Modul 3

Supervised Learning 1

Data Science Program

Purwadhika
Startup and Coding School

Agenda

Session 1: Predictive Modeling, Machine Learning

Session 2: Logistic Regression + Exercise

Session 3: Decision Tree + Exercise

Session 4: Assignment, Recap of the day

Purwadhika
Startup and Coding School

Objective

Understand concept of Machine Learning

Capable to perform predictive modeling (classification)

Capable to evaluate model quality

Familiar with LogReg and Decision Tree algorithm

Purwadhika
Startup and Coding School

Regression Overview

Which one is your preference?

1   2   3
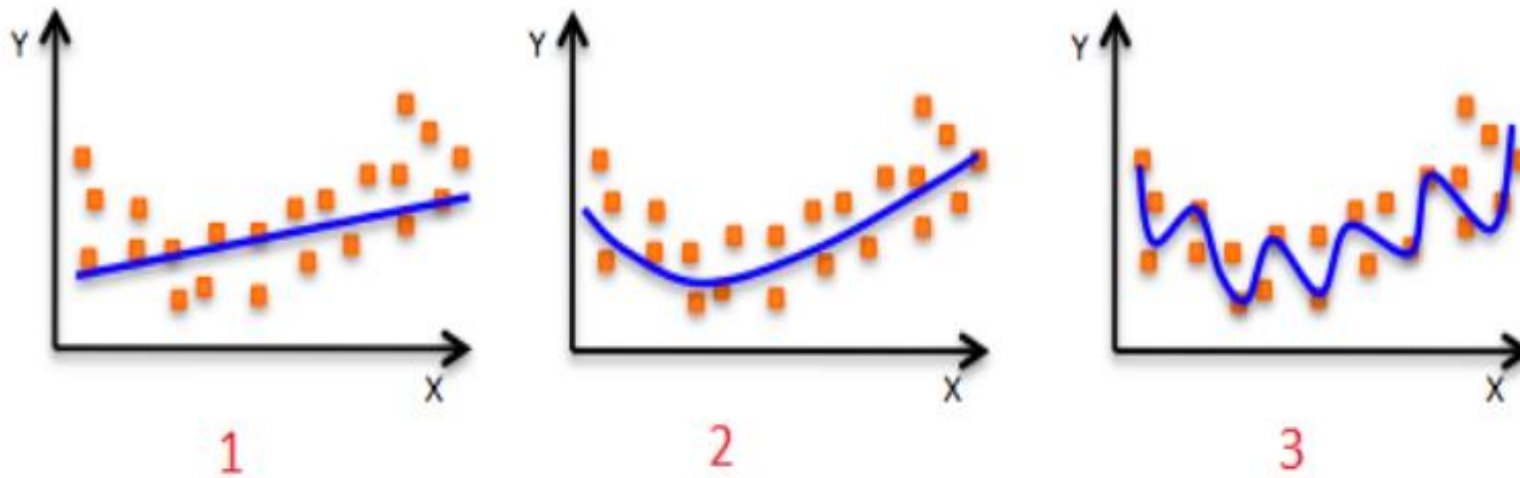
Regression Overview

OVERFITTING

- Fits the data too well
- Too complex
- High variance, low bias
- Learns noise in training data

Solutions:

- ✓ Reduce number of features
- ✓ Increase training samples

Question: How do we know that our model is overfitting?

UNDERFITTING

- Simple model
- High bias, low variance
- Doesn't capture underlying data trend

Solutions:

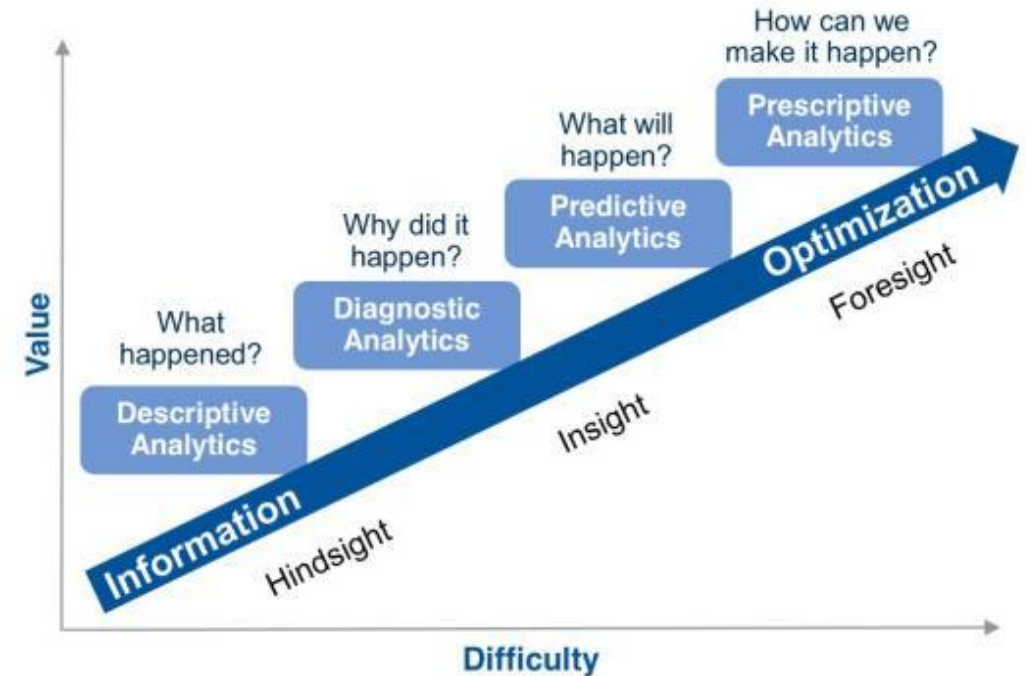- ✓ Train more complex model
- ✓ Obtain more features

Purwadhika
Startup and Coding School

Session 1

Data Analysis
Predictive Modeling
Machine Learning Concept

Data Analysis
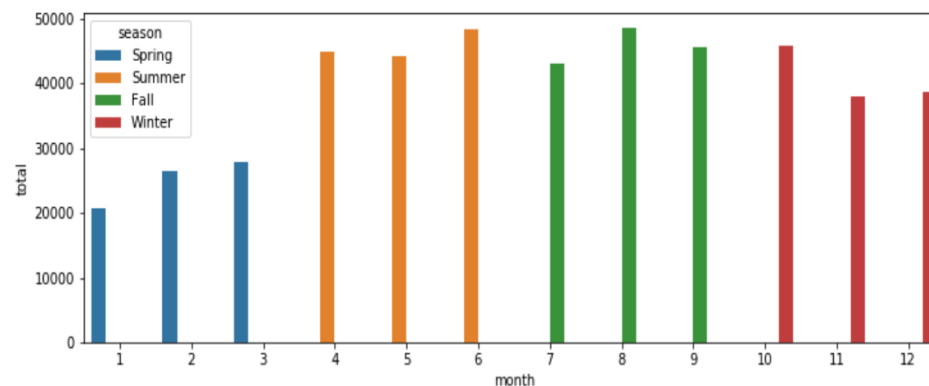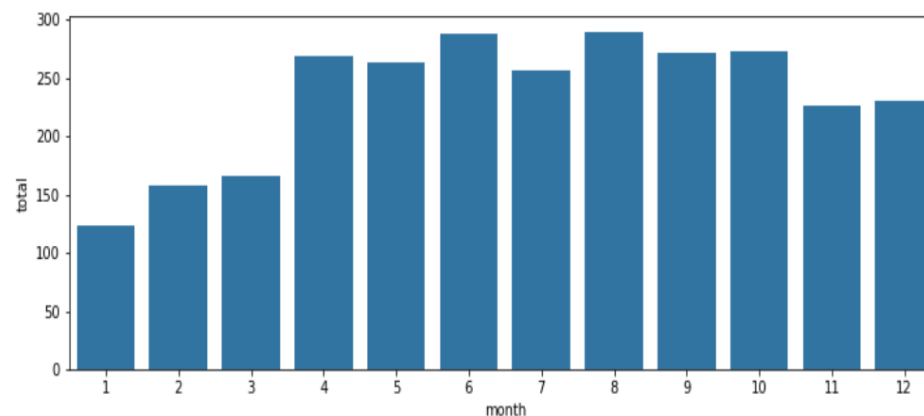
Types of Data Analysis

- **Descriptive** (BI, Data mining)
  Understand historical data.
  Look for reasons behind past success/failures.

- **Predictive** (Forecasting)
  Determine probable future outcome.

- **Prescriptive** (Optimization)
  Goes beyond predicting future outcome.
  Suggest action to benefit from prediction.

GOAL : Get actionable insights, smarter decision, better business outcomes

Purwadhika
Startup and Coding School

http://www.datascienceassn.org/content/descriptive-predictive-prescriptive-analytics

Descriptive Analytics

Bike Sharing usage
In Washington DC

Usage peaks during Summer and Fall

Purwadhika
Startup and Coding School

Predictive Modeling

Churn Analysis

Credit Scoring

Purwadhika
Startup and Coding School

Predictive Modeling

Likely to respond?

