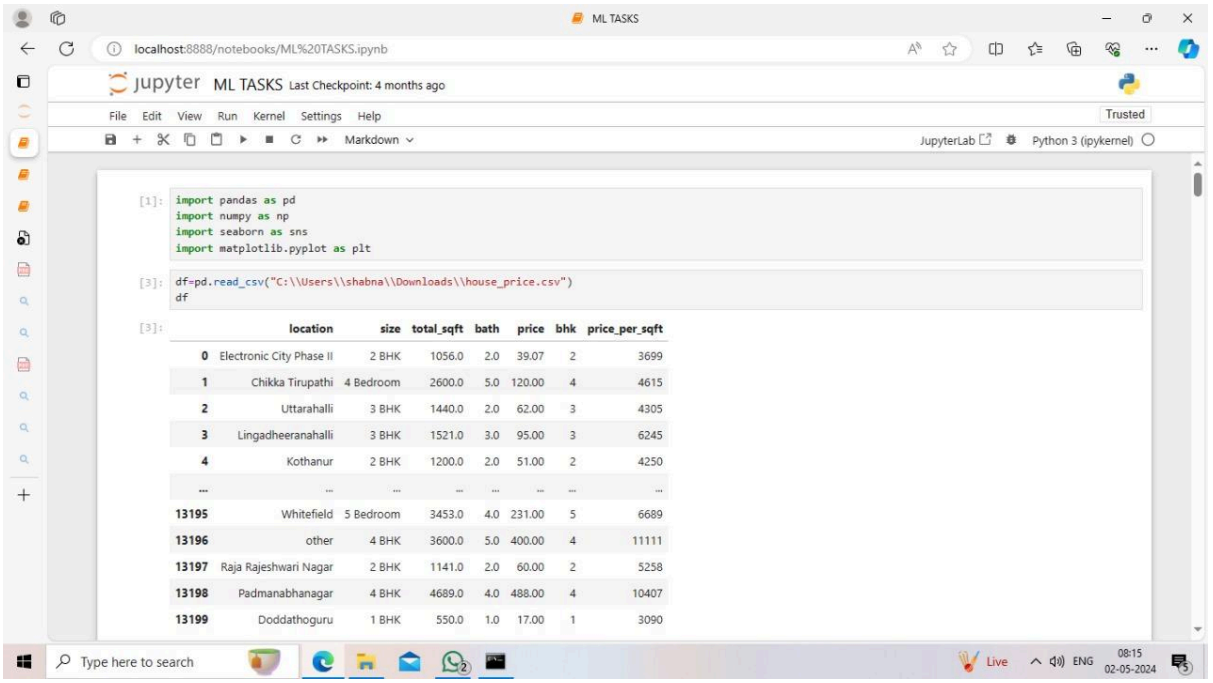
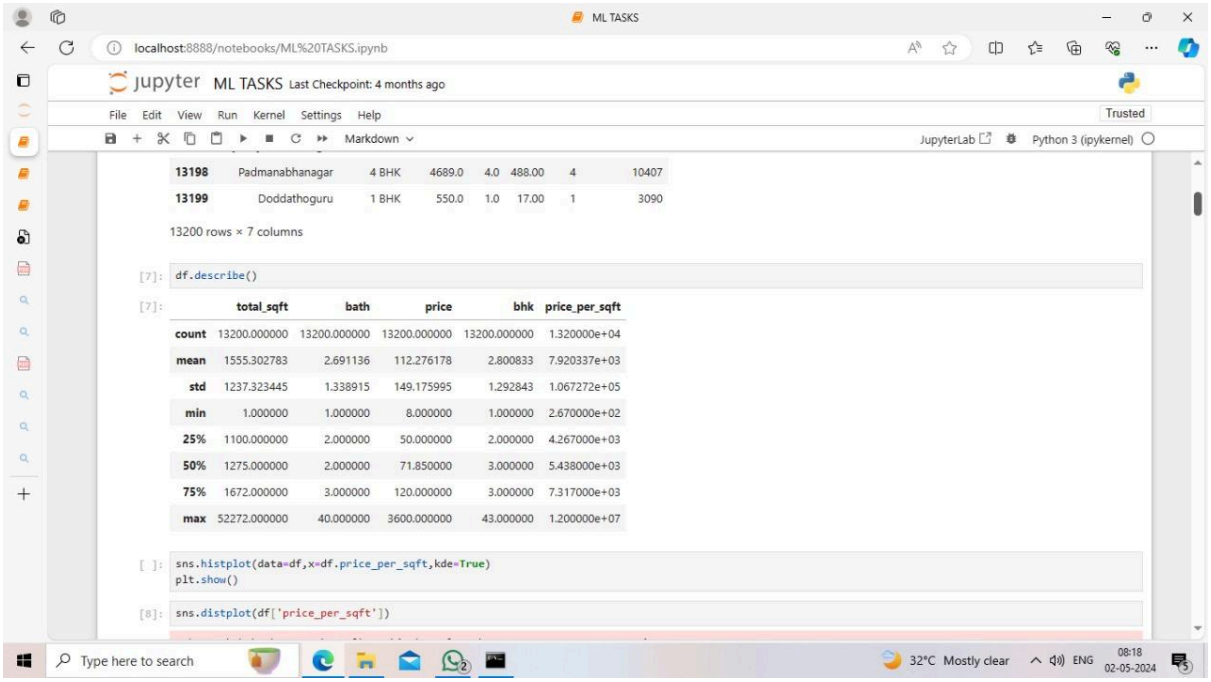
ML - Task1
- Measures of Descriptive statistics-Central Tendency, spread
You are given house_price.csv which contains property prices in the city of Bangalore. You need to examine price per square feet do the following:

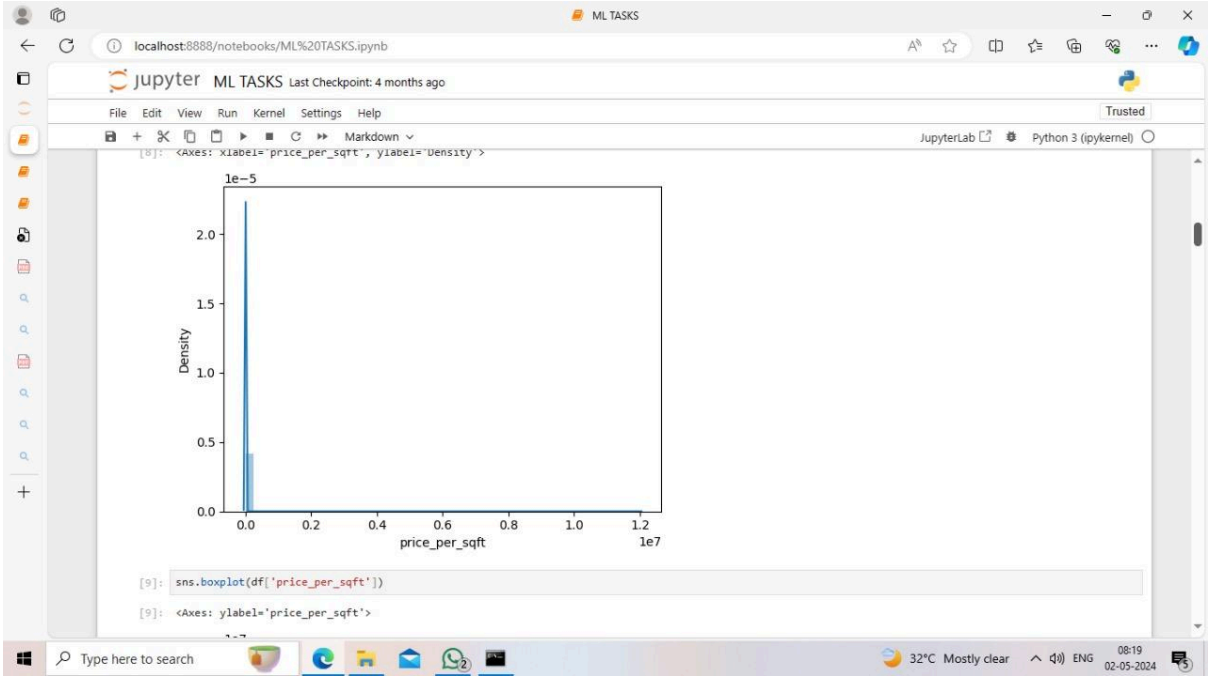
Detect the outliers and remove it using:
1. Mean Function
2. Percentile method
3. IQR(Inter quartile range method)
4. Normal distribution
5. Zscore method

Also, plot the box plot(for all the numerical columns), histplot(to check the normality of the column(price per sqft column))

Check the correlation between all the numerical columns and plot heatmap.
Scatter plot between the variables to check the correlation between them.




ML TASKS
localhost:8888/notebooks/ML%20TASKS.ipynb
jupyter ML TASKS Last Checkpoint: 4 months ago
File Edit View Run Kernel Settings Help
Trusted
Markdown
JupyterLab
Python 3 (ipykernel)
13198 Padmanabhanagar 4 BHK 4689.0 4.0 488.00 4 10407
13199 Doddathoguru 1 BHK 550.0 1.0 17.00 1 3090
13200 rows × 7 columns
[7]: df.describe()
[7]:
total_sqft bath price bhk price_per_sqft
count 13200.000000 13200.000000 13200.000000 13200.000000 1.320000e+04
mean 1555.302783 2.691136 112.276178 2.800833 7.920337e+03
std 1237.323445 1.338915 149.175995 1.292843 1.067272e+05
min 1.000000 1.000000 8.000000 1.000000 2.670000e+02
25% 1100.000000 2.000000 50.000000 2.000000 4.267000e+03
50% 1275.000000 2.000000 71.850000 3.000000 5.438000e+03
75% 1672.000000 3.000000 120.000000 3.000000 7.317000e+03
max 52272.000000 40.000000 3600.000000 43.000000 1.200000e+07
[ ]: sns.histplot(data=df,x=df.price_per_sqft,kde=True)
plt.show()
[8]: sns.distplot(df['price_per_sqft'])
Type here to search
32°C Mostly clear
ENG
08:18
02-05-2024


ML TASKS
localhost:8888/notebooks/ML%20TASKS.ipynb
jupyter ML TASKS Last Checkpoint: 4 months ago
File Edit View Run Kernel Settings Help
Trusted
Markdown
JupyterLab
Python 3 (ipykernel)
[8]: <Axes: xlabel='price_per_sqft', ylabel='Density'>
1e−5
2.0
1.5
1.0
0.5
0.0
Density
0.0 0.2 0.4 0.6 0.8 1.0 1.2
price_per_sqft
1e7
[9]: sns.boxplot(df['price_per_sqft'])
[9]: <Axes: ylabel='price_per_sqft'>
Type here to search
32°C Mostly clear
ENG
08:19
02-05-2024

ML TASKS

localhost:8888/notebooks/ML%20TASKS.ipynb

jupyter ML TASKS Last Checkpoint: 4 months ago

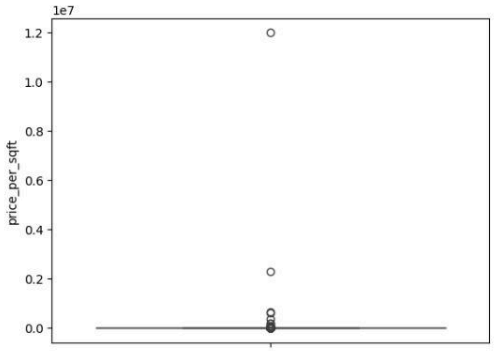
File Edit View Run Kernel Settings Help

Trusted

Markdown

JupyterLab Python 3 (ipykernel)

[9]: <Axes: ylabel='price_per_sqft'>



Z SCORE METHOD

[11]:
```
mean=df['price_per_sqft'].mean()
mean
```

Type here to search

32°C Mostly clear ENG 08:20 02-05-2024

ML TASKS

localhost:8888/notebooks/ML%20TASKS.ipynb

jupyter ML TASKS Last Checkpoint: 4 months ago

File Edit View Run Kernel Settings Help

Trusted

Markdown

JupyterLab Python 3 (ipykernel)

Z SCORE METHOD

[11]:
```
mean=df['price_per_sqft'].mean()
mean
```

[11]: 7920.336742424242

[13]:
```
s=df['price_per_sqft'].std()
s
```

[13]: 106727.16032810867

[15]:
```
positive3sigma=mean+3*s
negative3sigma=mean-3*s
print(positive3sigma)
print(negative3sigma)
```

```
328101.8177267502
-312261.14424190175
```

OUTLIERS

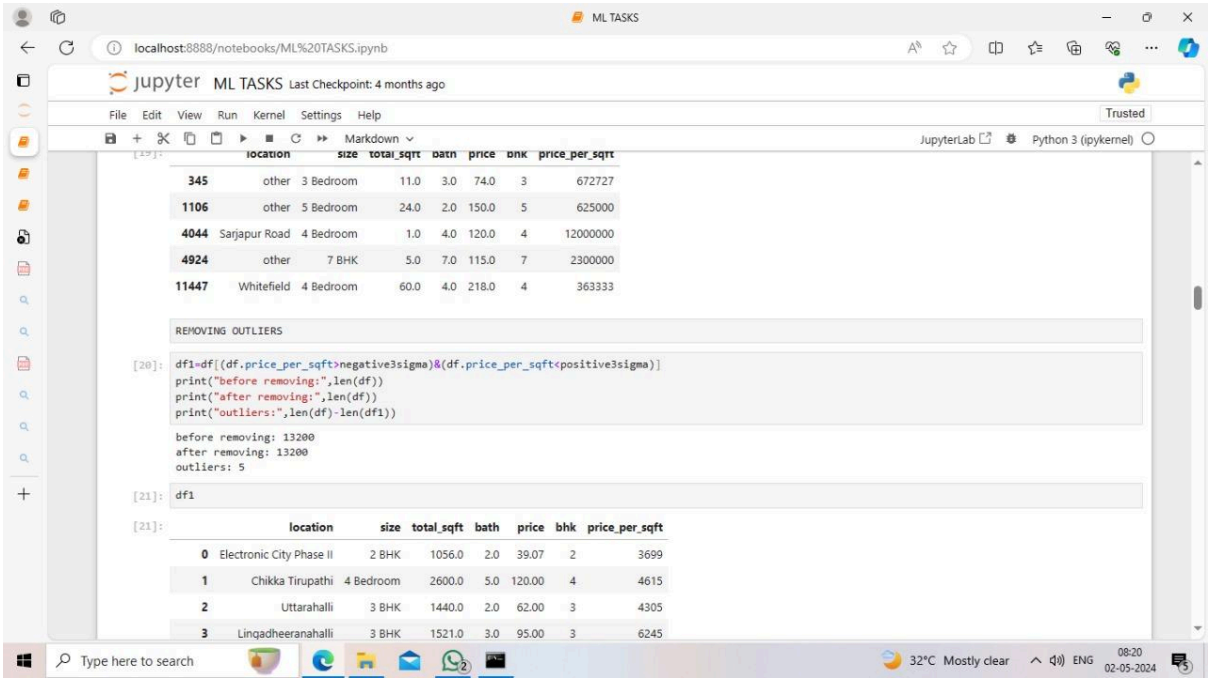
[19]:
```
df[(df.price_per_sqft<negative3sigma)|(df.price_per_sqft>positive3sigma)]
```

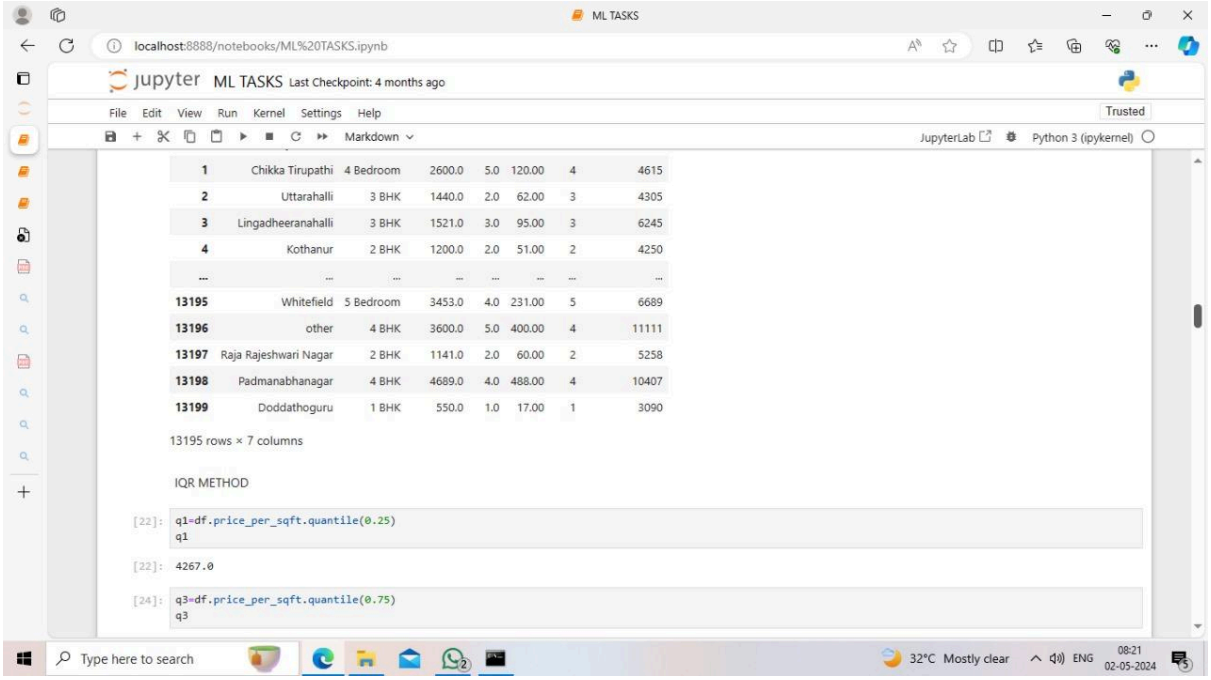
[19]:

	location	size	total_sqft	bath	price	bhk	price_per_sqft
345	other	3 Bedroom	11.0	3.0	74.0	3	672727
1106	other	5 Bedroom	24.0	2.0	150.0	5	625000

Type here to search

32°C Mostly clear ENG 08:20 02-05-2024


ML TASKS
localhost:8888/notebooks/ML%20TASKS.ipynb
jupyter ML TASKS Last Checkpoint: 4 months ago
File Edit View Run Kernel Settings Help
Trusted
Markdown
JupyterLab
Python 3 (ipykernel)
location size total_sqft bath price bhk price_per_sqft
345 other 3 Bedroom 11.0 3.0 74.0 3
672727
1106 other 5 Bedroom 24.0 2.0 150.0 5
625000
4044 Sarjapur Road 4 Bedroom 1.0 4.0 120.0 4
12000000
4924 other 7 BHK 5.0 7.0 115.0 7
2300000
11447 Whitefield 4 Bedroom 60.0 4.0 218.0 4
363333
REMOVING OUTLIERS
[20]:
df1=df[(df.price_per_sqft>negative3sigma)&(df.price_per_sqft<positive3sigma)]
print("before removing:",len(df))
print("after removing:",len(df))
print("outliers:",len(df)-len(df1))
before removing: 13200
after removing: 13200
outliers: 5
[21]:
df1
[21]:
location size total_sqft bath price bhk price_per_sqft
0 Electronic City Phase II 2 BHK 1056.0 2.0 39.07 2
3699
1 Chikka Tirupathi 4 Bedroom 2600.0 5.0 120.00 4
4615
2 Uttarahalli 3 BHK 1440.0 2.0 62.00 3
4305
3 Lingadheeranahalli 3 BHK 1521.0 3.0 95.00 3
6245
Type here to search
32°C Mostly clear
ENG
08:20
02-05-2024


