Pattern Recognition

Assignment 1

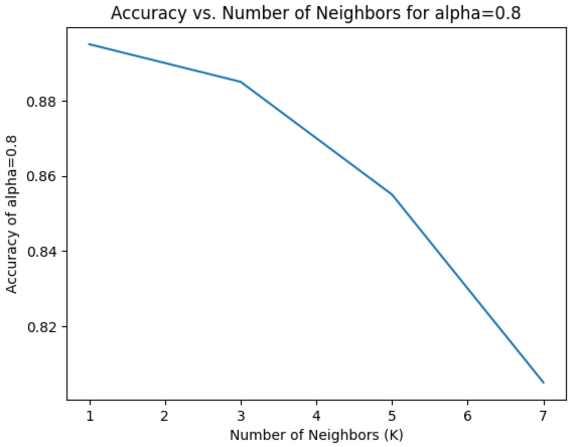
Face Recognition

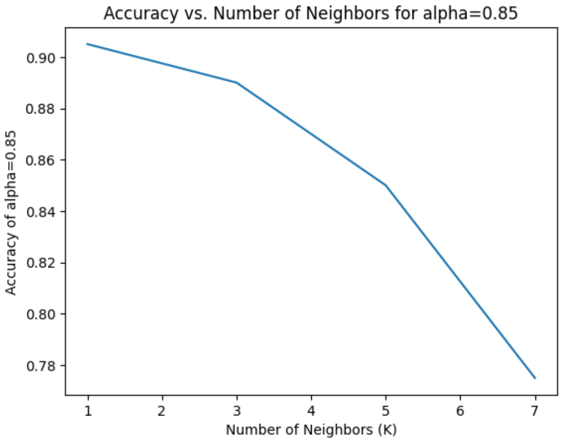
Authors:

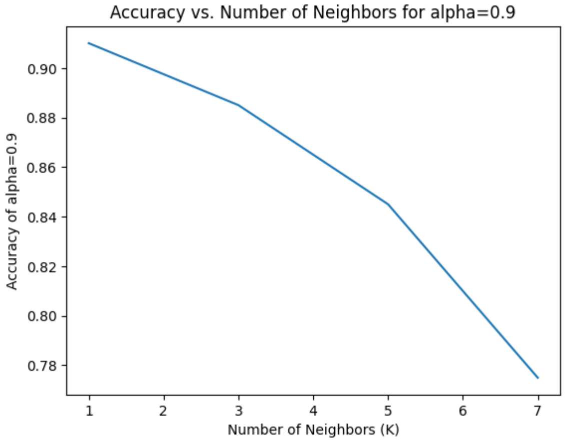
6207	Hayam Hiba
6848	Adel Yasser

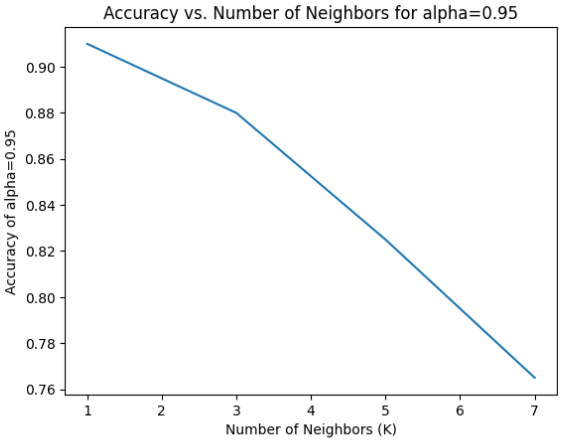
To get started, use google colab, open the notebook(.ipynb file) and upload archive.zip and run all cells.

Main part(Face recognition, 50:50 split):







