UNIT - III

ENSEMBLE TECHNIQUES & UNSUPERVISED LEARNING

UNIT III ENSEMBLE TECHNIQUES & UNSUPERVISED LEARNING 9

Combining multiple learners: Model combination schemes, Voting, Ensemble Learning - bagging, boosting, stacking, Unsupervised learning: K-means, Instance-Based Learning: KNN, Gaussian mixture models and Expectation maximization

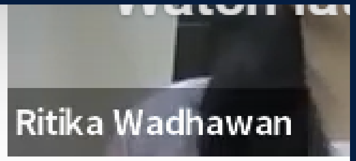
Ensemble Techniques

- **To purchase a new car,**
  1. **Browse web portals**
  2. **Look into posted reviews**
  3. **Ask friends and colleagues for their opinion**
- **You wouldn’t directly reach a conclusion, but will instead make a decision considering the opinions of other people as well**

What Is Ensemble?

- Ensemble means ‘a group producing a single effect’
- In machine learning, it is a technique that combines several base models in order to produce one optimal model
- In learning models, noise, variance, and bias are the major sources of error. The ensemble methods in machine learning help minimize these error-causing factors, thereby ensuring the accuracy and stability of machine learning (ML) algorithms

ML Learners



Weak learners

Weak learners have low prediction accuracy, like random guessing. They are prone to overfitting—that is, they can't classify data that varies too much from their original dataset. For example, if you train the model to identify cats as animals with pointed ears, it might fail to recognize a cat whose ears are curled.

Strong learners

Strong learners have higher prediction accuracy. Ensemble methods convert a system of weak learners into a single strong learning system. For example, to identify the cat image, it combines a weak learner that guesses for pointy ears and another learner that guesses for cat-shaped eyes. After analyzing the animal image for pointy ears, the system analyzes it once again for cat-shaped eyes. This improves the system's overall accuracy.

Ensemble Learning

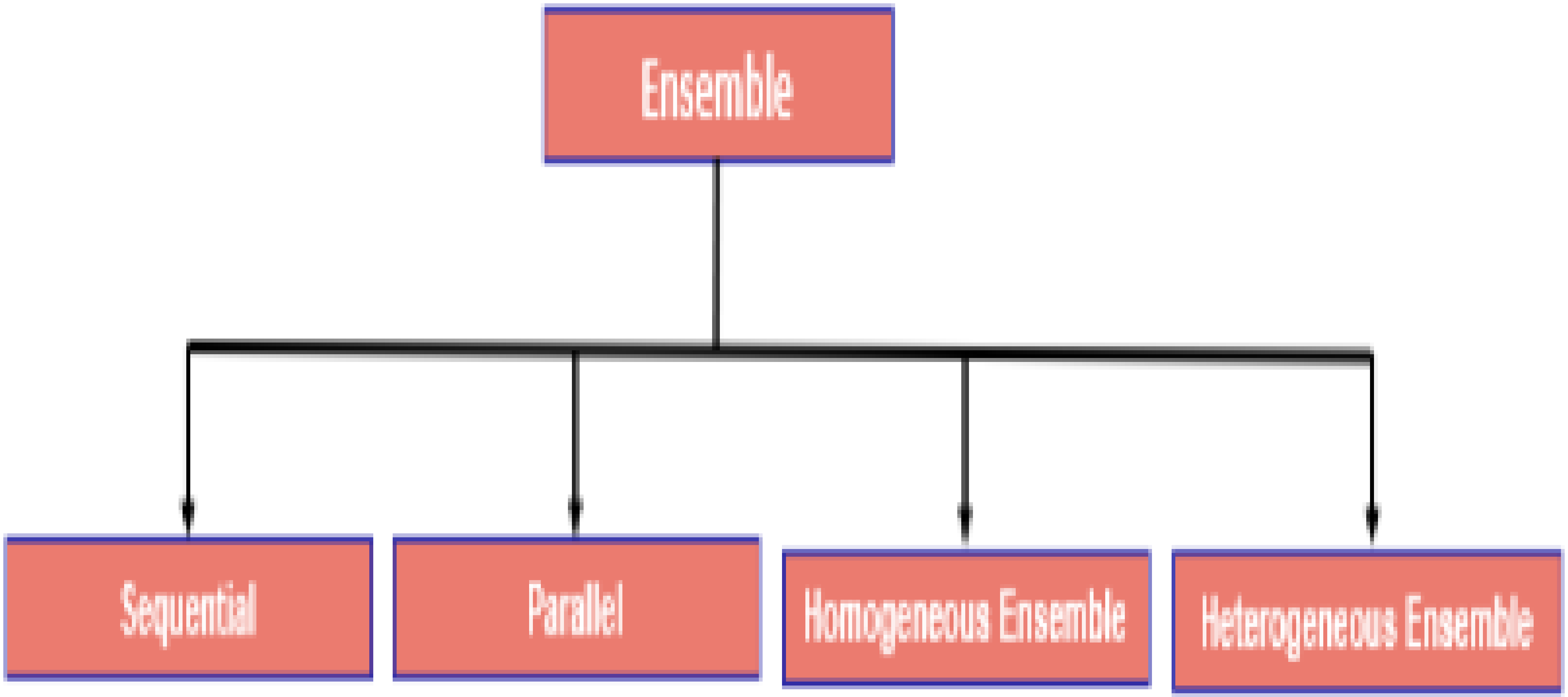
- **Ensemble Learning:** is a group of predictors that are trained and used for predictions in ML.
- Ensemble learning methods are based on the hypothesis that combining multiple models together can often produce a much more powerful model.

What are ensemble methods?

