

Random Forest Algorithm Report

Donal Loitam

July 19, 2022

Contents

1 Introduction	1
2 Key Points of the Algorithm	2
3 Some Questions	3

1 Introduction

- Random forest is a Supervised Machine Learning Algorithm that is used widely in Classification and Regression problems.
- It builds decision trees on different samples and takes their majority vote for classification and average in case of regression.
- **Has higher accuracy than Decision trees though built out of Decision Trees** : as it is based on the concept of **ensemble learning**, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.
- As it relies not only on one Decision Tree but greater number of trees in the forest it yields higher accuracy and solves the problem of overfitting.

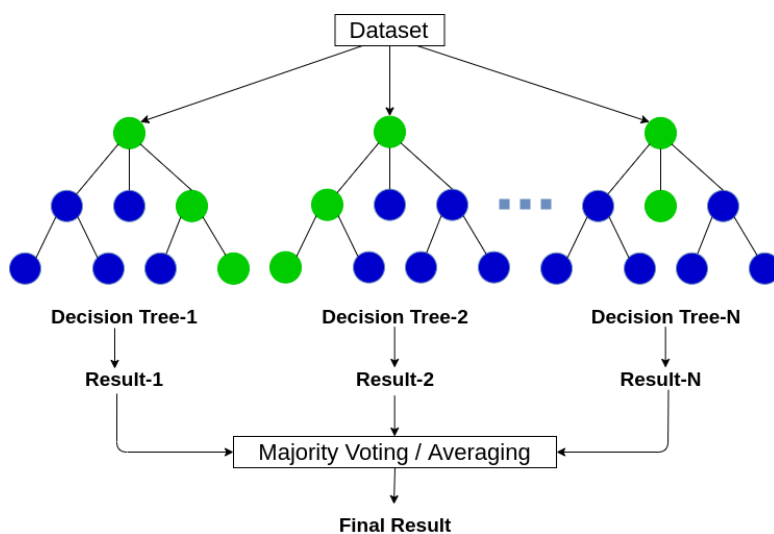


Figure 1: Ensemble learning : Working of a Random Forest

2 Key Points of the Algorithm

The fundamental concept behind random forest is a simple but powerful one — the wisdom of crowds. **“A large number of relatively uncorrelated models(trees) operating as a committee will outperform any of the individual constituent models.”**

The low correlation between models is the key.

The reason why Random forest produces exceptional results is that the trees protect each other from their individual errors.

Steps of the algorithm

Step 1: Bagging or Bootstrap Aggregation

The random forest allows each individual tree to randomly sample from the dataset with **replacement** (allowing repetition), resulting in different trees. This process is called Bagging.

NOTE : Here, we are not subsetting the training data into smaller chunks and training each tree on a different chunk. Rather, if we have a sample of size N , we are still feeding each tree a training set of size N . But instead of the original training data, we take a random sample of size N with replacement.

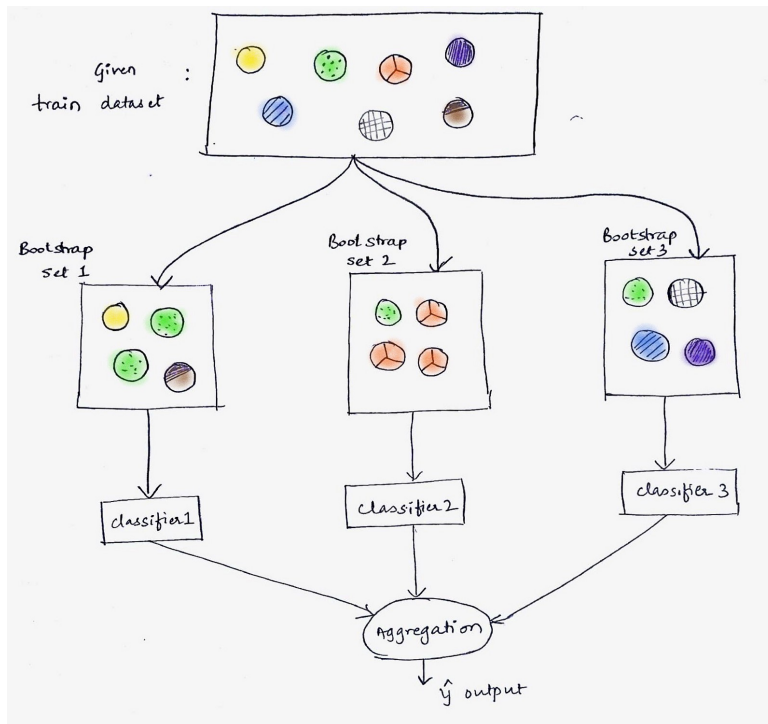


Figure 2: Bagging(with replacement)

Step 2: Random feature selection

In a normal decision tree, when it is time to split a node, we consider every possible feature and pick the one that separates the best. While in contrast, each tree in a random forest can pick only from a random subset of features. This ultimately results in low correlation across trees. And we just build the tree as usual, but only considering a random subset of variables at each step

We built a tree using

1. Using a bootstrapped dataset (N)
2. Only consid a random subset of variables/features (' k ') at each step.

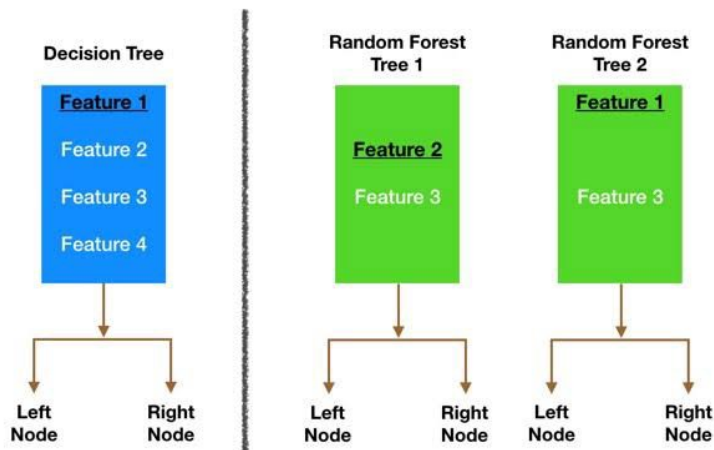


Figure 3: Bagging(with replacement)

3 Some Questions

1. Write a Pseudo Code for the Random Forest algorithm

Ans: Random Forest creation pseudocode:

- Randomly select “k” features from total “m” features where $k \ll m$
- Among the “k” features, calculate the node “d” using the best split point
- Split the node into daughter nodes using the best split
- Repeat the 1 to 3 steps until some “max_num” number of nodes has been reached
- Build forest by repeating steps 1 to 4 for “n” number times to create “n” number of trees.

2. What is Out-of-Bag Error?

Ans.

3. Why does the Random Forest algorithm not require split sampling methods?

Ans

4. Prove that in the Bagging method only about 63% of the total original examples (total training set) appear in any of sampled bootstrap datasets. Provide proper justification.

Ans.

5. How does random forest define the Proximity (Similarity) between observations?

Ans