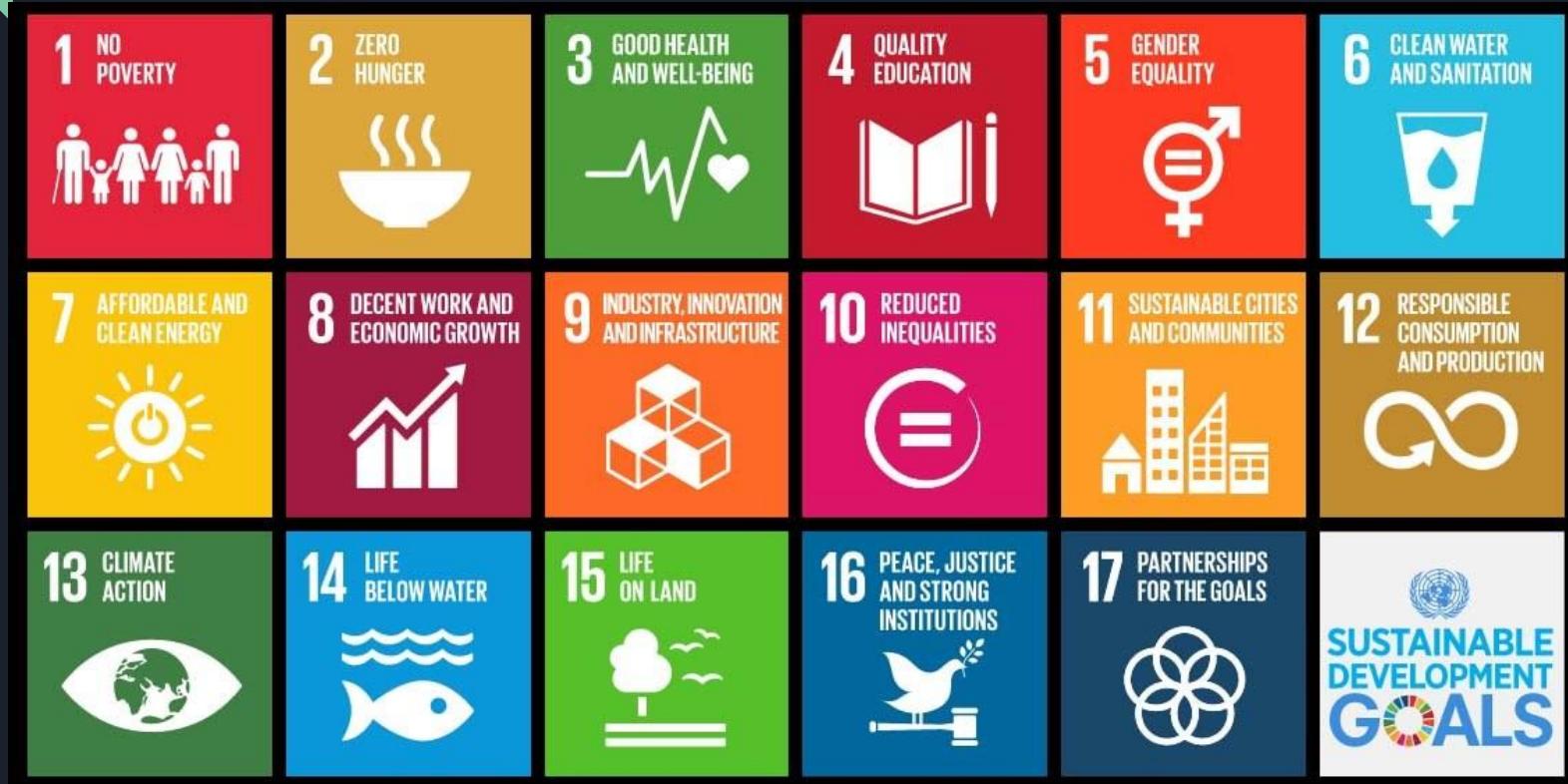




DATA SCIENCE PROJECT

Analysis of SDG index report 2020-21 of Indian States and UTs published by NITI Aayog, India





GROUP MEMBERS



1. Deepanshu Chaudhary - 2001ME21
2. Nitesh Srivastava - 2001ME42
3. Prateek Kumar - 2001ME48
4. Shresth Verma - 2001ME71
5. Ankur Singh - 2001ME84
6. Athul Krishna K - 2001ME85



INDEX

SL NO	TOPIC NAME	PAGE NO
1	Introduction	5
2	Data collection	7
3	Data Preprocessing and Cleaning	8
4	Statistical Overview	9
5	Data Visualization & Statistics	11
6	Hypothesis Testing	79
7	Conclusion	

INTRODUCTION



In 2015, the United Nations General Assembly adopted the 2030 Agenda for Sustainable Development, a plan of action for people, planet, prosperity and peace. It compiles a list of 17 Sustainable Development Goals and 169 targets that several countries including India have pledged to comply with. NITI Aayog, India, published the SDG scores for 2020-21, for each of the states and union territories in India, and along with the overall score for India. We have analyzed the SDG scores for 2020-21 given by NITI aayog and tried to analyze the performances of different states.

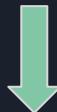




DATA COLLECTION



DATA PROCESSING AND CLEANING



STATISTICAL AND VISUAL (GRAPHICAL) ANALYSIS OF DATA



HYPOTHESIS TESTING



DATA COLLECTION

We collected data from Kaggle which contained 18 .csv files (16 SDG files and 2 composite scores) and a report pdf. The report can also be accessed on the NITI Aayog,s official website. All the references links are:

Source : <https://www.kaggle.com/chandramanaha/inian-states-sdg-index-202021>

NITI Aayog : <https://sdgindiaindex.niti.gov.in/#/>

Rankings : <https://sdgindiaindex.niti.gov.in/#/ranking>



NITI Aayog

DATA PREPROCESSING & CLEANING

Step 1: Import the libraries

```
In [1]: import pandas as pd  
import numpy as np  
import matplotlib.pyplot as plt  
import os  
import seaborn as sns
```

Step 2: Reading the csv file using pandas and storing the data in dataframe df

```
In [2]: df=pd.read_csv('cs.csv')  
  
In [3]: df.head()  
  
Out[3]:
```

	Category	State/UT	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5	SDG 6	SDG 7	SDG 8	SDG 9	SDG 10	SDG 11	SDG 12	SDG 13	SDG 15	SDG 16	Composite Score	Rank
0	State	Andhra Pradesh	81	52	77	50	58	92	100	67	52	74	78	84	63	69	77	72	4
1	State	Arunachal Pradesh	54	66	64	41	37	67	85	50	31	69	39	77	58	93	64	60	22
2	State	Assam	51	41	59	43	25	64	98	50	39	65	55	66	53	78	62	57	26
3	State	Bihar	32	31	66	29	48	91	78	50	24	48	67	59	16	62	73	52	28
4	State	Chhattisgarh	49	37	60	55	64	89	78	64	36	72	78	64	38	65	71	61	19

Step 3: Check out the missing values

```
In [6]: df.isnull().sum()

Out[6]: SNO          0
Category      0
States/UTs    0
Percentage of targeted habitations connected by all-weather roads under Pradhan Mantri Gram Sadak Yojana (PMGSY) 6
Percentage Share of GVA in manufacturing to total GVA (current prices) 3
Manufacturing employment as a percentage of total employment 0
Innovation score as per the India Innovation Index 0
Score as per Logistics Ease Across Different States (LEADS) report 2
Number of mobile connections per 100 persons (mobile tele density) 0
Number of internet subscribers per 100 population 6
dtype: int64
```

Step 4: After rejecting null values using .dropna() function in pandas

Step 5: Finally checking for missing values

```
In [8]: df.isnull().sum()

Out[8]: SNO          0
Category      0
States/UTs    0
Percentage of targeted habitations connected by all-weather roads under Pradhan Mantri Gram Sadak Yojana (PMGSY) 0
Percentage Share of GVA in manufacturing to total GVA (current prices) 0
Manufacturing employment as a percentage of total employment 0
Innovation score as per the India Innovation Index 0
Score as per Logistics Ease Across Different States (LEADS) report 0
Number of mobile connections per 100 persons (mobile tele density) 0
Number of internet subscribers per 100 population 0
dtype: int64
```

STATISTICAL OVERVIEW OF ALL SDGs

In [4]: df.describe()

Out[4]:

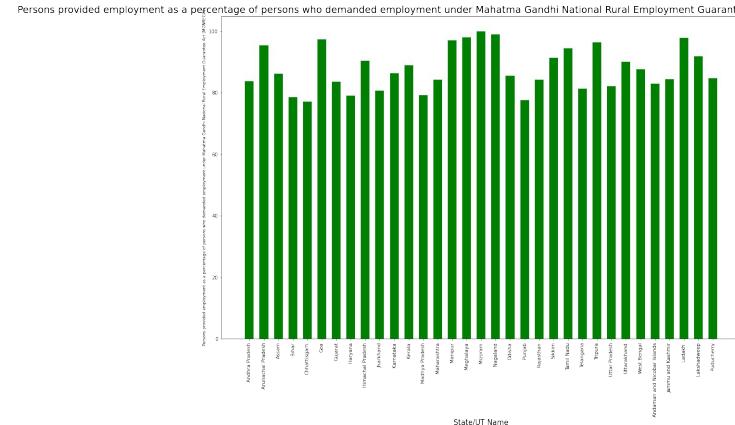
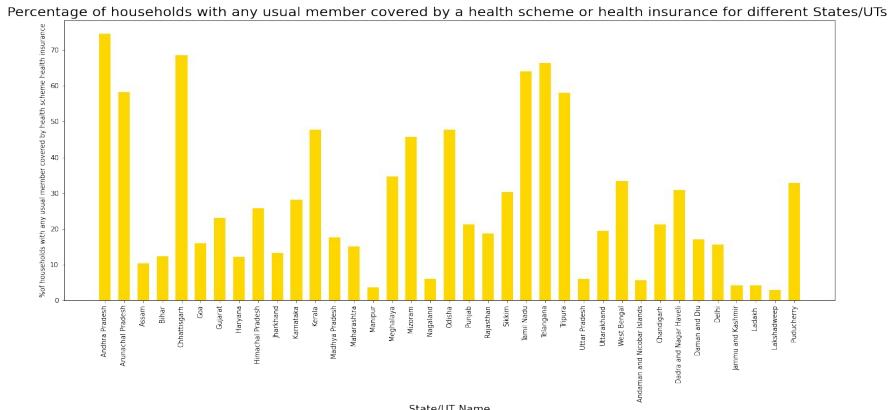
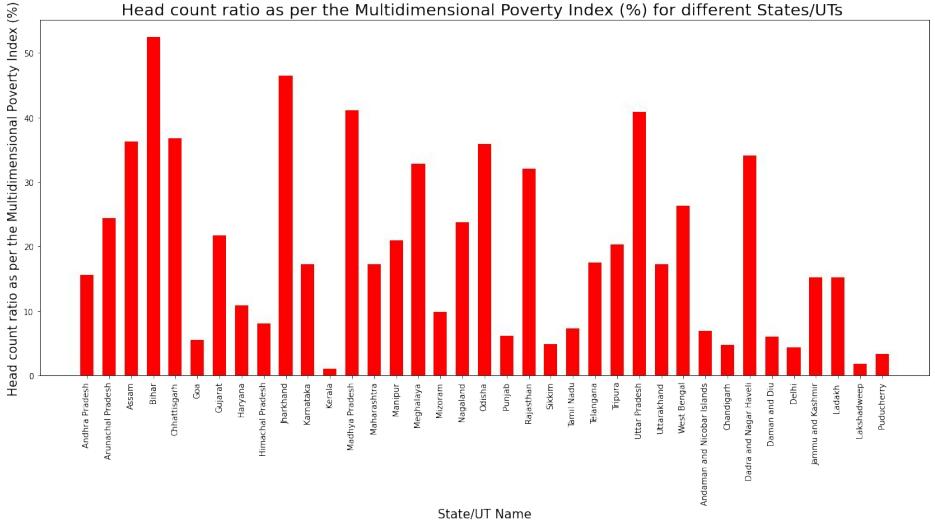
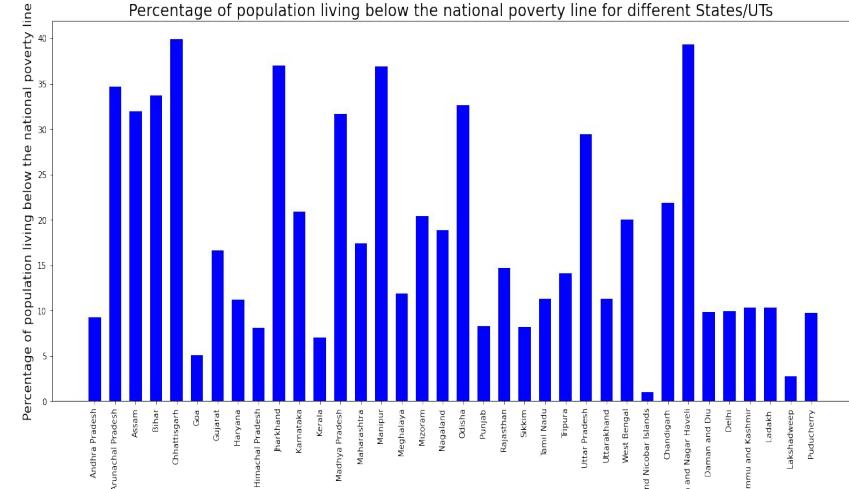
	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5	SDG 6	SDG 7	SDG 8	
count	36.000000	36.000000	36.000000	36.000000	36.000000	36.000000	36.000000	36.000000	
mean	66.500000	55.472222	71.805556	57.111111	50.111111	84.555556	92.333333	59.833333	
std	14.456832	16.538243	7.645052	12.101423	9.662232	10.656841	12.125534	9.060117	
min	32.000000	19.000000	59.000000	29.000000	25.000000	54.000000	50.000000	36.000000	
25%	59.750000	43.750000	67.000000	48.750000	44.500000	82.750000	84.500000	53.750000	
50%	69.000000	53.000000	71.000000	57.500000	50.500000	87.000000	100.000000	59.500000	
75%	79.250000	66.750000	77.250000	64.000000	58.000000	90.250000	100.000000	65.250000	
max	86.000000	97.000000	90.000000	80.000000	68.000000	100.000000	100.000000	78.000000	
	SDG 9	SDG 10	SDG 11	SDG 12	SDG 13	SDG 15	SDG 16	Composite Score	Rank
36.000000	36.000000	36.000000	36.000000	36.000000	36.000000	36.000000	36.000000	36.000000	36.000000
48.138889	67.527778	72.222222	74.333333	53.277778	64.527778	72.583333	66.055556	11.638889	
14.552668	11.596763	14.178678	12.897397	15.331780	13.635010	7.969137	5.980657	8.394395	
23.000000	41.000000	39.000000	47.000000	16.000000	27.000000	46.000000	52.000000	1.000000	
36.750000	64.750000	64.000000	65.750000	43.750000	57.500000	69.750000	61.000000	4.000000	
46.500000	67.500000	76.000000	76.500000	58.000000	64.000000	73.000000	67.000000	10.000000	
60.250000	72.500000	81.000000	82.000000	63.500000	72.250000	77.500000	70.250000	19.000000	
72.000000	100.000000	98.000000	99.000000	77.000000	93.000000	86.000000	79.000000	28.000000	

DATA VISUALIZATION & STATISTICS

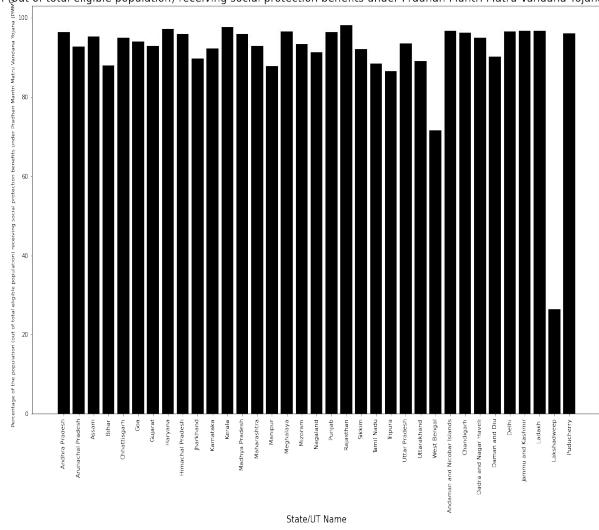


SDG 1 : No Poverty

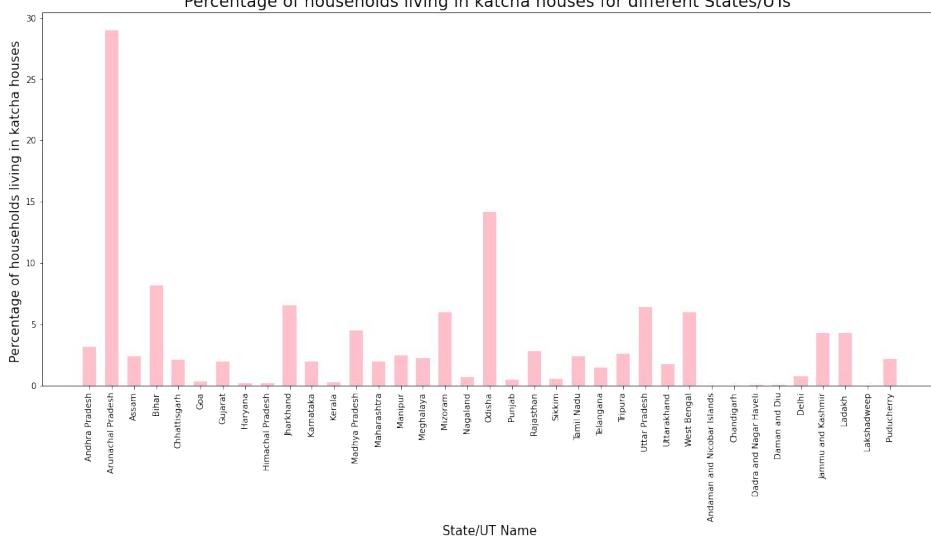
```
#code for plotting bar graph for SDGs indicators:  
#this code is for SDG 1 first indicator, similarly we have done for all SDGs.|  
fig=plt.figure(figsize=[5,2])  
ax= fig.add_axes([0,0,3,3])  
xaxis=df3['States/UTs']  
yaxis=df3['Percentage of population living below the national poverty line']  
plt.xlabel("State/UT Name",fontsize=15)  
plt.ylabel("Percentage of population living below the national poverty line",fontsize=15)  
plt.xticks(rotation=90)  
plt.title("Percentage of population living below the national poverty line for different States/UTs ",fontsize=20)  
ax.bar(xaxis,yaxis,color="blue",width=0.6)  
plt.show()
```



Percentage of the population (out of total eligible population) receiving social protection benefits under Pradhan Mantri Matru Vandana Yojana (PMMVY) for different States/UTs



Percentage of households living in katcha houses for different States/UTs

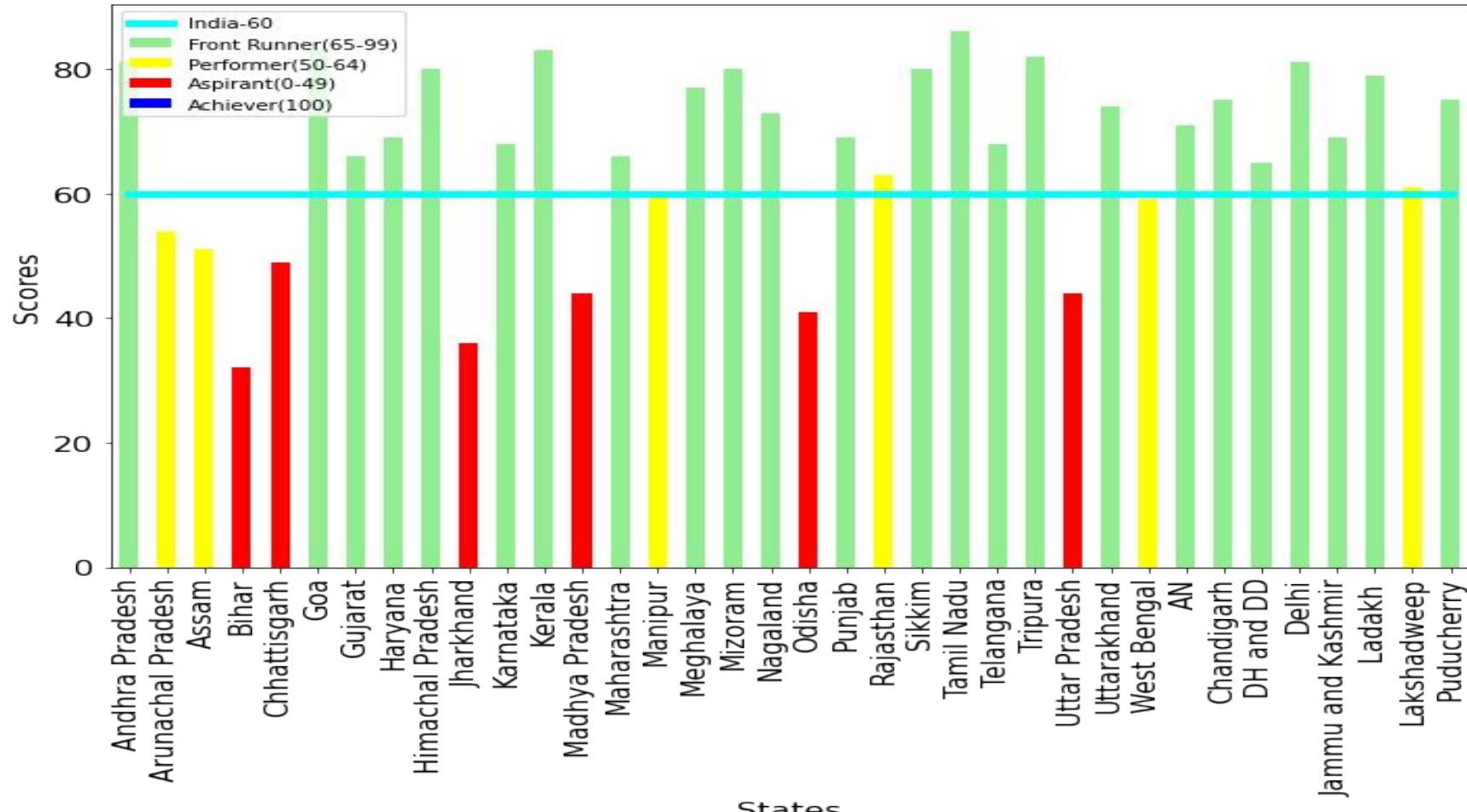


Code for plotting State Wise overall SDG 1 score and classification

```
# Created different array for States(SDG_states), all the scores (Final), score of India for each SDGs(India)
#In this code, intialising different arrays for each group from the scores(Final) array
#Finally using the for loop plotting all the graphs for each SDGs
for i in range(len(Final)):
    aspirant=[]
    for j in Final[i]:
        if j<50:
            aspirant.append(j)
        else:
            aspirant.append(0)
    Performer = []
    for j in Final[i]:
        if j>=50 and j<65:
            Performer.append(j)
        else:
            Performer.append(0)
    Front_Runner = []
    for j in Final[i]:
        if j>=65 and j<100:
            Front_Runner.append(j)
        else:
            Front_Runner.append(0)
    Achiever = []
    for j in Final[i]:
        if j==100:
            Achiever.append(j)
        else:
            Achiever.append(0)

plt.rcParams["figure.figsize"] = [10, 9]
plt.rcParams["figure.autolayout"] = True
sdg = pd.DataFrame({"Aspirant(0-49)":aspirant,"Performer(50-64)":Performer,"Front Runner(65-99)":Front_Runner,
                    "Achiever(100)":Achiever,("India-"+str(india[i])):India[i],index = SDG_states}
sdg[("India-"+str(india[i]))].plot(kind = 'line',color = 'cyan',legend = True, linewidth = 5)
sdg["Front Runner(65-99)"].plot(kind = 'bar',color = 'lightgreen', legend = True,fontsize = 16)
sdg["Performer(50-64)"].plot(kind = 'bar',color = 'yellow', legend = True,fontsize = 16)
sdg["Aspirant(0-49)"].plot(kind = 'bar',color = 'red', legend = True,fontsize = 16)
sdg["Achiever(100)"].plot(kind = 'bar',color = 'blue',legend = True,fontsize = 16)
plt. xlabel('States', fontsize=16)
plt. ylabel('Scores', fontsize=16)
plt.title(final_index[i],fontsize = 18)
plt.show()
```

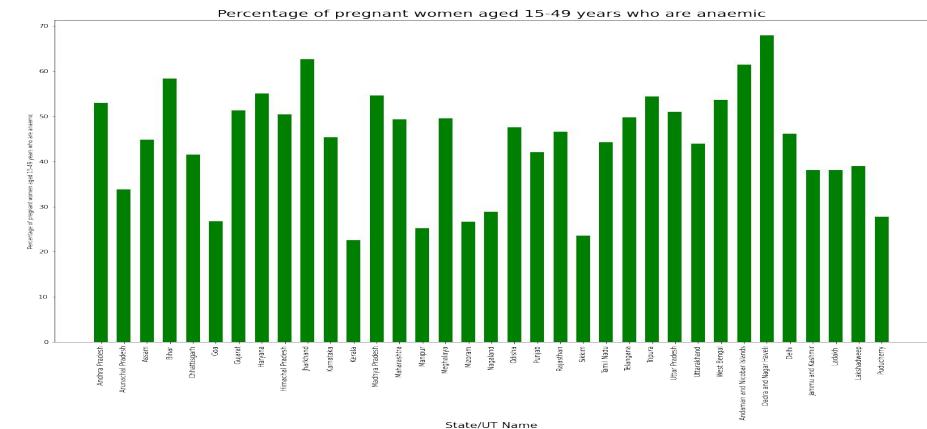
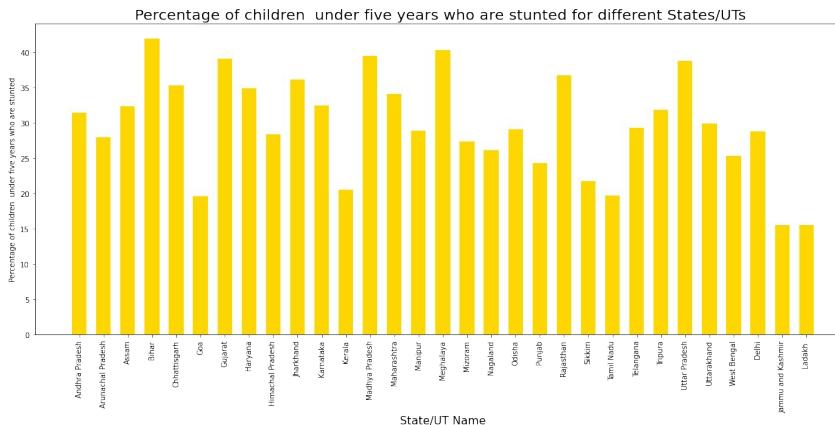
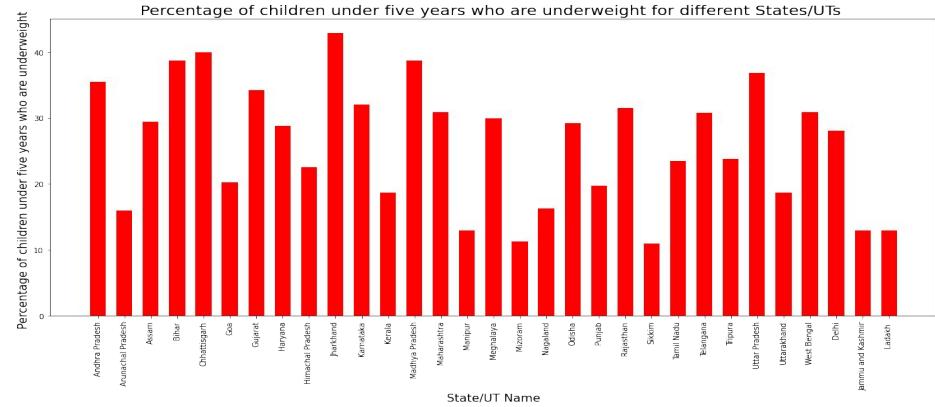
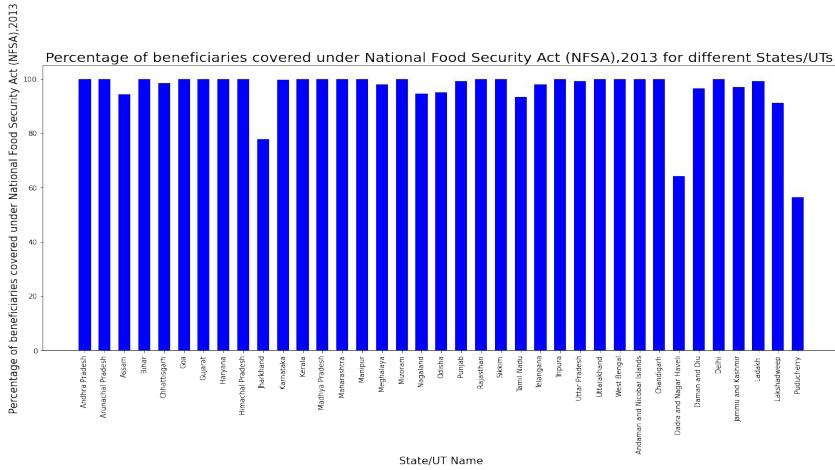
SDG 1



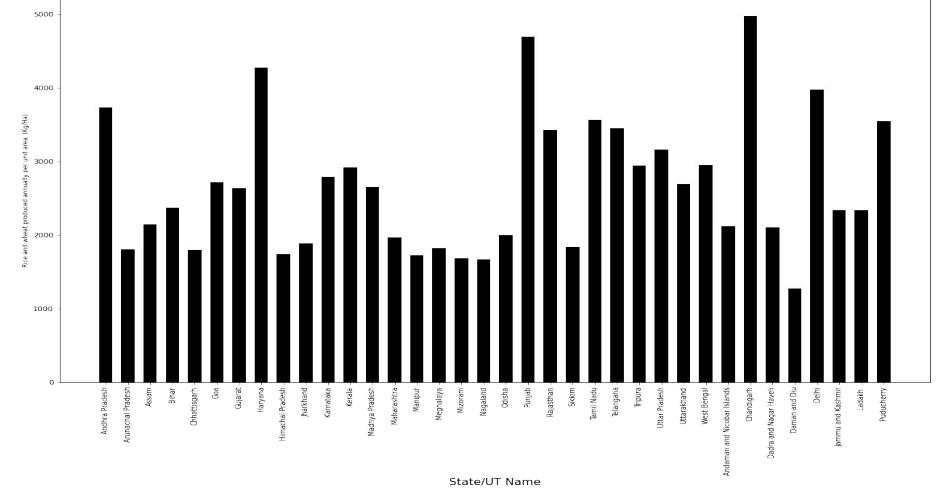
DATA STATISTICS FOR SDG 1

	Topic	Min_Value	Max_Value	Average_Val	No.of Null values
0	Percentage of population living below the national poverty line	1.000000	39.930000	17.762703	1
1	Head count ratio as per the Multidimensional Poverty Index	1.100000	52.500000	19.524324	0
2	Percentage of households with any usual member employed	2.900000	74.600000	27.375676	0
3	Persons provided employment as a percentage of total population	77.085593	99.924977	78.298874	4
4	Percentage of the population (out of total eligible) in employment	26.410000	98.150000	86.254054	2
5	Percentage of households living in katcha houses	0.000000	29.000000	3.383784	3

