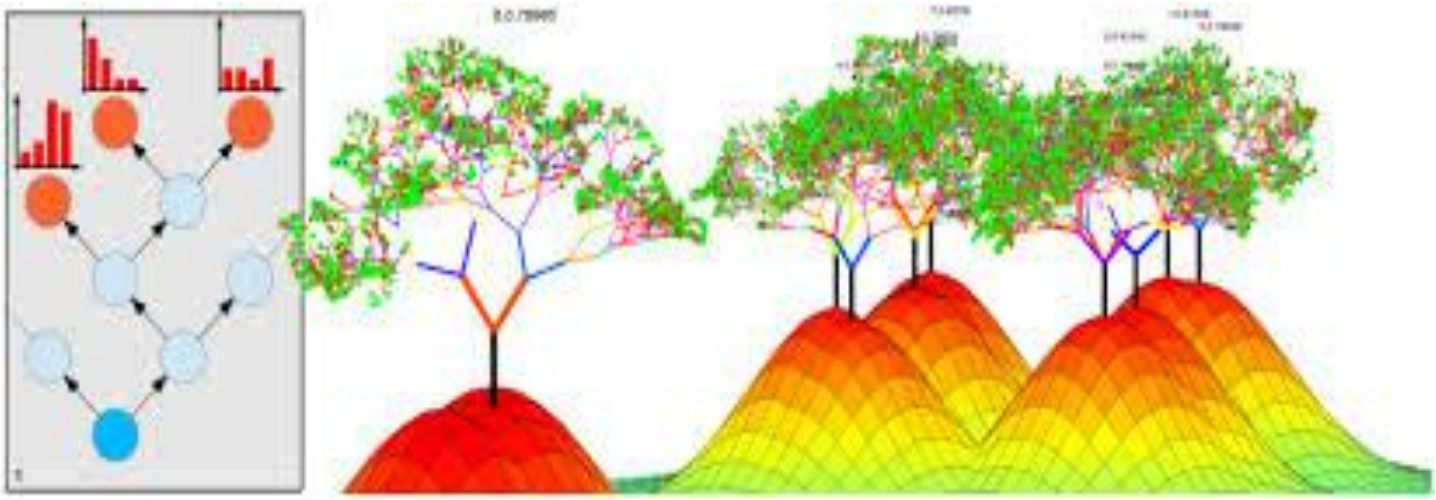
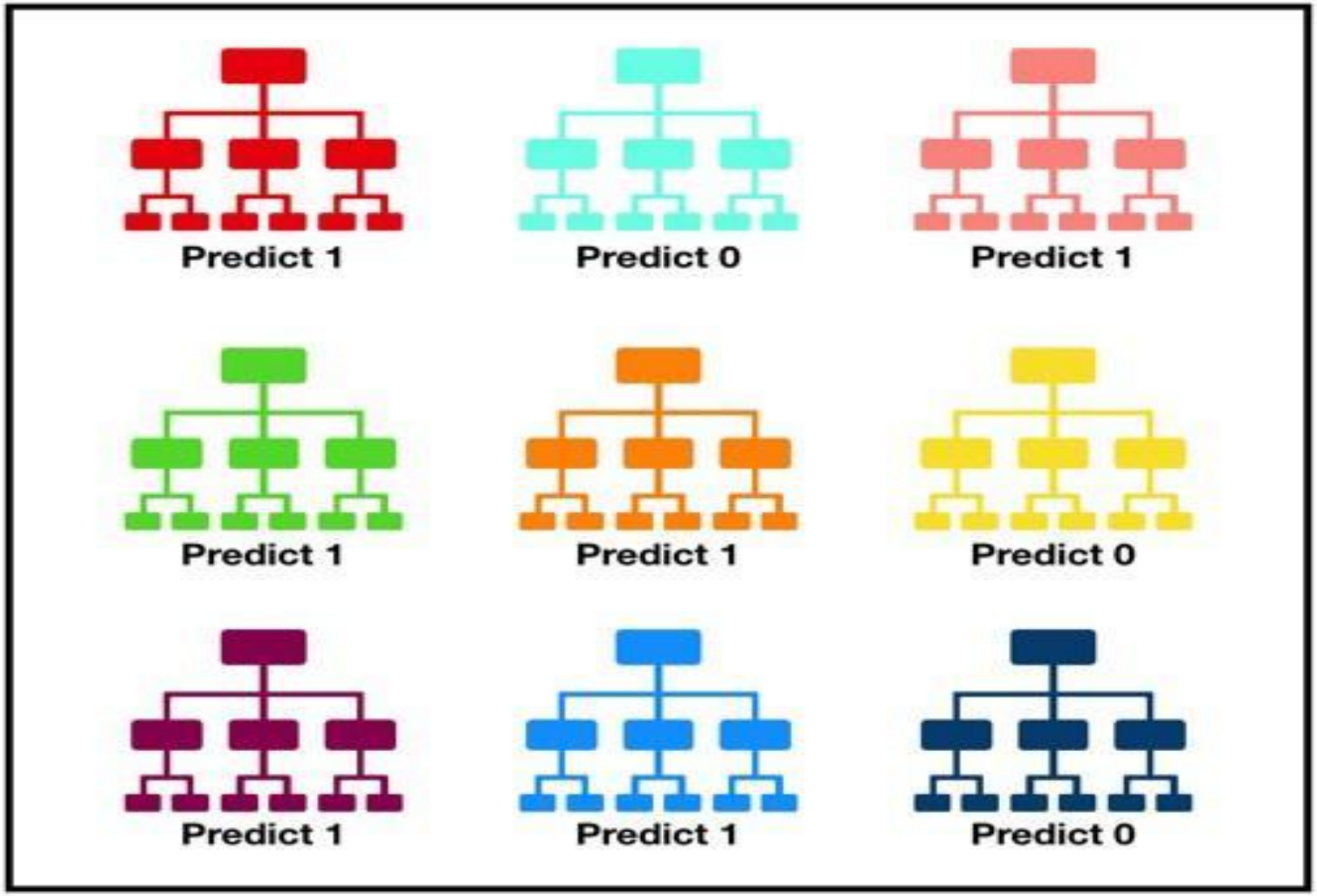
Random Forest



