Random Forest

Random forest is a supervised learning algorithm. It has two variations – one is used for classification problems and other is used for regression problems. It is one of the most flexible and easy to use algorithm. It creates decision trees on the given data samples, gets prediction from each tree and selects the best solution by means of voting. It is also a pretty good indicator of feature importance.

Random forest algorithm combines multiple decision-trees, resulting in a forest of trees, hence the name `Random Forest`. In the random forest classifier, the higher the number of trees in the forest results in higher accuracy.

Random Forest algorithm intuition

Random forest algorithm intuition can be divided into two stages.

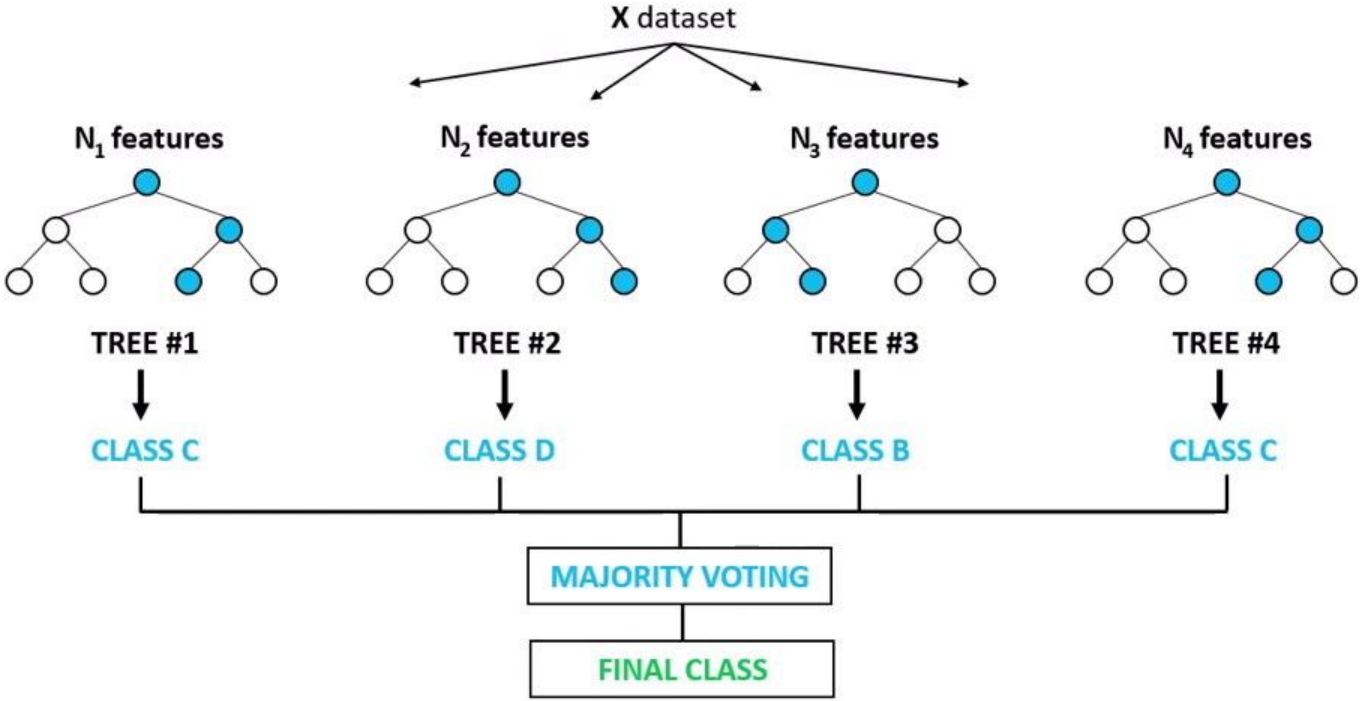
In the first stage, we randomly select “k” features out of total m features and build the random forest. In the first stage, we proceed as follows:-

1. **Randomly select k features from a total of m features where k < m.**
2. **Among the k features, calculate the node d using the best split point.**
3. **Split the node into daughter nodes using the best split.**
4. **Repeat 1 to 3 steps until l number of nodes has been reached.**
5. **Build forest by repeating steps 1 to 4 for n number of times to create n number of trees.**

In the second stage, we make predictions using the trained random forest algorithm.

