

###

HW 9

Youyu Zhang

zhang.youy@northeastern.edu

(530)574-2826

Code available on: https://github.com/kuohu233/IE_7300

Submitted by 11/22/2022

```
In [1]: ## imports
import numpy as np
import pandas as pd
from scipy import stats
import matplotlib.pyplot as plt
import seaborn as sns
from typing import Dict, Any
from abc import ABC, abstractmethod
from sklearn.preprocessing import StandardScaler
```

Part 1

K mean with an optimum number of clusters (k)

```
In [2]: df = pd.read_excel('EastWestAirlines.xlsx', sheet_name='data')
df.head()
```

```
Out[2]:
```

	ID#	Balance	Qual_miles	cc1_miles	cc2_miles	cc3_miles	Bonus_miles	Bonus_trans	Flight_miles_12mo	Flight
0	1	28143	0	1	1	1	174	1	0	
1	2	19244	0	1	1	1	215	2	0	
2	3	41354	0	1	1	1	4123	4	0	
3	4	14776	0	1	1	1	500	1	0	
4	5	97752	0	4	1	1	43300	26	2077	

```
In [3]: df.columns
```

```
Out[3]: Index(['ID#', 'Balance', 'Qual_miles', 'cc1_miles', 'cc2_miles', 'cc3_miles',
              'Bonus_miles', 'Bonus_trans', 'Flight_miles_12mo', 'Flight_trans_12',
              'Days_since_enroll', 'Award?'],
              dtype='object')
```

```
In [4]: df.isnull().sum()
```

```
Out[4]: ID#          0
Balance        0
Qual_miles     0
cc1_miles      0
cc2_miles      0
cc3_miles      0
Bonus_miles    0
Bonus_trans    0
Flight_miles_12mo  0
Flight_trans_12  0
Days_since_enroll  0
Award?         0
dtype: int64
```

```
In [5]: y = df['Award?']  
x = np.array(df.drop(['Award?'],axis=1))
```

```
In [8]: def init_centroids(k, X):  
    arr = []  
    for i in range(k):  
        cx1 = np.random.uniform(min(X[:,0]), max(X[:,0]))  
        cx2 = np.random.uniform(min(X[:,1]), max(X[:,1]))  
        arr.append([cx1, cx2])  
    return np.asarray(arr)  
  
def init_centroids2(k, X):  
    arr = []  
    for i in range(k):  
        cx_list = []  
        for j in range(X.shape[1]):  
            cx = np.random.uniform(min(X[:,j]), max(X[:,j]))  
            cx_list.append(cx)  
        arr.append(cx_list)  
    return np.asarray(arr)  
  
def dist(a, b):  
    return np.sqrt(sum(np.square(a-b)))  
  
def assign_cluster(k, X, cg):  
    cluster = [-1]*len(X)  
    for i in range(len(X)):  
        dist_arr = []  
        for j in range(k):  
            dist_arr.append(dist(X[i], cg[j]))  
        idx = np.argmin(dist_arr)  
        cluster[i] = idx  
    return np.asarray(cluster)  
  
def show_clusters(X,y, cluster, cg):  
    df1 = pd.DataFrame(dict(y=y, label=cluster))  
    df2 = pd.DataFrame(X)  
    df = pd.concat([df1,df2], axis=1)  
    # df = pd.DataFrame(dict(x=X[:,0], y=X[:,1], label=cluster))  
    colors = {0:'blue', 1:'orange', 2:'green'}  
    fig, ax = plt.subplots(figsize=(8, 8))  
    grouped = df.groupby('label')  
    for key, group in grouped:  
        group.plot(ax=ax, kind='scatter', x='x', y='y', label=key, color=colors[key])  
    ax.scatter(cg[:, 0], cg[:, 1], marker='*', s=150, c='#ff2222')  
    plt.xlabel('X_1')  
    plt.ylabel('X_2')  
    plt.show()  
  
def compute_centroids(k, X, cluster):  
    cg_arr = []  
    for i in range(k):  
        arr = []  
        for j in range(len(X)):  
            if i not in np.unique(cluster):  
                pass  
            if cluster[j]==i:  
                arr.append(X[j])  
        cg_arr.append(np.mean(arr, axis=0))  
    return np.asarray(cg_arr)  
  
def measure_change(cg_prev, cg_new):  
    res = 0  
    for a,b in zip(cg_prev,cg_new):
```

```

    res+=dist(a,b)
    return res

def k_means(k, X, threshold, itermax):
    cg_prev = init_centroids2(k, X)
    cluster = [0]*len(X)
    cg_change = 100
    iter = 0
    while cg_change>=threshold and iter <= itermax:
        # print(f"Iteration {iter} with cg_change={cg_change}")
        cluster = assign_cluster(k, X, cg_prev)
        # show_clusters(X, cluster, cg_prev)
        cg_new = compute_centroids(k, X, cluster)
        cg_change = measure_change(cg_new, cg_prev)
        cg_prev = cg_new
        iter += 1
    print(f"Iteration {iter} with cg_change={cg_change}")
    # show_clusters(X, cluster, cg_prev)
    return cluster, cg_prev

```

```

In [9]: group = []
        centroids = []
        k_range = range(2,15)
        for i in k_range:
            print(f"k = {i}")
            cluster, cg_prev = k_means(k=i, X=x,
                                      threshold=0.1, itermax=100)
            group.append(cluster)
            centroids.append(cg_prev)

```

```

k = 2
Iteration 23 with cg_change=0.0
k = 3
Iteration 23 with cg_change=0.0
k = 4
Iteration 50 with cg_change=0.0
k = 5
Iteration 24 with cg_change=0.0
k = 6
Iteration 59 with cg_change=0.0
k = 7

```

```

c:\Users\youyu\AppData\Local\Programs\Python\Python38\lib\site-packages\numpy\core\fromn
umeric.py:3440: RuntimeWarning: Mean of empty slice.
    return _methods._mean(a, axis=axis, dtype=dtype,
c:\Users\youyu\AppData\Local\Programs\Python\Python38\lib\site-packages\numpy\core\_meth
ods.py:189: RuntimeWarning: invalid value encountered in double_scalars
    ret = ret.dtype.type(ret / rcount)
C:\Users\youyu\AppData\Local\Temp\ipykernel_22884\119970047.py:57: VisibleDeprecationWar
ning: Creating an ndarray from ragged nested sequences (which is a list-or-tuple of list
s-or-tuples-or ndarrays with different lengths or shapes) is deprecated. If you meant to
do this, you must specify 'dtype=object' when creating the ndarray.
    return np.asarray(cg_arr)

```

```

Iteration 1 with cg_change=nan
k = 8
Iteration 1 with cg_change=nan
k = 9
Iteration 1 with cg_change=nan
k = 10
Iteration 1 with cg_change=nan
k = 11
Iteration 1 with cg_change=nan
k = 12
Iteration 1 with cg_change=nan
k = 13
Iteration 1 with cg_change=nan

```

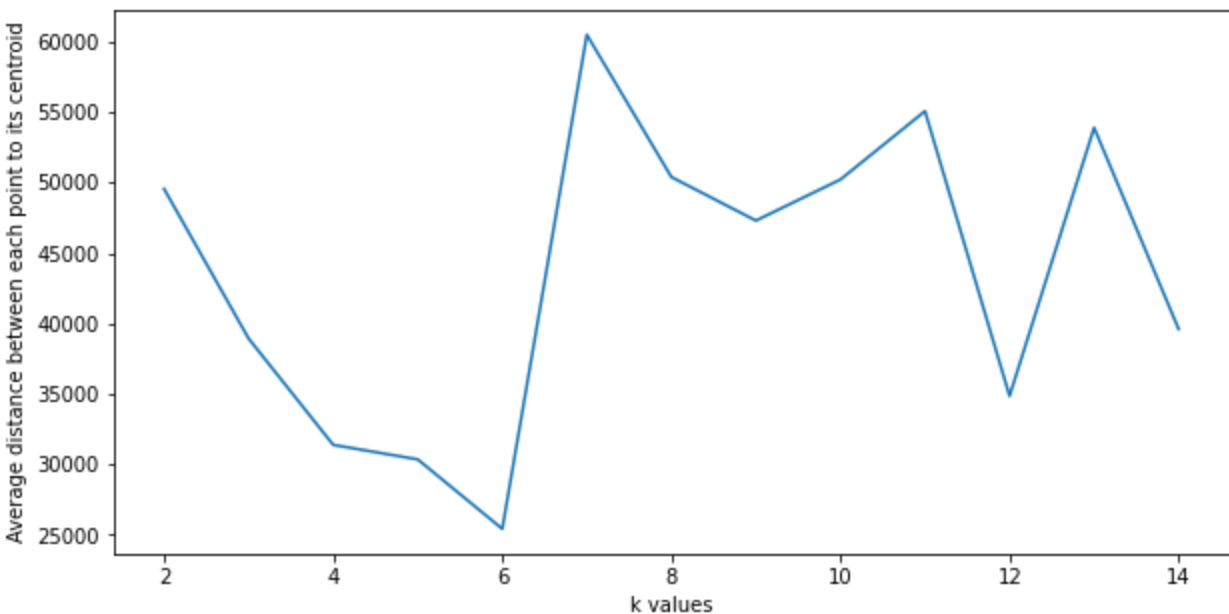
```
k = 14
Iteration 1 with cg_change=nan
```

```
In [10]: dist_k = []
for j in range(len(group)):
    # Calculate distance between each point and its centroid
    dist_list = []
    for i in range(len(x)):
        group_ind = group[j][i]
        cent = centroids[j][group_ind]
        distance = dist(x[i], cent)
        dist_list.append(distance)
    dist_mean = np.mean(dist_list)
    dist_k.append(dist_mean)
```

```
In [15]: centroids[4]
```

```
Out[15]: array([[1.16943590e+03, 3.63319265e+05, 4.49094017e+02, 3.18803419e+00,
 1.02564103e+00, 1.00000000e+00, 4.99647436e+04, 2.01111111e+01,
 1.58948718e+03, 4.73504274e+00, 5.66887179e+03],
 [1.88453484e+03, 8.28577991e+04, 1.47406335e+02, 2.68054299e+00,
 1.00904977e+00, 1.02171946e+00, 2.51761792e+04, 1.48361991e+01,
 5.80251584e+02, 1.72760181e+00, 4.38890498e+03],
 [1.55797699e+03, 1.76979146e+05, 2.68813808e+02, 3.07531381e+00,
 1.00836820e+00, 1.02928870e+00, 3.47681506e+04, 1.73284519e+01,
 8.97516736e+02, 2.43723849e+00, 4.96983054e+03],
 [2.23271598e+03, 2.31838785e+04, 9.64610304e+01, 1.46895641e+00,
 1.01805372e+00, 1.00352268e+00, 7.35145883e+03, 8.25451343e+00,
 2.35384852e+02, 7.40202554e-01, 3.69813298e+03],
 [9.07000000e+02, 7.20621217e+05, 2.47391304e+02, 3.47826087e+00,
 1.00000000e+00, 1.13043478e+00, 6.02981739e+04, 2.16086957e+01,
 1.53347826e+03, 5.52173913e+00, 6.25973913e+03],
 [3.91200000e+02, 1.28891580e+06, 1.52800000e+03, 3.00000000e+00,
 1.00000000e+00, 1.00000000e+00, 3.90980000e+04, 2.46000000e+01,
 2.75460000e+03, 1.14000000e+01, 7.82160000e+03]])
```

```
In [12]: plt.figure(figsize=(10,5))
plt.plot(k_range,dist_k)
plt.xlabel('k values')
plt.ylabel('Average distance between each point to its centroid')
plt.show()
```



From the figure above we can see that k=5 and k=6 has significant lower average distance between each point to its centroids. And thus k=6 can be a better solution.

Part 2

Hierarchy cluster with Dendrogram

```
In [52]: import math
def get_distance_measure(M):
    if M == 0:
        return single_link
    elif M == 1:
        return complete_link
    else:
        return average_link

def distance(p, q):
    return math.sqrt(sum([(pi - qi)**2 for pi, qi in zip(p, q)]))

def single_link(ci, cj):
    return min([distance(vi, vj) for vi in ci for vj in cj])

def complete_link(ci, cj):
    return max([distance(vi, vj) for vi in ci for vj in cj])

def average_link(ci, cj):
    distances = [distance(vi, vj) for vi in ci for vj in cj]
    return sum(distances) / len(distances)

class AgglomerativeHierarchicalClustering:
    def __init__(self, data, K, M):
        self.data = data
        self.N = len(data)
        self.K = K
        self.measure = get_distance_measure(M)
        self.clusters = self.init_clusters()

    # Replace self.measure into distance function
    # def measure(self, a, b):
    #     return np.sqrt(sum(np.square(np.array(a)-np.array(b))))

    def init_clusters(self):
        return {data_id: [data_point] for data_id, data_point in enumerate(self.data)}

    def find_closest_clusters(self):
        min_dist = math.inf
        closest_clusters = None

        clusters_ids = list(self.clusters.keys())

        for i, cluster_i in enumerate(clusters_ids[:-1]):
            for j, cluster_j in enumerate(clusters_ids[i+1:]):
                dist = self.measure(self.clusters[cluster_i], self.clusters[cluster_j])
                if dist < min_dist:
                    min_dist, closest_clusters = dist, (cluster_i, cluster_j)
                # distt = self.measure(self.clusters[cluster_i], self.clusters[cluster_j])
                # dist = np.sqrt(np.dot(distt,distt))
                # if dist < min_dist:
                #     min_dist, closest_clusters = dist, (cluster_i, cluster_j)
        return closest_clusters

    def merge_and_form_new_clusters(self, ci_id, cj_id):
        new_clusters = {0: self.clusters[ci_id] + self.clusters[cj_id]}
```

```

        for cluster_id in self.clusters.keys():
            if (cluster_id == ci_id) | (cluster_id == cj_id):
                continue
            new_clusters[len(new_clusters.keys())] = self.clusters[cluster_id]
        return new_clusters

    def run_algorithm(self):
        while len(self.clusters.keys()) > self.K:
            closest_clusters = self.find_closest_clusters()
            self.clusters = self.merge_and_form_new_clusters(*closest_clusters)
            print(f'Length of self.cluster.keys: {len(self.clusters.keys())}')

    def print(self):
        for id, points in self.clusters.items():
            print("Cluster: {}".format(id))
            for point in points:
                print("    {}".format(point))

```

```

In [77]: # As full dataset training will exceed 20 hours, a subset sample is used to show the den
# M=2 for average_link
agg_hierarchical_clustering = AgglomerativeHierarchicalClustering(x[0:400,:], 6, 2)

```

```

In [78]: agg_hierarchical_clustering.run_algorithm()

