

Modul 3

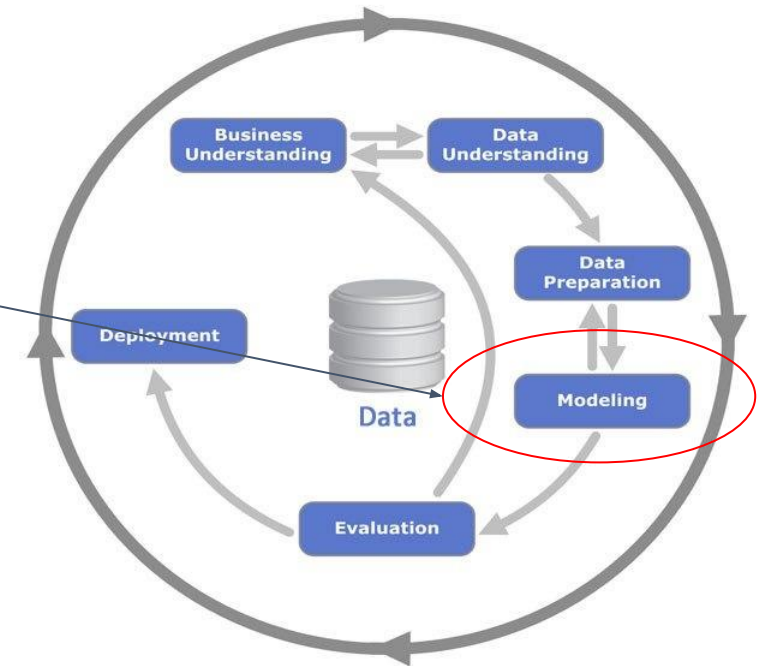
Supervised Part 2

Data Science Program

Outline

What is Classification ?
Methods in Classification
Logistics Regression
KNN
Decision Tree

CRISP-DM
Process
Diagram



Source: Kenneth Jensen

What is Classification ?

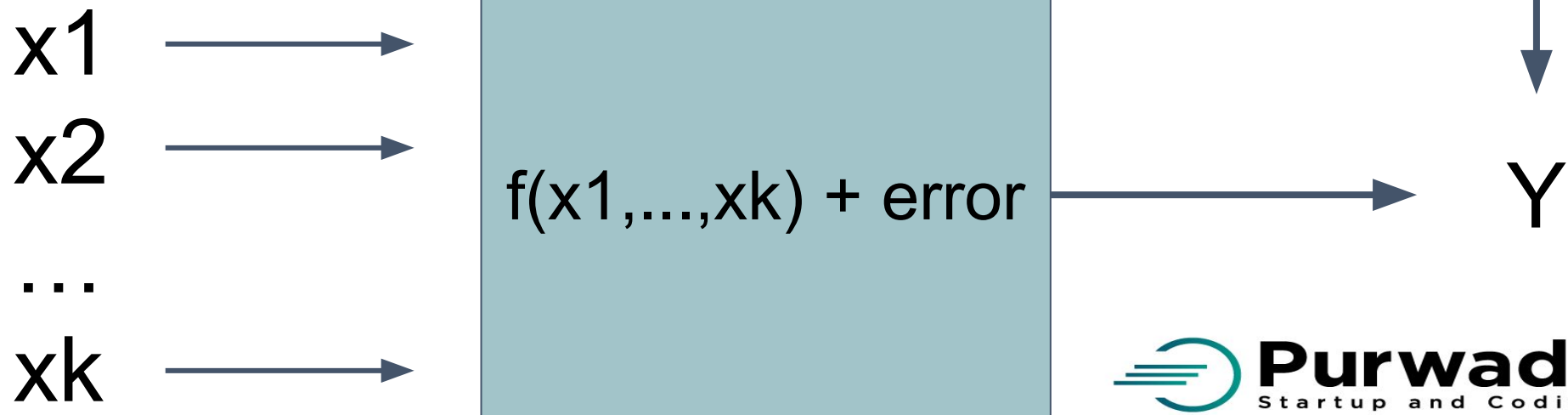
What Is Classification ?

Response variable = Model function + random error

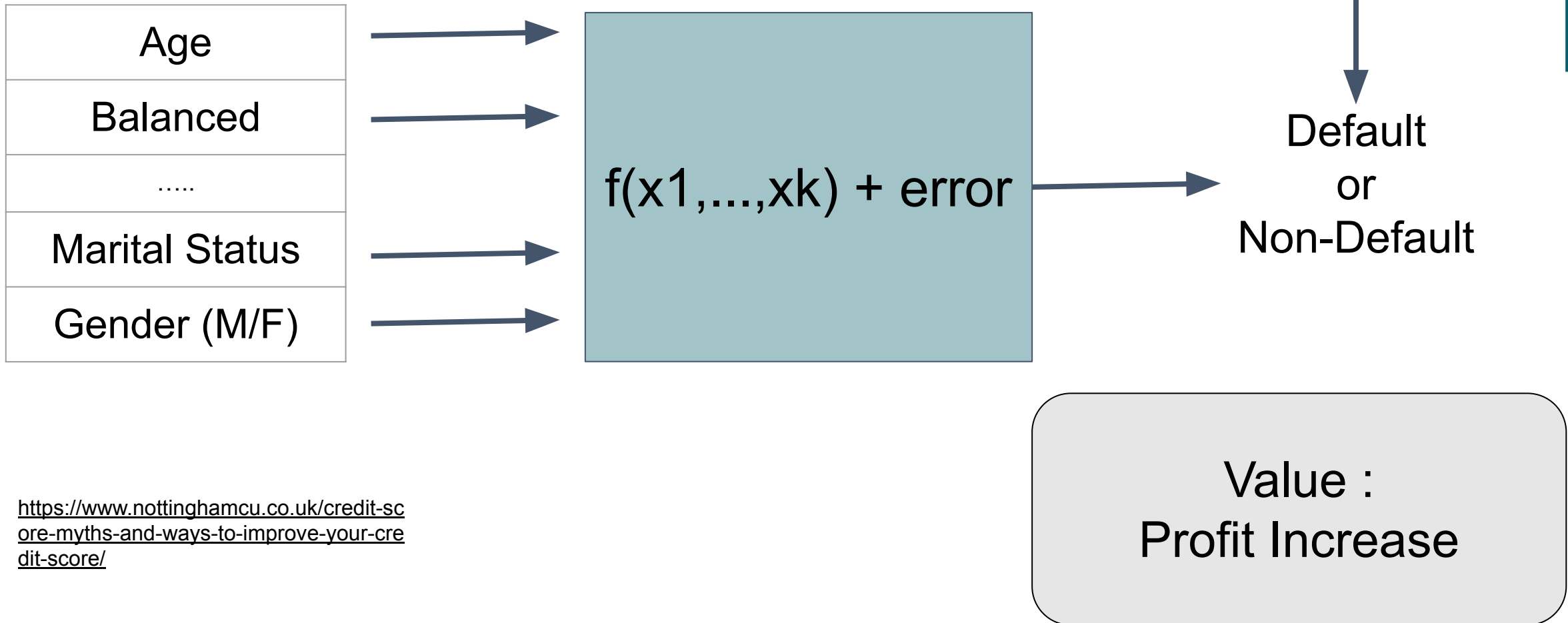
$$Y = f(x_1, x_2, \dots, x_k) + e$$

Y categorical - 2 categories (binary classification)

Y categorical - more than 2 categories (multiclass classification)

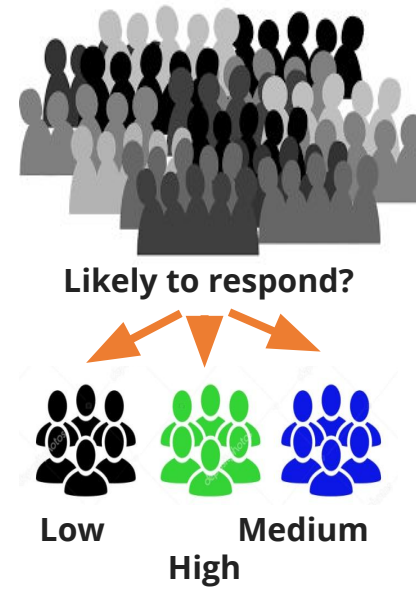


Credit Scoring (Binary)





Churn Analysis



Propensity Analysis



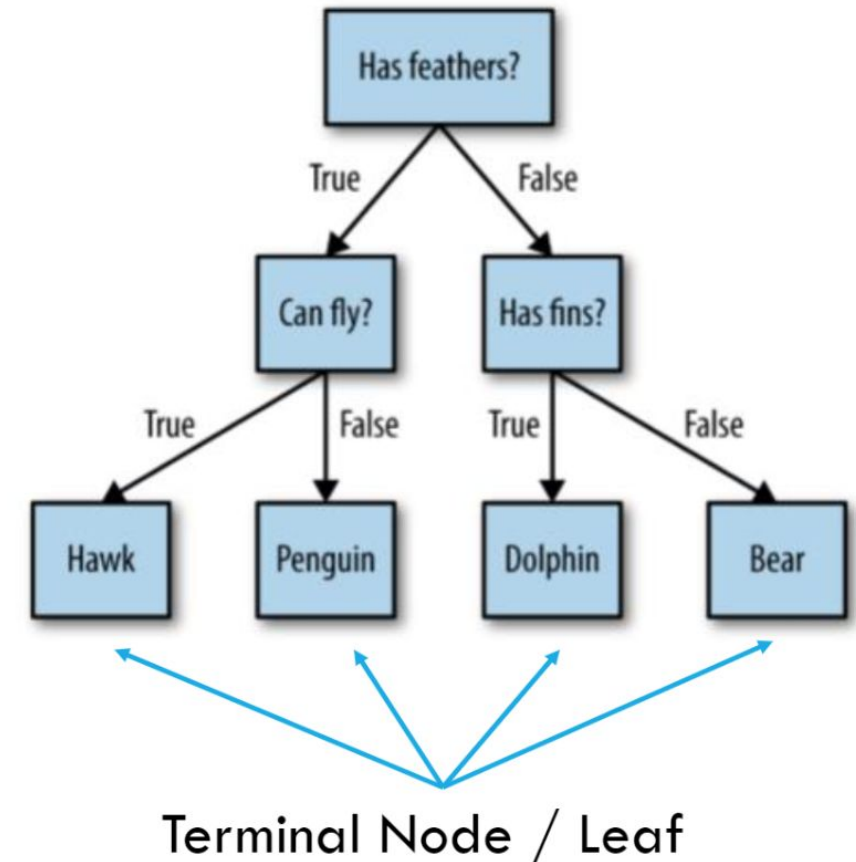
Human Resources
“The Rising Star”

Some Method Usually Used In Classification

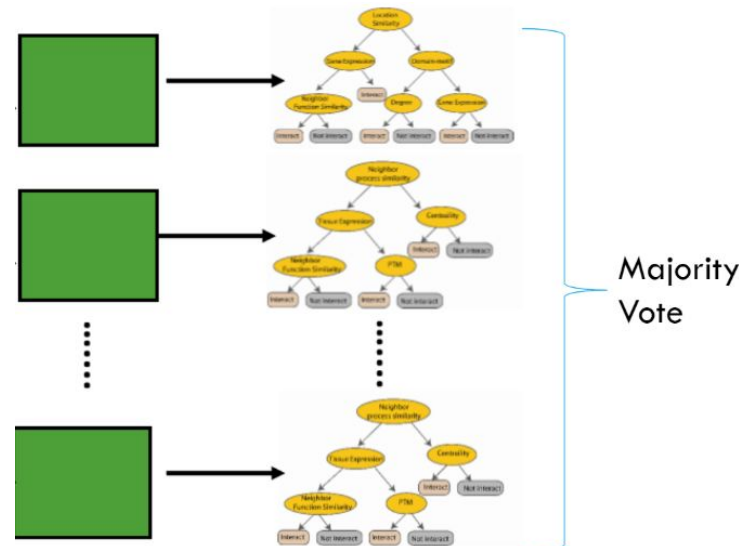
Logistic Regression:

$$P(Y = 1) = \frac{\exp(\beta_0 + \beta_1 x_1 + \dots + \beta_k x_k)}{1 + \exp(\beta_0 + \beta_1 x_1 + \dots + \beta_k x_k)}$$

Decision Tree Classifier:



Ensemble Method:



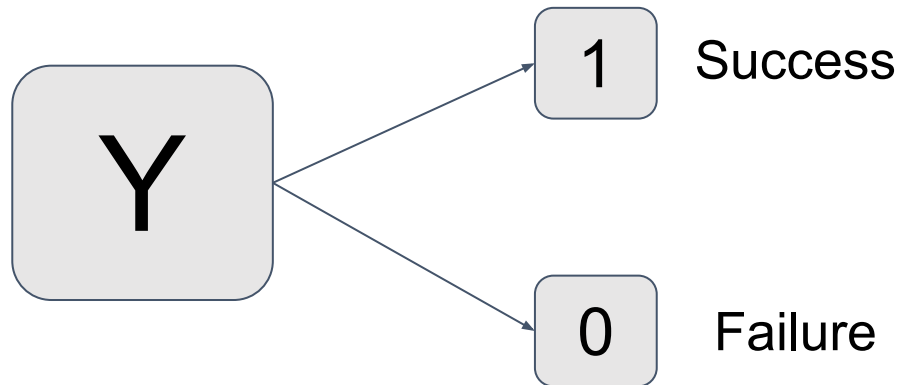
Note:
Other models:
Discriminant Analysis,
K-Nearest Neighbour
(KNN),
Support Vector
Machine (SVM),
Ensemble – Bagging,
Random Forest,
Boosting, etc

Binary Logistics Regression

Logistic Regression

1. Binary Logistic Regression, binary label
2. Multinomial Logistic Regression, multinomial label
3. Ordinal Logistic Regression, ordinal label

What is Binary Logistic Regression ?