1. **We take the test features and use the rules of each randomly created decision tree to predict the outcome and stores the predicted outcome.**
2. **Then, we calculate the votes for each predicted target.**
3. **Finally, we consider the high voted predicted target as the final prediction from the random forest algorithm.**

Random Forest Classifier



Feature selection with Random Forests

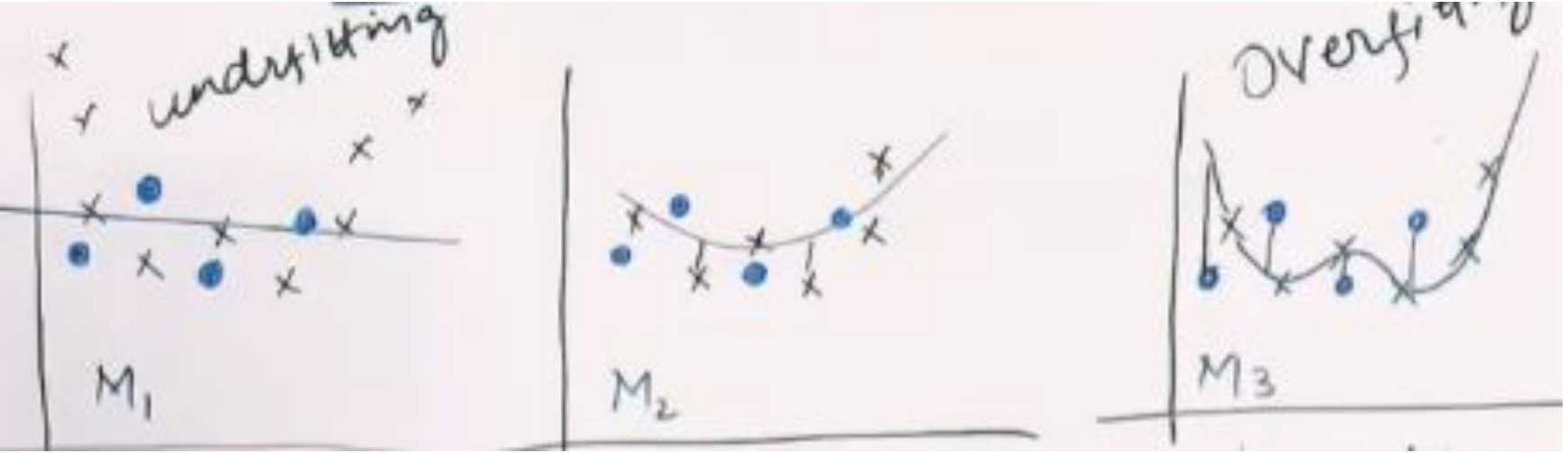
Random forests algorithm can be used for feature selection process. This algorithm can be used to rank the importance of variables in a regression or classification problem.

We measure the variable importance in a dataset by fitting the random forest algorithm to the data. During the fitting process, the out-of-bag error for each data point is recorded and averaged over the forest.

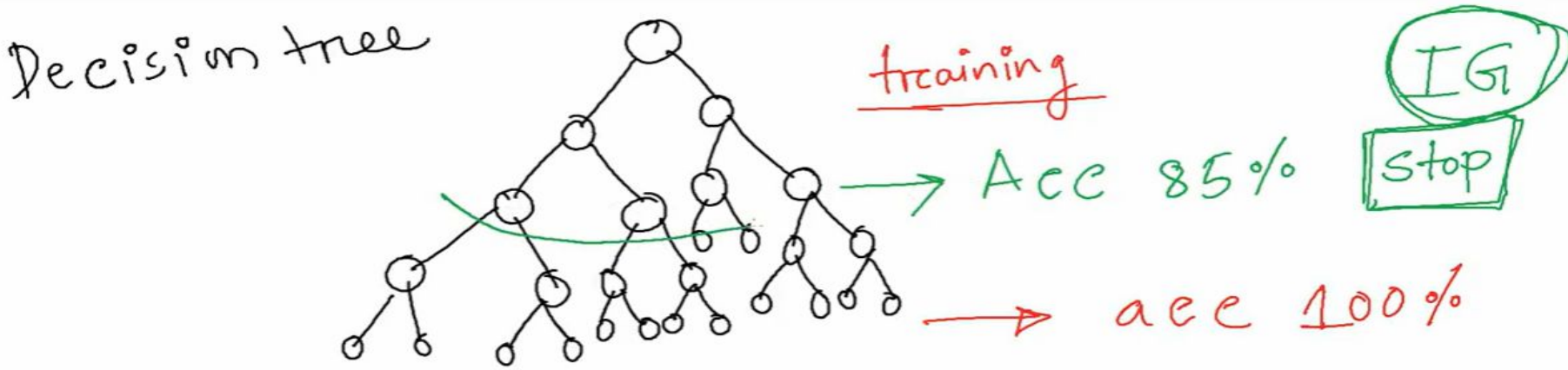
The importance of the j-th feature was measured after training. The values of the j-th feature were permuted among the training data and the out-of-bag error was again computed on this perturbed dataset. The importance score for the j-th feature is computed by averaging the difference in out-of-bag error before and after the permutation over all trees. The score is normalized by the standard deviation of these differences.

