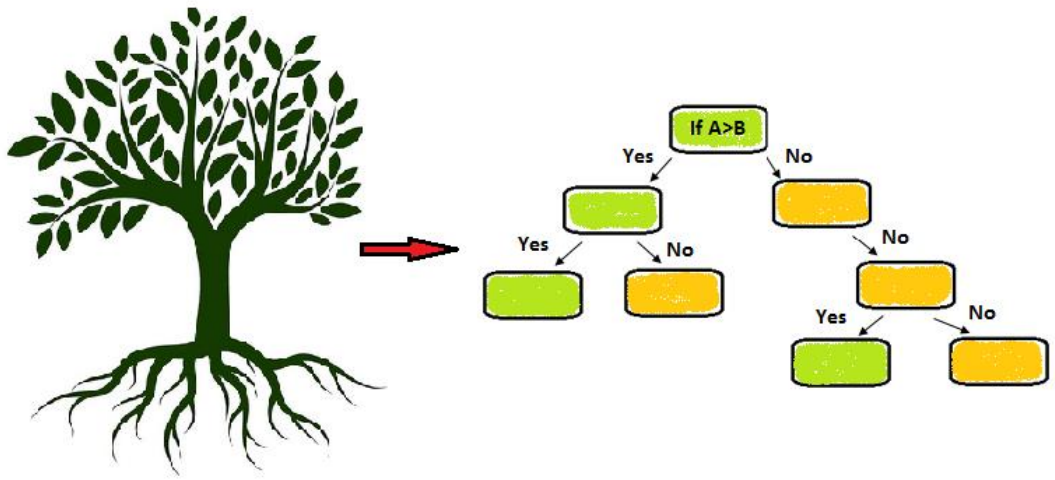
AI VIET NAM
@aivietnam.edu.vn

Decision Tree For Classification

(Basic, Advanced Concepts and Its Applications)



Vinh Dinh Nguyen
PhD in Computer Science

Outline

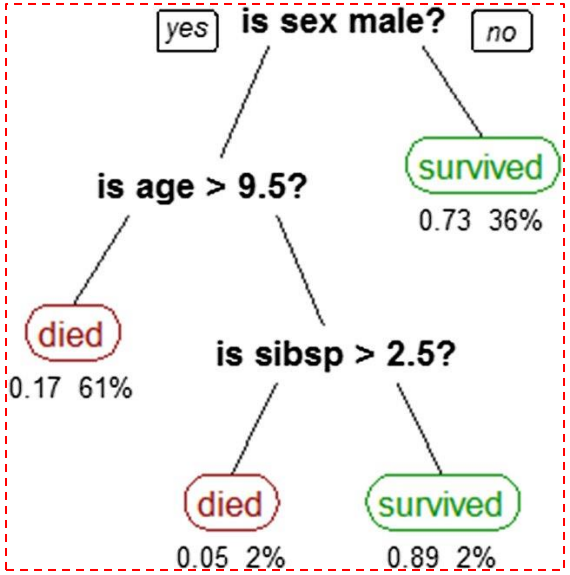
- Introduction to Decision Tree
- Classification Tree with GINI
- Classification Tree with Entropy
- Examples
- Summary

AI VIET NAM
@aivietnam.edu.vn

Sample Decision Tree

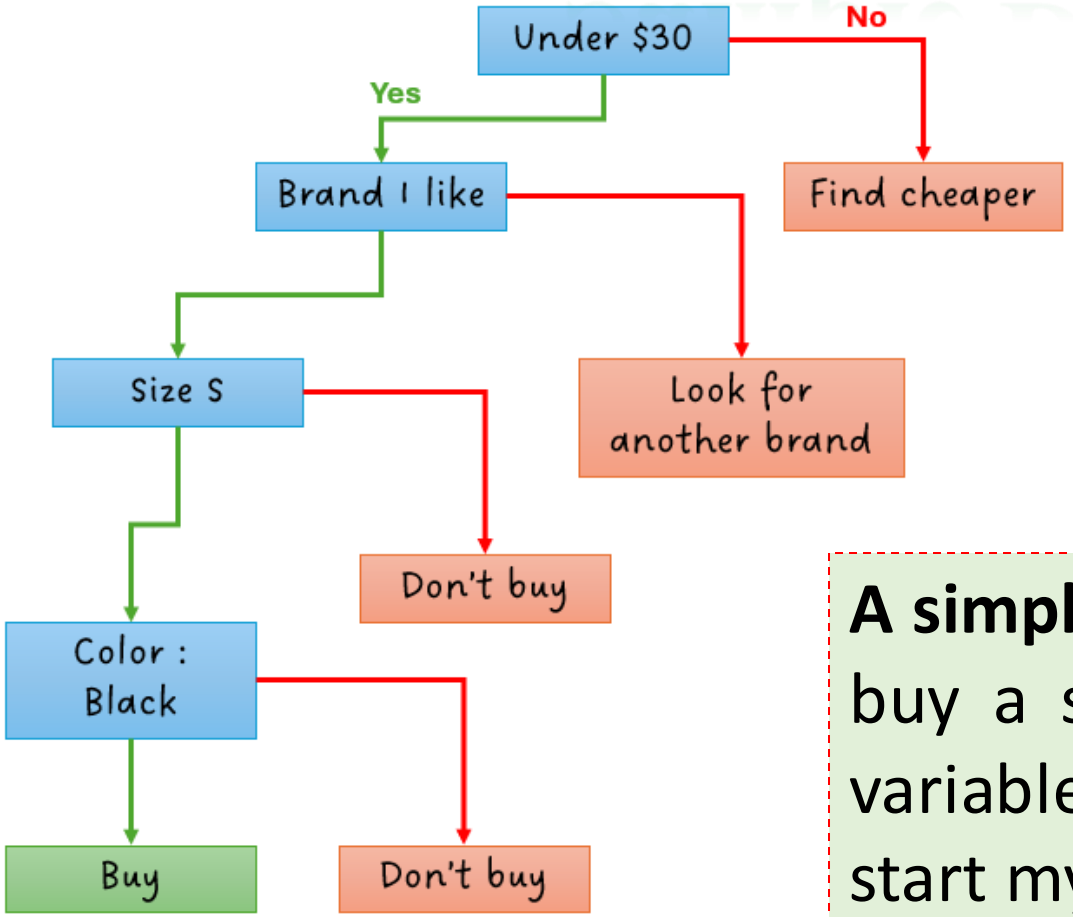
A decision tree predicting the survival of a passenger on the Titanic using the sex, age, and siblings or spouses onboard attributes

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S



Milborrow S (2011) Titanic decision tree. https://commons.wikimedia. org/wiki/File:CART_tree_titanic_survivors.png. Accessed Mar 2019

Sample Decision Tree

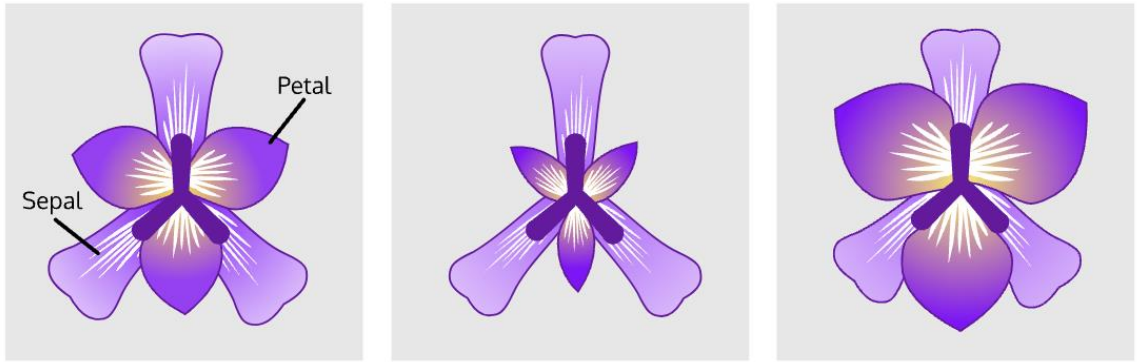


A simple example: buying a T-shirt. If I want to buy a shirt, I may have in mind a couple of variables like price, brand, size, and color. So I start my decision process from a budget

AI VIET NAM
@aivietnam.edu.vn

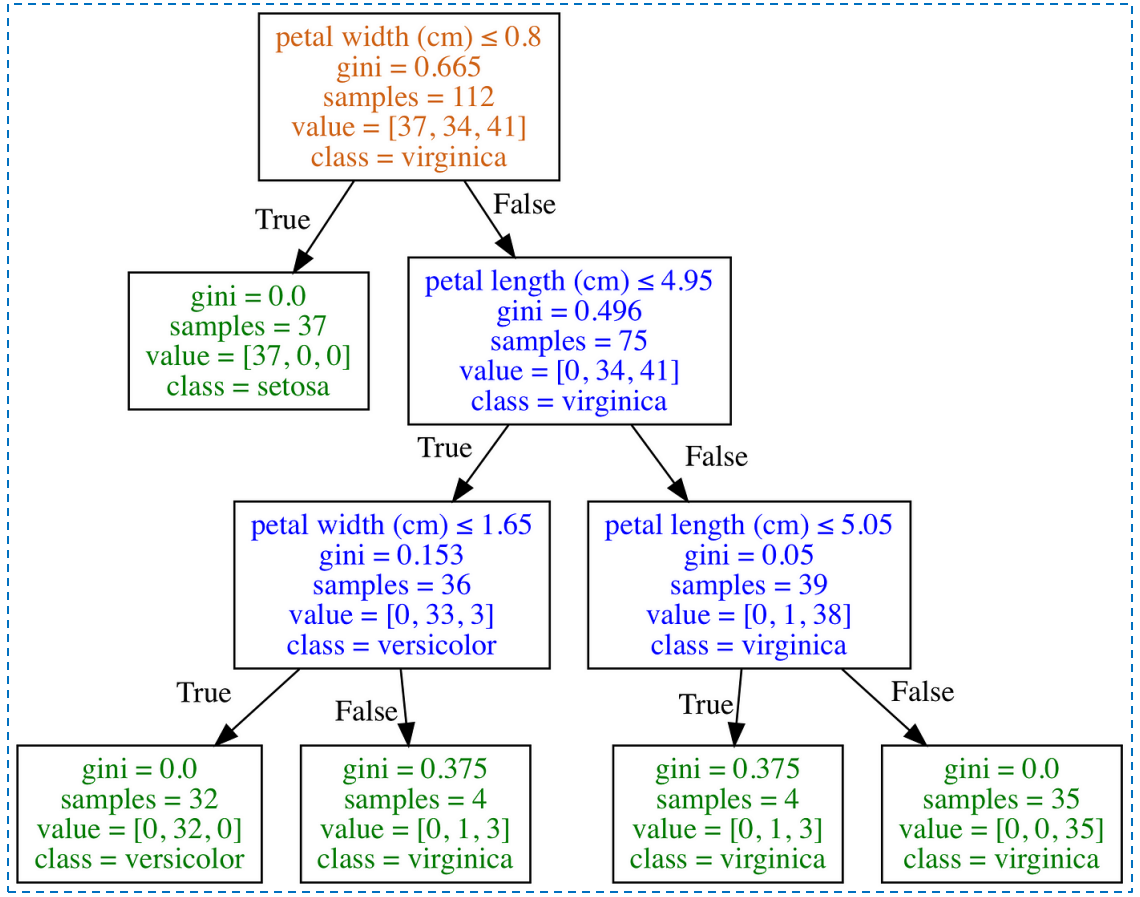
Sample Decision Tree

Classfying Iris flower based on its attributes: sepal length, sepal width, petal length, petal width

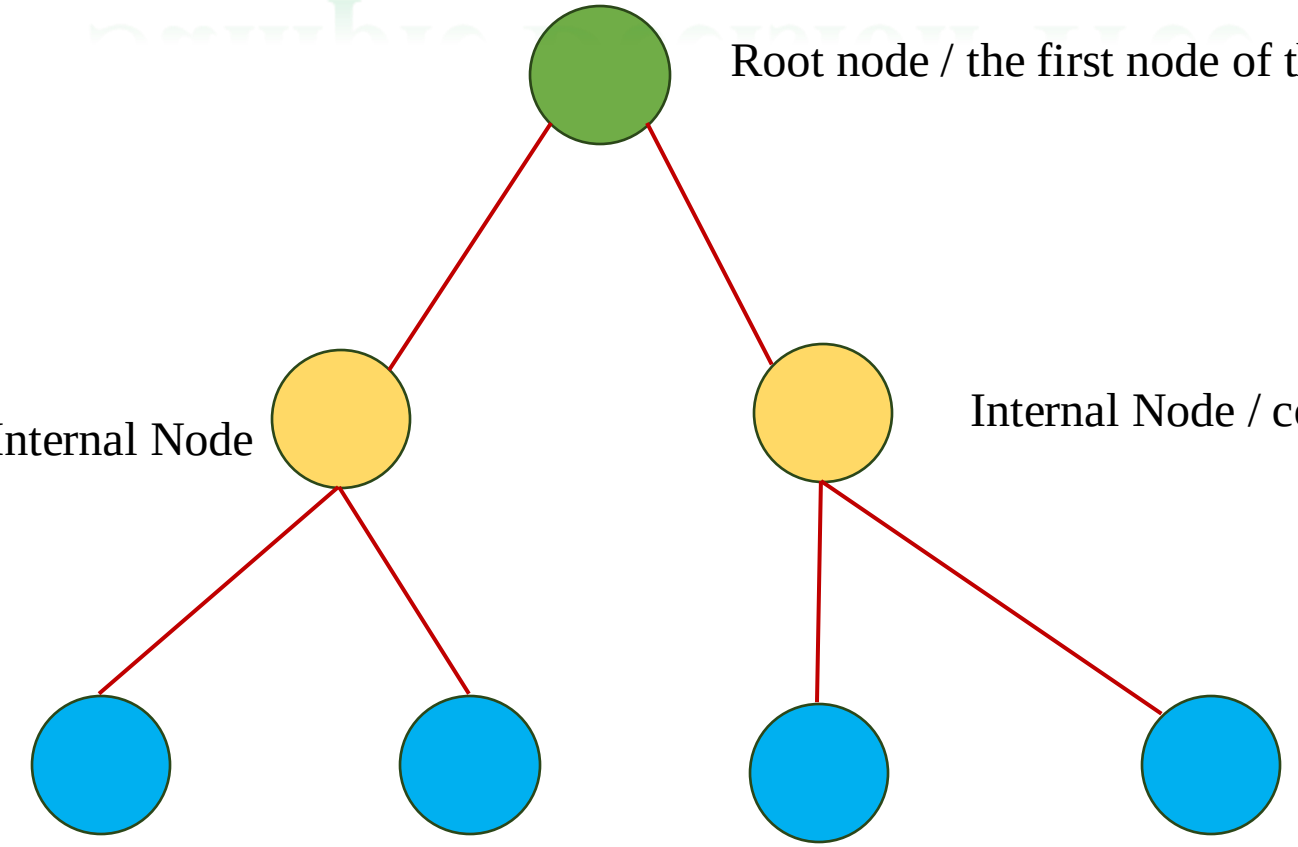


Iris Versicolor | Iris Setosa | Iris Virginica

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa



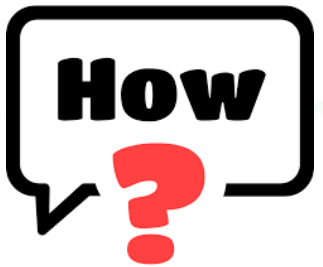
Sample Decision Tree

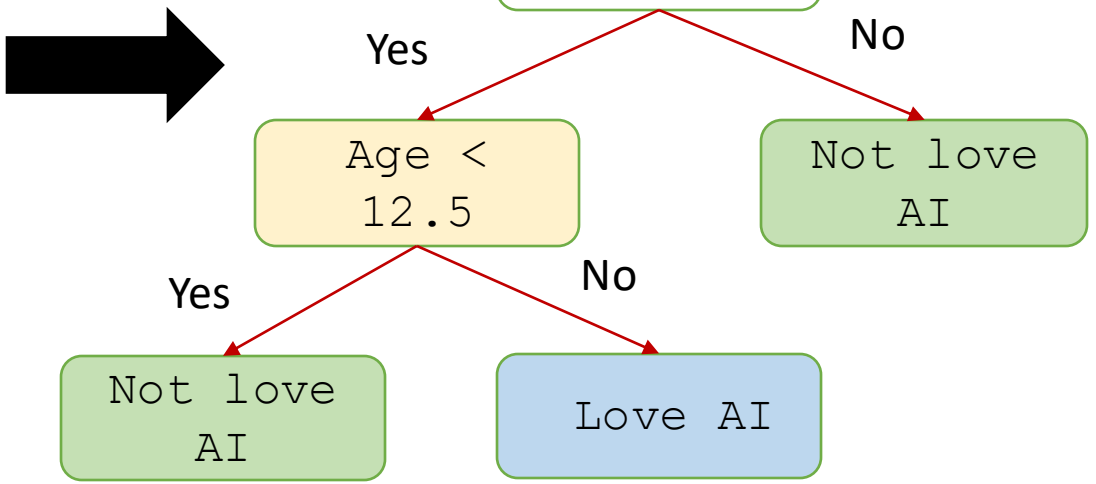


AI VIET NAM
@aivietnam.edu.vn

From Dataset to Decision Tree

Love Math	Love Art	Age	Love AI
Yes	Yes	7	No
Yes	No	12	No
No	Yes	18	Yes
No	Yes	35	Yes
Yes	Yes	38	Yes
Yes	No	50	No
No	No	83	No





Decision Tree Theory

•**Step-1:** Begin the tree with the root node, says S, which contains the complete dataset.

•**Step-2:** Find the best attribute in the dataset using **Attribute Selection Measure (ASM).**

•**Step-3:** Divide the S into subsets that contains possible values for the best attributes.

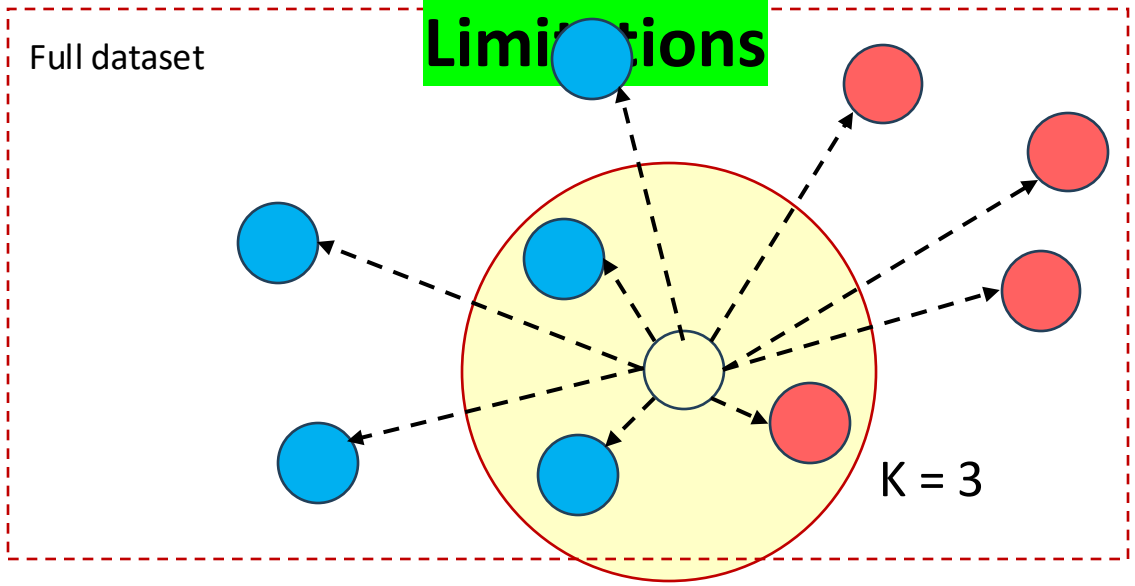
•**Step-4:** Generate the decision tree node, which contains the best attribute.

•**Step-5:** Recursively make new decision trees using the subsets of the dataset created in step -3. Continue this process until a stage is reached where you cannot further classify the nodes and called the final node as a leaf node.

AI VIET NAM
@aivietnam.edu.vn

Decision Tree Motivation

KNN Limitations



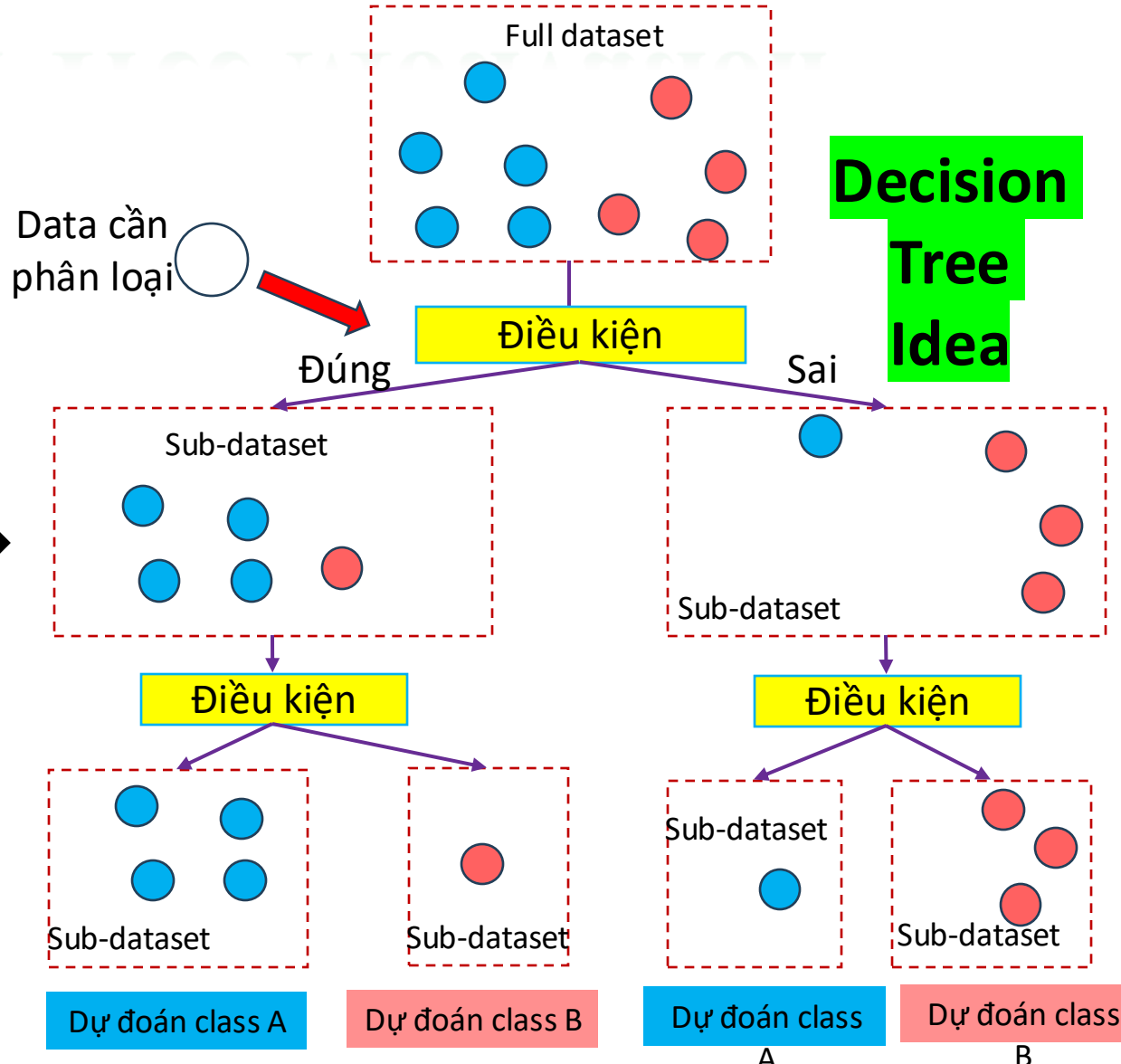
Data cần phân loại

Class A

Class B

KNN has some drawbacks and challenges, such as computational expense, slow speed, for large datasets

Decision Tree Idea



How to Build Decision Tree

No.	Love Math	Love Art	Age	Love AI
1	Yes	Yes	7	No
2	Yes	No	12	No
3	No	Yes	18	Yes
4	No	Yes	35	Yes
5	Yes	Yes	38	Yes
6	Yes	No	50	No
7	No	No	83	No

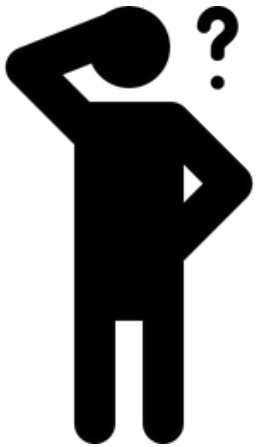
How to select the first node in the tree

Love Math

Love Art

Age

Which one is a Root Node ??



