ES1101: COMPUTATIONAL DATA ANALYSIS

PROJECT REPORT

ON

INDIAN EDUCATION SYSTEM

INSTITUTE OF ENGINEERING AND TECHNOLOGY(IET), JAIPUR

SUBMITTED BY:

YEARAGRA PALIWAL (2020BTechCSE087)

CHIRAG KUMAR (2020BTechCSE022)

PRATEEK SHARMA (2020BTechCSE093)

VIKAS JANU (2020BTechCSE083)

SUBMITTED TO:

Dr. UMESH GUPTA

Dr. SONAL JAIN

ABSTRACT

Quality education is one of the most important sustainable development goal. It is set and targeted in a way so that everyone can get access and opportunity to get the quality education. Today there are lots of people who are suffering from the insufficient availability of education so that everyone can get access to quality education. It is very important for the development of the society that everyone living in the society gets the quality education. Any society can become prosperous and well developed by getting the well education. It becomes necessary that the implementation and the availability of the education facilities are reaching to which sections of the society because it becomes necessary that it should reach to all the sections of the society because only the one section of the society can not help in the development of the whole world.

TABLE OF CONTENTS

INTRODUCTION

The sustainable development goals are the set of 17 goals to cure many environmental issues and to cure the various problems that are prevailing in the society. They are set and advised in such a way so that the problems that are connected with everyone's life including all castes, all genders, which means that these goals are made and targeted in such a way so that each section of the society can live a happy and peaceful life. These goals are accepted but the whole world for the betterment of the planet. These goals are set by the United Nations General Assembly which consists of 193 representatives.

We will work on the data collection of the sustainable development goal 'Quality Education' and to collect the information about the development in this field in our report.

Goal 4 : QUALITY EDUCATION

As we all know that there are 17 SDGs and one of them is Quality Education. It is the most important factor for deciding the development of the society. The more the number of persons will get quality education the lesser time we will take for the development because for the development, quality education is the most important and the strongest pillar on which the amount and the speed of the development depends. It is the tool which enlightens the future of any individual and even of the society. It is the productive tool which helps in the overall development of the society. By acquiring quality education, every individual can acquire skills and knowledge of any field with the help of the quality education. Its importance is very much because it increases the chances of employment in the

society and the society where there is no unemployment, will definitely develop smoothly and speedly. There are different targets in this goal which we have to achieve before 2030.

Quality education has many targets by which the goal can be achieved. These targets are described below:

- By 2030, ensure that all girls and boys complete free, equitable and quality primary and secondary education leading to relevant and Goal-4 effective learning outcomes

- By 2030, ensure that all girls and boys have access to quality early childhood development, care and pre-primary education so that they are ready for primary education

- By 2030, ensure equal access for all women and men to affordable and quality technical, vocational and tertiary education, including university

- By 2030, substantially increase the number of youth and adults who have relevant skills, including technical and vocational skills, for employment, decent jobs and entrepreneurship.

- By 2030, eliminate gender disparities in education and ensure equal access to all levels of education and vocational training for the vulnerable, including persons with disabilities, indigenous peoples and children in vulnerable situations.

- By 2030, ensure that all youth and a substantial proportion of adults, both men and women, achieve literacy and numeracy.

- By 2030, ensure that all learners acquire the knowledge and skills needed to promote sustainable development, including, among others, through education for sustainable development and sustainable lifestyles, human rights, gender equality, promotion of a culture of peace and non-violence, global citizenship and appreciation of cultural diversity and of culture's contribution to sustainable development.

- Build and upgrade education facilities that are child, disability and gender sensitive and provide safe, nonviolent, inclusive and effective learning environments for all

- By 2020, substantially expand globally the number of scholarships available to developing countries, in particular least developed countries, small island developing States and African countries, for enrolment in higher education, including vocational training and information and communications technology, technical, engineering and scientific programmes, in developed countries and other developing countries

- By 2030, substantially increase the supply of qualified teachers, including through international cooperation for teacher training in developing countries, especially least developed countries and small island developing states.

For providing the education the most important role is of the teachers and even of the students because teachers are those who are providing the education and students are those who are getting the education and from them some will provide the education to the future generation. So, it becomes necessary that each student gets the quality education in a proper manner. To get the quality education properly and efficiently it becomes very necessary that we have to take care of teacher- student ratio. To get the education properly it becomes necessary that each teacher has low number of students to control, so that he/she can focus on every student and can help in the progress of each student. The quality of education will definitely impact the future of every individual.

STUDENT DROP OUT RATE OVER INDIA

India is nisus to guarantee an opportunity for admittance to schooling to any or every one of its electors with value and quality. while the training framework has completely developed complex since autonomy, the development has keep organization with its own arrangement of difficulties. It isn't sufficient to feel vain essentially by the enrolments. It is similarly essential to research and guarantee that most quantities of understudies can support and finish the entire instruction. Proficiency is identified with partner degree investigation of information sources and yields. A framework is considered practical if per unit of information, a great deal of yield is determined. Instruction framework has frightfully curious attributes. intensity of partner degree training framework is estimated by the state rates, paces of profits on instruction and dropout rates among others. Dropout rates are considered to be a great wastage inside the schooling system. Presently, even with higher informative associations constantly attempting to make, the amount of understudies choosing universities isn't sufficient. Set on the backbones of "Access, Quality, Equity, Affordability, and Accountability", the new National Education Policy (NEP 2020) pledges to address

the current troubles, for instance, powerless capability level, high dropouts and a shortfall of multidisciplinary approach. In any case, it doesn't obviously describe the accomplishments or propose financial obligations to execute the game plan.

While it is recognized as a significantly hopeful course of action that considers reorienting Indian preparing system to meet the 21st Century needs, it will require

a strong execution plan and comprehensive considered existing issues and differences that the pandemic has annoyed. One of the basic focal points of NEP 2020 is to raise Gross Enrolment Ratio (GER) to half by 2035. Could the visionary methodology achieve it? The proper reaction is in reality, yet through a strong execution plan and dynamic speculation of the large number of accomplices.

History has seen that while improving GER has been one of the essential purposes behind guidance courses of action previously, there is basically no qualification made on the ground. Recalling, the National Policy on Education proposed comparable informational opportunities especially for Scheduled Tribes (ST), Scheduled Caste (SC) and women, and to ensure social joining and lift GER. Did it totally achieve the point? The examination suggests the utilization was not incredibly powerful.

At this moment, even with higher informational establishments ceaselessly attempting to make, the amount of understudies taking on schools is 37.4 million with a GER of 26.3. As such, the target GER of half is authentically not a basic one. NEP 2020 means to achieve it by modifying the instructive arrangement and instructional strategy and change examinations. While it sounds critical on paper, the execution on ground will require uncommon measures.

Coming up next are two proposition that address (a) the help around which quality preparing turns explicitly Teachers, and (b) effective and lucky execution of NEP 2020:

Setting up National Educational Service: To meet the goal of half higher GER, India will require 3.3 million extra teachers in high level training by 2035 reliant

on an educator understudy extent of 1:15 which is a 235% development from the current availability of 1.4 million. This is a gigantic test contemplating that instructing isn't among the pined for purposes for living and subsequently not very many youths will pick it.

To build present assumptions, India should take practices from countries, for instance, Finland and South Korea where educating is among the most idealistic purposes for living. In any case, there is a necessity for establishment of National Educational Service (NES) that would be just probably as wanted as the IAS, IPS, IRS. The NES would be the wellspring for delivering enlightening pioneers who may start the master progression of teachers and preparing in all of the 935 districts in the country. Close by having content data and substance express instructional strategy, instructional classes should be modified to join current social and strategy driven issues. There should similarly be phenomenal tasks to make "tutoring prepared experts" and "investigation topic specialists". Constant master improvement of educators can be ensured through a public-private association.

For better occupation advancement, teachers should be given opportunities for headway and sidelong segment to the NES reliant on showing experience, execution, and journey for capable improvement openings.

There are numerous different prerequisites likewise to accomplish the Quality Education objective, that resemble drinking water office ought to be there in the schools since it is additionally the significant necessities that the understudies need to read for such a long time hours. The office of the latrines and the cleanliness

and sterilization ought to likewise be dealt with so the understudies wellbeing ought to likewise stay great and they can likewise contribute in accomplishing the objective.

ENROLLMENT IN SCHOOL RATIO OVER INDIA

The fourth (SDG 4) is to: "Assurance thorough and reasonable quality guidance and advance profound pull learning openings for all". SDG 4 is to be refined through the accomplishment of ten targets, which address the most broad and driven arrangement for overall guidance anytime tried. Among these, Target 4.5 is of uncommon interest during the current year's arrival of Education at a Glance as it revolves around esteem.

Value is at the center of the Sustainable Development Goals (SDGs), with Target 4.5 expressly hoping to "dispose of sex differences and assurance comparable induction to all levels of guidance and expert getting ready for the frail, joining individuals with ineptitudes, local social orders and children in powerless conditions."

Anyway today, an enormous number of the most thought little of bundles stay imperceptible in training information at the worldwide and public levels. This incorporates kids and adults with handicaps, those evacuated by battle, kid officials and laborers, similarly as meandering peoples.

As the position wellspring of SDG 4 information, the UIS has set up the International Observatory on Equity and Inclusion in Education to develop and develop the ways of thinking, rules and investigation expected to create an overall chronicle of data and rules to gauge value in training. This information is

key to help countries, UN accessories and normal society get-togethers to reasonably arrive at the most minimized gatherings.

Based at the UIS, the Observatory will work through relationship with research establishments and overall associations engaged with value related issues.

SURVEY OF LITERATURE

As per Beena Pandey, India has begun gaining ground by actualizing different plans in the field of giving Quality Education. The public authority of India has begun to make a few key projects and different arrangements for giving free and mandatory training to all the youngsters in the age gathering of 6 to 14 years under different plans like Sarva Shiksha Abhiyan(SSA) and the significant right that is Right To Education (RTE). The different plans like these have helped in expanding the enrolment paces of the understudies the nation over exceptionally in essential instruction. Training are the significant markers of a general public that assumes exceptionally urgent part in upgrading by and large financial improvement of the country. The advancement has been made in expanding the enrolment of the understudies till the essential level yet at the same time parcel of the work must be finished. In India the training field has seen an improvement regarding monstrous enrolment and the development of foundation, and in diminishing the sexual orientation hole in proficiency and even the general education rate.

She additionally wrote in her report that by having a low Pupil-Teacher Ratio, instructors can know the individual understudy better. Through low understudy instructor proportion educator's ability to recognize zones where the understudy might be needing help increment. Instruction are the significant pointers of a general public that assumes urgent part in upgrading by and large financial advancement of the country. The advancement has been made in expanding the enrolment of the understudies till the essential level yet at the same time part of the work must be finished. In India the schooling field has seen an improvement

as far as huge enrolment and the extension of foundation, and in diminishing the sex hole in proficiency and even the general education rate. By noticing some more subtleties in the schooling we can say that educator factor is the most remarkable determinant of student's scholastic accomplishment. As per her the possibility of solid execution decreases with expanding PTRs, and when PTR is just about as high as 40:1, schools have an under 2% possibility of turning in a solid execution. Based on outcomes and findings of the investigation she presumed that administration should take some specific arranging and the executives approaches which can help the infrastructural offices, everything being equal. As indicated by the new arrangement of rules inside the system of the Right to Education (RTE) Act, 2009 of India. RTE has set schooling in India inside a rights based system whereby the public authority is compelled by a solemn obligation to give instructive freedoms to all kids up to the Elementary level. The RTE orders a greatest PTR of 30:1 to be kept up in each school exclusively which implies the estimation of the understudy educator proportion ought to be 30.

The connection for the overview of writing is joined underneath in the reference.

As per Sateesh Gouda and T.V. Shekhar, International Institute for Population Science's (IIPS), Mumbai. The improvement of any general public and the country relies upon the nature of the instruction that individuals arrive.

The advancement of an individual and the advancement of a country rely upon the schooling. The creator attempted to clarify the different elements which are related with school dropouts in India. As indicated by the Survey of National Family Health Survey, just 75% of the kids which are in the age gathering of 6 to 16 years goes to class. It is gigantic measure of the nonconformists which isn't useful for any general public or country. The report additionally said that almost 14% of the

youngsters never goes to the school and almost 11% drop out the tutoring instruction for different reasons.

The creator has referenced in his report that the drop out pace of the kids likewise relies upon the factor that their folks work or they are jobless, on the grounds that the youngsters whose guardians are jobless then the chance of the exiting the kids additionally increments. The creator said that until and except if the financial status doesn't improve the objective of giving the quality training can not be improved appropriately. So as per the creator, the drop out proportion in India can be improved with connected at the hip of the financial turn of events.

The connection for the review of writing is joined underneath in the supplement.

As per Rahul Kundal, Ph.D. Exploration researcher, Department of Economics, University of Jammu Gross Enrolment Ratio of the advanced education in India is increasing with a decent rate. In India, consistently the quantity of understudies selecting the advanced education are expanding each year. Both the quantity of young men and young ladies are expanding for the enrolment a seemingly endless amount of time after year. Because of the expansion in the enrolment in the advanced education the Adult Literacy rate is likewise showing an incredible ascent. It has ascended from 61 out of 2001 to 69.3 in 2011. With the extraordinary ascent in the GER in advanced education, it is missing fairly behind than the GER in advanced education of the world.

The connection for the overview of writing is appended beneath in the reference.

OBJECTIVES

1) To check the Drop Out Rate in Schools of India.

1.1 To comprehend and look at the general drop out pace of school going youngsters up to the essential, auxiliary and higher optional degrees of school in India

2 To study and analyse the Pupil Teacher Ratio so that we can know the status of the different levels of schooling in India.

2.1 To consider the accessibility of the educators for understudies as state astute with the goal that we can check where the improvement should be possible for the better working and usage to accomplish the objective.

2.2 To consider the accessibility of the instructors for understudies in Union Territories likewise, with the goal that we can check where the improvement should be possible for the better working and usage to accomplish the objective and to rank them in like manner.

3. To check the situation in GER of Higher Education.

3.1 To check the situations in equal access to affordable technical, vocational and higher education in India.

3.2 To check the situations in different states of our Country.

4. To check whether the student enrolment in primary schools has increased or decreased between 2013 to 2015.

4.1 To find year-wise changes in gross enrolment rate.

4.2 To find state wise changes in gross enrolment rate.

4.3 To find top five states with highest enrolment rate and top states with lowest enrolment rate.

DATA COLLECTION

We are giving the data according to the objectives:

1) To check the Drop Out Rate in Schools of India.

Data of Student Drop Out From Schools over India.

States	Girls Drop Out Rate In	Boys Drop Out Rate In
Andhra Pradesh	99.72	99.69
Arunachal Pradesh	96.56	95.65
Assam	83.94	82.8
Bihar	90.05	89.16
Chhattisgarh	99.16	98.5
Goa	100	100
Gujarat	99.95	99.87
Haryana	99.6	99.49
Himachal Pradesh	99.82	99.7
Jammu And Kashmir	95	92.14
Jharkhand	96.75	96.49
Karnataka	99.59	99.3
Kerala	99.15	98.16
Madhya Pradesh	96.65	96.2
Maharashtra	99.41	99.21
Manipur	98.74	98.67
Meghalaya	84.29	86.4
Mizoram	99.27	97.96
Nagaland	99.89	98.69
Odisha	97.06	93.75
Punjab	99.83	99.6
Rajasthan	99.67	99.19
Sikkim	99.83	99.84
Tamil Nadu	99.9	99.73
Telangana	100	100
Tripura	99.86	99.69
Uttar Pradesh	99.8	99.74
Uttarakhand	97.18	97.43
West Bengal	98.29	97.7

Data For Girls Drop Out Of Schools In India.

1	State_UT	Primary Level Schools	Secondary level schools	Higher secondary level schools	All Schools
2	Andhra Pradesh	99.6	100	100	99.72
3	Arunachal Pradesh	94.41	0	100	96.56
4	Assam	86.39	78.04	82.95	83.94
5	Bihar	84.35	91.9	89.07	90.05
6	Chhattisgarh	99.16	86.61	100	99.16
7	Goa	100	100	100	100
8	Gujarat	99.95	99.85	99.72	99.95
9	Haryana	99.37	0	100	99.6
10	Himachal Pradesh	99.85	100	100	99.82
11	Jammu And Kashmir	91.66	93.94	0	95
12	Jharkhand	96.16	94.79	95.33	96.75
13	Karnataka	99.54	99.95	93.99	99.59
14	Kerala	98.54	98.68	98.46	99.15
15	Madhya Pradesh	96.2	93.53	100	96.65
16	Maharashtra	99.06	99.82	99.43	99.41
17	Manipur	98.27	92.5	100	98.74
18	Meghalaya	84.12	67.24	81.82	84.29
19	Mizoram	99.04	99.51	99.28	99.27
20	Nagaland	99.84	100	100	99.89
21	Odisha	97.44	93.88	92.83	97.06
22	Punjab	99.82	100	100	99.83
23	Rajasthan	99.4	100	0	99.67
24	Sikkim	99.84	0	0	99.83
25	Tamil Nadu	99.84	100	100	99.9
26	Telangana	100	0	100	100
27	Tripura	99.73	0	0	99.86
28	Uttar Pradesh	99.8	98.96	92.31	99.8
29	Uttarakhand	96.71	91.88	100	97.18
30	West Bengal	98.1	0	100	98.29

Data For Boys Drop Out Of Schools In India.

State_UT	Primary Level Schools	Secondary Level Schools	Higher Secondary Level Schools	All Schools
Andhra Pradesh	99.58	100	100	99.69
Arunachal Pradesh	93.55	100	100	95.65
Assam	85.83	73.55	80.21	82.8
Bihar	83.78	94.29	87.46	89.16
Chhattisgarh	98.49	97.67	100	98.5
Goa	100	0	100	100
Gujarat	99.86	100	99.72	99.87
Haryana	99.15	84.78	100	99.49
Himachal Pradesh	99.68	100	100	99.7
Jammu And Kashmir	87.9	99.26	100	92.14
Jharkhand	96.07	96.13	94.93	96.49
Karnataka	98.92	99.88	93.65	99.3
Kerala	96.88	99.54	100	98.16
Madhya Pradesh	95.72	96.94	100	96.2
Maharashtra	98.8	99.61	99.61	99.21
Manipur	98.37	100	100	98.67
Meghalaya	86.48	93.33	82.82	86.4
Mizoram	97.5	0	100	97.96
Nagaland	98.66	100	100	98.69
Odisha	93.66	0	88.4	93.75
Punjab	99.62	96.67	98.08	99.6
Rajasthan	98.97	97.86	0	99.19
Sikkim	99.86	0	0	99.84
Tamil Nadu	99.6	100	100	99.73
Telangana	100	100	100	100
Tripura	99.57	0	0	99.69
Uttar Pradesh	99.74	99.81	92.31	99.74
Uttarakhand	97.18	98.13	100	97.43
West Bengal	97.53	100	100	97.7

The above data is collected for the analysis of the first objective.

Now, we are showing the data for the second objective on the next page.

2 To study and analyse the Pupil Teacher Ratio so that we can know the status of the different levels of schooling in India.

Here is the data for the pupil teacher ratio of the year 2015-16 for the different levels of the schools till senior secondary education. It is showing the data for all the states and the union territories of India.

	U.T./States	Primary Level	Upper-Primary Level	Secondary Level	Senior secondary Level
0	A and N Island	8	6	14	16
1	Andhra Pradesh	21	16	20	71
2	Arunachal Pradesh	12	7	22	37
3	Assam	21	13	14	20
4	Bihar	36	24	66	39
5	Chandigarh	13	9	13	28
6	Chhattisgarh	20	17	33	21
7	D and N Haveli	17	13	30	30
8	Daman and Diu	26	14	17	13
9	Delhi	24	17	30	21
10	Goa	20	16	13	18
11	Gujarat	19	13	34	29
12	Haryana	20	13	15	15
13	Himachal Pradesh	12	10	18	14
14	Jammu And Kashmir	9	6	15	29
15	Jharkhand	27	19	62	78
16	Karnataka	19	13	16	30
17	Kerala	18	14	17	21
18	Lakshadweep	7	7	7	12
19	Madhya Pradesh	20	18	39	38
20	Maharashtra	24	17	23	44
21	Manipur	12	8	12	19
22	Meghalaya	21	13	12	21

23	Mizoram	14	6	9	15
24	Nagaland	10	6	15	21
25	Odisha	17	14	20	45
26	Puducherry	14	9	11	17
27	Punjab	18	12	16	26
28	Rajasthan	17	10	21	32
29	Sikkim	5	5	17	15
30	Tamil Nadu	18	15	21	25
31	Telangana	23	15	22	47
32	Tripura	10	8	28	11
33	Uttar Pradesh	39	31	56	97
34	Uttarakhand	18	17	16	25
35	West Bengal	25	27	39	57

Here is the data for the pupil teacher ratio of the year 2015-16 for the different levels of the schools till senior secondary education. It is showing the data for all the states of India.

	States	Primary Level	Upper-Primary Level	Secondary Level	Senior Secondary level
0	Andhra Pradesh	21	16	20	71
1	Arunachal Pradesh	12	7	22	37
2	Assam	21	13	14	20
3	Bihar	36	24	66	39
4	Chhattisgarh	20	17	33	21
5	Goa	20	16	13	18
6	Gujarat	19	13	34	29
7	Haryana	20	13	15	15
8	Himachal Pradesh	12	10	18	14
9	Jammu And Kashmir	9	6	15	29
10	Jharkhand	27	19	62	78
11	Karnataka	19	13	16	30
12	Kerala	18	14	17	21
13	Madhya Pradesh	20	18	39	38
14	Maharashtra	24	17	23	44
15	Manipur	12	8	12	19
16	Meghalaya	21	13	12	21
17	Mizoram	14	6	9	15
18	Nagaland	10	6	15	21
19	Odisha	17	14	20	45
20	Punjab	18	12	16	26
21	Rajasthan	17	10	21	32

22	Sikkim	5	5	17	15
23	Tamil Nadu	18	15	21	25
24	Telangana	23	15	22	47
25	Tripura	10	8	28	11
26	Uttar Pradesh	39	31	56	97
27	Uttarakhand	18	17	16	25
28	West Bengal	25	27	39	57

Here is the data for the pupil teacher ratio of the year 2015-16 for the different levels of the schools till senior secondary education. It is showing the data for all the union territories of India.

	U.T.	Primary Level	Upper-Primary Level	Secondary Level	Senior secondary Level
0	A and N Island	8	6	14	16
1	Chandigarh	13	9	13	28
2	D and N Haveli	17	13	30	30
3	Daman and Diu	26	14	17	13
4	Delhi	24	17	30	21
5	Lakshadweep	7	7	7	12
6	Puducherry	14	9	11	17

So, here we have collected the data of the Pupil Teacher Ratio of the Indian states and Union territories together and separately also, so that we can analyse the situations accordingly and properly.

The above data is collected for the analysis of the second objective.

Now, we are showing the data for the next objective i.e third objective on the next page.

3. To check the situation in GER of Higher Education.

STATES/UTs	ALL CATEGORIES - MALE	ALL CATEGORIES - FEMALE	ALL CATEGORIES - TOTAL	SC - MALE	SC - FEMALE	SC – TOTAL	ST - MALE	ST - FEMALE	ST - TOTAL
Andaman and Nicobar Islands	9.6	13.4	11.4	0	0	0	4.4	8.8	6.6
Andhra Pradesh	32.1	24.4	28.4	29.8	21.9	25.9	31.5	16.8	23.8
Arunachal Pradesh	33.6	19.5	26.9	100	66	100	24.3	16.1	20.1
Assam	13.3	13.4	13.4	10.4	12.2	11.2	13.5	13.2	13.4
Bihar	11.8	9.1	10.5	7	4.7	5.9	20.9	19.7	20.3
Chandigarh	42.2	40.4	41.4	12.1	20.8	16	0	0	0
Chhatisgarh	15.8	11.4	13.6	12.2	10.3	11.3	5.9	4.9	5.4
Dadra and Nagar Haveli	3.5	3.9	3.6	2.3	4.4	3.3	1.9	1.1	1.5
Daman and Diu	2.7	5.6	3.5	11.1	23	16.1	10.7	5.3	8.4
Delhi	35	29.8	32.5	11.7	8.7	10.2	0	0	0
Goa	31	35.8	33.2	26.4	24.1	25.2	100	100	100
Gujarat	23.5	18.8	21.3	19.8	15.6	17.8	11.3	9.3	10.3
Haryana	27.2	20.7	24.1	16.5	11.5	14.2	0	0	0
Himachal Pradesh	25.9	26	26	13.5	13.9	13.7	29.6	29.7	29.6
Jammu and Kashmir	17	16.7	16.8	6.4	6.4	6.4	5.6	2.4	4.1
Jharkhand	8.8	7.5	8.1	5.7	4	4.9	5.9	6.5	6.2
Karnataka	26.6	24.3	25.5	21.2	18.3	19.8	19.5	16.3	18
Kerala	18.8	25.1	21.9	10.3	18.1	14.2	17.4	19.1	18.3
Lakshadweep	0	0	0	0	0	0	0	0	0
Madhya Pradesh	15.1	11.9	13.6	12.3	8.3	10.4	6.2	3.6	4.9
Maharashtra	30.7	24.2	27.6	32.7	24.9	28.9	16.2	8.7	12.3
Manipur	38.5	33.3	35.9	100	100	100	44.6	34.4	39.5
Meghalaya	15.3	19.8	17.5	35.3	32.1	33.9	12.9	17.2	15.1
Mizoram	22	21.1	21.6	100	100	100	21.4	19.5	20.4
Nagaland	25.8	16.9	21.5	0	0	0	15.9	15.6	15.8
Odisha	18.1	14.1	16.1	10.7	8.2	9.5	6.8	5.2	6
Puducherry	32.6	29.9	31.2	28.9	23.2	26	0	0	0

Punjab	23.6	14.5	19.4	8.1	6.4	7.3	0	0	0
Rajasthan	20.9	15.2	18.2	13.5	8.5	11.2	15.7	10.3	13
Sikkim	26	22.2	24.2	12.4	11.3	11.8	16.4	22.2	19.3
Tamil Nadu	36.5	29.1	32.9	23.4	19.9	21.7	30.6	17.5	23.7
Tripura	16	11	13.6	11.7	8.4	10.1	7.9	4.9	6.3
Uttar Pradesh	15.2	17.4	16.3	11.2	12.4	11.7	100	100	100
Uttrakhand	26.1	29.6	27.8	17.5	18	17.7	42.2	46.3	44.2
West Bengal	13.8	10.9	12.4	7.8	5.8	6.9	7.3	4.8	6

4. To check whether the student enrolment in primary schools has increased or decreased between 2013 to 2015.

	states	primary_2013-14	primary_2014-15	primary_2015-16
0	Andhra Pradesh	96.74	88.21	84.48
1	Arunachal Pradesh	128.46	128.13	126.76
2	Assam	113.43	114.96	106.11
3	Bihar	97.96	101.09	107.67
4	Chhattisgarh	103.99	103.08	100.02
5	Delhi	110.67	111.75	110.71
6	Goa	105.14	103.97	102.57
7	Gujarat	101.13	98.72	97.24
8	Haryana	98.39	97.57	91.41
9	Himachal Pradesh	100.49	99.43	98.80
10	Jammu And Kashmir	84.90	85.97	85.98
11	Jharkhand	110.23	108.40	109.22
12	Karnataka	100.96	101.86	102.98
13	Kerala	95.42	95.11	95.44
14	Madhya Pradesh	111.49	101.11	94.47
15	Maharashtra	99.81	98.95	97.74
16	Manipur	149.15	134.37	130.85
17	Meghalaya	135.35	138.40	140.90
18	Mizoram	125.96	122.66	122.99
19	Nagaland	118.78	100.57	99.50
20	Odisha	105.84	105.53	103.73
21	Punjab	105.61	105.11	101.70
22	Rajasthan	101.53	98.64	100.43
23	Sikkim	124.42	112.57	102.87
24	Tamil Nadu	102.56	103.11	103.89
25	Tripura	113.31	103.57	103.02
26	Uttar Pradesh	96.41	109.98	107.96
27	Uttaranchal	100.60	95.00	92.15
28	West Bengal	104.00	100.54	99.29

METHODOLOGY

We have used the following methods to solve many numerical problems and to get a detailed analysis of the data. We have used many methods like Descriptive Statistics, Building Hypotheses, Statistical Inferences and python code for drawing graphs and all. Methods Used in Descriptive statistics are:

1)MEAN

The mean is the average of the data or the numbers. Its computation is done by adding up all the numbers and then divide by how many numbers there are. In simple way, it is the sum divided by the count.

$$\overline{x} = \frac{\sum_{i=1}^{N} x_i}{N}$$

2.MEDIAN:

The median is found by ordering the set from lowest to highest and then finding the exact middle value.

$$\text{Med}(X) = \begin{cases} \frac{X[\frac{n}{2}]+X[\frac{n+1}{2}]}{2} & \text{if n is even} \\ X[\frac{n+1}{2}] & \text{if n is odd} \end{cases}$$

X = ordered list of values in data set

n = number of values in data set

3).MODE:

The mode of any data set is defined as the value that is arriving or we can say appearing the most number of times.

4) RANGE:

Range is defined as the difference in the highest value and the lowest value

$$\text{Range}(X) = \text{Max}(X) - \text{Min}(X)$$

5.VARIANCE:

Variance is defined as the square of the standard deviation or we can simply say that the standard deviation is the square root of the variance. It is basically a measure of the variability in the data. The more the variance that we get, the more the variability we will get or in other words we can simply say that it measures how far a data set is spread out.

$$S^2 = \frac{\sum (x_i - \bar{x})^2}{n - 1}$$

S^2 = sample variance
x_i = the value of the one observation
$\bar{x}$ = the mean value of all observations
n = the number of observations

6) STANDARD DEVIATION:

It is the measure of the amount of variation of a set of values. The low value of the Standard deviation means the values are tending close to the mean of the data. In mathematical ways we can say that it is the square root of the variance.

$$\sigma = \sqrt{\frac{\sum(x_i - \mu)^2}{N}}$$

σ = population standard deviation
N = the size of the population
x_i = each value from the population
μ = the population mean

7.) CORRELATION

Correlation is a measure in statistics which measures the amount or extent to which the two data or lets say variables are linearly related. Basically it's a statistical tool which is used to find the relationship between two variables. It can be in negative as well as positive. It is considered in categories like strongly related or weakly related or moderately related. We have found correlation between different levels of the schooling i.e. primary, upper primary and some others.

The coefficient of the correlation which indicates the strength of the relationship between the two variables can be calculated by using the below mentioned formula.

Correlation Coefficient Formula

$$r = \frac{n(\Sigma xy) - (\Sigma x)(\Sigma y)}{\sqrt{[n\Sigma x^2 - (\Sigma x)^2][n\Sigma y^2 - (\Sigma y)^2]}}$$

Now in the above mentioned formula the variables are :

r = the coefficient of the correlation of the linear relationship between x and y.

x(i) = the values of x in the given sample.

X = the mean of the sum of values of the x

Y(i) = The values of y in the given sample.

Y = the mean of the sum of values of the y

Example :

Lets say there are two variables, marks and the amount of hours spent on studying each day of different students of one subject only.

STUDENTS	HOURS	MARKS
AMAN	2	50
BHANU	4	62
DEV	7	71
MANAV	12	89

Now, we have to calculate the mean of the marks and the hours.

STUDENTS	HOURS	MARKS
AMAN	2	50
BHANU	4	62
DEV	7	71
MANAV	12	89
Mean	6.25	68

So, here we have calculated the mean of both the variables.

Now we have to subtract the mean of the hour from each value of the hour i.e. [x(i)-mean(x)]

Similarly, we have to subtract the mean of the hour from each value of the hour i.e. [x(i)-mean(x)]

Now the values that we get after this step are shown in the table :

STUDENTS	HOURS	MARKS	a= x(i)-x	b=y(i)-y
AMAN	2	50	-4.25	-15.75
BHANU	4	62	-2.25	-5.75
DEV	7	71	0.75	6.25
MANAV	12	89	5.75	15.25
Mean	6.25	68		

Now we have to make operations on a and b these are also shown in the table :

STUDENTS	HOURS	MARKS	a= x(i)-x	b=y(i)-y	Axb	a^2	b^2
AMAN	2	50	-4.25	-15.75	66.93	18.06	248
BHANU	4	62	-2.25	-5.75	12.93	5.06	33.06
DEV	7	71	0.75	6.25	4.68	0.56	39.06
MANAV	12	89	5.75	15.25	87.68	33.06	232.56
Mean	6.25	68	SUM		172.22	56.74	552.68

Now after putting the values in the formula, we are getting r as :

r = 0.972

The above coefficient suggests that the marks and hours have a very high positive correlation. This means that their respective marks tend to move in the same direction.

So similarly, in this way we have calculated the correlation between different levels of the education which we have shown below.

8.) HYPOTHESES

Hypotheses testing is the methodology in statistics by which one tests a claim which is made regarding a population. The methodology that we use to test a hypotheses test vary according to the type of data on which the claim is made.

In simple words we can say that hypotheses is a claim made for the given set of population regarding their mean. Hypotheses testing is a test which checks that whether the particular assumption is right or wrong. In simple words, we can say hypotheses testing is the test which checks whether the null hypotheses is accepted or rejected.

To test any hypotheses, we have to follow some steps which we are defining below:

STEP 1 → We have to first build the hypotheses and then we have to find the level of significance.

STEP 2 → Then we have to identify the type of test statistic that will be implemented on the type of the data on which we are working on.

STEP 3 → Then we have to find the criteria of rejection of null hypotheses.

STEP 4 → The we have to calculate the test statistic.

STEP 5 → Then we have to take the decision that the null hypotheses is accepted or rejected based on the calculation.

Building of Hypotheses and Statistical Inference:

The methodology for the hypotheses testing depends on the type of the data that we are working on. We can simply say that hypotheses testing is to check that is there any statistical evidence in favour of certain belief or not.
There are two types of hypotheses :
1. Null Hypotheses
2.Alternative Hypotheses.

Null hypotheses :-

In inferential statistics, t he null hypotheses is a claim regarding any data.

In other words, we can say that it is a basic assumption based on domain or problem knowledge.

Example : We take 2000 calories per day. i.e. Ho = 2000

And now through the data and mathematical calculations and methods we will check that the claim is accepted or not..

Alternative hypotheses :-

The alternative hypotheses is used in the hypotheses testing which is just opposite of the null hypotheses.

Example : Ho != 2000 Ha

STATISTICAL INFERENCE:

It is the process by which some inferences or conclusions about a population data are made through some certain statistics calculated from a sample of data drawn from that particular data on which one is working

Now, we will show the objective wise methodology objective wise:

1) To check the Drop Out Rate in Schools of India.

HYPOTHESES TESTING

Building of hypothesis

Null Hypothesis H0: $\mu1 \leq \mu2$ (Drop Out Rate from School In India)

Alternative Hypothesis: $\mu1 > \mu2$ (Drop Out Rate from School In India)

Where,

$\mu1$ is the mean of the Girls Drop Out Rate in school.

$\mu2$ is the mean of the Boys Drop Out Rate in school.

Level of significance:

$\alpha = 0.05$, level of significance

Sample size -n1 = 29, n2 = 29

Degree of freedom = n – 1 = 28

Test statistics:

We will use t-test as population variance is unknown and n < 30

Criteria of rejection:

We will reject the null hypothesis if (t<-t α, n1+n2-2) (Left Tailed)

Calculations:

$$t = \frac{\overline{x}_1 - \overline{x}_2}{\sqrt{(s^2(\frac{1}{n_1} + \frac{1}{n_2}))}}$$

$s_p^2 = \frac{s_1^2 + s_2^2}{2}$ if $n_1 = n_2$

Here,

μ**1 = 97.55**, where μ1 is the mean of the Girls Drop Out Rate in school.

μ**2 = 97.06**, where μ2 is the mean of the Boys Drop Out Rate in school.

n1 = n2 = 29

= 0

Variance 1 = 17.67

Variance 2 = 18.30

sp = (variance 1)*(variance 2)

t = 0.5188 (From Calculations)

t α,n1+n2-2= 1.7485

t = 0.5188

Since**, t < -1.7485**

Hence, **Null Hypothesis is Rejected.** Which signifies that the Drop Out Rate in school over India are not the same.