Features which produce large values for this score are ranked as more important than features which produce small values. Based on this score, we will choose the most important features and drop the least important ones for model building.

Bias variance Tradeoff



Decision tree (where to stop?)

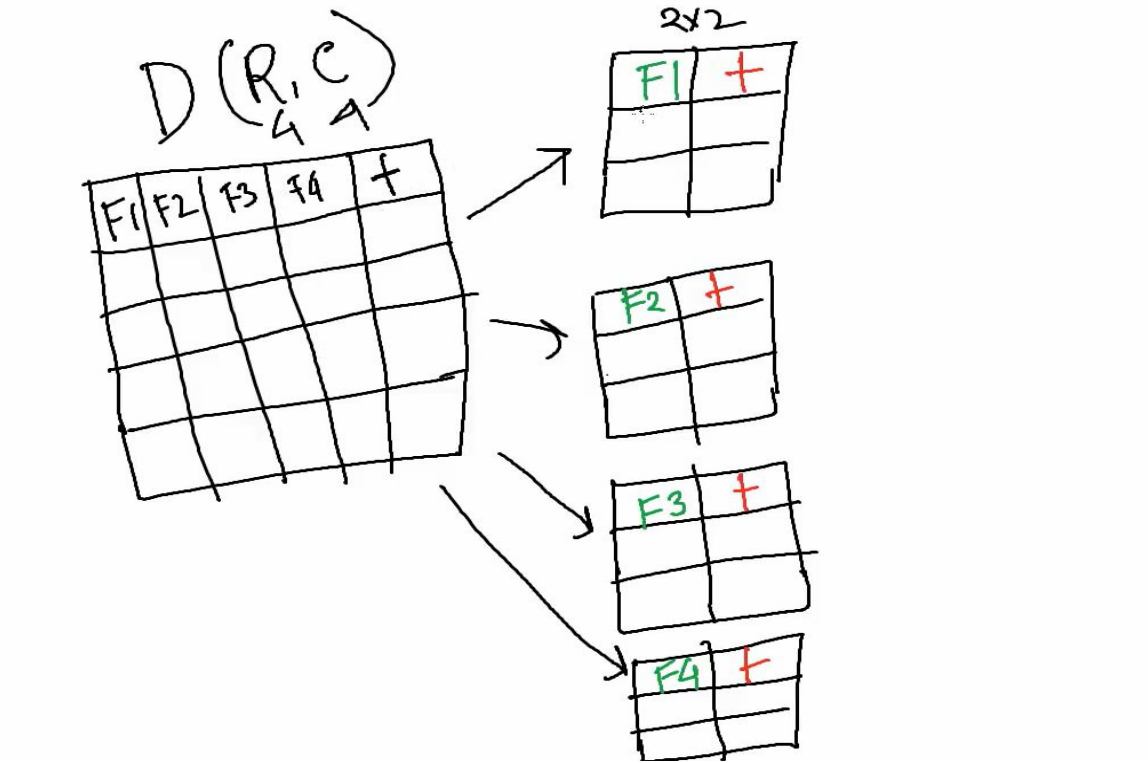


Problem with Decision tree

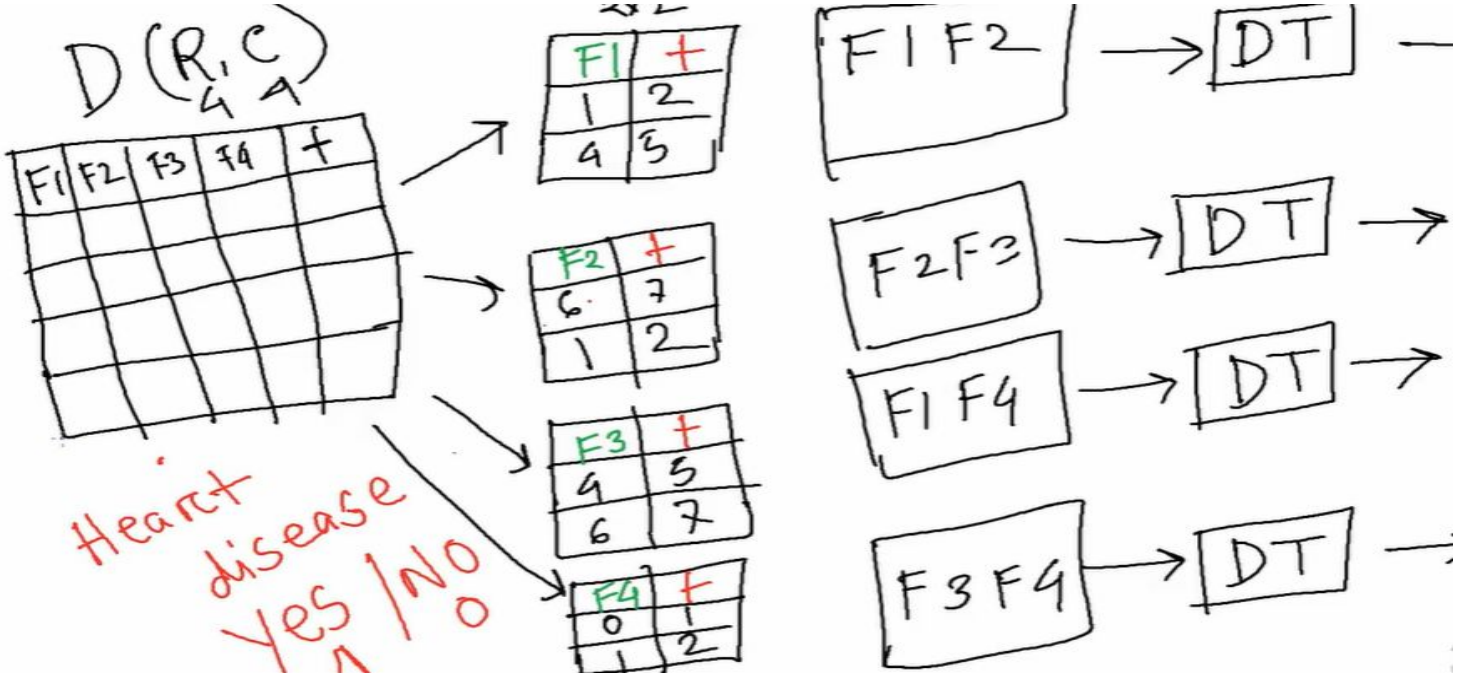
overfitting (becoming too complex and memorizing training data, failing on new data) and instability (small data changes causing huge tree structure shifts), leading to poor generalization.