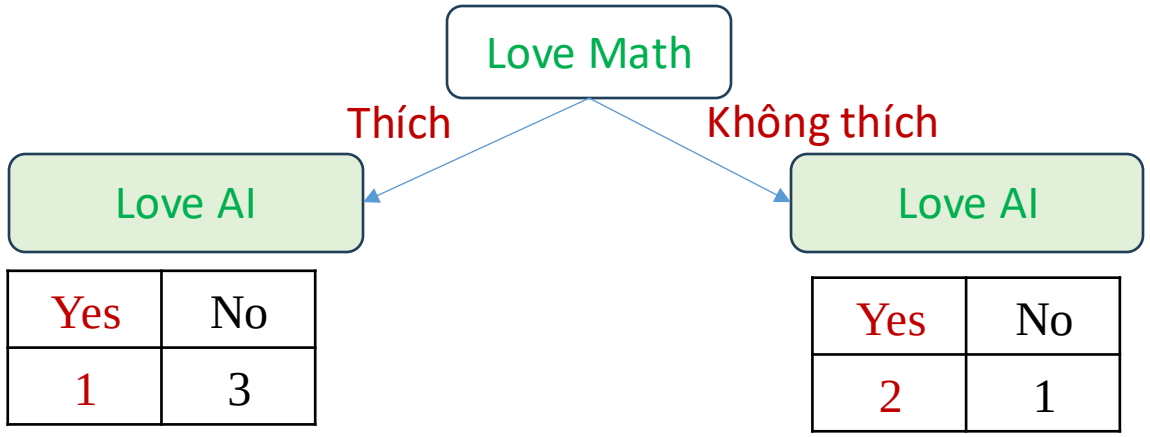
AI VIET NAM
@aivietnam.edu.vn

How to Build Decision Tree

No.	Love Math	Love Art	Age	Love AI
1	Yes	Yes	7	No
2	Yes	No	12	No
3	No	Yes	18	Yes
4	No	Yes	35	Yes
5	Yes	Yes	38	Yes
6	Yes	No	50	No
7	No	No	83	No

How to select the first node in the tree

Vì hiện tại chúng ta chưa biết chọn thông tin nào làm root node. Nên cách đơn giản là giả sử lấy thông tin "love math" làm root node.



Yes	No
1	3

Yes	No
2	1

Để cho công bằng, chúng ta cũng nên thử chọn thêm Love Art hoặc Age làm root node nhé.

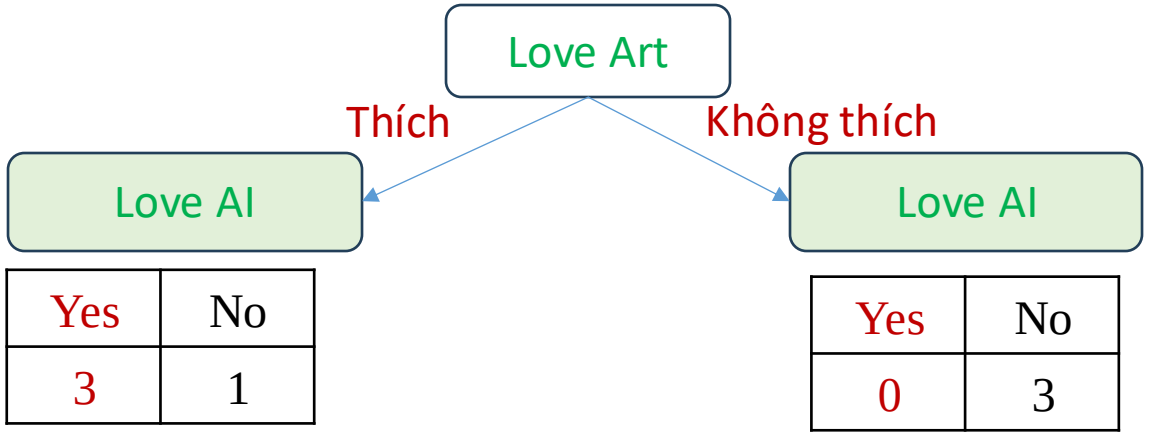
AI VIET NAM
@aivietnam.edu.vn

How to Build Decision Tree

No.	Love Math	Love Art	Age	Love AI
1	Yes	Yes	7	No
2	Yes	No	12	No
3	No	Yes	18	Yes
4	No	Yes	35	Yes
5	Yes	Yes	38	Yes
6	Yes	No	50	No
7	No	No	83	No

How to select the first node in the tree

Chọn Love Art như là root node.

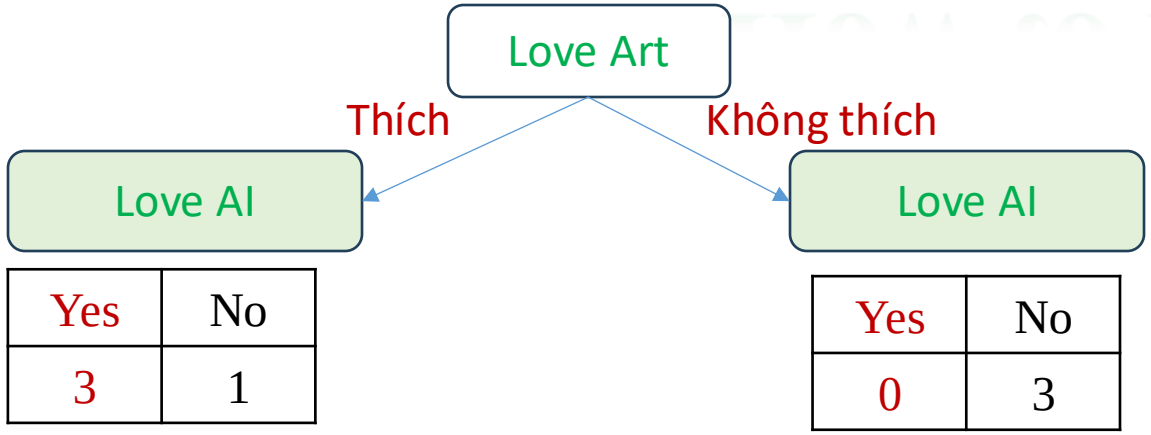


Yes	No
3	1

Yes	No
0	3

AI VIET NAM
@aivietnam.edu.vn

How to Build Decision Tree

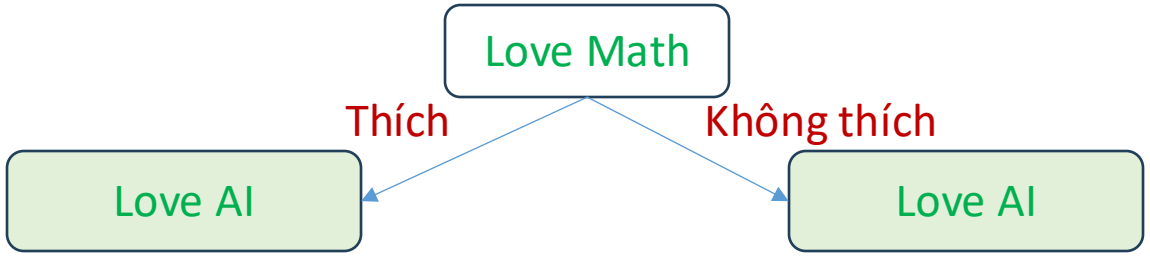


Yes	No
3	1

Yes	No
0	3

Review Two Approaches

Do you have any ***comments*** on these two simple trees.

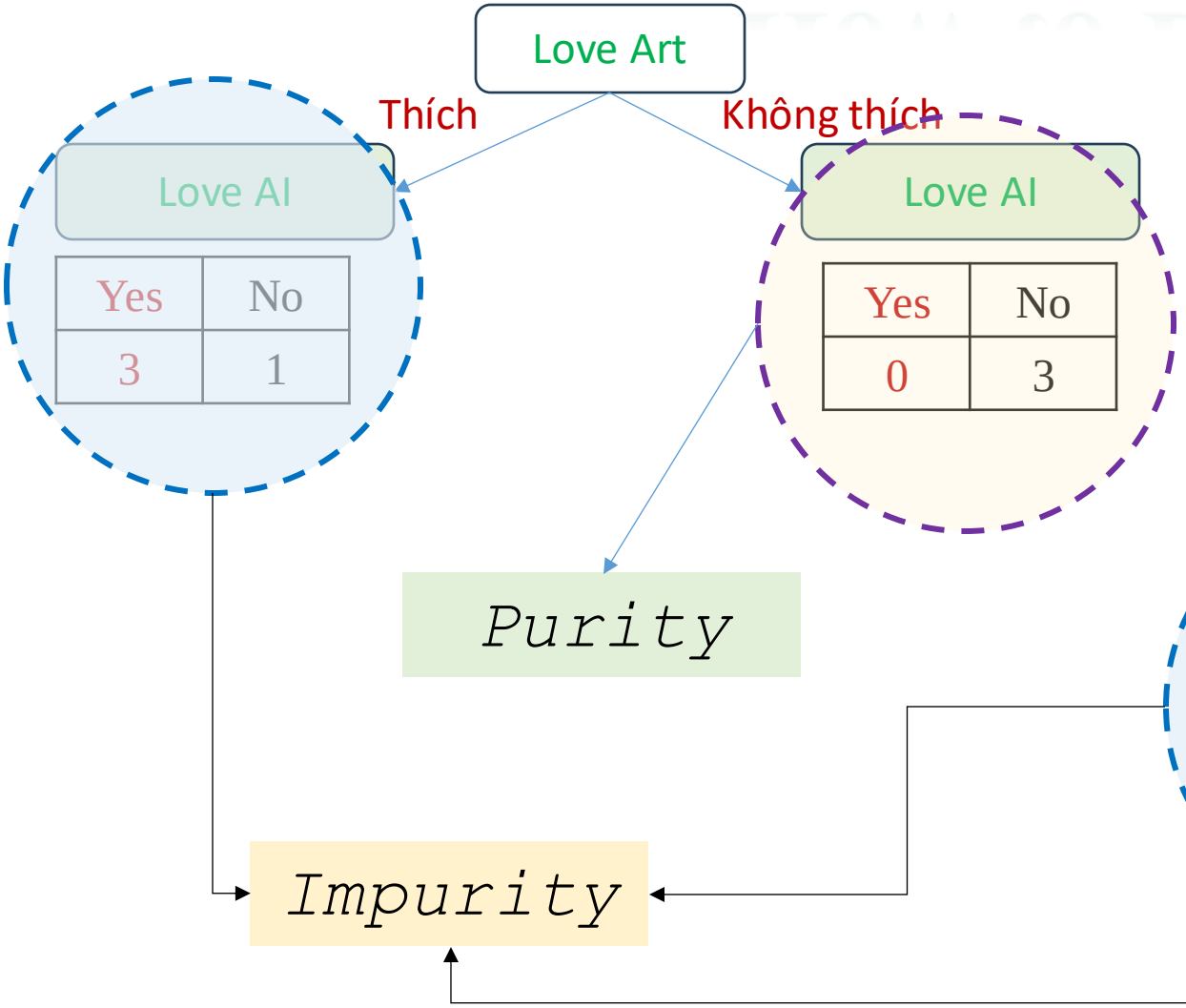


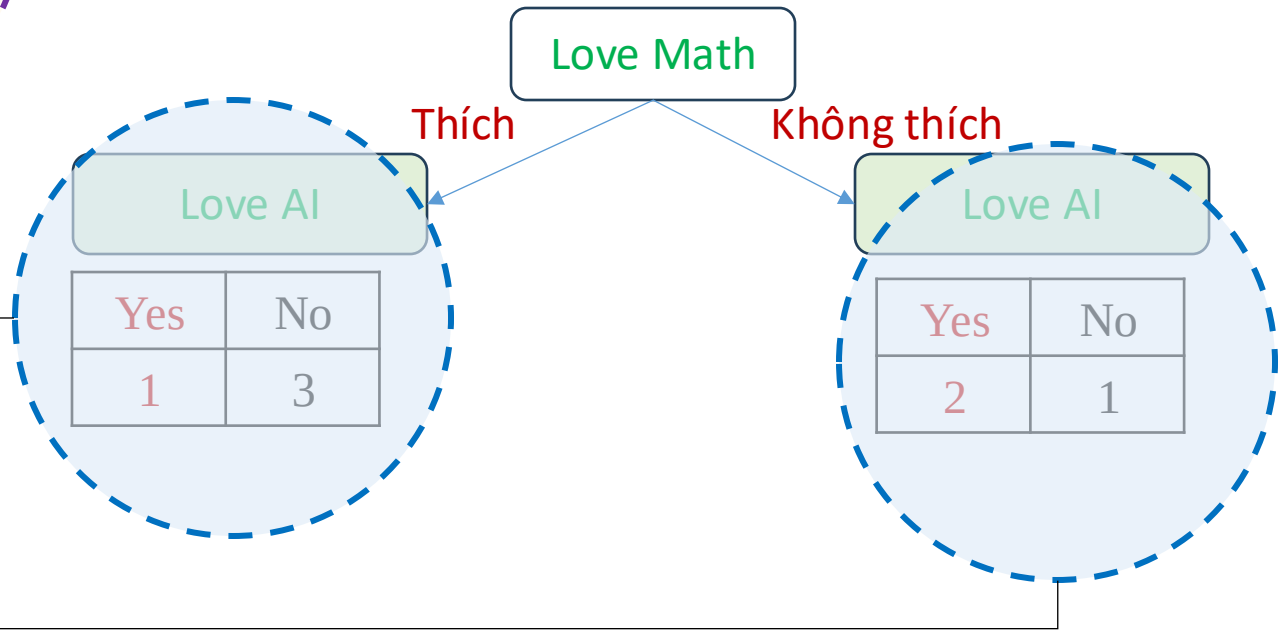
Yes	No
1	3

Yes	No
2	1

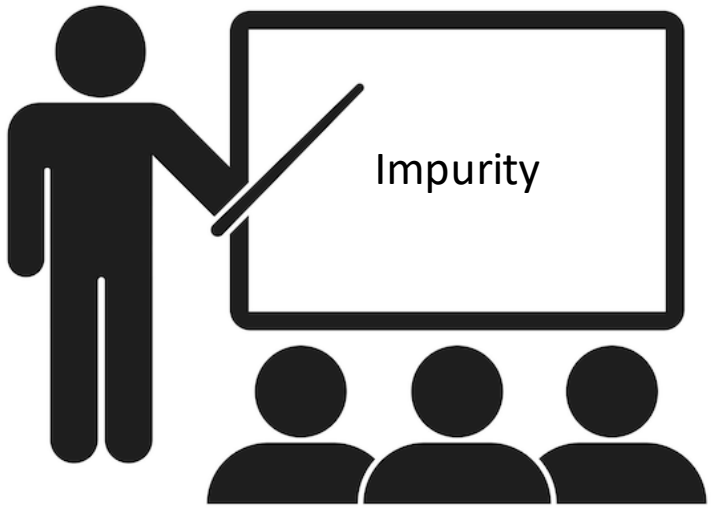
How to Build Decision Tree

Do you have any ***comments*** on these two simple trees.



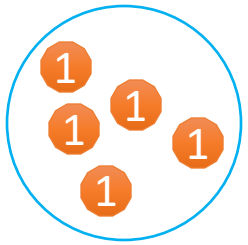


What is an Impurity

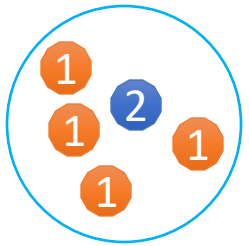


Set 1	1	1	1	1	1
Set 2	1	1	2	1	1
Set 3	1	2	4	6	7

	Impurity Score
Set 1	Thấp
Set 2	Trung Bình
Set 3	Cao



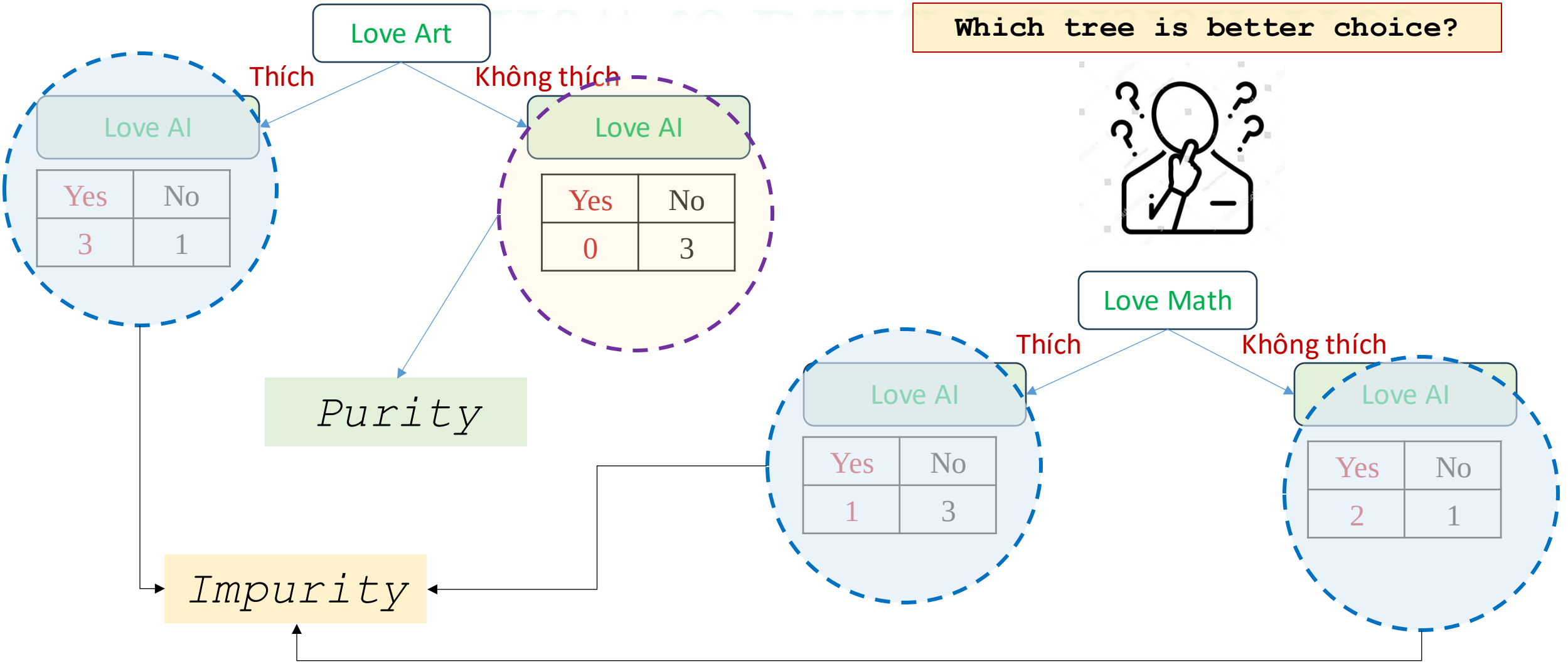
Set 1



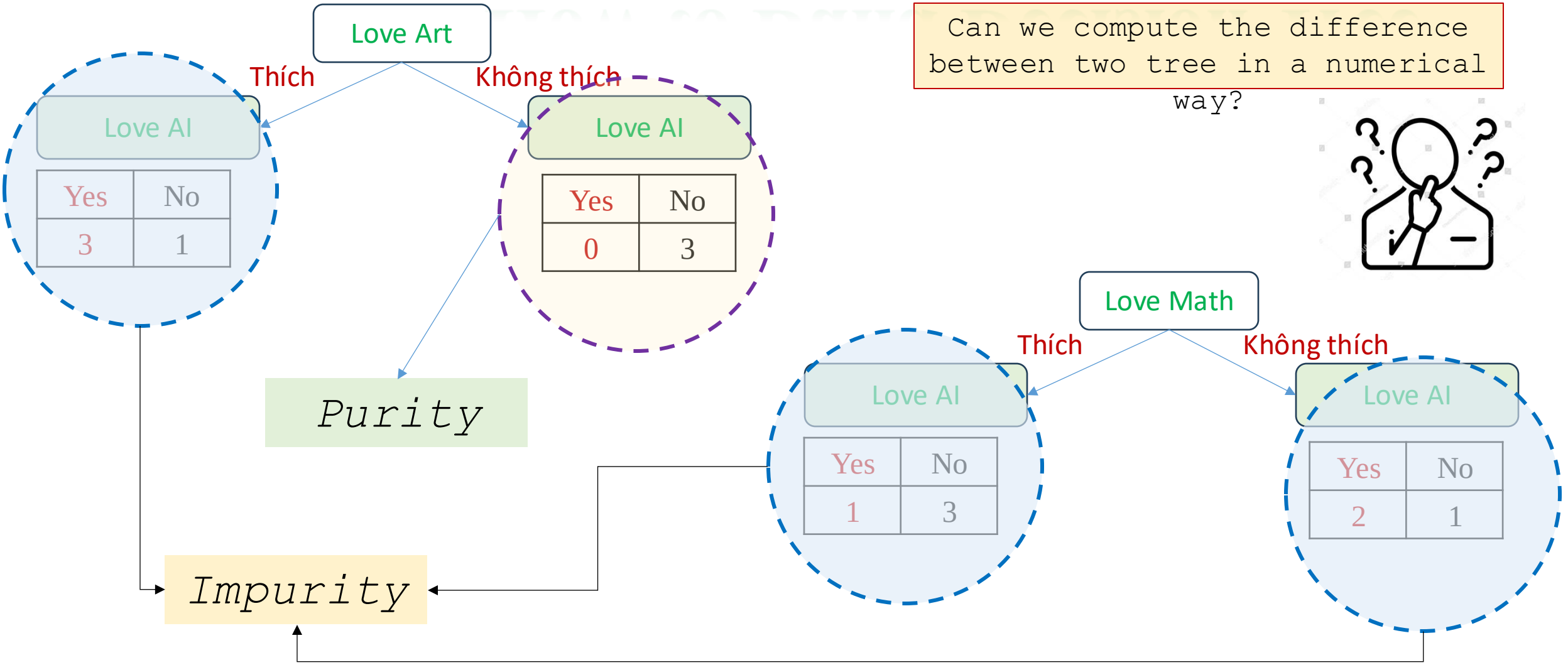
Set 2

Set 3

How to Build Decision Tree



How to Build Decision Tree



Evaluation Metrics

Entropy – Information Gain

GNI IMPURITY

Outline

- Introduction to Decision Tree
- Classification Tree with GINI
- Classification Tree with Entropy
- Examples
- Summary

Evaluation Metrics

Entropy – Information Gain

GNI IMPURITY

GINI Impurity

It was developed by statistician and sociologist Corrado Gini.