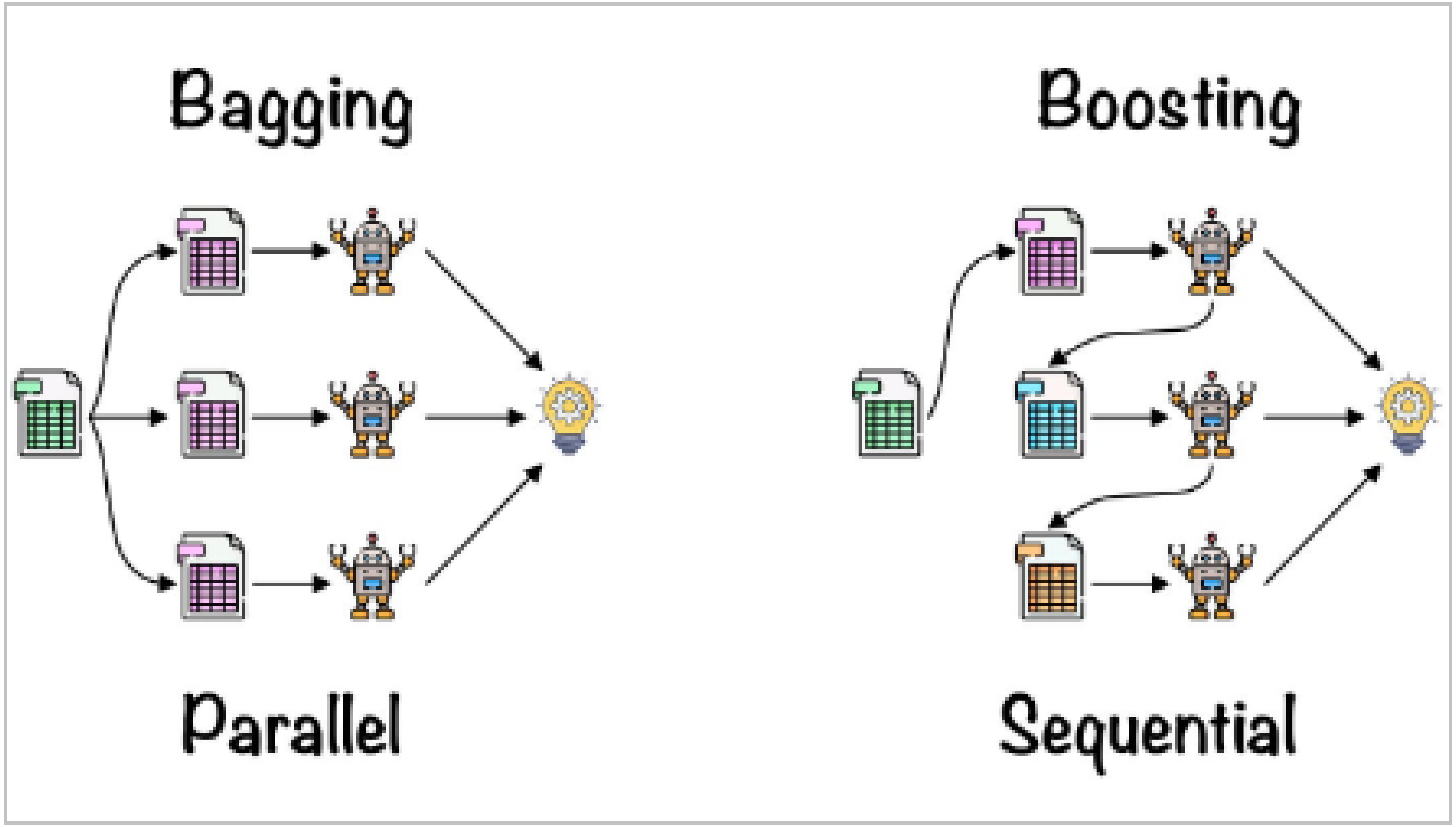
- Ensemble learning is a machine learning paradigm where multiple models (often called "weak learners") are trained to solve the same problem and combined to get better results.
- The main hypothesis is that when weak models are correctly combined we can obtain more accurate and/or robust models.

- The basic weak learner models do not perform so well by themselves either because they have a high bias or because they have too much variance to be robust.
- Then, the idea of ensemble methods is to try reducing bias and/or variance of such weak learners by combining several of them together in order to create a strong learner (or ensemble model) that achieves better performances.

Categories of Ensemble Learning



Parallel vs Sequential



Homogeneous Ensemble

- **HOMOGENEOUS ENSEMBLE** is a collection of classifiers of the same type, built upon a different subset of data as we use to do in the Random Forest model.
  - *Data 1 <> Data 2 <> Data 3 ...... Data n*
  - Examples: Bagging and Boosting

Heterogeneous Ensemble

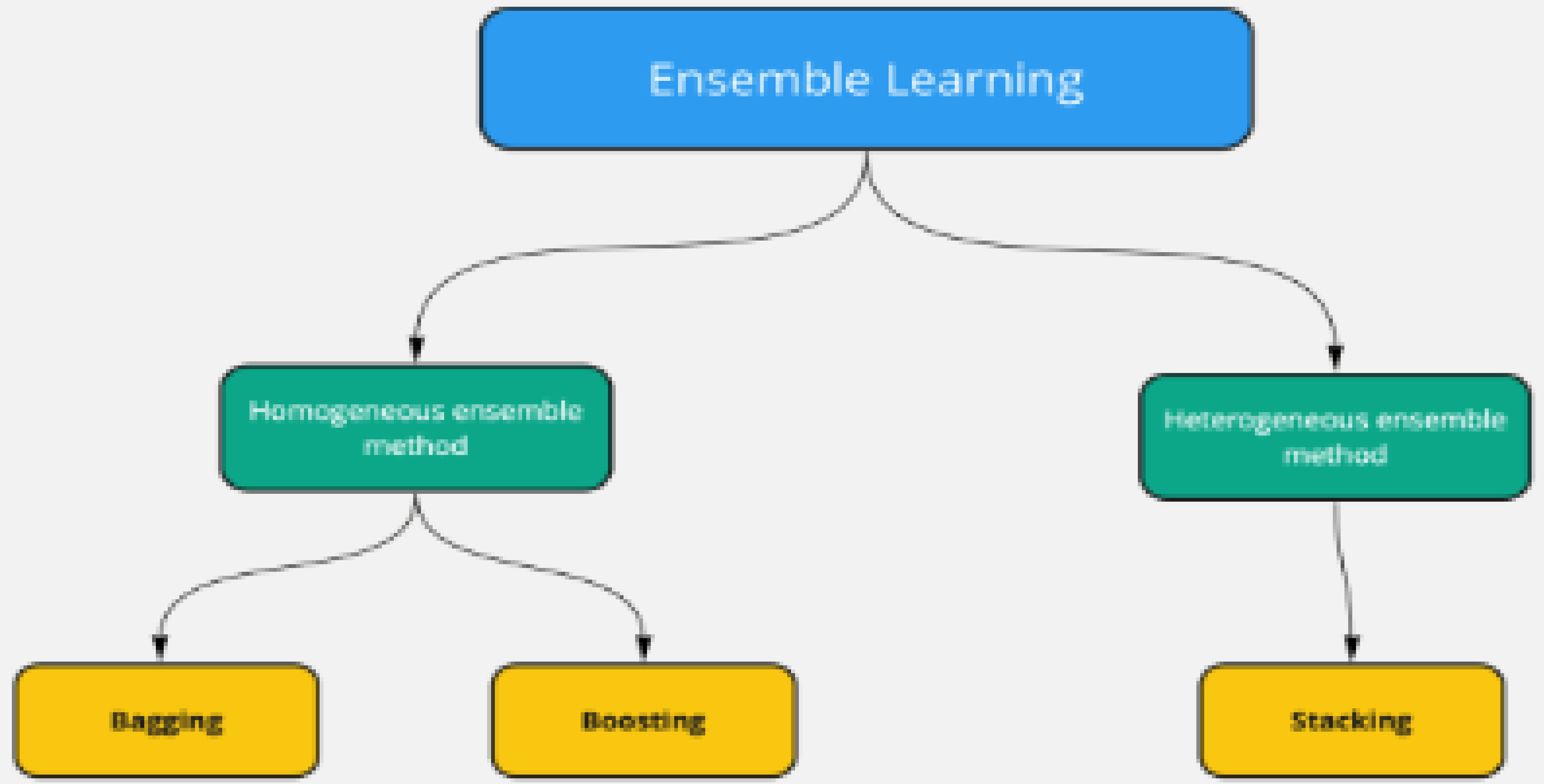
- **HETEROGENEOUS ENSEMBLE** is a set of classifiers of different types, built upon the same data.
  - ❑ *Data 1 = Data 2 = Data 3 …… Data n*
  - ❑ *Examples: Stacking*