```

```

Length of self.cluster.keys: 399
Length of self.cluster.keys: 398
Length of self.cluster.keys: 397
Length of self.cluster.keys: 396
Length of self.cluster.keys: 395
Length of self.cluster.keys: 394
Length of self.cluster.keys: 393
Length of self.cluster.keys: 392
Length of self.cluster.keys: 391
Length of self.cluster.keys: 390
Length of self.cluster.keys: 389
Length of self.cluster.keys: 388
Length of self.cluster.keys: 387
Length of self.cluster.keys: 386
Length of self.cluster.keys: 385
Length of self.cluster.keys: 384
Length of self.cluster.keys: 383
Length of self.cluster.keys: 382
Length of self.cluster.keys: 381
Length of self.cluster.keys: 380
Length of self.cluster.keys: 379
Length of self.cluster.keys: 378
Length of self.cluster.keys: 377
Length of self.cluster.keys: 376
Length of self.cluster.keys: 375
Length of self.cluster.keys: 374
Length of self.cluster.keys: 373
Length of self.cluster.keys: 372
Length of self.cluster.keys: 371
Length of self.cluster.keys: 370
Length of self.cluster.keys: 369
Length of self.cluster.keys: 368
Length of self.cluster.keys: 367
Length of self.cluster.keys: 366
Length of self.cluster.keys: 365
Length of self.cluster.keys: 364
Length of self.cluster.keys: 363
Length of self.cluster.keys: 362
Length of self.cluster.keys: 361
Length of self.cluster.keys: 360
Length of self.cluster.keys: 359

```

[illegible]

Length of self.cluster.keys: 292
Length of self.cluster.keys: 291
Length of self.cluster.keys: 290
Length of self.cluster.keys: 289
Length of self.cluster.keys: 288
Length of self.cluster.keys: 287
Length of self.cluster.keys: 286
Length of self.cluster.keys: 285
Length of self.cluster.keys: 284
Length of self.cluster.keys: 283
Length of self.cluster.keys: 282
Length of self.cluster.keys: 28
Length of self.cluster.keys: 280
Length of self.cluster.keys: 279
Length of self.cluster.keys: 278
Length of self.cluster.keys: 277
Length of self.cluster.keys: 276
Length of self.cluster.keys: 275
Length of self.cluster.keys: 274
Length of self.cluster.keys: 273
Length of self.cluster.keys: 272
Length of self.cluster.keys: 271
Length of self.cluster.keys: 270
Length of self.cluster.keys: 269
Length of self.cluster.keys: 268
Length of self.cluster.keys: 267
Length of self.cluster.keys: 266
Length of self.cluster.keys: 265
Length of self.cluster.keys: 264
Length of self.cluster.keys: 263
Length of self.cluster.keys: 262
Length of self.cluster.keys: 261
Length of self.cluster.keys: 260
Length of self.cluster.keys: 259
Length of self.cluster.keys: 258
Length of self.cluster.keys: 257
Length of self.cluster.keys: 256
Length of self.cluster.keys: 255
Length of self.cluster.keys: 254
Length of self.cluster.keys: 253
Length of self.cluster.keys: 252
Length of self.cluster.keys: 251
Length of self.cluster.keys: 250
Length of self.cluster.keys: 249
Length of self.cluster.keys: 248
Length of self.cluster.keys: 247
Length of self.cluster.keys: 246
Length of self.cluster.keys: 245
Length of self.cluster.keys: 244
Length of self.cluster.keys: 243
Length of self.cluster.keys: 242
Length of self.cluster.keys: 241
Length of self.cluster.keys: 240
Length of self.cluster.keys: 239
Length of self.cluster.keys: 238
Length of self.cluster.keys: 237
Length of self.cluster.keys: 236
Length of self.cluster.keys: 235
Length of self.cluster.keys: 234
Length of self.cluster.keys: 233
Length of self.cluster.keys: 232
Length of self.cluster.keys: 231
Length of self.cluster.keys: 230
Length of self.cluster.keys: 229
Length of self.cluster.keys: 228
Length of self.cluster.keys: 227

Length of self.cluster.keys: 226
Length of self.cluster.keys: 225
Length of self.cluster.keys: 224
Length of self.cluster.keys: 223
Length of self.cluster.keys: 222
Length of self.cluster.keys: 221
Length of self.cluster.keys: 220
Length of self.cluster.keys: 219
Length of self.cluster.keys: 218
Length of self.cluster.keys: 217
Length of self.cluster.keys: 216
Length of self.cluster.keys: 215
Length of self.cluster.keys: 214
Length of self.cluster.keys: 213
Length of self.cluster.keys: 212
Length of self.cluster.keys: 211
Length of self.cluster.keys: 210
Length of self.cluster.keys: 209
Length of self.cluster.keys: 208
Length of self.cluster.keys: 207
Length of self.cluster.keys: 206
Length of self.cluster.keys: 205
Length of self.cluster.keys: 204
Length of self.cluster.keys: 203
Length of self.cluster.keys: 202
Length of self.cluster.keys: 201
Length of self.cluster.keys: 200
Length of self.cluster.keys: 199
Length of self.cluster.keys: 198
Length of self.cluster.keys: 197
Length of self.cluster.keys: 196
Length of self.cluster.keys: 195
Length of self.cluster.keys: 194
Length of self.cluster.keys: 193
Length of self.cluster.keys: 192
Length of self.cluster.keys: 191
Length of self.cluster.keys: 190
Length of self.cluster.keys: 189
Length of self.cluster.keys: 188
Length of self.cluster.keys: 187
Length of self.cluster.keys: 186
Length of self.cluster.keys: 185
Length of self.cluster.keys: 184
Length of self.cluster.keys: 183
Length of self.cluster.keys: 182
Length of self.cluster.keys: 181
Length of self.cluster.keys: 180
Length of self.cluster.keys: 179
Length of self.cluster.keys: 178
Length of self.cluster.keys: 177
Length of self.cluster.keys: 176
Length of self.cluster.keys: 175
Length of self.cluster.keys: 174
Length of self.cluster.keys: 173
Length of self.cluster.keys: 172
Length of self.cluster.keys: 171
Length of self.cluster.keys: 170
Length of self.cluster.keys: 169
Length of self.cluster.keys: 168
Length of self.cluster.keys: 167
Length of self.cluster.keys: 166
Length of self.cluster.keys: 165
Length of self.cluster.keys: 164
Length of self.cluster.keys: 163
Length of self.cluster.keys: 162
Length of self.cluster.keys: 161

Length of self.cluster.keys: 160
Length of self.cluster.keys: 159
Length of self.cluster.keys: 158
Length of self.cluster.keys: 157
Length of self.cluster.keys: 156
Length of self.cluster.keys: 155
Length of self.cluster.keys: 154
Length of self.cluster.keys: 153
Length of self.cluster.keys: 152
Length of self.cluster.keys: 151
Length of self.cluster.keys: 150
Length of self.cluster.keys: 149
Length of self.cluster.keys: 148
Length of self.cluster.keys: 147
Length of self.cluster.keys: 146
Length of self.cluster.keys: 145
Length of self.cluster.keys: 144
Length of self.cluster.keys: 143
Length of self.cluster.keys: 142
Length of self.cluster.keys: 141
Length of self.cluster.keys: 140
Length of self.cluster.keys: 139
Length of self.cluster.keys: 138
Length of self.cluster.keys: 137
Length of self.cluster.keys: 136
Length of self.cluster.keys: 135
Length of self.cluster.keys: 134
Length of self.cluster.keys: 133
Length of self.cluster.keys: 132
Length of self.cluster.keys: 131
Length of self.cluster.keys: 130
Length of self.cluster.keys: 129
Length of self.cluster.keys: 128
Length of self.cluster.keys: 127
Length of self.cluster.keys: 126
Length of self.cluster.keys: 125
Length of self.cluster.keys: 124
Length of self.cluster.keys: 123
Length of self.cluster.keys: 122
Length of self.cluster.keys: 121
Length of self.cluster.keys: 120
Length of self.cluster.keys: 119
Length of self.cluster.keys: 118
Length of self.cluster.keys: 117
Length of self.cluster.keys: 116
Length of self.cluster.keys: 115
Length of self.cluster.keys: 114
Length of self.cluster.keys: 113
Length of self.cluster.keys: 112
Length of self.cluster.keys: 111
Length of self.cluster.keys: 110
Length of self.cluster.keys: 109
Length of self.cluster.keys: 108
Length of self.cluster.keys: 107
Length of self.cluster.keys: 106
Length of self.cluster.keys: 105
Length of self.cluster.keys: 104
Length of self.cluster.keys: 103
Length of self.cluster.keys: 102
Length of self.cluster.keys: 101
Length of self.cluster.keys: 100
Length of self.cluster.keys: 99
Length of self.cluster.keys: 98
Length of self.cluster.keys: 97
Length of self.cluster.keys: 96
Length of self.cluster.keys: 95

[illegible]

```

Length of self.cluster.keys: 28
Length of self.cluster.keys: 27
Length of self.cluster.keys: 26
Length of self.cluster.keys: 25
Length of self.cluster.keys: 24
Length of self.cluster.keys: 23
Length of self.cluster.keys: 22
Length of self.cluster.keys: 21
Length of self.cluster.keys: 20
Length of self.cluster.keys: 19
Length of self.cluster.keys: 18
Length of self.cluster.keys: 17
Length of self.cluster.keys: 16
Length of self.cluster.keys: 15
Length of self.cluster.keys: 14
Length of self.cluster.keys: 13
Length of self.cluster.keys: 12
Length of self.cluster.keys: 11
Length of self.cluster.keys: 10
Length of self.cluster.keys: 9
Length of self.cluster.keys: 8
Length of self.cluster.keys: 7
Length of self.cluster.keys: 6

```

```
In [81]: agg_hierarchical_clustering.print()
```

```

Cluster: 0
[ 73 252386      0      4      1      1 39787      13      0      0
7787]
[ 138 259484 1776      1      1      1 19172      26 7172      23
6723]
[ 69 230715      0      3      1      1 24047      12      0      0
6826]
[ 301 238868      0      3      1      1 20521      16      0      0
7220]
[ 158 220081      0      4      1      1 52574      21    500      1
7626]
[ 306 217846      0      4      1      1 49198      20      0      0
7215]
[ 130 213150      0      4      1      1 56308      41   5200     14
7645]
[ 362 224081      0      4      1      1 40108      15    150      2
7059]
[ 367 222227      0      4      1      1 38127      14    500      1
7047]
[ 246 227881      0      4      1      1 41186      13      0      0
7369]
[ 161 228829      0      5      1      1 59852      21      0      0
7612]
[ 259 236274      0      5      1      1 61515      12      0      0
7375]
[ 225 229744      0      5      1      1 68754      35    550      2
7467]
[ 159 212976      0      3      1      1 19926      12      0      0
7624]
[ 325 205523      0      3      1      1 32404      20   1000      2
7159]
[ 228 193976      0      3      1      1 10849      14      0      0
7416]
[ 286 190542 1745      1      1      1  8487      9      0      0
7267]
[ 262 198137      0      1      1      1  6303      6    500      1
7323]
[ 394 198859      0      3      1      1 17855      14    500      1
8296]
[ 25 205651    500      1      1      1  4025      21    700      4

```

7932]	[216	204582	0	1	1	1	4671	14	0	0
7498]	[191	211595	0	1	1	1	3250	8	0	0
7521]	[141	160447	0	1	1	1	8578	25	0	0
6716]	[266	164613	0	2	1	1	11095	13	0	0
7375]	[287	156230	0	1	1	1	5300	8	1300	5
7260]	[386	167670	0	1	3	1	15907	13	0	0
6994]	[250	176090	0	1	1	1	5300	9	2200	6
7344]	[284	168515	0	1	1	1	4600	4	0	0
7269]	[113	181019	0	3	1	1	18415	15	0	0
7714]	[239	182317	0	2	1	1	19489	17	273	1
7397]	[318	179559	0	3	1	1	17819	12	1000	2
7173]	[31	177926	0	3	1	1	20797	13	0	0
7886]	[22	185681	2024	1	1	1	13300	16	1800	9
6896]	[222	189053	0	3	1	1	24156	22	0	0
7439]	[314	179433	0	4	1	1	30476	15	0	0
7187]	[155	298121	0	5	1	1	60808	20	500	1
7640]	[296	294881	0	5	1	1	84800	20	0	0
7279]	[33	276571	0	4	1	1	42044	23	0	0
7872]	[340	278457	6727	4	1	1	57313	27	1000	2
7101]	[46	288865	967	1	1	1	23600	14	2000	4
6884]	[327	278727	0	1	1	1	21666	19	9166	16
7155]	[114	298911	0	1	1	1	8550	11	2550	8
7714]	[117	295638	0	1	1	1	5422	20	818	2
7700]	[241	285116	0	1	1	1	8471	18	500	1
7406]	[357	312512	0	1	1	1	100	1	100	1
7078]	[97	185549	0	5	1	1	110859	39	2950	16
6778]	[214	201346	0	5	1	1	99612	34	600	4
7463]	[124	205126	0	5	1	1	126630	45	2250	10
7668]	[374	161813	0	5	1	1	126941	20	0	0
7026]	[76	123867	0	3	1	1	25308	17	0	0
7759]	[121	122705	0	3	1	1	24222	15	0	0
7682]	[45	121260	0	3	1	1	18493	18	0	0
7808]	[78	129871	0	3	1	1	15776	22	0	0

7752]	[184	127807	0	3	1	1	17086	16	1500	3
7537]	[137	125948	0	3	1	1	15155	15	0	0
6761]	[308	125465	0	3	1	1	14750	9	0	0
7206]	[61	134457	0	3	1	1	15588	14	0	0
6861]	[337	133002	0	3	1	1	15672	21	150	1
7117]	[139	129298	0	4	1	1	29099	15	0	0
6722]	[146	126787	0	4	1	1	35021	17	0	0
6701]	[234	133309	0	3	1	1	23987	13	0	0
7415]	[281	120073	0	3	1	1	12423	29	0	0
7276]	[319	119977	2556	1	1	1	9275	16	1400	4
7226]	[54	121395	0	1	1	1	4970	8	650	2
6889]	[310	123370	0	1	1	1	7478	16	0	0
7197]	[312	112969	0	2	1	1	7572	7	0	0
7192]	[400	114713	0	1	1	1	7142	18	0	0
8296]	[230	117329	0	1	2	1	6448	11	0	0
7416]	[79	109380	0	1	1	1	7537	16	0	0
7749]	[179	106961	0	1	1	1	7357	19	200	2
7572]	[51	108137	0	1	1	1	6368	5	6368	5
6844]	[80	111157	0	4	1	1	32883	19	0	0
7771]	[219	114356	0	4	1	1	27308	14	0	0
7456]	[10	104860	0	3	1	1	28426	28	1150	3
6931]	[341	105166	1374	4	1	1	37663	22	0	0
7120]	[201	108341	0	3	1	1	19670	21	0	0
7488]	[376	107707	0	3	1	1	19407	25	2211	12
7024]	[269	105769	0	3	1	1	23097	16	0	0
7345]	[356	105423	0	3	1	1	20802	14	0	0
7078]	[348	112501	0	1	1	2	19140	11	0	0
7101]	[53	118531	0	4	1	1	44577	38	0	0
6868]	[177	120941	0	4	1	1	42301	21	0	0
7588]	[203	118940	0	4	1	1	36001	17	150	1
7484]	[231	119239	0	4	1	1	36362	15	250	1
7467]	[119	143967	0	2	1	1	7346	11	0	0
7689]	[135	145074	0	1	1	1	8000	4	0	0