Consider a dataset D that contains samples from k classes. The probability of samples belonging to class i at a given node can be denoted as p_i. Then the Gini Impurity of D is defined as:

$$Gini(D) = 1 - \sum_{i=1}^{k} p_i^2$$

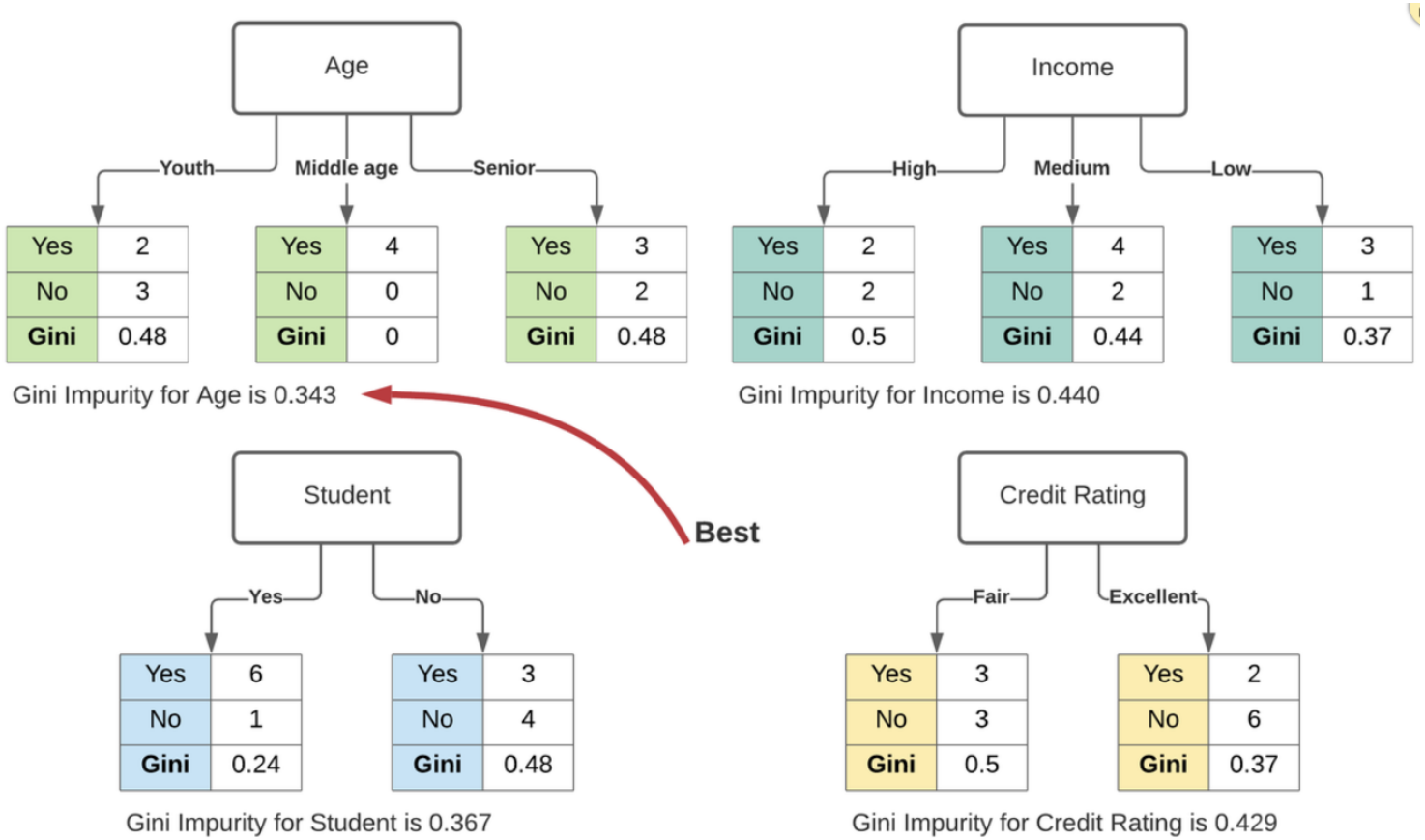
	Count		Probability		Gini Impurity
	n_1	n_2	p_1	p_2	$1 - p_1^2 - p_2^2$
Node A	0	10	0	1	$1 - 0^2 - 1^2 = 0$
Node B	3	7	0.3	0.7	$1 - 0.3^2 - 0.7^2 = 0.42$
Node C	5	5	0.5	0.5	$1 - 0.5^2 - 0.5^2 = 0.5$

AI VIET NAM
@aivietnam.edu.vn

GINI Impurity

It was developed by statistician and sociologist Corrado Gini.

$$Gini(D) = 1 - \sum_{i=1}^{k} p_i^2$$

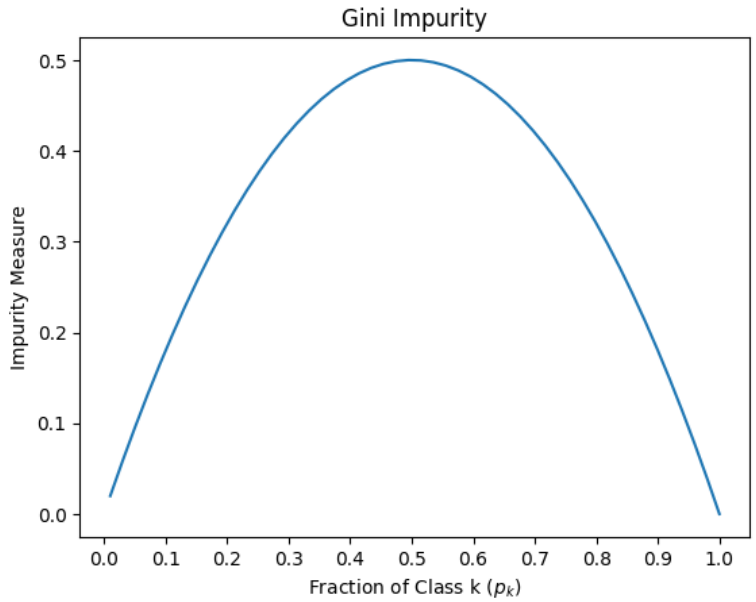


AI VIET NAM
@aivietnam.edu.vn

GINI Impurity

$$Gini(D) = 1 - \sum_{i=1}^{k} p_i^2$$

It was developed by statistician and sociologist Corrado Gini.




```
#A figure is created to show Gini ımpurity measures
plt.figure()
x = np.linspace(0.01,1)
y = 1 - (x*x) - (1-x)*(1-x)
plt.plot(x,y)
plt.title('Gini Impurity')
plt.xlabel("Fraction of Class k ($p_k$)")
plt.ylabel("Impurity Measure")
plt.xticks(np.arange(0,1.1,0.1))

plt.show()
```

This figure shows that Gini impurity is maximum for the 50-50 sample (p1=0.5) and minimum for the homogeneous sample (p1=0 or p1=1)

GINI Impurity

Set 1	1	1	1	1	1
Set 2	1	1	2	1	1
Set 3	1	2	4	6	7

	No of Unique Element	Count of unique elements	Probability	GNI Impurity
Set 1	1	5	5/5	0
Set 2	1,2	4, 1	4/5, 1/5	0.32
Set 3	1,2,4,6,7	1,1,1,1,1	1/5,1/5,1/5,1/5,1/5,1/5	0.8

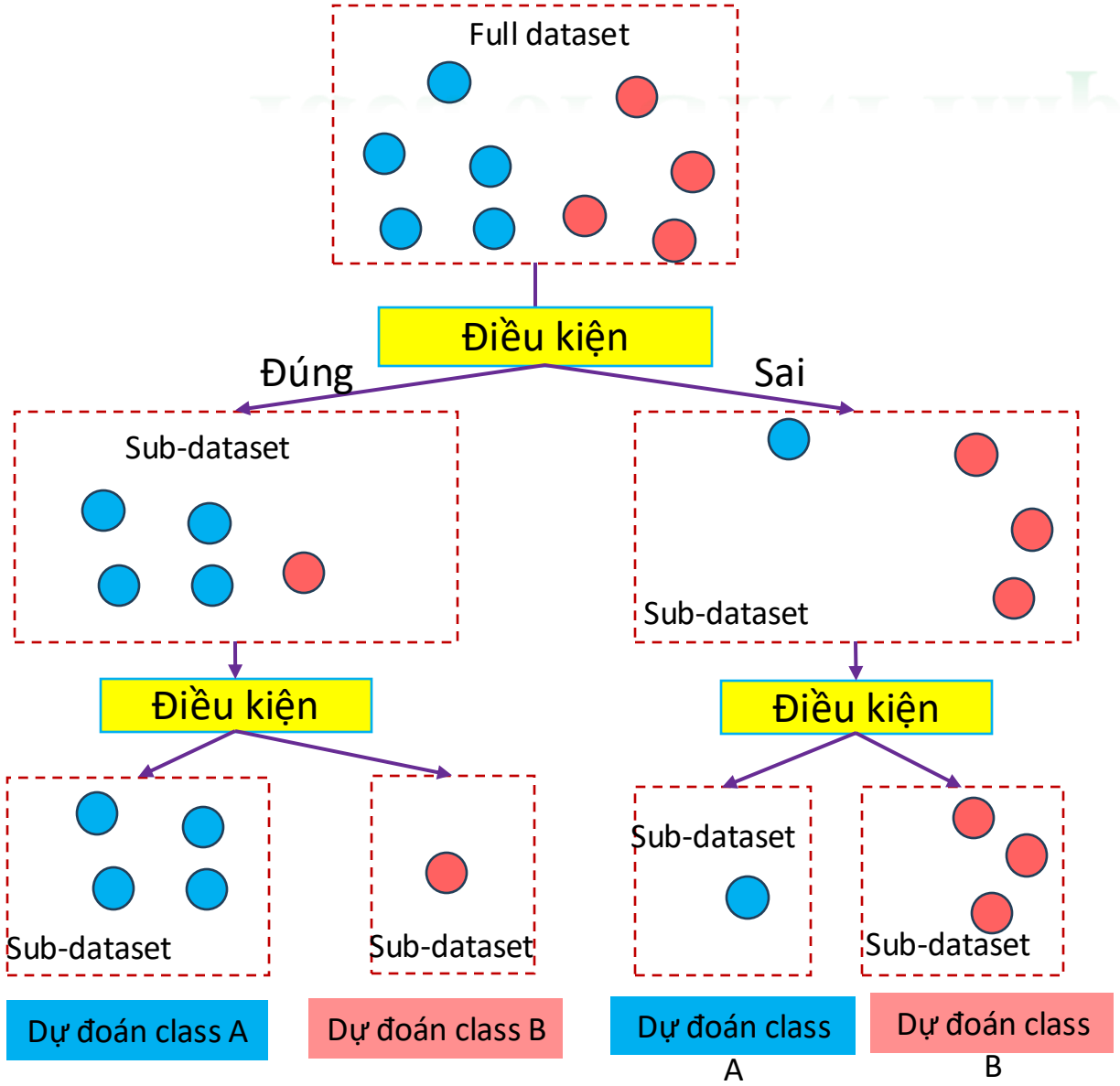
$$Gini(D) = 1 - \sum_{i=1}^{k} p_i^2$$

Idea of GINI Impurity

GINI cao,
Khó đưa ra quyết định

GINI vừa,
Đưa ra quyết định

GINI nhỏ,
Tự tin đưa ra quyết định



GINI = ???

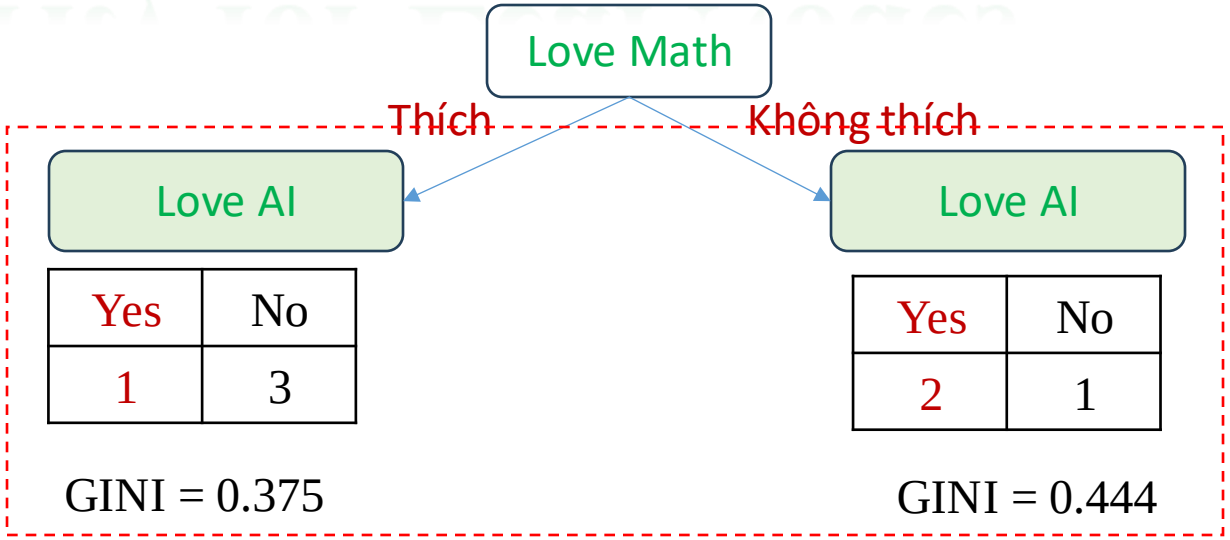
GINI = ???

GINI = ???

Ý tưởng: xây dựng Tree để làm cho GINI giảm dần

Gini Impurity for Leaf Nodes

$$Gini = 1 - \sum_{i=1}^{n} P_i^2$$

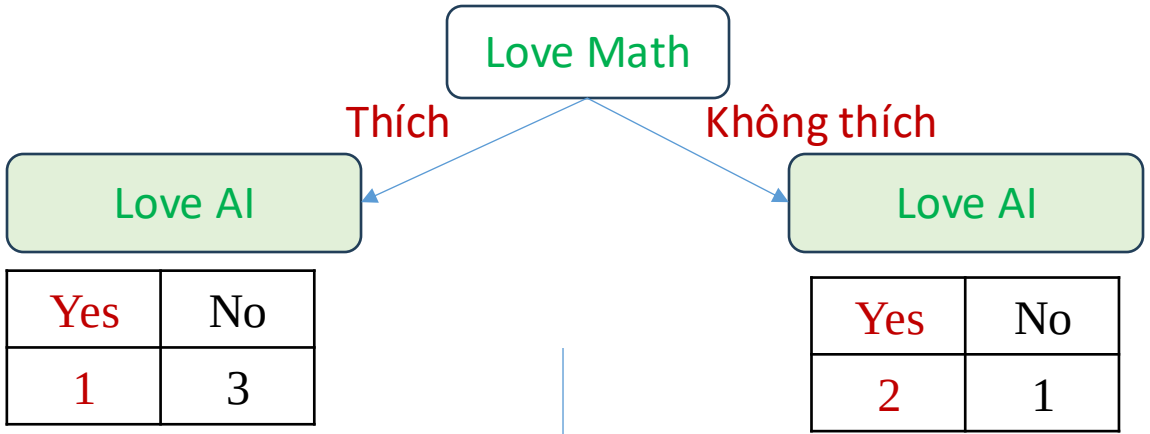


Cách tính Gini Impurity cho Leaf Nodes

1. Calculate the probabilties of all classess.
2. Square the calculated probabilities
3. Sum all the squared probabilities into a single integer
4. Substract the single integer from 1

Gini Impurity for Leaf Nodes

$$Gini = 1 - \sum_{i=1}^{n} P_i^2$$



Yes	No
1	3

GINI = 0.375

Yes	No
2	1

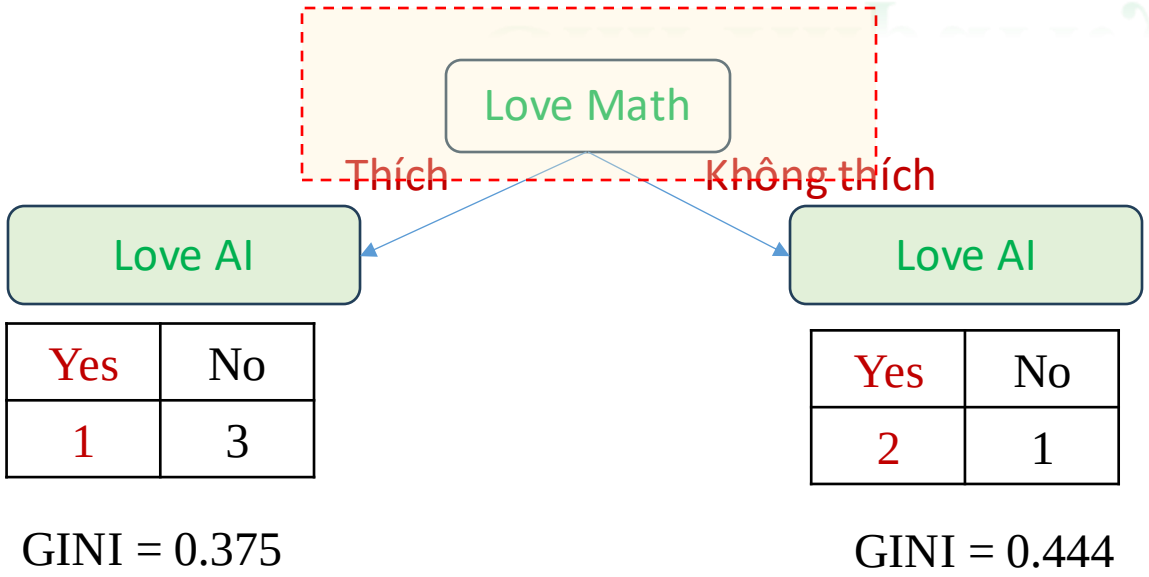
GINI = 0.444

Gini = 1 – (the probability of Yes)^2 – (the probability of No)^2

GINI = $1 - [(1/4)^2 + (3/4)^2] = 0.375$

GINI = $1 - [(2/3)^2 + (1/3)^2] = 0.444$

Gini Impurity for Roof/Internal Nodes



Yes	No
1	3

GINI = 0.375

Yes	No
2	1

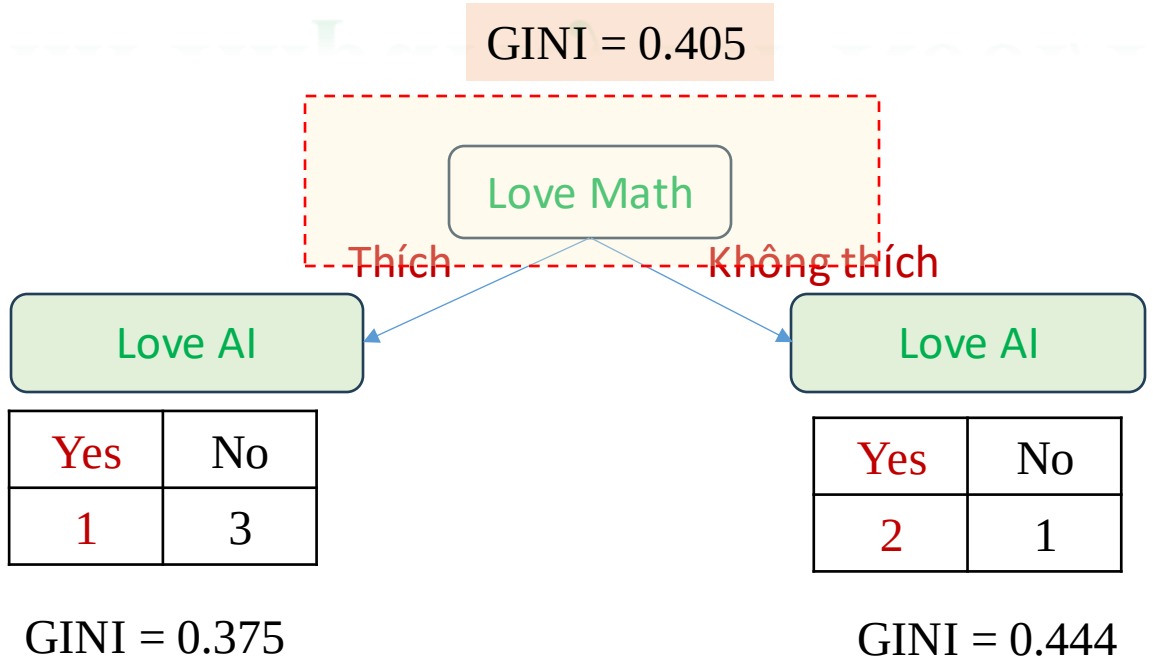
GINI = 0.444

Cách tính Gini Impurity cho Root/Internal Nodes

1. Loop over the leaf node, from k = 1 all the way to the K leaf nodes
2. Find the gini impurity of the urrent k'th leaf
3. Count the number of observations in the k'th leaf
4. Divide by the total number of observation in the all leaf nodes
5. The result of each leaf is summed for a final gini impurity

$$Gini(Internal_j) = \sum_{k'th\ leaf=1}^{K\ leaf\ nodes} \left(\frac{count(L_k)}{count(L_1, \ldots, L_k)} \right) Gini(L_k)$$

Gini Impurity for Roof/Internal Nodes



Total Impurity = Weight average of Gini Impurities for The leaves

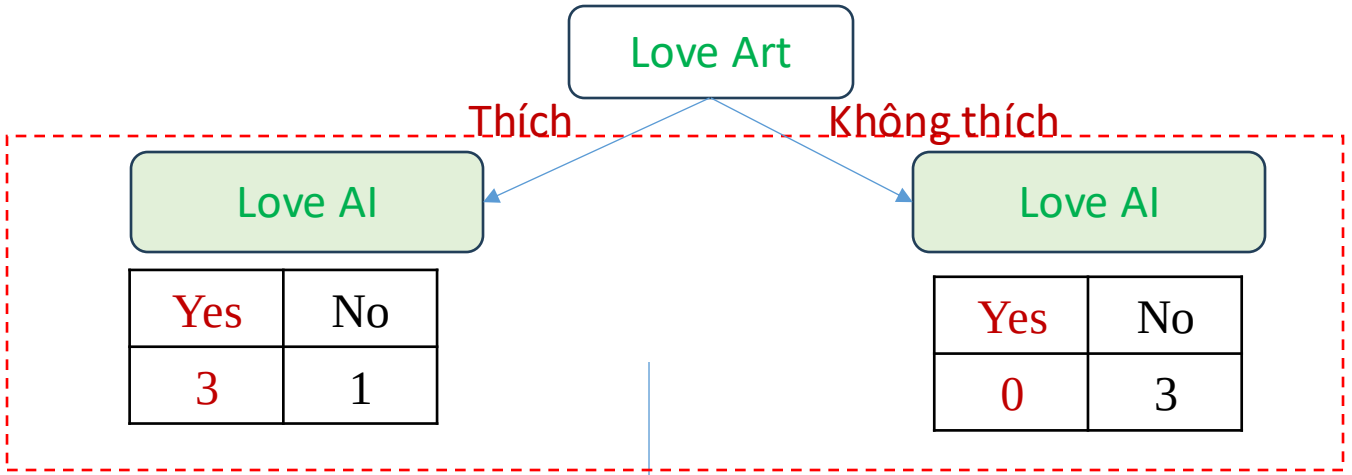
Total Impurity = $\frac{4}{7}$ ×0.375 + $\frac{3}{7}$ ×0.444 = 0.405