- The random forest is a model made up of many decision trees. Rather than just simply averaging the prediction of trees (which we could call a "forest"), this model uses two key concepts that gives it the name *random.*
- Random sampling of training data points when building trees
- Random subsets of features considered when splitting nodes

Random Forest Working

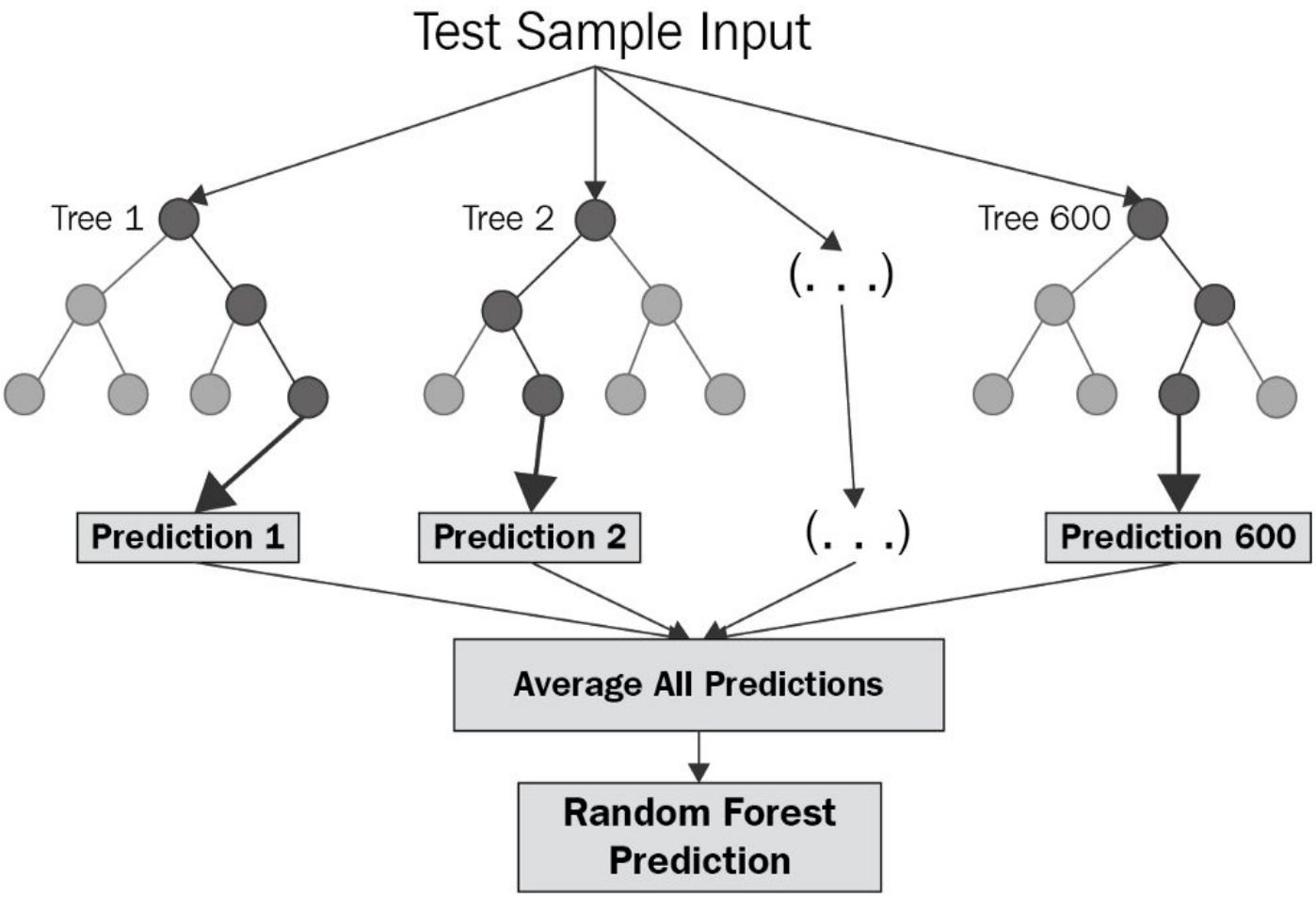


Tally: Six 1s and Three 0s

Prediction: 1

INTERNSHIPSTUDIO

This is an **ensemble algorithm**
Takes into account results of more than one algorithms of the same or different kind for classification.



How does Random Forest algorithm work?