2 To study and analyse the Pupil Teacher Ratio so that we can know the status of the different levels of schooling in India.

CORRELATION

As already defined above ,

Correlation is a measure in statistics which measures the amount or extent to which the two data or lets say variables are linearly related.

We have found correlation between different levels of the schooling i.e. primary, upper primary and some others.

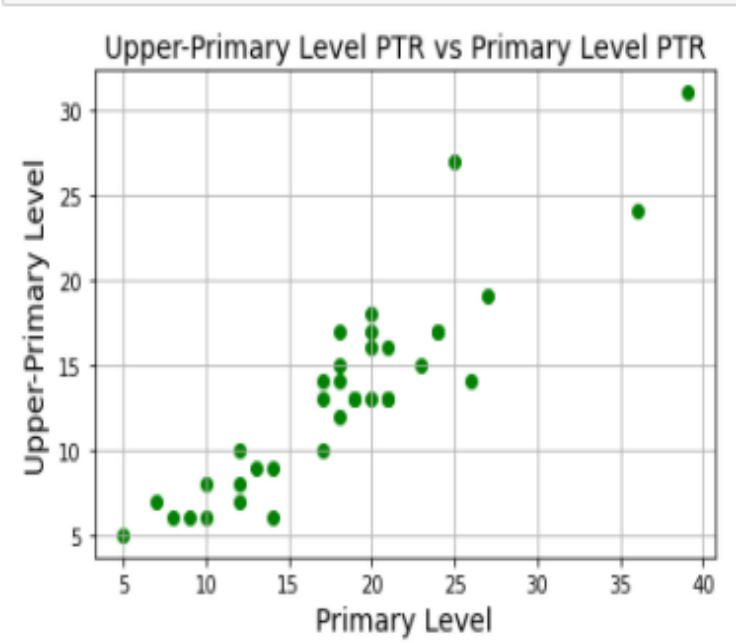
```
In [15]: correlation = d.corr(method = 'pearson')
         correlation
```

Out[15]:

	Primary Level	Upper-Primary Level	Secondary Level	Senior secondary Level
Primary Level	1.000000	0.908683	0.708292	0.637424
Upper-Primary Level	0.908683	1.000000	0.734287	0.692194
Secondary Level	0.708292	0.734287	1.000000	0.664914
Senior secondary Level	0.637424	0.692194	0.664914	1.000000

Correlation between primary and upper primary level

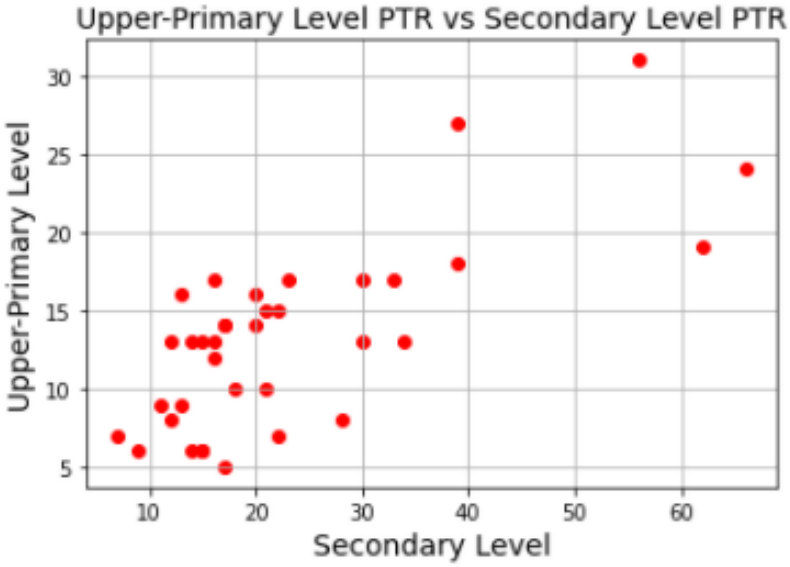
	Primary Level	Upper-Primary Level
Primary Level	1.000000	0.908683
Upper-Primary Level	0.908683	1.000000



As we can see in the above table the relation coefficient between primary and upper-primary i.e. 0.906 which means they are strongly correlated to each other. And we can also think it because it is one can study in upper primary level after completing the primary level.

Correlation between secondary and senior secondary level

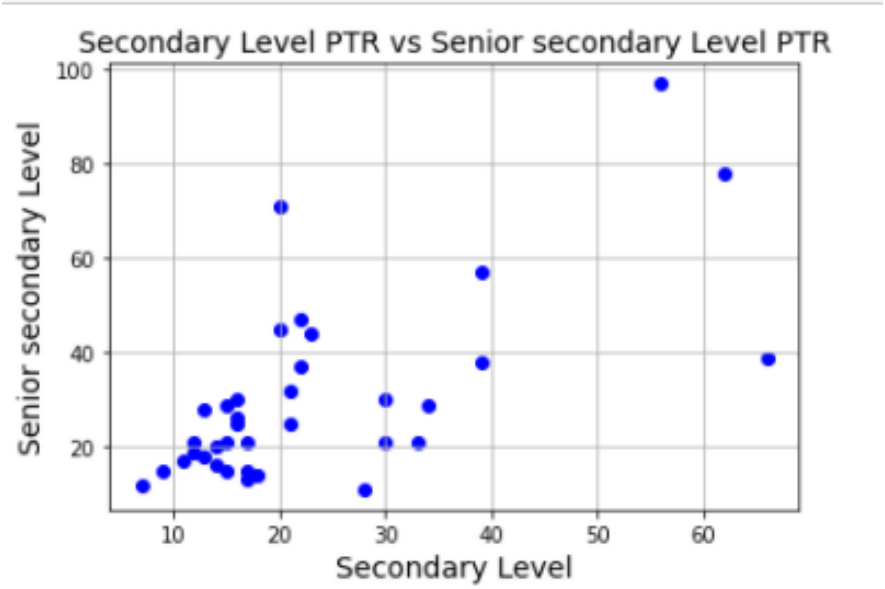
	Primary Level	Upper-Primary Level	Secondary Level	Senior secondary Level
Primary Level	1.000000	0.908683	0.708292	0.637424
Upper-Primary Level	0.908683	1.000000	0.734287	0.692194
Secondary Level	0.708292	0.734287	1.000000	0.664914
Senior secondary Level	0.637424	0.692194	0.664914	1.000000



As we can see in the marked area in the above table the relation coefficient between primary and upper-primary i.e. 0.664 which means they are moderately correlated to each other. The moderate correlation is because there are lots of students who study till primary or upper primary level only and later on they leave the studies due to various reasons.

Correlation between Upper-primary and secondary level

	Primary Level	Upper-Primary Level	Secondary Level	Senior secondary Level
Primary Level	1.000000	0.908683	0.708292	0.637424
Upper-Primary Level	0.908683	1.000000	0.734287	0.692194
Secondary Level	0.708292	0.734287	1.000000	0.664914
Senior secondary Level	0.637424	0.692194	0.664914	1.000000



As we can see in the marked area of the above table the relation coefficient between primary and upper-primary i.e., 0.734 which means they are moderately correlated to each other. The moderate correlation is because there are lots of students who study till primary or upper primary level only and later on they leave the studies due to various reasons.

Now, we are also showing the mean, maximum, minimum, variance and all the other things of our data the next page.

1.) Here we have shown the code and the output of the mean, mode, median and all other required calculations for the primary level.

```
import numpy as np
from scipy.stats import norm
import statistics

mean1 = statistics.mean(d['Primary Level'])
print("Mean of the Pupil-Teacher Ratio of primary level -> " , round(mean1,3))

std1 = d['Primary Level'].std()
print("Standard deviation of the Pupil-Teacher Ratio of primary level -> " , round(std1,3))

var1 = print("Variance of the Pupil-Teacher Ratio of primary level -> ", round((std1*std1),3))

modev1 = statistics.mode(d['Primary Level'])
print("Mode of the Pupil-Teacher Ratio of primary level -> " , modev1)

medv1 = statistics.median(d['Primary Level'])
print("Median of the Pupil-Teacher Ratio of primary level -> " , medv1)

minv1 = min(d['Primary Level'])
print("Minimum value of the Pupil-Teacher Ratio of primary level -> " , minv1)

maxv1 = max(d['Primary Level'])
print("Maximum value of the Pupil-Teacher Ratio of primary level -> " , maxv1)
```

```
Mean of the Pupil-Teacher Ratio of primary level ->  18.167
Standard deviation of the Pupil-Teacher Ratio of primary level ->  7.304
Variance of the Pupil-Teacher Ratio of primary level ->  53.343
Mode of the Pupil-Teacher Ratio of primary level ->  20
Median of the Pupil-Teacher Ratio of primary level ->  18.0
Minimum value of the Pupil-Teacher Ratio of primary level ->  5
Maximum value of the Pupil-Teacher Ratio of primary level ->  39
```

2.) Here we have shown the code and the output of the mean, mode, median and all other required calculations for the Upper - Primary level.

```
mean2 = statistics.mean(d['Upper-Primary Level'])
print("Mean of the Pupil-Teacher Ratio of Upper-primary level -> " , round(mean2,3))

std2 = d['Upper-Primary Level'].std()
print("Standard deviation of the Pupil-Teacher Ratio of Upper-primary level -> " , round(std2,3))

var2 = print("Variance of the Pupil-Teacher Ratio of Upper-primary level -> ", round((std2*std2),3))

modev2 = statistics.mode(d['Upper-Primary Level'])
print("Mode of the Pupil-Teacher Ratio of Upper-primary level -> " , modev2)

medv2 = statistics.median(d['Upper-Primary Level'])
print("Median of the Pupil-Teacher Ratio of Upper-primary level -> " , medv2)

minv2 = min(d['Upper-Primary Level'])
print("Minimum value of the Pupil-Teacher Ratio of Upper-primary level -> " , minv2)

maxv2 = max(d['Upper-Primary Level'])
print("Maximum value of the Pupil-Teacher Ratio of Upper-primary level -> " , maxv2)
```

```
Mean of the Pupil-Teacher Ratio of Upper-primary level ->  13.278
Standard deviation of the Pupil-Teacher Ratio of Upper-primary level ->  5.917
Variance of the Pupil-Teacher Ratio of Upper-primary level ->  35.006
Mode of the Pupil-Teacher Ratio of Upper-primary level ->  13
Median of the Pupil-Teacher Ratio of Upper-primary level ->  13.0
Minimum value of the Pupil-Teacher Ratio of Upper-primary level ->  5
Maximum value of the Pupil-Teacher Ratio of Upper-primary level ->  31
```

3.) Here we have shown the code and the output of the mean, mode, median and all other required calculations for the secondary level.

```
mean3 = statistics.mean(d['Secondary Level'])
print("Mean of the Pupil-Teacher Ratio of Secondary Level -> " , round(mean3,3))

std3 = d['Secondary Level'].std()
print("Standard deviation of the Pupil-Teacher Ratio of Secondary Level -> " , round(std3,3))

var3 = print("Variance of the Pupil-Teacher Ratio of Secondary Level -> ", round((std3*std3),3))

modev3 = statistics.mode(d['Secondary Level'])
print("Mode of the Pupil-Teacher Ratio of Secondary Level -> " , modev3)

medv3 = statistics.median(d['Secondary Level'])
print("Median of the Pupil-Teacher Ratio of Secondary Level -> " , medv3)

minv3 = min(d['Secondary Level'])
print("Minimum value of the Pupil-Teacher Ratio of Secondary Level -> " , minv3)

maxv3 = max(d['Secondary Level'])
print("Maximum value of the Pupil-Teacher Ratio of Secondary Level -> " , maxv3)
```

```
Mean of the Pupil-Teacher Ratio of Secondary Level ->  23.139
Standard deviation of the Pupil-Teacher Ratio of Secondary Level ->  14.157
Variance of the Pupil-Teacher Ratio of Secondary Level ->  200.409
Mode of the Pupil-Teacher Ratio of Secondary Level ->  17
Median of the Pupil-Teacher Ratio of Secondary Level ->  17.5
Minimum value of the Pupil-Teacher Ratio of Secondary Level ->  7
Maximum value of the Pupil-Teacher Ratio of Secondary Level ->  66
```

4.) Here we have shown the code and the output of the mean, mode, median and all other required calculations for the Senior -secondary level.

```
mean4 = statistics.mean(d['Senior secondary Level'])
print("Mean of the Pupil-Teacher Ratio of Senior secondary Level -> " , round(mean4,3))

std4 = d['Senior secondary Level'].std()
print("Standard deviation of the Pupil-Teacher Ratio of Senior secondary Level -> " , round(std4,3))

var4 = print("Variance of the Pupil-Teacher Ratio of Senior secondary Level -> ", round((std4*std4),3))

modev4 = statistics.mode(d['Senior secondary Level'])
print("Mode of the Pupil-Teacher Ratio of Senior secondary Level -> " , modev4)

medv4 = statistics.median(d['Senior secondary Level'])
print("Median of the Pupil-Teacher Ratio of Senior secondary Level -> " , medv4)

minv4 = min(d['Senior secondary Level'])
print("Minimum value of the Pupil-Teacher Ratio of Senior secondary Level -> " , minv4)

maxv4 = max(d['Senior secondary Level'])
print("Maximum value of the Pupil-Teacher Ratio of Senior secondary Level -> " , maxv4)
```

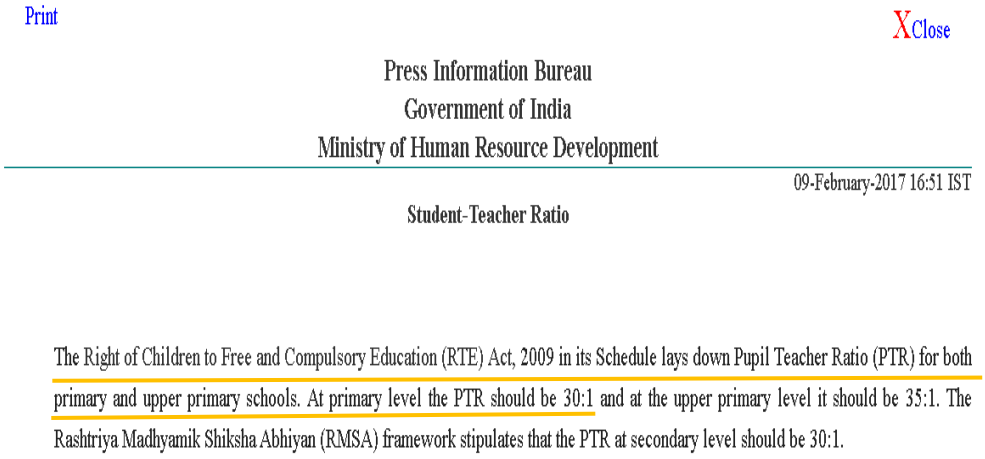
```
Mean of the Pupil-Teacher Ratio of Senior secondary Level ->  30.472
Standard deviation of the Pupil-Teacher Ratio of Senior secondary Level ->  19.431
Variance of the Pupil-Teacher Ratio of Senior secondary Level ->  377.571
Mode of the Pupil-Teacher Ratio of Senior secondary Level ->  21
Median of the Pupil-Teacher Ratio of Senior secondary Level ->  25.0
Minimum value of the Pupil-Teacher Ratio of Senior secondary Level ->  11
Maximum value of the Pupil-Teacher Ratio of Senior secondary Level ->  97
```

Test of Hypotheses:

1.) Problem Statement

- To verify the claim of the government of Pupil Teacher Ratio of the primary level is correct or not.

Student-Teacher Ratio (pib.gov.in)



Print

X Close

Press Information Bureau
Government of India
Ministry of Human Resource Development

09-February-2017 16:51 IST

Student-Teacher Ratio

The Right of Children to Free and Compulsory Education (RTE) Act, 2009 in its Schedule lays down Pupil Teacher Ratio (PTR) for both primary and upper primary schools. At primary level the PTR should be 30:1 and at the upper primary level it should be 35:1. The Rashtriya Madhyamik Shiksha Abhiyan (RMSA) framework stipulates that the PTR at secondary level should be 30:1.

Hypotheses :

Building of hypotheses

Null Hypotheses Ho : μo = 30

Alternative Hypotheses Ha: μo != 30

Where,

μo is the claimed mean of the primary level pupil teacher ratio in school.

Level of significance:

$\alpha = 0.05$, level of significance

Sample size n = 36

Degree of freedom = n – 1 = 35

Test statistics:

As in the above data n > 30 therefore we are using the z-test here.

Criteria of rejection:

$|z| > z_{\alpha/2}$, (Two Tailed), then the null hypotheses is rejected.

Calculations:

$$Z = \frac{\overline{x} - \mu_o}{s / \sqrt{n}}$$

It is the formula for one sampled z test, where:

x(bar) = mean of the sample

μo = hypotheses mean

s = Standard deviation.

n = 36

s = 7.303 through calculations.

z = - 9.723 (From Calculations)

Therefore $|z| = 9.723$

$z_{\alpha/2} = 1.96$

Hence, $|z| > z_{\alpha/2}$

Result of hypotheses :

Hence, **Hypotheses is Rejected by Two Tailed Test.**

```
import math
Ho = 30
n = 36
mean = 18.166
sd = 7.303
root = math.sqrt(n)
print("Root of number of samples" , root)
z = mean - Ho
zt = z/sd

zalpha = 1.96
ztest = zt*root
print("Value of z :-> " , round(abs(ztest),3))
if abs(ztest) > zalpha:
    print("NULL HYPOTHESES IS REJECTED BY TWO TAILED TEST")
else :
    print("NULL HYPOTHESES IS ACCEPTED")
```

```
Root of number of samples 6.0
Value of z :->  9.723
NULL HYPOTHESES IS REJECTED BY TWO TAILED TEST
```

2) Problem Statement

- To verify the claim of the government of Pupil Teacher Ratio of the Upper -primary level is correct or not.

Student-Teacher Ratio (pib.gov.in)

Print

Close

Press Information Bureau
Government of India
Ministry of Human Resource Development

09-February-2017 16:51 IST

Student-Teacher Ratio

The Right of Children to Free and Compulsory Education (RTE) Act, 2009 in its Schedule lays down Pupil Teacher Ratio (PTR) for both primary and upper primary schools. At primary level the PTR should be 30:1 and at the upper primary level it should be 35:1. The Rashtriya Madhyamik Shiksha Abhiyan (RMSA) framework stipulates that the PTR at secondary level should be 30:1.

Hypotheses :

Building of hypotheses

Null Hypotheses Ho : μo = 35

Alternative Hypotheses Ha: μo != 35

Where,

μo is the claimed mean of the upper - primary level pupil teacher ratio in school.

Level of significance:

α = 0.05, level of significance

Sample size n = 36

Degree of freedom = n – 1 = 35

Test statistics:

As in the above data n > 30 therefore we are using the z-test here.

Criteria of rejection:

$|z| > z_{\alpha/2}$, (Two Tailed), then the null hypotheses is rejected.

Calculations:

$$z = \frac{\bar{x} - \mu_o}{s / \sqrt{n}}$$

It is the formula for one tailed z test, where:

x(bar) = mean of the sample

μo = hypotheses mean

s = Standard deviation.

n = 36

s = 5.816 (through calculations)

z = -22.031 (From Calculations)

$z_{\alpha/2} = 1.96$

$|z| = 22.031$

Hence, $|z| > z_{\alpha/2}$

Result of hypotheses :

Hence, **Hypotheses is Rejected by Two Tailed Test.**

```
Ho = 35
n = 36
mean = 13.277
sd = 5.916
root = math.sqrt(n)
print("Root of number of samples" , root)
z = mean - Ho
zt = z/sd

zalpha = 1.96
ztest = zt*root
print("Value of z :-> " , round(abs(ztest),3))
if abs(ztest) > zalpha:
    print("NULL HYPOTHESES IS REJECTED BY TWO TAILED TEST")
else :
    print("NULL HYPOTHESES IS ACCEPTED")
```

```
Root of number of samples 6.0
Value of z :->  22.031
NULL HYPOTHESES IS REJECTED BY TWO TAILED TEST
```

Therefore, by testing both the hypotheses, we can say that both the claims of the government regarding the average pupil teacher ratio are not correct.

LINEAR ALGEBRA IMPLEMENTATION

Now we are applying power method to find the rank of the union territories according to the pupil teacher ratio. So, we are comparing the ranks of the union territories on the basis of their pupil teacher ratio. As we all know that the pupil teacher ratio should be low for the better implementation and development of the quality education. So, we will rank the union territories according on the basis of the low pupil teacher ratio. So for the implementation

of the power method we are using the python coding. The code and the matrix that is formed is shown below is shown below:

	U.T.	Primary Level	Upper-Primary Level	Secondary Level	Senior secondary Level	overall
0	A and N Island	8	6	14	16	44
1	Chandigarh	13	9	13	28	63
2	D and N Haveli	17	13	30	30	90
3	Daman and Diu	26	14	17	13	70
4	Delhi	24	17	30	21	92
5	Lakshadweep	7	7	7	12	33
6	Puducherry	14	9	11	17	51

So here we are calculating the overall pupil teacher ratio of each Union Territory i.e., if there are four levels of the education till school then we have calculated the pupil teacher ratio of total of that. And by that total we are comparing the union territories and giving them rank accordingly.

So, for the formation of the matrix, the methodology is

If Aij < Aji :

then 1 point

Else :

0 point

Then our matrix forms as :

	A and N Island	Chandigarh	D & N Haveli	Daman & Diu	Delhi	Lakshadweep	Puducherry
A and N Island	1	1	1	1	1	0	1
Chandigarh	0	1	1	1	1	0	0
D & N Haveli	0	0	1	0	1	0	0
Daman & Diu	0	0	1	1	1	0	0
Delhi	0	0	0	0	1	0	0
Lakshadweep	1	1	1	1	1	1	1
Puducherry	0	1	1	1	1	0	1

```
import numpy as np
l = []
a = np.array([[1,1,1,1,1,0,1],
                        [0,1,1,1,1,0,0],
                        [0,0,1,0,1,0,0],
                        [0,0,1,1,1,0,0],
                        [0,0,0,0,1,0,0],
                        [1,1,1,1,1,1,1],
                        [0,1,1,1,1,0,1]])
n = 7
# for initial guess vector
x = np.zeros((n))
print('Enter initial guess vector: ')
for i in range(n):
    x[i] = float(input( 'x['+str(i)+']='))
#Reading tolerable error
tolerable_error = float(input('Enter tolerable error: '))
#steps
max_iteration = int(input('Enter maximum number of steps: '))
# Power Method Implementation
lambda_old = 1.0
condition =  True
step = 1
while condition:
    x = np.matmul(a,x)
    # Finding new Eigen value and Eigen vector
    lambda_new = max(abs(x))
    x = x/lambda_new
    # Displaying Eigen value and Eigen Vector
    print('\nIteration %d' %(step))
    print('----------')
    print('Eigen Value = %0.4f' %(lambda_new))
    print('Eigen Vector: ')
    l.append(lambda_new)
    for i in range(n):
        print('%0.3f\t' % (x[i]))
    # Checking maximum iteration
    step = step + 1
    if step > max_iteration:
        print('Not convergent in given maximum iteration!')
        break
    # Calculating error
    error = abs(lambda_new - lambda_old)
    #print('errror='+ str(error))
    lambda_old = lambda_new
    condition = error > tolerable_error
print("Largest Dominant Eigen Value Possible: ",(max(l)))
print("Smallest Dominant Eigen Value Possible: ",(min(l)))
```

```
Enter initial guess vector:
x[0]=1
x[1]=1
x[2]=1
x[3]=1
x[4]=1
x[5]=1
x[6]=1
Enter tolerable error: 0.001
Enter maximum number of steps: 100

Iteration 1
----------
Eigen Value = 7.0000
Eigen Vector:
0.857
0.571
0.286
0.429
0.143
1.000
0.714

Iteration 2
----------
Eigen Value = 4.0000
Eigen Vector:
0.750
0.357
0.107
0.214
0.036
1.000
0.536

Iteration 3
----------
Eigen Value = 3.0000
Eigen Vector:
0.667
0.238
0.048
0.119
0.012
1.000
0.417
```

```
Iteration 4
----------
Eigen Value = 2.5000
Eigen Vector:
0.600
0.167
0.024
0.071
0.005
1.000
0.333

Iteration 5
----------
Eigen Value = 2.2000
Eigen Vector:
0.545
0.121
0.013
0.045
0.002
1.000
0.273

Iteration 6
----------
Eigen Value = 2.0000
Eigen Vector:
0.500
0.091
0.008
0.030
0.001
1.000
0.227

Iteration 7
----------
Eigen Value = 1.8571
Eigen Vector:
0.462
0.070
0.005
0.021
0.001
1.000
0.192
```

```
Iteration 8
----------
Eigen Value = 1.7500
Eigen Vector:
0.429
0.055
0.003
0.015
0.000
1.000
0.165

Iteration 9
----------
Eigen Value = 1.6667
Eigen Vector:
0.400
0.044
0.002
0.011
0.000
1.000
0.143

Iteration 10
----------
Eigen Value = 1.6000
Eigen Vector:
0.375
0.036
0.001
0.008
0.000
1.000
0.125

Iteration 11
----------
Eigen Value = 1.5455
Eigen Vector:
0.353
0.029
0.001
0.006
0.000
1.000
0.110
```

```
Iteration 12
----------
Eigen Value = 1.5000
Eigen Vector:
0.333
0.025
0.001
0.005
0.000
1.000
0.098

Iteration 13
----------
Eigen Value = 1.4615
Eigen Vector:
0.316
0.021
0.001
0.004
0.000
1.000
0.088

Iteration 14
----------
Eigen Value = 1.4286
Eigen Vector:
0.300
0.018
0.000
0.003
0.000
1.000
0.079

Iteration 15
----------
Eigen Value = 1.4000
Eigen Vector:
0.286
0.015
0.000
0.003
0.000
1.000
0.071
```

```
Iteration 16
----------
Eigen Value = 1.3750
Eigen Vector:
0.273
0.013
0.000
0.002
0.000
1.000
0.065

Iteration 17
----------
Eigen Value = 1.3529
Eigen Vector:
0.261
0.011
0.000
0.002
0.000
1.000
0.059

Iteration 18
----------
Eigen Value = 1.3333
Eigen Vector:
0.250
0.010
0.000
0.001
0.000
1.000
0.054

Iteration 19
----------
Eigen Value = 1.3158
Eigen Vector:
0.240
0.009
0.000
0.001
0.000
1.000
0.050
```

```
Iteration 20
----------
Eigen Value = 1.3000
Eigen Vector:
0.231
0.008
0.000
0.001
0.000
1.000
0.046

Iteration 21
----------
Eigen Value = 1.2857
Eigen Vector:
0.222
0.007
0.000
0.001
0.000
1.000
0.043

Iteration 22
----------
Eigen Value = 1.2727
Eigen Vector:
0.214
0.006
0.000
0.001
0.000
1.000
0.040

Iteration 23
----------
Eigen Value = 1.2609
Eigen Vector:
0.207
0.005
0.000
0.001
0.000
1.000
0.037
```

```
Iteration 24
----------
Eigen Value = 1.2500
Eigen Vector:
0.200
0.005
0.000
0.001
0.000
1.000
0.034

Iteration 25
----------
Eigen Value = 1.2400
Eigen Vector:
0.194
0.004
0.000
0.000
0.000
1.000
0.032

Iteration 26
----------
Eigen Value = 1.2308
Eigen Vector:
0.187
0.004
0.000
0.000
0.000
1.000
0.030

Iteration 27
----------
Eigen Value = 1.2222
Eigen Vector:
0.182
0.004
0.000
0.000
0.000
1.000
0.028
```

```
Iteration 28
----------
Eigen Value = 1.2143
Eigen Vector:
0.176
0.003
0.000
0.000
0.000
1.000
0.027

Iteration 29
----------
Eigen Value = 1.2069
Eigen Vector:
0.171
0.003
0.000
0.000
0.000
1.000
0.025

Iteration 30
----------
Eigen Value = 1.2000
Eigen Vector:
0.167
0.003
0.000
0.000
0.000
1.000
0.024

Iteration 31
----------
Eigen Value = 1.1935
Eigen Vector:
0.162
0.003
0.000
0.000
0.000
1.000
0.023
```

```
Iteration 32
----------
Eigen Value = 1.1875
Eigen Vector:
0.158
0.002
0.000
0.000
0.000
1.000
0.021

Iteration 33
----------
Eigen Value = 1.1818
Eigen Vector:
0.154
0.002
0.000
0.000
0.000
1.000
0.020

Iteration 34
----------
Eigen Value = 1.1765
Eigen Vector:
0.150
0.002
0.000
0.000
0.000
1.000
0.019

Iteration 35
----------
Eigen Value = 1.1714
Eigen Vector:
0.146
0.002
0.000
0.000
0.000
1.000
0.018
```

```
Iteration 36
----------
Eigen Value = 1.1667
Eigen Vector:
0.143
0.002
0.000
0.000
0.000
1.000
0.017

Iteration 37
----------
Eigen Value = 1.1622
Eigen Vector:
0.140
0.002
0.000
0.000
0.000
1.000
0.017

Iteration 38
----------
Eigen Value = 1.1579
Eigen Vector:
0.136
0.002
0.000
0.000
0.000
1.000
0.016

Iteration 39
----------
Eigen Value = 1.1538
Eigen Vector:
0.133
0.001
0.000
0.000
0.000
1.000
0.015
```

```
Iteration 40
----------
Eigen Value = 1.1500
Eigen Vector:
0.130
0.001
0.000
0.000
0.000
1.000
0.014

Iteration 41
----------
Eigen Value = 1.1463
Eigen Vector:
0.128
0.001
0.000
0.000
0.000
1.000
0.014

Iteration 42
----------
Eigen Value = 1.1429
Eigen Vector:
0.125
0.001
0.000
0.000
0.000
1.000
0.013

Iteration 43
----------
Eigen Value = 1.1395
Eigen Vector:
0.122
0.001
0.000
0.000
0.000
1.000
0.013
```

```
Iteration 44
----------
Eigen Value = 1.1364
Eigen Vector:
0.120
0.001
0.000
0.000
0.000
1.000
0.012

Iteration 45
----------
Eigen Value = 1.1333
Eigen Vector:
0.118
0.001
0.000
0.000
0.000
1.000
0.012

Iteration 46
----------
Eigen Value = 1.1304
Eigen Vector:
0.115
0.001
0.000
0.000
0.000
1.000
0.011

Iteration 47
----------
Eigen Value = 1.1277
Eigen Vector:
0.113
0.001
0.000
0.000
0.000
1.000
0.011
```

```
Iteration 48
----------
Eigen Value = 1.1250
Eigen Vector:
0.111
0.001
0.000
0.000
0.000
1.000
0.010

Iteration 49
----------
Eigen Value = 1.1224
Eigen Vector:
0.109
0.001
0.000
0.000
0.000
1.000
0.010

Iteration 50
----------
Eigen Value = 1.1200
Eigen Vector:
0.107
0.001
0.000
0.000
0.000
1.000
0.010

Iteration 51
----------
Eigen Value = 1.1176
Eigen Vector:
0.105
0.001
0.000
0.000
0.000
1.000
0.009
```

```
Iteration 52
----------
Eigen Value = 1.1154
Eigen Vector:
0.103
0.001
0.000
0.000
0.000
1.000
0.009

Iteration 53
----------
Eigen Value = 1.1132
Eigen Vector:
0.102
0.001
0.000
0.000
0.000
1.000
0.009

Iteration 54
----------
Eigen Value = 1.1111
Eigen Vector:
0.100
0.001
0.000
0.000
0.000
1.000
0.008

Iteration 55
----------
Eigen Value = 1.1091
Eigen Vector:
0.098
0.001
0.000
0.000
0.000
1.000
0.008
```

```
Iteration 56
----------
Eigen Value = 1.1071
Eigen Vector:
0.097
0.001
0.000
0.000
0.000
1.000
0.008

Iteration 57
----------
Eigen Value = 1.1053
Eigen Vector:
0.095
0.001
0.000
0.000
0.000
1.000
0.008

Iteration 58
----------
Eigen Value = 1.1034
Eigen Vector:
0.094
0.000
0.000
0.000
0.000
1.000
0.007

Iteration 59
----------
Eigen Value = 1.1017
Eigen Vector:
0.092
0.000
0.000
0.000
0.000
1.000
0.007
```

```
Iteration 60
----------
Eigen Value = 1.1000
Eigen Vector:
0.091
0.000
0.000
0.000
0.000
1.000
0.007

Iteration 61
----------
Eigen Value = 1.0984
Eigen Vector:
0.090
0.000
0.000
0.000
0.000
1.000
0.007

Iteration 62
----------
Eigen Value = 1.0968
Eigen Vector:
0.088
0.000
0.000
0.000
0.000
1.000
0.007

Iteration 63
----------
Eigen Value = 1.0952
Eigen Vector:
0.087
0.000
0.000
0.000
0.000
1.000
0.006
```

```
Iteration 64
----------
Eigen Value = 1.0938
Eigen Vector:
0.086
0.000
0.000
0.000
0.000
1.000
0.006

Iteration 65
----------
Eigen Value = 1.0923
Eigen Vector:
0.085
0.000
0.000
0.000
0.000
1.000
0.006

Iteration 66
----------
Eigen Value = 1.0909
Eigen Vector:
0.083
0.000
0.000
0.000
0.000
1.000
0.006

Iteration 67
----------
Eigen Value = 1.0896
Eigen Vector:
0.082
0.000
0.000
0.000
0.000
1.000
0.006
```

```
Iteration 68
----------
Eigen Value = 1.0882
Eigen Vector:
0.081
0.000
0.000
0.000
0.000
1.000
0.006

Iteration 69
----------
Eigen Value = 1.0870
Eigen Vector:
0.080
0.000
0.000
0.000
0.000
1.000
0.005

Iteration 70
----------
Eigen Value = 1.0857
Eigen Vector:
0.079
0.000
0.000
0.000
0.000
1.000
0.005

Iteration 71
----------
Eigen Value = 1.0845
Eigen Vector:
0.078
0.000
0.000
0.000
0.000
1.000
0.005
```

```
Iteration 72
----------
Eigen Value = 1.0833
Eigen Vector:
0.077
0.000
0.000
0.000
0.000
1.000
0.005

Iteration 73
----------
Eigen Value = 1.0822
Eigen Vector:
0.076
0.000
0.000
0.000
0.000
1.000
0.005

Iteration 74
----------
Eigen Value = 1.0811
Eigen Vector:
0.075
0.000
0.000
0.000
0.000
1.000
0.005

Iteration 75
----------
Eigen Value = 1.0800
Eigen Vector:
0.074
0.000
0.000
0.000
0.000
1.000
0.005
```

```
Iteration 76
----------
Eigen Value = 1.0789
Eigen Vector:
0.073
0.000
0.000
0.000
0.000
1.000
0.005

Iteration 77
----------
Eigen Value = 1.0779
Eigen Vector:
0.072
0.000
0.000
0.000
0.000
1.000
0.004

Iteration 78
----------
Eigen Value = 1.0769
Eigen Vector:
0.071
0.000
0.000
0.000
0.000
1.000
0.004
Largest Dominant Eigen Value Possible:  7.0
Smallest Dominant Eigen Value Possible:  1.076923076923077
```

So by this we can rank the union territories with the help of eigen vectors we get through python code :

So we can say that the top 3 rank holders in the pupil teacher ratio are:

1. Lakshadweep
2. A and N Island
3.Puducherry

3. To check the situation in GER of Higher Education.

One-Sample z-Test

A one-example z-test is utilized to test whether a populace boundary is fundamentally unique in relation to some theorized esteem.

Here is the manner by which to utilize the test.

Characterize theories. The table underneath shows three arrangements of invalid and elective theories. Each says something about how the genuine populace mean μ is identified with some speculated esteem M. (In the table, the image ≠ implies " not equivalent to ".)