If a data set D is split on an attribute A into two subsets D_1 and D_2 with sizes n_1 and n_2, respectively, the Gini Impurity can be defined as:

$$Gini_A(D) = \frac{n_1}{n} Gini(D_1) + \frac{n_2}{n} Gini(D_2)$$

Gini Impurity for Leaf Nodes

$$GINI = 1 - \sum_{i=1}^{n} (p_i)^2$$



GINI = 0.375

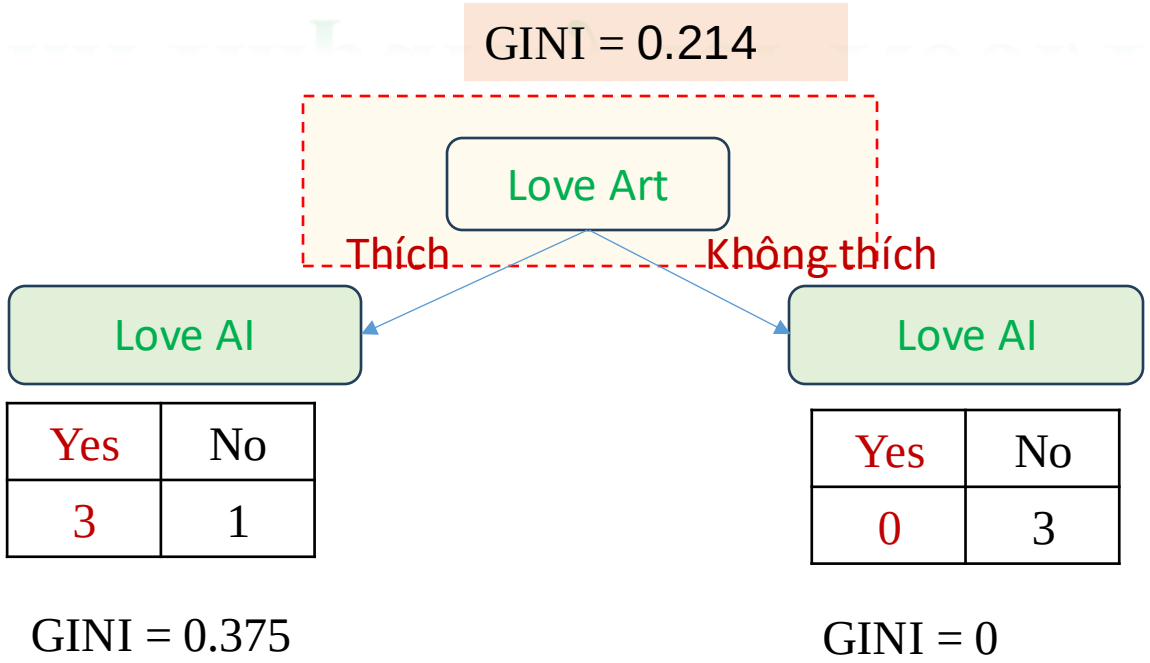
GINI = 0

Gnini = 1 – (the probability of Yes)^2 – (the probability of No)^2

GINI = $1 - [(3/4)^2 + (1/4)^2] = 0.375$

GINI = $1 - [(3/3)^2] = 0$

Gini Impurity for Roof/Internal Nodes



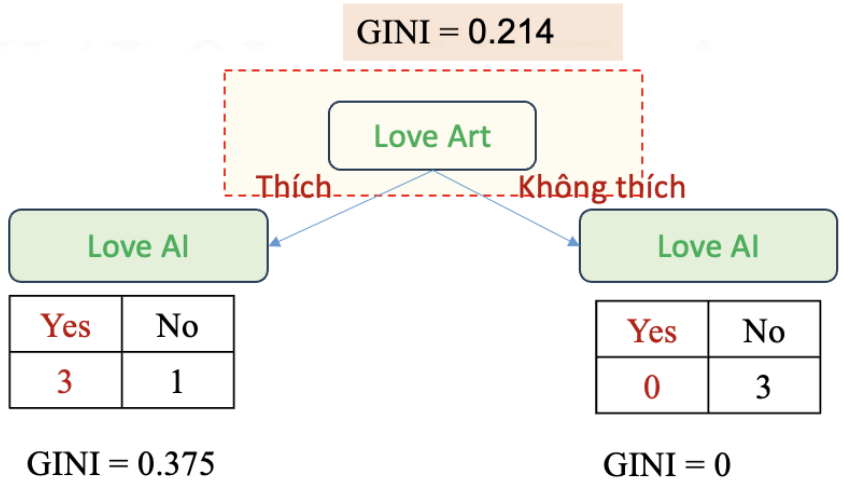
Total Impurity = Weight average of Gini Impurities for The leaves

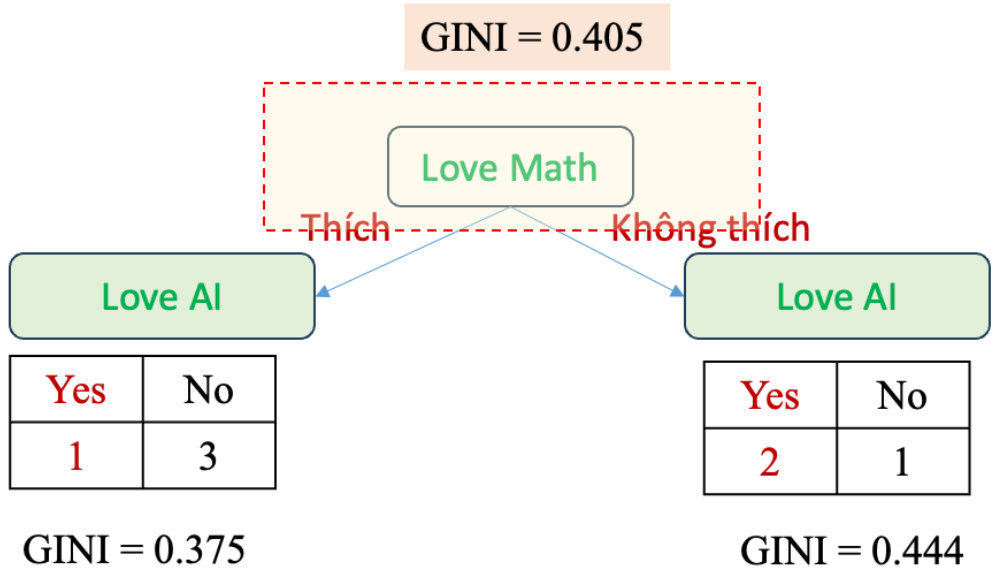
Total Impurity = $\frac{4}{7} \times 0.375 + \frac{3}{7} \times 0 = 0.214$

AI VIET NAM
@aivietnam.edu.vn

What is a GINI of "AGE"?

No.	Love Math	Love Art	Age	Love AI
1	Yes	Yes	7	No
2	Yes	No	12	No
3	No	Yes	18	Yes
4	No	Yes	35	Yes
		Yes	38	Yes
		No	50	No
		No	83	No





What is a GINI of "AGE"?

❖ Sort By Age

Before

Age	Love AI
7	No
12	No
18	Yes
35	Yes
38	Yes
50	No
83	No

Sort by Age →

After

Age	Love AI
7	No
12	No
18	Yes
35	Yes
38	Yes
50	No
83	No

Average of two adjacent cells →

(7 + 12)/2 = 9.5

(18 + 12)/2 = 15

(18 + 35)/2 = 26.5

(38 + 35)/2 = 36.5

(38 + 50)/2 = 44

(50 + 83)/2 = 66.5

Age	Love AI
7	No
12	No
18	Yes
35	Yes
38	Yes
50	No
83	No

What is a GINI of "AGE"?

❖ GINI IMPURITY FOR EACH OF AGE

Age	Love AI
7	No
12	No
18	Yes
35	Yes
38	Yes
50	No
83	No

(7 + 12)/2 = 9.5

(18 + 12)/2 = 15

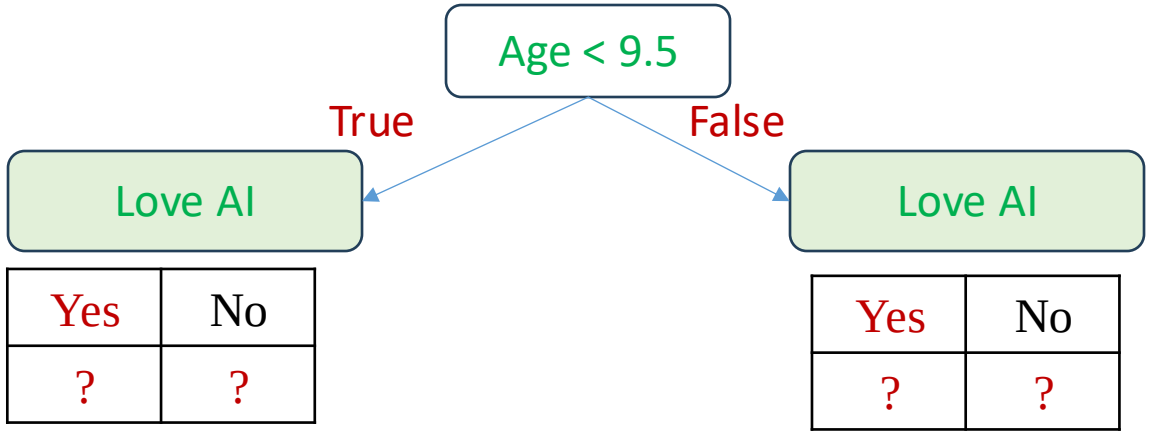
(18 + 35)/2 = 26.5

(38 + 35)/2 = 36.5

(38 + 50)/2 = 44

(50 + 83)/2 = 66.5

GINI = ???



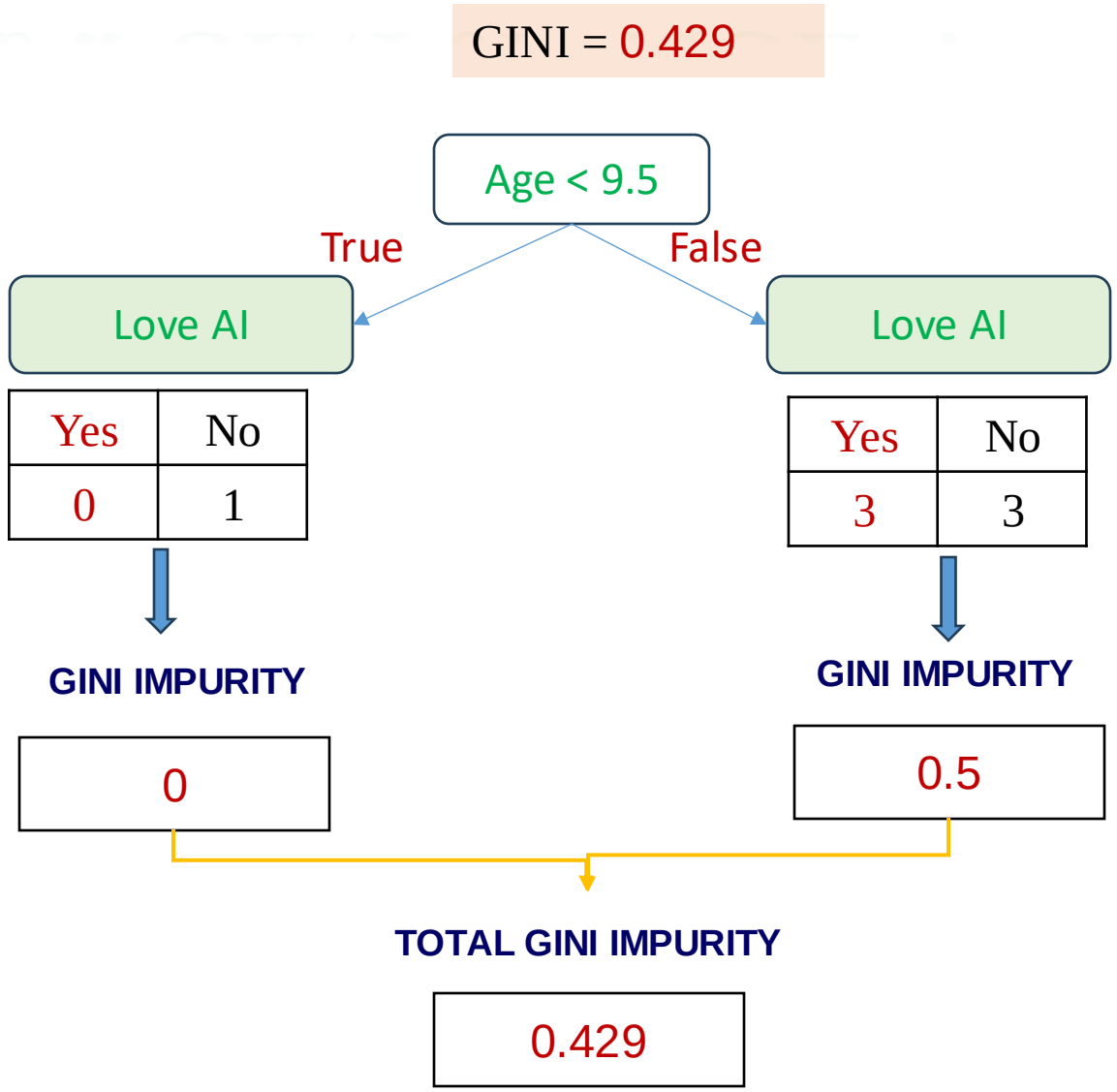
AI VIET NAM
@aivietnam.edu.vn

What is a GINI of "AGE"?

❖ GINI IMPURITY FOR EACH OF AGE

Age	Love AI
7	No
12	No
18	Yes
35	Yes
38	Yes
50	No
83	No

- (7 + 12)/2 = 9.5
- (18 + 12)/2 = 15
- (18 + 35)/2 = 26.5
- (38 + 35)/2 = 36.5
- (38 + 50)/2 = 44
- (50 + 83)/2 = 66.5



AI VIET NAM
@aivietnam.edu.vn

What is a GINI of "AGE"?

❖ GINI IMPURITY FOR EACH OF AGE

Age	Love AI
7	No
12	No
18	Yes
35	Yes
38	Yes
50	No
83	No

(7 + 12)/2 = 9.5

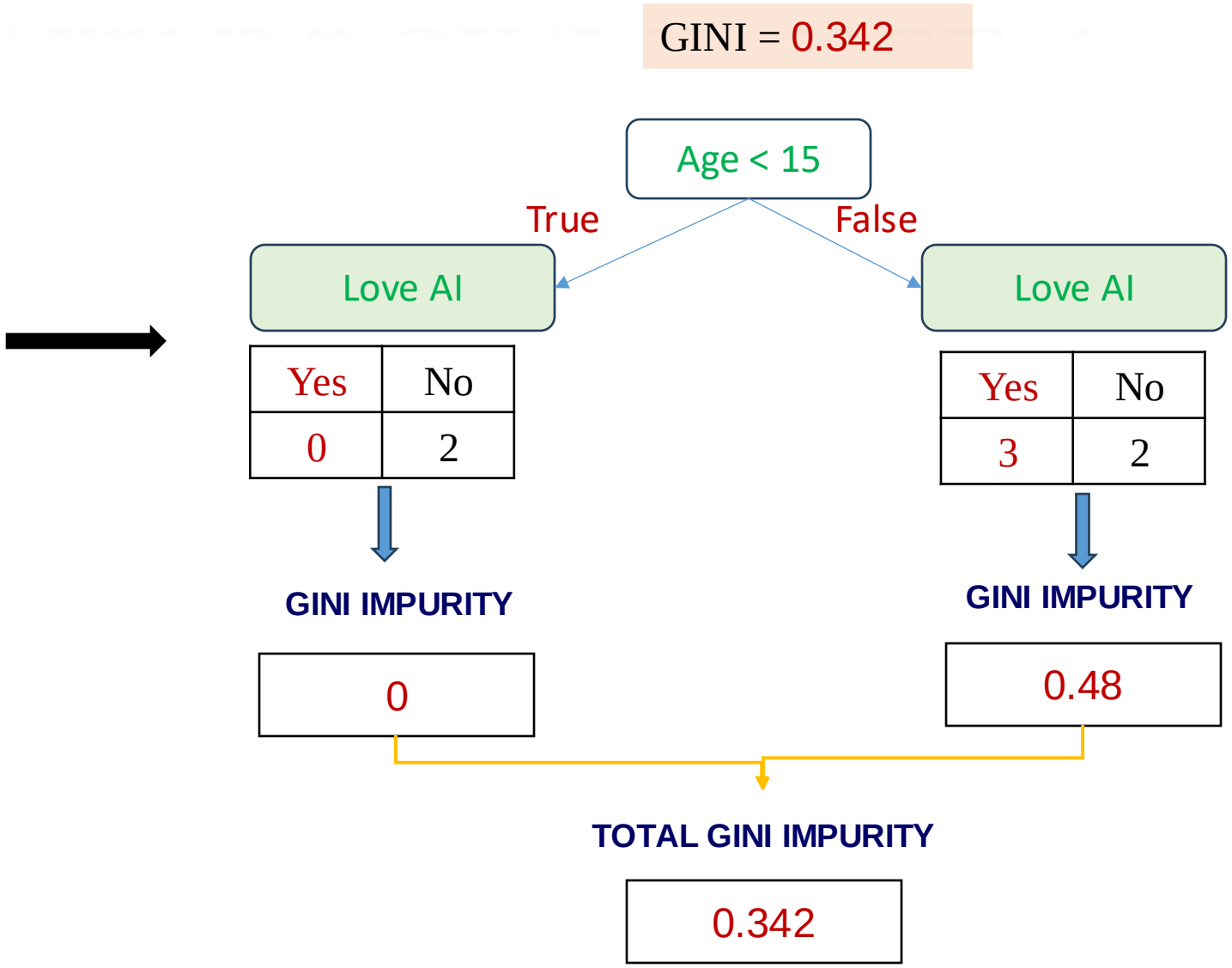
(18 + 12)/2 = 15

(18 + 35)/2 = 26.5

(38 + 35)/2 = 36.5

(38 + 50)/2 = 44

(50 + 83)/2 = 66.5



What is a GINI of "AGE"?

❖ GINI IMPURITY FOR EACH OF AGE

Age	Love AI
7	No
12	No
18	Yes
35	Yes
38	Yes
50	No
83	No

	GINI	GINI
(7 + 12)/2 = 9.5	0.429	0.429
(18 + 12)/2 = 15	0.343	0.343
(18 + 35)/2 = 26.5	0.476	0.476
(38 + 35)/2 = 36.5	0.476	0.476
(38 + 50)/2 = 44	0.343	0.343
(50 + 83)/2 = 66.5	0.429	0.429

GINI for each threshold

Best GINI

AI VIET NAM
@aivietnam.edu.vn

GINI for Three Attributes

GINI = 0.342

Age < 15

True | False

Love AI

Yes	No
0	2

GINI IMPURITY

0

Love AI

Yes	No
3	2

GINI IMPURITY

0.48

TOTAL GINI IMPURITY

0.342

Which attribute is in the first node?

GINI = 0.214

Love Art

Thích | Không thích

Love AI

Yes	No
3	1

GINI = 0.375

Love AI

Yes	No
0	3

GINI = 0

GINI = 0.405

Love Math

Thích | Không thích

Love AI

Yes	No
1	3

GINI = 0.375

Love AI

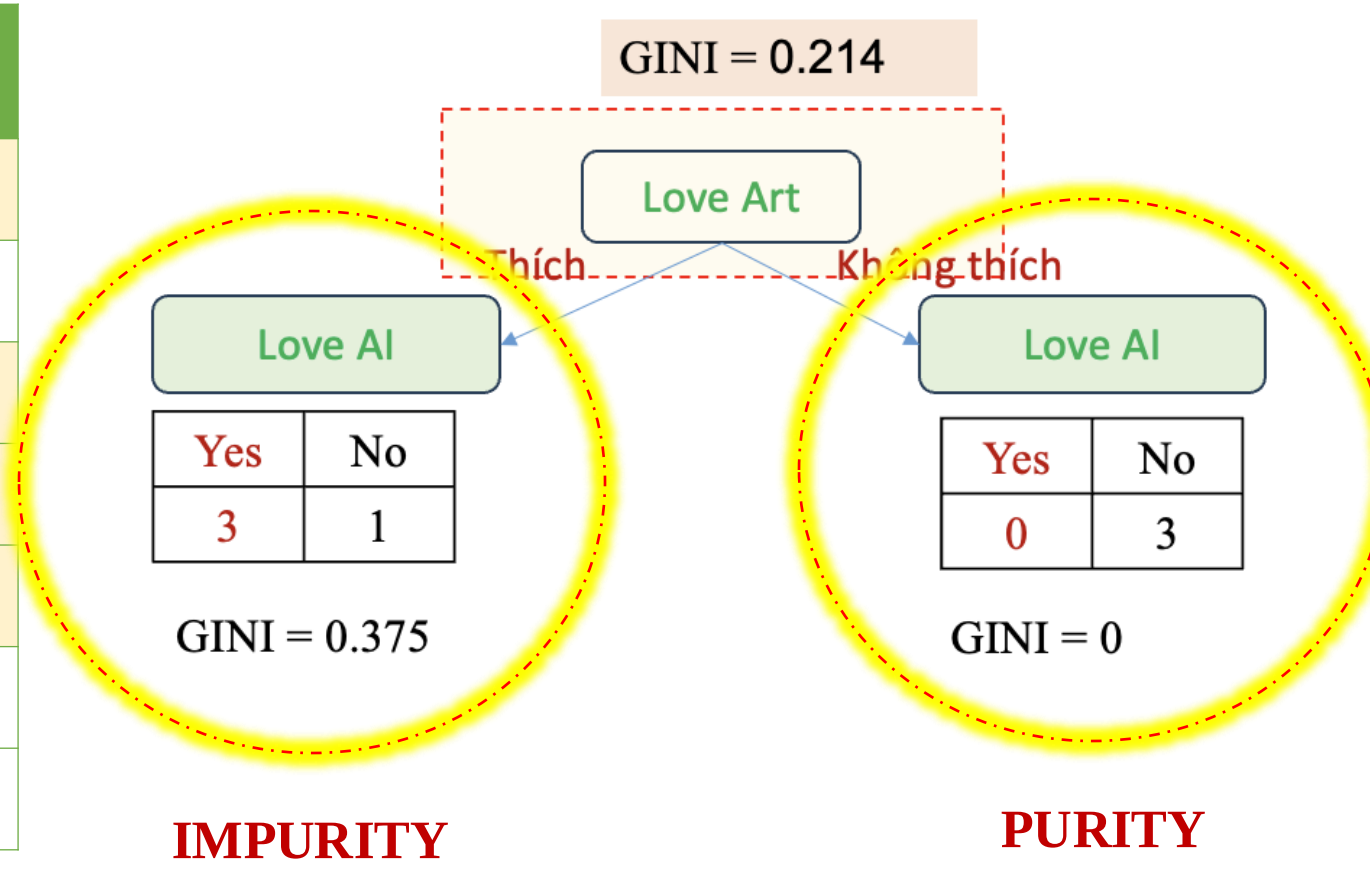
Yes	No
2	1

GINI = 0.444

Love Art is the best one

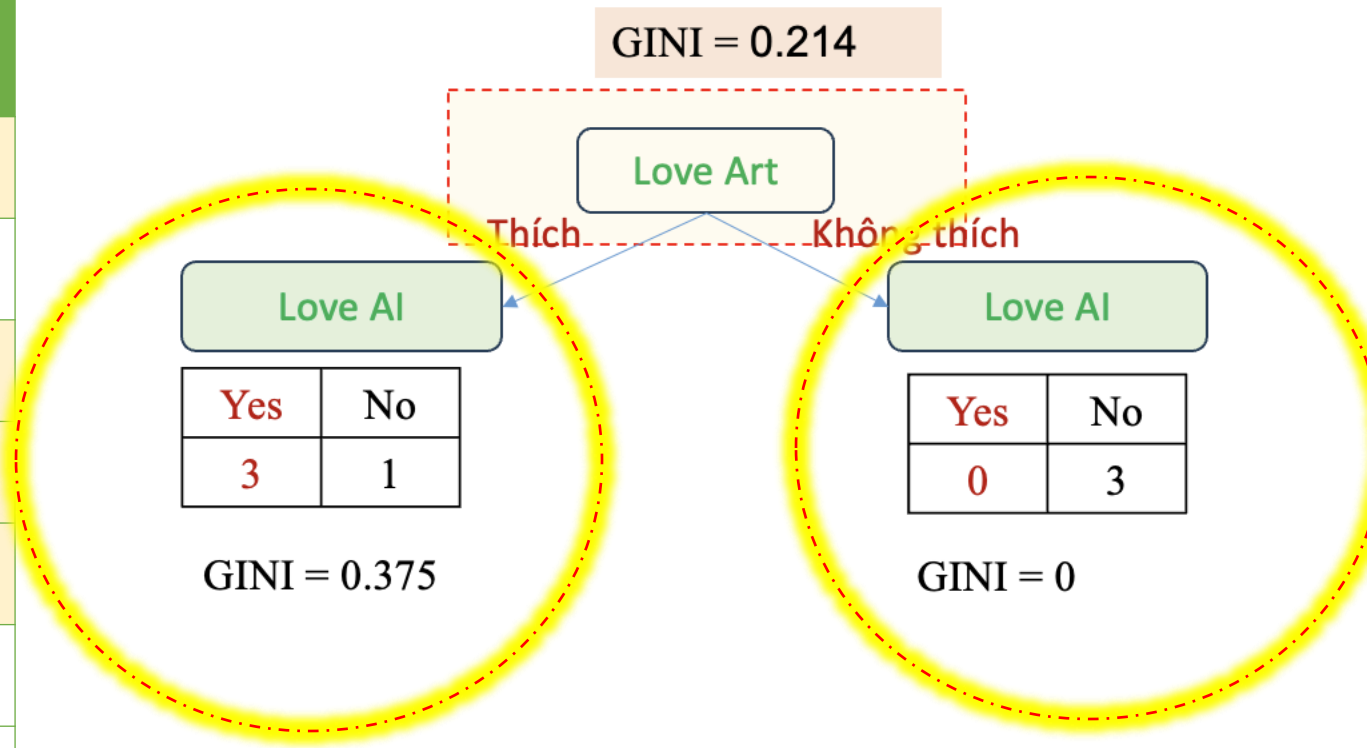
First Node in The Tree

No.	Love Math	Love Art	Age	Love AI
1	Yes	Yes	7	No
2	Yes	No	12	No
3	No	Yes	18	Yes
4	No	Yes	35	Yes
5	Yes	Yes	38	Yes
6	Yes	No	50	No
7	No	No	83	No