Random Forest works in two-phase first is to create the random forest by combining N decision tree, and second is to make predictions for each tree created in the first phase.

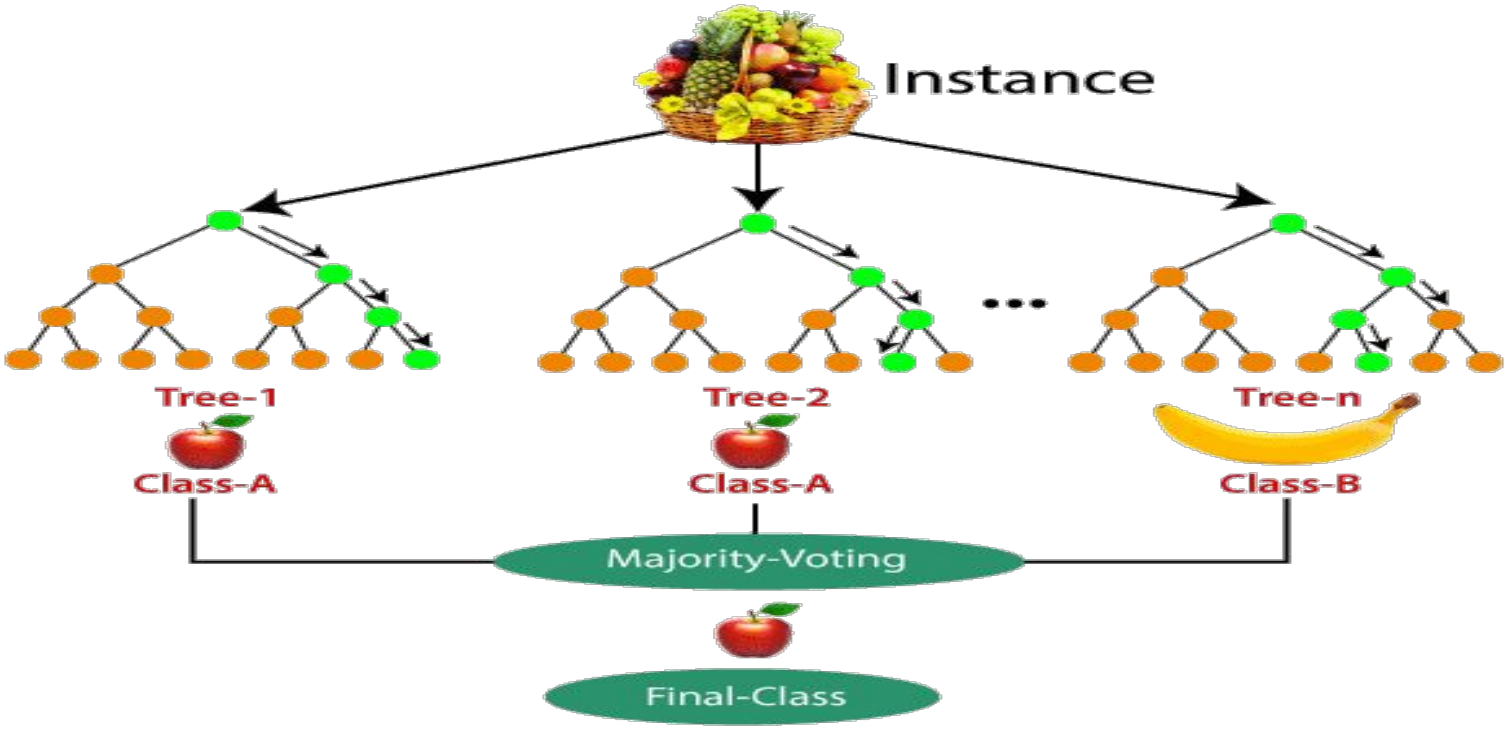
- **Step-1:** Select random K data points from the training set.
- **Step-2:** Build the decision trees associated with the selected data points (Subsets).
- **Step-3:** Choose the number N for decision trees that you want to build.
- **Step-4:** Repeat Step 1 & 2.
- **Step-5:** For new data points, find the predictions of each decision tree, and assign the new data points to the category that wins the majority votes.