Remember that
binary logistic
regression model
the success
rate/probability

Has more interest in success event

Case	1	0
Credit scoring	Bad	Good
Churn Analysis	Turn Over	Stay
Propensity	Buy	Not Buy

What is Binary Logistic Regression ?

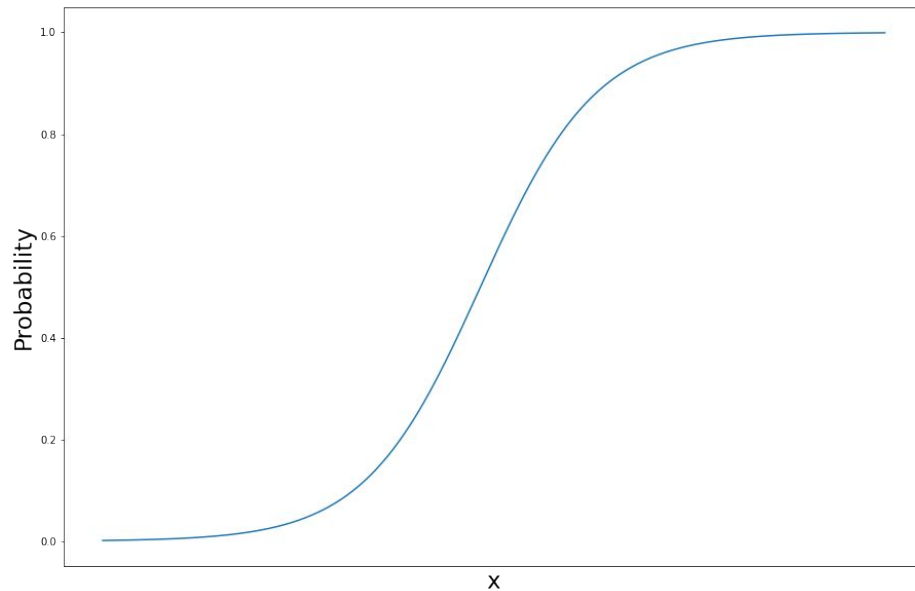
$$P(Y = 1) = \frac{\exp(\beta_0 + \beta_1 x_1 + \dots + \beta_k x_k)}{1 + \exp(\beta_0 + \beta_1 x_1 + \dots + \beta_k x_k)}$$

* $\exp(B_0 + B_1 x_1 + \dots + B_k x_k)$ is approximately equal to $2.71^{(B_0 + B_1 x_1 + \dots + B_k x_k)}$

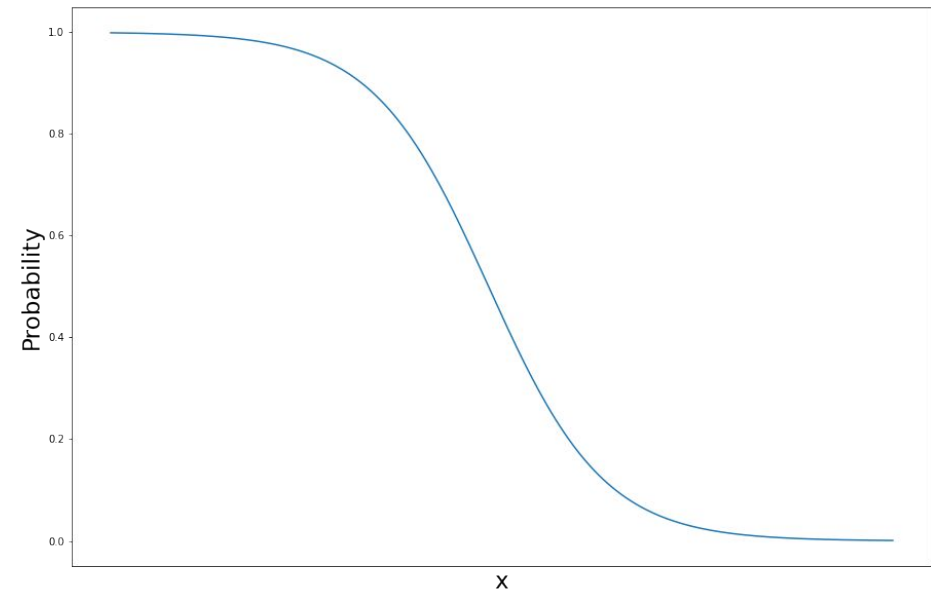
- Probability to success $P(Y = 1)$ and Probability to fail $P(Y = 0) = 1 - P(Y = 1)$
- Another notation Success (+) Failed (-)
- odds = $\exp(B_0 + B_1 x_1 + \dots + B_k x_k)$, ratio between probability to success and probability to fail
- $B_0 B_1 B_2 \dots B_k$, Regression Parameter
- $x_1 x_2 \dots x_k$, Features/Independent Variable

Sigmoid Curve

$b > 0$, success rate increase when
X increase



$b < 0$, success rate decrease when
X increase



$$P(Y = 1) = \text{odd} / (1 + \text{odd}),$$

with

- $0 < P(Y = 1) < 1$
- Y = dependent variable, succes (Y = 1) failure (Y = 0)
- $\text{odd} = \exp(a + bx)$
- x = independent variable

Why do we need to use Logistic Regression ?

- Instead of linear regression, logistic regression is more suitable when the response variable are categorical
 - Linear regression was designed for numerical variable
 - Linear regression can gives meaningless out-of-range prediction
 - Linear regression gives wrong p-value when Y categorical due to violation in normality assumption and equal variance assumption (homoscedasticity)
- Logistic Regression has high interpretability, remember that purpose of the modeling is not always about prediction.

Example : Loan repayment comparison by gender

Features (x)
Gender (M/F)

$x = 1$ (Male)

$x = 0$ (Female)



$$P(Y=1) = \frac{\exp(-2.1972 + 1.086x)}{1 + \exp(-2.1972 + 1.086x)}$$



Default ($Y=1$) or Non-Default ($Y=0$)

Analysis

MALE:

$$P(Y = 1 \mid \text{Male}) = \exp(-2.1972 + 1.086 (1)) / (1 + \exp(-2.1972 + 1.086 (1))) = 0.25$$

$$P(Y = 0 \mid \text{Male}) = 1 - P(Y = 1 \mid \text{Male}) = 1 - 0.25 = 0.75$$

$$\text{Odd}(\text{Male}) = 0.25/0.75 = 1/3, \text{ men is 3 times less likely to default}$$

FEMALE:

$$P(Y = 1 \mid \text{Female}) = \exp(-2.1972 + 1.086 (0)) / (1 + \exp(-2.1972 + 1.086 (0))) = 0.1$$

$$P(Y = 0 \mid \text{Female}) = 1 - P(Y = 1 \mid \text{Female}) = 1 - 0.1 = 0.9$$

$$\text{Odd}(\text{Female}) = 0.1/0.9 = 1/9, \text{ women is 9 times less likely to default}$$

Odds-Ratio (OR)

- Odds-ratio is used to interpret logistic regression
- Odds-ratios indicate how likely a successful event is to occur in one condition compared to other conditions

Example:

Odds-Ratio = $\text{Odd}(\text{Male}) / \text{Odd}(\text{Female}) = 3$, Men have three times greater tendency / risk to default than women

How To Interpret B_i

In binary logistic regression, If a unit observation with $X_i = c$ and another unit observation with $X_i = d$. Odd ratio between those unit observations are :

$$OR = \exp(B_i (c-d))$$

for $c > d$:

- if $B_i > 0$, then $OR > 1$
- if $B_i = 0$, then $OR = 1$
- if $B_i < 0$, then $OR < 1$

How To Interpret B_i

- when interpret B_i , it is assumed that no changes in another variable
- Binary logistic regression requires no multicollinearity between independent variables
- It is not recommended to interpret outside range of interval

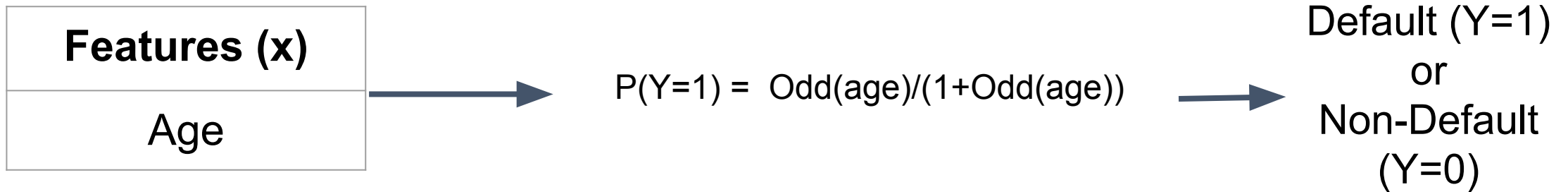
if $OR > 1$:

- Success rate increase when X_i increase
- Unit observations which have $X_i = c$ have OR times greater tendency to achieve success event than unit observations which have $X_i = d$.
- in general, unit observation which have $(c-d)$ higher different value in X_i have about OR times greater tendency to achieve success event

if $OR < 1$:

- Success rate decrease when X_i increase
- Unit observations which have $X_i = d$ have $1/OR$ times greater tendency to achieve success event than unit observations which have $X_i = c$.
- in general, unit observation which have $(c-d)$ lower different value in X_i have about $1/OR$ times greater tendency to achieve success event

Example : Loan repayment comparison by Age



age (20 - 56 year)

$$\text{Odd}(\text{age}) = \exp(0.3669 - 0.0411\text{age})$$

Analysis

$P(Y=1) = \text{Odd}(\text{age}) / (1 + \text{Odd}(\text{age}))$, with
 $\text{Odd}(\text{age}) = \exp(0.3669 - 0.0411\text{age})$

Age	Odd(Age)	Prob(Age)
20	0.6343	0.3881
21	0.6088	0.3784
22	0.5843	0.3688
...	...	
25	0.5165	0.3406
...	...	
30	0.4205	0.2960
...
56	0.1444	0.1262

$\text{OR} = \exp(B - \text{age}(c - d))$

$c = 38$

$d = 20$

$\text{OR} = \exp(-0.0411 \cdot 18) = 0.4772 < 1$

$\text{odd}(20)/\text{odd}(38) = 2.095$

$\text{odd}(21)/\text{odd}(39) = 2.095$

$\text{odd}(22)/\text{odd}(40) = 2.095$

...

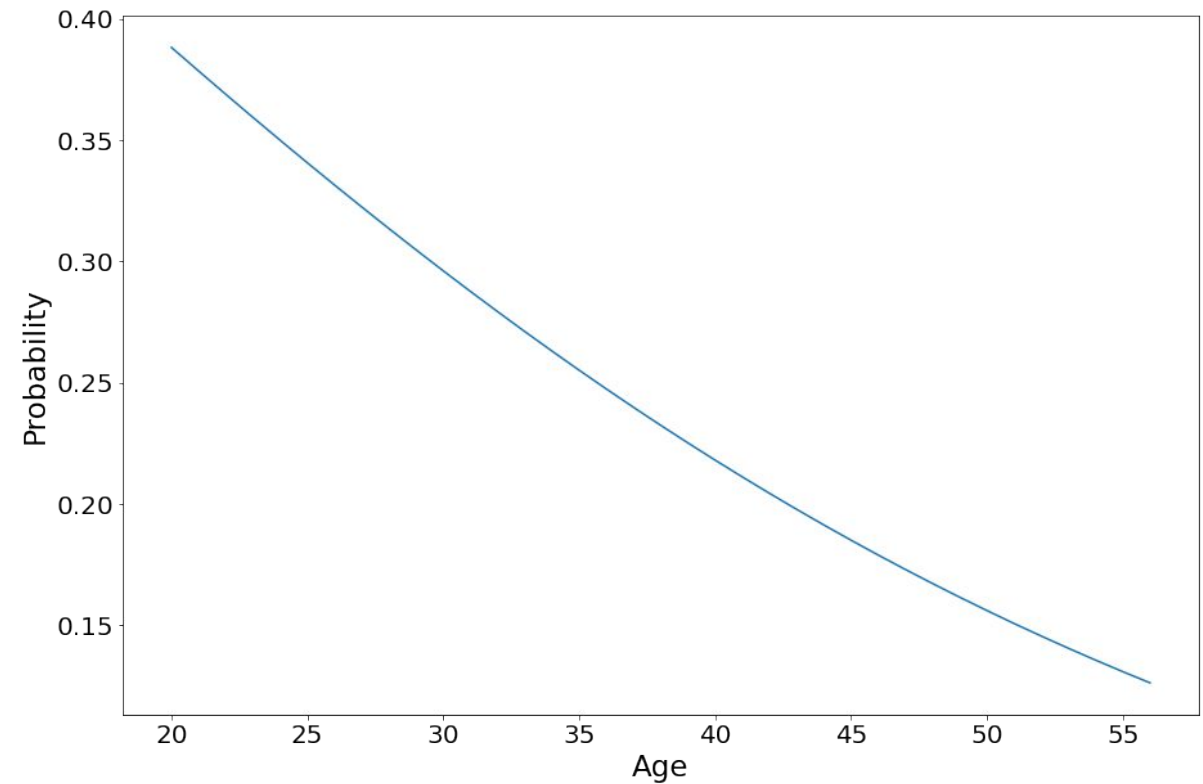
$\text{odd}(38)/\text{odd}(56) = 2.095$

Interpretation, $\text{OR} < 1$:

- Probability to default decrease when age increase
- people with age 20 have about 2.095 times greater tendency/risk to default than people with age 38.
- In General, younger people with difference in age 18 years have about 2.095 times greater tendency/risk to default