First Node in The Tree

No.	Love Math	Love Art	Age	Love AI
1	Yes	Yes	7	No
2	Yes	No	12	No
3	No	Yes	18	Yes
4	No	Yes	35	Yes
5	Yes	Yes	38	Yes
6	Yes	No	50	No
7	No	No	83	No

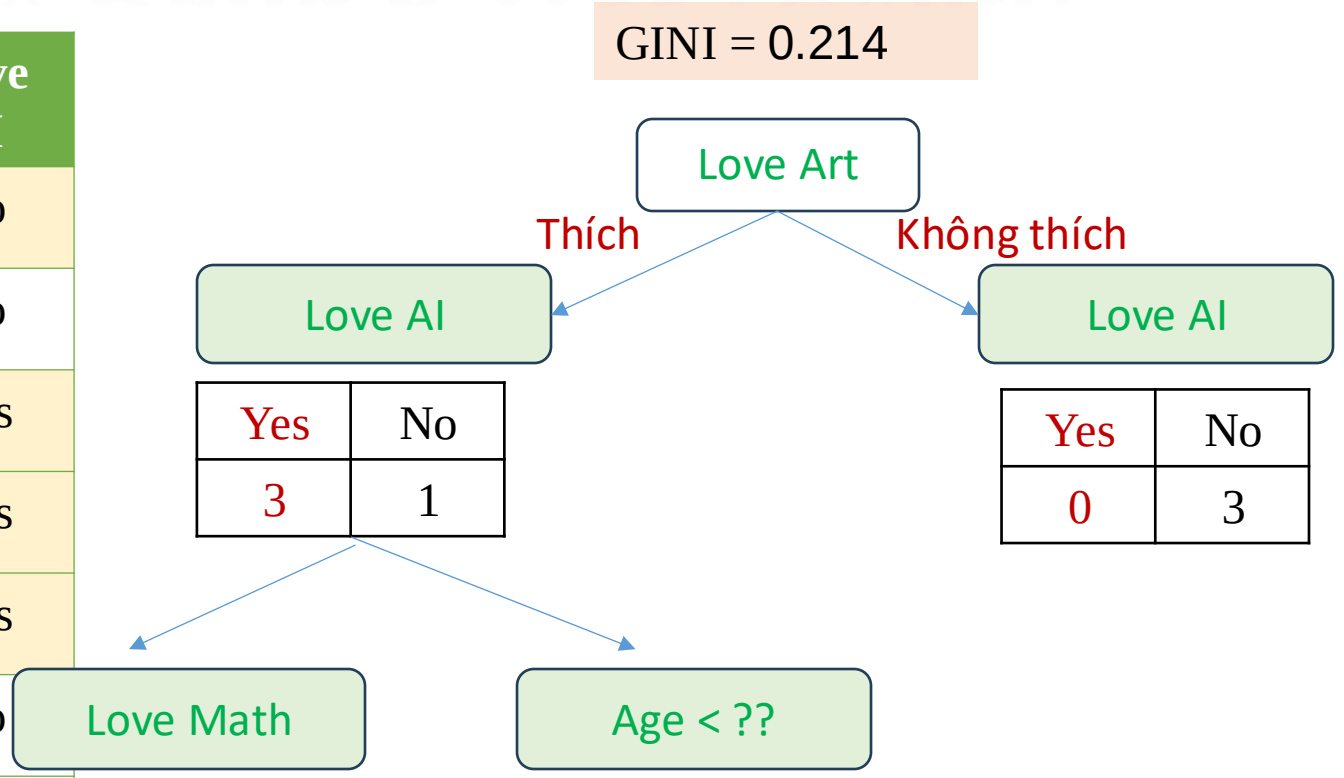


Not easy to make decision

Easy to make decision

How to Build a Tree (Cont.)

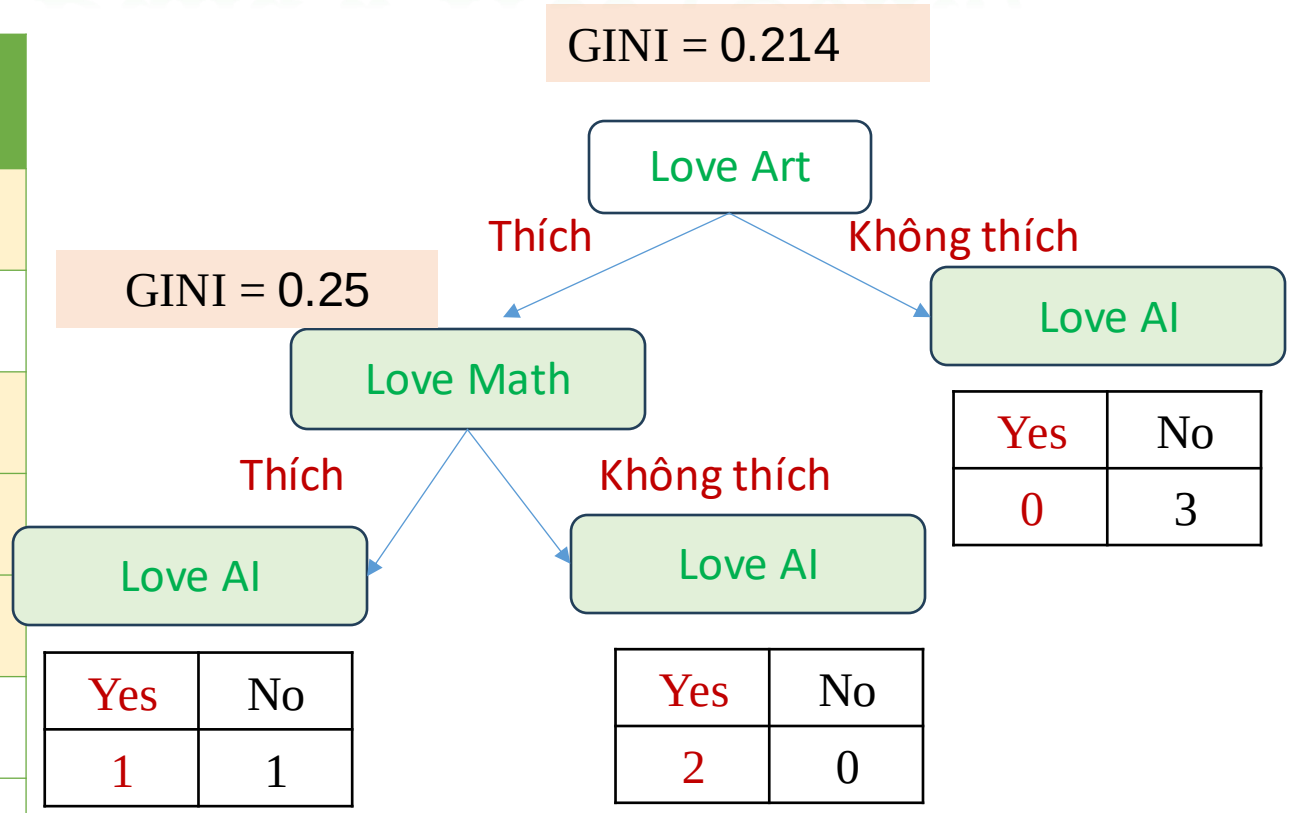
No.	Love Math	Love Art	Age	Love AI
1	Yes	Yes	7	No
2	Yes	No	12	No
3	No	Yes	18	Yes
4	No	Yes	35	Yes
5	Yes	Yes	38	Yes
6	Yes	No	50	No
7	No	No	83	No



WHICH ONE IS THE NEXT INTERNODE? Love Math or Age

How to Build a Tree (Cont.)

No.	Love Math	Love Art	Age	Love AI
1	Yes	Yes	7	No
2	Yes	No	12	No
3	No	Yes	18	Yes
4	No	Yes	35	Yes
5	Yes	Yes	38	Yes
6	Yes	No	50	No
7	No	No	83	No



Giả sử chúng ta chọn "Love Math" là node kế tiếp, cần tính GINI trong trường hợp này

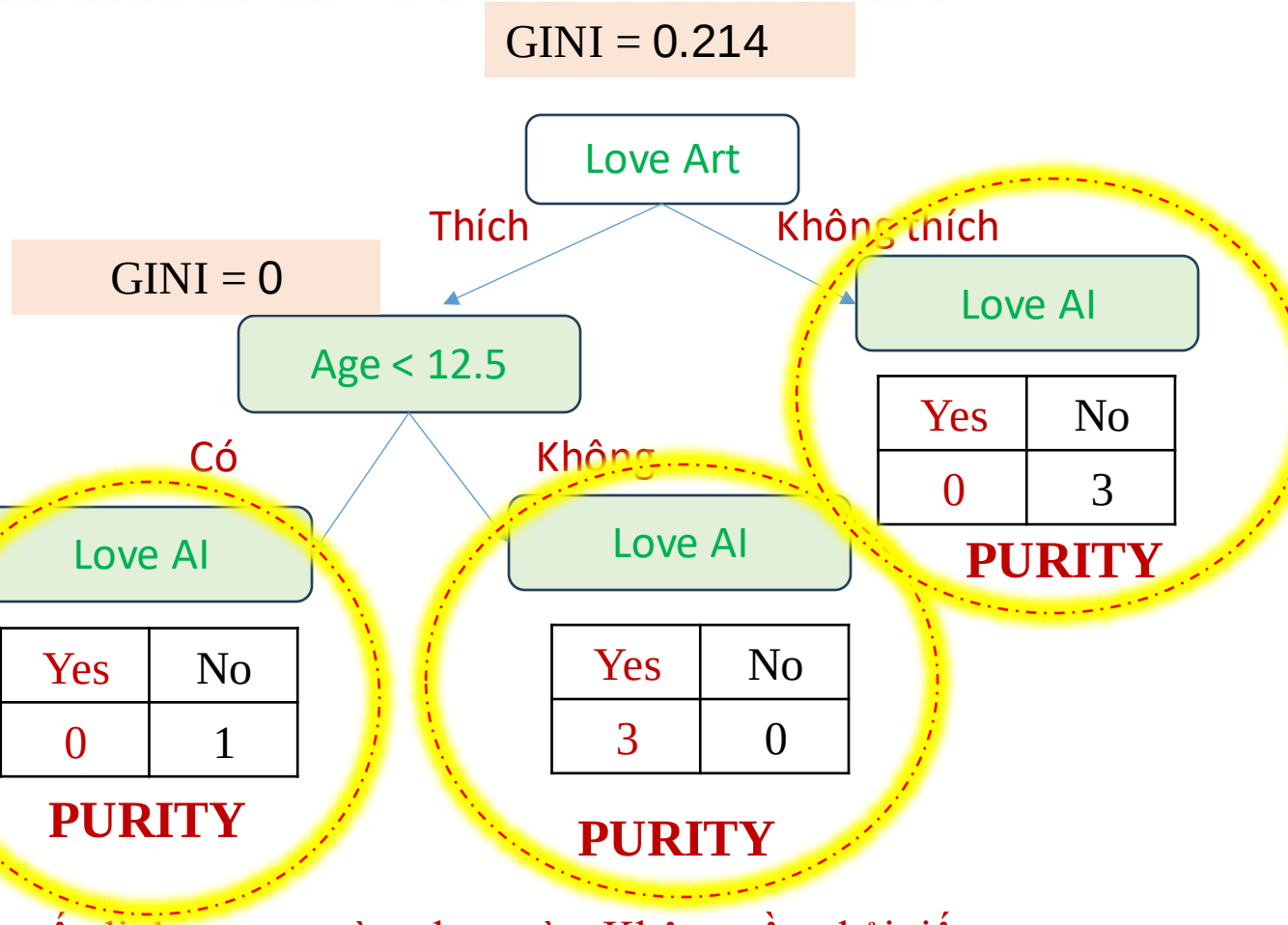
How to Build a Tree (Cont.)

Love Math	Love Art	Age	Love AI
Yes	Yes	7	No
No	Yes	18	Yes
No	Yes	35	Yes
Yes	Yes	38	Yes

12.5

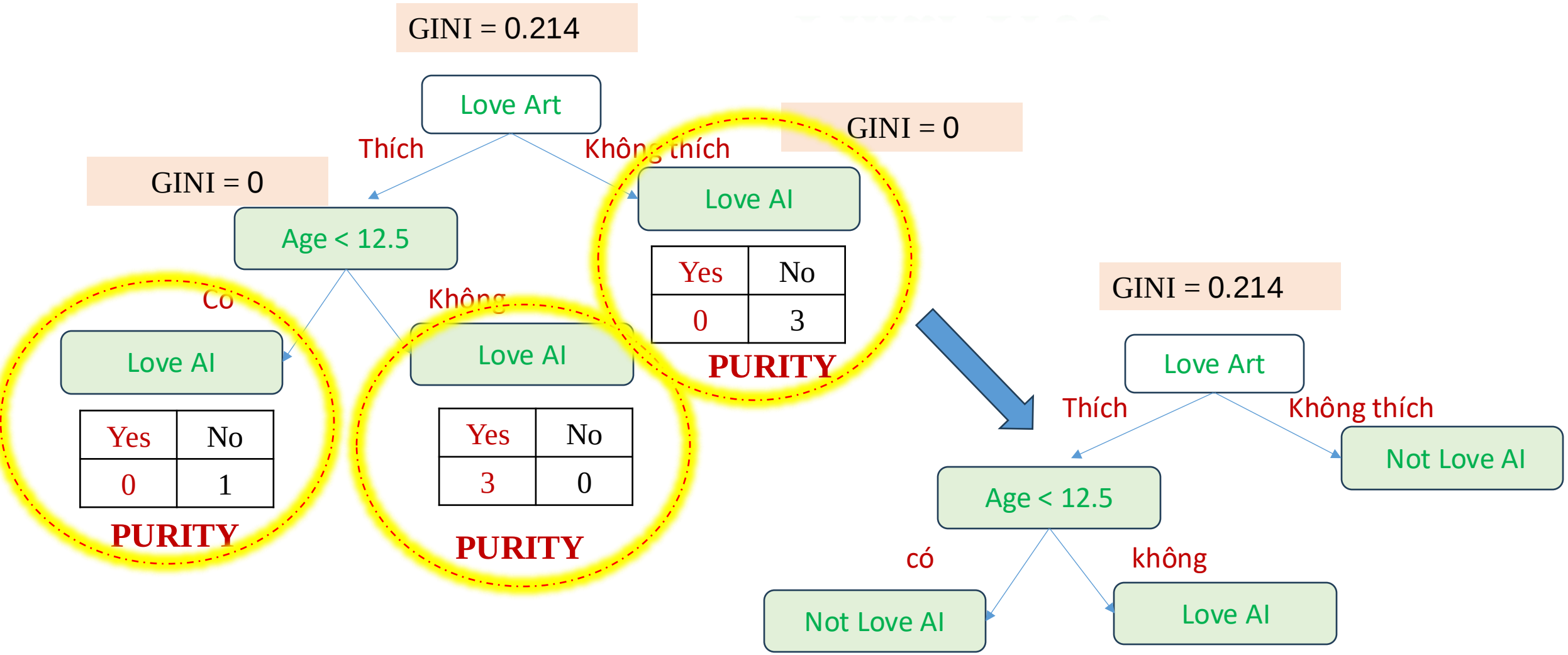
26.5

36.5

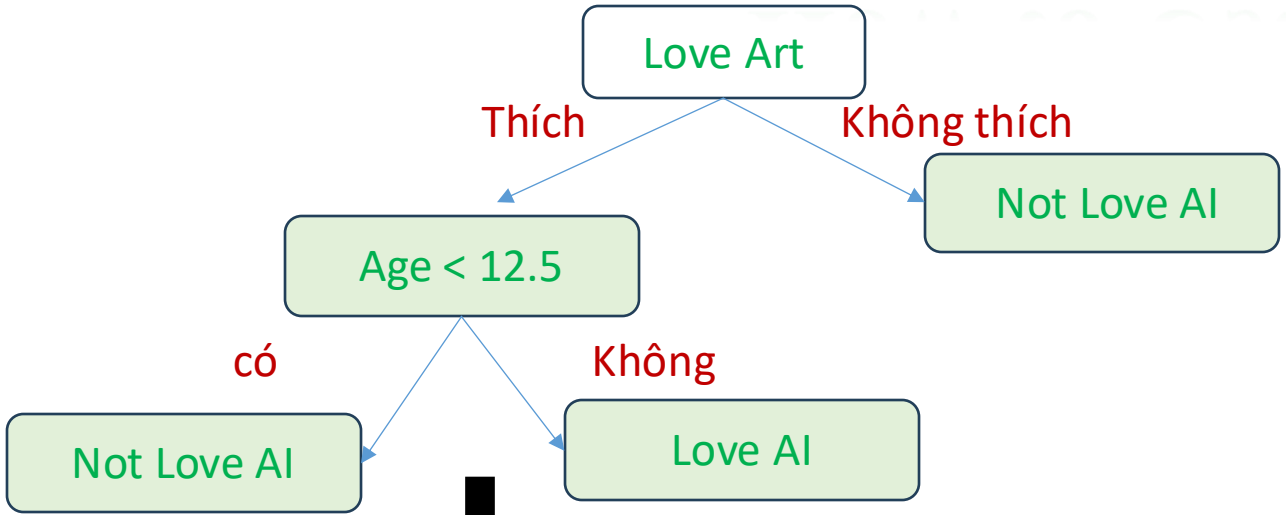


Dễ dàng đưa ra quyết định trong trường hợp này. Không cần phải tiếp tục xây dựng Tree.

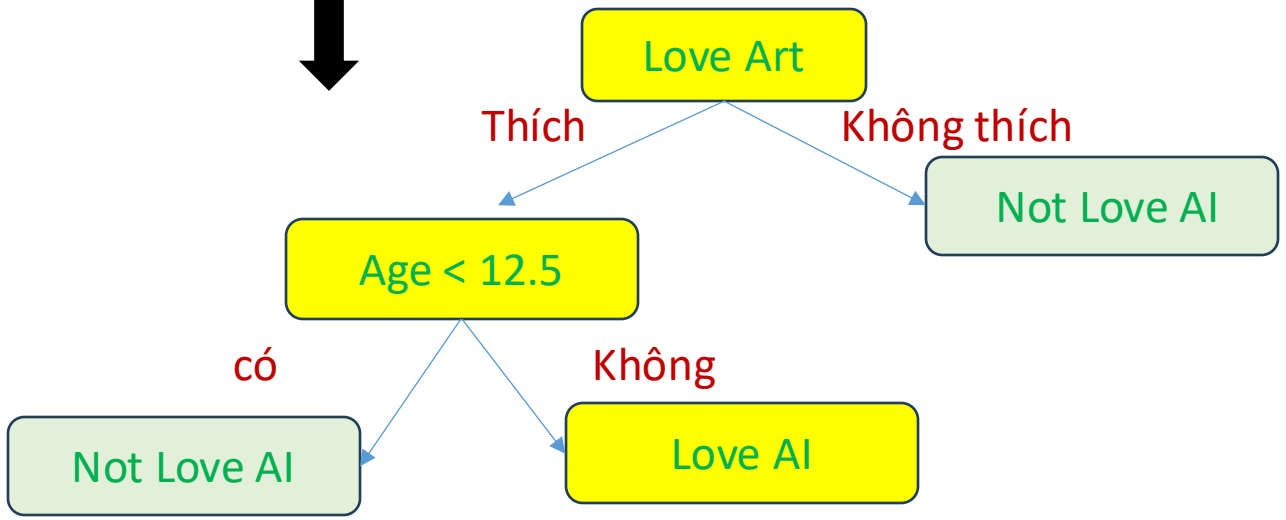
Final Tree



How to Use a Tree (Prediction)

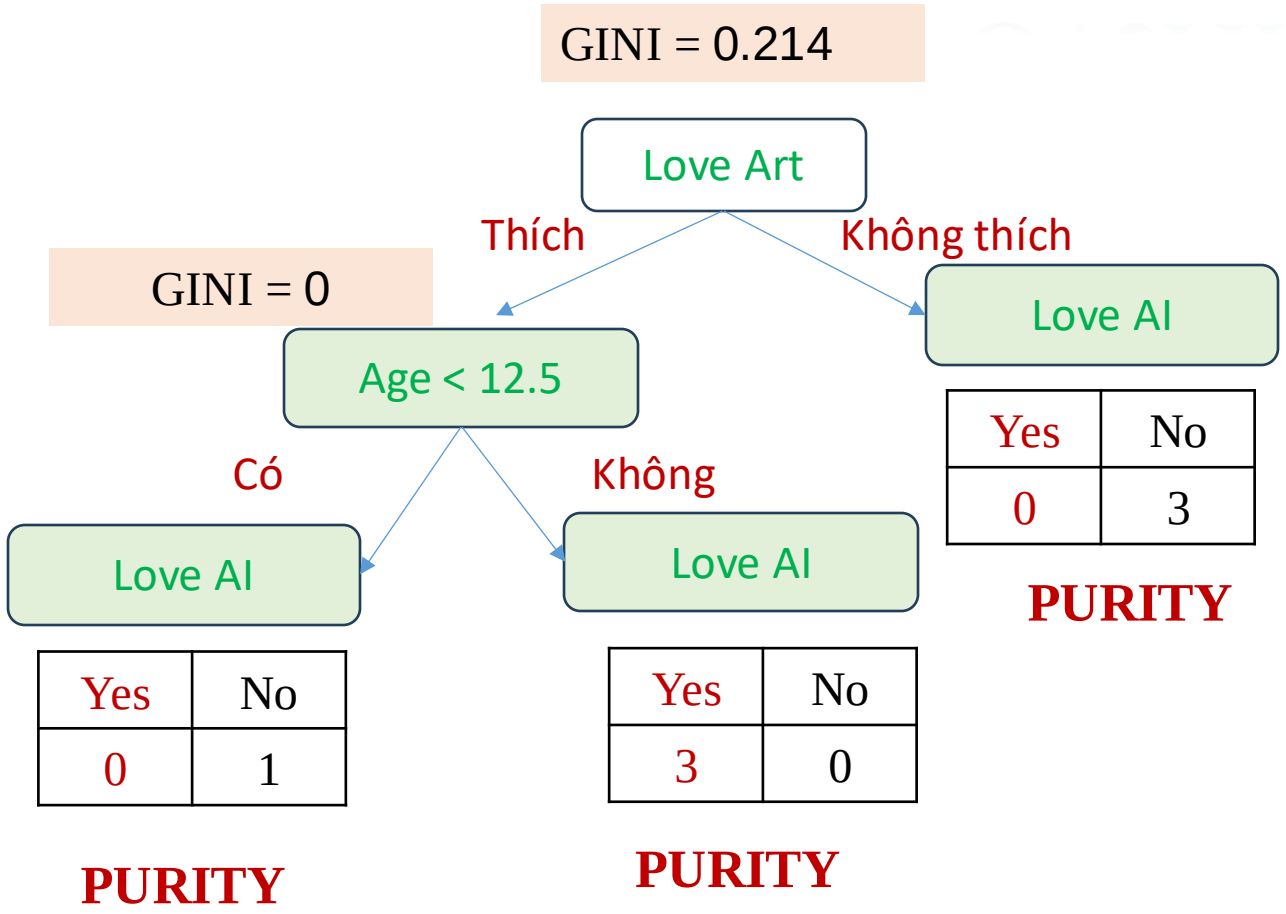


Love Math	Love Art	Age	Love AI
Yes	Yes	15	???