Low **Medium** **High**
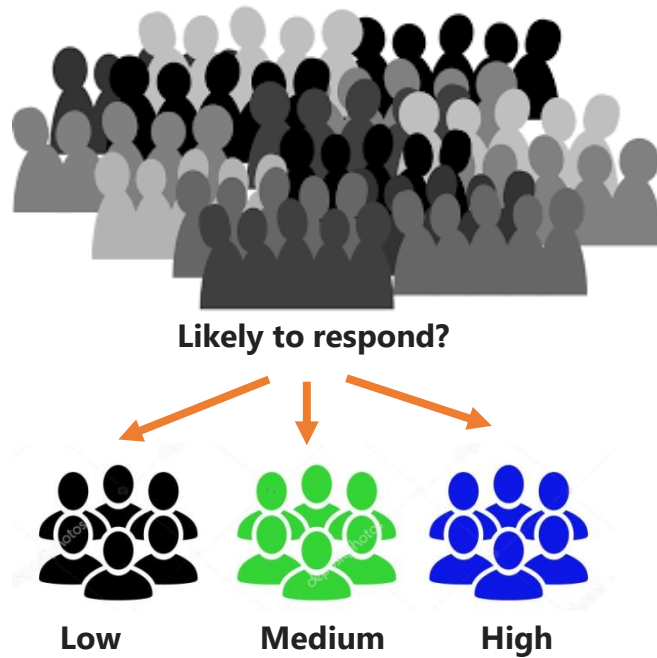
Propensity Analysis

Human Resources

"The Rising Star"

Purwadhika
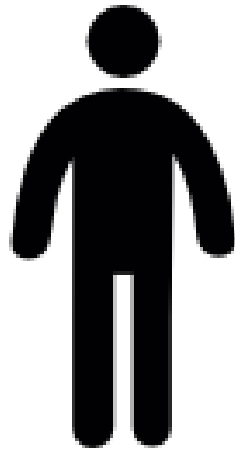Startup and Coding School
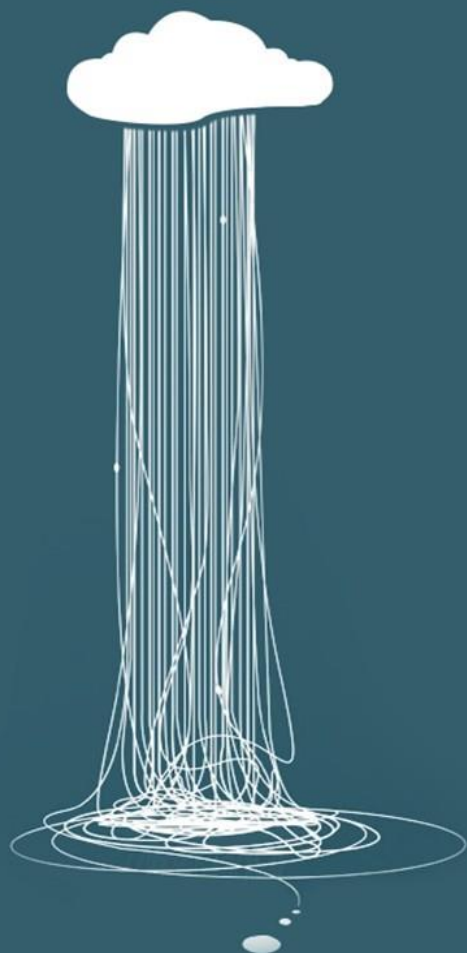
Will this evening be raining?

Machine Learning

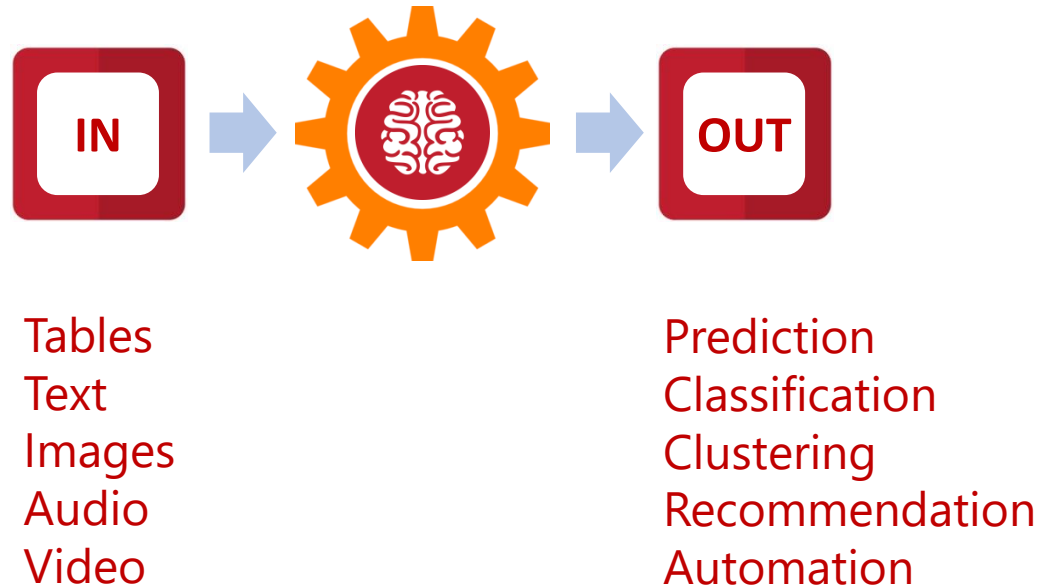What's the difference between **Human** and **Computer**?

VS

Learn from experience

Follow instruction

Learn from ~~experience~~ data

Purwadhika
Startup and Coding School

Machine Learning

Technique to give computers ability to **learn from data** without explicitly programmed

IN → 🧠 → **OUT**

Tables
Text
Images
Audio
Video

Prediction
Classification
Clustering
Recommendation
Automation

More Applications:
- Recommendation system
- Shopping basket analysis
- Decision making for self-driving cars

What is not Machine Learning?
- Calculating average
- Most occurring events
- Calculate highest grade

Purwadhika
Startup and Coding School

Illustration

Udemy

$10.99 ~~$199.99~~

★★★★☆ 4.3
(502 ratings)

The Complete Machine Learning Course with Python

Frequently Bought Together

$19.98 Total ~~$394.98~~

Buy Now

The Complete Machine Learning Course with...
Codestars by Rob Percival, Anth...
★★★★☆ 4.3 (508)
~~$199.99~~

Machine Learning with TensorFlow + Real-Life...
365 Careers, 365 Careers Team
★★★★★ 4.6 (409)
~~$194.99~~

BukaLapak

Rp435.000

Durable Selimut Mobil Car Body Cover For Daihatsu Ayla

Barang Terkait

Durable Selimut Mobil Car Body Cover For Daihatsu Terios &
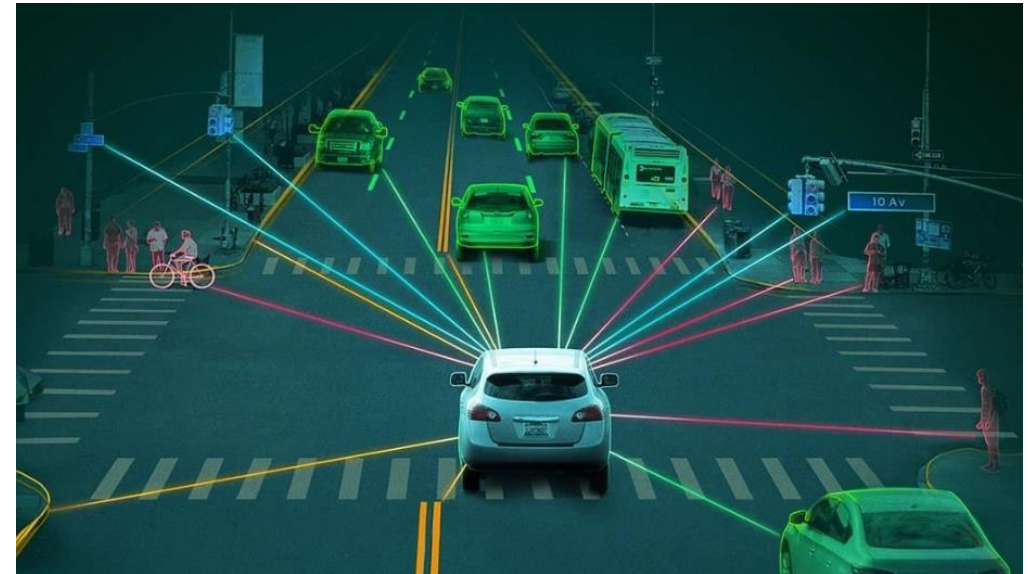Rp542.000
Cicilan mulai 45rb/bln

★★★★★ 1 ulasan
Cover Mobil Daihatsu Ayla
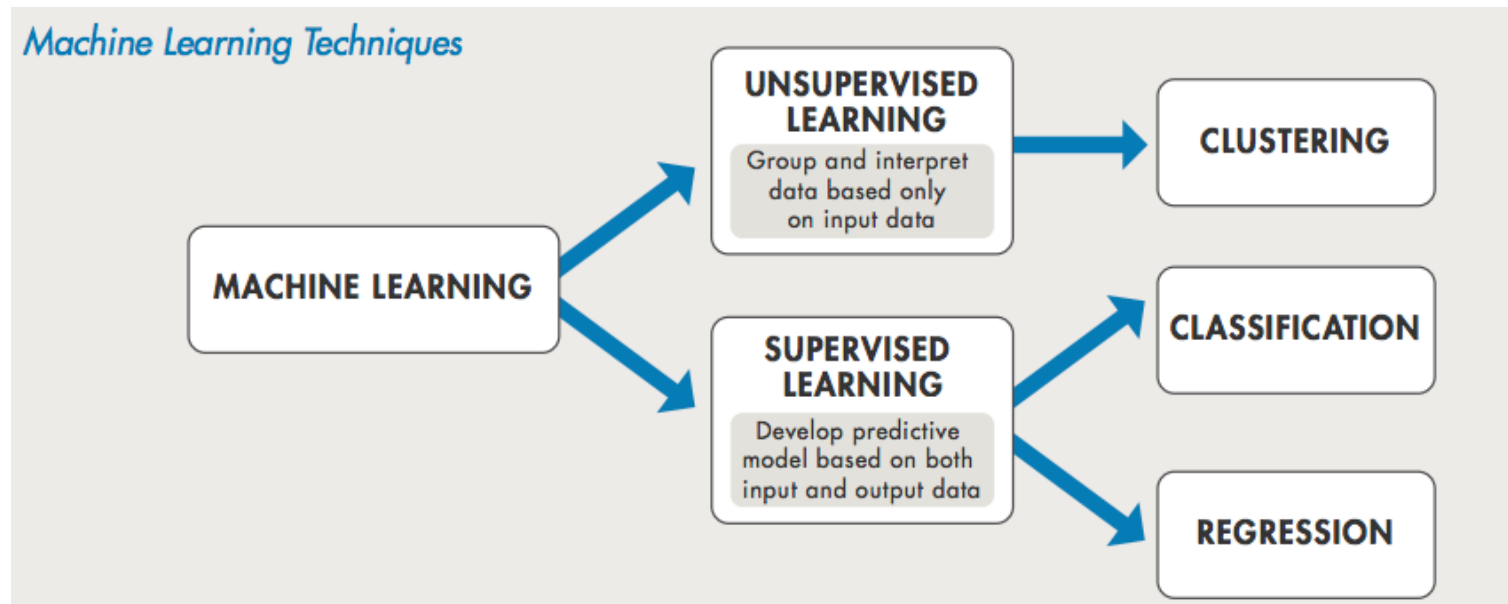Rp650.000
Cicilan mulai 54rb/bln