SDG 2 : Zero Hunger



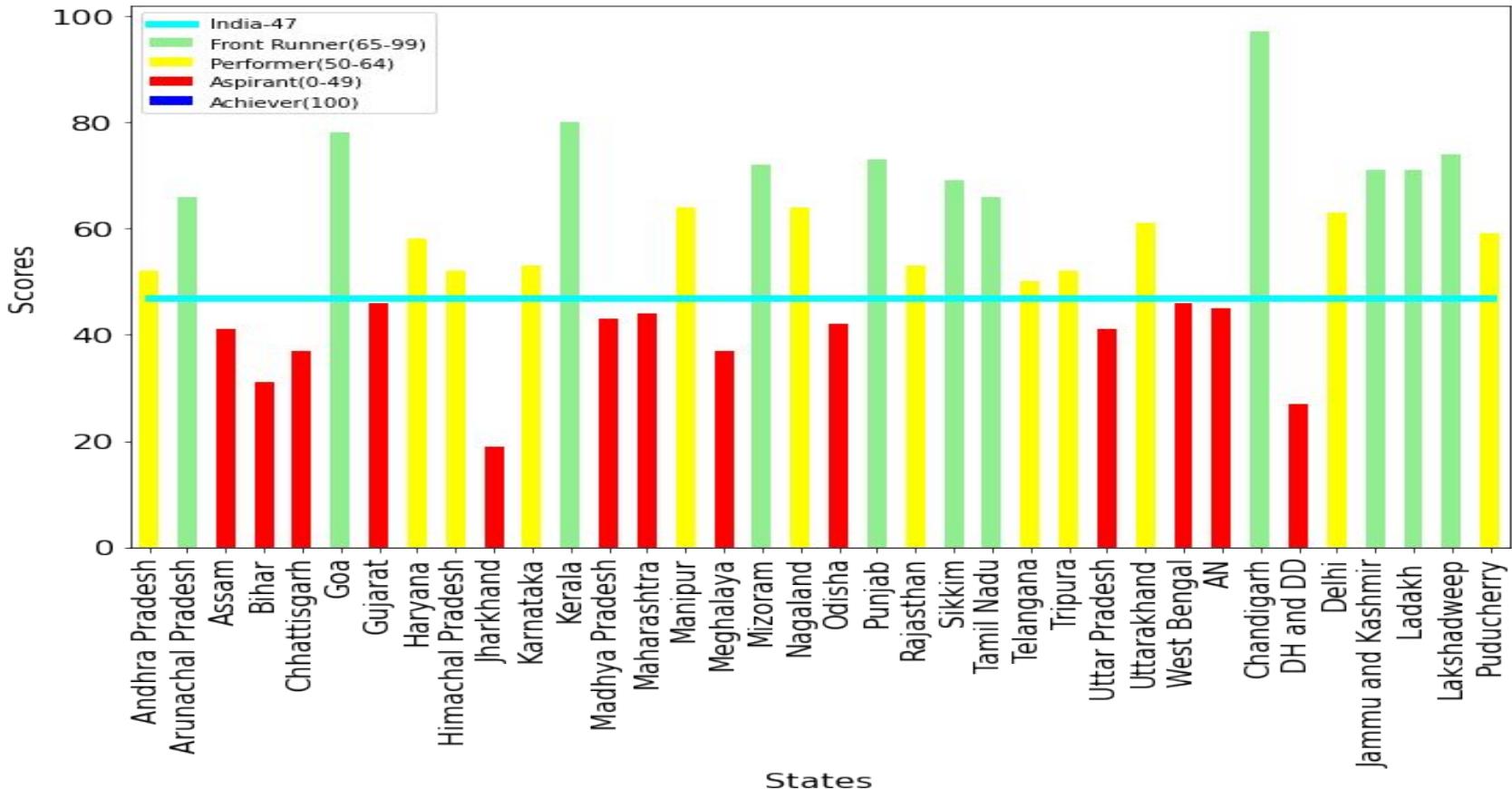
Rice and wheat produced annually per unit area (Kg/Ha) for different States/UTs



Gross Value Added (constant prices) in agriculture per worker (in Lakhs/worker)



SDG 2

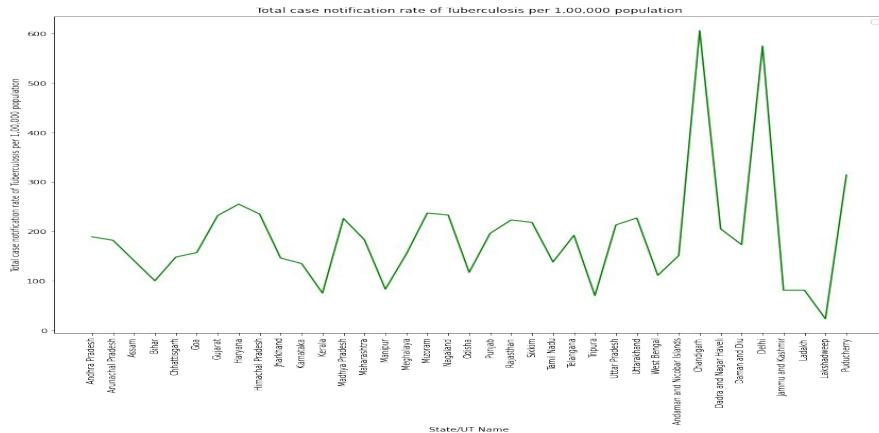
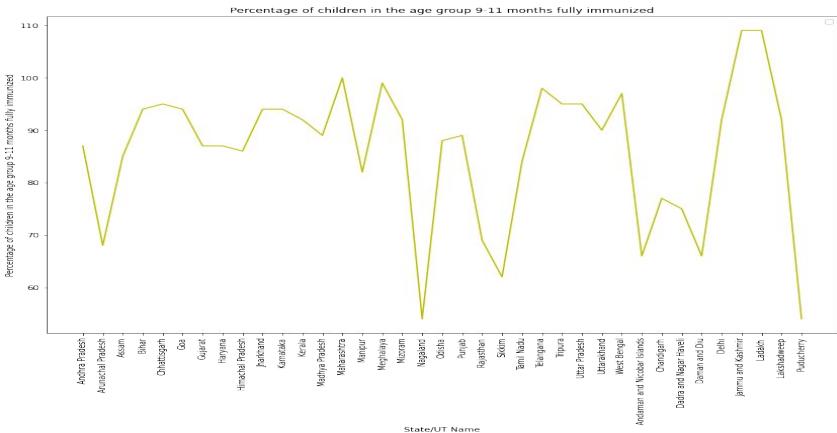
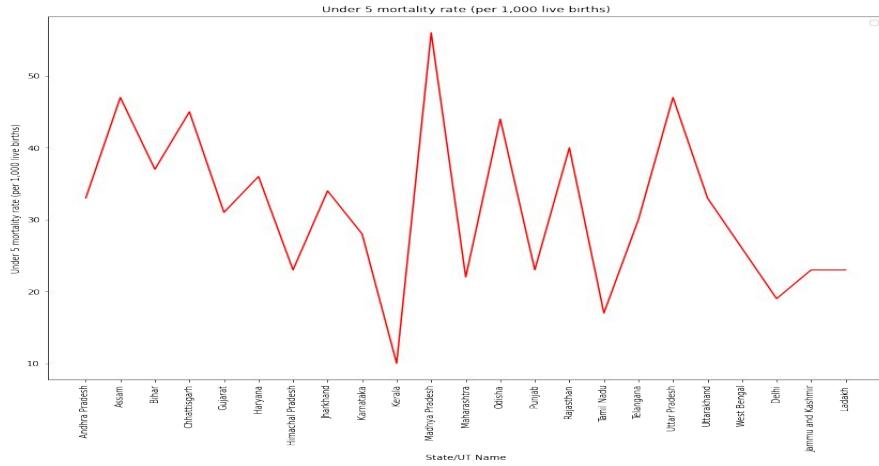
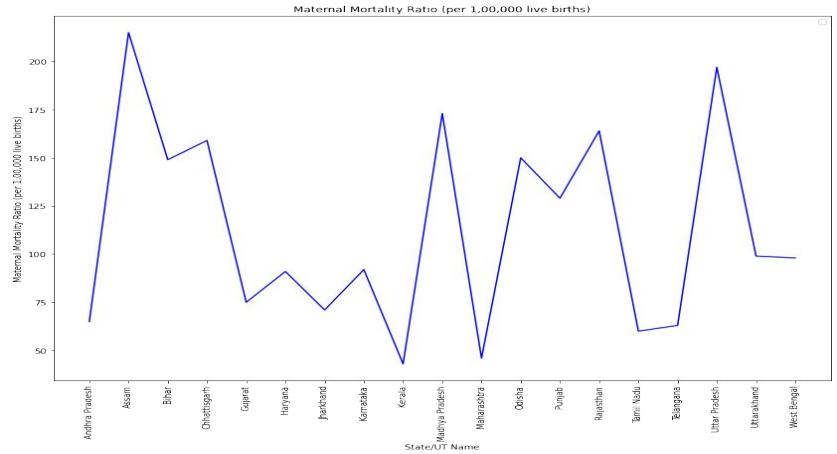


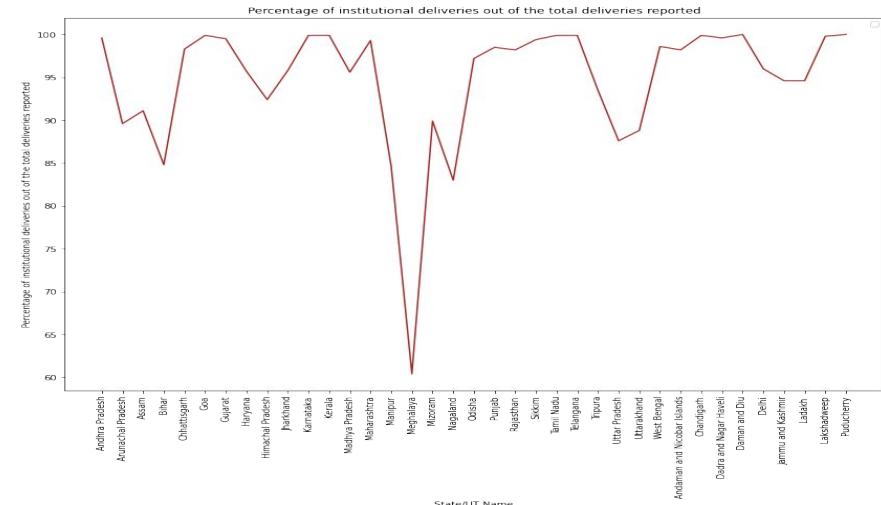
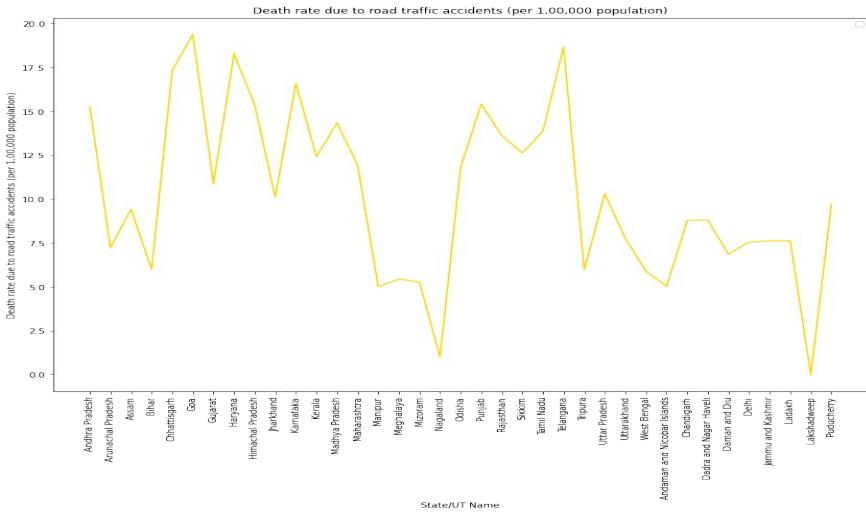
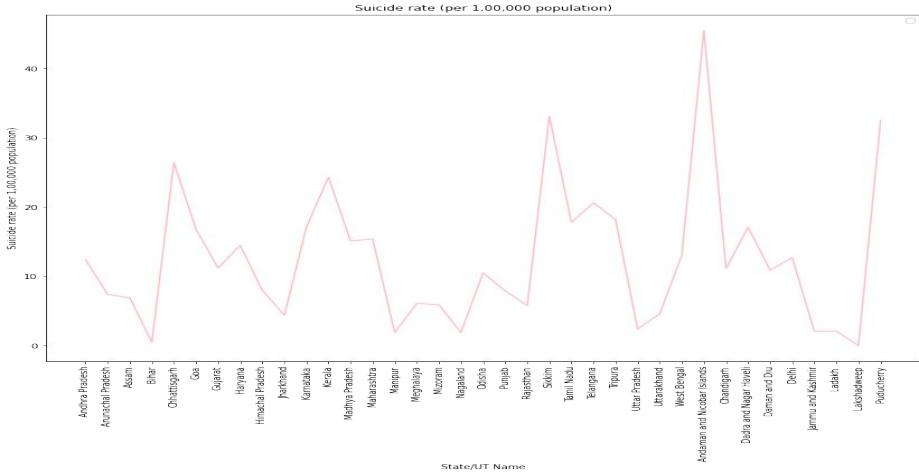
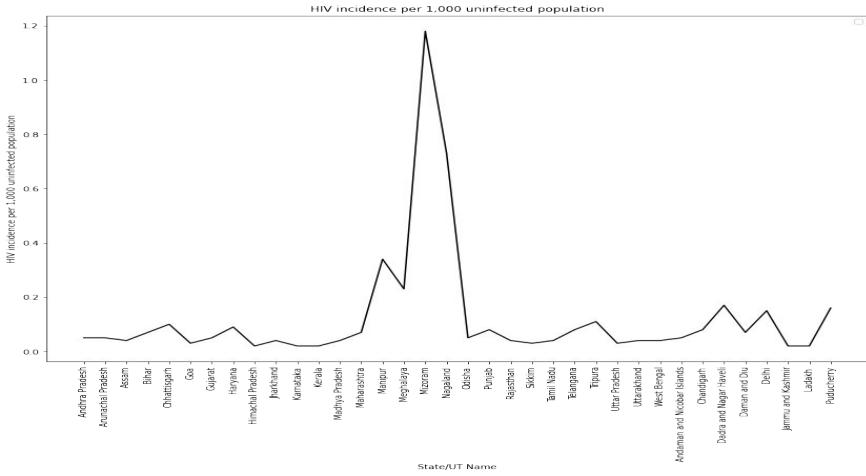
DATA STATISTICS FOR SDG 2

	Topic	Min_Value	Max_Value	Average_Val	No.of Null values
0	Percentage of beneficiaries covered under National Social Assistance Program	56.34000	100.00000	95.990811	0
1	Percentage of children under five years who are stunted	11.00000	42.90000	21.872973	6
2	Percentage of children under five years who are wasted	15.50000	42.00000	24.964865	6
3	Percentage of pregnant women aged 15-49 years who received antenatal care services	22.60000	67.90000	42.016216	2
4	Percentage of adolescents aged 10–19 years who are anemic	8.40000	45.50000	20.716216	6
5	Rice and wheat produced annually per unit area	1272.73000	4973.62000	2587.709189	1
6	Gross Value Added (constant prices) in agriculture, forestry, and fisheries	0.28555	4.20806	0.987473	5

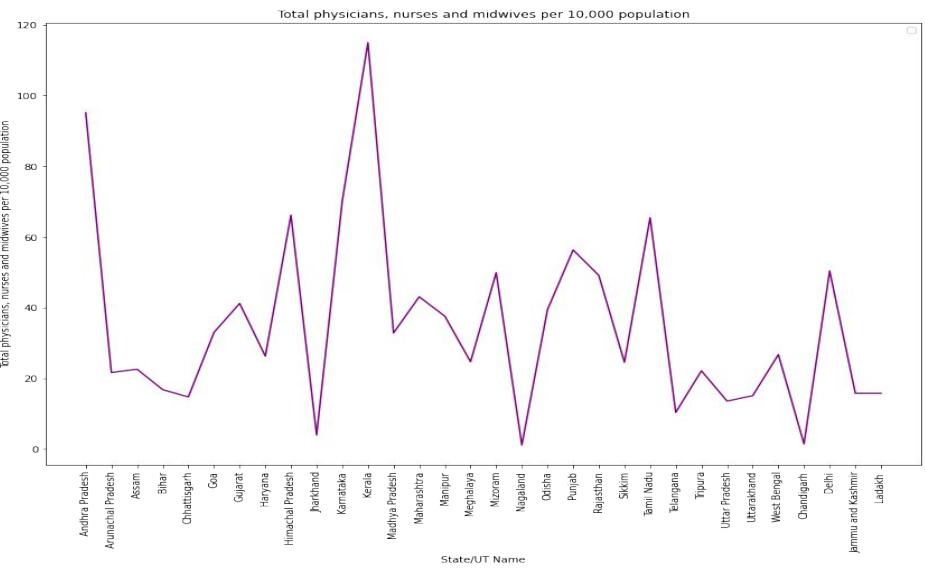
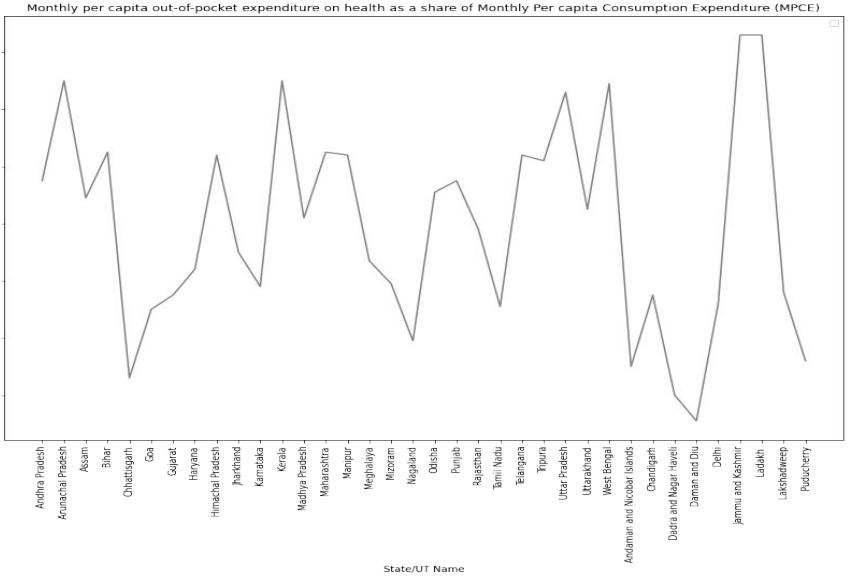
SDG 3 : Good Health and Well-being

```
#code for plotting line graph for SDGs indicators:  
#this code is for SDG 3 first indicator  
x = df3['States/UTs']  
y = df3['Maternal Mortality Ratio (per 1,00,000 live births)']  
plt.rcParams["figure.figsize"] = [14.00, 10]  
  
plt.ylabel("Maternal Mortality Ratio (per 1,00,000 live births)")  
plt.xlabel("State/UT Name")  
plt.xticks(rotation=90)  
  
plt.title("Maternal Mortality Ratio (per 1,00,000 live births)")  
# Plot a simple line chart  
plt.plot(x, y, 'b')  
  
plt.legend()  
plt.show()
```

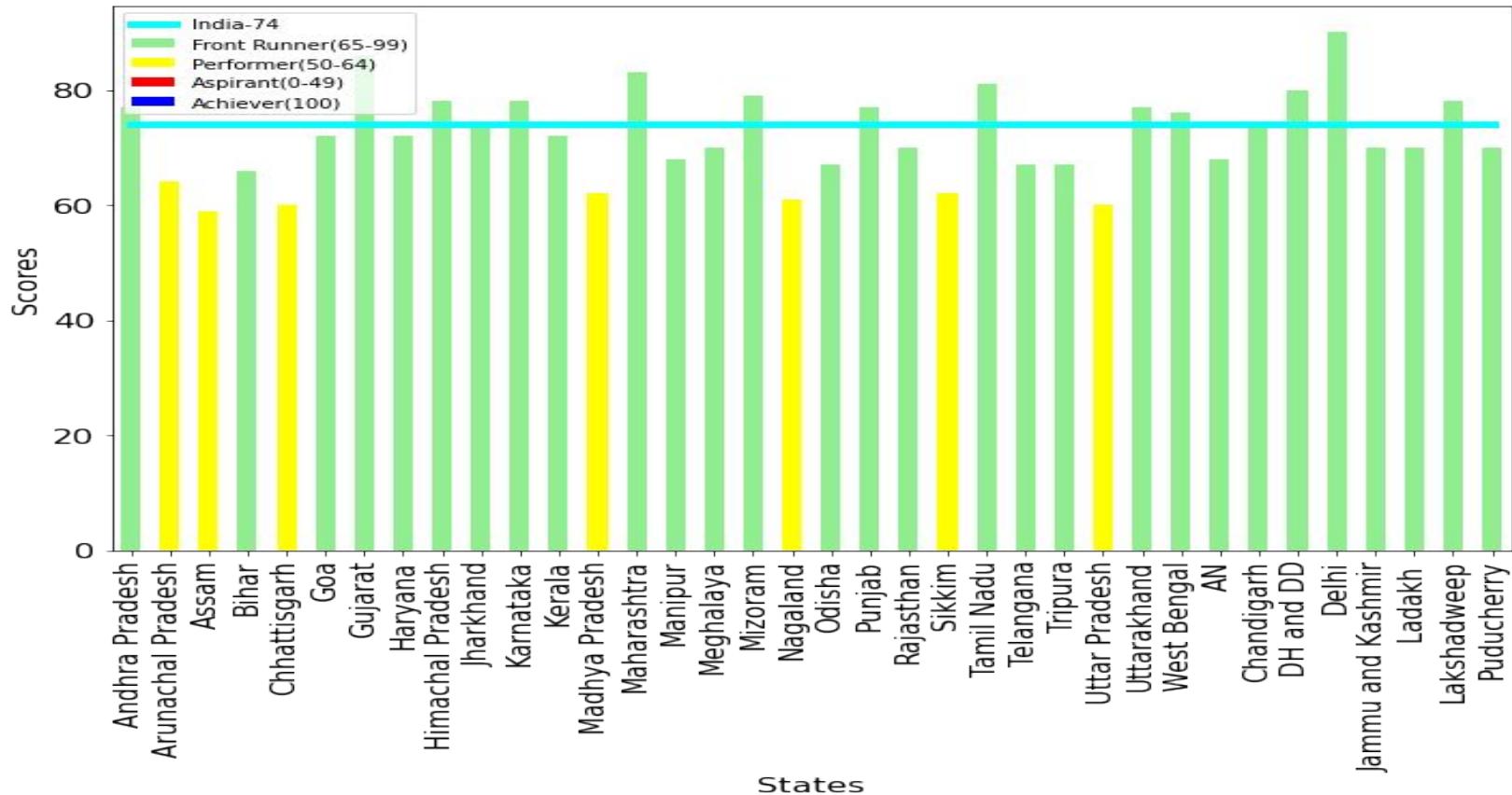




Monthly per capita out-of-pocket expenditure on health as a share of Monthly Per capita Consumption Expenditure (MPCE)



SDG 3

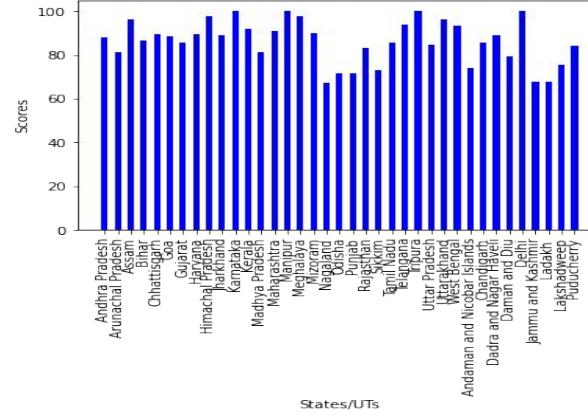


DATA STATISTICS FOR SDG 3

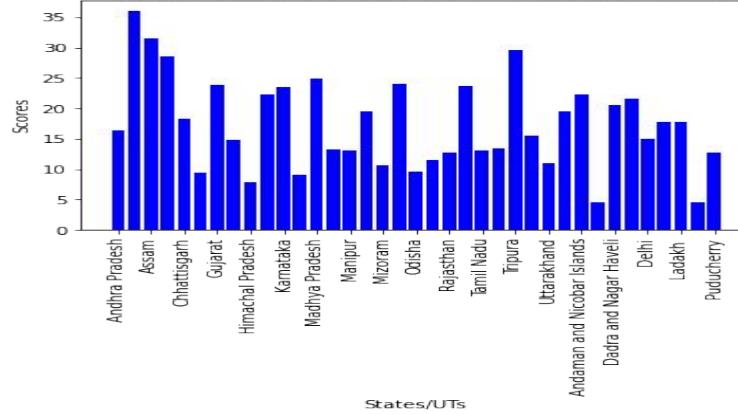
	Topic	Min_Value	Max_Value	Average_Val	No.of Null values
0	Maternal Mortality Ratio (per 1,00,000 live births)	43.00000	215.000000	57.810811	18
1	Under 5 mortality rate (per 1,000 live births)	10.00000	56.000000	19.648649	14
2	Percentage of children in the age group 9-11 months who are stunted	54.00000	109.000000	86.108108	0
3	Total case notification rate of Tuberculosis per 100,000 population	23.00000	606.000000	189.864865	0
4	HIV incidence per 1,000 uninfected population	0.02000	1.180000	0.119730	1
5	Suicide rate (per 1,00,000 population)	0.00000	45.500000	12.543243	1
6	Death rate due to road traffic accidents (per 100,000 population)	0.00000	19.377835	10.250897	1
7	Percentage of institutional deliveries out of total deliveries	60.40000	100.000000	94.691892	0
8	Monthly per capita out-of-pocket expenditure on health	5.10000	18.600000	11.570270	1
9	Total physicians, nurses and midwives per 10,000 population	1.10743	115.000718	30.320630	5

SDG 4 : Quality Education

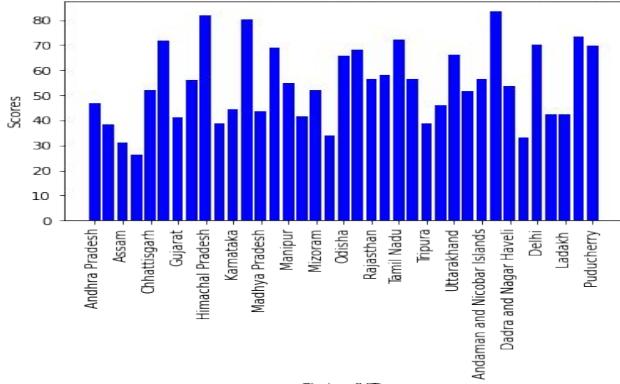
Adjusted Net Enrolment Ratio (ANER) in elementary education (class 1-8) v/s States/UTs



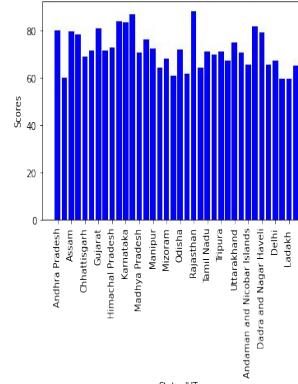
Average annual dropout rate at secondary level (class 9-10)v/s States/UTs



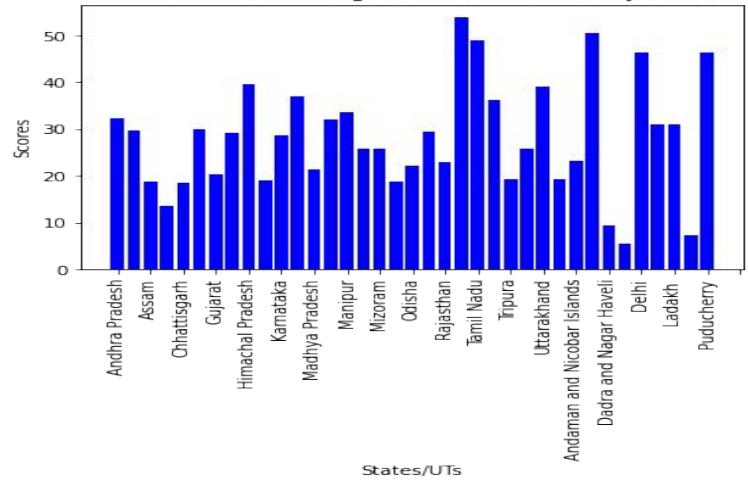
Gross Enrolment Ratio (GER) in higher secondary (class 11-12) v/s States/UTs



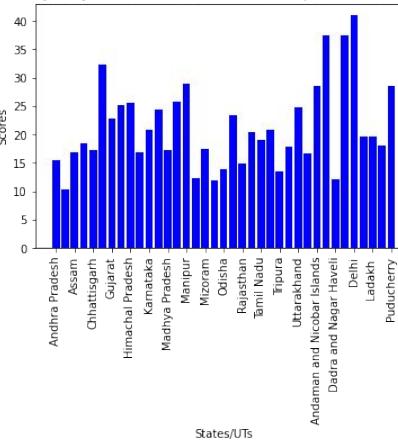
Percentage of students in grade VIII achieving atleast a minimum proficiency level in terms of nationally defined learning outcomes to be attained by the pupils at the end of the grade v/s States/UTs



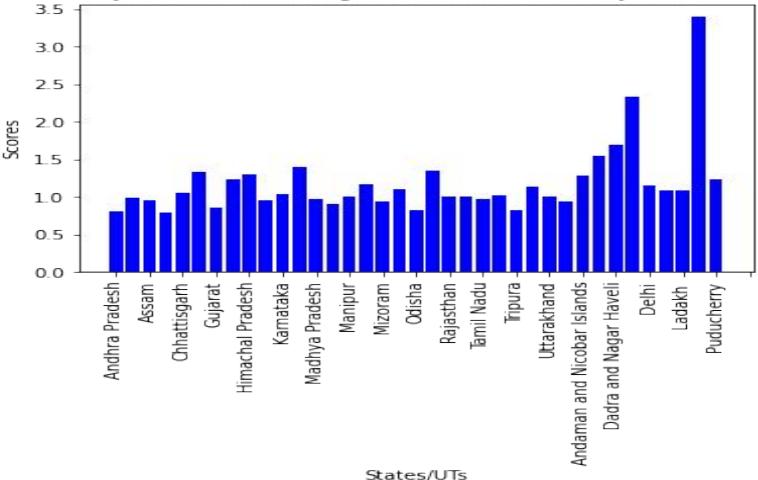
Gross Enrolment Ratio (GER) in higher education (18-23 years) v/s States/UTs



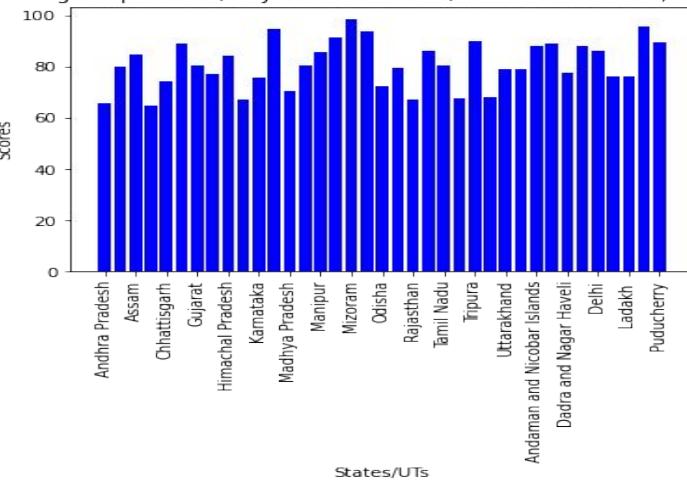
Percentage of persons with disability (15 years and above) who have completed at least secondary education v/s States/UTs

