A Presentation on Random Forest

PREPARED BY: SUBASH SAH

What is Random Forest?

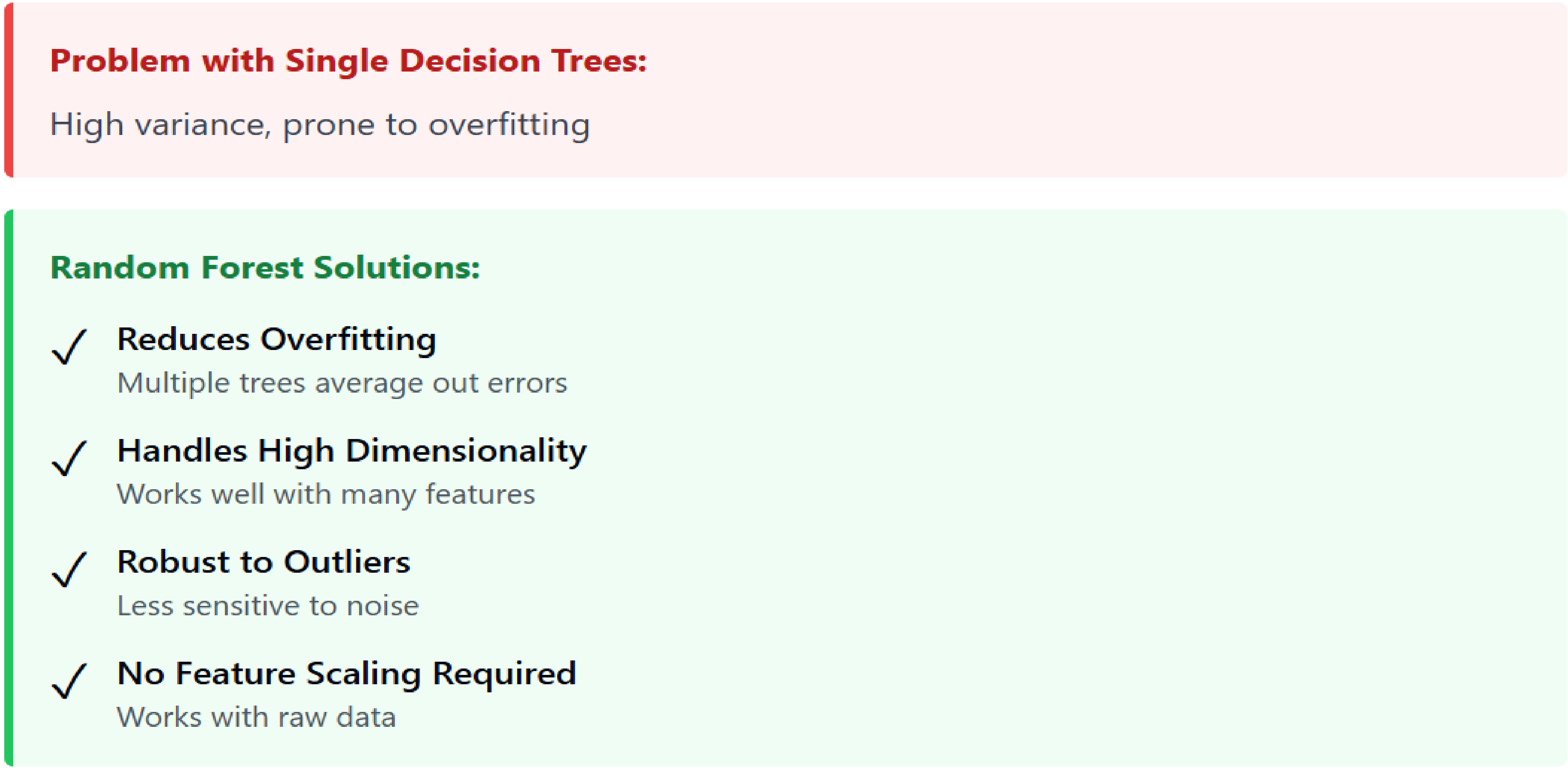
Random Forest is an **ensemble learning** method that combines multiple decision trees to make predictions.

