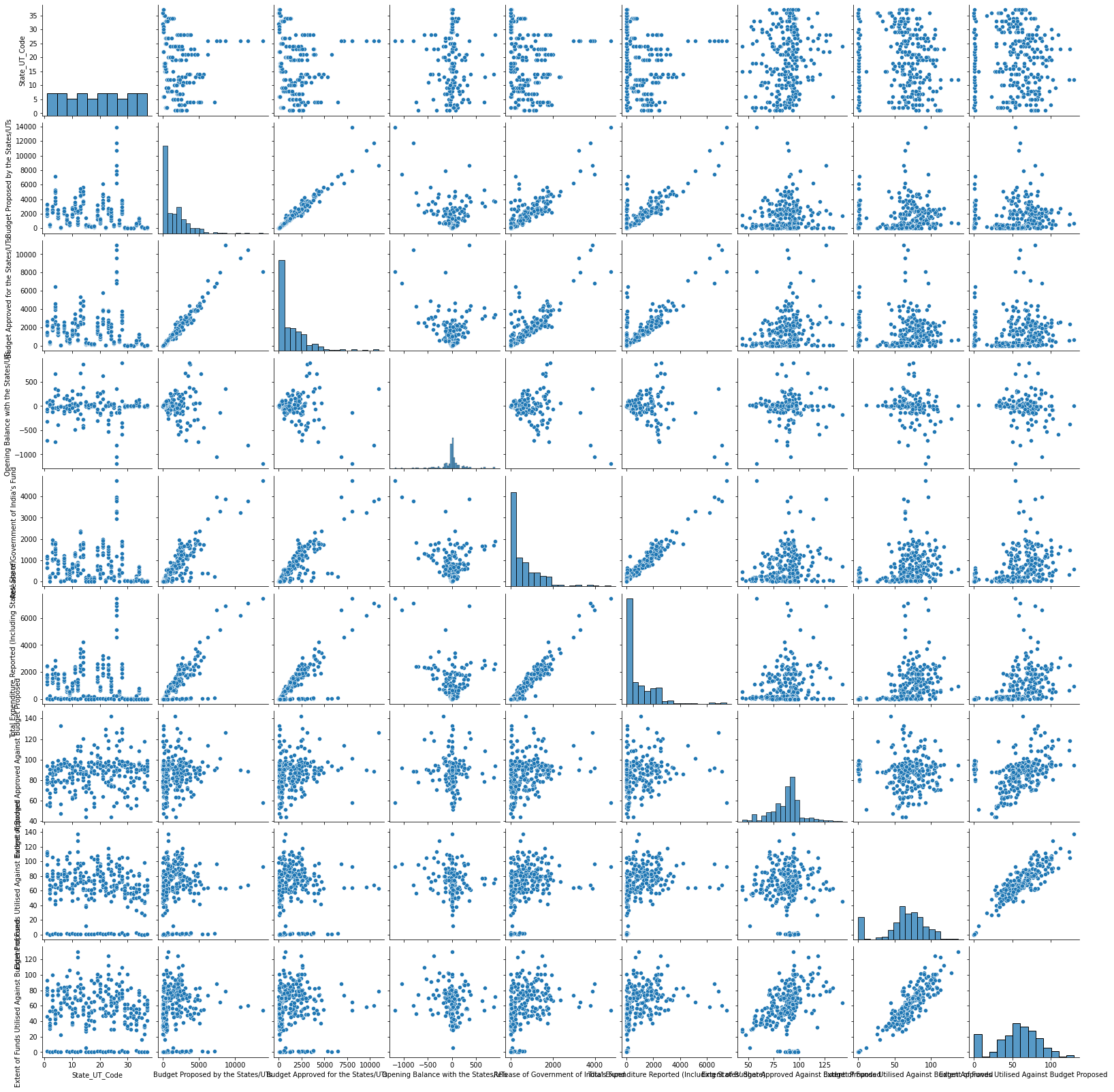


Figure 8: Plotting pairplot using seaborn of the dataframe.

Code:

This code is creating a stacked bar chart using the Plotly library to visualize the extent of budget approved against budget proposed and the extent of funds utilized against budget approved for each state/UT in each fiscal year.



Plots:

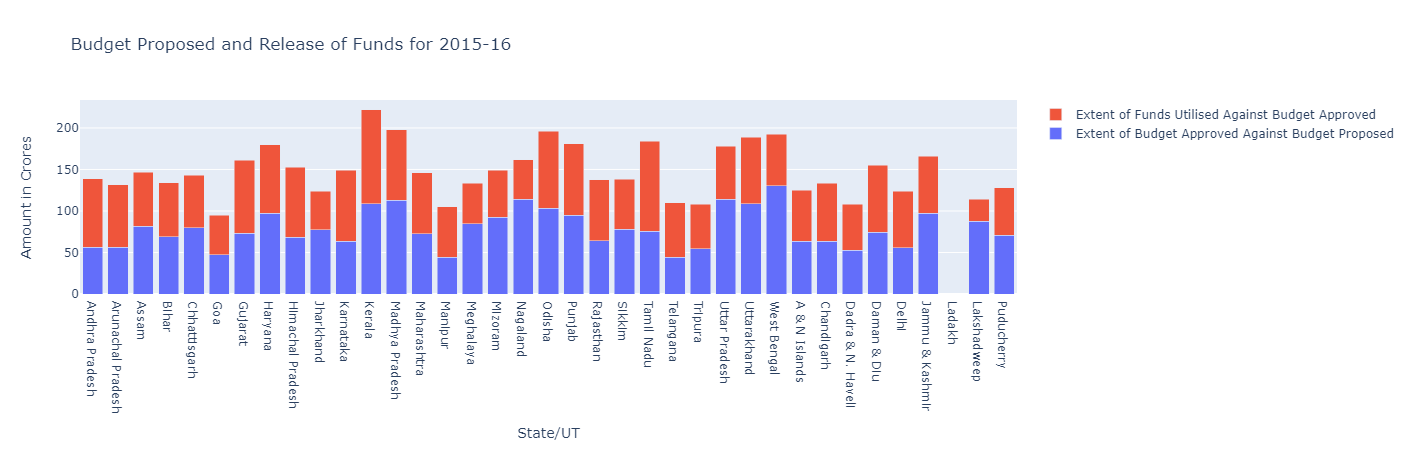


Figure 9: Budget proposed and Release of Funds for 2015 -16

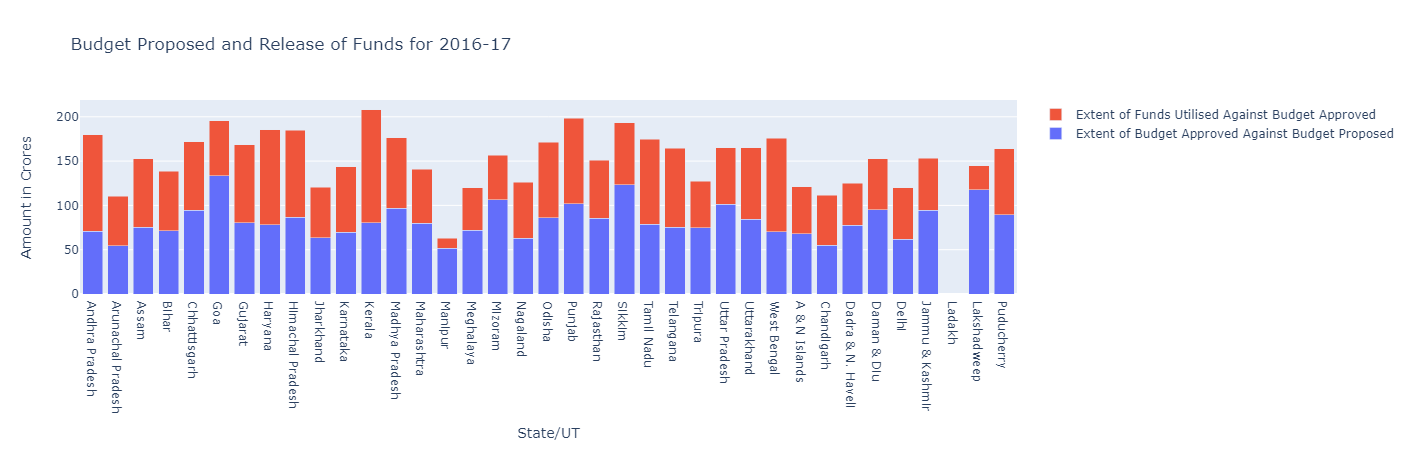


Figure 10: Budget proposed and release of funds for 2016 -17

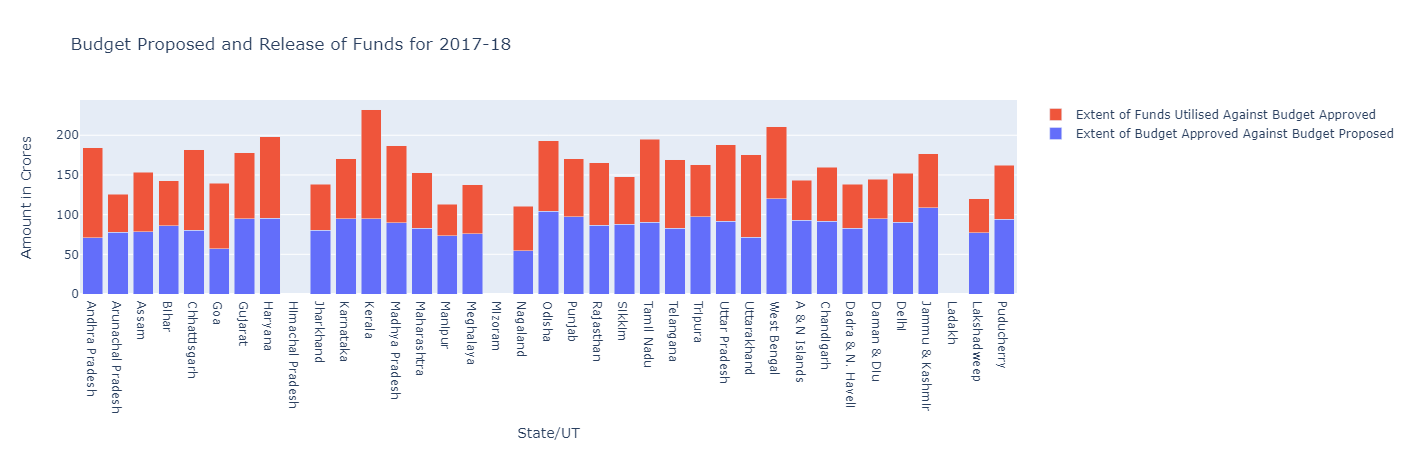


Figure 11: Budget proposed and release of funds for 2017 -18

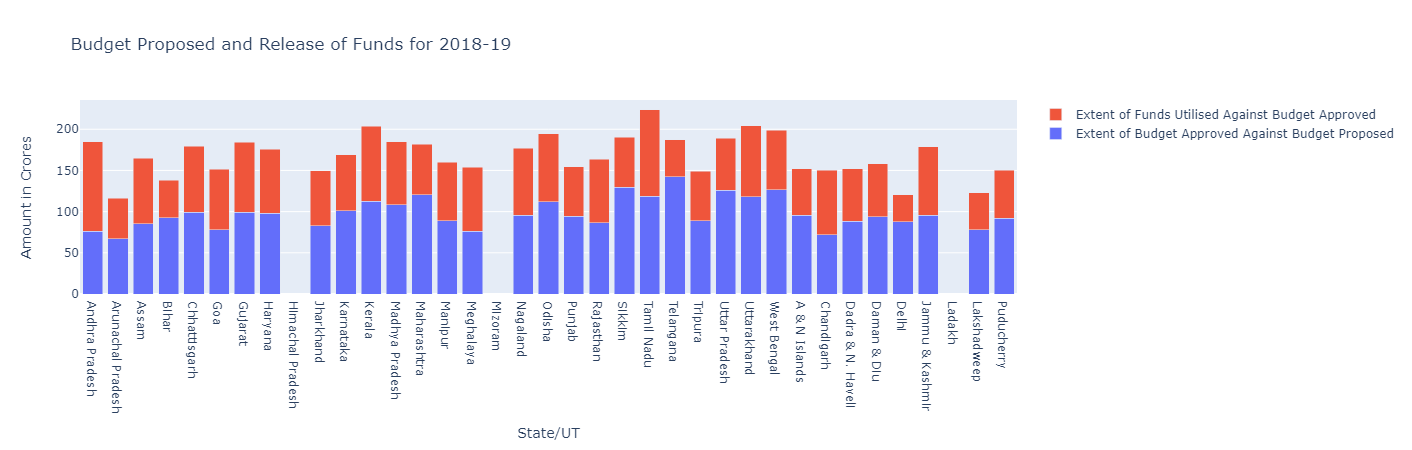


Figure 12: Budget proposed and release of funds for 2018 -19

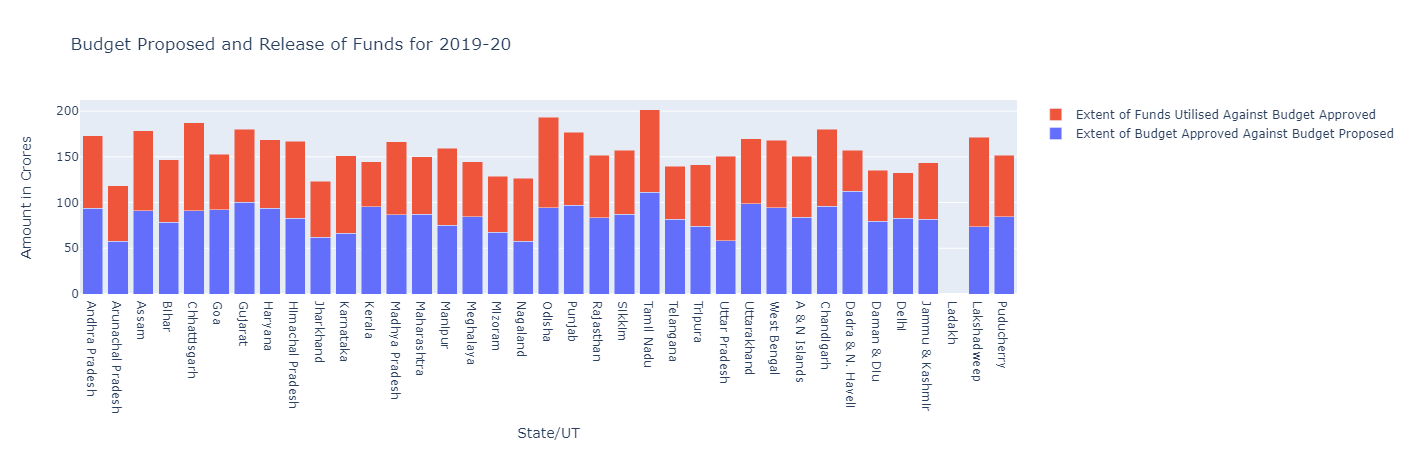


Figure 13: Budget proposed and release of funds for 2019 -20

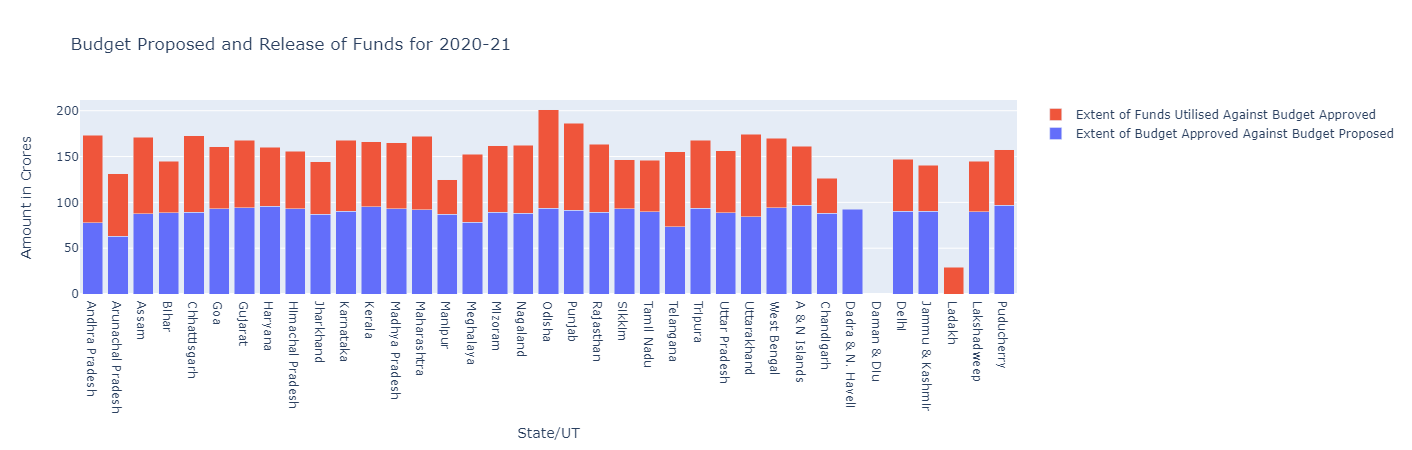


Figure 14: Budget proposed and release of funds for 2020 -21

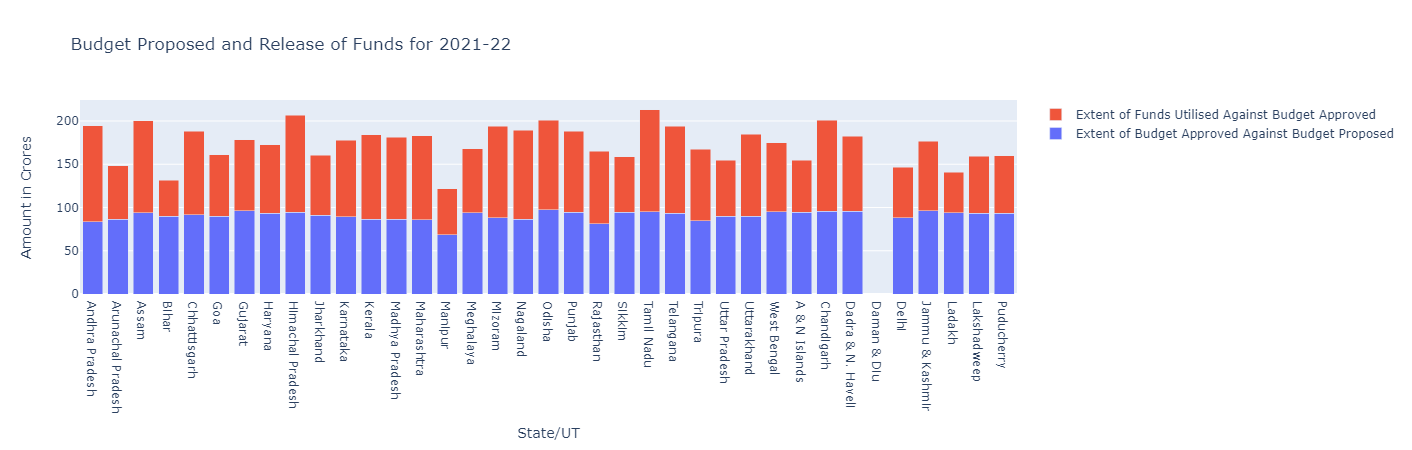


Figure 15: Budget proposed and release of funds for 2021 -22

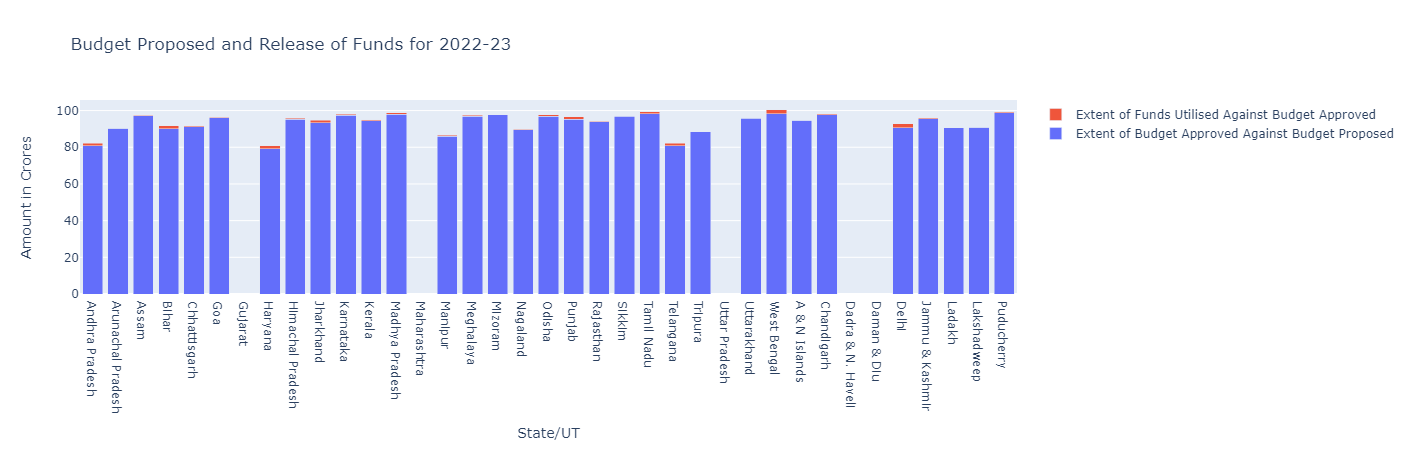


Figure 16: Budget proposed and release of funds for 2022 -23

Code:

This code is creating bar charts for each column in `df2` starting from the fourth column. It groups the data by fiscal year and calculates the mean for each year. Then, it creates a bar chart for each column with the fiscal year on the x-axis and the mean value on the y-axis. The code also updates the axis labels and displays the plot.