ML TASKS
localhost:8888/notebooks/ML%20TASKS.ipynb
jupyter ML TASKS Last Checkpoint: 4 months ago
File Edit View Run Kernel Settings Help
Trusted
Markdown
JupyterLab
Python 3 (ipykernel)
1 Chikka Tirupathi 4 Bedroom 2600.0 5.0 120.00 4
4615
2 Uttarahalli 3 BHK 1440.0 2.0 62.00 3
4305
3 Lingadheeranahalli 3 BHK 1521.0 3.0 95.00 3
6245
4 Kothanur 2 BHK 1200.0 2.0 51.00 2
4250
... ...
13195 Whitefield 5 Bedroom 3453.0 4.0 231.00 5
6689
13196 other 4 BHK 3600.0 5.0 400.00 4
11111
13197 Raja Rajeshwari Nagar 2 BHK 1141.0 2.0 60.00 2
5258
13198 Padmanabhanagar 4 BHK 4689.0 4.0 488.00 4
10407
13199 Doddathoguru 1 BHK 550.0 1.0 17.00 1
3090
13195 rows × 7 columns
IQR METHOD
[22]:
q1=df.price_per_sqft.quantile(0.25)
q1
[22]:
4267.0
[24]:
q3=df.price_per_sqft.quantile(0.75)
q3
Type here to search
32°C Mostly clear
ENG
08:21
02-05-2024

ML TASKS

localhost:8888/notebooks/ML%20TASKS.ipynb

jupyter ML TASKS Last Checkpoint: 4 months ago

File Edit View Run Kernel Settings Help

Trusted

Markdown

JupyterLab Python 3 (ipykernel)

[25]:
```
3050.0
```

[26]:
```
upperlimit=q3+(1.5*iqr)
lowerlimit=q1-(1.5*iqr)
(upperlimit,lowerlimit)
```

[26]:
```
(11892.0, -308.0)
```

[27]:
```
df[(df.price_per_sqft<lowerlimit)|(df.price_per_sqft>upperlimit)]
```

[27]:

	location	size	total_sqft	bath	price	bhk	price_per_sqft
7	Rajaji Nagar	4 BHK	3300.0	4.0	600.0	4	18181
9	other	6 Bedroom	1020.0	6.0	370.0	6	36274
22	Thanisandra	4 Bedroom	2800.0	5.0	380.0	4	13571
45	HSR Layout	8 Bedroom	600.0	9.0	200.0	8	33333
48	KR Puram	2 Bedroom	800.0	1.0	130.0	2	16250
...	...	...	...	...	...	...	...
13142	other	2 BHK	1140.0	1.0	185.0	2	16228
13157	other	7 Bedroom	1400.0	7.0	218.0	7	15571
13185	Hulimavu	1 BHK	500.0	1.0	220.0	1	44000
13186	other	4 Bedroom	1200.0	5.0	325.0	4	27083
13191	Ramamurthy Nagar	7 Bedroom	1500.0	9.0	250.0	7	16666

Type here to search

32°C Mostly clear ENG 08:22 02-05-2024

ML TASKS

localhost:8888/notebooks/ML%20TASKS.ipynb

jupyter ML TASKS Last Checkpoint: 4 months ago

File Edit View Run Kernel Settings Help

Trusted

Markdown

JupyterLab Python 3 (ipykernel)

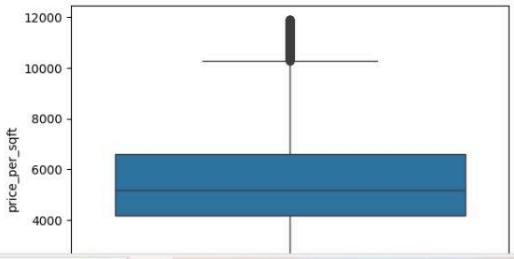
1265 rows × 7 columns

[29]:
```
df1=df[(df.price_per_sqft>lowerlimit)&(df.price_per_sqft<upperlimit)]
print("before removing :",len(df))
print("after removing:",len(df1))
print("outliers:",len(df)-len(df1))
```

```
before removing : 13200
after removing: 11935
outliers: 1265
```

[31]:
```
sns.boxplot(df1.price_per_sqft)
```

[31]:
```
<Axes: ylabel='price_per_sqft'>
```



Type here to search

32°C Mostly clear ENG 08:22 02-05-2024

ML TASKS

localhost:8888/notebooks/ML%20TASKS.ipynb

jupyter ML TASKS Last Checkpoint: 4 months ago

File Edit View Run Kernel Settings Help

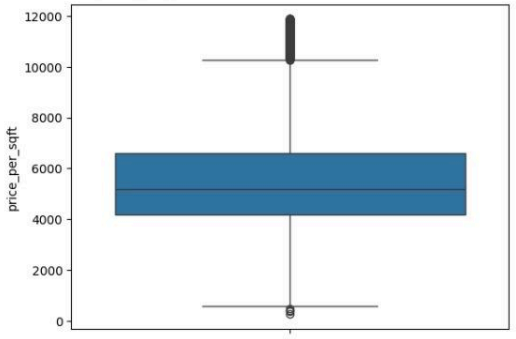
Trusted

Markdown

JupyterLab Python 3 (ipykernel)

[31]:
```
sns.boxplot(df1.price_per_sqft)
```

[31]: <Axes: ylabel='price_per_sqft'>



PERCENTILE METHOD

[33]:
```
upperlimit=df.price_per_sqft.quantile(0.95)
```

Type here to search

32°C Mostly clear ENG 08:23 02-05-2024

ML TASKS

localhost:8888/notebooks/ML%20TASKS.ipynb

jupyter ML TASKS Last Checkpoint: 4 months ago

File Edit View Run Kernel Settings Help

Trusted

Markdown

JupyterLab Python 3 (ipykernel)

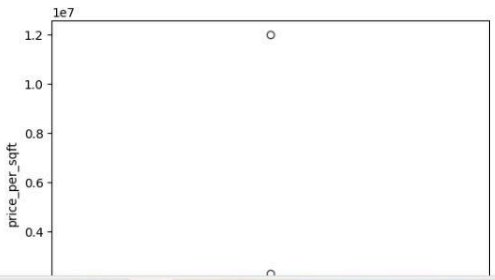
PERCENTILE METHOD

[33]:
```
upperlimit=df.price_per_sqft.quantile(0.95)
lowerlimit=df.price_per_sqft.quantile(0.05)
print(upperlimit)
print(lowerlimit)
```

```
15312.099999999984
3107.8500000000004
```

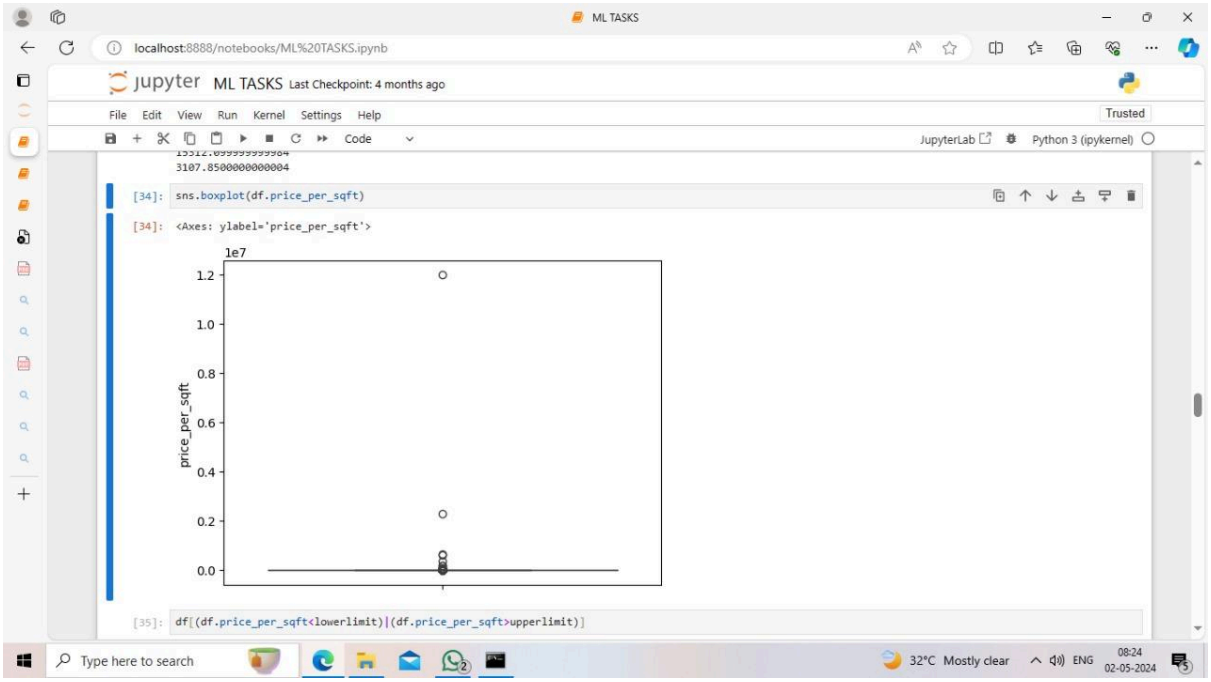
[34]:
```
sns.boxplot(df.price_per_sqft)
```

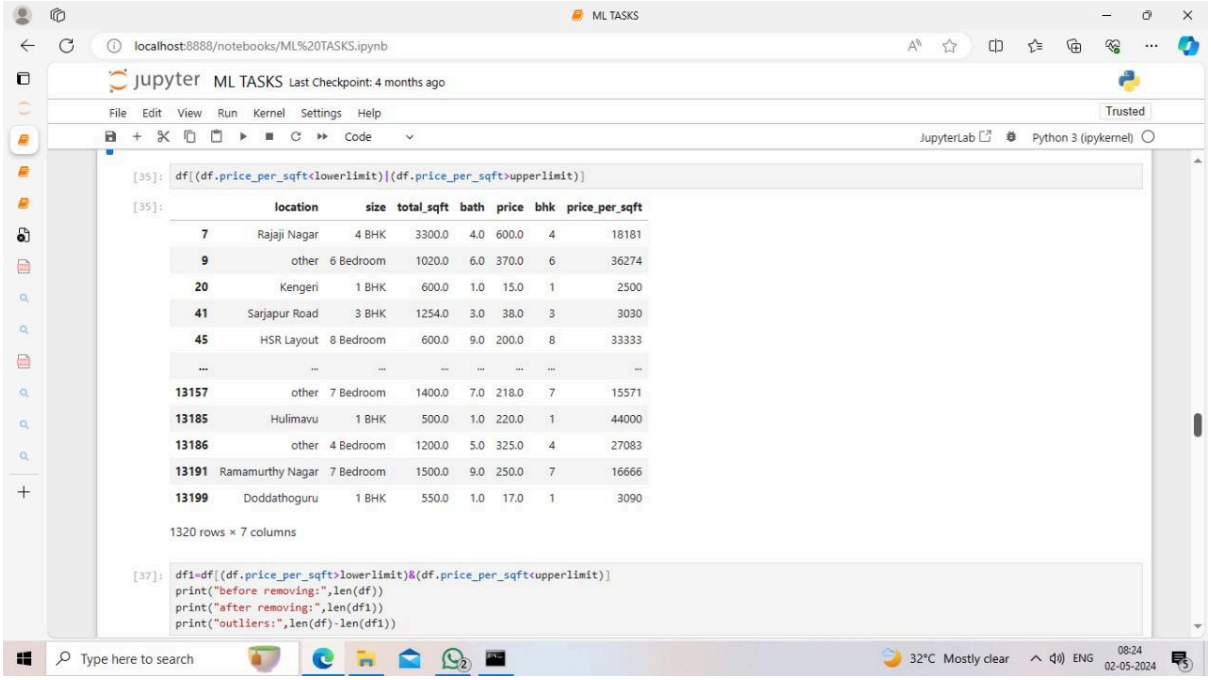
[34]: <Axes: ylabel='price_per_sqft'>



Type here to search

32°C Mostly clear ENG 08:23 02-05-2024





	location	size	total_sqft	bath	price	bhk	price_per_sqft
7	Rajaji Nagar	4 BHK	3300.0	4.0	600.0	4	18181
9	other	6 Bedroom	1020.0	6.0	370.0	6	36274
20	Kengeri	1 BHK	600.0	1.0	15.0	1	2500
41	Sarjapur Road	3 BHK	1254.0	3.0	38.0	3	3030
45	HSR Layout	8 Bedroom	600.0	9.0	200.0	8	33333
...	...	...	...	...	...	...	...
13157	other	7 Bedroom	1400.0	7.0	218.0	7	15571
13185	Hulimavu	1 BHK	500.0	1.0	220.0	1	44000
13186	other	4 Bedroom	1200.0	5.0	325.0	4	27083
13191	Ramamurthy Nagar	7 Bedroom	1500.0	9.0	250.0	7	16666
13199	Doddathoguru	1 BHK	550.0	1.0	17.0	1	3090

ML TASKS

localhost:8888/notebooks/ML%20TASKS.ipynb

jupyter ML TASKS Last Checkpoint: 4 months ago

File Edit View Run Kernel Settings Help

Trusted

Code

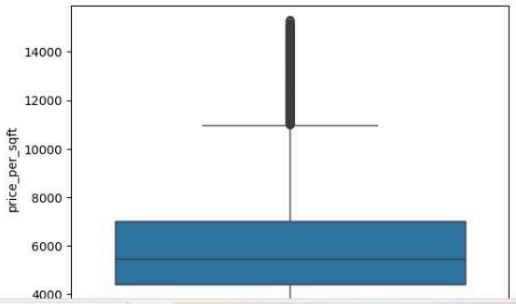
JupyterLab Python 3 (ipykernel)

```
before removing: 13200
after removing: 11880
outliers: 1320
```

[ ]:
```
df1=df.copy()
df1.loc[(df1.price_per_sqft>upperlimit),"price_per_sqft"]=upperlimit
df1.loc[(df1.price_per_sqft<upperlimit),"price_per_sqft"]=lowerlimit
```

[38]:
```
sns.boxplot(df1.price_per_sqft)
```

[38]:
```
<Axes: ylabel='price_per_sqft'>
```



Type here to search

32°C Mostly clear ENG 08:25 02-05-2024

ML TASKS

localhost:8888/notebooks/ML%20TASKS.ipynb

jupyter ML TASKS Last Checkpoint: 4 months ago

File Edit View Run Kernel Settings Help

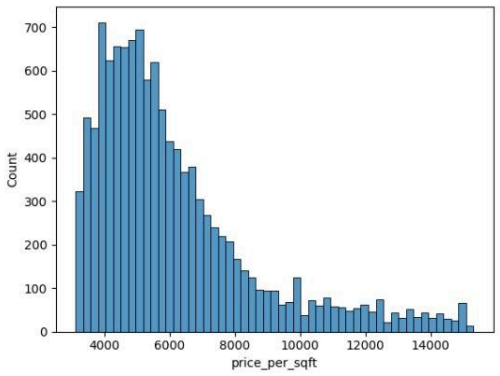
Trusted

Code

JupyterLab Python 3 (ipykernel)

[39]:
```
sns.histplot(df1.price_per_sqft)
```

[39]:
```
<Axes: xlabel='price_per_sqft', ylabel='Count'>
```



Type here to search

32°C Mostly clear ENG 08:25 02-05-2024

ML TASKS

localhost:8888/notebooks/ML%20TASKS.ipynb

jupyter ML TASKS Last Checkpoint: 4 months ago

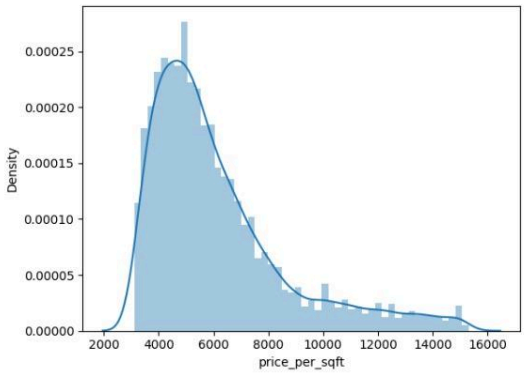
File Edit View Run Kernel Settings Help

Trusted

Code

JupyterLab Python 3 (ipykernel)

[40]: <Axes: xlabel='price_per_sqft', ylabel='Density'>



[41]:
```
df.price_per_sqft.skew()
```

[41]: 108.28105529705168

Type here to search

32°C Mostly clear ENG 08:26 02-05-2024

ML TASKS

localhost:8888/notebooks/ML%20TASKS.ipynb

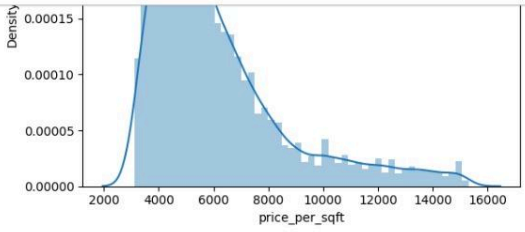
jupyter ML TASKS Last Checkpoint: 4 months ago

File Edit View Run Kernel Settings Help

Trusted

Code

JupyterLab Python 3 (ipykernel)



[41]:
```
df.price_per_sqft.skew()
```

[41]: 108.28105529705168

[45]:
```
from scipy.stats import skew,kurtosis
kurtosis(df["price_per_sqft"],fisher=False)
```

[45]: 12093.633538860382

[44]:
```
pip install scipy
```

```
Collecting scipyNote: you may need to restart the kernel to use updated packages.

[notice] A new release of pip is available: 23.2.1 -> 24.0
```

Type here to search

32°C Mostly clear ENG 08:26 02-05-2024

ML TASKS

localhost:8888/notebooks/ML%20TASKS.ipynb

jupyter ML TASKS Last Checkpoint: 4 months ago

File Edit View Run Kernel Settings Help

Trusted

Code

JupyterLab Python 3 (ipykernel)

```
Requirement already satisfied: numpy<2.3,>=1.22.4 in c:\users\shabna\appdata\local\programs\python\python311\lib\site-packages (from scipy) (1.26.2)
Downloading scipy-1.13.0-cp311-cp311-win_amd64.whl (46.2 MB)
```

[46]:
```
df["logprice"]=np.log(df["price_per_sqft"])
df["logprice"]
```

[46]:
```
0        8.215818
1        8.437067
2        8.367532
3        8.739536
4        8.354674
           ...
13195    8.808220
13196    9.315691
13197    8.567506
13198    9.250234
13199    8.035926
Name: logprice, Length: 13200, dtype: float64
```

[47]:
```
sns.distplot(df.logprice)
```

```
C:\Users\shabna\AppData\Local\Temp\ipykernel_240\4043559899.py:1: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df.logprice)
```

