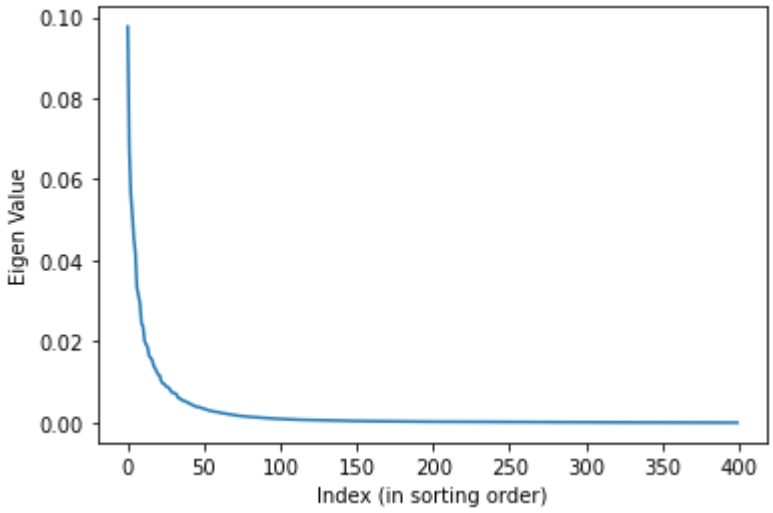
**Mustafa Yaşar**
**21702808**
**GE461 - Introduction to Data Science**
**Spring 2022**

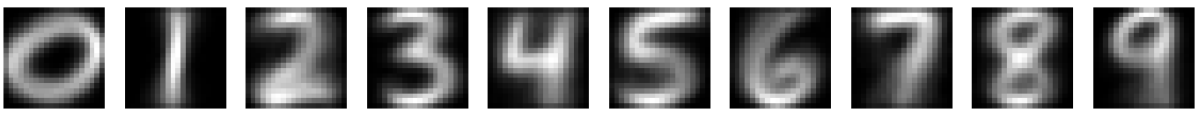
## Question 1

1. The PCA has been used to obtain the eigen values.




The figure above shows the eigenvalues in descending order. As we can see from the plot above, after the 100'th eigen-value of the eigen-vector, the explained variance will be quite low. Therefore, using more than 100 principal components might not affect the performance of the model.

2. The sample mean for the whole training data set has been calculated and the following images have been obtained.



As can be seen above, every number is legible and understandable. To obtain those images, every feature with labels has been combined and the means of the vectors have been computed. When the resulting matrices have been displayed, the above images have been printed.

Even though we have used 100 features instead of 400, the hand-written digits are completely readable and understandable. That way, thanks to PCA, we can reduce the computation work significantly. It is observable that when we display the sample mean, it is less legible than the generated bases with 100 principal components. Therefore, it's more efficient to use data that is projected onto 100 principal components than the sample mean.

3. Projecting data onto subspaces and Gaussian Classifier

First of all, the Gaussian classifier to use in this project has been obtained from 'scipy.stats' library and the name of the classifier is multivariate_normal. The details of the classifier can be seen in this link:
"https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.multivariate_normal.html"

To train the Gaussian Classifier, first of all, a Gaussian has been fitted to the corresponding subset of the training data for each class (from 0 to 9). The mean estimates and covariance matrices for each class is computed in the 'seperate_training_set' function.

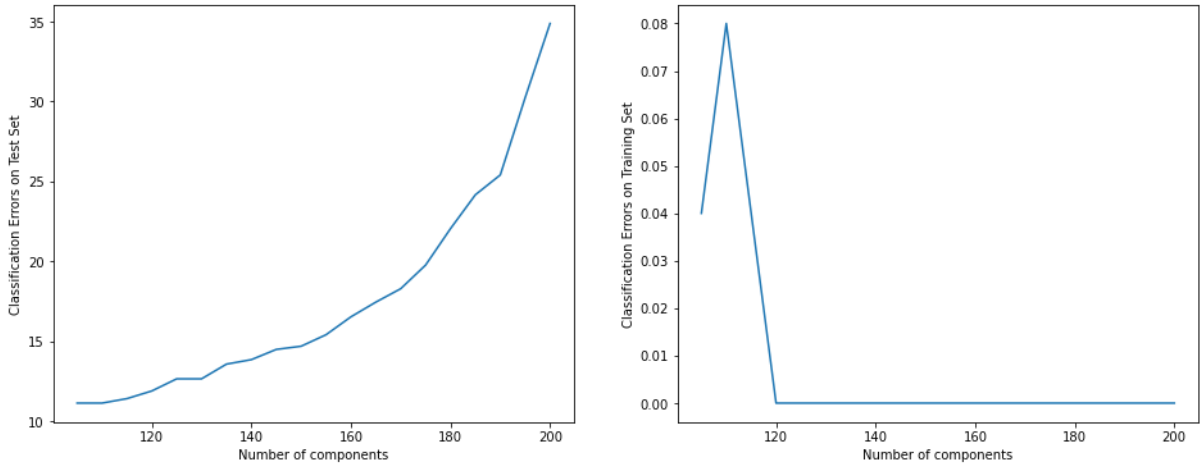
When the mean vector estimates and covariance matrices are learned, probabilities of each class for each test data (a vector) has been calculated, and the biggest of those probabilities has been taken as the prediction.

