## Machine Learning Lab 9

# Ensemble Learning

Ensemble learning is the process by which multiple models, such as classifiers or experts, are strategically generated and combined. Ensemble learning is primarily used to improve the (classification, prediction, function approximation, etc.) performance of a model, or reduce the likelihood of an unfortunate selection of a poor one. Some commonly used ensemble learning techniques are bagging, boosting, stacking, random forest, gradient boosting methods and voting.

## The dataset

The dataset used to perform this experiment is the wine quality dataset, it is a combination of data on two types of wine variants, namely red wine and white wine, of the portuguese “Vinho Verde” wine. The dataset contains information on the parameters for fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, alcohol.

## Experiment

In this experiment I used the sklearn’s decision tree, random forest classifier and gradient boosted tree algorithms to predict the quality of a wine and compared their performance.

Using the pandas library in I loaded the red wine and white wine datasets into the memory from their respective csv files and then merged the two datasets into one single pandas dataframe.

Using the pandas.Dataframe.describe() function in pandas I calculated the various statistical measures of each of the columns of the dataset.

I started with a decision tree with the training data containing all the 11 features from the dataset, the tree used gini index and has a max depth of 5. Next I used a gradient boosted tree with a max depth of 5 and finally the last model I used was random forest again with the max depth of five.

After training all the models over the training data I calculated the precision, recall, Fscore and support for all the algorithms using the sklearn’s function to calculate all of the above together.

I plotted the above score on a bar graph and it is evident from the graphs that GBM trees perform the best, followed by random forests and the decision trees.

The code and plots can be found in the accompanying jupyter notebook.