6746]	[96	143566	0	1	1	1	5000	5	1500	3
6785]	[375	138020	0	1	1	1	6568	15	0	0
7024]	[264	149733	0	1	1	1	605	3	500	1
7323]	[378	149678	0	1	1	1	0	0	0	0
7022]	[361	148691	0	1	1	1	3000	3	1000	2
7064]	[199	134601	0	1	1	1	0	0	0	0
7493]	[165	152945	0	1	1	1	18604	6	3604	2
7598]	[176	149285	0	1	1	1	15656	13	3250	6
7561]	[274	145726	0	3	1	1	19019	15	0	0
7292]	[382	146263	0	3	1	1	20853	21	350	1
7001]	[169	95658	0	1	1	1	5269	18	0	0
	[364	95305	0	1	1	1	3189	13	0	0
	[229	98224	0	2	1	1	5570	11	0	0
	[402	97763	0	1	1	1	2000	9	0	0
	[136	101101	0	1	1	1	10850	11	1850	6
6725]	[148	100640	0	1	1	1	7600	8	1600	5
6708]	[84	88443	0	1	1	1	1200	3	1200	3
	[156	87474	0	1	1	1	4581	15	0	0
	[395	91939	0	1	1	1	1244	8	0	0
	[258	96098	0	1	1	1	12500	4	0	0
	[351	93045	0	3	1	1	17636	15	0	0
	[253	88702	0	2	1	1	9620	13	50	1
	[354	87938	0	1	1	1	11268	2	0	0
	[396	88449	0	2	1	1	11402	18	1450	5
	[189	89999	0	3	1	1	11694	13	0	0
	[47	92336	0	2	1	1	11214	6	0	0
	[167	80331	0	3	1	1	15489	18	0	0
	[324	84700	0	3	1	1	13544	17	0	0
	[72	84409	5031	2	1	1	15436	16	1150	4
	[65	80250	0	1	1	1	895	2	0	0
	[370	80214	0	1	1	1	500	1	500	1
	[123	77540	0	1	1	1	1000	1	0	0
	[151	84631	0	3	1	1	20415	17	0	0
	[256	87293	0	3	1	1	21584	17	1050	2
	[7	84914	0	3	1	1	27482	25	0	0
	[19	91473	0	3	1	1	27408	17	0	0
	[60	95118	0	3	1	1	23188	23	2200	7
	[12	96522	0	5	1	1	61105	19	0	0
	[377	95656	0	5	1	1	62666	27	0	0
	[252	92875	0	5	1	1	60990	14	0	0
	[235	81380	0	5	1	1	56524	22	0	0
	[5	97752	0	4	1	1	43300	26	2077	4
	[289	98006	0	4	1	1	46349	20	2309	3
	[109	96627	1182	1	1	3	49059	26	2300	9
	[100	102062	0	4	1	1	44247	16	0	0
6758]	[309	94222	0	4	1	1	52081	24	0	0
	[140	103302	0	4	1	1	53552	22	700	3
6718]	[57	75971	0	4	1	1	34339	14	0	0
	[384	76150	0	4	1	1	36907	17	0	0
	[95	78629	0	4	1	1	36679	25	0	0
	[288	79211	0	4	1	1	36776	14	0	0

[68 83237	0	4	1	1 35287	18	0	0 6837]
[125 81974	0	4	1	1 37266	18	1050	3 7661]
[127 77699	0	4	1	1 42775	13	0	0 7654]
[153 77097	0	4	1	1 42403	22	1718	4 6683]
[111 84674	0	4	1	1 46335	14	50	1 6738]
[387 86853	0	4	1	1 42293	21	500	1 6980]
[173 71767	0	5	1	1 67335	13	0	0 7570]
[276 70676	0	5	1	1 74993	15	0	0 7288]
[334 172288	0	5	1	1 66879	10	0	0
7130]							
[345 174910	0	4	1	1 63960	20	200	1
7110]							
[352 197328	0	5	1	1 60447	12	0	0
7092]							
[207 152881	0	5	1	1 79687	18	500	1
7474]							
[244 163530	0	5	1	1 88987	25	0	0
7406]							
[295 145993	0	5	1	1 95082	17	500	1
7229]							
[292 137908	0	5	1	1 57713	19	0	0
7250]							
[332 138999	0	5	1	1 63738	29	600	4
7153]							
[233 153747	0	5	1	1 58689	15	0	0
7409]							
[195 143080	733	4	1	1 44373	38	3928	17
7575]							
[223 143481	0	4	1	1 45891	15	0	0
7437]							
[132 142175	0	4	1	1 37461	22	0	0
6732]							
[299 141813	0	4	1	1 39743	14	0	0
7254]							
[302 149286	0	4	1	1 42141	30	2400	13
7222]							
[353 151301	0	4	1	1 37772	14	0	0
7092]							
[128 133954	500	4	1	1 37670	18	1500	3
7654]							
[339 133370	0	4	1	1 44299	18	0	0
7101]							
[21 120576	0	5	1	1 58831	23	250	2
6896]							
[164 120700	0	5	1	1 55323	17	1500	1
7598]							
[36 123759	0	4	1	1 50572	23	2500	5
7865]							
[247 118934	0	5	1	1 66222	37	2250	17
7354]							
[275 121639	0	5	1	1 67286	7	0	0
7288]							
[115 124198	0	5	1	1 66330	23	4833	10
7710]							
[86 61990	0	1	1	1 1625	6	0	0 6833]
[254 62882	0	1	1	1 1925	8	0	0 7375]
[181 60230	0	1	1	1 500	1	0	0 7544]
[321 65316	0	1	1	1 3917	10	0	0 7254]
[333 65573	0	1	1	1 3875	23	0	0 7130]
[24 66275	0	1	1	1 2533	11	150	1 6884]
[193 72173	0	1	1	1 2954	13	232	1 7509]
[226 75039	0	1	1	1 2750	10	250	1 7435]
[38 68666	0	1	1	1 350	2	350	2 7861]
[168 69882	0	1	1	1 180	6	0	0 7582]
[265 70730	0	1	1	1 1319	12	0	0 7316]
[91 62553	0	2	1	1 7672	16	700	1 6819]

[291 62998	0	2	1	1	11326	24	0	0	7253]
[43 60313	0	1	1	1	110000	26	3250	9	7829]
[99 57346	0	1	1	1	5560	23	0	0	6766]
[221 59797	0	1	1	1	7069	4	0	0	7444]
[116 56906	0	1	1	1	2015	13	0	0	7707]
[232 53715	0	2	1	1	7164	15	300	2	7416]
[323 52060	0	1	1	1	6500	3	0	0	7162]
[278 51250	0	1	1	1	0	0	0	0	7283]
[363 51088	0	1	1	1	0	0	0	0	7057]
[209 51468	0	1	1	1	2278	9	0	0	7460]
[349 54005	0	1	1	1	1500	5	0	0	7099]
[371 53985	0	1	1	1	1000	1	0	0	7029]
[197 52738	2398	1	1	1	0	0	0	0	7498]
[163 69110	0	3	1	1	18500	19	0	0	7598]
[390 67086	0	3	1	1	18157	16	0	0	6974]
[102 64824	0	3	1	1	20184	13	0	0	6770]
[118 67018	0	3	1	1	23260	25	0	0	7698]
[330 73849	0	2	1	1	11793	21	0	0	7250]
[48 36924	0	1	1	1	5900	6	300	2	6879]
[268 35065	0	1	1	1	5400	5	1400	3	7311]
[58 36298	0	1	1	1	3100	5	600	3	6865]
[11 40091	0	2	1	1	7278	10	0	0	6959]
[89 40284	0	1	1	1	7719	5	0	0	6827]
[369 38227	0	2	1	1	8612	13	0	0	7036]
[391 35969	0	2	1	1	11665	21	0	0	6977]
[105 35119	0	1	1	1	0	0	0	0	6749]
[342 34956	0	1	1	1	0	0	0	0	7172]
[107 34327	0	1	1	1	0	0	0	0	6745]
[270 35900	0	1	1	1	0	0	0	0	7304]
[303 35945	0	1	1	1	0	0	0	0	7220]
[174 35820	0	1	1	1	0	0	0	0	7575]
[41 34616	0	1	1	1	1750	4	500	1	7850]
[55 38348	0	1	1	1	0	0	0	0	6861]
[347 38718	0	1	1	1	0	0	0	0	7314]
[143 39628	0	1	1	1	0	0	0	0	6711]
[208 37348	0	1	1	1	0	0	0	0	7472]
[145 31769	0	1	1	1	450	2	450	2	6707]
[186 32948	0	1	1	1	50	1	50	1	7569]
[215 31588	0	1	1	1	2947	8	0	0	7463]
[14 43097	0	1	1	1	3258	6	0	0	6918]
[271 43088	0	1	1	1	4000	1	0	0	7345]
[3 41354	0	1	1	1	4123	4	0	0	7034]
[110 43498	0	1	1	1	0	0	0	0	6759]
[373 44457	0	1	1	1	225	3	0	0	7026]
[35 43832	0	1	1	1	0	0	0	0	7865]
[338 42363	0	1	1	1	0	0	0	0	7120]
[313 46430	0	1	1	1	2100	4	500	1	7194]
[343 46963	0	1	1	1	1500	5	0	0	7117]
[240 46184	0	1	1	1	3125	12	500	1	7388]
[92 48433	0	1	1	1	3500	4	500	1	6807]
[335 47958	0	1	1	1	0	0	0	0	7130]
[162 46823	0	1	1	1	5860	12	0	0	7613]
[13 43382	0	2	1	1	11150	20	0	0	6924]
[160 44182	0	3	1	1	12313	12	0	0	7619]
[257 45225	0	3	1	1	14711	15	0	0	7339]
[147 41547	0	3	1	1	16857	17	0	0	6697]
[359 41279	0	3	1	1	14462	20	1650	6	7064]
[183 48145	0	1	1	1	10735	27	0	0	7540]
[267 47557	0	2	1	1	10267	18	200	3	7345]
[131 48333	0	2	1	1	10123	14	0	0	6737]
[71 47457	0	3	1	1	12621	16	0	0	7766]
[328 50325	0	3	1	1	11930	14	0	0	7155]
[29 59763	0	3	1	1	33772	20	100	1	7907]
[300 58139	500	4	1	1	38408	16	150	1	7227]
[39 59990	0	3	1	1	27878	17	0	0	7854]
[316 61455	0	4	1	1	28765	17	0	0	7183]

[66	53914	0	3	1	1	33767	45	5550	29	6826]
[49	70312	0	3	1	1	34678	24	500	1	6875]
[350	66863	0	4	1	1	36532	9	0	0	7099]
[344	55788	0	3	1	1	24831	12	0	0	7314]
[397	54842	0	3	1	1	20792	15	0	0	8296]
[393	53790	0	4	1	1	28717	19	0	0	8296]
[126	47258	0	3	1	1	24768	19	0	0	7659]
[188	49885	0	3	1	1	23640	17	0	0	7533]
[59	38077	0	3	1	1	34024	8	0	0	6837]
[204	41385	0	4	1	1	26948	14	0	0	7481]
[285	44873	0	4	1	1	38740	18	850	2	7267]
[293	45586	0	4	1	1	37459	15	0	0	7246]
[81	49238	0	4	1	1	38037	18	0	0	7801]
[157	48707	0	4	1	1	37655	14	0	0	7631]
[17	51890	0	4	1	1	48963	16	0	0	6910]
[272	58703	0	4	1	1	46473	34	950	4	7296]
[150	50593	0	5	1	1	58208	17	1500	2	6689]
[346	13812	0	2	1	1	15149	39	0	0	7110]
[379	15113	0	2	1	1	10574	19	0	0	7017]
[238	11449	0	2	1	1	9057	13	0	0	7397]
[398	11189	0	1	2	1	9044	20	0	0	8296]
[187	8933	0	2	1	1	8860	14	0	0	7533]
[202	8669	0	1	3	1	11397	16	0	0	7486]
[305	7061	0	1	1	1	7100	12	0	0	7218]
[217	8298	0	3	1	1	20087	15	0	0	7498]
[220	1344	0	1	1	1	1134	9	0	0	7498]
[282	2003	0	1	1	1	980	2	0	0	7274]
[401	1612	0	1	1	1	955	9	0	0	8296]
[52	1300	0	1	1	1	370	1	0	0	6868]
[40	2176	0	1	1	1	0	0	0	0	7847]
[94	2000	0	1	1	1	2000	4	2000	4	6809]
[245	2075	0	2	1	1	5002	14	0	0	7374]
[85	8454	0	1	1	1	498	9	0	0	6821]
[104	8053	0	1	1	1	0	0	0	0	6750]
[28	8828	0	1	1	1	0	0	0	0	7914]
[32	10021	0	1	1	1	0	0	0	0	7879]
[63	10120	0	1	1	1	0	0	0	0	6864]
[67	9375	0	1	1	1	1750	7	0	0	6826]
[144	6112	0	1	1	1	0	0	0	0	6709]
[213	6854	0	1	1	1	0	0	0	0	7467]
[182	6663	0	1	1	1	1125	7	750	4	7575]
[218	10097	0	1	1	1	2700	1	2700	1	7498]
[237	24051	0	2	1	1	8764	13	0	0	7395]
[360	25313	0	2	1	1	7491	13	0	0	7064]
[20	23354	0	3	1	1	10447	5	0	0	6896]
[365	22080	0	1	1	1	12200	47	1950	9	7050]
[180	28086	1745	2	1	1	7368	12	0	0	7549]
[255	30015	0	1	1	1	5875	32	0	0	7341]
[106	28667	0	1	1	1	3861	7	2050	5	6749]
[198	24523	0	1	1	1	100	1	0	0	7493]
[383	24598	0	1	1	1	0	0	0	0	7040]
[152	24868	0	1	1	1	205	5	0	0	6689]
[87	24093	0	1	1	1	1750	7	0	0	6819]
[101	22324	0	1	1	1	1375	4	0	0	6753]
[381	22380	0	1	1	1	300	2	200	1	7010]
[37	23193	0	1	1	1	650	4	550	3	7861]
[1	28143	0	1	1	1	174	1	0	0	7000]
[304	28218	0	1	1	1	750	4	750	4	7232]
[405	26720	0	1	1	1	913	3	0	0	8296]
[8	20856	0	1	1	1	5250	4	250	1	6938]
[236	22074	0	1	1	1	6039	15	0	0	7404]
[30	19221	0	1	1	1	4655	8	500	1	7896]
[211	23184	1250	1	1	1	3694	12	1150	3	7467]
[23	20584	0	1	1	1	3450	11	3450	11	6884]
[243	15525	0	1	1	1	3853	6	0	0	7406]
[251	15221	0	1	1	1	4425	9	0	0	7361]

