Lam Dang x Anh Dang
CS365: Machine Learning and Artificial Intelligence
Lab E Report

I/ Pick a single random seed and a single (sensible) training set percentage and run k-Nearest Neighbor with a k = 1 on each of the four data sets. What is the accuracy you observed on each data set? Include the terminal commands you used to run each of these in your README.

We pick seed = 12345 and the training set percentage to be 75%

```
python knn.py "iris.csv" 1 0.75 12345
            CONFUSION MATRIX FOR IRIS.CSV
Iris-virginica  Iris-versicolor Iris-setosa
11      1       0       Iris-virginica
0       12      0       Iris-versicolor
0       0       14      Iris-setosa

The accuracy score is 97.36842105263158%
```

```
python knn.py "monks1.csv" 1 0.75 12345
            CONFUSION MATRIX FOR MONKS1.CSV
yes     no
44      6       yes
18      40      no

The accuracy score is 77.77777777777779%
```

```
knn.py "mnist_small.csv" 1 0.75 12345
            CONFUSION MATRIX FOR MNIST_SMALL.CSV
eight   six   two   four   seven   five   one   zero   nine   three
13      0     0     0      0       0      0     0      0      0      eight
0       11    0     0      0       0      0     0      0      0      six
0       0     17    0      0       0      0     0      0      0      two
0       0     0     13     0       0      0     0      2      0      four
0       0     0     0      9       0      0     0      0      0      seven
0       0     0     0      0       13     0     0      1      0      five
0       0     0     0      0       0      9     0      0      0      one
0       0     0     0      0       0      0     10     0      0      zero
1       0     0     0      0       0      0     0      14     0      nine
0       0     0     0      0       0      0     0      0      12     three

The accuracy score is 96.8%
```

---

knn.py "mnist_large.csv" 1 0.75 12345

```
              CONFUSION MATRIX
six    four   nine   eight  seven  five   zero   one    three  two
147    0      0      0      0      0      0      1      0      0      six
0      130    0      0      0      0      0      0      0      0      four
0      1      142    0      1      0      0      0      3      0      nine
0      1      0      130    0      0      0      5      0      0      eight
0      0      0      0      146    0      0      0      0      0      seven
0      0      2      0      0      143    0      0      0      0      five
1      0      0      0      0      0      140    0      0      0      zero
0      0      1      0      0      0      0      129    0      0      one
0      0      1      0      1      0      0      0      142    0      three
0      0      0      0      0      0      0      1      0      137    two
```

The accuracy score is 98.6476868327402%

2/
I conducted a series of test for both mnist_small and mnist_large, using different percentage for training set of each files.
The result is:

- knn.py "mnist_small.csv" 1 0.5 12345: Accuracy = 94%
- knn.py "mnist_small.csv" 1 0.75 12345: Accuracy = 96.8%
- knn.py "mnist_small.csv" 1 0.8 12345: Accuracy = 98%
- knn.py "mnist_large.csv" 1 0.5 12345: Accuracy = 98.398%
- knn.py "mnist_large.csv" 1 0.75 12345: Accuracy = 98.647%
- knn.py "mnist_large.csv" 1 0.8 12345: Accuracy = 98.754%

The average accuracy of mnist_small is smaller than the average accuracy of mnist_large. This is due to the large amount of data points that exist for each categorized value. Those data points provide the classifier more data values to find the nearest neighbor, thus provide more accuracy

3/
I test the mnist_small with three k value: 3, 5, 7

k=3:

```
             CONFUSION MATRIX
eight   six    two    four   seven   five   one    zero   nine   three
13      0      0      0      0       0      0      0      0      0       eight
0       11     0      0      0       0      0      0      0      0       six
0       0      17     0      0       0      0      0      0      0       two
0       1      0      12     0       0      0      0      2      0       four
0       0      0      0      8       0      1      0      0      0       seven
0       0      0      0      0       13     0      0      1      0       five
0       0      0      0      0       0      8      0      1      0       one
0       0      0      0      0       0      0      10     0      0       zero
1       0      0      1      0       0      0      0      13     0       nine
0       0      0      0      0       0      0      0      0      12      three
```

The accuracy score is 93.60000000000001%

----

k = 5:

```
             CONFUSION MATRIX
eight   six    two    four   seven   five   one    zero   nine   three
13      0      0      0      0       0      0      0      0      0       eight
0       11     0      0      0       0      0      0      0      0       six
0       0      17     0      0       0      0      0      0      0       two
0       1      0      13     0       0      0      0      1      0       four
0       0      0      0      8       0      1      0      0      0       seven
0       0      0      0      0       13     0      0      1      0       five
0       0      0      0      0       0      8      0      1      0       one
0       0      0      0      0       0      0      10     0      0       zero
1       0      0      0      0       0      0      0      13     1       nine
0       0      0      0      0       0      0      0      0      12      three
```

The accuracy score is 94.39999999999999%

----

k = 7:

```
             CONFUSION MATRIX
eight   six    two    four   seven   five   one    zero   nine   three
13      0      0      0      0      0      0      0      0      0      eight
0       11     0      0      0      0      0      0      0      0      six
0       0      17     0      0      0      0      0      0      0      two
0       1      0      12     0      0      1      0      1      0      four
0       0      0      0      8      0      1      0      0      0      seven
0       0      0      0      0      13     0      0      1      0      five
0       0      0      0      0      0      8      0      1      0      one
0       0      0      0      0      0      0      10     0      0      zero
1       0      0      1      0      0      0      0      13     0      nine
0       0      0      0      0      0      0      0      0      12     three
```

The accuracy score is 93.60000000000001%

You can see here that the accuracy of k = 3 and k = 7 is less than the accuracy of k = 5. Our speculation is that if you choose a k value too small, the classifier would have a smaller pool of options to consider and would be easy to make mistakes. On the other hand, if the k value is too large, we may smoothing things out too much and eliminating some important details in the distribution.

4/

There is a general formula for choosing a k value that will presumably produce the best accuracy : k = sqrt(n)^(½), with n represent the number of samples in the training set:

Testing:

Monks1.csv: training_set: 75%, seed: 12345, k = 4 (according to formula)

```
            CONFUSION MATRIX
yes    no
45     5      yes
11     47     no
```

The accuracy score is 85.18518518518519%

Comparing to the test in the first problem which is k = 1, this k value definitely improves the accuracy of the test

----

However:

Mnist_small.csv: training_set: 75%, seed: 12345, k = 9 (according to formula)

```
              CONFUSION MATRIX
eight   six    two    four   seven   five   one    zero   nine   three
13      0      0      0      0       0      0      0      0      0      eight
0       11     0      0      0       0      0      0      0      0      six
0       0      17     0      0       0      0      0      0      0      two
0       1      0      11     0       0      1      0      2      0      four
0       0      0      0      8       0      1      0      0      0      seven
0       0      0      0      0       13     0      0      1      0      five
0       0      0      0      0       0      8      0      1      0      one
0       0      0      0      0       0      0      10     0      0      zero
1       0      0      0      0       0      0      0      14     0      nine
0       0      0      0      0       0      0      0      0      12     three
```

The accuracy score is 93.60000000000001%

Comparing to the tests in problem 1 and 3, this k value has smaller accuracy.

So, to answer the question, the formula does not always give us the best accuracy. It still depends on the data we have and the k value we choose.