Analysis

Age	Odd(Age)	Prob(Age)
20	0.6343	0.3881
21	0.6088	0.3784
22	0.5843	0.3688
...	...	
25	0.5165	0.3406
...	...	
30	0.4205	0.2960
...
56	0.1444	0.1262



Log-Likelihood Ratio Test (LLR-Test)

- LLR-Test in Logistic Regression is analogous to F-Test in Linear Regression
- LLR-Test check for overall significance of multiple regression model.
- LLR-Test checks if there is a statistically significant relationship between Y (dependent variable) and any of the independent variables

Hypothesis:

$$H_0 : \beta_1 = \beta_2 = \dots = \beta_k = 0$$

H_A : Not all β values are zero

Test Statistics : Log Likelihood Ratio

Rejection Criteria : $P\text{-value} \leq \alpha$ (two-sided)

Wald Test

- Wald-Test in Logistic Regression is analogous to T-Test in Linear Regression
- Wald-test checks if there is a statistically significant relationship between Y (dependent variable) and each of the independent variables

Hypothesis:

$H_0 : \mathbf{B_i} = 0$

$H_a : \mathbf{B_i} \neq 0$ (two sided)

$\mathbf{B_i} > 0$ or $\mathbf{B_i} < 0$ (one sided)

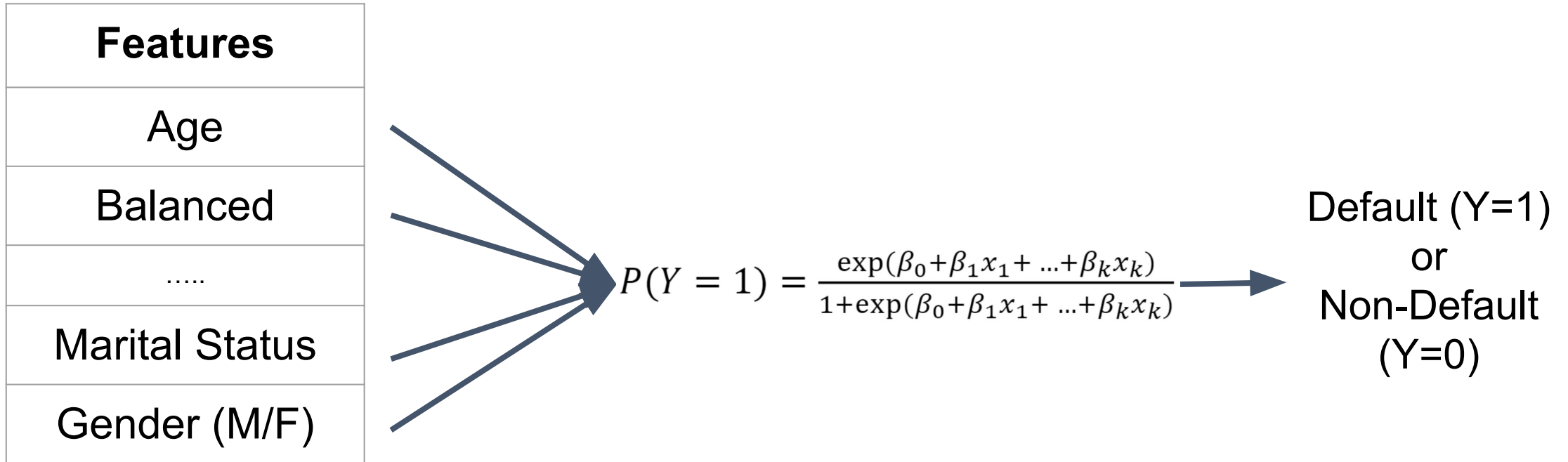
Rejection Criteria:

$P\text{-value} \leq \alpha$ (two-sided)

$P\text{-value}/2 \leq \alpha$ (one-sided)

Test Statistics : Wald-Statistics

Example : Credit Scoring



<https://www.nottinghamcu.co.uk/credit-score-myths-and-ways-to-improve-your-credit-score/>

Problem

How to predict **default risk of the new applicant** so we can **allocate loan efficiently** and **increase profit** from loan ?

Data

- What is being predicted ? default risk of the new applicant
- What is needed in prediction ? Demographical, Transaction behaviour, income, ect

ML Objective

Maximize (profit - potential revenue lost)

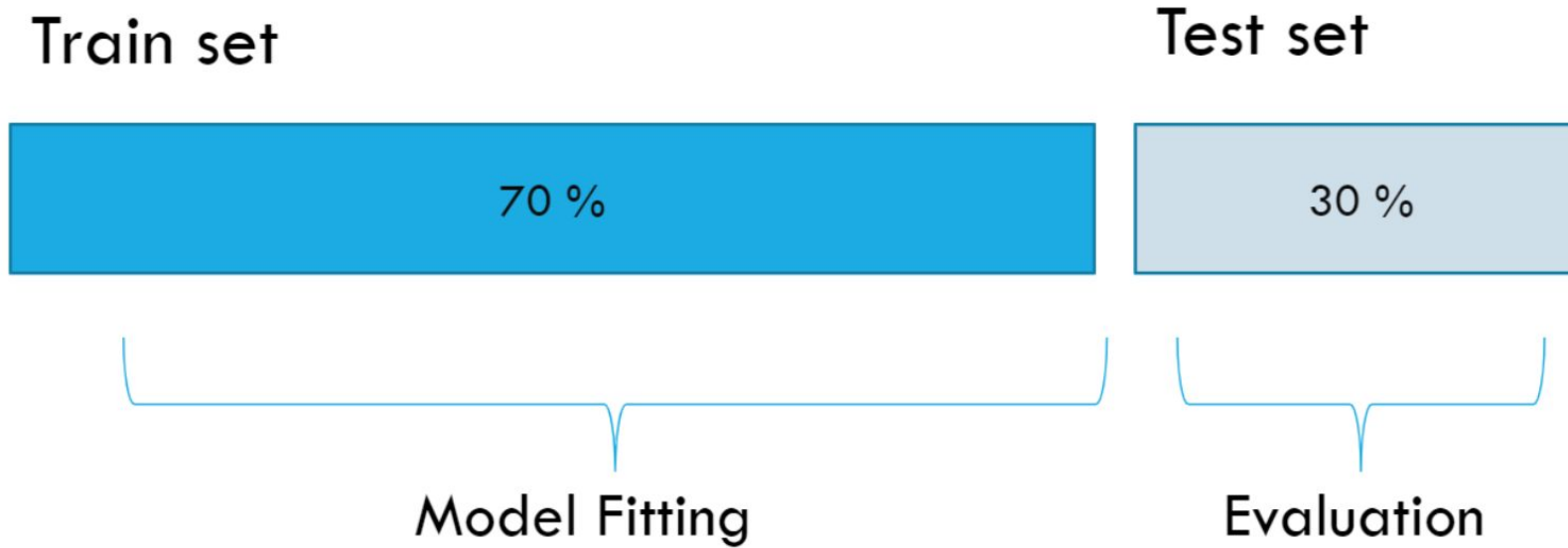
Action

Do not allocate loan to a customer when the risk is too high, higher than 50%

Value

Profit Increase

Validation Method



Measuring Performance of Classification Method

No	Prediction	Actual
1	1	1
2	1	0
3	0	1
..
499	0	0
500	0	1

Prediction	Actual	
	0	1
0	120	23
1	27	330

Accuracy Of Prediction = $(120+330)/500 \times 100\%$
= 90.0%

Our model will correctly predict 9 of 10 People

Python Exercise : Logistic Regression

Analyze data bankloan.csv

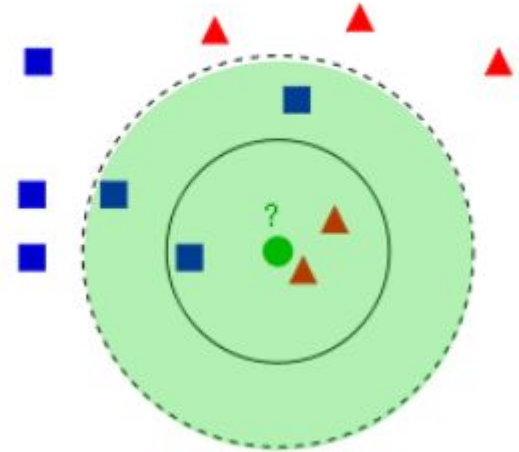
- build a logistics regression model
 - target : default
 - features : employ, debtinc, creddebt, othdebt
- Interpret The Result
- Validate the model using accuracy in 20% testing data

K-Nearest Neighbour

What is K-Nearest Neighbour ?



"Predict based on majority class of the top k similar observation or also called nearest neighbour"



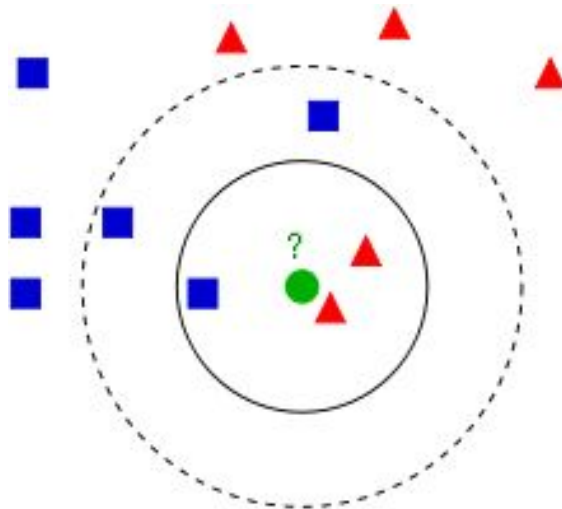
- Works both for Classification and Regression
- Non-parametric method, knn doesn't produce a model

Why Do We Need K-Nearest Neighbour ?

- Parametric method like linear regression and logistic regression make strong assumption about the form of the model. Ex.
 - $y = a + bx + e$
 - $P(Y = 1) = \text{odd}/(1+\text{odd})$
- If the specified functional form is far from the truth, the model will perform poorly because of the bad prediction result.
- A nonparametric method like KNN do not explicitly assume any form of model.
- In term of prediction, it's more flexible approach and It can capture any type of relationship.

Basic Idea

- Store/keep training data
- Classify new observation based on similarity to the observation in training data
- Use majority decision rule to classify the records



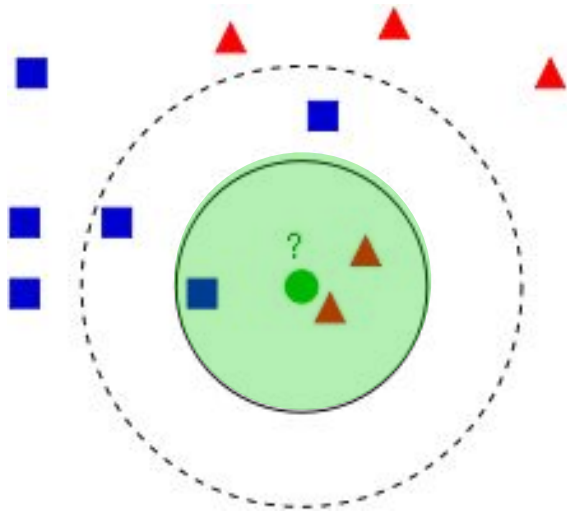
■ ▲ Training data

● Test data (New observation)

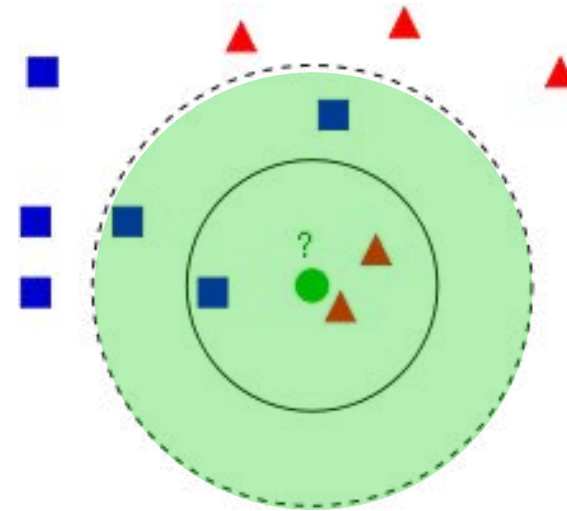
Test data will be classified as ■ or ▲ ?
(Ignore circle line for now)

Basic Idea

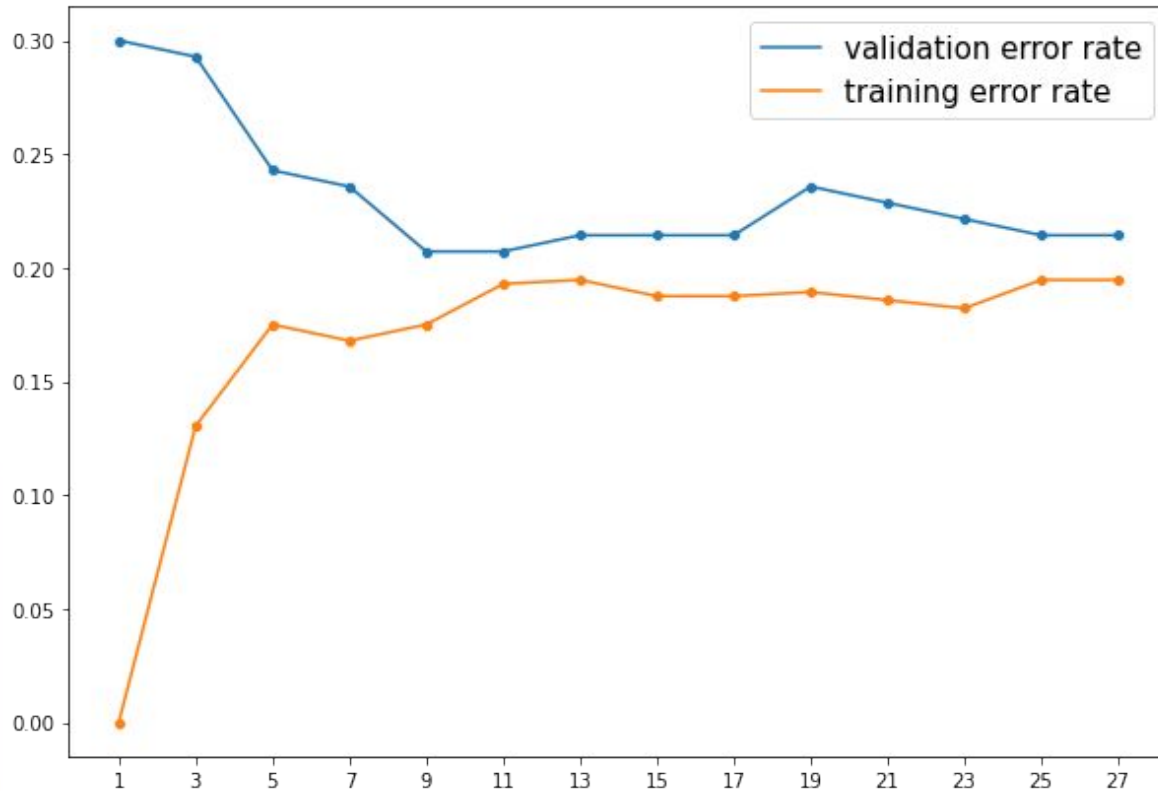
With **k=3** nearest neighbors
Test data classified into ▲



With **k=5** nearest neighbors
Test data classified into ■



How do we choose factor K?