Purwadhika
Startup and Coding School

Illustration

What self driving cars see

Purwadhika
Startup and Coding School

Techniques

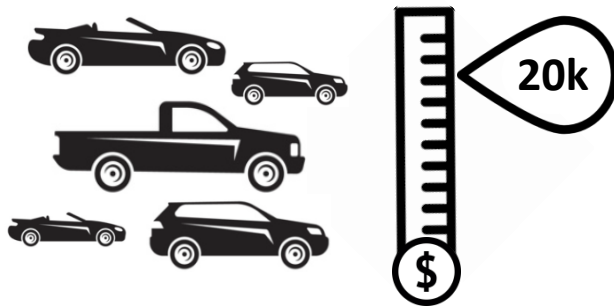Some call it Machine Learning Algorithm

Unsupervised Learning: Do not have target variable to estimate / predict.

Supervised Learning: Generate function that **map** inputs to desired outputs.
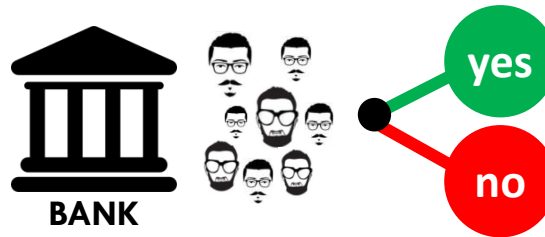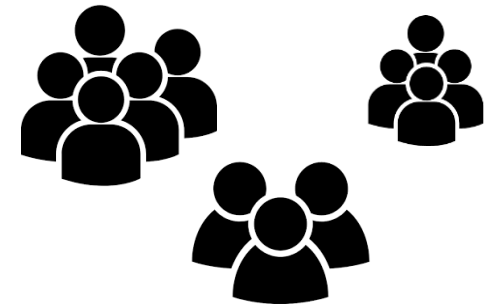
Applications

REGRESSION
PREDICT VALUE

CLASSIFICATION
PREDICT CATEGORY

BANK

yes

no

CLUSTERING
GROUP OBJECT

Purwadhika
Startup and Coding School

Illustration

- **Regression**

BRAND	TYPE	CYLINDER	ENG-SIZE	STROKE	MAX-RPM		PRICE
Brand-A	sedan	four	109	3.4	5500		13950
Brand-A	sedan	five	136	3.4	5500		17450
Brand-B	sedan	four	108	2.8	5800		16430
Brand-B	sedan	four	108	2.8	5800		16925
Brand-C	hatchback	three	61	3.03	5100		5151
Brand-C	hatchback	four	90	3.11	5400		6295
Brand-D	hatchback	four	90	3.23	5500		5572
Brand-D	hatchback	four	90	3.23	5500		6377

$f(x)$

Input
Predictor
Independent Var
X

Output
Response
Dependent Var
Y

Purwadhika
Startup and Coding School

Illustration

- **Classification**

| BRAND	TYPE	CYLINDER	ENG-SIZE	STROKE	MAX-RPM		RISK
Brand-A	sedan	four	109	3.4	5500		POS
Brand-A	sedan	five	136	3.4	5500		POS
Brand-B	sedan	four	108	2.8	5800		POS
Brand-B	sedan	four	108	2.8	5800		POS
Brand-C	hatchback	three	61	3.03	5100		NEG
Brand-C	hatchback	four	90	3.11	5400		NEG
Brand-D	hatchback	four	90	3.23	5500		NEG
Brand-D	hatchback	four	90	3.23	5500		NEG

$f(x)$

Input
Predictor
Independent Var
X

Output
Response
Dependent Var
Y

Illustration

- **Clustering**

Input
Predictor
Independent Var
X
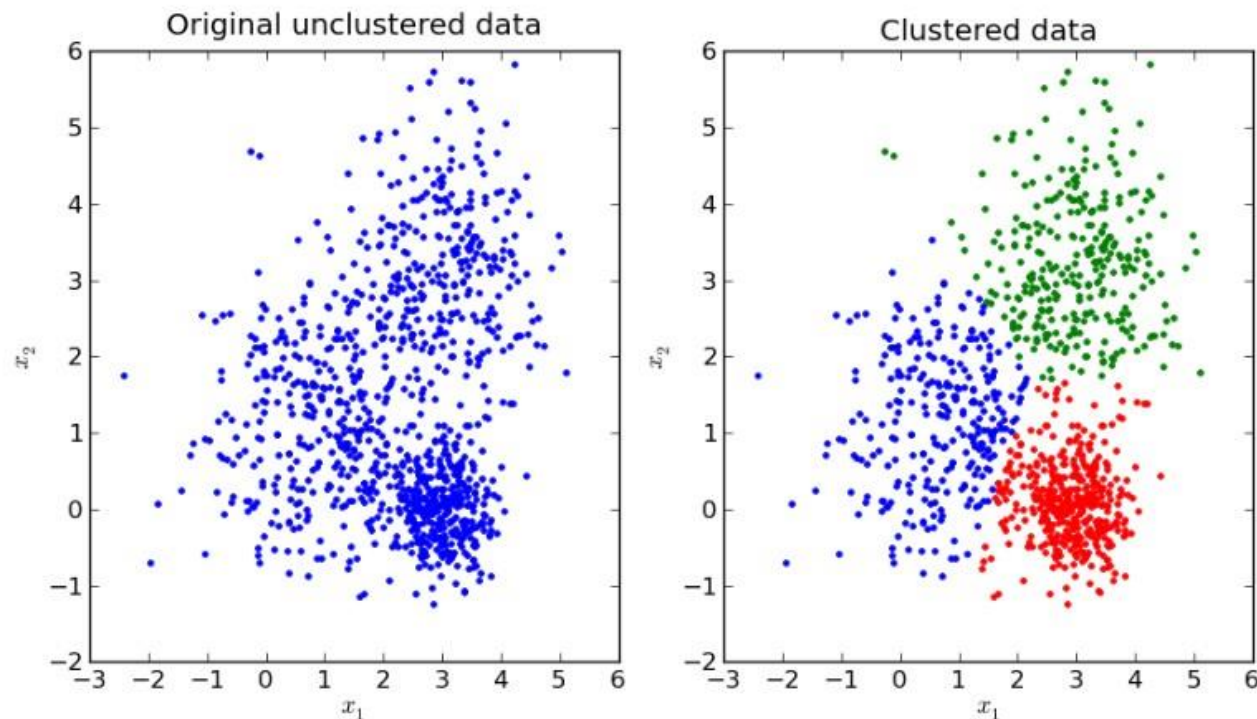
OUR FOCUS
Binary Classification

Session 2

Logistic Regression

Predictive Model

Today we're focusing on two predictive Models
- Logistic Regression
- Decision Tree

Practical session
- Implement CRISP-DM
- Goal: Actionable insights

Note:
Other models: Discriminant Analysis, K-Nearest Neighbour (KNN),
Support Vector Machine (SVM), Ensemble – Bagging, Random Forest, Boosting, etc

Purwadhika
Startup and Coding School

Why not Linear Model?

Simple Linear Model

$$y = \alpha + \beta x + \varepsilon$$

where,
y: Response variable (continuous)
x: Predictor variable

Why can't we use Linear Regression to model binary responses?
- Model must produce predicted probability that are between 0 and 1
- Linear model produces responses that vary from –inf to +inf
- Response (y) is not normally distributed
- Variability of Response (y) is not constant

Purwadhika
Startup and Coding School

GLM – Generalized Linear Model

- **GLM** is a **flexible generalization of ordinary linear regression** that allows for response variables that have **error distribution models other than a normal distribution.**

- GLM generalizes linear regression by allowing the linear model to be related to the response variable via a ***link function***

$$g\big(E(y)\big) = \alpha + \beta x + \varepsilon$$

$g()$ is the *link function*

Link function relates the **expected value of the response** to the **linear predictors**

GLM introduced by John Nelder and Robert Wedddernburn in 1972

Purwadhika
Startup and Coding School

GLM – Generalized Linear Model

- In logistic regression, expected value of response is probability of **'success'** event to occur.

    - 'Success' → Y=1.   P('Success' | x) = $\pi(x)$

    - 'Failure' → Y=0.   P('Failure' | x) = 1- $\pi(x)$

- Link function relates **expected value** of the response to the **linear predictor**.