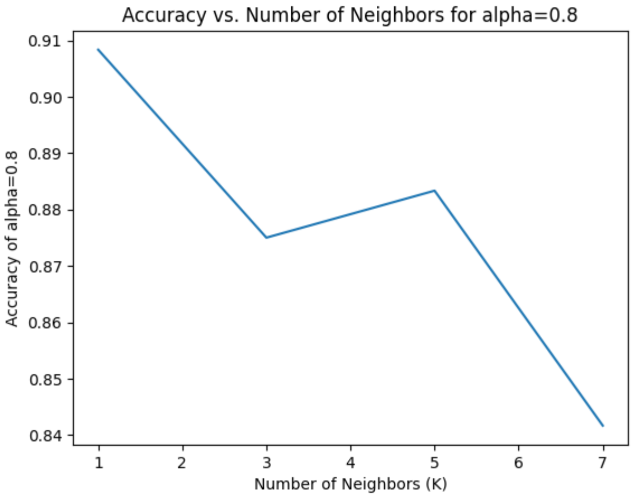


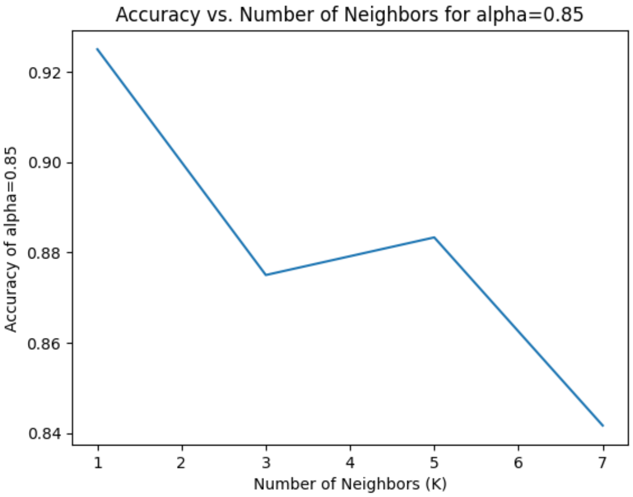
Observation:

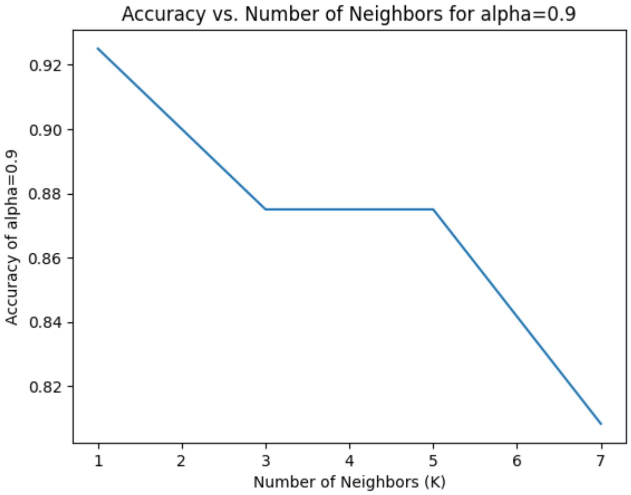
as alpha increases, the performance at higher number of nearest neighbors leads to lower accuracy

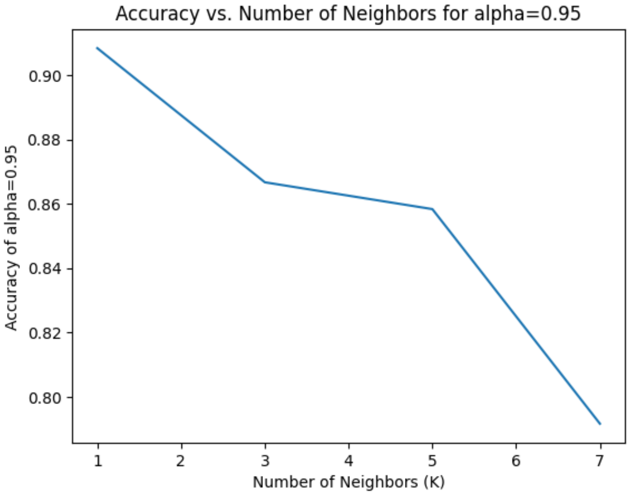
Note: rerunning the code could lead to slightly different results since the data matrix is randomized before splitting into 50% training data and 50% testing data

Bonus part 1: Face Recognition 70:30 split









Observation:

Overall Accuracy Performance is slightly improved over 50:50 split

Conclusion that as more percentage of the dataset is increased, the model's predictions will become more accurate.

Note: we must not keep increasing the ratio of the training data indefinitely as it may lead to the model overfitting and not being able to make good predictions on its own. A good balance between training and test data of data splitting is recommended.

Bonus part 2: Faces vs No Faces

Dataset used: https://www.kaggle.com/datasets/aneeshtickoo/hcaptcha-dataset

Hcaptcha images dataset. Only trucks images were used (most in dataset, over 800 images could be used)

