Assignment-07-Clustering-Hierarchical (Airlines)

Perform clustering (hierarchical,K means clustering and DBSCAN) for the airlines data to obtain optimum number of clusters. Draw the inferences from the clusters obtained.

Using Normalize Function

In [40]:
```
# Import libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import scipy.cluster.hierarchy as sch
from sklearn.cluster import AgglomerativeClustering
from sklearn.preprocessing import normalize
```

Import Dataset

In [41]:
```
airline=pd.read_excel("EastWestAirlines.xlsx",sheet_name="data")
airline
```

Out[41]:

	ID#	Balance	Qual_miles	cc1_miles	cc2_miles	cc3_miles	Bonus_miles	Bonus_trans	Flight_miles_12mo	Flight_trans_12	Days_since_enroll	Award?
0	1	28143	0	1	1	1	174	1	0	0	7000	0
1	2	19244	0	1	1	1	215	2	0	0	6968	0
2	3	41354	0	1	1	1	4123	4	0	0	7034	0
3	4	14776	0	1	1	1	500	1	0	0	6952	0
4	5	97752	0	4	1	1	43300	26	2077	4	6935	1
...	...	...	...	...	...	...	...	...	...	...	...	...
3994	4017	18476	0	1	1	1	8525	4	200	1	1403	1
3995	4018	64385	0	1	1	1	981	5	0	0	1395	1
3996	4019	73597	0	3	1	1	25447	8	0	0	1402	1
3997	4020	54899	0	1	1	1	500	1	500	1	1401	0
3998	4021	3016	0	1	1	1	0	0	0	0	1398	0

3999 rows × 12 columns

In [42]:
```
airline.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3999 entries, 0 to 3998
Data columns (total 12 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   ID#                3999 non-null   int64
 1   Balance            3999 non-null   int64
 2   Qual_miles         3999 non-null   int64
 3   cc1_miles          3999 non-null   int64
 4   cc2_miles          3999 non-null   int64
 5   cc3_miles          3999 non-null   int64
 6   Bonus_miles        3999 non-null   int64
 7   Bonus_trans        3999 non-null   int64
 8   Flight_miles_12mo  3999 non-null   int64
 9   Flight_trans_12    3999 non-null   int64
 10  Days_since_enroll  3999 non-null   int64
 11  Award?             3999 non-null   int64
dtypes: int64(12)
memory usage: 375.0 KB
```

In [43]:
```
airline2=airline.drop(['ID#'],axis=1)
airline2
```

Out[43]:

	Balance	Qual_miles	cc1_miles	cc2_miles	cc3_miles	Bonus_miles	Bonus_trans	Flight_miles_12mo	Flight_trans_12	Days_since_enroll	Award?
0	28143	0	1	1	1	174	1	0	0	7000	0
1	19244	0	1	1	1	215	2	0	0	6968	0
2	41354	0	1	1	1	4123	4	0	0	7034	0
3	14776	0	1	1	1	500	1	0	0	6952	0
4	97752	0	4	1	1	43300	26	2077	4	6935	1
...	...	...	...	...	...	...	...	...	...	...	...
3994	18476	0	1	1	1	8525	4	200	1	1403	1
3995	64385	0	1	1	1	981	5	0	0	1395	1
3996	73597	0	3	1	1	25447	8	0	0	1402	1
3997	54899	0	1	1	1	500	1	500	1	1401	0
3998	3016	0	1	1	1	0	0	0	0	1398	0

3999 rows × 11 columns

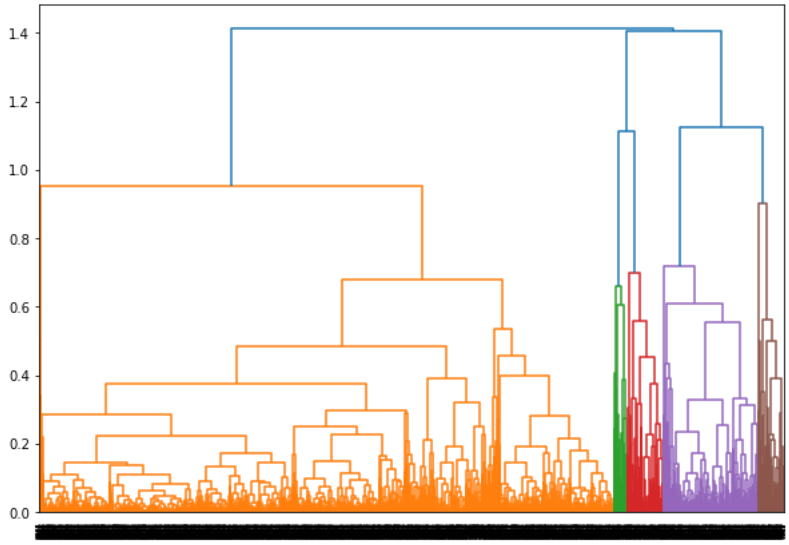
In [44]:
```
# Normalize heterogenous numerical data
airline2_norm=pd.DataFrame(normalize(airline2),columns=airline2.columns)
airline2_norm
```

Out[44]:

	Balance	Qual_miles	cc1_miles	cc2_miles	cc3_miles	Bonus_miles	Bonus_trans	Flight_miles_12mo	Flight_trans_12	Days_since_enroll	Award?
0	0.970414	0.0	0.000034	0.000034	0.000034	0.006000	0.000034	0.000000	0.000000	0.241371	0.000000
1	0.940209	0.0	0.000049	0.000049	0.000049	0.010504	0.000098	0.000000	0.000000	0.340437	0.000000
2	0.981113	0.0	0.000024	0.000024	0.000024	0.097817	0.000095	0.000000	0.000000	0.166880	0.000000
3	0.904428	0.0	0.000061	0.000061	0.000061	0.030605	0.000061	0.000000	0.000000	0.425527	0.000000
4	0.912226	0.0	0.000037	0.000009	0.000009	0.404078	0.000243	0.019383	0.000037	0.064718	0.000009
...	...	...	...	...	...	...	...	...	...	...	...
3994	0.905810	0.0	0.000049	0.000049	0.000049	0.417949	0.000196	0.009805	0.000049	0.068784	0.000049
3995	0.999649	0.0	0.000016	0.000016	0.000016	0.015231	0.000078	0.000000	0.000000	0.021659	0.000016
3996	0.944948	0.0	0.000039	0.000013	0.000013	0.326726	0.000103	0.000000	0.000000	0.018001	0.000013
3997	0.999592	0.0	0.000018	0.000018	0.000018	0.009104	0.000018	0.009104	0.000018	0.025509	0.000000
3998	0.907271	0.0	0.000301	0.000301	0.000301	0.000000	0.000000	0.000000	0.000000	0.420546	0.000000

3999 rows × 11 columns

In [45]:
```
# Create Dendrograms
plt.figure(figsize=(10, 7))
dendograms=sch.dendrogram(sch.linkage(airline2_norm,'complete'))
```



In [46]:
```
# Create Clusters (y)
hclusters=AgglomerativeClustering(n_clusters=5,affinity='euclidean',linkage='ward')
hclusters
```

Out[46]:
```
AgglomerativeClustering(n_clusters=5)
```
In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

In [47]:
```
y=pd.DataFrame(hclusters.fit_predict(airline2_norm),columns=['clustersid'])
y['clustersid'].value_counts()
```

Out[47]:
```
2    1547
4    1191
3     579
1     453
0     229
Name: clustersid, dtype: int64
```

In [48]:
```
# Adding clusters to dataset
airline2['clustersid']=hclusters.labels_
airline2
```

Out[48]:

	Balance	Qual_miles	cc1_miles	cc2_miles	cc3_miles	Bonus_miles	Bonus_trans	Flight_miles_12mo	Flight_trans_12	Days_since_enroll	Award?	cluster
0	28143	0	1	1	1	174	1	0	0	7000	0	
1	19244	0	1	1	1	215	2	0	0	6968	0	
2	41354	0	1	1	1	4123	4	0	0	7034	0	
3	14776	0	1	1	1	500	1	0	0	6952	0	
4	97752	0	4	1	1	43300	26	2077	4	6935	1	
...	...	...	...	...	...	...	...	...	...	...	...	
3994	18476	0	1	1	1	8525	4	200	1	1403	1	
3995	64385	0	1	1	1	981	5	0	0	1395	1	
3996	73597	0	3	1	1	25447	8	0	0	1402	1	
3997	54899	0	1	1	1	500	1	500	1	1401	0	
3998	3016	0	1	1	1	0	0	0	0	1398	0	

3999 rows × 12 columns

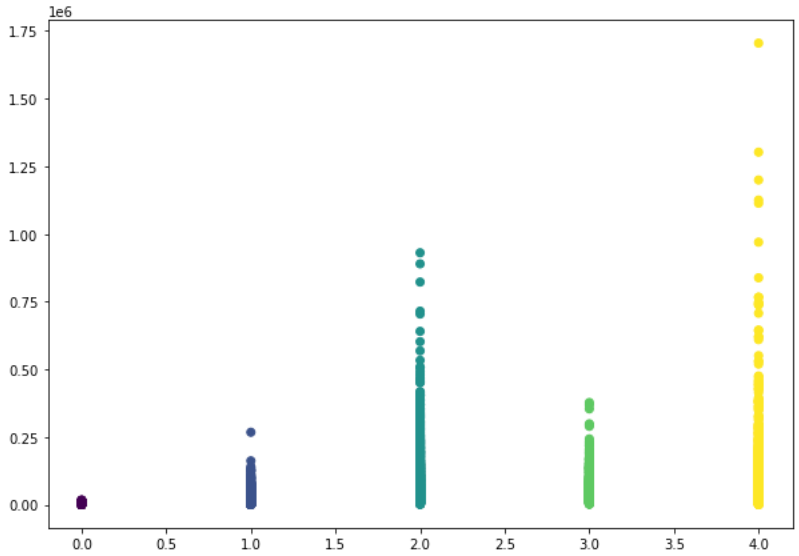
In [49]:
```
airline2.groupby('clustersid').agg(['mean']).reset_index()
```

Out[49]:

	clustersid	Balance	Qual_miles	cc1_miles	cc2_miles	cc3_miles	Bonus_miles	Bonus_trans	Flight_miles_12mo	Flight_trans_12	Days_since_enroll	A
		mean	mean	mean	mean	mean	mean	mean	mean	mean	mean	
0	0	5524.222707	8.755459	1.000000	1.000000	1.000000	584.532751	2.401747	66.982533	0.209607	4875.301310	0.
1	1	31066.514349	111.415011	3.200883	1.026490	1.070640	40266.935982	17.289183	626.754967	1.812362	4205.624724	0.
2	2	81201.080802	136.521008	2.115061	1.013575	1.000646	16350.149968	13.574014	488.550743	1.340659	4285.891403	0.
3	3	69569.894646	97.257340	3.326425	1.032815	1.022453	35743.675302	17.784111	406.804836	1.274611	4090.832470	0.
4	4	94957.590260	215.220823	1.141058	1.005038	1.002519	3524.928631	5.640638	461.104954	1.521411	3736.071369	0.

In [50]:
```
# Plot Clusters
plt.figure(figsize=(10, 7))
plt.scatter(airline2['clustersid'],airline2['Balance'], c=hclusters.labels_)
```

Out[50]: <matplotlib.collections.PathCollection at 0x20128c35dc0>



In [ ]:
```
1
```