Assignment 7

Dixitha Kasturi
dkasturi@syr.edu

Topic: KNN, SVM, Random Forests

Dataset: Digits dataset

Outcome: Classify the testset of digits with the appropriate number.

I. Data Processing:

I used the trainset as it is. On observing the columns of the trainset, there were 4198 rows with 785 features which were pixel values(0 when there was no number).The dataset was clean with no missing values. I dropped 127 columns which had only 0's . This means that the whole pixel was 'black' as in 0-255 0 represents black. If the column was only black, I assumed that it could have been a pixel that is outside the shape of the number(not contributing to the number), I did run the experiment with and without these columns and found that the accuracy values did not change. Removing those features only reduced the number of columns that had to be processed. Because columns from the training set were dropped, to match the same number of features, I dropped the columns in the test set as well. Although in the training set 129 columns were present which had all 0 values. I only dropped the ones that were not used in training set.

II. Models

1. KNN:
   The following accuracies were obtained (2 to 9 neighbors were considered and best model was picked based on the accuracy score of the predictions using the test set)

```
[[3, 0.9359218675559791, KNeighborsClassifier(n_neighbors=3)],
 [5, 0.9344926155312053, KNeighborsClassifier()],
 [4, 0.9333015721772272, KNeighborsClassifier(n_neighbors=4)],
 [6, 0.9306812767984755, KNeighborsClassifier(n_neighbors=6)],
 [7, 0.9304430681276799, KNeighborsClassifier(n_neighbors=7)],
 [8, 0.9304430681276799, KNeighborsClassifier(n_neighbors=8)],
 [9, 0.92663172939495, KNeighborsClassifier(n_neighbors=9)],
 [2, 0.9232968080038113, KNeighborsClassifier(n_neighbors=2)]]

Best model number of neighbors =  3
Best model accuracy =  0.9359218675559791
```

2. SVM
   The following accuracies were obtained (4 different kernels(gaussian(rbf), linear, polynomial and sigmoid were considered and best model was picked based on the accuracy score of the predictions using the test set). Gaussian kernel gave the best accuracy

```
[['rbf', 0.9530728918532635, SVC()],
 ['poly', 0.9387803716055264, SVC(kernel='poly')],
 ['linear', 0.9090042877560743, SVC(kernel='linear')],
 ['sigmoid', 0.8492139113863745, SVC(kernel='sigmoid')]]

Best model kernel =  rbf
Best model accuracy =  0.9537875178656503
```

3. Random forests
   The following accuracies were obtained (10 estimators with values from 100 to 1000 varying in steps of 100 were considered and best model was picked based on the accuracy score of the predictions using the test set). The best model accuracy was obtained for 800 estimators.

```
[[800, 0.9435445450214388, RandomForestClassifier(n_estimators=800)],
 [300, 0.9418770843258695, RandomForestClassifier(n_estimators=300)],
 [500, 0.9416388756550739, RandomForestClassifier(n_estimators=500)],
 [900, 0.9416388756550739, RandomForestClassifier(n_estimators=900)],
 [400, 0.9411624583134827, RandomForestClassifier(n_estimators=400)],
 [200, 0.940924249642687, RandomForestClassifier(n_estimators=200)],
 [700, 0.9406860409718913, RandomForestClassifier(n_estimators=700)],
 [600, 0.939018580276322, RandomForestClassifier(n_estimators=600)],
 [100, 0.9349690328727965, RandomForestClassifier()]]

Best model number of estimators =  800
Best model accuracy =  0.9435445450214388
```

Final Observations:

All 3 algorithms gave high accuracy numbers (over 90%). Overall SVM gave the highest (95%) followed by RandomForests(94%) and KNN(93%). The number of features were reduced as stated above with no effect on accuracy.