Random Forest Classifier-

- Ensemble Learning:-

--Ensemble Learning is when we take multiple machine learning algorithms and putting them together to form a Big Machine Learning algorithm.

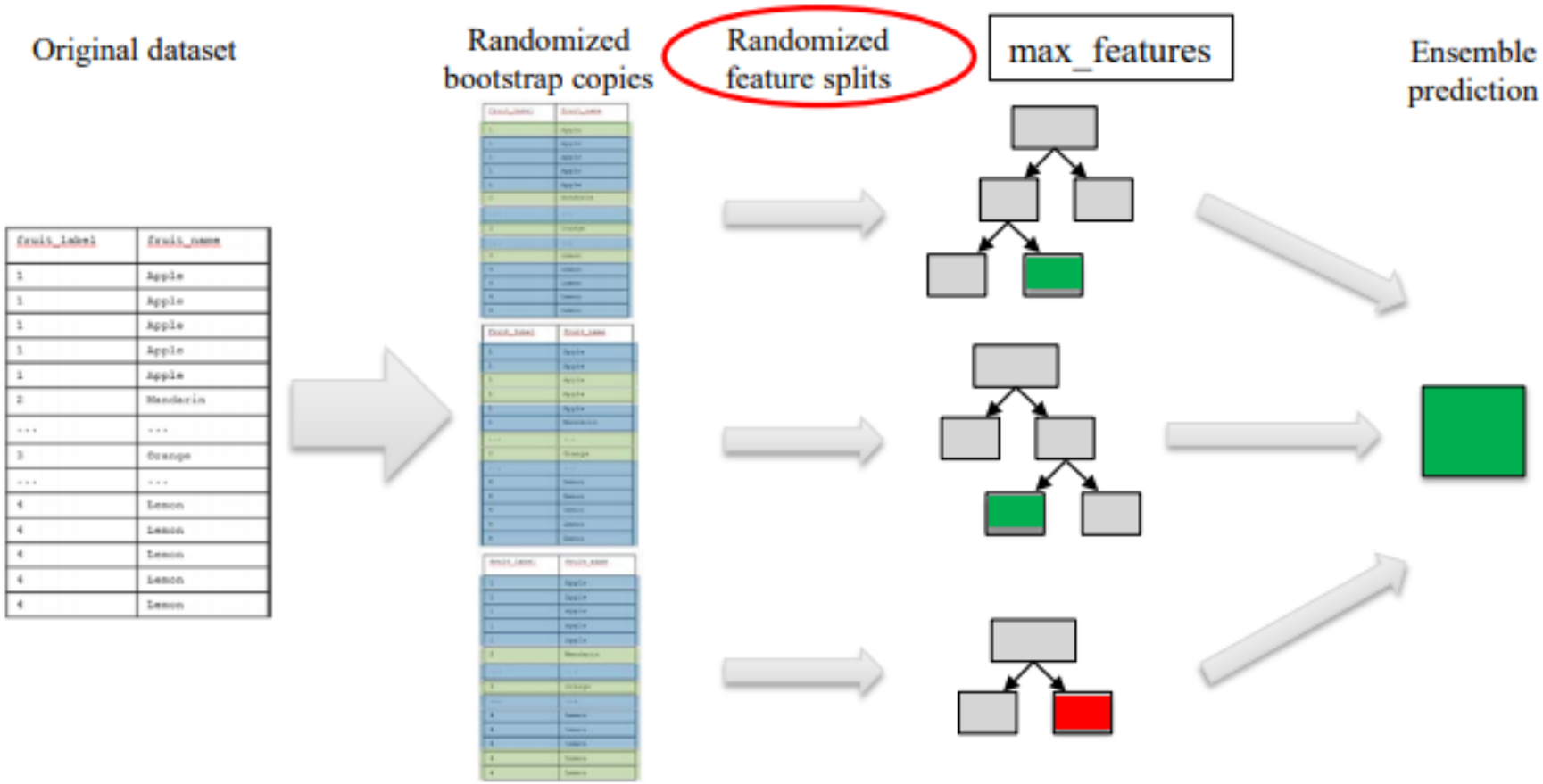
--That final algorithm have all the features of the remaining algorithms, thus increasing the accuracy of model.

--Mostly, random forests are run with multiple decision trees.

Steps For Building a Random Forest:-

- Step 1:- Pick a random k data points from the Training set.
- Step 2:- Build the decision tree associated to these K data points.
- Step 3:- Choose the number of trees we want to built and repeat step 1, Step 2.
- Step 4:- Now, for your new data point, train the model with different trees and predict the category to which the data point belongs to. Finally, assign the new data point to the category in which have the highest majority.

Random Forest Process



- n_estimators: number of trees to use in ensemble (default: 10). – Should be larger for larger datasets to reduce overfitting (but uses more computation).
- max_features: has a strong effect on performance. Influences the diversity of trees in the forest. – Default works well in practice, but adjusting may lead to some further gains.
- max_depth: controls the depth of each tree (default: None. Splits until all leaves are pure).
- n_jobs: How many cores to use in parallel during training.
- Criterion: Just like decision tree, we can use entropy and gini in this while using decision trees.
- Choose a fixed setting for the random_state parameter if you need reproducible results.

Gradient Boosted Decision Trees (GBDT): -

- Training builds a series of small decision trees.
- Each tree attempts to correct errors from the previous stage
- The learning rate controls how hard each new tree tries to correct remaining mistakes from previous round.
- High learning rate: more complex trees
- Low learning rate: simpler trees