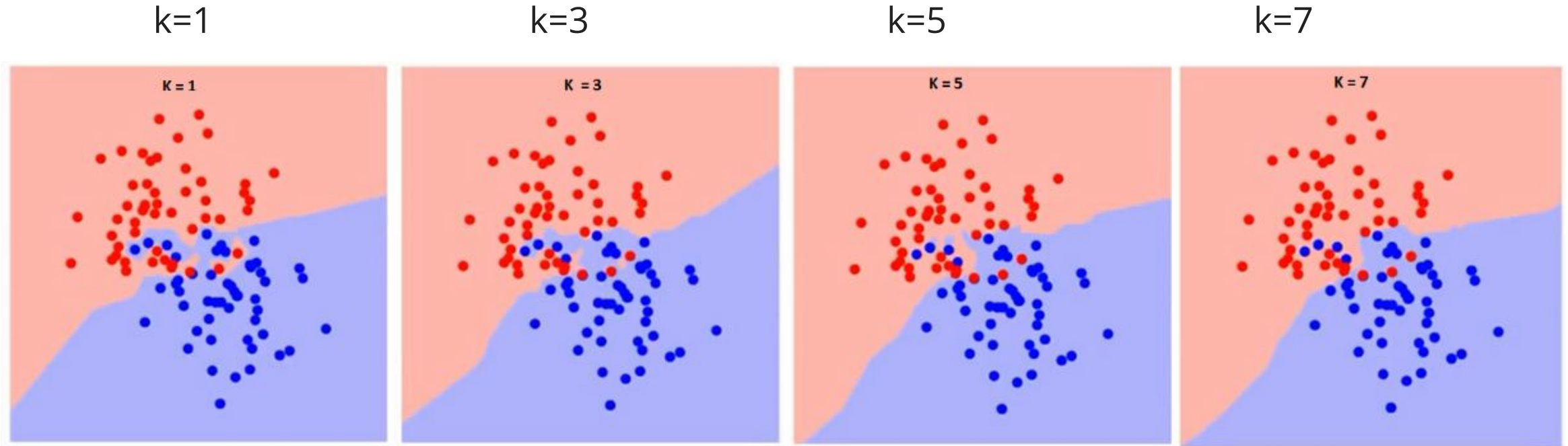


- Tips 1: Use odd number of K
- Tips 2: Evaluate using validation data set

Best k = 9

- Error rate at K=1 can perfectly predict training sample, closest point to any data point is itself
- Our goal is to predict new data so we want good performance in validation data set
- Performance at K=1 not acceptable to predict new data
- Error rate in validation set generally decreases with increases K
- We choose k with minimum error rate in validation dataset

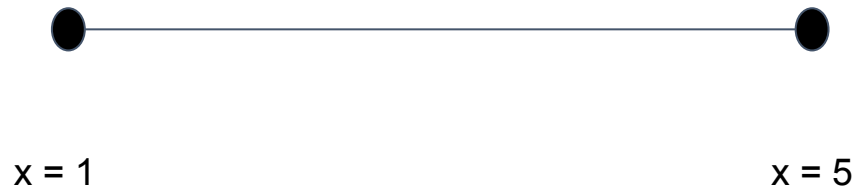
How do we choose factor K?



- Boundary becomes smoother with increase value of K
- With K increases to inf, finally becomes all-blue/all-red depending on total majority

Measuring distance 1 Dimension

- Closest neighbor is identified based on the distance
- There are several method to measure distance. The popular one is Euclidean.

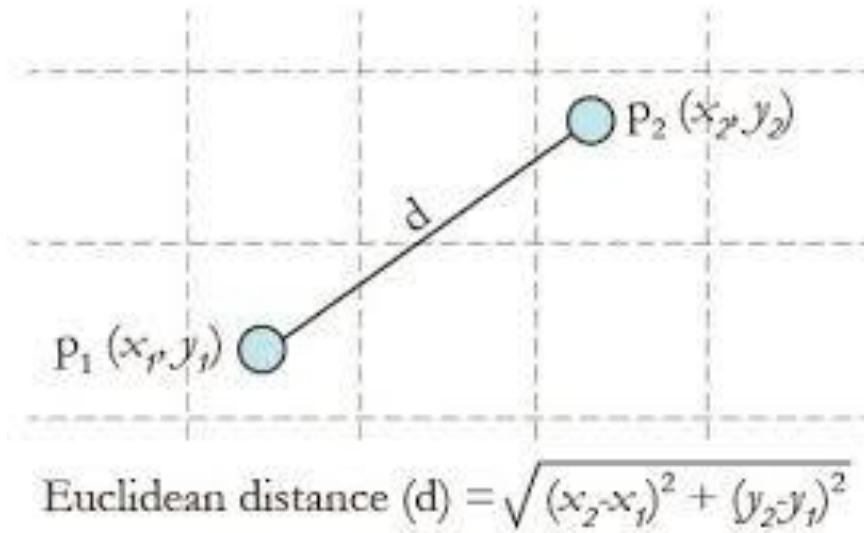


Simple Illustration

The distance is simply :

$$\begin{aligned}\text{Distance} &= 5 - 1 \\ &= 4\end{aligned}$$

Measuring distance 2 Dimension



Illustration

$$(x_1, y_1) = (1, 2)$$

$$(x_2, y_2) = (5, 4)$$

Euclidean distance

$$= \sqrt{(5 - 1)^2 + (4 - 2)^2}$$

$$= \sqrt{(4)^2 + (2)^2}$$

$$= \sqrt{20}$$

$$= 4.47$$

Measuring distance > 2 Dimension

$$Distance = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2 + \dots}$$

data points	x	y	z
1	12	15	16
2	12	16	17

$$Distance = \sqrt{(12 - 12)^2 + (16 - 15)^2 + (17 - 16)^2} = 1.414.$$

Measuring distance

- Distance calculations are performed for each data point against other data points.

data points	x1	x2	x3
1	12	15	16
2	12	16	17
3	20	13	18
4	9	14	18
5	17	15	20

data points	1	2	3	4	5
1	0				
2	1.414	0			
3	8.485	8.602	0		
4	3.741	3.7741	11.045	0	
5	6.403	5.916	4.123	8.306	0

Measuring distance

- The Closest Data Points

data points	1	2	3	4	5
1	0				
2	1.414	0			
3	8.485	8.602	0		
4	3.741	3.7741	11.045	0	
5	6.403	5.916	4.123	8.306	0

Distance Matrix

Data Points	1st closest	2nd closest	...
1	2	4	...
2	1	4	...
3	5	1	...
4	1	2	...
5	3	2	...

Issue with Euclidean Distance

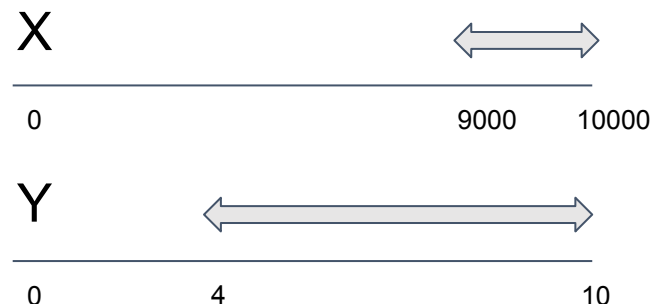
- Given X is Area with unit of hectare
- Given Y is Corn Production with unit of kwintals.

X	Y
4000	5
5000	4.5
2000	3
6000	4.5
7000	4.6
8000	4
9000	10

Look at the two last data points:

- in term of area (X) the diff is $9000 - 8000 = 1000$ hectare
- in term of production (Y) the diff is $10 - 6 = 4$ kg

Distance contribution from X (1000) is far surpassed by contribution from Y (4) due to different scale. So when euclidean distance used, variable with large scale will have larger effect on the distance.



Relatively, distance contribution from X is surpassed by distance contribution from Y. So when euclidean distance used, there should be any treatment like scaler or normalization.

Issue with distance

Solution to solve scale issue is **Normalization**.

◦Min-Max Scaling

Uses *MinMaxScaler*

Transform to defined range

$$y = \frac{x - \min x_i}{\max x_i - \min x_i}$$

◦Standardization

Uses *StandardScaler*

Transform to mean=0, sd=1

$$y = \frac{x - \bar{x}}{s}$$

Where

\bar{x} = mean

s = Standard deviation

Min-Max Scaling

Bigger Contribution
"area"

Look at the two last data points as example:

Variable	diff before	diff after
area	1000 hectare	0.143
production	6 Kw	0.858

Bigger Contribution
"production"

The Process

	area	production
0	4000	5.0
1	5000	4.5
2	2000	3.0
3	6000	4.5
4	7000	4.6
5	8000	4.0
6	9000	10.0



	area	production
0	0.285714	0.285714
1	0.428571	0.214286
2	0.000000	0.000000
3	0.571429	0.214286
4	0.714286	0.228571
5	0.857143	0.142857
6	1.000000	1.000000

X1

0 9000 10000

X2

0 4 10

Contribution in
reality

Advantages and Disadvantages

Advantages ?

- Able to provide decent accuracy in any situation
- Easy to learn
- Easy to program
- Training is fast

Disadvantages ?

- Need more space as the training data grow
- Low interpretability
- KNN doesn't know which feature are actually important

Python Exercise : Voting Classifier

Analyze data white_wine.csv

- Apply KNN Method ($k = 3$)
 - target : quality (quality >6 $\rightarrow Y = 1$)
 - features : density alcohol
- Validate the model using accuracy in 20% testing data
- Apply scaling and Validate the model using accuracy in 20% testing data
- Apply scaling Choose Factor K based on accuracy:
 - $K = (1, 3, 5, \dots, 29)$

Decision Tree Classifier

What is Decision Tree ?

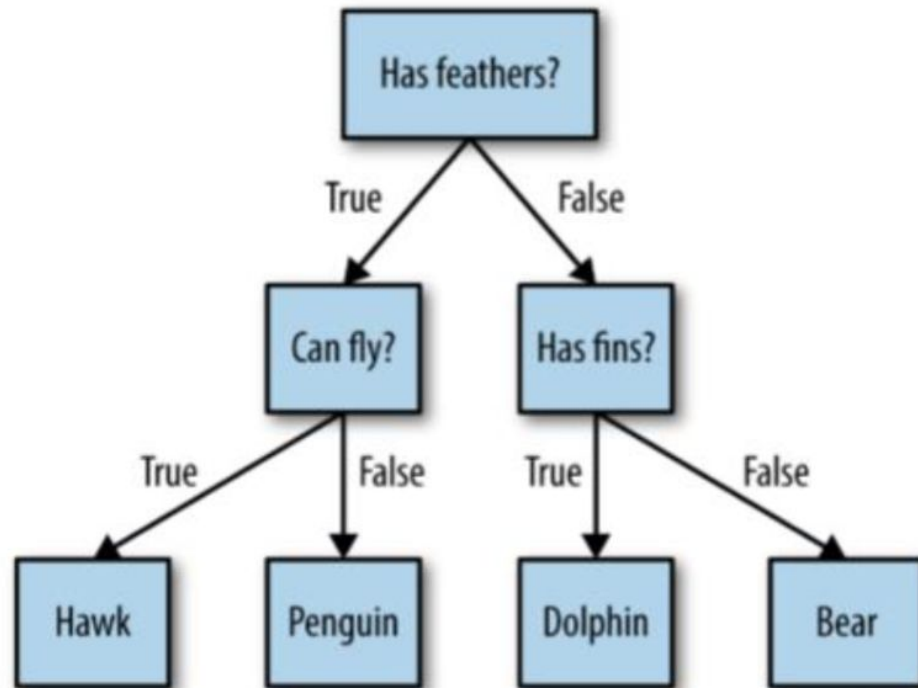
Essentially, Decision Tree is a hierarchy of if/else questions, leading to a decision.