Lets see how it looks like in next slide

Random Forest - Example

INTERNSHIPSTUDIO

- Suppose there is a dataset that contains multiple fruit images & given to the Random forest classifier.
- The dataset is divided into subsets and given to each decision tree.
- During the training phase, each decision tree produces a prediction result, and when a new data point occurs, then based on the majority of results, the Random Forest classifier predicts the final decision.



Why a Random Forest is better than a single decision tree?

- A random forest combines the predictions made by many decision trees into a single model.
- Individually, predictions made by decision trees may not be accurate but combined together, the predictions will be closer to the true value on average.
- Each individual tree brings their own information sources to the problem as they consider a random subset of features
- If we only build one tree we would only take advantage of their limited scope of information, but by combining many trees' predictions together, our net information would be much greater.
- This increased diversity in the forest leading to more robust overall predictions and the name ‘random forest.’
- When it comes time to make a prediction, the random forest regression model takes the average of all the individual decision tree estimates.

Applications of Random Forest

There are mainly four sectors where Random forest mostly used:

- **Banking:** Banking sector mostly uses this algorithm for the identification of loan risk.
- **Medicine:** With the help of this algorithm, disease trends and risks of the disease can be identified.
- **Land Use:** We can identify the areas of similar land use by this algorithm.
- **Marketing:** Marketing trends can be identified using this algorithm.

Load *RandomForestClassifier* class of sklearn

```
from sklearn.ensemble import RandomForestClassifier
classifier = RandomForestClassifier(n_estimators = 50)
classifier.fit(X_train, y_train)
```

At last, we need to make prediction. It can be done with the help of following script

```
y_pred = classifier.predict(X_test)
y_pred
```

Pros and Cons of Random Forest

Pros:-

- Random forests work well for a large range of data items than a single decision tree does.
- Random forest has less variance then single decision tree.
- Random forests are very flexible and possess very high accuracy.
- Scaling of data does not require in random forest algorithm. It maintains good accuracy even after providing data without scaling.
- Random Forest algorithms maintains good accuracy even a large proportion of the data is missing.
- It overcomes the problem of overfitting by averaging or combining the results of different decision trees.

Cons:-

- Complexity is the main disadvantage of Random forest algorithms.
- Construction of Random forests are much harder and time-consuming than decision trees.
- More computational resources are required to implement Random Forest algorithm.
- It is less intuitive in case when we have a large collection of decision trees.
- The prediction process using random forests is very time-consuming in comparison with other algorithms.

INTERNSHIPSTUDIO

Q.1 Discuss the implementing of Decision trees?

Q.2 Discuss the implementing of Random Forest?

Q.3 What are the advantages and disadvantages of Random Forest?

Q.4 Write the code for printing Confusion Matrix, Classification Report and Accuracy ?