In short : low bias, high variance or vice versa

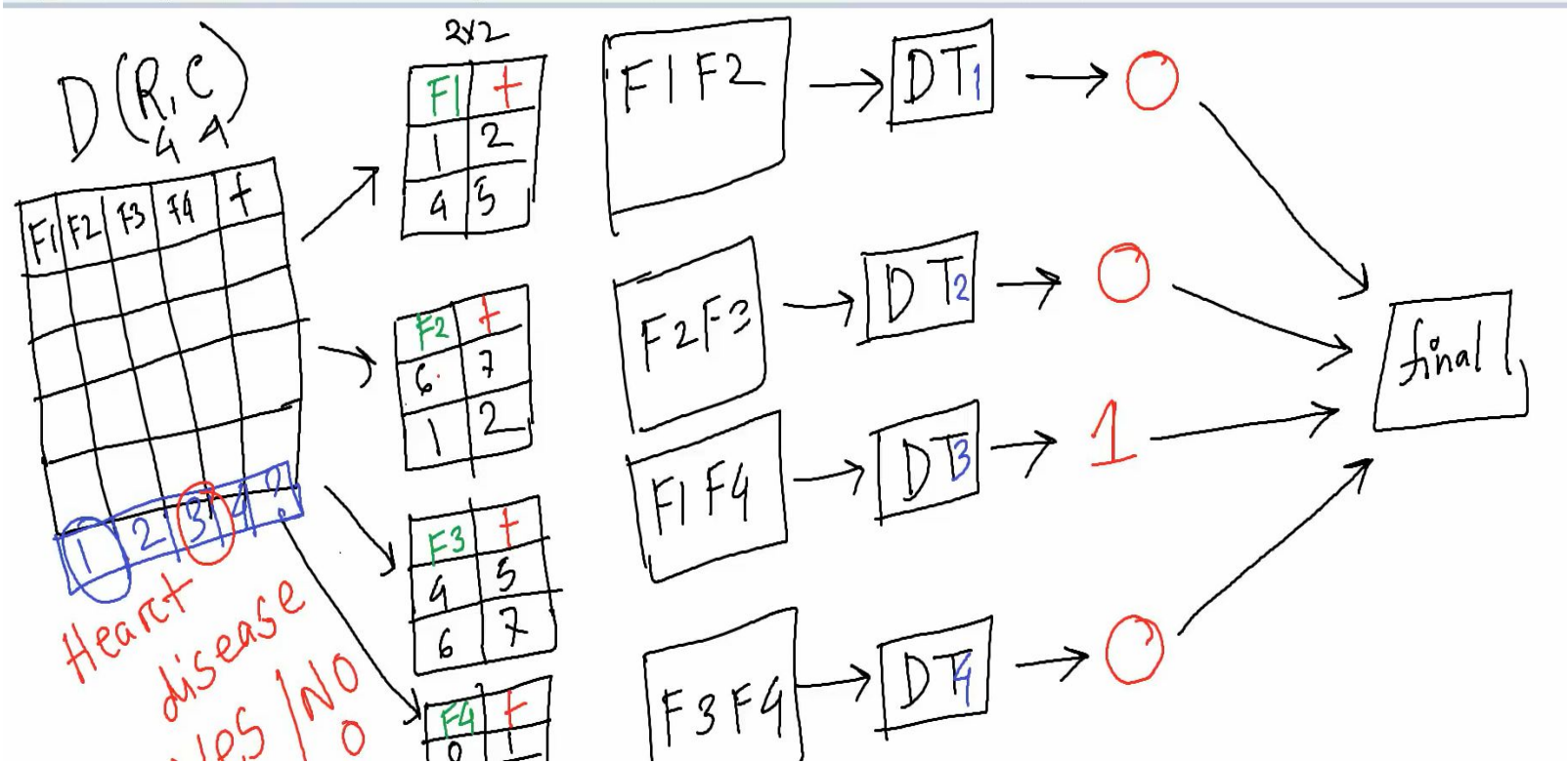
Solution is Random forest



How the Tree splitting will look like



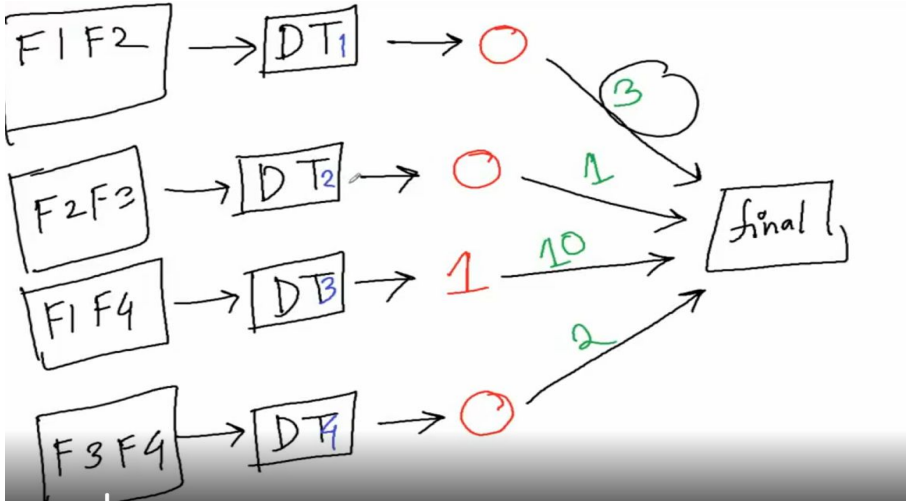
Continue

