Gabriel Rosca - 10159887

COMP24111 - Machine Learning and Optimisation
Exercise 2 lab report

Face Regocnition

Part 1: k-Nearest Neighbour

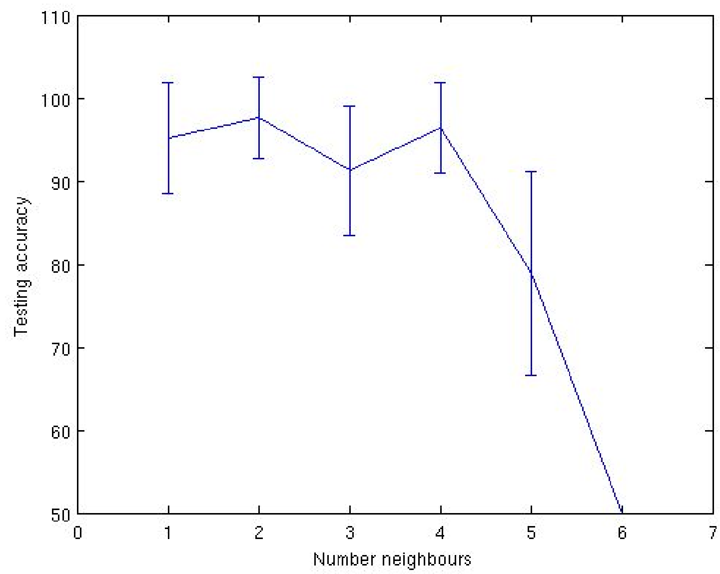
Q1: The training accuracy obtained for k = 1 is 100% because we calculate the distance from one point from the set of training points to the all points from the same set(training points/data). The minimum distance will always be 0 because the nearest point will always be same point from the training set points. Testing and training accuracies differ because one is based on a set which the classifier has not seen yet and one is based on a set of points the classifier was made. In result, the training accuracies for k=1,2 will always be 100%. The testing accuracies will be lower than the training accuracies. For the training set, the classifier will try to classify a point which is already in the training set.

Q2: We don't get the same behavioUr on different data sets because the samples included in the data sets differ from one to another. For example, in one data set we might have the samples 1,3,4 for training and the rest for the testing and in another data set we have 2,5,6. So we will get a different accuracy thus a different behavioUr.
Q3: It is not a good idea to set the k an even value because we might have the first k distances equal and half of them from Class1 and the other half from Class2. The classifier will then randomly assign the point/object to either of class 1 or class 2. The accuracy will differ over different values of k. When k gets bigger the classifier will become lousier and might include some points in a wrong class.

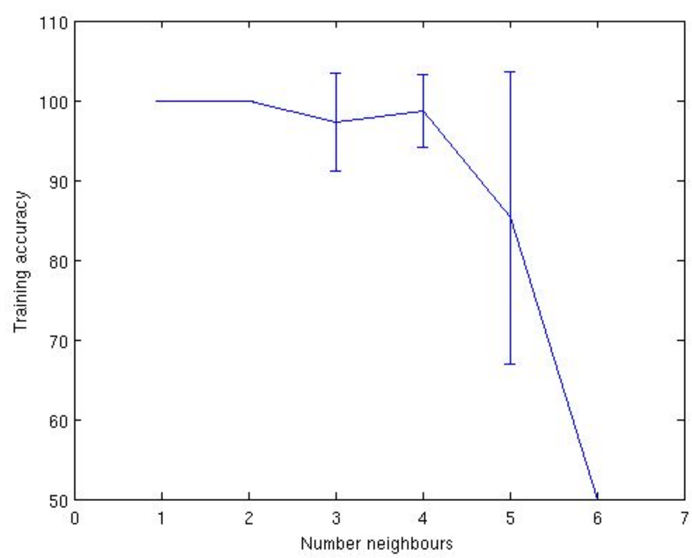
Q4: The first subject is harder to classify because the average accuracy is always lower than the subject 30. (80% vs 100% using k = 1).

