CmpE 462 Spring - 2020 Project 3 Report

Alperen Bağ - 2018400279
Cihat Kapusuz - 2016400126
İbrahim Özgürcan Öztaş - 2016400198

June 21, 2020

Contents

Chapter 1

Introduction:

This report is composed to explain the overall structure of the requested project with elaborated statements. Our project description was to implement and apply several unsupervised learning algorithms to resolve different problems with the power of computation and mathematics.

In first part, we've implemented K-Means clustering algorithm to classify a data set into several clusters with no additional data except the data set itself. And, we've implemented the K-Means clustering algorithm to group all data around several centroids, which are the center points of the current clusters. At each iteration, we've found the new centroid for each cluster and with the new center point, we've found out new set of points related to these centroids.

In the second part of the project, we've implemented the PCA(Principal Component Analysis) to understand the decomposition of the set of features in a data set and reconstruction of the original data with the supporting components derived from the PCA. It is mostly used in facial recognition and handwritten digit recognition. By extracting the weights of the features that defines the data set, we can reconstruct a new data set with the most weighted features, which leads to the efficiency of computational power with a decrease in resemblance ratio between the newly constructed data and the original data.

Chapter 2

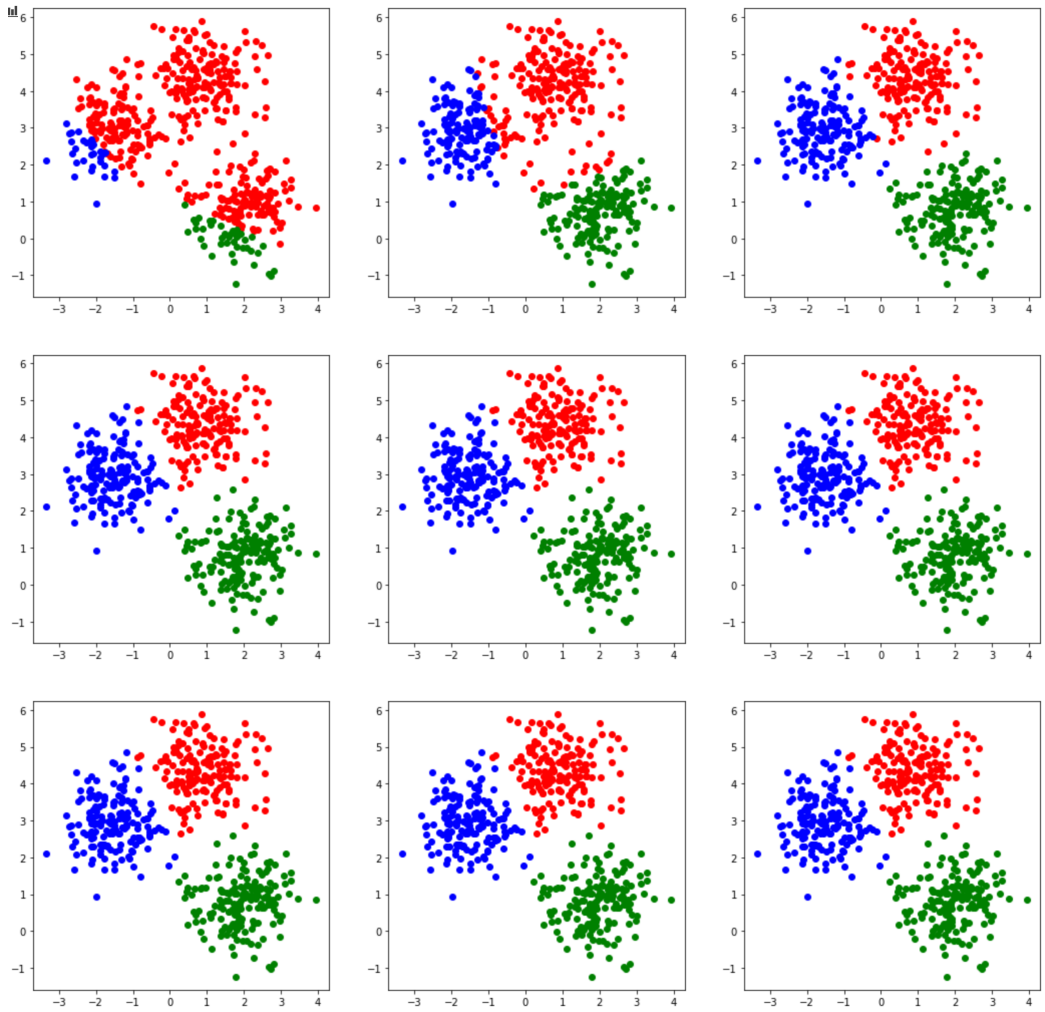
K-Means:

K-Means clustering algorithm is based on the random selected starting points for each centroid and for each iteration, these points and its related set of points that constructs a cluster changes into a new set of points that constructs a new cluster.