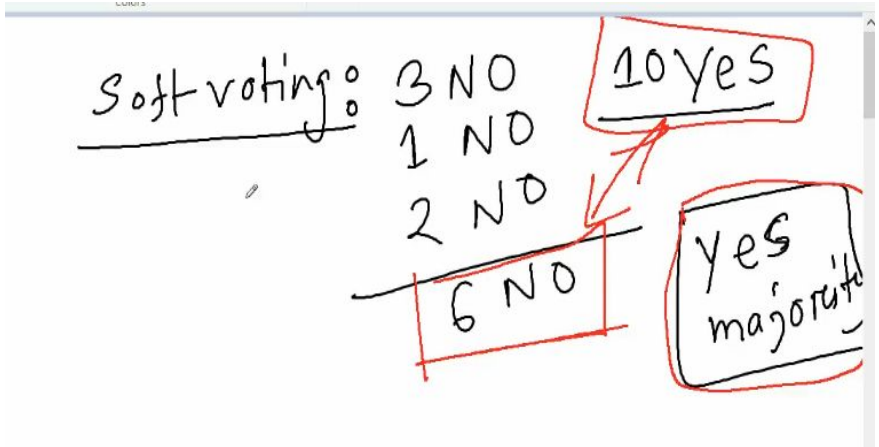

D(R, C)
4 4
F1 F2 F3 F4 t
1 2 3 4 5
Heart disease
yes / no
0
2x2
F1 t
1 2
4 5
F2 t
6 7
1 2
F3 t
4 5
6 7
F4 t
0 1
F1 F2
DT1
F2 F3
DT2
F1 F4
DT3
1
F3 F4
DT4
final

