

In [127]:

```

import pandas as pd
import numpy as np
from sklearn.preprocessing import StandardScaler
from sklearn import cluster
import matplotlib.pyplot as plt

# Clustering demo for k-means with computer-generated appreciations data
# Vesa Ollikainen 2018

num_clusters = 3

# read data (replace with your own path)
df = pd.read_csv('C:\\Users\\olliv\\Documents\\cogsys\\appreciation_v3.csv', sep=';')
df.head(10)

```

Out[127]:

	ID	Sports and exercise	Bars and clubs	Literature	Music	Family	Nature and wildlife	Gardening	Home decoration	Travel	Housing
0	1	4	4	5	7	8	9	6	9	5	
1	2	6	5	6	6	10	8	10	8	5	
2	3	5	6	7	5	6	4	5	6	6	
3	4	2	4	2	4	6	3	1	3	1	
4	5	10	8	3	7	10	5	7	9	9	
5	6	5	7	3	4	2	1	2	2	10	
6	7	7	9	2	7	4	4	2	3	10	
7	8	3	4	1	4	3	3	1	1	5	
8	9	7	6	6	9	9	7	10	10	7	
9	10	7	9	2	9	5	5	5	4	10	

In [128]:

```
# re-encode gender column
df['Gender'].replace(['M','F'], [1,2], inplace=True)

# drop ID
df.drop(columns=['ID'], inplace=True)

df.head(10)
```

Out[128]:

	Sports and exercise	Bars and clubs	Literature	Music	Family	Nature and wildlife	Gardening	Home decoration	Travel	Handic
0	4	4	5	7	8	9	6	9	5	
1	6	5	6	6	10	8	10	8	5	
2	5	6	7	5	6	4	5	6	6	
3	2	4	2	4	6	3	1	3	1	
4	10	8	3	7	10	5	7	9	9	
5	5	7	3	4	2	1	2	2	10	
6	7	9	2	7	4	4	2	3	10	
7	3	4	1	4	3	3	1	1	5	
8	7	6	6	9	9	7	10	10	7	
9	7	9	2	9	5	5	5	4	10	

In [129]:

```
df.describe()
```

Out[129]:

	Sports and exercise	Bars and clubs	Literature	Music	Family	Nature and wildlife	Gardening
count	1000.000000	1000.000000	1000.000000	1000.000000	1000.000000	1000.000000	1000.000000
mean	5.966000	5.931000	4.801000	6.183000	5.878000	4.849000	5.966000
std	2.451296	2.521626	2.445686	2.508746	2.520609	2.166308	2.451296
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	4.000000	4.000000	3.000000	4.750000	4.000000	3.000000	4.000000
50%	6.000000	6.000000	5.000000	6.000000	6.000000	5.000000	6.000000
75%	8.000000	8.000000	7.000000	8.000000	8.000000	6.000000	8.000000
max	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000

In [130]:

```
# save column headings into a list as they will be lost
colnames = df.columns.get_values()

# normalize all variables to mean=0, sd=1
scaler = StandardScaler()
df_scaled = pd.DataFrame(scaler.fit_transform(df))
df_scaled.head()
```

Out[130]:

	0	1	2	3	4	5	6	7	
0	-0.802426	-0.766159	0.081408	0.325824	0.842281	1.917122	0.038732	1.272310	-0.40
1	0.013877	-0.369391	0.490496	-0.072981	1.636137	1.455276	1.669556	0.863075	-0.40
2	-0.394274	0.027377	0.899584	-0.471786	0.048425	-0.392107	-0.368974	0.044607	-0.00
3	-1.618729	-0.766159	-1.145855	-0.870591	0.048425	-0.853953	-1.999797	-1.183097	-1.97
4	1.646483	0.820913	-0.736767	0.325824	1.636137	0.069739	0.446438	1.272310	1.17

In [131]:

```
kmeans = cluster.KMeans(n_clusters=num_clusters)
kmeans.fit(df_scaled)

centroids = kmeans.cluster_centers_
print(centroids)
```

```
[[ 0.73975902  0.29107507  0.91594762  0.76880405  0.46123039  0.45768917
  0.74249515  0.874408    0.14453563  0.50771461 -0.69815738  0.73037298
 -0.15378122  0.78039529]
 [-0.57820095 -0.15845104 -0.55345439 -0.47885377 -0.85496291 -0.81069135
 -0.88299291 -0.85674517 -0.06914773 -0.90206474  0.17312539 -0.14648396
 -0.01917564 -0.12543537]
 [-0.04297253 -0.11432585 -0.28238748 -0.21683599  0.67075169  0.61240751
  0.38382598  0.1936848  -0.07021688  0.68324401  0.56613079 -0.64110734
  0.20554741 -0.7288625 ]]
```

In [132]:

```
# create a pandas data frame with columns
centroids_df = pd.DataFrame(centroids, columns=colnames)
print(centroids_df)
```

	Sports and exercise	Bars and clubs	Literature	Music	Family	\
0	0.739759	0.291075	0.915948	0.768804	0.461230	
1	-0.578201	-0.158451	-0.553454	-0.478854	-0.854963	
2	-0.042973	-0.114326	-0.282387	-0.216836	0.670752	

	Nature and wildlife	Gardening	Home decoration	Travel	Handicraft	\
0	0.457689	0.742495	0.874408	0.144536	0.507715	
1	-0.810691	-0.882993	-0.856745	-0.069148	-0.902065	
2	0.612408	0.383826	0.193685	-0.070217	0.683244	

	Fashion and clothing	Monthly income	Gender	Age
0	-0.698157	0.730373	-0.153781	0.780395
1	0.173125	-0.146484	-0.019176	-0.125435
2	0.566131	-0.641107	0.205547	-0.728862

In [133]:

```
# visualize with matplotlib
fig=plt.figure()
ax = fig.add_subplot(111)
cax = ax.matshow(centroids_df, vmin=-1, vmax=1)
fig.colorbar(cax)
ticks = np.arange(0, len(colnames), 1)
ax.set_xticks(ticks)
#ax.set_yticks(ticks)
ax.set_xticklabels(colnames)
for tick in ax.get_xticklabels():
    tick.set_rotation(90)
#ax.set_yticklabels(cnames)
plt.show()
```