Key Concept:
Wisdom of Crowd – Multiple trees vote together to make better predictions than any single tree.

Classification
Majority Voting

Regression
Average Prediction

Why Random Forest?



Problem with Single Decision Trees:

High variance, prone to overfitting

Random Forest Solutions:

- ✓ **Reduces Overfitting**
  Multiple trees average out errors
- ✓ **Handles High Dimensionality**
  Works well with many features
- ✓ **Robust to Outliers**
  Less sensitive to noise
- ✓ **No Feature Scaling Required**
  Works with raw data

How Random Forest Works?

Step-by-Step Process:

1. Bootstrap Sampling:

   Create multiple random subsets of data with replacement

2. Random Feature Selection:

   At each split, consider only random subset of features

3. Build Decision Trees:

   Train individual trees on each bootstrap sample

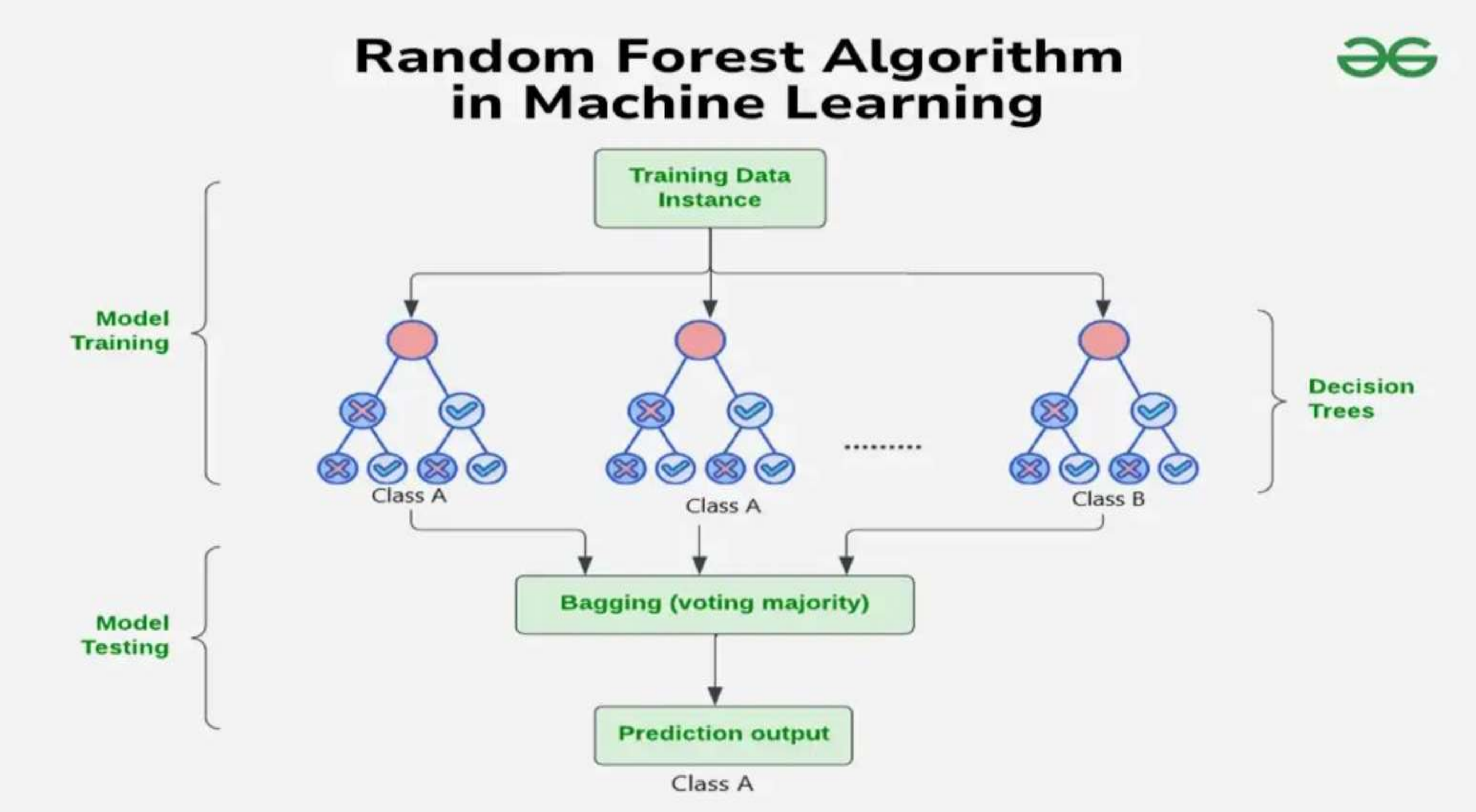
4. Aggregate Predictions:

   Majority vote or average the results

Randomness Explained

- **Random sampling of data:** Each tree gets a random subset of training data.
- **Random subset of features:** Each tree splits using a random subset of features.

→ This helps trees less correlated, improving generalization.


Random Forest Algorithm
in Machine Learning
Training Data Instance
Model Training
Decision Trees
Class A
Class A
Class B
Model Testing
Bagging (voting majority)
Prediction output
Class A

Hyperparameters to Tune

n_estimator
Number of trees in the forest (default: 100)

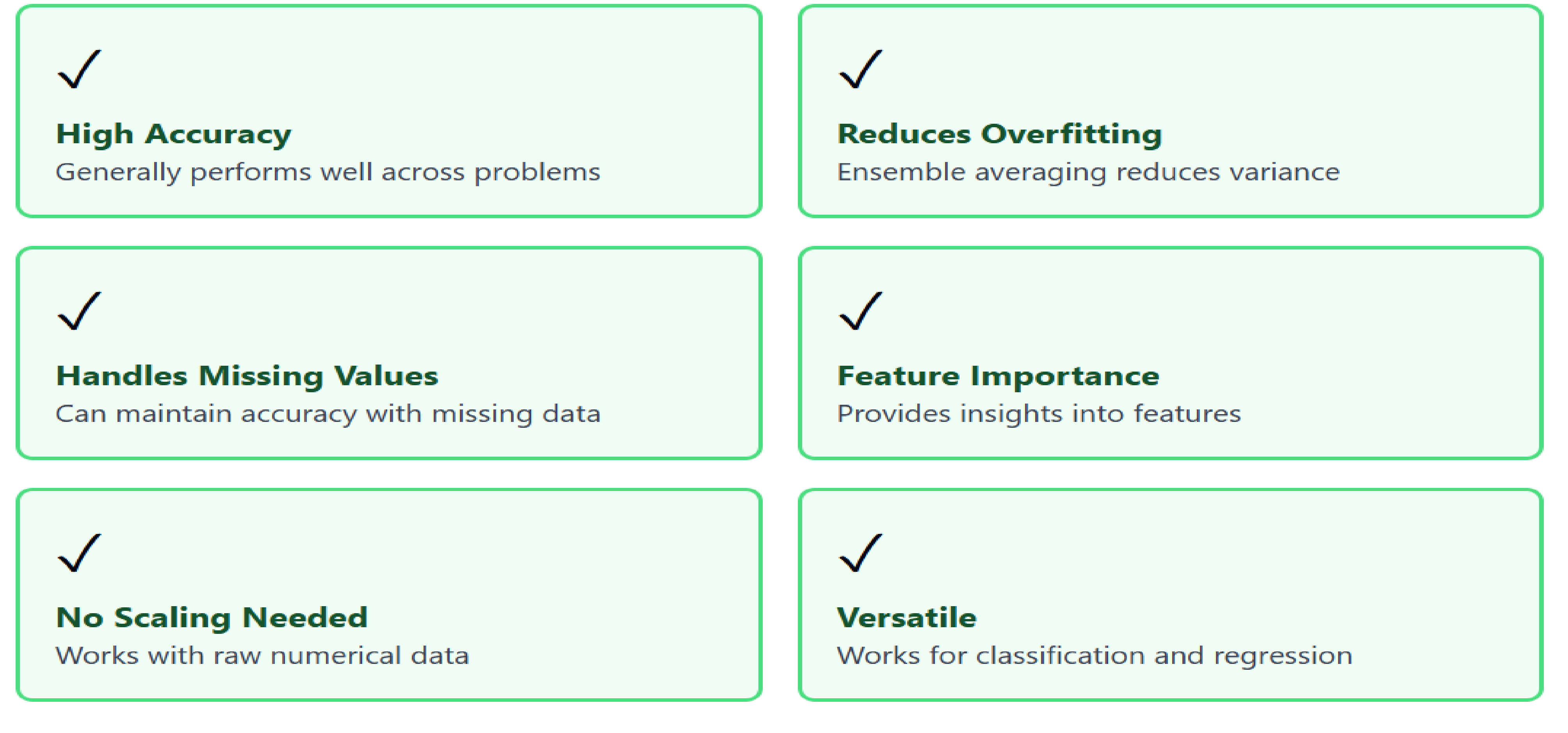
max_depth
Number of depth of each tree

max_features
Number of features to consider at each split

min_samples_split
Minimum samples required to split a node

min_samples_leaf