- Bagging, Boosting - Combine Multiple weak learners of the same kind
- Stacking - Combine models of different kind i.e stacking together of different models Eg: Random Forest in stack with Extreme Gradient Boosting

Homogeneous vs Heterogeneous

Heterogeneous Ensemble	Homogeneous Ensemble
Different types classifiers	Same type of classifiers
The Data for each model is the same	the datasets should be different for each model
Number of algorithms should always be odd number to avoid ties in the results	Just one algorithm on all the estimators.
Applicable algorithms: Decision Tree, SVM and Logistic regression.	Applicable algorithm is Random Forest
Using Averaging or Stacking to aggregate the results of the models.	Aggregating the results of each model (Collection of Datasets)
Well tunned algorithms	Not be fine-tuned algorithm
Little Expensive	Expensive than Heterogeneous

Combine weak learners

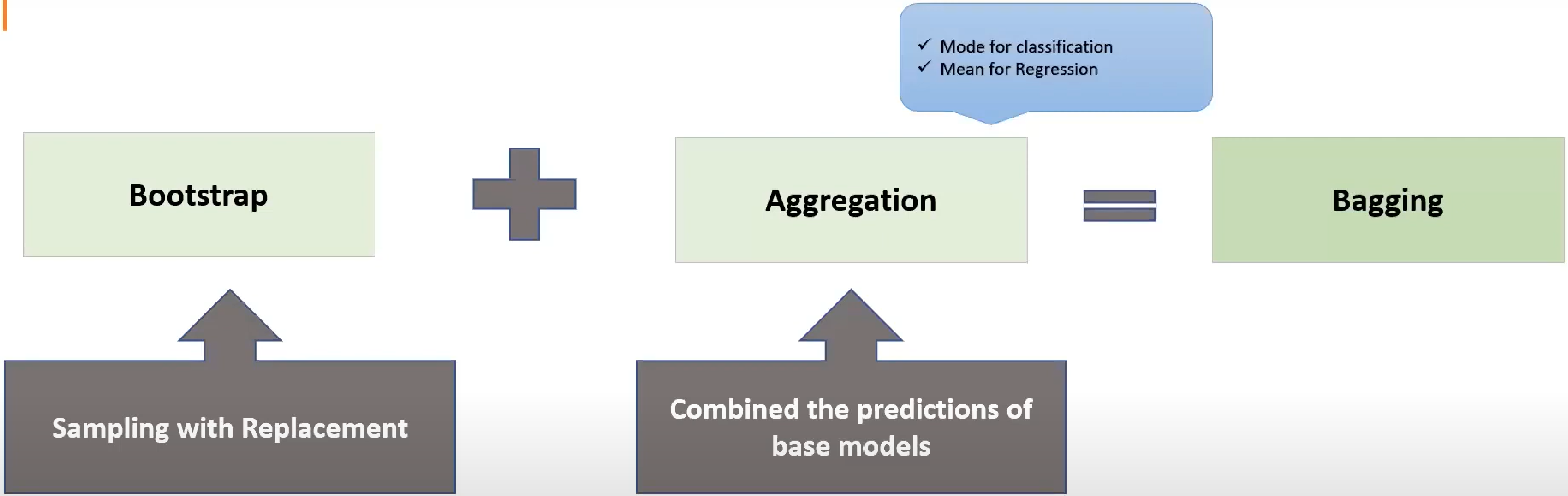


- **Bagging** – Homogeneous weak learners + parallel learning
- **Boosting** – Homogeneous weak learner + sequential learning
- **Stacking** – Heterogeneous weak learner + parallel learning + combine them by using meta model