Type here to search

32°C Mostly clear ENG 08:27 02-05-2024

ML TASKS

localhost:8888/notebooks/ML%20TASKS.ipynb

jupyter ML TASKS Last Checkpoint: 4 months ago

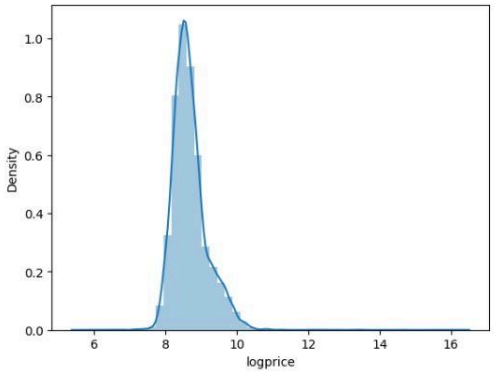
File Edit View Run Kernel Settings Help

Trusted

Code

JupyterLab Python 3 (ipykernel)

[47]:
```
<Axes: xlabel='logprice', ylabel='Density'>
```



[59]:
```
df.logprice.skew()
```

[59]:
```
1.3998626551353
```

Type here to search

31°C Partly cloudy ENG 11:27 02-05-2024

ML TASKS

localhost:8888/notebooks/ML%20TASKS.ipynb

jupyter ML TASKS Last Checkpoint: 4 months ago

File Edit View Run Kernel Settings Help

Trusted

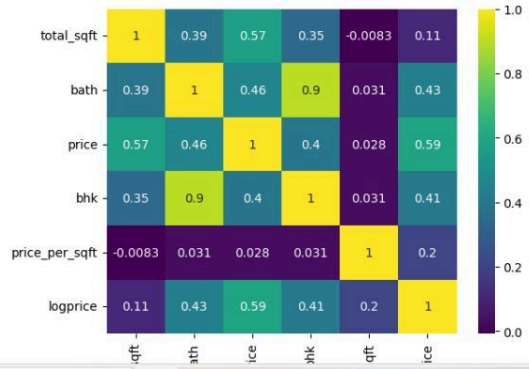
Code

JupyterLab Python 3 (ipykernel)

[59]:
```
1.3998626551353
```

[60]:
```
df1=df.drop(["location","size"],axis="columns")
correlate= df1.corr()
sns.heatmap(correlate,annot=True,cmap="viridis")
```

[60]:
```
<Axes: >
```



Type here to search

31°C Partly cloudy ENG 11:27 02-05-2024

ML TASKS

localhost:8888/notebooks/ML%20TASKS.ipynb

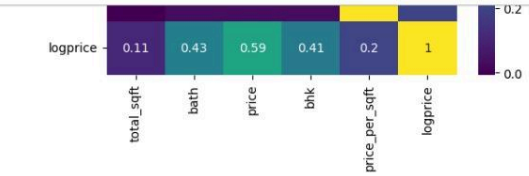
jupyter ML TASKS Last Checkpoint: 4 months ago

File Edit View Run Kernel Settings Help

Trusted

Code

JupyterLab Python 3 (ipykernel)



MEAN

[62]:
```
mean=df['price_per_sqft'].mean()
s=df['price_per_sqft'].std()
mean,s
```

[62]:
```
(7920.336742424242, 106727.16032810867)
```

[57]:
```
threshold=3*s
```

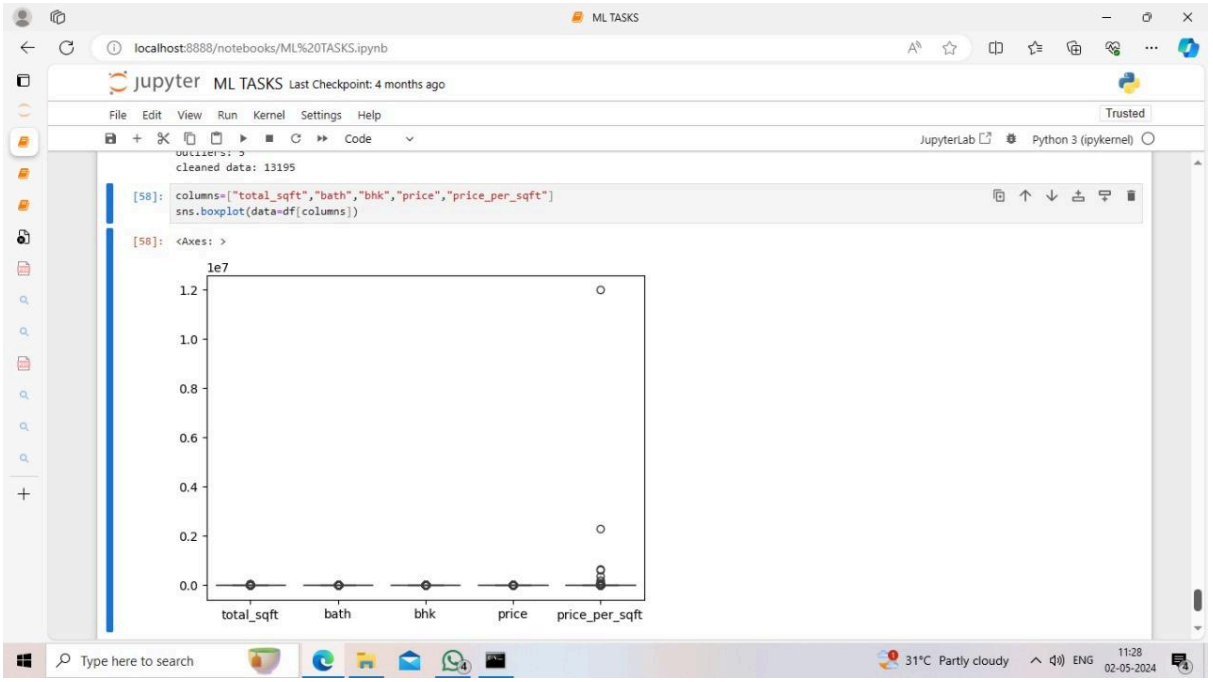
[57]:
```
320181.480984326
```

[66]:
```
outliers=df[(df['price_per_sqft']<mean-threshold)|(df['price_per_sqft']>mean+threshold)]
cleaned_data=df[(df['price_per_sqft']>=mean-threshold)&(df['price_per_sqft']<=mean+threshold)]
print("outliers:",len(outliers))
print("cleaned data:",len(cleaned_data))
```

```
outliers: 5
cleaned data: 13195
```

Type here to search

31°C Partly cloudy ENG 11:27 02-05-2024

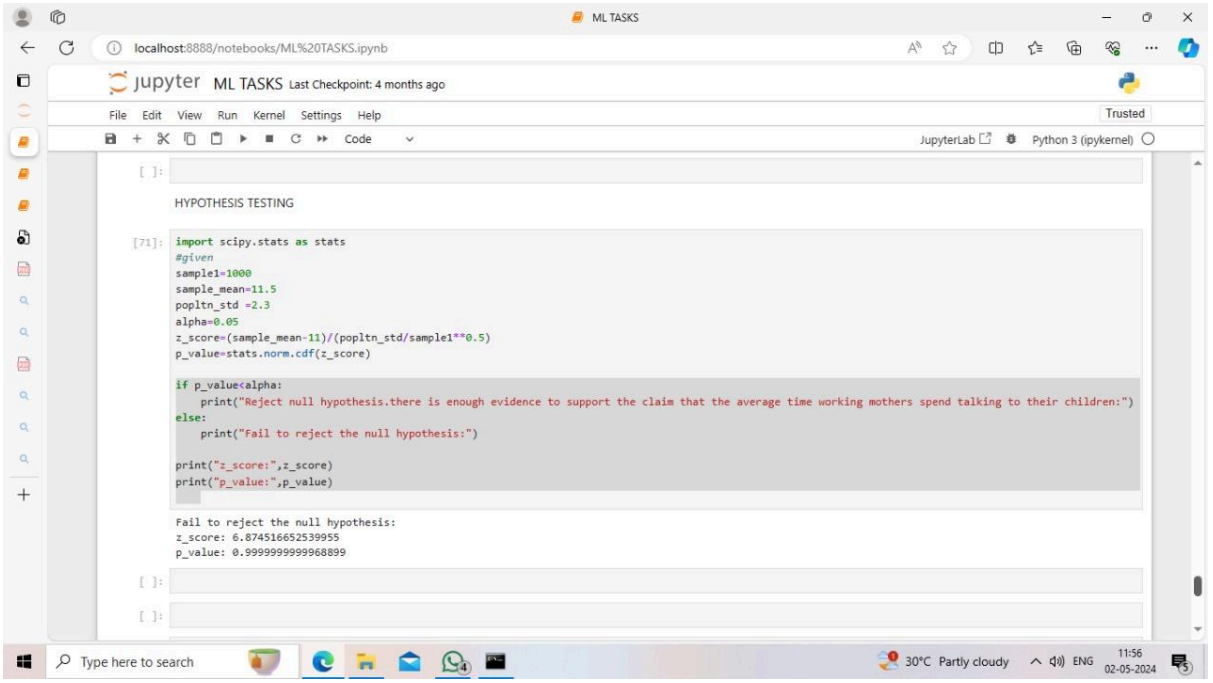


.

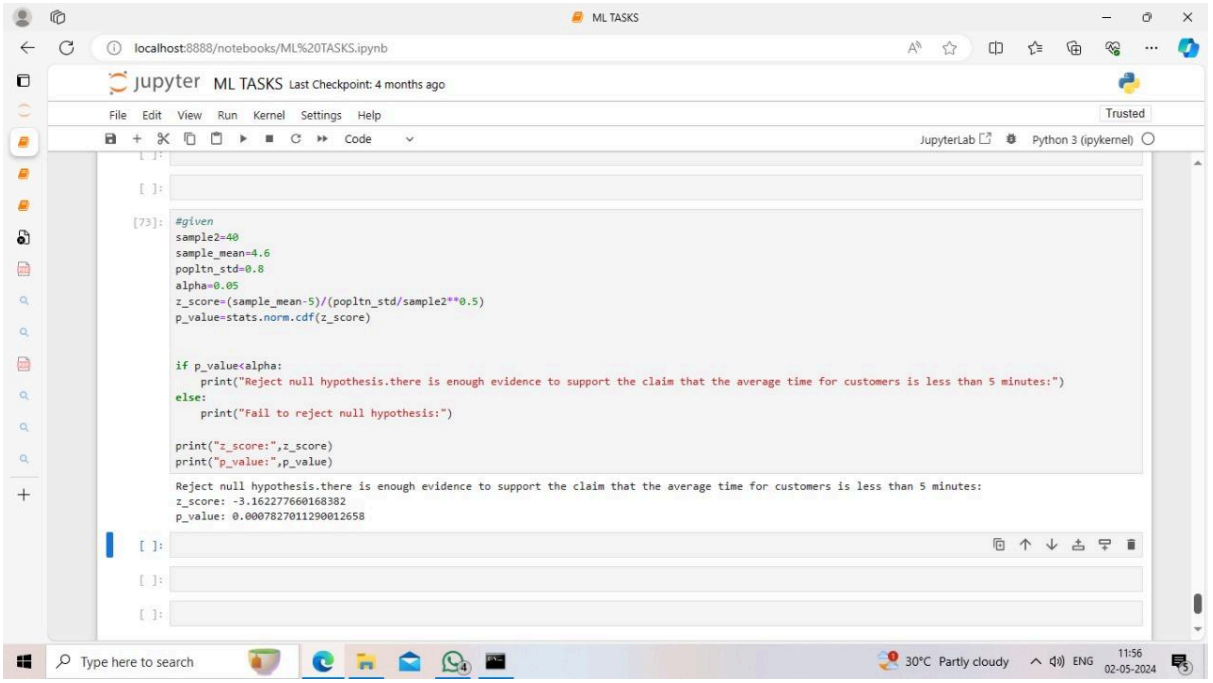
ML - Task2
- Hypothesis testing

Q1.Suppose a child psychologist claims that the average time working mothers spend talking to their children is at least 11 minutes per day. You conduct a random sample of 1000 working mothers and find they spend an average of 11.5 minutes per day talking with their children. Assume prior research suggests the population standard deviation is 2.3 minutes.Conduct a test with a level of significance of alpha = 0.05.

Hi



Q2. A coffee shop claims that their average wait time for customers is less than 5 minutes. To test this claim, a sample of 40 customers is taken, and their wait times are recorded. The sample mean wait time is found to be 4.6 minutes with a standard deviation of 0.8 minutes. Perform a hypothesis test at a significance level of 0.05 and determine whether there is enough evidence to support the coffee shop's claim.



ML - Task3

- Data Preprocessing

Objective:

The main objective of this project is to design and implement a robust data preprocessing system that addresses common challenges such as missing values, outliers, inconsistent formatting, and noise. By performing effective data preprocessing, the project aims to enhance the quality, reliability, and usefulness of the data for machine learning.

Dataset:

https://drive.google.com/file/d/1F3lRf32JM8ejnXq-Cbf9y7fa57zSHGz_/view?usp=sharing

Key Components to be fulfilled:

Data Exploration: Explore the data, list down the unique values in each feature and find its length. Perform the statistical analysis and renaming of the columns.

Data Cleaning:

Find the missing and inappropriate values, treat them appropriately. Remove all duplicate rows. Find the outliers.

- Replace the value 0 in age as NaN
- Treat the null values in all columns using any measures(removing/ replace the values with mean/median/mode)

Data Analysis:

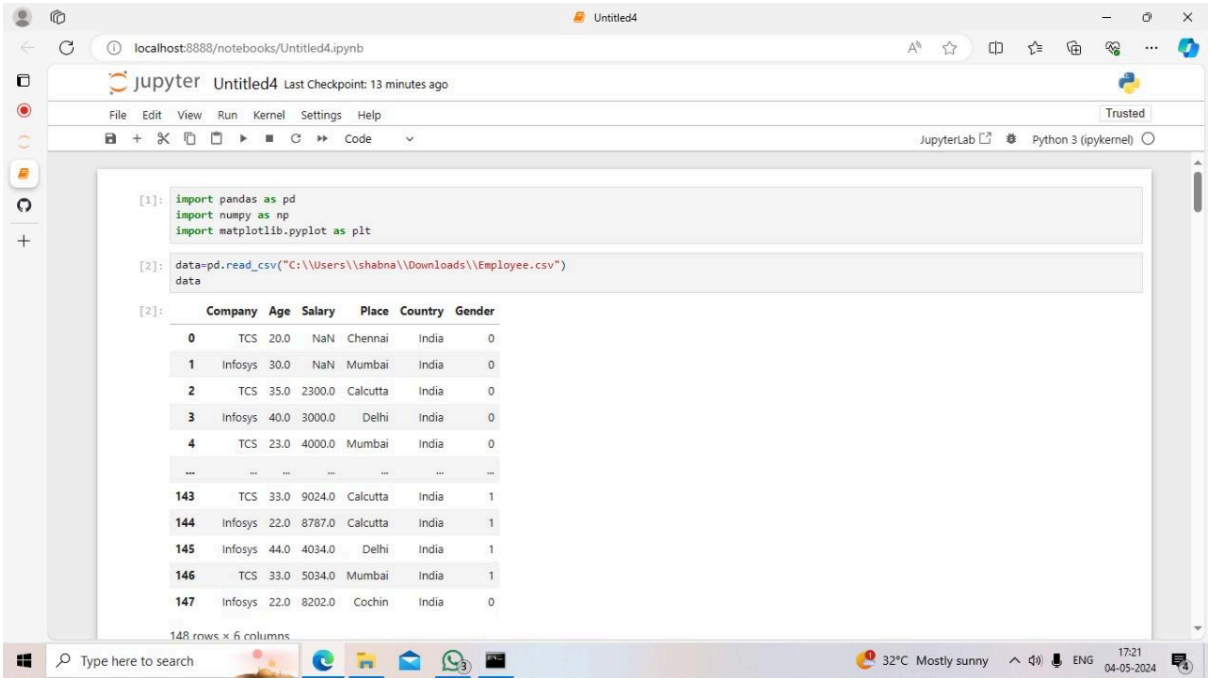
- Filter the data with age >40 and salary<5000
- Plot the chart with age and salary
- Count the number of people from each place and represent it visually

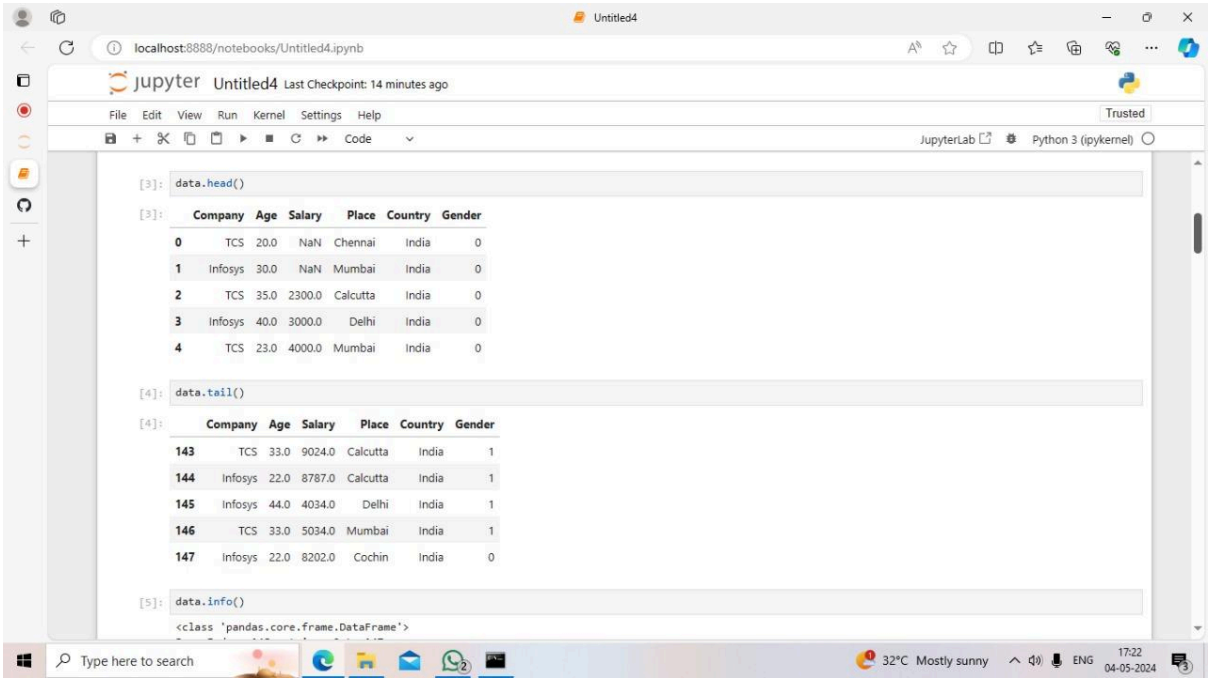
Data Encoding:

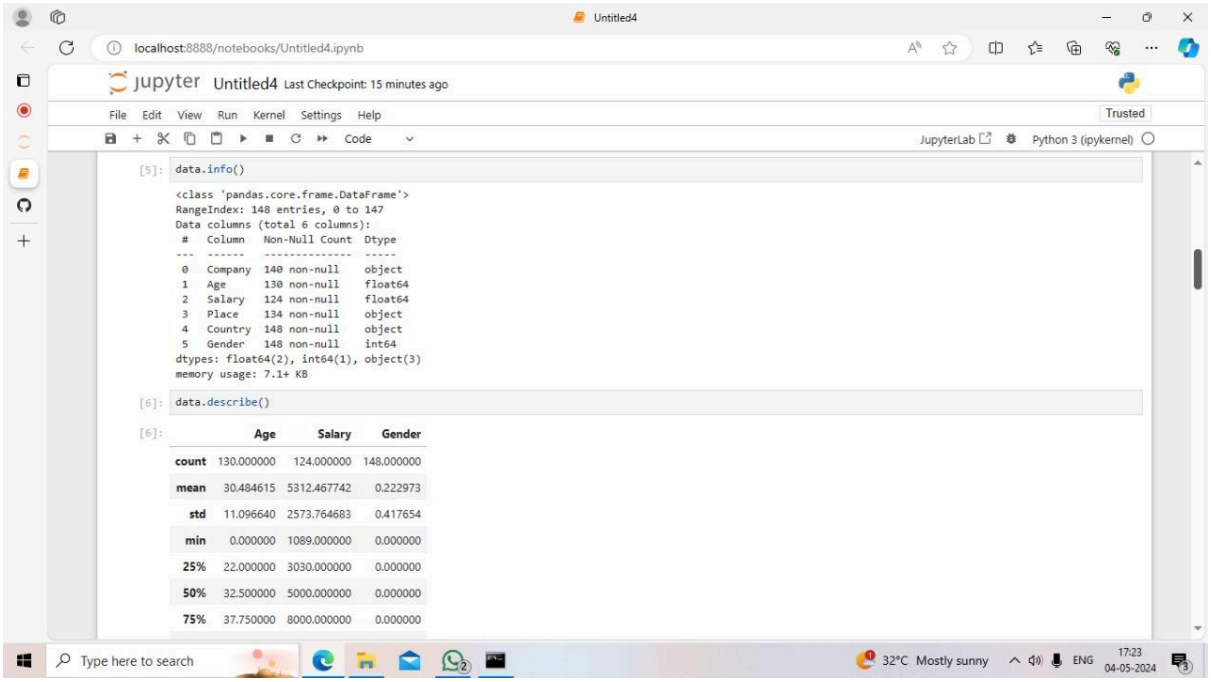
Convert categorical variables into numerical representations using techniques such as one-hot encoding, label encoding, making them suitable for analysis by machine learning algorithms.