For **Part1a** we built a k-NN classifier to regocnize face images for subjects “1” and “30”. Using the PartitionData function we partitioned the data(photos for each subject) and labels(the label for each photo in the data matrix) randomly into 2 set: Training and Testing. We prepared 50 datasets containing different training/testing sets.

In the right graph we are showing the testing accuracies for different k values ranging from 1 to 6. We can see from the graph the accuracies tend to decrease as we increase k until we reach a 50% accuracy for k = 6. The 50% accuracy for k = 6 is because we take into consideration all the neighbours and it will always randomly assign a photo to a random class between the 2 available in the test sets. We can also observe the standard deviation for each k.



In the right graph we are showing the training accuracies for different k values. For k = 1 we have a 100% accuracy and a 0 standard deviation.

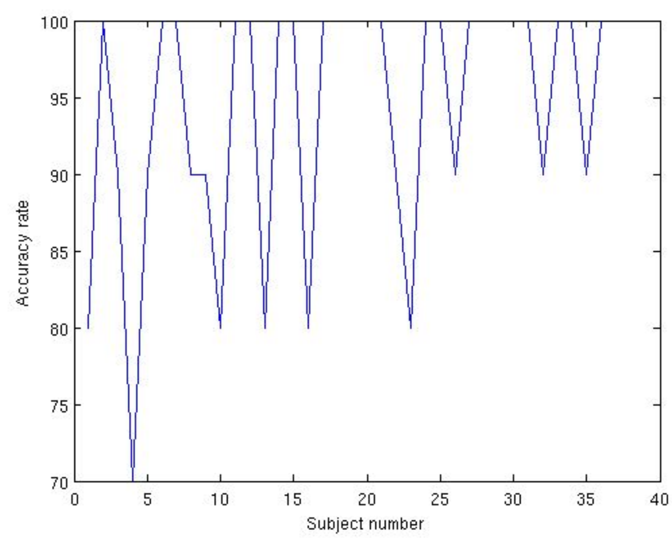


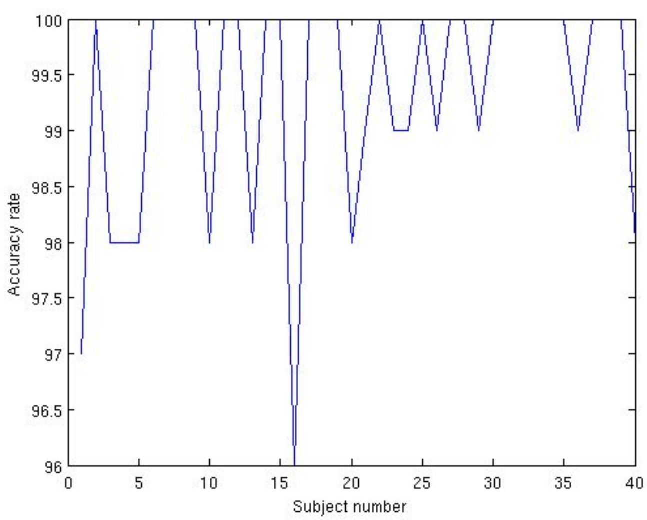
For **Part1b** we partitioned the whole data set into 200 photos/samples for testing and 200 photos/samples for training. We choose a random data set to implement the leave-one-out cross-validation to decide the value of the hyperparameter k(number of neighbours taken into consideration).

Q1: It is not a good idea to randomly select 200 training samples from the whole image set without specifying the number of selected samples for each subject because we might end up with no training photos/samples for one subject. Therefore, we will not have a training sample for a subject so we will missclassify every photo of that subject.

Q2: Based on my my k-NN classifier the most difficult subject to classify are: Subject 4(70% accuracy), Subject 23(80% accuracy), Subject 13(80% accuracy). All of these accuracies are based on a value of k =1.

The graph on the right is showing the testing accuracies for each subject(applying the L00) .