Ensemble methods in Machine Learning

- Bagging
- Boosting
- Stacking

Bagging



For effective ensemble, one must ensure the following

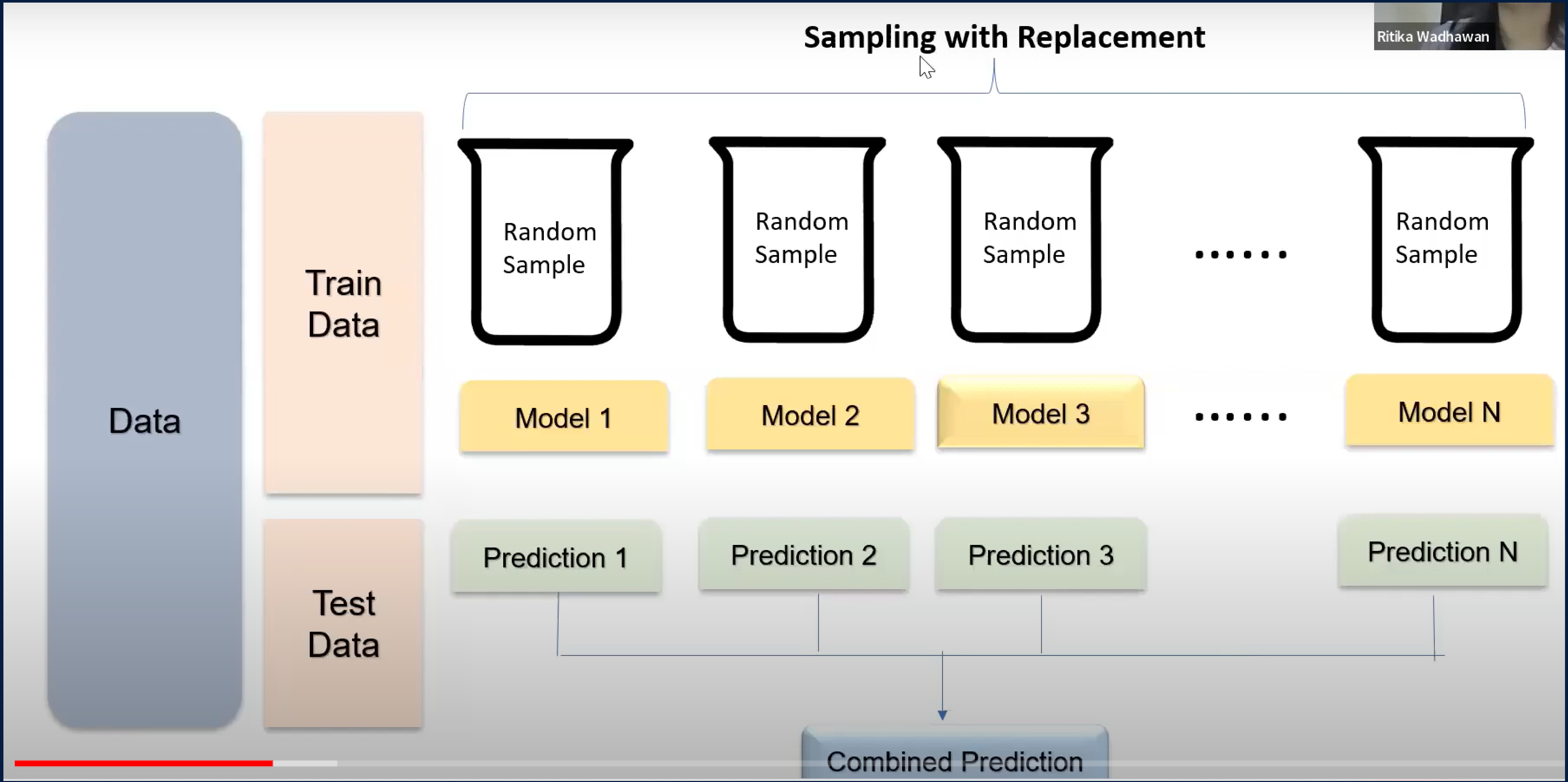
- The base estimators are as different from each other as possible
- Errors should be independent

Bagging

- Sampling with Replacement is called Bootstrap.
- The process is called Bootstrapping.
- In Bagging, all the models run in parallel.
- Aggregation function for classification is mode and aggregation function for regression is mean.

Bagging

- Bagging in ensemble machine learning takes several weak models, aggregating the predictions to select the best prediction.
- Bagging significantly raises the stability of models, which **reduces the variance** to a large extent level and eventually **increases accuracy**.
- So, it is **eliminating over fitting**, which was a big challenge in many predictive models.
- This is often considered as a homogeneous weak learner and learns them independently from each other in parallel and combines them following by averaging process.



Sampling with Replacement
Ritika Wadhawan
Data
Train Data
Test Data
Random Sample
Random Sample
Random Sample
Random Sample
Model 1
Model 2
Model 3
Model N
Prediction 1
Prediction 2
Prediction 3
Prediction N
Combined Prediction

Reason for Sampling with Replacement

1. If you do not withdraw samples with replacement, the size of the dataset would eventually decrease
2. Helps to make base models independent

Model 1 : y = avg(x1,x2..xn)			
Actual Value	Predicted Value	Absolute Error	% Error
2.6	3.3	0.7	25%
4.1	3.3	0.9	21%
3.4	3.3	0.2	4%
2.4	3.3	0.9	35%
1.9	3.3	1.4	71%
3.3	3.3	0.0	2%
2.2	3.3	1.1	48%
2.8	3.3	0.5	16%
2.7	3.3	0.6	20%
7.1	3.3	3.9	54%
		MAE= 0.98	MAPE= 29.65%

Model 2 : y = avg(x1,x2..xn) +/- Stddev			
Actual Value	Predicted Value	Error	% Error
2.6	1.7	0.9	35%
4.1	1.7	2.4	59%
3.4	4.6	1.2	36%
2.4	1.7	0.7	30%
1.9	1.7	0.2	11%
3.3	4.6	1.3	40%
2.2	1.7	0.5	23%
2.8	1.7	1.1	40%
2.7	1.7	1.0	38%
7.1	4.6	2.5	35%
		MAE= 1.19	MAPE= 34.70%

Combined Prediction	Error	% Error
2.5	0.1	5%
2.5	1.6	40%
3.9	0.5	16%
2.5	0.1	3%
2.5	0.6	30%
3.9	0.6	19%
2.5	0.3	12%
2.5	0.3	12%
2.5	0.2	9%
3.9	3.2	45%
	MAE= 0.76	MAPE= 18.97%

Regression Classification

Records	Actual Value	Model 1	Model 2	Model 3	Model 4	Model 5	Final Model : Mode
1	1	1	0	0	1	1	1
2	1	0	1	1	0	1	1
3	1	1	1	1	1	0	1
4	1	1	0	0	1	1	1
5	0	0	1	0	1	0	0
6	1	0	1	1	1	0	1
7	0	0	0	0	1	0	1
8	0	1	1	0	0	1	1
9	1	0	1	0	1	1	1
10	1	0	0	0	0	0	0
	Accuracy	**50%**	**50%**	**60%**	**60%**	**60%**	**80%**

ession | **Classification** | +

Records	Actual Value	Model 1	Model 2	Model 3	Model 4	Model 5	Final Model : Mode
1	1	1	0	0	1	1	1
2	1	0	1	1	0	1	1
3	1	1	1	1	1	0	1
4	1	1	0	0	1	1	1
5	0	0	1	0	1	0	0
6	1	0	1	1	1	0	1
7	0	0	0	0	1	0	1
8	0	1	1	0	0	1	1
9	1	0	1	0	1	1	1
10	1	0	0	0	0	0	0
	Accuracy	**50%**	**50%**	**60%**	**60%**	**60%**	**80%**

Why Random Forest?

- Decision Trees are highly sensitive to training data, which could result in high variance.
- Our model may fail to generalize.

Random Forest

- Random Forest Algorithm is a supervised machine learning algorithm that is used for Classification and Regression problems in Machine Learning.
- A forest comprises numerous trees, “the more trees more it will be robust”.
- Similarly, the greater the number of trees in a Random Forest Algorithm, the higher its accuracy and problem-solving ability.
- Random Forest is a classifier that contains several decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset.

Random Forest Algorithm

- Random forest is a commonly-used machine learning algorithm.
- A random forest is an ensemble learning method where multiple decision trees are constructed and then they are merged to get a more accurate prediction.
- Random forest became popular because of its ease of use and flexibility in handling both classification and regression problems.

