Modul 3

High Performance Model

Data Science Program

Purwadhika
Startup and Coding School

High Performance

- What do we mean by performance?
- Is it speed? Accuracy?

- We want model that give us **high accuracy.**

- How do we know if the accuracy is **high enough**?

Introduction

- High performance model means we need to do **optimization**. And it is a hard thing to do!

- How do we get high performance?
  - Improve Performance with data.
  - Examine your nature of data and apply model for it.
  - Tuning Algorithm.
  - Start with simple, and improve later.

Purwadhika
Startup and Coding School

Improve by Data

- Add new data points
- Clean your dataset
- Transform your data
- Add new feature(s)
- Feature Selection / Feature Extraction
- Etc.

Use Existing Models

- Many **researches** produce new models / new architectures for specific cases with specific dataset.

- Even when the model is not specifically addressing your dataset or prediction goals, you can do **transfer learning** for it.
- Example: Image Recognition on Deep Learning case.

- Notes: Always look for latest researches for your problem and you can use other ideas to solve your problem!

Use Existing Models

- For example, Inception v4 architecture by Google.
- Google's goals: Classify images to one of 1000 classes.
- Our goals: Classify images to only 10 classes (not included in Google's).
- How?
- Retrain the model to get new weights.

Algorithm Tuning

- We already got an **acceptable** accuracy, but we want to **boost** it without changing the algorithm.

- How?
- We choose the hyperparameters for our algorithm.

- We tried to find the best possible hyperparameters.

Parameter vs Hyperparameter

- Parameter
- ML model done training to minimize error by **learning** some **characteristics**.

- Hyperparameter
- Can’t be obtained by learning from the training process and it express higher level properties.

Parameter

- “Live inside the model”
- It is a standard that exist on model and it is required by model when making prediction.

- Estimated from learning process
- If we done training it means we change the parameter.

Parameter

- Example:

- Weights in Perceptron / Neural Network
- Support Vectors in SVM
- Means in k-Means Clustering

Purwadhika
Startup and Coding School

Hyperparameter

- External to model
- We can’t estimate hyperparameter from learning process of the model.

- Manually set before the training process
- Often, we have to decide the value of hyperparameter by ourselves, which sometimes suboptimal.

Hyperparameter

- Example:

- Learning Rate in Perceptron / Neural Network
-
- C, gamma, and kernel in SVM
- k in k-Means Clustering

Hyperparameter

- We can search for the best possible hyperparameters for our solutions.

- There are two search algorithm that tried possible combination of hyperparameters:
- Grid Search
- Randomize Search

Purwadhika
Startup and Coding School

Ensemble

- You have several models that were build but non-optimal accuracy.
- You want to leverage each superiority as well as decrease each disadvantage.
- We can combine these classifiers using several ensemble methods.

Ensemble Idea

- You have the task to predict Astra International stock for the next years. In doing so, you can ask input from various people from various domains.

- We consider these 4-independent perspective:
  - Employee
  - Competitor
  - Market Researcher
  - Trader

Ensemble Idea

1. Employee of AI knows the inside culture and company plans ahead, but lacking outside perspective. He can predict **80%** stock performance based on that info.

2. Employee of AI’s competitor know their company info and how to beat AI, but they don’t know AI inside condition. He can predict **60%** AI’s stock performance based on that.

Ensemble Idea

3. Market researcher knows the industry and customer preferences toward product toward future. But he doesn't know any info on inside the company. He can predict the stock **70%** accuracy.

4. Stock trader know the trend of stock price and macro economic trend affect stock price. He can predict the stock price with **70%** accuracy.

Ensemble Idea

- Now, if I have the access to all those information. I can surely predict better the stock price with this accuracy.

$$\text{Accuracy} = 1 - 0.2*0.4*0.3*0.3 = 99.28\%$$

- But life is not that simple!
- Many information based on similar information thus is not independent. But the idea is we still can **combine** these knowledge.

Purwadhika
Startup and Coding School

Benefit

- + Improve Performance
- + Learn complicated problems by combining simple classifier.
- + Speedy execution.

