SPLEX TME 2

Clustering

The goal of the TME is to learn how to use some popular clustering methods (unsupervised learning), and how to interpret the results.

We will use the *scikit-learn Python* library `http://scikit-learn.org` which is already installed on the computers.

Data (simulated data sets + data sets of TME 1)

We explore two data sets downloadable from the Machine Learning Repository (`http://archive.ics.uci.edu/ml/index.php`)

- Breast Cancer Wisconsin (Diagnostic) Data Set (`https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)`)
- Mice Protein Expression Data Set (`https://archive.ics.uci.edu/ml/datasets/Mice+Protein+Expression`)

Libraries

You will need to load the following packages:

```
import matplotlib.pyplot as plt
from sklearn import cluster
from sklearn.cluster import KMeans
from sklearn import metrics
from sklearn.cluster import AgglomerativeClustering
from sklearn.datasets import make_classification
from sklearn.datasets import make_blobs
from sklearn.datasets import make_moons
```

Analysis

Before running analysis on the Breast and Mice data sets, we will do analysis on three simulated data sets to better understand what different clustering methods do, and why they produce different clustering. Generate and visualize the artificial data as follows:

```
# First simulated data set
plt.title("Two informative features, one cluster per class", fontsize='small')
X1, Y1 = make_classification(n_samples=200, n_features=2, n_redundant=0, n_informative=2,
                             n_clusters_per_class=1)
plt.scatter(X1[:, 0], X1[:, 1], marker='o', c=Y1,s=25, edgecolor='k')

# Second simulated data set
plt.title("Three blobs", fontsize='small')
X2, Y2 = make_blobs(n_samples=200, n_features=2, centers=3)
plt.scatter(X2[:, 0], X2[:, 1], marker='o', c=Y2, s=25, edgecolor='k')

# Third simulated data set
plt.title("Non-linearly separated data sets", fontsize='small')
X3, Y3 = make_moons(n_samples=200, shuffle=True, noise=None, random_state=None)
plt.scatter(X3[:, 0], X3[:, 1], marker='o', c=Y3, s=25, edgecolor='k')
```

Apply the following clustering methods to the three simulated data sets.

Clustering Methods

1. *K-means*

   `http://scikit-learn.org/stable/modules/clustering.html#k-means`

   An example of k-means clustering (where k is the number of clusters you want to produce, and X is the data matrix):

   ```
   km = KMeans(n_clusters=k, init='k-means++', max_iter=100, n_init=1)
   km.fit(X)
   ```

   You can also visualize the clustering (and compare it to the true repartition):

   ```
   plt.scatter(X[:, 0], X[:, 1], s=10, c=km.labels_)
   ```

2. *Hierarchical clustering*

   `http://scikit-learn.org/stable/modules/clustering.html#hierarchical-clustering`

   An example of hierarchical clustering (where k is the number of clusters you want to produce, and X is the data matrix):

   ```
   for linkage in ('ward', 'average', 'complete'):
       clustering = AgglomerativeClustering(linkage=linkage, n_clusters=k)
       clustering.fit(X)
   ```

3. *Spectral clustering*

   `http://scikit-learn.org/stable/modules/clustering.html#spectral-clustering`

   An example of spectral clustering (where k is the number of clusters you want to produce, and X is the data matrix):

   ```
   spectral = cluster.SpectralClustering(n_clusters=k, eigen_solver='arpack',
   affinity="nearest_neighbors")
   spectral.fit(X)
   ```

4. *Analyse the results of clustering* in terms of

   - Homogeneity `metrics.homogeneity_score()`
   - Completeness `metrics.completeness_score()`
   - V-measure `metrics.v_measure_score()`
   - Adjusted Rand-Index `metrics.adjusted_rand_score()`
   - Silhouette Coefficient `metrics.silhouette_score()`

5. What is an optimal clustering method for each simulated data set?

6. Re-run the clustering methods on the Breast cancer and Mice data sets. Do not include the class variables in your clustering analysis but compare the obtained clustering with the true class labels.