Steps followed in Random Forest Algorithm

- **Step 1:** Select random samples from a given data or training set.
- **Step 2:** This algorithm will construct a decision tree for every training data.
- **Step 3:** Voting will take place by averaging the decision tree.
- **Step 4:** Finally, select the most voted prediction result as the final prediction result.

1. Build random forests : 100 10 $n = 30$

   n_1 n_2

   a) If the number of examples in the training set is $\boldsymbol{N}$, take a sample of $\boldsymbol{n}$ examples at random - but with replacement, from the original data. This sample will be the training set for generating the tree. 10

   b) If there are $\boldsymbol{M}$ input variables, $\boldsymbol{m}$ variables are selected at random out of the $\boldsymbol{M}$ and the best split on these $\boldsymbol{m}$ is used to split the node. The value of $\boldsymbol{m}$ is held constant during the generation of the various trees in the forest.

   c) Each tree is grown to the largest extent possible.

Random Forest Algorithm - Steps

100 10 $n = 30$

1. Build random forests :

n_1 n_2

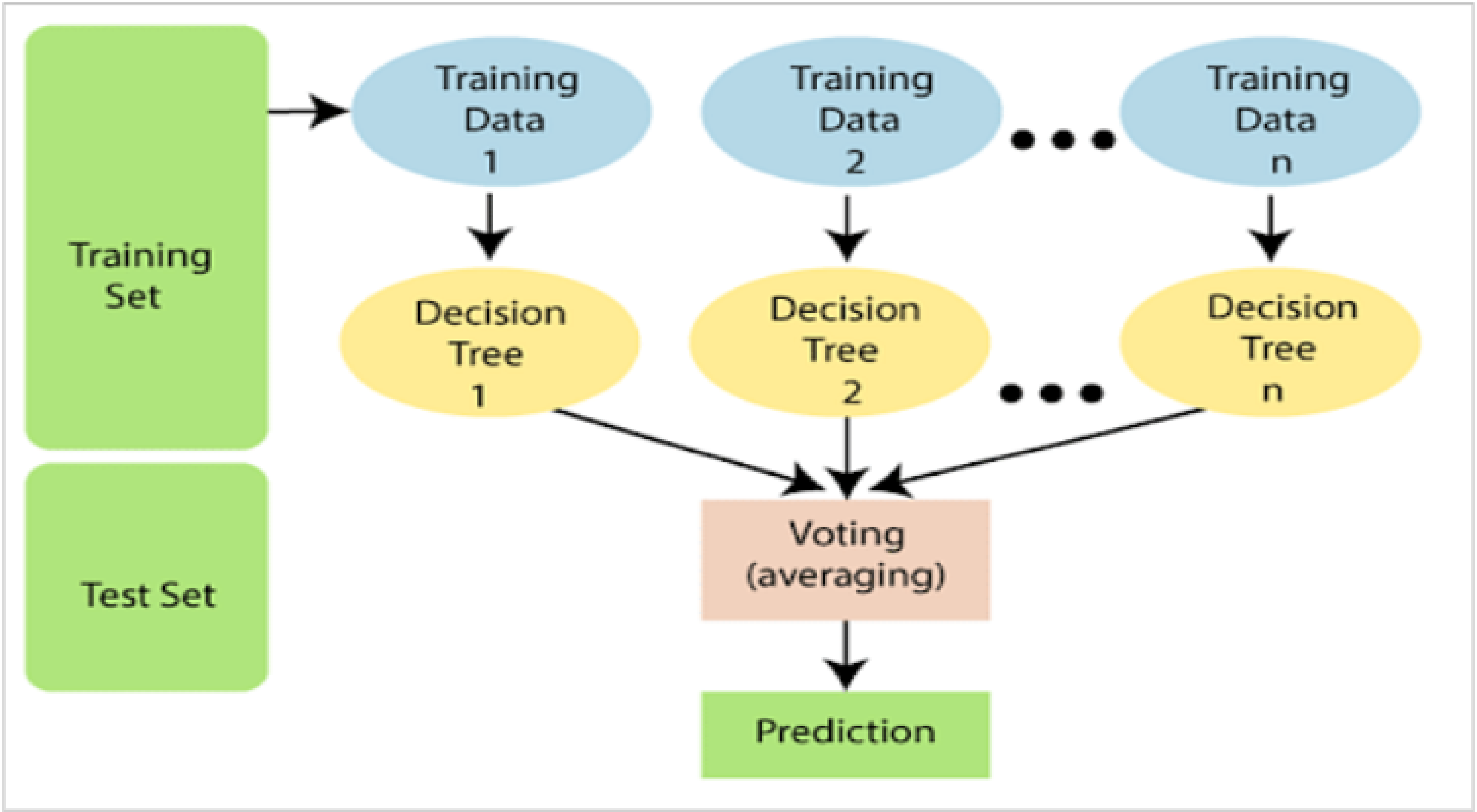
   a) If the number of examples in the training set is N, take a sample of n examples at random - but with replacement, from the original data. This sample will be the training set for generating the tree. 10 5

