NetID : iv447
University ID: N17385760

PART A) BOOSTING :

Boosting is a machine learning algorithm whose prime focus is to reduce bias and also variance in supervised learning. Boosting is based on the idea : can a set of weak learners create a single strong learner. A weak learner is a classifier that is only slightly correlated with the true classification. In contrast, a strong learner is a classifier that is arbitrarily well-correlated with the true classification. Boosting can be done for binary categorization or multi class categorization :

Boosting for Binary Categorization : Example in this category can be : faces v/s background in images. We can run the algorithm in following steps :;

1. First form a large set of simple features.
2. Initialize weights for training images.
3. For N rounds :
   - Normalize the weights
   - For available features from the set, train a classifier using a single feature and evaluate the training error.
   - Choose the classifier with the lowest error
   - Update the weights of the training images : increase if classified wrongly by this classifier, decrease if correctly
4. From the final strong classifier as the linear combination of the N classifiers (coefficient larger if training error is small)

Boosting for Multi class categorization : Compared with binary categorization, multi class categorization looks for common features that can be shared across the categories at the same time. They turn to be more generic edge like features. During learning, the detectors for each category can be trained jointly.

Gradient boosting machine learning technique for regression problems is a prediction model in the form of an ensemble of weak prediction models. Gradient boosting decision trees can use decision tree as the weak prediction model in gradient boosting. It is highly accurate. The general idea is additive training. At each iteration, a new tree learns the gradients of the residuals between the target values and the current predicted values, and then the algorithm conducts gradient descent based on the learned gradients. Algorithm is :

Input : training set : {(x (i), y(i)} i = 1 to n, a differentiable loss function :
L(y, F(x)), number of iterations : m.

Algorithm :

1. Initialize model with constant value.
2. For m = 1 to M :
   - Compute Pseudo residuals
   - Fit a base learner to pseudo residuals i.e train it using training set.
   - Compute multiplier by solving one dimensional optimization problem.
3. Output

We can only parallelize the algorithm in the tree building step. Therefore, the problem reduces to parallel decision tree building.

PART B)
Three ways to improve course from student’s perspective :

Over all this course was very helpful in learning new concepts for r, mongo and spark (in java). These things can add up more learning if possible to incorporate :

1. Learning the concepts in python along with java.
2. Few assignments on sql, r and mongo.
3. Few more tutorials on introduction to data visualization.