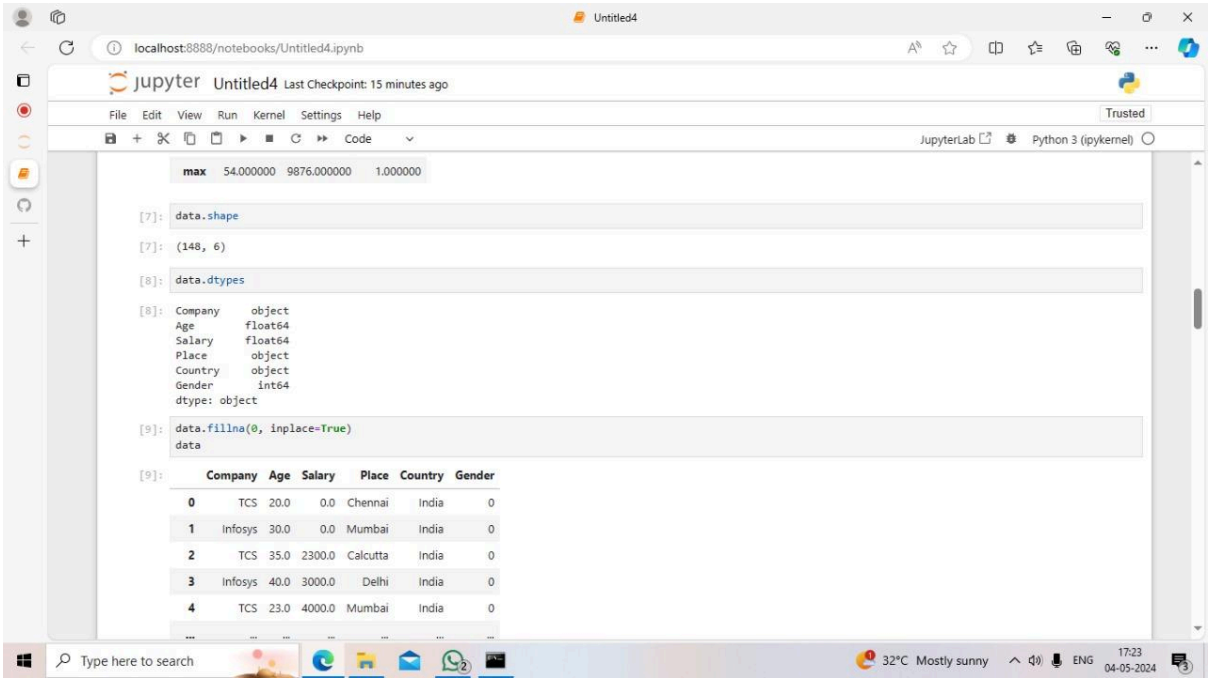
Feature Scaling:

After the process of encoding, perform the scaling of the features using standardscaler and minmaxscaler.


Untitled4
localhost:8888/notebooks/Untitled4.ipynb
jupyter Untitled4 Last Checkpoint: 13 minutes ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
[1]:
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
[2]:
data=pd.read_csv("C:\\Users\\shabna\\Downloads\\Employee.csv")
data
[2]:
Company Age Salary Place Country Gender
0 TCS 20.0 NaN Chennai India 0
1 Infosys 30.0 NaN Mumbai India 0
2 TCS 35.0 2300.0 Calcutta India 0
3 Infosys 40.0 3000.0 Delhi India 0
4 TCS 23.0 4000.0 Mumbai India 0
... ...
143 TCS 33.0 9024.0 Calcutta India 1
144 Infosys 22.0 8787.0 Calcutta India 1
145 Infosys 44.0 4034.0 Delhi India 1
146 TCS 33.0 5034.0 Mumbai India 1
147 Infosys 22.0 8202.0 Cochin India 0
148 rows × 6 columns
Type here to search
32°C Mostly sunny
ENG
17:21
04-05-2024


Untitled4
localhost:8888/notebooks/Untitled4.ipynb
jupyter Untitled4 Last Checkpoint: 14 minutes ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
[3]: data.head()
[3]:
Company Age Salary Place Country Gender
0 TCS 20.0 NaN Chennai India 0
1 Infosys 30.0 NaN Mumbai India 0
2 TCS 35.0 2300.0 Calcutta India 0
3 Infosys 40.0 3000.0 Delhi India 0
4 TCS 23.0 4000.0 Mumbai India 0
[4]: data.tail()
[4]:
Company Age Salary Place Country Gender
143 TCS 33.0 9024.0 Calcutta India 1
144 Infosys 22.0 8787.0 Calcutta India 1
145 Infosys 44.0 4034.0 Delhi India 1
146 TCS 33.0 5034.0 Mumbai India 1
147 Infosys 22.0 8202.0 Cochin India 0
[5]: data.info()
<class 'pandas.core.frame.DataFrame'>
Type here to search
32°C Mostly sunny
ENG
17:22
04-05-2024


Untitled4
localhost:8888/notebooks/Untitled4.ipynb
jupyter Untitled4 Last Checkpoint: 15 minutes ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
[5]: data.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 148 entries, 0 to 147
Data columns (total 6 columns):
Column Non-Null Count Dtype
--- ------ -------------- -----
0 Company 140 non-null object
1 Age 130 non-null float64
2 Salary 124 non-null float64
3 Place 134 non-null object
4 Country 148 non-null object
5 Gender 148 non-null int64
dtypes: float64(2), int64(1), object(3)
memory usage: 7.1+ KB
[6]: data.describe()
[6]:
Age Salary Gender
count 130.000000 124.000000 148.000000
mean 30.484615 5312.467742 0.222973
std 11.096640 2573.764683 0.417654
min 0.000000 1089.000000 0.000000
25% 22.000000 3030.000000 0.000000
50% 32.500000 5000.000000 0.000000
75% 37.750000 8000.000000 0.000000
Type here to search
32°C Mostly sunny
ENG
17:23
04-05-2024


Untitled4
localhost:8888/notebooks/Untitled4.ipynb
jupyter Untitled4 Last Checkpoint: 15 minutes ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
max 54.000000 9876.000000 1.000000
[7]: data.shape
[7]: (148, 6)
[8]: data.dtypes
[8]: Company object
Age float64
Salary float64
Place object
Country object
Gender int64
dtype: object
[9]: data.fillna(0, inplace=True)
data
[9]:
Company Age Salary Place Country Gender
0 TCS 20.0 0.0 Chennai India 0
1 Infosys 30.0 0.0 Mumbai India 0
2 TCS 35.0 2300.0 Calcutta India 0
3 Infosys 40.0 3000.0 Delhi India 0
4 TCS 23.0 4000.0 Mumbai India 0
Type here to search
32°C Mostly sunny
ENG
17:23
04-05-2024

Untitled4

localhost:8888/notebooks/Untitled4.ipynb

jupyter Untitled4 Last Checkpoint: 43 minutes ago

File Edit View Run Kernel Settings Help

Trusted

Code

JupyterLab Python 3 (ipykernel)

4	TCS	23.0	4000.0	Mumbai	India	0
...	...	...	...	...	...	...
143	TCS	33.0	9024.0	Calcutta	India	1
144	Infosys	22.0	8787.0	Calcutta	India	1
145	Infosys	44.0	4034.0	Delhi	India	1
146	TCS	33.0	5034.0	Mumbai	India	1
147	Infosys	22.0	8202.0	Cochin	India	0

148 rows × 6 columns

[10]:
```
filtered_data = data[(data['Age'] > 40) & (data['Salary'] < 5000)]

print(filtered_data)
```

```
     Company   Age  Salary      Place Country  Gender
12       CTS  45.0     0.0    Chennai   India       0
21   Infosys  50.0  3184.0      Delhi   India       0
32   Infosys  45.0  4034.0   Calcutta   India       0
39   Infosys  41.0  3000.0     Mumbai   India       0
48       CTS  43.0     0.0     Mumbai   India       0
50   Infosys  41.0  3000.0    Chennai   India       0
57   Infosys  51.0  3184.0  Hyderabad   India       0
66       CTS  41.0     0.0   Calcutta   India       0
68   Infosys  43.0  4034.0     Mumbai   India       0
75   Infosys  44.0  3000.0     Cochin   India       0
84       CTS  43.0     0.0     Mumbai   India       0
86   Infosys  41.0  3000.0      Delhi   India       0
```

Type here to search

31°C Mostly sunny ENG 17:51 04-05-2024

Untitled4

localhost:8888/notebooks/Untitled4.ipynb

jupyter Untitled4 Last Checkpoint: 44 minutes ago

File Edit View Run Kernel Settings Help

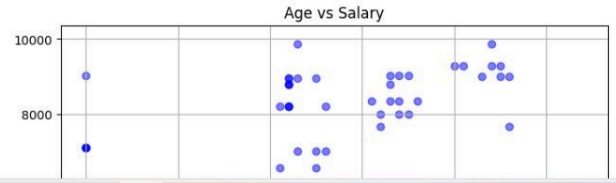
Trusted

Code

JupyterLab Python 3 (ipykernel)

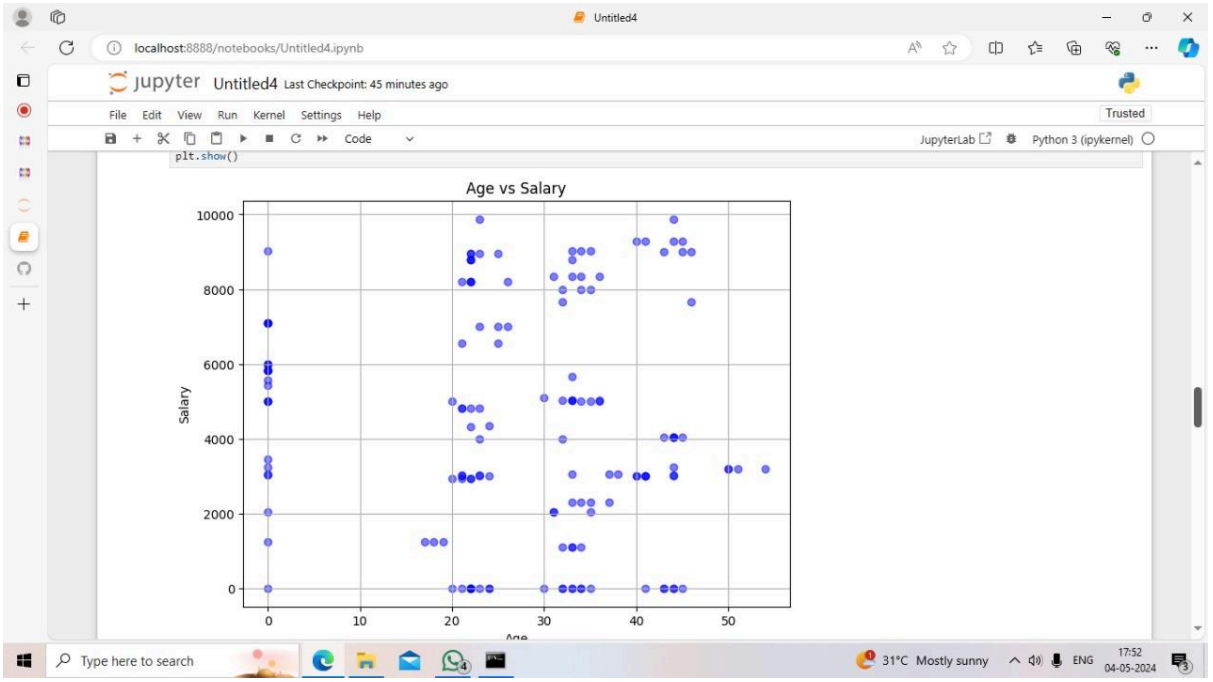
```
84       CTS  43.0     0.0     Mumbai   India       0
86   Infosys  41.0  3000.0      Delhi   India       0
93   Infosys  54.0  3184.0     Mumbai   India       0
102      CTS  44.0     0.0     Mumbai   India       0
104  Infosys  44.0  4034.0      Delhi   India       0
120      CTS  44.0     0.0  Hyderabad   India       0
122  Infosys  44.0  3234.0     Mumbai   India       0
129  Infosys  50.0  3184.0   Calcutta   India       0
138      CTS  44.0  3033.0     Cochin   India       0
140  Infosys  44.0  4034.0  Hyderabad   India       0
145  Infosys  44.0  4034.0      Delhi   India       1
```

[11]:
```
plt.figure(figsize=(8, 6))
plt.scatter(data['Age'], data['Salary'], color='blue', alpha=0.5)
plt.title('Age vs Salary')
plt.xlabel('Age')
plt.ylabel('Salary')
plt.grid(True)
plt.show()
```



Type here to search

31°C Mostly sunny ENG 17:52 04-05-2024


Untitled4
localhost:8888/notebooks/Untitled4.ipynb
jupyter Untitled4 Last Checkpoint: 45 minutes ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
plt.show()
Age vs Salary
Salary
10000
8000
6000
4000
2000
0
0
10
20
30
40
50
Type here to search
31°C Mostly sunny
ENG
17:52
04-05-2024

Untitled4

localhost:8888/notebooks/Untitled4.ipynb

jupyter Untitled4 Last Checkpoint: 45 minutes ago

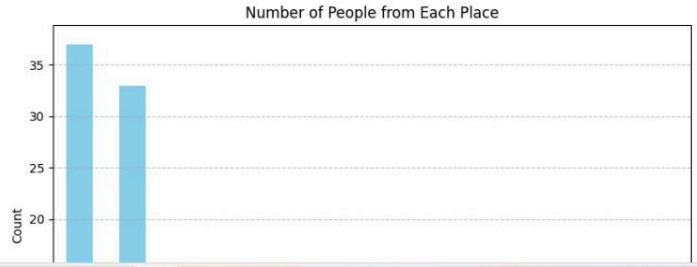
File Edit View Run Kernel Settings Help

Trusted

Code

JupyterLab Python 3 (ipykernel)

[12]:
```
place_counts = data['Place'].value_counts()

# Plotting
plt.figure(figsize=(8, 6))
place_counts.plot(kind='bar', color='skyblue')
plt.title('Number of People from Each Place')
plt.xlabel('Place')
plt.ylabel('Count')
plt.xticks(rotation=45)
plt.grid(axis='y', linestyle='--', alpha=0.7)
plt.tight_layout()
plt.show()
```



Type here to search 31°C Mostly sunny ENG 17:53 04-05-2024

Untitled4

localhost:8888/notebooks/Untitled4.ipynb

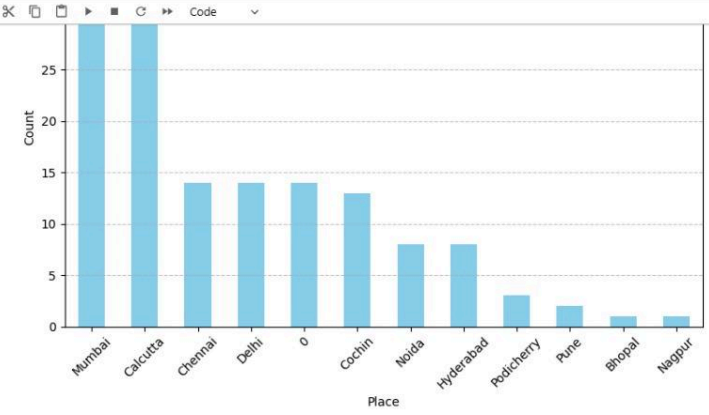
jupyter Untitled4 Last Checkpoint: 45 minutes ago