Soft voting vs hard voting

Hard voting : majority count

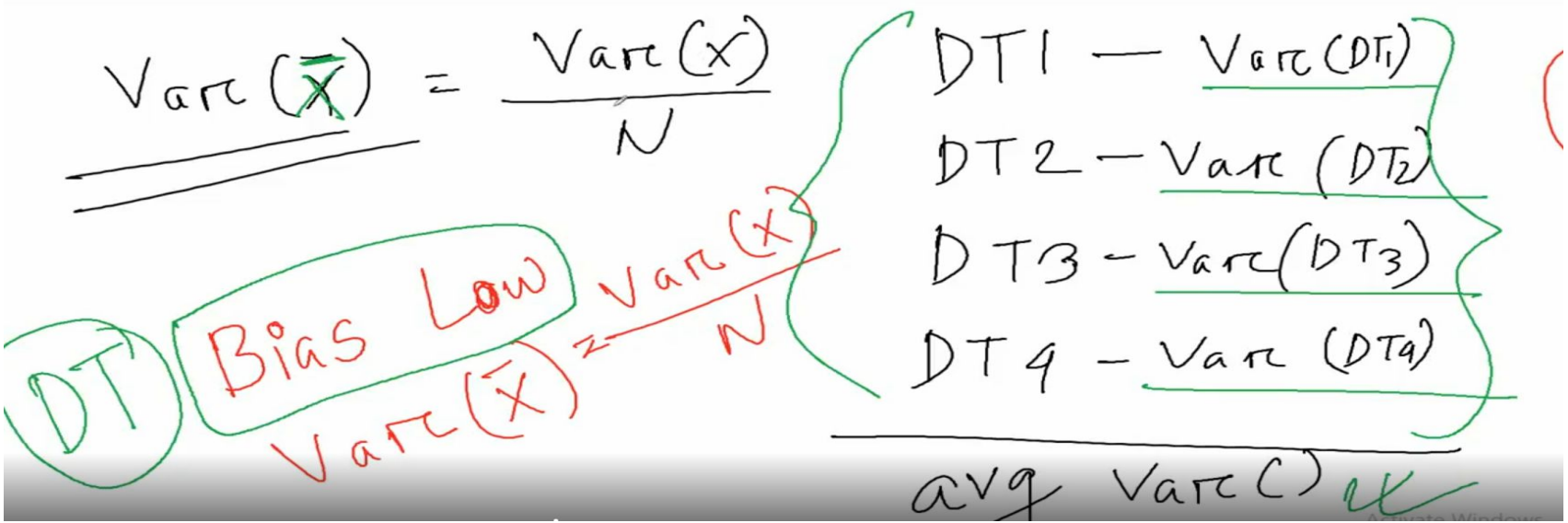
Soft voting : Assign weight



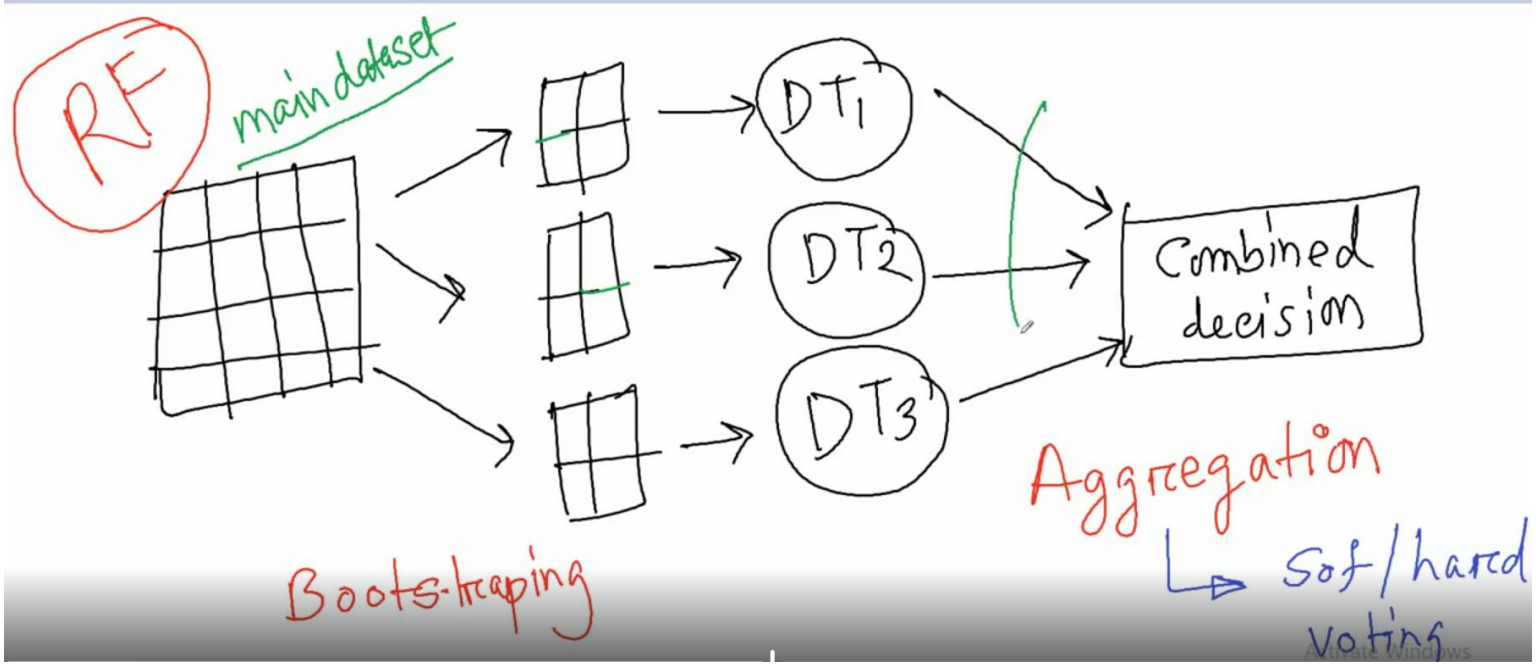


We can Decrease Variance in Decision tree (that's the magic!)