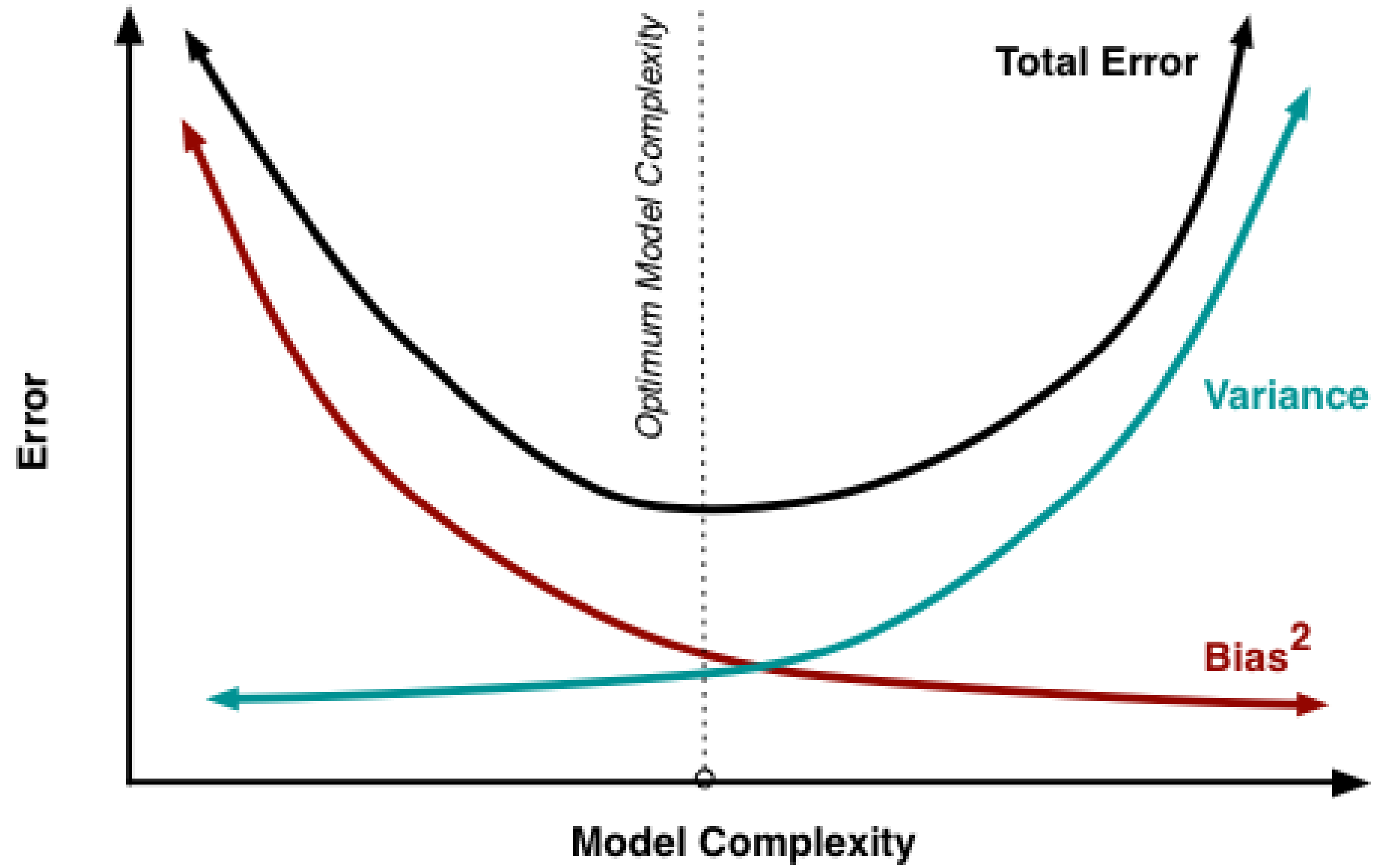
Bias-Variance

- Model source of error:

$$Err(x) = \left(E[\hat{f}(x)] - f(x)\right)^2 + E\left[\hat{f}(x) - E[\hat{f}(x)]\right]^2 + \sigma_e^2$$