File Edit View Run Kernel Settings Help

Trusted

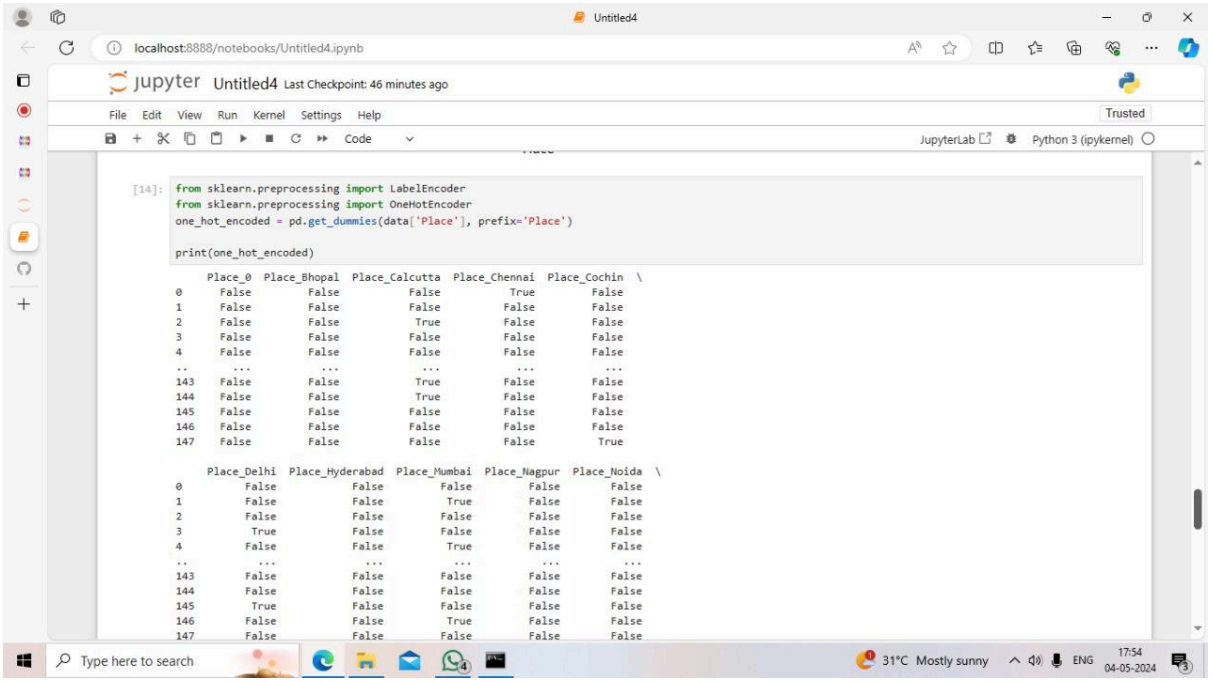
Code

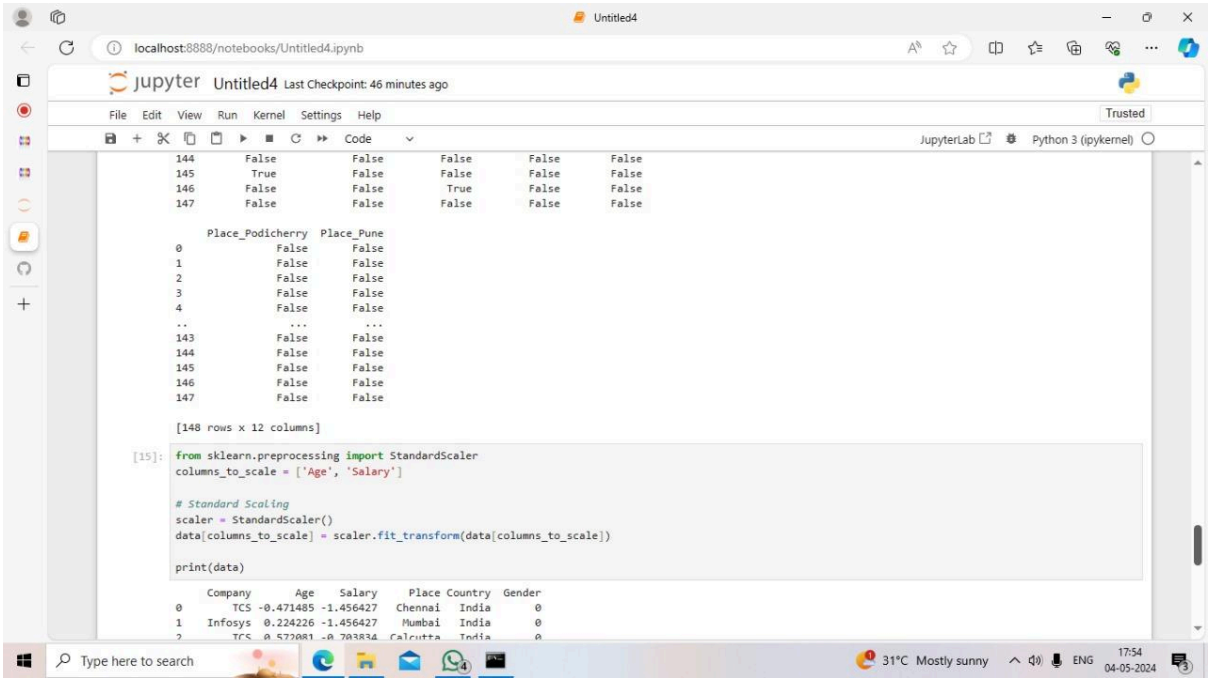
JupyterLab Python 3 (ipykernel)

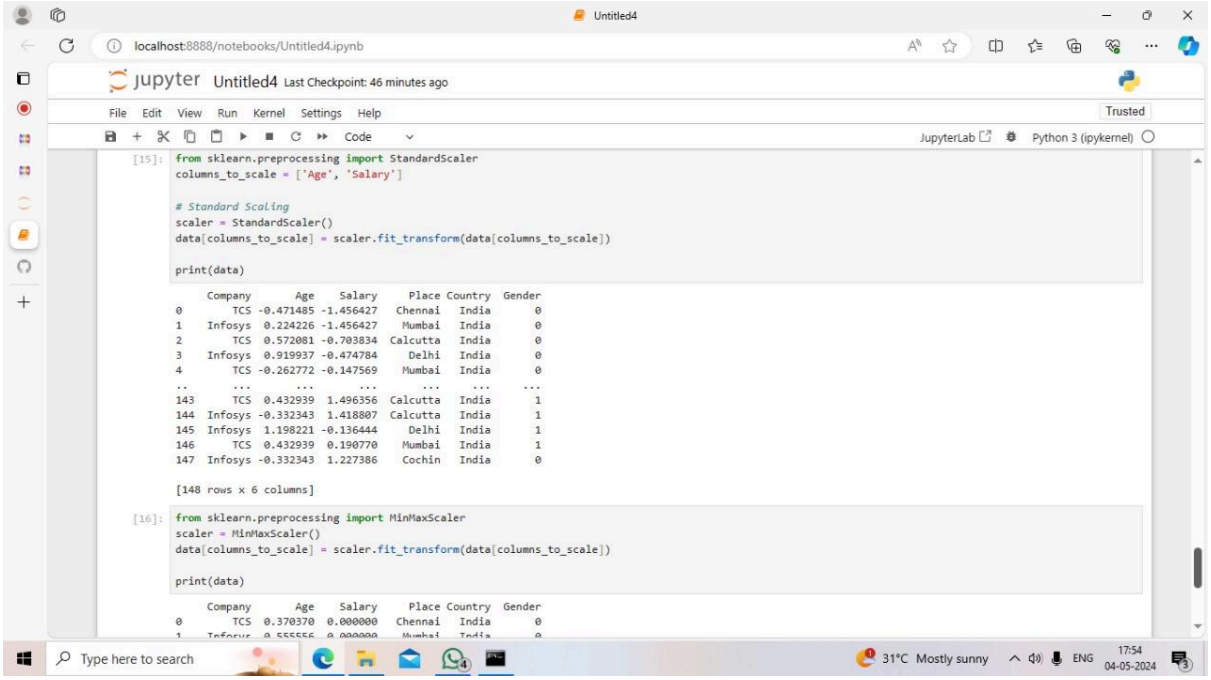


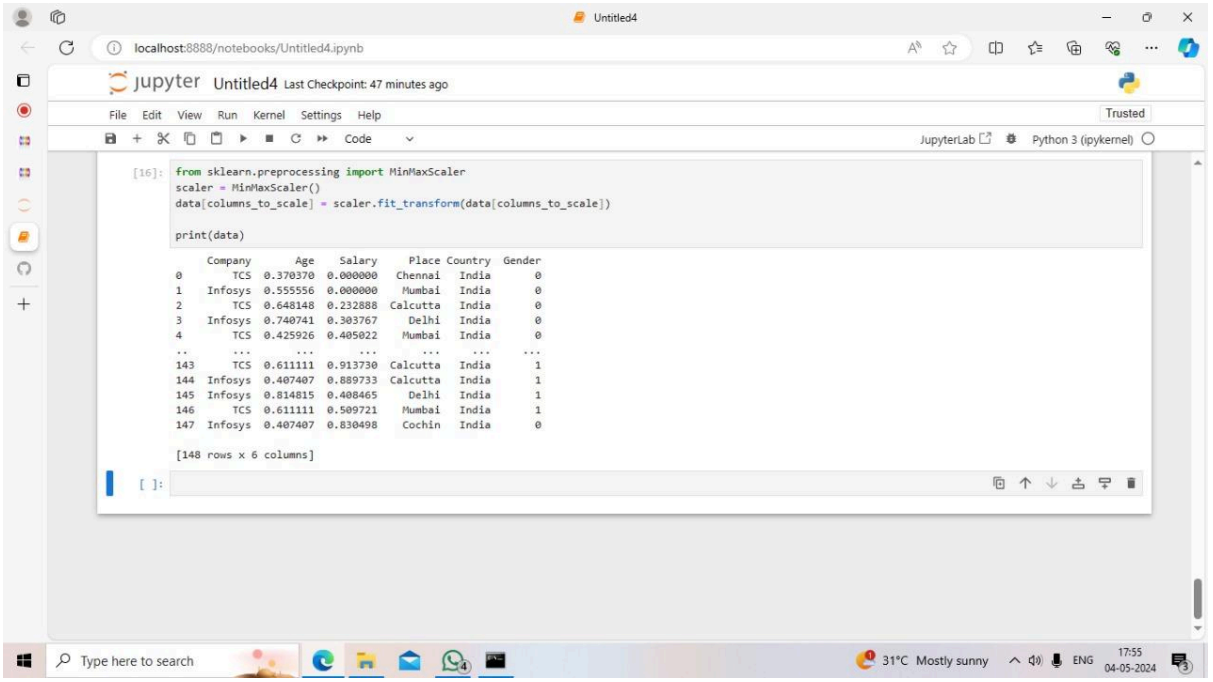
[14]:
```
from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import OneHotEncoder
one_hot_encoded = pd.get_dummies(data['Place'], prefix='Place')

print(one_hot_encoded)
```

Type here to search 31°C Mostly sunny ENG 17:53 04-05-2024


Untitled4
localhost:8888/notebooks/Untitled4.ipynb
jupyter Untitled4 Last Checkpoint: 46 minutes ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
[14]:
from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import OneHotEncoder
one_hot_encoded = pd.get_dummies(data['Place'], prefix='Place')
print(one_hot_encoded)
Place_0 Place_Bhopal Place_Calcutta Place_Chennai Place_Cochin \
0 False False False True False
1 False False False False False
2 False False True False False
3 False False False False False
4 False False False False False
.. ..
143 False False True False False
144 False False True False False
145 False False False False False
146 False False False False False
147 False False False False True
Place_Delhi Place_Hyderabad Place_Mumbai Place_Nagpur Place_Noida \
0 False False False False False
1 False False True False False
2 False False False False False
3 True False False False False
4 False False True False False
.. ..
143 False False False False False
144 False False False False False
145 True False False False False
146 False False True False False
147 False False False False False
Type here to search
31°C Mostly sunny
ENG
17:54
04-05-2024


Untitled4
localhost:8888/notebooks/Untitled4.ipynb
jupyter Untitled4 Last Checkpoint: 46 minutes ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
144 False False False False False
145 True False False False False
146 False False True False False
147 False False False False False
Place_Podicherry Place_Pune
0 False False
1 False False
2 False False
3 False False
4 False False
.. ..
143 False False
144 False False
145 False False
146 False False
147 False False
[148 rows x 12 columns]
[15]:
from sklearn.preprocessing import StandardScaler
columns_to_scale = ['Age', 'Salary']
# Standard Scaling
scaler = StandardScaler()
data[columns_to_scale] = scaler.fit_transform(data[columns_to_scale])
print(data)
Company Age Salary Place Country Gender
0 TCS -0.471485 -1.456427 Chennai India 0
1 Infosys 0.224226 -1.456427 Mumbai India 0
Type here to search
31°C Mostly sunny
ENG
17:54
04-05-2024


Untitled4
localhost:8888/notebooks/Untitled4.ipynb
jupyter Untitled4 Last Checkpoint: 46 minutes ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
[15]:
from sklearn.preprocessing import StandardScaler
columns_to_scale = ['Age', 'Salary']
# Standard Scaling
scaler = StandardScaler()
data[columns_to_scale] = scaler.fit_transform(data[columns_to_scale])
print(data)
Company Age Salary Place Country Gender
0 TCS -0.471485 -1.456427 Chennai India 0
1 Infosys 0.224226 -1.456427 Mumbai India 0
2 TCS 0.572081 -0.703834 Calcutta India 0
3 Infosys 0.919937 -0.474784 Delhi India 0
4 TCS -0.262772 -0.147569 Mumbai India 0
.. ..
143 TCS 0.432939 1.496356 Calcutta India 1
144 Infosys -0.332343 1.418807 Calcutta India 1
145 Infosys 1.198221 -0.136444 Delhi India 1
146 TCS 0.432939 0.190770 Mumbai India 1
147 Infosys -0.332343 1.227386 Cochin India 0
[148 rows x 6 columns]
[16]:
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()
data[columns_to_scale] = scaler.fit_transform(data[columns_to_scale])
print(data)
Company Age Salary Place Country Gender
0 TCS 0.370370 0.000000 Chennai India 0
Type here to search
31°C Mostly sunny
ENG
17:54
04-05-2024



ML - Task4
- Regression

Problem Description
A Chinese automobile company aspires to enter the US market by setting up their manufacturing unit there and producing cars locally to give competition to their US and European counterparts. They have contracted an automobile consulting company to understand the factors on which the pricing of cars depends. Specifically, they want to understand the factors affecting the pricing of cars in the American market, since those may be very different from the Chinese market. Essentially, the company wants to know:

• Which variables are significant in predicting the price of a car
• How well those variables describe the price of a car
Based on various market surveys, the consulting firm has gathered a large dataset of different types of cars across the American market.

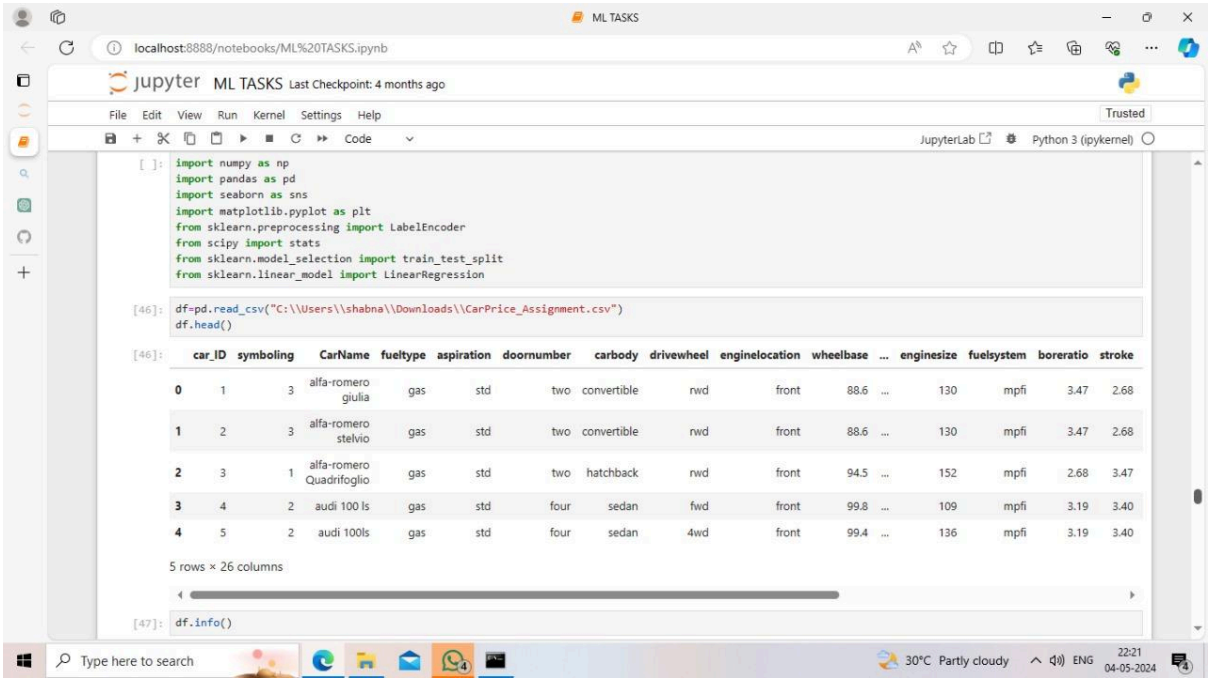
Business Goal
You are required to model the price of cars with the available independent variables. It will be used by the management to understand how exactly the prices vary with the independent variables. They can accordingly manipulate the design of the cars, the business strategy etc. to meet certain price levels. Further, the model will be a good way for the management to

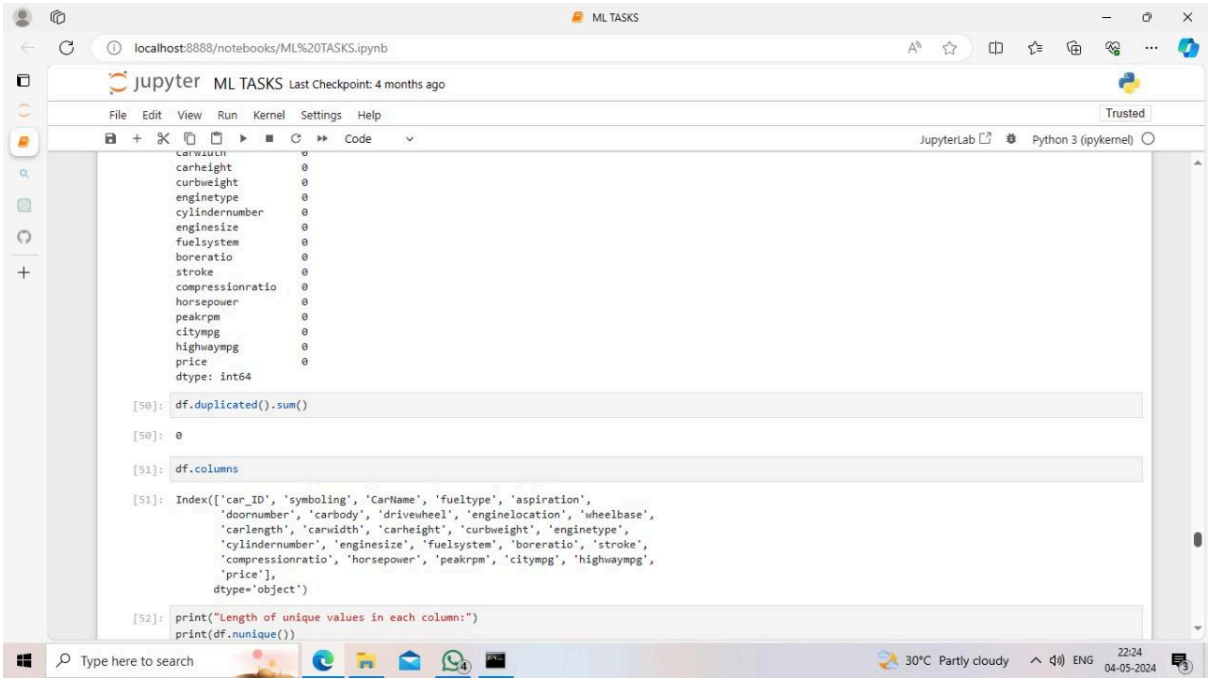
understand the pricing dynamics of a new market.