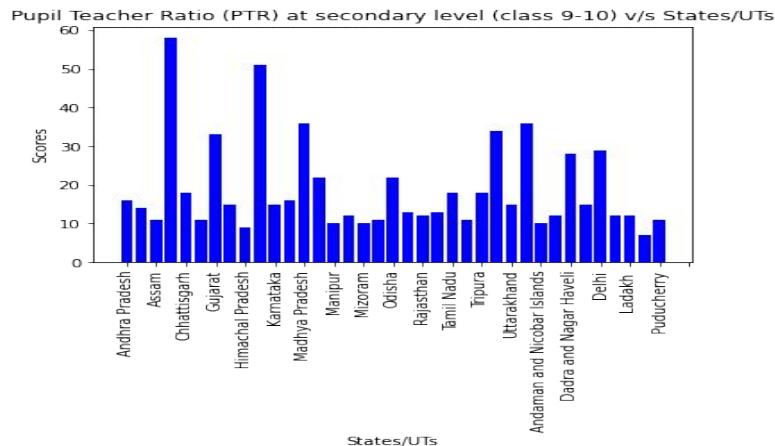
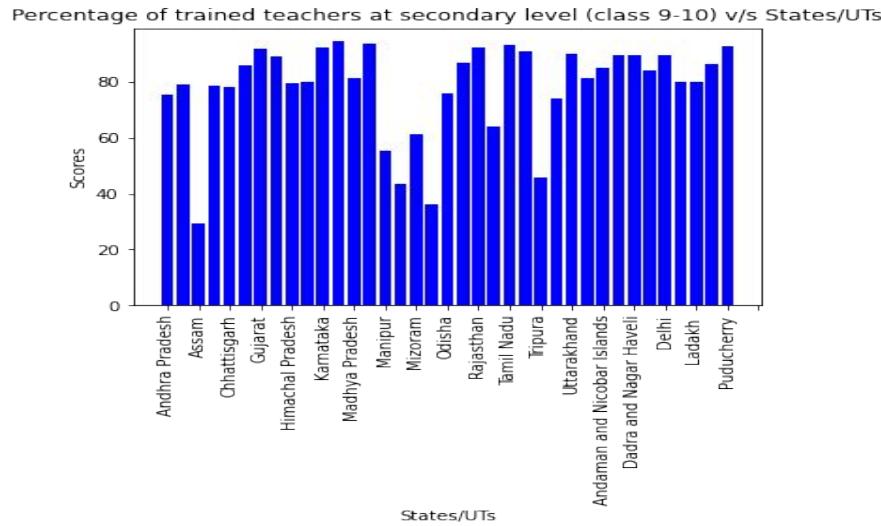
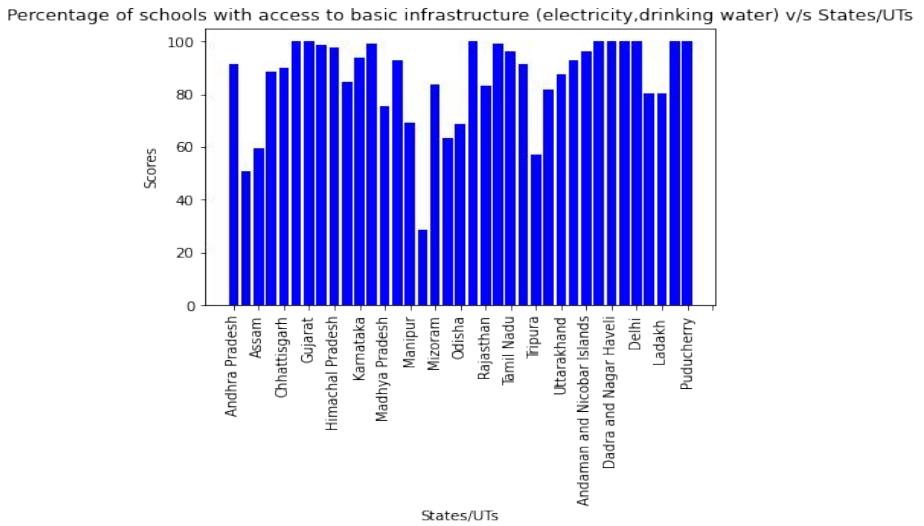


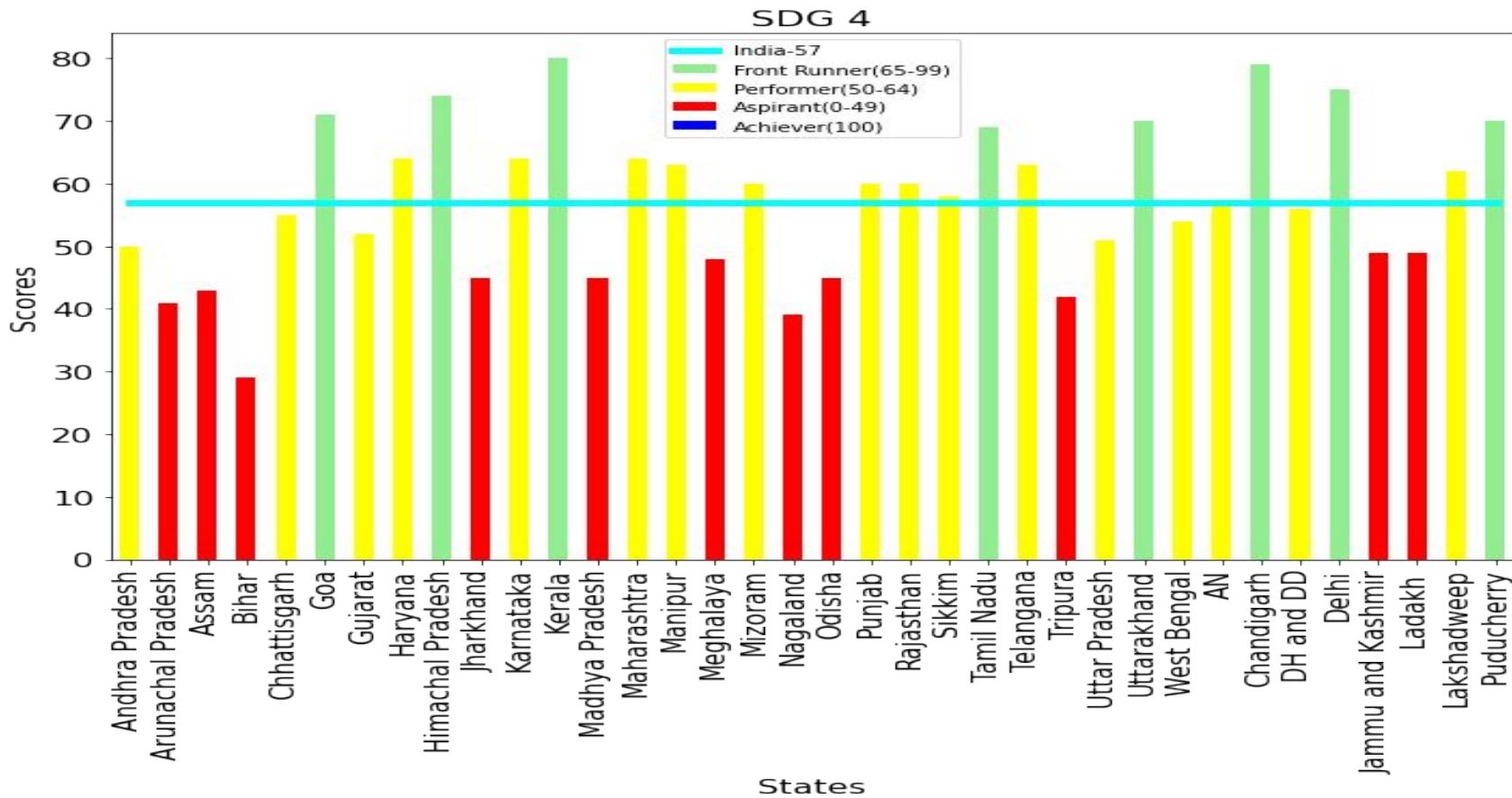
Gender Parity Index (GPI) for higher education (18-23 years) v/s States/UTs



Percentage of persons (15 years and above) who are literate v/s States/UTs





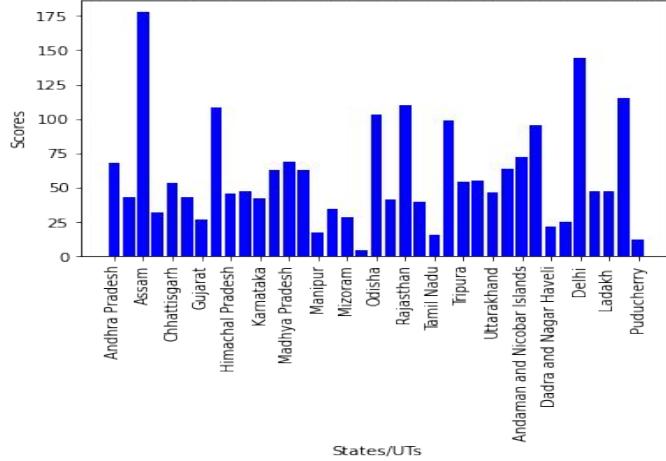


DATA STATISTICS FOR SDG 4

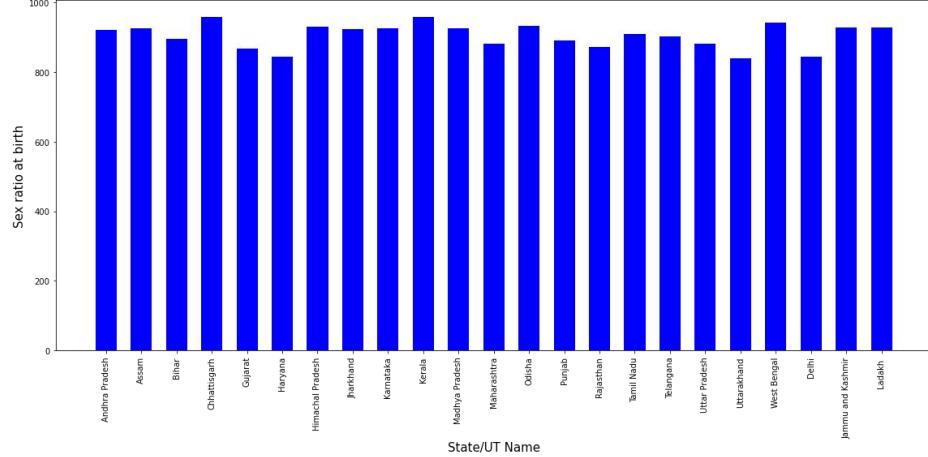
	Topic	Min_Value	Max_Value	Average_Val	No.of Null values
0	Adjusted Net Enrolment Ratio (ANER) in elementary education	67.380000	100.000000	86.097838	0
1	Average annual dropout rate at secondary level	4.520000	35.980000	17.371622	0
2	Gross Enrolment Ratio (GER) in higher secondary education	26.390000	83.430000	54.294595	0
3	Percentage of students in grade VIII achieving proficiency in English	59.750000	88.100000	71.547297	0
4	Gross Enrolment Ratio (GER) in higher education	5.500000	53.900000	28.167568	0
5	Percentage of persons with disability (15 years and above)	10.300000	41.000000	21.270270	0
6	Gender Parity Index (GPI) for higher education	0.790000	3.400000	1.181892	0
7	Percentage of persons (15 years and above) who have completed primary education	64.700000	98.500000	80.951351	0
8	Percentage of schools with access to basic infrastructure	28.389802	100.000000	85.965629	0
9	Percentage of trained teachers at secondary level	29.294607	94.526443	78.218547	0
10	Pupil Teacher Ratio (PTR) at secondary level (2018-19)	7.000000	58.000000	18.810811	0

SDG 5 : Gender Equality

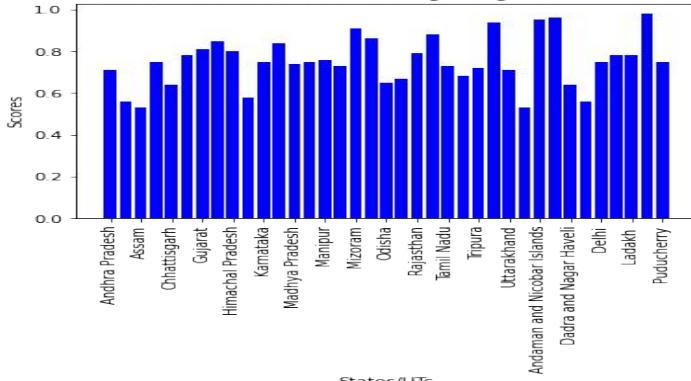
Rate of crimes against women per 1.00,000 female population v/s States/UTs



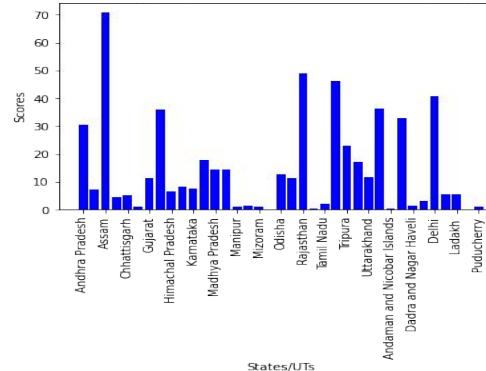
Sex ratio at birth for different States/UTs



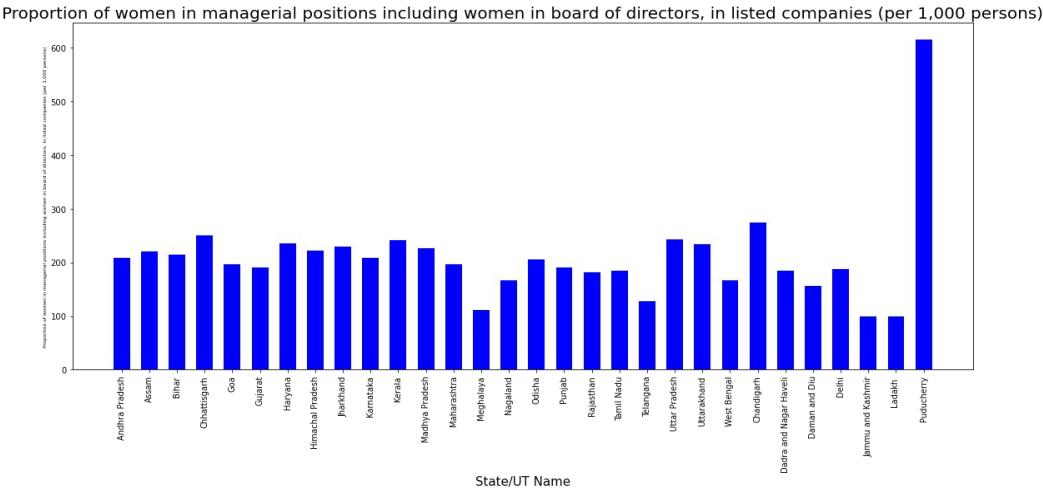
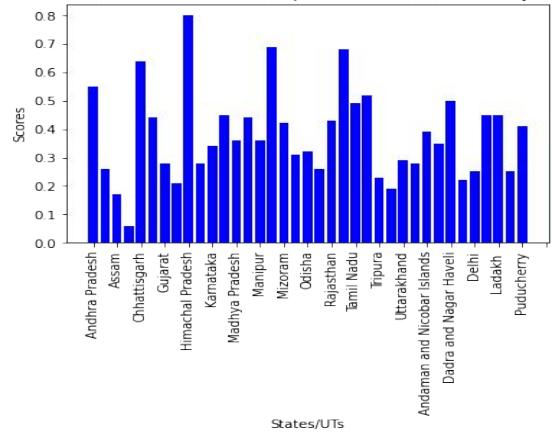
Ratio of female to male average wage v/s States/UTs



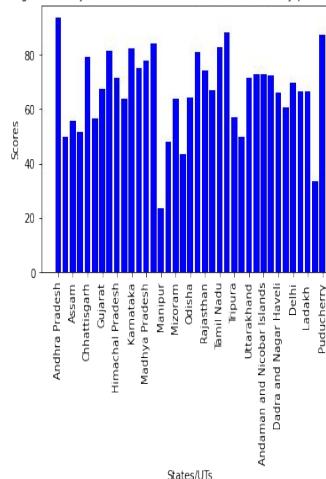
Per lakh women who have experienced cruelty/physical violence by husband or his relatives during the year v/s States/UTs



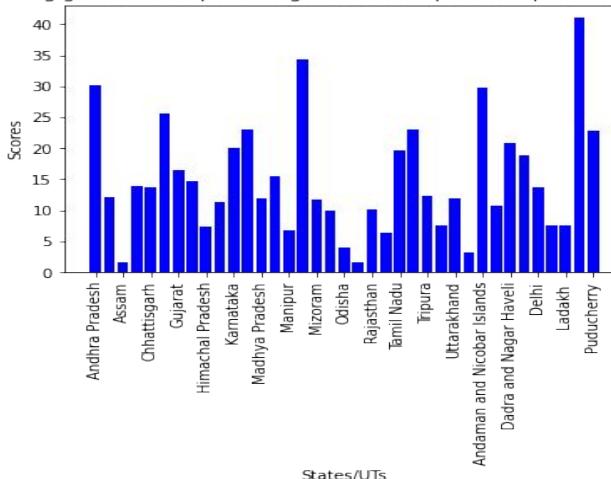
Ratio of female to male Labour Force Participation Rate (LFPR) (15-59 years) v/s States/UTs



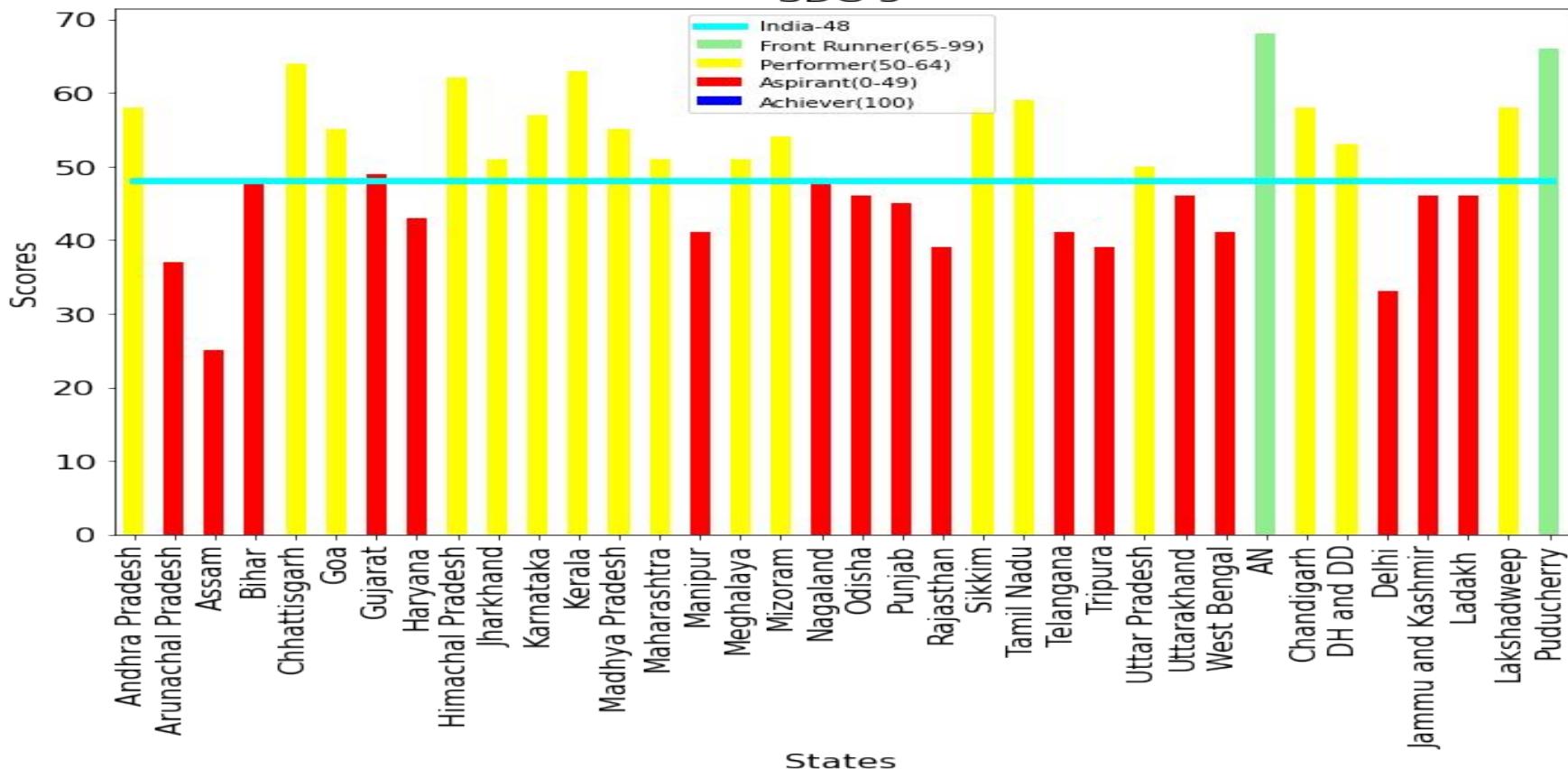
Percentage of currently married women aged 15-49 years who have their demand for family planning satisfied by modern methods v/s States/UTs



Operational land holding gender wise (percentage of female operated operational holdings) v/s States/UTs



SDG 5

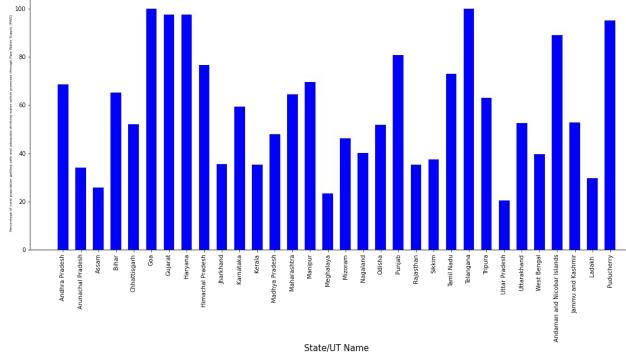


DATA STATISTICS FOR SDG 5

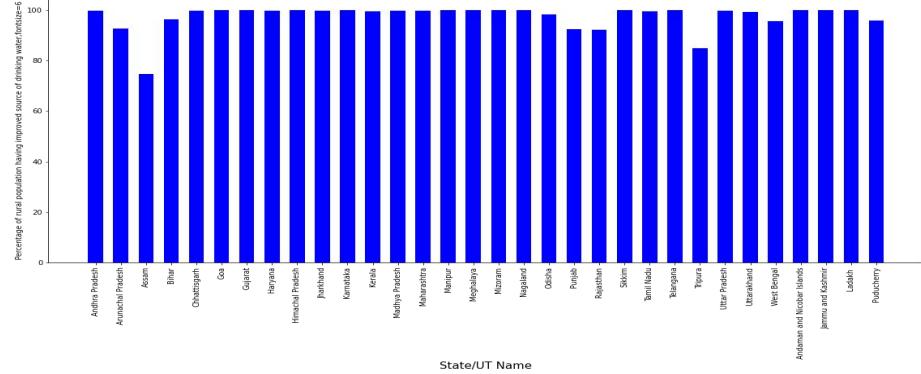
	Topic	Min_Value	Max_Value	Average_Val	No.of Null values
0	Rate of crimes against women per 1,00,000 females	4.100000	177.800000	58.924324	0
1	Sex ratio at birth	562.378378	958.000000	562.378378	14
2	Ratio of female to male average wage/salary etc.	0.530000	0.980000	0.751351	0
3	Per lakh women who have experienced cruelty/physical violence	0.000000	70.728242	14.567448	2
4	Percentage of elected women over total seats in the Lok Sabha	0.000000	14.444444	6.440270	7
5	Ratio of female to male Labour Force Participation Rate	0.060000	0.800000	0.378919	0
6	Proportion of women in managerial positions in the private sector	100.000000	615.000000	169.405405	7
7	Percentage of currently married women aged 15-49 years	23.600000	93.600000	66.851351	0
8	Operational land holding gender wise (percentage)	1.548623	41.030249	14.948995	0

SDG 6 : Clean Water and Sanitation

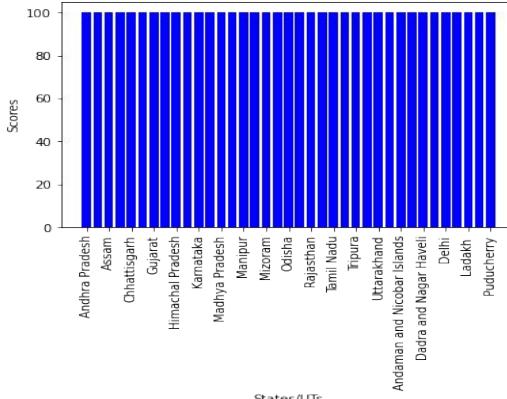
Percentage of rural population getting safe and adequate drinking water within premises through Pipe Water Supply (PWS) for different States/UTs



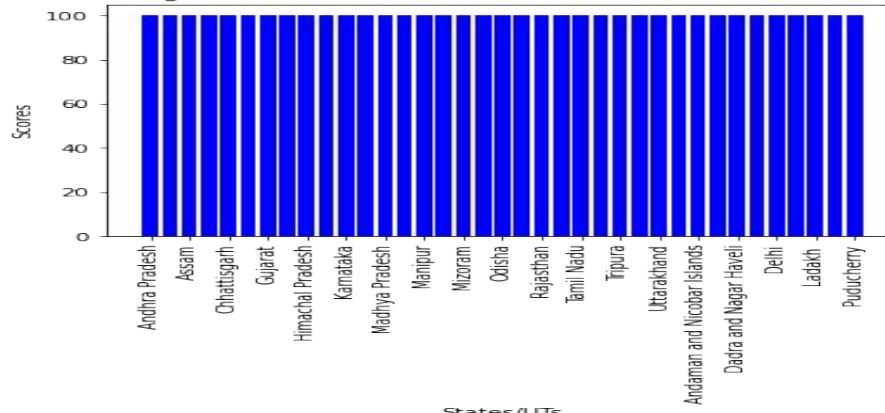
Percentage of rural population having improved source of drinking water

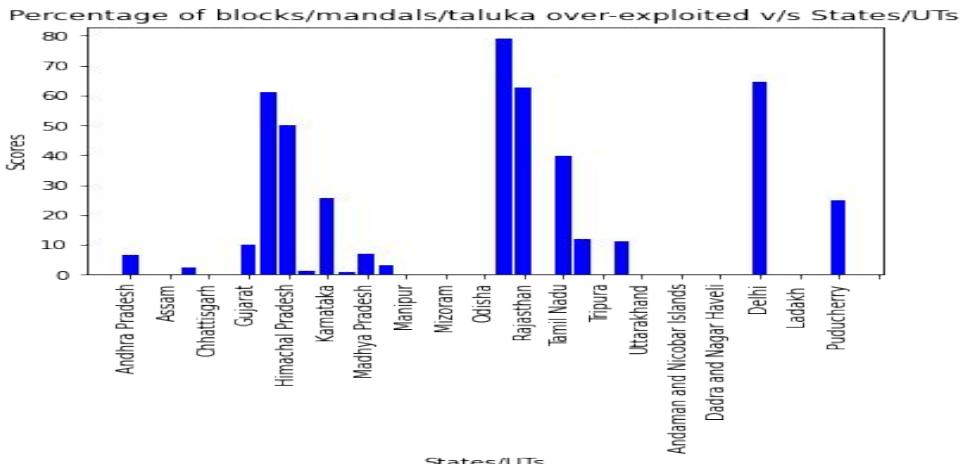
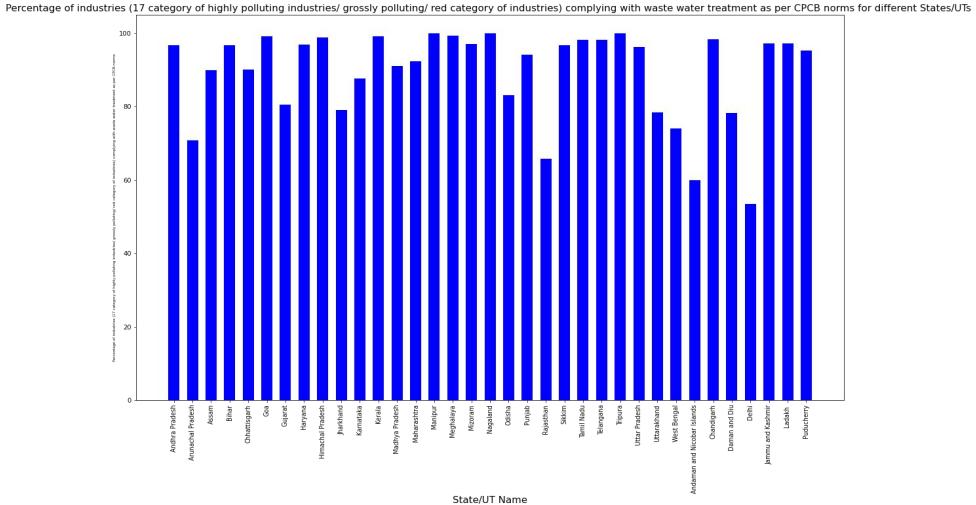
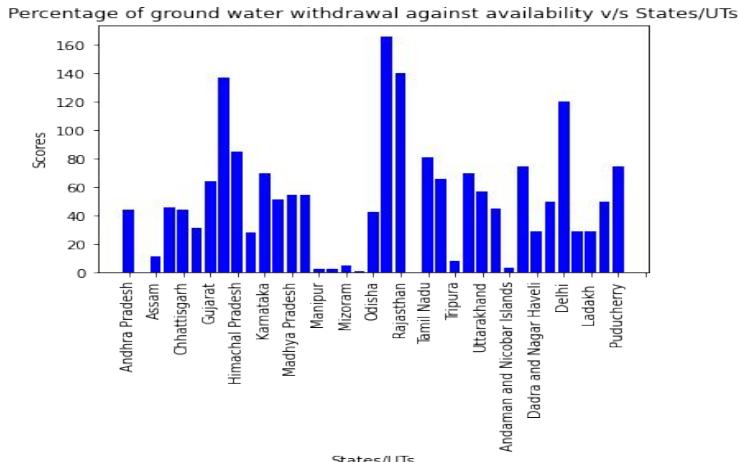
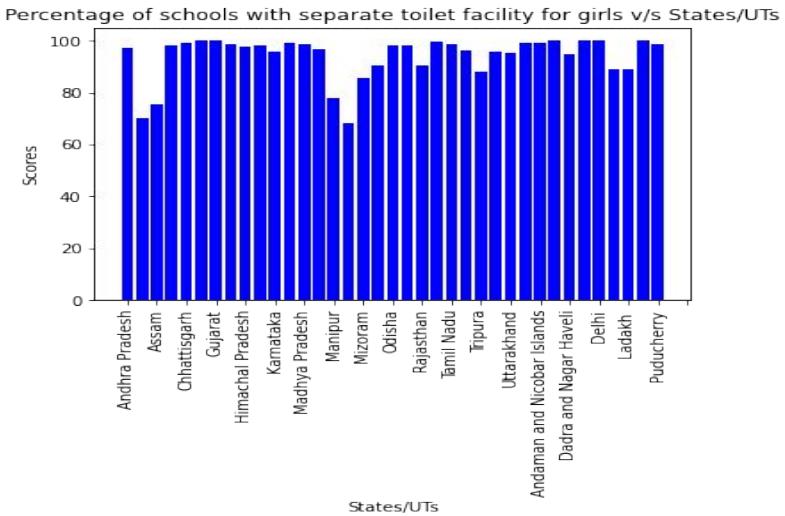


Percentage of individual household toilets constructed against target (SBM(G)) v/s States/UTs

