CAPSTONE PROJECT REPORT

Spherical K-Means: Pattern Discovery in Textures

Name - Paleti Samuel Yashaswi
Course - Machine learning and AI
Duration - 24 months
Question - 7

Generate a dummy dataset using Scikit-Learn having high dimensionality (number of features >10) and total 4 classes. For this dataset, first implement K-Means clustering and then use the clusters for classification purpose. Now using the same dataset, implement spherical clustering and then check accuracy for classification. Notice the change in accuracy. You may also plot the obtained clusters from both the methods using t-SNE plots or by projecting data into two dimensions using PCA.

Dataset used is make blobs from sklearn

```
: import numpy as np
  import matplotlib.pyplot as plt
  from sklearn.cluster import KMeans, SpectralClustering
  from sklearn.datasets import make_blobs
```

```
: plt.figure(figsize=(6,5))
```

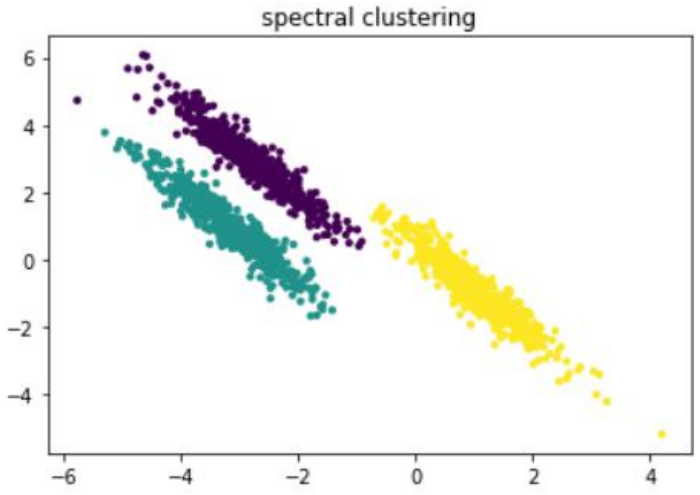
```
: <Figure size 432x360 with 0 Axes>

  <Figure size 432x360 with 0 Axes>
```

```
: n_samples =1500
  random_state = 170
  X,y = make_blobs(n_samples=n_samples, random_state=random_state)
  transformation = [[0.60834549,-0.63667341],[-0.40887718,0.85253229]]
  X_aniso= np.dot(X,transformation)
  y_pred=KMeans(n_clusters=3, random_state=random_state).fit_predict(X_aniso)
  y_pred2=SpectralClustering(n_clusters=3,gamma=5,random_state=random_state).fit
```

```
plt.scatter(X_aniso[:, 0],X_aniso[:, 1], c=y_pred2,s=10)
plt.title("spectral clustering")
```

```
Text(0.5, 1.0, 'spectral clustering')
```




```
plt.scatter(X_aniso[:, 0],X_aniso[:, 1], c=y_pred,s=10)
plt.title("Kmeans clustering")
```

```
Text(0.5, 1.0, 'Kmeans clustering')
```