- Link function for Logistic Regression is **Logit**

$$logit[\pi(x)] = ln\left[\frac{\pi(x)}{1 - \pi(x)}\right] = \alpha + \beta x$$
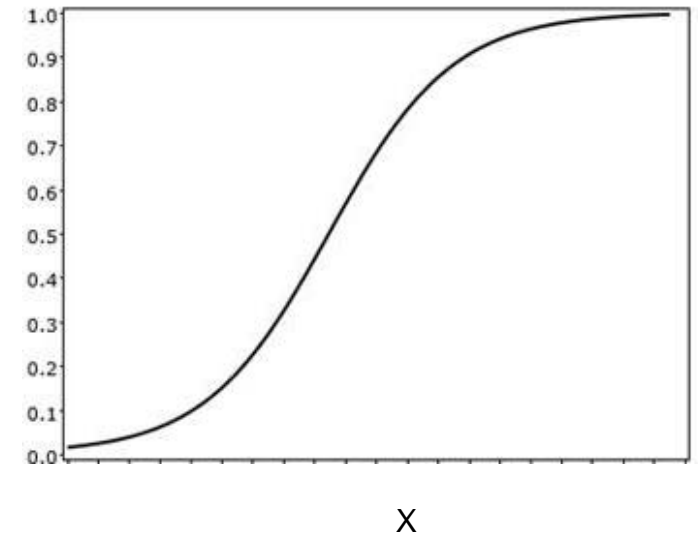
Purwadhika
Startup and Coding School

Logistic Regression Model

$$\text{logit}[\pi(x)] = \log\left[\frac{\pi(x)}{1-\pi(x)}\right] = \alpha + \beta x$$

$$\pi(x) = \frac{\exp(\alpha + \beta x)}{1 + \exp(\alpha + \beta x)}$$

P('Success'|X)

$$\pi(x) = \frac{\exp(\alpha + \beta x)}{1 + \exp(\alpha + \beta x)}$$

X

- With logit link function, $\pi(x)$ varies between 0 and 1
- Value of α and β is obtained using Maximum Likelihood Estimation (MLE). Different to Linear regression which using Ordinary Least Square (OLS)
- MLE link : https://en.wikipedia.org/wiki/Maximum_likelihood_estimation

Purwadhika
Startup and Coding School

Assumptions

- Doesn't need linear relationship between dependent and independent variables. However it assumes **linear relationship** between **link function** and **independent variable** in logit model.

- Error / residuals do not need to be normally distributed

- Homoscedasticity is not required

- Binary Reglog requires dependent variable to be binary. While ordinal reglog requires dependent variable to be ordinal

- Observation to be independent of each other. Not a repeated measurement.

- Requires no or only little of multicollinearity among independent variables.

- Requires large sample data for stable result. Complex issue, but refer to work by Peduzzi et all (1996), minimum sample is N = 10 k / p, where k is number of independent variables and p is the smallest proportion of dependent variable. If result is <100, then make it 100 at the minimum. Long (1997)

Purwadhika
Startup and Coding School

Encoding Categorical Var

- There are ML algorithms that supports categorical value without further manipulation, but there are also algorithms that do not.
- Need to turn categorical value into numerical value
- *One Hot Encoding*: Process by which categorical variables are converted into form that could be accepted by ML algorithm.
- Pandas support encoding using *get_dummies.* It creates dummy variables.

Sample	Category	Numerical
1	Human	1
2	Human	1
3	Penguin	2
4	Octopus	3
5	Alien	4
6	Octopus	3
7	Alien	4

One Hot Encoder

Sample	Human	Penguin	Octopus	Alien
1	1	0	0	0
2	1	0	0	0
3	0	1	0	0
4	0	0	1	0
5	0	0	0	1
6	0	0	1	0
7	0	0	0	1

Model Evaluation

- **Confusion Matrix**

  NxN matrix where N is the number of classes being predicted. Some definitions:

  o **Accuracy** : Proportion of total number of predictions that were correct

  o **Precision** (Positive Predictive Value) : Proportion of positive cases that were correctly identified.

  o **Negative Predictive Value** : Proportion of negative cases that were correctly identified.

  o **Sensitivity (Recall)** : Proportion of actual positive cases which are correctly identified.

  o **Specificity** : Proportion of actual negative cases which are correctly identified.

		Predicted	
		No	Yes
Actual	No	TN	FP
	Yes	FN	TP

TN	True Negative
FP	False Positive
FN	False Negative
TP	True Positive

Model Performance

Accuracy	$= (TN+TP)/(TN+FP+FN+TP)$
Precision	$= TP/(FP+TP)$
Sensitivity	$= TP/(TP+FN)$
Specificity	$= TN/(TN+FP)$

Model Evaluation

Accuracy : Proportion of total number of predictions that were correct

Accuracy (Correct Rate) = (TP + TN) / N

Model Evaluation

Sensitivity : Proportion of total number of predictions that were correct

Sensitivity = TP / (TP + FN)

Model Evaluation

Specificity : Proportion of actual negative cases which are correctly identified

		Predicted	
		No	Yes
Actual	No	TN	FP
	Yes	FN	TP

Specificity = TN / (TN + FP)

Purwadhika
Startup and Coding School

Imbalanced Classification

- It is scenario where the number of observations belonging to one class is **significantly lower** than those belonging to the other class.
- Example: Fraud case, Rare disease identification, Customer Default
- In this situation, predictive model result could be inaccurate.
- ML algorithm are usually designed to improve accuracy by reducing error. They don't take into account class distribution. The model works better in the majority class.
- Handling: Resample, Use advance ML algorithm

Illustration:
Total observation : 1000
Fraud event : 50
Non-fraud event : 950
Event rate : 5%

Purwadhika
Startup and Coding School

Model Evaluation

- **AUC - ROC**

AUC : Area under curve

ROC : Receiver Operating Characteristic

Plot between True Positive Rate (TPR) and False Negative Rate (FPR) for different cut-off points.

Y-axis : True Positive Rate = Sensitivity = Recall

X-axis : False Positive Rate = 1-Specificity

Specificity = TN / (TN+FP)

Sensitivity = TP / (TP+FN)

Purwadnika

Startup and Coding School
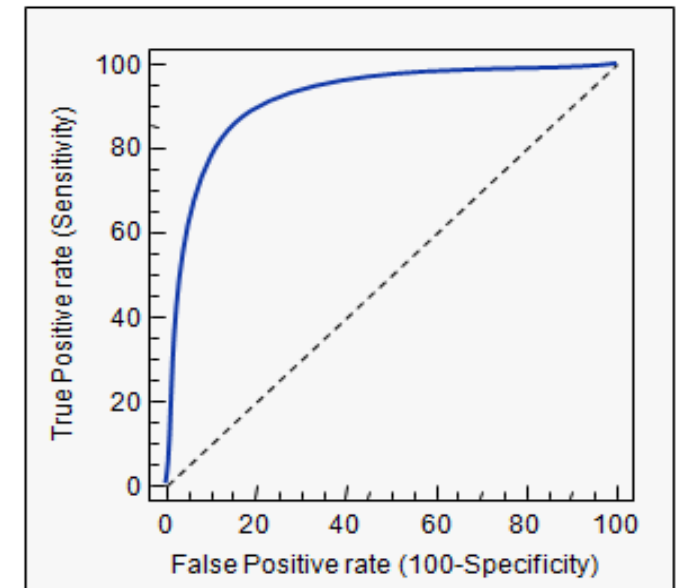
Model Evaluation

How ROC is drawn?

- Plot between True Positive Rate (TPR) and False Negative Rate (FPR) for different cut-off points.
- Suppose now you decrease this threshold level. Then there are chances that what were being earlier classified as zeros will be reclassified as ones

Interpretation guide
- The larger the AUC, the better the model
- ROC curve along diagonal is a random classification
- Model with ROC below diagonal has worse prediction
Than model which makes prediction randomly

Actual class	Model output	Predicted class (cut off 0.5)	Predicted class (cut off 0.45)
1	0.6	1	1
1	0.5	1	1
1	0.5	1	1
1	0.45	0	1
1	0.44	0	0
—	—	—	—
0	0.5	1	1
0	0.5	1	1
0	0.45	0	1
0	0.45	0	1
0	0.4	0	0
—	—	TPR=3/5, FPR=2/5	TPR=4/5,FPR=4/5

Purwadhika
Startup and Coding School

Exercise

- Case : Predict Customer Default of Credit card service from a bank
- Data : Originally from UCI Data Repository (German Credit Data), modified as necessary for training purposes.

Nama Peubah	Deskripsi	Tipe & Satuan	Keterangan
ID	Nomor urut	character ID	
AGE	Umur	Kontinyu (tahun)	
LIMIT_BAL	Batas maksimal kredit	Kontinyu (USD)	
EDUCATION	Tingkat pendidikan	Kategorik	1: S2/S3, 2: Dipl/S1, 3: SMA, 4: Lainnya
MARRIAGE	Status Pernikahan	Kategorik	1: Belum Menikah, 2: Menikah, 3: Lainnya
SEX	Jenis kelamin	Kategorik	1: Pria, 2: Wanita
BILL_AMT1 … 3	Jumlah tagihan	Kontinyu (USD)	
TARGET	Status bayar April 2015	Kategorik	1: Terlambat, 0: Tidak terlambat.