[142	15472	0	1	1	1	3500	14	0	0	6715]
[18	13958	0	1	1	1	4291	5	0	0	6905]
[98	16796	0	1	1	1	2875	16	0	0	6773]
[122	16583	0	1	1	1	2625	7	0	0	7682]
[297	12195	0	1	1	1	3875	17	0	0	7229]
[56	14448	0	1	1	1	1625	6	0	0	6856]
[355	15283	0	1	1	1	1250	3	0	0	7132]
[4	14776	0	1	1	1	500	1	0	0	6952]
[372	14414	0	1	1	1	0	0	0	0	7024]
[403	15129	0	1	1	1	0	0	0	0	8296]
[404	15669	0	1	1	1	1205	4	0	0	8296]
[280	12895	0	1	1	1	0	0	0	0	7277]
[358	13680	0	1	1	1	0	0	0	0	7073]
[75	12646	0	1	1	1	631	4	631	4	7787]
[366	11660	0	1	1	1	140	1	0	0	7050]
[15	17648	0	1	1	1	0	0	0	0	6912]
[133	17469	0	1	1	1	0	0	0	0	6730]
[34	18047	0	1	1	1	100	1	0	0	7868]
[2	19244	0	1	1	1	215	2	0	0	6968]
[149	18707	0	1	1	1	100	1	0	0	6693]
[6	16420	0	1	1	1	0	0	0	0	6942]
[298	16622	0	1	1	1	0	0	0	0	7229]
[368	16200	0	1	1	1	160	1	0	0	7043]
[206	16230	0	1	1	1	486	2	0	0	7477]
[320	16928	0	1	1	1	1000	1	0	0	7173]
[399	16999	0	1	1	1	140	1	0	0	8296]
[27	18521	0	1	1	1	1227	2	1227	2	7917]
[277	18263	0	1	1	1	2100	4	1000	2	7285]
[50	17051	0	1	1	1	1150	4	1150	4	6868]
[26	20726	0	1	1	1	1375	4	0	0	7924]
[227	20203	0	1	1	1	750	6	0	0	7467]
[134	19823	0	1	1	1	1095	3	335	1	6730]
[62	19918	0	1	1	1	17601	11	0	0	6863]
[93	20508	0	1	1	1	22250	8	1250	5	6794]
[212	27381	0	3	1	1	18009	18	0	0	7524]
[261	25279	0	2	1	1	14938	26	0	0	7327]
[205	28621	0	2	1	1	13878	20	1400	10	7479]
[108	3734	0	5	1	1	61096	18	150	2	6760]
[249	12526	0	4	1	1	56076	19	3850	11	7375]
[172	21694	0	4	1	1	45230	20	700	2	7575]
[326	20457	0	4	1	1	51399	16	0	0	7193]
[16	28495	0	4	1	1	49442	15	0	0	6912]
[103	25076	1182	5	1	1	57203	14	0	0	6750]
[283	22652	0	5	1	1	57642	14	0	0	7271]
[88	35418	0	5	1	1	58557	18	900	3	6813]
[200	32742	0	5	1	1	61857	26	0	0	7488]
[290	33982	0	5	1	1	68320	17	0	0	7255]
[82	38896	0	5	1	1	76988	16	556	1	7771]
[112	15098	0	4	1	1	32917	26	550	3	6737]
[166	10302	0	4	1	1	30298	19	0	0	7591]
[42	10470	0	4	1	1	38094	26	0	0	7840]
[315	20746	0	4	1	1	37534	17	0	0	7185]
[83	4340	0	1	1	1	32685	5	0	0	7733]
[380	231	0	1	1	1	29900	24	5300	15	7015]
[210	95989	0	5	1	1	92159	30	1329	4	7518]
[273	70354	0	5	1	1	120907	22	0	0	7290]
[242	30962	0	5	1	1	97683	37	6400	18	7360]
[389	28193	0	5	1	1	4103456	32	0	0	6980]
[194	123516	0	5	1	1	240544	31	500	2	7507]
Cluster: 1											
[120	969559	0	1	1	1	2500	3	1000	2	7718]
[154	930410	0	5	1	1	211284	18	2250	6	

```

7640]
Cluster: 2
[ 196 386061      0      5      1      1 107813      24      0      0
7500]
[ 331 388455      0      5      1      1 114329      26    6078      8
7164]
[ 170 402874      0      5      1      1  74800      15      0      0
7582]
[ 263 402312      0      5      1      1  77122      18     250      3
7375]
[ 192 410795      0      5      1      1  73679      31      0      0
7514]
[  64 362642      0      1      1      1  28079       8      0      0
6835]
[ 307 364387      0      4      1      1  28200      27      0      0
7213]
[ 175 352508      0      3      1      1  23740       6    1000      2
7563]
[ 171 370941      0      4      1      1  44615      16      0      0
7577]
[ 311 383030    2998      1      1      1   8001      25    3226     15
7326]
[ 178 451673      0      4      1      1  43533      19     900      4
7575]
[ 388 479989      0      4      1      1  66516      26    1600      6
6980]
[   9 443003      0      3      2      1   1753      43    3850     12
6948]
[ 329 455228      0      1      1      1    258       2     258      2
7141]
[ 260 423540      0      1      1      1   8534      14    2100      9
7375]
[ 248 468175      0      5      1      1 141615      22      0      0
7348]
[ 385 377252      0      5      1      1 230629      30    6393     13
6996]
Cluster: 3
[ 190 707079      0      4      1      1   57173      40    3450     12
7523]
[ 317 714717      0      5      1      1 119162      20    1750      4
7183]
[ 322 766419      0      1      1      1  11398       3     398      1
7162]
[  74 550367      0      3      1      1  12500      13     50      1
7801]
[ 294 568174      0      5      1      1   67121      16    1000      2
7243]
[  44 619393      0      3      1      1  15008      14      0      0
7819]
[  90 609477      0      3      1      1  21422      22    1200      8
6820]
[ 129 602064      0      5      1      1 194753      26    2250     10
7652]
Cluster: 4
[ 224 1302051    2706      5      1      1   90653      32    3050
7  7467]
Cluster: 5
[ 279 1704838      0      1      1      1   17108      32    4823
23  7283]

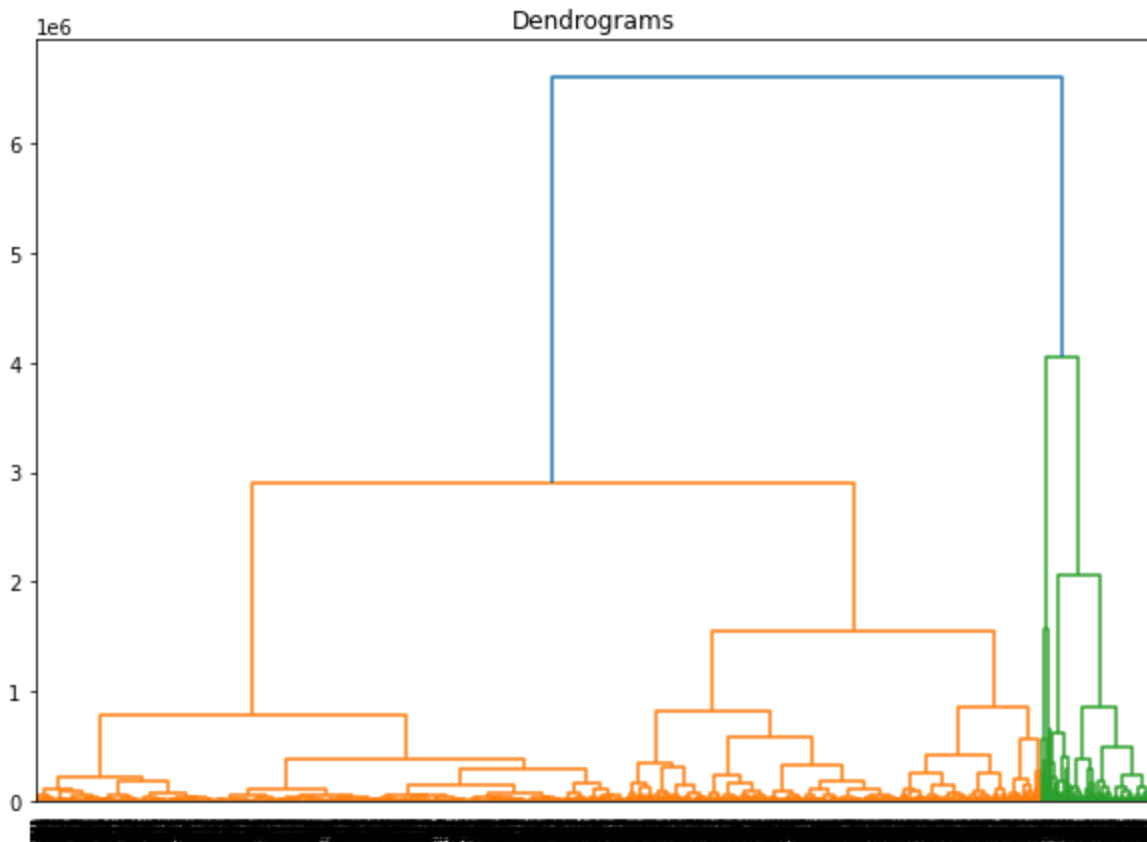
```

The codes above tried to cluster 500 samples out of 4000 observations. Considering about the time complexity and the time for fitting, the total 4000 observations clustering may take over 20 hours. Thus only 500 samples are clustered. The printed result indicated over 380 observations are in the same group, and rest are in the other groups.

The following code use scipy dendrograms to describe all 4000 observations. The outcome indicated that main 2 groups exist (yellow and green). It can also be viewed as 3 groups for 1 yellow group and 2 green groups if we divide the dendrogram from the top part. The yellow group size is obviously larger than green group, but the distances within green group are also larger than that in the yellow group.

```
In [90]: import scipy.cluster.hierarchy as shc

plt.figure(figsize=(10, 7))
plt.title("Dendrograms")
dend = shc.dendrogram(shc.linkage(x, method='ward'))
```



Part 3

DBScan cluster

```
In [82]: from sklearn.preprocessing import PolynomialFeatures
from sklearn.datasets import load_iris
import numpy as np
import scipy as scipy
import matplotlib.pyplot as plt
from sklearn.decomposition import PCA
from sklearn.preprocessing import QuantileTransformer
from sklearn.preprocessing import MinMaxScaler, StandardScaler
from sklearn.metrics.pairwise import euclidean_distances
from sklearn.metrics import pairwise_distances, f1_score, precision_score, recall_score
from sklearn.model_selection import GridSearchCV
from sklearn.base import BaseEstimator, ClassifierMixin

#Custom estimator for gridsearch
class MyClassifier(BaseEstimator, ClassifierMixin):
    def __init__(self, e=0, minp=0):
        self.e = e
        self.minp = minp

    def fit(self, X, Y):
```

```

self.Y=Y
DistanceMatrix = scipy.spatial.distance.squareform(
    scipy.spatial.distance.pdist(X, 'euclidean')
)

core_point_array=np.zeros(150)
cluster_array=np.zeros(150)
PointNeighbors=[]

e=self.e
k=self.minp
w=0
for i in range(len(DistanceMatrix)):
    PointNeighbors=np.where(DistanceMatrix[i]<=e)[0]
    if len(PointNeighbors)>=k:
        core_point_array[i]=1
        if cluster_array[i]==0:
            cluster_array[i]=w
            w=w+1
        for x in range(len(PointNeighbors)):
            #print(cluster_array[PointNeighbors[x]])
            if cluster_array[PointNeighbors[x]]==0:
                cluster_array[PointNeighbors[x]]=cluster_array[i]

for x in range(len(cluster_array)):
    cluster_array[x]=cluster_array[x]-1

self.cluster_array=cluster_array
return cluster_array

def predict(self, X):
    # Some code
    return self.cluster_array

def score(self, X, Y):
    dt=f1_score(self.Y,self.cluster_array,average='weighted')
    print('Accuracy -'+str(dt))
    return (dt)

def DBSCAN(normalised_distance,e,k):
    DistanceMatrix = scipy.spatial.distance.squareform(
        scipy.spatial.distance.pdist(normalised_distance, 'euclidean')
    )

    core_point_array=np.zeros(150)
    cluster_array=np.zeros(150)
    PointNeighbors=[]
    #e=0.3
    #k=18
    w=0
    for i in range(len(DistanceMatrix)):
        PointNeighbors=np.where(DistanceMatrix[i]<=e)[0]
        if len(PointNeighbors)>=k:
            core_point_array[i]=1
            if cluster_array[i]==0:
                cluster_array[i]=w
                w=w+1

            for x in range(len(PointNeighbors)):
                #print(cluster_array[PointNeighbors[x]])
                if cluster_array[PointNeighbors[x]]==0:
                    cluster_array[PointNeighbors[x]]=cluster_array[i]

    for x in range(len(cluster_array)):
        cluster_array[x]=cluster_array[x]-1

    return cluster_array