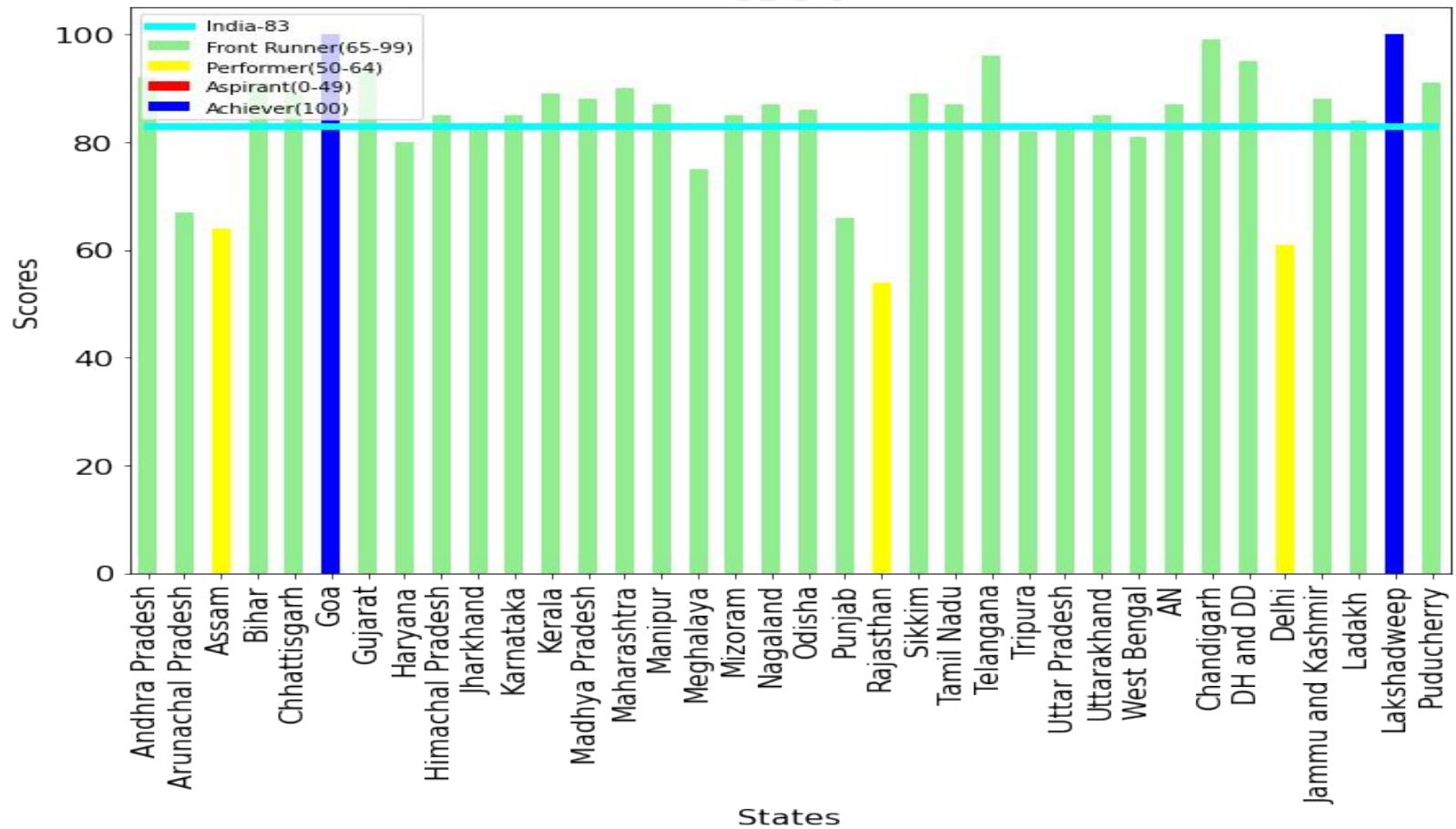


Percentage of districts verified to be ODF (SBM(G)) v/s States/UTs





SDG 6



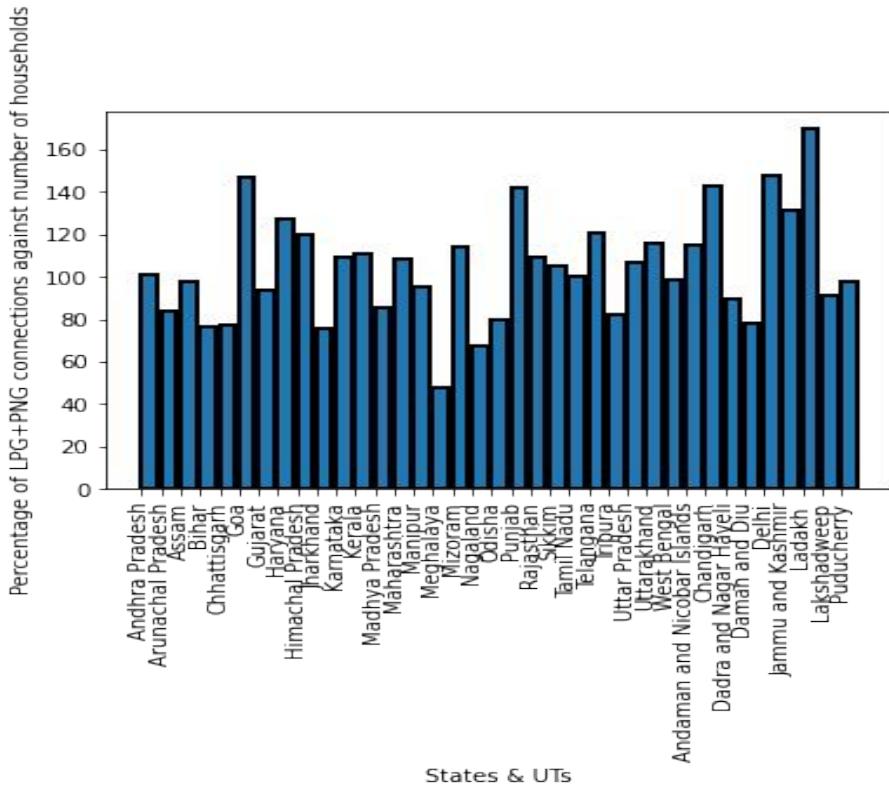
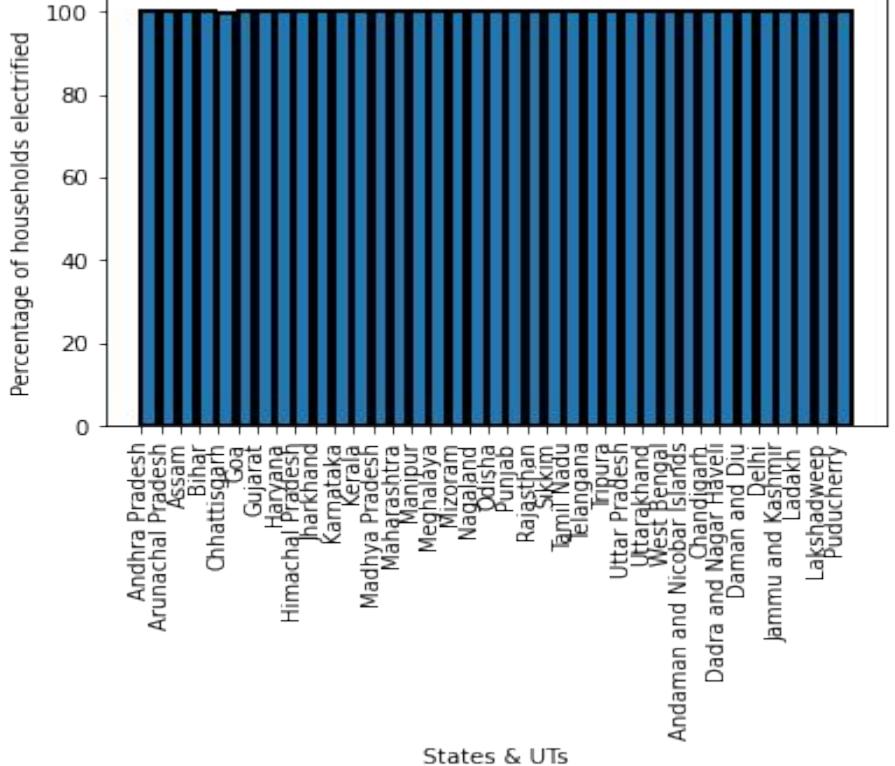
DATA STATISTICS FOR SDG 6

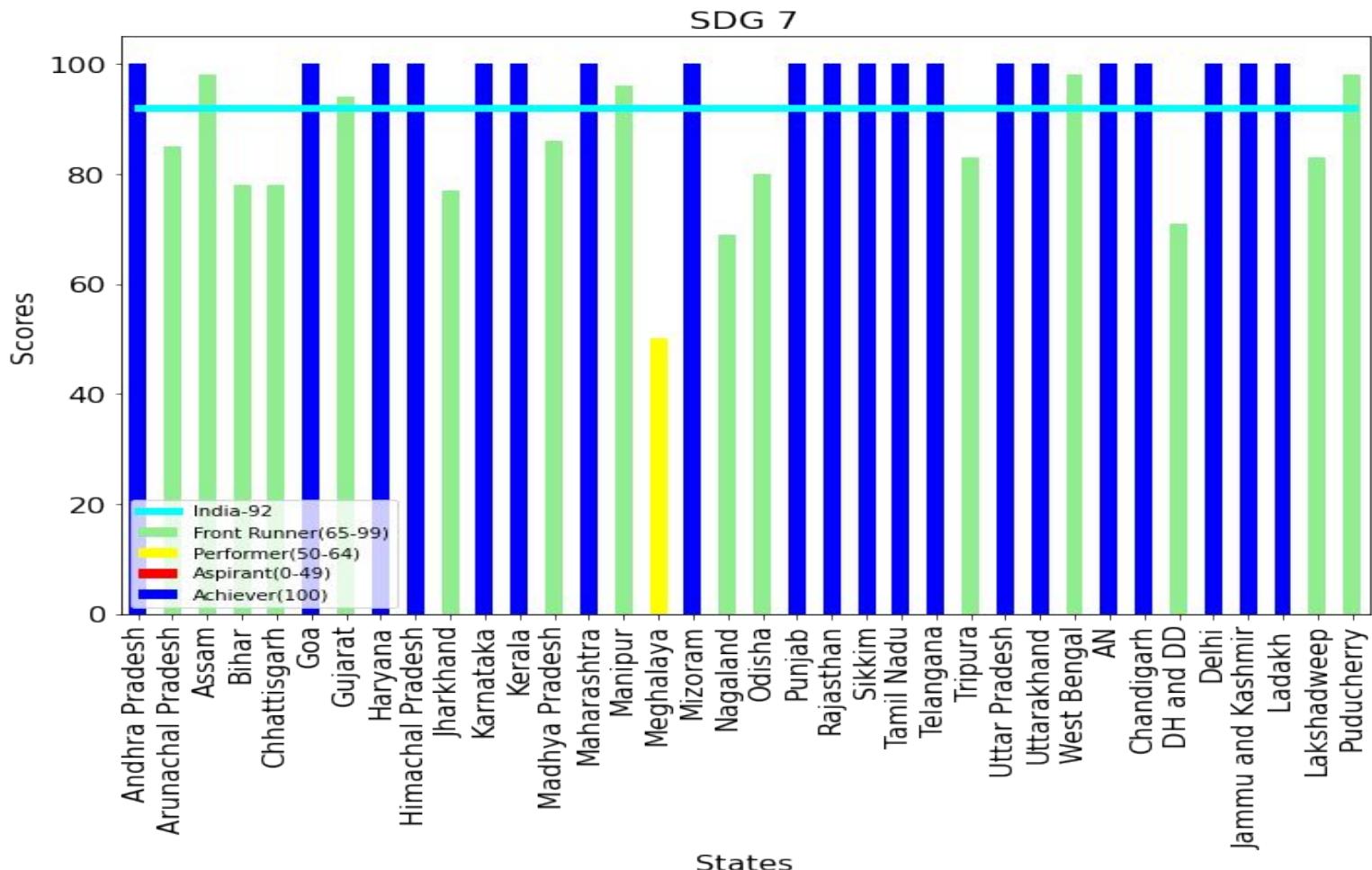
	Topic	Min_Value	Max_Value	Average_Val	No.of Null values
0	Percentage of rural population getting safe an...	20.348594	100.000000	50.261359	5
1	Percentage of rural population having improved...	74.724722	100.000000	84.280529	5
2	Percentage of individual household toilets con...	100.000000	100.000000	100.000000	0
3	Percentage of districts verified to be ODF (SB...	100.000000	100.000000	100.000000	0
4	Percentage of schools with separate toilet fac...	67.977068	100.000000	93.931012	0
5	Percentage of industries (17 category of highl...	53.440367	100.000000	84.633842	2
6	Percentage of ground water withdrawal against ...	0.060000	165.801668	50.441442	0
7	Percentage of blocks/mandals/taluka over-explo...	0.000000	78.985507	12.472504	20



SDG 7 : Affordable and Clean Energy

```
#code for histogram for the SDGs
#we have used similar codes for all SDGs
x=data['States_UTs']
y=data['Percentage_of_population_in_the_lowest_two_wealth_quintiles']
plt.bar(x,y,align='edge',edgecolor='black',linewidth=2")
plt.xticks(rotation=90)
plt.xlabel("States & UTs")
plt.ylabel("Percentage_of_population_in_the_lowest_two_wealth_quintiles")
```

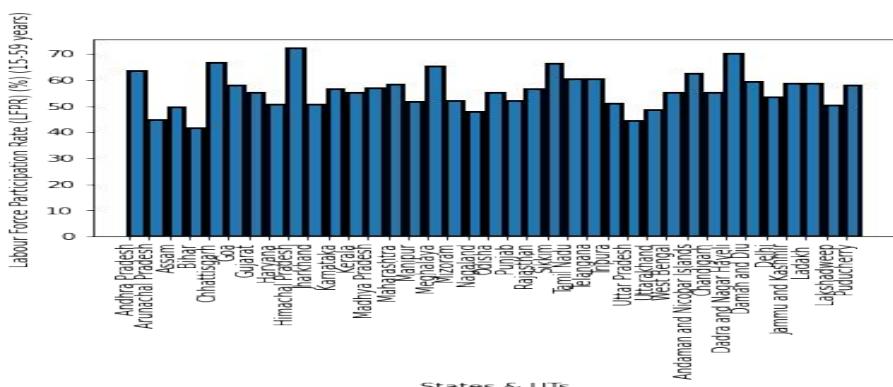
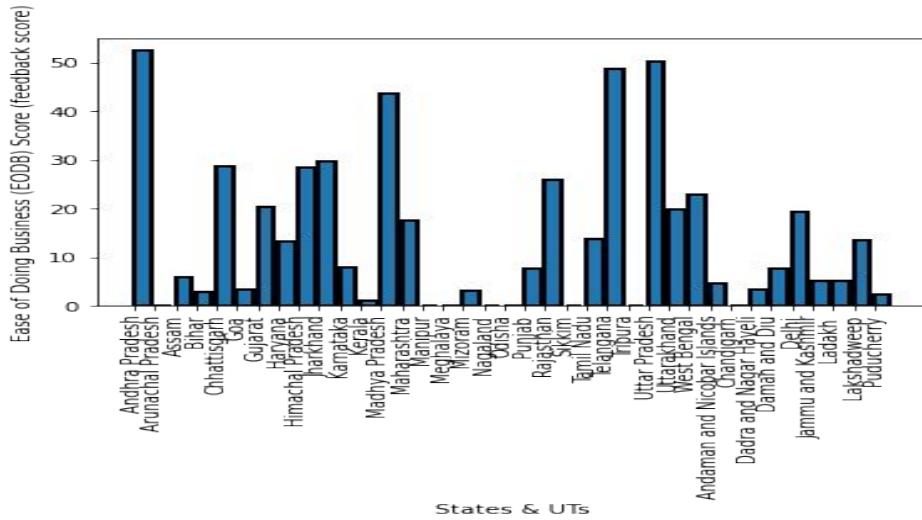
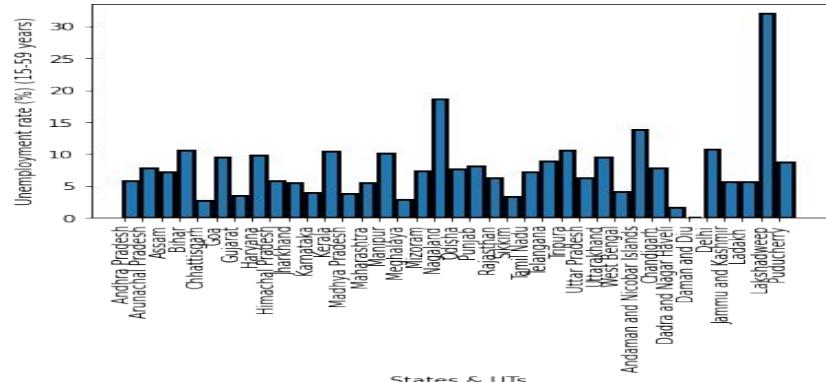
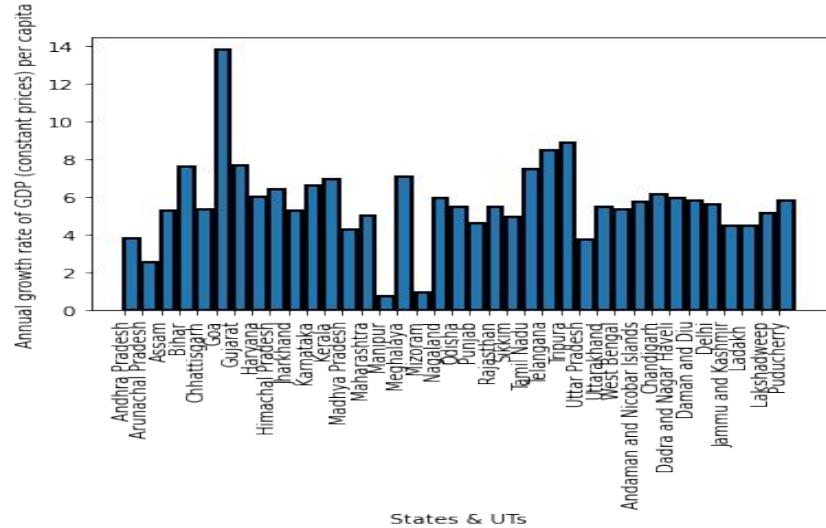




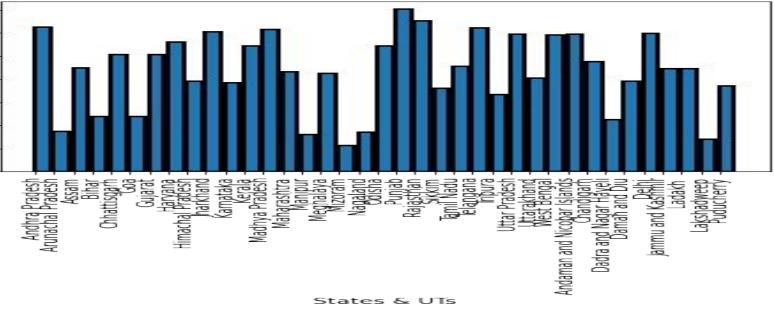
DATA STATISTICS FOR SDG 7

	Topic	Min_Value	Max_Value	Average_Val	No.of Null values
0	Percentage of households electrified	86.477568	100.000000	86.477568	5
1	Percentage of LPG+PNG connections against numb...	47.500000	169.811321	104.300824	0

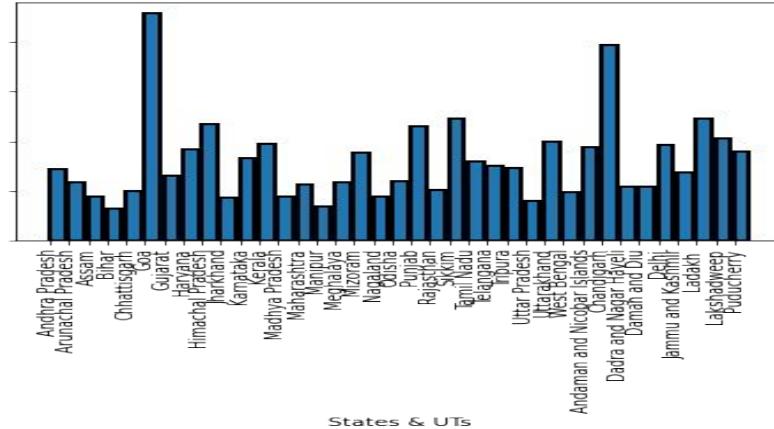
SDG 8 : Decent Work and Economic Growth



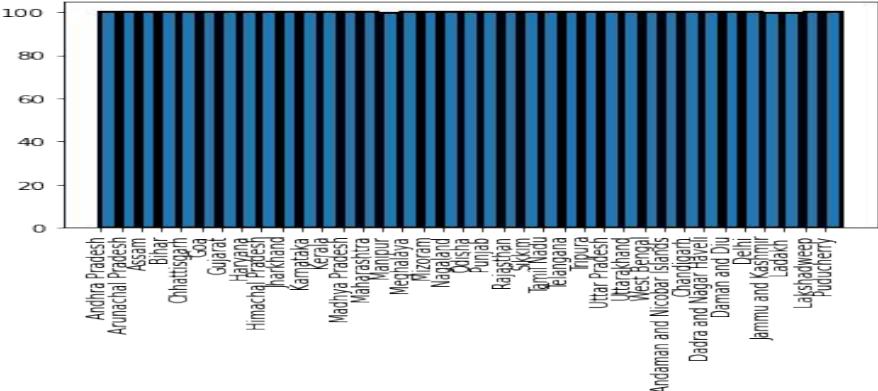
Percentage of regular wage salaried employees in non-agriculture sector without any social security benefit



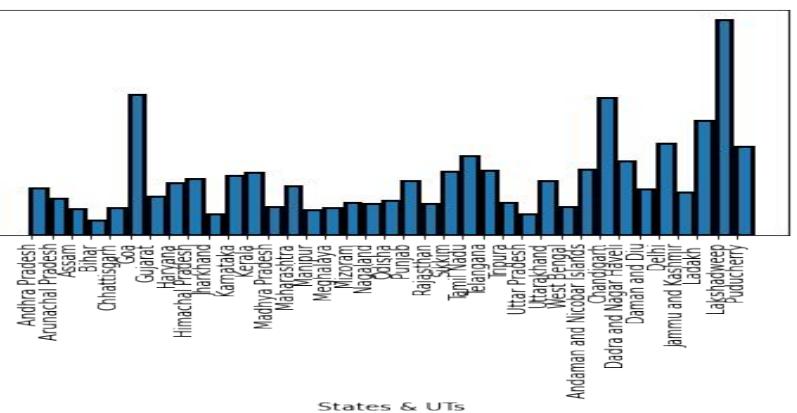
Number of functioning branches of commercial banks per 1,00,000 population

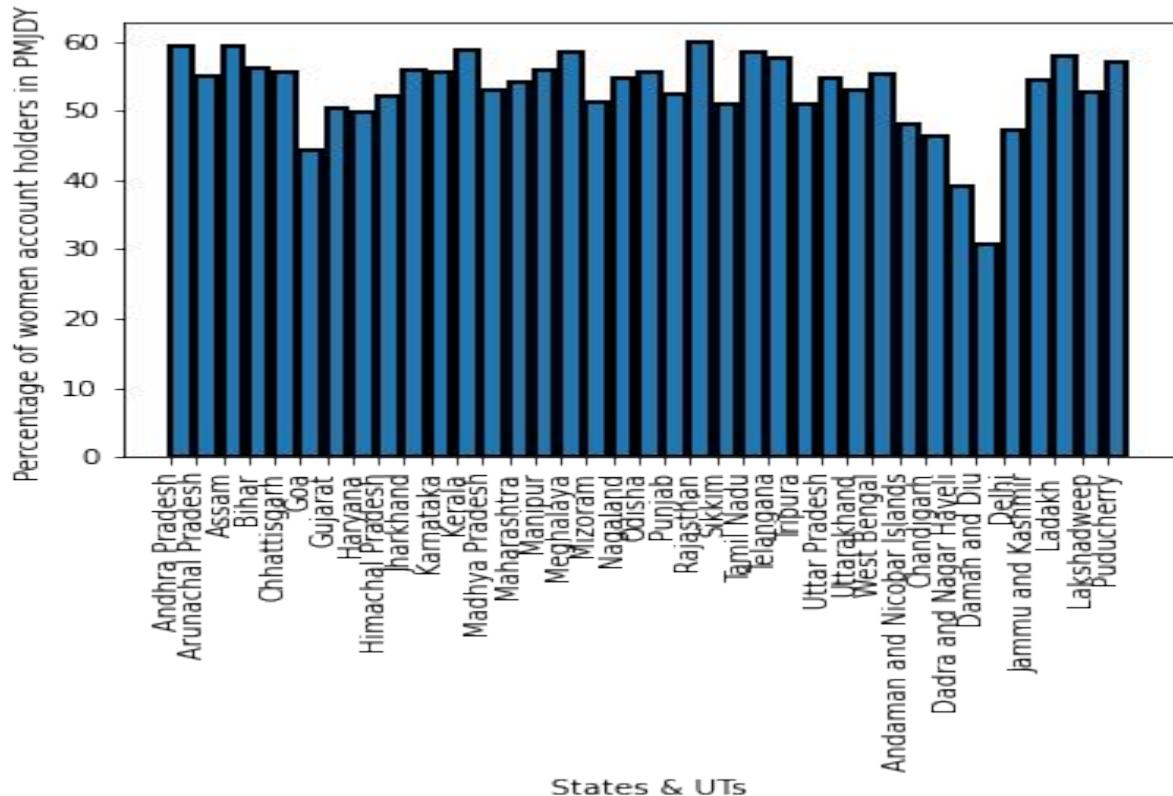


Percentage of households covered with a bank account under PNB/PDY against target.

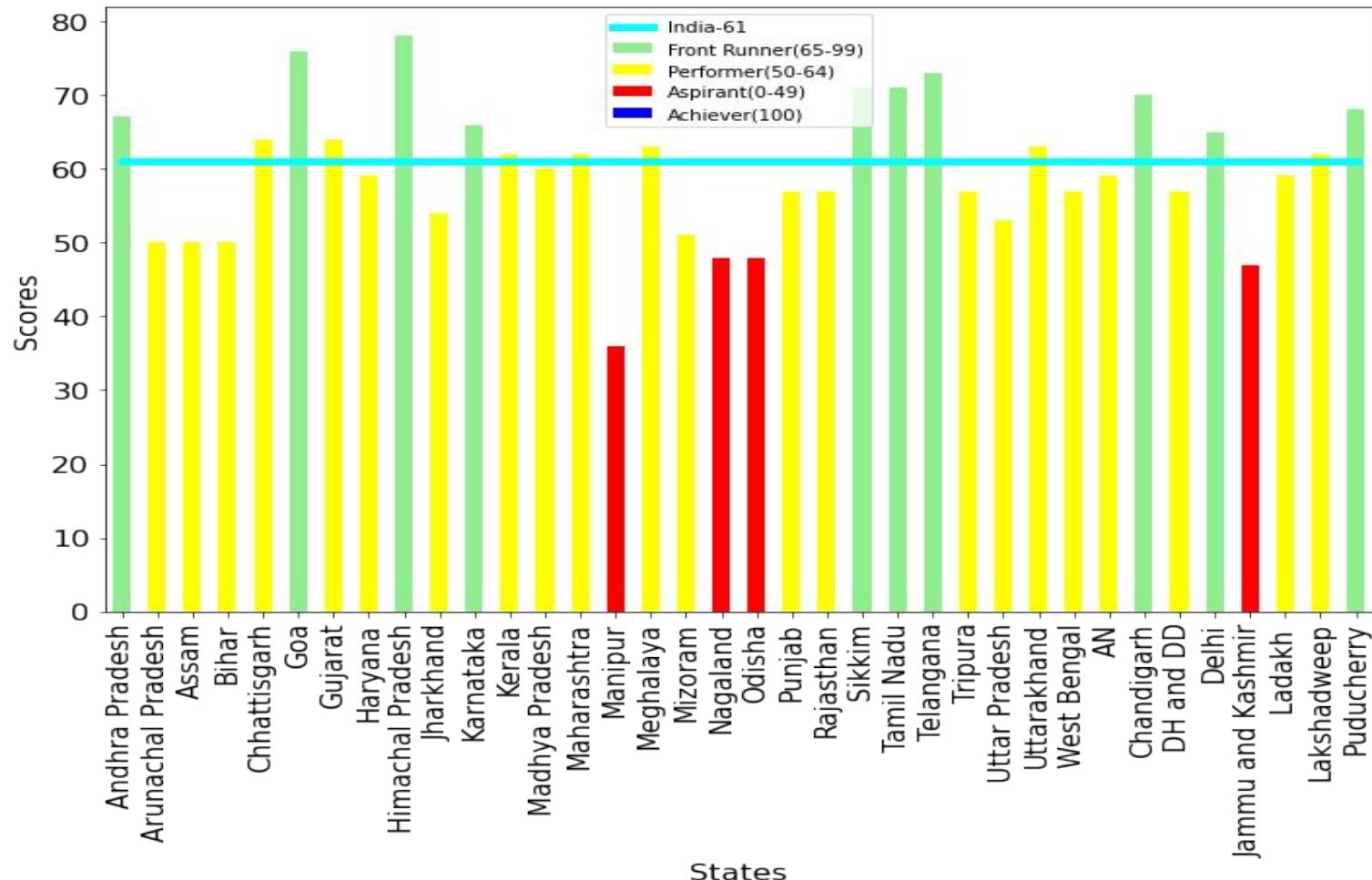


Automated Teller Machines (ATMs) per 1,00,000 population





SDG 8

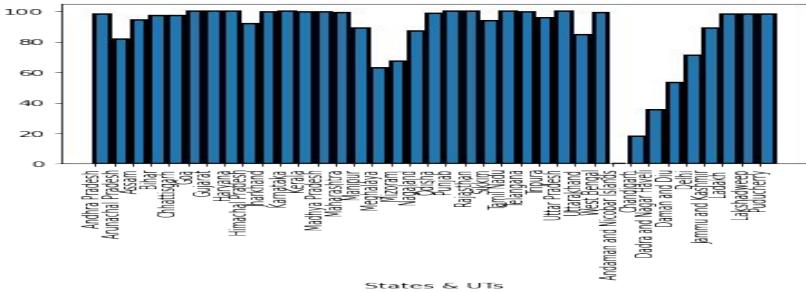


DATA STATISTICS FOR SDG 8

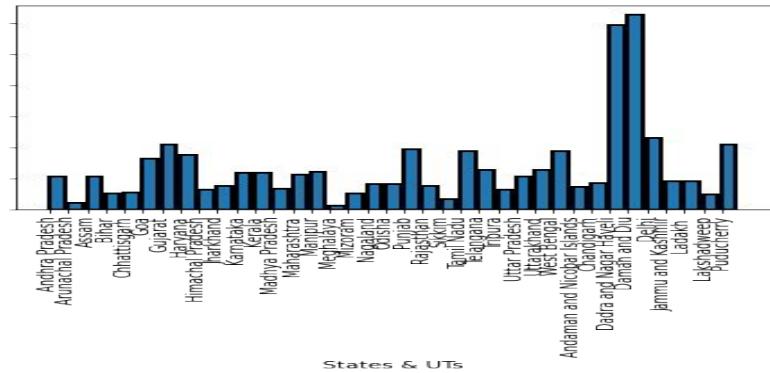
	Topic	Min_Value	Max_Value	Average_Val	No.of Null values
0	Annual growth rate of GDP (constant prices) per ...	0.735896	13.793487	5.072370	4
1	Ease of Doing Business (EODB) Score (feedback ...	0.000000	52.400000	13.684865	8
2	Unemployment rate (%) (15-59 years)	0.000000	32.000000	7.751351	1
3	Labour Force Participation Rate (LFPR) (%) (15...	41.400000	72.100000	55.913514	0
4	Percentage of regular wage/ salaried employees...	11.400000	70.300000	43.459459	0
5	Percentage of households covered with a bank a...	99.710000	100.000000	99.966757	0
6	Number of functioning branches of commercial b...	6.372698	45.782357	15.998302	0
7	Automated Teller Machines (ATMs) per 1,00,000 ...	6.841954	97.058824	25.795441	0
8	Percentage of women account holders in PMJDY	30.830000	59.850000	52.993514	0

SDG 9 : Industry, Innovation and Infrastructure

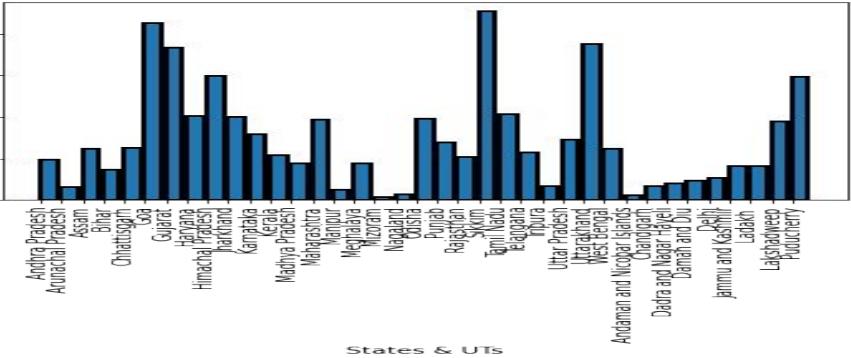
Percentage of target habitats corrected by all weather roads under Pradhan Mantri Gram Sadak Yojana (PMGSY)



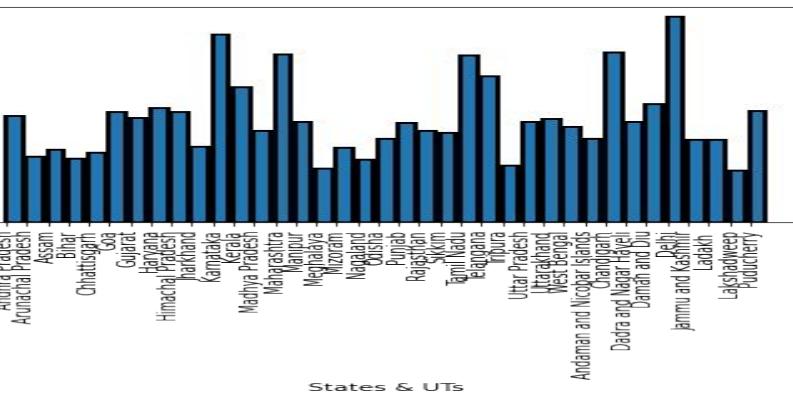
Manufacturing employment as a percentage of total employment