- Decision Tree is widely used ML Algorithm
- Decision Tree also can be applied to regression problem

Why do We Need Decision Tree ?

- It's also a nonparametric approach like KNN
 - we don't need to make any strong assumption about the form of the model
 - it's more flexible approach to capture any type of relationship.
- You want a fast, flexible model with high interpretability

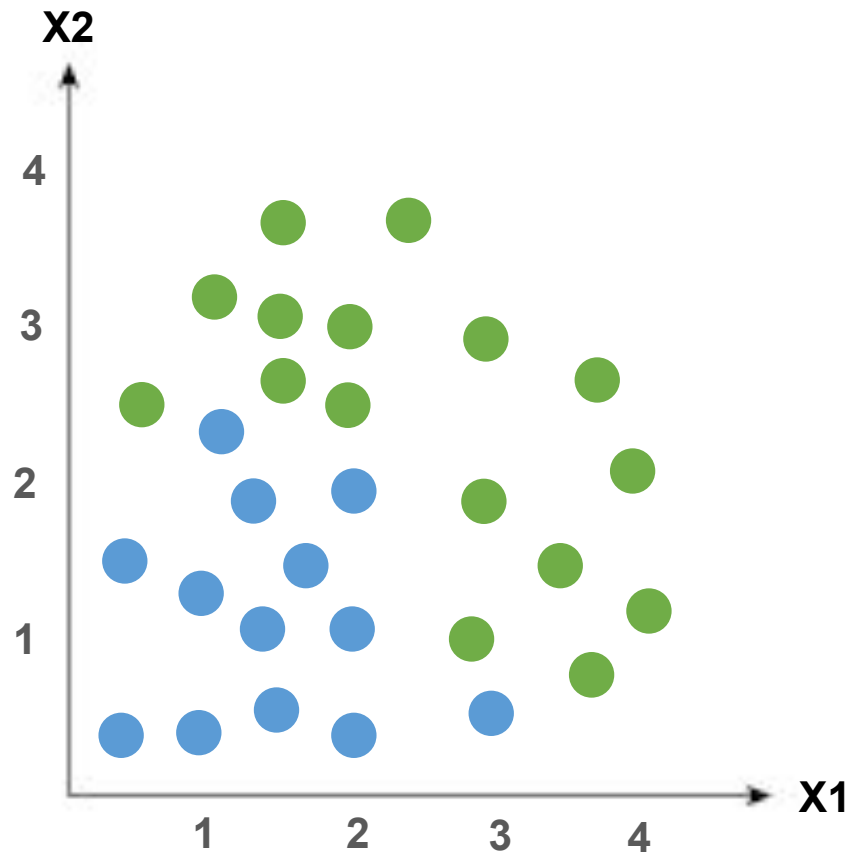
Decision Analogy



Analogyes:

- Here we want to differentiate Hawk, Penguin, Dolphin, Bear by using characteristics as few as possible
- In term of classification:
 - Animals (Hawk, Penguin, Dolphin, Bear) → Target
 - Characteristics → feature
- We must be thinking what characteristics can differentiate them:
 - Among These animals, which one has feathers
 - feathers yes : Hawk and Penguin
 - feathers no : Dolphin and Bear
 - This is not enough we still need additional information
 - feather yes : add can fly or not
 - fly : Hawk
 - do not fly : Penguin
 - feather no : has fins or not
 - Has Fins : Dolphin
 - No Fins : Bear

Basic Idea

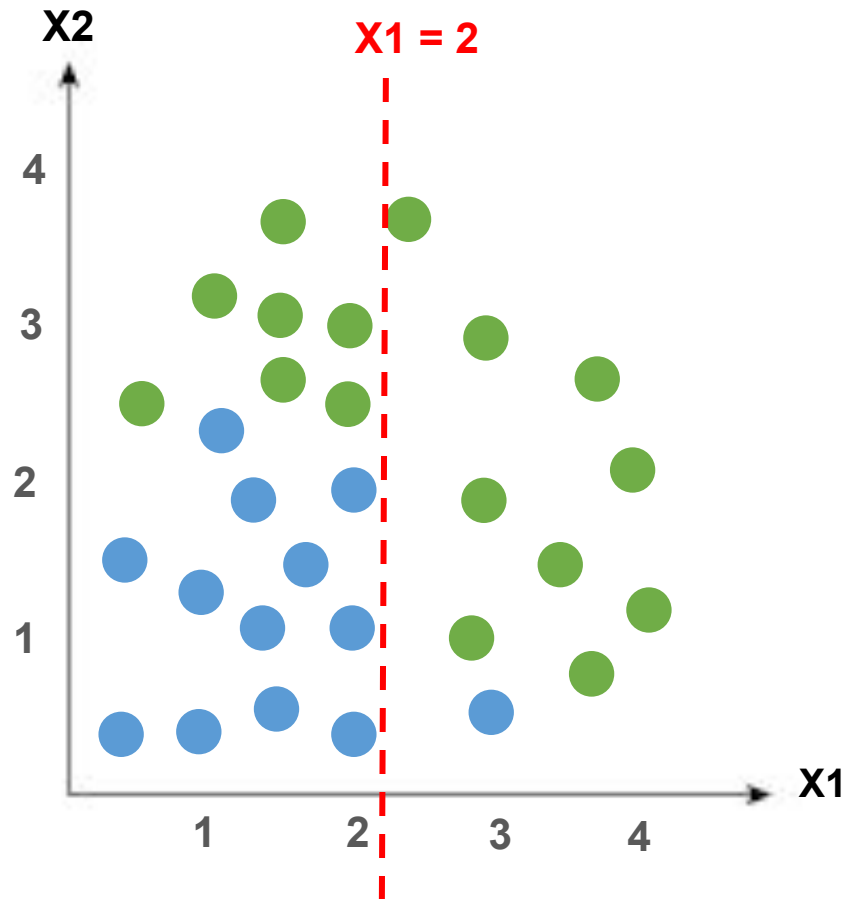


● 16 obs.

● 13 obs.

- Find the best splitter between X1 & X2
- Best splitter is the one results most homogenous element in each class
- Splitter X1 = 2, X1 = 3, etc

Basic Idea



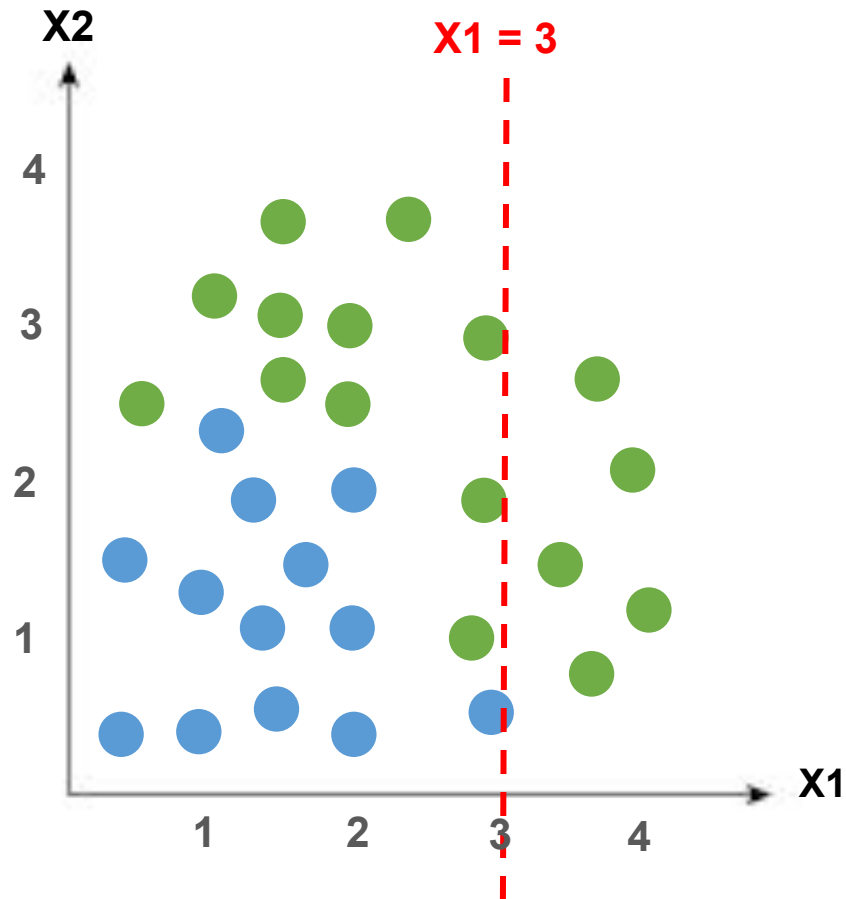
- Splitting at $X1 = 2$
- Split to 2 subset

Subset $X1 < 2$ ● 7 obs.
● 12 obs.

Subset $X1 > 2$ ● 9 obs.
● 1 obs.

How good is this split?

Basic Idea



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

Class $X1 < 3$ ● 11 obs.
 ● 13 obs.

Class $X1 > 3$ ● 5 obs.
 ● 0 obs.

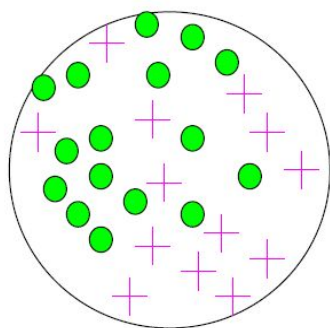
Is this split better?

Entropy

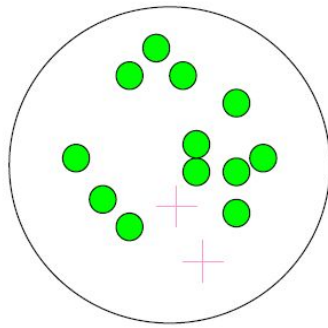
Entropy is a measure of heterogeneity

- given dataset or subset D, consist of 2 class YES and NO
- entropy : $E(D) = -(p \log_2 p + q \log_2 q)$, with p is YES class proportion and q is NO proportion
- Dataset or subset with all-YES or all-NO will have $E(D) = 0$
- Entropy will have its maximum value when $p = 0.5$

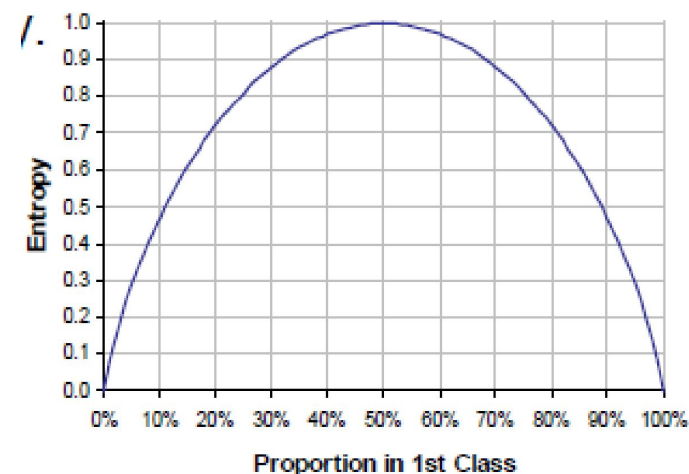
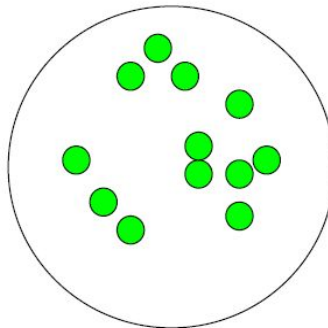
Very impure group



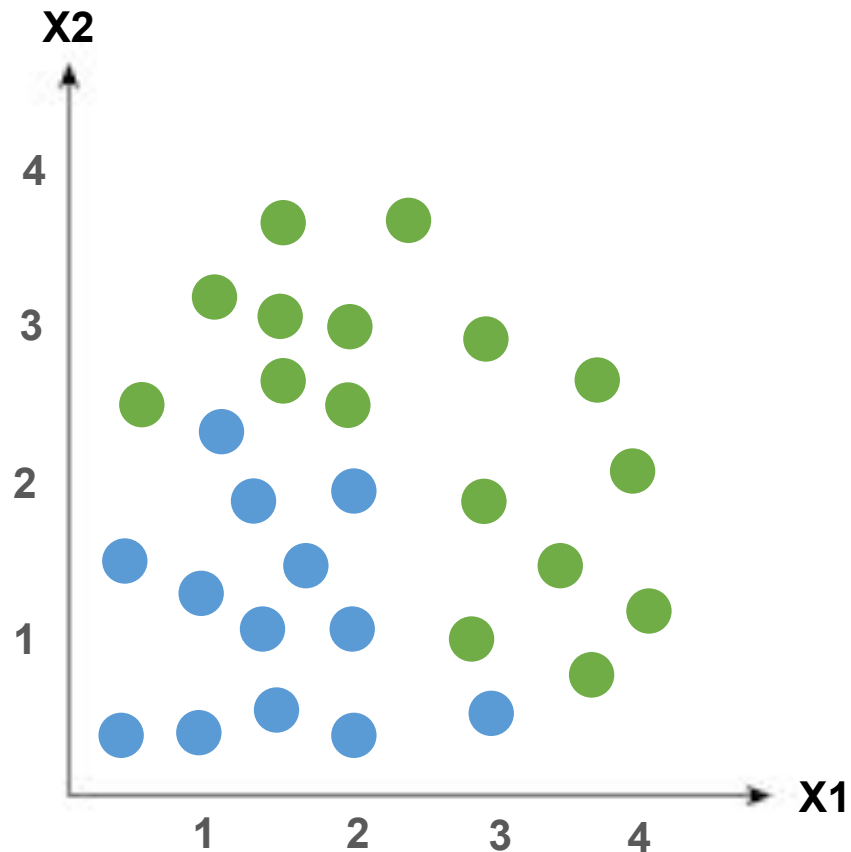
Less impure



Minimum impurity



Entropy Dataset



● 16 obs.