```

As the maximum of the input data is limited to 150 observations, only 150 samples are in the clustering method.

```
In [109... # Data Transformation
input_data=x[0:150,:]
target_data=y[0:150]

poly = PolynomialFeatures(x.shape[1])
input_data=poly.fit_transform(input_data)
input_data=QuantileTransformer(n_quantiles=40, random_state=0).fit_transform(input_data)

scaler = MinMaxScaler()
scaler.fit(input_data)
normalised_input_data=scaler.transform(input_data)
distan=pairwise_distances(normalised_input_data,metric='euclidean')
scaler.fit(distan)
normalised_distance=scaler.transform(distan)

sscaler = StandardScaler()
sscaler.fit(normalised_distance)
normalised_distance=sscaler.transform(normalised_distance)

pca = PCA(n_components=4)
normalised_distance = pca.fit_transform(normalised_distance)

scaler.fit(normalised_distance)
normalised_distance=scaler.transform(normalised_distance)

print(normalised_distance)
print('normalised_distance')
```

```
[[0.03242504 0.22853599 0.91830355 0.60398854]
 [0.0304068  0.22678839 0.91141609 0.6009346 ]
 [0.02076049 0.21547078 0.78413479 0.54567316]
 [0.0308371  0.22722787 0.91409663 0.60228907]
 [0.82512055 0.10599629 0.41713096 0.58795337]
 [0.05120071 0.24665953 0.9984805  0.64282823]
 [0.02836692 0.21519269 0.34190953 0.3998951 ]
 [0.63166211 0.01960462 0.77770667 0.02376806]
 [0.85160635 0.13728906 0.42577481 0.62379793]
 [0.80953308 0.0755992  0.42695348 0.53053593]
 [0.00988447 0.20435989 0.60260656 0.47430996]
 [0.04543761 0.22861449 0.19154728 0.37608961]
 [0.01021313 0.20335812 0.49778337 0.43725084]
 [0.0125267  0.20840973 0.73058005 0.52188798]
 [0.04967352 0.24549244 0.99899243 0.64302716]
 [0.02297106 0.21186896 0.35016667 0.40152667]
 [0.02436302 0.21310503 0.27777137 0.37970863]
 [0.01701058 0.21314205 0.789012  0.54802037]
 [0.01434843 0.20608872 0.29150032 0.37159794]
 [0.0085804  0.20356964 0.59340805 0.47215976]
 [0.7992833  0.09049183 0.31838748 0.63846855]
 [0.93693096 0.93575606 0.63935866 0.17289195]
 [0.77524853 0.05402645 0.6843151  0.21879489]
 [0.66109086 0.00430227 0.69910841 0.05870322]
 [0.89974214 0.90179343 0.63968149 0.14978099]
 [0.01502627 0.21246474 0.81310982 0.55663605]
 [0.66308595 0.02542746 0.80906088 0.00931682]
 [0.04983473 0.24567171 0.99897218 0.64359863]
 [0.73448606 0.03333863 0.43433133 0.40220695]
 [0.67482309 0.00571715 0.72211874 0.04890616]
 [0.02182084 0.21187457 0.20799605 0.35016727]
 [0.04896094 0.24500756 0.99924256 0.64371927]
 [0.05999874 0.24163225 0.0503064  0.35435547]
```

[0.02533377	0.2229222	0.90537085	0.59781648]
[0.04707116	0.24247002	0.97591099	0.63326002]
[0.883105	0.14119741	0.28309249	0.81787072]
[0.66684838	0.01696991	0.79700417	0.]
[0.6493516	0.01915226	0.78287609	0.0125733]
[0.01075466	0.20427533	0.23843864	0.35313231]
[0.0503538	0.24615065	1.	0.64428731]
[0.66194456	0.00412884	0.76012271	0.00315409]
[0.03343785	0.21927603	0.29271614	0.39644282]
[0.84353333	0.07901159	0.49540065	0.49244738]
[0.04411414	0.22868979	0.12443314	0.34934359]
[0.01653507	0.20887025	0.17148676	0.33543172]
[0.94675388	0.93938938	0.60524214	0.18217446]
[0.00105274	0.19718789	0.42182595	0.40333893]
[0.69817812	0.	0.68933911	0.07436516]
[0.78917087	0.05222979	0.36857551	0.54381799]
[0.70112155	0.02293681	0.78415102	0.03281573]
[0.81558326	0.06103737	0.58687638	0.34546276]
[0.0294401	0.22642745	0.91755061	0.60427735]
[0.05859078	0.24106525	0.02777397	0.34903035]
[0.74651814	0.01179912	0.6006272	0.20612796]
[0.04526078	0.24141726	0.98220802	0.63640228]
[0.00955292	0.2077318	0.77073404	0.53902294]
[0.01782959	0.2101954	0.15284395	0.3344178]
[0.71849495	0.00521035	0.71248087	0.0695665]
[0.00758679	0.20167058	0.32453534	0.37893799]
[0.87824636	0.10841231	0.34410946	0.72493912]
[0.01185769	0.20556422	0.19678467	0.33786277]
[0.0041224	0.19960802	0.51967114	0.4415769]
[0.04689367	0.24346547	0.99943555	0.64414314]
[0.03182356	0.2180797	0.23312543	0.36360112]
[0.01218393	0.20831531	0.72967431	0.52041571]
[0.90733963	0.16102651	0.31315626	0.83317074]
[0.01131577	0.20875646	0.76175693	0.53658302]
[0.02731138	0.2178213	0.08431014	0.32502076]
[0.03078363	0.21965013	0.11185339	0.33177893]
[0.00467816	0.20009657	0.28233833	0.36206024]
[0.93832329	0.94425519	0.6113371	0.18067638]
[0.05498704	0.23873518	0.02995826	0.3421062]
[0.75575067	0.07366287	0.345691	0.53176187]
[0.68269361	0.02271907	0.78601056	0.02656528]
[0.02645251	0.21734946	0.0794531	0.31948243]
[0.02929122	0.21887841	0.09621358	0.32732599]
[0.00692039	0.20083887	0.33952864	0.37718348]
[0.04262682	0.22978105	0.03012767	0.32789134]
[0.03058377	0.21968269	0.10569371	0.33670258]
[0.8077104	0.08995093	0.30473853	0.66851002]
[0.02150378	0.21207083	0.58443968	0.47577002]
[0.7327125	0.021578	0.69297439	0.12222069]
[0.0167829	0.21298485	0.77981782	0.54521279]
[0.00287455	0.20000276	0.6109873	0.47210417]
[0.00307993	0.20133468	0.67316464	0.49799179]
[0.844415	0.10316406	0.31220666	0.72028023]
[0.	0.19737088	0.55101546	0.44939002]
[0.88905518	0.15710097	0.26363014	0.84850915]
[0.766291	0.02157265	0.49890258	0.34716104]
[0.69653146	0.00151957	0.68314894	0.07991844]
[0.78926898	0.04031148	0.56747023	0.32355618]
[0.70380828	0.04026654	0.76729221	0.08826225]
[0.04587195	0.23187578	0.03582328	0.33538037]
[0.78539187	0.03532695	0.56426883	0.30569197]
[0.91962452	0.20921168	0.2109791	0.98312347]
[0.00699331	0.20240407	0.58294796	0.46656627]
[0.00874964	0.20201118	0.38557685	0.39627007]
[0.04377602	0.2305695	0.0336434	0.33093256]
[0.00747024	0.20547695	0.73016505	0.52195243]


```

[0.01022793 0.20454753 0.20714603 0.34246794]
[0.27788864 0.48389013 0.38234559 0.55485783]
[0.0479332 0.24399223 0.9904039 0.64135727]
[0.04614592 0.24181208 0.96998335 0.63323324]
[0.78047376 0.0339028 0.62883182 0.24055282]
[0.04628467 0.24191639 0.96963906 0.6331994 ]
[0.75059014 0.07627755 0.42773364 0.48541153]
[0.9875608 0.98411534 0.55346056 0.29258704]
[0.04713021 0.24216737 0.95968628 0.62922879]
[0.75602593 0.06812276 0.31609274 0.5668996 ]
[0.81072314 0.08119541 0.37743516 0.59489484]
[0.03815143 0.22585626 0.06949052 0.32949808]
[0.86279607 0.10779533 0.40956861 0.61913503]
[0.91393809 0.19689529 0.21764777 0.96366649]
[0.00598847 0.20073993 0.48131971 0.42720304]
[0.80829081 0.07356768 0.40724135 0.52804325]
[0.03561368 0.22360033 0.08380032 0.33436396]
[0.01669096 0.20838736 0.23869745 0.35445697]
[0.7797791 0.07504976 0.50007054 0.40776786]
[0.03394538 0.22301386 0.06649634 0.32488557]
[0.00662552 0.20287416 0.62128045 0.48086847]
[0.01721356 0.21095201 0.68238788 0.50529468]
[0.91725566 0.21207465 0.20314063 0.98963935]
[0.86286128 0.11993897 0.26630116 0.79832357]
[0.02648653 0.216423 0.13670209 0.33964391]
[0.04376504 0.23014104 0.05810683 0.3381127 ]
[0.97274326 0.9640057 0.53232145 0.28531405]
[0.92214367 0.22349898 0.20241136 1. ]
[0.92277561 0.21541705 0.21202201 0.98856881]
[0.00758649 0.20191343 0.30747109 0.37220285]
[0.06618108 0.24787747 0. 0.35225995]
[0.04790341 0.24341611 0.97473762 0.63577112]
[0.65046802 0.01363154 0.75249115 0.03094333]
[0.01490849 0.20621764 0.40342799 0.4041569 ]
[0.83863814 0.07162784 0.46487978 0.50540413]
[0.02788697 0.21780023 0.12014731 0.33264219]
[1. 1. 0.57334309 0.29706439]
[0.04744073 0.23336051 0.03630113 0.33522682]
[0.86461071 0.13965603 0.23369167 0.84510563]
[0.03282145 0.21928594 0.20697292 0.35979961]
[0.01139302 0.20498572 0.54525342 0.45604814]
[0.05000514 0.24403059 0.94790546 0.62564803]
[0.05098444 0.24591176 0.97694631 0.63701888]
[0.66616512 0.02241056 0.75689573 0.04620376]
[0.05700243 0.24075625 0.01697235 0.34360472]
[0.02024975 0.21124779 0.20803346 0.3543154 ]
[0.8179253 0.05790418 0.50178393 0.43028757]
[0.02622036 0.22155571 0.83779511 0.57154136]
[0.8591531 0.12932486 0.26973329 0.79828562]
[0.03016612 0.21954966 0.1101312 0.33426723]
[0.01959586 0.21390538 0.73172371 0.52720163]]
normalised_distance

```

```

In [110... eps_values= np.arange(0.1,0.5 ,0.001)
min_sample_values = np.arange(2,30,1)

params = {
    'e':eps_values,
    'minp':min_sample_values
}
cv = [(slice(None), slice(None))]

```

```

In [111... gs = GridSearchCV(MyClassifier(), param_grid=params, cv=cv)
Y=target_data
gs.fit(normalised_distance,Y)

```

[illegible]

Accuracy	-0.2579045792375929
Accuracy	-0.33086311013245256
Accuracy	-0.33086311013245256
Accuracy	-0.33086311013245256
Accuracy	-0.3241291551835596
Accuracy	-0.3241291551835596
Accuracy	-0.3241291551835596
Accuracy	-0.30355194852672374
Accuracy	-0.30355194852672374
Accuracy	-0.2974698911263128
Accuracy	-0.18593272171253825
Accuracy	-0.1859327217125382
Accuracy	-0.18593272171253825
Accuracy	-0.1737142857142857
Accuracy	-0.1737142857142857
Accuracy	-0.1737142857142857
Accuracy	-0.1737142857142857
Accuracy	-0.1737142857142857
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.2579045792375929
Accuracy	-0.33939105739383135
Accuracy	-0.33939105739383135
Accuracy	-0.33939105739383135
Accuracy	-0.33265710244493835
Accuracy	-0.3241291551835596
Accuracy	-0.3241291551835596
Accuracy	-0.3126120455778622
Accuracy	-0.3126120455778622
Accuracy	-0.3361659302896416
Accuracy	-0.18593272171253825
Accuracy	-0.18593272171253825
Accuracy	-0.18593272171253825
Accuracy	-0.1737142857142857
Accuracy	-0.1737142857142857
Accuracy	-0.1737142857142857
Accuracy	-0.1737142857142857
Accuracy	-0.1737142857142857
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.18934053346424481
Accuracy	-0.2579045792375929
Accuracy	-0.33939105739383135
Accuracy	-0.33939105739383135
Accuracy	-0.33939105739383135
Accuracy	-0.3408936908096572
Accuracy	-0.3241291551835596
Accuracy	-0.3241291551835596
Accuracy	-0.32148339060710196
Accuracy	-0.32148339060710196
Accuracy	-0.32148339060710196

[illegible]

Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.2579045792375929
Accuracy -0.33939105739383135
Accuracy -0.33939105739383135
Accuracy -0.33939105739383135
Accuracy -0.3408936908096572
Accuracy -0.3323657435482784
Accuracy -0.3323657435482784
Accuracy -0.318546882670594
Accuracy -0.32148339060710196
Accuracy -0.32148339060710196
Accuracy -0.17777777777777773
Accuracy -0.18593272171253825
Accuracy -0.18593272171253825
Accuracy -0.1876543209876543
Accuracy -0.1876543209876543
Accuracy -0.1876543209876543
Accuracy -0.1876543209876543
Accuracy -0.1737142857142857
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.2579045792375929
Accuracy -0.33939105739383135
Accuracy -0.33939105739383135
Accuracy -0.33939105739383135
Accuracy -0.3408936908096572
Accuracy -0.3323657435482784
Accuracy -0.3323657435482784
Accuracy -0.3241291551835596
Accuracy -0.32148339060710196
Accuracy -0.32148339060710196
Accuracy -0.17777777777777773
Accuracy -0.18593272171253825
Accuracy -0.18593272171253825
Accuracy -0.1876543209876543
Accuracy -0.1876543209876543
Accuracy -0.1876543209876543
Accuracy -0.1876543209876543
Accuracy -0.1737142857142857
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.18934053346424481
Accuracy -0.26832247106687207
Accuracy -0.3379173207744636

Accuracy	-0.3379173207744636
Accuracy	-0.3379173207744636
Accuracy	-0.33939105739383135
Accuracy	-0.33086311013245256
Accuracy	-0.33086311013245256
Accuracy	-0.3226925409338994
Accuracy	-0.32
Accuracy	-0.32
Accuracy	-0.12657294028722602
Accuracy	-0.18593272171253825
Accuracy	-0.18593272171253825
Accuracy	-0.1876543209876543
Accuracy	-0.1876543209876543
Accuracy	-0.1876543209876543
Accuracy	-0.1876543209876543
Accuracy	-0.1737142857142857
Accuracy	-0.1978684807256236
Accuracy	-0.1978684807256236
Accuracy	-0.1978684807256236
Accuracy	-0.1978684807256236
Accuracy	-0.1978684807256236
Accuracy	-0.1978684807256236
Accuracy	-0.1978684807256236
Accuracy	-0.1978684807256236
Accuracy	-0.1978684807256236
Accuracy	-0.1978684807256236
Accuracy	-0.1978684807256236
Accuracy	-0.19773029211684676
Accuracy	-0.4389059829059829
Accuracy	-0.3379173207744636
Accuracy	-0.3379173207744636
Accuracy	-0.33939105739383135
Accuracy	-0.33939105739383135
Accuracy	-0.33939105739383135
Accuracy	-0.33122048819527816
Accuracy	-0.3285279472613788
Accuracy	-0.3285279472613788
Accuracy	-0.13646578140960164
Accuracy	-0.18593272171253825
Accuracy	-0.18593272171253825
Accuracy	-0.18593272171253825
Accuracy	-0.18593272171253825
Accuracy	-0.1876543209876543
Accuracy	-0.1876543209876543
Accuracy	-0.1876543209876543
Accuracy	-0.1978684807256236
Accuracy	-0.1978684807256236
Accuracy	-0.1978684807256236
Accuracy	-0.1978684807256236
Accuracy	-0.1978684807256236
Accuracy	-0.1978684807256236
Accuracy	-0.1978684807256236
Accuracy	-0.1978684807256236
Accuracy	-0.1978684807256236
Accuracy	-0.1978684807256236
Accuracy	-0.1978684807256236
Accuracy	-0.1978684807256236
Accuracy	-0.19773029211684676
Accuracy	-0.4389059829059829
Accuracy	-0.3379173207744636
Accuracy	-0.3379173207744636
Accuracy	-0.33939105739383135
Accuracy	-0.33939105739383135
Accuracy	-0.33939105739383135
Accuracy	-0.33122048819527816
Accuracy	-0.3285279472613788
Accuracy	-0.3285279472613788
Accuracy	-0.13646578140960164
Accuracy	-0.18593272171253825