Set	Null theory	Alternative speculation	Number of tails
1	μ = M	μ ≠ M	2
2	μ > M	μ < M	1
3	μ < M	μ > M	1

Indicate importance level. Frequently, scientists pick importance levels equivalent to 0.01, 0.05, or 0.10; yet any incentive somewhere in the range of 0 and 1 can be utilized.

Register test measurement. The test measurement is a z-score (z) characterized by the accompanying condition.

$z = (x - M) / [\sigma/\sqrt{n}]$

where x is the noticed example mean, M is the theorized populace mean (from the invalid speculation), and σ is the standard deviation of the populace.

Process P-esteem. The P-esteem is the likelihood of noticing an example measurement as outrageous as the test measurement. Since the test measurement is a z-score, utilize the Normal Distribution Calculator to survey the likelihood related with the z-score.

Assess invalid theory. The assessment includes contrasting the P-esteem with the importance level, and dismissing the invalid speculation when the P-esteem is not exactly the importance level.

The one-example z-test can be utilized when the populace is regularly circulated, and the populace difference is known.

Claim1- Average Gross Enrolment Ratio (Higher Education) of all categories of India is equals to 19.4.

Building of hypothesis:-

Null Hypothesis H0: $\mu = 19.4$

Alternative Hypothesis: $\mu \neq 19.4$

Where,

μ is the Gross Enrolment Ratio (Higher Education) of all categories of India in year 2015-

16.

Level of significance –

$\alpha = 0.05$

Test statistics –

We will use z-test here.

Criteria of rejection –

We will reject the null hypothesis if |z |> zα/2 (Two Tailed)

Calculations -

$$z = \frac{\bar{x} - \mu}{\sigma / \sqrt{n}}$$

Here,

$\bar{x}$ Is mean of the sample.

σ is standard deviation.

n is sample size.

Z = 8.33 (Calculation by PYTHON)

Result and decision –

-z α= -1.96

In [15]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import statistics as sts
data=pd.read_csv('Gross Enrolment Ratio in Higher Education.DATA.csv',index_col=0)
a=list(data['ALL CATEGORIES - TOTAL'])
μ0 = 19.4
n = len(a)
print('Null Hypothesis H0: μ = 19.4')
print('Alternative Hypothesis: H0: μ ≠ 19.4')
print('Level of significance(α) = 0.05')
print('Sample size: n =',len(a))
x=round(sts.mean(a),2)
s=round(np.std(a),2)
print('mean of the sample=',x)
print('standard deviation=',s)
z=round((x - μ0 /(s/(n)**0.5)),2)
print('from statistical table,we get that: zα/2=1.96  and z=',z)
zα=1.96
if zα>0:
    zα=1.96
else:
    zα=-(zα)
if zα==z:
    print('hence,null hypothesis is accepted...')
else:
    print('hence,null hypothesis is rejected...')
```

```
Null Hypothesis H0: μ = 19.4
Alternative Hypothesis: H0: μ ≠ 19.4
Level of significance(α) = 0.05
Sample size: n = 35
mean of the sample= 20.34
standard deviation= 9.56
from statistical table,we get that: zα/2=1.96  and z= 8.33
hence,null hypothesis is rejected...
```

Since z < -1.96 we cannot accept the null hypothesis which signifies that the Average Gross Enrolment Ratio (Higher Education) of all categories of India is equals to 19.4.

Claim2- Average Gross Enrolment Ratio (Higher Education) of SC category of India is equals to 13.5.

Building of hypothesis:-

Null Hypothesis H0: μ = 13.5

Alternative Hypothesis: μ ≠ 13.5

Where,

μ is the Gross Enrolment Ratio (Higher Education) of SC category of India in year 2015-16.

Level of significance –

α = 0.05

Test statistics –

We will use z-test here.

Criteria of rejection –

We will reject the null hypothesis if |z |> zα/2 (Two Tailed)

Calculations -

$$z = \frac{\bar{x} - \mu}{\sigma / \sqrt{n}}$$

Here,

$\bar{x}$ is mean of the sample.

σ is standard deviation.

n is sample size.

Z = 17.54 **(Calculation by PYTHON)**

Result and decision –

-z α= -1.96

Since z < -1.96 we cannot accept the null hypothesis which signifies that the Average Gross Enrolment Ratio (Higher Education) of SC category of India is equals to 13.5.

Python implementation:

In [16]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import statistics as sts
data=pd.read_csv('Gross Enrolment Ratio in Higher Education.DATA.csv',index_col=0)
a=list(data['SC - TOTAL'])
μ0 = 13.5
n = len(a)
print('Null Hypothesis H0: μ = 13.5')
print('Alternative Hypothesis: H0: μ ≠ 13.5')
print('Level of significance(α) = 0.05')
print('Sample size: n =',len(a))
x=round(sts.mean(a),2)
s=round(np.std(a),2)
print('mean of the sample=',x)
print('standard deviation=',s)
z=round((x - μ0 /(s/(n)**0.5)),2)
print('from statistical table,we get that: zα/2=1.96  and z=',z)
zα=1.96
if zα>0:
    zα=1.96
else:
    zα=-(zα)
if zα==z:
    print('hence,null hypothesis is accepted...')
else:
    print('hence,null hypothesis is rejected...')
```

```
Null Hypothesis H0: μ = 13.5
Alternative Hypothesis: H0: μ ≠ 13.5
Level of significance(α) = 0.05
Sample size: n = 35
mean of the sample= 20.66
standard deviation= 25.58
from statistical table,we get that: zα/2=1.96  and z= 17.54
hence,null hypothesis is rejected...
```

Claim3- Average Gross Enrolment Ratio (Higher Education) of ST category of India is equals to 13.5.

Building of hypothesis:-

Null Hypothesis H0: μ = 13.5

Alternative Hypothesis: μ ≠ 13.5

Where,

μ is the Gross Enrolment Ratio (Higher Education) of ST category of India in year 2015-16.

Level of significance –

α = 0.05

Test statistics –

We will use z-test here.

Criteria of rejection –

We will reject the null hypothesis if |z |> zα/2 (Two Tailed)

Calculations -

$$z = \frac{\bar{x} - \mu}{\sigma / \sqrt{n}}$$

Here,

- $\bar{x}$ is mean of the sample.
- σ is standard deviation.
- n is sample size.

Z = 14.62 **(Calculation by PYTHON)**

Result and decision –

-z α= -1.96

Since z < -1.96 we cannot accept the null hypothesis which signifies that the Average Gross Enrolment Ratio (Higher Education) of ST category of India is equals to11.2.

In [17]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import statistics as sts
data=pd.read_csv('Gross Enrolment Ratio in Higher Education.DATA.csv',index_col=0)
a=list(data['ST - TOTAL'])
μ0 = 11.2
n = len(a)
print('Null Hypothesis H0: μ = 11.2')
print('Alternative Hypothesis: H0: μ ≠ 11.2')
print('Level of significance(α) = 0.05')
print('Sample size: n =',len(a))
x=round(sts.mean(a),2)
s=round(np.std(a),2)
print('mean of the sample=',x)
print('standard deviation=',s)
z=round((x - μ0 /(s/(n)**0.5)),2)
print('from statistical table,we get that: zα/2=1.96  and z=',z)
zα=1.96
if zα>0:
    zα=1.96
else:
    zα=-(zα)
if zα==z:
    print('hence,null hypothesis is accepted...')
else:
    print('hence,null hypothesis is rejected...')
```

```
Null Hypothesis H0: μ = 11.2
Alternative Hypothesis: H0: μ ≠ 11.2
Level of significance(α) = 0.05
Sample size: n = 35
mean of the sample= 17.5
standard deviation= 22.98
from statistical table,we get that: zα/2=1.96  and z= 14.62
hence,null hypothesis is rejected...
```

4. **To check whether the student enrolment in primary schools has increased or decreased between 2013 to 2015**

Claim : Gross Enrolment Rate in primary schools has increased from 2013 to 2015:

Null hypothesis: μ2- μ1<=0(gross enrolment rate has decreased)

Alternate hypothesis: μ2- μ1>0(gross enrolment rate has increased)

Here, μ1 is average gross enrolment rate in primary schools in 2013

μ2 is average gross enrolment rate in primary schools in 2015

level of significance(α)=0.05

sample size(n)=29 ; here n1=n2

test statistics:

$$t = \frac{\overline{x}_1 - \overline{x}_2}{\sqrt{(s^2(\frac{1}{n_1} + \frac{1}{n_2}))}}$$

$$s_p^2 = \frac{s_1^2 + s_2^2}{2} \text{ if } n_1 = n_2$$

Here,

X1= 108.370

X2= 104.68

S1=185.341

S2=158.165

t= 0.092(from calculations)

criteria of rejection:

we will reject the hypothesis if t>tα,n1+n2-2 (right tailed test)

result and discussion:

$t_{\alpha,n1+n2-2}=1.645$

$t= 0.092$

since $t< t_{\alpha,n1+n2-2}$, null hypothesis cannot be rejected which signifies that gross enrolment rate has been decreased from 2013 to 2015.

RESULT AND DISCUSSION

1) To check the Drop Out Rate in Schools of India.

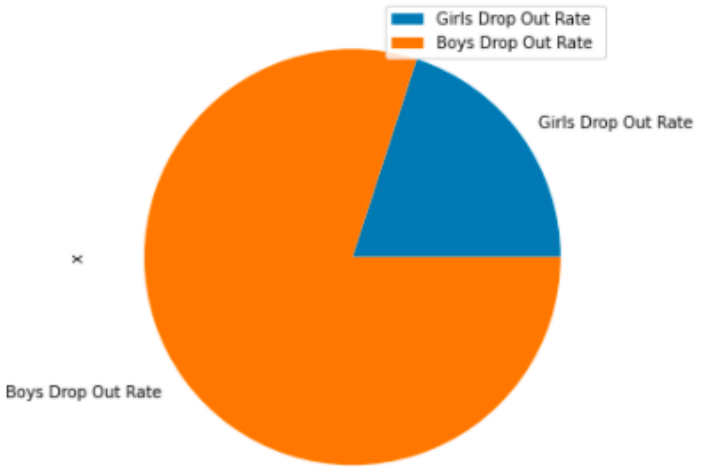
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import statistics as stat
data = pd.read_csv("Project Data.csv")
a = list(data["Girls Drop Out Rate In School"])
b = list(data["Boys Drop Out Rate In School"])

print("Null Hypothesis = H0: (u1-u2=0) (Left Tailed)")
print("Alternate hypothesis = Ha: (u1-u2>0) (Right tailed)")
print("Level of Significance: (a=0.05)")
print("Sample size: n1=29  n2=29")
n1 = n2 = 29
m1 = round(stat.mean(a),2)
m2 = round(stat.mean(b),2)
v1 = round(stat.variance(a),2)
v2 = round(stat.variance(b),2)
print("Mean of the Girls Drop Out Rate in school is: ",m1)
print("Mean of the Boys Drop Out Rate in school is: ",m2)
print("Variance of Girls Drop Out Rate in school is: ",v1)
print("Variance of boys Drop Out Rate in school is: ",v2)
sp = ((v1+v2)/2)**0.05
t = (m1-m2)/(sp*(1/(n1+n2)**0.05))
print("Criteria for Rejection of Null Hypothesis: if(t > ta)")
print("From the Table, We got that: ta = 1.7485 and t=",t)
ta = 1.7485
if (t>ta):
    print('hypothesis is accepted...')
else:
    print('hypothesis is rejected...')
```

```
Null Hypothesis = H0: (u1-u2=0) (Left Tailed)
Alternate hypothesis = Ha: (u1-u2>0) (Right tailed)
Level of Significance: (a=0.05)
Sample size: n1=29  n2=29
Mean of the Girls Drop Out Rate in school is:  97.55
Mean of the Boys Drop Out Rate in school is:  97.06
Variance of Girls Drop Out Rate in school is:  18.3
Variance of boys Drop Out Rate in school is:  18.64
Criteria for Rejection of Null Hypothesis: if(t > ta)
From the Table, We got that: ta = 1.7485 and t= 0.5188527643727852
hypothesis is rejected...
```

Python Code For The Hypothesis.

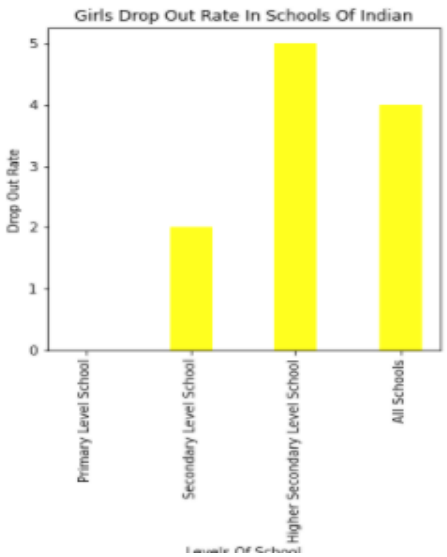
```
import pandas as pd
df = pd.DataFrame([1,4], index=['Girls Drop Out Rate', 'Boys Drop Out Rate '], columns=['x'])
df.plot(kind='pie', subplots=True, figsize=(6,6))
```

```
array([<matplotlib.axes._subplots.AxesSubplot object at 0x000002400B2D3E20>],
      dtype=object)
```



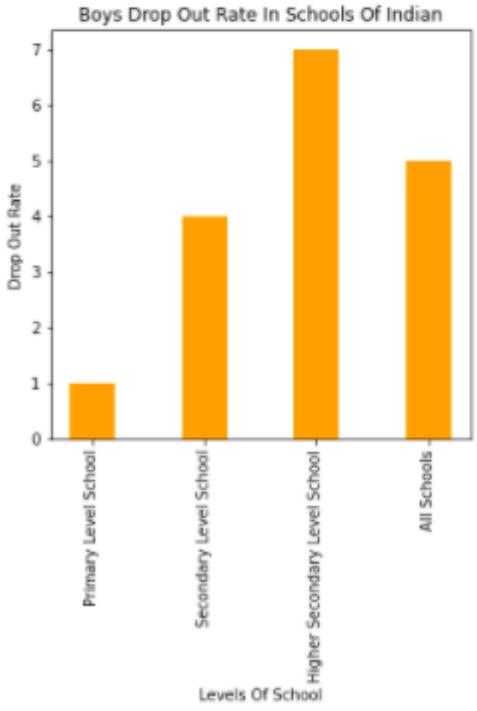
The above given pie graph sates the no. of Drop Out Rate covered by Girls and Boys In Schools In India.

```
import numpy as np
import matplotlib.pyplot as plt

data = {'Primary Level School':0, 'Secondary Level School':2, 'Higher Secondary Level School':5,  'All Schools':4}
courses = list(data.keys())
values = list(data.values())

fig = plt.figure(figsize = (5, 5))

plt.bar(courses, values, color ='yellow',
        width = 0.4)

plt.xlabel("Levels Of School")
plt.ylabel("Drop Out Rate")
plt.title("Girls Drop Out Rate In Schools Of Indian")
plt.xticks(size=10, rotation=90)
plt.show()
```



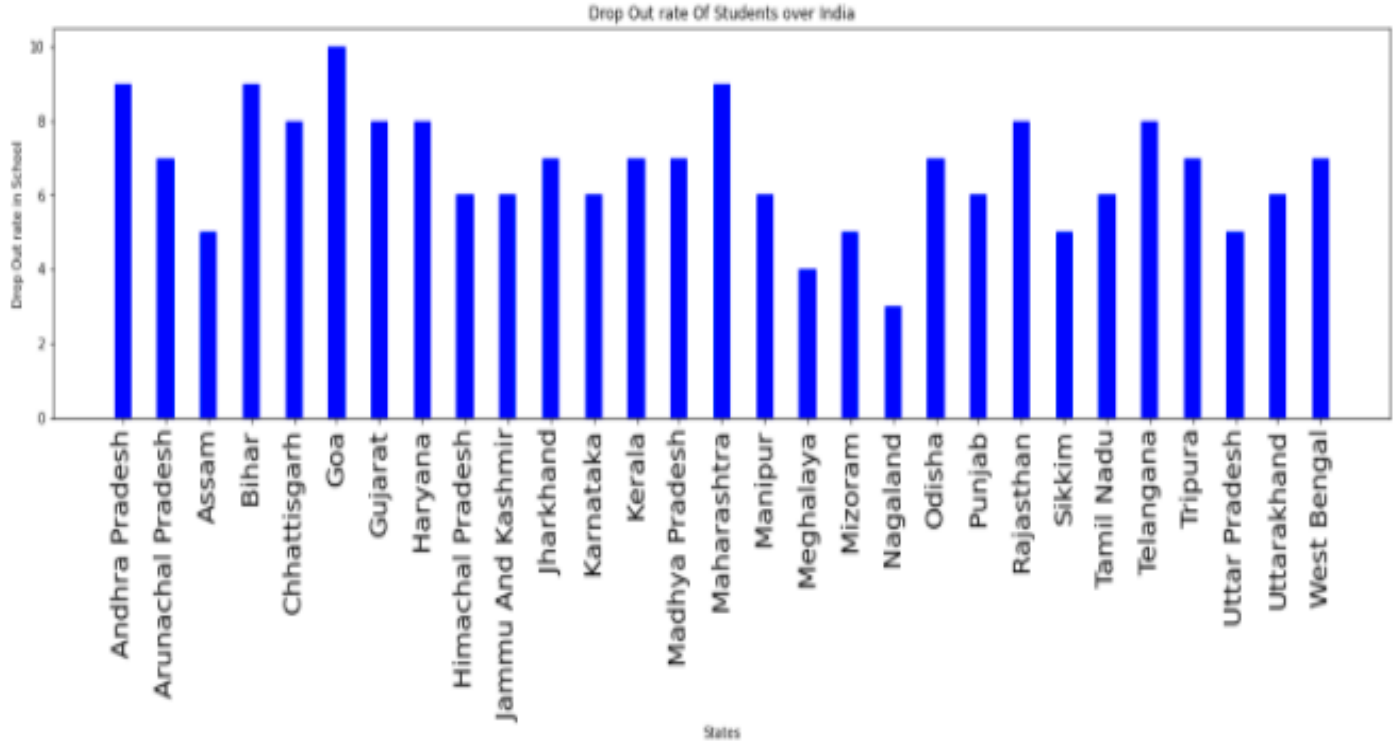
The above given bar graph sates the no. of Drop Out Rate covered by Girls at Levels of Schools In India.

```
import numpy as np
import matplotlib.pyplot as plt

data = {'Primary Level School':1, 'Secondary Level School':4, 'Higher Secondary Level School':7,  'All Schools':5}
courses = list(data.keys())
values = list(data.values())

fig = plt.figure(figsize = (5, 5))

plt.bar(courses, values, color ='orange',
        width = 0.4)

plt.xlabel("Levels Of School")
plt.ylabel("Drop Out Rate")
plt.title("Boys Drop Out Rate In Schools Of Indian")
plt.xticks(size=10, rotation=90)
plt.show()
```



The above given bar graph sates the no. of Drop Out Rate covered by Boys at Levels of Schools In India.

```
import numpy as np
import matplotlib.pyplot as plt

data = {'Andhra Pradesh':9, 'Arunachal Pradesh':7, 'Assam':5,  'Bihar':9, 'Chhattisgarh':8, 'Goa':10,
        'Gujarat':8, 'Haryana':8, 'Himachal Pradesh':6, 'Jammu And Kashmir':6, 'Jharkhand':7, 'Karnataka':6,
        'Kerala':7, 'Madhya Pradesh':7, 'Maharashtra':9, 'Manipur':6, 'Meghalaya':4, 'Mizoram':5, 'Nagaland':3,
        'Odisha':7, 'Punjab':6, 'Rajasthan':8, 'Sikkim':5, 'Tamil Nadu':6, 'Telangana':8, 'Tripura':7,
        'Uttar Pradesh':5, 'Uttarakhand':6, 'West Bengal':7, }
courses = list(data.keys())
values = list(data.values())

fig = plt.figure(figsize = (20,5))

plt.bar(courses, values, color ='blue',
        width = 0.4)

plt.xlabel("States")
plt.ylabel("Drop Out rate in School")
plt.title("Drop Out rate Of Students over India")
plt.xticks(size=20, rotation=90)
plt.show()
```



The above given bar graph sates the Student - Drop Out Rate over India.

```
import numpy as np
import pandas as pd
import sys
n = int(input(pd.read_csv("Project Data.csv")))
a = np.zeros((n,n))
print('Enter Matrix Coefficients:')
for i in range(n):
    for j in range(n):
        a[i][j] = float(input( 'a['+str(i)+']['+ str(j)+']='))
x = np.zeros((n))
print('Enter initial guess vector: ')
for i in range(n):
    x[i] = float(input( 'x['+str(i)+']='))
tolerable_error = float(input('Enter tolerable error: '))
max_iteration = int(input('Enter maximum number of steps: '))
lambda_old = 1.0
condition =  True
step = 1
while condition:
    x = np.matmul(a,x)
    lambda_new = max(abs(x))
    x = x/lambda_new
    print('\nSTEP %d' %(step))
    print('----------')
    print('Eigen Value = %0.4f' %(lambda_new))
    print('Eigen Vector: ')
    for i in range(n):
        print('%0.3f\t' % (x[i]))
    step = step + 1
    if step > max_iteration:
        print('Not convergent in given maximum iteration!')
        break
    error = abs(lambda_new - lambda_old)
    print('errror='+ str(error))
    lambda_old = lambda_new
    condition = error > tolerable_error
```

Python Code Of Power Method For The Data.

2. To study and analyse the Pupil Teacher Ratio so that we can know the status of the different levels of schooling in India.

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
from sklearn.linear_model import LinearRegression
```

```
d = pd.read_csv("Both.csv")
d
```

	U.T./States	Primary Level	Upper-Primary Level	Secondary Level	Senior secondary Level
0	A and N Island	8	6	14	16
1	Andhra Pradesh	21	16	20	71
2	Arunachal Pradesh	12	7	22	37
3	Assam	21	13	14	20
4	Bihar	36	24	66	39
5	Chandigarh	13	9	13	28
6	Chhattisgarh	20	17	33	21
7	D and N Haveli	17	13	30	30
8	Daman and Diu	26	14	17	13
9	Delhi	24	17	30	21
10	Goa	20	16	13	18
11	Gujarat	19	13	34	29
12	Haryana	20	13	15	15
13	Himachal Pradesh	12	10	18	14
14	Jammu And Kashmir	9	6	15	29
15	Jharkhand	27	19	62	78
16	Karnataka	19	13	16	30
17	Kerala	18	14	17	21
18	Lakshadweep	7	7	7	12
19	Madhya Pradesh	20	18	39	38
20	Maharashtra	24	17	23	44
21	Manipur	12	8	12	19
22	Meghalaya	21	13	12	21
23	Mizoram	14	6	9	15
24	Nagaland	10	6	15	21
25	Odisha	17	14	20	45
26	Puducherry	14	9	11	17
27	Punjab	18	12	16	26
28	Rajasthan	17	10	21	32
29	Sikkim	5	5	17	15
30	Tamil Nadu	18	15	21	25
31	Telangana	23	15	22	47
32	Tripura	10	8	28	11
33	Uttar Pradesh	39	31	56	97
34	Uttarakhand	18	17	16	25
35	West Bengal	25	27	39	57

```
import numpy as np
from scipy.stats import norm
import statistics

mean1 = statistics.mean(d['Primary Level'])
print("Mean of the Pupil-Teacher Ratio of primary level -> " , round(mean1,3))

std1 = d['Primary Level'].std()
print("Standard deviation of the Pupil-Teacher Ratio of primary level -> " , round(std1,3))

var1 = print("Variance of the Pupil-Teacher Ratio of primary level -> ", round((std1*std1),3))

modev1 = statistics.mode(d['Primary Level'])
print("Mode of the Pupil-Teacher Ratio of primary level -> " , modev1)

medv1 = statistics.median(d['Primary Level'])
print("Median of the Pupil-Teacher Ratio of primary level -> " , medv1)

minv1 = min(d['Primary Level'])
print("Minimum value of the Pupil-Teacher Ratio of primary level -> " , minv1)

maxv1 = max(d['Primary Level'])
print("Maximum value of the Pupil-Teacher Ratio of primary level -> " , maxv1)
```

```
Mean of the Pupil-Teacher Ratio of primary level ->  18.167
Standard deviation of the Pupil-Teacher Ratio of primary level ->  7.304
Variance of the Pupil-Teacher Ratio of primary level ->  53.343
Mode of the Pupil-Teacher Ratio of primary level ->  20
Median of the Pupil-Teacher Ratio of primary level ->  18.0
Minimum value of the Pupil-Teacher Ratio of primary level ->  5
Maximum value of the Pupil-Teacher Ratio of primary level ->  39
```

```
mean2 = statistics.mean(d['Upper-Primary Level'])
print("Mean of the Pupil-Teacher Ratio of Upper-primary level -> " , round(mean2,3))

std2 = d['Upper-Primary Level'].std()
print("Standard deviation of the Pupil-Teacher Ratio of Upper-primary level -> " , round(std2,3))

var2 = print("Variance of the Pupil-Teacher Ratio of Upper-primary level -> ", round((std2*std2),3))

modev2 = statistics.mode(d['Upper-Primary Level'])
print("Mode of the Pupil-Teacher Ratio of Upper-primary level -> " , modev2)

medv2 = statistics.median(d['Upper-Primary Level'])
print("Median of the Pupil-Teacher Ratio of Upper-primary level -> " , medv2)

minv2 = min(d['Upper-Primary Level'])
print("Minimum value of the Pupil-Teacher Ratio of Upper-primary level -> " , minv2)

maxv2 = max(d['Upper-Primary Level'])
print("Maximum value of the Pupil-Teacher Ratio of Upper-primary level -> " , maxv2)
```

```
Mean of the Pupil-Teacher Ratio of Upper-primary level ->  13.278
Standard deviation of the Pupil-Teacher Ratio of Upper-primary level ->  5.917
Variance of the Pupil-Teacher Ratio of Upper-primary level ->  35.006
Mode of the Pupil-Teacher Ratio of Upper-primary level ->  13
Median of the Pupil-Teacher Ratio of Upper-primary level ->  13.0
Minimum value of the Pupil-Teacher Ratio of Upper-primary level ->  5
Maximum value of the Pupil-Teacher Ratio of Upper-primary level ->  31
```

```
mean3 = statistics.mean(d['Secondary Level'])
print("Mean of the Pupil-Teacher Ratio of Secondary Level -> " , round(mean3,3))

std3 = d['Secondary Level'].std()
print("Standard deviation of the Pupil-Teacher Ratio of Secondary Level -> " , round(std3,3))

var3 = print("Variance of the Pupil-Teacher Ratio of Secondary Level -> ", round((std3*std3),3))

modev3 = statistics.mode(d['Secondary Level'])
print("Mode of the Pupil-Teacher Ratio of Secondary Level -> " , modev3)

medv3 = statistics.median(d['Secondary Level'])
print("Median of the Pupil-Teacher Ratio of Secondary Level -> " , medv3)

minv3 = min(d['Secondary Level'])
print("Minimum value of the Pupil-Teacher Ratio of Secondary Level -> " , minv3)

maxv3 = max(d['Secondary Level'])
print("Maximum value of the Pupil-Teacher Ratio of Secondary Level -> " , maxv3)
```

```
Mean of the Pupil-Teacher Ratio of Secondary Level ->  23.139
Standard deviation of the Pupil-Teacher Ratio of Secondary Level ->  14.157
Variance of the Pupil-Teacher Ratio of Secondary Level ->  200.409
Mode of the Pupil-Teacher Ratio of Secondary Level ->  17
Median of the Pupil-Teacher Ratio of Secondary Level ->  17.5
Minimum value of the Pupil-Teacher Ratio of Secondary Level ->  7
Maximum value of the Pupil-Teacher Ratio of Secondary Level ->  66
```

```
mean4 = statistics.mean(d['Senior secondary Level'])
print("Mean of the Pupil-Teacher Ratio of Senior secondary Level -> " , round(mean4,3))

std4 = d['Senior secondary Level'].std()
print("Standard deviation of the Pupil-Teacher Ratio of Senior secondary Level -> " , round(std4,3))

var4 = print("Variance of the Pupil-Teacher Ratio of Senior secondary Level -> ", round((std4*std4),3))

modev4 = statistics.mode(d['Senior secondary Level'])
print("Mode of the Pupil-Teacher Ratio of Senior secondary Level -> " , modev4)

medv4 = statistics.median(d['Senior secondary Level'])
print("Median of the Pupil-Teacher Ratio of Senior secondary Level -> " , medv4)

minv4 = min(d['Senior secondary Level'])
print("Minimum value of the Pupil-Teacher Ratio of Senior secondary Level -> " , minv4)

maxv4 = max(d['Senior secondary Level'])
print("Maximum value of the Pupil-Teacher Ratio of Senior secondary Level -> " , maxv4)
```

```
Mean of the Pupil-Teacher Ratio of Senior secondary Level ->  30.472
Standard deviation of the Pupil-Teacher Ratio of Senior secondary Level ->  19.431
Variance of the Pupil-Teacher Ratio of Senior secondary Level ->  377.571
Mode of the Pupil-Teacher Ratio of Senior secondary Level ->  21
Median of the Pupil-Teacher Ratio of Senior secondary Level ->  25.0
Minimum value of the Pupil-Teacher Ratio of Senior secondary Level ->  11
Maximum value of the Pupil-Teacher Ratio of Senior secondary Level ->  97
```

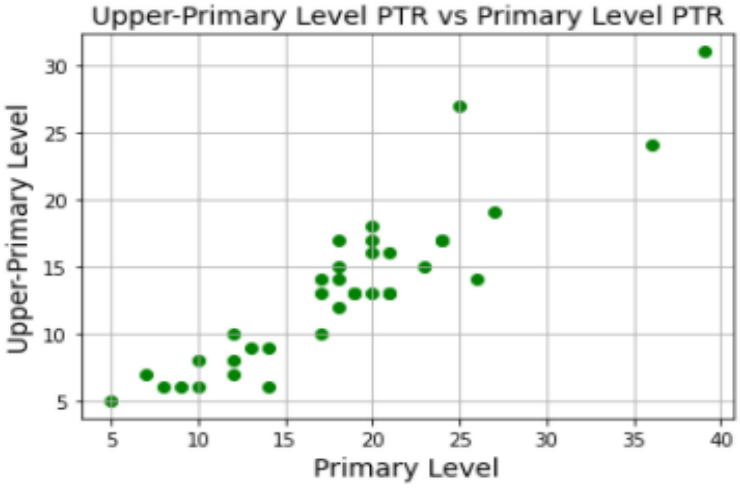
```
d.describe()
```

	Primary Level	Upper-Primary Level	Secondary Level	Senior secondary Level
count	36.000000	36.000000	36.000000	36.000000
mean	18.166667	13.277778	23.138889	30.472222
std	7.303619	5.916616	14.156579	19.431177
min	5.000000	5.000000	7.000000	11.000000
25%	12.750000	8.750000	14.750000	17.750000
50%	18.000000	13.000000	17.500000	25.000000
75%	21.000000	16.250000	28.500000	37.250000
max	39.000000	31.000000	66.000000	97.000000

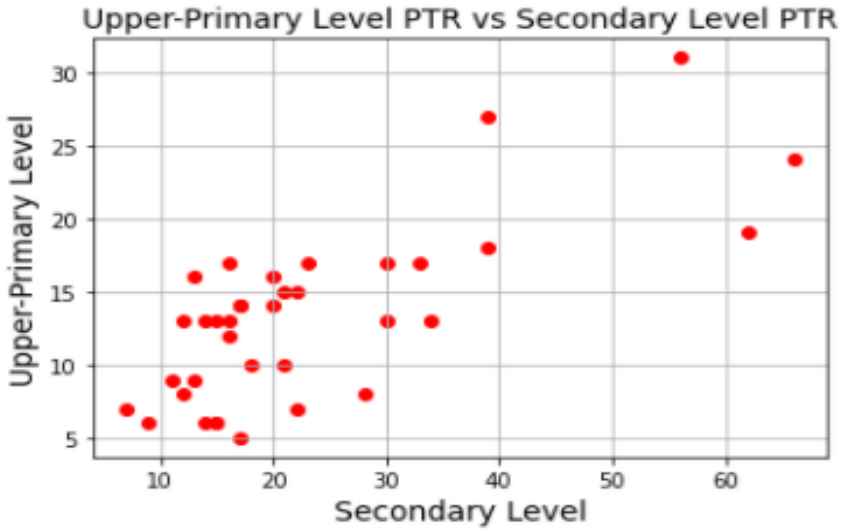
```
correlation = d.corr(method = 'pearson')
correlation
```

	Primary Level	Upper-Primary Level	Secondary Level	Senior secondary Level
Primary Level	1.000000	0.908683	0.708292	0.637424
Upper-Primary Level	0.908683	1.000000	0.734287	0.692194
Secondary Level	0.708292	0.734287	1.000000	0.664914
Senior secondary Level	0.637424	0.692194	0.664914	1.000000

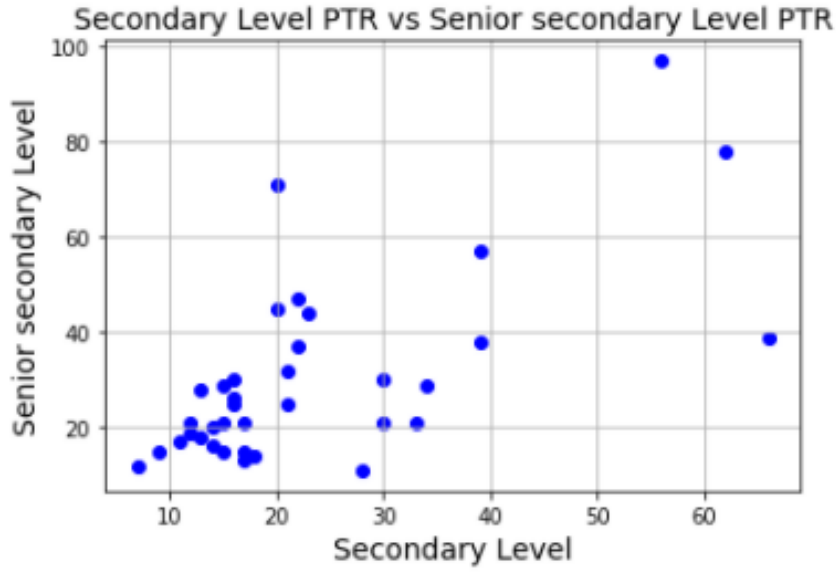
```
plt.scatter(d['Primary Level'] , d['Upper-Primary Level'] , color='green')
plt.title('Upper-Primary Level PTR vs Primary Level PTR', fontsize=14)
plt.xlabel('Primary Level', fontsize=14)
plt.ylabel('Upper-Primary Level', fontsize=14)
plt.grid(True)
plt.show()
```



```
plt.scatter(d['Secondary Level'] , d['Upper-Primary Level'] , color='red')
plt.title('Upper-Primary Level PTR vs Secondary Level PTR', fontsize=14)
plt.xlabel('Secondary Level', fontsize=14)
plt.ylabel('Upper-Primary Level', fontsize=14)
plt.grid(True)
plt.show()
```




```
plt.scatter(d['Secondary Level'] , d['Senior secondary Level'] , color='blue')
plt.title('Secondary Level PTR vs Senior secondary Level PTR', fontsize=14)
plt.xlabel('Secondary Level', fontsize=14)
plt.ylabel('Senior secondary Level', fontsize=14)
plt.grid(True)
plt.show()
```



