Andrew Floyd

CS3001 - Intro to Data Science

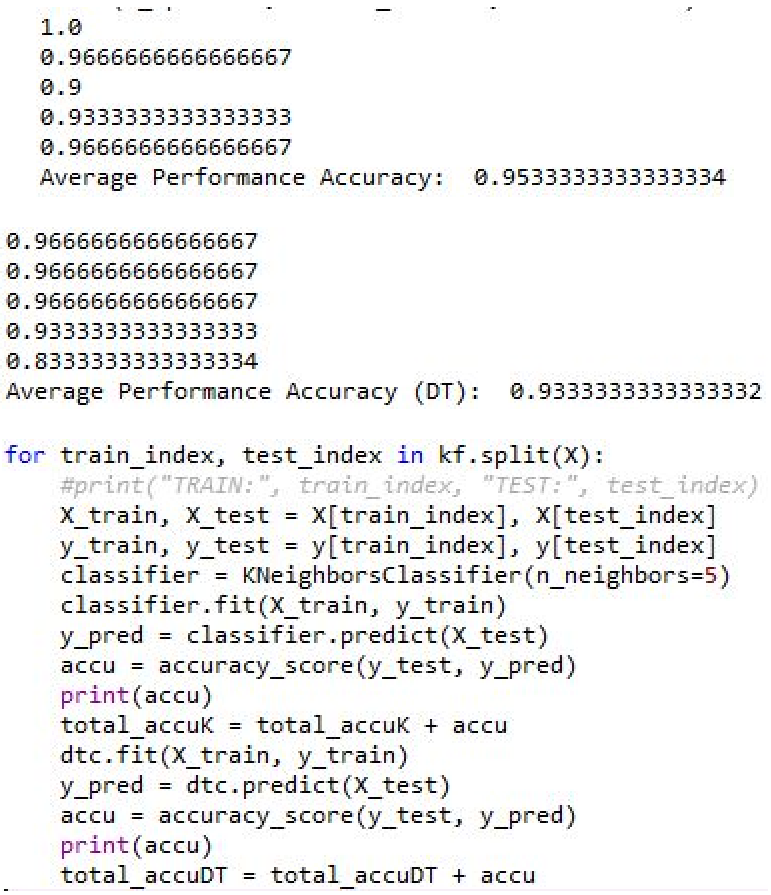
Dr. Fu

November 4th, 2018

HW#5 - kNN

<u>Github repo (for all code and photos):</u> https://github.com/afzm4/cs3001hw5

1. The average accuracy of the kNN (using a default k=5) was: 0.953, while the average accuracy of the Decision Tree testing was 0.933. The full results are pictured below along with the main code used. I decided to combine the testing.py and training.py into one file, with the for-loop shown splitting the dataset into 5 parts and running like specified (5 total runs).