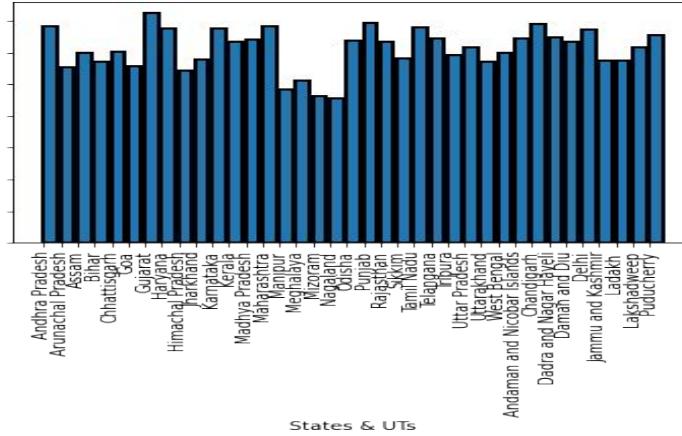
Percentage Share of GVA in manufacturing to total GVA (current prices)



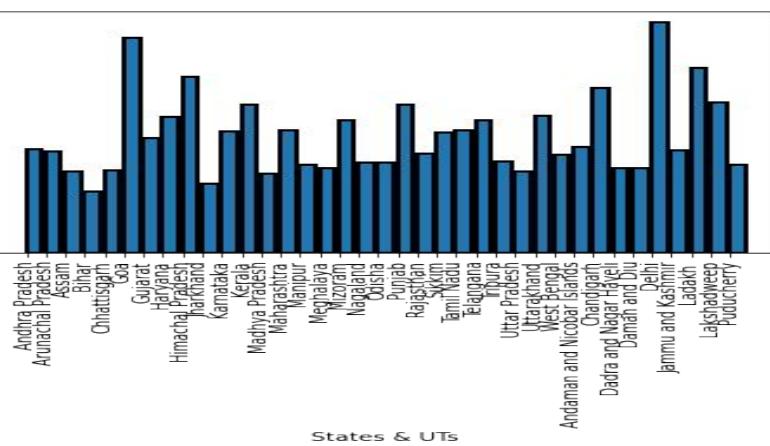
Innovation score as per the India Innovation Index



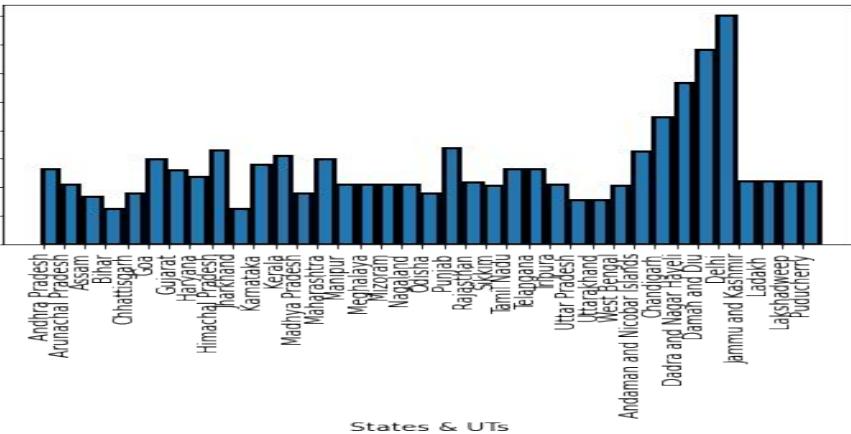
Score as per Logistics Ease Across Different States (LEADS) report

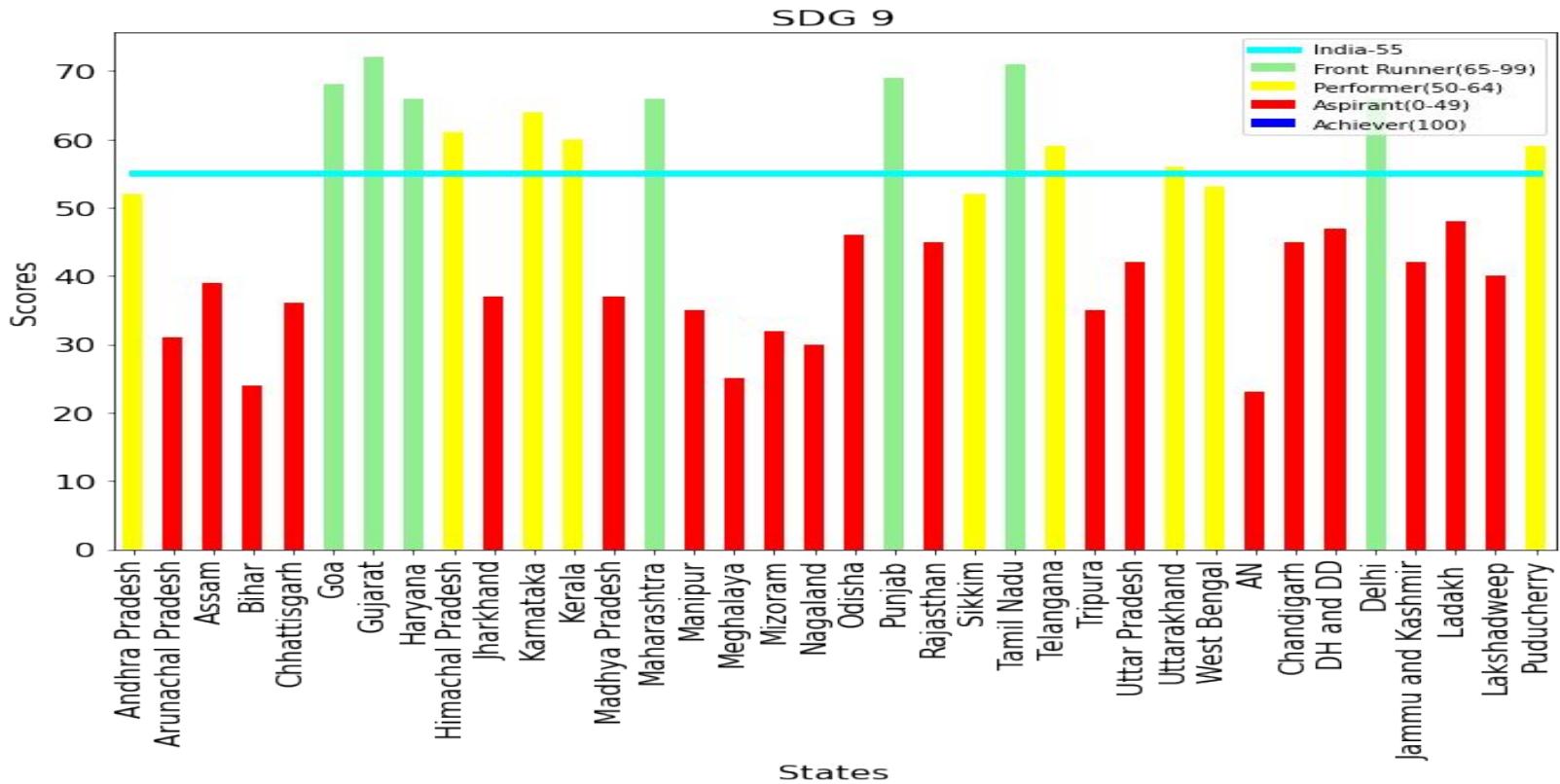


Number of mobile connections per 100 persons (mobile tele density)



Number of internet subscribers per 100 population

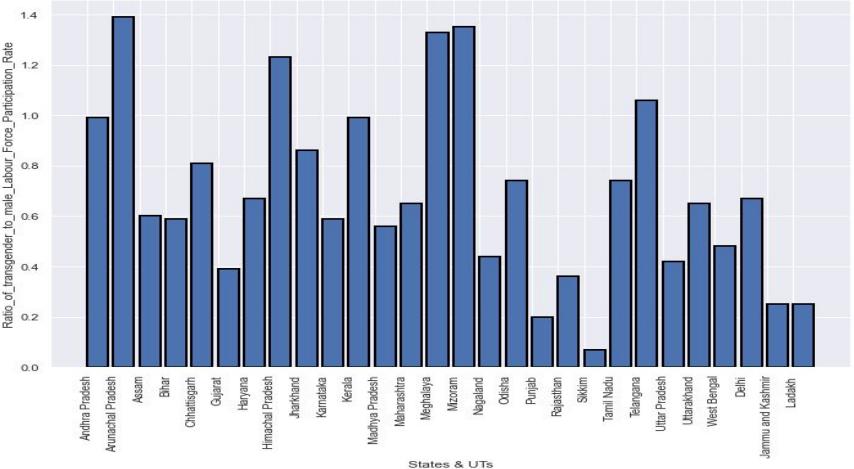
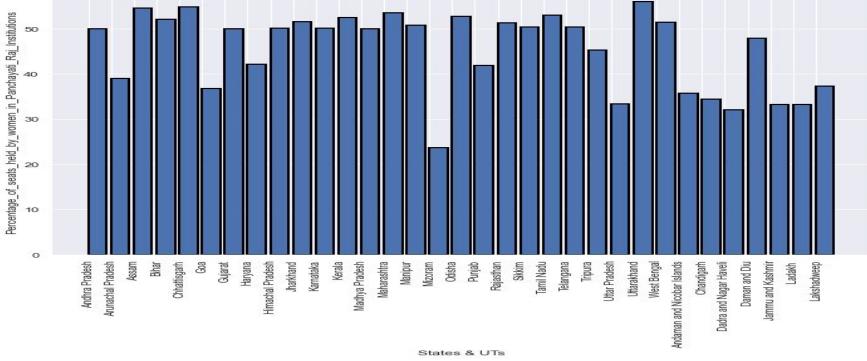
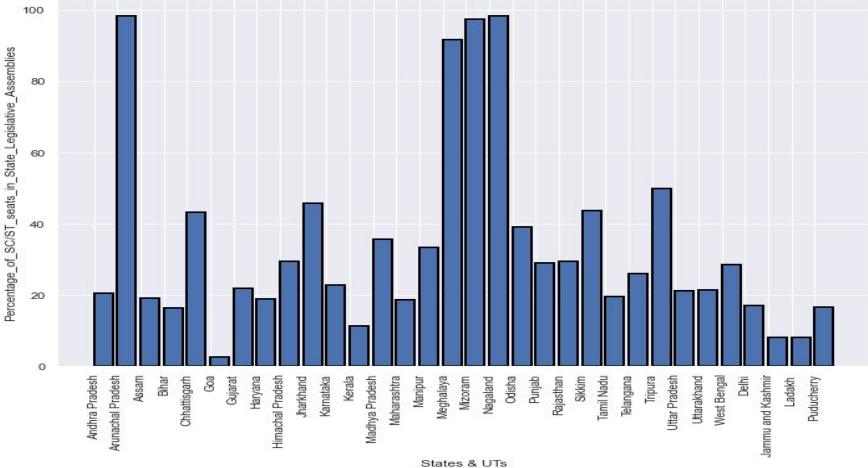
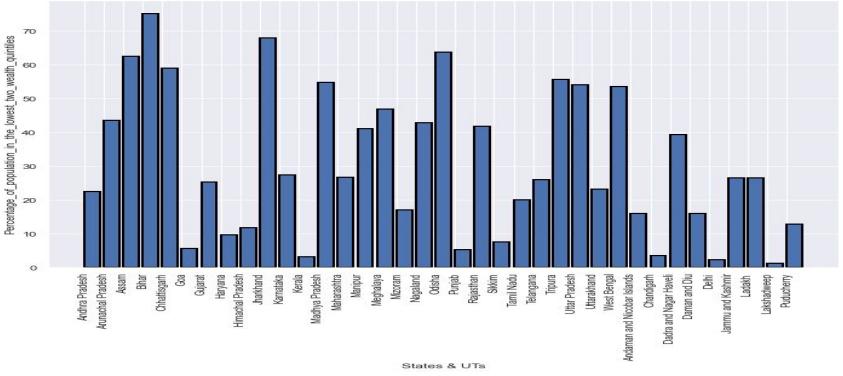


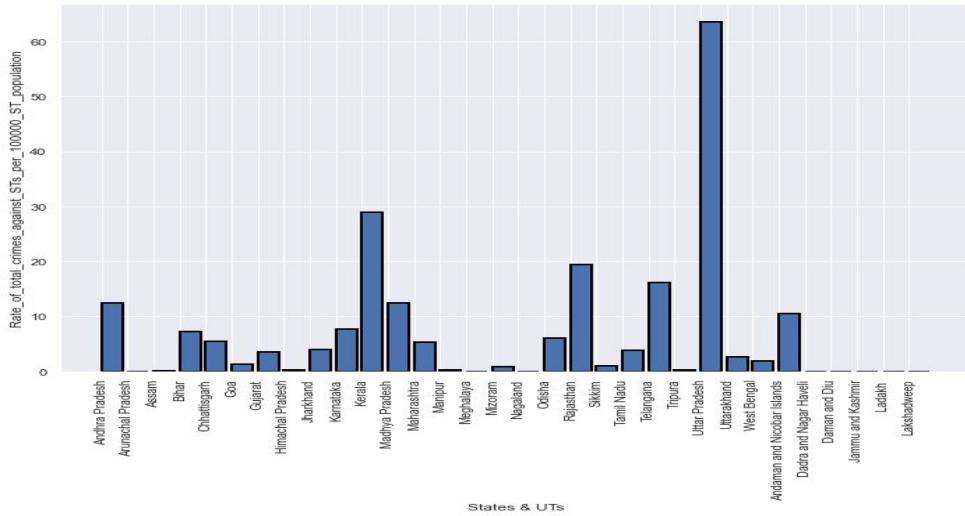
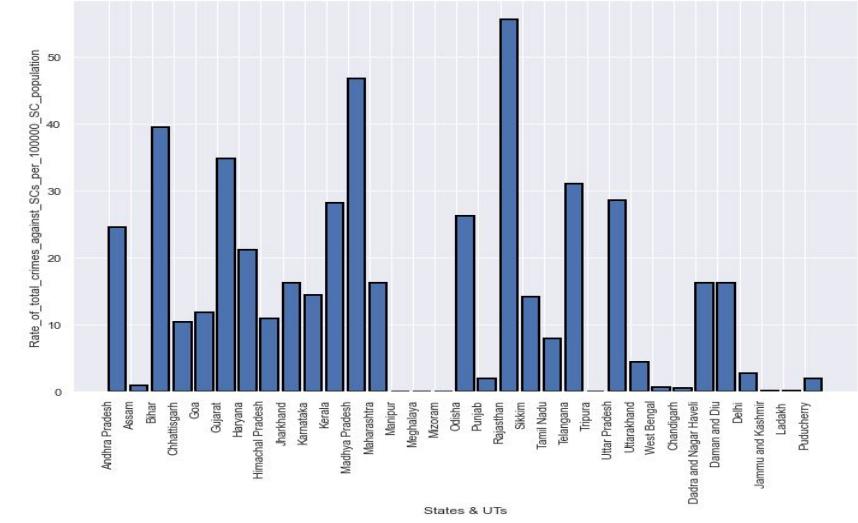


DATA STATISTICS FOR SDG 9

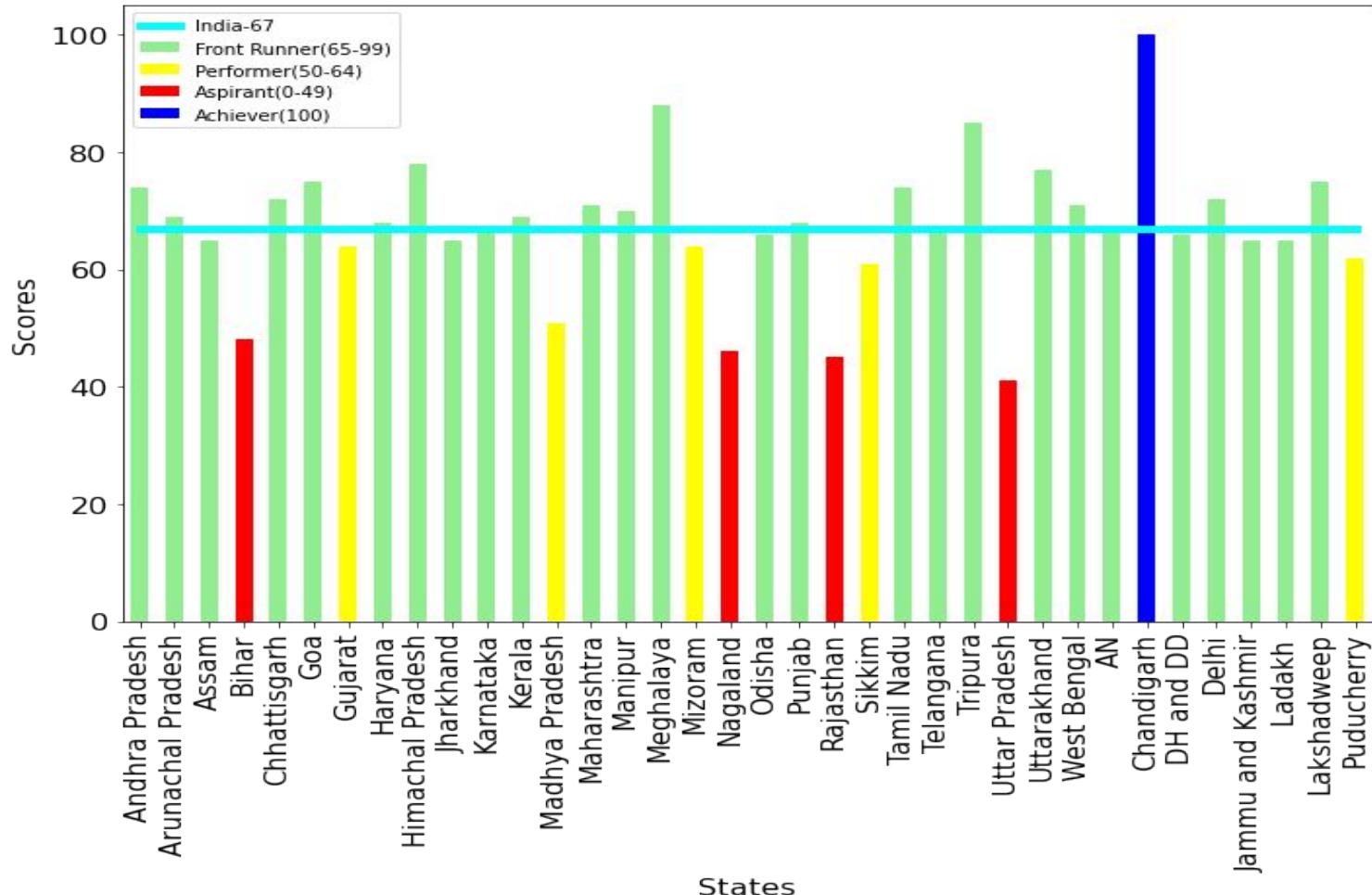
	Topic	Min_Value	Max_Value	Average_Val	No.of Null values
0	Percentage of targeted habitations connected b...	0.000000	100.006256	76.350289	7
1	Percentage Share of GVA in manufacturing to to...	0.644401	45.426237	13.694486	3
2	Manufacturing employment as a percentage of to...	1.270000	62.570000	13.352432	0
3	Innovation score as per the India Innovation I...	11.710000	46.600000	23.311081	0
4	Score as per Logistics Ease Across Different S...	2.280000	3.620000	2.880000	2
5	Number of mobile connections per 100 persons ...	50.650000	190.610000	96.747568	0
6	Number of internet subscribers per 100 population	30.990000	199.880000	50.817297	6

SDG 10 : Reduce Inequalities





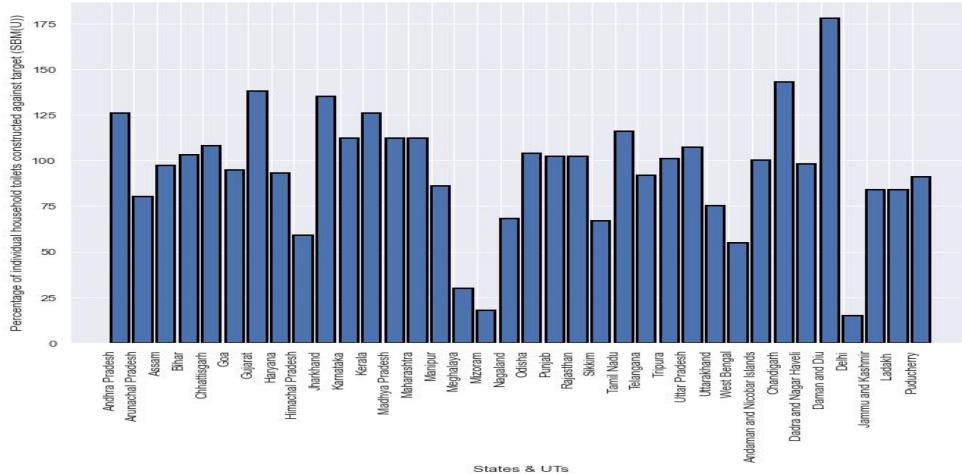
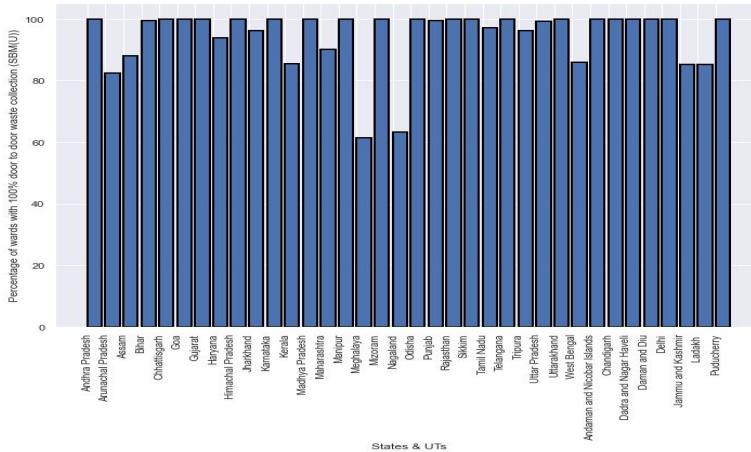
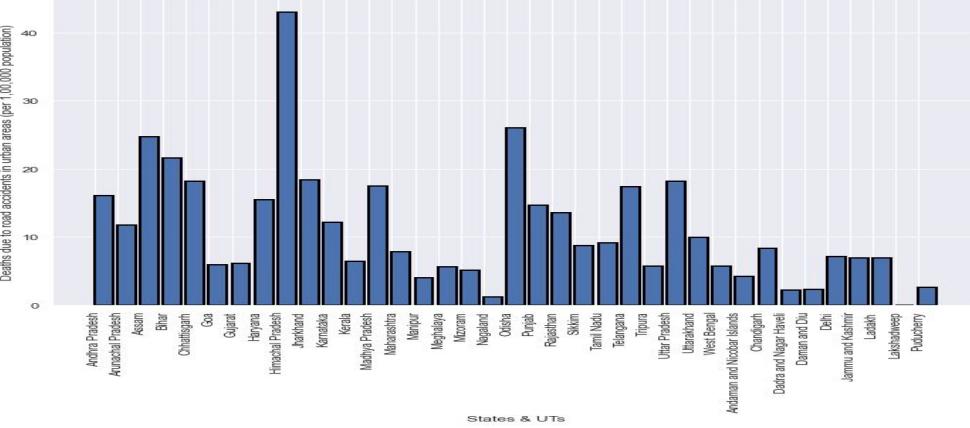
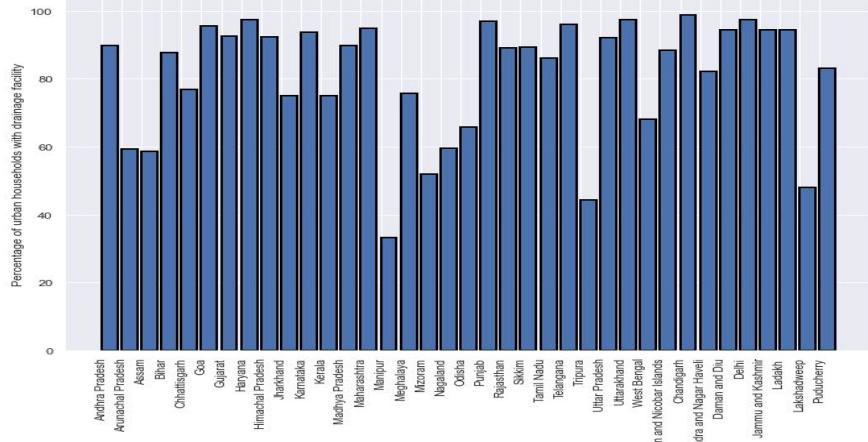
SDG 10

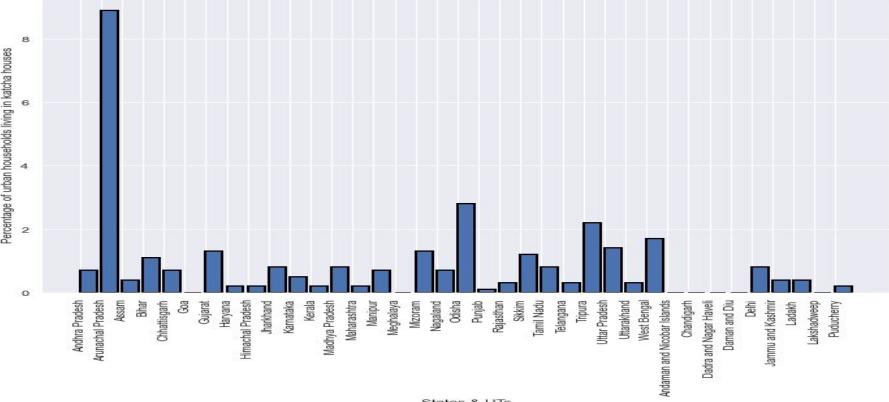
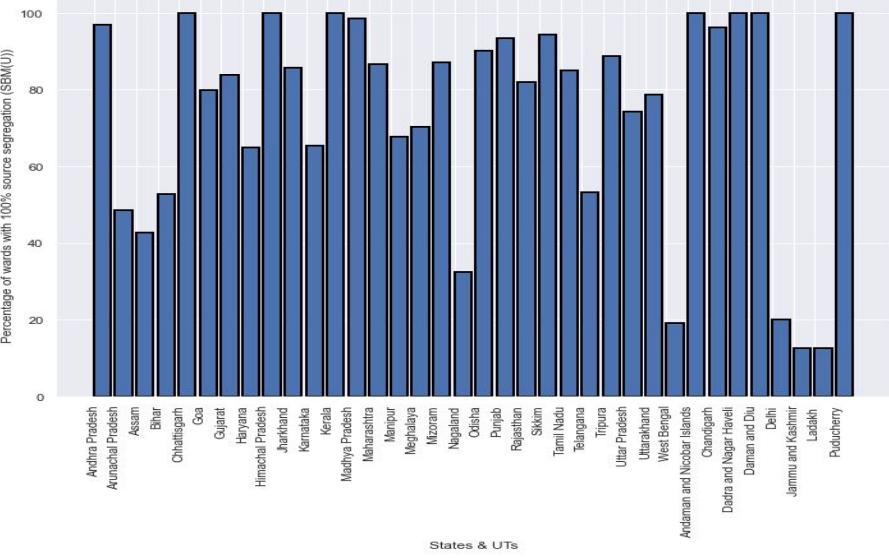
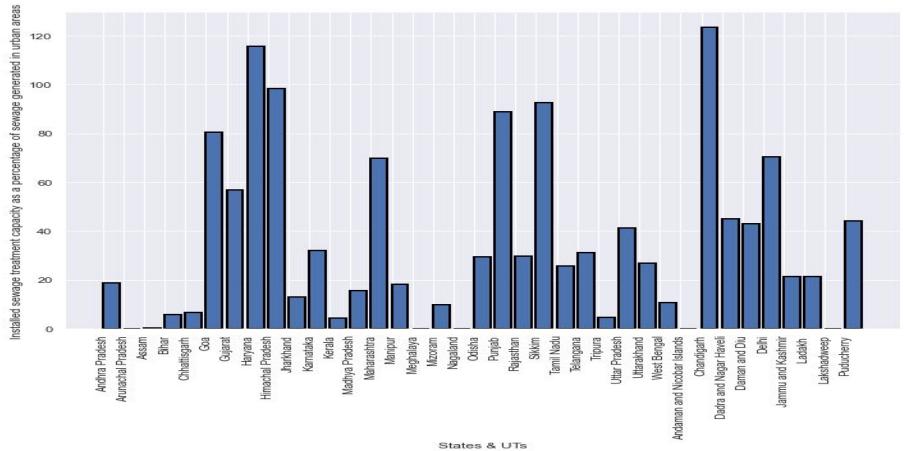
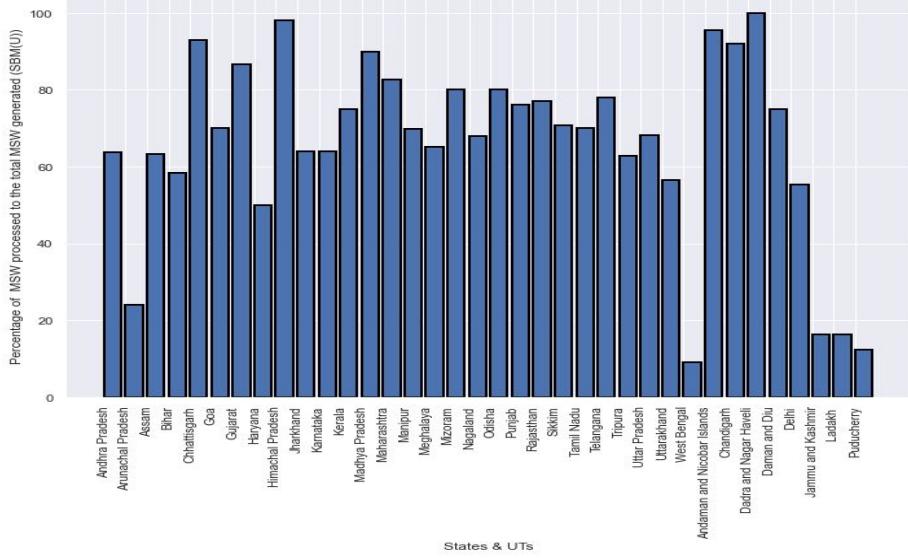