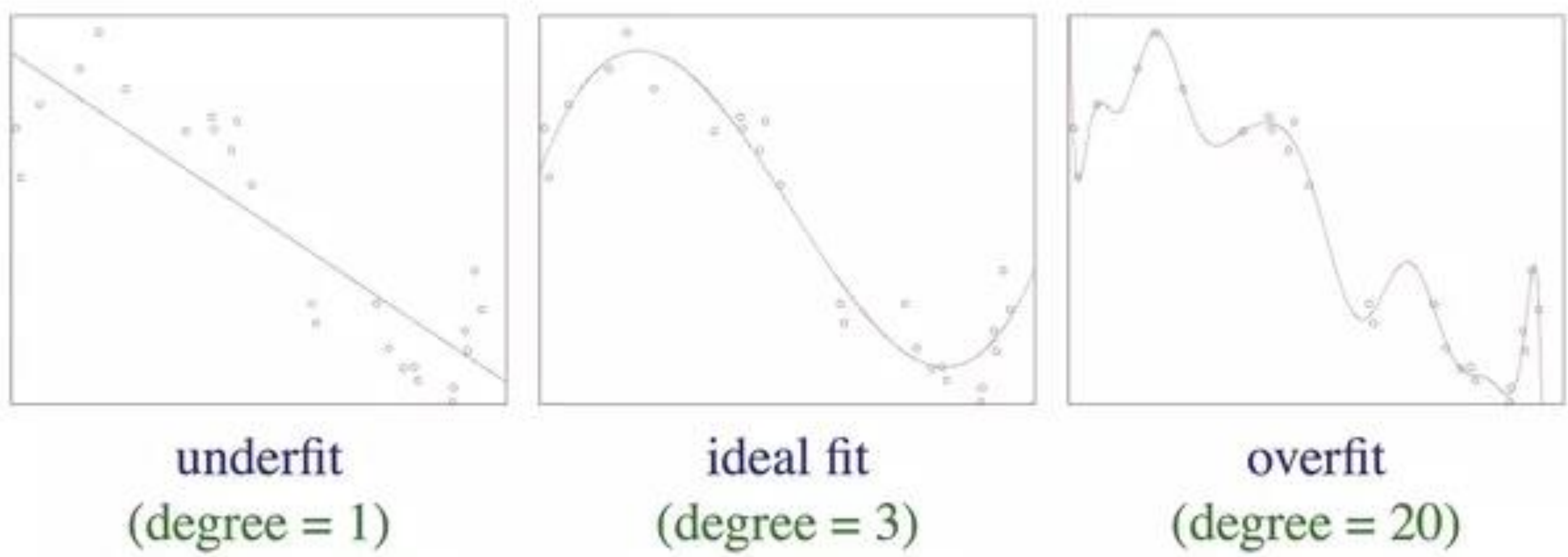
$$Err(x) = \mathrm{Bias}^2 + \mathrm{Variance} + \mathrm{Irreducible\ Error}$$