The dimensions of subspaces have been selected as [105, 110, 115, 120, …, 195, 200]. For each subspace dimension, a PCA whose number of principal components the subspace dimension is fitted and the training and test data has been projected using the transformation matrix.

After training, the testing has been done and the following results have been obtained.

(Test) Accuracy when subspace dimension is 105 : 88.88000000000001
(Training) Accuracy when subspace dimension is 105 : 99.96000000000001
(Test) Accuracy when subspace dimension is 110 : 88.88000000000001
(Training) Accuracy when subspace dimension is 110 : 99.92
(Test) Accuracy when subspace dimension is 115 : 88.6
(Training) Accuracy when subspace dimension is 115 : 99.96000000000001
(Test) Accuracy when subspace dimension is 120 : 88.12
(Training) Accuracy when subspace dimension is 120 : 100.0
(Test) Accuracy when subspace dimension is 125 : 87.36
(Training) Accuracy when subspace dimension is 125 : 100.0
(Test) Accuracy when subspace dimension is 130 : 87.36
(Training) Accuracy when subspace dimension is 130 : 100.0
(Test) Accuracy when subspace dimension is 135 : 86.44

(Training) Accuracy when subspace dimension is  135 : 100.0
(Test) Accuracy when subspace dimension is  140 : 86.16
(Training) Accuracy when subspace dimension is  140 : 100.0
(Test) Accuracy when subspace dimension is  145 : 85.52
(Training) Accuracy when subspace dimension is  145 : 100.0
(Test) Accuracy when subspace dimension is  150 : 85.32
(Training) Accuracy when subspace dimension is  150 : 100.0
(Test) Accuracy when subspace dimension is  155 : 84.6
(Training) Accuracy when subspace dimension is  155 : 100.0
(Test) Accuracy when subspace dimension is  160 : 83.48
(Training) Accuracy when subspace dimension is  160 : 100.0
(Test) Accuracy when subspace dimension is  165 : 82.56
(Training) Accuracy when subspace dimension is  165 : 100.0
(Test) Accuracy when subspace dimension is  170 : 81.72
(Training) Accuracy when subspace dimension is  170 : 100.0
(Test) Accuracy when subspace dimension is  175 : 80.24
(Training) Accuracy when subspace dimension is  175 : 100.0
(Test) Accuracy when subspace dimension is  180 : 77.96
(Training) Accuracy when subspace dimension is  180 : 100.0
(Test) Accuracy when subspace dimension is  185 : 75.84
(Training) Accuracy when subspace dimension is  185 : 100.0
(Test) Accuracy when subspace dimension is  190 : 74.6
(Training) Accuracy when subspace dimension is  190 : 100.0
(Test) Accuracy when subspace dimension is  195 : 69.76
(Training) Accuracy when subspace dimension is  195 : 100.0
(Test) Accuracy when subspace dimension is  200 : 65.12
(Training) Accuracy when subspace dimension is  200 : 100.0