DATA STATISTICS FOR SDG 10

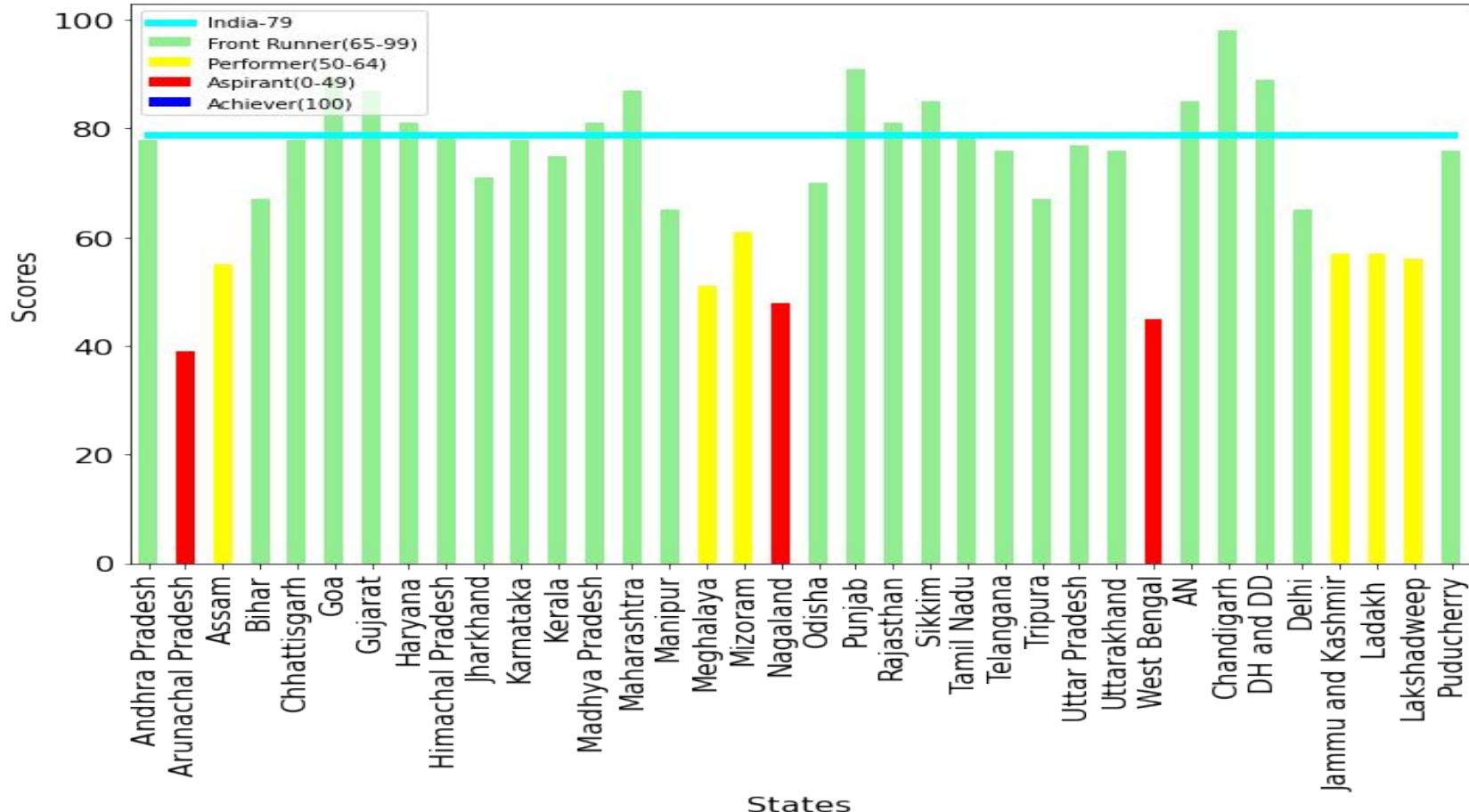
	Topic	Min_Value	Max_Value	Average_Val	No.of Null values
0	Percentage of population in the lowest two wea...	1.20	75.200000	30.727027	0
1	Percentage of elected women over total seats i...	0.00	100.000000	13.421351	14
2	Percentage of seats held by women in Panchayat...	23.64	56.020000	40.543514	4
3	Percentage of SC/ST seats in State Legislative...	2.50	98.333333	29.296930	5
4	Ratio of transgender to male Labour Force Par...	0.07	1.390000	0.522432	9
5	Rate of total crimes against SCs (per 1,00,000...)	0.00	55.600000	13.089189	8
6	Rate of total crimes against STs (per 1,00,000...)	0.00	63.600000	5.821622	13

SDG 11 : Sustainable Cities and Communities





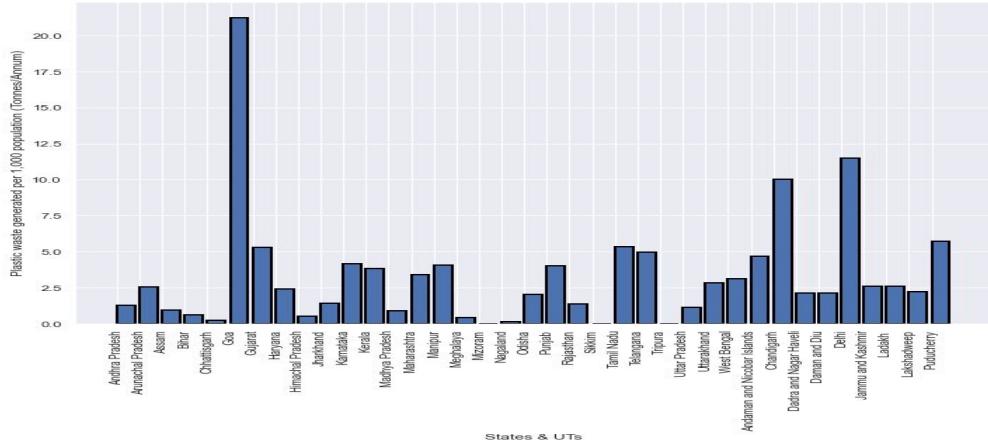
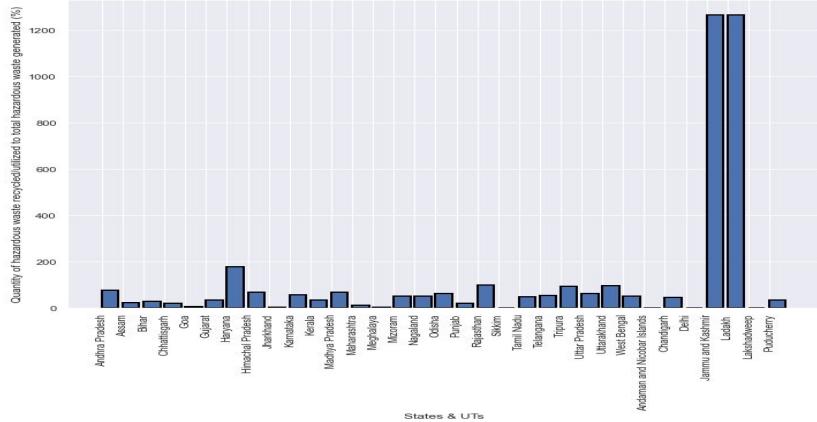
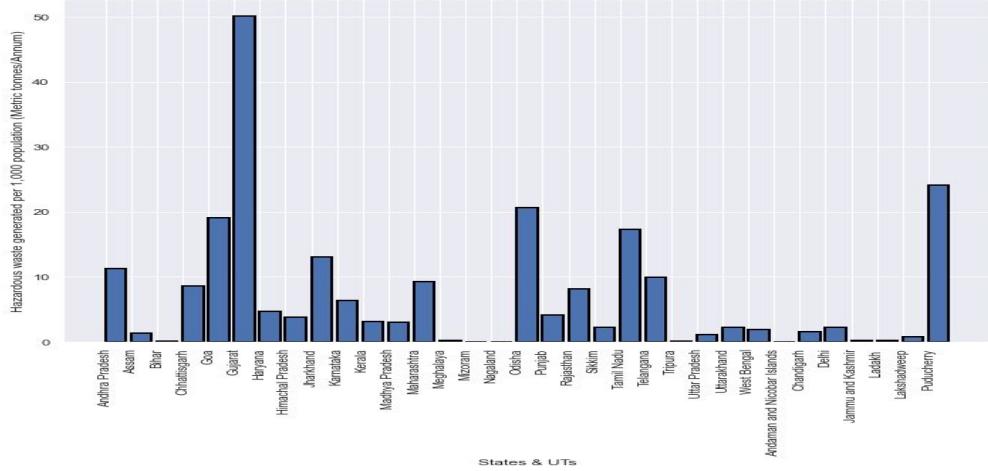
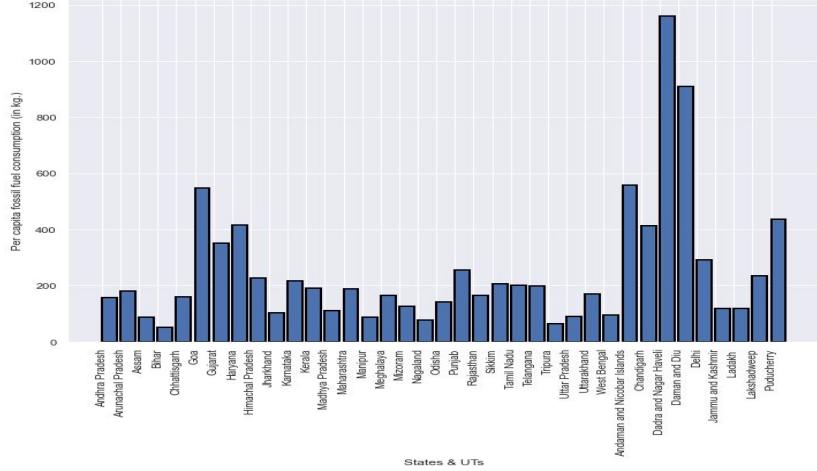
SDG 11

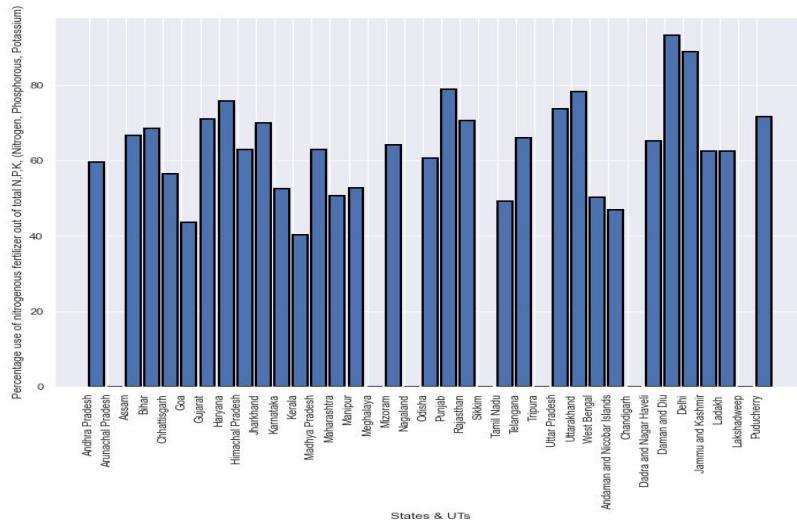
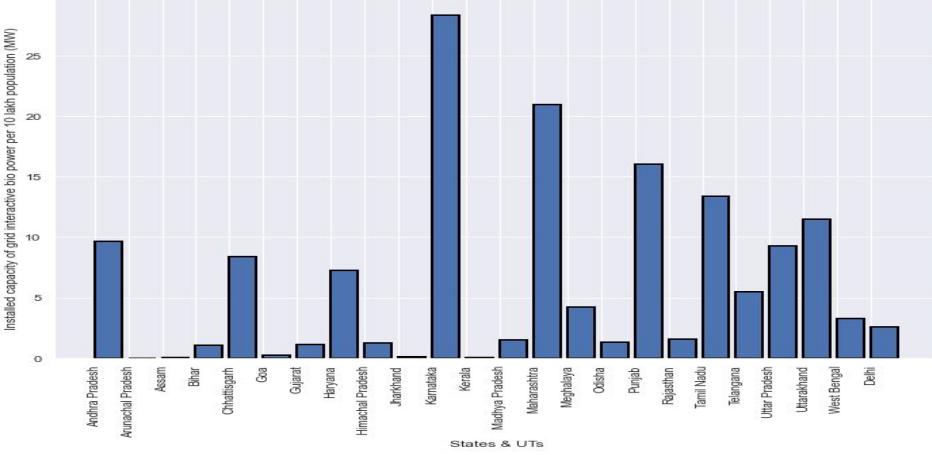
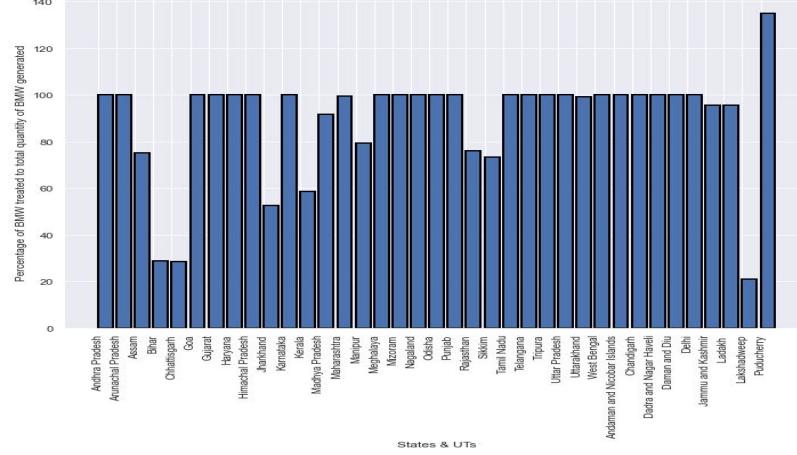


DATA STATISTICS FOR SDG 11

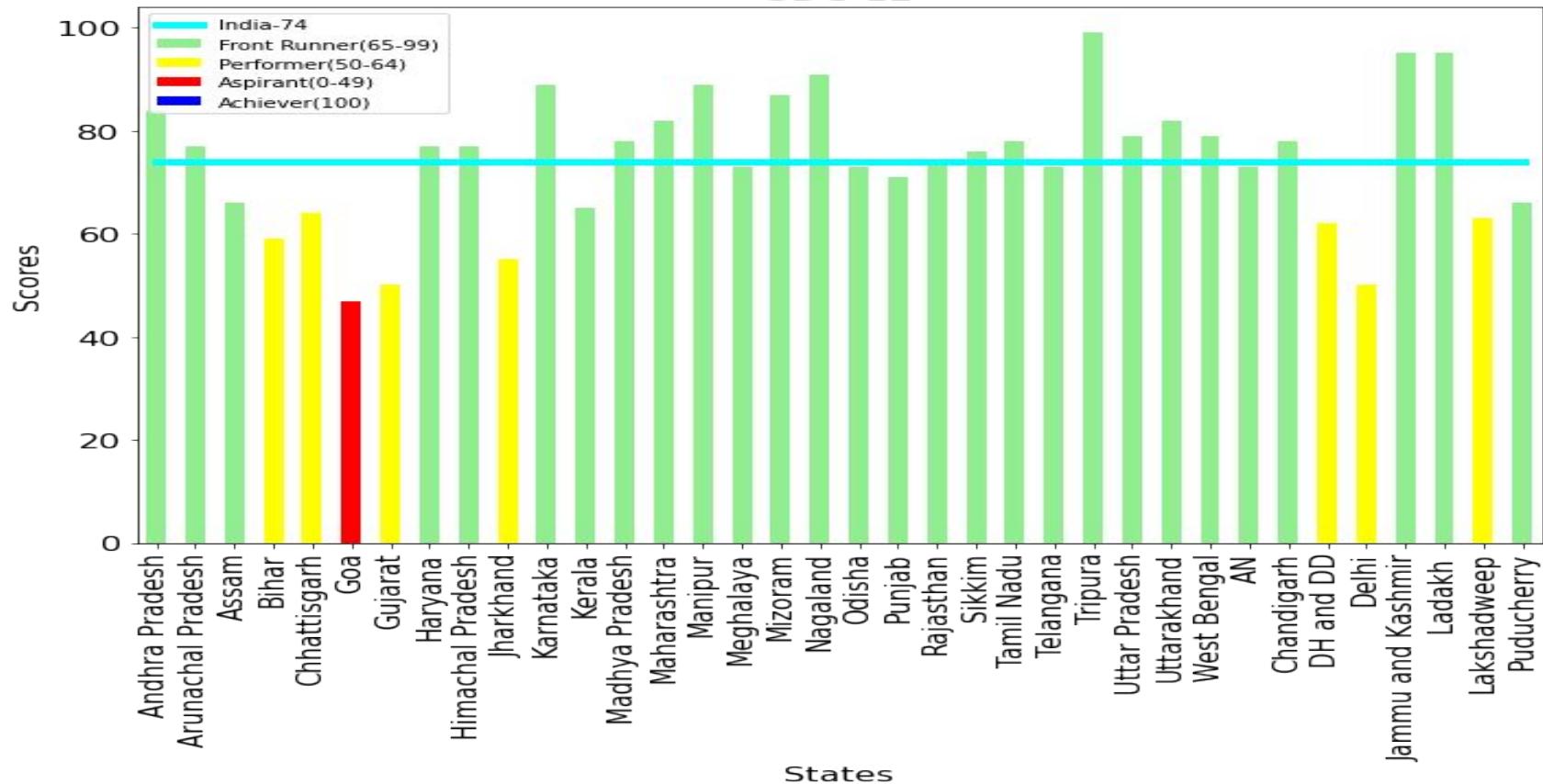
	Topic	Min_Value	Max_Value	Average_Val	No.of Null values
0	Percentage of urban households living in katch...	0.000000	8.900000	0.854054	7
1	Deaths due to road accidents in urban areas (p...	0.000000	43.048128	11.086974	1
2	Percentage of wards with 100% door to door was...	61.403509	100.000000	92.084140	1
3	Percentage of individual household toilets con...	15.000000	178.000000	92.183514	1
4	Percentage of MSW processed to the total MSW ...	9.090909	100.000000	64.194806	1
5	Percentage of wards with 100% source segregati...	12.454545	100.000000	71.940664	1
6	Installed sewage treatment capacity as a perce...	0.000000	123.566845	34.976038	5
7	Percentage of urban households with drainage f...	33.300000	98.900000	81.245946	0

SDG 12 : Responsible Consumption and Production



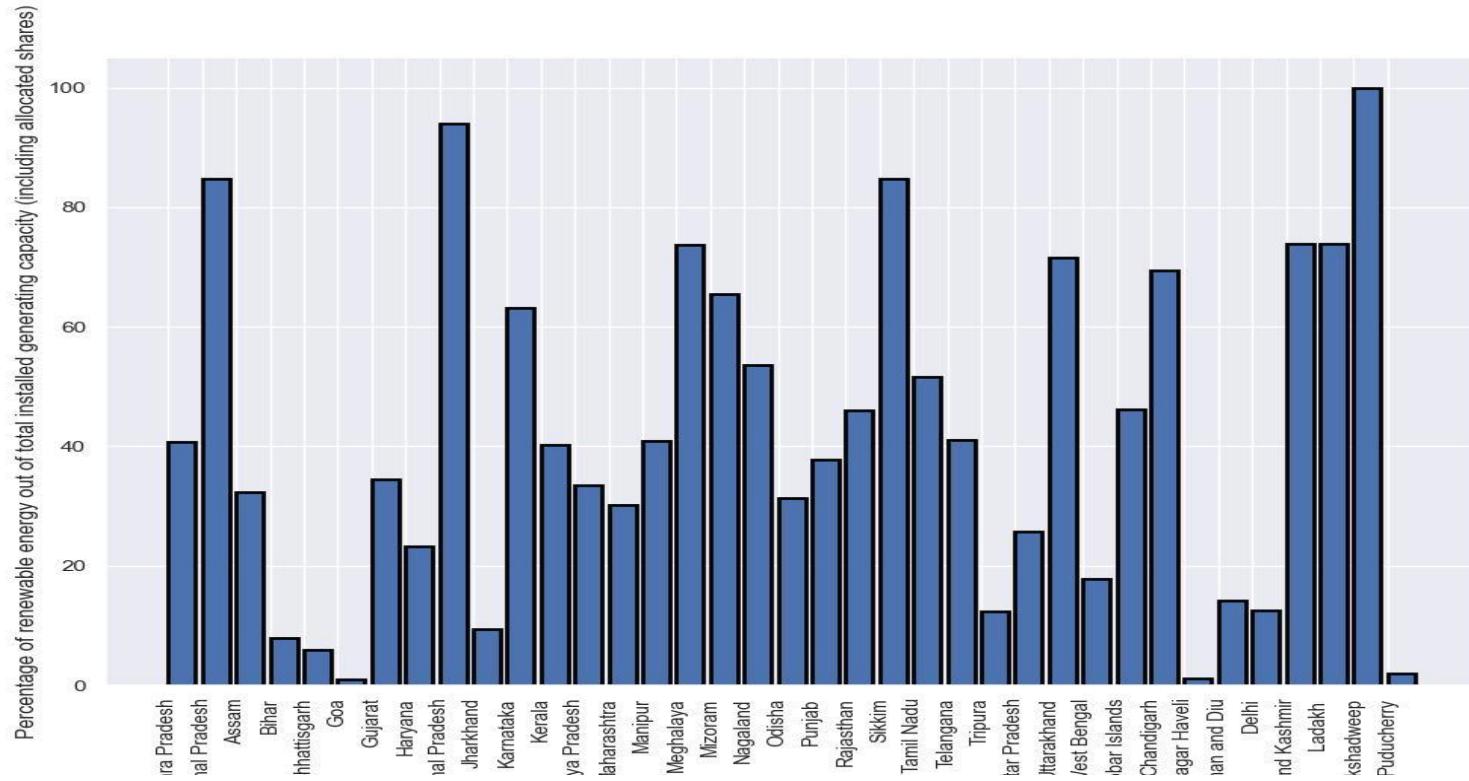


SDG 12

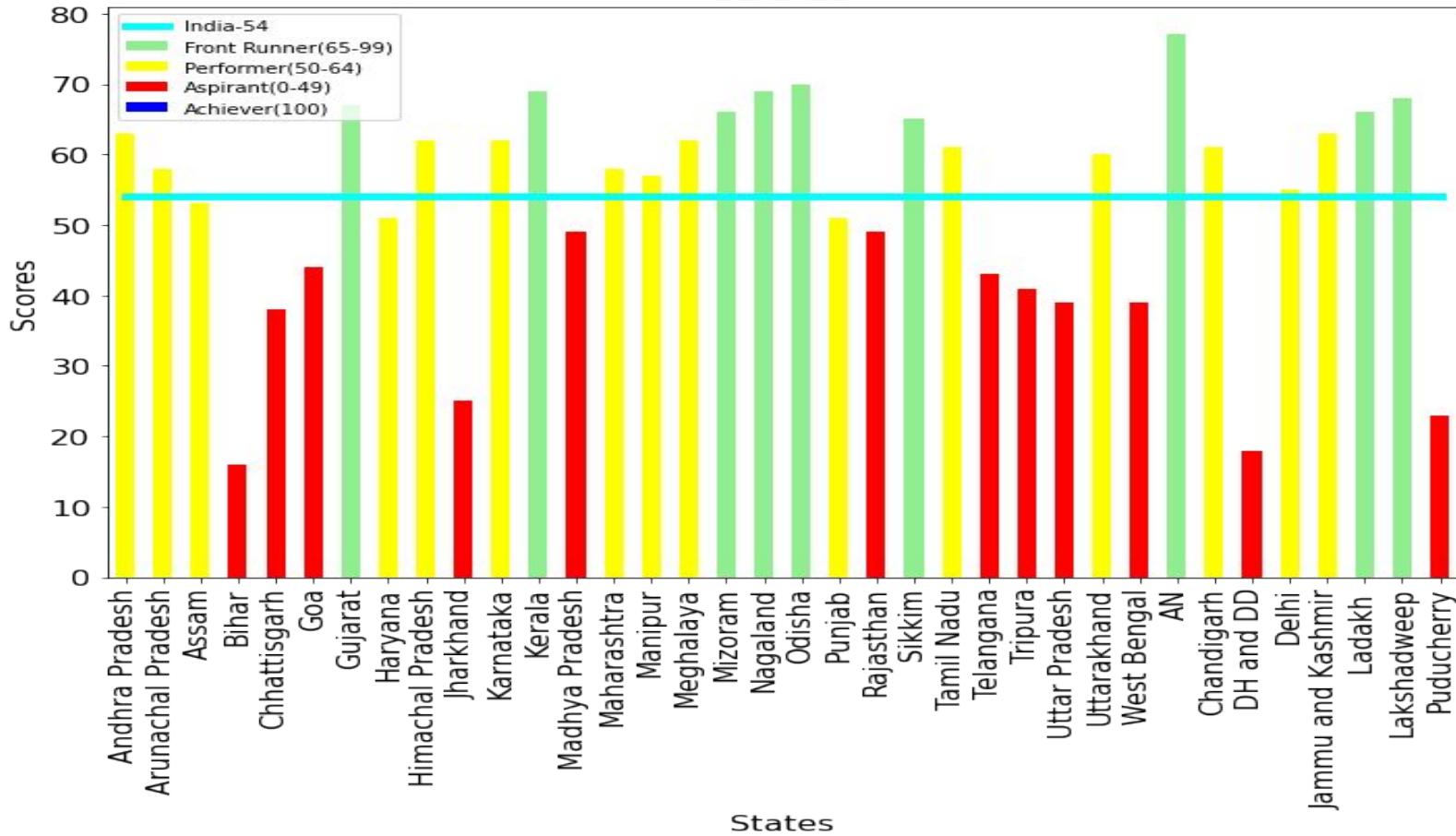


DATA STATISTICS FOR SDG 12

	Topic	Min_Value	Max_Value	Average_Val	No.of Null values
0	Per capita fossil fuel consumption (in kg.)	50.500000	1160.500000	250.735135	0
1	Percentage use of nitrogenous fertilizer out o...	0.000000	93.333333	51.816304	7
2	Hazardous waste generated per 1,000 population...	0.000017	50.116802	6.224314	4
3	Quantity of hazardous waste recycled/utilized ...	0.000000	1264.493554	104.933953	7
4	Plastic waste generated per 1,000 population (...	0.006611	21.238931	3.283970	0
5	Percentage of BMW treated to total quantity of...	20.872865	135.052549	89.451511	0
6	Installed capacity of grid interactive bio pow...	0.000000	28.397197	4.014676	14



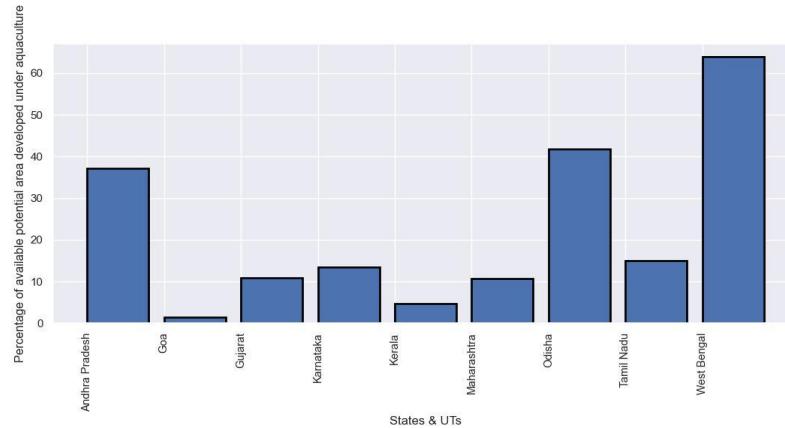
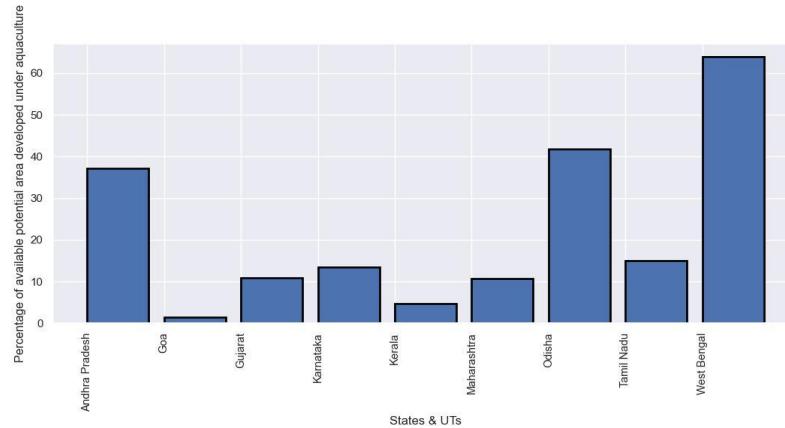
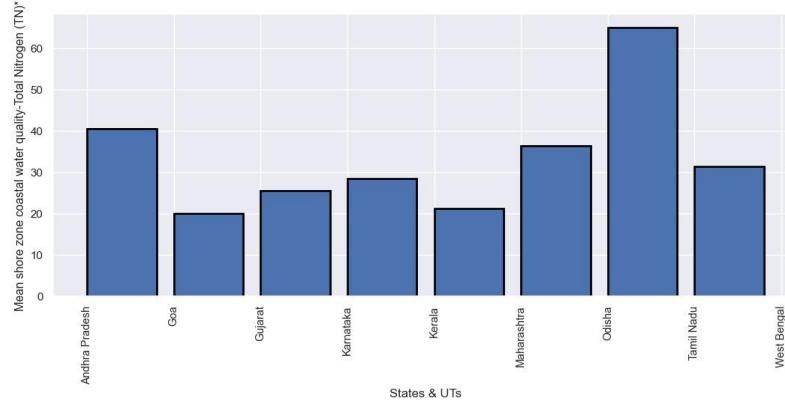
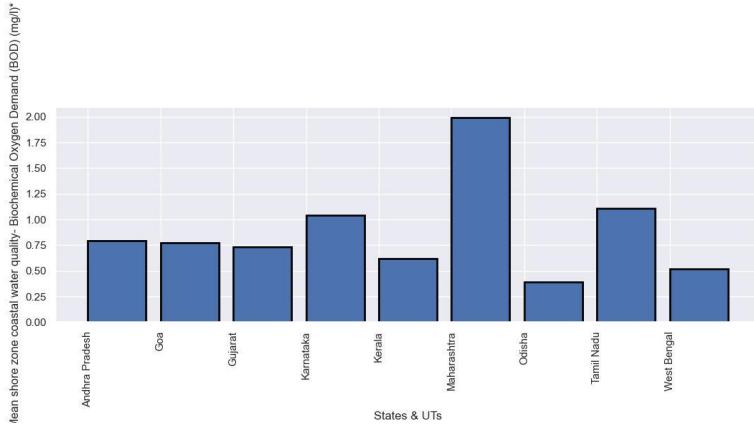
SDG 13



DATA STATISTICS FOR SDG 13

	Topic	Min_Value	Max_Value	Average_Val	No.of Null values
0	Number of human lives lost per 1 crore populat...	1.056301	471.931756	36.442719	12
1	Disaster preparedness score as per Disaster Re...	7.500000	27.500000	17.878378	0
2	Percentage of renewable energy out of total in...	1.055007	100.000000	41.838301	0
3	CO2 saved from LED bulbs per 1,000 population ...	5.968658	309.382353	49.192928	0
4	Disability Adjusted Life Years (DALY) rate at...	1408.000000	4528.000000	2395.324324	6

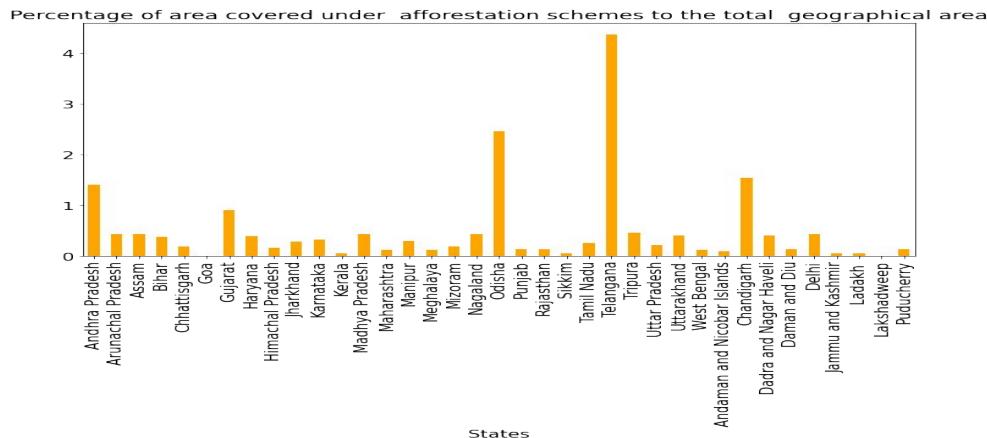
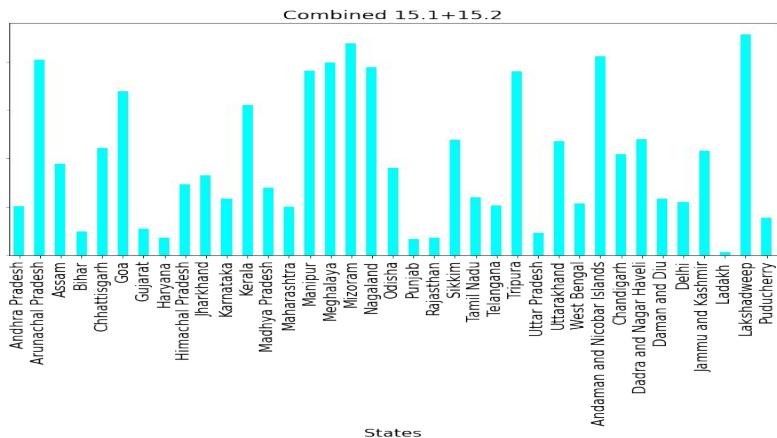
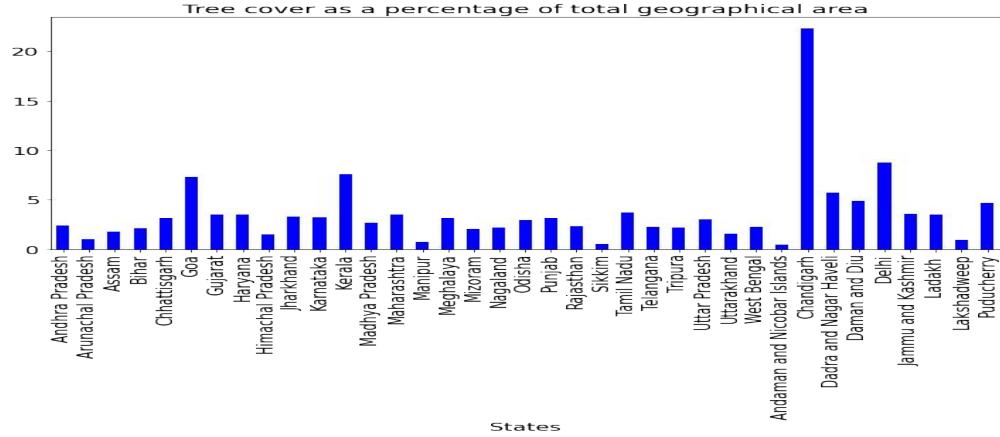
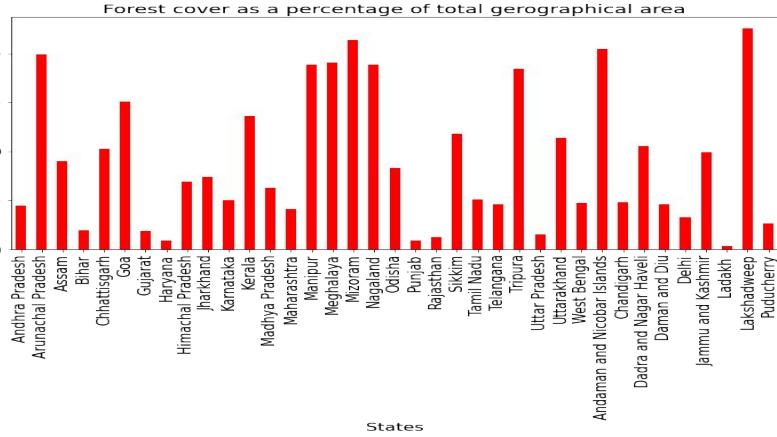
SDG 14 : Life Below Water

