# Bagging Report

- Setup
    - A training file is taken as an input and a decision tree is built from that.
    - Next, the number of bags to be created is taken as an input.
    - Each bag is created by bootstrapping the training file i.e. randomly selecting training instances from the training file with replacement.
    - Every bag has exactly as many training instances as the training file has and a decision tree is built per bag.
    - Once building decision tree is done the test file is taken as an input.
    - Every test instance is run on each bag's decision tree and the output class label is noted down.
    - After every bag's decision tree is run we get the vote for class label which has got the highest count and that becomes our output class label for the given test instance.
- Result
    - We find out the accuracy of test data on just training file and then find out accuracy of data on entire bag.
    - For example for one of the training and test file we recorded following results –
        - Accuracy of test file with train data ( 216 instances ) = 82.87
        - Accuracy of test file with bag data ( 216 instances, 50 bags ) = 93.22
    - Therefore we see that by bagging we are getting an increased accuracy.
- Observation
    - With small bag size such as 10 – 15 the accuracy was different on different run as the randomness in selecting training instance played vital role and in some cases even error in training file was considered in bags.
    - But as the bag size was increased to 50 – 100 the accuracy increased as well as constant.