K-Medoids clustering on socio-economic dataset

In [6]:
```
import pandas as pd
from sklearn_extra.cluster import KMedoids
from sklearn.metrics import silhouette_score

df = pd.read_excel('Preprocessed_data_standardscaler.xlsx')

# Dropping the ID Column, as it is not required in clustering.

new_df = df.drop(columns=['ID'])
new_df
```

Out[6]:

	Normalised Age (Yrs)	Gender (Male)	Education (<Class 10)	Education (Class 10-12)	Graduate	Higher Edu	Income <20k	Income 20-40k	Income 40-60k	Income 60-80k	Income >80k	Experience (<1yr)	Experience (1-2yr)	Experience (2-5yr)	Experience (5-10yr)	Experience (10-20yr)	Experience (20-30yr)	Experience (>30yr)	Helmet (Full mask)
0	-0.927352	1	1	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1
1	0.387063	1	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0
2	-0.739578	1	0	0	1	0	0	0	1	0	0	0	0	0	1	0	0	0	1
3	0.762611	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0
4	-1.678446	1	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
476	-0.739578	1	0	0	1	0	0	0	1	0	0	0	0	0	1	0	0	0	1
477	-1.302899	1	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0
478	2.077026	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	1	0	0
479	-0.551804	1	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0
480	-0.551804	1	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0

481 rows × 19 columns

In [7]:
```
# Now the Silhouette Score method is used to determine the number of clusters to achieve the best possible clustering
# Here the scores are calculated for k = {2,3,4,5,6}

scores = {}

for i in range(2,7):
    kmedoids = KMedoids(n_clusters=i,method='pam',init='build',random_state=42)
    kmedoids.fit(new_df)
    score = silhouette_score(new_df,kmedoids.labels_)
    scores[i]=score

df_scores = pd.DataFrame(list(scores.items()), columns=['no. of clusters', 'silhouette score'])
df_scores

# since setting k = 5 yields the highest score, the number of clusters are chosen to be 5.
# As can be seen in the table below:
```

Out[7]:

	no. of clusters	silhouette score
0	2	0.137599
1	3	0.152380
2	4	0.170523
3	5	0.172540
4	6	0.162440

In [8]:
```
# Now applying the K-Medoids algorithm.

optimal_clusters = 5

kmedoids_1 = KMedoids(n_clusters=optimal_clusters,method='pam',init='build',random_state=42)
kmedoids_1.fit(new_df)

list1 = list(kmedoids_1.labels_)

df_clustered = new_df.copy()

df_clustered['Cluster'] = list1
df_clustered

# As can be seen in the below dataset, a new 'Cluster' column has been added at the very end:
```

Out[8]:

	Normalised Age (Yrs)	Gender (Male)	Education (<Class 10)	Education (Class 10-12)	Graduate	Higher Edu	Income <20k	Income 20-40k	Income 40-60k	Income 60-80k	Income >80k	Experience (<1yr)	Experience (1-2yr)	Experience (2-5yr)	Experience (5-10yr)	Experience (10-20yr)	Experience (20-30yr)	Experience (>30yr)	Helmet (Full mask)
0	-0.927352	1	1	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1
1	0.387063	1	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0
2	-0.739578	1	0	0	1	0	0	0	1	0	0	0	0	0	1	0	0	0	1
3	0.762611	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0
4	-1.678446	1	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
476	-0.739578	1	0	0	1	0	0	0	1	0	0	0	0	0	1	0	0	0	1
477	-1.302899	1	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0
478	2.077026	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	1	0	0
479	-0.551804	1	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0
480	-0.551804	1	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0

481 rows × 20 columns

In [27]:
```
final_df = df_clustered.copy()

old_df = pd.read_excel('Asansol socio-economic data 1.xlsx')
age = list(old_df['Age (Yrs)'])

final_df['Age (Yrs)'] = age

final_df = final_df.drop(columns = ['Normalised Age (Yrs)'])

list2 = list(final_df.columns)
new_order = [list2[-1]] + list2[0:-1]

final_df = final_df[new_order]
final_df
```

Out[27]:

	Age (Yrs)	Gender (Male)	Education (<Class 10)	Education (Class 10-12)	Graduate	Higher Edu	Income <20k	Income 20-40k	Income 40-60k	Income 60-80k	Income >80k	Experience (<1yr)	Experience (1-2yr)	Experience (2-5yr)	Experience (5-10yr)	Experience (10-20yr)	Experience (20-30yr)	Experience (>30yr)	Helmet (Full-mask)	Cluster
0	26	1	1	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1	0
1	40	1	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1
2	28	1	0	0	1	0	0	0	1	0	0	0	0	0	1	0	0	0	1	0
3	44	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	3
4	18	1	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1	4
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
476	28	1	0	0	1	0	0	0	1	0	0	0	0	0	1	0	0	0	1	0
477	22	1	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0	1
478	58	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	1	0	1	2
479	30	1	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	1	0
480	30	1	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	1	0