● 13 obs.

Entropy for this data set is
 $p = \text{green proportion} = 16/29$
 $q = \text{blue} = 13/29$

$$\begin{aligned} E(D) &= - (p \log_2 p + q \log_2 q) \\ &= - (16/29 \log_2 (16/29) + 13/29 \log_2 (13/29)) \\ &= 0.9922... \end{aligned}$$

Information Gain

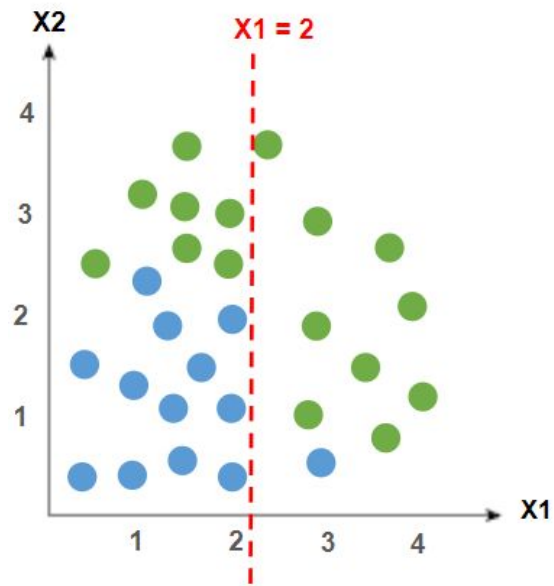
- Information gain measure the quality of a split.
- Better splitter = Higher Information Gain
- Let's say dataset D is split into several groups D1, D2, ... , Dk based on variable V.
Ex. $X1 < 2$ and $X1 > 2$; Single, Married and Divorce, variable marital status.
- For every Di, Entropy can be calculated as $E(D_i) = -p_i \log p_i - (1-p_i) \log (1-p_i)$
- Information Gain:

$$\text{Information Gain} = E(D) - E(\text{Split})$$

$$E(\text{split}) = \text{Weighted_Average}(E(D_1), E(D_2), \dots, E(D_k))$$

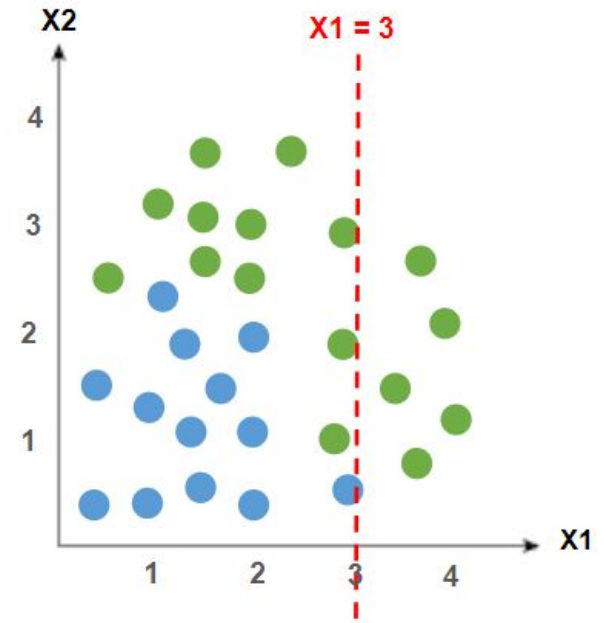
Back to Basic Idea

Which one is better?



● 7 obs.
● 12 obs.

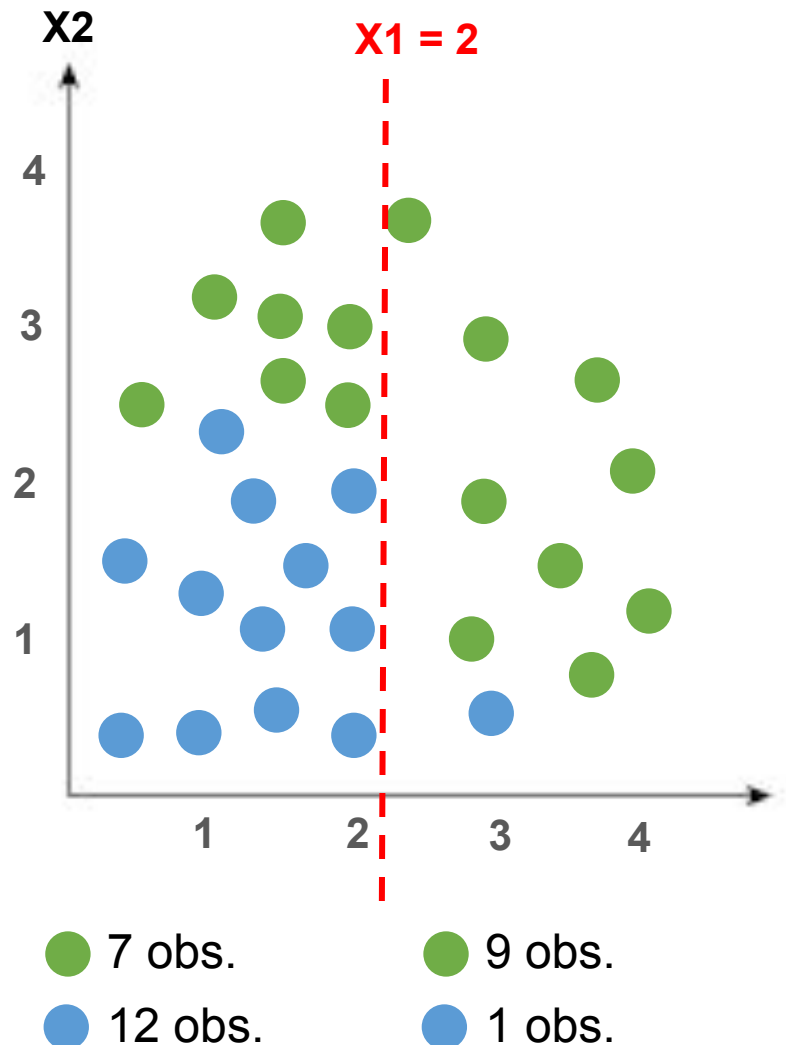
● 9 obs.
● 1 obs.



● 11 obs.
● 13 obs.

● 5 obs.
● 0 obs.

Entropy Split



$X_1 < 2$:

p = green proportion = $7/19$

q = blue = $12/19$

$$\begin{aligned} E(X_1 < 2) &= - (p \log_2 p + q \log_2 q) \\ &= - (7/19 \log_2 (7/19) + 12/19 \log_2 (12/19)) \\ &= 0.9494... \end{aligned}$$

$X_1 > 2$:

p = green proportion = $9/10$

q = blue = $1/10$

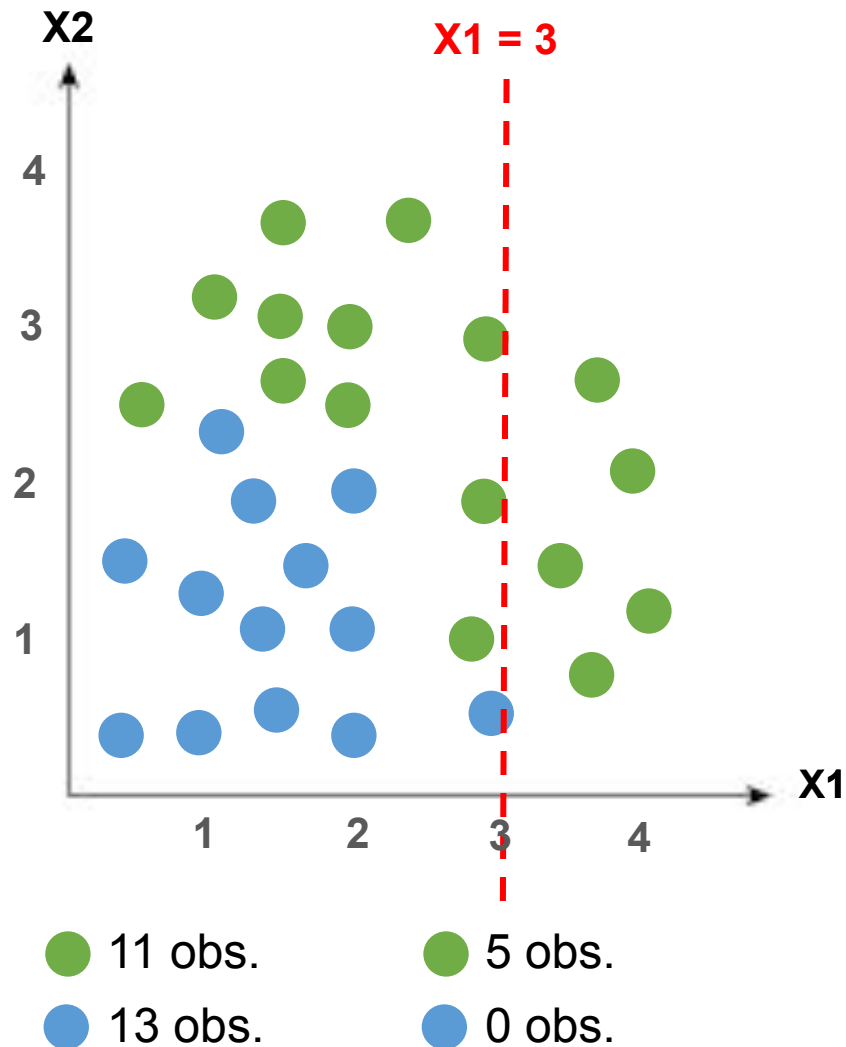
$$\begin{aligned} E(X_1 > 2) &= - (p \log_2 p + q \log_2 q) \\ &= - (9/10 \log_2 (9/10) + 1/10 \log_2 (1/10)) \\ &= 0.4689... \end{aligned}$$

Entropy Split :

$$E(S) = 19/29 (0.9494...) + 10/29 (0.4689...) = 0.78377$$

$$IG = E(D) - E(S) = 0.9922... - 0.7837... = 0.2084...$$

Entropy Split



$X_1 < 3$:

$p = \text{green proportion} = 11/24$

$q = \text{blue} = 13/24$

$$\begin{aligned} E(X_1 < 3) &= - (p \log_2 p + q \log_2 q) \\ &= - (11/24 \log_2 (11/24) + 13/24 \log_2 (13/24)) \\ &= 0.9949... \end{aligned}$$

$X_1 > 3$:

$p = \text{green proportion} = 5/5$

$q = \text{blue} = 0/5$

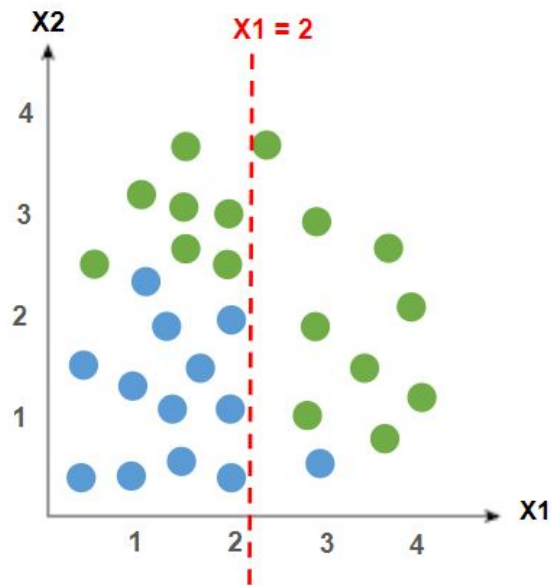
$$\begin{aligned} E(X_1 > 3) &= - (p \log_2 p + q \log_2 q) \\ &= - (5/5 \log_2 (5/5) + 0/5 \log_2 (0/5)) \\ &= 0 \end{aligned}$$

Entropy Split :

$$E(S) = 24/29 (0.9949...) + 5/29 (0) = 0.6518...$$

$$IG = E(D) - E(S) = 0.9922... - 0.8234... = 0.1688...$$

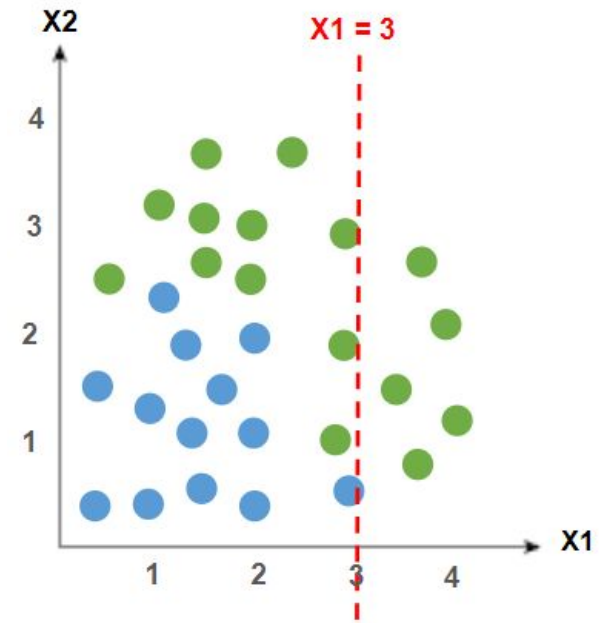
Information Gain



● 16 obs.
● 13 obs.

● 7 obs. ● 9 obs.
● 12 obs. ● 1 obs.