[illegible]

Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.36649808429118774
Accuracy -0.39802540138988735
Accuracy -0.3379173207744636
Accuracy -0.33939105739383135
Accuracy -0.33939105739383135
Accuracy -0.33939105739383135
Accuracy -0.33939105739383135
Accuracy -0.3242557823129252
Accuracy -0.3285279472613788
Accuracy -0.3285279472613788
Accuracy -0.18593272171253825
Accuracy -0.18593272171253825
Accuracy -0.18593272171253825
Accuracy -0.18593272171253825
Accuracy -0.1876543209876543
Accuracy -0.1876543209876543
Accuracy -0.1876543209876543
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.3655959595959596
Accuracy -0.32471537807986406
Accuracy -0.3379173207744636
Accuracy -0.33939105739383135
Accuracy -0.33939105739383135
Accuracy -0.33939105739383135
Accuracy -0.33939105739383135
Accuracy -0.33122048819527816
Accuracy -0.3242557823129252
Accuracy -0.3285279472613788
Accuracy -0.17777777777777773
Accuracy -0.1842424242424242
Accuracy -0.18593272171253825
Accuracy -0.18593272171253825
Accuracy -0.18593272171253825
Accuracy -0.18593272171253825
Accuracy -0.1876543209876543
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.1978684807256236
Accuracy -0.3466285325161729
Accuracy -0.33491966817179464
Accuracy -0.27481158755637086

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Accuracy -0.433785663877407
Accuracy -0.433785663877407
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.45127355233303734
Accuracy -0.494761255942108
Accuracy -0.494761255942108
Accuracy -0.45558358817533134
Accuracy -0.433785663877407
Accuracy -0.433785663877407
Accuracy -0.34899494897433253
Accuracy -0.34899494897433253
Accuracy -0.24501560283687945
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.3481990068165695
Accuracy -0.19370546405806716
Accuracy -0.24734577412543515
Accuracy -0.19751412429378534
Accuracy -0.19218390804597704
Accuracy -0.19218390804597704
Accuracy -0.1850434782608696
Accuracy -0.433785663877407
Accuracy -0.433785663877407
Accuracy -0.433785663877407
Accuracy -0.433785663877407
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.45127355233303734
Accuracy -0.494761255942108
Accuracy -0.494761255942108
Accuracy -0.45558358817533134
Accuracy -0.433785663877407
Accuracy -0.433785663877407
Accuracy -0.40606338615512927
Accuracy -0.3589724770642202
Accuracy -0.29120000000000007
Accuracy -0.2706525836622924
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.3481990068165695
Accuracy -0.19370546405806716
Accuracy -0.24734577412543515
Accuracy -0.19751412429378534
Accuracy -0.19920227920227923
Accuracy -0.19920227920227923
Accuracy -0.19920227920227923
Accuracy -0.433785663877407
Accuracy -0.433785663877407
Accuracy -0.433785663877407
Accuracy -0.433785663877407
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.45127355233303734
Accuracy -0.494761255942108
Accuracy -0.494761255942108
Accuracy -0.45558358817533134
Accuracy -0.45127355233303734
Accuracy -0.433785663877407

Accuracy -0.40606338615512927
Accuracy -0.3578882279799711
Accuracy -0.290115750915751
Accuracy -0.3140225470322558
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.3481990068165695
Accuracy -0.19370546405806716
Accuracy -0.24734577412543515
Accuracy -0.19751412429378534
Accuracy -0.19751412429378534
Accuracy -0.19920227920227923
Accuracy -0.19920227920227923
Accuracy -0.433785663877407
Accuracy -0.433785663877407
Accuracy -0.433785663877407
Accuracy -0.433785663877407
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.45127355233303734
Accuracy -0.494761255942108
Accuracy -0.494761255942108
Accuracy -0.45558358817533134
Accuracy -0.45127355233303734
Accuracy -0.433785663877407
Accuracy -0.40606338615512927
Accuracy -0.3578882279799711
Accuracy -0.3578882279799711
Accuracy -0.3140225470322558
Accuracy -0.4296358456358456
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.3481990068165695
Accuracy -0.3481990068165695
Accuracy -0.24734577412543515
Accuracy -0.19751412429378534
Accuracy -0.19751412429378534
Accuracy -0.19920227920227923
Accuracy -0.19920227920227923
Accuracy -0.433785663877407
Accuracy -0.433785663877407
Accuracy -0.433785663877407
Accuracy -0.433785663877407
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.45127355233303734
Accuracy -0.494761255942108
Accuracy -0.494761255942108
Accuracy -0.45558358817533134
Accuracy -0.45127355233303734
Accuracy -0.433785663877407
Accuracy -0.4155118029069168
Accuracy -0.3568275495279883
Accuracy -0.3568275495279883
Accuracy -0.312961868580273
Accuracy -0.3736727362814319
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914
Accuracy -0.35314491449144914

Accuracy -0.3481990068165695
Accuracy -0.24734577412543515
Accuracy -0.19751412429378534
Accuracy -0.19751412429378534
Accuracy -0.19920227920227923
Accuracy -0.19920227920227923
Accuracy -0.433785663877407
Accuracy -0.433785663877407
Accuracy -0.433785663877407
Accuracy -0.433785663877407
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.45127355233303734
Accuracy -0.494761255942108
Accuracy -0.494761255942108
Accuracy -0.46374223651439195
Accuracy -0.46374223651439195
Accuracy -0.433785663877407
Accuracy -0.4155118029069168
Accuracy -0.36639900036171125
Accuracy -0.36639900036171125
Accuracy -0.3225333194139959
Accuracy -0.38324418711515484
Accuracy -0.30675325597075837
Accuracy -0.3664863965388135
Accuracy -0.3664863965388135
Accuracy -0.3664863965388135
Accuracy -0.3664863965388135
Accuracy -0.3664863965388135
Accuracy -0.3622692614088995
Accuracy -0.24734577412543515
Accuracy -0.19751412429378534
Accuracy -0.19751412429378534
Accuracy -0.19920227920227923
Accuracy -0.19920227920227923
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.45127355233303734
Accuracy -0.494761255942108
Accuracy -0.494761255942108
Accuracy -0.46374223651439195
Accuracy -0.46374223651439195
Accuracy -0.433785663877407
Accuracy -0.424750254841998
Accuracy -0.36639900036171125
Accuracy -0.36639900036171125
Accuracy -0.3225333194139959
Accuracy -0.38324418711515484
Accuracy -0.30675325597075837
Accuracy -0.3664863965388135
Accuracy -0.3664863965388135
Accuracy -0.3664863965388135
Accuracy -0.3664863965388135
Accuracy -0.3664863965388135
Accuracy -0.3622692614088995
Accuracy -0.24734577412543515
Accuracy -0.19751412429378534
Accuracy -0.19751412429378534
Accuracy -0.19920227920227923
Accuracy -0.19920227920227923
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636

Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.45127355233303734
Accuracy -0.494761255942108
Accuracy -0.494761255942108
Accuracy -0.46374223651439195
Accuracy -0.46374223651439195
Accuracy -0.4426246509772636
Accuracy -0.433785663877407
Accuracy -0.36639900036171125
Accuracy -0.36639900036171125
Accuracy -0.36639900036171125
Accuracy -0.38324418711515484
Accuracy -0.30675325597075837
Accuracy -0.3664863965388135
Accuracy -0.3664863965388135
Accuracy -0.3664863965388135
Accuracy -0.3664863965388135
Accuracy -0.3664863965388135
Accuracy -0.3622692614088995
Accuracy -0.24734577412543515
Accuracy -0.19751412429378534
Accuracy -0.19751412429378534
Accuracy -0.19920227920227923
Accuracy -0.19920227920227923
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.45127355233303734
Accuracy -0.494761255942108
Accuracy -0.494761255942108
Accuracy -0.46374223651439195
Accuracy -0.46374223651439195
Accuracy -0.4426246509772636
Accuracy -0.433785663877407
Accuracy -0.3757668033053549
Accuracy -0.3757668033053549
Accuracy -0.3757668033053549
Accuracy -0.3926119900587985
Accuracy -0.31612105891440206
Accuracy -0.31612105891440206
Accuracy -0.3664863965388135
Accuracy -0.3664863965388135
Accuracy -0.3664863965388135
Accuracy -0.3622692614088995
Accuracy -0.3622692614088995
Accuracy -0.19751412429378534
Accuracy -0.19751412429378534
Accuracy -0.19920227920227923
Accuracy -0.19920227920227923
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.45127355233303734
Accuracy -0.494761255942108
Accuracy -0.494761255942108
Accuracy -0.46374223651439195
Accuracy -0.46374223651439195
Accuracy -0.4426246509772636
Accuracy -0.433785663877407
Accuracy -0.3757668033053549

Accuracy -0.3757668033053549
Accuracy -0.3757668033053549
Accuracy -0.3926119900587985
Accuracy -0.3240884438881936
Accuracy -0.3240884438881936
Accuracy -0.37445378151260506
Accuracy -0.37445378151260506
Accuracy -0.37445378151260506
Accuracy -0.37445378151260506
Accuracy -0.370236646382691
Accuracy -0.24684745762711865
Accuracy -0.19751412429378534
Accuracy -0.19920227920227923
Accuracy -0.19920227920227923
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.45127355233303734
Accuracy -0.48953139962314274
Accuracy -0.494761255942108
Accuracy -0.4795650696568128
Accuracy -0.4795650696568128
Accuracy -0.4426246509772636
Accuracy -0.433785663877407
Accuracy -0.3757668033053549
Accuracy -0.3757668033053549
Accuracy -0.3757668033053549
Accuracy -0.3926119900587985
Accuracy -0.3240884438881936
Accuracy -0.3240884438881936
Accuracy -0.37445378151260506
Accuracy -0.37445378151260506
Accuracy -0.37445378151260506
Accuracy -0.37445378151260506
Accuracy -0.370236646382691
Accuracy -0.24684745762711865
Accuracy -0.19751412429378534
Accuracy -0.19751412429378534
Accuracy -0.19751412429378534
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.45127355233303734
Accuracy -0.48953139962314274
Accuracy -0.494761255942108
Accuracy -0.4795650696568128
Accuracy -0.4795650696568128
Accuracy -0.4426246509772636
Accuracy -0.433785663877407
Accuracy -0.3757668033053549
Accuracy -0.3757668033053549
Accuracy -0.3757668033053549
Accuracy -0.3926119900587985
Accuracy -0.3240884438881936
Accuracy -0.3240884438881936
Accuracy -0.3812941176470588
Accuracy -0.37445378151260506
Accuracy -0.37445378151260506
Accuracy -0.37445378151260506
Accuracy -0.370236646382691

Accuracy -0.24684745762711865
Accuracy -0.19751412429378534
Accuracy -0.19751412429378534
Accuracy -0.19751412429378534
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.45127355233303734
Accuracy -0.48953139962314274
Accuracy -0.494761255942108
Accuracy -0.4795650696568128
Accuracy -0.4795650696568128
Accuracy -0.4426246509772636
Accuracy -0.433785663877407
Accuracy -0.3757668033053549
Accuracy -0.3757668033053549
Accuracy -0.3757668033053549
Accuracy -0.40498480243161095
Accuracy -0.3240884438881936
Accuracy -0.3240884438881936
Accuracy -0.3812941176470588
Accuracy -0.37445378151260506
Accuracy -0.37445378151260506
Accuracy -0.37445378151260506
Accuracy -0.370236646382691
Accuracy -0.24684745762711865
Accuracy -0.19751412429378534
Accuracy -0.19751412429378534
Accuracy -0.19751412429378534
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.45127355233303734
Accuracy -0.48953139962314274
Accuracy -0.494761255942108
Accuracy -0.494761255942108
Accuracy -0.4795650696568128
Accuracy -0.4680251086431675
Accuracy -0.4426246509772636
Accuracy -0.3757668033053549
Accuracy -0.3757668033053549
Accuracy -0.3757668033053549
Accuracy -0.3976354445435623
Accuracy -0.32287306501547985
Accuracy -0.32287306501547985
Accuracy -0.3812941176470588
Accuracy -0.37445378151260506
Accuracy -0.37445378151260506
Accuracy -0.37445378151260506
Accuracy -0.370236646382691
Accuracy -0.24684745762711865
Accuracy -0.19751412429378534
Accuracy -0.19751412429378534
Accuracy -0.19751412429378534
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636
Accuracy -0.4426246509772636

Accuracy -0.45127355233303734
Accuracy -0.48953139962314274
Accuracy -0.49978801104480275
Accuracy -0.49978801104480275
Accuracy -0.49978801104480275
Accuracy -0.48953139962314274
Accuracy -0.45127355233303734
Accuracy -0.43297247706422015
Accuracy -0.43297247706422015
Accuracy -0.43297247706422015
Accuracy -0.45462949992996216
Accuracy -0.37986712040187975
Accuracy -0.37986712040187975
Accuracy -0.3812941176470588
Accuracy -0.37445378151260506
Accuracy -0.37445378151260506
Accuracy -0.37445378151260506
Accuracy -0.3688553712083124
Accuracy -0.25370308123249297
Accuracy -0.20436974789915965
Accuracy -0.20436974789915965
Accuracy -0.19751412429378534
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.48953139962314274
Accuracy -0.49978801104480275
Accuracy -0.49978801104480275
Accuracy -0.49978801104480275
Accuracy -0.48953139962314274
Accuracy -0.45127355233303734
Accuracy -0.43297247706422015
Accuracy -0.43297247706422015
Accuracy -0.43297247706422015
Accuracy -0.45462949992996216
Accuracy -0.37986712040187975
Accuracy -0.37986712040187975
Accuracy -0.41889719626168226
Accuracy -0.41205686012722853
Accuracy -0.37445378151260506
Accuracy -0.37445378151260506
Accuracy -0.3688553712083124
Accuracy -0.25370308123249297
Accuracy -0.20436974789915965
Accuracy -0.20436974789915965
Accuracy -0.19751412429378534
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.4872391437308869
Accuracy -0.49978801104480275
Accuracy -0.49978801104480275
Accuracy -0.49978801104480275
Accuracy -0.48953139962314274
Accuracy -0.45127355233303734
Accuracy -0.43297247706422015
Accuracy -0.43297247706422015
Accuracy -0.43297247706422015