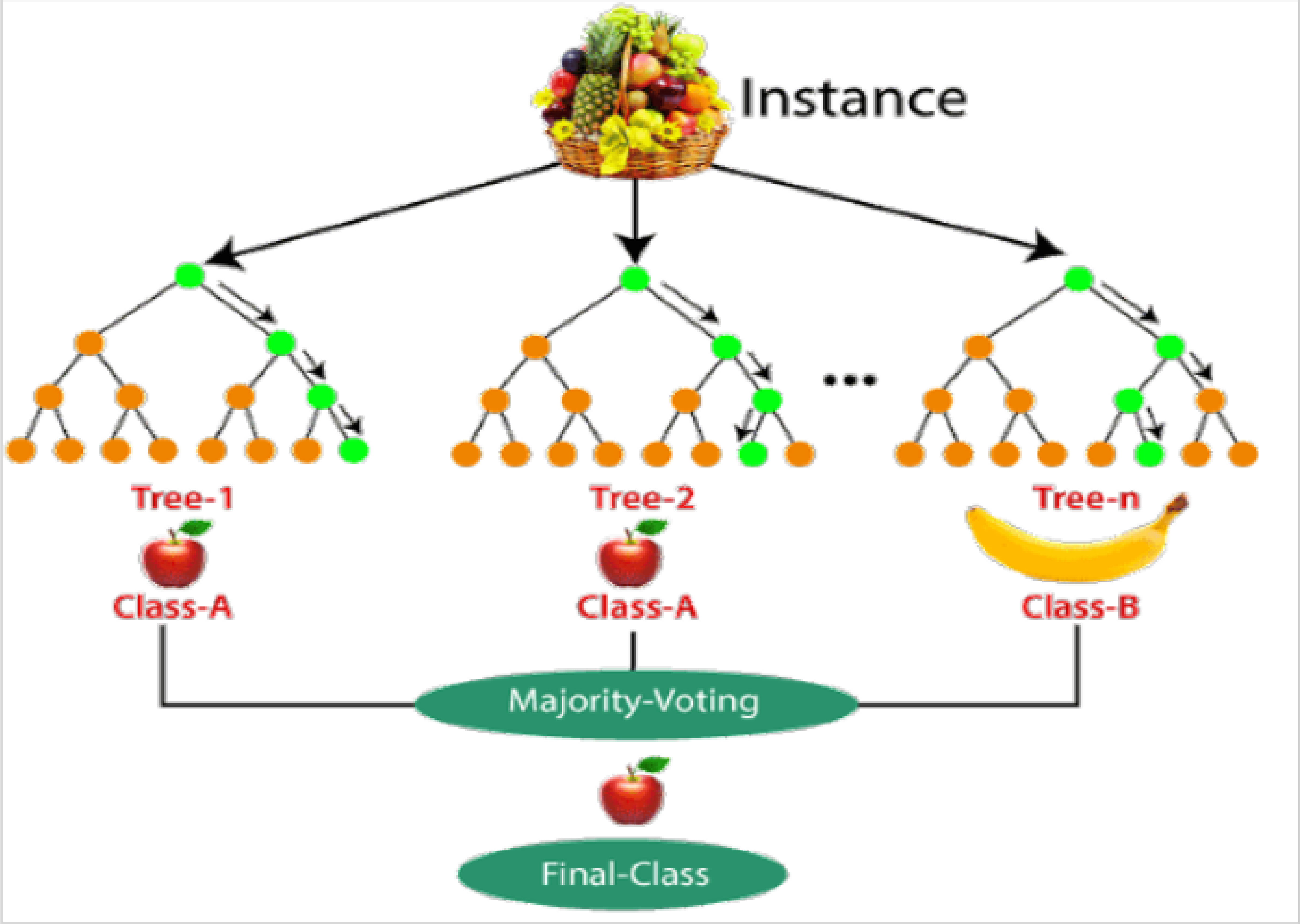
   b) If there are M input variables, m variables are selected at random out of the M and the best split on these m is used to split the node. The value of m is held constant during the generation of the various trees in the forest.

   c) Each tree is grown to the largest extent possible.

2. 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**.

Working of Random Forest Algorithm




Instance
Tree-1
Class-A
Tree-2
Class-A
Tree-n
Class-B
Majority-Voting
Final-Class

id	x_0	x_1	x_2	x_3	x_4	y
0	4.3	4.9	4.1	4.7	5.5	0
1	3.9	6.1	5.9	5.5	5.9	0
2	2.7	4.8	4.1	5.0	5.6	0
3	6.6	4.4	4.5	3.9	5.9	1
4	6.5	2.9	4.7	4.6	6.1	1
5	2.7	6.7	4.2	5.3	4.8	1

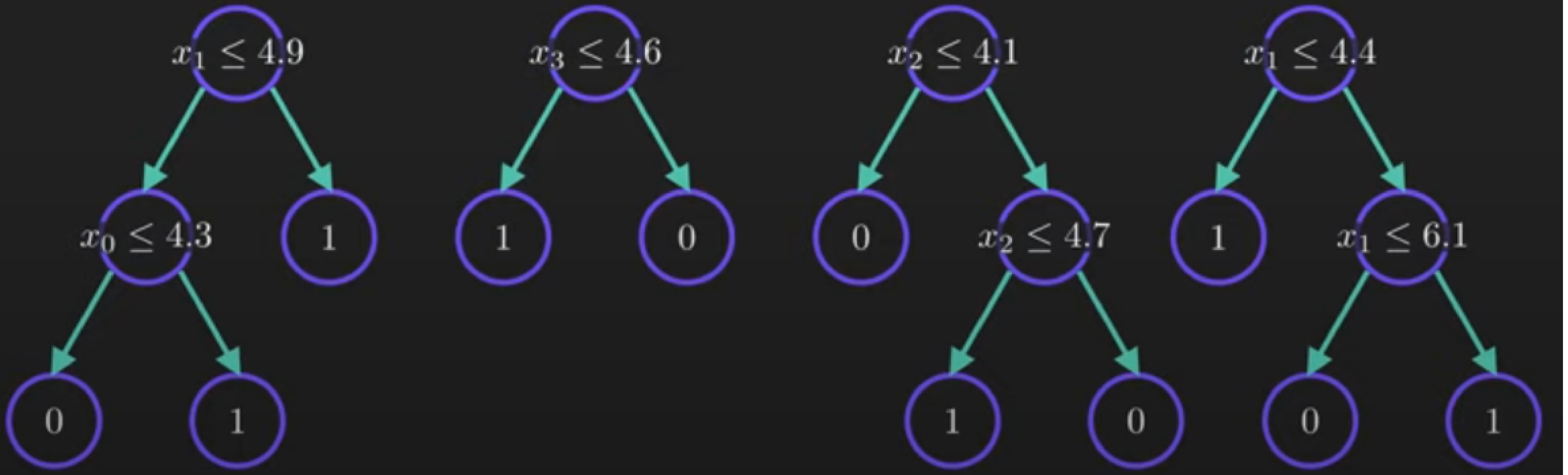
id	id	id	id
2	2	4	3
0	1	1	3
2	3	3	2
4	1	0	5
5	4	0	1
5	4	2	2

x_0, x_1 x_2, x_3 x_2, x_4 x_1, x_3

2.8	6.2	4.3	5.3	5.5

Bootstrap + Aggregating

(Bagging)



1 0 1 1

Why it is called Random?

Because we use two random processes.

1. **Bootstrapping**
2. **Feature Selection**

Why Bootstrapping and Feature Selection?

- **Bootstrapping ensures we are not using the same data for every tree, which helps our model to be less sensitive to training data.**
- **If you use every features, then most of your trees will have same decision node.**

How many features to consider?

- Value close to the log or square root of the total number of features.

Random Forest Algorithm - Strengths

1. It takes less training time as compared to other algorithms.
2. It predicts output with high accuracy, even for the large dataset it runs efficiently.
3. It can also maintain accuracy when a large proportion of data is missing.

Random Forest Algorithm - Weaknesses

1. A weakness of random forest algorithms is that when used for regression they cannot predict beyond the range in the training data, and that they may over-fit data sets that are particularly noisy.