Purwadhika
Startup and Coding School

Session 3

Decision Tree

Purwadhika
Startup and Coding School
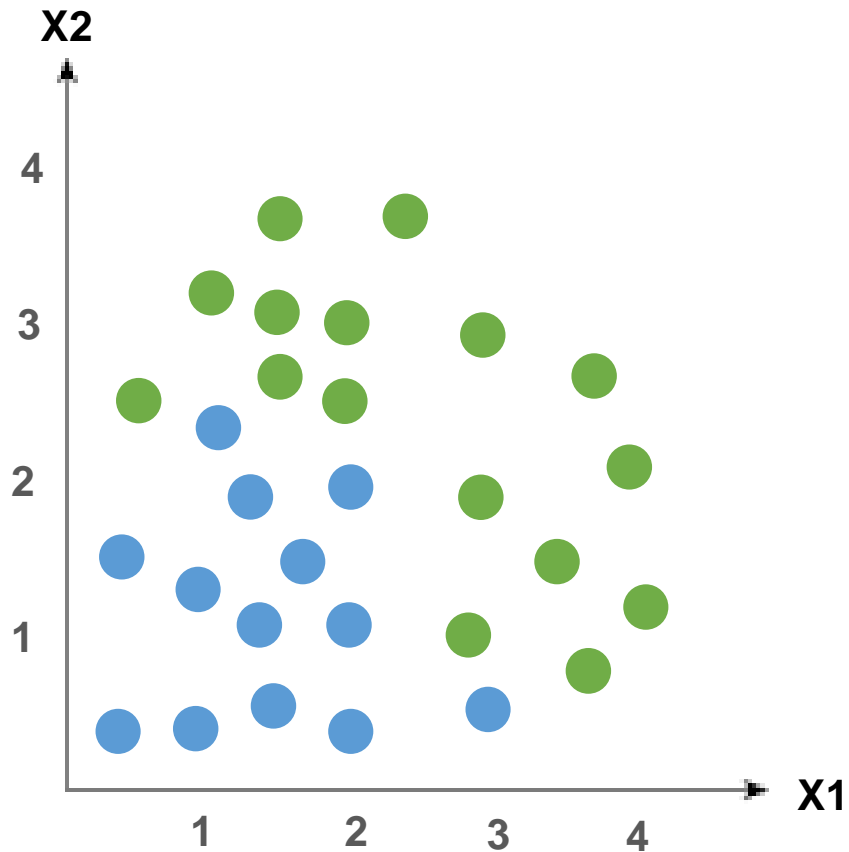
Basic Idea

- 🟢 **16 obs.**
- 🔵 **13 obs.**

- ○ Find the best splitter between 🟢 & 🔵
- ○ Best splitter is the one results most homogenous element in each class

Purwadhika
Startup and Coding School
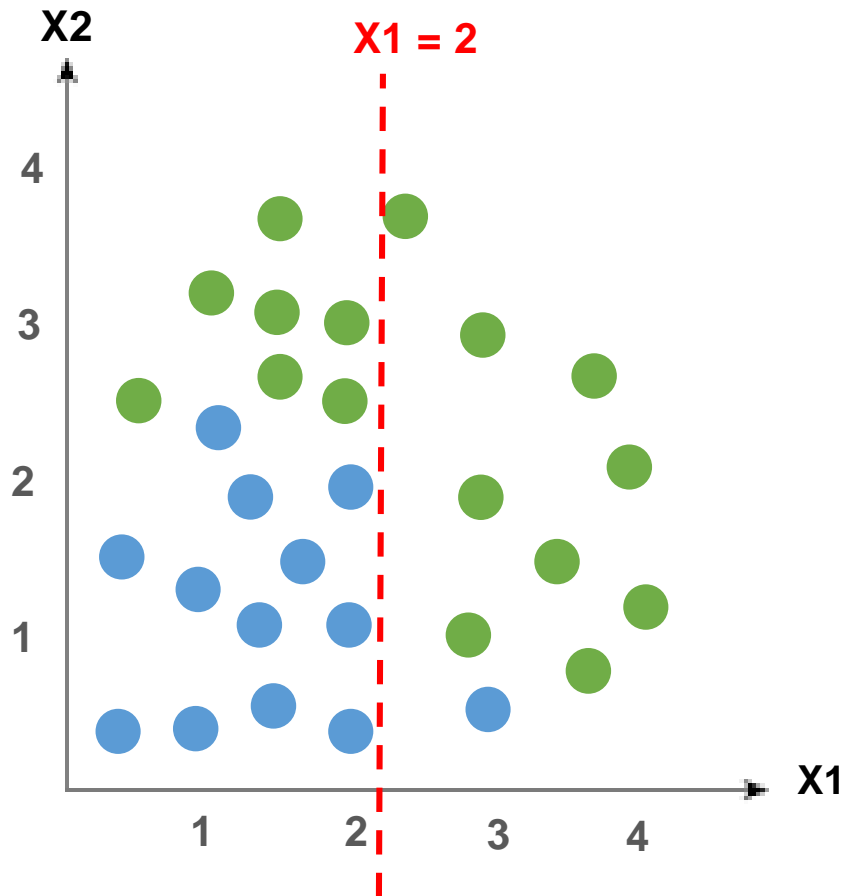
Basic Idea

- o Splitting at X1 = 2
- o Split to 2 classes

Class X1 < 2   🟢 7 obs.
                  🔵 12 obs.

Class X1 > 2   🟢 9 obs.
                  🔵 1 obs.

How good is this split?

Basic Idea

X2

X1 = 3
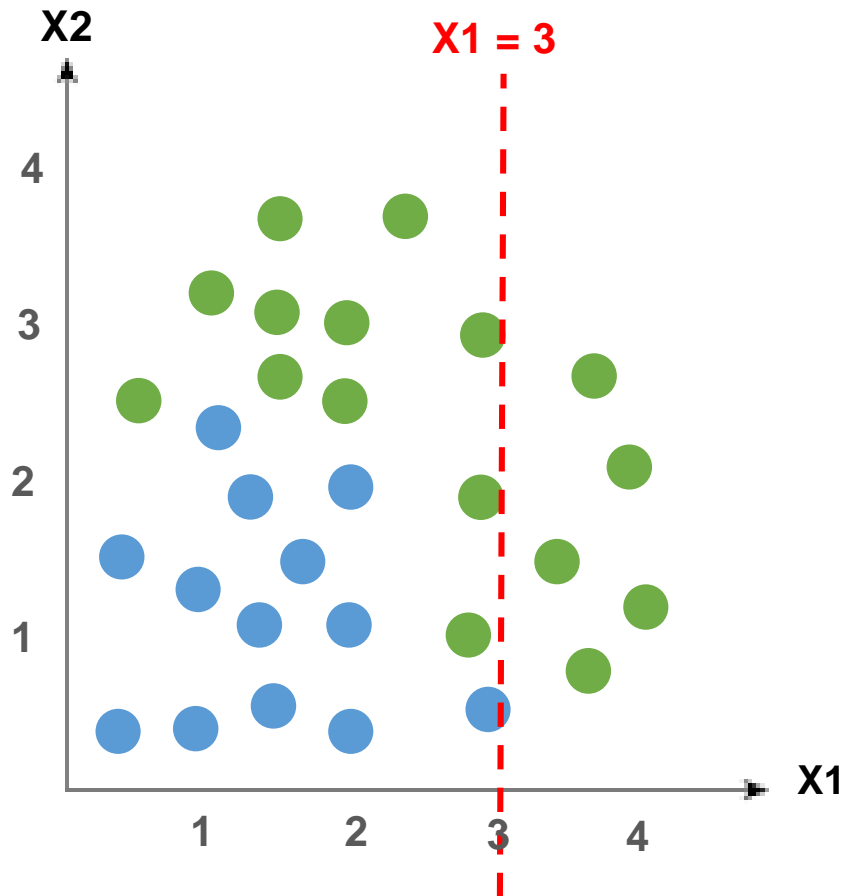
- Splitting at X1 = 3
- Split to 2 classes

Class X1 < 3   ● 11 obs.
                ● 13 obs.

Class X1 > 3   ● 5 obs.
                ● 0 obs.

Is this split better?

Purwadhika
Startup and Coding School

Entropy

Given dataset D, consists of 2 class YES and NO.

Proportion of Yes is **p**, while proportion of No is **(1-p)**

Entropy of the dataset:

$$D(E) = -p.log_2(p) - (1-p).log_2(1-p)$$

Dataset with all-YES or all-NO will have D(E) = 0.

Entropy is measure of heterogeneity)
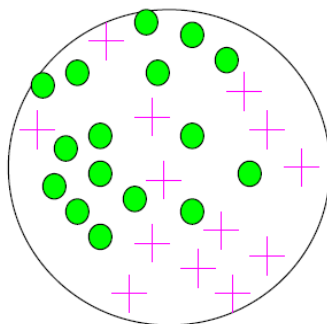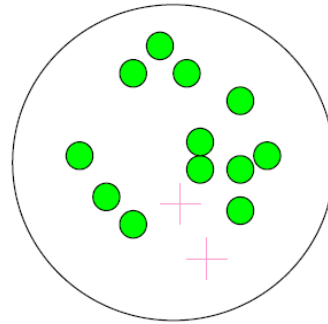
Very impure group   Less impure   Minimum impurity

Entropy

Given dataset D, consists of 2 class YES and NO.