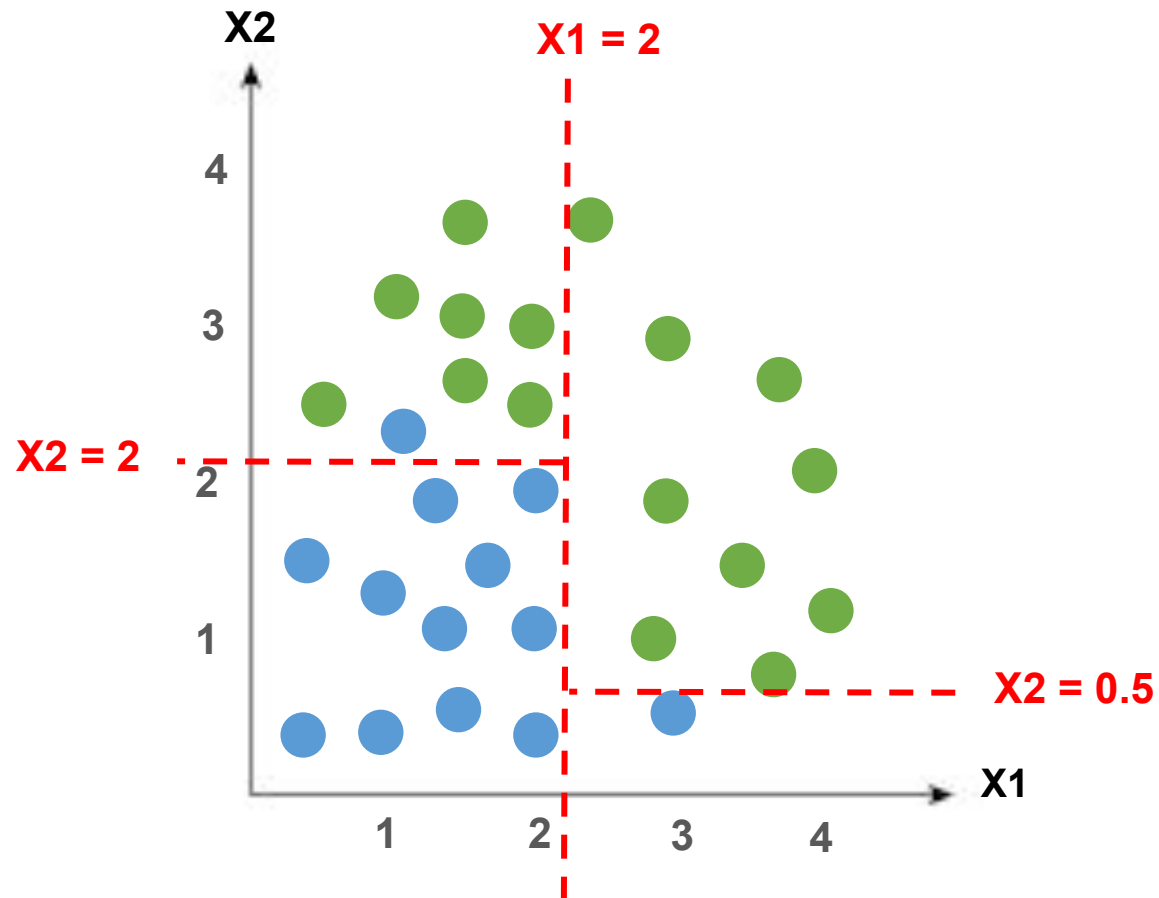
IG = 0.2084



● 11 obs. ● 5 obs.
● 13 obs. ● 0 obs.

IG = 0.1688

Basic Idea – Continue Splitting

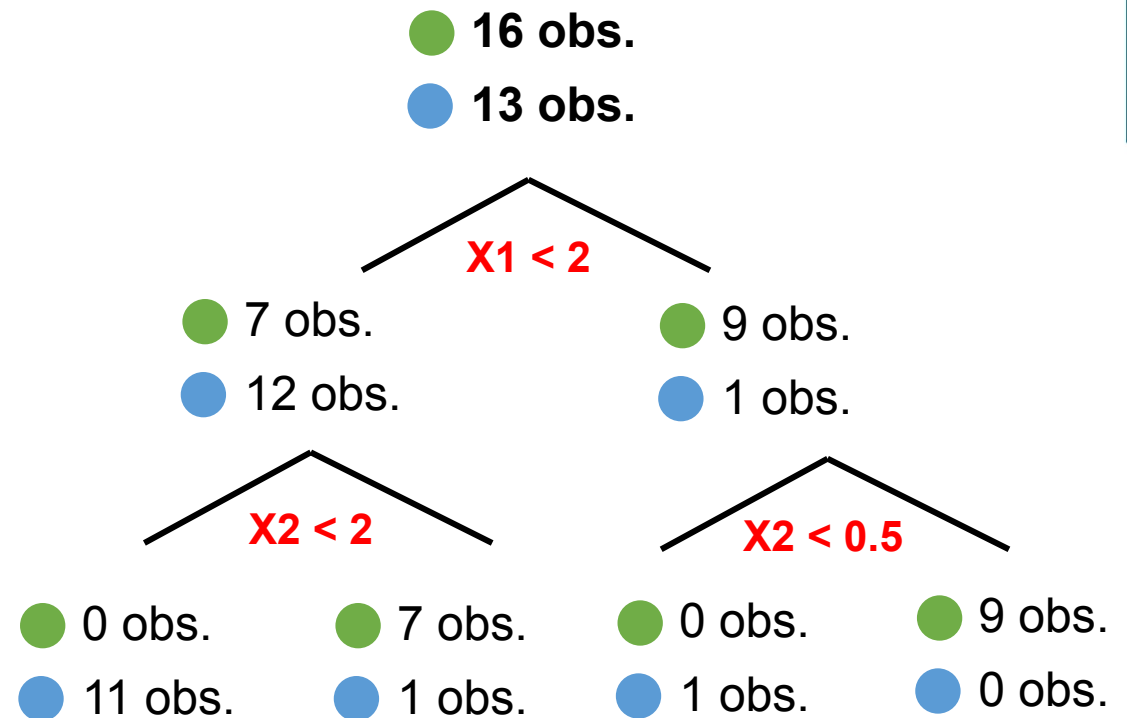
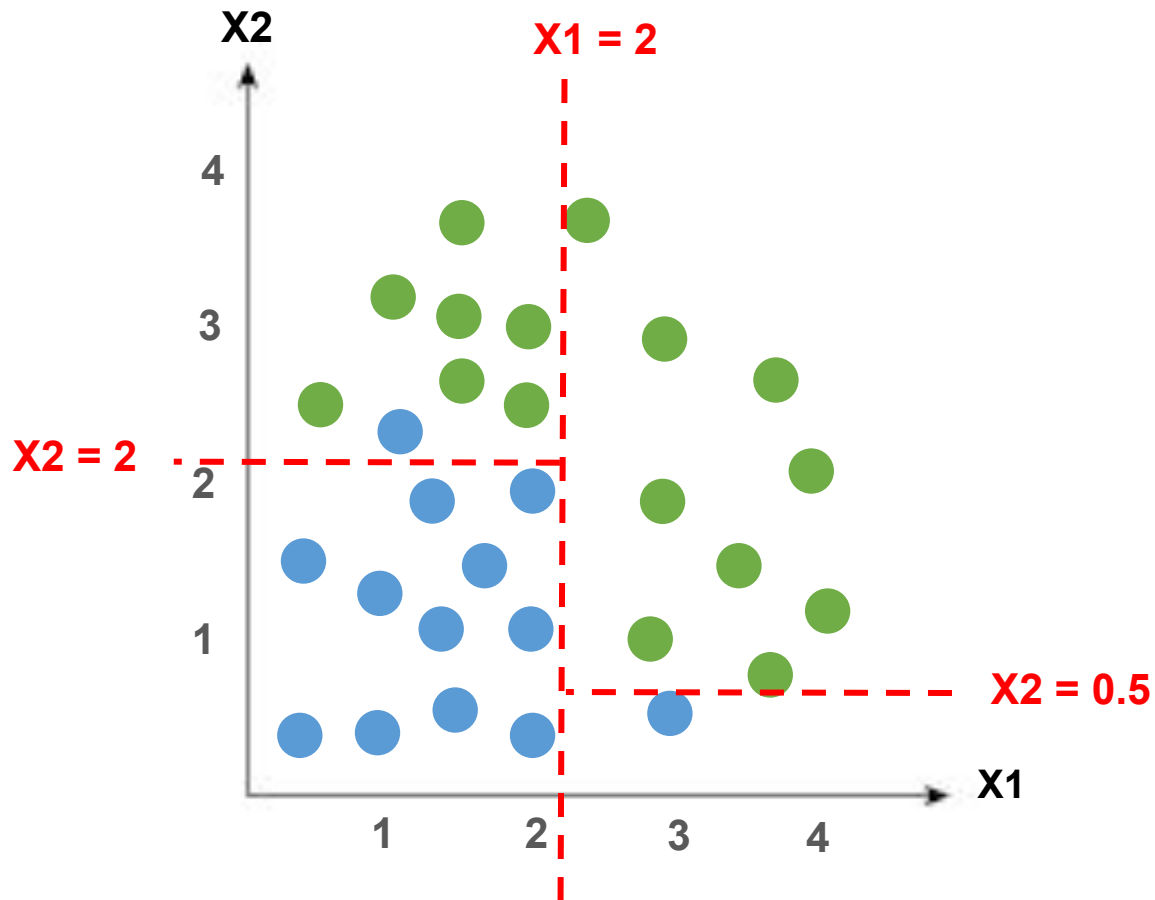


Continue splitting on each subset.

At $X_2 = 2$ for class $X_1 < 2$

At $X_2 = 0.5$ for class $X_1 > 2$

Basic Idea – Continue Splitting

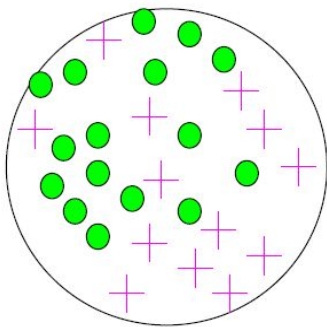


Gini Index

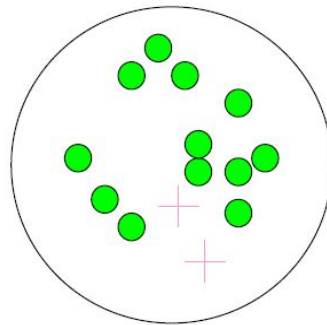
Gini is a measure of heterogeneity

- given dataset or subset D, consist of 2 class YES and NO
- gini : $G(D) = p q + q p$, with p is YES class proportion and q is NO proportion
- Dataset or subset with all-YES or all-NO will have $G(D) = 0$
- Entropy will have its maximum value when $p = 0.5$

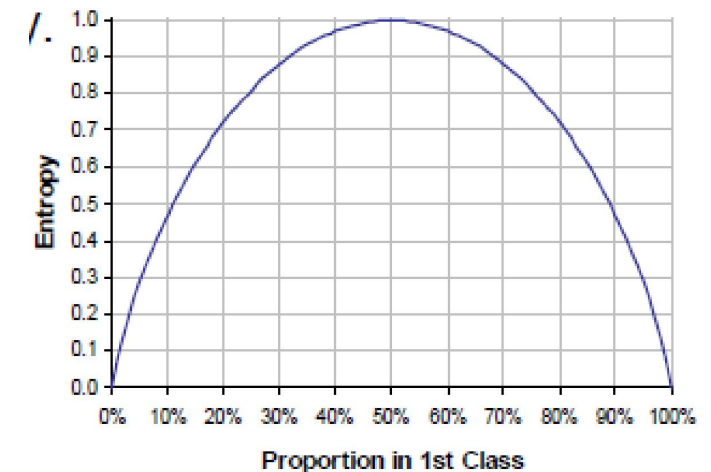
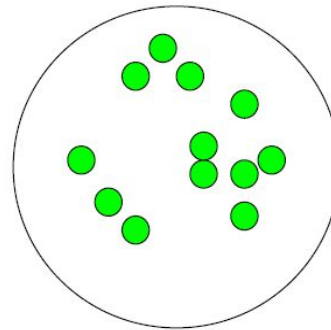
Very impure group



