Problem IV: GMM

1452669, Yang LI, April 8

Data Preprocessing

As the Gaussian Mixture Model works for *N* random variables that are observed, each distributed according to a mixture of *K* components, with the components belonging to the same Gaussian distributions. I use standardization to preprocess the data. Same as in Problem III, Z-score standardization has mean 0 and standard deviation 1.

Gaussian Mixture Model

Here introduce the EM algorithms.

1. Initial the parameter.
2. E-step: calculate every possibility of data from submodel.

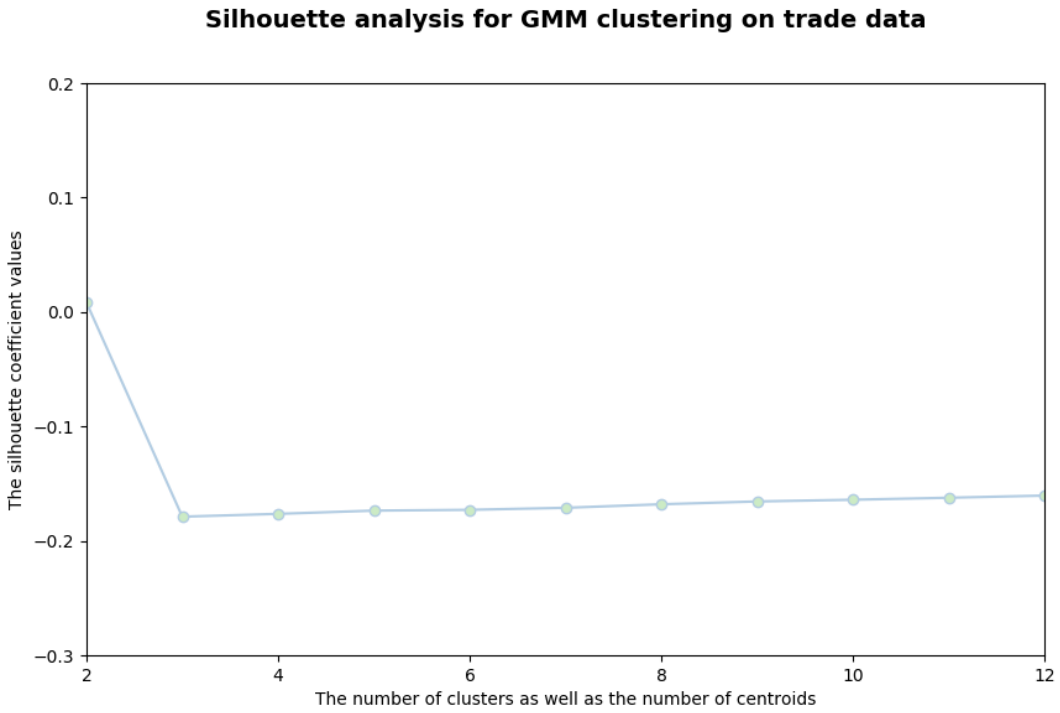
   $P(j|x_i) = \pi_j \phi(x_i;\theta_j)/f_k(x_i)$
3. M-step: calculate new round model parameter.

   $\pi_j = \Sigma P(j|x_i)/n$

   $m_j = \Sigma P(j|x_i)x_i/(n\pi_j)$

   $C_j = \Sigma P(j|x_i)(x_i - m_j)(x_i - m_j)^T(n\pi_j)$

Following figure shows the silhouette score with number of clusters, detailed data in the fiollowing table.



number of clusters	silhouette score
2	0.008170664390796289
3	-0.17888166675017764
4	-0.17640449270412467
5	-0.17358065858275754
6	-0.17289004898253502
7	-0.17111480858695352
8	-0.1680206784271202
9	-0.16557530656981118
10	-0.16408006833335792
11	-0.16232669961509005
12	-0.1604467452945715

The accuracy please see the screenshot of running result below.

Performance

Time & Space Complexity in Theory (EM algorithm)

- time complexity: $O(nki)$ where i stands for the number of iterators. in theoretical is **infinite**

