HW 3: Boosting
Advanced Machine Learning

Prof: Yannet Interian

Due: Feb 11th 2020

For this assignment submit **hw3.ipynb** to github.

1 AdaBoost on a toy dataset (5 points)

Now we will apply AdaBoost to classify a toy dataset. The dataset consists of 4 points: $(x^{(1)} = (0, -1), y^{(1)} = -1), (x^{(2)} = (1, 0), y^{(2)} = 1), (x^{(3)} = (-1, 0), y^{(3)} = 1)$ and $(x^{(4)} = (0, 1), y^{(4)} = -1)$. You may want to use python or R as a calculator rather than doing the computations by hand but you don't have to submit your code.

1. (3 points) For M = 4 (use 4 trees), show how Adaboost works for this dataset, using simple decision stumps (depth-1 decision trees that simply split on a single variable once) as weak classifiers. For each timestep fill the following table:

m	w_1	w_2	w_3	w_4	err	α	$T_m(x^{(1)})$	$T_m(x^{(2)})$	$T_m(x^{(3)})$	$T_m(x^{(4)})$
1										
2										
3										
4										

2. (1 points) What is the training error of AdaBoost for this toy dataset? (show me the computation)

3. (1 points) Is the above dataset linearly separable? Explain why AdaBoost does better than a decision stump on the above dataset.

Note: In the Adaboost algorithm all log functions are **natural** log and not $\log 10$.

2 Implement AdaBoost (10 points)

For this exercise you will implement AdaBoost from scratch and applied it to a spam dataset. You will be classifying data into spam and not spam. You can call DecisionTreeClassifier from sklearn to learn your base classifiers.

Write a program implementing AdaBoost with trees using the template and tests given to you (hw3.ipynb). Hint: Here is how you train a decision tree classifier with weights.

```
h = DecisionTreeClassifier(max_depth=1, random_state=0)
h.fit(X, Y, sample_weight=w)
```

3 Implement Gradient Boosting for MSE (15 points)

Implement gradient boosting for "rent-ideal.csv" dataset.

1. (10 points)Write a program implementing gradient boosting from MSE. Use the template given on hw3.ipynb.
2. (2 points) Fix the Shrinkage to 0.1. Apply gradient boosting to your dataset using different values for the number of trees $numTrees$. How do you find the best value for $numTrees$? Report train and validation R^2 for the best value of $numTrees$. Make a plot that shows your experiment (training and validation metric as a function of the number of trees). Try as least 2000 trees.
3. (3 points) Compare your results with the results of running the gradient boosting package (XGBoost). Make a plot that shows the result of your experiments. Explore the hyper parameters given in the package.