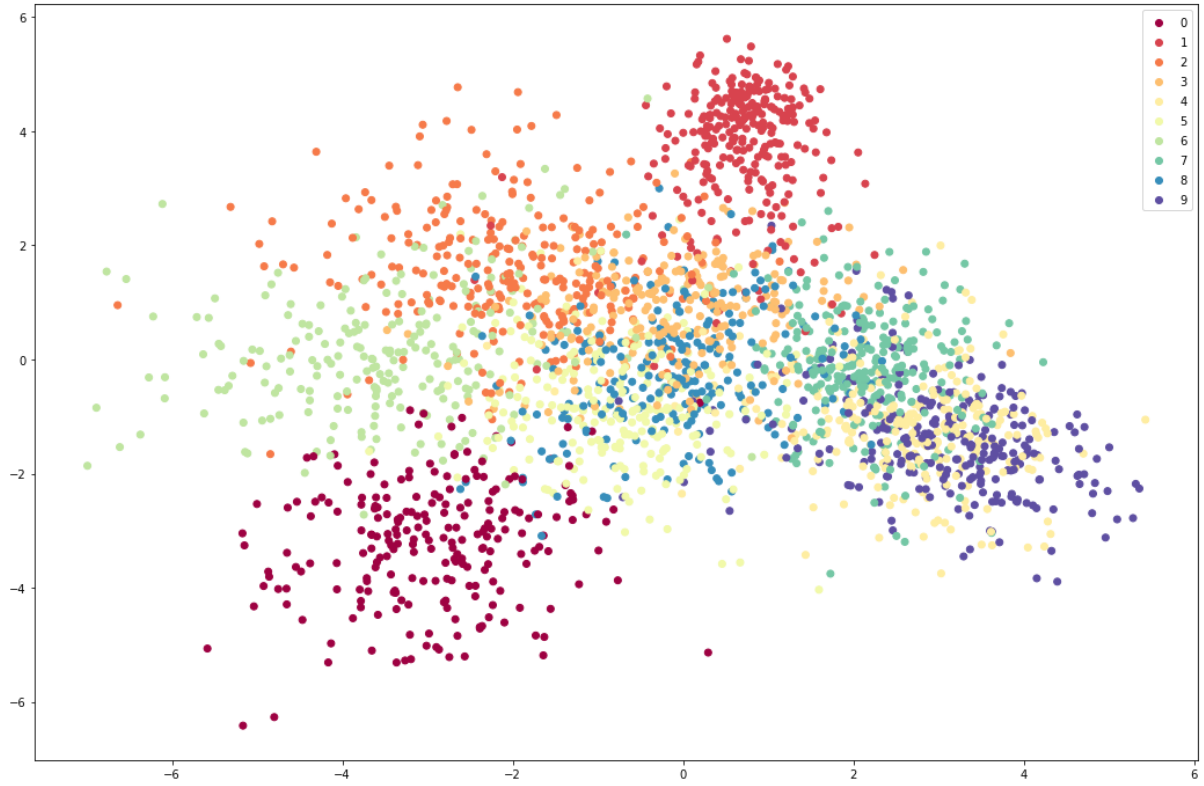

The plot on the left shows the classification errors on the test data set according to the number of components used. The plot on the right hand-side shows the classification error on the training data set according to the number of components.

It can be observed that the error increases as the number of components increases. This is because of overfitting to the training data. As can be observed from the plot on the right, the training accuracy is 100% when the number of components exceeds 120. As the model overfits, the bias decreases, however, variance increases. In this case, the training accuracy increases whereas the test accuracy decreases (Bias-Variance trade-off).

## Question 2

1. LDA is another dimensionality reduction technique which is used to avoid the curse of dimensionality and reduce the number of features so that the computation work of the model decreases. When the LDA is fitted with training data, it estimates the mean and variance from the data for each class with some assumptions. The first assumption is that each variable is taken from a Gaussian distribution, and the second assumption each feature has the same variance.

LDA makes predictions the same way the Gaussian classifier does. Therefore, LDA is not only a dimensionality reduction technique, but also a classification method.