Plots:

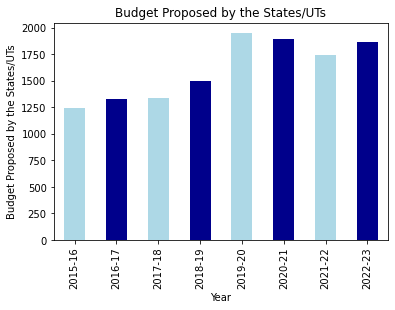


Figure 17: Bar chart of Budget Proposed by State/UTs

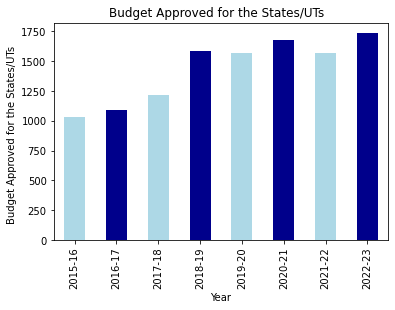


Figure 18: Bar chart of Budget approved for the state/uts

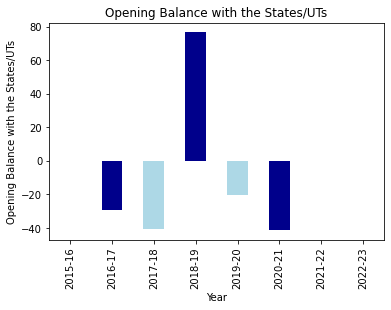


Figure 19: Opening balance with the states/ UTs

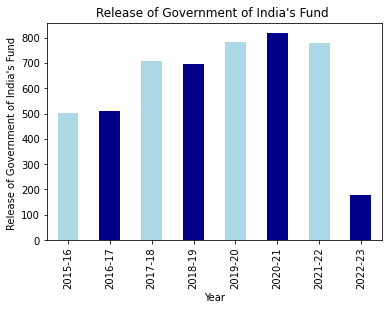


Figure 20: Release of Government of India’s Fund

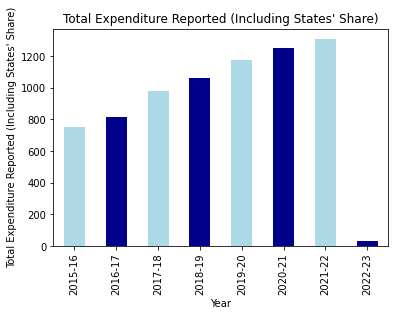


Figure 21: Total Expenditure reported (Including State’s share)

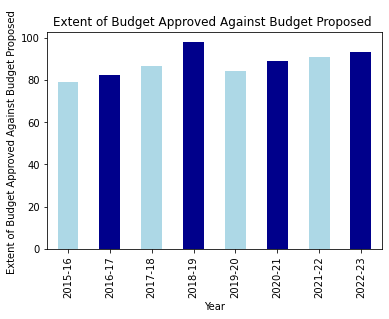


Figure 22: Extent of Budget approved against budget proposed

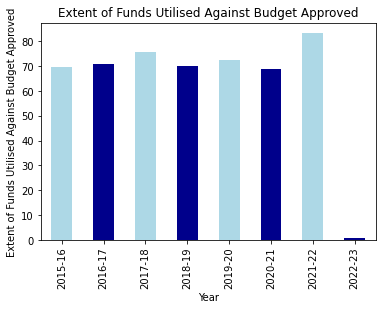


Figure 23: Extent of Funds utilised against budget approved

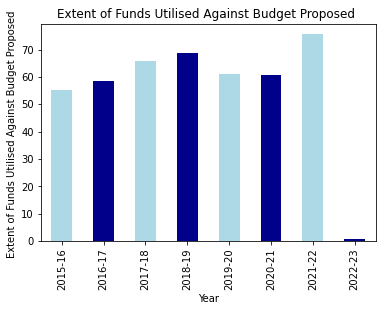


Figure 24: Extent of Funds utilised against budget approved

Code:

This code is creating grouped bar charts for each column in `df2` starting from the fourth column.