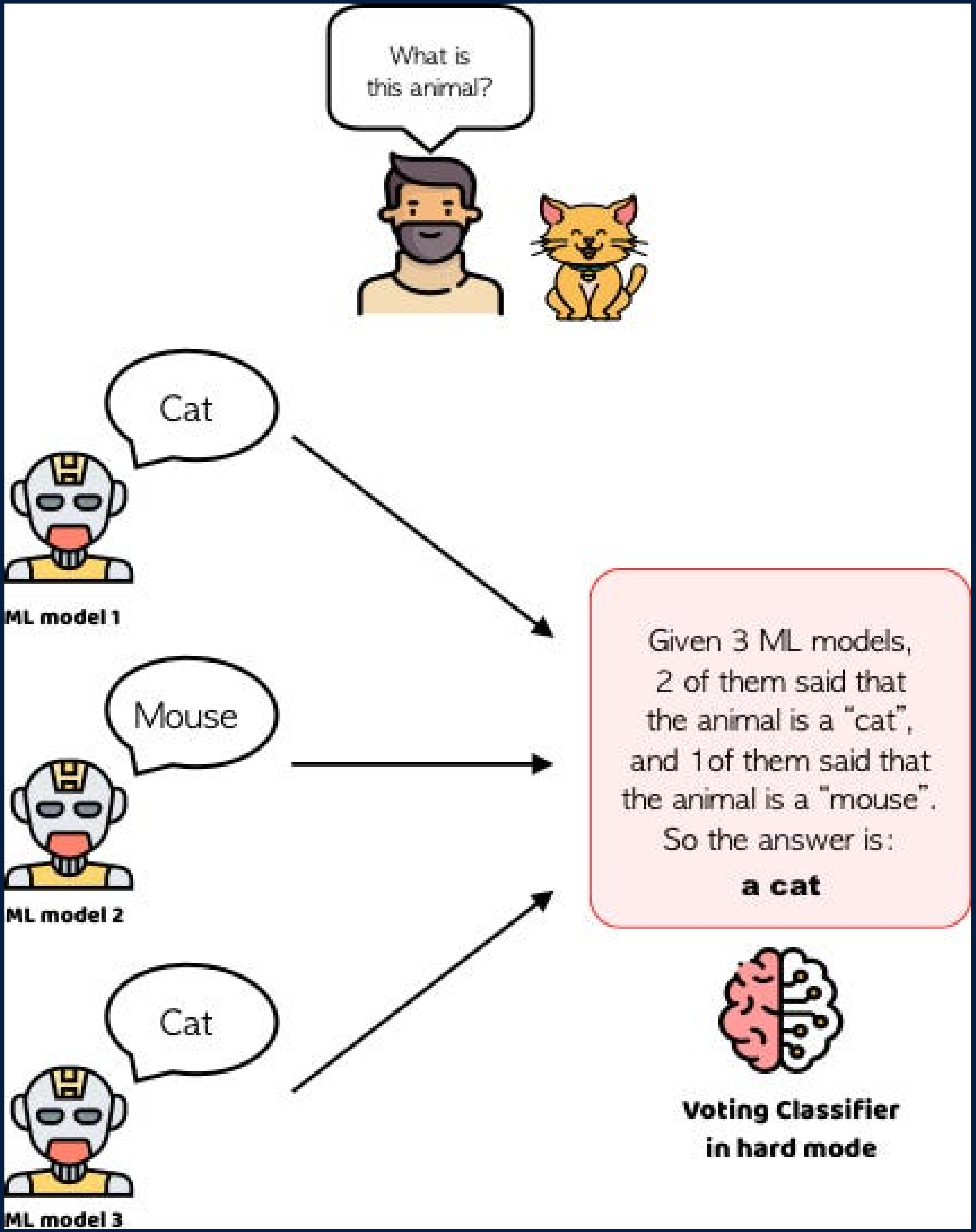
2. The sizes of the models created by random forests may be very large. It may take hundreds of megabytes of memory and may be slow to evaluate.

3. Random forest models are black boxes that are very hard to interpret.

VOTING

Voting

- A Voting Classifier is a machine learning model that trains on an ensemble of numerous models and predicts an output (class) based on their highest probability of chosen class as the output.
- The aggregated decision, whether by majority vote or weighted voting, often yields better generalization and predictive performance than any individual model.

