Project Preparation Phase
Data Collection

Date	25 November 2022
Team ID	PNT2022TMID30140
Project Name	University Admit Eligibility Predictor
Maximum Marks	

Importing the Libraries:

Import Libraries

In [165…

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as mtp
import sklearn
from scipy.stats import iqr
```

Reading the Dataset:

In [54]:

```
import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove those credentials before you share the notebook.
cos_client = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='1mfUrBZXNXedJeOJMz4P9zJkfDzxFuKdNzVPR-6_n6cM',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')

bucket = 'universityadmiteligibilitypredict-donotdelete-pr-5vceqmhls7eowb'
object_key = 'university.csv'

body = cos_client.get_object(Bucket=bucket,Key=object_key)['Body']
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

df = pd.read_csv(body)
df.head()
```

Out[54]:

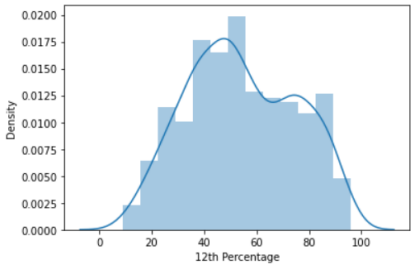
	S.NO	University Name	District	12th Percentage	Entrance Percentage	Department	Output
0	1	Aligarh Muslim University	Aligarh	55	80	Computer Science and Engineering	Yes
1	2	Aligarh Muslim University	Aligarh	85	80	Computer Science and Engineering	Yes
2	3	Aligarh Muslim University	Aligarh	74	80	Computer Science and Engineering	Yes
3	4	Aligarh Muslim University	Aligarh	92	80	Computer Science and Engineering	Yes
4	5	Aligarh Muslim University	Aligarh	19	80	Computer Science and Engineering	No

Analyse the Data:

Univariate Analysis

In [60]:
```
sns.distplot(df1['12th Percentage'])
```

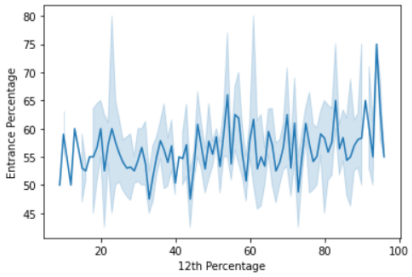
Out[60]:



Bivariate Analysis

In [61]:
```
sns.lineplot(df1['12th Percentage'],df1['Entrance Percentage'])
```

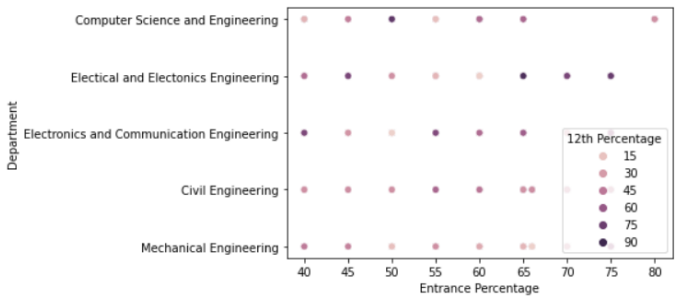
Out[61]:



Multi variate Analysis

In [62]:
```
sns.scatterplot(df1['Entrance Percentage'],df1['Department'],hue = df1['12th Percentage'])
```

Out[62]:



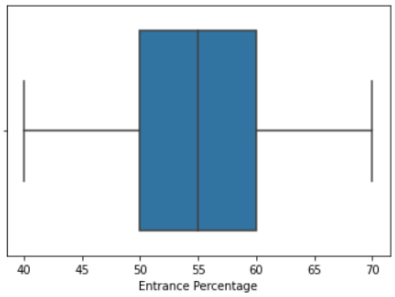
Handling Missing Values:

handling outliers

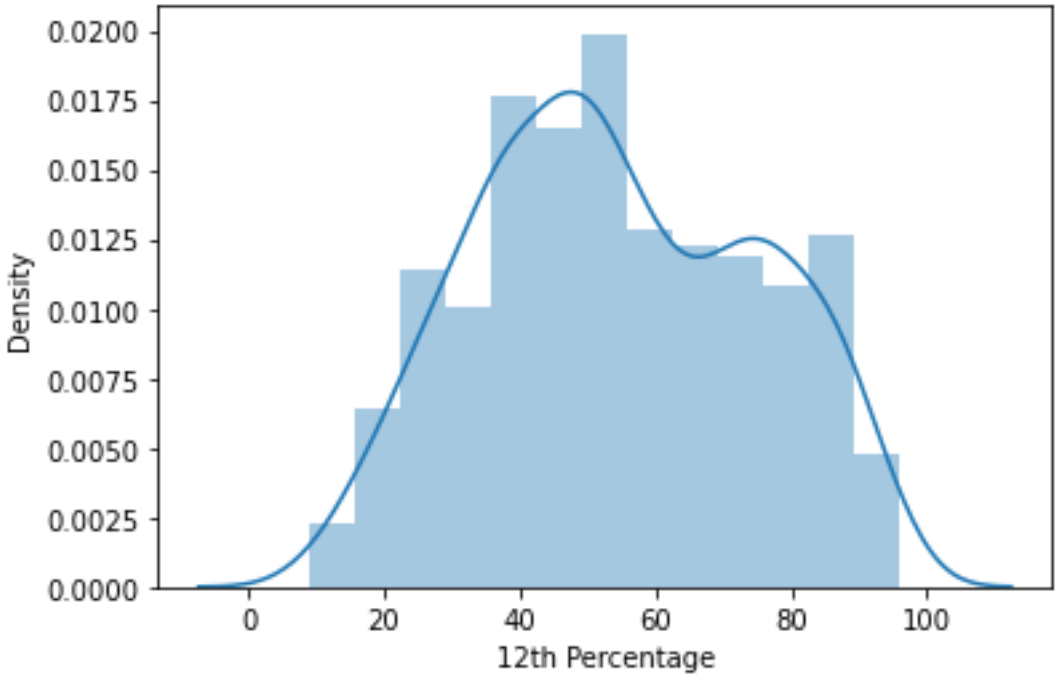
```
In [76]: df1['Entrance Percentage'] = np.where(df1['Entrance Percentage'] >70,40,df1['Entrance Percentage'])

In [77]: sns.boxplot(df1['Entrance Percentage'])
```

Out[77]:



Data Visualisation:



Splitting Independent and Dependant Columns:

In [82]:
```
x = df.iloc[:,1:5]
```

In [83]:
```
x.head()
```

Out[83]:

	University Name	12th Percentage	Entrance Percentage	Department
0	0	55	80	1
1	0	85	80	1
2	0	74	80	1
3	0	92	80	1
4	0	19	80	1

In [84]:
```
y = df.iloc[:,5:6]
y.head()
```

Out[84]:

	Output
0	1
1	1
2	1
3	1
4	0

Splitting the Data into Test and Train:

Splitting dataset into train and test

In [86]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size = 0.3,random_state =0)
```

In [87]:
```
x_train
```

Out[87]:

	University Name	12th Percentage	Entrance Percentage	Department
738	36	55	55	4
351	17	63	50	1
385	18	45	50	0
776	37	78	65	2
250	12	60	75	4
...	...	...	...	...
763	36	44	55	4
192	9	59	50	0
629	32	93	55	4
559	29	29	50	4
684	30	26	60	3

552 rows × 4 columns

In [88]:
```
y_test
```

Out[88]:

	Output
453	1
85	0
545	1
312	0
334	1
...	...
402	1
92	1
261	0
493	0
775	1

237 rows × 1 columns