

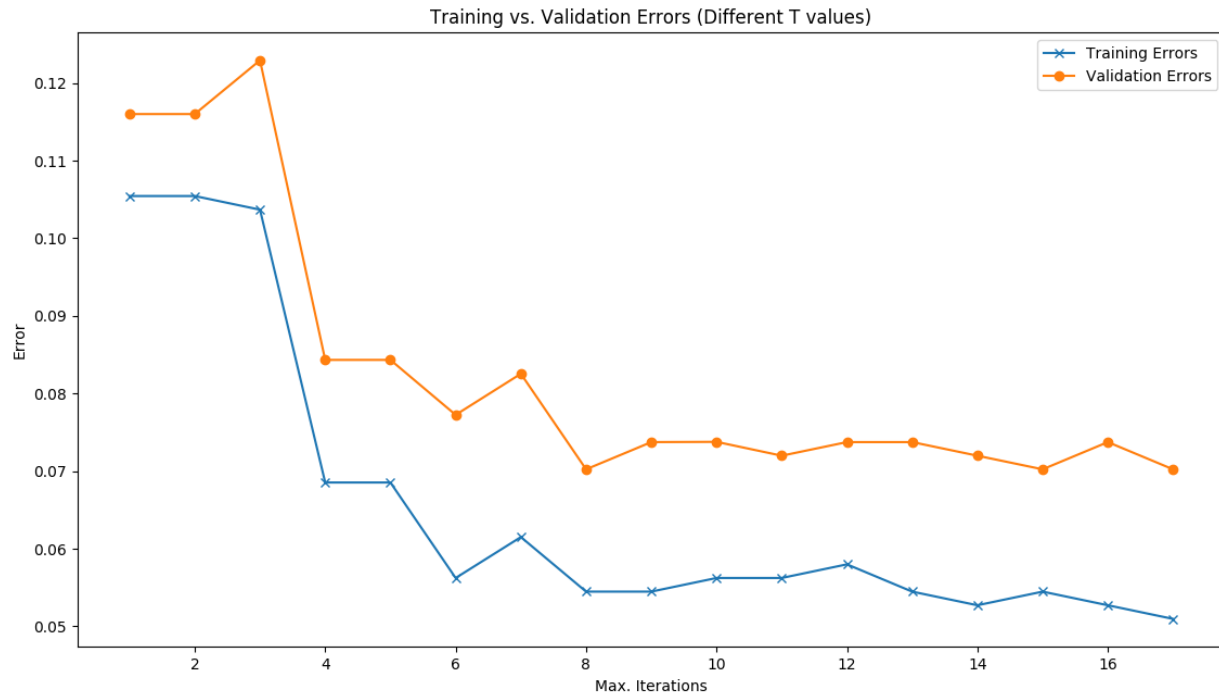
1. Perceptron Performance:-

Linearly Separable Dataset	Loss	Output Hypothesis
1. ERM	0.00	[-10. -40.02891027 20.0112814]
2. Cross-validation	0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.017391 0.000000 Mean Error : 0.000000	[-10. -39.83630621 19.89736048] [-5. -20.47529912 10.16219857] [-9. -36.27687235 18.01580433] [-6. -24.20438331 12.03766337] [-7. -28.42557748 14.15344513] [-6. -24.51203492 12.16531701]
Breast Cancer Dataset		
1. ERM	0.1335676	[2151. , 18878.004 -32605.2200, 35949.0799 -4617.2 -572.72431]
2. Cross-validation	0.087719 0.210526 0.157895 0.122807 0.087719 0.087719 0.070175 0.140351 0.157895 0.107143 Mean Error : 0.133521303	

As the Perceptron Learning Algorithm terminated on a linearly separable dataset with training error = 0.00 for 1000 epochs, it was because it could generate the separating boundary in that data. This was not the case with the Breast cancer dataset where the accuracy came to a minimum but never actually converged(training stopped) because it was not possible to generate a linear decision boundary for this data. This was an interesting observation. So in order to terminate, I had set a max iteration count after which the looping should stop if it has not already converged.

2. Adaboost Performance :

1. ERM - error - 0.10369068
2. Cross-Validation - MEAN error 0.1229323,
Fold errors - [0.122807, 0.175439, 0.017544, 0.105263, 0.140351, 0.210526, 0.175439,
0.105263, 0.105263, 0.071429]
3. Output Hypothesis: Very long vector, please refer to printing in code output for it.



For at least $T = 15$ rounds of boosting the validation error starts to kink and increase. Therefore, at least 15 min rounds of boosting will be needed in order to stabilize the result.

Running the code instructions:

For perceptron.py : `python perceptron.py --dataset [data path] --mode [erm, cv]`

1. `python3 perceptron.py --dataset linearly-separable-dataset.csv --mode erm`
2. `python3 perceptron.py --dataset linearly-separable-dataset.csv --mode cv`
3. `python3 perceptron.py --dataset Breast_cancer_data.csv --mode erm`
4. `python3 perceptron.py --dataset Breast_cancer_data.csv --mode cv`

For adaboost.py : `python adaboost.py --dataset [data path] --mode [erm, cv, plots]`

1. `python3 adaboost.py --dataset Breast_cancer_data.csv --mode erm`
2. `python3 adaboost.py --dataset Breast_cancer_data.csv --mode cv`