Random Forest Demonstration in R

Vinh Nguyen

**Abstract**—The purpose of this documentation is to demo how Random Forest algorithm works in R

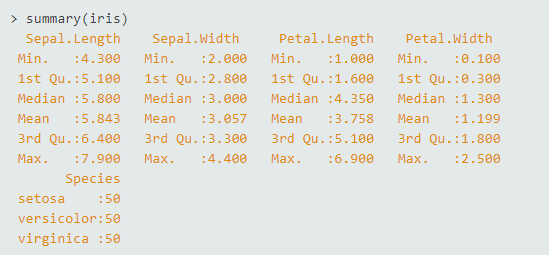
**Index Terms**—Randome Forest, algorithm, R, bootstraping

xxxx-xxxx/0x/$xx.00 © 200x IEEE Published by the IEEE Computer Society

—————————— ◆ ——————————

# 1 Dataset

I

n this experiment, we use a very popular dataset for machine learning, that is, the IRIS dataset introduced by Fisher. This dataset contains 150 observations for three species of flowers (setosa, versicolor, and virginica) with 4 variables (Sepal.Length, Sepal.Width, Petal.Length, Petal.Width)

# 2 Data processing

We split our dataset in two parts: training part and testing part with the following commands

data\_set\_size <- floor(nrow(iris)/2)

indexes <- sample(1:nrow(iris), data\_set\_size)

training <- iris[indexes,]

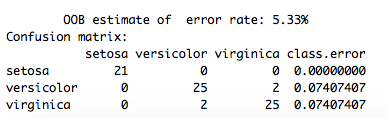
test <- iris[-indexes,]

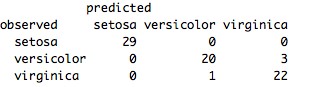
# 3 Running algorithm

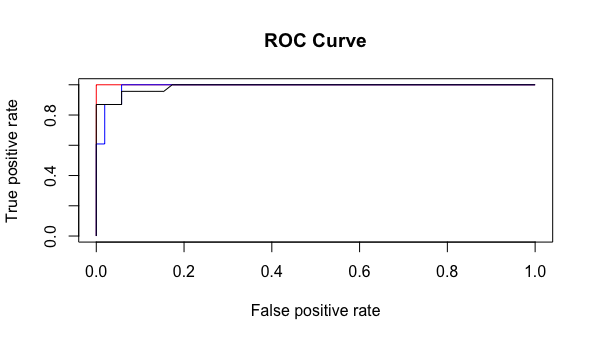
rf\_classifier <- randomForest(Species ~., data = training, ntree=100, mtry=2, importance=TRUE)

In the command above, we have 5 parameters. The first parameter (Species ~.) indicates that our target class is Species in which we need our algorithm have to classify, the dot (.) shows that we will put all other variables/features into training, data=training points out the training data which we created in the previous step. ntree = 100 shows that we will create 100 random trees, mtry = sqrt (features/variables) = sqrt (4) = 2, important = TRUE means that we want to output the important value of variables.

# 4 Result



The result shows that our out-of-bag error rate is 5.33% = 4/75 as depicted in the confusion matrix when training the model.

The figure above shows the confusion matrix with testing data, the error rate is the same compared to training data.

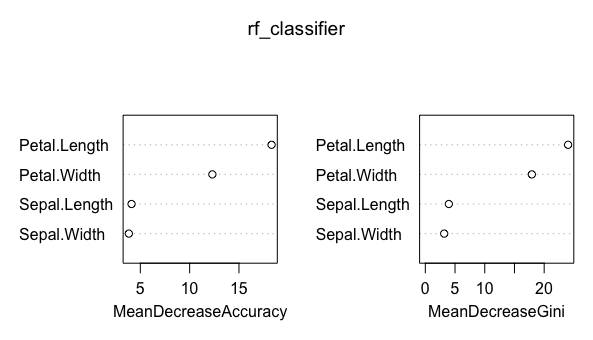
The ROC curve:

Red: setosa

Blue: versicolor

Black: virginica

It can quickly be seen that setosa is the ideal case with 100 accuracy, followed by virginica and versicolor.

The figure below shows the importance of each variable, it can be seen that Petal.Length plays the most important role for classification task.

————————————————

1. Vinh Nguyen. Email: vinh.nguyen@ttu.edu