Problem assignment 10

Problem 1. Bagging and Boosting

In this problem you will have an opportunity to experiment with the bagging and boosting approaches that let you combine multiple classification models into an ensemble with the hope that the ensemble will improve the classification performance.

You have received a bagging and boosting code that implements both schemes and lets you combine them with multiple classification models. The bagging and boosting function code is implemented in files: **Bag\_classifier.m and Boost\_classifier.m.** The inputs and outputs of these functions are the same.

**Part a.** Use the dataset in hw10 train.txt hw10 test.txt files (last columns are class labels) and the code provided to test and compare the performance of the base SVM model, the bagged SVM model and the boosted SVM model, and by reporting their the train and test errors.

**parta.m**

Using base model SVM, I have:

Error on training data: 0

Error on testing data: 0.1286

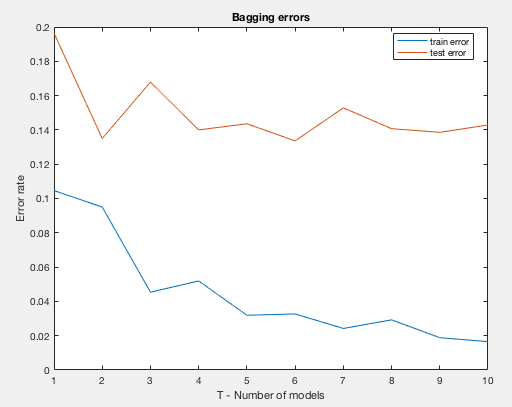
**Error from Bagging with T = 1:10**

train\_error\_T =

[ 0.1046 0.0950 0.0454 0.0519 0.0319 0.0327 0.0242 0.0292 0.0188 0.0165 ]

test\_error\_T =

[ 0.1964 0.1350 0.1679 0.1400 0.1436 0.1336 0.1529 0.1407 0.1386 0.1429 ]



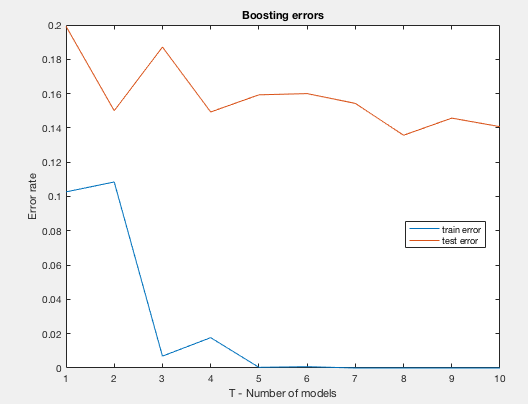
Error from Boosting

train\_error\_T =

[ 0.1027 0.1085 0.0069 0.0177 0.0004 0.0008 0 0 0 0 ]

test\_error\_T =

[ 0.1993 0.1500 0.1871 0.1493 0.1593 0.1600 0.1543 0.1357 0.1457 0.1407 ]



**Part b.** Following the *SVM\_base.m* code definition, write *DT\_base\_full.m* function that implements the decision tree algorithm *without pruning*. Submit the code.

Run and compare the *DT\_base\_full.m* alone, and in combination with the bagging and boosting procedures for T = 2 . . . , 10, similarly to the experiment in Part a.

Plot the graphs showing the train and test errors performances for different T.

Analyze and discuss the results.

*partb.m*

*DT\_base\_full.m*

Using the base tree, the error rate:

misclass\_train = 0.0308

misclass\_test = 0.2571

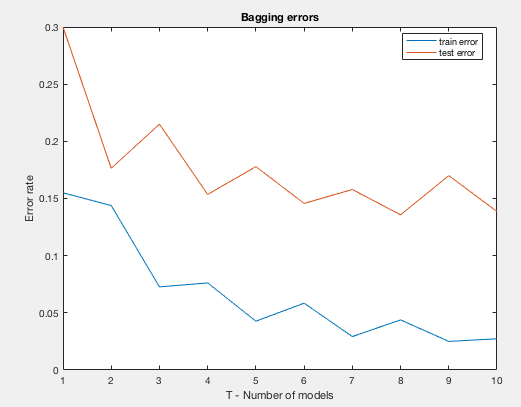
Using bagging with number of models ranging from 1 – 10, the error rates are:

train\_error\_T =

[ 0.1550 0.1438 0.0727 0.0762 0.0427 0.0585 0.0292 0.0438 0.0250 0.0273]

test\_error\_T

[ 0.3000 0.1764 0.2150 0.1536 0.1779 0.1457 0.1579 0.1357 0.1700 0.1386]



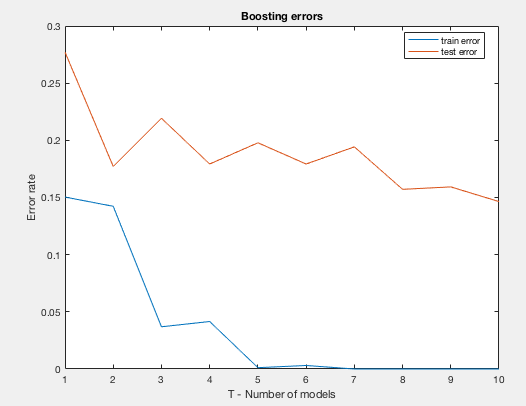
Using the boosting with the number of the models ranging from 1-10:

train\_error\_T =

[ 0.1504 0.1423 0.0369 0.0415 0.0012 0.0031 0 0 0 0 ]

test\_error\_T =

[ 0.2771 0.1771 0.2193 0.1793 0.1979 0. 1793 0.1943 0.1571 0.1593 0.1464 ]



**Part c.** Following the *SVM­­­\_base.m* code definition, write *DT­\_base\_simple.m* function that

uses a decision tree with just one decision node (with one splitting test) that lets you divide

examples into two subpopulations. Submit the code. Run the same set of experiments as in

Parts a and b comparing the base level model to its bagged and boosted version. Analyze

and discuss the results.

*partc.m*

*DT\_base\_simple.m*

Using the base simple tree, the error rate:

misclass\_train = 0.2154

misclass\_test = 0.1429

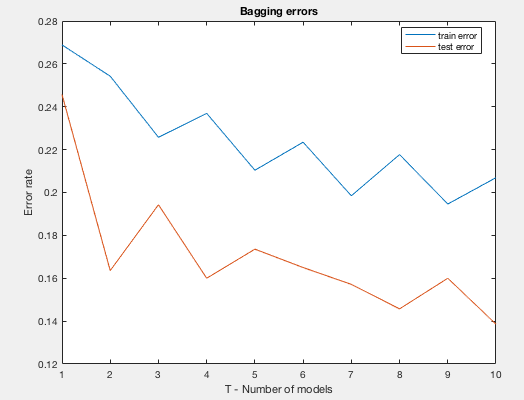
Using bagging with number of models ranging from 1 – 10, the error rates are:

train\_error\_T =

[ 0.2688 0.2542 0.2258 0.2369 0.2104 0.2235 0.1985 0.2177 0.1946 0.2069 ]

test\_error\_T =

[ 0.2457 0.1636 0.1943 0.1600 0.1736 0.1650 0.1571 0.1457 0.1600 0.1386 ]



Using the boosting with the number of the models ranging from 1-10:

train\_error\_T =

[ 0.2627 0.2454 0.2385 0.1992 0.2012 0.1750 0.1654 0.1546 0.1377 0.1254 ]

test\_error\_T =

[ 0.1936 0.1864 0.1814 0.1621 0.1629 0.1621 0.1621 0.1500 0.1557 0.1636 ]