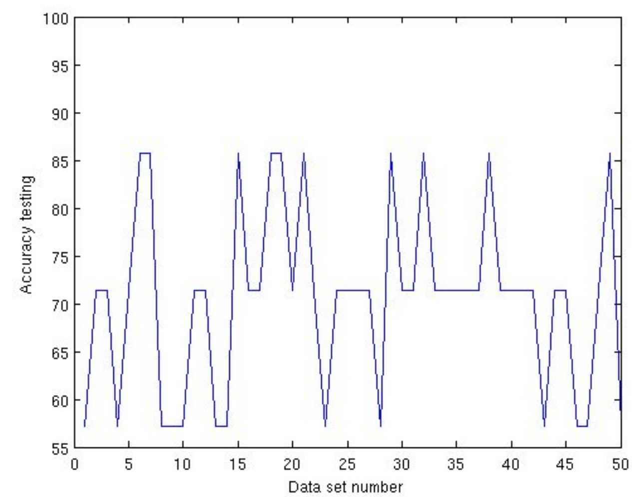
The graph on the left is showing the testing accuracies for each subject using the training sets from data set number 1.

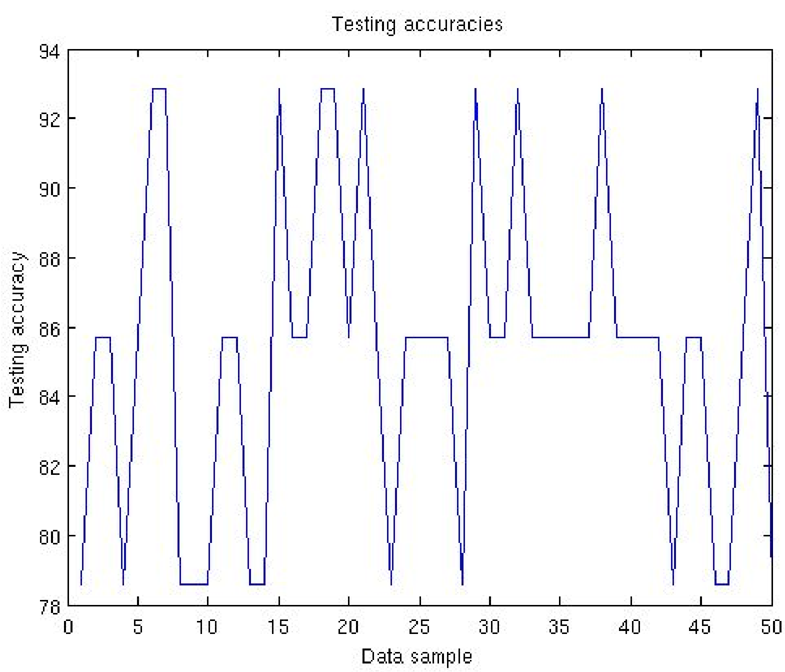
Part 2: Linear Least Squares

For Part2a we implemented a linear binary classifier to classify face images for subjects “1” and “30”.

Q1: With the linear classifier we obtain a higher training accuracy(100%) than the testing accuracy(96.5%).

Q2: The subject “1” is more difficult for the linear classifier to recognise. The result from linear classification for the input vector from subject 30 will always be greater than 15 therefore we can classify it more easily.(See the graph below where blue=subject “1” and the green=”30”).





The graph on the left is showing the testing accuracies for each data set.

For Part2b we implemented a linear multi-class classifier based on the least squares approach using the normal equations.

Q1:

	Mean accuracy	Time
kNN classifier	85.404%	0.873 s
Linear classifier	95.000%	0.022 s

Both k-NN and linear classifier have the same space complexity because we use the same structures(50 data sets). The linear classifier also stores a W for saving the coefficient vector [w0,w1,...wd] where d = number of features of the photo.

Q2: For the linear classification the subject 1 is considered the most difficult to recognise(74%) while the accuracy for the kNN is 60% while using 1 neighbour. The two classifiers find almost the same subjects difficult.

Green = kNN

Blue = Linear multi-class classifier

We can see that the testing accuracies are lower for the kNN classifier than the ones for the Linear multi-class classifier.