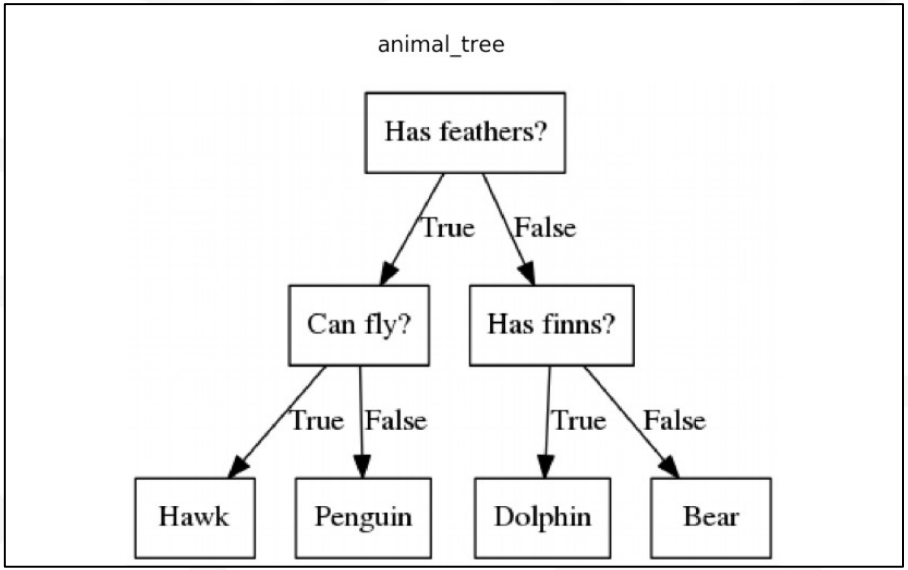
Random Forests

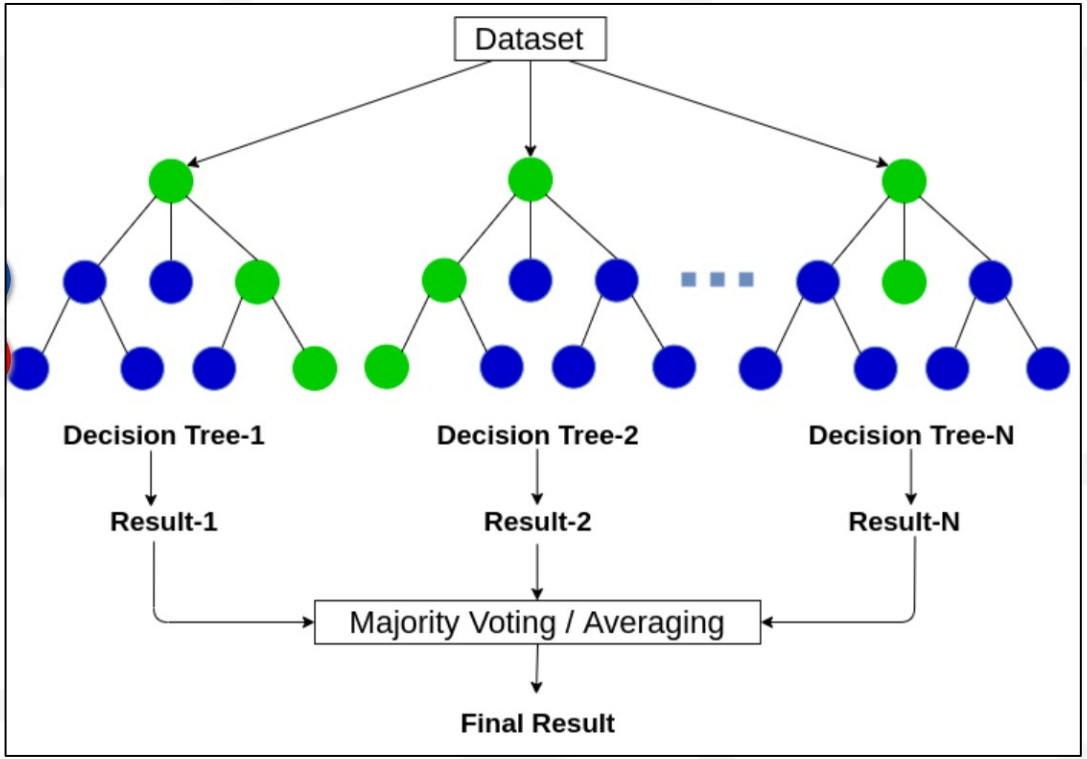
Basic Decision Tree

- Recall that basic decision trees have **high variance** - the results are sensitive to the data we have in the training set.
- By creating multiple decision trees and aggregating the results, we can reduce the variance. This is the idea behind random forests.



Random Forest

- Creates multiple decision trees – each taking a different subset of the data
- When a data instance is presented to be predicted each tree is followed to a terminal node
- Then the results are found by:
  - Voting for classification
  - Averaging for regression

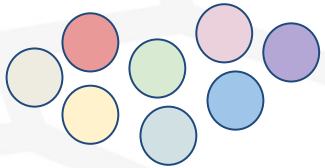


How Do Random Forests Work?

1. Draw a random bootstrap sample of size *n*, where *n* is the sample size of your training data.
2. For each split in the tree, randomly select *m* predictors to be split candidates. *m* is typically chosen to be the square root of the total number of predictors.
3. Repeat steps 1-2 multiple times.
4. Aggregate results to get the prediction.

What is Bootstrapping?

Bootstrapping is a very popular technique in data science and statistics. It works by simply taking a sample of size *n* (where *n* is the size of your original sample) **with replacement**.



Original Data

Bootstrap Sample 1

Bootstrap Sample 2

Bootstrap Sample 3

Decision Tree 1

Decision Tree 2

Decision Tree 3

Pros and Cons

Pros

- Same benefits as decision trees - can be used with:
  - Continuous & categorical predictors
  - Continuous & categorical response
- Robust to outliers.
- Works well with non-linear data.
- Lower risk of overfitting.
- Runs efficiently on a large dataset.
- Good accuracy generally.
- Can get feature importance.

Cons

- Takes longer to run
- Not as easily interpretable as decision trees or linear regression
- Considered more of a ‘black box’ approach.

https://towardsdatascience.com/pros-and-cons-of-various-classification-ml-algorithms-3b5bfb3c87d6