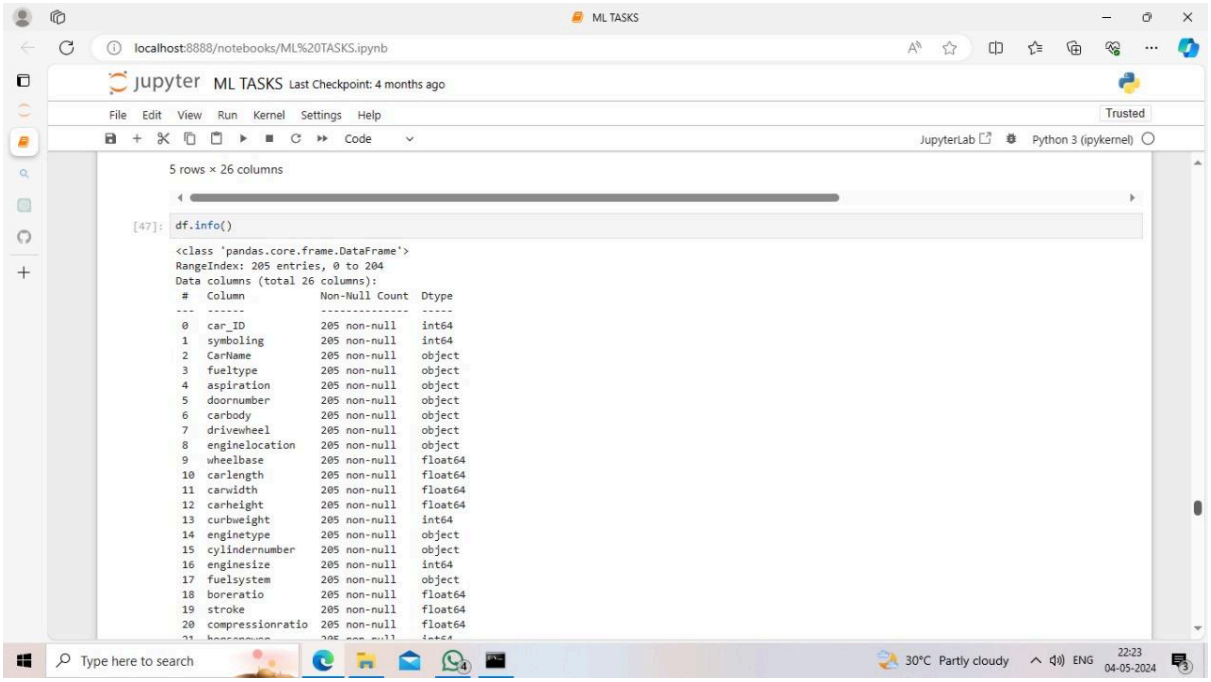
Dataset:
https://drive.google.com/file/d/1FHmYNLs9v0Enc-UExEMpitOFGsWvB2dP/view?usp=drive_link

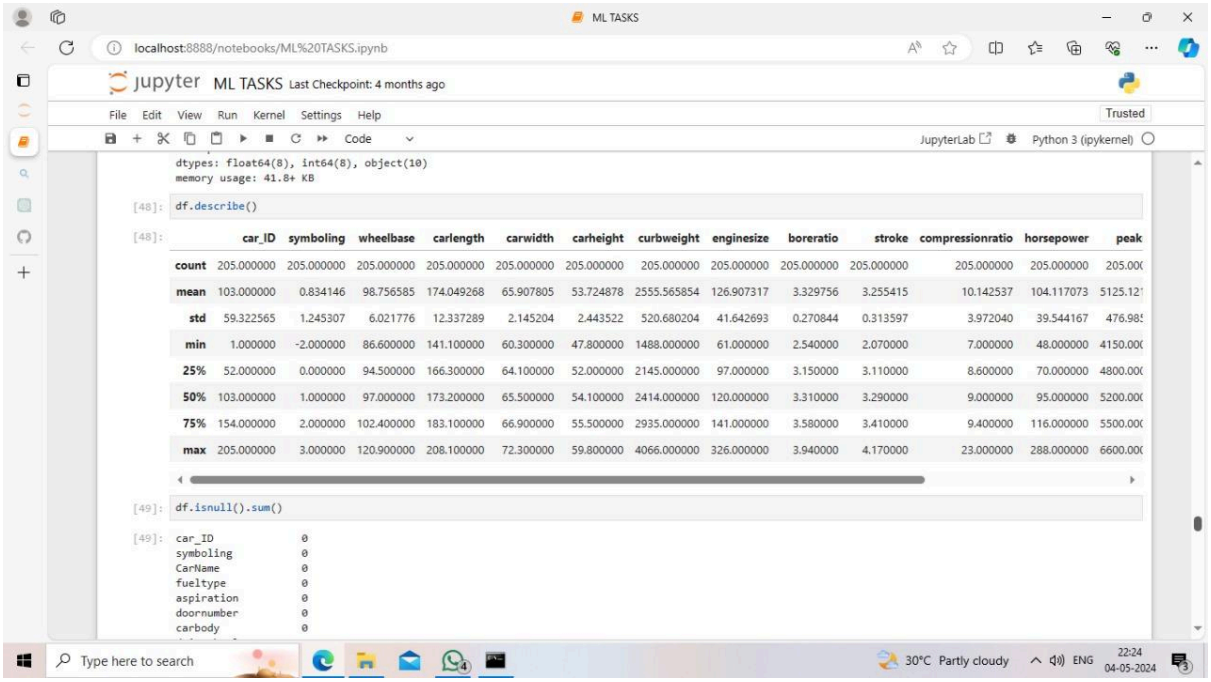
Dear students,
Apply any 5 algorithms to the regression problem provided.
For example:
Linear Regression
Decision Tree Regressor
Random Forest Regressor
Gradient Boosting Regressor
Support Vector Regressor

Rootmap:
1. Understand problem statement
2. Import necessary libraries and data
3. Check the data
Info()
Describe((
Isnull()
Duplicated()
Df. Columns
Length of unique values in each column.
4. Data preprocessing
Drop car id
Find unique values in categorical or count plot
extract company name from car name and address this new col to df also remove car name column.
There are spelling mistakes in company name. Treat this.
Label encoding all the categorical columns
Outliers detection and removal(if present)
5. Feature selection
Find correlation matrix
Remove multicolinearity (remove features with High correlation .85 to 1)
6. Data splitting
Test, train
7. Model selection and implementation
8. Model evaluation


ML TASKS
localhost:8888/notebooks/ML%20TASKS.ipynb
jupyter ML TASKS Last Checkpoint: 4 months ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
[ ]:
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.preprocessing import LabelEncoder
from scipy import stats
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
[46]:
df=pd.read_csv("C:\\Users\\shabna\\Downloads\\CarPrice_Assignment.csv")
df.head()
[46]:
car_ID symboling CarName fueltype aspiration doornumber carbody drivewheel enginelocation wheelbase ... enginesize fuelsystem boreratio stroke
0 1 3 alfa-romero giulia gas std two convertible rwd front 88.6 ... 130 mpfi 3.47 2.68
1 2 3 alfa-romero stelvio gas std two convertible rwd front 88.6 ... 130 mpfi 3.47 2.68
2 3 1 alfa-romero Quadrifoglio gas std two hatchback rwd front 94.5 ... 152 mpfi 2.68 3.47
3 4 2 audi 100 ls gas std four sedan fwd front 99.8 ... 109 mpfi 3.19 3.40
4 5 2 audi 100ls gas std four sedan 4wd front 99.4 ... 136 mpfi 3.19 3.40
5 rows × 26 columns
[47]:
df.info()
Type here to search
30°C Partly cloudy
ENG
22:21
04-05-2024


ML TASKS
localhost:8888/notebooks/ML%20TASKS.ipynb
jupyter ML TASKS Last Checkpoint: 4 months ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
carwidth 0
carheight 0
curbweight 0
enginetype 0
cylindernumber 0
enginesize 0
fuelsystem 0
boreratio 0
stroke 0
compressionratio 0
horsepower 0
peakrpm 0
citympg 0
highwaympg 0
price 0
dtype: int64
[50]:
df.duplicated().sum()
[50]:
0
[51]:
df.columns
[51]:
Index(['car_ID', 'symboling', 'CarName', 'fueltype', 'aspiration',
'doornumber', 'carbody', 'drivewheel', 'enginelocation', 'wheelbase',
'carlength', 'carwidth', 'carheight', 'curbweight', 'enginetype',
'cylindernumber', 'enginesize', 'fuelsystem', 'boreratio', 'stroke',
'compressionratio', 'horsepower', 'peakrpm', 'citympg', 'highwaympg',
'price'],
dtype='object')
[52]:
print("Length of unique values in each column:")
print(df.nunique())
Type here to search
30°C Partly cloudy
ENG
22:24
04-05-2024


ML TASKS
localhost:8888/notebooks/ML%20TASKS.ipynb
jupyter ML TASKS Last Checkpoint: 4 months ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
5 rows × 26 columns
[47]: df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 205 entries, 0 to 204
Data columns (total 26 columns):
Column Non-Null Count Dtype
--- ------ -------------- -----
0 car_ID 205 non-null int64
1 symboling 205 non-null int64
2 CarName 205 non-null object
3 fueltype 205 non-null object
4 aspiration 205 non-null object
5 doornumber 205 non-null object
6 carbody 205 non-null object
7 drivewheel 205 non-null object
8 enginelocation 205 non-null object
9 wheelbase 205 non-null float64
10 carlength 205 non-null float64
11 carwidth 205 non-null float64
12 carheight 205 non-null float64
13 curbweight 205 non-null int64
14 enginetype 205 non-null object
15 cylindernumber 205 non-null object
16 enginesize 205 non-null int64
17 fuelsystem 205 non-null object
18 boreratio 205 non-null float64
19 stroke 205 non-null float64
20 compressionratio 205 non-null float64
Type here to search
30°C Partly cloudy
ENG
22:23
04-05-2024


ML TASKS
localhost:8888/notebooks/ML%20TASKS.ipynb
jupyter ML TASKS Last Checkpoint: 4 months ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
dtypes: float64(8), int64(8), object(10)
memory usage: 41.8+ KB
[48]: df.describe()
[48]:
car_ID symboling wheelbase carlength carwidth carheight curbweight enginesize boreratio stroke compressionratio horsepower peak
count 205.000000 205.000000 205.000000 205.000000 205.000000 205.000000 205.000000 205.000000 205.000000 205.000000 205.000000 205.000000 205.00(
mean 103.000000 0.834146 98.756585 174.049268 65.907805 53.724878 2555.565854 126.907317 3.329756 3.255415 10.142537 104.117073 5125.12
std 59.322565 1.245307 6.021776 12.337289 2.145204 2.443522 520.680204 41.642693 0.270844 0.313597 3.972040 39.544167 476.98
min 1.000000 -2.000000 86.600000 141.100000 60.300000 47.800000 1488.000000 61.000000 2.540000 2.070000 7.000000 48.000000 4150.00(
25% 52.000000 0.000000 94.500000 166.300000 64.100000 52.000000 2145.000000 97.000000 3.150000 3.110000 8.600000 70.000000 4800.00(
50% 103.000000 1.000000 97.000000 173.200000 65.500000 54.100000 2414.000000 120.000000 3.310000 3.290000 9.000000 95.000000 5200.00(
75% 154.000000 2.000000 102.400000 183.100000 66.900000 55.500000 2935.000000 141.000000 3.580000 3.410000 9.400000 116.000000 5500.00(
max 205.000000 3.000000 120.900000 208.100000 72.300000 59.800000 4066.000000 326.000000 3.940000 4.170000 23.000000 288.000000 6600.00(
[49]: df.isnull().sum()
[49]: car_ID 0
symboling 0
CarName 0
fueltype 0
aspiration 0
doornumber 0
carbody 0
Type here to search
30°C Partly cloudy
ENG
22:24
04-05-2024

ML TASKS

localhost:8888/notebooks/ML%20TASKS.ipynb

jupyter ML TASKS Last Checkpoint: 4 months ago

File Edit View Run Kernel Settings Help

Trusted

Code

JupyterLab Python 3 (ipykernel)

```
dtype='object')
```

[52]:
```
print("Length of unique values in each column:")
print(df.nunique())
```

```
Length of unique values in each column:
car_ID              205
symboling             6
CarName             147
fueltype              2
aspiration            2
doornumber            2
carbody               5
drivewheel            3
enginelocation        2
wheelbase            53
carlength            75
carwidth             44
carheight            49
curbweight          171
enginetype            7
cylindernumber        7
enginesize           44
fuelsystem            8
boreratio            38
stroke               37
compressionratio     32
horsepower           59
peakrpm              23
citympg              29
highwaympg           30
price               189
dtype: int64
```

Type here to search

30°C Partly cloudy ENG 22:25 04-05-2024

ML TASKS

localhost:8888/notebooks/ML%20TASKS.ipynb

jupyter ML TASKS Last Checkpoint: 4 months ago

File Edit View Run Kernel Settings Help

Trusted

Code

JupyterLab Python 3 (ipykernel)

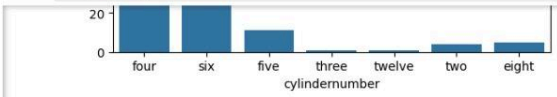
```
price               189
dtype: int64
```

[53]:
```
len(df['car_ID'])
```

[53]:
```
205
```

[ ]:
```
df.drop(['car_ID'],axis=1,inplace=True)
```

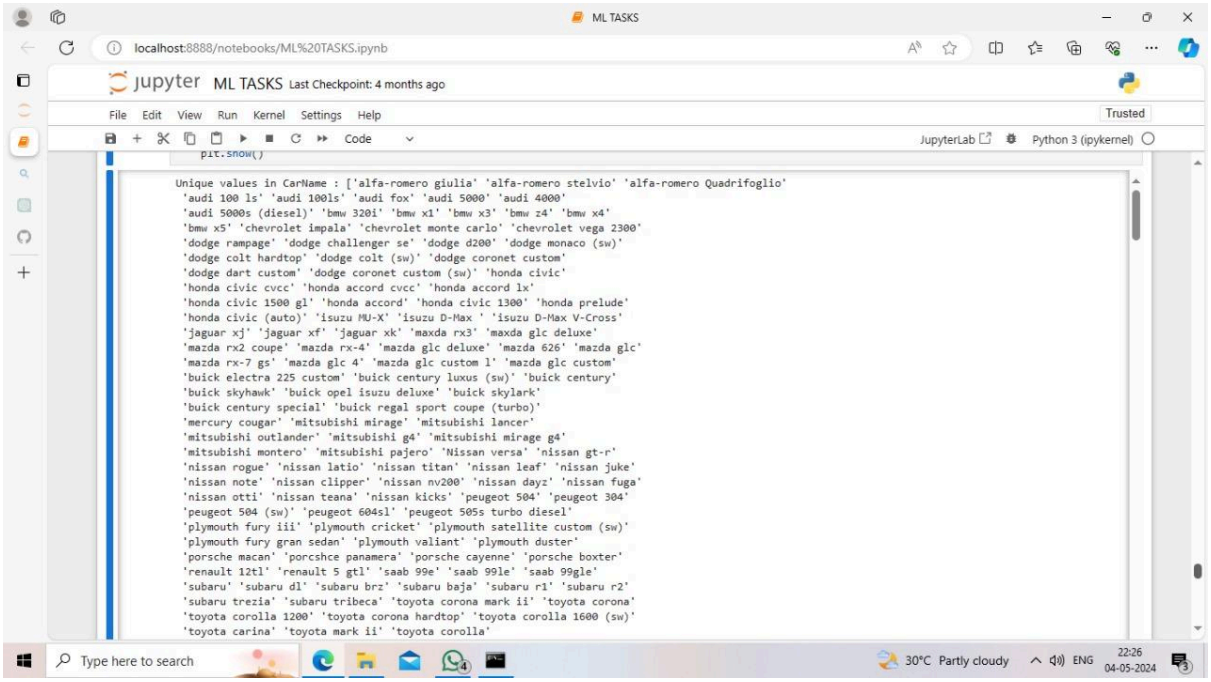
[54]:
```
categorical_cols = df.select_dtypes(include=['object']).columns
for col in categorical_cols:
    print("Unique values in", col, ":", df[col].unique())

    sns.countplot(x=col, data=df)
    plt.show()
```

