**Name: Pranav Shivkumar**

**RUID: 194007405**

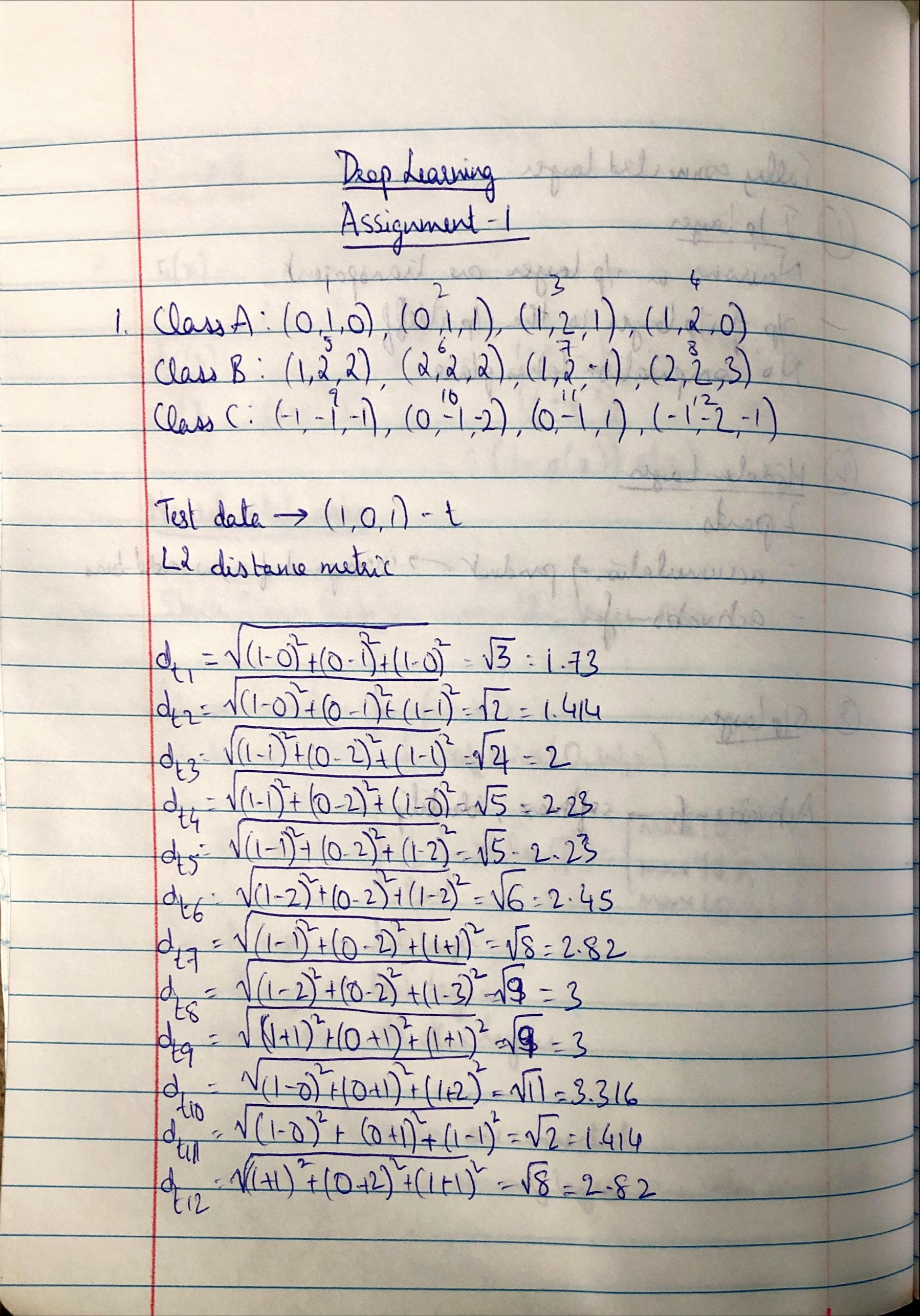
**NetID: ps1029**

**Intro to Deep Learning**

**Assignment 1**

1. **Computation of KNN**

Based on the given data, the computed L2 distances between the test data and the labelled samples are shown below:



1. K = 1

For K = 1, the test data is classified based on the 1 labeled sample to which it is closest. From the distances, the minimum distance is and , each of which are at a distance of 1.414 from the test sample. This implies that the test sample can be either classified into Class A or Class C.

1. K = 2

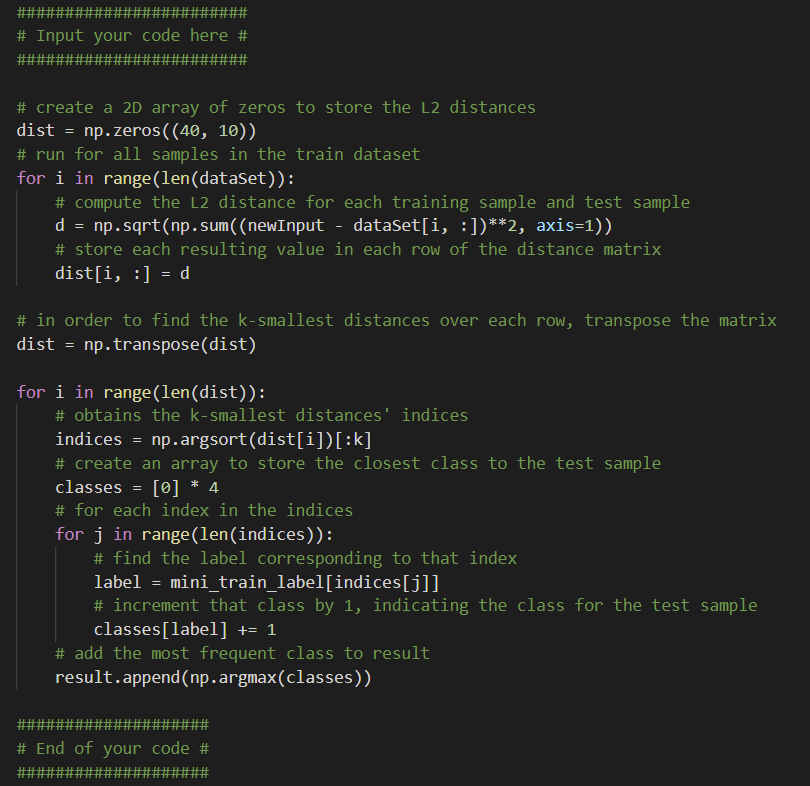
For K = 2, the test data is classified based on the 2 closest labeled samples. Once again, the two closest samples are at a distance of 1.414 from the test sample, so the test sample can be classified into Class A or Class C.