Proportion of Yes is **p**, while proportion of No is **(1-p)**

Entropy of the dataset:

$$D(E) = -p.log_2(p) - (1-p).log_2(1-p)$$
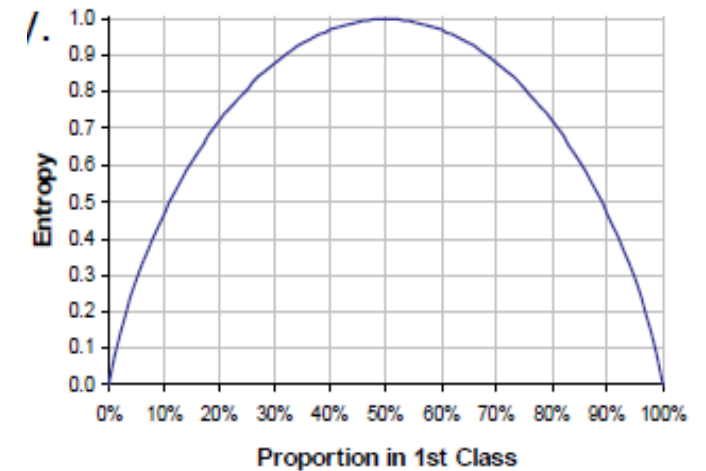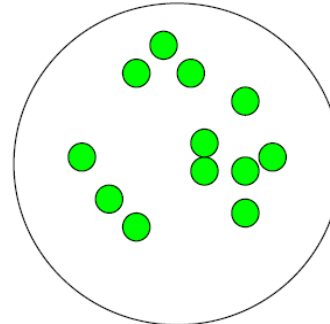
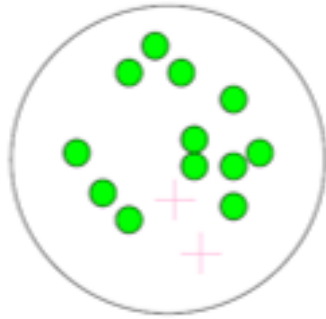Dataset with all-YES or all-NO will have D(E) = 0.
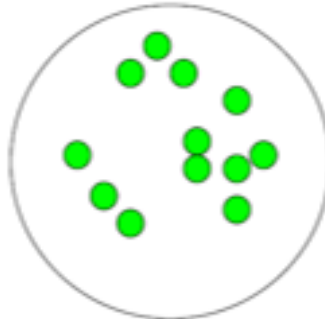
Entropy is measure of heterogeneity)
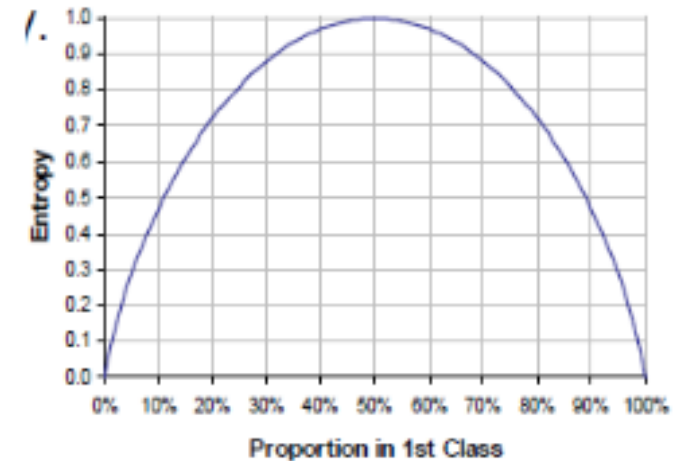
Very impure group   Less impure   Minimum impurity

Information Gain

- Let's say dataset D is split into several groups D1, D2, ... , Dk based on variable V.

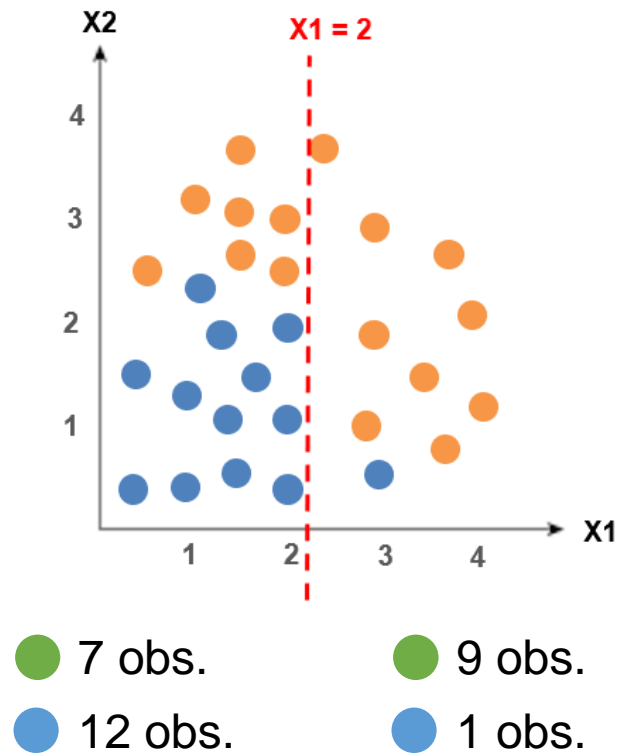- For every Di, Entropy can be calculated as E(Di)

- Information Gain:

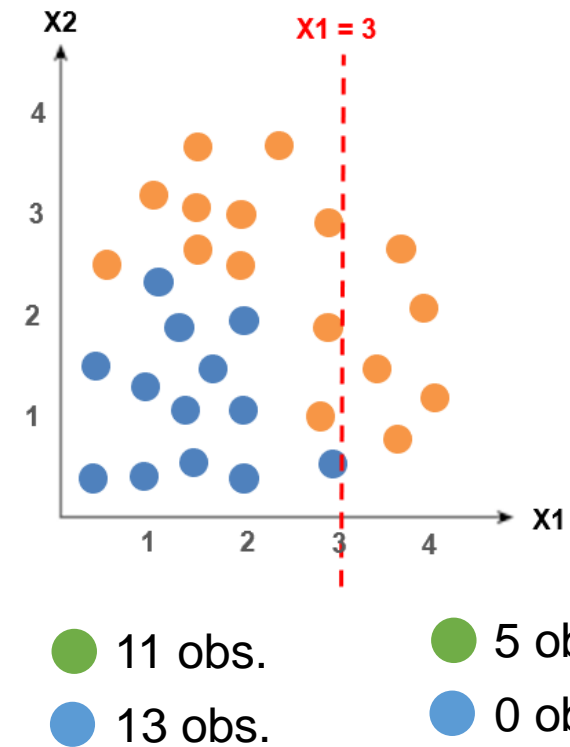$$IG(D,V) = E(D) - \sum_{i=1}^{k} \frac{|D_i|}{|D|} E(D_i)$$

- More homogenous group will have higher Information Gain Value

Purwadhika
Startup and Coding School

Back to Basic Idea

Which one is better?
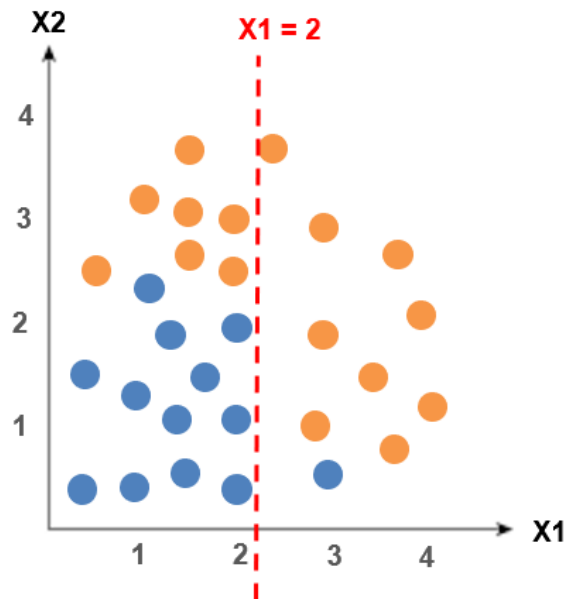
🟢 7 obs.	🟢 9 obs.
🔵 12 obs.	🔵 1 obs.

🟢 11 obs.	🟢 5 obs.
🔵 13 obs.	🔵 0 obs.

Purwadhika
Startup and Coding School
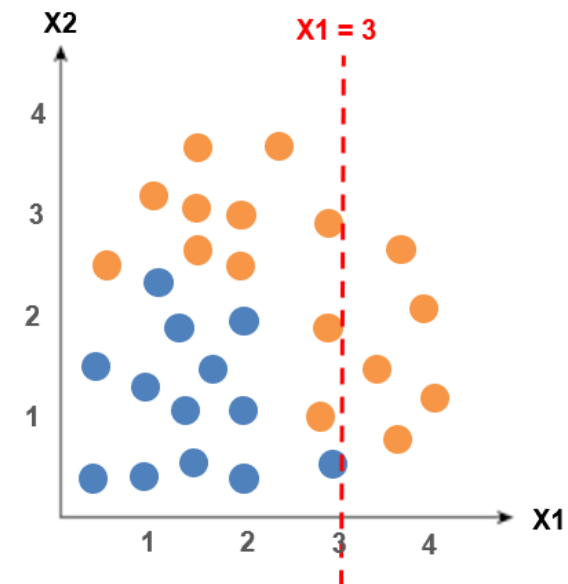
Basic Idea