Bias-Variance



Purwadhika
Startup and Coding School

Bias-Variance



Ensemble

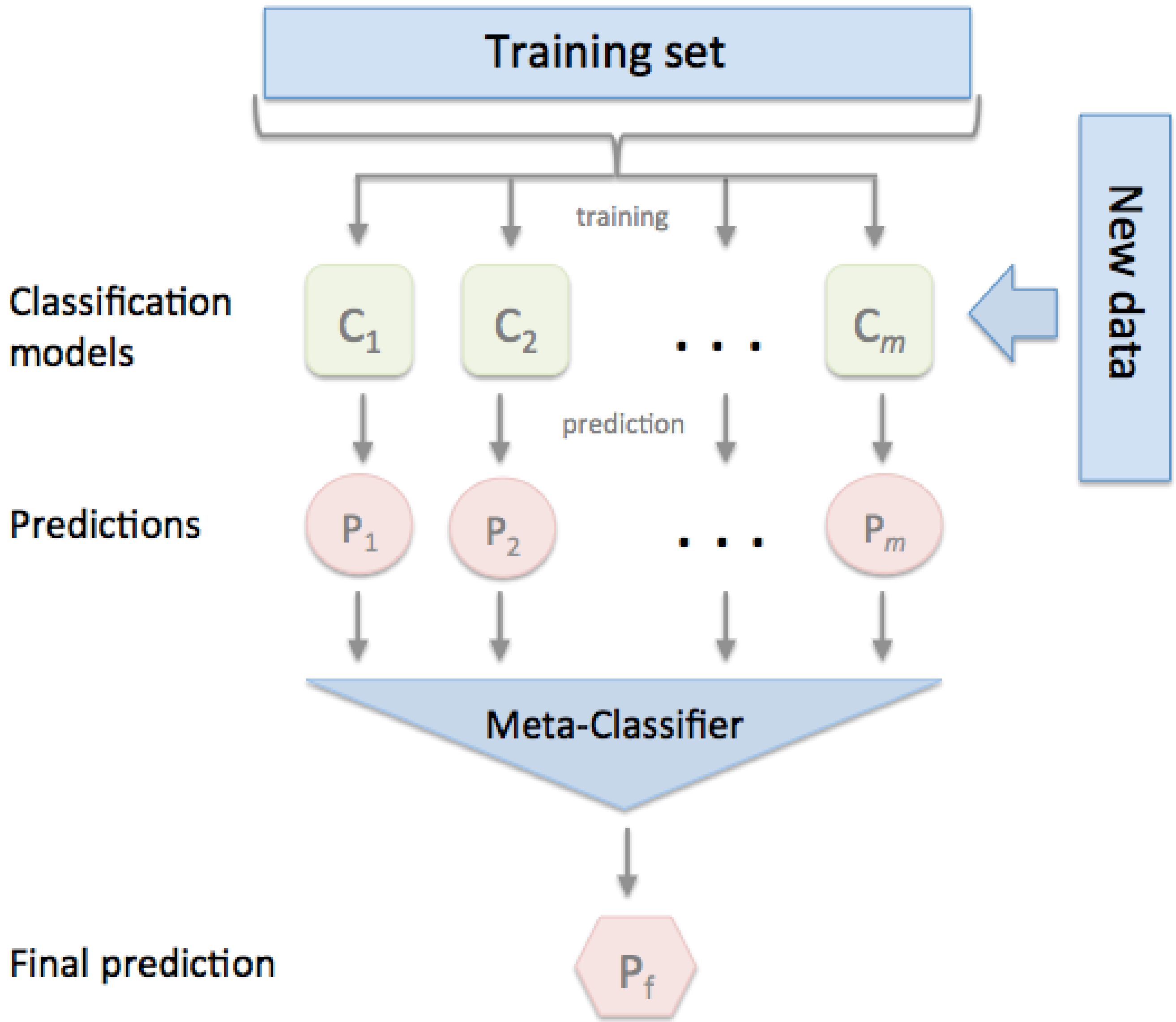
- There are three basic techniques that we will discuss in this session:

1. Stacking
2. Boosting
3. Bagging

Stacking

- Create base model with different settings for each (algorithm, hyperparameter, etc.)
- Stacking algorithm is a meta algorithm that trained based on the output of the base models.
- Use the output of the base models for the input for stacking algorithm as features.

Stacking



urwadhika
rtup and Coding School

Stacking

- Two different example for stacking:

1. Stack Classifier (Logistic Regression)
2. Voting Classifier (Majority Vote)

Purwadhika
Startup and Coding School

Boosting

- Convert weak learners to strong learners (boost it) iteratively. Weak classifier means only slightly better than baseline.

- **Sequence** ensemble that attempt to correct the mistakes of previous models before them in sequence.

- Weight the data point and calculate the next weight of the data.

AdaBoost

- Adaptive Boosting is one of the widely used boosting algorithm out there.
- Calculate weight of each data point depend on misclassified or not for the next iteration.
- Prediction are combined through weighted vote.

Purwadhika
Startup and Coding School

AdaBoost

1. **Sample** the training set according to a set of object weights (initially equal)
2. Use it for **training a** simple (weak) **classifier** w_i
3. **Classify** the entire data set, using the **weights error** ϵ_i

   Store **classifier weight** $\alpha_i = 0.5 \log(\frac{1-\epsilon_i}{\epsilon_i})$

4. **Multiply weights** of erroneously classified objects with $\exp(\alpha_i)$
5. Multiply weights of correctly classified objects with $\exp(-\alpha_i)$
6. **Iterate from 1**
7. **Final classifier**: weighted voting, weights α_i

Purwadhika
Startup and Coding School

Bagging

- Bootstrap Aggregating is a technique to **reduce variance** by average multiple estimate.

- Bagging uses bootstrap **sampling with replacement** to obtain the data subsets for training the base learners. For aggregating the outputs of base learners, bagging uses voting.