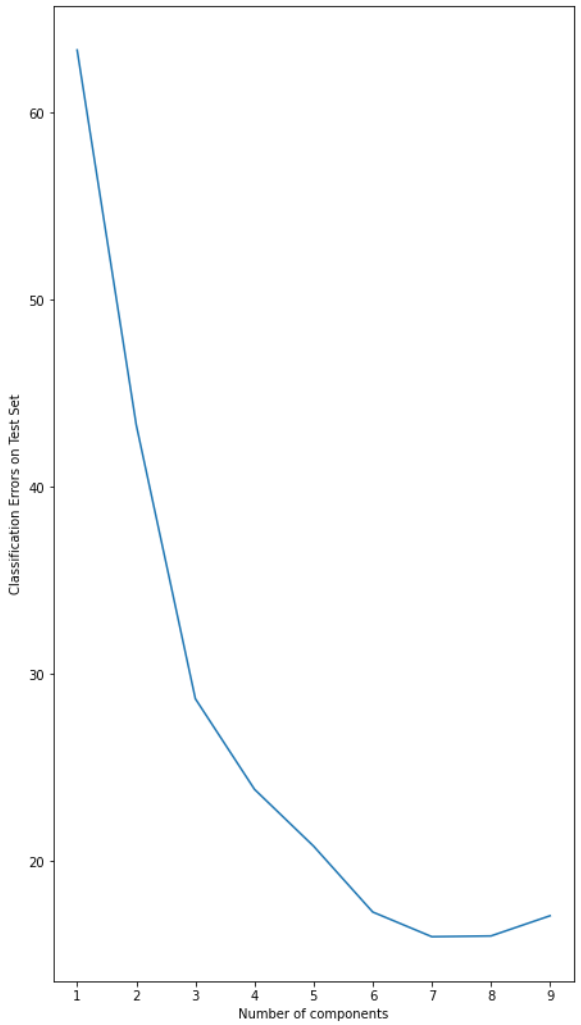

As can be seen above, the LDA groups the classes quite accurately. Since the data is not linearly or polynomially separable, there are mixtures in the groups. When we make the LDA make predictions, it has obtained 81.24% accuracy which is considerably high.

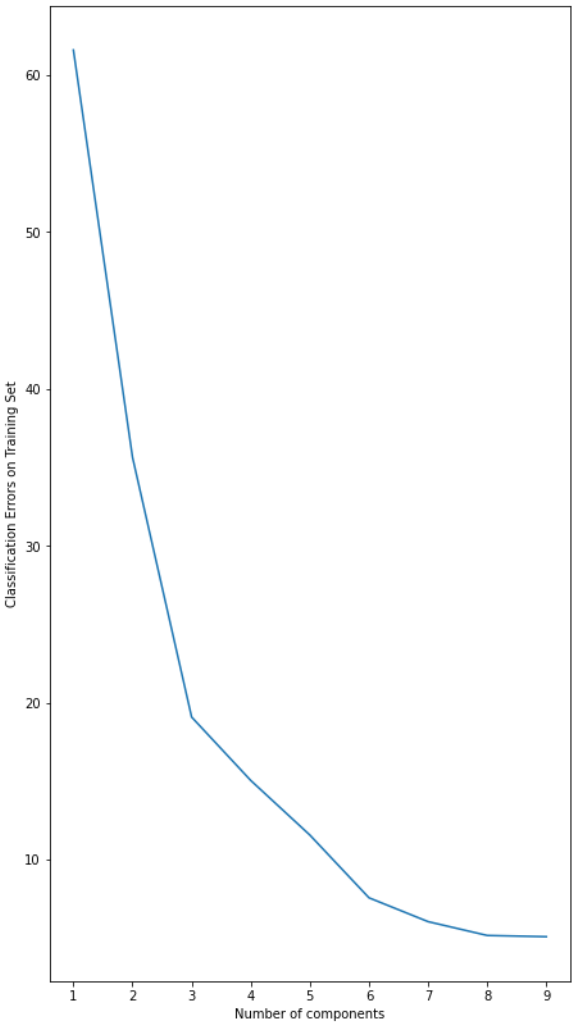
2. The LDA has been computed for each possible dimension (1, …, 9) and both training data and the test data has been projected using the transformation matrix onto these

subspaces. The Gaussian classifier has been trained using data in each subspace and the following accuracies has been obtained.

(Test) Accuracy when subspace dimension is 1 : 36.64
(Training) Accuracy when subspace dimension is 1 : 38.4
(Test) Accuracy when subspace dimension is 2 : 56.68
(Training) Accuracy when subspace dimension is 2 : 64.36
(Test) Accuracy when subspace dimension is 3 : 71.32
(Training) Accuracy when subspace dimension is 3 : 80.92
(Test) Accuracy when subspace dimension is 4 : 76.16000000000001
(Training) Accuracy when subspace dimension is 4 : 84.96000000000001
(Test) Accuracy when subspace dimension is 5 : 79.2
(Training) Accuracy when subspace dimension is 5 : 88.44
(Test) Accuracy when subspace dimension is 6 : 82.72
(Training) Accuracy when subspace dimension is 6 : 92.44
(Test) Accuracy when subspace dimension is 7 : 84.04
(Training) Accuracy when subspace dimension is 7 : 93.96
(Test) Accuracy when subspace dimension is 8 : 84.0
(Training) Accuracy when subspace dimension is 8 : 94.84
(Test) Accuracy when subspace dimension is 9 : 82.92
(Training) Accuracy when subspace dimension is 9 : 94.92

3.