Less impure



Minimum impurity



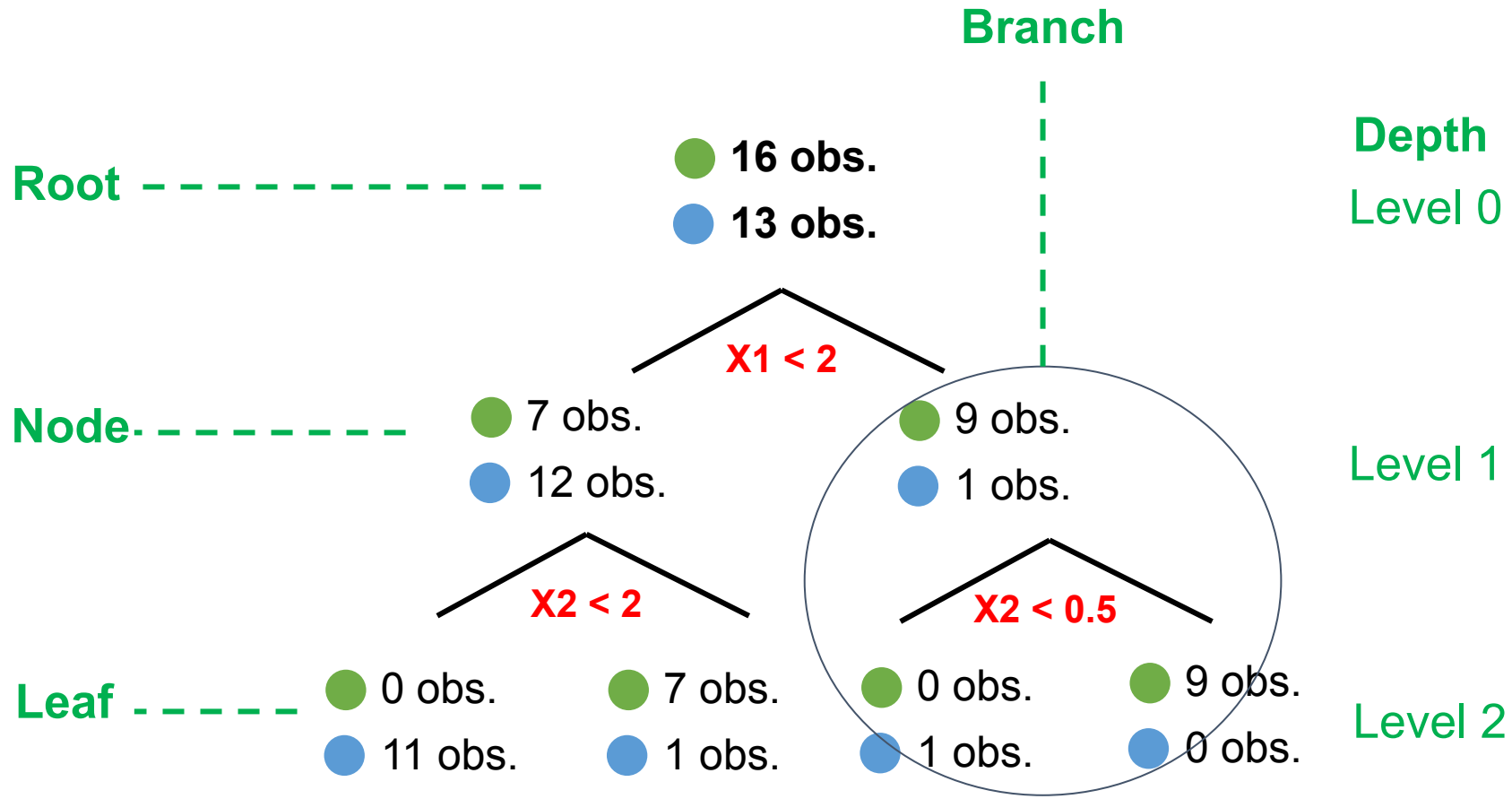
Gini Impurity

- Gini gain measure the quality of a split.
- Better splitter = Higher Gini Gain
- Let's say dataset D is split into several groups D1, D2, ... , Dk based on variable V.
Ex. $X1 < 2$ and $X1 > 2$; Single, Married and Divorce, variable marital status.
- For every Di, Gini can be calculated as $G(D_i) = p_i q_i + q_i p_i$
- Gini Gain:

$$\text{Gini Impurity} = G(D) - G(\text{Split})$$

$$G(\text{split}) = \text{Weighted_Average}(G(D1), G(D2), \dots, G(Dk))$$

Terminologies



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 reached 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

Advantages and Disadvantages

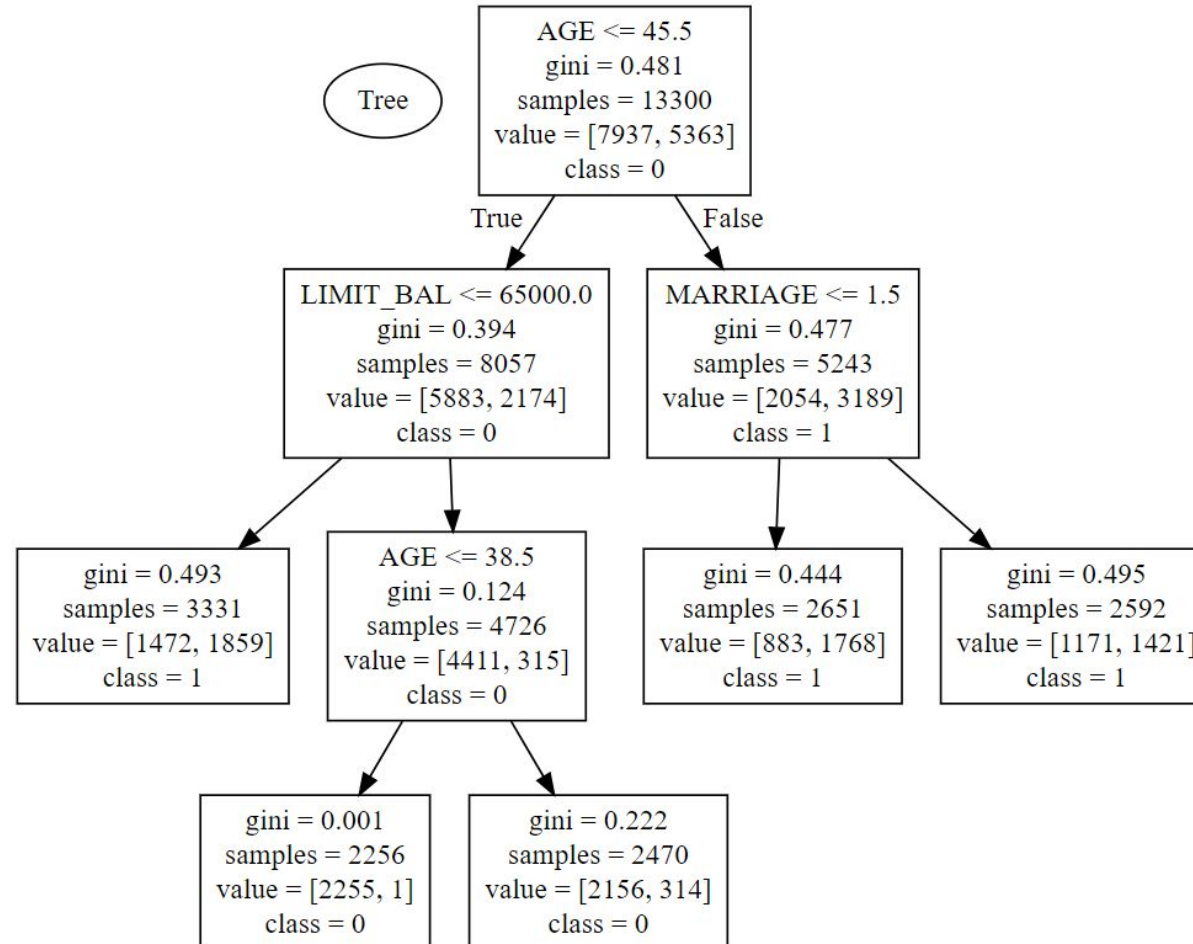
Advantages

- Easy to understand
- Useful in data exploration.
 - Can be visualized graphically
 - Information of importance variables, variables which relates each other.
- Data type is not a constraint (works for numerical and categorical too)

Disadvantages

- Often not stable/overfitting
- Loses information of continuous numeric variable when it categorized into different categories
- Cant compete with method like bagging, random forest and boosting in many situation
- Deeper tree are harder to interpret

Use Case : Credit Scoring



Use Case : Direct Marketing

Store 'M' gathered their customer data that has used a certain product 'N' to predict whether new customer will be spend money to buy it or not.

Target:

- Buy ($Y = 1$)
- No Buy ($Y = 0$)

Featurest:

- Gender
- Marital Status
- Domicili
- Age
- Smoker Status
- Budget
- Preference

No	Jenis Kelamin	Single	Tinggal di Kota	usia	Perokok	Budget	Kesukaan	Tertarik Beli?
1	1	0	1	32	0	low	Tekno	0
2	0	1	0	38	0	medium	Tekno	0
3	0	0	0	33	0	low	Tekno	0
4	0	1	0	27	0	medium	Lainnya	0
5	1	1	1	30	0	medium	Busana	1
6	0	1	0	44	0	medium	Tekno	0
7	1	1	1	36	0	medium	Seni	1
8	1	0	0	32	0	low	Seni	0
9	0	1	0	31	0	medium	Seni	0
10	1	1	0	40	0	high	Tekno	0
11	1	0	1	34	0	low	Lainnya	0

How to Build Decision Tree ?

Some Questions :

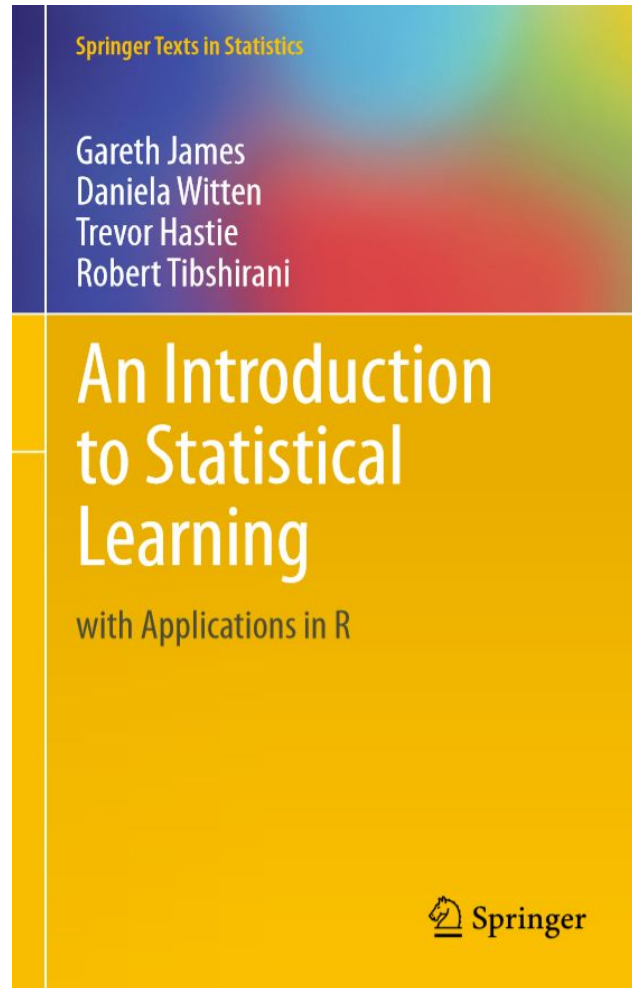
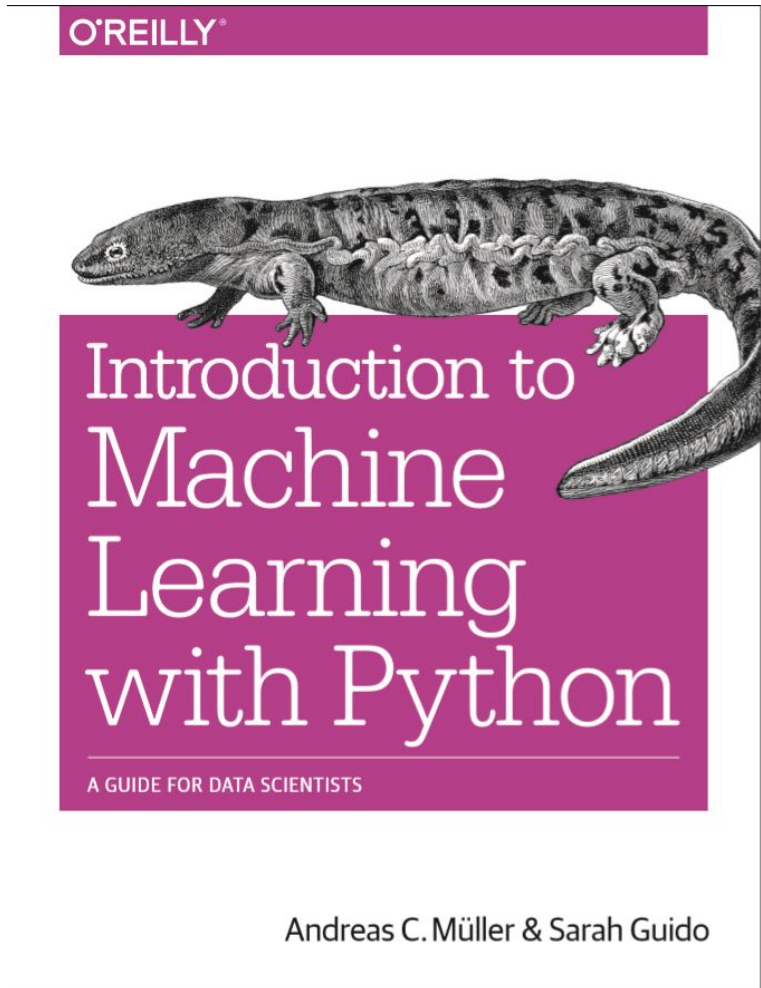
- How to determine which Features can differentiate the most so we can make the first split? Gender, Marital, Age,... or which one and What metric to measure how good feature can differentiate target variable ?
- After first Split made how to find the next feature that can differentiate the most ?
- How many times we should do the splitting ?
2 times enough???, 3 times ???,
as many as possible ?

Python Exercise : Decision Tree Classifier

Analyze data `white_wine.csv`

- Apply Decision Tree Classification Method (`max_depth = 2`)
 - target : quality ($\text{quality} > 6 \rightarrow Y = 1$)
 - features : density alcohol
- Validate the model using accuracy in 20% testing data
- Apply scaling and Validate the model using accuracy in 20% testing data
- Check the performance: is there any difference whether we applied scaling or not ?
- Check the tree: is there any difference whether we applied scaling or not ?

References



References

<https://www.the-modeling-agency.com/crisp-dm.pdf>

<https://scikit-learn.org/stable/>

<https://victorzhou.com/blog/information-gain/#:~:text=Information%20Gain%20is%20calculated%20for,chosen%20by%20maximizing%20Information%20Gain.>

<https://victorzhou.com/blog/gini-impurity/>