```
import math
Ho = 30
n = 36
mean = 18.166
sd = 7.303
root = math.sqrt(n)
print("Root of number of samples" , root)
z = mean - Ho
zt = z/sd

zalpha = 1.96
ztest = zt*root
print("Value of z :-> " , ztest)
if ztest < -zalpha:
    print("NULL HYPOTHESES IS REJECTED BY LEFT TAILED TEST")
else :
    print("NULL HYPOTHESES IS ACCEPTED")
```

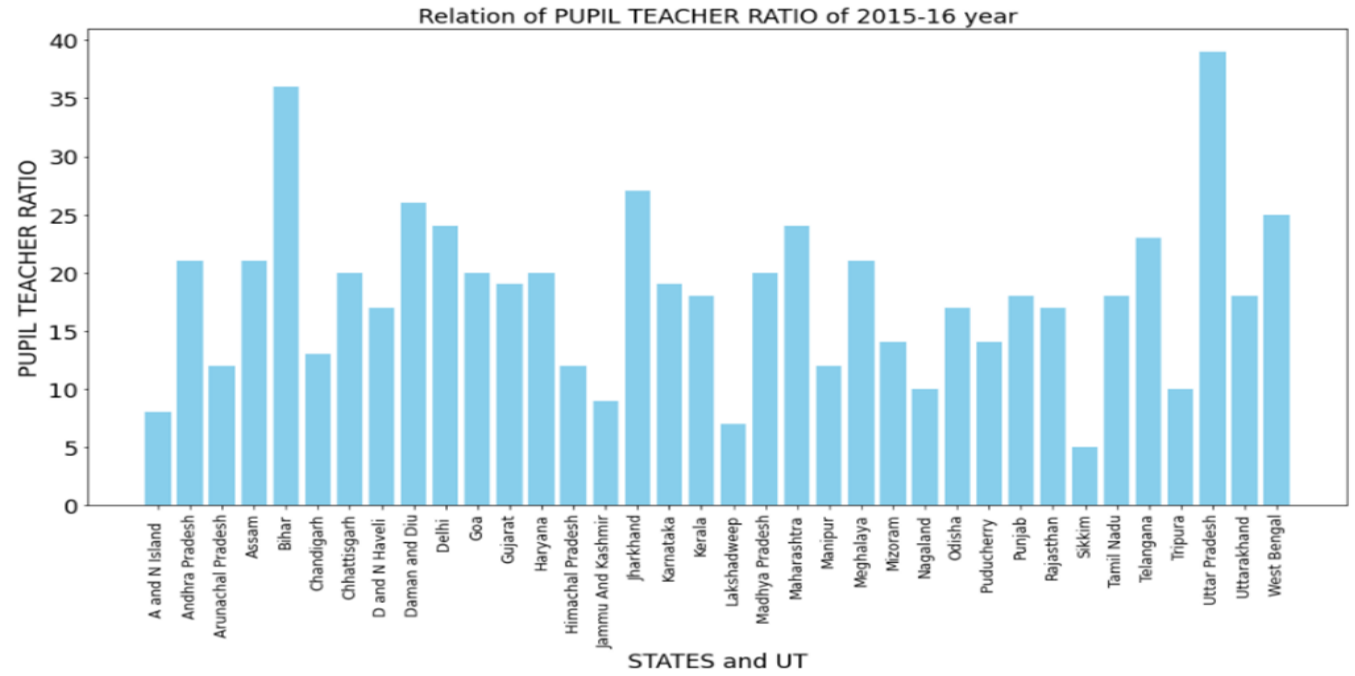
```
Root of number of samples 6.0
Value of z :->  -9.72257976174175
NULL HYPOTHESES IS REJECTED BY LEFT TAILED TEST
```

```
Ho = 35
n = 36
mean = 13.277
sd = 5.916
root = math.sqrt(n)
print("Root of number of samples" , root)
z = mean - Ho
zt = z/sd

zalpha = 1.96
ztest = zt*root
print("Value of z :-> " , ztest)
if ztest < -zalpha:
    print("NULL HYPOTHESES IS REJECTED BY LEFT TAILED TEST")
else :
    print("NULL HYPOTHESES IS ACCEPTED")
```

```
Root of number of samples 6.0
Value of z :->  -22.031440162271803
NULL HYPOTHESES IS REJECTED BY LEFT TAILED TEST
```

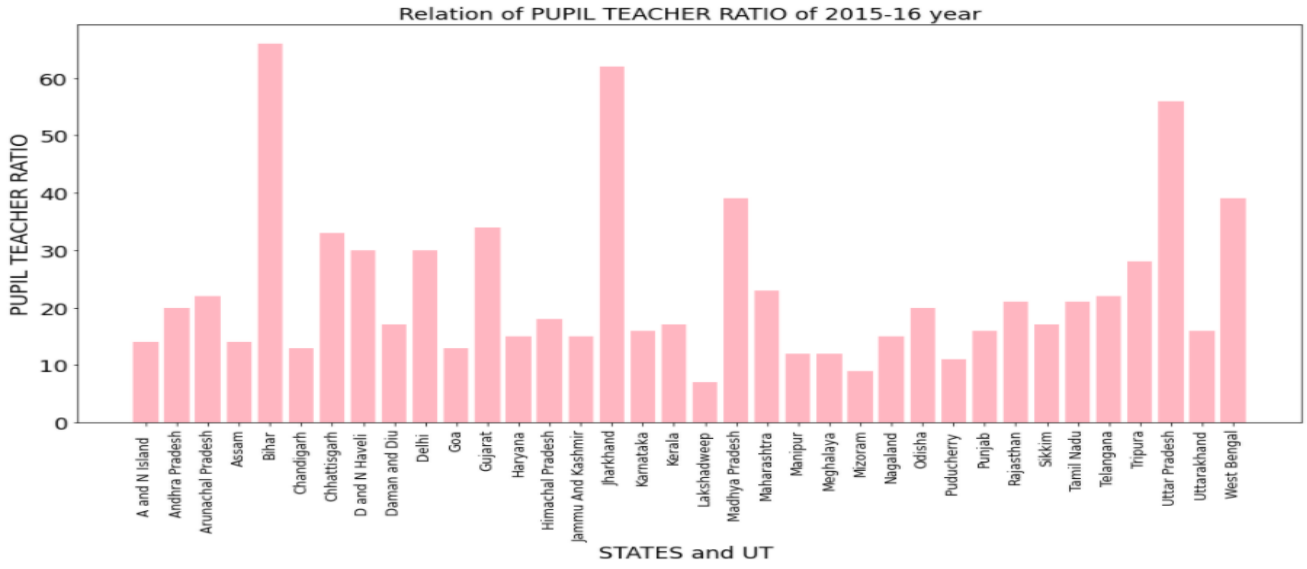
```
import matplotlib.pyplot as plt

fig = plt.figure(figsize = (17,8))        # Assigning the dimensions of the graph.


plt.xlabel('STATES and UT' , size=18)           # Labelling the X axis.
plt.ylabel('PUPIL TEACHER RATIO' , size=18)     # Labelling the Y axis.
plt.title("Relation of PUPIL TEACHER RATIO of 2015-16 year" , size=18)
plt.bar(d['U.T./States'],d['Primary Level'],color = 'skyblue')          # Assigning the type of the graph i.e. bar graph with x
plt.yticks(size=18)
plt.xticks(size=12,rotation=90)
plt.show()
```

```
import matplotlib.pyplot as plt

fig = plt.figure(figsize = (17,8))          # Assigning the dimensions of the graph.


plt.xlabel('STATES and UT' , size=18)              # Labelling the X axis.
plt.ylabel('PUPIL TEACHER RATIO' , size=18)        # Labelling the Y axis.
plt.title("Relation of PUPIL TEACHER RATIO of 2015-16 year" , size=18)
plt.bar(d['U.T./States'],d['Primary Level'],color = 'skyblue')            # Assigning the type of the graph i.e. bar graph with x
plt.yticks(size=18)
plt.xticks(size=12,rotation=90)
plt.show()
```



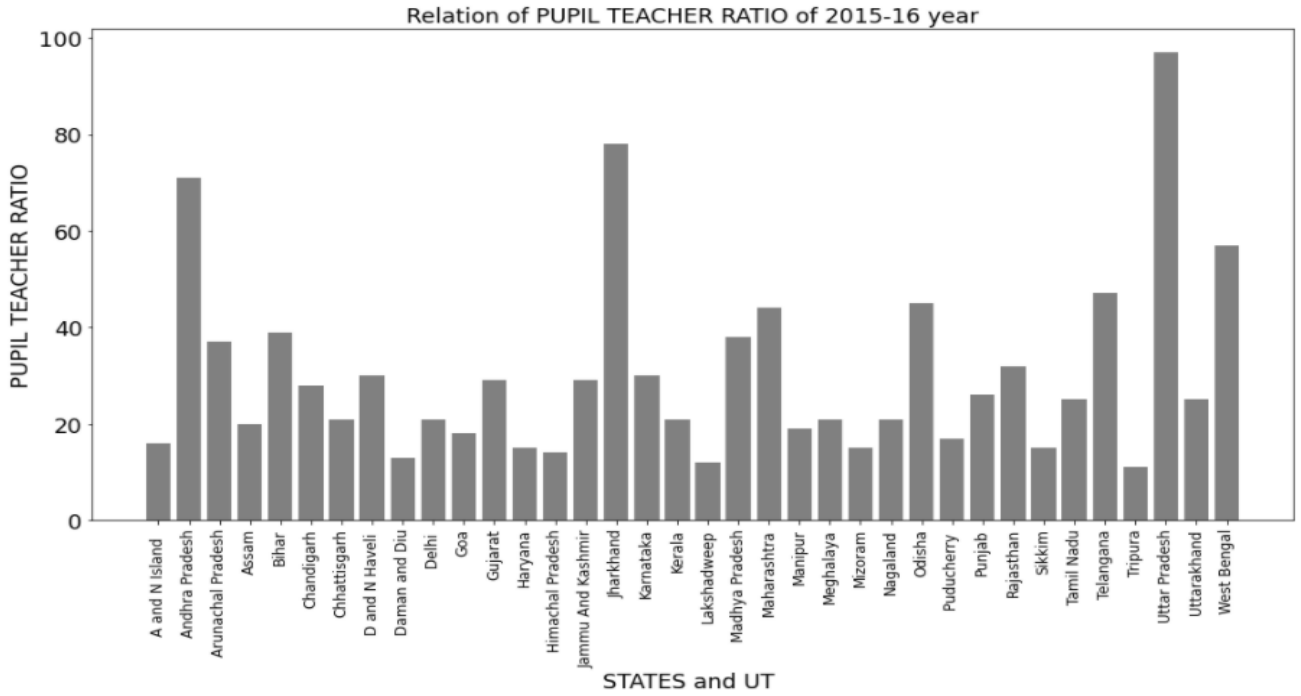
In the above graph we are seeing the details of the pupil teacher ratio of the different states. This graph is showing the details of the Upper - Primary Level Pupil Teacher Ratio for the year 2015-16

```
fig = plt.figure(figsize = (17,8))          # Assigning the dimensions of the graph.

plt.xlabel('STATES and UT ' , size=18)              # Labelling the X axis.
plt.ylabel('PUPIL TEACHER RATIO' , size=18)        # Labelling the Y axis.
plt.title("Relation of PUPIL TEACHER RATIO of 2015-16 year" , size=18)
plt.bar(d['U.T./States'],d['Secondary Level'] , color = 'lightpink')            # Assigning the type of the graph i.e. bar graph w
plt.yticks(size=18)
plt.xticks(size=12,rotation=90)
plt.show()
```



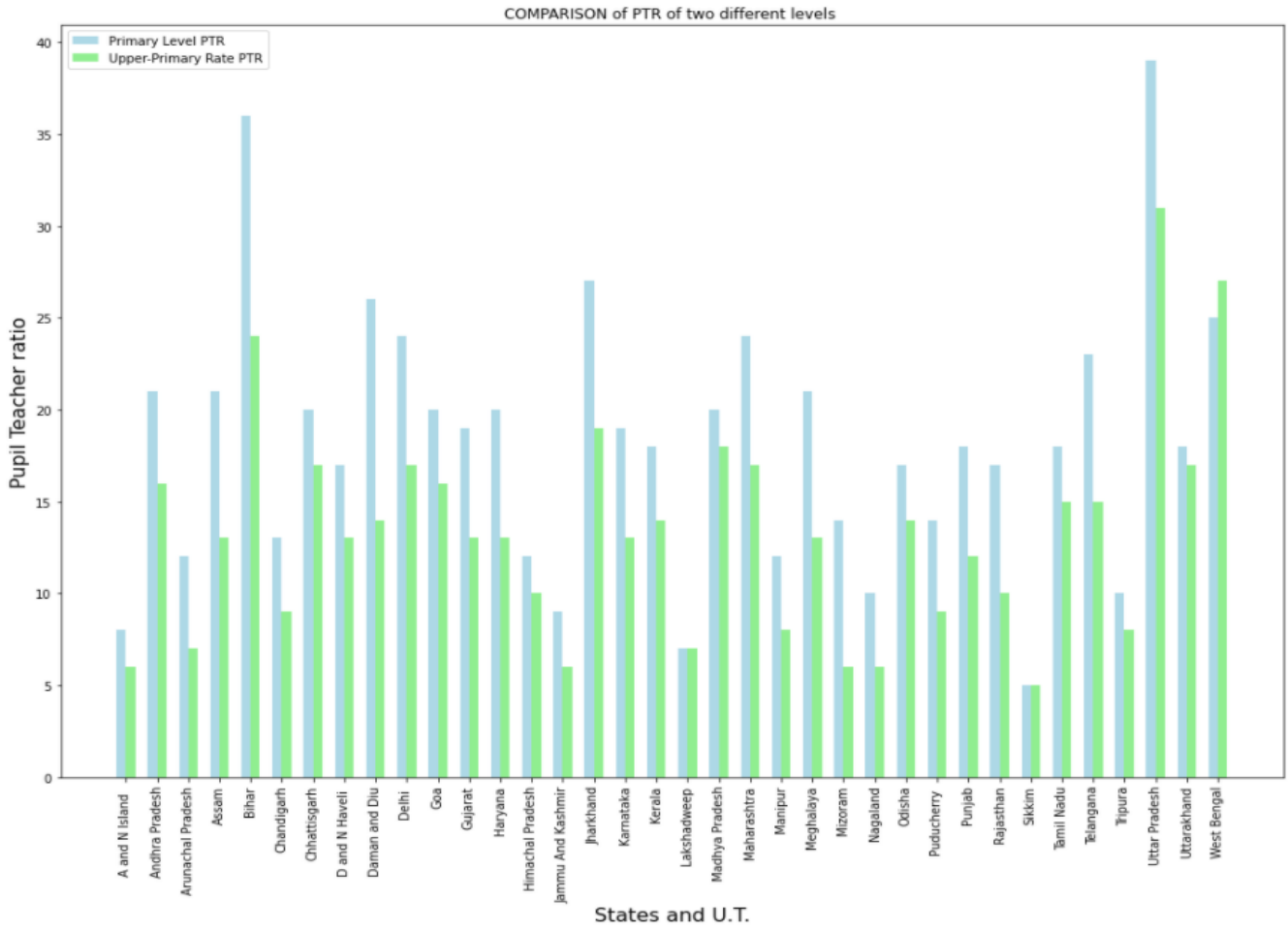
#In the above graph we are seeing the details of the pupil teacher ratio of the different states. This graph is showing the details of the Secondary Level Pupil Teacher Ratio for the year 2015-16

```
fig = plt.figure(figsize = (17,8))          # Assigning the dimensions of the graph.

plt.xlabel('STATES and UT ' , size=18)            # Labelling the X axis.
plt.ylabel('PUPIL TEACHER RATIO' , size=18)       # Labelling the Y axis.
plt.title("Relation of PUPIL TEACHER RATIO of 2015-16 year" , size=18)
plt.bar(d['U.T./States'],d['Senior secondary Level'] , color = 'Grey')          # Assigning the type of the graph i.e. bar graph
plt.yticks(size=18)
plt.xticks(size=12,rotation=90)
plt.show()
```



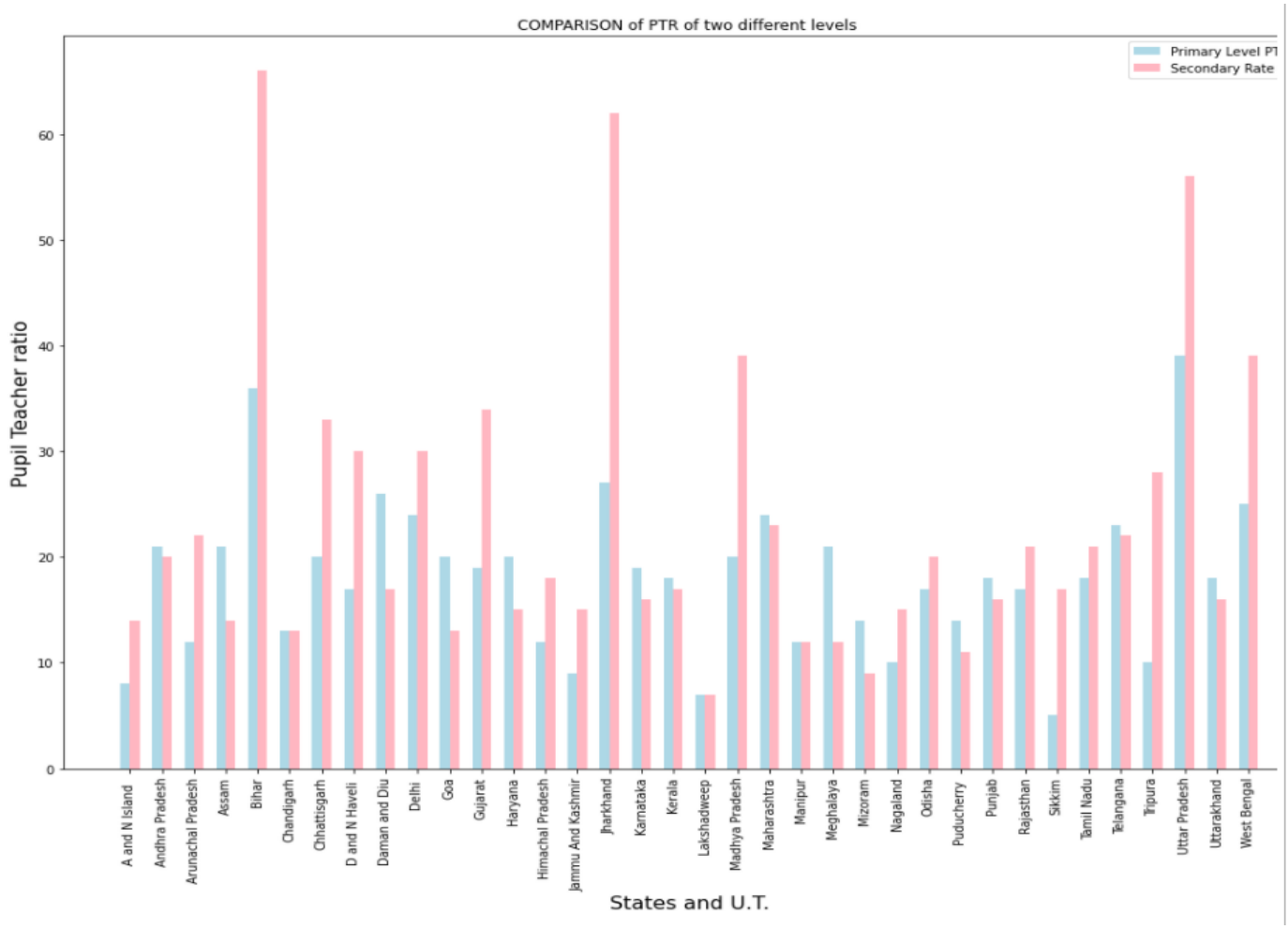
In the above graph we are seeing the details of the pupil teacher ratio of the different states. This graph is showing the details of the Senior Secondary Level Pupil Teacher Ratio for the year 2015-16

```
import matplotlib.pyplot as plt

n = 36

ind = np.arange(n)
plt.figure(figsize = (17,12))
width = 0.3

x = d['U.T./States']
y = d['Primary Level']
z = d['Upper-Primary Level']

plt.bar(ind,y, width , label = 'Primary Level PTR ' , color = 'lightblue')
plt.bar(ind+width , z ,width , label = 'Upper-Primary Rate PTR' , color = 'lightgreen')

plt.xlabel("States and U.T." ,size =16)
plt.ylabel("Pupil Teacher ratio" ,size =16)
plt.xticks(ind +width/2 , (x) , rotation = 90)
plt.title("COMPARISON of PTR of two different levels ")
plt.legend()
plt.show()
```



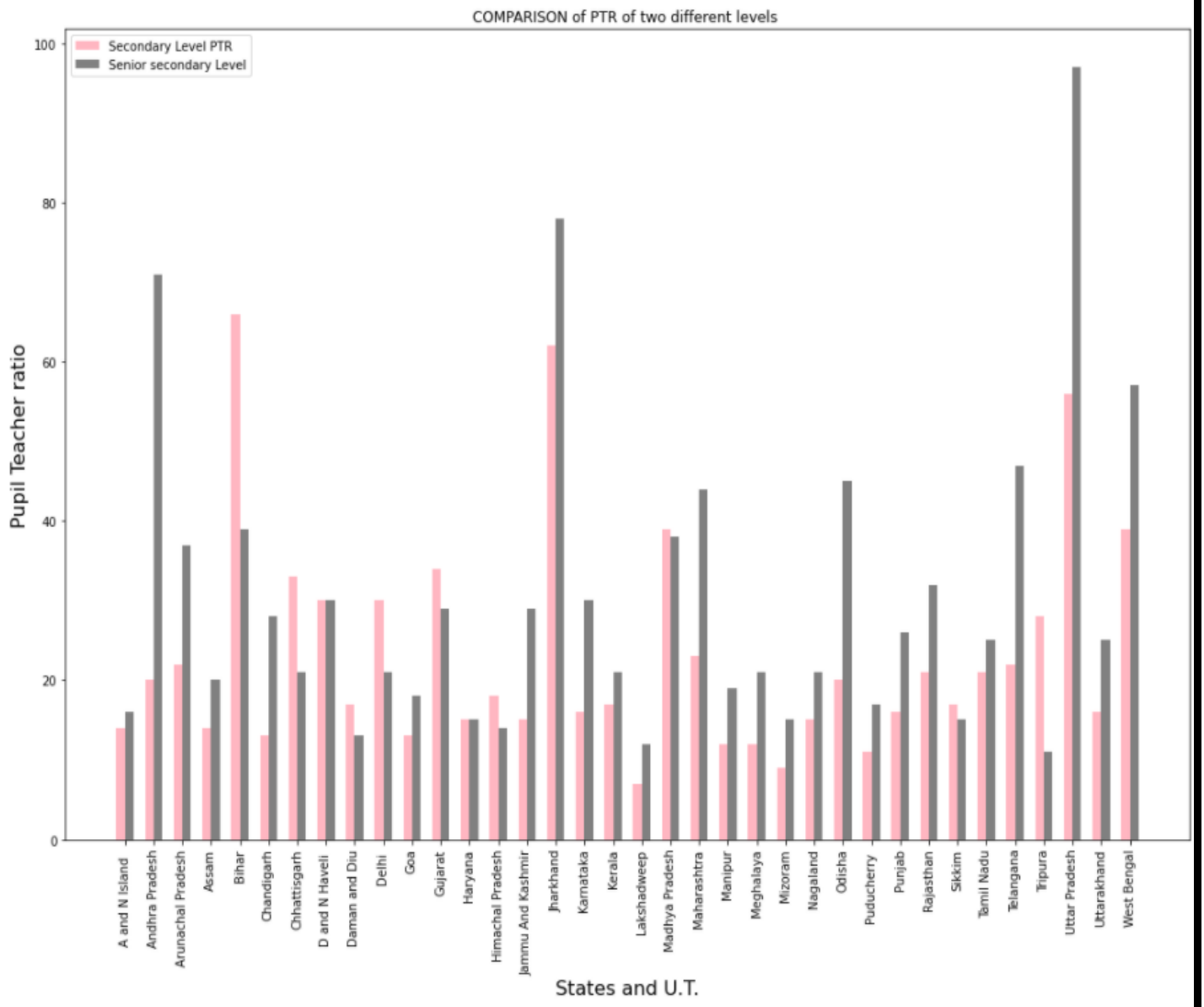
Here in this graph we are comparing the PTR of Primary Level and Upper-Primary Level for the year 2015-16.

```
n = 36

ind = np.arange(n)
plt.figure(figsize = (17,12))
width = 0.3

x = d['U.T./States']
y = d['Primary Level']
z = d['Secondary Level']

plt.bar(ind,y, width , label = 'Primary Level PTR ' , color = 'lightblue')
plt.bar(ind+width , z ,width , label = 'Secondary Rate PTR' , color = 'lightpink')

plt.xlabel("States and U.T." ,size =16)
plt.ylabel("Pupil Teacher ratio" ,size =16)
plt.xticks(ind +width/2 , (x) , rotation = 90)
plt.title("COMPARISON of PTR of two different levels ")
plt.legend()
plt.show()
```



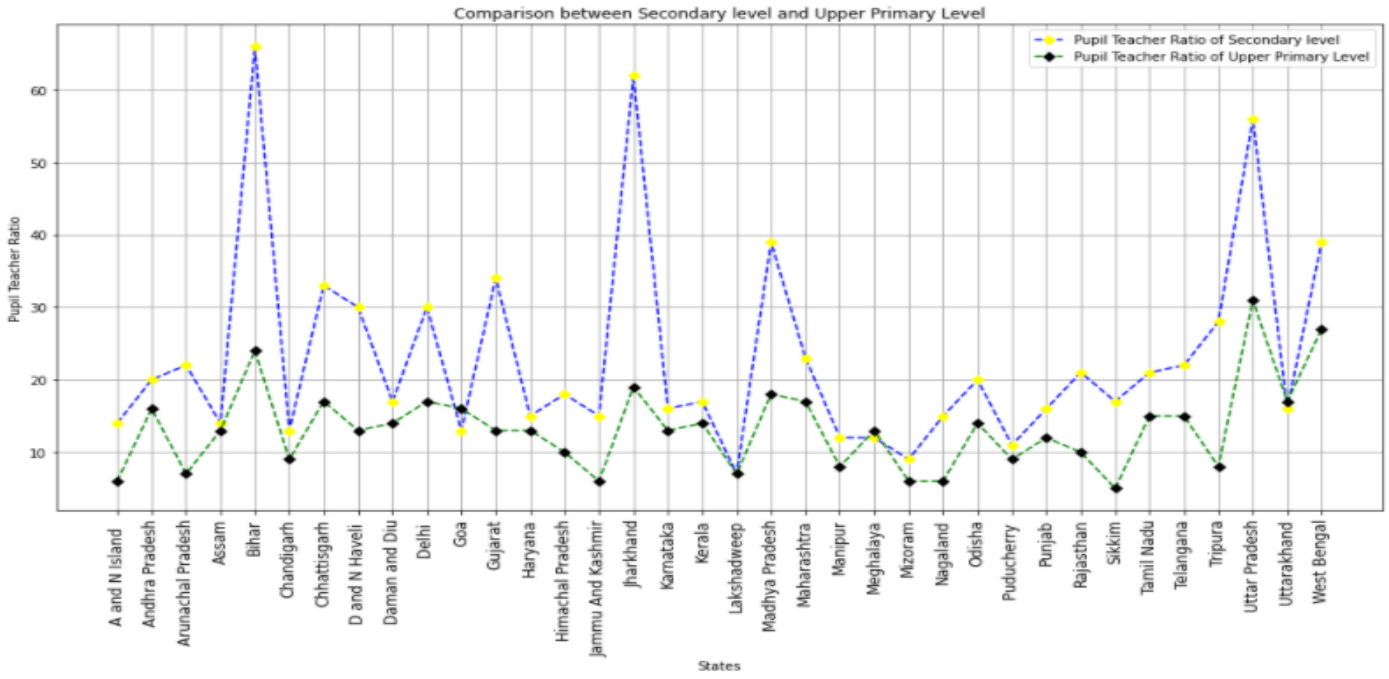
Here in this graph we are comparing the PTR of Primary Level and Secondary Level for the year 2015-16.

```
n = 36

ind = np.arange(n)
plt.figure(figsize = (17,12))
width = 0.3

x = d['U.T./States']
y = d['Secondary Level']
z = d['Senior secondary Level']

plt.bar(ind,y, width , label = 'Secondary Level PTR ' , color = 'lightpink')
plt.bar(ind+width , z ,width , label = 'Senior secondary Level' , color = 'grey')

plt.xlabel("States and U.T." ,size =16)
plt.ylabel("Pupil Teacher ratio" ,size =16)
plt.xticks(ind +width/2 , (x) , rotation = 90)
plt.title("COMPARISON of PTR of two different levels ")
plt.legend()
plt.show()
```



Here in this graph we are comparing the PTR of Secondary Level and Senior Secondary Level for the year 2015-16.

```
plt.figure(figsize = (17 , 8))
x = d['U.T./States']
plt.plot(d['U.T./States'] , d['Secondary Level'] , color = 'blue' ,
         marker = 'D' , markersize = 5 , markerfacecolor = 'yellow' ,
         markeredgecolor = "yellow" , linestyle = 'dashed' ,  label = 'Pupil Teacher Ratio of Secondary level')
plt.plot(d['U.T./States'] , d['Upper-Primary Level'] , color = 'green' ,
         marker = 'D' , markersize = 5 , markerfacecolor = 'black' ,
         markeredgecolor = "black" , linestyle = 'dashed' , label = 'Pupil Teacher Ratio of Upper Primary Level')
plt.xticks(x , size = 12 , rotation = 90)
plt.grid(linewidth =1)
plt.title("Comparison between Secondary level and Upper Primary Level")
plt.xlabel("States")
plt.ylabel("Pupil Teacher Ratio")
plt.legend()
plt.show()
```



Here in this graph we are comparing the PTR of Secondary Level and Upper Primary Level for the year 2015-16.

Now for the Union Territories

In [2]:
```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
from sklearn.linear_model import LinearRegression
```

In [3]:
```
d = pd.read_csv("Ut.csv")
d
```

Out[3]:

	U.T.	Primary Level	Upper-Primary Level	Secondary Level	Senior secondary Level
0	A and N Island	8	6	14	16
1	Chandigarh	13	9	13	28
2	D and N Haveli	17	13	30	30
3	Daman and Diu	26	14	17	13
4	Delhi	24	17	30	21
5	Lakshadweep	7	7	7	12
6	Puducherry	14	9	11	17

```
a1 = d['Primary Level']+d['Upper-Primary Level']+d['Secondary Level']+d['Senior secondary Level']
d['overall'] = a1
d
```

	U.T.	Primary Level	Upper-Primary Level	Secondary Level	Senior secondary Level	overall
0	A and N Island	8	6	14	16	44
1	Chandigarh	13	9	13	28	63
2	D and N Haveli	17	13	30	30	90
3	Daman and Diu	26	14	17	13	70
4	Delhi	24	17	30	21	92
5	Lakshadweep	7	7	7	12	33
6	Puducherry	14	9	11	17	51

```
l = []
a = np.array([[1,1,1,1,1,0,1],
                        [0,1,1,1,1,0,0],
                        [0,0,1,0,1,0,0],
                        [0,0,1,1,1,0,0],
                        [0,0,0,0,1,0,0],
                        [1,1,1,1,1,1,1],
                        [0,1,1,1,1,0,1]])
n = 7
# for initial guess vector
x = np.zeros((n))
print('Enter initial guess vector: ')
for i in range(n):
    x[i] = float(input( 'x['+str(i)+']='))
#Reading tolerable error
tolerable_error = float(input('Enter tolerable error: '))
#steps
max_iteration = int(input('Enter maximum number of steps: '))
# Power Method Implementation
lambda_old = 1.0
condition =  True
step = 1
while condition:
    x = np.matmul(a,x)
    # Finding new Eigen value and Eigen vector
    lambda_new = max(abs(x))
    x = x/lambda_new
    # Displaying Eigen value and Eigen Vector
    print('\nIteration %d' %(step))
    print('----------')
    print('Eigen Value = %0.4f' %(lambda_new))
    print('Eigen Vector: ')
    l.append(lambda_new)
    for i in range(n):
        print('%0.3f\t' % (x[i]))
    # Checking maximum iteration
    step = step + 1
    if step > max_iteration:
        print('Not convergent in given maximum iteration!')
        break
    # Calculating error
    error = abs(lambda_new - lambda_old)
    #print('errror='+ str(error))
    lambda_old = lambda_new
    condition = error > tolerable_error
print("Largest Dominant Eigen Value Possible: ",(max(l)))
print("Smallest Dominant Eigen Value Possible: ",(min(l)))
```

```
Enter initial guess vector:
x[0]=1
x[1]=1
x[2]=1
x[3]=1
x[4]=1
x[5]=1
x[6]=1
Enter tolerable error: 0.001
Enter maximum number of steps: 100

Iteration 1
----------
Eigen Value = 7.0000
Eigen Vector:
0.857
0.571
0.286
0.429
0.143
1.000
0.714

Iteration 2
----------
Eigen Value = 4.0000
Eigen Vector:
0.750
0.357
0.107
0.214
0.036
1.000
0.536

Iteration 3
----------
Eigen Value = 3.0000
Eigen Vector:
0.667
0.238
0.048
0.119
0.012
1.000
0.417
```

```
Iteration 4
----------
Eigen Value = 2.5000
Eigen Vector:
0.600
0.167
0.024
0.071
0.005
1.000
0.333

Iteration 5
----------
Eigen Value = 2.2000
Eigen Vector:
0.545
0.121
0.013
0.045
0.002
1.000
0.273

Iteration 6
----------
Eigen Value = 2.0000
Eigen Vector:
0.500
0.091
0.008
0.030
0.001
1.000
0.227

Iteration 7
----------
Eigen Value = 1.8571
Eigen Vector:
0.462
0.070
0.005
0.021
0.001
1.000
0.192
```

```
Iteration 8
----------
Eigen Value = 1.7500
Eigen Vector:
0.429
0.055
0.003
0.015
0.000
1.000
0.165

Iteration 9
----------
Eigen Value = 1.6667
Eigen Vector:
0.400
0.044
0.002
0.011
0.000
1.000
0.143

Iteration 10
----------
Eigen Value = 1.6000
Eigen Vector:
0.375
0.036
0.001
0.008
0.000
1.000
0.125

Iteration 11
----------
Eigen Value = 1.5455
Eigen Vector:
0.353
0.029
0.001
0.006
0.000
1.000
0.110
```

```
Iteration 12
----------
Eigen Value = 1.5000
Eigen Vector:
0.333
0.025
0.001
0.005
0.000
1.000
0.098

Iteration 13
----------
Eigen Value = 1.4615
Eigen Vector:
0.316
0.021
0.001
0.004
0.000
1.000
0.088

Iteration 14
----------
Eigen Value = 1.4286
Eigen Vector:
0.300
0.018
0.000
0.003
0.000
1.000
0.079

Iteration 15
----------
Eigen Value = 1.4000
Eigen Vector:
0.286
0.015
0.000
0.003
0.000
1.000
0.071
```

```
Iteration 16
----------
Eigen Value = 1.3750
Eigen Vector:
0.273
0.013
0.000
0.002
0.000
1.000
0.065

Iteration 17
----------
Eigen Value = 1.3529
Eigen Vector:
0.261
0.011
0.000
0.002
0.000
1.000
0.059

Iteration 18
----------
Eigen Value = 1.3333
Eigen Vector:
0.250
0.010
0.000
0.001
0.000
1.000
0.054

Iteration 19
----------
Eigen Value = 1.3158
Eigen Vector:
0.240
0.009
0.000
0.001
0.000
1.000
0.050
```

```
Iteration 20
----------
Eigen Value = 1.3000
Eigen Vector:
0.231
0.008
0.000
0.001
0.000
1.000
0.046

Iteration 21
----------
Eigen Value = 1.2857
Eigen Vector:
0.222
0.007
0.000
0.001
0.000
1.000
0.043

Iteration 22
----------
Eigen Value = 1.2727
Eigen Vector:
0.214
0.006
0.000
0.001
0.000
1.000
0.040

Iteration 23
----------
Eigen Value = 1.2609
Eigen Vector:
0.207
0.005
0.000
0.001
0.000
1.000
0.037
```

```
Iteration 24
----------
Eigen Value = 1.2500
Eigen Vector:
0.200
0.005
0.000
0.001
0.000
1.000
0.034

Iteration 25
----------
Eigen Value = 1.2400
Eigen Vector:
0.194
0.004
0.000
0.000
0.000
1.000
0.032

Iteration 26
----------
Eigen Value = 1.2308
Eigen Vector:
0.187
0.004
0.000
0.000
0.000
1.000
0.030