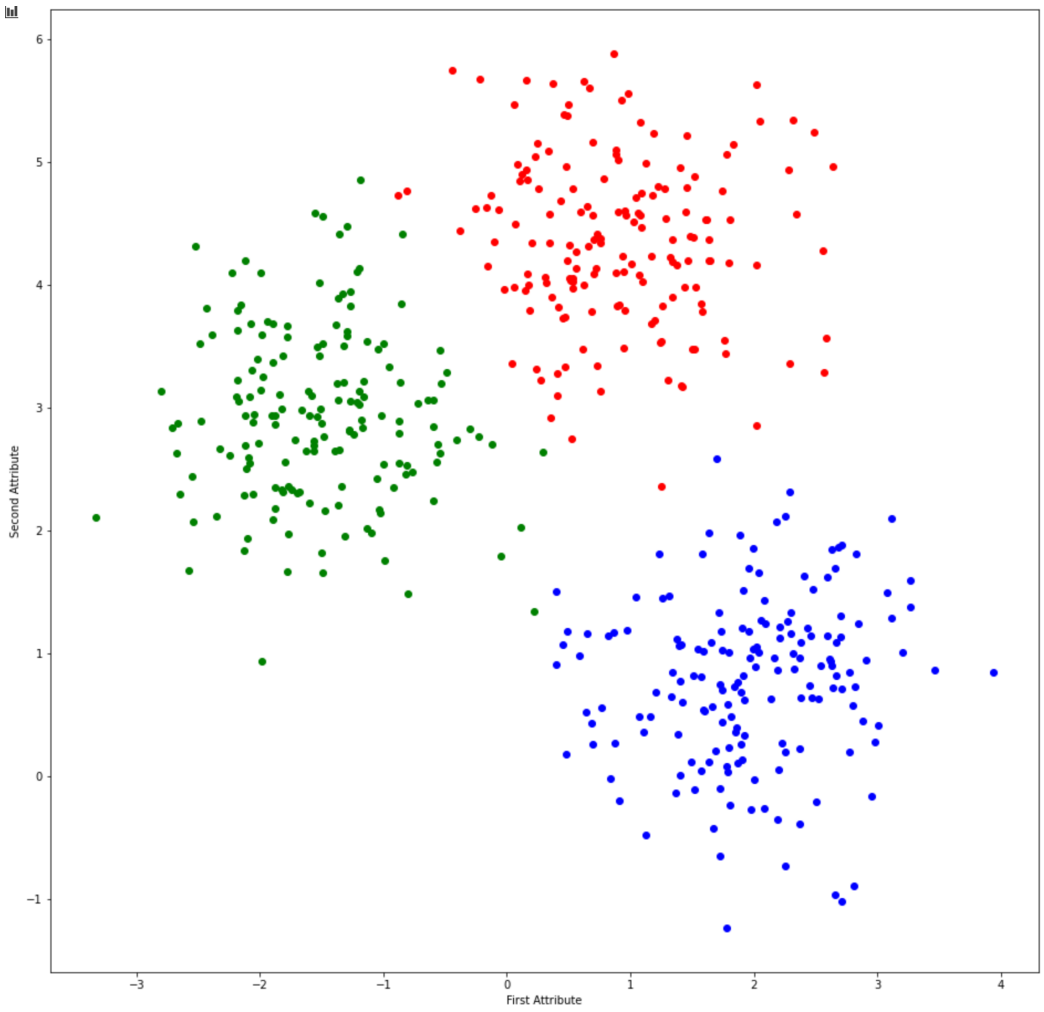
At first, we've selected our centroids with uniform random distribution. We've extracted the minimum value in the data set and the range of the data set. After these information found out, we've used numpy.random.rand() to get a value between 0 and 1 and by adding its multiplied value with range into the minimum value results in a point that is certain to be into the data set. To prevent uncertainty in random numbers, we've set our seed value as 1 at the beginning of the function k_means(n, seed) in our code, where n means the number of iterations that the K-Means algorithm runs.

Then, we've initialized our predictions as an array of zeros, and started to iterate over our data set. At each iteration, we've calculated the norm value(the absolute value of the shortest distance between the current point and the centroid) regarding the three centroids for all points in the data set. After that, we've found out the minimum value of these three values to decide which cluster that specific point in the data set should reside. Assigning the cluster to the current point, we've recalculated our centroid point of each cluster by averaging the x and y coordinates of all points that resides in that clusters separately.

Finally, we've calculated all iterations on the data set and found out the results. Plotting them shows us the alteration at each iteration and the flow of the points inside clusters. At the fourth iteration, we've reached to the given final cluster and the remaining iterations won't create any difference in the clusters.



The given final cluster is plotted below:



Each centroid is uniquely colorized with a color among red, green and blue and it is quite spectacular to classify a data set into such distinct clusters.

