ADABOOST USING DECISION STUMPS AS WEAK LEARNERS

Username on Miner**: Spydyyy**

Objective:

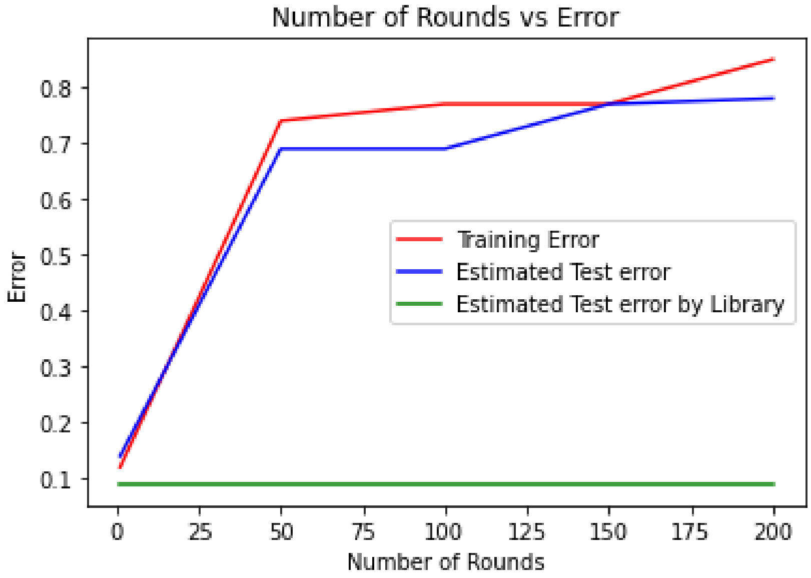
To encode the Adaboost algorithm and a Gini-Index based method for learning decision stumps.

Analyzing Technical Results:

Number of Classifiers	Training Error	Estimated Test Error
1	0.12	0.14
50	0.74	0.69
100	0.77	0.69
150	0.77	0.77
200	0.85	0.78

- We can clearly see that the training error is first minimum at the first round or when trained using 1 – 5 weak classifiers. But when we increase the number of classifiers around 50-100, the model is starting to overfit the training data and hence the training error maximizes.
- By estimating test errors using these classifiers, show that the model is starting to get completely unaware of the classes in the test set and result in mostly predicting the wrong class, when we increase the number of classifiers.
- Also, it takes such a long time to compile and run all the calculations in the code with so many features and examples in the training set and will keep running to overfit the training data more and more as we increase the number of boosting rounds.
- These technicalities show that, the Adaboost algorithm would fit the training dataset perfectly for a few classifiers but will try to fit so many classifiers to the training data when we increase the number of boosting rounds, ultimately resulting in overfitting.
- I tried implementing the model using broadcasting, vector calculations, and loops. Although, it still runs very long and becomes impossible to get any output at a point of time.
- Estimation of Test error using a Single Decision Tree by Sci-kit Learn results in a great Accuracy of 0.91 and a minimal error of 0.09.

Graph:



From the graph above, we can clearly notice that the training error is at the minimum in the first round of boosting, and then hits the peak at 50 rounds of boosting and stays in that range. This implies evidence of overfitting.

The estimated test error also is at the minimum while starting to boost, and eventually raises with the training set error, which is logical.

The green horizontal line denotes the test error estimated by a Single Decision Tree learned using Sci-kit Learn Library, which proves to be an excellent output from the built-in libraries.

Conclusion:

Therefore, by analyzing and examining the above technical results, I successfully implemented the Adaboost algorithm using Decision stumps (learned using the Gini index) as weak classifiers and was able to efficiently test the model on an unseen test dataset.