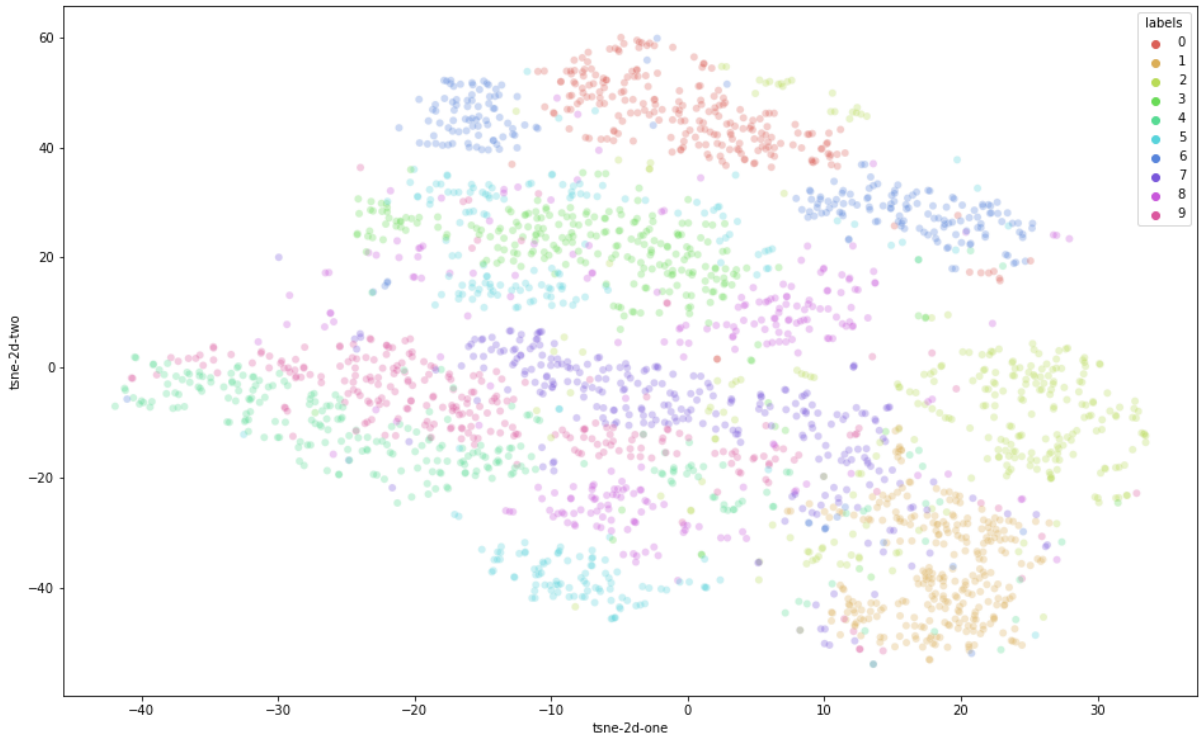




As can be seen, generally the accuracy increases as the LDA dimension increases because we are using more components. However, when the dimension is greater than 7, as can be observed above, the accuracy of tests decreases even though the dimension increases. This is not unexpected because LDA does not guarantee the fact that higher dimensions will provide higher accuracy. Moreover, it's likely that this is because of overfitting to the training data. As the model overfits, the bias decreases, however, variance increases. In this case, the training accuracy increases whereas the test accuracy decreases. (Bias-Variance Trade-off)

## Question 3

**t-SNE:**

By using TSNE of sklearn.manifold, the number of iterations which gives the lowest KL divergence has been obtained. The number of iterations which results in the lowest KL divergence is 1000, however, 600 iterations resulted in quite close KL divergence to 1000 case, therefore, n_iter has been selected as 600 to decrease the computation burden.




The dimension of the data has been reduced to 2 and the above scatter has been obtained.

We can use t-SNE and PCA together to increase the model quality. When t-SNE is used on PCA-Reduced data (120 principal components explains 95% of the variance) the following scatter has been obtained.