Basic Idea – Continue Splitting
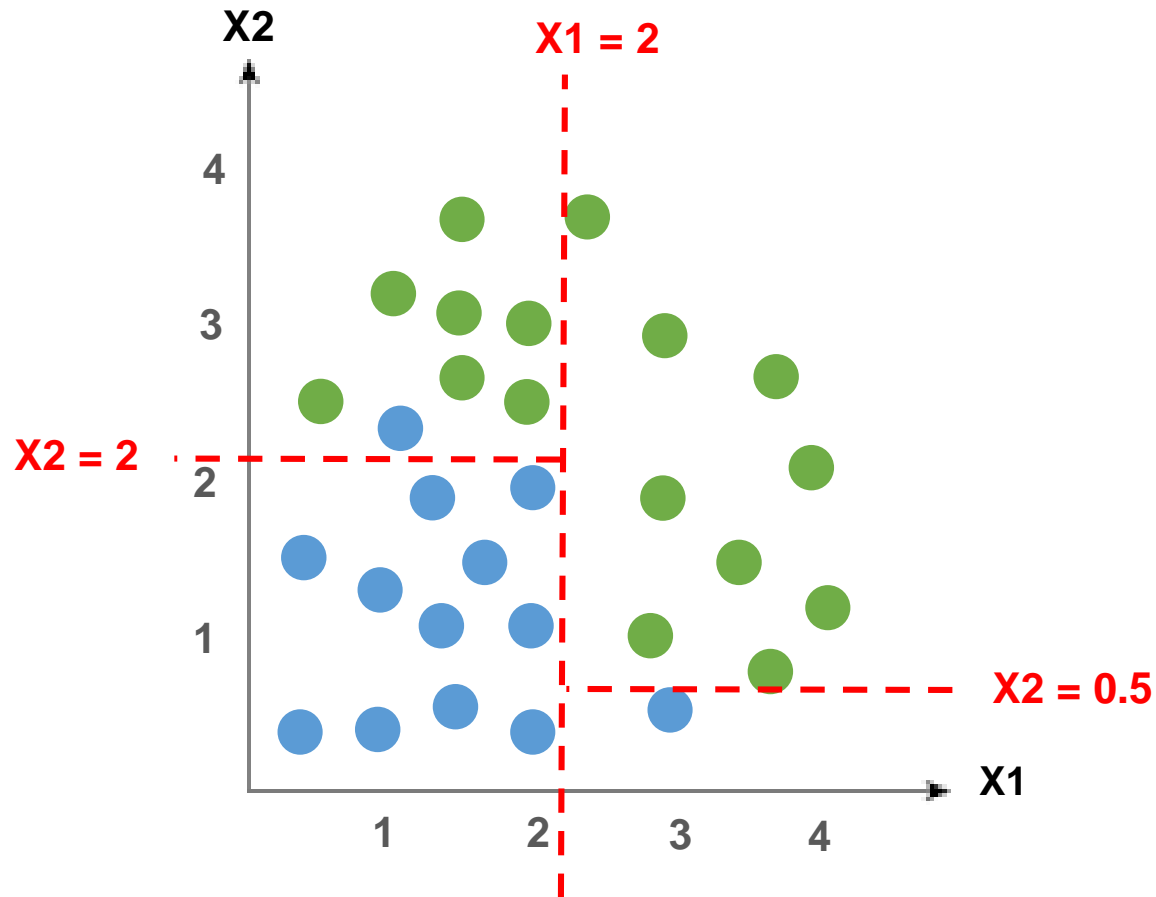
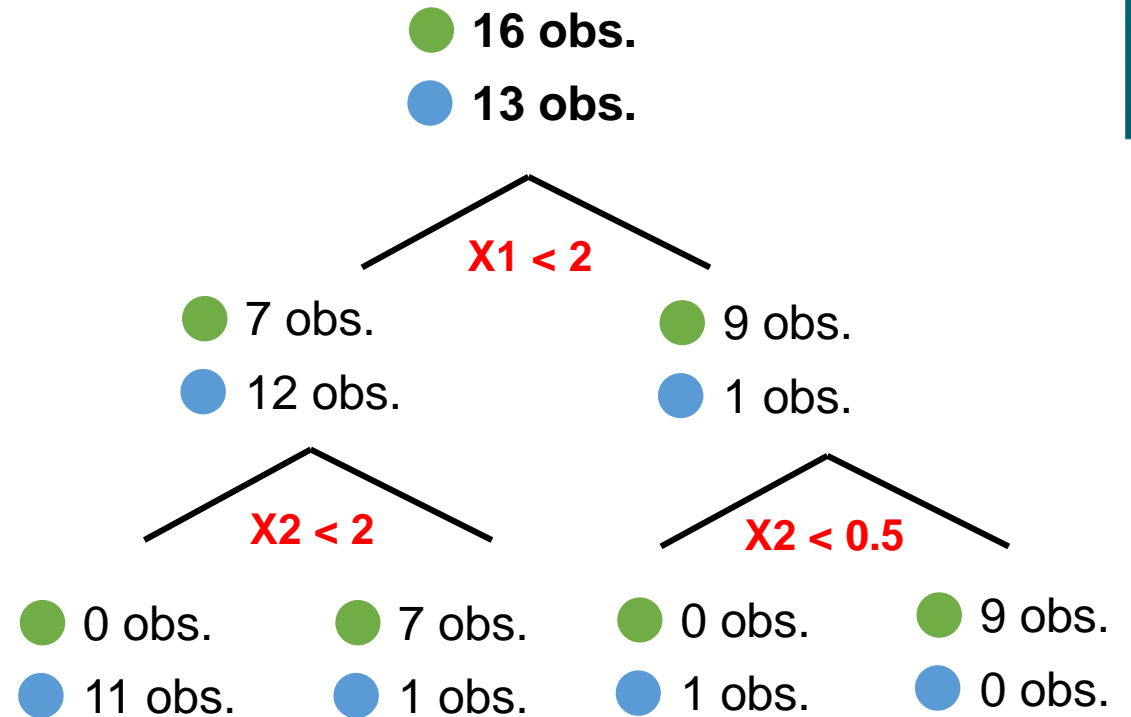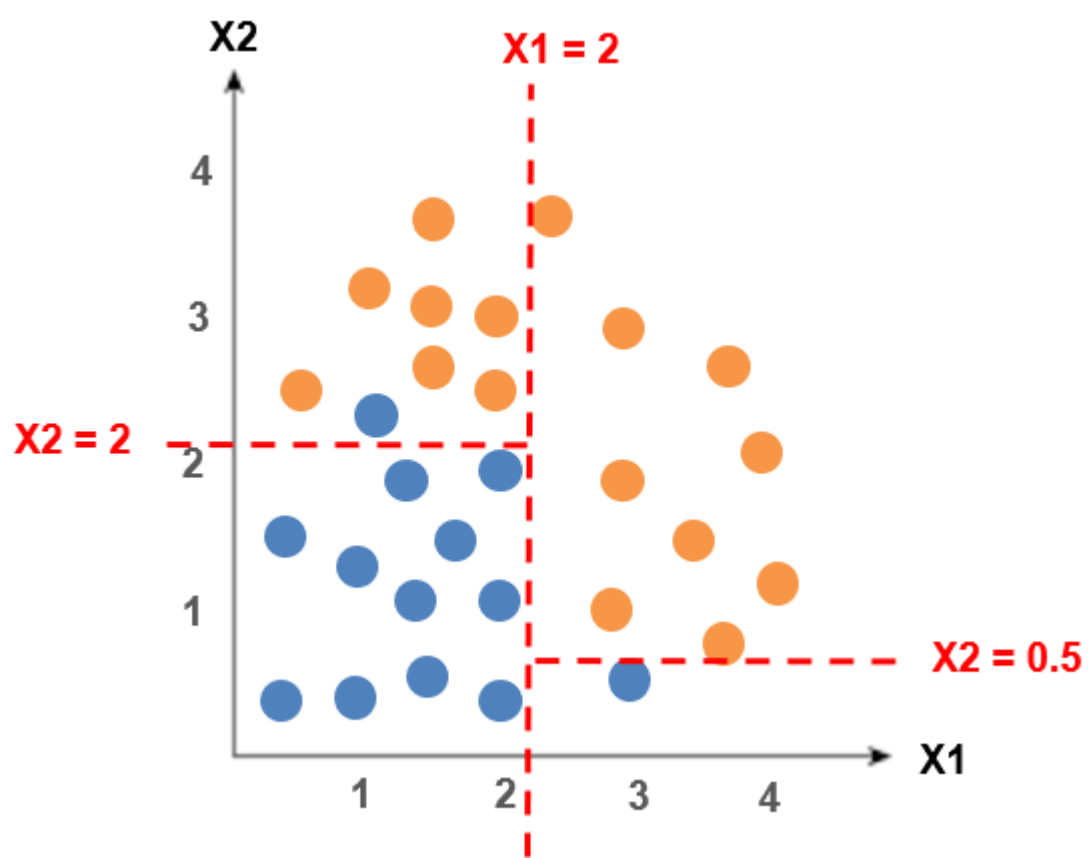
Continue splitting on each group.

At X2 = 2 for class X1 < 2
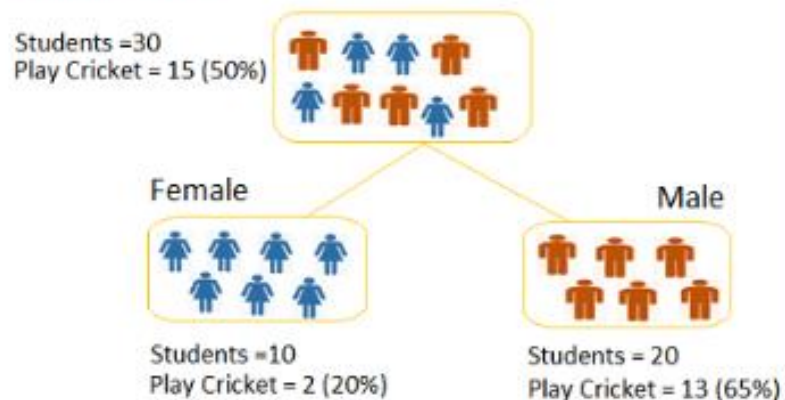
At X2 = 0.5 for class X1 > 2

Basic Idea – Continue Splitting

GINI index

- The higher the GINI value means more homogenous class
- Steps in calculate GINI for a split
  - Calculate Gini for sub-nodes.
    Formula sum of square of probability for success and failure (p^2+q^2).
  - Calculate Gini for split using weighted Gini score of each node of that split.