If we take the average

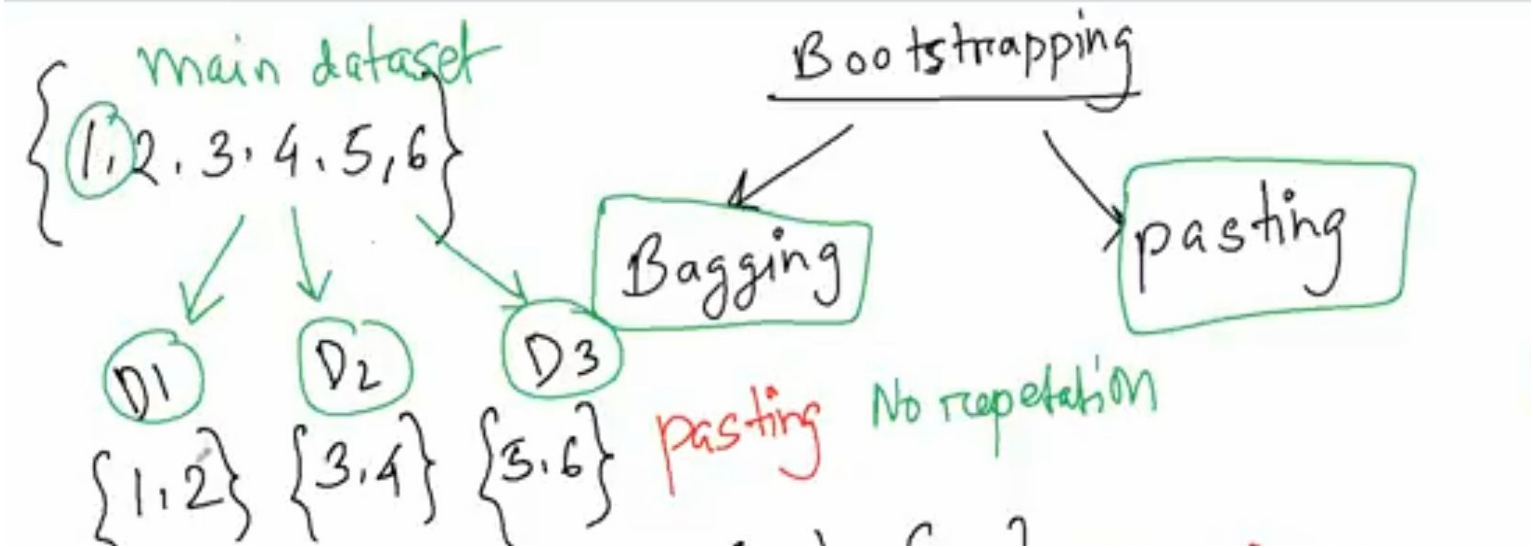


Overall process

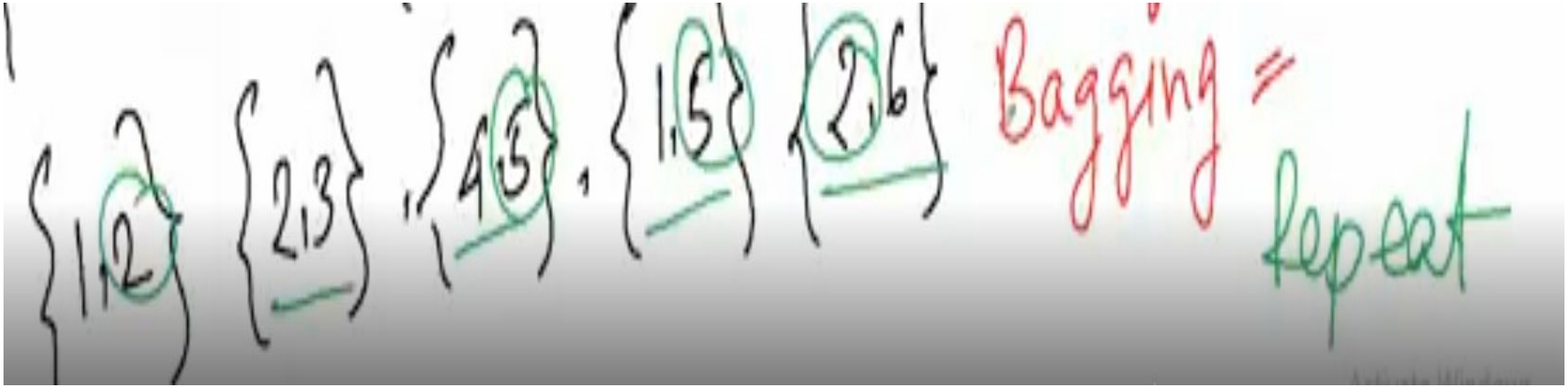


How BootStrapping is done?

Pasting :



Bagging



Question : In our Dataset, we used bagging or pasting?

Now which one to use?

Bagging brings Diversity

Let’s code now!

Advantages and disadvantages of Random Forest algorithm

The advantages of Random forest algorithm are as follows:-

1. Random forest algorithm can be used to solve both classification and regression problems.
2. It is considered as very accurate and robust model because it uses large number of decision-trees to make predictions.
3. 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.
4. 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.
5. 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.