Experiment-3.1

Student Name: YANA SRIVASTAVA
UID: 20BCS2279
Branch: CSE
Section/Group: 20BCS_WM-906/B
Semester: 5th
Subject Code: 21CST-317
Subject Name: Machine learning lab

1. Aim/Overview of the practical: Implement Kmeans.

2. Requirements: Any Dataset for classification, Jupyter notebook or Google collab**.**

3. Description/ Code:

 # importing all necessary libraries

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.cluster import KMeans
from sklearn.model_selection import train_test_split
```

 # Reading the data

```
df=pd.read_csv("3D_spatial_network.txt",names=['OSM_ID','LONGITUDE','LATITUDE','ALTITUDE'])
df.head()
```

	OSM_ID	LONGITUDE	LATITUDE	ALTITUDE
0	144552912	9.349849	56.740876	17.052772
1	144552912	9.350188	56.740679	17.614840
2	144552912	9.350549	56.740544	18.083536
3	144552912	9.350806	56.740485	18.279465
4	144552912	9.351053	56.740486	18.422974

 # Preprocessing the data and Splitting the dataset into train and test

```
data  = df.drop('OSM_ID',axis=1)

train_data,test_data = train_test_split(data,test_size=0.3)

std_scaler = MinMaxScaler()
train_data['LONGITUDE']=train_data['LONGITUDE']/train_data['LONGITUDE'].max()
train_data['LATITUDE']=train_data['LATITUDE']/train_data['LATITUDE'].max()
train_data['ALTITUDE']=train_data['ALTITUDE']/train_data['ALTITUDE'].max()

test_data['LONGITUDE']=test_data['LONGITUDE']/test_data['LONGITUDE'].max()
test_data['LATITUDE']=test_data['LATITUDE']/test_data['LATITUDE'].max()
test_data['ALTITUDE']=test_data['ALTITUDE']/test_data['ALTITUDE'].max()
```

```
train_data.head()
```

	LONGITUDE	LATITUDE	ALTITUDE
212414	0.871023	0.989723	0.063734
135673	0.929270	0.997163	0.032702
291118	0.922388	0.991339	0.122259
307128	0.899849	0.984240	0.341964
358740	0.760997	0.980894	0.146135

```
test_data.head()
```

	LONGITUDE	LATITUDE	ALTITUDE
308941	0.883420	0.982314	0.416726
380476	0.914637	0.997535	0.050558
212702	0.885900	0.988021	0.022134
237919	0.878948	0.985133	0.120827
100458	0.877304	0.990803	0.052138

Training Kmeans model for clustering

```
clustering_Kmeans = KMeans(n_clusters=6)
clustering_Kmeans.fit(train_data)
```

Predicting the test data for SVM

```
pred=clustering_Kmeans.predict(test_data)
print(pred)
```

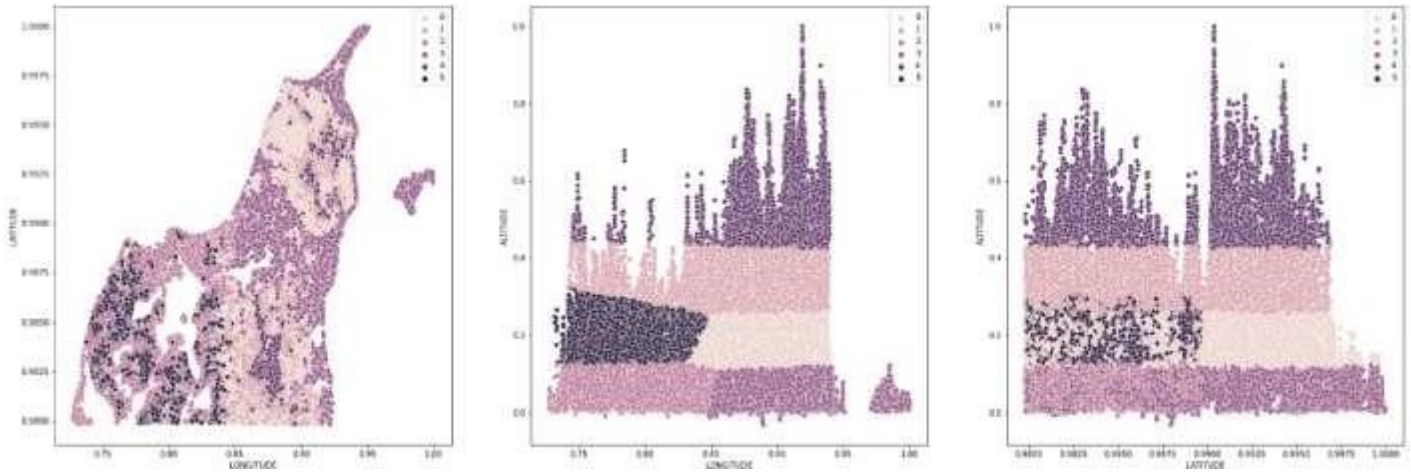
```
array([1, 3, 3, ..., 2, 1, 0])
```

Clustering

```
plt.figure(figsize=(30,5))
plt.subplot(1,3,1)
sns.scatterplot(test_data['LONGITUDE'],test_data['LATITUDE'],hue=pred)

plt.subplot(1,3,2)
sns.scatterplot(test_data['LONGITUDE'],test_data['ALTITUDE'],hue=pred)

plt.subplot(1,3,3)
sns.scatterplot(test_data['LATITUDE'],test_data['ALTITUDE'],hue=pred)
```



Learning outcomes (What I have learnt):

- **Kmeans clustering**

Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.			
2.			
3.			