

In []:

```
1 *****
2 #
3 #         Problem : Compress the Face Data into 5 clusters
4 #
5 *****
6 # Problem Statement :In this assignment students have to compress racoon grey scale image into 5 clusters.
7 # In the end, visualize both raw and compressed image and look for quality difference.
8
9 # Step0 : Loading the libraries
10 import numpy as np
11 import pandas as pd
12 import scipy
13 from sklearn import cluster, datasets
14 from scipy import misc
15 from sklearn.metrics import silhouette_score
16 import matplotlib.pyplot as plt
17 %matplotlib inline
18
19 # importing the Face Dataset from scipy.misc
20 from scipy.misc import face
21
22
23 face = face(gray=True)
24
25 # the number of rows and features and the Data type of the image
26 print(" The number of rows and columns  of Face DataSet ",face.shape)
27 X = face
28
```

```
In [ ]: 1 # for clusters =5
2 n_clusters = 5
3
4 X = face.reshape((-1,1))
5 clusterer = cluster.KMeans(n_clusters=n_clusters, random_state=10)
6 Z = clusterer.fit(X)
7
8 values = clusterer.cluster_centers_.squeeze()
9 labels = clusterer.labels_
10 print(" The Cluster Centers are\n", '-'*80)
11 print(values)
12 print("-"*80)
13 print(" The Labels are\n", '-'*80)
14 print(labels)
15 print("-"*80)
16 face_compressed = np.choose(labels, values)
17 face_compressed.shape = face.shape
18
19 vmin = face.min()
20 vmax = face.max()
21
```

```
In [ ]: 1 # Displaying the raw image and compressed image
2 face_original = scipy.misc.face(gray=True)
3
4 #plt.figure(1,figsize=(20, 3.6))
5 plt.subplot(1,2,1)
6 plt.imshow(face_original, cmap=plt.cm.gray,vmax = 256)
7 plt.title("Original Racoon Face")
8
9 #plt.figure(2,figsize=(10, 3.6))
10 plt.subplot(1,2,2)
11 plt.imshow(face_compressed,cmap=plt.cm.gray,vmin = vmin , vmax = vmax )
12 plt.title("Compressed Racoon Face")
13 #plt.show()
14
15
```

```
In [ ]: 1 # doing model evaluation to see the performance for various cluster numbers
2 X = face
3 #n_clusters = 5
4 clusters_range = range(2,10)
5 cluster_error=[]
6 sil_score=[]
7 print( " The Performance of Model by sihouette score and cluster Error \n",'-'*80)
8
9 for n_clusters in clusters_range:
10     cluster1 = cluster.KMeans(n_clusters=n_clusters, random_state=10)
11     y = cluster1.fit_predict(X)
12     silscore = round(silhouette_score(X, y),2)
13     message = "For n_clusters = {} The average silhouette_score : {} , Cluster Error: {} "
14     print(message.format(n_clusters, silscore,round(cluster1.inertia_,2)))
15     sil_score.append(silscore)
16     cluster_error.append(cluster1.inertia_)
17     # print("The Cluster error , SSE is : {} ".format())
18
19 clusters_df = pd.DataFrame( { "num_clusters":clusters_range, "cluster_errors": cluster_error , "sil_score":sil_score}
20
21 #print(clusters_df[0:n_clusters])
```

```
In [ ]: 1 plt.figure(figsize=(10,4))
2 plt.plot(clusters_df.num_clusters, clusters_df.cluster_errors, marker="o")
3 plt.plot(clusters_df.num_clusters, clusters_df.sil_score, marker="x" )
4 plt.xlabel("Clusters")
5 plt.ylabel("Cluster Error")
6 plt.text(25,15,"Silhouette score",verticalalignment='center', horizontalalignment='center')
7
```