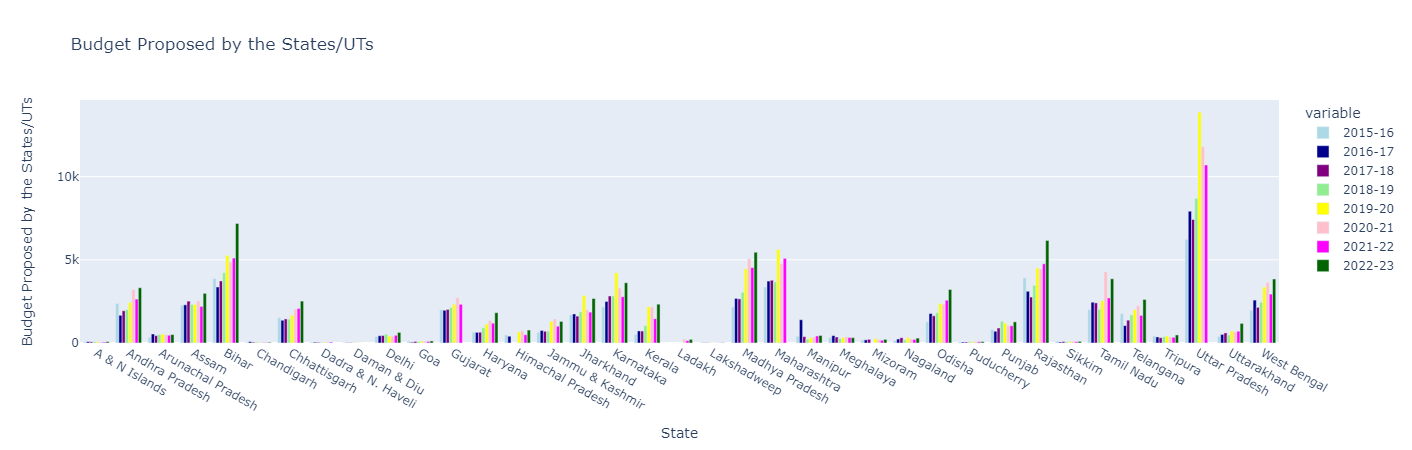


Figure 25: Budget Propsed by the State/UTs

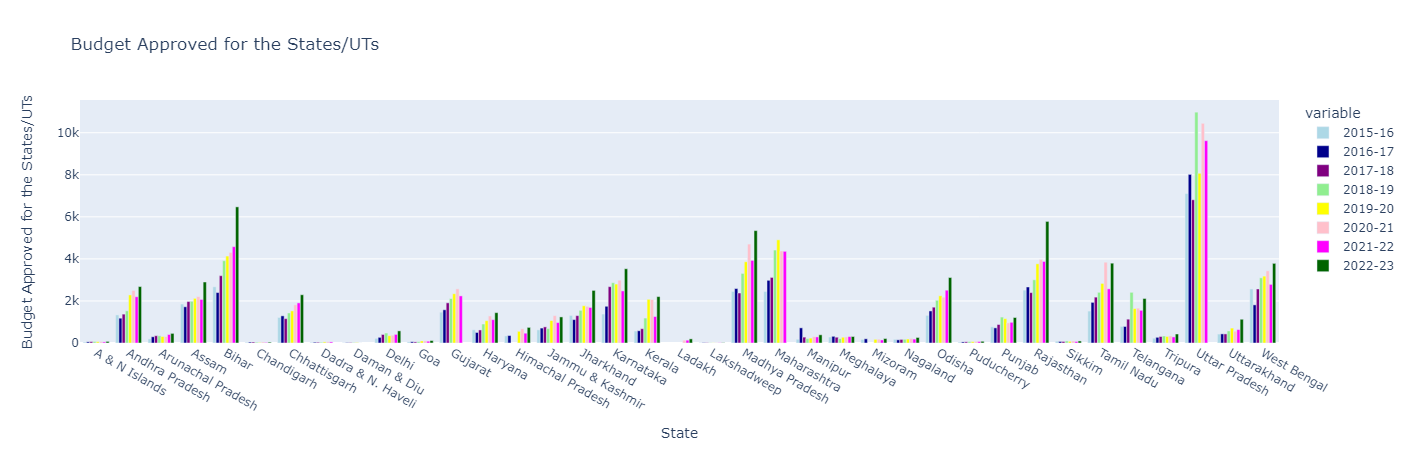


Figure 26: Budget Approved for the States/Uts

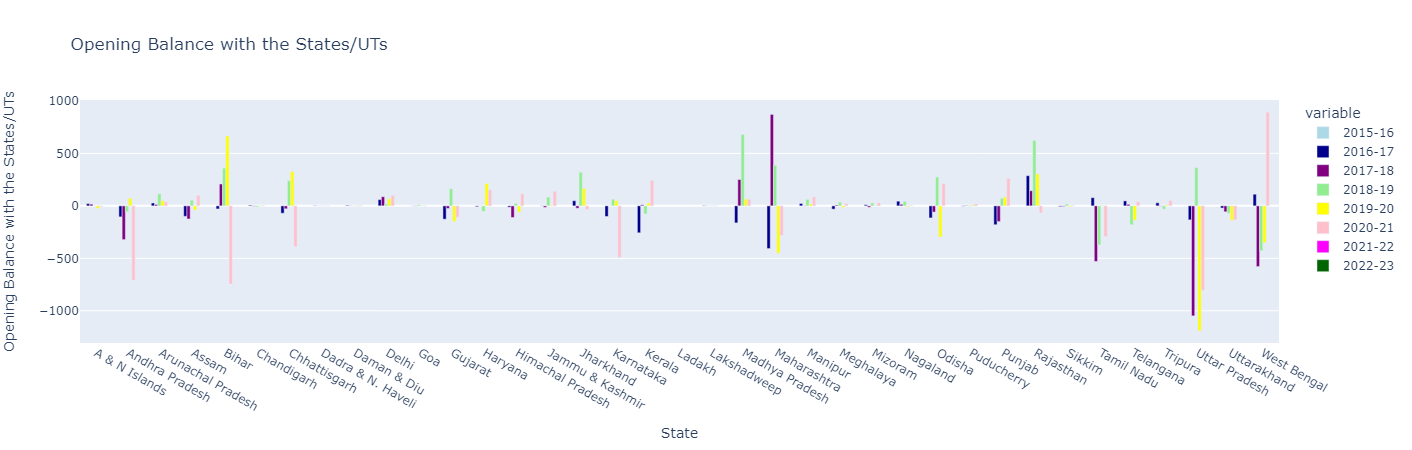


Figure 27: Opening Balance with the states/uts

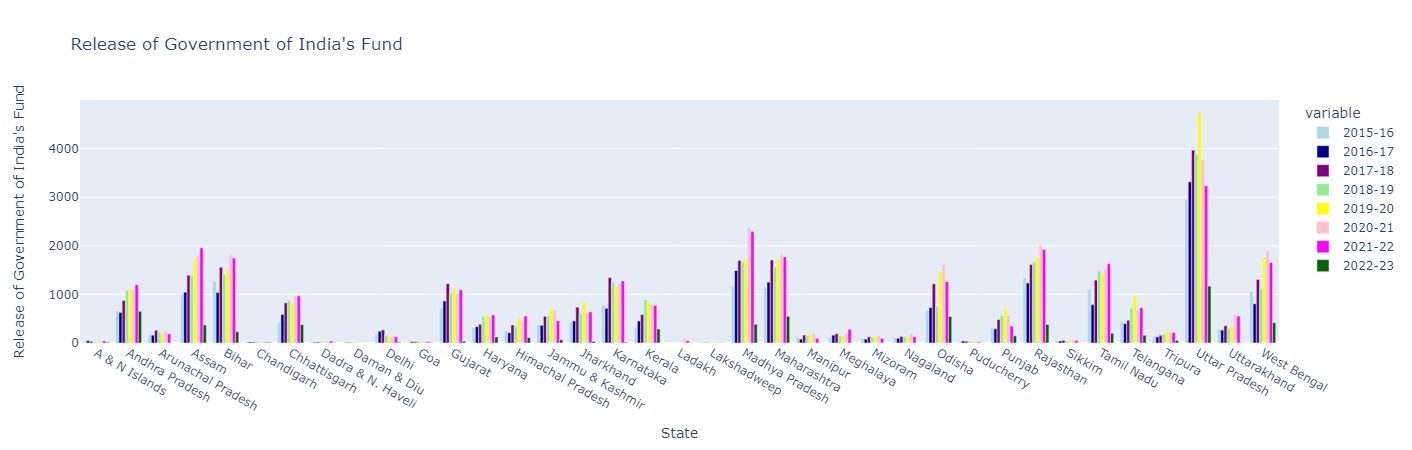


Figure 28: Release of Government of India’s Fund

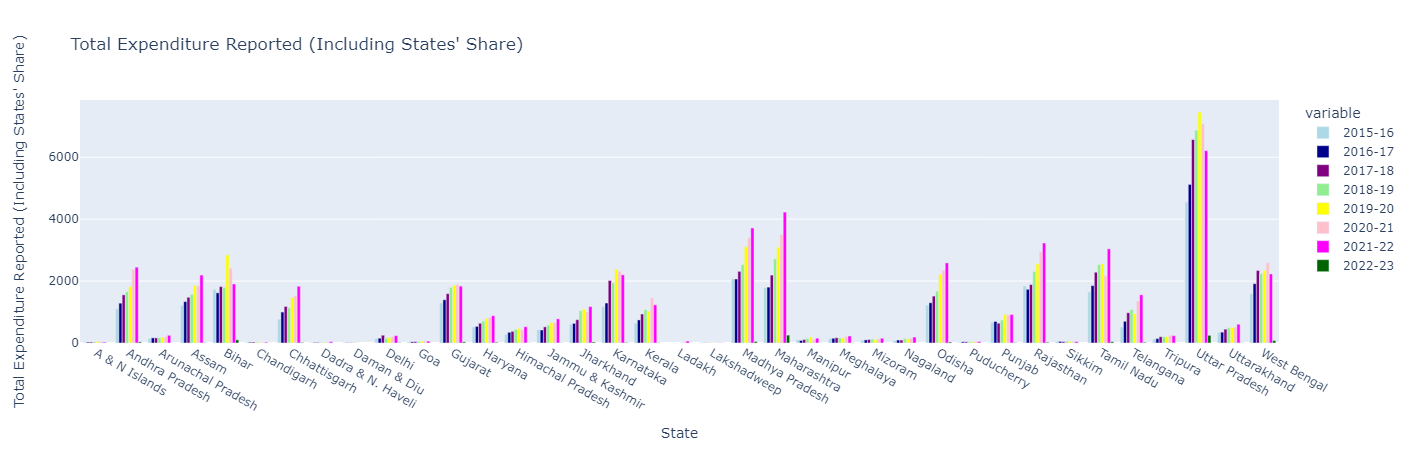


Figure 29: Total Expenditure Reported (Including State’s share)