Iteration 27
----------
Eigen Value = 1.2222
Eigen Vector:
0.182
0.004
0.000
0.000
0.000
1.000
0.028
```

```
Iteration 28
----------
Eigen Value = 1.2143
Eigen Vector:
0.176
0.003
0.000
0.000
0.000
1.000
0.027

Iteration 29
----------
Eigen Value = 1.2069
Eigen Vector:
0.171
0.003
0.000
0.000
0.000
1.000
0.025

Iteration 30
----------
Eigen Value = 1.2000
Eigen Vector:
0.167
0.003
0.000
0.000
0.000
1.000
0.024

Iteration 31
----------
Eigen Value = 1.1935
Eigen Vector:
0.162
0.003
0.000
0.000
0.000
1.000
0.023
```

```
Iteration 32
----------
Eigen Value = 1.1875
Eigen Vector:
0.158
0.002
0.000
0.000
0.000
1.000
0.021

Iteration 33
----------
Eigen Value = 1.1818
Eigen Vector:
0.154
0.002
0.000
0.000
0.000
1.000
0.020

Iteration 34
----------
Eigen Value = 1.1765
Eigen Vector:
0.150
0.002
0.000
0.000
0.000
1.000
0.019

Iteration 35
----------
Eigen Value = 1.1714
Eigen Vector:
0.146
0.002
0.000
0.000
0.000
1.000
0.018
```

```
Iteration 36
----------
Eigen Value = 1.1667
Eigen Vector:
0.143
0.002
0.000
0.000
0.000
1.000
0.017

Iteration 37
----------
Eigen Value = 1.1622
Eigen Vector:
0.140
0.002
0.000
0.000
0.000
1.000
0.017

Iteration 38
----------
Eigen Value = 1.1579
Eigen Vector:
0.136
0.002
0.000
0.000
0.000
1.000
0.016

Iteration 39
----------
Eigen Value = 1.1538
Eigen Vector:
0.133
0.001
0.000
0.000
0.000
1.000
0.015
```

```
Iteration 40
----------
Eigen Value = 1.1500
Eigen Vector:
0.130
0.001
0.000
0.000
0.000
1.000
0.014

Iteration 41
----------
Eigen Value = 1.1463
Eigen Vector:
0.128
0.001
0.000
0.000
0.000
1.000
0.014

Iteration 42
----------
Eigen Value = 1.1429
Eigen Vector:
0.125
0.001
0.000
0.000
0.000
1.000
0.013

Iteration 43
----------
Eigen Value = 1.1395
Eigen Vector:
0.122
0.001
0.000
0.000
0.000
1.000
0.013
```

```
Iteration 44
----------
Eigen Value = 1.1364
Eigen Vector:
0.120
0.001
0.000
0.000
0.000
1.000
0.012

Iteration 45
----------
Eigen Value = 1.1333
Eigen Vector:
0.118
0.001
0.000
0.000
0.000
1.000
0.012

Iteration 46
----------
Eigen Value = 1.1304
Eigen Vector:
0.115
0.001
0.000
0.000
0.000
1.000
0.011

Iteration 47
----------
Eigen Value = 1.1277
Eigen Vector:
0.113
0.001
0.000
0.000
0.000
1.000
0.011
```

```
Iteration 48
----------
Eigen Value = 1.1250
Eigen Vector:
0.111
0.001
0.000
0.000
0.000
1.000
0.010

Iteration 49
----------
Eigen Value = 1.1224
Eigen Vector:
0.109
0.001
0.000
0.000
0.000
1.000
0.010

Iteration 50
----------
Eigen Value = 1.1200
Eigen Vector:
0.107
0.001
0.000
0.000
0.000
1.000
0.010

Iteration 51
----------
Eigen Value = 1.1176
Eigen Vector:
0.105
0.001
0.000
0.000
0.000
1.000
0.009
```

```
Iteration 52
----------
Eigen Value = 1.1154
Eigen Vector:
0.103
0.001
0.000
0.000
0.000
1.000
0.009

Iteration 53
----------
Eigen Value = 1.1132
Eigen Vector:
0.102
0.001
0.000
0.000
0.000
1.000
0.009

Iteration 54
----------
Eigen Value = 1.1111
Eigen Vector:
0.100
0.001
0.000
0.000
0.000
1.000
0.008

Iteration 55
----------
Eigen Value = 1.1091
Eigen Vector:
0.098
0.001
0.000
0.000
0.000
1.000
0.008
```

```
Iteration 56
----------
Eigen Value = 1.1071
Eigen Vector:
0.097
0.001
0.000
0.000
0.000
1.000
0.008

Iteration 57
----------
Eigen Value = 1.1053
Eigen Vector:
0.095
0.001
0.000
0.000
0.000
1.000
0.008

Iteration 58
----------
Eigen Value = 1.1034
Eigen Vector:
0.094
0.000
0.000
0.000
0.000
1.000
0.007

Iteration 59
----------
Eigen Value = 1.1017
Eigen Vector:
0.092
0.000
0.000
0.000
0.000
1.000
0.007
```

```
Iteration 60
----------
Eigen Value = 1.1000
Eigen Vector:
0.091
0.000
0.000
0.000
0.000
1.000
0.007

Iteration 61
----------
Eigen Value = 1.0984
Eigen Vector:
0.090
0.000
0.000
0.000
0.000
1.000
0.007

Iteration 62
----------
Eigen Value = 1.0968
Eigen Vector:
0.088
0.000
0.000
0.000
0.000
1.000
0.007

Iteration 63
----------
Eigen Value = 1.0952
Eigen Vector:
0.087
0.000
0.000
0.000
0.000
1.000
0.006
```

```
Iteration 64
----------
Eigen Value = 1.0938
Eigen Vector:
0.086
0.000
0.000
0.000
0.000
1.000
0.006

Iteration 65
----------
Eigen Value = 1.0923
Eigen Vector:
0.085
0.000
0.000
0.000
0.000
1.000
0.006

Iteration 66
----------
Eigen Value = 1.0909
Eigen Vector:
0.083
0.000
0.000
0.000
0.000
1.000
0.006

Iteration 67
----------
Eigen Value = 1.0896
Eigen Vector:
0.082
0.000
0.000
0.000
0.000
1.000
0.006
```

```
Iteration 68
----------
Eigen Value = 1.0882
Eigen Vector:
0.081
0.000
0.000
0.000
0.000
1.000
0.006

Iteration 69
----------
Eigen Value = 1.0870
Eigen Vector:
0.080
0.000
0.000
0.000
0.000
1.000
0.005

Iteration 70
----------
Eigen Value = 1.0857
Eigen Vector:
0.079
0.000
0.000
0.000
0.000
1.000
0.005

Iteration 71
----------
Eigen Value = 1.0845
Eigen Vector:
0.078
0.000
0.000
0.000
0.000
1.000
0.005
```

```
Iteration 72
----------
Eigen Value = 1.0833
Eigen Vector:
0.077
0.000
0.000
0.000
0.000
1.000
0.005

Iteration 73
----------
Eigen Value = 1.0822
Eigen Vector:
0.076
0.000
0.000
0.000
0.000
1.000
0.005

Iteration 74
----------
Eigen Value = 1.0811
Eigen Vector:
0.075
0.000
0.000
0.000
0.000
1.000
0.005

Iteration 75
----------
Eigen Value = 1.0800
Eigen Vector:
0.074
0.000
0.000
0.000
0.000
1.000
0.005

Iteration 76
---------
Eigen Value = 1.0789
Eigen Vector:
0.073
0.000
0.000
0.000
0.000
1.000
0.005

Iteration 77
---------
Eigen Value = 1.0779
Eigen Vector:
0.072
0.000
0.000
0.000
0.000
1.000
0.004

Iteration 78
---------
Eigen Value = 1.0769
Eigen Vector:
0.071
0.000
0.000
0.000
0.000
1.000
0.004
Largest Dominant Eigen Value Possible:  7.0
Smallest Dominant Eigen Value Possible:  1.076923076923077
```

```
from scipy.stats import norm
import statistics

mean1 = statistics.mean(d['Primary Level'])
print("Mean of the Pupil-Teacher Ratio of primary level -> " , round(mean1,3))

std1 = d['Primary Level'].std()
print("Standard deviation of the Pupil-Teacher Ratio of primary level -> " , round(std1,3))

var1 = print("Variance of the Pupil-Teacher Ratio of primary level -> ", round((std1*std1),3))

modev1 = statistics.mode(d['Primary Level'])
print("Mode of the Pupil-Teacher Ratio of primary level -> " , modev1)

medv1 = statistics.median(d['Primary Level'])
print("Median of the Pupil-Teacher Ratio of primary level -> " , medv1)

minv1 = min(d['Primary Level'])
print("Minimum value of the Pupil-Teacher Ratio of primary level -> " , minv1)

maxv1 = max(d['Primary Level'])
print("Maximum value of the Pupil-Teacher Ratio of primary level -> " , maxv1)
```

```
Mean of the Pupil-Teacher Ratio of primary level ->  15.571
Standard deviation of the Pupil-Teacher Ratio of primary level ->  7.323
Variance of the Pupil-Teacher Ratio of primary level ->  53.619
Mode of the Pupil-Teacher Ratio of primary level ->  8
Median of the Pupil-Teacher Ratio of primary level ->  14
Minimum value of the Pupil-Teacher Ratio of primary level ->  7
Maximum value of the Pupil-Teacher Ratio of primary level ->  26
```

```
mean2 = statistics.mean(d['Upper-Primary Level'])
print("Mean of the Pupil-Teacher Ratio of Upper-primary level -> " , round(mean2,3))

std2 = d['Upper-Primary Level'].std()
print("Standard deviation of the Pupil-Teacher Ratio of Upper-primary level -> " , round(std2,3))

var2 = print("Variance of the Pupil-Teacher Ratio of Upper-primary level -> ", round((std2*std2),3))

modev2 = statistics.mode(d['Upper-Primary Level'])
print("Mode of the Pupil-Teacher Ratio of Upper-primary level -> " , modev2)

medv2 = statistics.median(d['Upper-Primary Level'])
print("Median of the Pupil-Teacher Ratio of Upper-primary level -> " , medv2)

minv2 = min(d['Upper-Primary Level'])
print("Minimum value of the Pupil-Teacher Ratio of Upper-primary level -> " , minv2)

maxv2 = max(d['Upper-Primary Level'])
print("Maximum value of the Pupil-Teacher Ratio of Upper-primary level -> " , maxv2)
```

```
Mean of the Pupil-Teacher Ratio of Upper-primary level ->  10.714
Standard deviation of the Pupil-Teacher Ratio of Upper-primary level ->  4.03
Variance of the Pupil-Teacher Ratio of Upper-primary level ->  16.238
Mode of the Pupil-Teacher Ratio of Upper-primary level ->  9
Median of the Pupil-Teacher Ratio of Upper-primary level ->  9
Minimum value of the Pupil-Teacher Ratio of Upper-primary level ->  6
Maximum value of the Pupil-Teacher Ratio of Upper-primary level ->  17
```

```
mean3 = statistics.mean(d['Secondary Level'])
print("Mean of the Pupil-Teacher Ratio of Secondary Level -> " , round(mean3,3))

std3 = d['Secondary Level'].std()
print("Standard deviation of the Pupil-Teacher Ratio of Secondary Level -> " , round(std3,3))

var3 = print("Variance of the Pupil-Teacher Ratio of Secondary Level -> ", round((std3*std3),3))

modev3 = statistics.mode(d['Secondary Level'])
print("Mode of the Pupil-Teacher Ratio of Secondary Level -> " , modev3)

medv3 = statistics.median(d['Secondary Level'])
print("Median of the Pupil-Teacher Ratio of Secondary Level -> " , medv3)

minv3 = min(d['Secondary Level'])
print("Minimum value of the Pupil-Teacher Ratio of Secondary Level -> " , minv3)

maxv3 = max(d['Secondary Level'])
print("Maximum value of the Pupil-Teacher Ratio of Secondary Level -> " , maxv3)
```

```
Mean of the Pupil-Teacher Ratio of Secondary Level ->  17.429
Standard deviation of the Pupil-Teacher Ratio of Secondary Level ->  9.108
Variance of the Pupil-Teacher Ratio of Secondary Level ->  82.952
Mode of the Pupil-Teacher Ratio of Secondary Level ->  30
Median of the Pupil-Teacher Ratio of Secondary Level ->  14
Minimum value of the Pupil-Teacher Ratio of Secondary Level ->  7
Maximum value of the Pupil-Teacher Ratio of Secondary Level ->  30
```

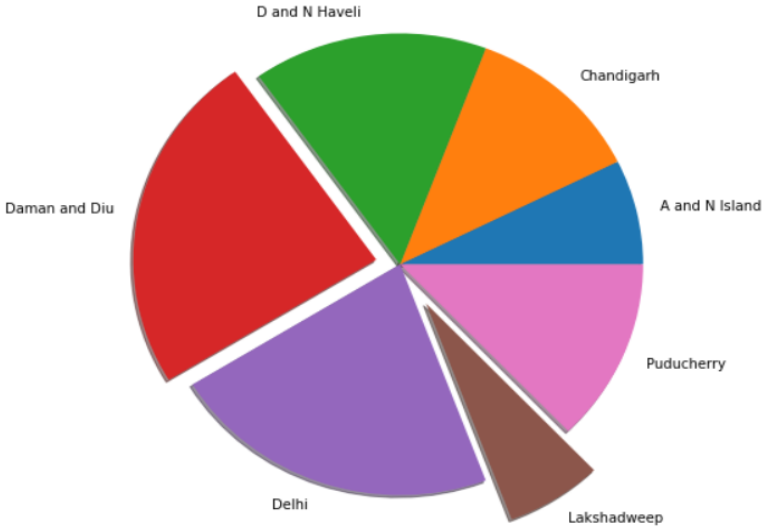
```
mean4 = statistics.mean(d['Senior secondary Level'])
print("Mean of the Pupil-Teacher Ratio of Senior secondary Level -> " , round(mean4,3))

std4 = d['Senior secondary Level'].std()
print("Standard deviation of the Pupil-Teacher Ratio of Senior secondary Level -> " , round(std4,3))

var4 = print("Variance of the Pupil-Teacher Ratio of Senior secondary Level -> ", round((std4*std4),3))

modev4 = statistics.mode(d['Senior secondary Level'])
print("Mode of the Pupil-Teacher Ratio of Senior secondary Level -> " , modev4)

medv4 = statistics.median(d['Senior secondary Level'])
print("Median of the Pupil-Teacher Ratio of Senior secondary Level -> " , medv4)

minv4 = min(d['Senior secondary Level'])
print("Minimum value of the Pupil-Teacher Ratio of Senior secondary Level -> " , minv4)

maxv4 = max(d['Senior secondary Level'])
print("Maximum value of the Pupil-Teacher Ratio of Senior secondary Level -> " , maxv4)
```

```
Mean of the Pupil-Teacher Ratio of Senior secondary Level ->  19.571
Standard deviation of the Pupil-Teacher Ratio of Senior secondary Level ->  7.091
Variance of the Pupil-Teacher Ratio of Senior secondary Level ->  50.286
Mode of the Pupil-Teacher Ratio of Senior secondary Level ->  16
Median of the Pupil-Teacher Ratio of Senior secondary Level ->  17
Minimum value of the Pupil-Teacher Ratio of Senior secondary Level ->  12
Maximum value of the Pupil-Teacher Ratio of Senior secondary Level ->  30
```

```
d.describe()
```

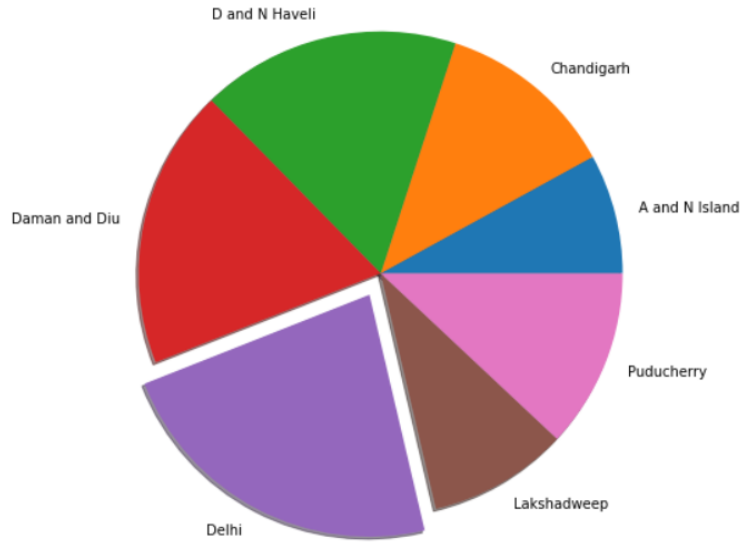
	Primary Level	Upper-Primary Level	Secondary Level	Senior secondary Level
count	7.000000	7.000000	7.000000	7.000000
mean	15.571429	10.714286	17.428571	19.571429
std	7.322503	4.029652	9.107820	7.091242
min	7.000000	6.000000	7.000000	12.000000
25%	10.500000	8.000000	12.000000	14.500000
50%	14.000000	9.000000	14.000000	17.000000
75%	20.500000	13.500000	23.500000	24.500000
max	26.000000	17.000000	30.000000	30.000000

```
a = d['Primary Level']
plt.pie(a , labels = d['U.T.'], radius = 2 , shadow =True , explode = (0,0,0,0.2,0,0.4,0))
plt.show()
```



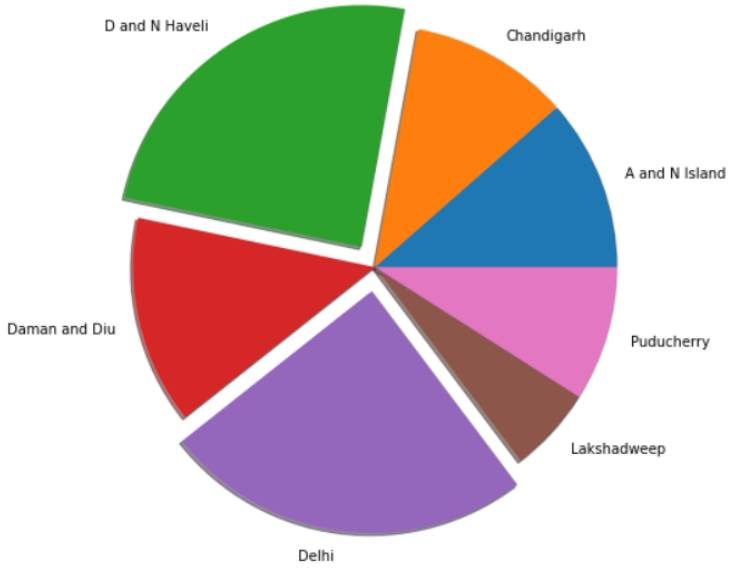
In the above pie chart we are seeing the amount(number) of the pupil teacher ratio of the different Union Territories. This graph is showing the details of the Primary Level Pupil Teacher Ratio for the year 2015-16. ¶

```
a = d['Upper-Primary Level']
plt.pie(a , labels = d['U.T.'], radius = 2 , shadow =True , explode = (0,0,0,0,0.2,0.,0))
plt.show()
```



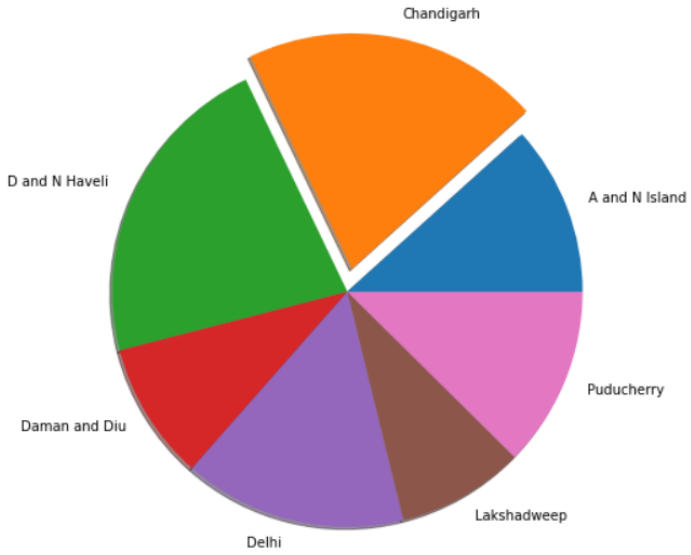
In the above pie chart we are seeing the amount(number) of the pupil teacher ratio of the different Union Territories. This graph is showing the details of the Upper Primary Level Pupil Teacher Ratio for the year 2015-16

```
a = d['Secondary Level']
plt.pie(a , labels = d['U.T.'], radius = 2 , shadow =True , explode = (0,0,0.2,0,0.2,0.,0))
plt.show()
```



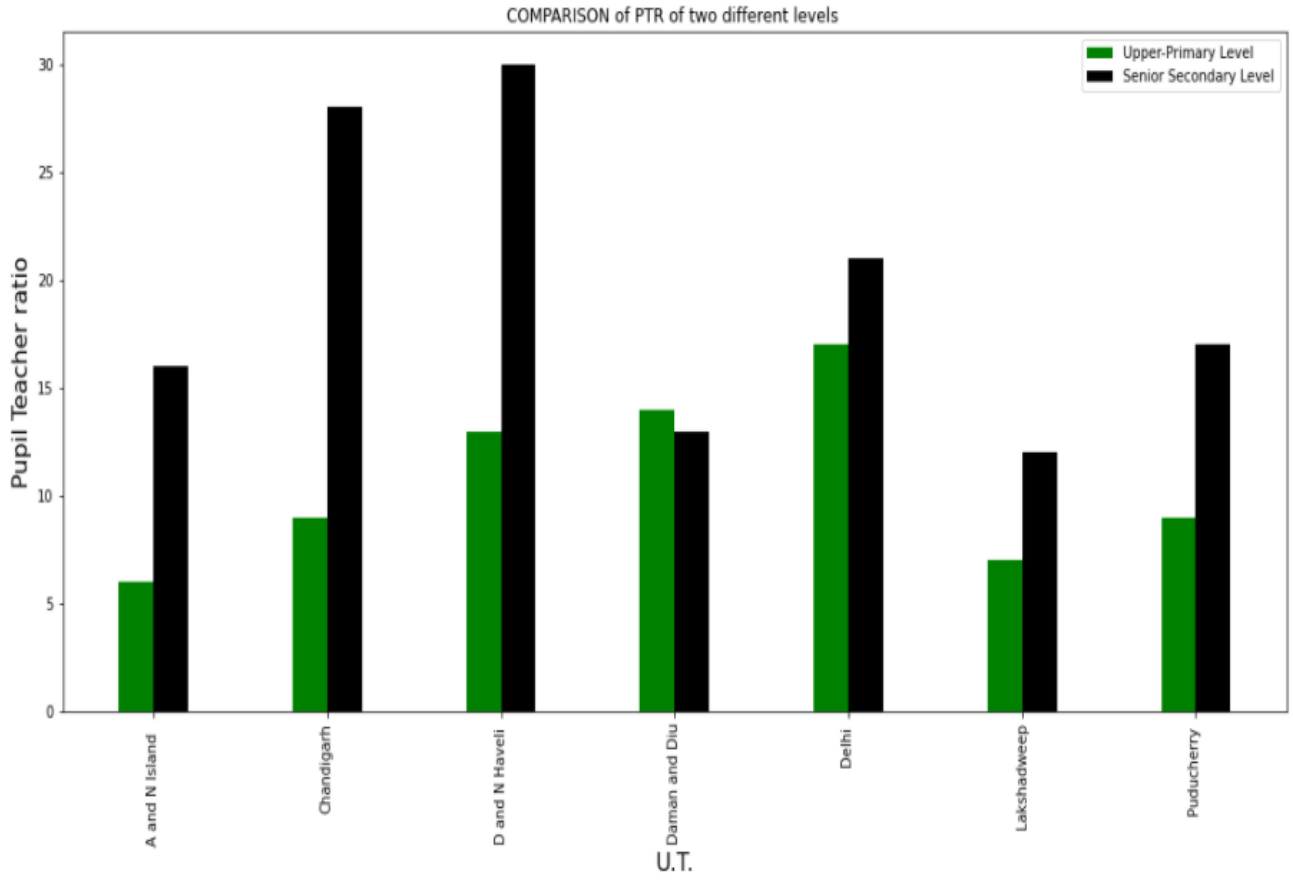
In the above pie chart we are seeing the amount(number) of the pupil teacher ratio of the different Union Territories. This graph is showing the details of the Secondary Level Pupil Teacher Ratio for the year 2015-16. ¶

```
a = d['Senior secondary Level']
plt.pie(a , labels = d['U.T.'], radius = 2 , shadow =True , explode = (0,0.2,0,0,0,0.,0))
plt.show()
```



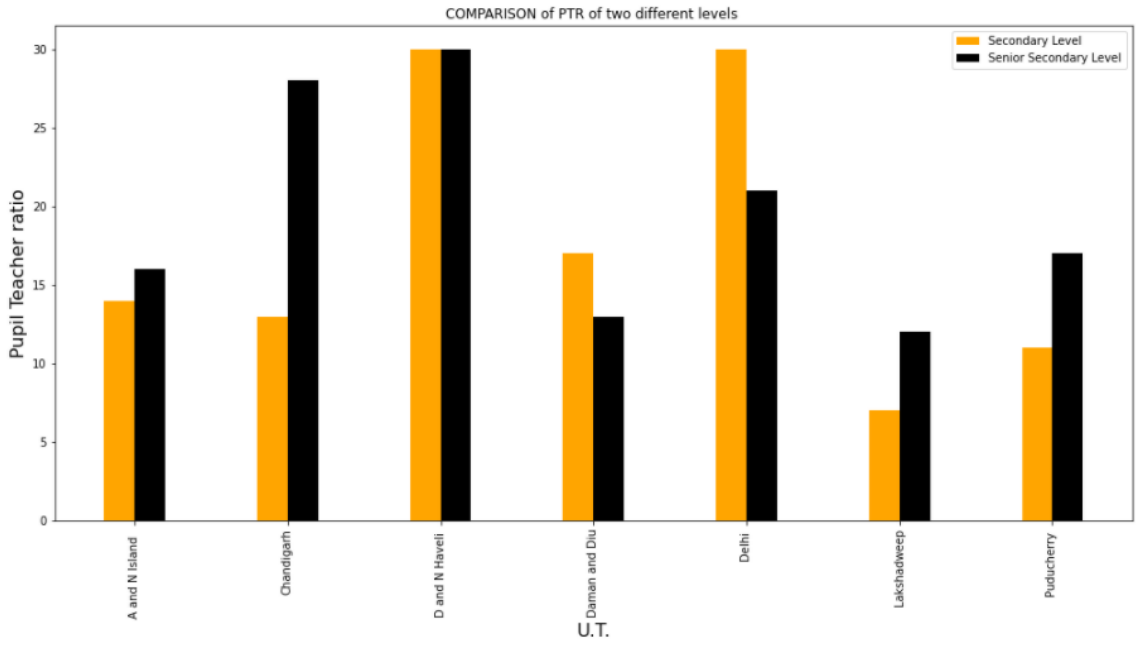
In the above pie chart we are seeing the amount(number) of the pupil teacher ratio of the different Union Territories. This graph is showing the details of the Senior Secondary Level Pupil Teacher Ratio for the year 2015-16

```
n = 29

ind = np.arange(7)
plt.figure(figsize = (17,8))
width = 0.2

x = d['U.T.']
y = d['Upper-Primary Level']
z = d['Senior secondary Level']

plt.bar(ind,y, width , label = 'Upper-Primary Level' , color = 'green')
plt.bar(ind+width , z ,width , label = 'Senior Secondary Level' , color = 'black')

plt.xlabel("U.T." ,size =16)
plt.ylabel("Pupil Teacher ratio" ,size =16)
plt.xticks(ind +width/2 , (x) , rotation = 90)
plt.title("COMPARISON of PTR of two different levels ")
plt.legend()
plt.show()
```



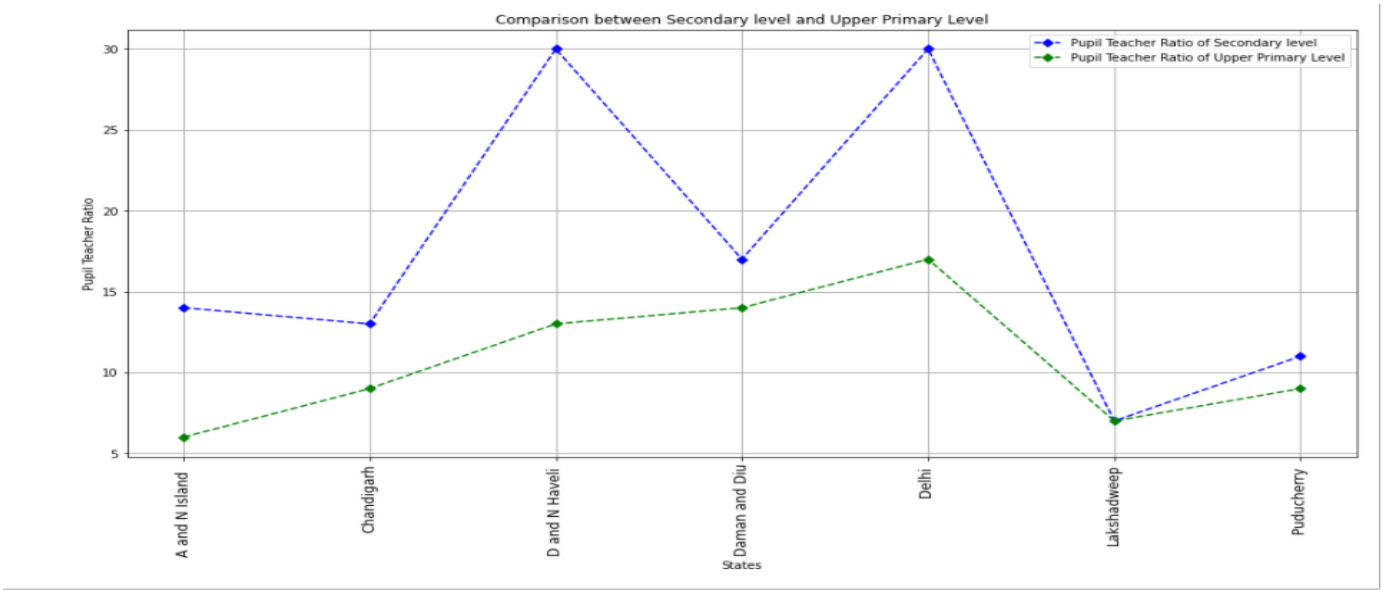
Here in this graph we are comparing the PTR of Senior Secondary Level and Upper Primary Level for the year 2015-16.

```
n = 29

ind = np.arange(7)
plt.figure(figsize = (17,8))
width = 0.2

x = d['U.T.']
y = d['Secondary Level']
z = d['Senior secondary Level']

plt.bar(ind,y, width , label = 'Secondary Level' , color = 'orange')
plt.bar(ind+width , z ,width , label = 'Senior Secondary Level' , color = 'black')

plt.xlabel("U.T." ,size =16)
plt.ylabel("Pupil Teacher ratio" ,size =16)
plt.xticks(ind +width/2 , (x) , rotation = 90)
plt.title("COMPARISON of PTR of two different levels ")
plt.legend()
plt.show()
```



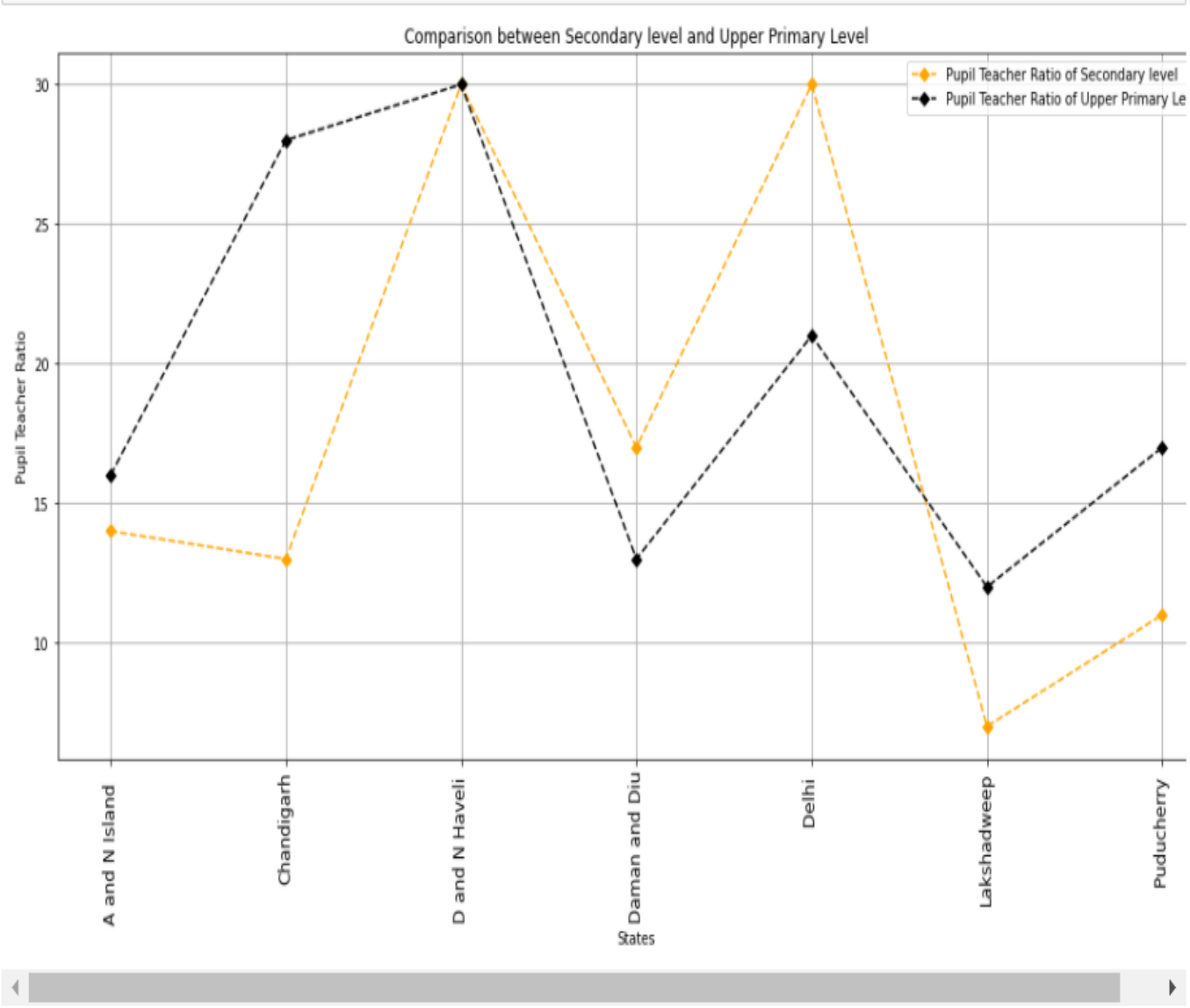
Here in this graph we are comparing the PTR of Secondary and Senior Secondary Level for the year 2015-16.

```
plt.figure(figsize = (17 , 8))
x = d['U.T.']
plt.plot(d['U.T.'] , d['Secondary Level'] , color = 'blue' ,
         marker = 'D' , markersize = 5 , markerfacecolor = 'blue' ,
         markeredgecolor = "blue" , linestyle = 'dashed' ,  label = 'Pupil Teacher Ratio of Secondary level')
plt.plot(d['U.T.'] , d['Upper-Primary Level'] , color = 'green' ,
         marker = 'D' , markersize = 5 , markerfacecolor = 'green' ,
         markeredgecolor = "green" , linestyle = 'dashed' , label = 'Pupil Teacher Ratio of Upper Primary Level')
plt.xticks(x , size = 12 , rotation = 90)
plt.grid(linewidth =1)
plt.title("Comparison between Secondary level and Upper Primary Level")
plt.xlabel("States")
plt.ylabel("Pupil Teacher Ratio")
plt.legend()
plt.show()
```



Here in this graph we are comparing the PTR of Secondary Level and Upper Primary Level for the year 2015-16.

