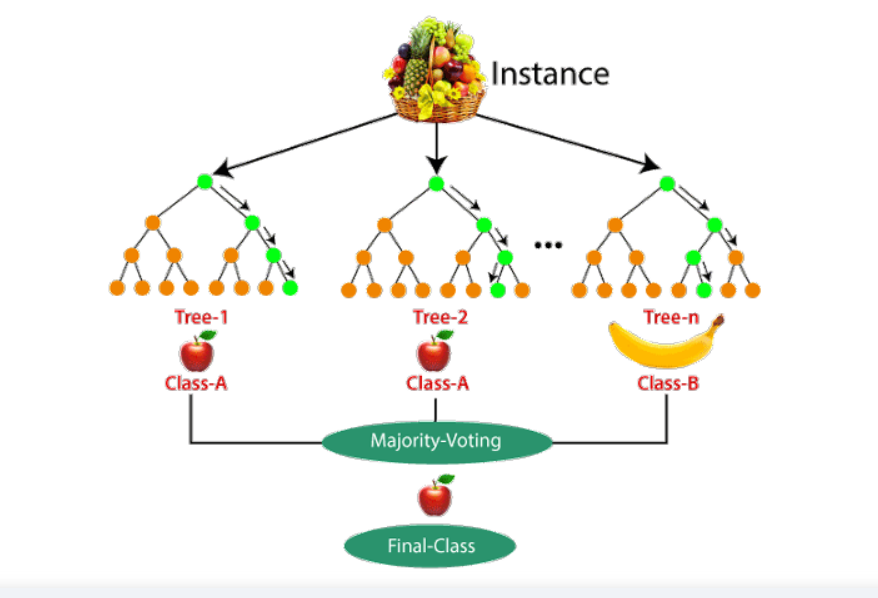
**Random Forest**

1. **Introduction**
   1. Random Forest is a widely-used machine learning algorithm developed by Leo Breiman and Adele Cutler, which combines the output of multiple decision trees to reach a single result. It tackle both classification and regression problems effectively
2. **Working of Random Forest Algorithm:**
   1. Random Forest is based on the Bagging technique, which combines multiple decision trees.
   2. Bagging involves selecting random subsets of data with replacement, training individual models, and combining their results through majority voting.
3. **Steps Involved in Random Forest Algorithm:**
   1. Random Forest selects subsets of data points and features for each decision tree.
   2. Each decision tree is constructed independently.
   3. The final output is determined by majority voting for classification or averaging for regression.
4. **Important Features of Random Forest:**
5. **Diversity**: Not all attributes/variables/features are considered while making an individual tree; each tree is different.
6. **Immune to the curse of dimensionality:** Since each tree does not consider all the features, the feature space is reduced.
7. **Parallelization:** Each tree is created independently out of different data and attributes. This means we can fully use the CPU to build random forests.
8. **Train-Test split:** In a random forest, we don’t have to segregate the data for train and test as there will always be 30% of the data which is not seen by the decision tree.
9. **Stability:** Stability arises because the result is based on majority voting/ averaging.
10. **Difference Between Decision Tree and Random Forest:**
    1. Decision trees can suffer from overfitting, while Random Forest mitigates this issue by aggregating results.
    2. Decision trees are faster in computation compared to Random Forest.
11. **Important Hyperparameters in Random Forest:**
    1. Hyperparameters to increase the Predictive Power
       1. **n\_estimators:** Number of trees the algorithm builds before averaging the predictions.
       2. **max\_features:** Maximum number of features random forest considers splitting a node.
       3. **mini\_sample\_leaf:** Determines the minimum number of leaves required to split an internal node.
       4. **criterion:** How to split the node in each tree? (Entropy/Gini impurity/Log Loss)
       5. **max\_leaf\_nodes:**Maximum leaf nodes in each tree
    2. Hyperparameters to increase the speed
       1. ***n\_jobs:***it tells the engine how many processors it is allowed to use. If the value is 1, it can use only one processor, but if the value is -1, there is no limit.
       2. ***random\_state:***controls randomness of the sample. The model will always produce the same results if it has a definite value of random state and has been given the same hyperparameters and training data.
       3. ***oob\_score:****OOB* means out of the bag. It is a random forest cross-validation method. In this, one-third of the sample is not used to train the data; instead used to evaluate its performance. These samples are called out-of-bag samples.
12. **Random Forest Algorithm Use Cases:**
    1. Random Forest finds applications in various domains, including E-commerce, banking, medicine, and the stock market.
    2. It can be used for tasks such as predicting loan defaults in the banking industry.
13. **Advantages and Disadvantages of Random Forest Algorithm:**
14. Random forest algorithm can be used to solve both classification and regression problems.
15. It is considered as very accurate and robust model because it uses large number of decision-trees to make predictions.
16. Random forests takes the average of all the predictions made by the decision-trees, which cancels out the biases. So, it does not suffer from the overfitting problem.
17. Random forest classifier can handle the missing values. There are two ways to handle the missing values. First is to use median values to replace continuous variables and second is to compute the proximity-weighted average of missing values.
18. Random forest classifier can be used for feature selection. It means selecting the most important features out of the available features from the training dataset.

The disadvantages of Random Forest algorithm are listed below:-

1. The biggest disadvantage of random forests is its computational complexity. Random forests is very slow in making predictions because large number of decision-trees are used to make predictions. All the trees in the forest have to make a prediction for the same input and then perform voting on it. So, it is a time-consuming process.
2. The model is difficult to interpret as compared to a decision-tree, where we can easily make a prediction as compared to a decision-tree.
3. **Conclusion:**
   1. Random Forest is a fast and efficient model known for its ability to handle missing values.
   2. It's versatile and widely used in different industries due to its high performance and robustness.
4. **Key Takeaways:**
   1. Random Forest is an ensemble learning technique that combines multiple decision trees for improved performance.
   2. It works for both classification and regression problems.



Consider the fruit basket as the data as shown in the figure above. Now n number of samples are taken from the fruit basket, and an individual decision tree is constructed for each sample. Each decision tree will generate an output, as shown in the figure. The final output is considered based on majority voting. In the below figure, you can see that the majority decision tree gives output as an apple when compared to a banana, so the final output is taken as an apple.

Reference :- [Random Forest](https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/)