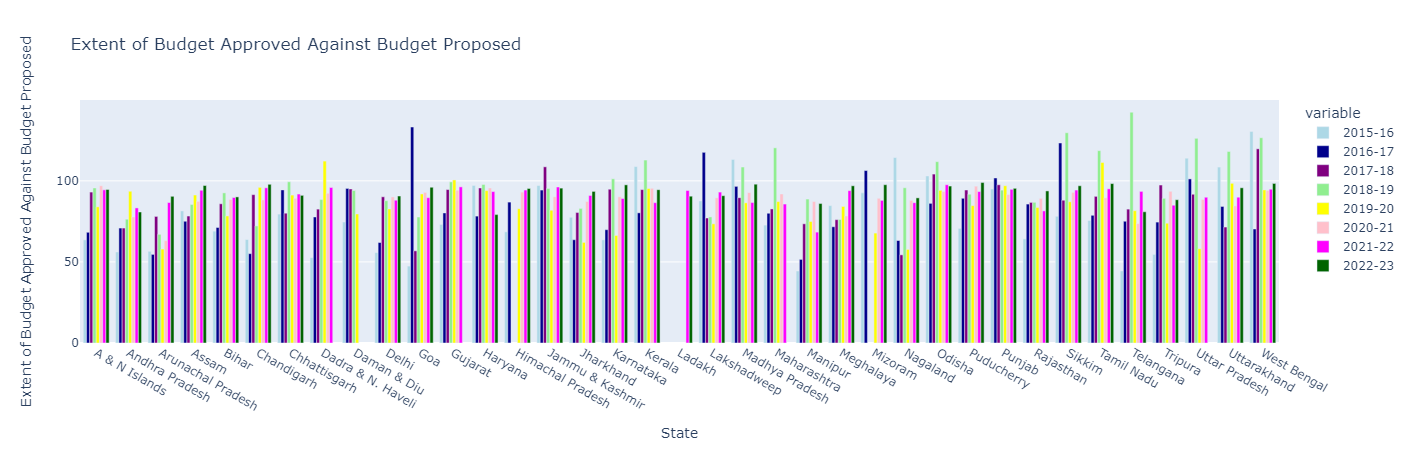


Figure 30: Extent of Budget Approved Against Budget Proposed

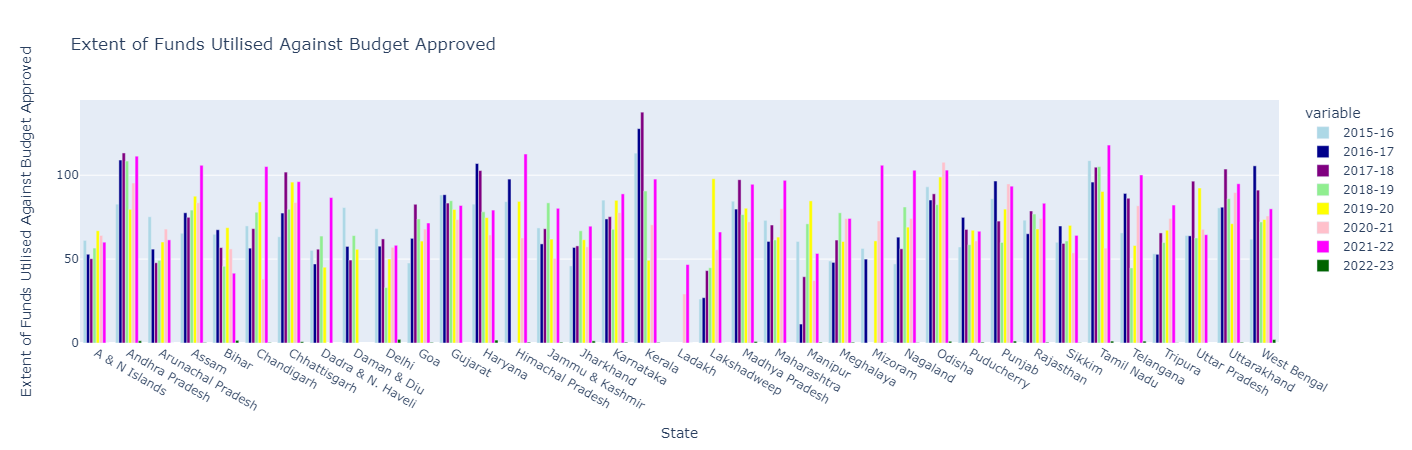


Figure 31:Extent of Funds utilised against Budget Approved

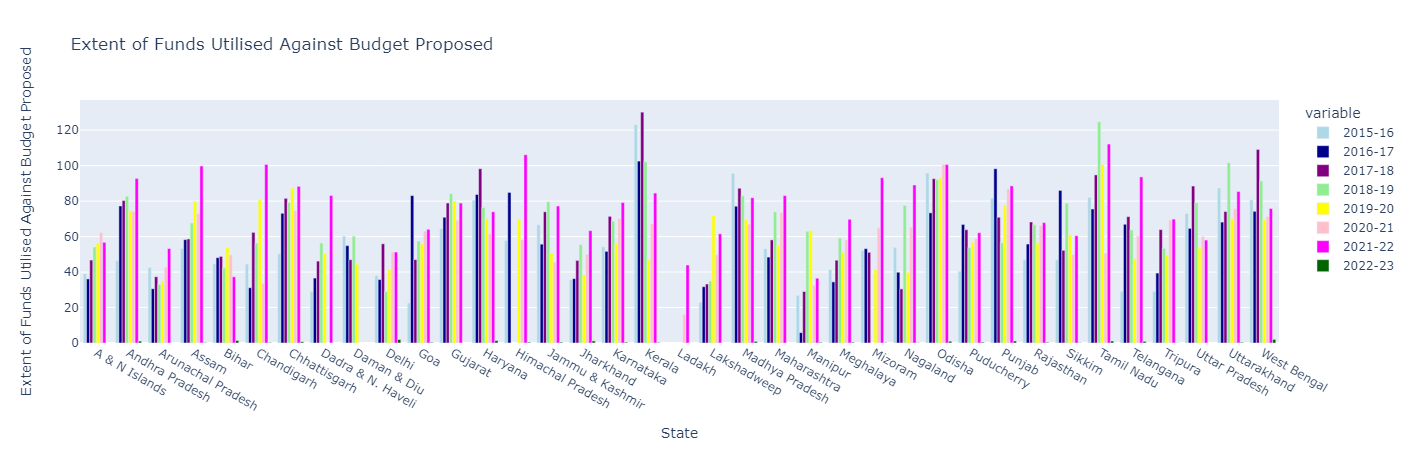
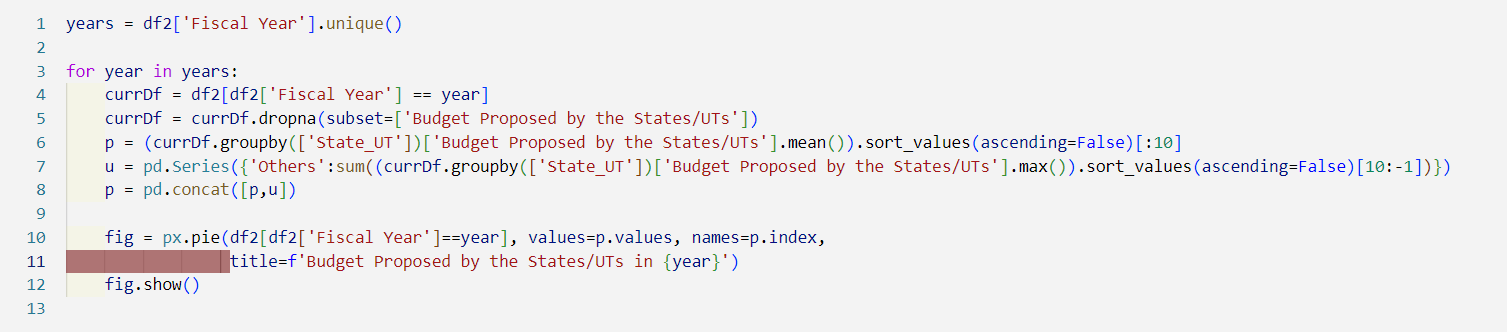


Figure 32: Extent of Funds utilised against Budget Approved

Code:



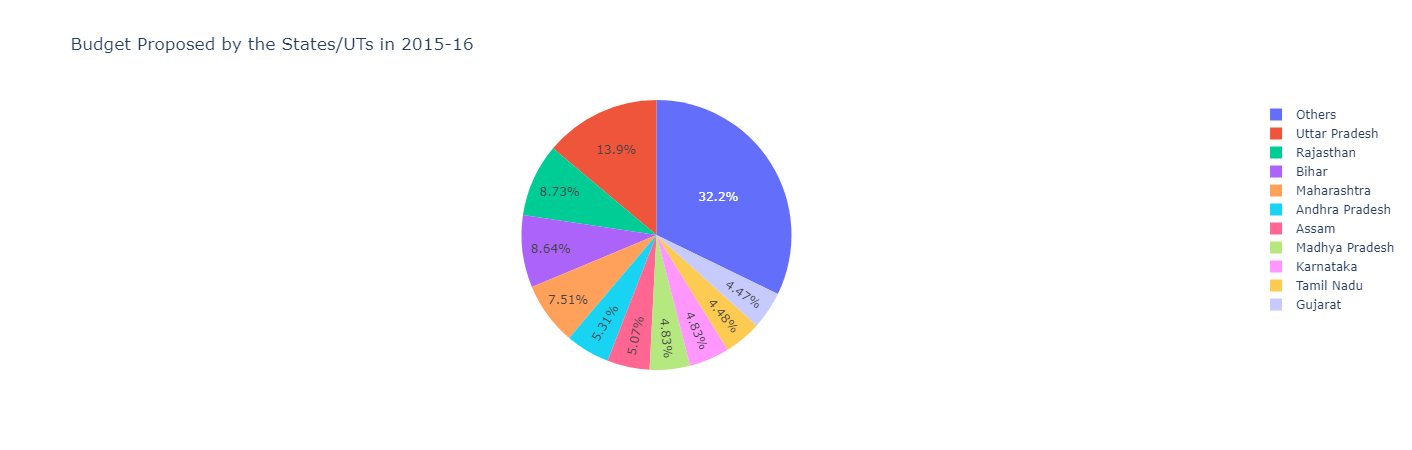


Figure 33: Budget Proposed by the States/ UTs in 2015 -16

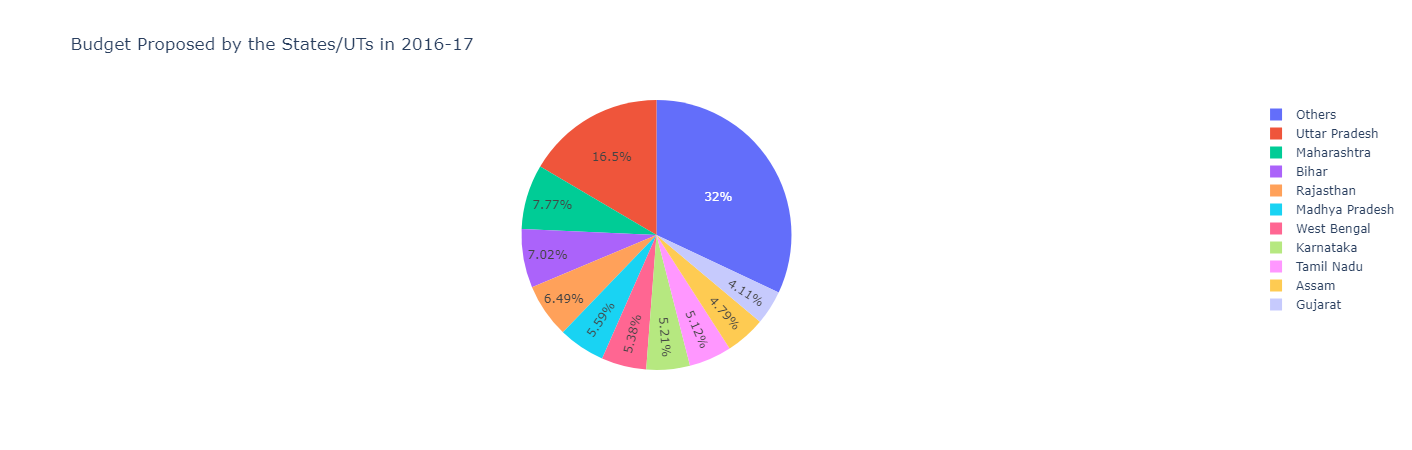


Figure 34: Budget Proposed by the States/ UTs in 2016 -17

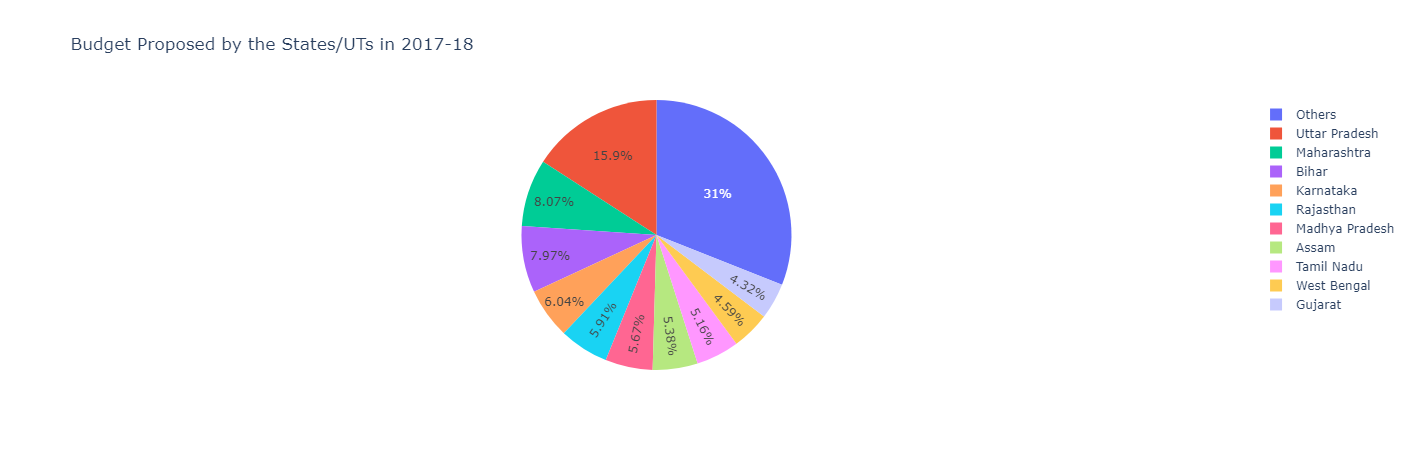


Figure 35: Budget Proposed by the States/ UTs in 2017 -18

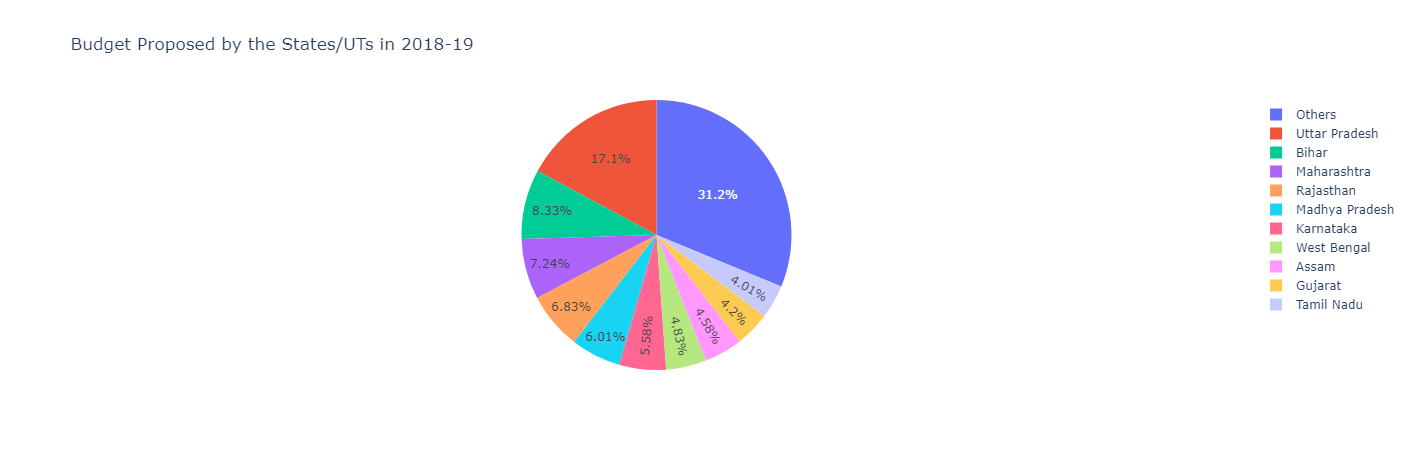


Figure 36: Budget Proposed by the States/ UTs in 2018 -19

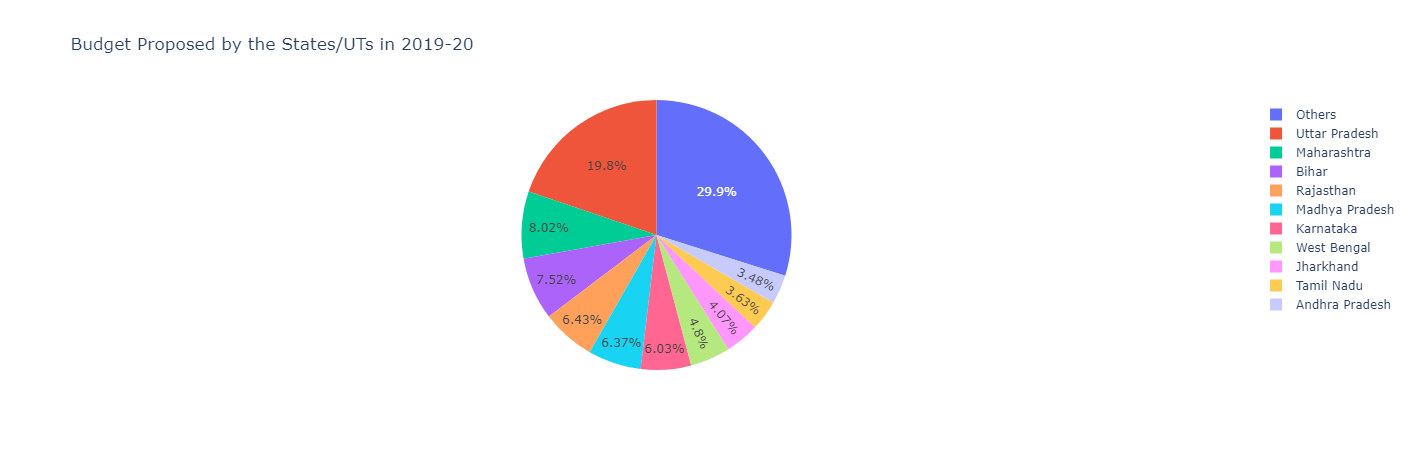


Figure 37: Budget Proposed by the States/ UTs in 2019 -20

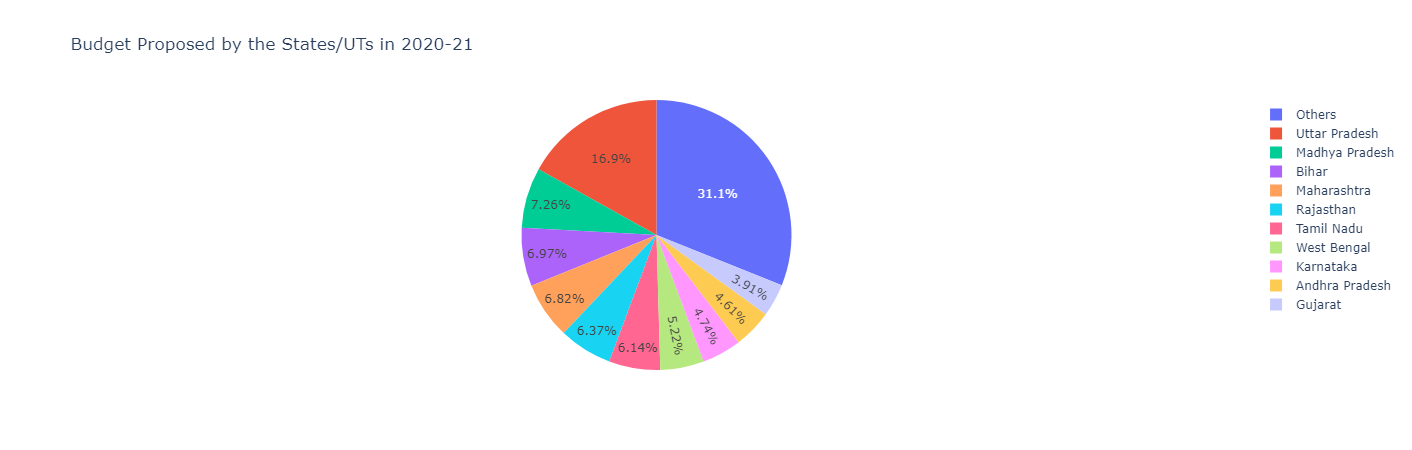


Figure 38: Budget Proposed by the States/ UTs in 2020 -21

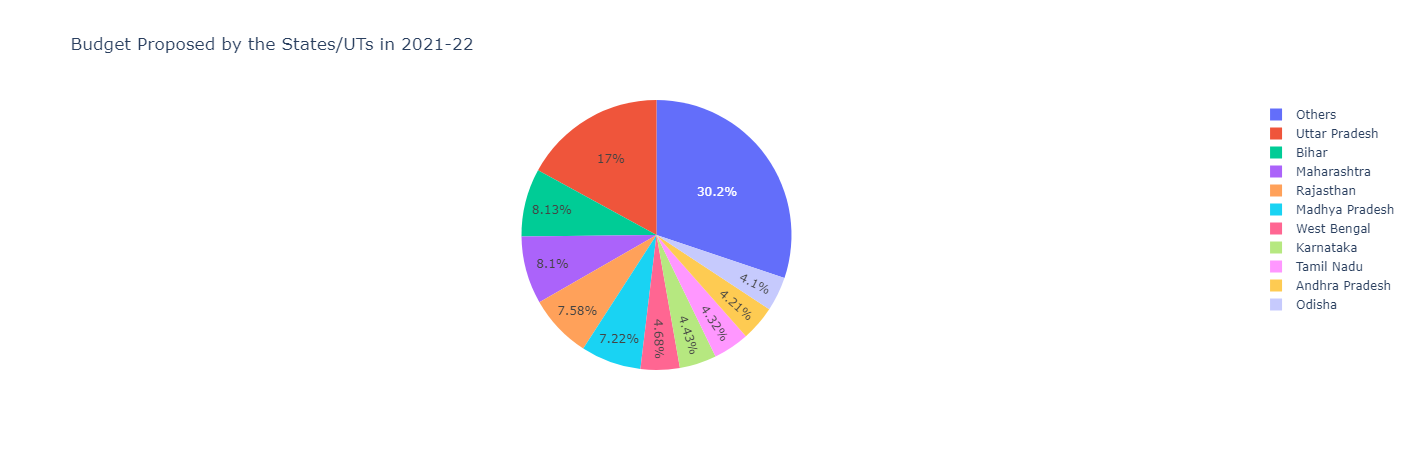