```
plt.figure(figsize = (17 , 8))
x = d['U.T.']
plt.plot(d['U.T.'] , d['Secondary Level'] , color = 'orange' ,
        marker = 'D' , markersize = 5 , markerfacecolor = 'orange' ,
        markeredgecolor = "orange" , linestyle = 'dashed' ,  label = 'Pupil Teacher Ratio of Secondary level')
plt.plot(d['U.T.'] , d['Senior secondary Level'] , color = 'black' ,
        marker = 'D' , markersize = 5 , markerfacecolor = 'black' ,
        markeredgecolor = "black" , linestyle = 'dashed' , label = 'Pupil Teacher Ratio of Upper Primary Level')
plt.xticks(x , size = 12 , rotation = 90)
plt.grid(linewidth =1)
plt.title("Comparison between Secondary level and Upper Primary Level")
plt.xlabel("States")
plt.ylabel("Pupil Teacher Ratio")
plt.legend()
plt.show()
```



Here in this graph we are comparing the PTR of Secondary Level and Upper Primary Level for the year 2015-16.

Now the code for Pupil Teacher Ratio of only States

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
from sklearn.linear_model import LinearRegression
```

```
df = pd.read_csv("Pupilrat.csv")
df
```

	States	Primary Level	Upper-Primary Level	Secondary Level	Senior Secondary level
0	Andhra Pradesh	21	16	20	71
1	Arunachal Pradesh	12	7	22	37
2	Assam	21	13	14	20
3	Bihar	36	24	66	39
4	Chhattisgarh	20	17	33	21
5	Goa	20	16	13	18
6	Gujarat	19	13	34	29
7	Haryana	20	13	15	15
8	Himachal Pradesh	12	10	18	14
9	Jammu And Kashmir	9	6	15	29
10	Jharkhand	27	19	62	78
11	Karnataka	19	13	16	30
12	Kerala	18	14	17	21
13	Madhya Pradesh	20	18	39	38
14	Maharashtra	24	17	23	44
15	Manipur	12	8	12	19
16	Meghalaya	21	13	12	21
17	Mizoram	14	6	9	15
18	Nagaland	10	6	15	21
19	Odisha	17	14	20	45
20	Punjab	18	12	16	26
21	Rajasthan	17	10	21	32
22	Sikkim	5	5	17	15
23	Tamil Nadu	18	15	21	25
24	Telangana	23	15	22	47
25	Tripura	10	8	28	11
26	Uttar Pradesh	39	31	56	97
27	Uttarakhand	18	17	16	25
28	West Bengal	25	27	39	57

```
import numpy as np
from scipy.stats import norm
import statistics

mean1 = statistics.mean(df['Primary Level'])
print("Mean of the Pupil-Teacher Ratio of primary level -> " , round(mean1,3))

std1 = df['Primary Level'].std()
print("Standard deviation of the Pupil-Teacher Ratio of primary level -> " , round(std1,3))

var1 = print("Variance of the Pupil-Teacher Ratio of primary level -> ", round((std1*std1),3))

modev1 = statistics.mode(df['Primary Level'])
print("Mode of the Pupil-Teacher Ratio of primary level -> " , modev1)

medv1 = statistics.median(df['Primary Level'])
print("Median of the Pupil-Teacher Ratio of primary level -> " , medv1)

minv1 = min(df['Primary Level'])
print("Minimum value of the Pupil-Teacher Ratio of primary level -> " , minv1)

maxv1 = max(df['Primary Level'])
print("Maximum value of the Pupil-Teacher Ratio of primary level -> " , maxv1)
```

```
Mean of the Pupil-Teacher Ratio of primary level ->  18.793
Standard deviation of the Pupil-Teacher Ratio of primary level ->  7.287
Variance of the Pupil-Teacher Ratio of primary level ->  53.099
Mode of the Pupil-Teacher Ratio of primary level ->  20
Median of the Pupil-Teacher Ratio of primary level ->  19
Minimum value of the Pupil-Teacher Ratio of primary level ->  5
Maximum value of the Pupil-Teacher Ratio of primary level ->  39
```

```
mean2 = statistics.mean(df['Upper-Primary Level'])
print("Mean of the Pupil-Teacher Ratio of Upper-primary level -> " , round(mean2,3))

std2 = df['Upper-Primary Level'].std()
print("Standard deviation of the Pupil-Teacher Ratio of Upper-primary level -> " , round(std2,3))

var2 = print("Variance of the Pupil-Teacher Ratio of Upper-primary level -> ", round((std2*std2),3))

modev2 = statistics.mode(df['Upper-Primary Level'])
print("Mode of the Pupil-Teacher Ratio of Upper-primary level -> " , modev2)

medv2 = statistics.median(df['Upper-Primary Level'])
print("Median of the Pupil-Teacher Ratio of Upper-primary level -> " , medv2)

minv2 = min(df['Upper-Primary Level'])
print("Minimum value of the Pupil-Teacher Ratio of Upper-primary level -> " , minv2)

maxv2 = max(df['Upper-Primary Level'])
print("Maximum value of the Pupil-Teacher Ratio of Upper-primary level -> " , maxv2)
```

```
Mean of the Pupil-Teacher Ratio of Upper-primary level ->  13.897
Standard deviation of the Pupil-Teacher Ratio of Upper-primary level ->  6.184
Variance of the Pupil-Teacher Ratio of Upper-primary level ->  38.239
Mode of the Pupil-Teacher Ratio of Upper-primary level ->  13
Median of the Pupil-Teacher Ratio of Upper-primary level ->  13
Minimum value of the Pupil-Teacher Ratio of Upper-primary level ->  5
Maximum value of the Pupil-Teacher Ratio of Upper-primary level ->  31
```

```
mean3 = statistics.mean(df['Secondary Level'])
print("Mean of the Pupil-Teacher Ratio of Secondary Level  -> " , round(mean3,3))

std3 = df['Secondary Level'].std()
print("Standard deviation of the Pupil-Teacher Ratio of Secondary Level -> " , round(std3,3))

var3 = print("Variance of the Pupil-Teacher Ratio of Secondary Level -> ", round((std3*std3),3))

modev3 = statistics.mode(df['Secondary Level'])
print("Mode of the Pupil-Teacher Ratio of Secondary Level -> " , modev3)

medv3 = statistics.median(df['Secondary Level'])
print("Median of the Pupil-Teacher Ratio of Secondary Level -> " , medv3)

minv3 = min(df['Secondary Level'])
print("Minimum value of the Pupil-Teacher Ratio of Secondary Level -> " , minv3)

maxv3 = max(df['Secondary Level'])
print("Maximum value of the Pupil-Teacher Ratio of Secondary Level -> " , maxv3)
```

```
Mean of the Pupil-Teacher Ratio of Secondary Level  ->  24.517
Standard deviation of the Pupil-Teacher Ratio of Secondary Level ->  14.92
Variance of the Pupil-Teacher Ratio of Secondary Level ->  222.616
Mode of the Pupil-Teacher Ratio of Secondary Level ->  15
Median of the Pupil-Teacher Ratio of Secondary Level ->  20
Minimum value of the Pupil-Teacher Ratio of Secondary Level ->  9
Maximum value of the Pupil-Teacher Ratio of Secondary Level ->  66
```

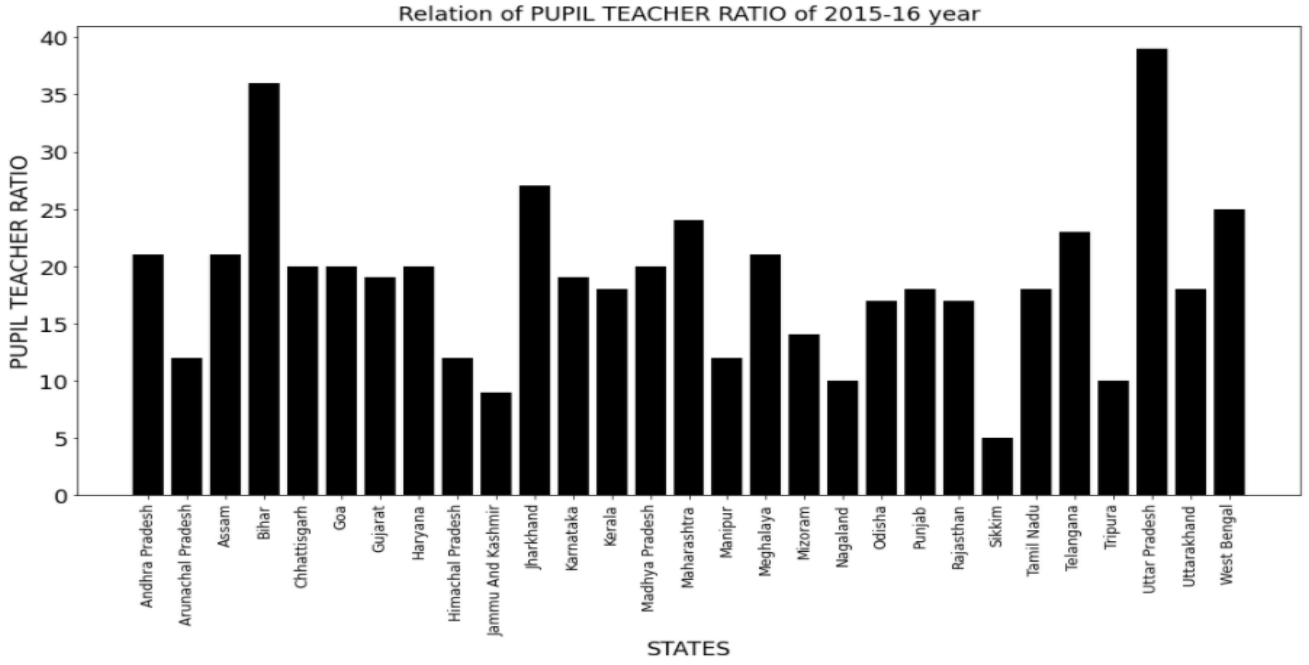
```
mean4 = statistics.mean(df['Senior Secondary level'])
print("Mean of the Pupil-Teacher Ratio of Senior Secondary level -> " , round(mean4,3))

std4 = df['Senior Secondary level'].std()
print("Standard deviation of the Pupil-Teacher Ratio of Senior Secondary level -> " , round(std4,3))

var4 = print("Variance of the Pupil-Teacher Ratio of Senior Secondary level -> ", round((std4*std4),3))

modev4 = statistics.mode(df['Senior Secondary level'])
print("Mode of the Pupil-Teacher Ratio of Senior Secondary level -> " , modev4)

medv4 = statistics.median(df['Senior Secondary level'])
print("Median of the Pupil-Teacher Ratio of Senior Secondary level  -> " , medv4)

minv4 = min(df['Senior Secondary level'])
print("Minimum value of the Pupil-Teacher Ratio of Senior Secondary level-> " , minv4)

maxv4 = max(df['Senior Secondary level'])
print("Maximum value of the Pupil-Teacher Ratio ofSenior Secondary level -> " , maxv4)
```

```
Mean of the Pupil-Teacher Ratio of Senior Secondary level ->  33.103
Standard deviation of the Pupil-Teacher Ratio of Senior Secondary level ->  20.599
Variance of the Pupil-Teacher Ratio of Senior Secondary level ->  424.31
Mode of the Pupil-Teacher Ratio of Senior Secondary level ->  21
Median of the Pupil-Teacher Ratio of Senior Secondary level  ->  26
Minimum value of the Pupil-Teacher Ratio of Senior Secondary level->  11
Maximum value of the Pupil-Teacher Ratio ofSenior Secondary level ->  97
```

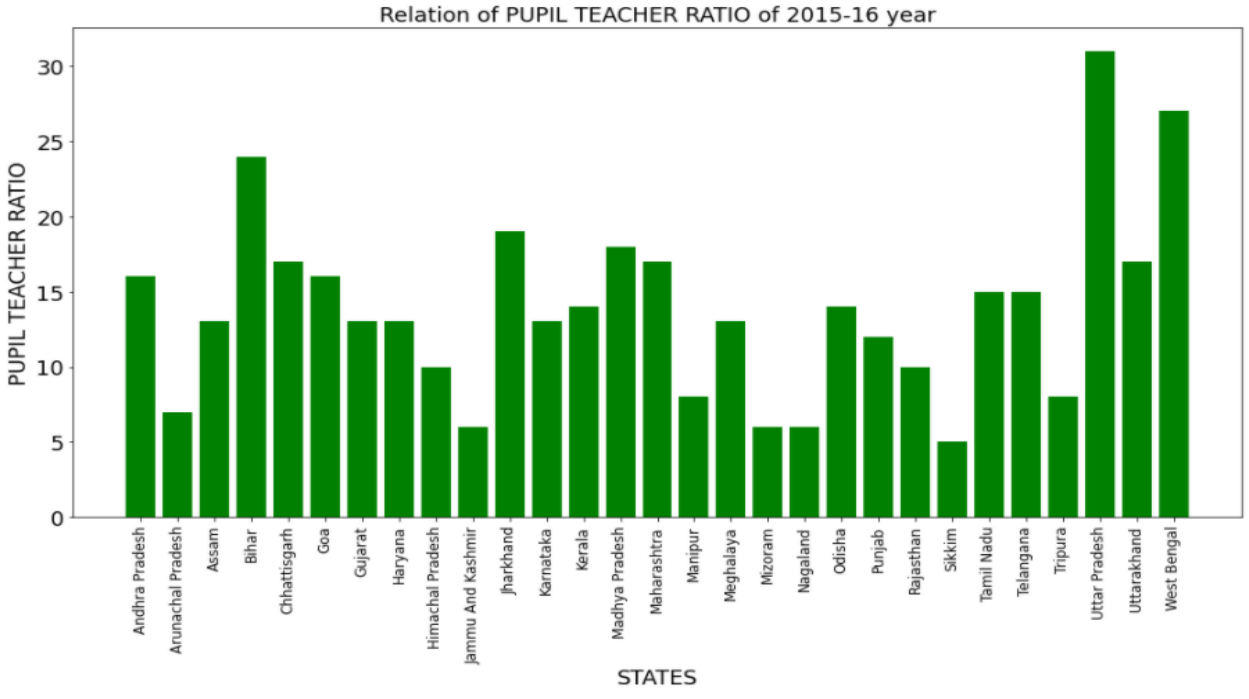
```
df.describe()
```

	Primary Level	Upper-Primary Level	Secondary Level	Senior Secondary level
count	29.000000	29.000000	29.000000	29.000000
mean	18.793103	13.896552	24.517241	33.103448
std	7.286873	6.183762	14.920314	20.598795
min	5.000000	5.000000	9.000000	11.000000
25%	14.000000	10.000000	15.000000	20.000000
50%	19.000000	13.000000	20.000000	26.000000
75%	21.000000	17.000000	28.000000	39.000000
max	39.000000	31.000000	66.000000	97.000000

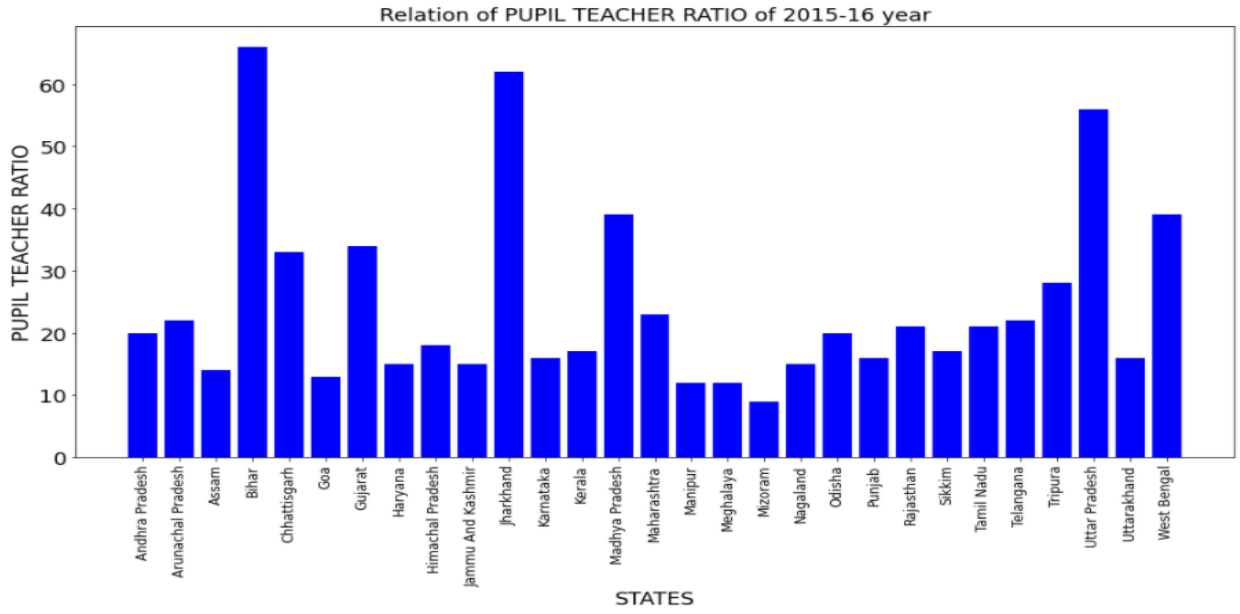
```
import matplotlib.pyplot as plt

fig = plt.figure(figsize = (17,8))         # Assigning the dimensions of the graph.


plt.xlabel('STATES' , size=18)             # Labelling the X axis.
plt.ylabel('PUPIL TEACHER RATIO' , size=18)      # Labelling the Y axis.
plt.title("Relation of PUPIL TEACHER RATIO of 2015-16 year" , size=18)
plt.bar(df['States'],df['Primary Level'],color = 'black')           # Assigning the type of the graph i.e. bar graph with x and y
plt.yticks(size=18)
plt.xticks(size=12,rotation=90)
plt.show()
```



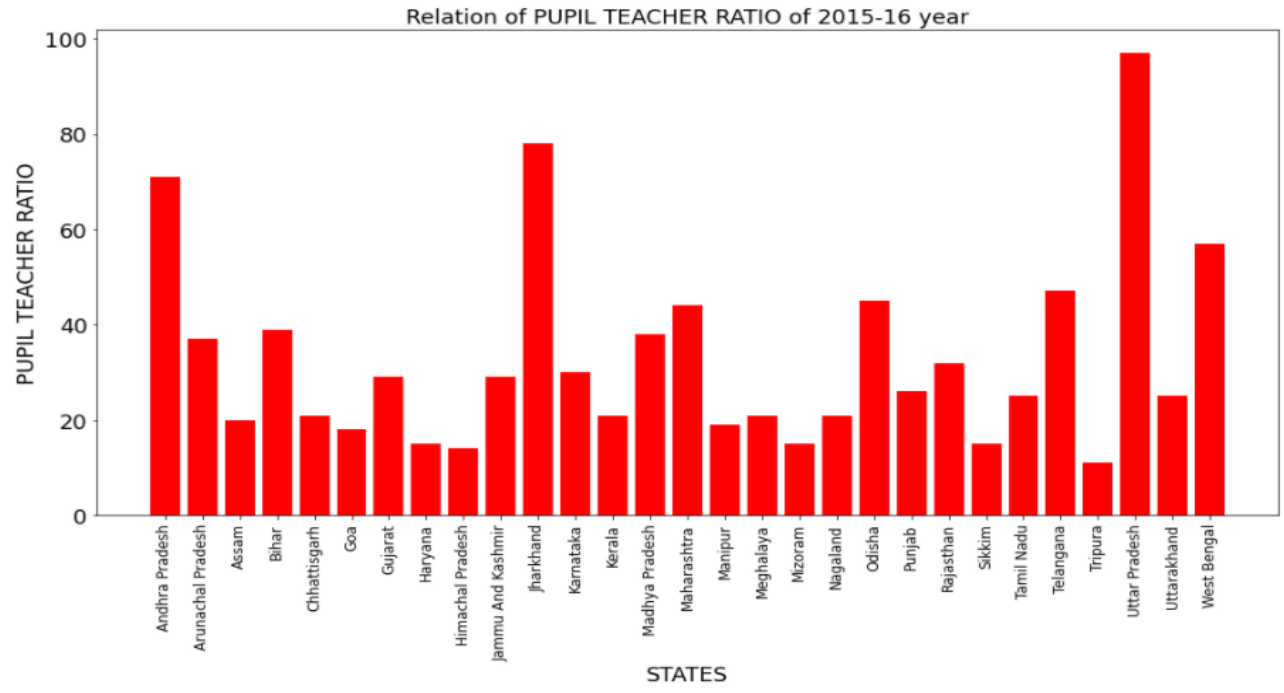

```
fig = plt.figure(figsize = (17,8))         # Assigning the dimensions of the graph.

plt.xlabel('STATES' , size=18)             # Labelling the X axis.
plt.ylabel('PUPIL TEACHER RATIO' , size=18)      # Labelling the Y axis.
plt.title("Relation of PUPIL TEACHER RATIO of 2015-16 year" , size=18)
plt.bar(df['States'],df['Upper-Primary Level'],color = 'green')           # Assigning the type of the graph i.e. bar graph with x
plt.yticks(size=18)
plt.xticks(size=12,rotation=90)
plt.show()
```



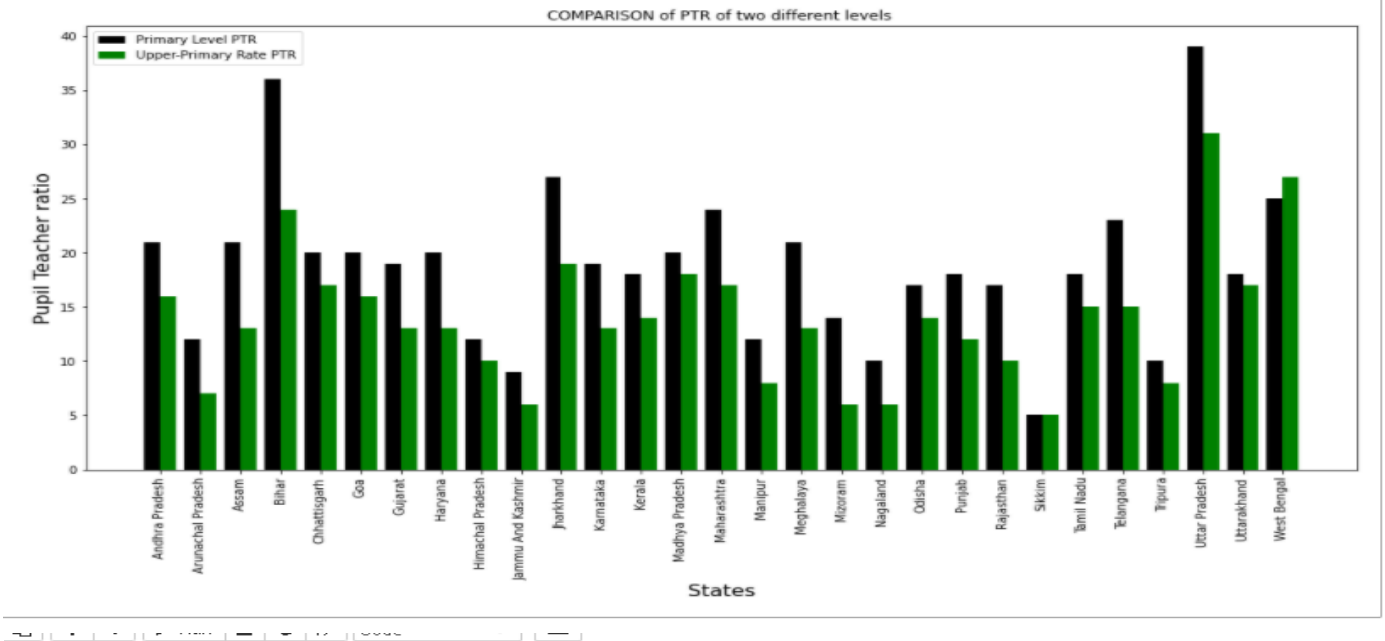
```
fig = plt.figure(figsize = (17,8))          # Assigning the dimensions of the graph.

plt.xlabel('STATES' , size=18)              # Labelling the X axis.
plt.ylabel('PUPIL TEACHER RATIO' , size=18)      # Labelling the Y axis.
plt.title("Relation of PUPIL TEACHER RATIO of 2015-16 year" , size=18)
plt.bar(df['States'],df['Secondary Level'],color = 'blue')          # Assigning the type of the graph i.e. bar graph with x and
plt.yticks(size=18)
plt.xticks(size=12,rotation=90)
plt.show()
```



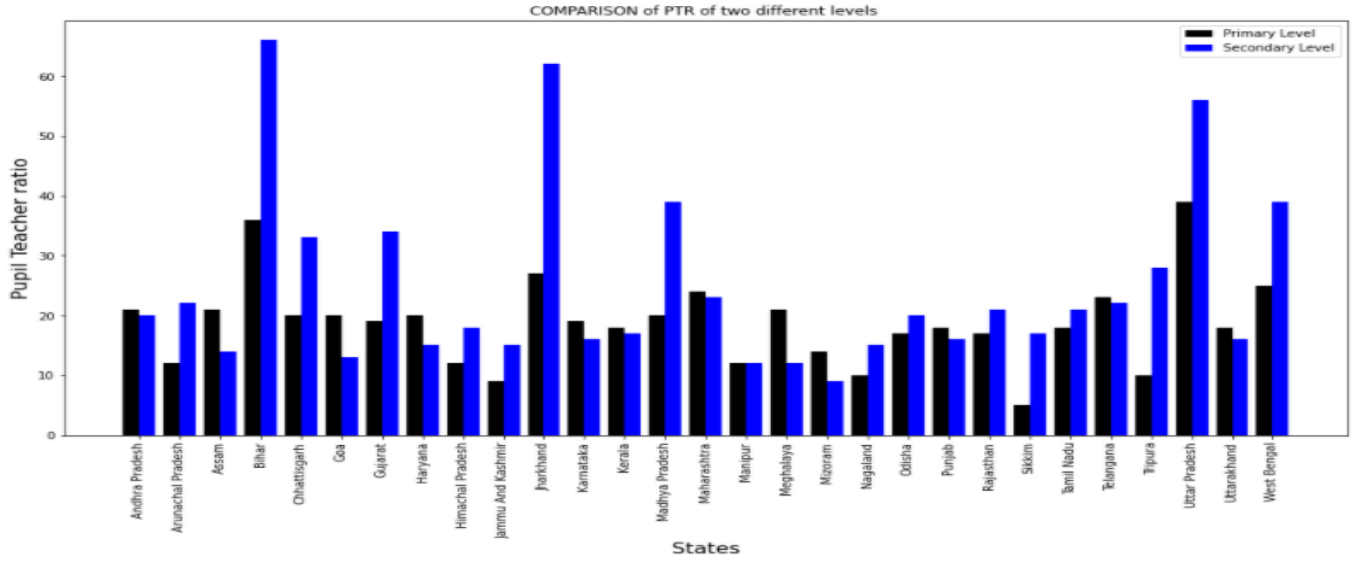

```
fig = plt.figure(figsize = (17,8))          # Assigning the dimensions of the graph.

plt.xlabel('STATES' , size=18)              # Labelling the X axis.
plt.ylabel('PUPIL TEACHER RATIO' , size=18)      # Labelling the Y axis.
plt.title("Relation of PUPIL TEACHER RATIO of 2015-16 year" , size=18)
plt.bar(df['States'],df['Senior Secondary level'],color = 'red')          # Assigning the type of the graph i.e. bar graph with
plt.yticks(size=18)
plt.xticks(size=12,rotation=90)
plt.show()
```



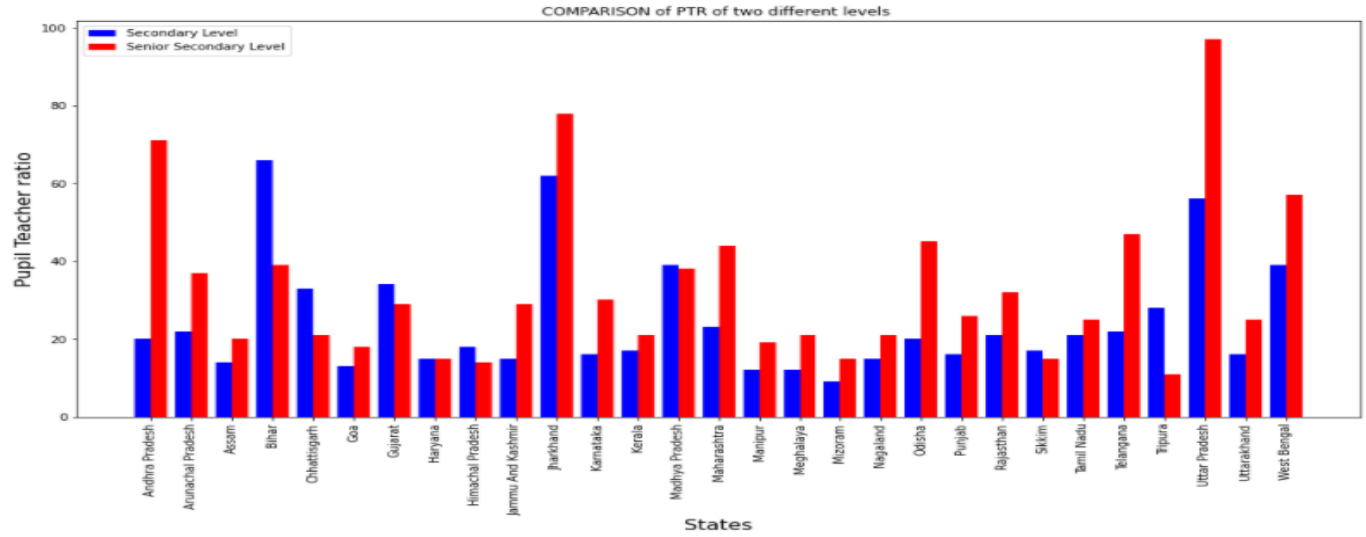
```
n = 29

ind = np.arange(n)
plt.figure(figsize = (17,8))
width = 0.4

x = df['States']
y = df['Primary Level']
z = df['Upper-Primary Level']

plt.bar(ind,y, width , label = 'Primary Level PTR ' , color = 'black')
plt.bar(ind+width , z ,width , label = 'Upper-Primary Rate PTR' , color = 'green')

plt.xlabel("States" ,size =16)
plt.ylabel("Pupil Teacher ratio" ,size =16)
plt.xticks(ind +width/2 , (x) , rotation = 90)
plt.title("COMPARISON of PTR of two different levels ")
plt.legend()
plt.show()
```



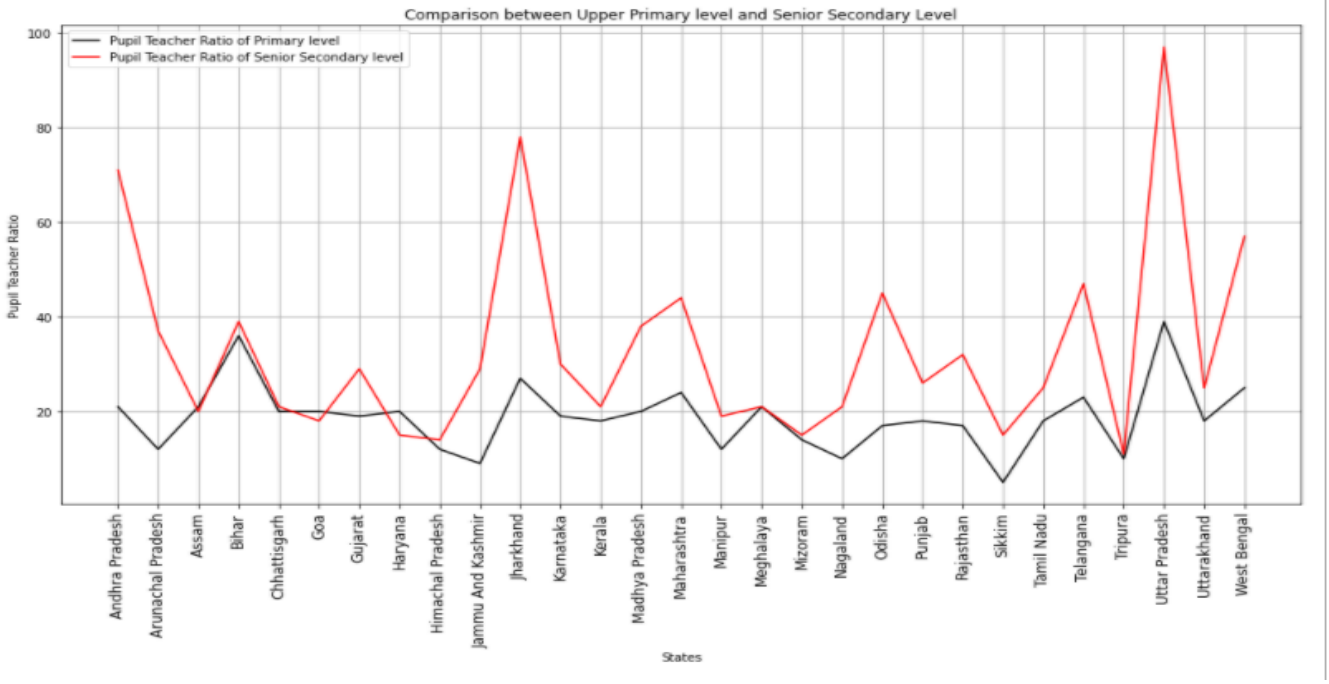

```
n = 29

ind = np.arange(n)
plt.figure(figsize = (17,8))
width = 0.4

x = df['States']
y = df['Primary Level']
z = df['Secondary Level']

plt.bar(ind,y, width , label = 'Primary Level' , color = 'black')
plt.bar(ind+width , z ,width , label = 'Secondary Level' , color = 'blue')


plt.xlabel("States" ,size =16)
plt.ylabel("Pupil Teacher ratio" ,size =16)
plt.xticks(ind +width/2 , (x) , rotation = 90)
plt.title("COMPARISON of PTR of two different levels ")
plt.legend()
plt.show()
```



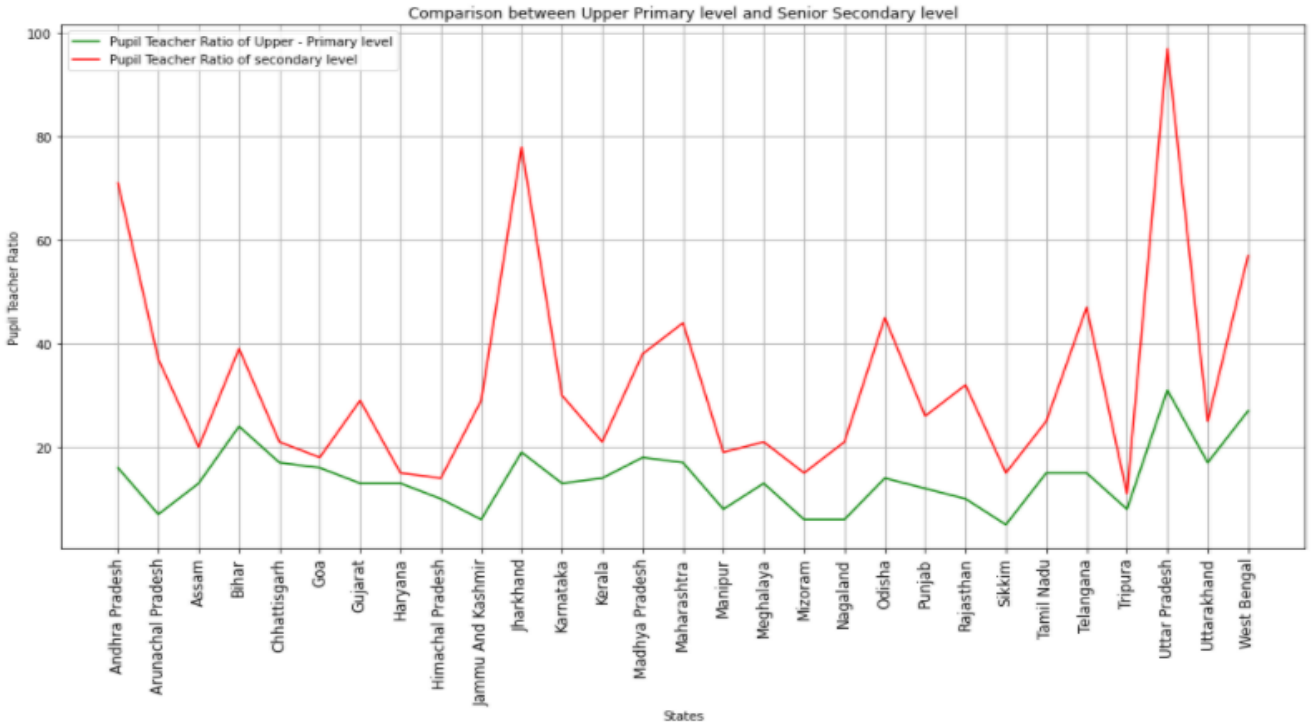
```
n = 29

ind = np.arange(n)
plt.figure(figsize = (17,8))
width = 0.4

x = df['States']
y = df['Secondary Level']
z = df['Senior Secondary level']

plt.bar(ind,y, width , label = 'Secondary Level' , color = 'blue')
plt.bar(ind+width , z ,width , label = 'Senior Secondary Level' , color = 'red')

plt.xlabel("States" ,size =16)
plt.ylabel("Pupil Teacher ratio" ,size =16)
plt.xticks(ind +width/2 , (x) , rotation = 90)
plt.title("COMPARISON of PTR of two different levels ")
plt.legend()
plt.show()
```