Split on Gender

Students =30
Play Cricket = 15 (50%)

Female

Male

Students =10
Play Cricket = 2 (20%)

Students = 20
Play Cricket = 13 (65%)

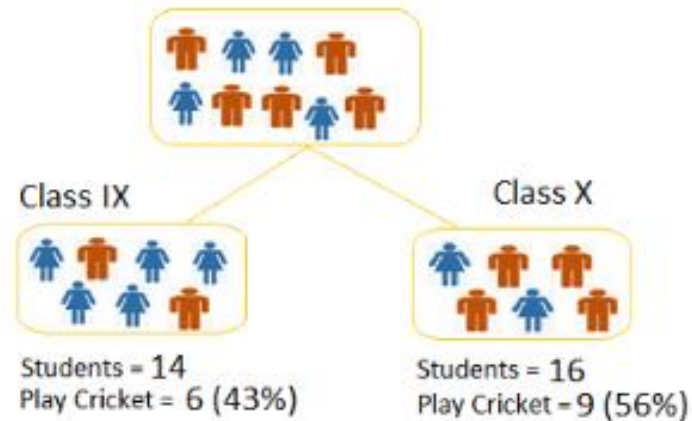Calculate, Gini for sub-node Female
= (0.2)*(0.2)+(0.8)*(0.8)=0.68

Gini for sub-node Male
= (0.65)*(0.65)+(0.35)*(0.35)=0.55

Calculate weighted Gini for Split Gender
= (10/30)*0.68+(20/30)*0.55 = **0.59**

Purwadhika
Startup and Coding School

GINI index

Split on Class

Class IX
Students = 14
Play Cricket = 6 (43%)

Class X
Students = 16
Play Cricket = 9 (56%)
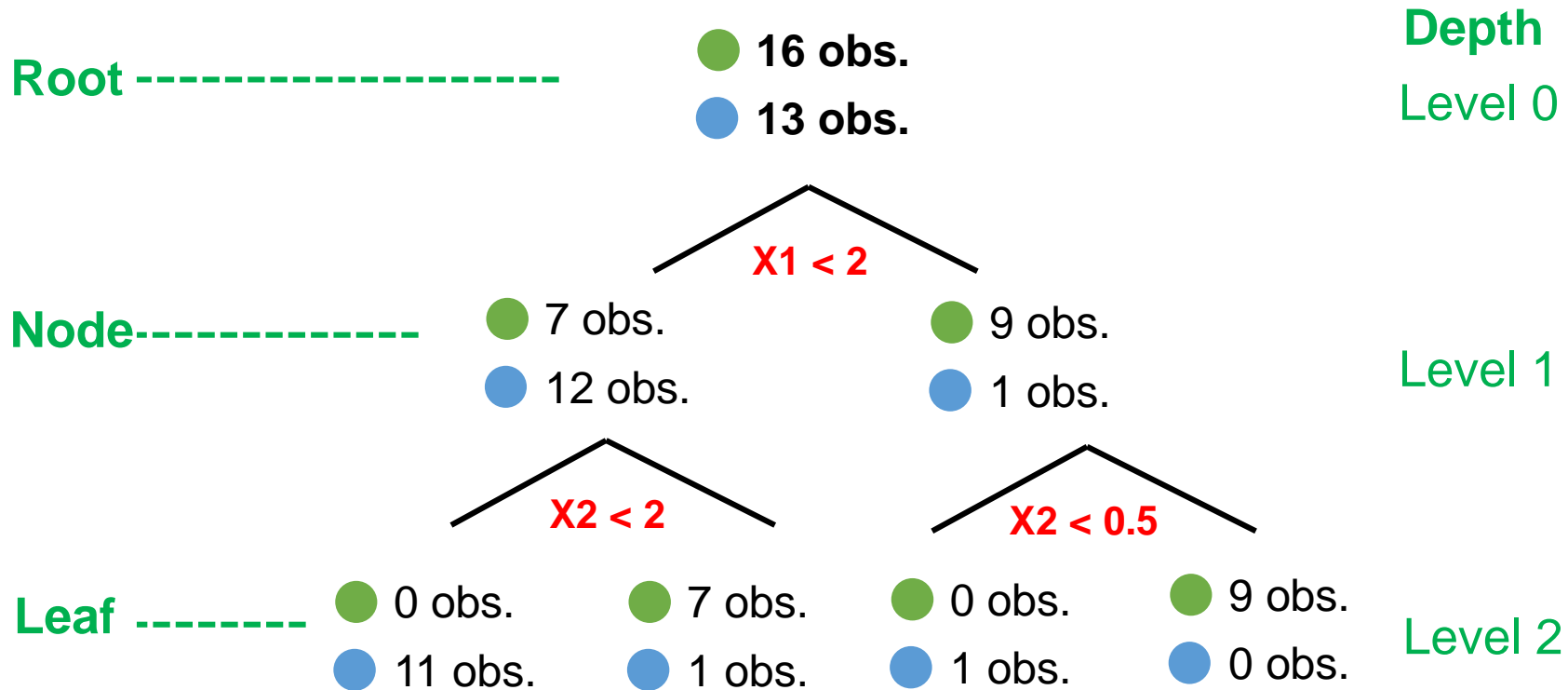
Gini for sub-node Class IX
= (0.43)*(0.43)+(0.57)*(0.57)=0.51

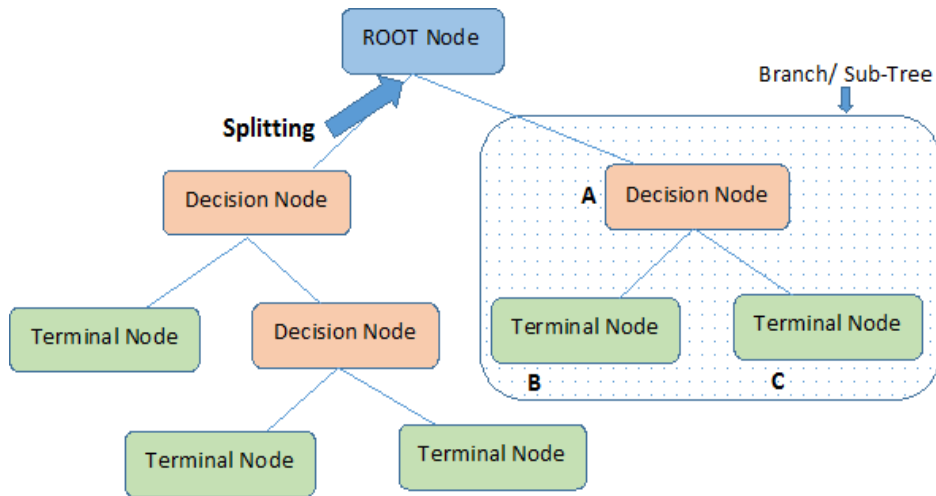Gini for sub-node Class X
= (0.56)*(0.56)+(0.44)*(0.44)=0.51

Calculate weighted Gini for Split Class
= (14/30)*0.51+(16/30)*0.51 = **0.51**

**Gini score for *Split on Gender* is higher
than *Split on Class,* hence, the node split will
take place on Gender.**

Purwadhika
Startup and Coding School

Terminologies

Terminologies

Root Node: Represent entire population / sample

Splitting: Process of dividing a node into two or more sub-nodes.

Decision Node: Sub-node that splits into further sub-nodes

Leaf/Terminal Node: Nodes that do not split

Pruning: Remove sub-nodes of a decision node. It is the opposite of splitting.

Branch / Sub-Tree: A sub section of entire tree is called branch or sub-tree.

Parent and Child Node: A node, which is divided into sub-nodes is called parent node of sub-nodes where as sub-nodes are the child of parent node

Basic Algorithm

Perform 3 steps for every single Node and its splitting result

- **Step-1**
  Find best splitter on each variable

- **Step-2**
  Select best variable for splitting

- **Step-3**
  Perform splitting based on result on Step-2.
  Check if the splitting should stop.

Stop-splitting Condition

Splitting will stop if any of below conditions met

- Node contains only 1 class of response variable
- Number of observation in a node before splitting is less than pre-defined number
- Number of observation in a node after splitting is less than pre-defined number
- Tree depth has reach its maximum

There are parameters in the software to control the Tree Size
- Minimum sample of node split
- Minimum sample of terminal
- Maximum depth of tree
- Maximum number of terminal node

Purwadhika
Startup and Coding School

Illustration

Adv-Disadvantages

Advantages
- Easy to understand
- Useful in data exploration. Information of importance variables, variables which relates each other.
- Data type is not a constraint (works for numerical and categoric too)
- Non parametric, have no assumption about distribution

Disadvantages
- Over fitting
- Loses information of continuous numeric variable when it categorized into different categories

Purwadhika
Startup and Coding School

Adv-Disadvantages

Advantages
- Easy to understand
- Useful in data exploration. Information of importance variables, variables which relates each other.
- Data type is not a constraint (works for numerical and categoric too)
- Non parametric, have no assumption about distribution

Disadvantages
- Over fitting
- Loses information of continuous numeric variable when it categorized into different categories

Purwadhika
Startup and Coding School

Exercise

- Using same use case and dataset on Logistic Regression.
- Perform prediction using Decision Tree
- Compare the result

Nama Peubah	Deskripsi	Tipe & Satuan	Keterangan
ID	Nomor urut	character ID	
AGE	Umur	Kontinyu (tahun)	
LIMIT_BAL	Batas maksimal kredit	Kontinyu (USD)	
EDUCATION	Tingkat pendidikan	Kategorik	1: S2/S3, 2: Dipl/S1, 3: SMA, 4: Lainnya
MARRIAGE	Status Pernikahan	Kategorik	1: Belum Menikah, 2: Menikah, 3: Lainnya
SEX	Jenis kelamin	Kategorik	1: Pria, 2: Wanita
PAY_1 … 3	Lama terlambat bayar	Kategorik	0: Tepat waktu, 1: Terlambat 1 bulan, dst
BILL _AMT1 … 3	Jumlah tagihan	Kontinyu (USD)	
PAY_AMT1 …3	Jumlah dibayar	Kontinyu (USD)	
TARGET	Status bayar April 2015	Kategorik	1: Terlambat, 0: Tidak terlambat.

Purwadhika
Startup and Coding School

Session 4

✓ Assignment
✓ Recap of the day

Purwadhika
Startup and Coding School

It's a wrap

- **CRISP-DM Concept**
- **Supervised VS Unsupervised Learning**
- **Machine Learning Application**
- **Regression Logistic Concept**
- **Regression Logistic Interpretation**
- **Decision Tree Splitting Algorithm**
- **Decision Tree Stop-splitting condition**
- **Entropy and Information Gain**
- **Model Evaluation**

Purwadhika
Startup and Coding School