481 rows × 20 columns

In [32]:
```
cluster_1 = final_df.loc[final_df['Cluster']==0]
cluster_1
cluster_1.describe()
```

Out[32]:

	Age (Yrs)	Gender (Male)	Education (<Class 10)	Education (Class 10-12)	Graduate	Higher Edu	Income <20k	Income 20-40k	Income 40-60k	Income 60-80k	Income >80k	Experience (<1yr)	Experience (1-2yr)	Experience (2-5yr)	Experience (5-10yr)	Experience (10-20yr)	Experience (20-30yr)
count	71.000000	71.000000	71.000000	71.000000	71.000000	71.000000	71.000000	71.000000	71.000000	71.000000	71.000000	71.0	71.000000	71.000000	71.000000	71.000000	71.000000
mean	51.873239	0.943662	0.140845	0.183099	0.577465	0.098592	0.070423	0.239437	0.267606	0.380282	0.042254	0.0	0.014085	0.028169	0.056338	0.169014	0.591549
std	8.349388	0.232214	0.350338	0.389500	0.497479	0.300235	0.257679	0.429777	0.445862	0.488911	0.202599	0.0	0.118678	0.166633	0.232214	0.377432	0.495046
min	35.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.000000	0.000000
25%	46.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.000000	0.000000
50%	51.000000	1.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.000000	1.000000
75%	58.500000	1.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000	1.000000	1.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.000000	1.000000
max	73.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	0.0	1.000000	1.000000	1.000000	1.000000	1.000000

In [30]:
```
cluster_2 = final_df.loc[final_df['Cluster']==1]
cluster_2.describe()
```

Out[30]:

	Age (Yrs)	Gender (Male)	Education (<Class 10)	Education (Class 10-12)	Graduate	Higher Edu	Income <20k	Income 20-40k	Income 40-60k	Income 60-80k	Income >80k	Experience (<1yr)	Experience (1-2yr)	Experience (2-5yr)	Experience (5-10yr)	Experience (10-20yr)	Experience (20-30
count	82.000000	82.000000	82.0	82.000000	82.000000	82.000000	82.000000	82.000000	82.000000	82.000000	82.000000	82.0	82.000000	82.000000	82.000000	82.000000	82.000
mean	38.329268	0.951220	0.0	0.085366	0.878049	0.036585	0.073171	0.073171	0.658537	0.170732	0.024390	0.0	0.048780	0.048780	0.024390	0.853659	0.024
std	6.299235	0.216734	0.0	0.281145	0.329243	0.188897	0.262019	0.262019	0.477119	0.378590	0.155207	0.0	0.216734	0.216734	0.155207	0.355623	0.155
min	28.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.000000	0.000
25%	35.000000	1.000000	0.0	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	1.000000	0.000
50%	37.000000	1.000000	0.0	0.000000	1.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	1.000000	0.000
75%	40.750000	1.000000	0.0	0.000000	1.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	1.000000	0.000
max	58.000000	1.000000	0.0	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	0.0	1.000000	1.000000	1.000000	1.000000	1.000

In [31]:
```
cluster_3 = final_df.loc[final_df['Cluster']==2]
cluster_3.describe()
```

Out[31]:

	Age (Yrs)	Gender (Male)	Education (<Class 10)	Education (Class 10-12)	Graduate	Higher Edu	Income <20k	Income 20-40k	Income 40-60k	Income 60-80k	Income >80k	Experience (<1yr)	Experience (1-2yr)	Experience (2-5yr)	Experience (5-10yr)	Experience (10-20yr)	Experience (20-30yr)
count	97.000000	97.000000	97.000000	97.000000	97.000000	97.000000	97.000000	97.000000	97.000000	97.000000	97.000000	97.0	97.000000	97.000000	97.000000	97.000000	97.000000
mean	32.298969	0.958763	0.742268	0.195876	0.030928	0.030928	0.865979	0.092784	0.020619	0.020619	0.0	0.051546	0.103093	0.14433	0.494845	0.175258	0.030928
std	8.214780	0.199871	0.439658	0.398935	0.174022	0.174022	0.342444	0.291636	0.142842	0.142842	0.0	0.222258	0.305660	0.35325	0.502571	0.382162	0.174022
min	18.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.000000	0.000000
25%	26.000000	1.000000	0.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.000000	0.000000
50%	32.000000	1.000000	1.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.000000	0.000000
75%	38.000000	1.000000	1.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.000000	0.000000
max	50.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	0.0	1.000000	1.000000	1.000000	1.000000	1.000000

In [33]:
```
cluster_4 = final_df.loc[final_df['Cluster']==3]
cluster_4.describe()
```

Out[33]:

	Age (Yrs)	Gender (Male)	Education (<Class 10)	Education (Class 10-12)	Graduate	Higher Edu	Income <20k	Income 20-40k	Income 40-60k	Income 60-80k	Income >80k	Experience (<1yr)	Experience (1-2yr)	Experience (2-5yr)	Experience (5-10yr)	Experience (10-20yr)	Experience (20-30yr)	Exp (20
count	114.000000	114.000000	114.000000	114.000000	114.000000	114.000000	114.000000	114.000000	114.0	114.000000	114.000000	114.000000	114.000000	114.000000	114.000000	114.000000	114.000000	114.0
mean	35.350877	0.964912	0.061404	0.894737	0.017544	0.026316	0.070175	0.912281	0.0	0.008772	0.008772	0.052632	0.149123	0.096491	0.263158	0.412281	0.
std	9.370432	0.184814	0.241129	0.308247	0.131866	0.160779	0.256570	0.284135	0.0	0.093659	0.093659	0.224283	0.357782	0.296567	0.442292	0.494418	0.
min	16.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
25%	28.000000	1.000000	0.000000	1.000000	0.000000	0.000000	0.000000	1.000000	0.0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
50%	34.500000	1.000000	0.000000	1.000000	0.000000	0.000000	0.000000	1.000000	0.0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
75%	42.000000	1.000000	0.000000	1.000000	0.000000	0.000000	0.000000	1.000000	0.0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	1.000000	1.000000	0.0
max	59.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	0.0	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.0

In [34]:
```
cluster_5 = final_df.loc[final_df['Cluster']==4]
cluster_5.describe()
```

Out[34]:

	Age (Yrs)	Gender (Male)	Education (<Class 10)	Education (Class 10-12)	Graduate	Higher Edu	Income <20k	Income 20-40k	Income 40-60k	Income 60-80k	Income >80k	Experience (<1yr)	Experience (1-2yr)	Experience (2-5yr)	Experience (5-10yr)	Experience (10-20yr)	Experience (20-30yr)	Exp (20
count	117.000000	117.000000	117.000000	117.000000	117.000000	117.000000	117.000000	117.000000	117.000000	117.000000	117.000000	117.000000	117.000000	117.000000	117.000000	117.000000	117.0	
mean	27.931624	0.931624	0.017094	0.051282	0.888889	0.042735	0.111111	0.299145	0.495726	0.068376	0.025641	0.034188	0.059829	0.282051	0.623932	0.000000	0.0	
std	4.872029	0.253476	0.130179	0.221521	0.315621	0.203129	0.315621	0.459853	0.502132	0.253476	0.158742	0.182493	0.238190	0.451934	0.486481	0.0	0.0	
min	17.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	
25%	25.000000	1.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	
50%	28.000000	1.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	1.000000	0.0	
75%	30.000000	1.000000	0.000000	0.000000	1.000000	0.000000	0.000000	1.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	1.000000	1.000000	0.0	
max	45.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	0.0	

In [37]:
```
overall_mean = final_df.groupby('Cluster').mean()

overall_mean
```

Out[37]:

	Age (Yrs)	Gender (Male)	Education (<Class 10)	Education (Class 10-12)	Graduate	Higher Edu	Income <20k	Income 20-40k	Income 40-60k	Income 60-80k	Income >80k	Experience (<1yr)	Experience (1-2yr)	Experience (2-5yr)	Experience (5-10yr)	Experience (10-20yr)	Experience (20-30yr)	Exp
Cluster																		
0	51.873239	0.943662	0.140845	0.183099	0.577465	0.098592	0.070423	0.239437	0.267606	0.380282	0.042254	0.000000	0.014085	0.028169	0.056338	0.169014	0.591549	0.
1	38.329268	0.951220	0.000000	0.085366	0.878049	0.036585	0.073171	0.073171	0.658537	0.170732	0.024390	0.000000	0.048780	0.048780	0.024390	0.853659	0.024390	0.
2	32.298969	0.958763	0.742268	0.195876	0.030928	0.030928	0.865979	0.092784	0.020619	0.020619	0.000000	0.051546	0.103093	0.144330	0.494845	0.175258	0.030928	0.
3	35.350877	0.964912	0.061404	0.894737	0.017544	0.026316	0.070175	0.912281	0.000000	0.008772	0.008772	0.052632	0.149123	0.096491	0.263158	0.412281	0.017544	0.
4	27.931624	0.931624	0.017094	0.051282	0.888889	0.042735	0.111111	0.299145	0.495726	0.068376	0.025641	0.034188	0.059829	0.282051	0.623932	0.000000	0.000000	0.

In [48]:
```
# Exporting the datasets to excel:

final_df.to_excel('clustered data KMedoids.xlsx')
overall_mean.to_excel('mean_for_clusters_kmedoids.xlsx')

cluster_1.to_excel('cluster_1_kmedoids.xlsx')
cluster_2.to_excel('cluster_2_kmedoids.xlsx')
cluster_3.to_excel('cluster_3_kmedoids.xlsx')
cluster_4.to_excel('cluster_4_kmedoids.xlsx')
cluster_5.to_excel('cluster_5_kmedoids.xlsx')
```