Clustering+Categorical+Data+-+Exercise

January 27, 2021

1 Clustering Categorical Data

You are given much more country data. Using the same methodology as the one in the lecture, group all the countries in 2 clusters.

Already done that? Okay!

There are other features: name and continent.

Encode the continent one and use it in the clustering solution. Think about the difference with the previous exercise.

1.1 Import the relevant libraries

[1]:
```
import pandas as pd
import numpy as np
import seaborn as sb
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
```

1.2 Load the data

Load data from the csv file: 'Categorical.csv'.

[2]:
```
data = pd.read_csv('Categorical.csv')
data.head()
```

[2]:
```
          name  Longitude   Latitude      continent
0        Aruba -69.982677  12.520880  North America
1  Afghanistan  66.004734  33.835231           Asia
2       Angola  17.537368 -12.293361         Africa
3     Anguilla -63.064989  18.223959  North America
4      Albania  20.049834  41.142450         Europe
```

Remove the duplicate index column from the dataset.

[3]:
```
data['continent'].unique()
```

[3]:
```
array(['North America', 'Asia', 'Africa', 'Europe', 'South America',
       'Oceania', 'Antarctica', 'Seven seas (open ocean)'], dtype=object)
```

1.3 Map the data

Use the ‘continent’ category for this analysis.

[4]:
```
data_mapped = data.copy()
data_mapped['continent'] = data_mapped['continent'].map({'North America':
 ↪0,'Asia':1,'Africa':2,'Europe':3,'South America':4,'Oceania':5,'Antarctica':
 ↪6,'Seven seas (open ocean)':7})
data_mapped
```

[4]:
```
             name   Longitude   Latitude  continent
0           Aruba  -69.982677  12.520880          0
1     Afghanistan   66.004734  33.835231          1
2          Angola   17.537368 -12.293361          2
3        Anguilla  -63.064989  18.223959          0
4         Albania   20.049834  41.142450          3
..            …          …          …        …
236         Samoa -172.164851 -13.753243          5
237         Yemen   47.586762  15.909280          1
238  South Africa   25.083901 -29.000341          2
239        Zambia   27.774759 -13.458242          2
240      Zimbabwe   29.851441 -19.004204          2

[241 rows x 4 columns]
```

1.4 Select the features

[5]:
```
x = data_mapped.iloc[:,3:4]
x.head()
```

[5]:
```
   continent
0          0
1          1
2          2
3          0
4          3
```

1.5 Clustering

Use 4 clusters initially.

[12]:
```
kmeans = KMeans(7)
```

[13]:
```
kmeans.fit(x)
```

[13]:
```
KMeans(n_clusters=7)
```

1.6 Clustering results

[14]:

```
identified_clusters = kmeans.fit_predict(x)
identified_clusters
```

[14]:

```
array([4, 0, 3, 4, 5, 5, 5, 0, 1, 0, 6, 2, 6, 2, 4, 6, 5, 0, 3, 5, 3, 3,
       0, 5, 0, 4, 4, 5, 4, 5, 4, 4, 1, 1, 4, 0, 0, 3, 3, 5, 1, 0, 3, 3,
       3, 3, 6, 1, 3, 3, 4, 4, 4, 4, 0, 0, 5, 5, 3, 4, 5, 4, 3, 1, 3, 3,
       5, 5, 3, 5, 6, 1, 5, 5, 6, 3, 5, 0, 5, 3, 3, 3, 3, 3, 5, 4, 4, 4,
       6, 1, 0, 2, 4, 5, 4, 5, 0, 5, 0, 0, 2, 5, 0, 0, 5, 0, 5, 4, 5, 0,
       0, 0, 0, 3, 0, 0, 6, 4, 0, 5, 0, 0, 0, 3, 3, 4, 5, 0, 3, 5, 5, 5,
       0, 4, 3, 5, 5, 3, 2, 4, 6, 5, 3, 5, 0, 5, 0, 6, 3, 3, 4, 2, 3, 0,
       3, 6, 3, 6, 3, 4, 6, 5, 5, 0, 6, 6, 0, 0, 4, 6, 1, 0, 6, 6, 5, 4,
       0, 5, 1, 0, 6, 0, 5, 5, 3, 3, 0, 3, 3, 3, 0, 2, 2, 6, 3, 4, 5, 3,
       3, 4, 5, 3, 1, 5, 5, 5, 3, 4, 2, 0, 4, 3, 3, 0, 0, 0, 0, 6, 4, 3,
       0, 0, 3, 3, 5, 1, 4, 0, 5, 4, 1, 4, 4, 0, 6, 6, 6, 0, 3, 3, 3])
```

