

Sean McGlincy
Machine Learning
HW 1 – KNN

I had already written the program when you went over how to use vectors in Python. This is my original program with slow for loop but I have removed normalizing data.

The Python program uses numpy to import the data and strip the data labels from the top of the file. The program then calls a driver function 'knn_main' to process each sample of test data. This function aggregates the results and returns the accuracy to the user. I leave the training and test data labels at pos[0] for each vector. The KNN function calculates pos[1]...pos[n].

The 'knn' function processes KNN for each test data point using all of the training data. The function creates an array to aggregate the distances after calculating the distance of each training data sample to the test data. The distance array is sorted and processed for k elements. The results are added to an array which accumulates the results and returns the position with the max.

This process is carried out for each test data sample, which produced 90% accuracy with k=7. To determine the optimal k value I ran through a loop with k ranging from 1...25, increasing by odd number. The global maximum with the smallest k value was when k=7.