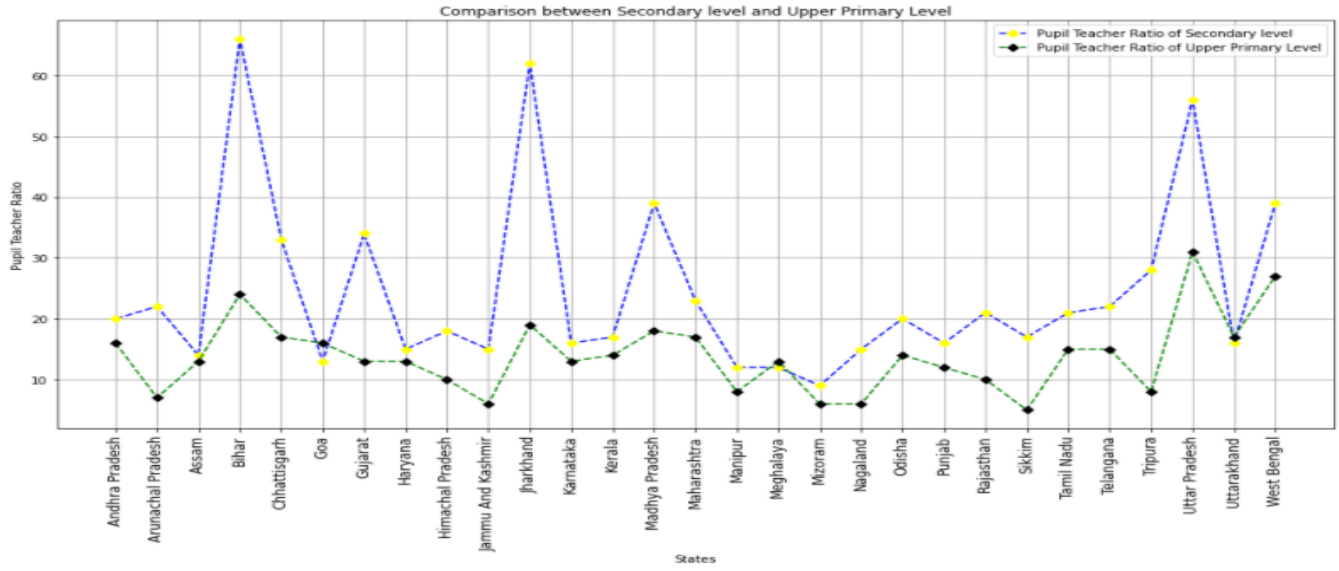

```
plt.figure(figsize = (17,8))
x = df['States']
plt.plot(df['States'] , df['Primary Level'] , color = 'black' , label = 'Pupil Teacher Ratio of Primary level')
plt.plot(df['States'] , df['Senior Secondary level'] , color = 'red' , label = 'Pupil Teacher Ratio of Senior Secc
plt.xticks(x , size = 12 , rotation = 90)
plt.grid(linewidth =1)
plt.title("Comparison between Upper Primary level and Senior Secondary Level")
plt.xlabel("States")
plt.ylabel("Pupil Teacher Ratio")
plt.legend()
plt.show()
```



```
plt.figure(figsize = (17, 8))
x = df['States']
plt.plot(df['States'] , df['Upper-Primary Level'] , color = 'green' , label = 'Pupil Teacher Ratio of Upper - Prim
plt.plot(df['States'] , df['Senior Secondary level'] , color = 'red' , label = 'Pupil Teacher Ratio of secondary l
plt.xticks(x , size = 12 , rotation = 90)
plt.grid(linewidth =1)
plt.title("Comparison between Upper Primary level and Senior Secondary level")
plt.xlabel("States")
plt.ylabel("Pupil Teacher Ratio")
plt.legend()
plt.show()
```



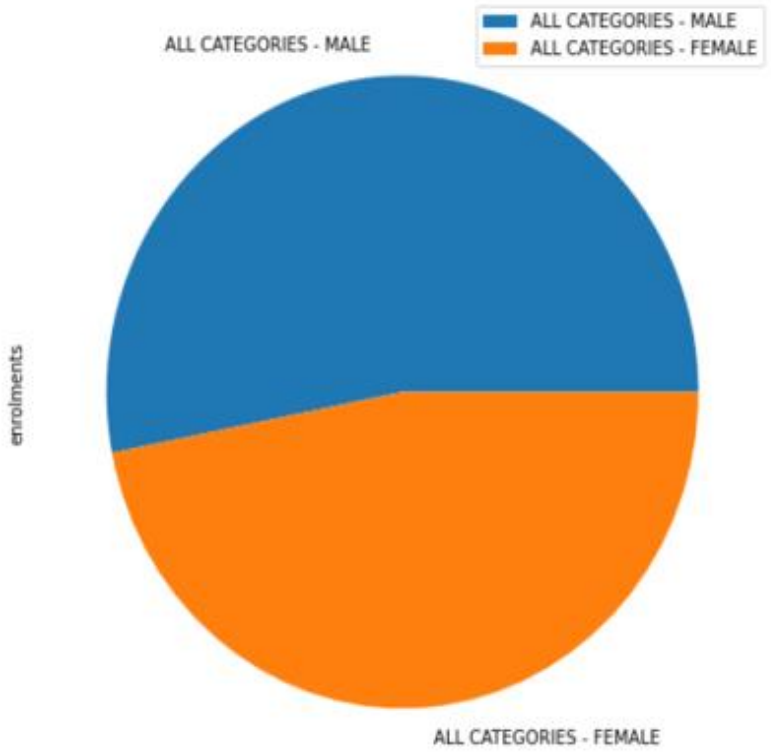

```
plt.figure(figsize = (17 , 8))
x = df['States']
plt.plot(df['States'] , df['Secondary Level'] , color = 'blue' ,
         marker = 'D' , markersize = 5 , markerfacecolor = 'yellow' ,
         markeredgecolor = "yellow" , linestyle = 'dashed' ,  label = 'Pupil Teacher Ratio of Secondary level')
plt.plot(df['States'] , df['Upper-Primary Level'] , color = 'green' ,
         marker = 'D' , markersize = 5 , markerfacecolor = 'black' ,
         markeredgecolor = "black" , linestyle = 'dashed' , label = 'Pupil Teacher Ratio of Upper Primary Level')
plt.xticks(x , size = 12 , rotation = 90)
plt.grid(linewidth =1)
plt.title("Comparison between Secondary level and Upper Primary Level")
plt.xlabel("States")
plt.ylabel("Pupil Teacher Ratio")
plt.legend()
plt.show()
```



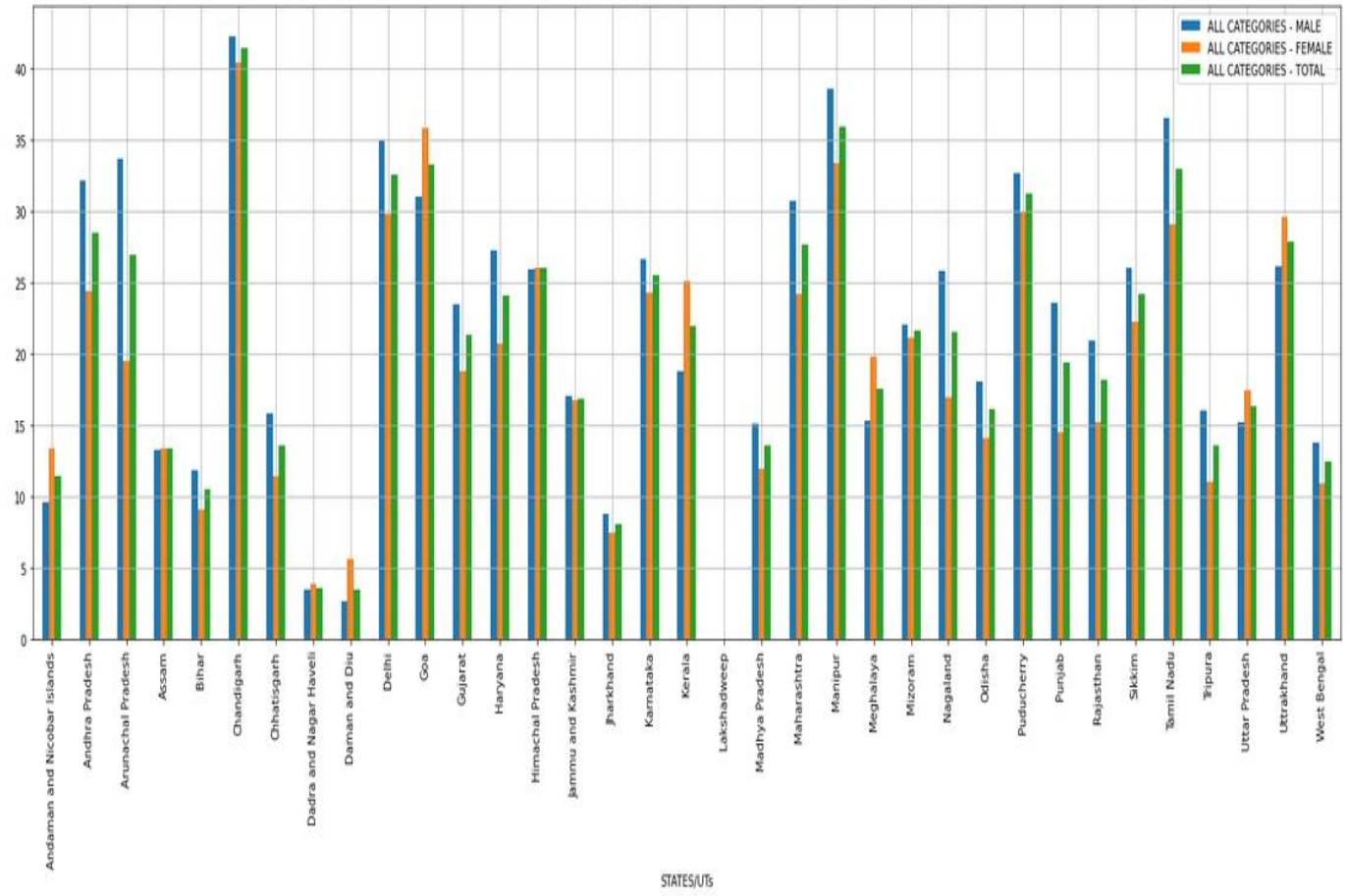
3. <u>To check the situation in GER of Higher Education.</u>

GRAPHS:

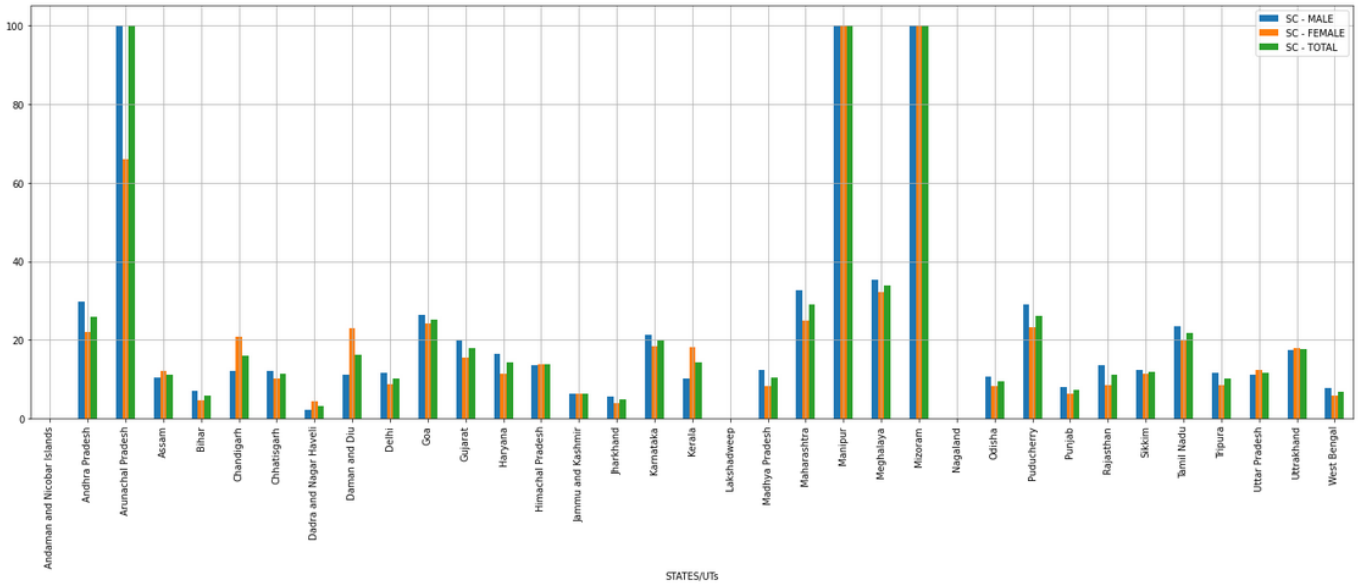
```
In [4]: E=data["ALL CATEGORIES - MALE"].sum()
        F=data["ALL CATEGORIES - FEMALE"].sum()
        names2=["ALL CATEGORIES - MALE","ALL CATEGORIES - FEMALE"]
        t_enrolments=[E,F]
        t_enrolments_series = pd.Series(t_enrolments,index=names2)
        frame = { 'enrolments': t_enrolments_series }
        result = pd.DataFrame(frame)
        result.plot(kind='pie', subplots=True, figsize=(8, 8))

Out[4]: array([<matplotlib.axes._subplots.AxesSubplot object at 0x0000017A592ACB20>],
              dtype=object)
```



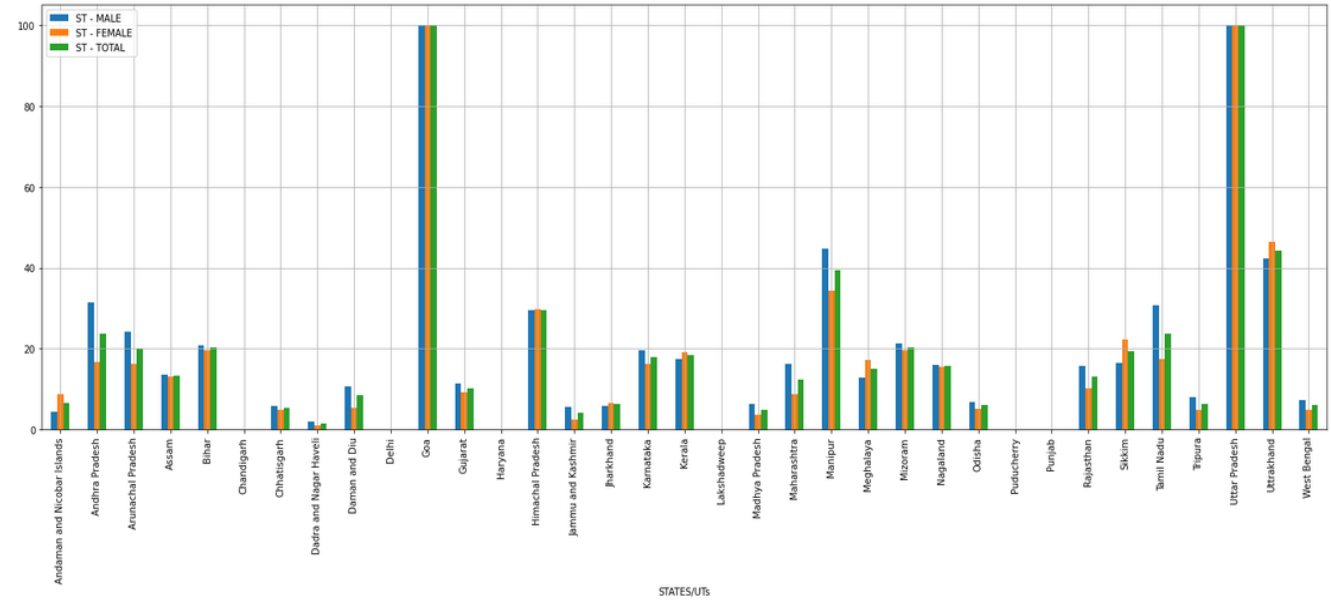
```
data1=data[['ALL CATEGORIES - MALE','ALL CATEGORIES - FEMALE','ALL CATEGORIES - TOTAL']]
# print(data1)
df=data1.plot.bar(rot=90,figsize=(25,8))
df.grid(True)
```



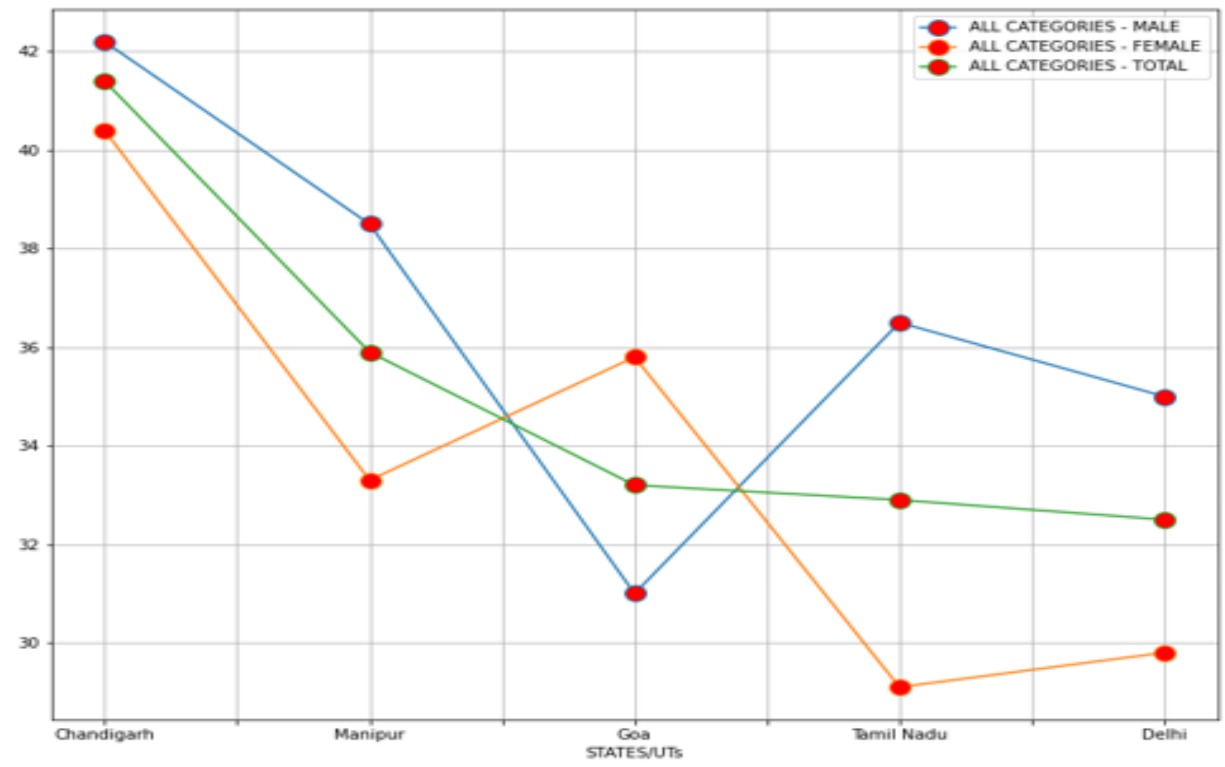

```
data1=data[['SC - MALE','SC - FEMALE','SC - TOTAL']]
# print(data1)
df=data1.plot.bar(rot=90,figsize=(25,8))
df.grid(True)
```



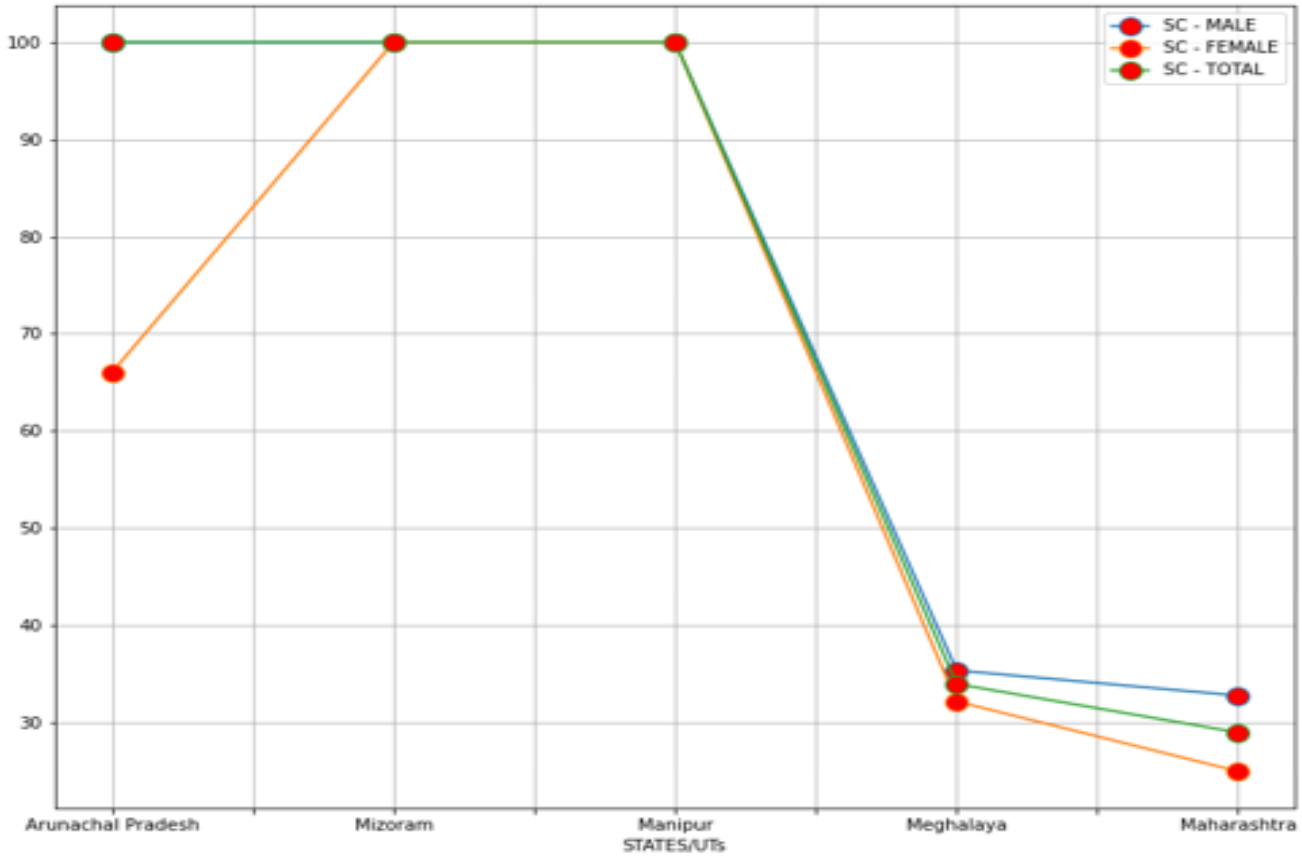
```
data1=data[['ST - MALE','ST - FEMALE','ST - TOTAL']]
# print(data1)
df=data1.plot.bar(rot=90,figsize=(25,8))
df.grid(True)
```



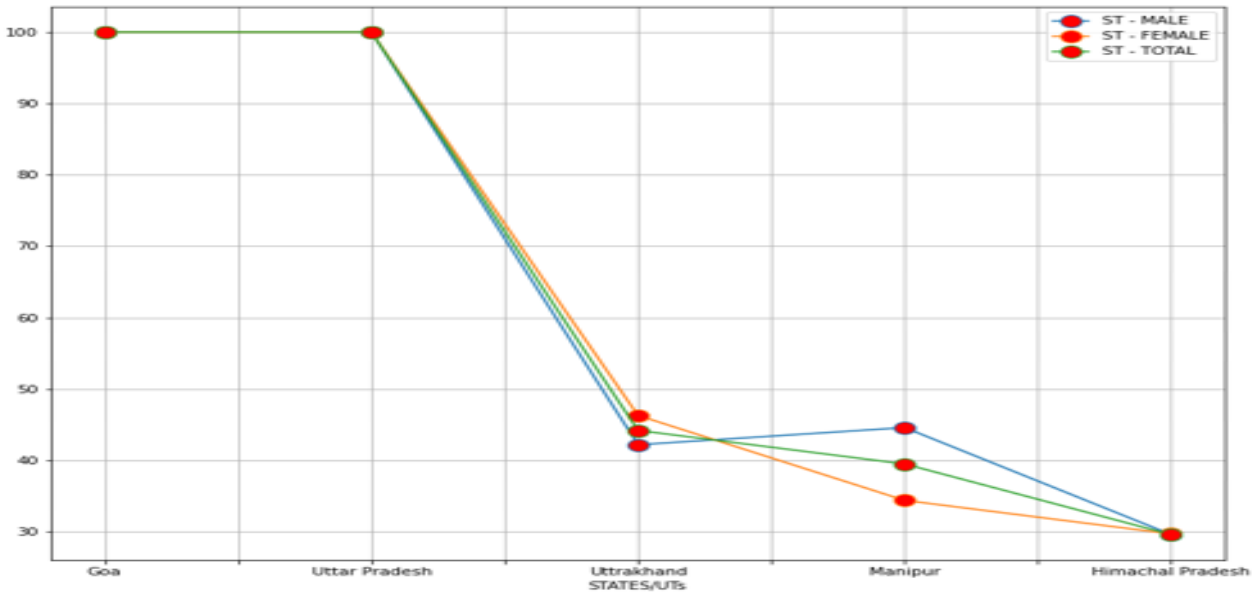

```
sort=data.sort_values('ALL CATEGORIES - TOTAL',ascending=False)
df=sort.head(5)
df=df[['ALL CATEGORIES - MALE','ALL CATEGORIES - FEMALE','ALL CATEGORIES - TOTAL']]
df=df.plot(figsize=(12, 10),marker='o', markerfacecolor='red', markersize=12)
df.grid(True)
```



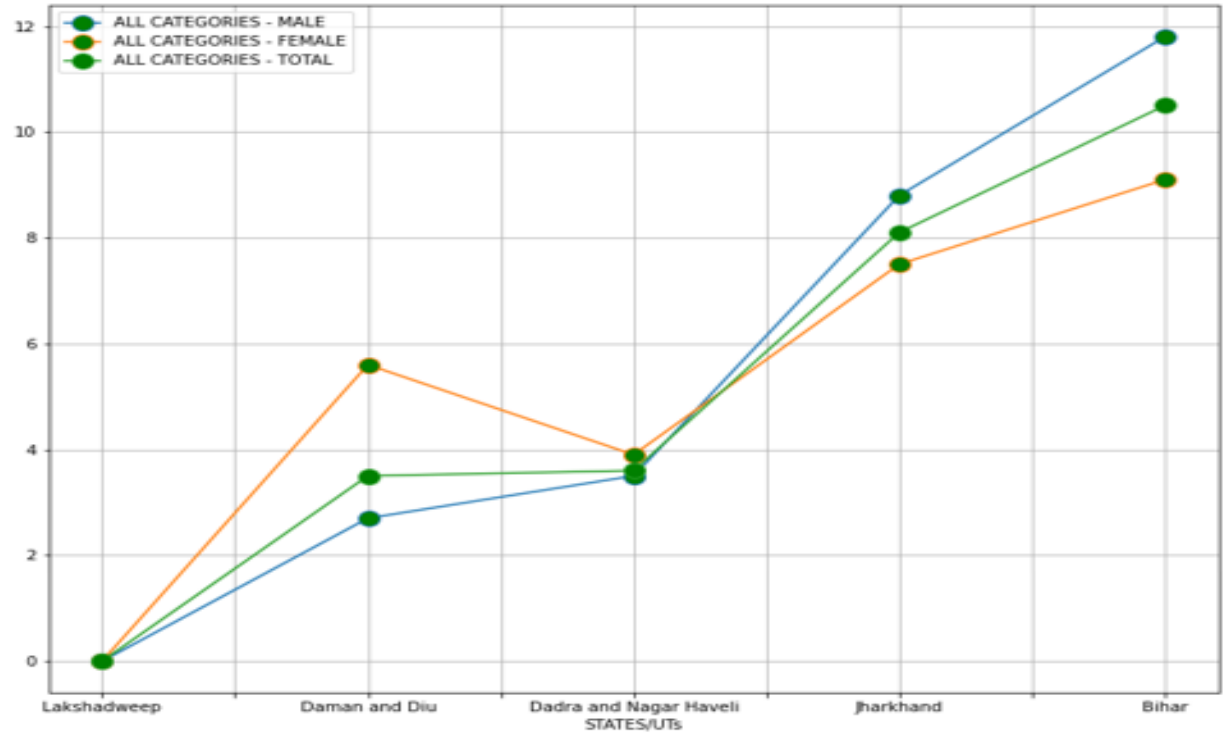
```
sort=data.sort_values('SC - TOTAL',ascending=False)
df=sort.head(5)
df=df[['SC - MALE','SC - FEMALE','SC - TOTAL']]
df=df.plot(figsize=(12, 10),marker='o', markerfacecolor='red', markersize=12)
df.grid(True)
```



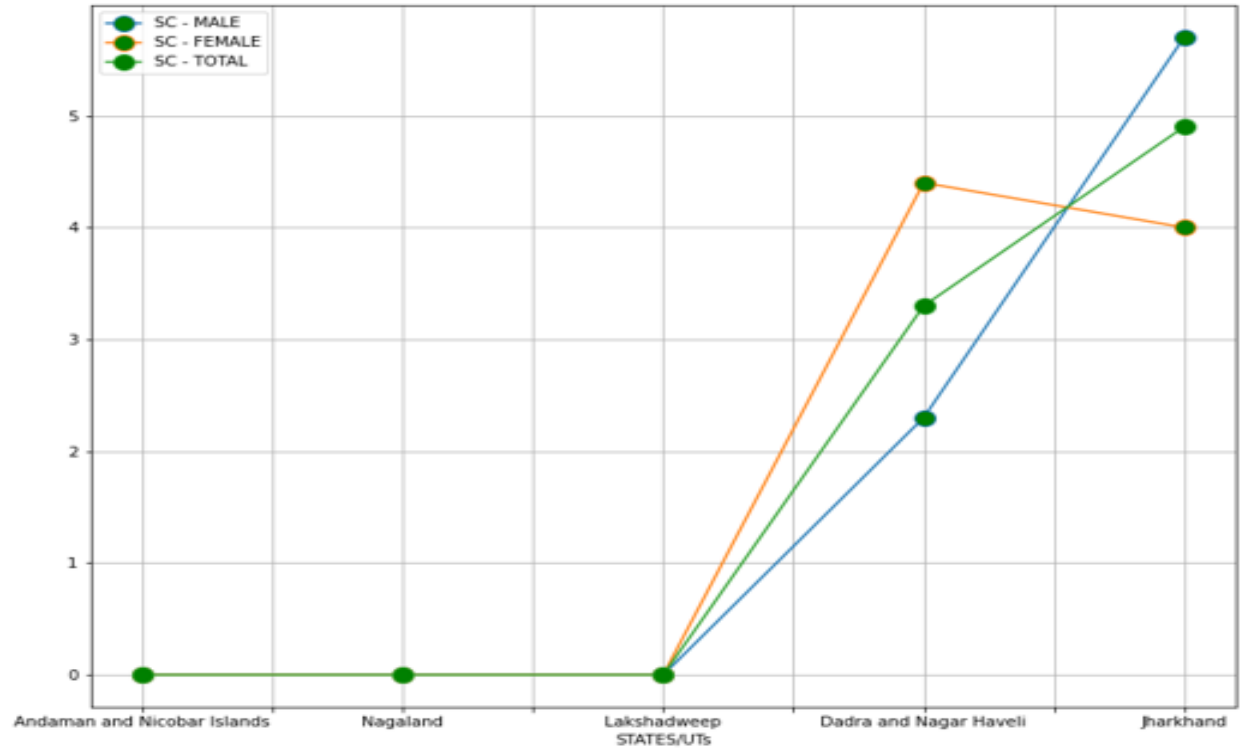

```
sort=data.sort_values('ST - TOTAL',ascending=False)
df=sort.head(5)
df=df[['ST - MALE','ST - FEMALE','ST - TOTAL']]
df=df.plot(figsize=(12, 10),marker='o', markerfacecolor='red', markersize=12)
df.grid(True)
```



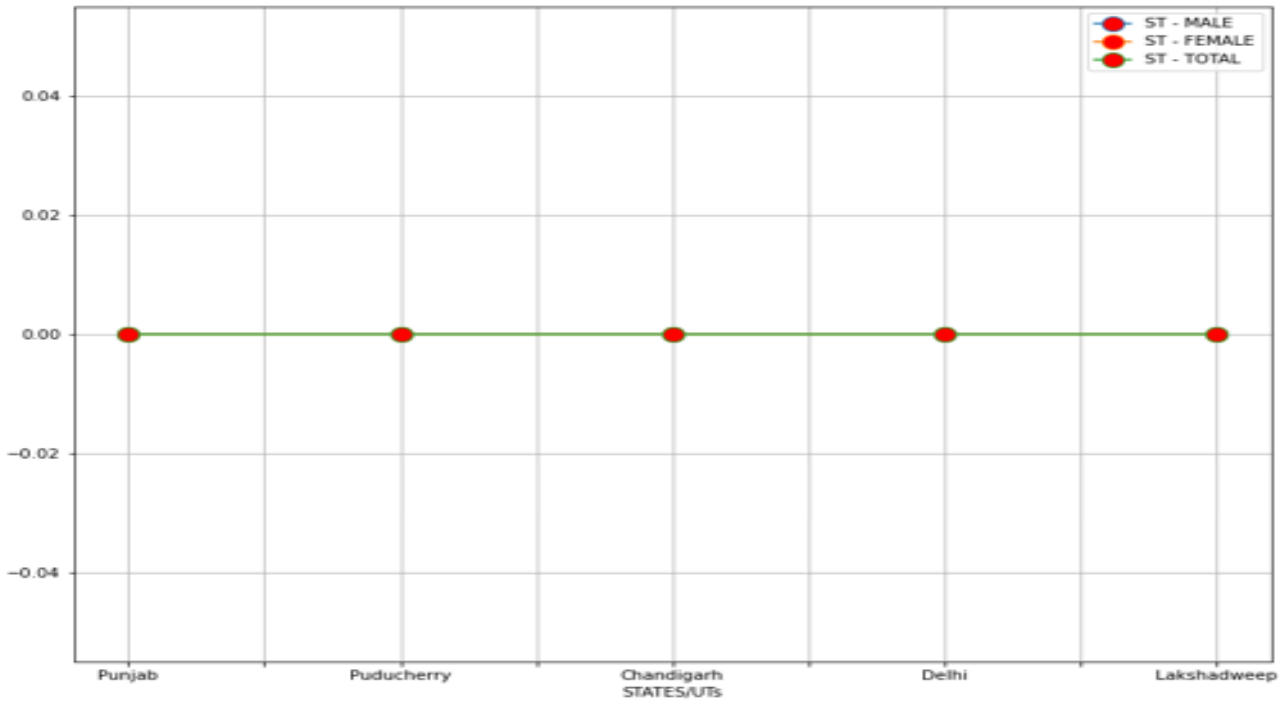
```
sort=data.sort_values('ALL CATEGORIES - TOTAL',ascending=True)
df=sort.head(5)
df=df[['ALL CATEGORIES - MALE','ALL CATEGORIES - FEMALE','ALL CATEGORIES - TOTAL']]
df=df.plot(figsize=(12, 10),marker='o', markerfacecolor='green', markersize=12)
df.grid(True)
```