[15]:

```
data_with_clusters = data_mapped.copy()
data_with_clusters['Cluster'] = identified_clusters
data_with_clusters.head()
```

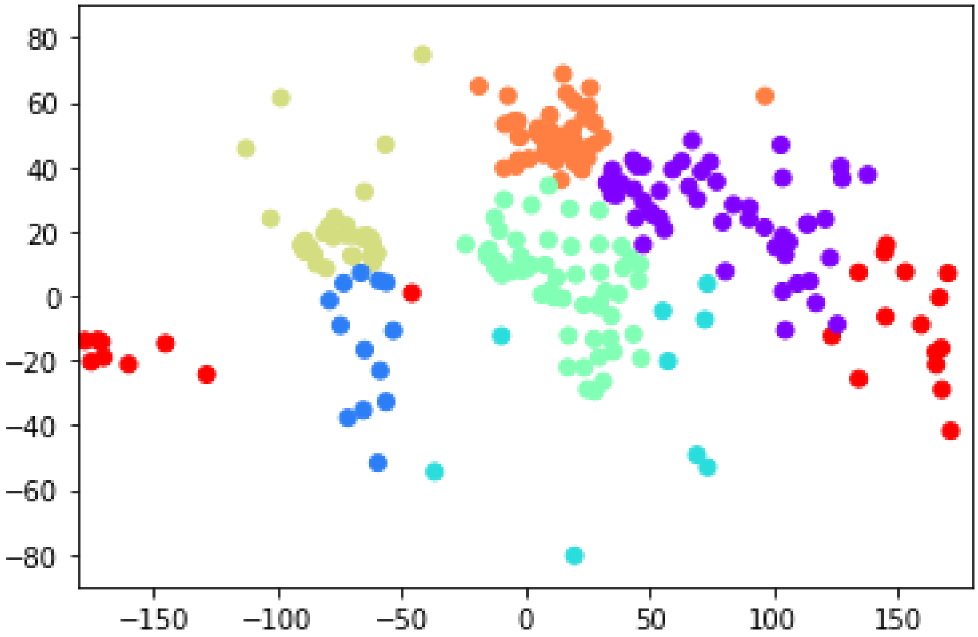
[15]:

```
          name  Longitude   Latitude  continent  Cluster
0        Aruba -69.982677  12.520880          0        4
1  Afghanistan  66.004734  33.835231          1        0
2       Angola  17.537368 -12.293361          2        3
3     Anguilla -63.064989  18.223959          0        4
4      Albania  20.049834  41.142450          3        5
```

1.7 Plot the data

[16]:

```
plt.scatter(data_with_clusters['Longitude'], data_with_clusters['Latitude'],␣
 ↪c=data_with_clusters['Cluster'],cmap='rainbow')
plt.xlim(-180,180)
plt.ylim(-90,90)
plt.show()
```



Since you already have all the code necessary, go back and play around with the number of clusters. Try 3, 7 and 8 and see if the results match your expectations.

Simply go back to the beggining of the Clustering section and change kmeans = KMeans(4) to kmeans = KMeans(3) . Then run the remaining cells until the end.