1. K = 3

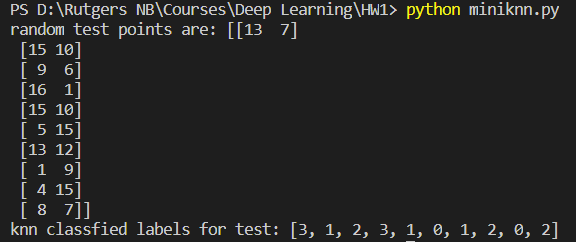
In this case, the test sample is classified based on the 3 closest labelled samples. From the distances, the 3 least distances are , and , with distances of 1.414, 1.414 and 1.73 respectively. Since and correspond to class A and these 2 distances form the majority of the k nearest distances, the test sample will be labelled as Class A.

1. **KNN for simple data**

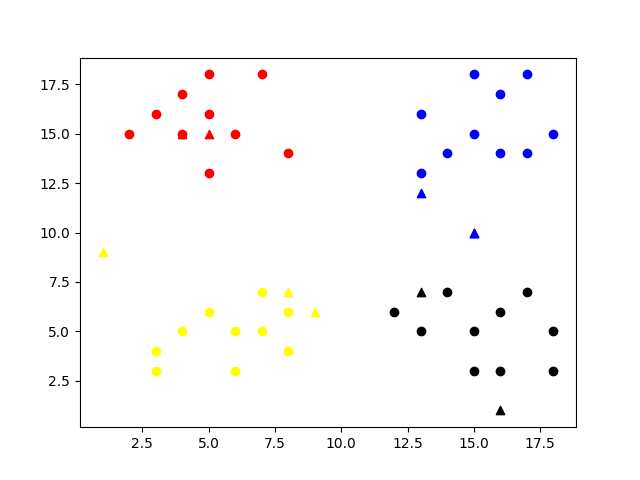
The code for the KNN algorithm is shown below:



The output of the code, with the classified labels for the test samples is shown below:

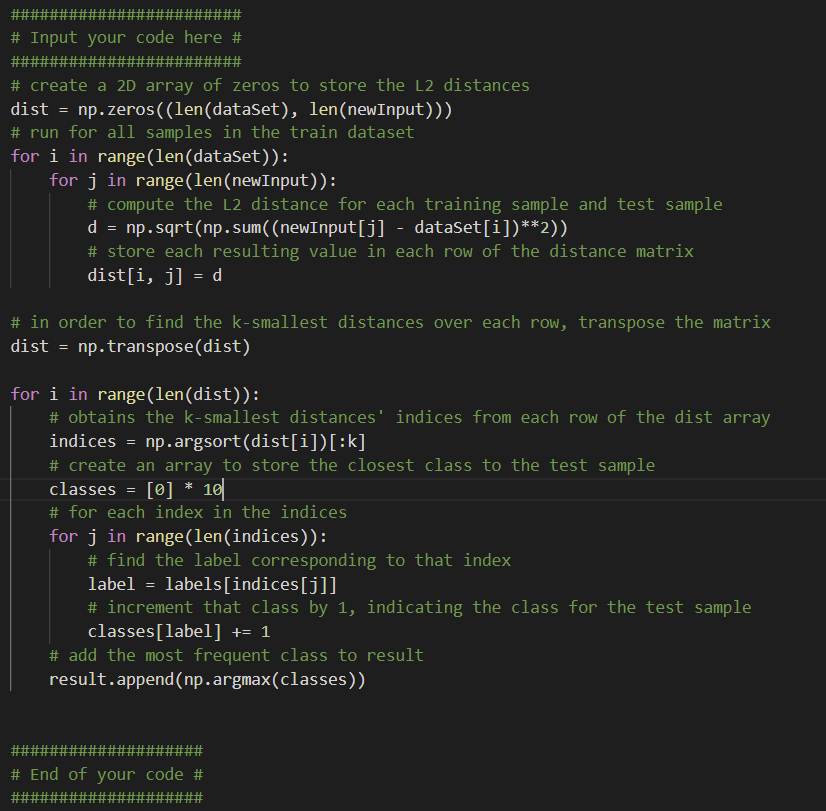


The visualized result of the KNN algorithm for the randomly generated test samples for k=9 is shown below:



1. **KNN for Handwritten Digit Recognition**

The code for the KNN algorithm is shown below:



To determine the number of classes, I found the maximum value of y\_train, which was 9. From this, I was able to deduce that there were 10 classes in which the test sample could be classified into.

On running the KNN algorithm for 25 test samples and k=9, the obtained accuracy was 100% with an execution time of about 15s, as shown in the screenshot below:

