Machine Learning Assignment Report

Task 1: - Unsupervised Learning

Learning Objective: -

- To apply the K-means clustering algorithm to perform the task of clustering. Vary the values of K and observe the differences.
- To find the gaussian clusters using GMM. The parameters of GMM are to be estimated using EM (Expectation-Maximization) Algorithm.

Steps / Procedures: -

- Import libraries.
- Upload data files(csv) from the local drive in google collab.
  V1=1 (as year yy is 22 which is greater than 20)
  Roll no-m22ma002
  So, 2%4 =2
  Binary conversion of 2 =10
  So v2 v3 v4 =[0 1 0]
  So,<v1 v2 v3 v4>=<1 0 1 0 >
  So total no. of 1's = 2
  So dataset =2:2 CITRUS (Orange Vs Grapefruit) will be taken
- Read and print csv file.
- Check the dataset for null values
- Find the correlated matrix .and remove the correlated features and class label.
- Convert the dataset into numpy array.
- Randomly selected the k rows as k centroids where k is the no. of clusters.
- Here initially we have taken as k=2.
- Initialise the closest class as a list of n zeroes, where n is a no. of rows.
- Set converged as False
- Calculate Euclidian distance between all the rows with all the centroids using scipy.spatial.distance.
- Closest of the tuple will be the centroids with the minimum distance.
- Reinitialise the centroids with the mean of the closest distance.
- Repeat the above three steps until converged becomes True.
- Returned the final centroid and the closest class.
- Print the centroids and values obtained at different values of k like k=2,3,4,etc.

Results And Observations: -

For k=2

[[150.8670125 156.3408054 81.1725848 7.85062488]

[199.63446461 151.31357128 70.76325872 14.93385763]]

Class-[0 0 0 ... 1 1 1]

For k=3

[[142.88744749 156.94340723 82.41511085   7.23249708]
 [209.46002293 150.64624959 69.67834916 16.01801507]
 [176.52978119 153.60983234 75.26541631 11.3486786]]
[0 0 0 ... 1 1 1]

For k=4

[[162.93518992 156.53418508 81.05041436   7.89502762]
 [215.23795909 150.85061846 70.29733587 15.73691722]
 [135.90783425 156.76144689 82.33470696   7.19276557]
 [187.86185593 151.06458481 70.19162527 14.89709013]]
[2 2 2 ... 1 1 1]

Graphs/Plots : -

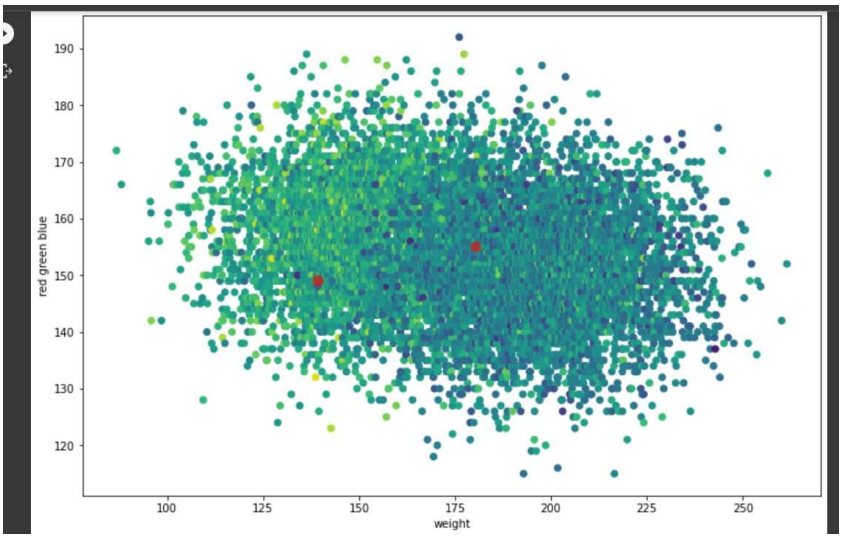

Fig1:Scatter Plot with Random Centroids

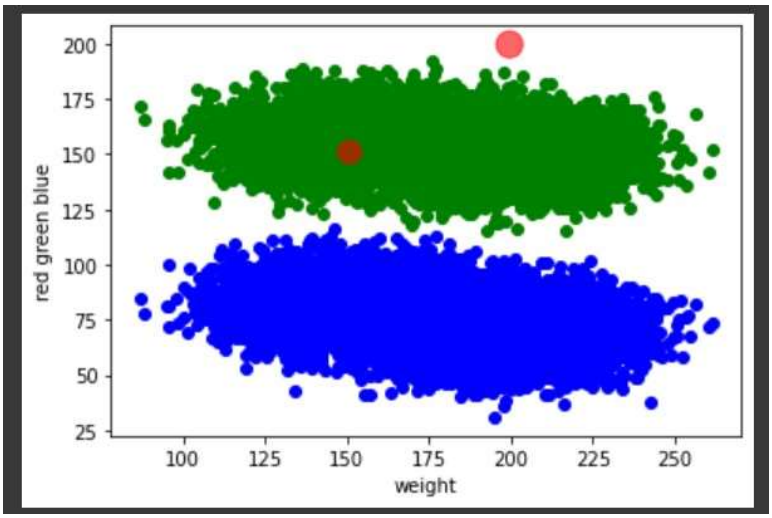

Fig2: Final Clustered Result with 2 clusters

Ans 1.3.

K-Means is not good at finding clusters of different sizes, shapes, and densities, which shows in the three examples here. GMM works very well on the data of different size and densities, as it shows in the citrus data and data plot in. Also, it's very fast. GMM may find some hidden parameter or failures which k Means cannot find.

References: -

- https://github.com/python-engineer/MLfromscratch/blob/master/mlfromscratch/kmeans.py
- Gaussian Mixture Models: implemented from scratch | by Vasile Păpăluță | Towards Data Science
- Associate Proff Deepak Mishra sir Class lectures

Collab Link:

https://colab.research.google.com/drive/1K8JFR284O6P5O7uKmS8AOkwQGB3N0t1m#scrollTo=R7z42rLdSu89

Task 2: - Using Principle Component Analysis (Dimensionally Reduction Techniques)

Learning Objectives:

- To reduce dimensionality of the dataset using PCA.
- To apply KMeans and GMM on the set of uncorrelated features obtained after using PCA
- Plot the cluster results.

Procedures:

- Import libraries.
- Upload data files(csv) from the local drive in google collab.
- Read and print csv file.
- Check the dataset for null values
- Normalize the dataset using standardscaler from sklearn preprocessing.
- Reduce the features using sklearn decomposition. PCA and fit and transform the dataset.
- Use this reduced feature set received from reducing the original dataset with PCA using the best components found.
- Apply Kmeans clustering on this new dataset .

Results :

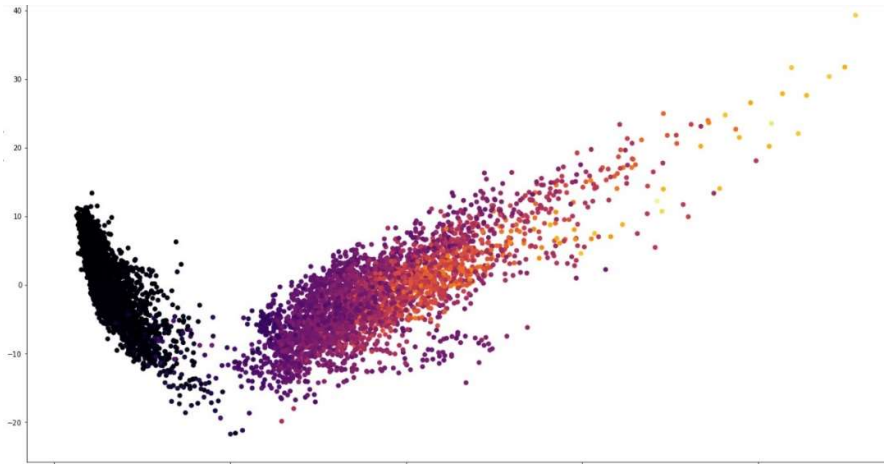

Fig1:Principle Components Analysis plot

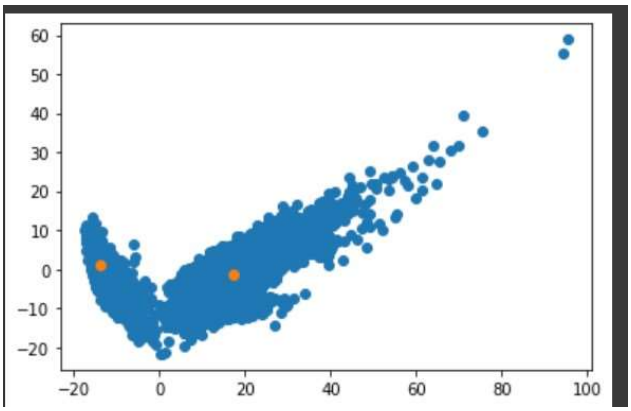

Fig: Applying KMeans Clustering On PCA

Pca array representation:-

array([[2.22657610e-05, -2.47966030e-03, -1.53186110e-03, ...,
2.65870280e-02, -5.68138307e-03, 4.47017443e-02],
[1.56144791e-02, -4.42026377e-06, -3.98348818e-03, ...,
-1.70395525e-02, -6.90784277e-03, -4.48424175e-02]])

Ans 2.3 we have plotted the scatter results obtained after applying K Means in PCA dimensionality reduction in the result section

References:

https://www.geeksforgeeks.org/implementing-pca-in-python-with-scikit-learn/

https://www.analyticsvidhya.com/blog/2021/04/k-means-clustering-simplified-in-python/#:~:text=Step%2D1%3A%20Select%20the%20value,will%20form%20the%20predefined%20clusters.

Collab Link:

https://colab.research.google.com/drive/1_0jKMFjrJtzjTaPfo_X41VpSFFWTGOJa#scrollTo=CEszKfQphGOL