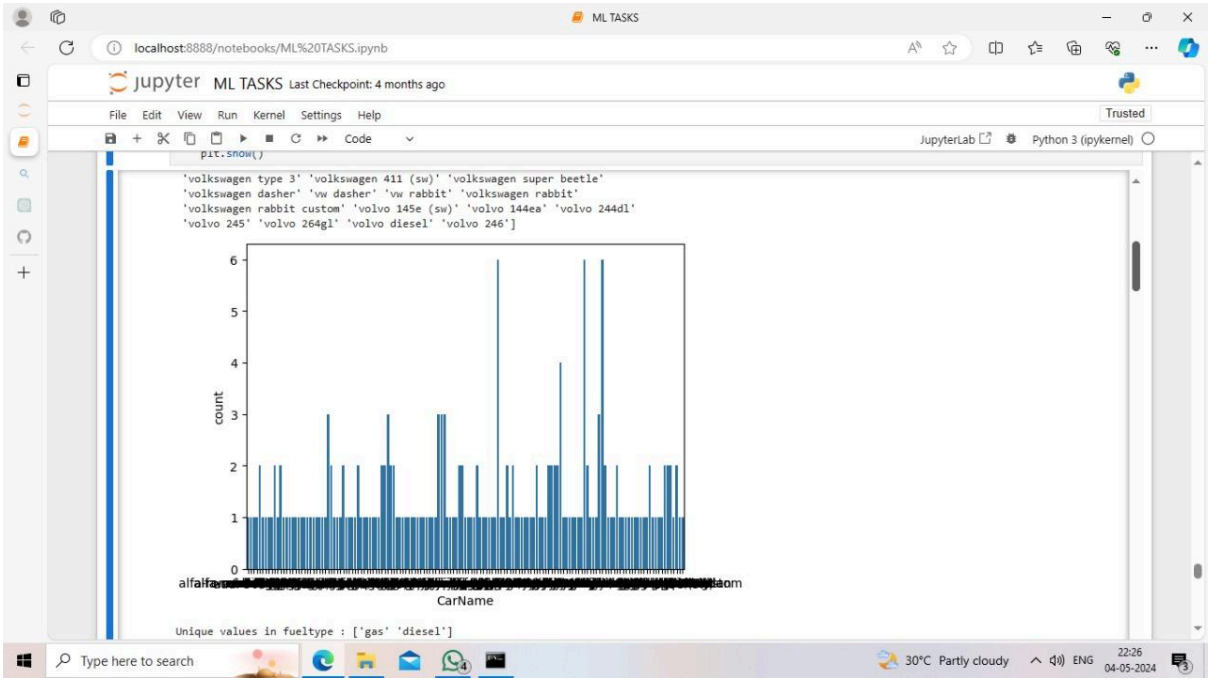


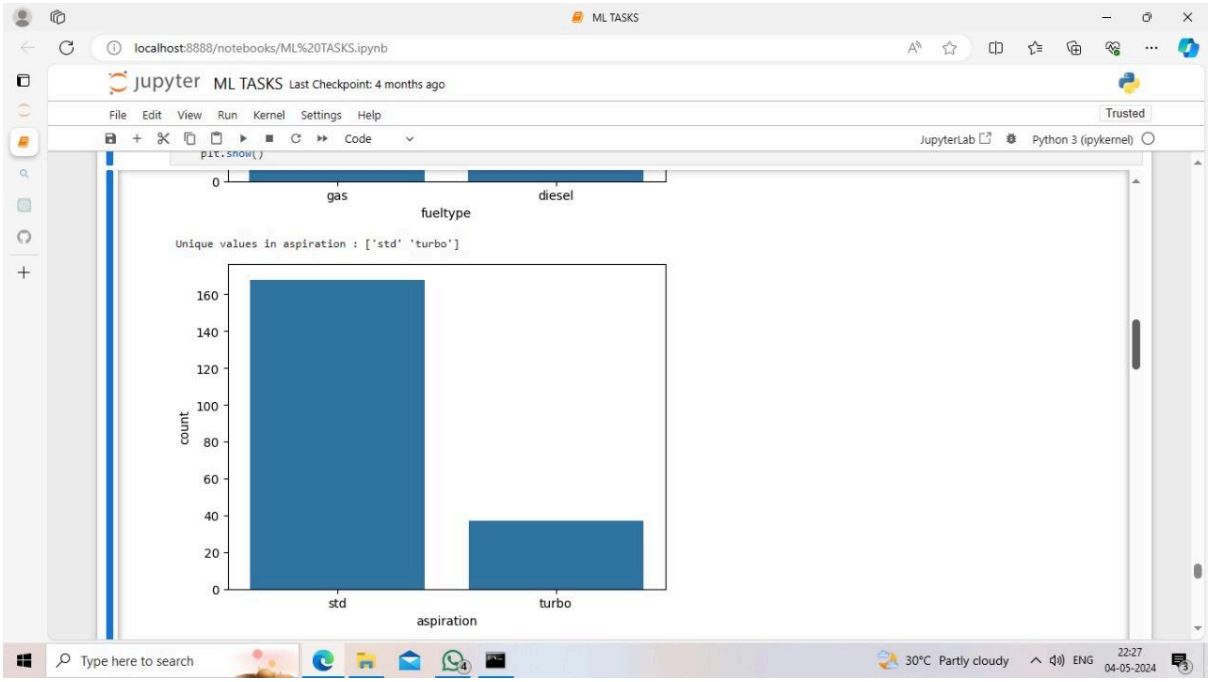

```
Unique values in fuelsystem : ['mpfi' '2bbl' 'mfi' '1bbl' 'spfi' '4bbl' 'idi' 'spdi']
```

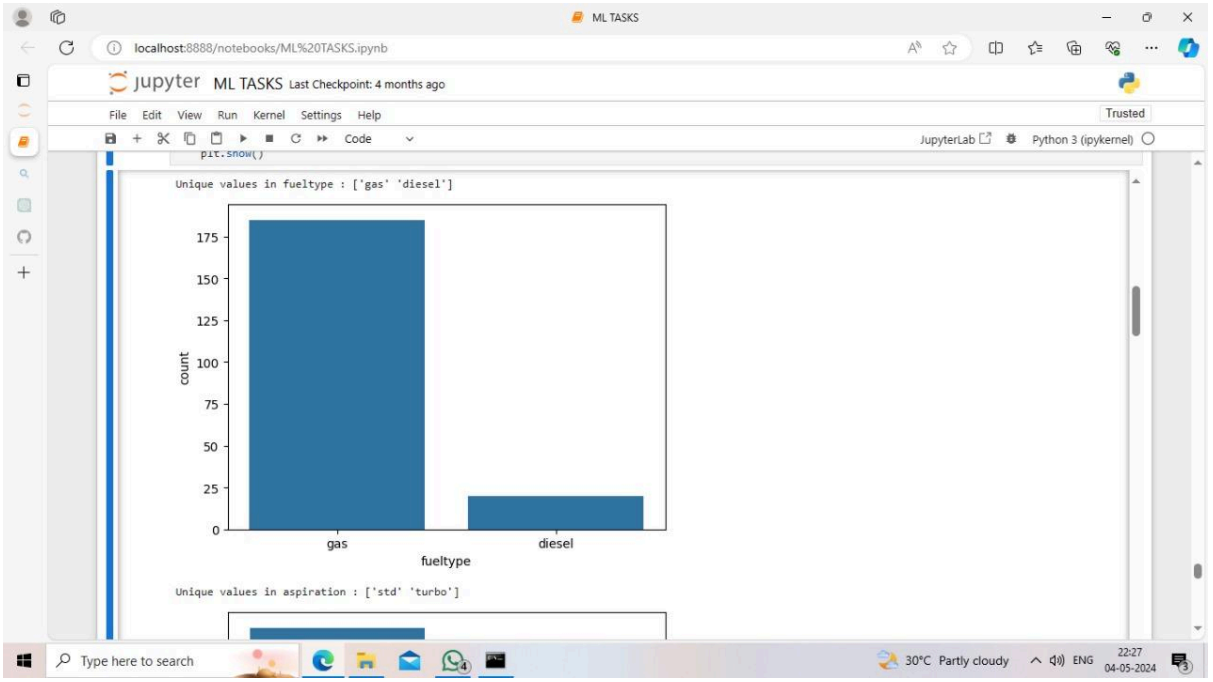
Type here to search

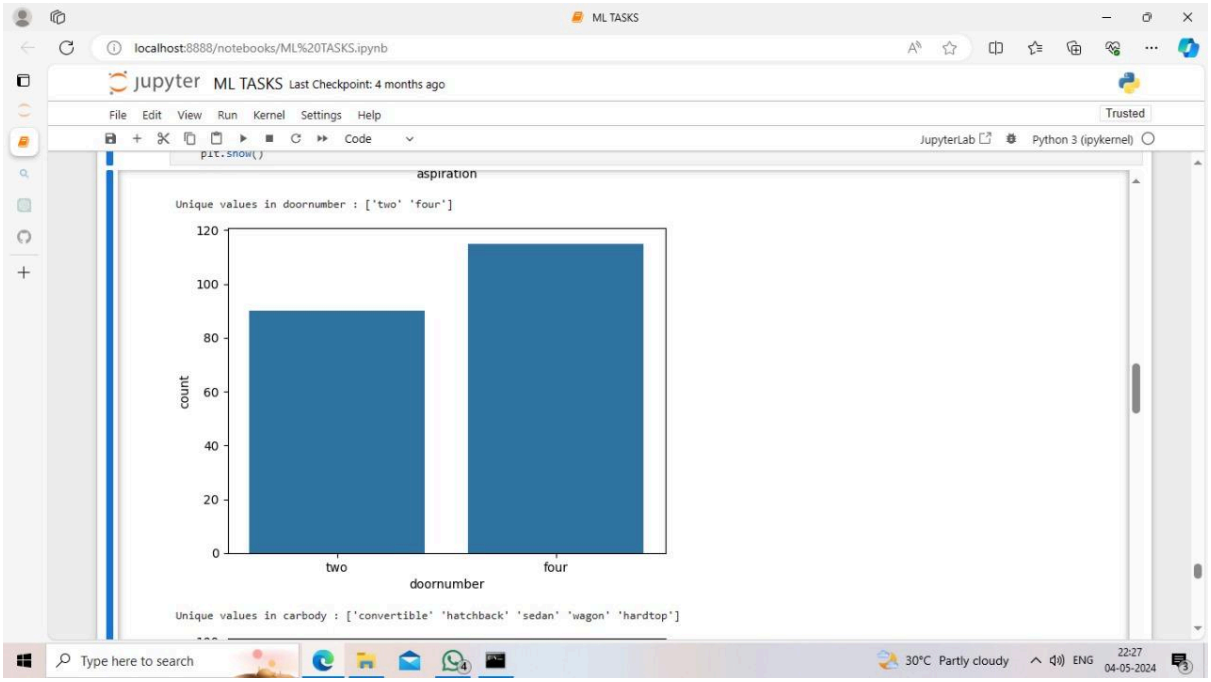
30°C Partly cloudy ENG 22:25 04-05-2024

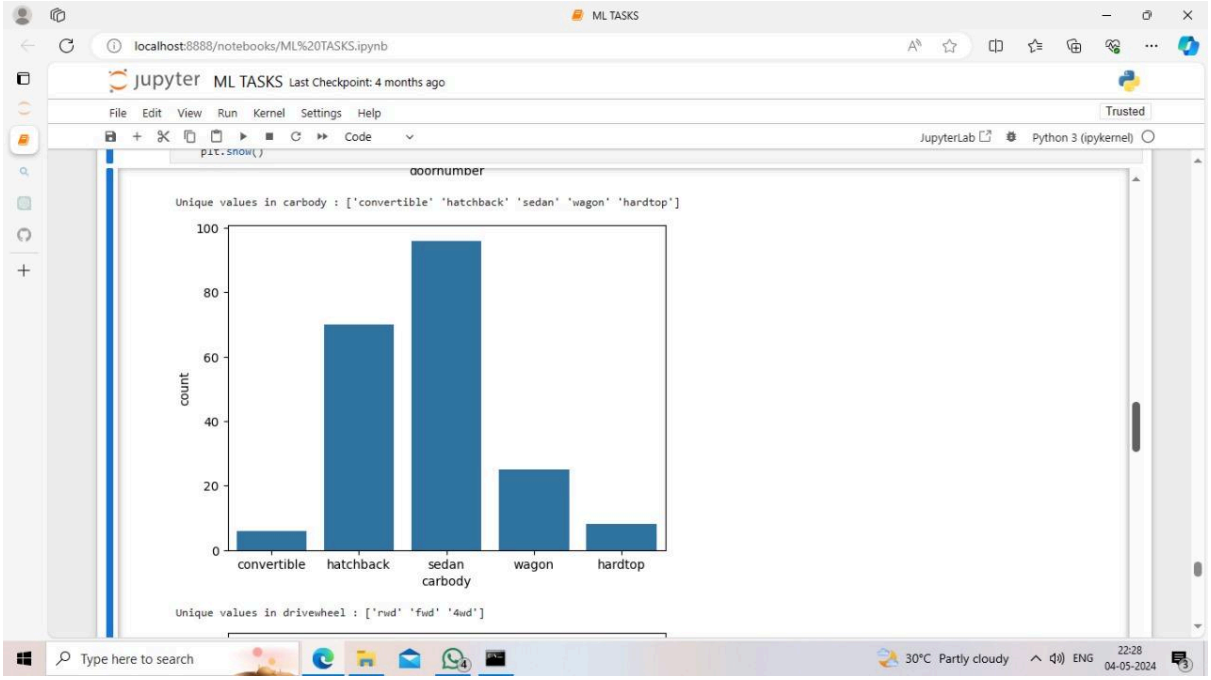

ML TASKS
localhost:8888/notebooks/ML%20TASKS.ipynb
jupyter ML TASKS Last Checkpoint: 4 months ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
plt.show()
Unique values in CarName : ['alfa-romero giulia' 'alfa-romero stelvio' 'alfa-romero Quadrifoglio'
'audi 100 ls' 'audi 100ls' 'audi fox' 'audi 5000' 'audi 4000'
'audi 5000s (diesel)' 'bmw 320i' 'bmw x1' 'bmw x3' 'bmw z4' 'bmw x4'
'bmw x5' 'chevrolet impala' 'chevrolet monte carlo' 'chevrolet vega 2300'
'dodge rampage' 'dodge challenger se' 'dodge d200' 'dodge monaco (sw)'
'dodge colt hardtop' 'dodge colt (sw)' 'dodge coronet custom'
'dodge dart custom' 'dodge coronet custom (sw)' 'honda civic'
'honda civic cvcc' 'honda accord cvcc' 'honda accord lx'
'honda civic 1500 gl' 'honda accord' 'honda civic 1300' 'honda prelude'
'honda civic (auto)' 'isuzu MU-X' 'isuzu D-Max ' 'isuzu D-Max V-Cross'
'jaguar xj' 'jaguar xf' 'jaguar xk' 'maxda rx3' 'maxda glc deluxe'
'mazda rx2 coupe' 'mazda rx-4' 'mazda glc deluxe' 'mazda 626' 'mazda glc'
'mazda rx-7 gs' 'mazda glc 4' 'mazda glc custom l' 'mazda glc custom'
'buick electra 225 custom' 'buick century luxus (sw)' 'buick century'
'buick skyhawk' 'buick opel isuzu deluxe' 'buick skylark'
'buick century special' 'buick regal sport coupe (turbo)'
'mercury cougar' 'mitsubishi mirage' 'mitsubishi lancer'
'mitsubishi outlander' 'mitsubishi g4' 'mitsubishi mirage g4'
'mitsubishi montero' 'mitsubishi pajero' 'Nissan versa' 'nissan gt-r'
'nissan rogue' 'nissan latio' 'nissan titan' 'nissan leaf' 'nissan juke'
'nissan note' 'nissan clipper' 'nissan nv200' 'nissan dayz' 'nissan fuga'
'nissan otti' 'nissan teana' 'nissan kicks' 'peugeot 504' 'peugeot 304'
'peugeot 504 (sw)' 'peugeot 604sl' 'peugeot 505s turbo diesel'
'plymouth fury iii' 'plymouth cricket' 'plymouth satellite custom (sw)'
'plymouth fury gran sedan' 'plymouth valiant' 'plymouth duster'
'porsche macan' 'porcshce panamera' 'porsche cayenne' 'porsche boxter'
'renault 12tl' 'renault 5 gtl' 'saab 99e' 'saab 99le' 'saab 99gle'
'subaru' 'subaru dl' 'subaru brz' 'subaru baja' 'subaru r1' 'subaru r2'
'subaru trezia' 'subaru tribeca' 'toyota corona mark ii' 'toyota corona'
'toyota corolla 1200' 'toyota corona hardtop' 'toyota corolla 1600 (sw)'
'toyota carina' 'toyota mark ii' 'toyota corolla'
Type here to search
30°C Partly cloudy
ENG
22:26
04-05-2024

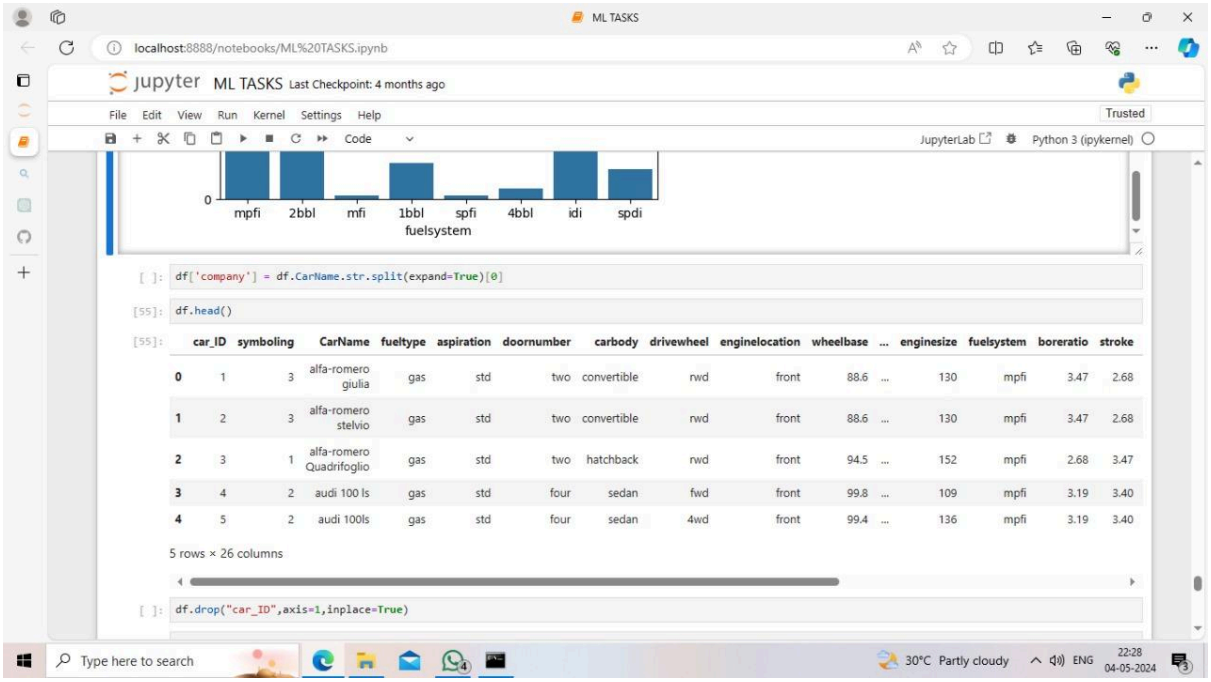

ML TASKS
localhost:8888/notebooks/ML%20TASKS.ipynb
jupyter ML TASKS Last Checkpoint: 4 months ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
plt.show()
'volkswagen type 3' 'volkswagen 411 (sw)' 'volkswagen super beetle'
'volkswagen dasher' 'vw dasher' 'vw rabbit' 'volkswagen rabbit'
'volkswagen rabbit custom' 'volvo 145e (sw)' 'volvo 144ea' 'volvo 244dl'
'volvo 245' 'volvo 264gl' 'volvo diesel' 'volvo 246']
count
CarName
Unique values in fueltype : ['gas' 'diesel']
Type here to search
30°C Partly cloudy
ENG
22:26
04-05-2024