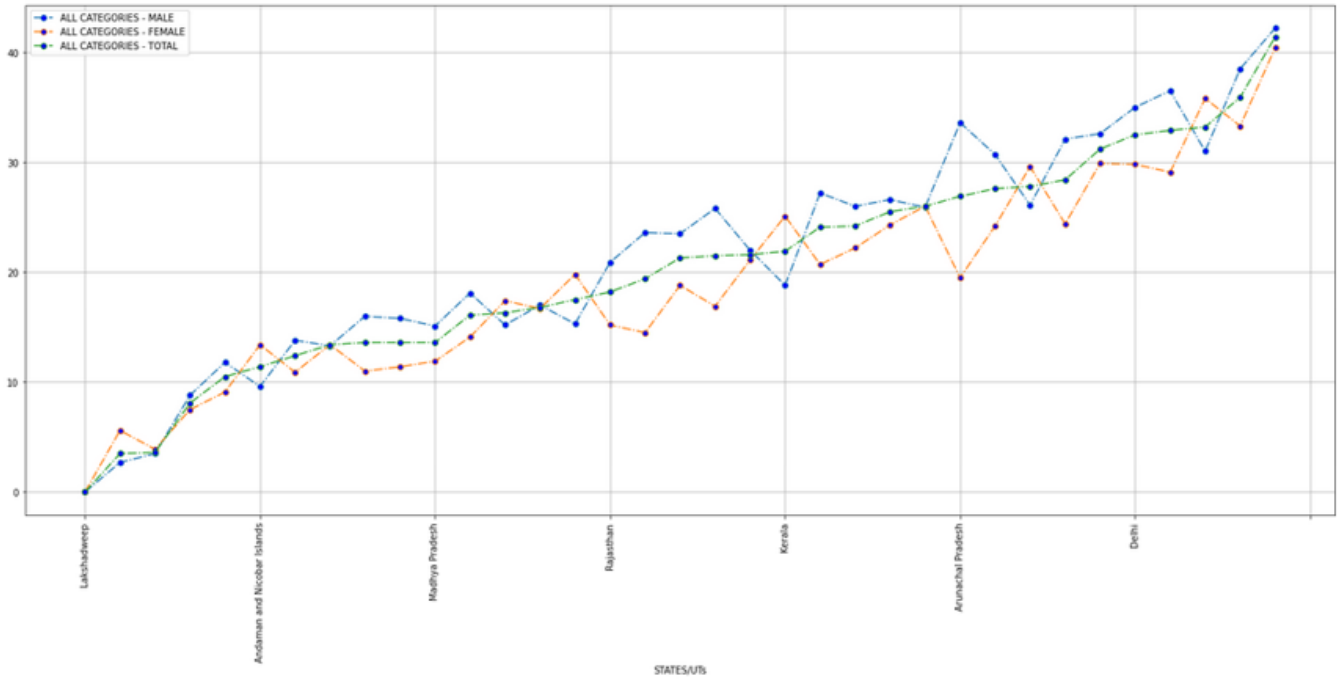

```
sort=data.sort_values('SC - TOTAL',ascending=True)
df=sort.head(5)
df=df[['SC - MALE','SC - FEMALE','SC - TOTAL']]
df=df.plot(figsize=(12, 10),marker='o', markerfacecolor='green', markersize=12)
df.grid(True)
```



```
sort=data.sort_values('ST - TOTAL',ascending=True)
df=sort.head(5)
df=df[['ST - MALE','ST - FEMALE','ST - TOTAL']]
df=df.plot(figsize=(12, 10),marker='o', markerfacecolor='red', markersize=12)
df.grid(True)
```

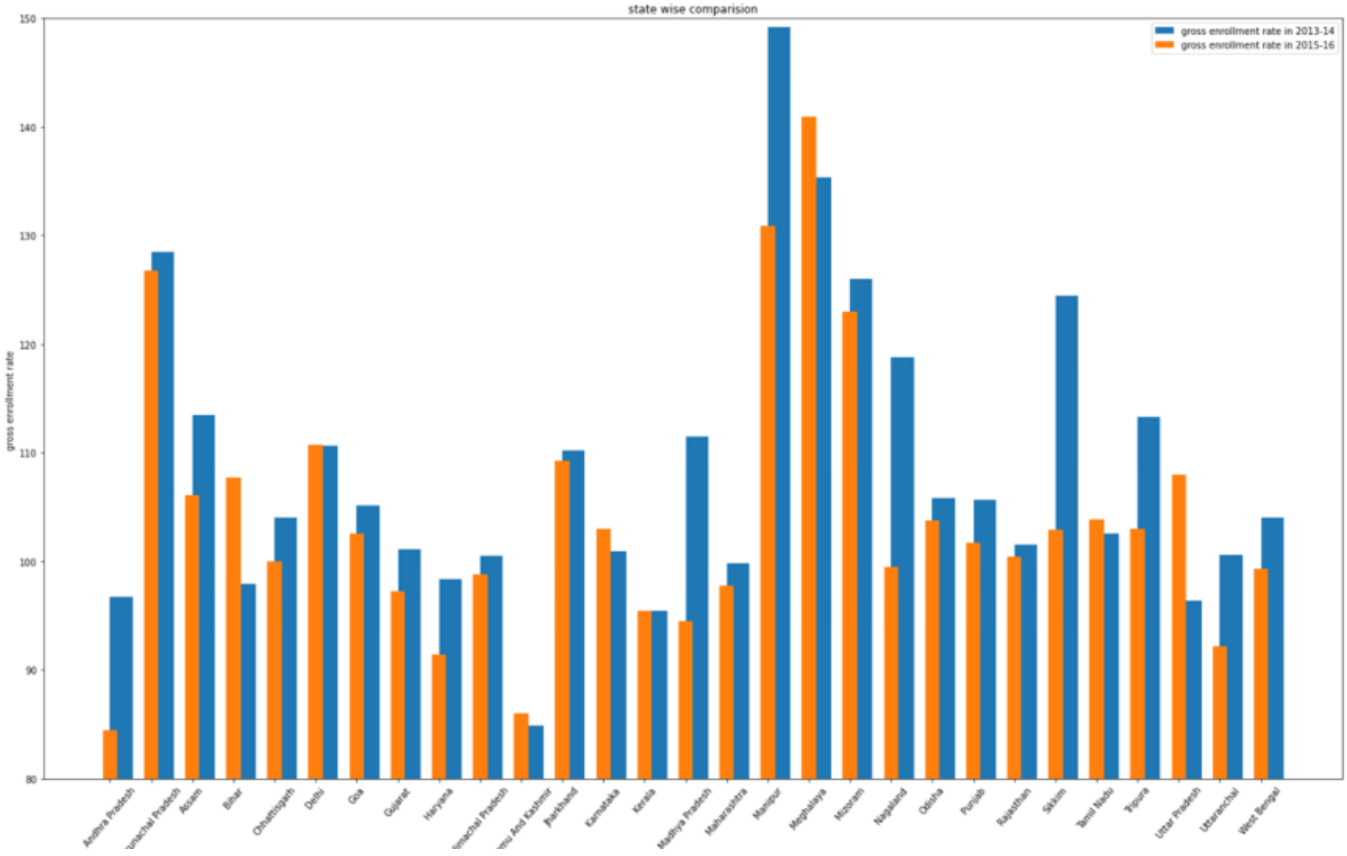



```
sort=data.sort_values('ALL CATEGORIES - TOTAL',ascending=True)
df=sort[['ALL CATEGORIES - MALE','ALL CATEGORIES - FEMALE','ALL CATEGORIES - TOTAL']]
df=df.plot(linestyle="-.",rot=90,figsize=(25, 10),marker='o', markerfacecolor='blue', markersize=6)
df.grid(True)
```

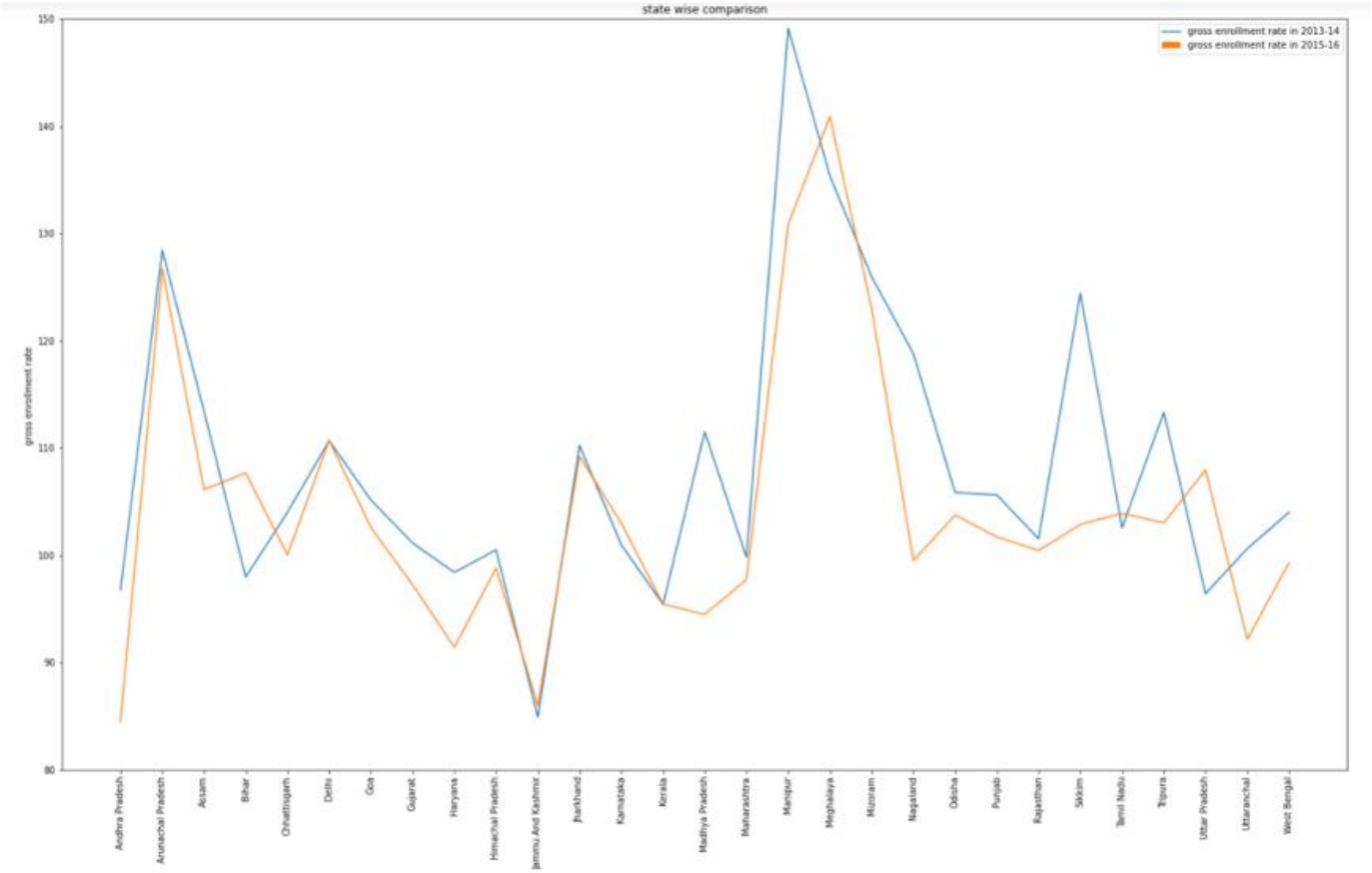


4. To check whether the student enrolment in primary schools has increased or decreased between 2013 to 2015.

```
fig=plt.figure(figsize=(25,10))
ax=fig.add_subplot(111)
xpos=np.arange(29)
p1=plt.bar(xpos, data['primary_2013-14'], width=0.55, align='edge')
p2=plt.bar(xpos, data['primary_2015-16'], width=0.35)
plt.ylabel('gross enrollment rate ')
plt.xlabel('states')
plt.xticks(xpos,data['states'],rotation=50)
plt.title('state wise comparision')
plt.ylim(60,100)
plt.legend((p1[0],p2[0]),('gross enrollment rate in 2013-14',
                          'gross enrollment rate in 2015-16'))
plt.ylim(80,150)
plt.show()
```

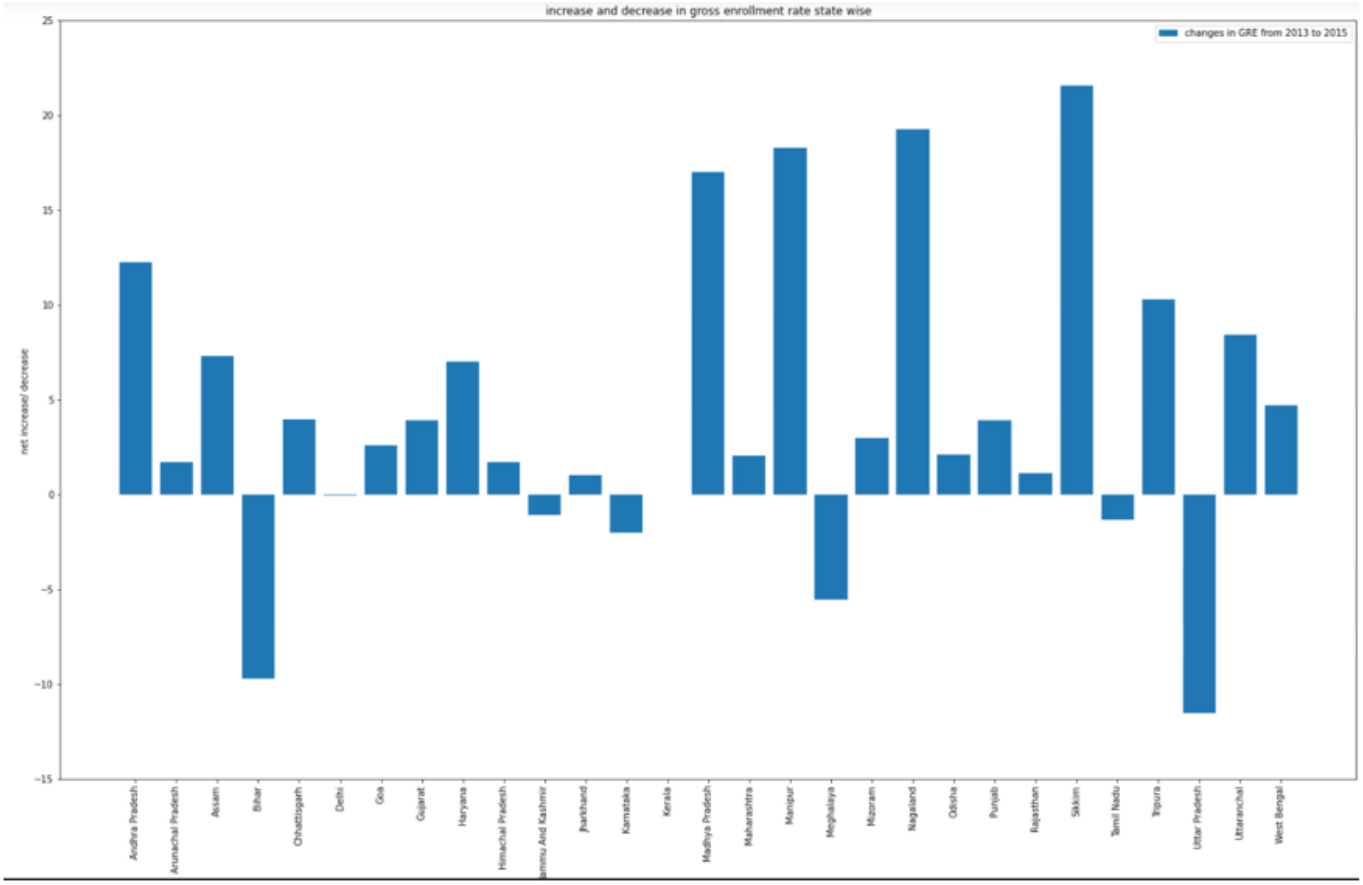


```
new=data.describe().transpose()
fig=plt.figure(figsize=(25,10))
ax=fig.add_subplot(111)
xpos=np.arange(29)
p1=plt.plot(xpos, data[['primary_2013-14','primary_2015-16']] )

plt.ylabel('gross enrollment rate ')
plt.xlabel('year')
plt.xticks(xpos,data['states'],rotation=90)
plt.title('year wise comparison')
plt.ylim(80,150)
plt.legend((p1[0],p2[0]),('gross enrollment rate in 2013-14',
                          'gross enrollment rate in 2015-16'))

plt.show()
```



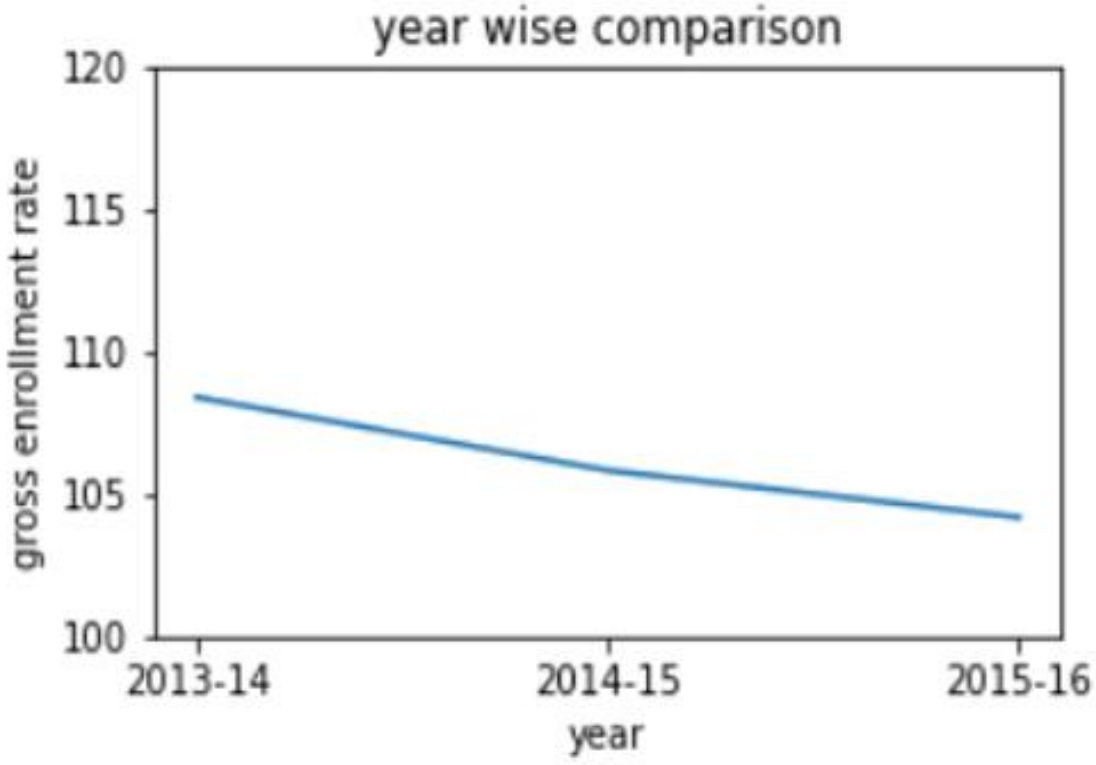
In the above graphs we can see comparative changes in gross enrolment rate between 2013 and 2015

```
changes=data[['states','primary_2013-14','primary_2015-16']]
changes['net increase/decrease']=changes['primary_2013-14']-changes['primary_2015-16']
changes
fig=plt.figure(figsize=(25,10))
ax=fig.add_subplot(111)
xpos=np.arange(29)
p1=plt.bar(xpos, changes['net increase/decrease'])

plt.ylabel('net increase/ decrease ')
plt.xlabel('states')
plt.xticks(xpos,data['states'],rotation=90)
plt.title('state wise comparision')

plt.legend((p1[0],p2[0]),('gross enrollment rate in 2013-14',
                          'gross enrollment rate in 2015-16'))
plt.ylim(-15,25)
plt.show()
```

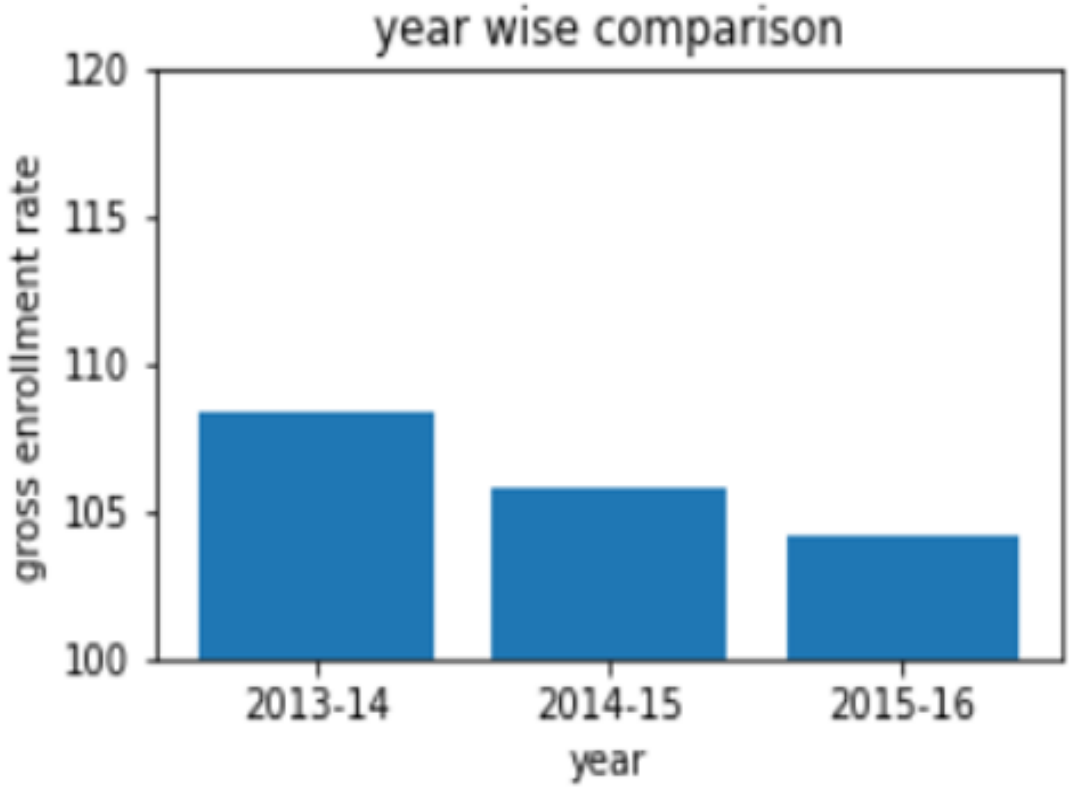


in the above graph we can see net increment/decrement in gross enrolment rate state wise

```
new=data.describe().transpose()
fig=plt.figure(figsize=(10,6))
ax=fig.add_subplot(222)
xpos=np.arange(3)
p1=plt.plot(xpos, new['mean'] )

plt.ylabel('gross enrollment rate ')
plt.xlabel('year')
plt.xticks(xpos,['2013-14','2014-15','2015-16'])
plt.title('year wise comparison')
plt.ylim(100,120)

plt.show()
```



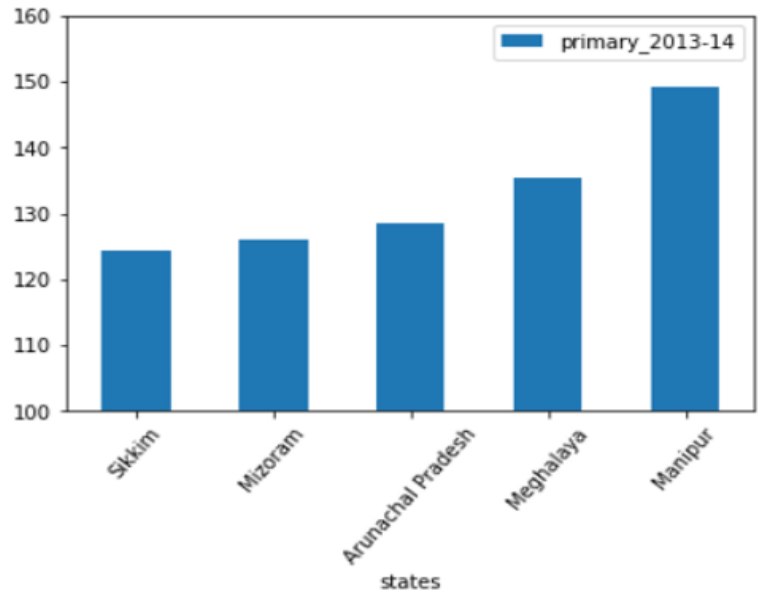
```
new=data.describe().transpose()
fig=plt.figure(figsize=(10,6))
ax=fig.add_subplot(222)
xpos=np.arange(3)
p1=plt.bar(xpos, new['mean'] )
plt.ylabel('gross enrollment rate')
plt.xlabel('year')
plt.xticks(xpos,['2013-14','2014-15','2015-16'])
plt.title('year wise comparison')
plt.ylim(100,120)

plt.show()
```



in above graphs we can see decrement in gross enrolment rate year wise

```
topfive=data[['states','primary_2013-14']].sort_values(by=['primary_2013-14']).tail(5)
topfive.plot.bar(x='states',y='primary_2013-14',rot=50)
plt.ylim(100,160)
```

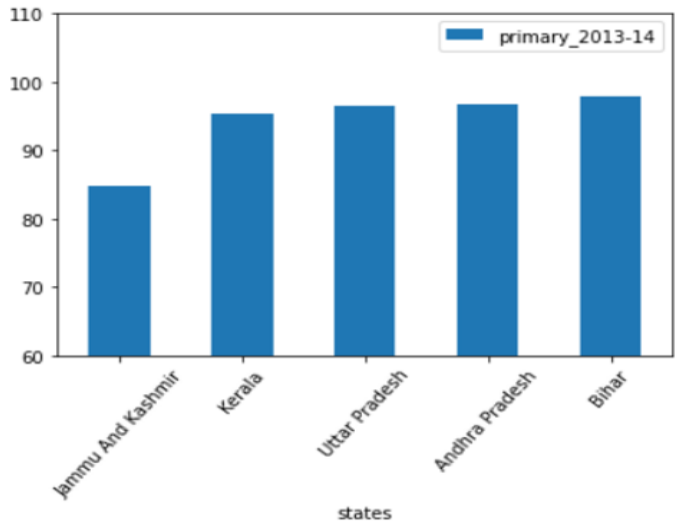
```
(100.0, 160.0)
```



above graph shows top five states with highest GRE in 2013

```
topfive=data[['states','primary_2013-14']].sort_values(by=['primary_2013-14']).head(5)
topfive.plot.bar(x='states',y='primary_2013-14',rot=50)
plt.ylim(60,110)
```

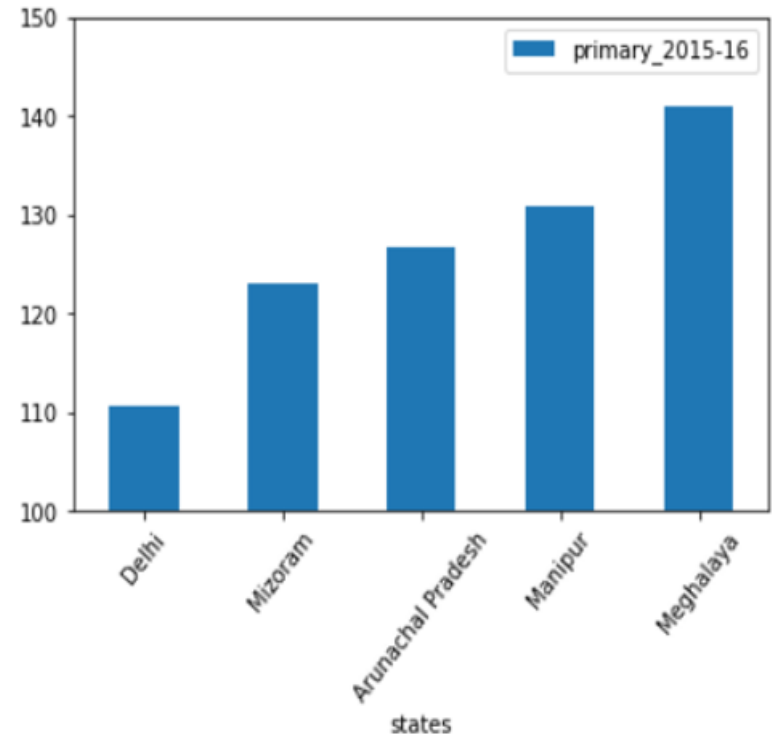
```
(60.0, 110.0)
```



above graph shows top 5 states with lowest GRE in 2013

```
topfive=data[['states','primary_2015-16']].sort_values(by=['primary_2015-16']).tail(5)
topfive.plot.bar(x='states',y='primary_2015-16',rot=50)
plt.ylim(100,150)
```

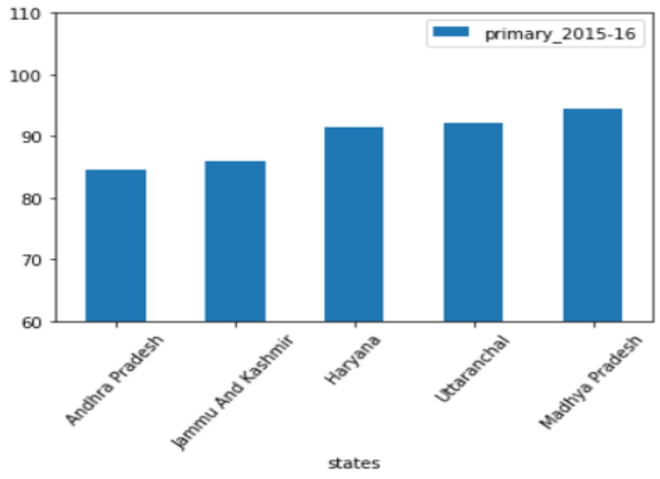
```
(100.0, 150.0)
```



above graph shows top 5 states with highest GRE in 2015

```
topfive=data[['states','primary_2015-16']].sort_values(by=['primary_2015-16']).head(5)
topfive.plot.bar(x='states',y='primary_2015-16',rot=50)
plt.ylim(60,110)
```

```
(60.0, 110.0)
```



Above graph shows top 5 states with lowest GRE in 2015

Conclusion

We have studied about Pupil Teacher Ratio or we can say that it is the number of the students that a single has teacher has to care of. We have applied many operations so that we can get clarity about the data. At last we can say that the ratio is good in various states for a specific level of the education. But in some states like Uttar Pradesh, Bihar, West Bengal and all other such places, the pupil teacher ratio is very high which means that the teachers which are there have a very huge load and pressure and also the quality of the education can also lack there due to so high pupil teacher ratio. Therefore, a lot of work is required to be done at such places. But there are some places where it is very good like Kerala.

We have compared the various levels of the school education i.e. primary, upper primary and some others state wise and also union territories wise. We have reached to a conclusion that the main and most important level is the primary level where a child can get basic knowledge. So the government is also facing towards increasing the number of students in primary level than any other level.

We have also seen the correlation between the levels of the education. By seeing the correlation between (Primary-Upper Primary) we can say that the there are good number of teachers for the student enrolling, but as the level starts increasing the correlation between the levels becomes moderate, so at that place the government should have to focus for the progress in the education field.

We have analysed that the in some portions the work for achieving the Quality Education SDG is going on very nicely like in the primary level the enrolment of the student is increasing and the gender gap is also decreasing. We have to work in some more portions with more dedication and more hard work so that we can achieve the goal successfully. The facility that are available in the schools are some what good but the government should still need to implement the facilities more properly. We also have to contribute individually so that the speed of the development in the field of education increases. At last we can say that we are lagging in some portions only, so we need to focus on those portions.

APPENDIX

CSV FILES

Objective 1

https://jklujaipur-my.sharepoint.com/:f:/g/personal/chiragkumar_jklu_edu_in/ErSK5nrvC-5JrK56CpXc9T4BvSMScsYWTzGiBkzGj8J5vg?e=IXgFfd

Objective 2

https://jklujaipur-my.sharepoint.com/:f:/g/personal/chiragkumar_jklu_edu_in/Es010U4ssa9Eus7nqMbRFs4B5agicGaLHOCWYWboDRuTqQ?e=TFaPWF

Objective 3

https://jklujaipur-my.sharepoint.com/:u:/g/personal/chiragkumar_jklu_edu_in/EZkEJ5rVDMNGjVUxaL7WhjsBFIHGwUdx_tzT3JxqIIEj8Q?e=MLVvyL

Objective 4

https://jklujaipur-my.sharepoint.com/:u:/g/personal/chiragkumar_jklu_edu_in/EccfKRpiRZdJuWLah8miNpYBkySN91xjNOICDcI3YrlMUQ?e=WU2JwA

Jupyter Files

1. Objective 1
   https://jklujaipur-my.sharepoint.com/:u:/g/personal/chiragkumar_jklu_edu_in/ER5XHVI7oVFFv8pN3prC1L4Bhm_IbeTUHCjtlzdBljTn0A?e=Mto0YG
2. Objective 2
   a. https://jklujaipur-my.sharepoint.com/:u:/g/personal/chiragkumar_jklu_edu_in/EdZpwWBlufZEjSjoVQhz6w4B7vQpALFWSct36L0otO2Zuw?e=C1W4YK
   b. https://jklujaipur-my.sharepoint.com/:u:/g/personal/chiragkumar_jklu_edu_in/ERHSn2yRoXxEvEoGuHnfa_YBXHSEkEifdIZ0kasbiA2HmA?e=tJgyd9
   c. https://jklujaipur-my.sharepoint.com/:u:/g/personal/chiragkumar_jklu_edu_in/EZHkacbnlbFPpDhWZIdXVJsByA3DoxkN-skCaFDaBKL3Jw?e=KOkAxc

3. Objective 3
   https://jklujaipur-my.sharepoint.com/:u:/g/personal/chiragkumar_jklu_edu_in/EU_CVtfcBh5NlvxNLJgM2LUB4DcA3DFsKNTke4-Yyvq-Gw?e=oajEL0

4. Objective 4
   https://jklujaipur-my.sharepoint.com/:u:/g/personal/chiragkumar_jklu_edu_in/EbuhUTpKCH1Dssn9JRXaCqMB0e0zFjkD8zgdZMF8E6ZYqA?e=1UA9U0

REFERENCES

1. http://ictactjournals.in/paper/IJMS_V_2_I_2_Paper_2_281_288.pdf
2. https://www.researchgate.net/publication/269932850_Factors_Leading_to_School_Dropouts_in_India_An_Analysis_of_National_Family_Health_Survey-3_Data
3. https://www.researchgate.net/publication/318291008_Analysis_of_Higher_Education_GER_-_A_Study_for_West_Bengal_and_Orissa
4. https://www.education.gov.in/sites/upload_files/mhrd/files/statistics-new/ESAG-2018.pdf
5. https://data.gov.in/major-indicator/drop-out-rate
6. https://data.gov.in/major-indicator/gross-enrolment-ratio-schools