Love Math	Love Art	Age	Love AI
Yes	Yes	15	Love AI

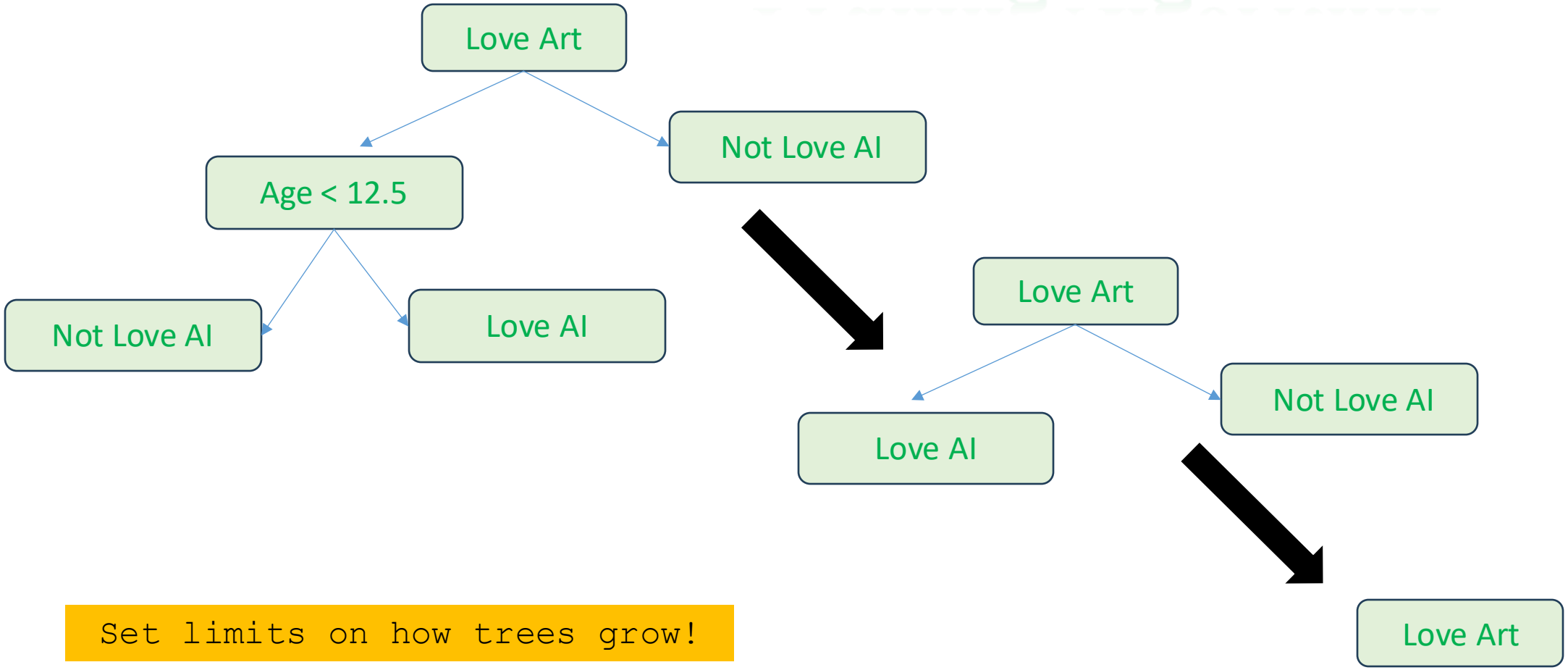
Overfitting in Tree



How to Reduce overfitting in Decision Tree

Set limits on how trees grow!

Pruning Algorithm



Set limits on how trees grow!

Outline

- Introduction to Decision Tree
- Classification Tree with GINI
- Classification Tree with Entropy
- Examples
- Summary

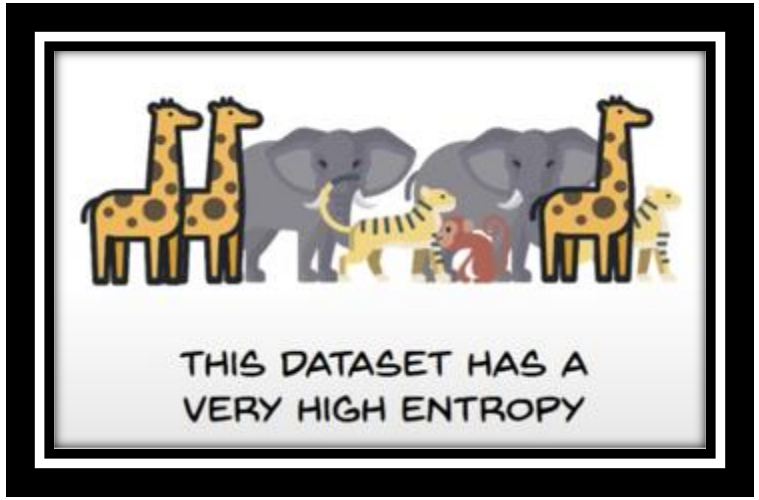
Evaluation Metrics

Entropy – Information Gain

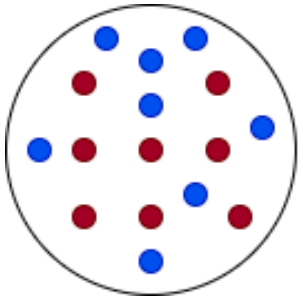
GNI IMPURITY

What is Entropy?

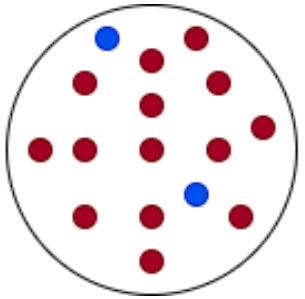
Entropy is an information theory metric that measures the impurity or uncertainty in a group of observations (dataset)



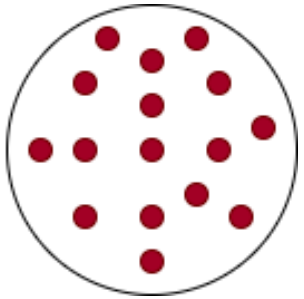
Very impure



Less Impure



Minimum Impurity



$$E = -\sum_{i=1}^{N} p_i log_2 p_i$$

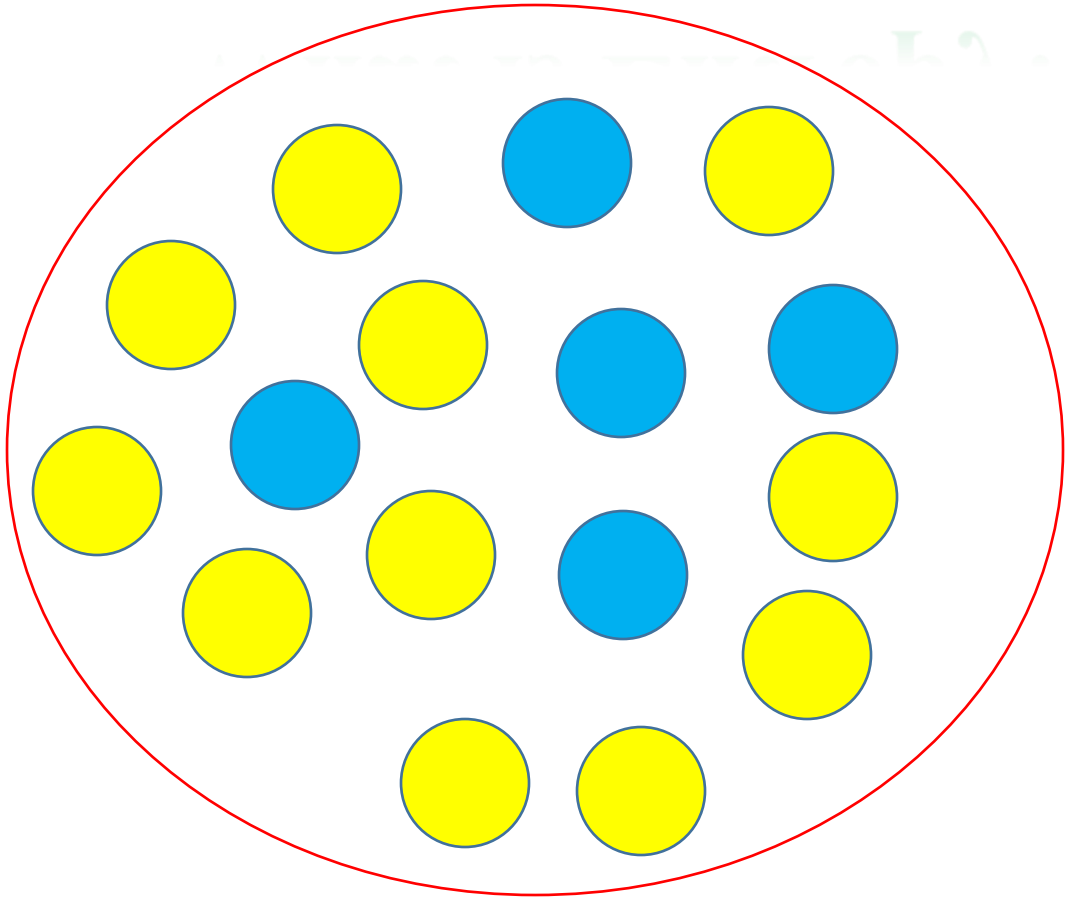
Này là gì vậy?
Nhìn rối quá

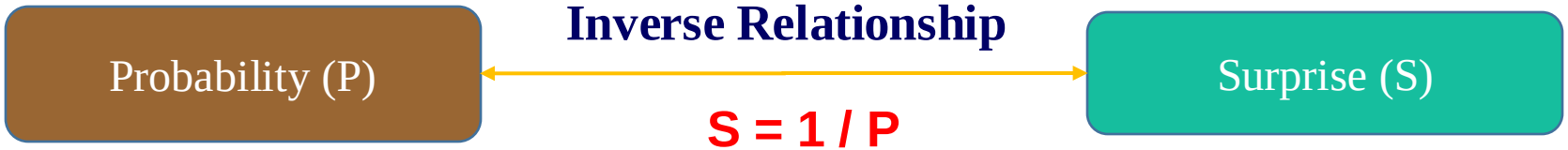
What is Entropy?

$$E = -\sum_{i=1}^{N} p_i log_2 p_i$$

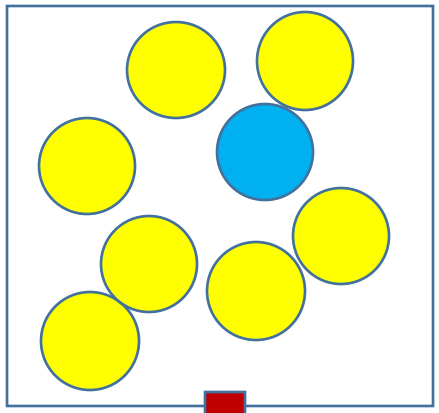
What is Entropy?

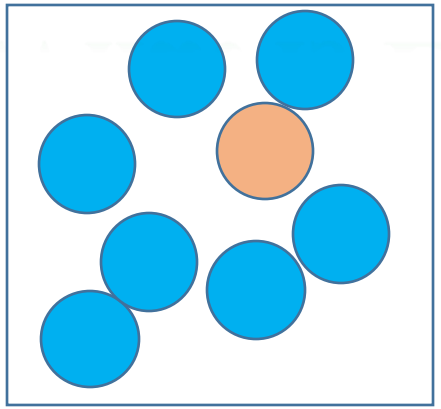
- Sample dataset

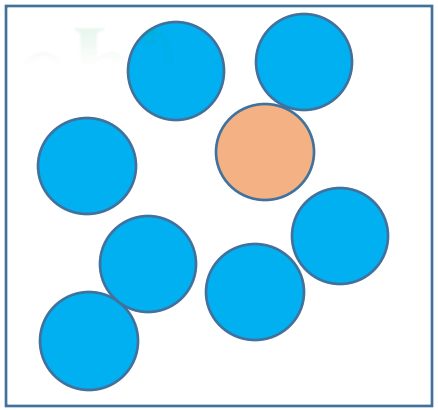


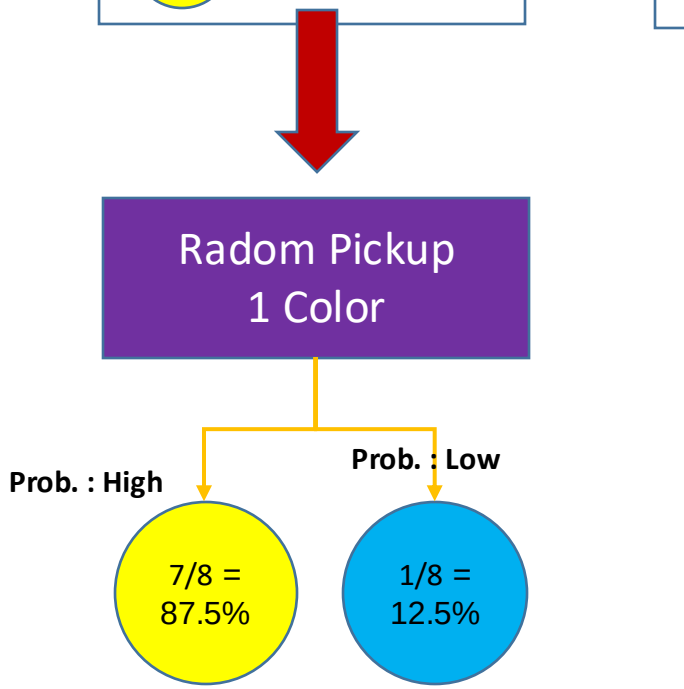


What is Entropy?









Inverse Relationship: S = 1/P

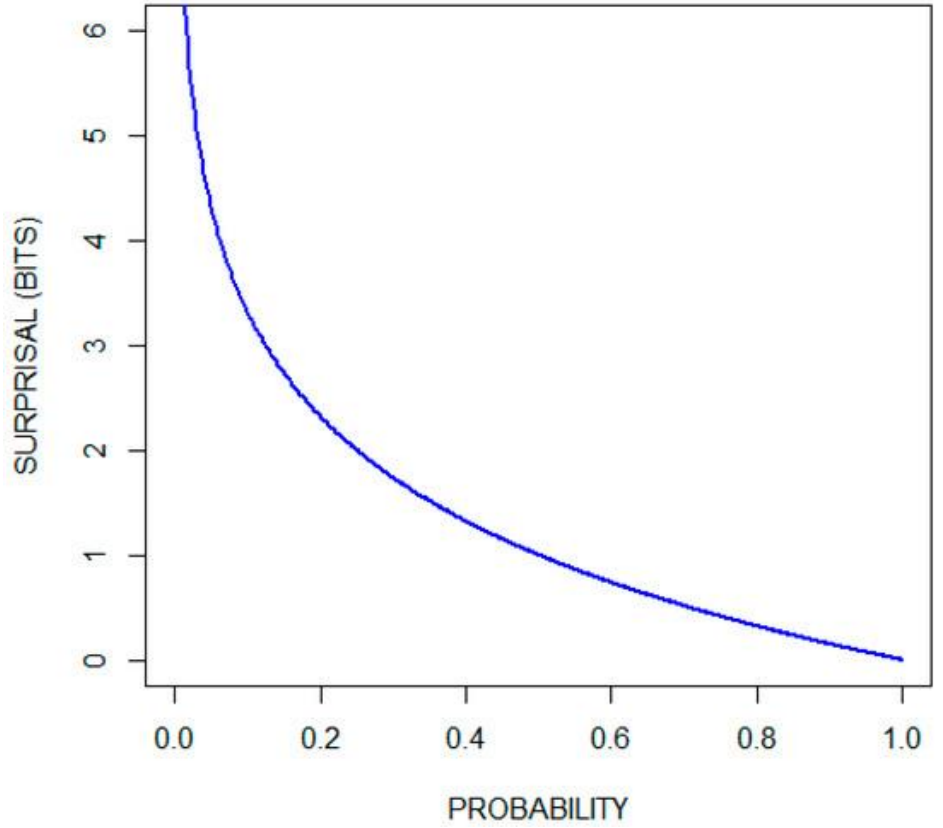
For Yellow: P = 7/8 => S = 1/(7/8) = 8/7
For Blue: P = 1/8 => S = 1/(1/8) = 8

Probability Vs. Surprise

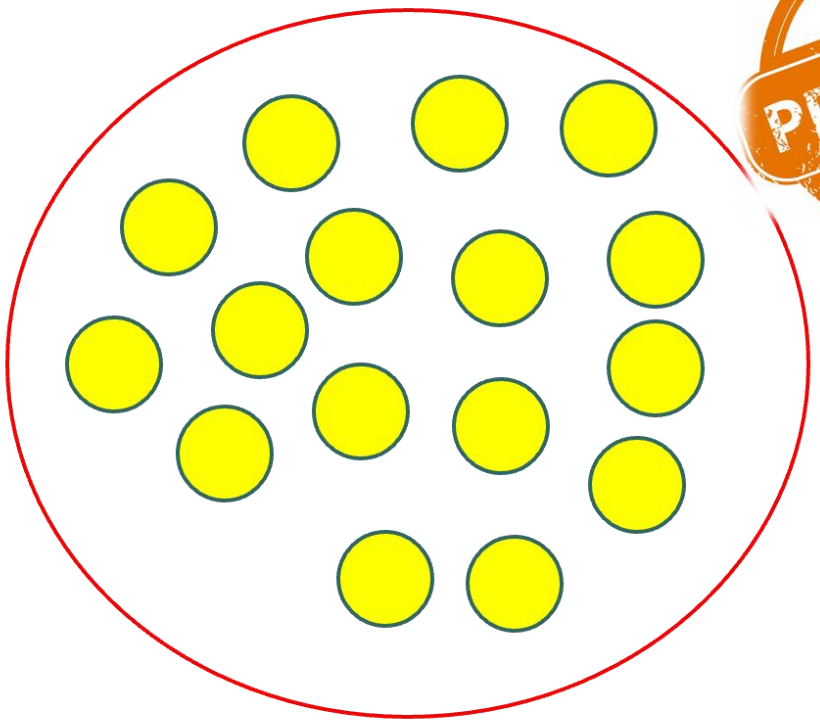
Probability (P) ⟷ Surprise (S)

Inverse Relationship

$$S = 1 / P$$



Probability Vs. Surprise



P = to get  = 1

S = to get  = 1/P = 1

S should be 0

Probability Vs. Surprise

Logs (Logarithms)

$$b^x = a \iff \log_b a = x$$

Argument

base

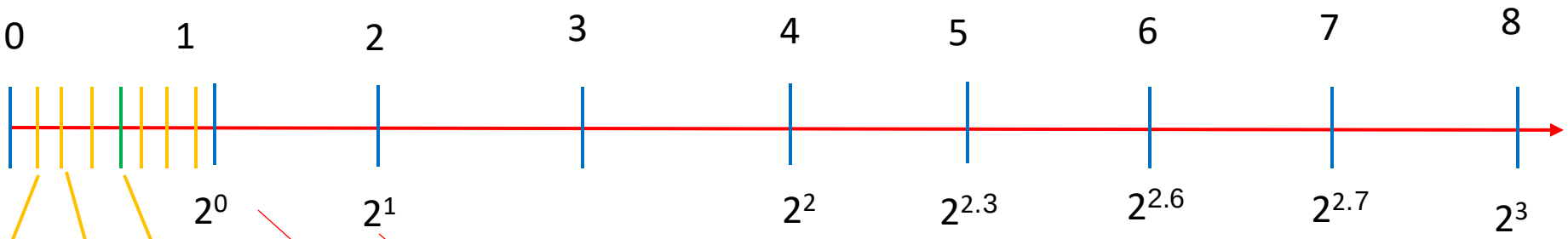
Logs (Logarithms)

Nhận xét: cả 2 measurement đều thể hiện độ lớn so với 1. Nhưng distance không đối xứng tại 1

8 time less than 1

8 time greater than 1

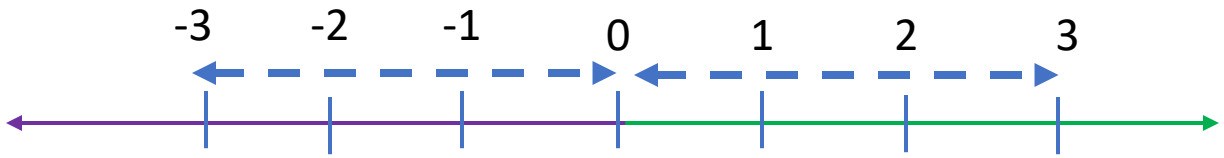
Ordinary number axis



$\frac{1}{8} = 2^{-3}$ $\frac{1}{4} = 2$ $\frac{1}{2} = 2^{-1}$

The log function is used to isolate the exponent

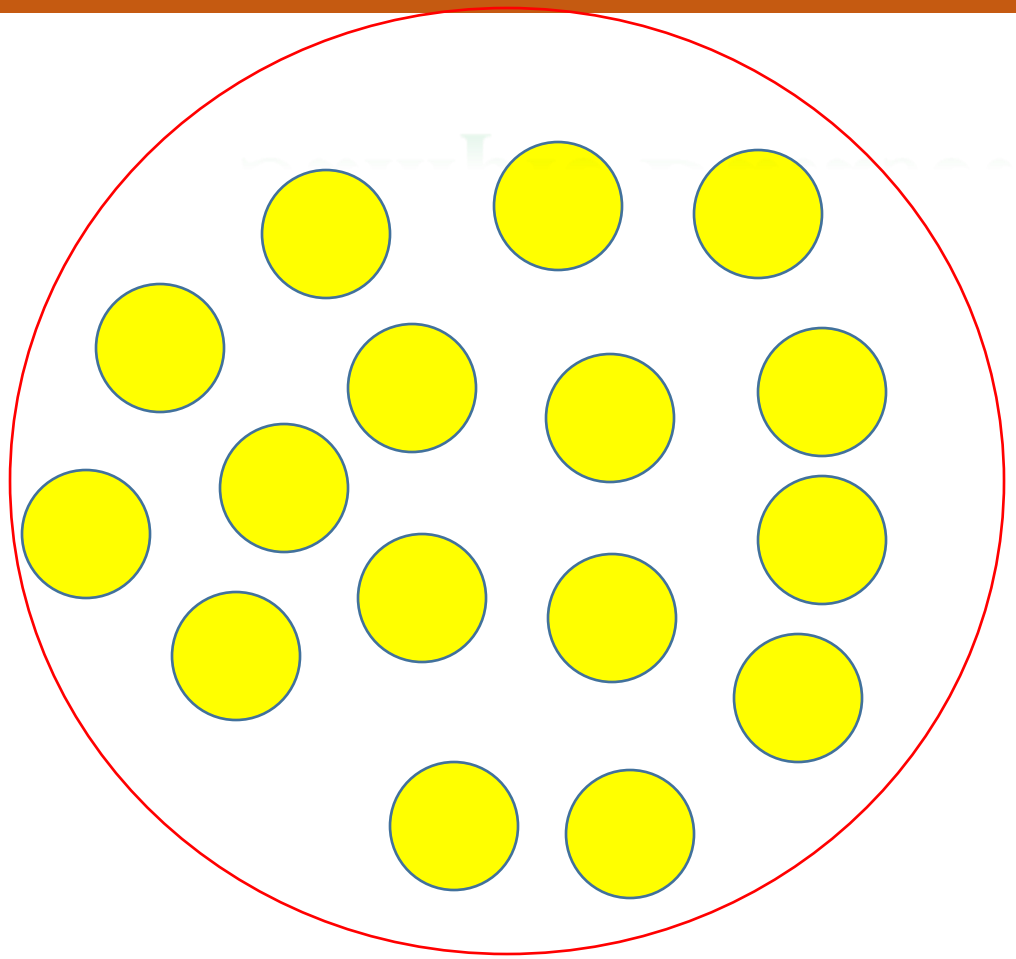
$\log_2(2^3)$=3 chính là phần luỹ thừa cần có của 2 để đạt được 8



