In [1]:

```
import pandas as pd
dps = {}
for i in range(2,6):
    dp_temp = {}
    for x in range(1,18):
        filename = "DP%02d_PA%02d.xlsx" %(i,x)
        dp_temp["PA%02d" %x] = pd.read_excel(filename)
        dps["DP%02d" %i] = dp_temp
    print("Imported DP0"+str(i))
```

```
Imported DP02
Imported DP03
Imported DP04
Imported DP05
```

In [2]:

```
def getDataVector(table, row , col):
    row = row-2
#     print(dps[table]["PA01"]["Table ID: "+table][row])
    rtn = []
    for x in range(1,18):
        df = dps[table]["PA%02d" %x]
#         print("PA%02d" % x ,"\t",df["Unnamed: "+str(1)][row])
        rtn.append(df["Unnamed: "+str(col)][row])
    return dps[table]["PA01"]["Table ID: "+table][row] , rtn
```

In [3]:

```
vectors = []
things = [ ("DP05" , 52 , 3) , ("DP02" , 93 , 3) ,("DP03",84,1) , ("DP03",53,3) ,("
#             White                 HS              median income  Manufacturing
          ("DP03" , 83 , 3), ("DP03" , 85 , 1) , ("DP03" , 128 , 3),("DP02" , 130 ,
#           200+              Meanincome             public health       foreign bo
         ("DP02" , 204 , 3) , ("DP03" , 43 , 3) , ("DP03" , 37 , 3)]
#               internet           job                 work from home
for table , row , col in things:
    vectors.append(getDataVector(table,row,col))


df = pd.DataFrame({
    "District" : ["PA%02d" %x for x in range(1,18)],
    **{vectors[x][0] : vectors[x][1] for x in range(0,len(vectors))}
})
df
```

Out[3]:

	District	White	Bachelor's degree or higher	Median household income (dollars)	Manufacturing	Educational services, and health care and social assistance	$200,000 or more	Me househo incon (dolla
0	PA01	81.6	44.6	100136	13.1	25.3	17.0	1286
1	PA02	37.6	26.6	52293	8.0	31.6	5.4	740
2	PA03	32.7	43.7	54392	5.8	34.2	7.6	819
3	PA04	76.9	48.7	99271	13.6	25.2	18.1	1336
4	PA05	60.1	42.7	75243	8.6	28.9	13.9	1118
5	PA06	71.3	47.9	94356	12.7	21.8	17.3	1263
6	PA07	70.1	31.3	71407	13.6	25.2	8.4	951
7	PA08	75.9	27.9	63058	11.1	26.5	4.8	788
8	PA09	88.7	23.2	62659	15.2	24.0	4.7	798
9	PA10	73.7	33.6	72359	10.2	23.6	6.3	914
10	PA11	83.4	28.9	75875	15.9	23.0	7.9	950
11	PA12	74.3	40.6	61514	8.0	30.2	7.7	864
12	PA13	91.3	21.9	60754	13.3	25.5	4.3	783
13	PA14	91.1	26.0	58075	12.0	24.1	4.5	776
14	PA15	91.5	25.2	57945	15.8	28.7	4.6	756
15	PA16	88.2	29.3	60630	15.1	25.8	5.5	803
16	PA17	83.0	45.2	77984	8.6	26.2	11.1	1059

In [4]:

```
import numpy as np
from sklearn.preprocessing import minmax_scale
from sklearn.metrics.pairwise import cosine_similarity
vector_scaled = np.array([minmax_scale(x[1]) for x in  vectors])
len(vector_scaled)
# for i in range(1,18):
#     print(cosine_similarity(vector_scaled[15]))
print(vector_scaled.T.shape)
Final_scores = []
for i in range(1,18):
    v = vector_scaled.T[i-1]
    Final_scores .append( ("PA" + str(i) , cosine_similarity(v.reshape(1, -1) , vec
# for v, in vector_scaled.T:
#     Final_scores .append( cosine_similarity(v.reshape(1, -1) , vector_scaled.T[15
Final_scores.sort(key = lambda x:-x[1])
print(Final_scores[0])
print(Final_scores[1])
print(Final_scores[2])
print(Final_scores[3])
```

```
(17, 13)
('PA16', 1.0)
('PA9', 0.9752702518856211)
('PA14', 0.9751573173135198)
('PA15', 0.9702141915411692)
```

In [5]:

```
[print(x,y) for x,y in Final_scores]
pass
```

```
PA16 1.0
PA9 0.9752702518856211
PA14 0.9751573173135198
PA15 0.9702141915411692
PA13 0.964508563094101
PA8 0.9498110836484239
PA11 0.9209997948393922
PA7 0.8788752760900211
PA10 0.866657660286038
PA12 0.7825011365202779
PA17 0.7564135871413133
PA1 0.6778928336232144
PA5 0.6600740668636202
PA4 0.6583414632706031
PA6 0.6304461161623047
PA3 0.5231973897839859
PA2 0.4622191969095474
```

In [6]:

```
df.iloc[[15,8,13]]
```

Out[6]:

	District	White	Bachelor's degree or higher	Median household income (dollars)	Manufacturing	Educational services, and health care and social assistance	$200,000 or more	Me househ inco (dolla
15	PA16	88.2	29.3	60630	15.1	25.8	5.5	803
8	PA09	88.7	23.2	62659	15.2	24.0	4.7	798
13	PA14	91.1	26.0	58075	12.0	24.1	4.5	776