Figure 39: Budget Proposed by the States/ UTs in 2021 -22

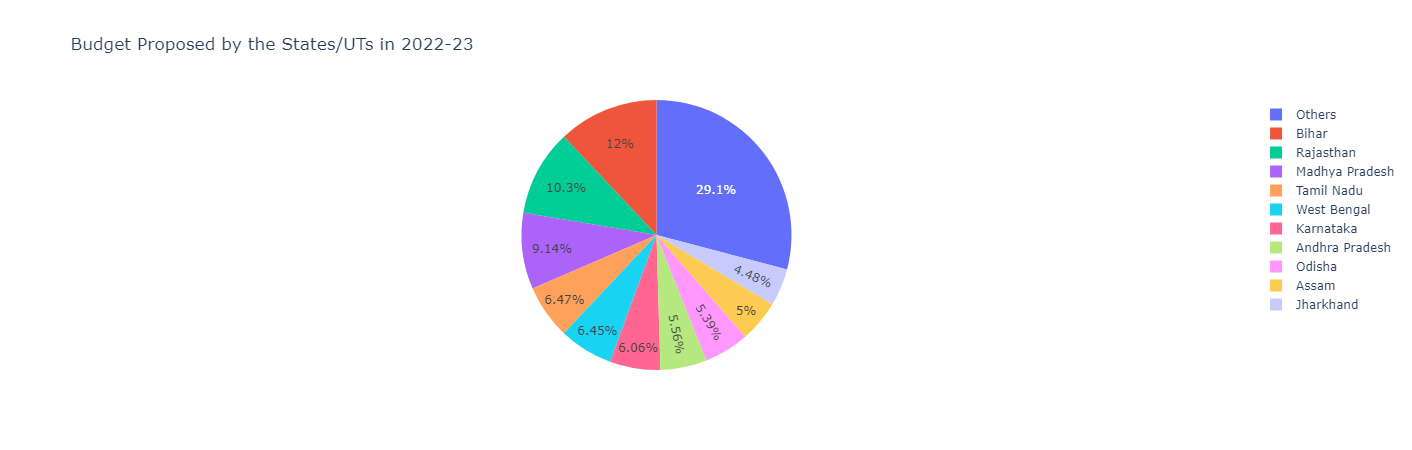


Figure 40: Budget Proposed by the States/ UTs in 2022 -23

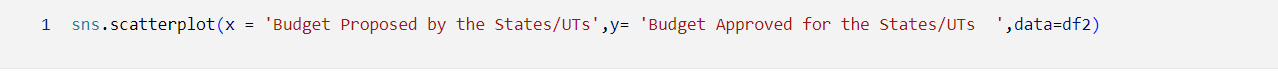
# Regression:

Applying regression model for data that seems to follow linear relationship

Regression on Budget Proposed by the States/UTs and Budget Approved for the States/UTs.

Code:

It is creating a scatter plot using the Seaborn library. The x-axis represents the budget proposed by the states/UTs and the y-axis represents the budget approved for the states/UTs. The data used for the plot is from the DataFrame `df2`.



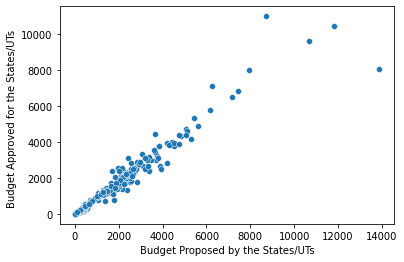
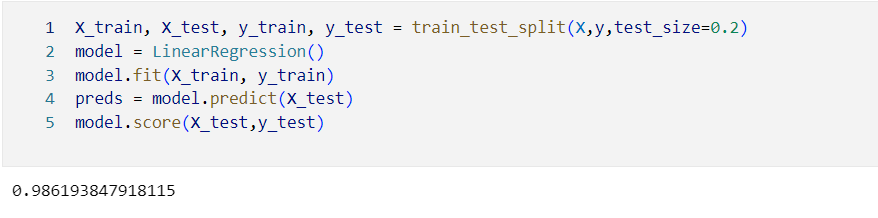


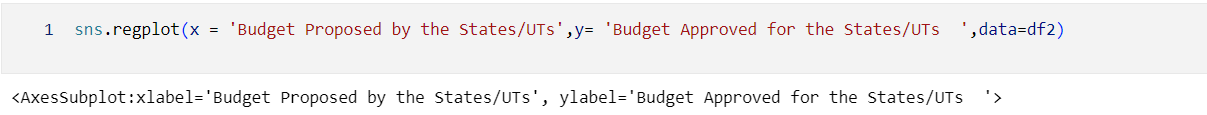
Figure 41: Linear Regression plot for the budget proposed vs the budget approved

This code is selecting two columns 'Budget Proposed by the States/UTs' and 'Budget Approved for the States/UTs' from a pandas DataFrame `df2`. The `dropna()` method is then used to remove any rows with missing values. The resulting DataFrame is stored in the variable `data`.



This code is performing a simple linear regression analysis on a dataset.





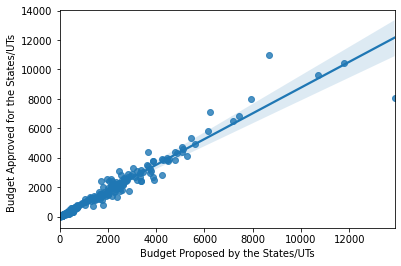
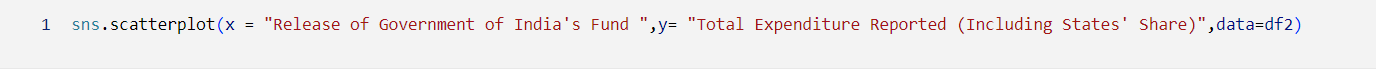


Figure 42: Linear Regression

## Regression on Release of Government of India's Fund and Total Expenditure Reported



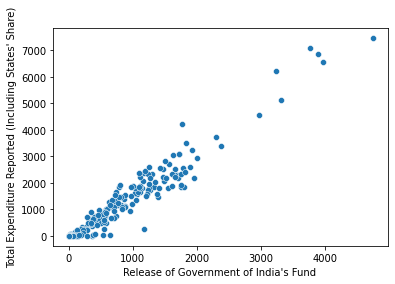
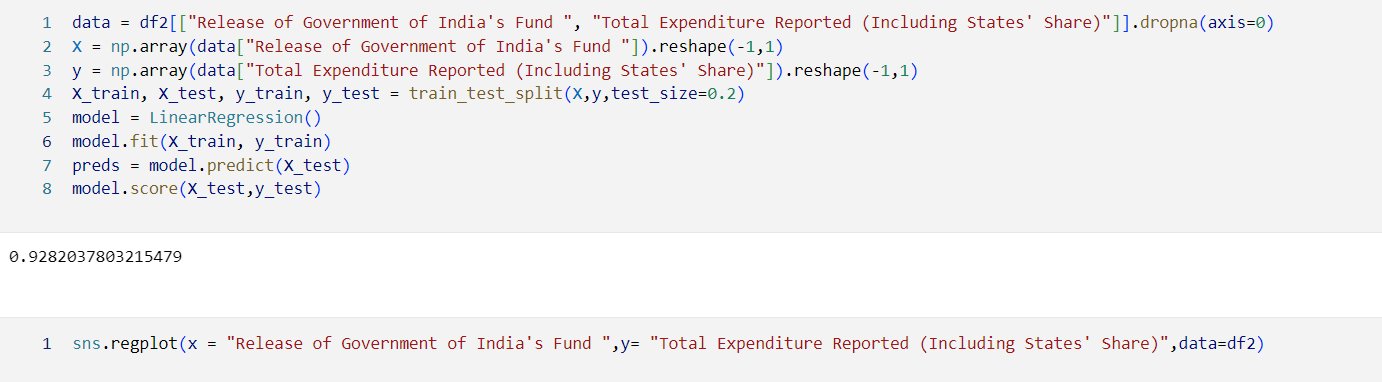
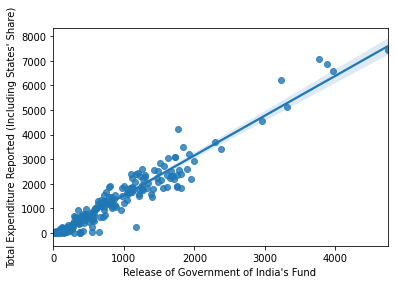


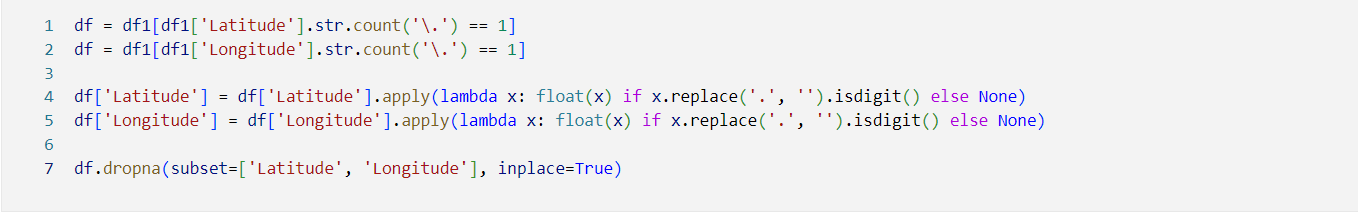
Figure 43: Linear Regression on release of Govt. Funds and total expenditure

Code:

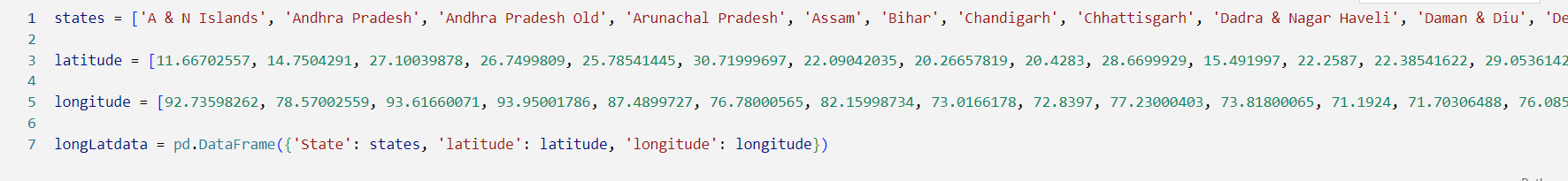
This code is performing linear regression analysis on two variables - "Release of Government of India's Fund" and "Total Expenditure Reported (Including States' Share)" - from a pandas dataframe `df2`.

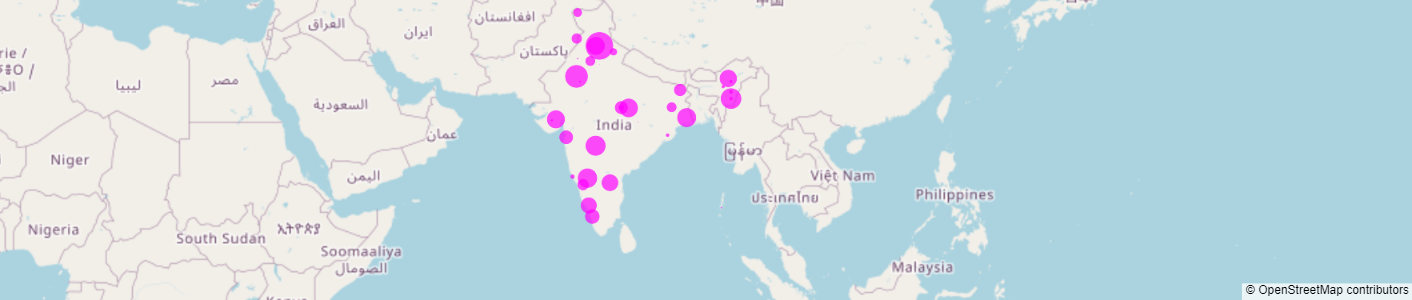






This code is creating a pandas DataFrame called `longLatdata` that contains latitude and longitude data for various states in India. The latitude and longitude data are stored in separate lists called `latitude` and `longitude`, respectively. The `states` list contains the names of the states. The `pd.DataFrame()` function is used to create the DataFrame by passing a dictionary with keys 'State', 'latitude', and 'longitude' and values `states`, `latitude`, and `longitude`, respectively.





# Conclusion:

After performing data analysis on several datasets, it can be clearly seen that the Health Sector is Improving. It is noticed that the funds allocated is used and other services like Ambulances, MMUs, ASHAs provided from the Government Of India is been properly.