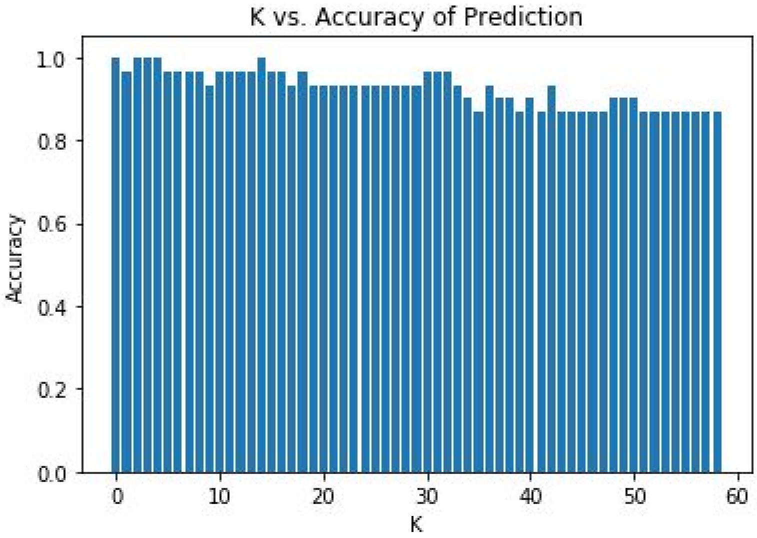


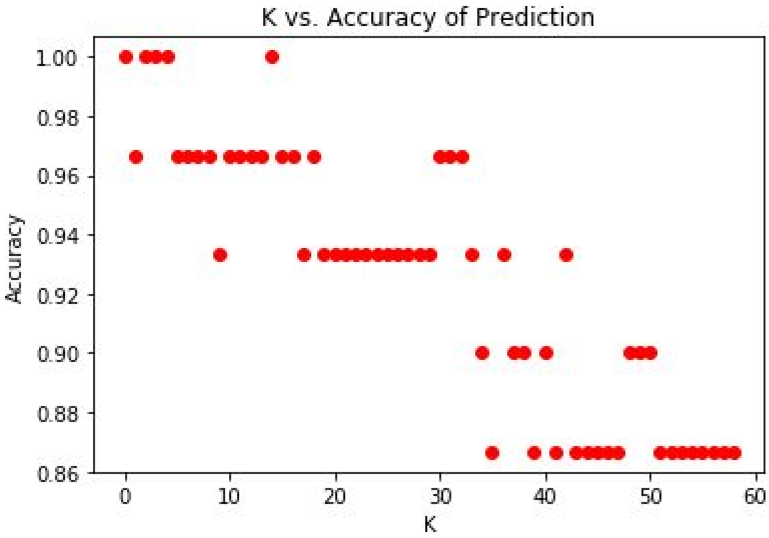
```
 1.0
 0.9666666666666667
 0.9
 0.9333333333333333
 0.9666666666666667
 Average Performance Accuracy:  0.9533333333333334

0.9666666666666667
0.9666666666666667
0.9666666666666667
0.9333333333333333
0.8333333333333334
Average Performance Accuracy (DT):  0.9333333333333332
```

```
for train_index, test_index in kf.split(X):
    #print("TRAIN:", train_index, "TEST:", test_index)
    X_train, X_test = X[train_index], X[test_index]
    y_train, y_test = y[train_index], y[test_index]
    classifier = KNeighborsClassifier(n_neighbors=5)
    classifier.fit(X_train, y_train)
    y_pred = classifier.predict(X_test)
    accu = accuracy_score(y_test, y_pred)
    print(accu)
    total_accuK = total_accuK + accu
    dtc.fit(X_train, y_train)
    y_pred = dtc.predict(X_test)
    accu = accuracy_score(y_test, y_pred)
    print(accu)
    total_accuDT = total_accuDT + accu
```

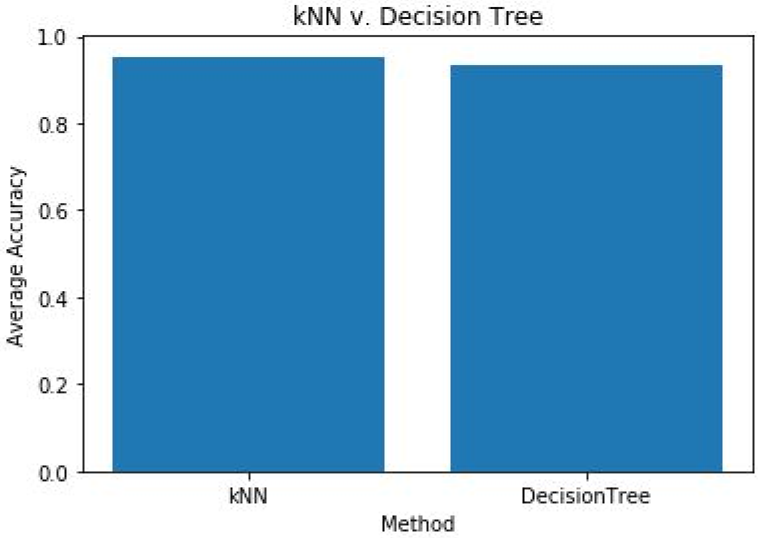
2. For this part, I ran the kNN for k=0 to 60. Below is the histogram for the average accuracies along with a scatter plot showing the same data (I think this is easier to read and interpret then the histogram):





These charts indicated that as the k values increased, the accuracy generally decreased as well. The best k seems to be around 4 or 5.

3. In this histogram, you can see the averages of the kNN with the best k value vs. the averages of the Decision Tree. We see that the kNN is slightly more accurate.


kNN v. Decision Tree
Average Accuracy
kNN
DecisionTree
Method