Log base 2 Axis

Nhận xét: cả 2 measurement đều thể hiện độ lớn so với 1. Nhưng distance đối xứng tại 1

Sample Dataset

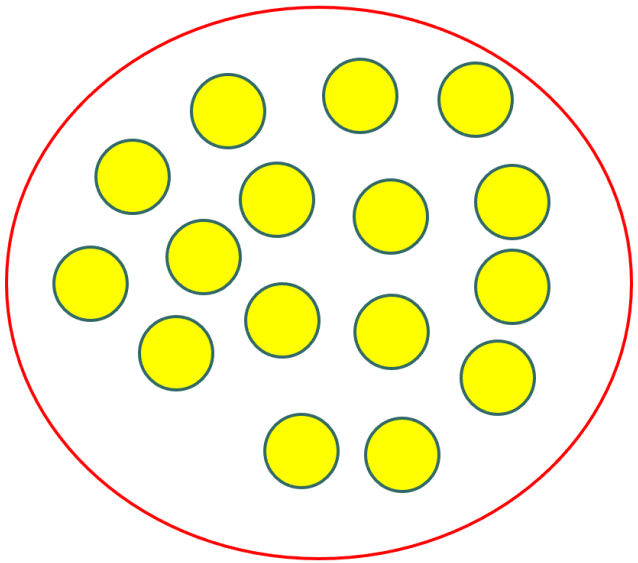


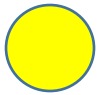
P = to get  = 1

S = to get  = 1/P = 1 

We use the log of 1/P to calculate S

Sample Dataset



P = to get  = 1

S = to get  = log2(1/1) = 0

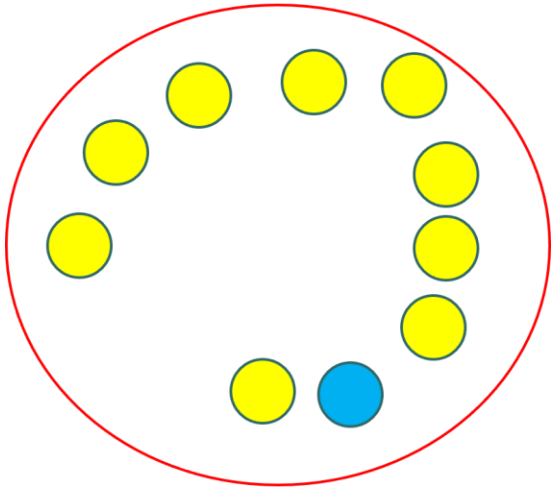
No surprise

P = to get  = 0

S = to get  = log2(1/0) = log2(1) – log2(0) = Undefined

Big surprise

Sample Dataset



P = to get  = 0.9

S = to get  = log2(1/0.9) = 0.15

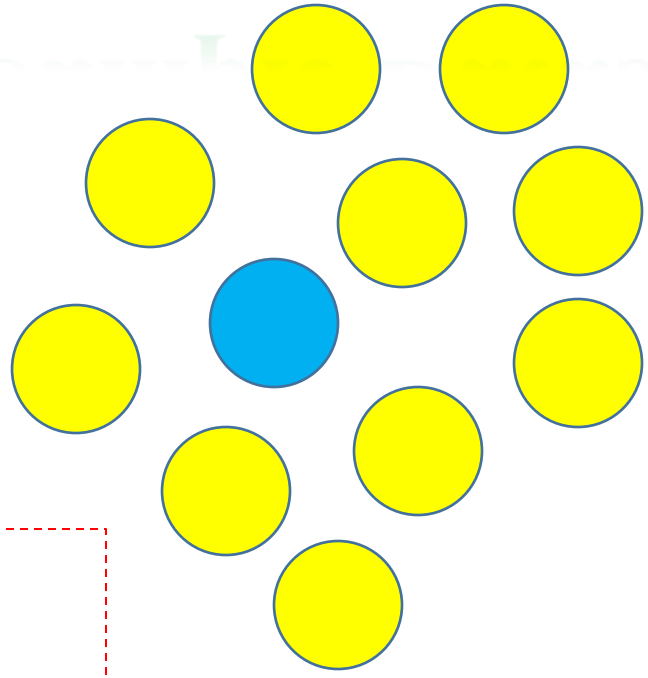
P = to get  = 0.1

S = to get  = log2(1/0.1) = 3.32

Sample Dataset

P to get = 9/10

P to get = 1/10



S to get = $\log_2(1 / 0.9) = 0.15$

S to get = $\log_2(1 / 0.1) = 3.32$

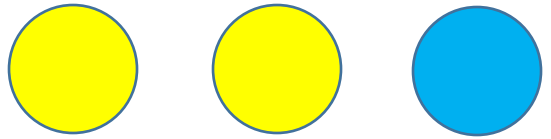
$$\log_2 \frac{1}{0.9 \times 0.9 \times 0.1} = \log_2 1 - \log_2(0.9 \times 0.9 \times 0.1)$$
$$= 0 - \log_2(0.9 \times 0.9 \times 0.1)$$
$$= 0 - \log_2(0.9) - \log_2(0.9) - \log_2(0.1)$$
$$= 0.15 + 0.15 + 3.32$$

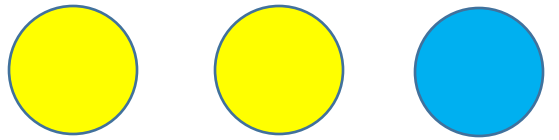
Gợi nhớ!

If two events A and B are independent , then the probability of happening of both A and B is:

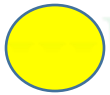
$P (A \cap B) = P(A). P(B)$

Flip the coin 3 times

P to get  **= 0.9 * 0.9 * 0.1**

S to get  **= S + S + S**

Sample Dataset

		
Probability	9/10	1/10
Surprise	0.15	3.32

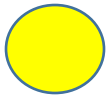
HOW MUCH SURPRISE
FROM GETTING YELLOW COLORS IN 100 TIMES

(0.9*100) * 0.15

Expected number of yellows

Total surprise from getting Yellow

Sample Dataset

		
Probability	9/10	1/10
Surprise	0.15	3.32

HOW MUCH SURPRISE
FROM GETTING BLUE COLORS IN 100 TIMES

(0.1*100) * 3.32

Expected number of blues

Total surprise from getting Blue

AVERAGE SURPRISE FOR 100 TIMES

S_B = HOW MUCH SURPRISE FROM GETTING BLUE COLORS IN 100 TIMES

(0.1*100) * 3.32

+

S_Y = HOW MUCH SURPRISE FROM GETTING YELLOW COLORS IN 100 TIMES

(0.9*100) * 0.15

$$E = -\sum_{i=1}^{N} p_i log_2 p_i$$

How ?

Total Surprise= S_B + S_Y = (0.1 x 100) x 3.32 + (0.9 x 100) x 0.15 = 46.7

Average = Total Surprise / 100 = (S_B + S_Y = (0.1 x 100) x 3.32 + (0.9 x 100) x 0.15) / 100 = 0.467

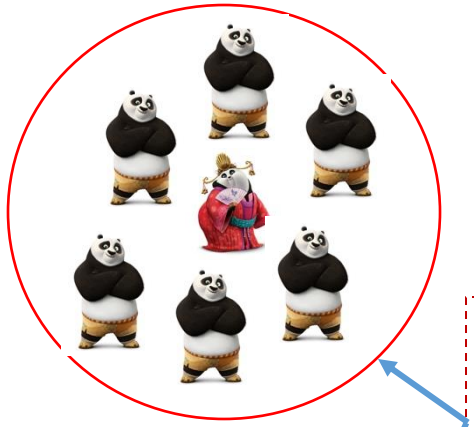
Entropy = Total Surprise / 100 = (0.1 x 3.32) + (0.9 x 0.15) = 0.467

$p(x)$ $\log(\frac{1}{p(x)})$

$$\text{Entropy} = \sum \log(\frac{1}{p(x)}) p(x)$$

Surprise — The probability of the Surprise.

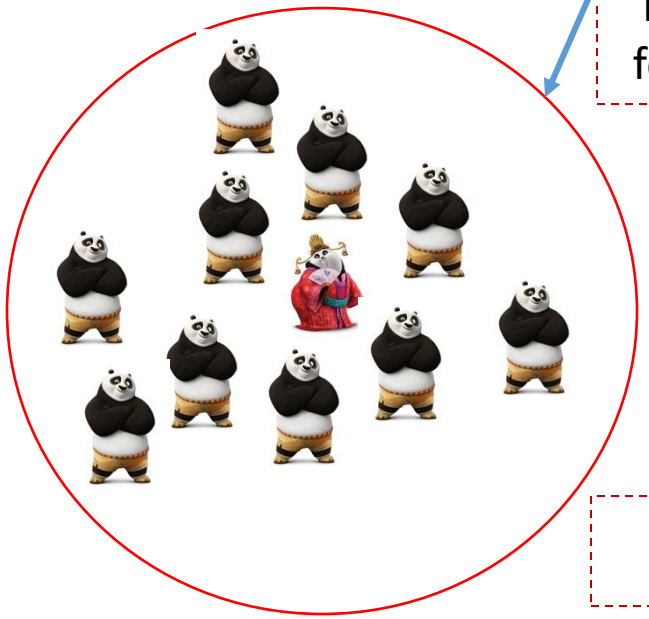
Sample Dataset



$$\text{Entropy} = \sum \; p(x)\log\left(\frac{1}{p(x)}\right) = \frac{6}{7}\log_2\frac{7}{6} + \frac{1}{7}\log_2 7$$

Entropy = 0.59

When we increase the difference in the number of male and femal panda. Entropy is lower



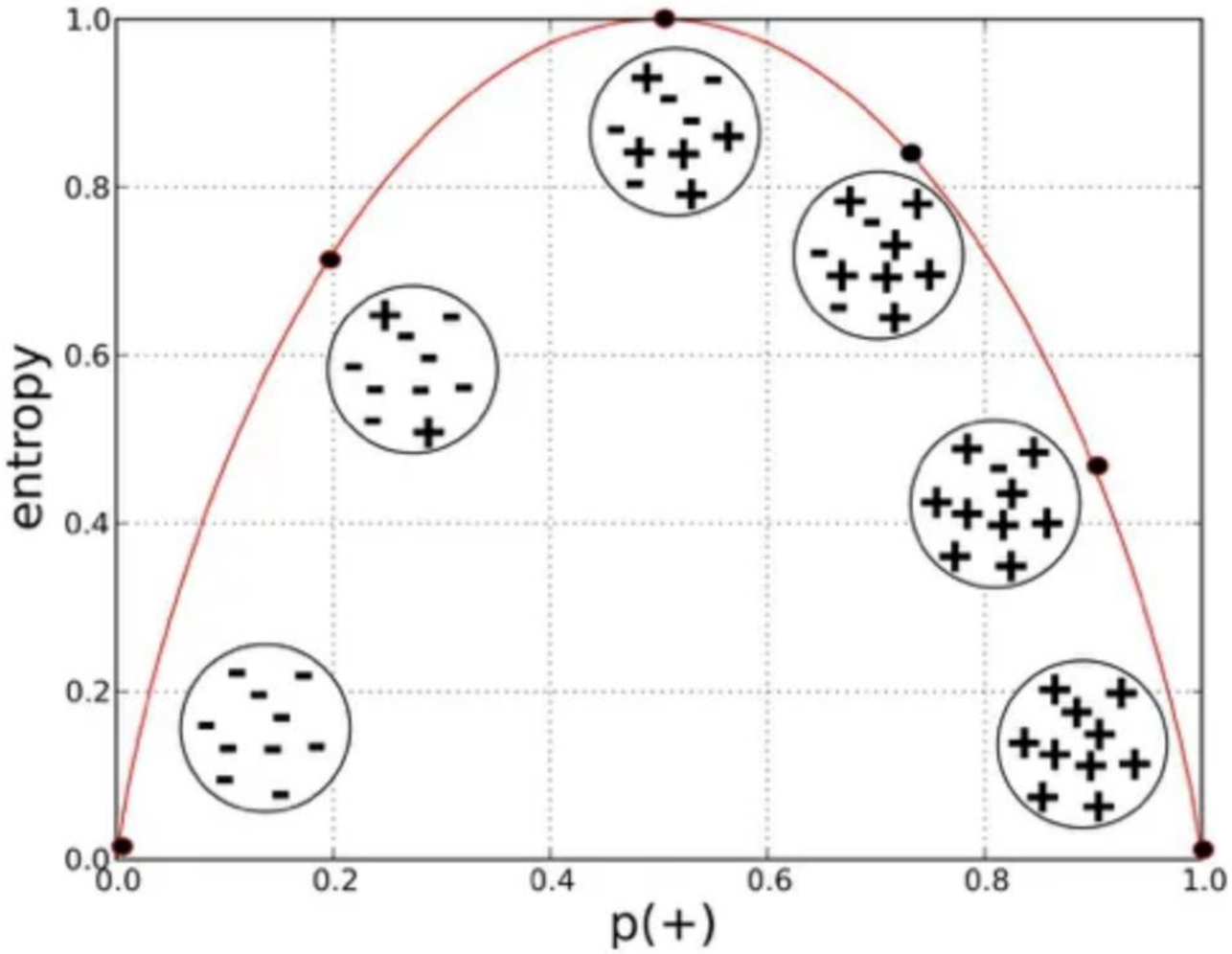
Entropy = 0.44

Entropy is highest when we have same number of female and male panda

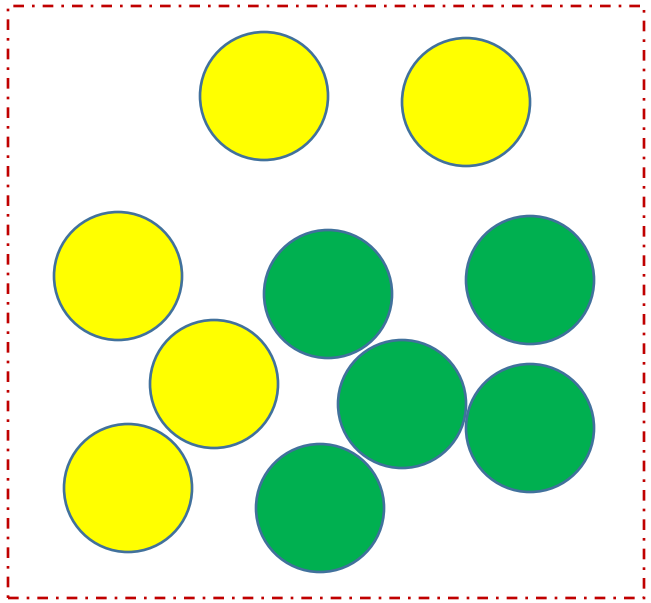
Entropy = 1.0

Entropy is used to quantify the similarity or difference in the dataset

Entropy and Probability

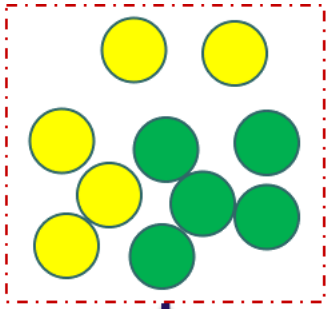


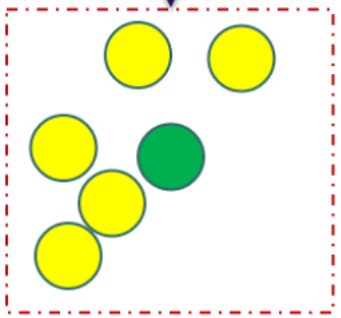
Sample Dataset



$$E_{initial} = -((0.5log_20.5) + (0.5log_20.5))$$

$$= 1$$



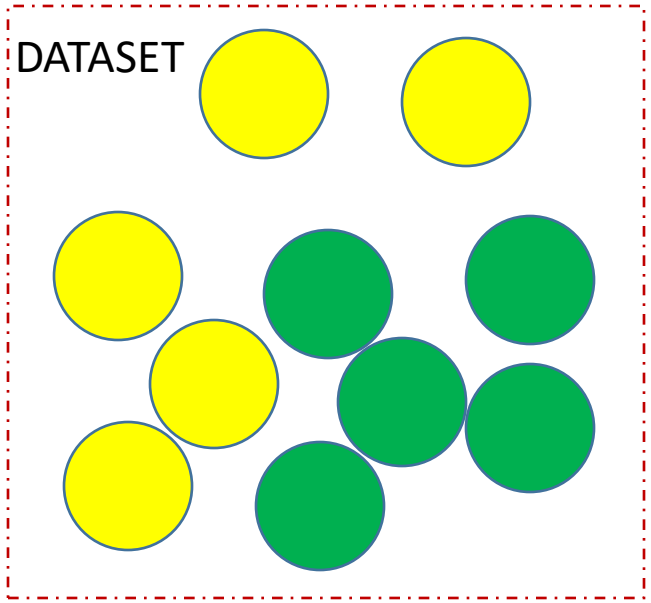


E= 0.65

E= 0

AI VIET NAM
@aivietnam.edu.vn

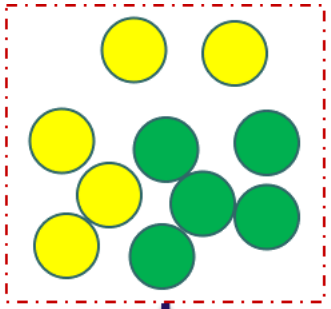
Information Gain (IG)



Maximum the information gain

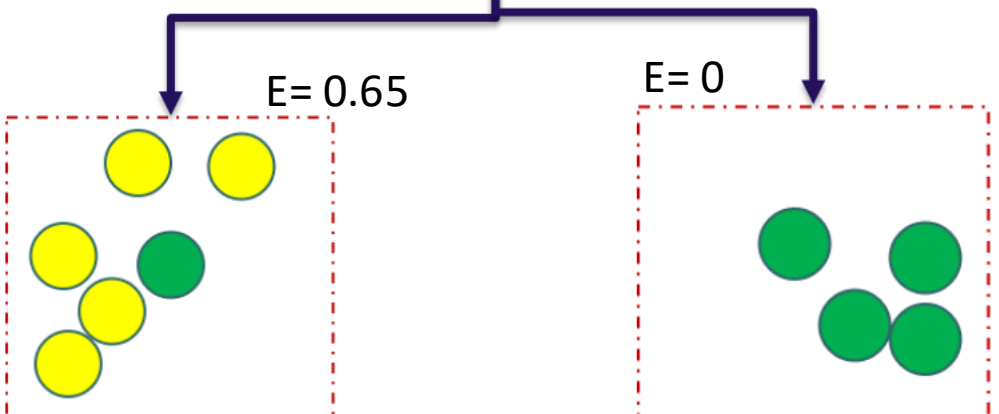
$$E_{initial} = -((0.5log_20.5) + (0.5log_20.5))$$

$$= 1$$



$$IG(Y, X) = E(Y) - E(Y|X)$$

$$Gain = E_{parent} - E_{children}$$



$E_{\_after\_slit} = 0.65 * 0.6 + 0.4 * 0 = 0.39$

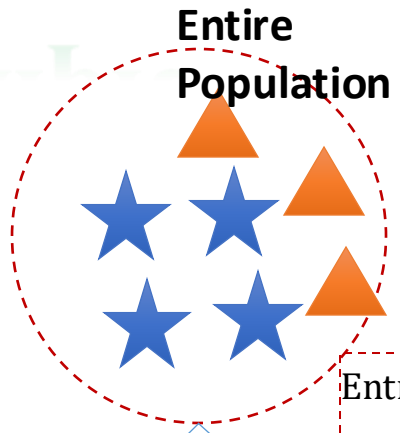
$Gain = 0.61 = E_{initial} - E_{\_after\_slit}$

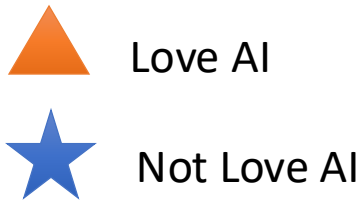
Example

No.	Love Math	Love Art	Love AI
1	Yes	Yes	No
2	Yes	No	No
3	No	Yes	Yes
4	No	Yes	Yes
5	Yes	Yes	Yes
6	Yes	No	No
7	No	No	No

Cần xác định Root node nên là Love Math hay Love Art?

Giả sử ta chọn root node là **Love Math**?