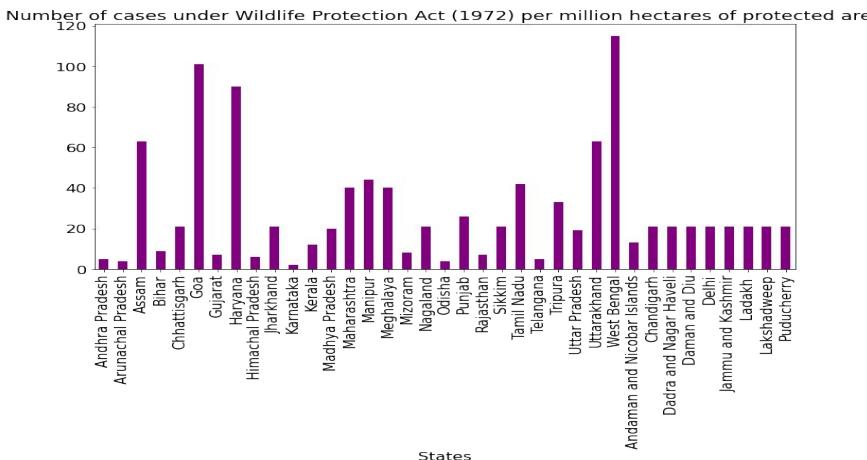
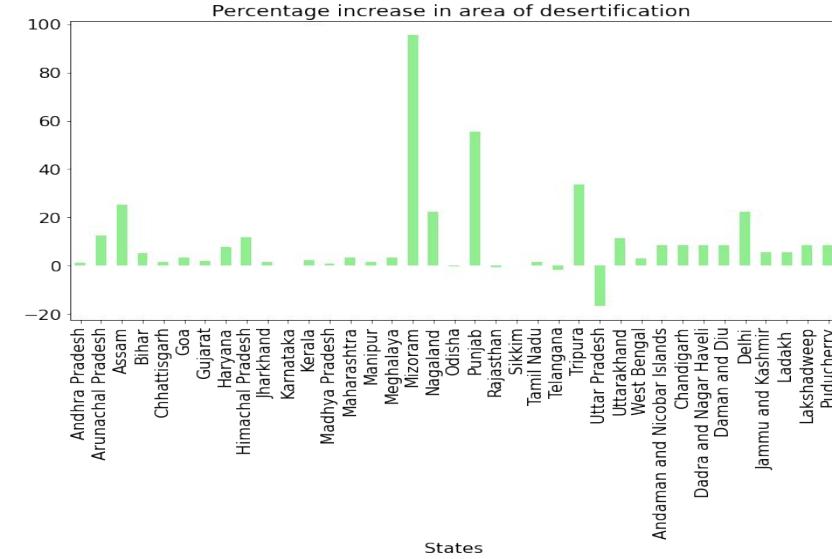
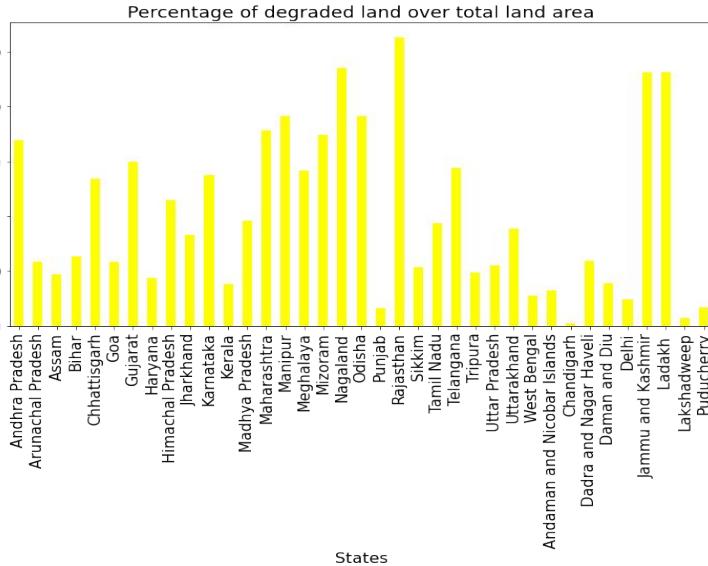


DATA STATISTICS FOR SDG 14

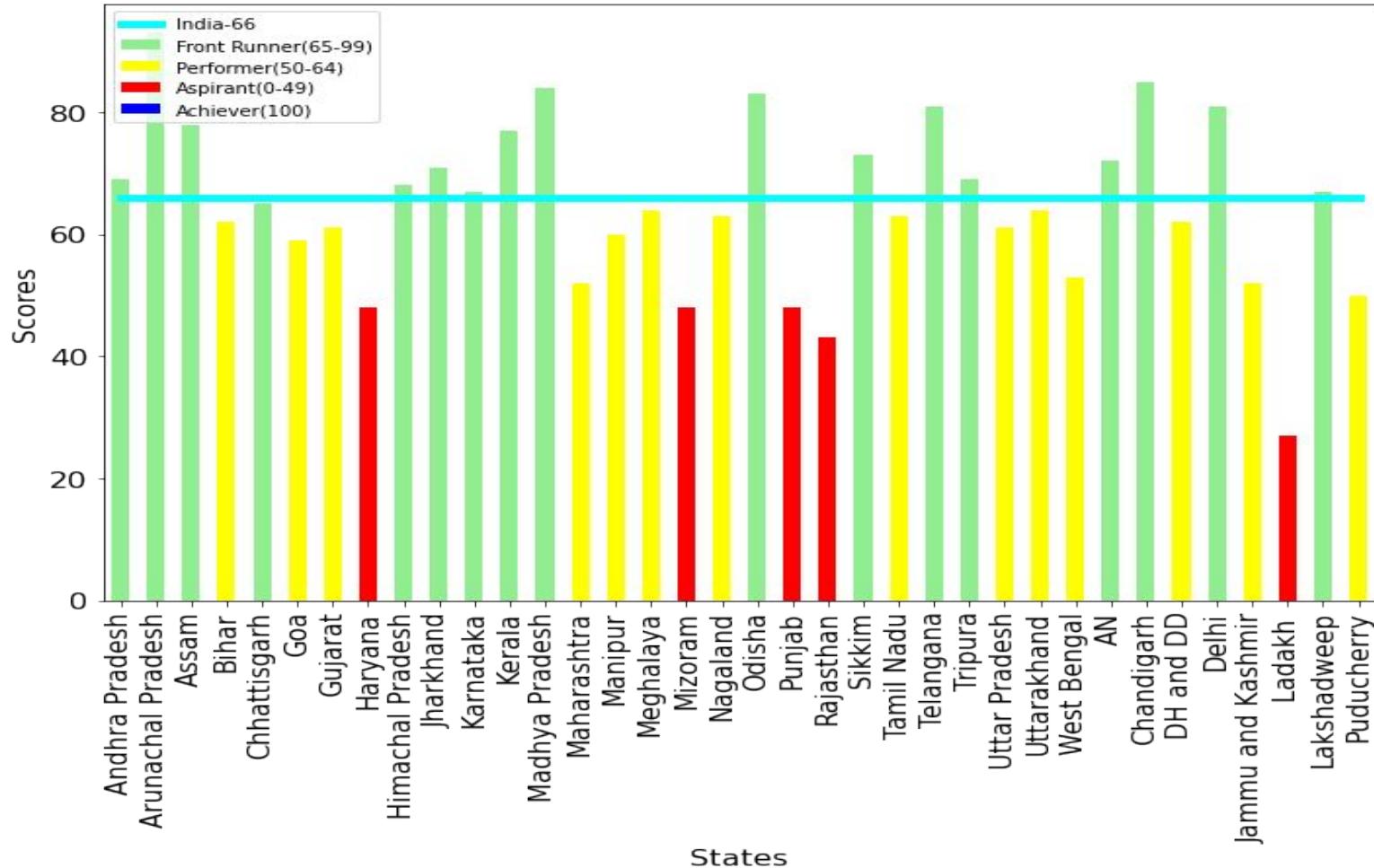
	Topic	Min_Value	Max_Value	Average_Val	No.of Null values
0	Mean shore zone coastal water quality- Biochem...	0.390000	1.990000	0.884444	0
1	Mean shore zone coastal water quality-Total Ni...	20.120000	65.000000	29.872222	1
2	Percentage increase in area under mangroves	-8.163265	5.263158	0.393676	4
3	Average marine acidity (pH) measured at repres...	7.970000	8.140000	8.041111	0
4	Percentage of available potential area develop...	1.447368	63.882222	22.078959	0

SDG 15 : Life on Land





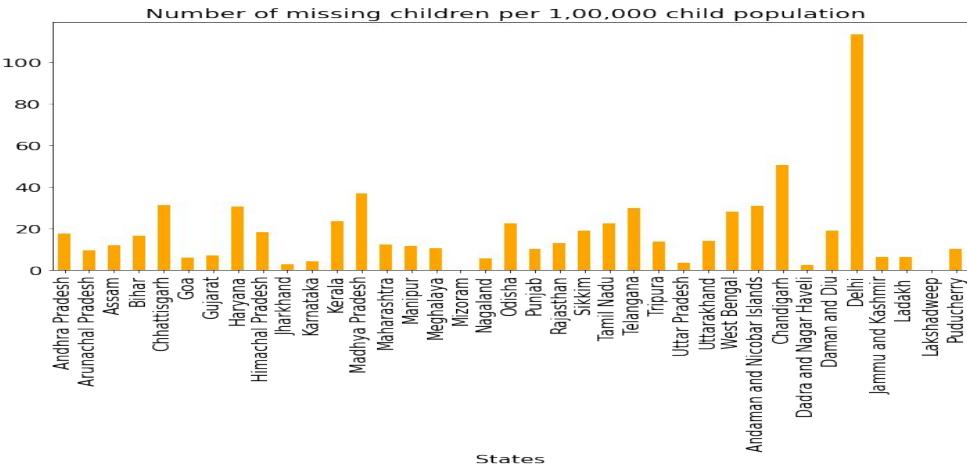
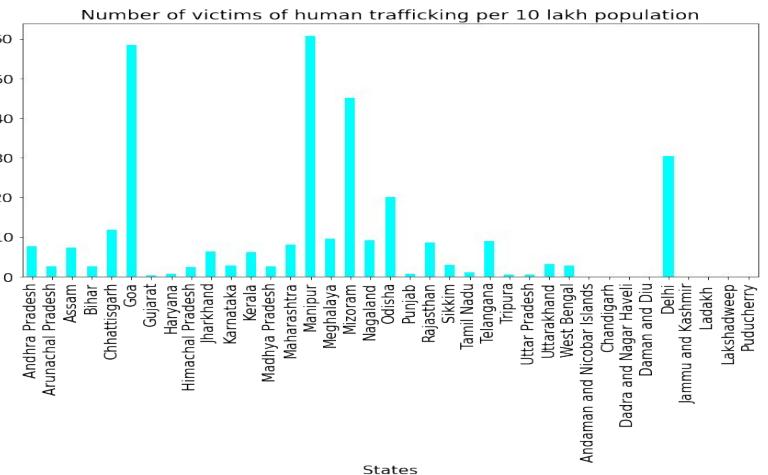
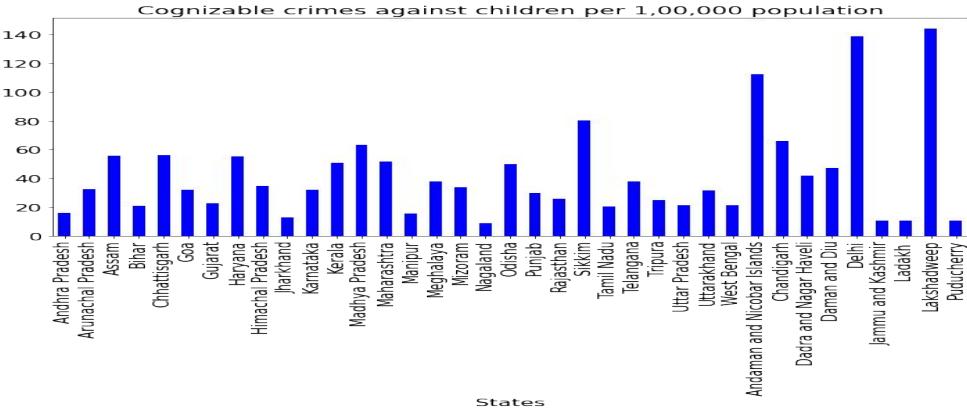
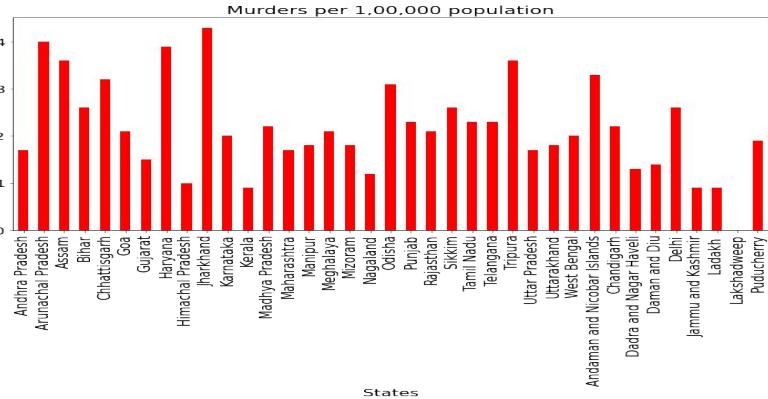
SDG 15

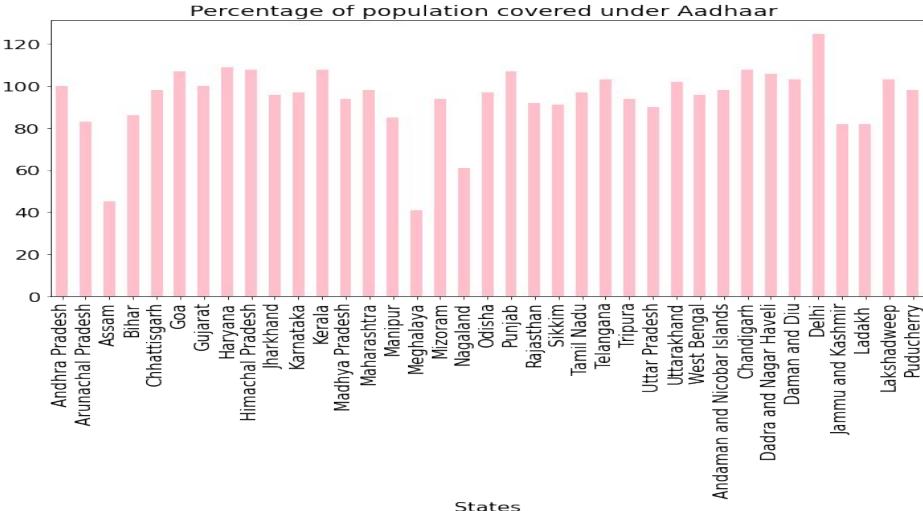
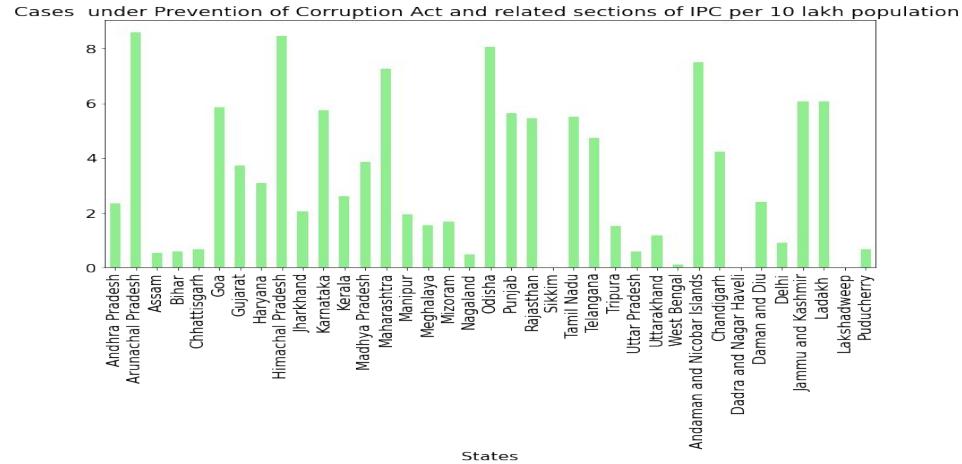
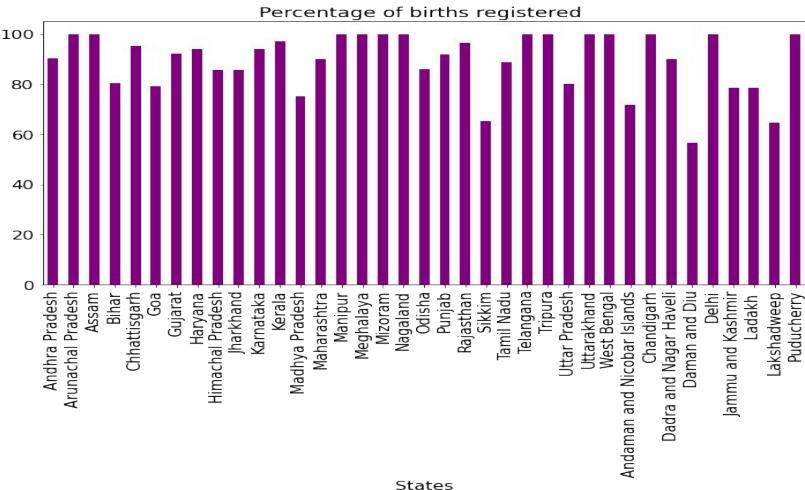
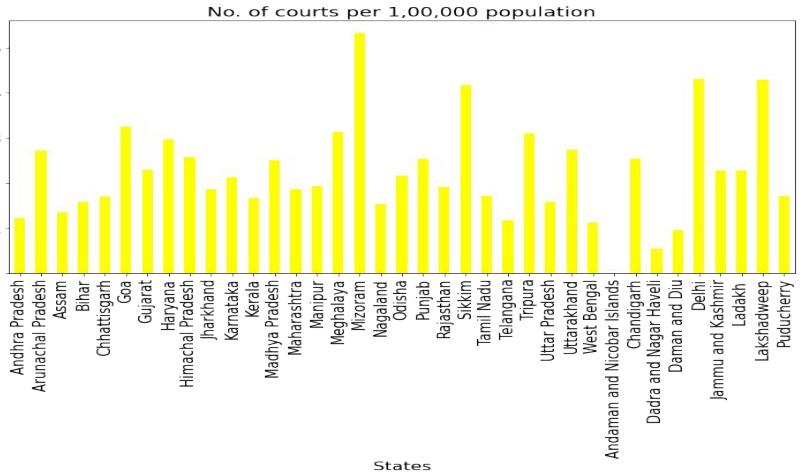


DATA STATISTICS FOR SDG 15

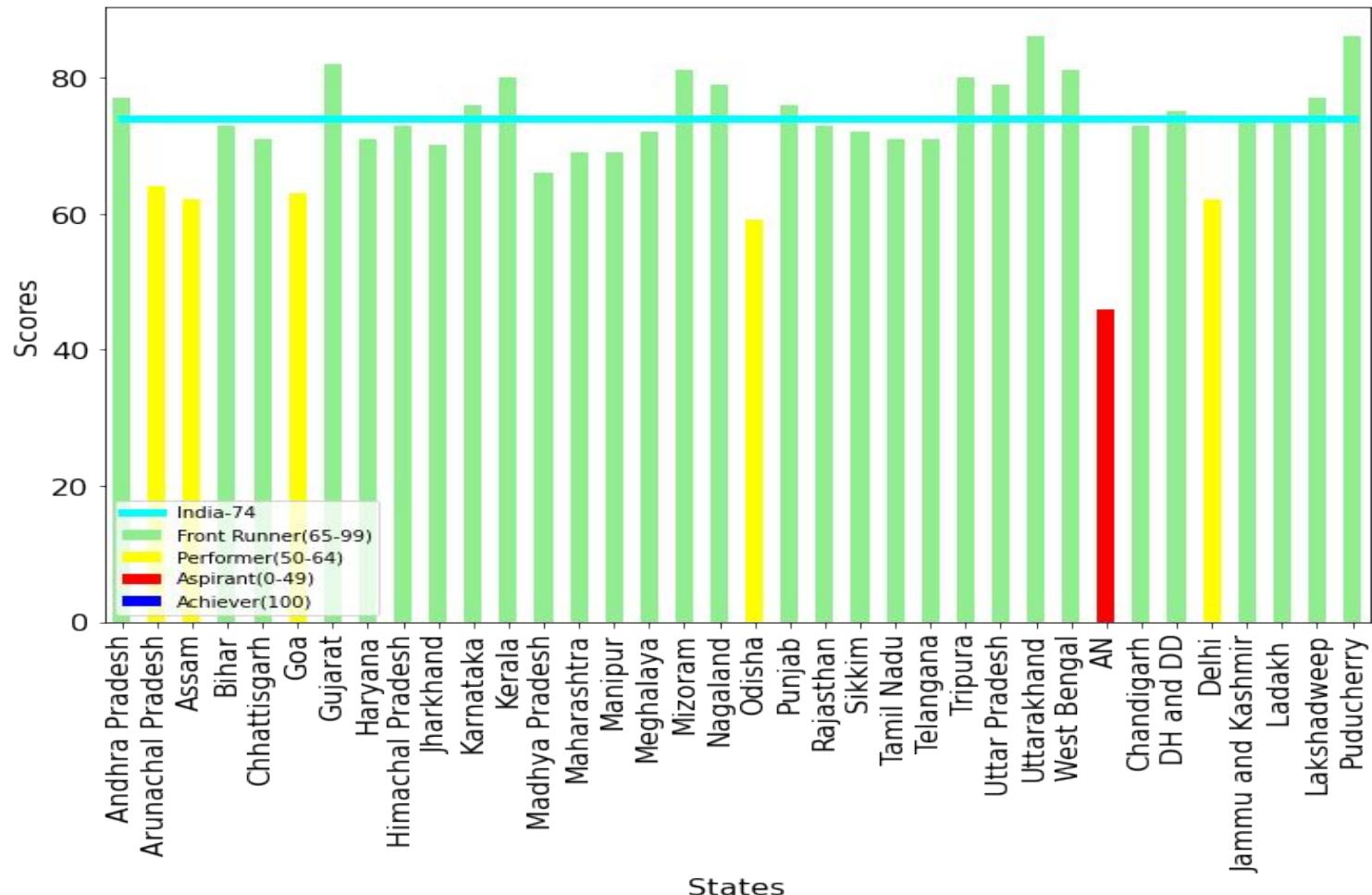
	Topic	Min_Value	Max_Value	Average_Val	No.of Null values
0	Forest cover as a percentage of total geographic area	1.470000	90.330000	35.915946	0
1	Tree cover as a percentage of total geographic area	0.500000	22.340000	3.524324	1
2	Combined 15.1+15.2	1.470000	91.300000	39.440270	0
3	Percentage of area covered under afforestation	0.000000	4.369077	0.427222	6
4	Percentage of degraded land over total land area	0.500000	52.694363	20.229236	0
5	Percentage increase in area of desertification	-16.687853	95.522201	8.719197	6
6	Number of cases under Wildlife Protection Act	2.000000	115.000000	21.027027	12

SDG 16 : Peace, Justice and Strong Institutions





SDG 16



DATA STATISTICS FOR SDG 16

	Topic	Min_Value	Max_Value	Average_Val	No.of Null values
0	Murders per 1,00,000 population	0.0	4.300000	2.159459	1
1	Cognizable crimes against children per 1,00,00...	8.8	144.400000	42.137838	0
2	Number of victims of human trafficking per 10 ...	0.0	60.771704	8.755643	8
3	Number of missing children per 1,00,000 child ...	0.0	113.482143	18.064847	2
4	No. of courts per 1,00,000 population	0.0	5.333333	2.233843	1
5	Cases under Prevention of Corruption Act and ...	0.0	8.609272	3.286479	3
6	Percentage of births registered	56.7	100.000000	89.378378	0
7	Percentage of population covered under Aadhaar	41.0	125.000000	94.162162	0



HYPOTHESIS TESTING

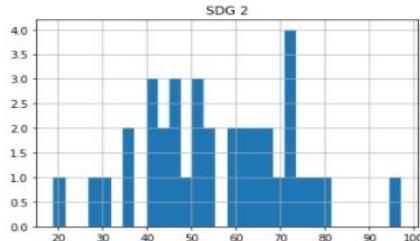
SDG 2: Hypothesis Testing (Zero Hunger)

```
In [283]: from scipy.stats import shapiro
DataToTest=df["SDG 2"]
stat, p=shapiro(DataToTest)
print('stat-%.2f, p-%.30f' % (stat,p))

if p > 0.05:
    print("yes normal distribution")
else:
    print("not a normal distribution")

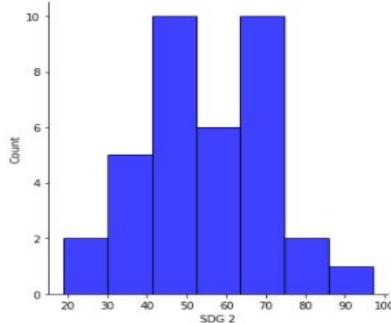
stat-0.99, p-0.9678865671157836914062500000000
yes normal distribution
```

```
In [284]: hist=df.hist('SDG 2',bins=30)
```



```
In [285]: import seaborn as sns
sns.distplot(df['SDG 2'],color='blue')

Out[285]: <seaborn.axisgrid.FacetGrid at 0x23976bf5d60>
```



```
In [286]: mean=np.mean(df['SDG 2'])
print(mean)
```

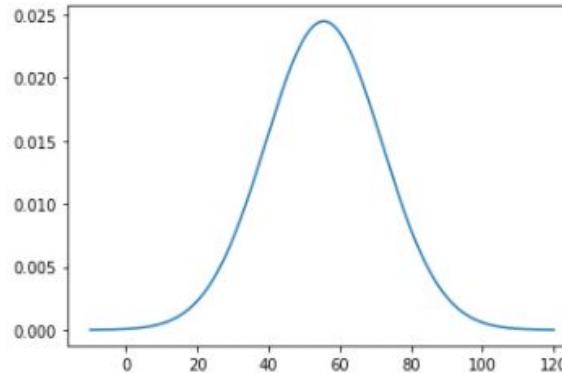
55.47222222222222

```
In [287]: std=np.std(df['SDG 2'])
print(std)
```

16.306927824140523

```
In [288]: import math
def fx(x,mu,sigma):
    return 1/(sigma*np.sqrt(2*np.pi))*np.exp(-np.power((x-mu)/sigma,2)/2)
```

```
In [289]: x=np.arange(-10,120,0.001)
plt.plot(x,fx(x,mean,std))
plt.show()
```



Two-Tail test (SDG 2)

Null Hypothesis : SDG 1 and SDG 2 are not related.

Alternate Hypothesis : There exists some relation between SDG 1 and SDG 2

Result : Null Hypothesis is accepted.

SDG 2

```
In [56]: ##sorting the data of SDG 2 wrt SDG 1 in descending order and taking a sample of 7 data
data3 = data2.sort_values('SDG 1',ascending=False).head(7)
data3
```

	Category	State/UT	SDG 1	SDG 2
22	State	Tamil Nadu	86.0	66.0
5	State	Goa	83.0	78.0
11	State	Kerala	83.0	80.0
24	State	Tripura	82.0	52.0
0	State	Andhra Pradesh	81.0	52.0
31	Union Territory	Delhi	81.0	63.0
8	State	Himachal Pradesh	80.0	52.0

```
In [58]: #applying t-test to the sample
from scipy.stats import ttest_1samp
mean01 = np.mean(data3['SDG 2'])
print(mean01)
tset, pval = ttest_1samp(data3['SDG 2'], mean1)
# print("p-values",pval)
if pval < 0.025 or pval>0.975:    # alpha value is 0.05 or 5%
    print(" we are rejecting null hypothesis")
else:
    print("we are accepting null hypothesis")
```

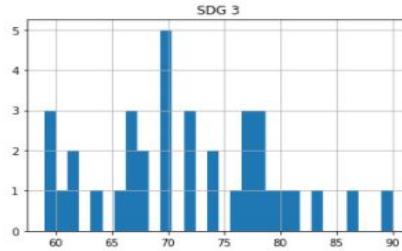
63.285714285714285

we are accepting null hypothesis

SDG 3: Hypothesis Testing (Good Health and Well-Being)

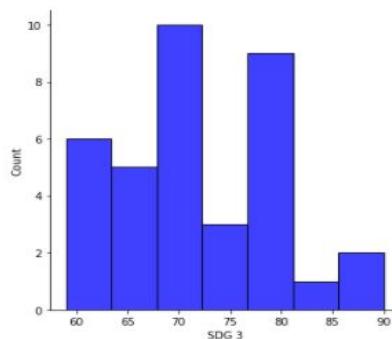
```
In [290]: from scipy.stats import shapiro  
DataToTest=df['SDG 3']  
stat, p=shapiro(DataToTest)  
print('stat=%.2f, p=%.30f' % (stat,p))  
  
if p > 0.05:  
    print("yes normal distribution")  
else:  
    print("not a normal distribution")  
  
stat=0.98, p=0.588117778301239013671875000000  
yes normal distribution
```

```
In [291]: hist=df.hist('SDG 3',bins=30)
```



```
In [292]: import seaborn as sns  
sns.distplot(df['SDG 3'],color='blue')
```

```
Out[292]: <seaborn.axisgrid.FacetGrid at 0x239093cbee0>
```



```
In [293]: mean=np.mean(df['SDG 3'])  
print(mean)
```

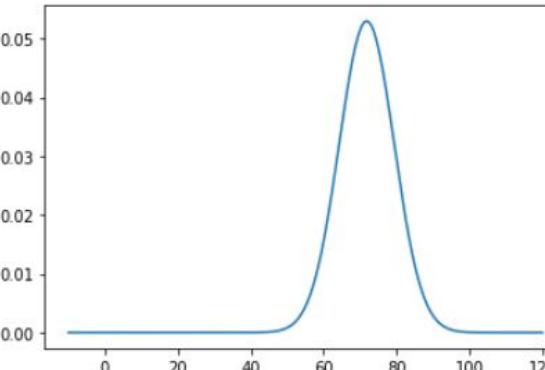
```
71.80555555555556
```

```
In [294]: std=np.std(df['SDG 3'])  
print(std)
```

```
7.538123272349411
```

```
In [295]: import math  
def fx(x,mu,sigma):  
    return 1/(sigma*np.sqrt(2*np.pi))*np.exp(-np.power((x-mu)/sigma,2)/2)
```

```
In [296]: x=np.arange(-10,120,0.001)  
plt.plot(x,fx(x,mean,std))  
plt.show()
```



Two-Tail test (SDG 3)

Null Hypothesis : SDG 2 and SDG 3 are not related.

