Summary

The K-nearest neighbour algorithm, also known as KNN, is a supervised learning classifier which uses proximity to make classifications or predictions about the grouping of an individual data point. Part 1 asks us to implement the K-nearest neighbours (KNN) algorithm from scratch to classify handwritten digits (digits 5 and 6) from a modified version of the Optical Recognition of Handwritten Digits Dataset. The implementation involved calculating distances between data points, performing majority voting for classification, and evaluating the model's performance using cross-validation and test accuracy.

Step 1: Import libraries

- Loaded libraries such as Pandas, NumPy, and PyPlot to help load CSV file datasets

Step 2: Load CSV datasets

- Putting the CSV file under the project directory, data was then loaded utilizing Pandas

Step 3: Implement KNN

- Defined a function called `def euclidean_distance(point1, point2)`, used to calculate the Euclidean distance given its formula:
- Defined a function called `knn_predict(train_inputs, train_labels, test_point, k)`, used to find the nearest K-neighbour

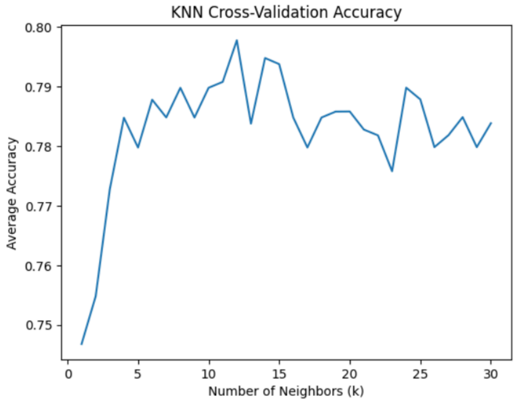
Step 4: 10-fold cross-validation

- Implemented libraries to perform function to evaluate performance for different values of K (from 1 to 30)

Step 5: Determine the best K & test accuracy

- Identified best K and used best K value to train KNN model to extract test accuracy

Results:

1) 

2) The best number of neighbours: 12 has an accuracy of: 0.7977575757575759
3) Test Accuracy: 0.7522935779816514