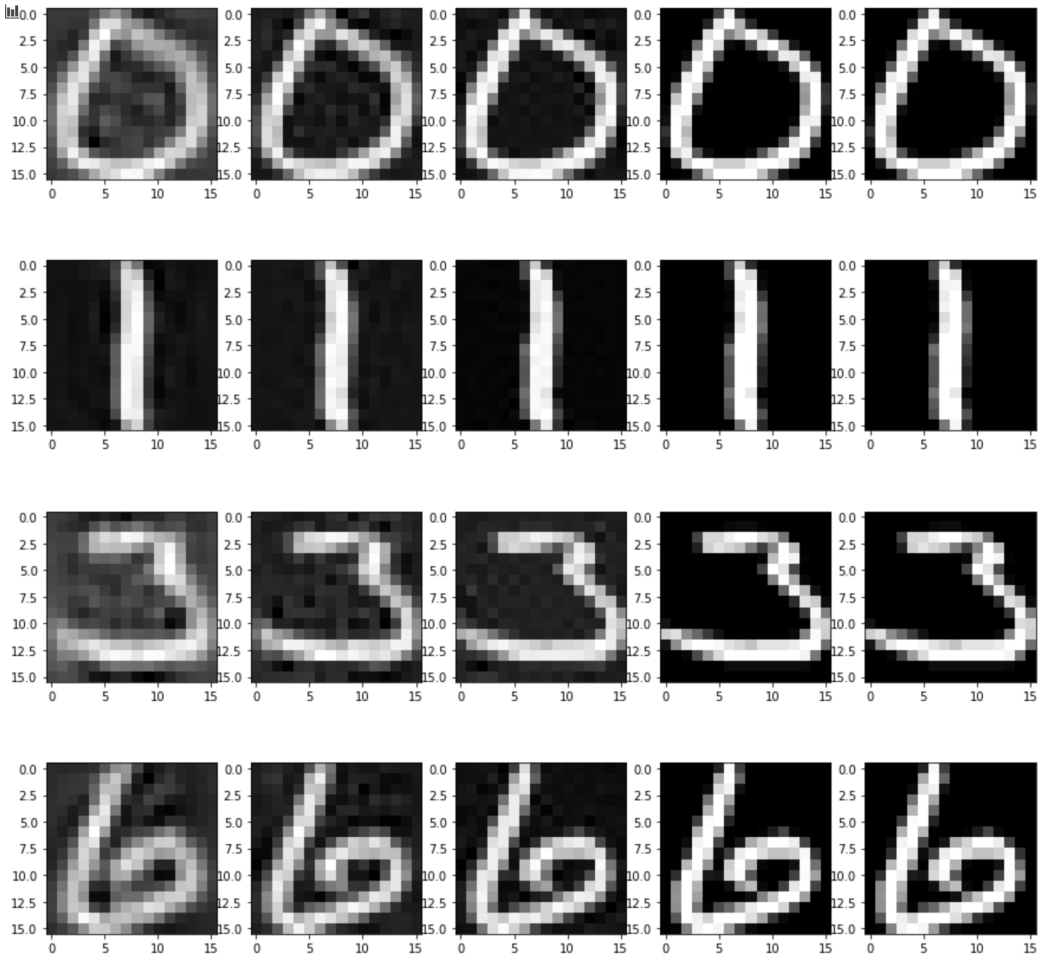
Chapter 3

Principal Component Analysis:

In Principal Component Analysis, we've calculated the rank of features by applying eigen-decomposition over feature covariance matrix to find out the relations among features and their weights.

At first, we've standardized our data by subtracting the mean of each feature from all images to set the mean 0 while protecting the variance. Then, we've calculated our covariance matrix to see the relations among features and its weights. Then, we've decomposed the covariance matrix to find out the eigenvectors of the covariance matrix, which are the foundation of the features of the data set. All eigenvectors are sorted by the eigenvalues, which leads to a ranking of eigenvectors starting from the most fundamental feature and ending with the least fundamental feature.

By selecting the first d features and multiplying with the original image, we can have a newly constructed image which has a resemblance with the original image. The more feature included into the transformation matrix, the more elaborate the image gets! If one chooses to use all features in the feature set, one will get the original image, since there's no loss of information. Even though reconstructing the original image is the ultimate step in this algorithm, its cost gets increased by adding another feature. Therefore, there's a tradeoff between the number of features included in the transformation matrix and the cost. By adding sufficient number of features with an awareness of negligible amount of information loss, the overall system results in the efficiency of computational power and energy.



As you can see, using first 50 features among 256 feature constructs an image that has the fundamental traits, but it lacks of details that defines the digits. Adding 50 more features sharpens the edges of the digits, yet it has still areas that is mostly blur. Furthermore, adding another 100 results in a very well defined image of a digit, which has a very high percentage of resemblance with the original image of the digit. We can deduct that by using more than half of the all features, between 130 and 150 features, we can construct a new image which has a very high resemblance rate with the original image for this data set. Yet, it is still not a fact since we have not run the algorithm for all images of digits in the data set.

Chapter 4

Conclusion:

In this project, we've learned to classify a data set by using only its points by K-Means clustering algorithm. Also, we've grasped the fundamental process of the Principal Component Analysis to efficiency reconstruct the data with a high rate of resemblance rate. These techniques has a great power in their functioning and we can solve complex problems by applying the suitable algorithms that we've learned in this course. Thanks for everything you've taught us! :)