Alternate Hypothesis : There exists some relation between SDG 2 and SDG 3

Result : Null Hypothesis is accepted.

SDG 3

```
In [41]: #sorting the data of SDG 3 wrt SDG 6 in descending order and taking a sample of 7 data
data3 = data2.sort_values('SDG 6',ascending=False).head(7)
data3
```

Out[41]:

	Category		State/UT	SDG 6	SDG 3
34	Union Territory		Lakshadweep	100.0	78.0
5	State		Goa	100.0	72.0
29	Union Territory		Chandigarh	99.0	74.0
23	State		Telangana	96.0	67.0
30	Union Territory	Dadra and Nagar Haveli and Daman and Diu		95.0	80.0
6	State		Gujarat	93.0	86.0
0	State		Andhra Pradesh	92.0	77.0

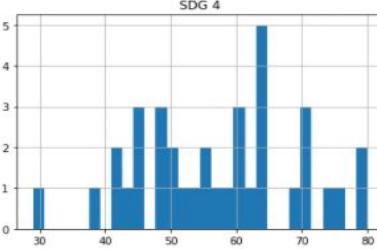
```
In [42]: #applying t-test to the sample
from scipy.stats import ttest_1samp
mean01 = np.mean(data3['SDG 3'])
print(mean01)
tset, pval = ttest_1samp(data3['SDG 3'], mean1)
# print("p-values",pval)
if pval < 0.025 or pval>0.975:    # alpha value is 0.05 or 5%
    print(" we are rejecting null hypothesis")
else:
    print("we are accepting null hypothesis")
```

76.28571428571429
we are accepting null hypothesis

SDG 4: Hypothesis Testing (Quality Education)

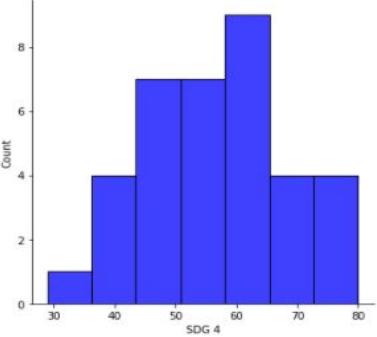
```
In [297]: from scipy.stats import shapiro  
DataToTest=df["SDG 4"]  
stat, p=shapiro(DataToTest)  
print('stat-%.2f, p-%.30f' % (stat,p))  
  
if p > 0.05:  
    print("yes normal distribution")  
else:  
    print("not a normal distribution")  
  
stat-0.99, p-0.9042539596557617187500000000000  
yes normal distribution
```

```
In [298]: hist=df.hist('SDG 4',bins=30)
```



```
In [299]: import seaborn as sns  
sns.displot(df['SDG 4'],color='blue')
```

```
Out[299]: <seaborn.axisgrid.FacetGrid at 0x2390cddefa0>
```



```
In [300]: mean=np.mean(df[ 'SDG 4'])  
print(mean)
```

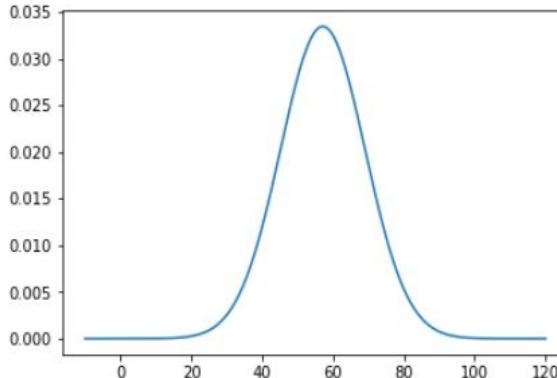
57.111111111111114

```
In [301]: std=np.std(df[ 'SDG 4'])  
print(std)
```

11.932164229923949

```
In [302]: import math  
def fx(x,mu,sigma):  
    return 1/(sigma*np.sqrt(2*np.pi))*np.exp(-np.power((x-mu)/sigma,2)/2)
```

```
In [303]: x=np.arange(-10,120,0.001)  
plt.plot(x,fx(x,mean,std))  
plt.show()
```



Two-Tail test (SDG 4)

Null Hypothesis: SDG 4 and SDG 5 are not related.

Alternate Hypothesis: There exists some relation between SDG 4 and SDG 5

Result: Null Hypothesis is accepted.

Hypothesis testing for SDG 4

```
In [4]: # sorting the entire data with respect to SDG5
data=data.sort_values('SDG 5')
d1=data['SDG 4'].head()
d1
```

```
Out[4]: 2    43
31   75
1    41
24   42
20   60
Name: SDG 4, dtype: int64
```

```
In [5]: # taking top 5 as the sample population and applying t-test
# Ho:Mean score of the sample=57.11
# H1:Mean score of the sample != 57.11
```

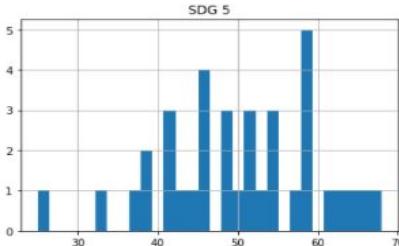
```
d1_mean = np.mean(data['SDG 4'])
print(d1_mean)
tset, pval = ttest_1samp(data['SDG 4'].head(), d1_mean)
print(pval)
if pval < 0.025 or pval>0.975:    # alpha value is 0.05 or 5%
    print(" we are rejecting null hypothesis")
else:
    print("we are accepting null hypothesis")
```

```
57.11111111111111
0.5035269517021073
we are accepting null hypothesis
```

SDG 5: Hypothesis Testing (Gender Equality)

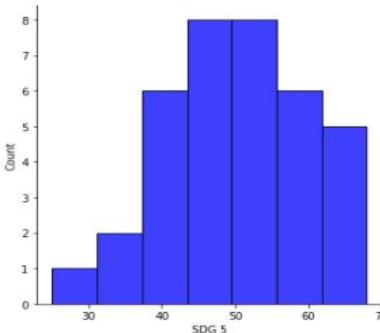
```
In [304]: from scipy.stats import shapiro  
DataToTest=df['SDG 5']  
stat, p=shapiro(DataToTest)  
print('stat=%2f, p=%.30f' % (stat,p))  
  
if p> 0.05:  
    print("yes normal distribution")  
else:  
    print("not a normal distribution")  
  
stat=0.98, p=0.885340034961700439453125000000  
yes normal distribution
```

```
In [305]: hist=df.hist('SDG 5',bins=30)
```



```
In [306]: import seaborn as sns  
sns.displot(df['SDG 5'],color='blue')
```

```
Out[306]: <seaborn.axisgrid.FacetGrid at 0x2390fa2af40>
```



```
In [307]: mean=np.mean(df['SDG 5'])  
print(mean)
```

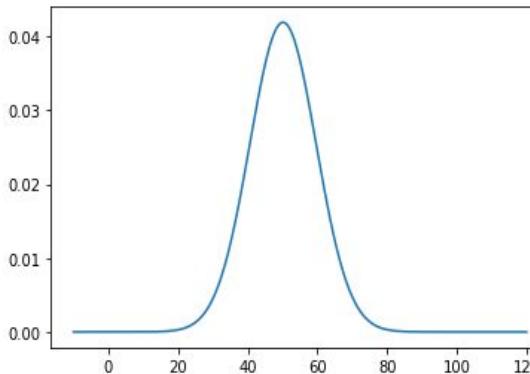
```
50.11111111111114
```

```
In [308]: std=np.std(df['SDG 5'])  
print(std)
```

```
9.527089382322673
```

```
In [309]: import math  
def fx(x,mu,sigma):  
    return 1/(sigma*np.sqrt(2*np.pi))*np.exp(-np.power((x-mu)/sigma,2)/2)
```

```
In [310]: x=np.arange(-10,120,0.001)  
plt.plot(x,fx(x,mean,std))  
plt.show()
```

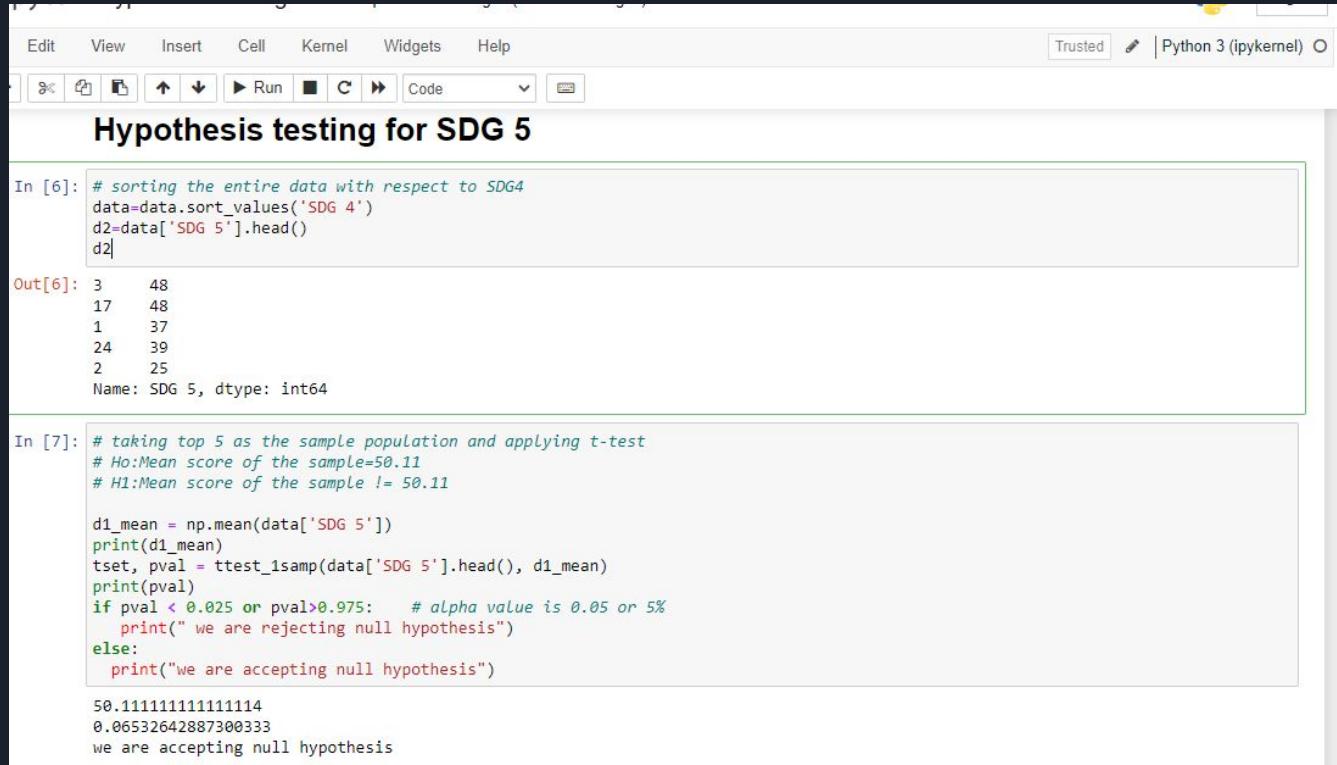


Two-Tail test (SDG 5)

Null Hypothesis : SDG 4 and SDG 5 are not related.

Alternate Hypothesis : There exists some relation between SDG 4 and SDG 5

Result : Null Hypothesis is accepted.



The screenshot shows a Jupyter Notebook interface with the following details:

- Toolbar:** Edit, View, Insert, Cell, Kernel, Widgets, Help, Trusted, Python 3 (ipykernel).
- Cell 6 (In [6]):** Contains Python code for sorting data by SDG 4 and displaying the top 5 rows of SDG 5.
- Output of Cell 6 (Out[6]):**

```
3    48
17   48
1    37
24   39
2    25
Name: SDG 5, dtype: int64
```
- Cell 7 (In [7]):** Contains Python code for performing a t-test on the top 5 values of SDG 5 against a mean of 50.11.
- Output of Cell 7 (Out[7]):**

```
50.11111111111111
0.06532642887300333
we are accepting null hypothesis
```

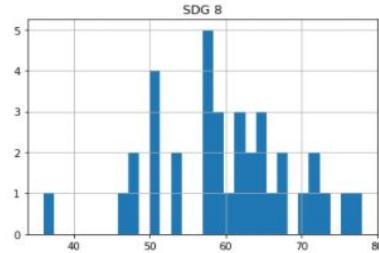
SDG 8: Hypothesis Testing (Decent Work and Economic Growth)

```
In [311]: from scipy.stats import shapiro
DataToTest=df["SDG 8"]
stat, p=shapiro(DataToTest)
print('stat=%.2f, p=%.30f' % (stat,p))

if p > 0.05:
    print("yes normal distribution")
else:
    print("not a normal distribution")

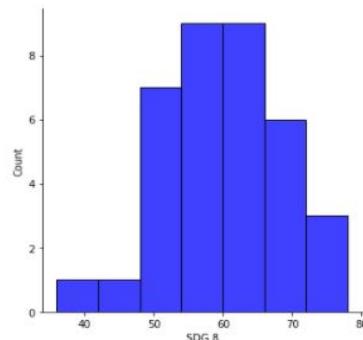
stat=0.98, p=0.887121379375457763671875000000
yes normal distribution
```

```
In [312]: hist=df.hist('SDG 8',bins=30)
```



```
In [313]: import seaborn as sns
sns.distplot(df['SDG 8'],color='blue')
```

```
Out[313]: <seaborn.axisgrid.FacetGrid at 0x2390fb9130>
```



```
In [314]: mean=np.mean(df['SDG 8'])
print(mean)
```

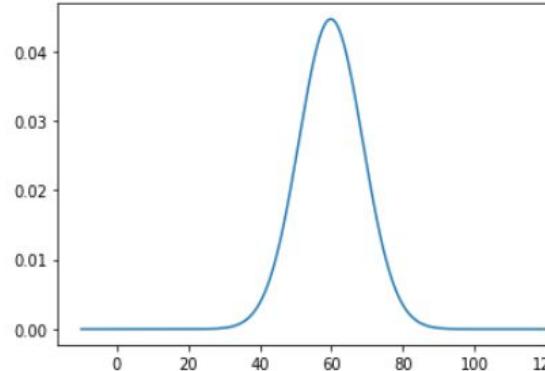
59.83333333333336

```
In [315]: std=np.std(df['SDG 8'])
print(std)
```

8.933395522171597

```
In [316]: import math
def fx(x,mu,sigma):
    return 1/(sigma*np.sqrt(2*np.pi))*np.exp(-np.power((x-mu)/sigma,2)/2)
```

```
In [317]: x=np.arange(-10,120,0.001)
plt.plot(x,fx(x,mean,std))
plt.show()
```



Two-Tail test (SDG 8)

Null Hypothesis : There is no relation between SDG 4 and SDG 8

Alternate Hypothesis : There exist some relation between SDG 4 and SDG 8

Result : Null Hypothesis is rejected.

The screenshot shows a Jupyter Notebook interface with the title "Hypothesis testing" and a subtitle "Last Checkpoint: 2 hours ago (unsaved changes)". The notebook has a "Trusted" status and is using Python 3 (ipykernel). The code in cell [8] sorts data by SDG 4 and prints the top 5 rows. The code in cell [9] performs a t-test on the top 5 rows against a mean of 59.83, rejecting the null hypothesis.

```
In [8]: # sorting the entire data with respect to SDG 4
data=data.sort_values('SDG 4')
d3=data['SDG 8'].head()
d3

Out[8]: 3    50
17   48
1    50
24   57
2    50
Name: SDG 8, dtype: int64

In [9]: # taking top 5 as the sample population and applying t-test
# Ho:Mean score of the sample=59.83
# H1:Mean score of the sample != 59.83

d1_mean = np.mean(data['SDG 8'])
print(d1_mean)
tset, pval = ttest_1samp(data['SDG 8'].head(), d1_mean)
print(pval)
if pval < 0.025 or pval>0.975 : # alpha value is 0.05 or 5%
    print(" we are rejecting null hypothesis")
else:
    print("we are accepting null hypothesis")

59.83333333333336
0.004676098152994367
we are rejecting null hypothesis
```

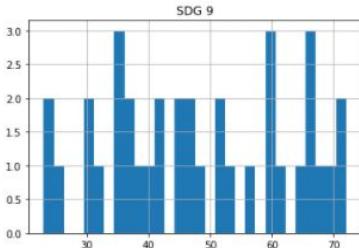
SDG 9: Hypothesis Testing (Industry, Innovation and Infrastructure)

```
In [318]: from scipy.stats import shapiro
DataToTest=df['SDG 9']
stat, p=shapiro(DataToTest)
print('stat-%.2f, p-%.30f' % (stat,p))

if p> 0.05:
    print("yes normal distribution")
else:
    print("not a normal distribution")

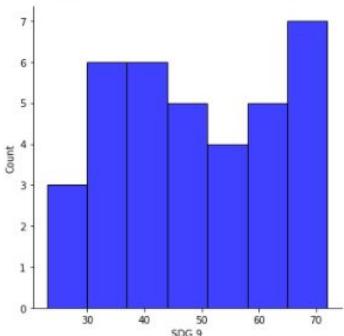
stat-0.95, p-0.146500617265701293945312500000
yes normal distribution
```

```
In [319]: hist=df.hist('SDG 9',bins=30)
```



```
In [320]: import seaborn as sns
sns.distplot(df['SDG 9'],color='blue')
```

```
Out[320]: <seaborn.axisgrid.FacetGrid at 0x2390fb3220>
```



```
In [321]: mean=np.mean(df['SDG 9'])
print(mean)
```

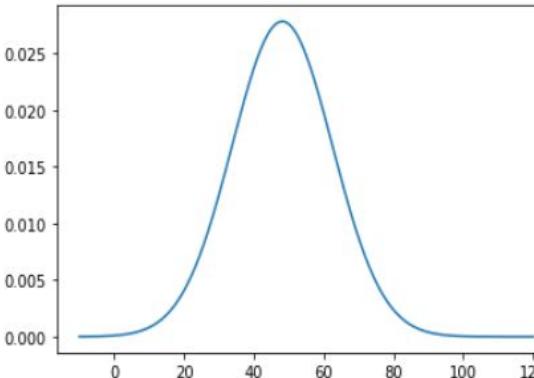
```
48.13888888888886
```

```
In [322]: std=np.std(df['SDG 9'])
print(std)
```

```
14.349124591528568
```

```
In [323]: import math
def fx(x,mu,sigma):
    return 1/(sigma*np.sqrt(2*np.pi))*np.exp(-np.power((x-mu)/sigma,2)/2)
```

```
In [324]: x=np.arange(-10,120,0.001)
plt.plot(x,fx(x,mean,std))
plt.show()
```



Two-Tail test (SDG 9)

Null Hypothesis : SDG 4 and SDG 9 are not related.

Alternate Hypothesis : There exists some relation between SDG 4 and SDG 9

Result : Null Hypothesis is rejected.



Jupyter Hypothesis testing Last Checkpoint: 2 hours ago (unsaved changes) Trusted Python 3 (ipykernel)

File Edit View Insert Cell Kernel Widgets Help

Hypothesis testing for SDG 9

```
In [10]: # sorting the entire data with respect to SDG4
data=data.sort_values('SDG 4')
d4=data['SDG 9'].head()
d4
```

```
Out[10]: 3    24
          17   30
          1    31
          24   35
          2    39
Name: SDG 9, dtype: int64
```

```
In [11]: # taking top 5 as the sample population and applying t-test
# H0:Mean score of the sample=48.14
# H1:Mean score of the sample != 48.14

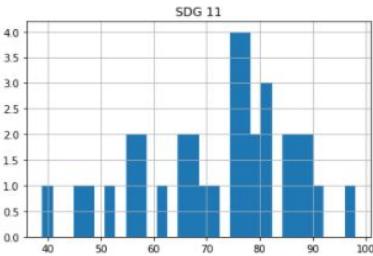
d1_mean = np.mean(data['SDG 9'])
print(d1_mean)
tset, pval = ttest_isamp(data['SDG 9'].head(), d1_mean)
print(pval)
if pval < 0.025 or pval>0.975 : # alpha value is 0.05 or 5%
    print(" we are rejecting null hypothesis")
else:
    print("we are accepting null hypothesis")

48.13888888888886
0.002908222405840632
we are rejecting null hypothesis
```

SDG 11: Hypothesis Testing (Sustainable Cities and communities)

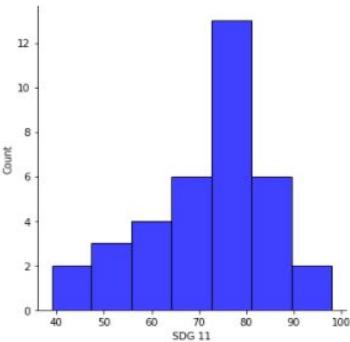
```
In [325]: from scipy.stats import shapiro  
DataToTest=df["SDG 11"]  
stat, p=shapiro(DataToTest)  
print('stat=%.2f, p=%.30f' % (stat,p))  
  
if p > 0.05:  
    print("yes normal distribution")  
else:  
    print("not a normal distribution")  
  
stat=0.96, p=0.197570458054542541503906250000  
yes normal distribution
```

```
In [326]: hist=df.hist('SDG 11',bins=30)
```



```
In [327]: import seaborn as sns  
sns.displot(df['SDG 11'],color='blue')
```

```
Out[327]: <seaborn.axisgrid.FacetGrid at 0x239091e1130>
```



```
In [328]: mean=np.mean(df['SDG 11'])  
print(mean)
```

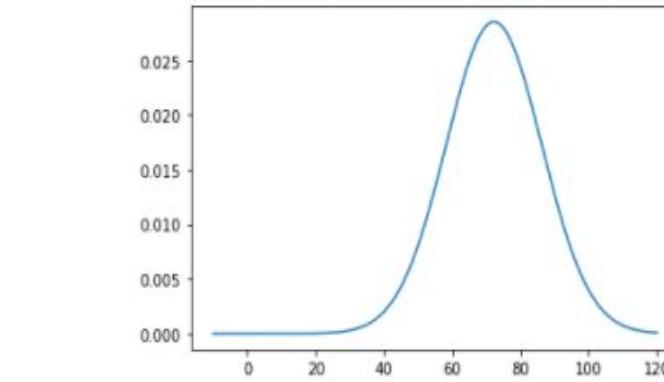
72.22222222222223

```
In [329]: std=np.std(df['SDG 11'])  
print(std)
```

13.980365420258176

```
In [330]: import math  
def fx(x,mu,sigma):  
    return 1/(sigma*np.sqrt(2*np.pi))*np.exp(-np.power((x-mu)/sigma,2)/2)
```

```
In [331]: x=np.arange(-10,120,0.001)  
plt.plot(x,fx(x,mean,std))  
plt.show()
```



Two-Tail test (SDG 11)

Null Hypothesis: SDG 11 and SDG 12 are not related.

Alternate Hypothesis : There exists some relation between SDG 11 and SDG 12

Result : Null Hypothesis is accepted.

SDG 11

```
In [8]: #sorting the data of SDG 11 wrt SDG 12 in descending order and taking a sample of 7 data
data3 = data2.sort_values('SDG 12', ascending=False).head(7)
data3
```

Out[8]:

	Category	State/UT	SDG 11	SDG 12
24	State	Tripura	67.0	99.0
33	Union Territory	Ladakh	57.0	95.0
32	Union Territory	Jammu and Kashmir	57.0	95.0
17	State	Nagaland	48.0	91.0
10	State	Karnataka	78.0	89.0
14	State	Manipur	65.0	89.0
16	State	Mizoram	61.0	87.0

```
In [9]: #applying t-test to the sample
from scipy.stats import ttest_1samp
mean01 = np.mean(data3['SDG 11'])
print(mean01)
tset, pval = ttest_1samp(data3['SDG 11'], mean1)
# print("p-values",pval)
if pval < 0.025 or pval>0.975:    # alpha value is 0.05 or 5%
    print(" we are rejecting null hypothesis")
else:
    print("we are accepting null hypothesis")
```

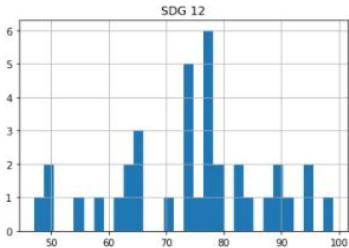
61.857142857142854

we are accepting null hypothesis

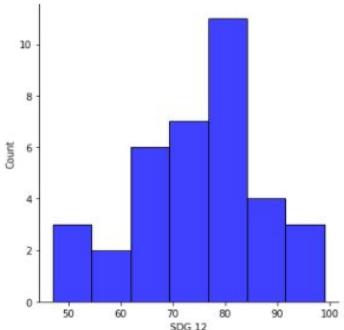
SDG 12: Hypothesis Testing (Responsible Production and Consumption)

```
In [332]: from scipy.stats import shapiro  
DataToTest=df["SDG 12"]  
stat, p=shapiro(DataToTest)  
print('stat=%.2f, p=%.30f' % (stat,p))  
  
if p > 0.05:  
    print("yes normal distribution")  
else:  
    print("not a normal distribution")  
  
stat=0.97, p=0.560765802860260009765625000000  
yes normal distribution
```

```
In [333]: hist=df.hist('SDG 12',bins=30)
```



```
In [334]: import seaborn as sns  
sns.distplot(df['SDG 12'],color='blue')  
Out[334]: <seaborn.axisgrid.FacetGrid at 0x2390e1a7490>
```



```
In [335]: mean=np.mean(df['SDG 12'])  
print(mean)
```

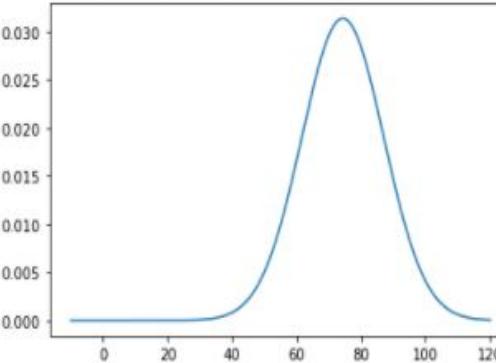
74.33333333333333

```
In [336]: std=np.std(df['SDG 12'])  
print(std)
```

12.71700523795686

```
In [337]: import math  
def fx(x,mu,sigma):  
    return 1/(sigma*np.sqrt(2*np.pi))*np.exp(-np.power((x-mu)/sigma,2)/2)
```

```
In [338]: x=np.arange(-10,120,0.001)  
plt.plot(x,fx(x,mean,std))  
plt.show()
```



Two-Tail test (SDG 12)

Null Hypothesis : SDG 8 and SDG 12 are not related.

Alternate Hypothesis : There exists some relation between SDG 8 and SDG 12

Result : Null Hypothesis is accepted.

```
"""SDG 12"""
"""Hypothesis Testing"""
#T-test b/w SDG 8 and 12
import numpy as np
from scipy.stats import ttest_1samp
df = pd.read_csv('Composite Score.csv')
print("T-test b/w SDG 8 and 12")
df=df.sort_values('SDG 8')#sorting the entire data with respect to SDG5
#print(df)
d1=df['SDG 12']#extracting the column 'SDG 12'
MEAN=np.mean(df['SDG 12'])#This is our sample mean
#print(df.head())
XBAR=np.mean(d1.head())
print("XBAR is", XBAR)
(tset, pval) = ttest_1samp(d1.head(), MEAN)
print("p-value is", pval)
if pval < 0.025:
    print("we are rejecting null hypothesis")
else:
    print("we are accepting null hypothesis")
```

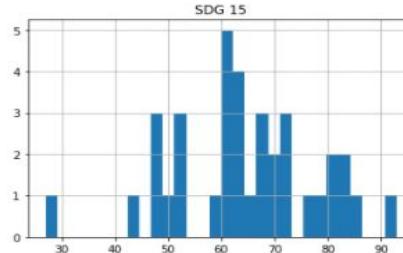
output2.txt

```
T-test b/w SDG 8 and 12
XBAR is 85.0
p-value is 0.060509908737843704
we are accepting null hypothesis
```

SDG 15: Hypothesis Testing (Life on Land)

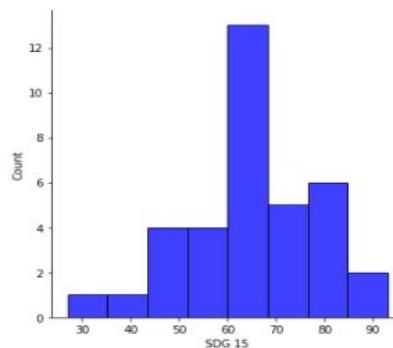
```
In [339]: from scipy.stats import shapiro  
DataToTest=df["SDG 15"]  
stat, p=shapiro(DataToTest)  
print('stat-%.2f, p-%.30f' % (stat,p))  
  
if p > 0.05:  
    print("yes normal distribution")  
else:  
    print("not a normal distribution")  
  
stat-0.98, p-0.743561863899230957031250000000  
yes normal distribution
```

```
In [340]: hist=df.hist('SDG 15',bins=30)
```



```
In [341]: import seaborn as sns  
sns.distplot(df['SDG 15'],color='blue')
```

```
Out[341]: <seaborn.axisgrid.FacetGrid at 0x2390e146700>
```



```
In [342]: mean=np.mean(df['SDG 15'])  
print(mean)
```

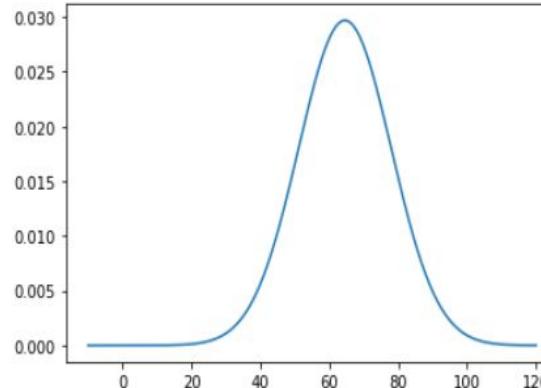
64.52777777777777

```
In [343]: std=np.std(df['SDG 15'])  
print(std)
```

13.444300963421703

```
In [344]: import math  
def fx(x,mu,sigma):  
    return 1/(sigma*np.sqrt(2*np.pi))*np.exp(-np.power((x-mu)/sigma,2)/2)
```

```
In [345]: x=np.arange(-10,120,0.001)  
plt.plot(x,fx(x,mean,std))  
plt.show()
```



Two-Tail test (SDG 15)

Null Hypothesis : SDG 13 and SDG 15 are not related.

Alternate Hypothesis : There exists some relation between SDG 13 and SDG 15

Result : Null Hypothesis is accepted.

SDG 15

```
In [8]: #sorting the data of SDG 15 wrt SDG 13 in descending order and taking a sample of 7 data
data3 = data2.sort_values('SDG 13', ascending=False).head(7)
data3
```

```
Out[8]:
```

	Category	State/UT	SDG 15	SDG 13
28	Union Territory	Andaman and Nicobar Islands	72.0	77.0
18	State	Odisha	83.0	70.0
11	State	Kerala	77.0	69.0
17	State	Nagaland	63.0	69.0
34	Union Territory	Lakshadweep	67.0	68.0
6	State	Gujarat	61.0	67.0
33	Union Territory	Ladakh	27.0	66.0

```
In [9]: #applying t-test to the sample
from scipy.stats import ttest_1samp
mean01 = np.mean(data3['SDG 15'])
print(mean01)
tset, pval = ttest_1samp(data3['SDG 15'], mean1)
# print("p-values",pval)
if pval < 0.025 or pval>0.975:    # alpha value is 0.05 or 5%
    print(" we are rejecting null hypothesis")
else:
    print("we are accepting null hypothesis")
```

```
64.28571428571429
we are accepting null hypothesis
```



THE END.