Benchmark in Practice

```
Timer unit: 1e-06 s

Total time: 30.8365 s
File: /Users/Yang/Developer/420235DataMining/hw1/q4/gmm.py
Function: gmm at line 10

Line #      Hits         Time  Per Hit   % Time  Line Contents
==============================================================
    10                                           @profile
    11                                           def gmm(df, eps, random_vip, knns):
    12         1        295.0    295.0      0.0      silhouette_avgs = []
    13         1          2.0      2.0      0.0      ks = []
    14         1          1.0      1.0      0.0      hits = []
    15         1          1.0      1.0      0.0      gmm_labels = []
    16         1      15867.0  15867.0      0.1      X = StandardScaler().fit_transform(df.T)
    17        11         16.0      1.5      0.0      for k in range(2, 12):
    18        10         12.0      1.2      0.0          clusterer = GaussianMixture(n_components=k, covariance_type='tied',
    19        10   26690285.0 2669028.5     86.6                                      max_iter=20, random_state=0).fit(X)
    20        10    3246283.0 324628.3     10.5          cluster_labels = clusterer.predict(X)
    21        10     174095.0  17409.5      0.6          silhouette_avg = silhouette_score(X, cluster_labels)
    22        10         38.0      3.8      0.0          silhouette_avgs.append(silhouette_avg)
    23        10         23.0      2.3      0.0          logging.info(
    24        10         11.0      1.1      0.0              "For n_clusters = %s ,the average silhouette_score in GMM is : %s." % (
    25        10       1980.0    198.0      0.0              k, silhouette_avg))
    26        10         23.0      2.3      0.0          ks.append(k)
    27        10         13.0      1.3      0.0          gmm_labels.append(cluster_labels)
    28
    29        10         13.0      1.3      0.0          hit = 0
    30        10        208.0     20.8      0.0          no = cluster_labels[df.columns.get_loc(random_vip)]
    31        60         79.0      1.3      0.0          for neighbor in knns:
    32        50        261.0      5.2      0.0              if cluster_labels[df.columns.get_loc(neighbor)] == no:
    33        50         62.0      1.2      0.0                  hit += 1
    34                                                       else:
    35                                                           logging.debug(
    36                                                               "GMM: vipno: {} is not in the same cluster.".format(
    37                                                                   neighbor))
    38        10         14.0      1.4      0.0          logging.info(
    39        10         13.0      1.3      0.0              "For k = {} in kNN, there has {} in the same cluster in GMM.".format(
    40        10       1238.0    123.8      0.0                  len(knns), hit))
    41        10         20.0      2.0      0.0          hits.append(hit)
    42
    43                                               # plot_kmeans_clusterno(11, silhouette_avgs)
    44
    45                                               # Compare with Kmeans
    46         1     142201.0 142201.0      0.5      kmeans_labels = KMeans(n_clusters=2, random_state=10).fit_predict(X)
    47         1          4.0      4.0      0.0      gmm_label = gmm_labels[ks.index(2)]
    48         1          1.0      1.0      0.0      hit = 0
    49       299        313.0      1.0      0.0      for index, kmeans_label in enumerate(kmeans_labels):
    50       298        969.0      3.3      0.0          if kmeans_label == gmm_label[index]:
    51         2          2.0      1.0      0.0              hit += 1
    52         1          2.0      2.0      0.0      logging.info(
    53         1        204.0    204.0      0.0          "The accuracy of KMeans is {}".format(hit / len(kmeans_labels)))
    54
    55                                               # Compare with DBSCAN
    56         1     561180.0 561180.0      1.8      dbscan_labels = DBSCAN(eps, min_samples=10).fit_predict(X)
    57         1         43.0     43.0      0.0      gmm_label = gmm_labels[ks.index(len(set(dbscan_labels)))]
    58         1         19.0     19.0      0.0      dbscan_labels[dbscan_labels == -1] = 1
    59         1          1.0      1.0      0.0      hit = 0
    60       299        259.0      0.9      0.0      for index, dbscan_label in enumerate(dbscan_labels):
    61       298        276.0      0.9      0.0          if dbscan_label == gmm_label[index]:
    62         2          3.0      1.5      0.0              hit += 1
    63         1          2.0      2.0      0.0      logging.info(
    64         1        160.0    160.0      0.0          "The accuracy of DBSCAN is {}".format(hit / len(dbscan_labels)))
```