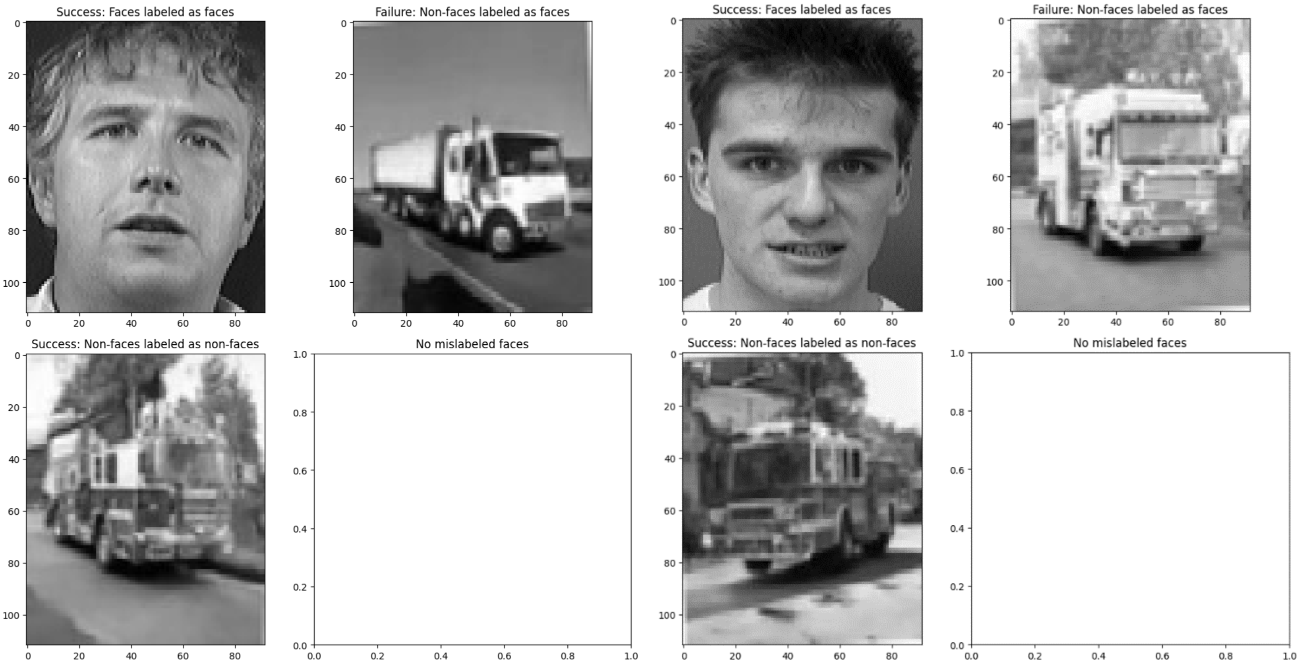
We use only the first 800 images of the trucks images, also included in archive.zip

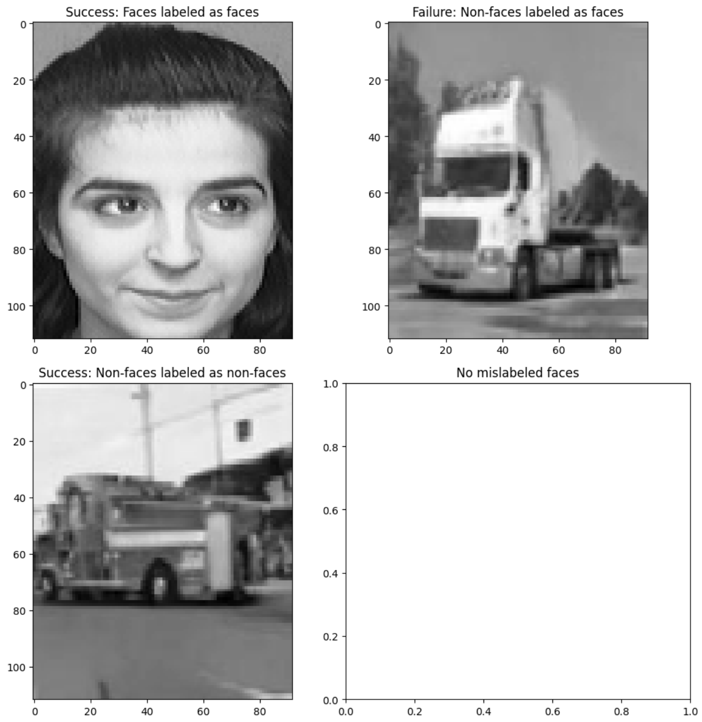
Images were reduced to 92x112 resolution and converted to greyscale as in main dataset