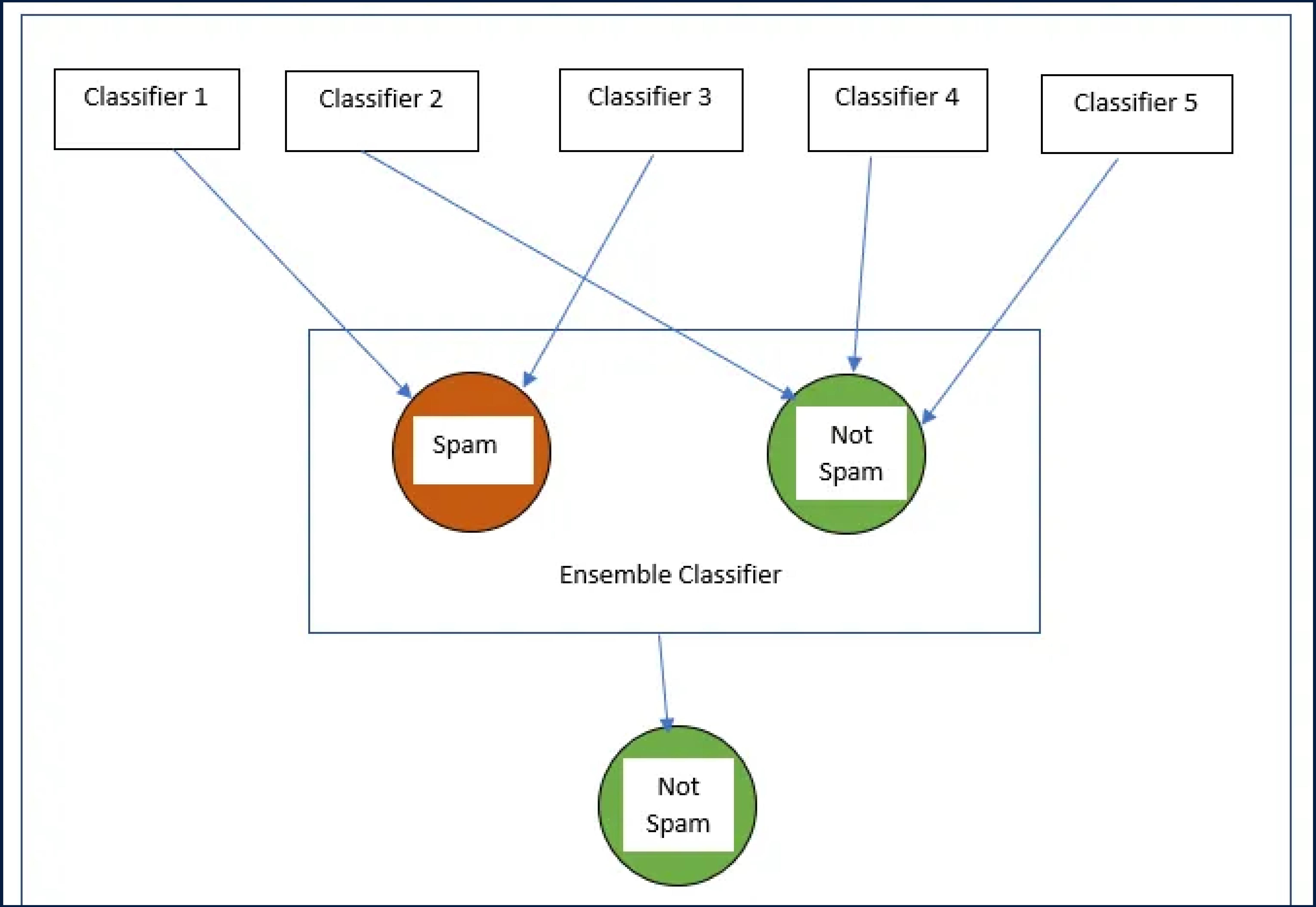

What is this animal?
Cat
ML model 1
Mouse
ML model 2
Cat
ML model 3
Given 3 ML models, 2 of them said that the animal is a "cat", and 1of them said that the animal is a "mouse". So the answer is:
a cat
Voting Classifier in hard mode

Voting Strategies

1. Hard Voting
2. Soft Voting

Hard Voting

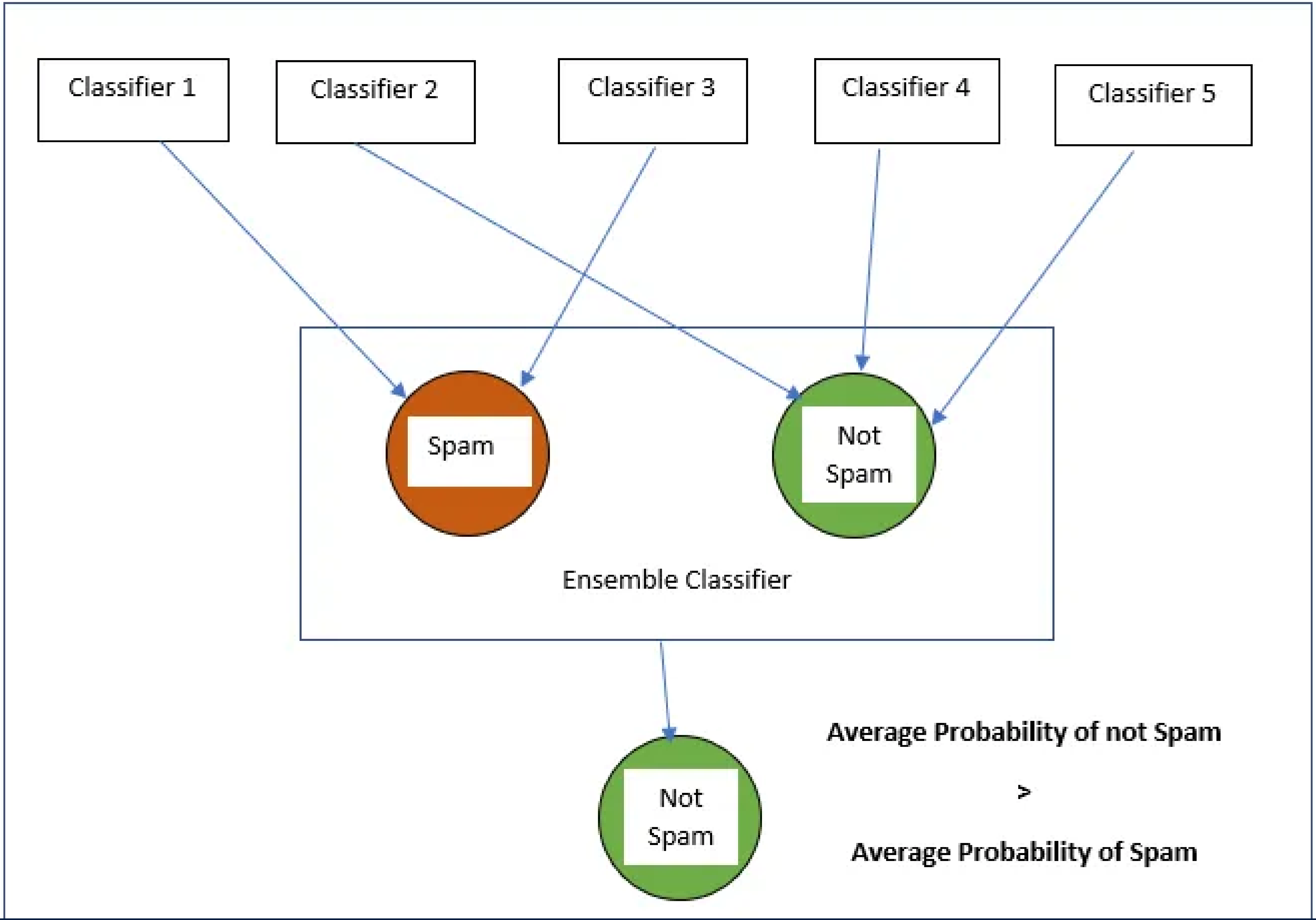
- Hard Voting also known as Majority Voting.
- The class that receives the **majority of votes** is selected as the final prediction. It is commonly used in **classification** problems.
- For example, classifiers predicted the output classes as (Cat, Dog, Dog).
- As the classifiers predicted class “dog” a maximum number of times, we will proceed with Dog as our final prediction.
- In **regression**, it predicts the **average of the individual predictions**.



Classifier 1
Classifier 2
Classifier 3
Classifier 4
Classifier 5
Spam
Not Spam
Ensemble Classifier
Not Spam

Soft Voting

- Soft voting is also called weighted voting.
- Each classifier assigns a probability to each class, and the ensemble's prediction is the class with the highest total probability.
- The average probabilities of the classes determine which one will be the final prediction.
- For example, let's say the probabilities of the class being a "dog" is (0.30, 0.47, 0.53) and a "cat" is (0.20, 0.32, 0.40).
- So, the average for a class dog is **0.4333**, and the cat is **0.3067**, from this, we can confirm our final prediction to be a dog as it has the highest average probability.


Classifier 1
Classifier 2
Classifier 3
Classifier 4
Classifier 5
Spam
Not Spam
Ensemble Classifier
Not Spam
Average Probability of not Spam
>
Average Probability of Spam

Voting Classifier – Combines multiple classifiers for classification tasks.

Voting Regressor – Combines multiple regressors for regression tasks.