Screenshot

```
/usr/local/bin/python3.6 /Users/Yang/Developer/420235DataMining/hw1/main.py
INFO:root:DataFrame shape: (2635, 298)
<class 'pandas.core.frame.DataFrame'>
Index: 2635 entries, 10000004 to 40000700
Columns: 298 entries, 13205496418 to 6222021615015662822
dtypes: float64(298)
memory usage: 6.0+ MB
INFO:root:random vipno: 1591015587123
INFO:root:vipno in ranked order using kNN(k = 5):
INFO:root:1595151614620
INFO:root:1595150991142
INFO:root:1595132332932
INFO:root:2900000549289
INFO:root:1595151110818
INFO:root:For n_clusters = 2 ,the average silhouette_score in K-means is : 0.9898142095571695.
```

```
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in K-means.
INFO:root:For n_clusters = 3 ,the average silhouette_score in K-means is : 0.9921749295338916.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in K-means.
INFO:root:For n_clusters = 4 ,the average silhouette_score in K-means is : 0.9532367586659553.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in K-means.
INFO:root:For n_clusters = 5 ,the average silhouette_score in K-means is : 0.9354794005524527.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in K-means.
INFO:root:For n_clusters = 6 ,the average silhouette_score in K-means is : 0.9121777723154195.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in K-means.
INFO:root:For n_clusters = 7 ,the average silhouette_score in K-means is : 0.8925583134911551.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in K-means.
INFO:root:For n_clusters = 8 ,the average silhouette_score in K-means is : 0.7841269502455196.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in K-means.
INFO:root:For n_clusters = 9 ,the average silhouette_score in K-means is : 0.8054482659092047.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in K-means.
INFO:root:For n_clusters = 10 ,the average silhouette_score in K-means is : 0.638603594695715.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in K-means.
INFO:root:For n_clusters = 11 ,the average silhouette_score in K-means is : 0.6411969263383369.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in K-means.
INFO:root:For n_clusters = 12 ,the average silhouette_score in K-means is : 0.6363110531546554.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in K-means.
INFO:root:For n_clusters = 13 ,the average silhouette_score in K-means is : 0.6173050922283423.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in K-means.
INFO:root:DBSCAN: eps = 10
INFO:root:For n_clusters = 1 The average silhouette_score in DBSCAN is : -0.31939086805466427.
INFO:root:For k = 5 in kNN, there has 3 in the same cluster in DBSCAN.
INFO:root:DBSCAN: eps = 20
INFO:root:For n_clusters = 1 The average silhouette_score in DBSCAN is : -0.1942397862954418.
INFO:root:For k = 5 in kNN, there has 3 in the same cluster in DBSCAN.
INFO:root:DBSCAN: eps = 30
INFO:root:For n_clusters = 1 The average silhouette_score in DBSCAN is : -0.029129862530624537.
INFO:root:For k = 5 in kNN, there has 1 in the same cluster in DBSCAN.
INFO:root:DBSCAN: eps = 40
INFO:root:For n_clusters = 1 The average silhouette_score in DBSCAN is : 0.08708130699171691.
INFO:root:For k = 5 in kNN, there has 1 in the same cluster in DBSCAN.
INFO:root:DBSCAN: eps = 50
INFO:root:For n_clusters = 1 The average silhouette_score in DBSCAN is : 0.20268451888941394.
INFO:root:For k = 5 in kNN, there has 1 in the same cluster in DBSCAN.
INFO:root:DBSCAN: eps = 60
INFO:root:For n_clusters = 1 The average silhouette_score in DBSCAN is : 0.2775394743004492.
INFO:root:For k = 5 in kNN, there has 0 in the same cluster in DBSCAN.
INFO:root:DBSCAN: eps = 70
INFO:root:For n_clusters = 1 The average silhouette_score in DBSCAN is : 0.3387778176593896.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in DBSCAN.
INFO:root:DBSCAN: eps = 80
INFO:root:For n_clusters = 1 The average silhouette_score in DBSCAN is : 0.4067160039619981.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in DBSCAN.
INFO:root:DBSCAN: eps = 90
INFO:root:For n_clusters = 1 The average silhouette_score in DBSCAN is : 0.43392130605710816.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in DBSCAN.
INFO:root:DBSCAN: eps = 100
INFO:root:For n_clusters = 1 The average silhouette_score in DBSCAN is : 0.4576647790706404.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in DBSCAN.
INFO:root:DBSCAN: eps = 110
INFO:root:For n_clusters = 1 The average silhouette_score in DBSCAN is : 0.4883338136442235.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in DBSCAN.
INFO:root:DBSCAN: eps = 120
INFO:root:For n_clusters = 1 The average silhouette_score in DBSCAN is : 0.5001686210200447.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in DBSCAN.
INFO:root:DBSCAN: eps = 130
INFO:root:For n_clusters = 1 The average silhouette_score in DBSCAN is : 0.5243956411873608.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in DBSCAN.
INFO:root:For n_clusters = 2 ,the average silhouette_score in GMM is : 0.008170664390796289.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in GMM.
INFO:root:For n_clusters = 3 ,the average silhouette_score in GMM is : -0.17888166675017764.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in GMM.
INFO:root:For n_clusters = 4 ,the average silhouette_score in GMM is : -0.17640449270412467.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in GMM.
INFO:root:For n_clusters = 5 ,the average silhouette_score in GMM is : -0.17358065858275754.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in GMM.
INFO:root:For n_clusters = 6 ,the average silhouette_score in GMM is : -0.17289004898253502.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in GMM.
INFO:root:For n_clusters = 7 ,the average silhouette_score in GMM is : -0.17111480858695352.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in GMM.
INFO:root:For n_clusters = 8 ,the average silhouette_score in GMM is : -0.1680206784271202.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in GMM.
INFO:root:For n_clusters = 9 ,the average silhouette_score in GMM is : -0.16557530656981118.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in GMM.
INFO:root:For n_clusters = 10 ,the average silhouette_score in GMM is : -0.16408006833335792.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in GMM.
INFO:root:For n_clusters = 11 ,the average silhouette_score in GMM is : -0.16232669961509005.
INFO:root:For k = 5 in kNN, there has 5 in the same cluster in GMM.
```

```
INFO:root:The accuracy of KMeans is 0.996711409395973154
INFO:root:The accuracy of DBSCAN is 0.974524338384257921

Process finished with exit code 0
```