Exercise 2 Report

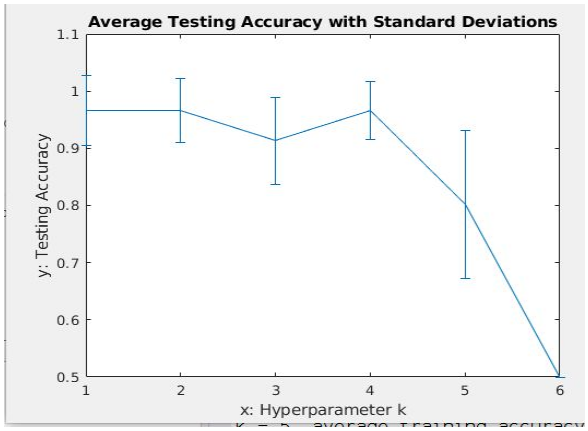
Name: Wenchang Liu, ID: 10406141

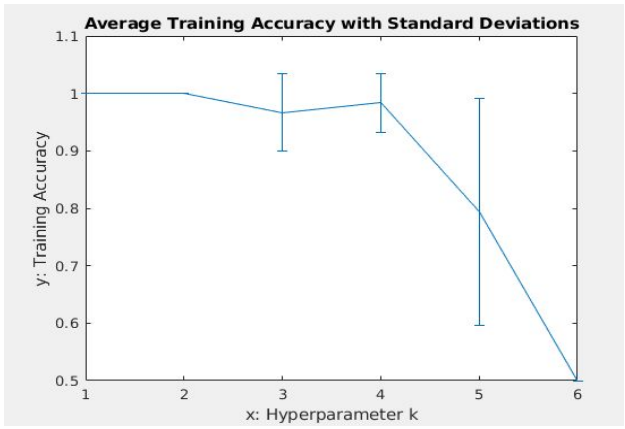
Part 1.a: This part is to build a k-NN binary classifier to recognise face images for subjects "1" and "30", the whole process can be divided into the following steps:

1. Prepare datasets of the face images taken for subject "1" and "30".

2. Use k-NN model to classify the prepared datasets and compute training and testing accuracies.

3. Use errorbar command to display average training and testing accuracies on different value of k with the standard deviation.





4. Observe the classification results by using ShowResult.m function.

Q&A-1. The training accuracy obtained when k = 1 is 100%, because the nearest neighbor is itself, so every prediction is exactly the same with the training label. And the testing and training accuracies do differ, this is because the model is trained by the training set so it has to fit most patterns of the training set, but there might be overfitting and underfitting problems that which might cause lower testing accuracies.

Q&A-2. No, because we split the data into training and testing set randomly, so we have different training sets, different models and different testing sets, and that's why we got little difference between each dataset.

Q&A-3. No, for the machine cannot vote if the labels for each subjects are equal.

Q&A-4. Subject "1", he has different postures and expressions of his eyes, for it may contain more different features.

Part 1.b: This part is to implement the same classifier on 40 subjects, the process should contain the these steps:

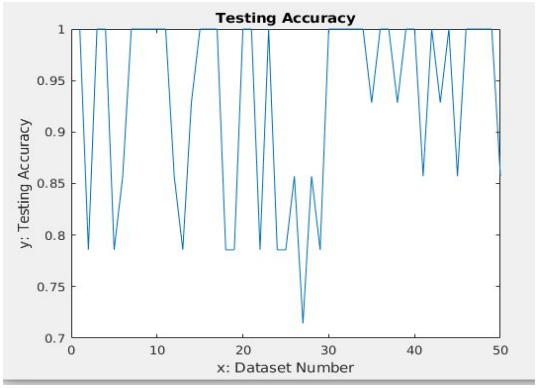
1. Implement leave one out cross validation to decide the best k value.
2. Prepare datasets for all 40 subjects.
3. Try k-NN classifier with the selected k value, and compute average testing accuracy and the standard deviation.
4. Count the wrong classifications of each subjects.

Q&A-1. No, we wish the training set represent all features from the different subjects.

Q&A-2. Subjects that are most difficult to recognise: 1, 3, 4, 10, 13, 16, 40

Part 2.a: This part is to implement a linear binary classifier for subject "1" and "30", the process should be as the following steps:

1. Prepare datasets of the face images taken for subject "1" and "30".
2. Add one "1" column for matrix Xtr and Xte.
3. Use normal equation to make predictions, compare with the ground truth label, and compute the training and testing accuracies.
4. We can use ShowResult.m function to observe some classification results.
5. Draw a graph to display the testing accuracies.

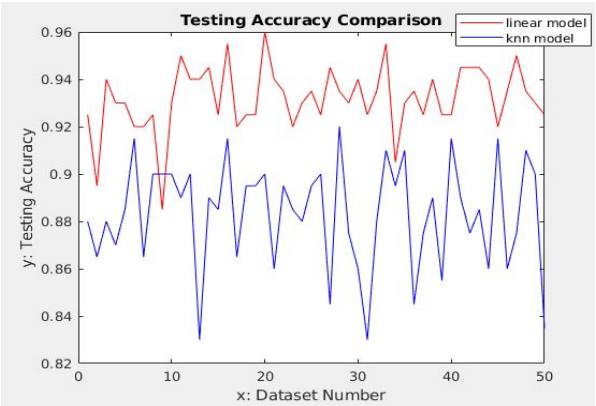


Q&A-1. 100%, because the training data can be divided with a linear boundary.

Q&A-2. The same as k-NN, subject "1".

Part 2.b: This part is to build a linear multi-class classifier to recognise all 40 subjects, the whole process should be as the following steps:

1. Prepare datasets of all 40 subjects.
2. Make 1 of k matrix for Y(labels)
3. Use the same method as part 1.a and get the prediction-matrix
4. Select the most likely prediction(close to 1) for each subject and compare with the ground truth labels, then compute the testing accuracies.
5. Count error classifications of each subjects
6. Do the k-NN classification again and compare with linear model.
7. Display the testing accuracies of k-NN and linear model in the same graph.



Q&A-1. Linear classification is slightly better in terms of accuracy, much better in terms of time and space complexity than the k-NN model.

Q&A-2. Subjects 1, 3, 8 ,13, 16, 36 are difficult for linear classifier, and subjects 1, 3, 4, 10, 13, 16, 40 are difficult for k-NN classifier, they have some same difficulties, but not completely the same.