Accuracy -0.45462949992996216
Accuracy -0.37986712040187975
Accuracy -0.37986712040187975
Accuracy -0.4174701990165031
Accuracy -0.41889719626168226
Accuracy -0.37445378151260506
Accuracy -0.37445378151260506
Accuracy -0.3688553712083124
Accuracy -0.25370308123249297
Accuracy -0.20436974789915965
Accuracy -0.20436974789915965
Accuracy -0.19751412429378534
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.494761255942108
Accuracy -0.5046232707150138
Accuracy -0.49978801104480275
Accuracy -0.49978801104480275
Accuracy -0.48953139962314274
Accuracy -0.45127355233303734
Accuracy -0.43297247706422015
Accuracy -0.43297247706422015
Accuracy -0.43297247706422015
Accuracy -0.45462949992996216
Accuracy -0.37986712040187975
Accuracy -0.37986712040187975
Accuracy -0.4174701990165031
Accuracy -0.41889719626168226
Accuracy -0.37445378151260506
Accuracy -0.37445378151260506
Accuracy -0.3688553712083124
Accuracy -0.25370308123249297
Accuracy -0.20436974789915965
Accuracy -0.20436974789915965
Accuracy -0.19751412429378534
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.494761255942108
Accuracy -0.5046232707150138
Accuracy -0.49978801104480275
Accuracy -0.49978801104480275
Accuracy -0.48953139962314274
Accuracy -0.45127355233303734
Accuracy -0.43297247706422015
Accuracy -0.43297247706422015
Accuracy -0.43297247706422015
Accuracy -0.45462949992996216
Accuracy -0.37986712040187975
Accuracy -0.38767979887132575
Accuracy -0.4174701990165031
Accuracy -0.41889719626168226
Accuracy -0.41205686012722853
Accuracy -0.3822664599820511
Accuracy -0.3766680496777584
Accuracy -0.25321463238760844
Accuracy -0.25321463238760844

[illegible]

Accuracy -0.5046232707150138
Accuracy -0.49978801104480275
Accuracy -0.49978801104480275
Accuracy -0.48953139962314274
Accuracy -0.45127355233303734
Accuracy -0.43297247706422015
Accuracy -0.43297247706422015
Accuracy -0.43297247706422015
Accuracy -0.4469870627247676
Accuracy -0.37986712040187975
Accuracy -0.38767979887132575
Accuracy -0.4174701990165031
Accuracy -0.41889719626168226
Accuracy -0.41205686012722853
Accuracy -0.41205686012722853
Accuracy -0.3822664599820511
Accuracy -0.3766680496777584
Accuracy -0.20148083242059148
Accuracy -0.20436974789915965
Accuracy -0.20436974789915965
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.4924626731426516
Accuracy -0.5046232707150138
Accuracy -0.49978801104480275
Accuracy -0.49978801104480275
Accuracy -0.48953139962314274
Accuracy -0.45127355233303734
Accuracy -0.4315454798190411
Accuracy -0.4315454798190411
Accuracy -0.4315454798190411
Accuracy -0.4469870627247676
Accuracy -0.37986712040187975
Accuracy -0.38767979887132575
Accuracy -0.4174701990165031
Accuracy -0.4174701990165031
Accuracy -0.4174701990165031
Accuracy -0.41205686012722853
Accuracy -0.3822664599820511
Accuracy -0.3766680496777584
Accuracy -0.20148083242059148
Accuracy -0.20436974789915965
Accuracy -0.20436974789915965
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.4924626731426516
Accuracy -0.5046232707150138
Accuracy -0.49978801104480275
Accuracy -0.49978801104480275
Accuracy -0.48953139962314274
Accuracy -0.45127355233303734
Accuracy -0.4315454798190411
Accuracy -0.4315454798190411
Accuracy -0.4315454798190411
Accuracy -0.4469870627247676
Accuracy -0.37986712040187975

Accuracy -0.38767979887132575
Accuracy -0.4174701990165031
Accuracy -0.4174701990165031
Accuracy -0.4174701990165031
Accuracy -0.41205686012722853
Accuracy -0.3822664599820511
Accuracy -0.3766680496777584
Accuracy -0.20148083242059148
Accuracy -0.20436974789915965
Accuracy -0.20436974789915965
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.49978801104480275
Accuracy -0.5116265651145346
Accuracy -0.5116265651145346
Accuracy -0.49978801104480275
Accuracy -0.48953139962314274
Accuracy -0.45127355233303734
Accuracy -0.4315454798190411
Accuracy -0.4315454798190411
Accuracy -0.4315454798190411
Accuracy -0.45462949992996216
Accuracy -0.37986712040187975
Accuracy -0.38767979887132575
Accuracy -0.38767979887132575
Accuracy -0.4174701990165031
Accuracy -0.4174701990165031
Accuracy -0.41205686012722853
Accuracy -0.3822664599820511
Accuracy -0.3766680496777584
Accuracy -0.20148083242059148
Accuracy -0.20436974789915965
Accuracy -0.20436974789915965
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.45763914373088693
Accuracy -0.5069724770642203
Accuracy -0.5116265651145346
Accuracy -0.5116265651145346
Accuracy -0.49978801104480275
Accuracy -0.48953139962314274
Accuracy -0.45127355233303734
Accuracy -0.45127355233303734
Accuracy -0.4315454798190411
Accuracy -0.4315454798190411
Accuracy -0.45462949992996216
Accuracy -0.37986712040187975
Accuracy -0.38767979887132575
Accuracy -0.38767979887132575
Accuracy -0.4174701990165031
Accuracy -0.4174701990165031
Accuracy -0.41205686012722853
Accuracy -0.3822664599820511
Accuracy -0.3822664599820511
Accuracy -0.20148083242059148
Accuracy -0.20436974789915965
Accuracy -0.20436974789915965

Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.45763914373088693
Accuracy -0.5140200961118392
Accuracy -0.518498956814999
Accuracy -0.518498956814999
Accuracy -0.518498956814999
Accuracy -0.4872391437308869
Accuracy -0.44924198061031945
Accuracy -0.45127355233303734
Accuracy -0.4315454798190411
Accuracy -0.4315454798190411
Accuracy -0.45462949992996216
Accuracy -0.38767979887132575
Accuracy -0.38767979887132575
Accuracy -0.38767979887132575
Accuracy -0.4174701990165031
Accuracy -0.4174701990165031
Accuracy -0.41205686012722853
Accuracy -0.3822664599820511
Accuracy -0.3822664599820511
Accuracy -0.25321463238760844
Accuracy -0.20436974789915965
Accuracy -0.20436974789915965
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.45763914373088693
Accuracy -0.5140200961118392
Accuracy -0.518498956814999
Accuracy -0.518498956814999
Accuracy -0.518498956814999
Accuracy -0.4872391437308869
Accuracy -0.4872391437308869
Accuracy -0.45127355233303734
Accuracy -0.4315454798190411
Accuracy -0.4315454798190411
Accuracy -0.45462949992996216
Accuracy -0.38767979887132575
Accuracy -0.38767979887132575
Accuracy -0.38767979887132575
Accuracy -0.4174701990165031
Accuracy -0.4174701990165031
Accuracy -0.41205686012722853
Accuracy -0.3822664599820511
Accuracy -0.3822664599820511
Accuracy -0.25321463238760844
Accuracy -0.20436974789915965
Accuracy -0.20436974789915965
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.45763914373088693
Accuracy -0.5140200961118392
Accuracy -0.518498956814999
Accuracy -0.518498956814999

Accuracy -0.518498956814999
Accuracy -0.4872391437308869
Accuracy -0.4872391437308869
Accuracy -0.45127355233303734
Accuracy -0.4315454798190411
Accuracy -0.4315454798190411
Accuracy -0.45462949992996216
Accuracy -0.38767979887132575
Accuracy -0.38767979887132575
Accuracy -0.38767979887132575
Accuracy -0.4174701990165031
Accuracy -0.4174701990165031
Accuracy -0.41205686012722853
Accuracy -0.3822664599820511
Accuracy -0.3822664599820511
Accuracy -0.25321463238760844
Accuracy -0.20436974789915965
Accuracy -0.20436974789915965
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.5140200961118392
Accuracy -0.518498956814999
Accuracy -0.518498956814999
Accuracy -0.518498956814999
Accuracy -0.4872391437308869
Accuracy -0.4872391437308869
Accuracy -0.45127355233303734
Accuracy -0.4315454798190411
Accuracy -0.4315454798190411
Accuracy -0.45462949992996216
Accuracy -0.38767979887132575
Accuracy -0.38767979887132575
Accuracy -0.38767979887132575
Accuracy -0.4174701990165031
Accuracy -0.4174701990165031
Accuracy -0.41205686012722853
Accuracy -0.3822664599820511
Accuracy -0.3822664599820511
Accuracy -0.3766680496777584
Accuracy -0.20436974789915965
Accuracy -0.20436974789915965
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.5069724770642203
Accuracy -0.518498956814999
Accuracy -0.518498956814999
Accuracy -0.518498956814999
Accuracy -0.5021358757570307
Accuracy -0.4872391437308869
Accuracy -0.45127355233303734
Accuracy -0.45127355233303734
Accuracy -0.4315454798190411
Accuracy -0.45462949992996216
Accuracy -0.38767979887132575
Accuracy -0.45462949992996216
Accuracy -0.38767979887132575

Accuracy -0.4245730027548209
Accuracy -0.4245730027548209
Accuracy -0.4191596638655462
Accuracy -0.4191596638655462
Accuracy -0.3822664599820511
Accuracy -0.3766680496777584
Accuracy -0.20436974789915965
Accuracy -0.20436974789915965
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.5069724770642203
Accuracy -0.5069724770642203
Accuracy -0.518498956814999
Accuracy -0.518498956814999
Accuracy -0.5021358757570307
Accuracy -0.4872391437308869
Accuracy -0.45127355233303734
Accuracy -0.45127355233303734
Accuracy -0.4315454798190411
Accuracy -0.4315454798190411
Accuracy -0.38767979887132575
Accuracy -0.45462949992996216
Accuracy -0.38767979887132575
Accuracy -0.4245730027548209
Accuracy -0.4245730027548209
Accuracy -0.4191596638655462
Accuracy -0.4191596638655462
Accuracy -0.4191596638655462
Accuracy -0.3766680496777584
Accuracy -0.20436974789915965
Accuracy -0.20436974789915965
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.5069724770642203
Accuracy -0.5069724770642203
Accuracy -0.518498956814999
Accuracy -0.518498956814999
Accuracy -0.5021358757570307
Accuracy -0.4872391437308869
Accuracy -0.45127355233303734
Accuracy -0.45127355233303734
Accuracy -0.4315454798190411
Accuracy -0.4315454798190411
Accuracy -0.38767979887132575
Accuracy -0.45462949992996216
Accuracy -0.38767979887132575
Accuracy -0.4245730027548209
Accuracy -0.4245730027548209
Accuracy -0.4191596638655462
Accuracy -0.4191596638655462
Accuracy -0.4191596638655462
Accuracy -0.3766680496777584
Accuracy -0.20436974789915965
Accuracy -0.20436974789915965
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945

Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.5069724770642203
Accuracy -0.5069724770642203
Accuracy -0.518498956814999
Accuracy -0.5021358757570307
Accuracy -0.4872391437308869
Accuracy -0.45127355233303734
Accuracy -0.45127355233303734
Accuracy -0.4315454798190411
Accuracy -0.4315454798190411
Accuracy -0.38767979887132575
Accuracy -0.45462949992996216
Accuracy -0.45462949992996216
Accuracy -0.4245730027548209
Accuracy -0.4245730027548209
Accuracy -0.4191596638655462
Accuracy -0.4191596638655462
Accuracy -0.4191596638655462
Accuracy -0.3822664599820511
Accuracy -0.20266666666666666
Accuracy -0.20266666666666666
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.5069724770642203
Accuracy -0.5069724770642203
Accuracy -0.518498956814999
Accuracy -0.5021358757570307
Accuracy -0.4872391437308869
Accuracy -0.45127355233303734
Accuracy -0.45127355233303734
Accuracy -0.4315454798190411
Accuracy -0.4315454798190411
Accuracy -0.38767979887132575
Accuracy -0.45462949992996216
Accuracy -0.45462949992996216
Accuracy -0.4245730027548209
Accuracy -0.4245730027548209
Accuracy -0.4245730027548209
Accuracy -0.4191596638655462
Accuracy -0.4191596638655462
Accuracy -0.3822664599820511
Accuracy -0.20936639118457304
Accuracy -0.20266666666666666
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.5069724770642203
Accuracy -0.5069724770642203
Accuracy -0.518498956814999
Accuracy -0.5021358757570307

Accuracy -0.4872391437308869
Accuracy -0.45127355233303734
Accuracy -0.45127355233303734
Accuracy -0.4315454798190411
Accuracy -0.4315454798190411
Accuracy -0.38767979887132575
Accuracy -0.45462949992996216
Accuracy -0.45462949992996216
Accuracy -0.4245730027548209
Accuracy -0.4245730027548209
Accuracy -0.4245730027548209
Accuracy -0.4191596638655462
Accuracy -0.4191596638655462
Accuracy -0.3822664599820511
Accuracy -0.20936639118457304
Accuracy -0.20266666666666666
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.44924198061031945
Accuracy -0.5069724770642203
Accuracy -0.5069724770642203
Accuracy -0.518498956814999
Accuracy -0.5021358757570307
Accuracy -0.4872391437308869
Accuracy -0.45127355233303734
Accuracy -0.45127355233303734
Accuracy -0.4315454798190411
Accuracy -0.4315454798190411
Accuracy -0.38627619502360866
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.38627619502360866
Accuracy -0.4231693989071038
Accuracy -0.4231693989071038
Accuracy -0.4191596638655462
Accuracy -0.4191596638655462
Accuracy -0.3822664599820511
Accuracy -0.20936639118457304
Accuracy -0.20266666666666666
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.5046060606060606
Accuracy -0.5161325403568394
Accuracy -0.49742159458664315
Accuracy -0.4826258625862586
Accuracy -0.4693679653679653
Accuracy -0.44890713587487774
Accuracy -0.4291790633608815
Accuracy -0.4291790633608815
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.38627619502360866
Accuracy -0.4231693989071038