Minimum samples required at a leaf node

Advantages of Random Forest



✓

High Accuracy

Generally performs well across problems

✓

Reduces Overfitting

Ensemble averaging reduces variance

✓

Handles Missing Values

Can maintain accuracy with missing data

✓

Feature Importance

Provides insights into features

✓

No Scaling Needed

Works with raw numerical data

✓

Versatile

Works for classification and regression

Disadvantages of Random Forest

Computational cost
Training many trees is time intensive

Memory Usage
Stores multiple trees requiring memory

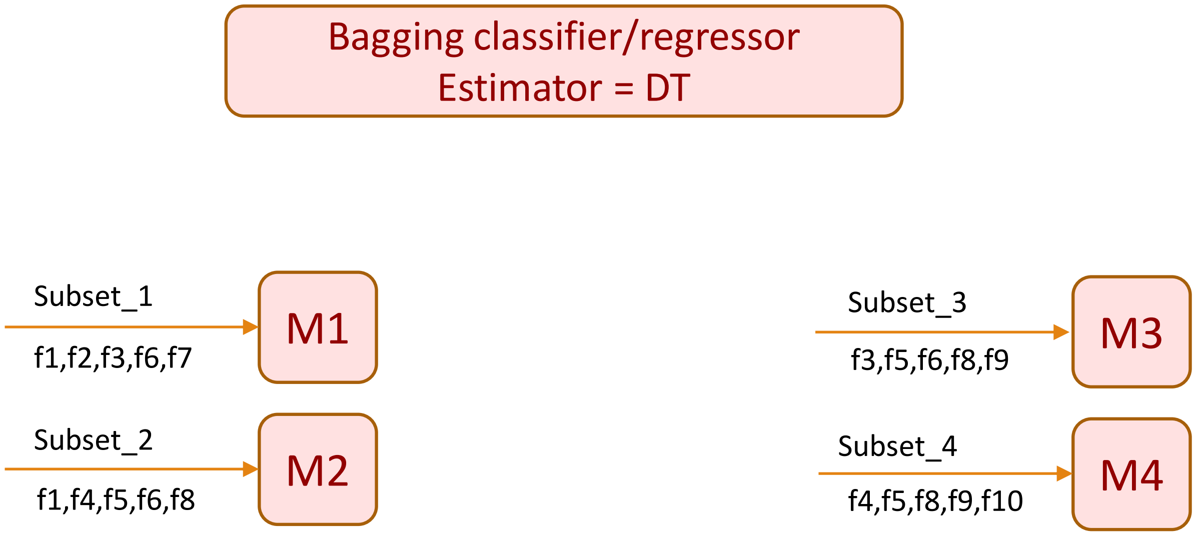
Black Box Model
Less interpretable

Prediction Time
Slower predictions compared to single models

Not Ideal for Linear Data
Overkill for simple linear problems

Why the name Random? Why not only Forest?