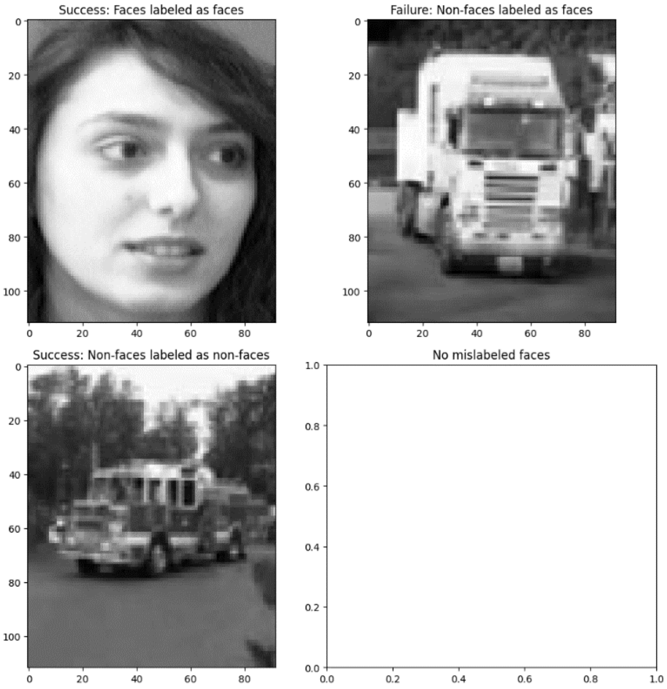
Face images number was 400 and labeled 1.

Non-face(truck) images number was (200,400,800) and labeled 0.

Examples of success vs failure cases:


Success: Faces labeled as faces
Failure: Non-faces labeled as faces
Success: Faces labeled as faces
Failure: Non-faces labeled as faces
Success: Non-faces labeled as non-faces
No mislabeled faces
Success: Non-faces labeled as non-faces
No mislabeled faces
0
20
40
60
80
100
0.0
0.2
0.4
0.6
0.8
1.0





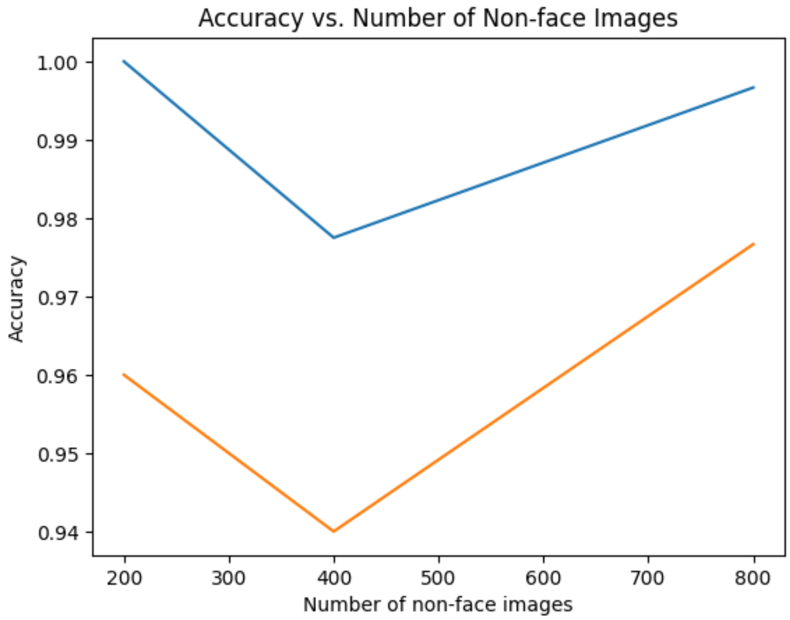
Obersvation:

Extremely high accuracy for classification to faces non faces over all 3 iterations(200 , 400 & 800 non faces vs 400 faces)

Non-faces labeled faces were at 24 and faces labeled non-faces were zero (Despite the dataset positions getting randomized)

Possible Explanations: unlike the ORL AT&T dataset which always consisted of images with dark/dark grey backgrounds and were always profile shots upclose in profile shot position, the truck images were naturally photographed where they are not covering most of the image, have complex backgrounds with details, and heavy pixellation due to compression of images all could lead to a vast difference in pixel values in the data vector between actual faces and non faces and make most points group in two big groups with very small number of outlier points

Accuracy vs number of non-faces



Blue line is maximum accuracy value during which KNN value(1,3,5 or7).

Yellow line is minimum accuracy value

Observation:

Prediction is very accurate (possible explanation answered in the previous question) for both the maximum and minimum line.

Accuracy is at minimum during the time when number of non-faces equals number of faces, which makes sense where at other settings there is more of one group than the other whether faces or non faces. It is more probable for less accuracy when number of faces equals number of non-faces. Also the image positions are randomized before splitting so there could be more of one group in training phase or test phase than there would be in un-randomized split.

iii. Criticize the accuracy measure for large numbers of non-faces images in the training data.

→ large number of non-faces acts similar relatively to large number of faces. Which indicates higher accuracies over relatively equal number of faces and non-faces