Accuracy -0.4231693989071038
Accuracy -0.4191596638655462
Accuracy -0.4191596638655462
Accuracy -0.3822664599820511
Accuracy -0.20936639118457304
Accuracy -0.20266666666666666
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.5046060606060606
Accuracy -0.5161325403568394
Accuracy -0.49742159458664315
Accuracy -0.4826258625862586
Accuracy -0.4693679653679653
Accuracy -0.44890713587487774
Accuracy -0.4291790633608815
Accuracy -0.4291790633608815
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.38627619502360866
Accuracy -0.4231693989071038
Accuracy -0.4231693989071038
Accuracy -0.4191596638655462
Accuracy -0.4191596638655462
Accuracy -0.3822664599820511
Accuracy -0.20936639118457304
Accuracy -0.20936639118457304
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.5046060606060606
Accuracy -0.5161325403568394
Accuracy -0.49742159458664315
Accuracy -0.4826258625862586
Accuracy -0.4693679653679653
Accuracy -0.44488676236044655
Accuracy -0.4291790633608815
Accuracy -0.4291790633608815
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.38627619502360866
Accuracy -0.4231693989071038
Accuracy -0.4231693989071038
Accuracy -0.4191596638655462
Accuracy -0.4191596638655462
Accuracy -0.3822664599820511
Accuracy -0.20936639118457304
Accuracy -0.20936639118457304
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984

Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.5116536796536796
Accuracy -0.5228776655443321
Accuracy -0.5228776655443321
Accuracy -0.4826258625862586
Accuracy -0.4693679653679653
Accuracy -0.44488676236044655
Accuracy -0.4291790633608815
Accuracy -0.4291790633608815
Accuracy -0.4482131837307152
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.4231693989071038
Accuracy -0.4231693989071038
Accuracy -0.4191596638655462
Accuracy -0.4191596638655462
Accuracy -0.4191596638655462
Accuracy -0.24695369277187462
Accuracy -0.24695369277187462
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.44687556415215984
Accuracy -0.5116536796536796
Accuracy -0.5228776655443321
Accuracy -0.5228776655443321
Accuracy -0.4826258625862586
Accuracy -0.4826258625862586
Accuracy -0.44488676236044655
Accuracy -0.4291790633608815
Accuracy -0.4291790633608815
Accuracy -0.4291790633608815
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.4231693989071038
Accuracy -0.4231693989071038
Accuracy -0.4191596638655462
Accuracy -0.4191596638655462
Accuracy -0.4191596638655462
Accuracy -0.24695369277187462
Accuracy -0.24695369277187462
Accuracy -0.4552727272727272
Accuracy -0.4552727272727272
Accuracy -0.4552727272727272
Accuracy -0.4552727272727272
Accuracy -0.4552727272727272
Accuracy -0.4552727272727272
Accuracy -0.4552727272727272
Accuracy -0.4552727272727272
Accuracy -0.4552727272727272
Accuracy -0.5116536796536796
Accuracy -0.5228776655443321
Accuracy -0.5228776655443321
Accuracy -0.4826258625862586
Accuracy -0.4826258625862586

Accuracy -0.44488676236044655
Accuracy -0.4291790633608815
Accuracy -0.4291790633608815
Accuracy -0.4291790633608815
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.4231693989071038
Accuracy -0.4231693989071038
Accuracy -0.4191596638655462
Accuracy -0.4191596638655462
Accuracy -0.4191596638655462
Accuracy -0.2559072716877177
Accuracy -0.2559072716877177
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.5116536796536796
Accuracy -0.5228776655443321
Accuracy -0.5228776655443321
Accuracy -0.4826258625862586
Accuracy -0.4826258625862586
Accuracy -0.44488676236044655
Accuracy -0.4291790633608815
Accuracy -0.4291790633608815
Accuracy -0.4291790633608815
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.430141875971324
Accuracy -0.430141875971324
Accuracy -0.4315454798190411
Accuracy -0.42613214092976637
Accuracy -0.42613214092976637
Accuracy -0.2554723101876883
Accuracy -0.2554723101876883
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.5116536796536796
Accuracy -0.5228776655443321
Accuracy -0.5228776655443321
Accuracy -0.49742159458664315
Accuracy -0.4826258625862586
Accuracy -0.44488676236044655
Accuracy -0.4291790633608815
Accuracy -0.4291790633608815
Accuracy -0.4291790633608815
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.430141875971324
Accuracy -0.430141875971324
Accuracy -0.4315454798190411

Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.5116536796536796
Accuracy -0.5228776655443321
Accuracy -0.5228776655443321
Accuracy -0.49742159458664315
Accuracy -0.4826258625862586
Accuracy -0.4693679653679653
Accuracy -0.44890713587487774
Accuracy -0.4277754595131645
Accuracy -0.4277754595131645
Accuracy -0.45788526434195725
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.43535957383841367
Accuracy -0.430141875971324
Accuracy -0.42613214092976637
Accuracy -0.42613214092976637
Accuracy -0.2554723101876883
Accuracy -0.2554723101876883
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.5116536796536796
Accuracy -0.5228776655443321
Accuracy -0.5228776655443321
Accuracy -0.49742159458664315
Accuracy -0.4826258625862586
Accuracy -0.4693679653679653
Accuracy -0.44890713587487774
Accuracy -0.4277754595131645
Accuracy -0.4277754595131645
Accuracy -0.45788526434195725
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.43535957383841367
Accuracy -0.430141875971324
Accuracy -0.42613214092976637
Accuracy -0.42613214092976637
Accuracy -0.2554723101876883
Accuracy -0.2554723101876883
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.4634949494949495
Accuracy -0.5116536796536796
Accuracy -0.5228776655443321
Accuracy -0.5228776655443321
Accuracy -0.49742159458664315
Accuracy -0.4826258625862586
Accuracy -0.4693679653679653
Accuracy -0.44890713587487774

Accuracy -0.4277754595131645
Accuracy -0.4277754595131645
Accuracy -0.45788526434195725
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.43535957383841367
Accuracy -0.430141875971324
Accuracy -0.42613214092976637
Accuracy -0.42613214092976637
Accuracy -0.2554723101876883
Accuracy -0.2554723101876883
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.5184590304590305
Accuracy -0.5296830163496831
Accuracy -0.5296830163496831
Accuracy -0.5042269453919941
Accuracy -0.4894312133916095
Accuracy -0.47617331617331626
Accuracy -0.4557124866802287
Accuracy -0.4345808103185153
Accuracy -0.4345808103185153
Accuracy -0.4345808103185153
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.43535957383841367
Accuracy -0.430141875971324
Accuracy -0.43297247706422015
Accuracy -0.42613214092976637
Accuracy -0.2554723101876883
Accuracy -0.2554723101876883
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.5184590304590305
Accuracy -0.5296830163496831
Accuracy -0.5296830163496831
Accuracy -0.5042269453919941
Accuracy -0.4894312133916095
Accuracy -0.47617331617331626
Accuracy -0.4557124866802287
Accuracy -0.4345808103185153
Accuracy -0.4345808103185153
Accuracy -0.4345808103185153
Accuracy -0.45788526434195725
Accuracy -0.45788526434195725
Accuracy -0.4469870627247676
Accuracy -0.43535957383841367
Accuracy -0.430141875971324
Accuracy -0.43297247706422015
Accuracy -0.42613214092976637

Accuracy -0.4703003003003004
Accuracy -0.5184590304590305
Accuracy -0.5296830163496831
Accuracy -0.5296830163496831
Accuracy -0.5042269453919941
Accuracy -0.4894312133916095
Accuracy -0.47617331617331626
Accuracy -0.4557124866802287
Accuracy -0.4345808103185153
Accuracy -0.4345808103185153
Accuracy -0.4345808103185153
Accuracy -0.45788526434195725
Accuracy -0.45788526434195725
Accuracy -0.4469870627247676
Accuracy -0.4482131837307152
Accuracy -0.430141875971324
Accuracy -0.430141875971324
Accuracy -0.43297247706422015
Accuracy -0.25375619222715895
Accuracy -0.25375619222715895
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.4703003003003004
Accuracy -0.5184590304590305
Accuracy -0.5296830163496831
Accuracy -0.5296830163496831
Accuracy -0.5042269453919941
Accuracy -0.4894312133916095
Accuracy -0.47617331617331626
Accuracy -0.4536809149575108
Accuracy -0.4345808103185153
Accuracy -0.4345808103185153
Accuracy -0.4345808103185153
Accuracy -0.4345808103185153
Accuracy -0.44974474474474485
Accuracy -0.44974474474474485
Accuracy -0.4482131837307152
Accuracy -0.430141875971324
Accuracy -0.430141875971324
Accuracy -0.43297247706422015
Accuracy -0.25375619222715895
Accuracy -0.25375619222715895
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.5308652828652828
Accuracy -0.5420892687559353
Accuracy -0.5420892687559353
Accuracy -0.5166331977982463
Accuracy -0.5093078598960952
Accuracy -0.4964102564102564
Accuracy -0.47242877492877483
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676

Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.44974474474474485
Accuracy -0.4482131837307152
Accuracy -0.430141875971324
Accuracy -0.430141875971324
Accuracy -0.43297247706422015
Accuracy -0.25375619222715895
Accuracy -0.25375619222715895
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.5420892687559353
Accuracy -0.5420892687559353
Accuracy -0.5166331977982463
Accuracy -0.5166331977982463
Accuracy -0.4964102564102564
Accuracy -0.47242877492877483
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.44974474474474485
Accuracy -0.44974474474474485
Accuracy -0.430141875971324
Accuracy -0.430141875971324
Accuracy -0.43297247706422015
Accuracy -0.25375619222715895
Accuracy -0.25375619222715895
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.4827065527065527
Accuracy -0.5420892687559353
Accuracy -0.5420892687559353
Accuracy -0.5166331977982463
Accuracy -0.5166331977982463
Accuracy -0.4964102564102564
Accuracy -0.47242877492877483
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.4469870627247676
Accuracy -0.44974474474474485
Accuracy -0.44974474474474485
Accuracy -0.43535957383841367
Accuracy -0.430141875971324
Accuracy -0.43297247706422015
Accuracy -0.43297247706422015
Accuracy -0.25375619222715895

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Out[111]:

In [112...

```
print(gs.best_params_)
```

```
para=gs.best_params_
```

```
{'e': 0.46600000000000003, 'minp': 24}
```

The outcome of best estimation for MyClassifier e is 0.46, and the minimum p is 24.

In [113...

```
e=para['e']
k=para['minp']
cluster_array=DBSCAN(normalised_distance,e,k)

print(target_data)
print(cluster_array.astype(int))

print('precision_score- '+str(precision_score(target_data,cluster_array,average='weighte
print('recall_score- '+str(recall_score(target_data,cluster_array,average='weighted',lab

plt.subplot(2, 1, 1)
plt.scatter(normalised_distance[:,0], normalised_distance[:,1],c=cluster_array, cmap='Pa
plt.title("custom DBSCAN predicted cluster outputs")

plt.subplot(2, 1, 2)
plt.scatter(normalised_distance[:,0], normalised_distance[:,1],c=target_data, cmap='Pair
plt.title("Actual target outputs")

plt.tight_layout()
plt.show()
```

```
0      0
```

```
1      0
```

```
2      0
```

```
3      0
```

```
4      1
```

```
..
```

```
145    1
```

```
146    0
```

```
147    1
```

```
148    0
```

```
149    0
```

```
Name: Award?, Length: 150, dtype: int64
```

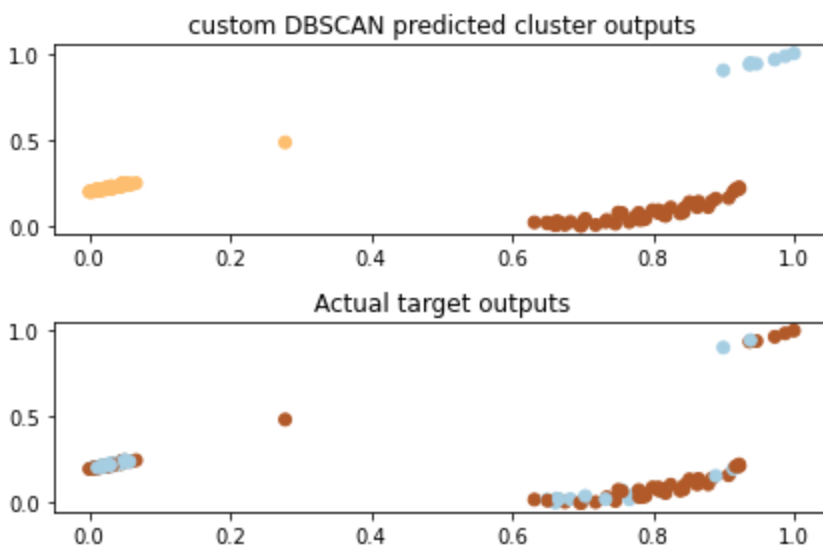
```
[ 0  0  0  0  1  0  0  1  1  1  0  0  0  0  0  0  0  0  0  1 -1  1  1
 -1  0  1  0  1  1  0  0  0  0  0  1  1  1  0  0  1  0  1  0  0 -1  0  1
  1  1  1  0  0  1  0  0  0  1  0  1  0  0  0  0  1  0  0  0  0 -1  0
  1  1  0  0  0  0  0  1  0  1  0  0  0  1  0  1  1  1  1  1  0  1  1  0
  0  0  0  0  0  0  0  1  0  1 -1  0  1  1  0  1  1  0  1  0  0  1  0  0
  0  1  1  0  0 -1  1  1  0  0  0  1  0  1  0 -1  0  1  0  0  0  0  1  0
  0  1  0  1  0  0]
```

```
precision_score- 0.6858588399720474
```

```
recall_score- 0.6466666666666666
```

```
c:\Users\youyu\AppData\Local\Programs\Python\Python38\lib\site-packages\sklearn\metrics
\_classification.py:1248: UndefinedMetricWarning: Recall is ill-defined and being set to
0.0 in labels with no true samples. Use `zero_division` parameter to control this behavi
or.
```

```
_warn_prf(average, modifier, msg_start, len(result))
```



The prediction output group number doesn't match with the actual target output. It misclassified the left corner group observations. This could happen because of the wrongly normalized data. The original data did not get properly processed with correct grouping methods.

The overall precision score is 0.686 and recall score is 0.647, which is moderate.

Part 4

Draw the inferences from the clusters obtained.

Draw the inferences based on each feature, no obvious inferences are observed based on the plots.

```
In [190... X = x[0:150]
cluster = cluster_array.astype(int)
cg = centroids[4][0:150]
df1 = pd.DataFrame(dict(y=y[0:150], label=cluster))
df2 = pd.DataFrame(X)
df = pd.concat([df1, df2], axis=1)
colors = {0:'blue', 1:'orange', -1:'green', 3:'black', 4:'red', 5:'pink', 6:'purple'}

for i in range(1, X.shape[1]):
    fig, ax = plt.subplots(figsize=(8,6))
    df['a'] = X[:,i]
    grouped = df.groupby('label')
    for key, group in grouped:
        group.plot(ax=ax, kind='scatter', x='a', y='y', label=key, color=colors[key])
    ax.scatter(cg[:, 0], cg[:, 1], marker='*', s=150, c='#ff2222')
    plt.xlabel(f'X_{i}')
    plt.ylabel('y')
    plt.show()
```