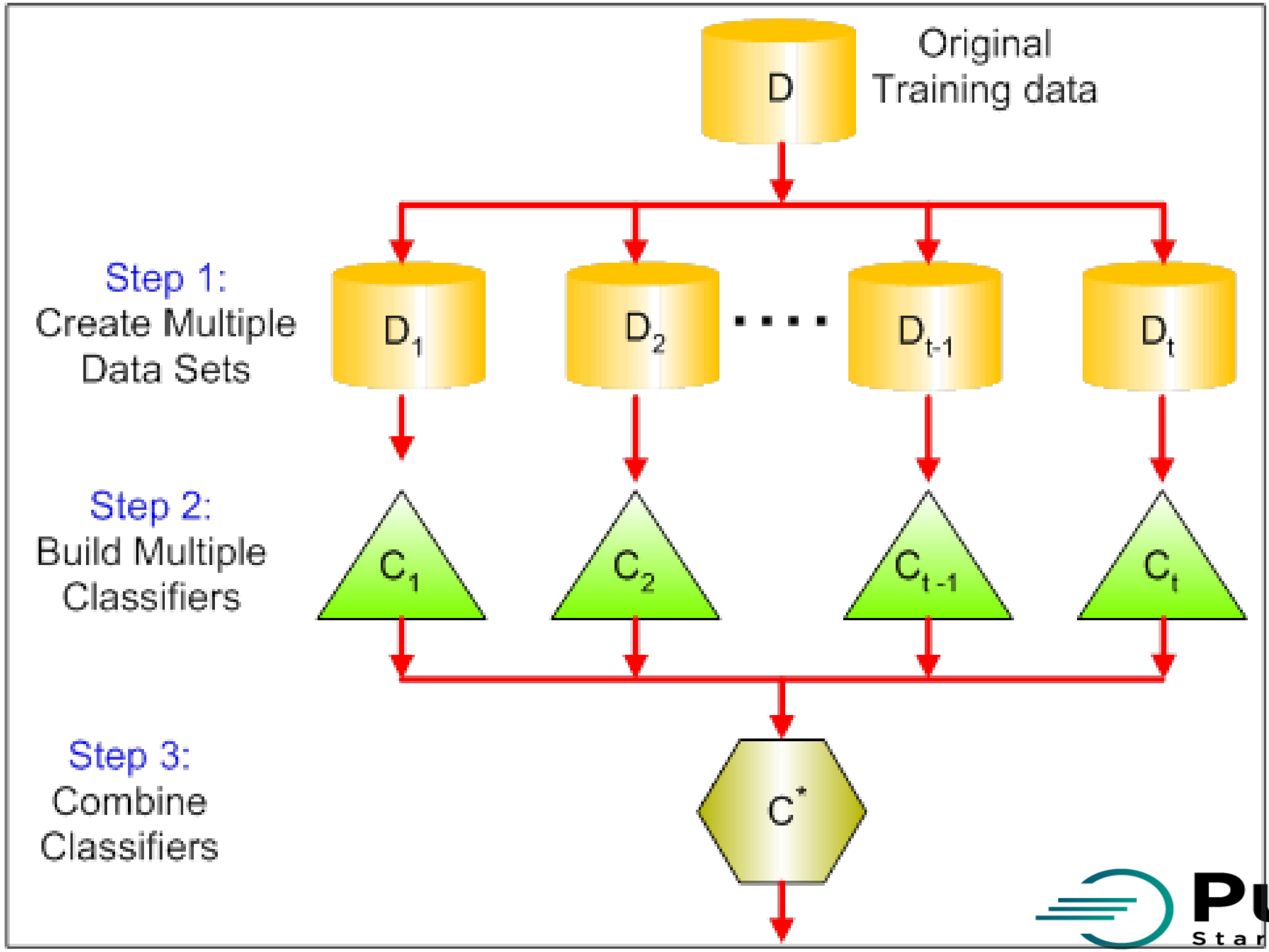
Bagging

- Bootstrap is a method to estimate quantity from data sample.

- Example:
- We have 100 samples data and want to calculate mean.
- But we know, because our data is small, our calculated mean might not the true mean.
- We can divide to sub samples, let say 5 samples, and calculate each mean’s sample.
- The means are 5, 5.5, 4, 4.5, and 3.6. Then the mean would be 4.52

Purwadhika
Startup and Coding School

Bagging



Purwadhika
Startup and Coding School

Random Forest

- Forest: a collection of tree.
- Random: chosen without method or conscious decision.
- Random Forest: a collection of (decision) tree with a random sample of features & training data each.
- But actually it is not *that random.*

Random Forest

- Bagging for decision tree are grown deep and correlated to each other, which result to high variance error.

- Random Forest is the improvement of bagging for decision tree. Minimize correlation between trees so that it is uncorrelated or at least weakly correlated.

- As a result, RF should provide higher accuracy (if not similar) to bagging DT.

Purwadhika
Startup and Coding School