Q. If we put estimator = DT in Bagging classifier/regressor, is it same as Random Forest?



Bagging classifier/regressor
Estimator = DT

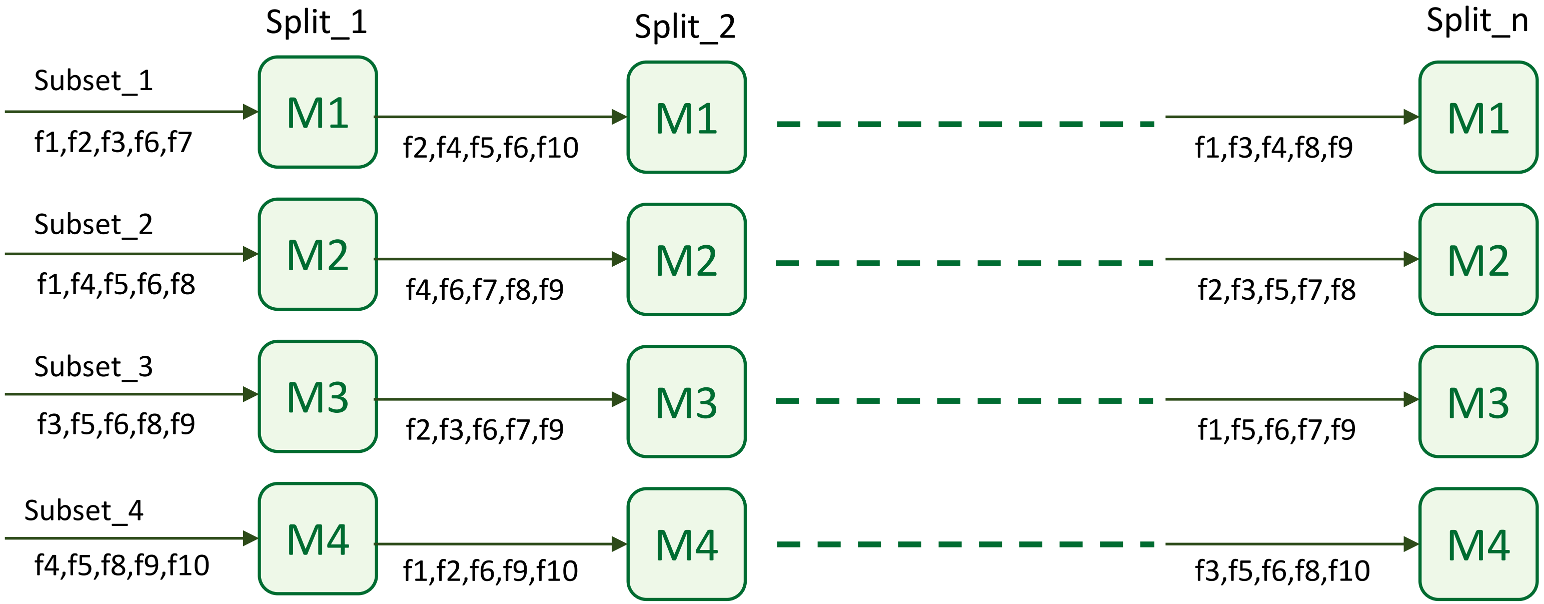
Subset_1
f1,f2,f3,f6,f7
M1

Subset_2
f1,f4,f5,f6,f8
M2

Subset_3
f3,f5,f6,f8,f9
M3

Subset_4
f4,f5,f8,f9,f10
M4

Random Forest



Thank you!