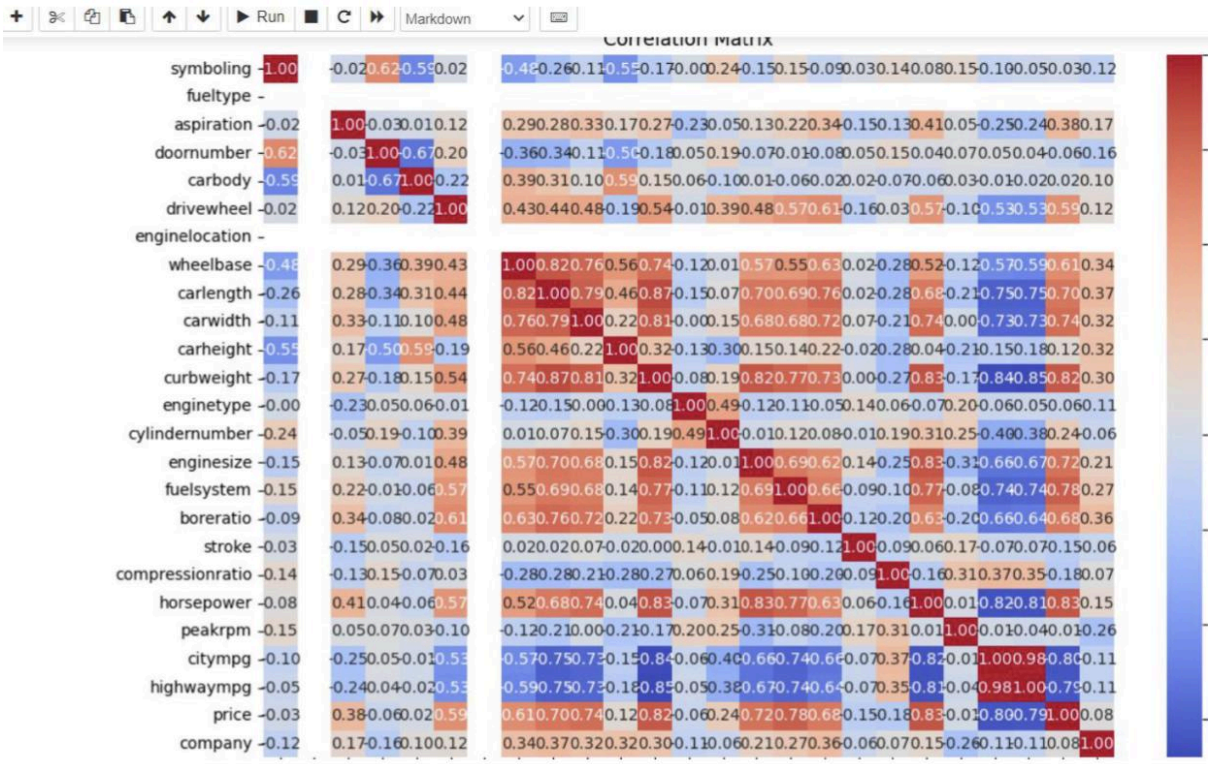

ML TASKS
localhost:8888/notebooks/ML%20TASKS.ipynb
jupyter ML TASKS Last Checkpoint: 4 months ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
plt.show()
gas
diesel
fueltype
Unique values in aspiration : ['std' 'turbo']
count
std
turbo
aspiration
Type here to search
30°C Partly cloudy
ENG
22:27
04-05-2024


ML TASKS
localhost:8888/notebooks/ML%20TASKS.ipynb
jupyter ML TASKS Last Checkpoint: 4 months ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
plt.show()
Unique values in fueltype : ['gas' 'diesel']
count
fueltype
gas
diesel
Unique values in aspiration : ['std' 'turbo']
Type here to search
30°C Partly cloudy
ENG
22:27
04-05-2024


ML TASKS
localhost:8888/notebooks/ML%20TASKS.ipynb
jupyter ML TASKS Last Checkpoint: 4 months ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
plt.show()
aspiration
Unique values in doornumber : ['two' 'four']
120
100
80
60
40
20
0
count
two
four
doornumber
Unique values in carbody : ['convertible' 'hatchback' 'sedan' 'wagon' 'hardtop']
Type here to search
30°C Partly cloudy
ENG
22:27
04-05-2024


ML TASKS
localhost:8888/notebooks/ML%20TASKS.ipynb
jupyter ML TASKS Last Checkpoint: 4 months ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
plt.show()
doornumber
Unique values in carbody : ['convertible' 'hatchback' 'sedan' 'wagon' 'hardtop']
100
80
60
40
20
0
count
convertible
hatchback
sedan
wagon
hardtop
carbody
Unique values in drivewheel : ['rwd' 'fwd' '4wd']
Type here to search
30°C Partly cloudy
ENG
22:28
04-05-2024


ML TASKS
localhost:8888/notebooks/ML%20TASKS.ipynb
jupyter ML TASKS Last Checkpoint: 4 months ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
0
mpfi
2bbl
mfi
1bbl
spfi
4bbl
idi
spdi
fuelsystem
[ ]: df['company'] = df.CarName.str.split(expand=True)[0]
[55]: df.head()
[55]:
car_ID symboling CarName fueltype aspiration doornumber carbody drivewheel enginelocation wheelbase ... enginesize fuelsystem boreratio stroke
0 1 3 alfa-romero giulia gas std two convertible rwd front 88.6 ... 130 mpfi 3.47 2.68
1 2 3 alfa-romero stelvio gas std two convertible rwd front 88.6 ... 130 mpfi 3.47 2.68
2 3 1 alfa-romero Quadrifoglio gas std two hatchback rwd front 94.5 ... 152 mpfi 2.68 3.47
3 4 2 audi 100 ls gas std four sedan fwd front 99.8 ... 109 mpfi 3.19 3.40
4 5 2 audi 100ls gas std four sedan 4wd front 99.4 ... 136 mpfi 3.19 3.40
5 rows × 26 columns
[ ]: df.drop("car_ID",axis=1,inplace=True)
Type here to search
30°C Partly cloudy
ENG
22:28
04-05-2024



Features with relatively high positive correlations with the target variable (price) include: wheelbase, carlength, carwidth, curbweight, enginesi and company. Some features exhibit multicollinearity, such as: carlength, carwidth, and curbweight; enginesize and horsepower.

```
]: # Selected features
selected_features = ['wheelbase', 'curbweight', 'enginesize', 'horsepower', 'company', 'price']
```

```
selected_features = ['wheelbase', 'curbweight', 'enginesize', 'horsepower', 'company', 'price']

# Subset the DataFrame with selected features
selected_df = car_price[selected_features]

# Split the data into features (X) and target variable (y)
X = selected_df.drop('price', axis=1)
y = selected_df['price']

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# List of regression algorithms to evaluate
regressors = {
    'Linear Regression': LinearRegression(),
    'Decision Tree Regressor': DecisionTreeRegressor(),
    'Random Forest Regressor': RandomForestRegressor(),
    'Gradient Boosting Regressor': GradientBoostingRegressor(),
    'Support Vector Regressor': SVR()
}

# Train and evaluate each regression algorithm
for name, regressor in regressors.items():
    # Train the model
    regressor.fit(X_train, y_train)

    # Evaluate the model
    train_score = regressor.score(X_train, y_train)
    test_score = regressor.score(X_test, y_test)
```

ML - Task5
- Classification and clustering

Problem Description
Use sklearn.datasets iris flower dataset to train your model using logistic regression. You need to figure out the accuracy of your model and use that to predict different samples in your test dataset. In iris dataset there are 150 samples containing following features,

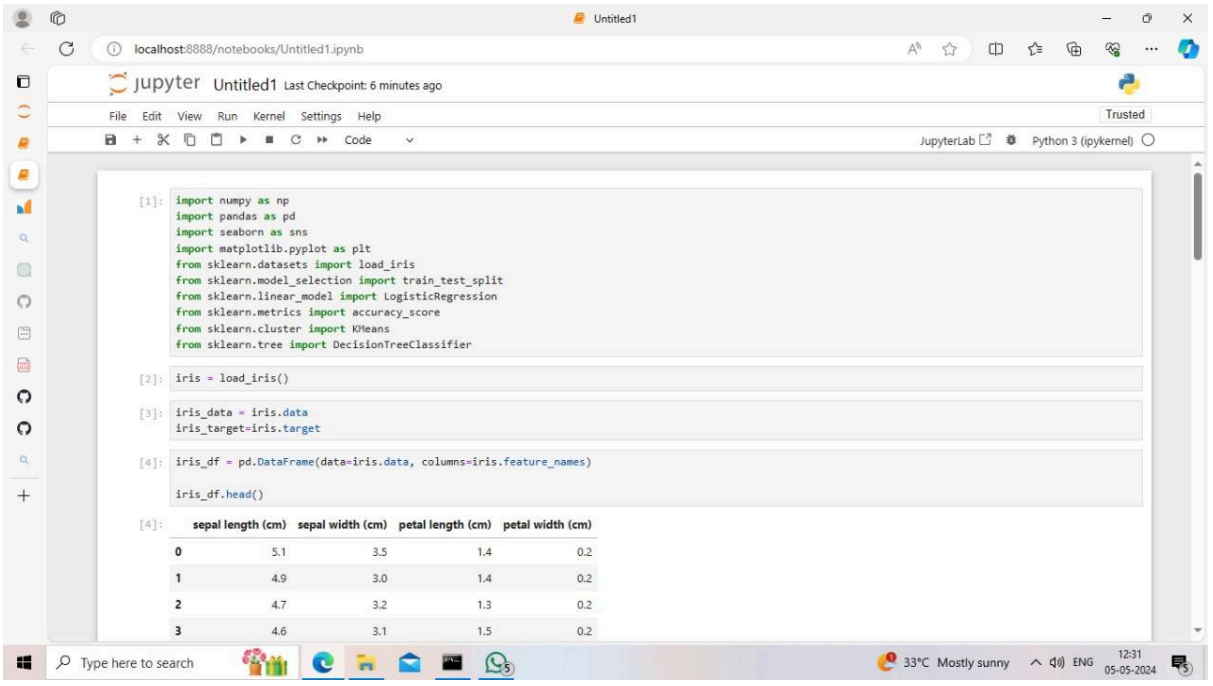
1. Sepal Length
2. Sepal Width
3. Petal length
4. Petal width

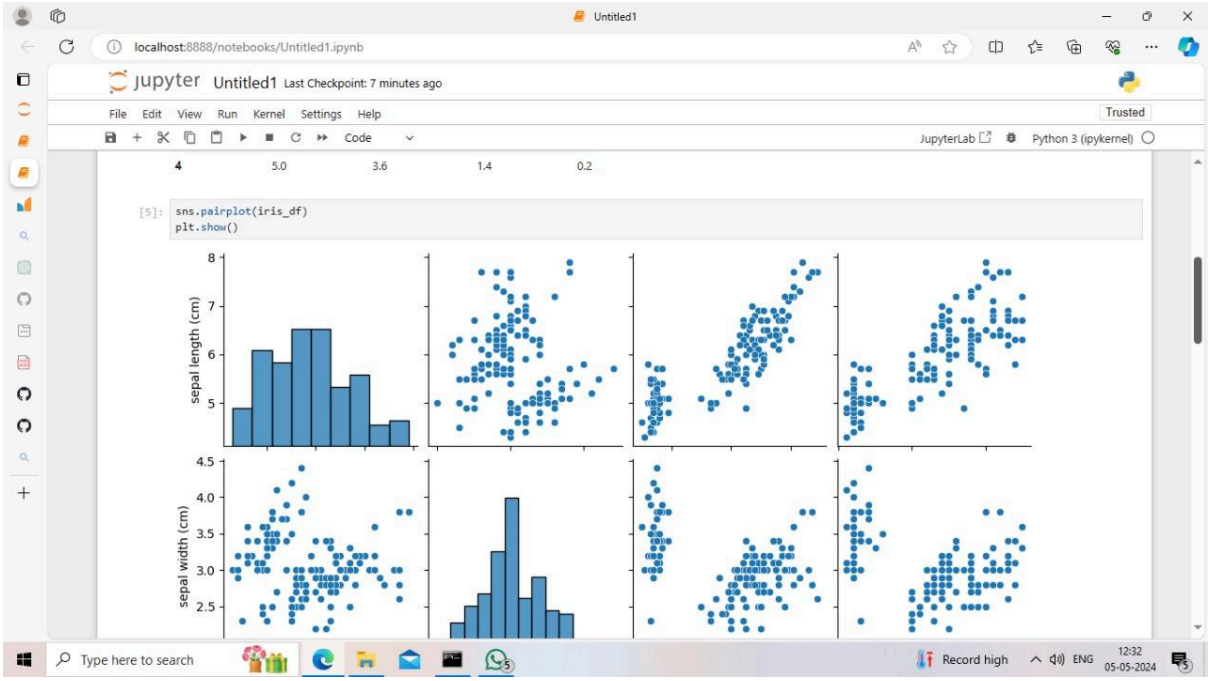
Using above 4 features you will classify a flower in one of the three categories,

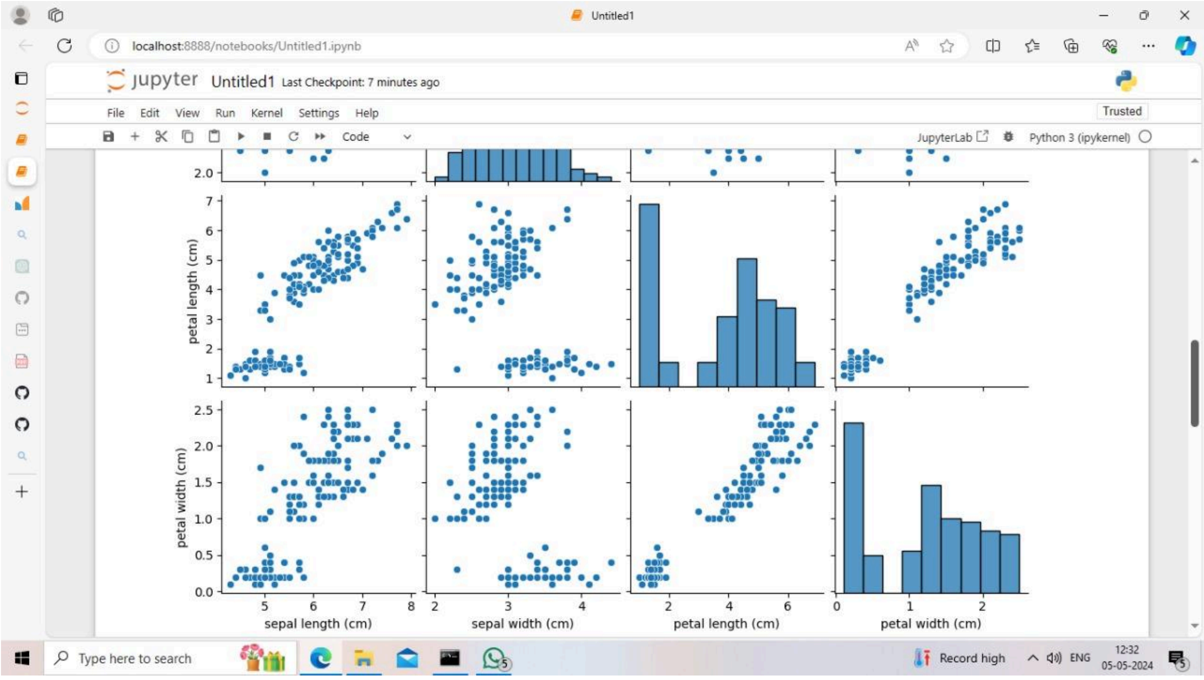
1. Setosa
2. Versicolour
3. Virginica

Download the iris dataset directly from sklearn.dataset

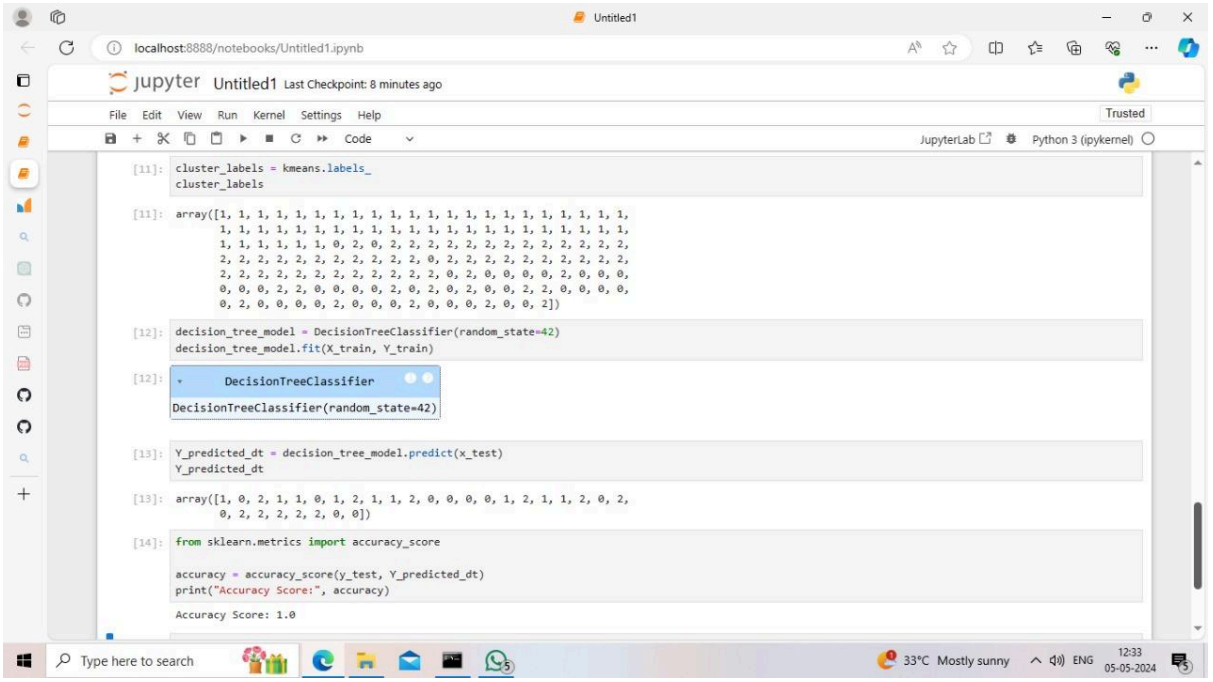
Use the same data for different clustering(exclude target variable) and classification algorithm








```
[6]: X_train, x_test, Y_train, y_test = train_test_split(iris_data,iris_target , test_size=0.2, random_state=42)

[7]: log_reg_model = LogisticRegression(max_iter=1000)
     log_reg_model.fit(X_train, Y_train)

[7]: LogisticRegression
     LogisticRegression(max_iter=1000)

[8]: target_predicted = log_reg_model.predict(x_test)

[9]: accuracy_score = accuracy_score(y_test, target_predicted)
     accuracy_score

[9]: 1.0

[10]: kmeans = KMeans(n_clusters=3, random_state=42)
      kmeans.fit(iris_data)

[10]: KMeans
      KMeans(n_clusters=3, random_state=42)

[11]: cluster_labels = kmeans.labels_
      cluster_labels

[11]: array([1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
             1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
```


Untitled1
localhost:8888/notebooks/Untitled1.ipynb
jupyter Untitled1 Last Checkpoint: 8 minutes ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
[11]: cluster_labels = kmeans.labels_
cluster_labels
[11]: array([1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 0, 2, 0, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 0, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 0, 2, 0, 0, 0, 0, 2, 0, 0,
0, 0, 0, 2, 2, 0, 0, 0, 0, 2, 0, 2, 0, 2, 0, 0, 2, 2, 0, 0, 0, 0,
0, 2, 0, 0, 0, 0, 2, 0, 0, 0, 2, 0, 0, 0, 2, 0, 0, 2])
[12]: decision_tree_model = DecisionTreeClassifier(random_state=42)
decision_tree_model.fit(X_train, Y_train)
[12]: DecisionTreeClassifier
DecisionTreeClassifier(random_state=42)
[13]: Y_predicted_dt = decision_tree_model.predict(x_test)
Y_predicted_dt
[13]: array([1, 0, 2, 1, 1, 0, 1, 2, 1, 1, 2, 0, 0, 0, 0, 1, 2, 1, 1, 2, 0, 2,
0, 2, 2, 2, 2, 0, 0])
[14]: from sklearn.metrics import accuracy_score
accuracy = accuracy_score(y_test, Y_predicted_dt)
print("Accuracy Score:", accuracy)
Accuracy Score: 1.0
Type here to search
33°C Mostly sunny
ENG
12:33
05-05-2024