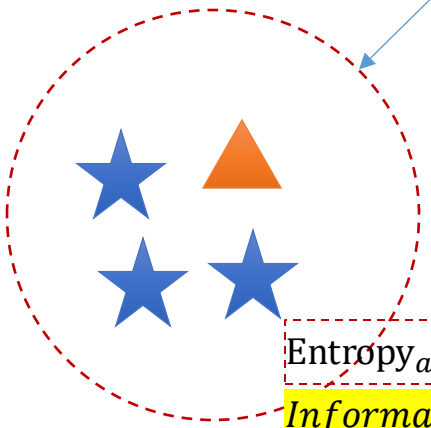


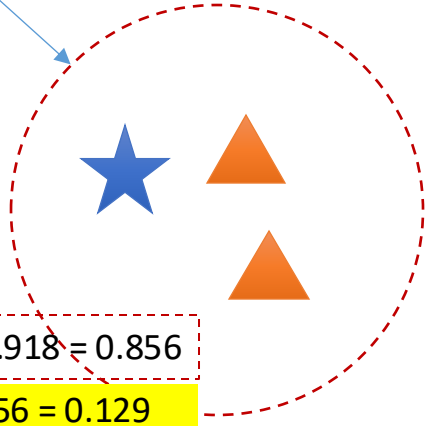


$$\text{Entropy\_}before = -\frac{3}{7}\log\left(\frac{3}{7}\right) - \frac{4}{7}\log\left(\frac{4}{7}\right) = 0.985$$

Love Math is Yes

Love Math is No





$$\text{Entropy}_{after} = \frac{4}{7} \times 0.811 + \frac{3}{7} \times 0.918 = 0.856$$

$$Information\ Gain = 0.985 - 0.856 = 0.129$$

$$\text{Entropy} = -\frac{3}{4}\log\left(\frac{3}{4}\right) - \frac{1}{4}\log\left(\frac{1}{4}\right) = 0.811$$

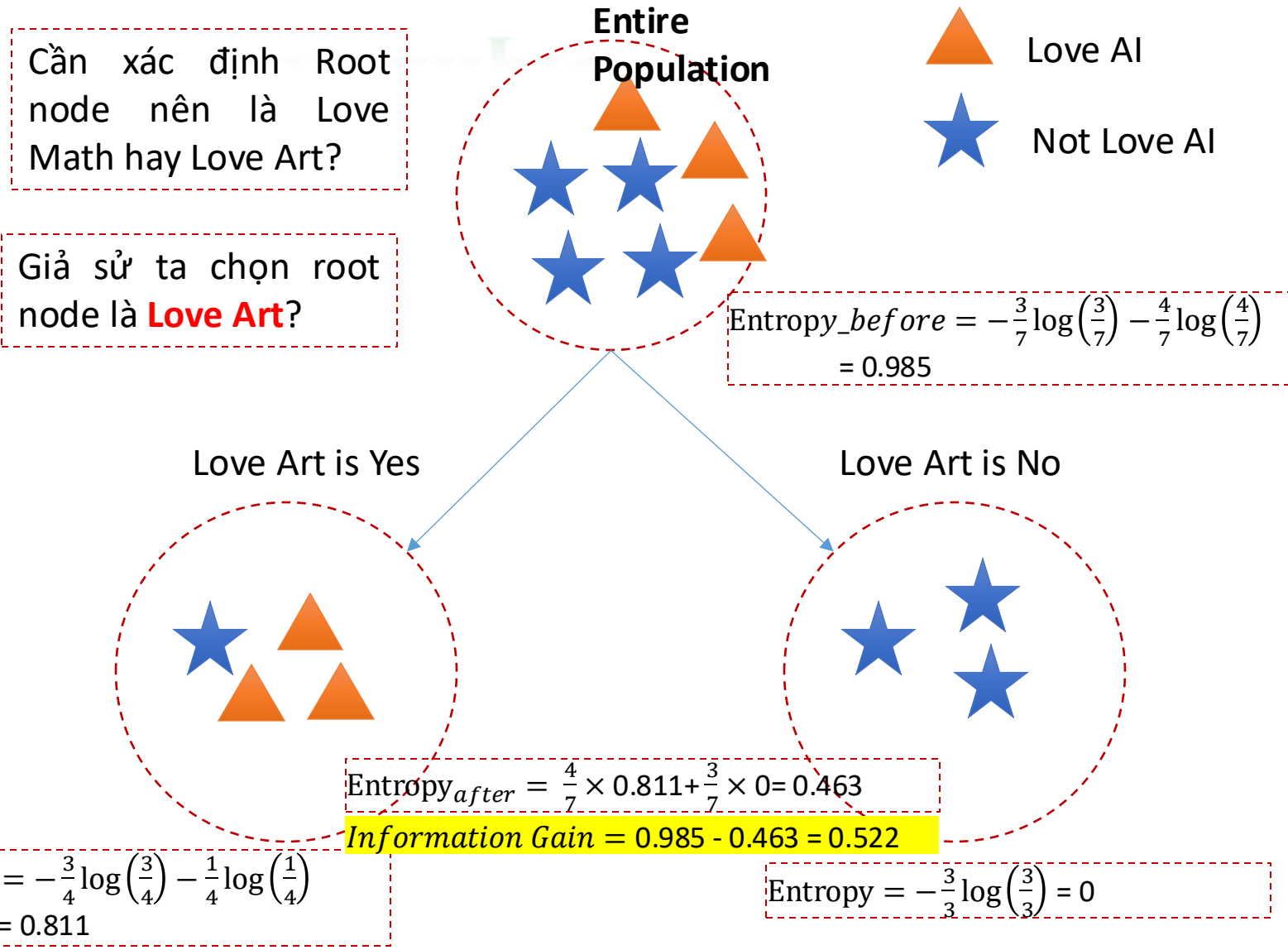
$$\text{Entropy} = -\frac{3}{4}\log\left(\frac{3}{4}\right) - \frac{1}{4}\log\left(\frac{1}{4}\right) = 0.918$$

Example

No.	Love Math	Love Art	Love AI
1	Yes	Yes	No
2	Yes	No	No
3	No	Yes	Yes
4	No	Yes	Yes
5	Yes	Yes	Yes
6	Yes	No	No
7	No	No	No

Cần xác định Root node nên là Love Math hay Love Art?

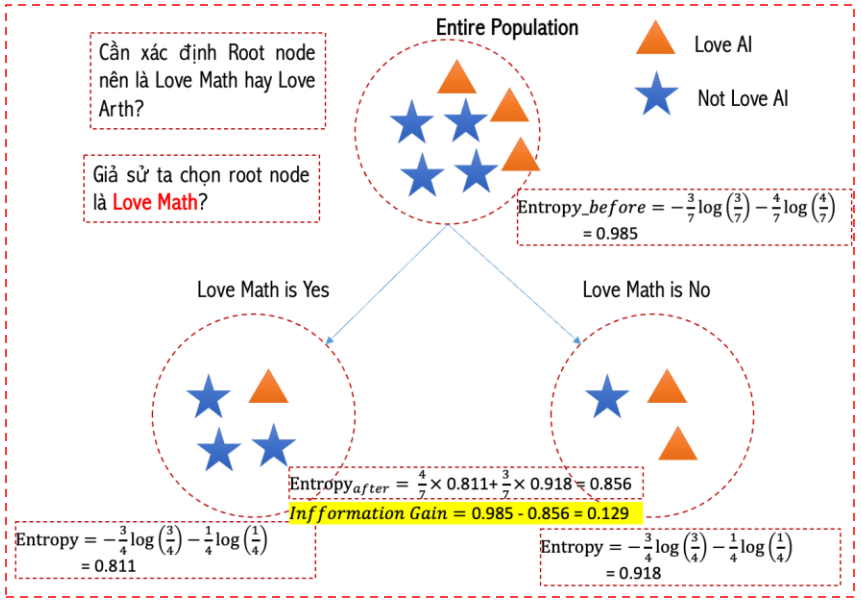
Giả sử ta chọn root node là **Love Art**?



AI VIET NAM
@aivietnam.edu.vn

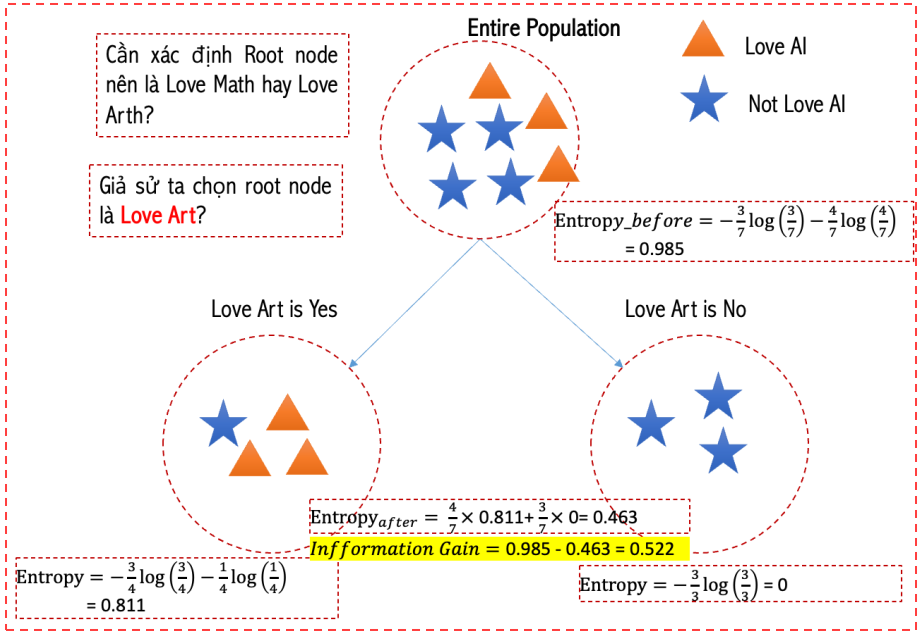
Example

Love Math is a Root Node



$Information\ Gain$ = 0.129

Love Art is a Root Node



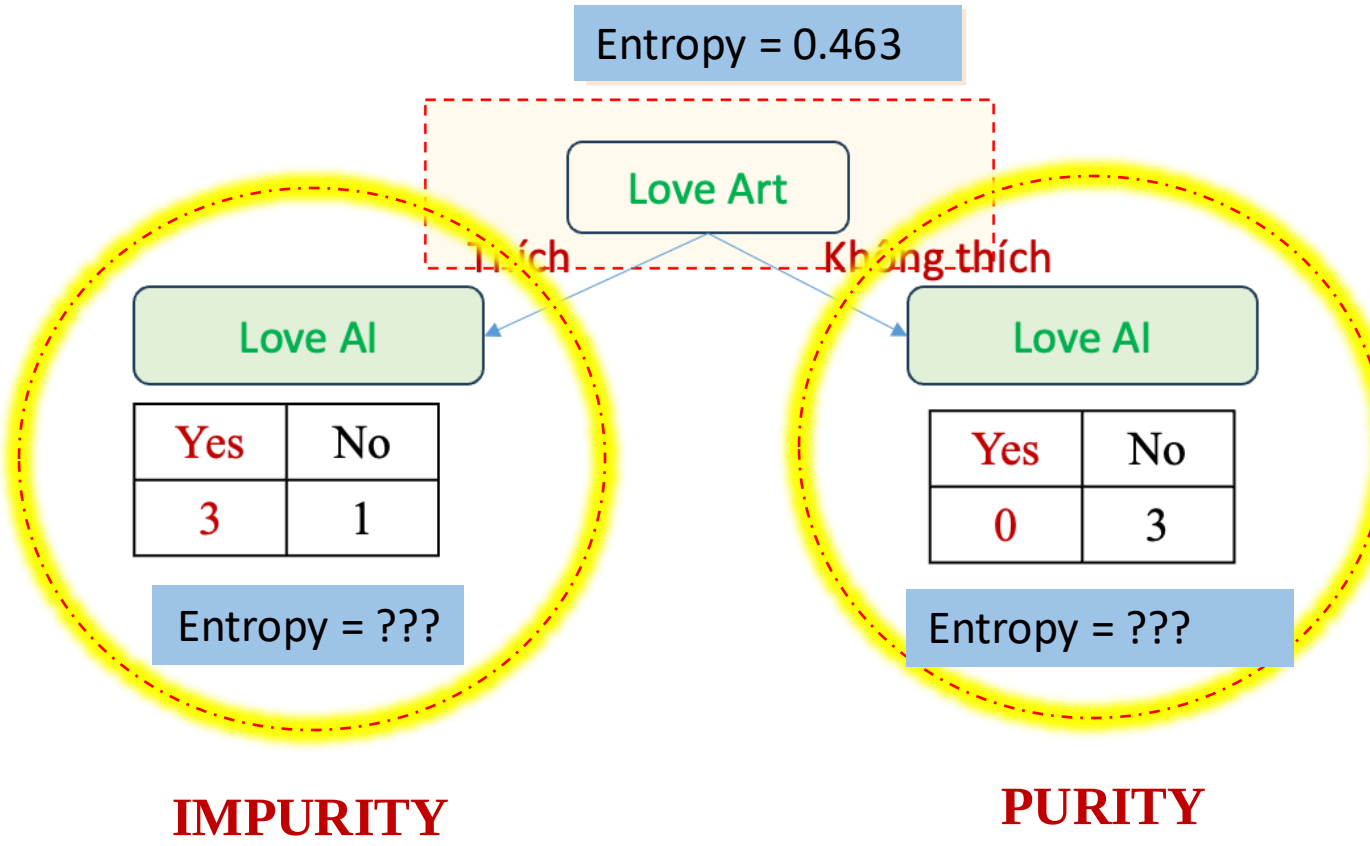
Infformation Gain = 0.522

Which attribute is in the first node?

Maximum the information gain

Example

No.	Love Math	Love Art	Love AI
1	Yes	Yes	No
2	Yes	No	No
3	No	Yes	Yes
4	No	Yes	Yes
5	Yes	Yes	Yes
6	Yes	No	No
7	No	No	No



Continue to expand this tree using Entropy metric

Outline

- Introduction to Decision Tree
- Classification Tree with GINI
- Classification Tree with Entropy
- Examples
- Summary

Credit Card Default Prediction

	age	income	student	credit_rate	Default
0	youth	high	no	fair	no
1	youth	high	no	excellent	no
2	middle_age	high	no	fair	yes
3	senior	medium	no	fair	yes
4	senior	low	yes	fair	yes
5	senior	low	yes	excellent	no
6	middle_age	low	yes	excellent	yes
7	youth	medium	no	fair	no
8	youth	low	yes	fair	yes
9	senior	medium	yes		
10	youth	medium	yes	ex	
11	middle_age	medium	no	ex	
12	middle_age	high	yes		
13	senior	medium	no	ex	

This data set is used to predict whether a person will default on their credit card. There are two classes (default = 'yes', no_default = 'no'):

```
# Defining a simple dataset
attribute_names =  ['age', 'income','student', 'credit_rate']
class_name = 'default'
data1 ={
    'age' : ['youth', 'youth', 'middle_age', 'senior', 'senior', 'senior','middle_age', 'youth', 'youth', 'senior',
    'income' : ['high', 'high', 'high', 'medium', 'low', 'low', 'low', 'medium','low', 'medium', 'medium', 'medium',
    'student' : ['no','no','no','no','yes','yes','yes','no','yes','yes','yes','no','yes','no'],
    'credit_rate' : ['fair', 'excellent', 'fair', 'fair', 'fair', 'excellent', 'excellent', 'fair', 'fair', 'fair','
    'default' : ['no', 'no', 'yes', 'yes', 'yes', 'no', 'yes', 'no', 'yes', 'yes','yes', 'yes', 'yes', 'no']
}
df1 = pd.DataFrame (data1, columns=data1.keys())
```

Credit Card Default Prediction

```python
# STEP 1: Calculate gini(D)
def gini_impurity (value_counts):
    n = value_counts.sum()
    p_sum = 0
    for key in value_counts.keys():
        p_sum = p_sum  +  (value_counts[key] / n ) * (value_counts[key] / n )
    gini = 1 - p_sum
    return gini

class_value_counts = df1[class_name].value_counts()
print(f'Number of samples in each class is:\n{class_value_counts}')

gini_class = gini_impurity(class_value_counts)
print(f'\nGini Impurity of the class is {gini_class:.3f}')
```

```
Number of samples in each class is:
yes    9
no     5
Name: default, dtype: int64

Gini Impurity of the class is 0.459
```

Credit Card Default Prediction

```python
# STEP 2:
# Calculating  gini impurity for the attiributes
def gini_split_a(attribute_name):
    attribute_values = df1[attribute_name].value_counts()
    gini_A = 0
    for key in attribute_values.keys():
        df_k = df1[class_name][df1[attribute_name] == key].value_counts()
        n_k = attribute_values[key]
        n = df1.shape[0]
        gini_A = gini_A + (( n_k / n) * gini_impurity(df_k))
    return gini_A

gini_attiribute ={}
for key in attribute_names:
    gini_attiribute[key] = gini_split_a(key)
    print(f'Gini for {key} is {gini_attiribute[key]:.3f}')
```

```
Gini for age is 0.343
Gini for income is 0.440
Gini for student is 0.367
Gini for credit_rate is 0.429
```

Credit Card Default Prediction

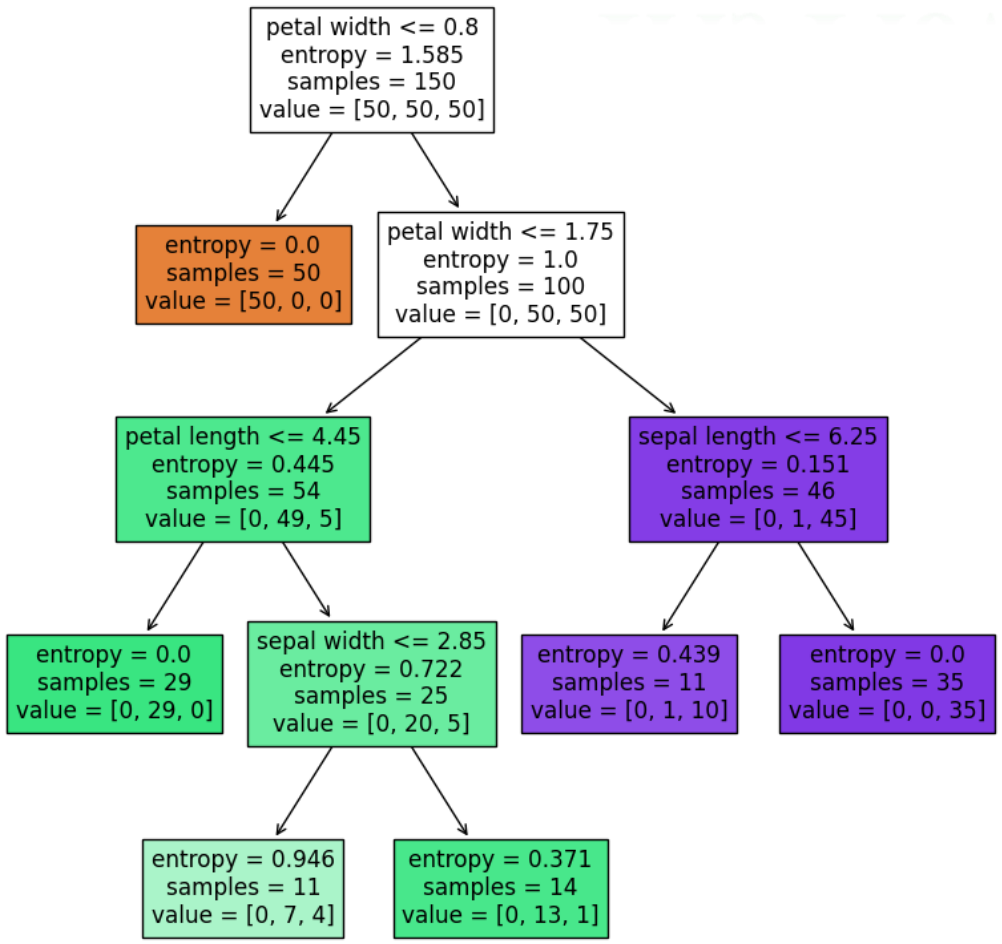
```
# STEP 3:
# Compute Gini gain values to find the best split
# An attribute has maximum Gini gain is selected for splitting.

min_value = min(gini_attiribute.values())
print('The minimum value of Gini Impurity : {0:.3} '.format(min_value))
print('The maximum value of Gini Gain     : {0:.3} '.format(1-min_value))

selected_attribute = min(gini_attiribute.keys())
print('The selected attiribute is: ', selected_attribute)
```

```
The minimum value of Gini Impurity : 0.343
The maximum value of Gini Gain     : 0.657
The selected attiribute is    :  age
```

Iris Flower Classification

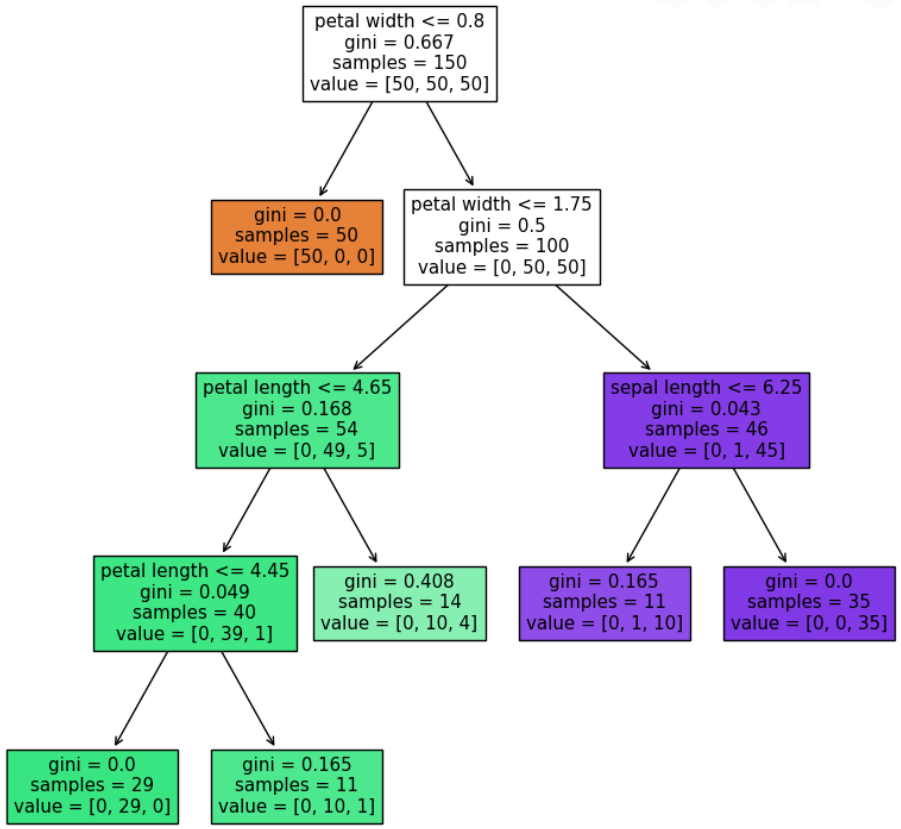
	Sepal length	Sepal width	Petal length	Petal width	Class
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
⋮	⋮	⋮	⋮	⋮	⋮
150	5.9	3.0	5.1	1.8	virginica

Iris Flower Classification (Entropy)




```
dataset = load_iris()
X = dataset.data
y = dataset.target

classifier = tree.DecisionTreeClassifier(criterion="entropy",
                                         max_depth=4, min_samples_leaf=10)
classifier.fit(X,y)
fig, ax = plt.subplots(figsize=(10,10))
tree.plot_tree(classifier,ax=ax, feature_names=["sepal length", "sepal width",
                                                "petal length", "petal width"],
                                                 filled=True)

plt.show()
```

Iris Flower Classification (GINI)




```
dataset = load_iris()
X = dataset.data
y = dataset.target

classifier = tree.DecisionTreeClassifier(criterion="gini",
                                         max_depth=4, min_samples_leaf=10)
classifier.fit(X,y)
fig, ax = plt.subplots(figsize=(10,10))
tree.plot_tree(classifier,ax=ax, feature_names=["sepal length", "sepal width",
                                                "petal length", "petal width"],
                                                 filled=True)

plt.show()
```

Credit Card Fraud Detection: 01

It is important that credit card companies are able to recognize fraudulent credit card transactions so that customers are not charged for items that they did not purchase.

The dataset contains transactions made by credit cards in September 2013 by European cardholders. This dataset presents transactions that occurred in two days, where we have 492 frauds out of 284,807 transactions. The dataset is highly unbalanced, the positive class (frauds) account for 0.172% of all transactions

Feature 'Class' is the response variable and it takes value 1 in case of fraud and 0 otherwise.

V8	V9	...	V21	V22	V23	V24	V25	V26	V27	V28	Amount	Class
8698	0.363787	...	-0.018307	0.277838	-0.110474	0.066928	0.128539	-0.189115	0.133558	-0.021053	149.62	0
5102	-0.255425	...	-0.225775	-0.638672	0.101288	-0.339846	0.167170	0.125895	-0.008983	0.014724	2.69	0
7676	-1.514654	...	0.247998	0.771679	0.909412	-0.689281	-0.327642	-0.139097	-0.055353	-0.059752	378.66	0
7436	-1.387024	...	-0.108300	0.005274	-0.190321	-1.175575	0.647376	-0.221929	0.062723	0.061458	123.50	0
0533	0.817739	...	-0.009431	0.798278	-0.137458	0.141267	-0.206010	0.502292	0.219422	0.215153	69.99	0
0314	-0.568671	...	-0.208254	-0.559825	-0.026398	-0.371427	-0.232794	0.105915	0.253844	0.081080	3.67	0
1213	0.464960	...	-0.167716	-0.270710	-0.154104	-0.780055	0.750137	-0.257237	0.034507	0.005168	4.99	0
7864	0.615375	...	1.943465	-1.015455	0.057504	-0.649709	-0.415267	-0.051634	-1.206921	-1.085339	40.80	0

AI VIET NAM
@aivietnam.edu.vn

Credit Cad Fraud Detection: 01

V8	V9	...	V21	V22	V23	V24	V25	V26	V27	V28	Amount	Class
8698	0.363787	...	-0.018307	0.277838	-0.110474	0.066928	0.128539	-0.189115	0.133558	-0.021053	149.62	0
5102	-0.255425	...	-0.225775	-0.638672	0.101288	-0.339846	0.167170	0.125895	-0.008983	0.014724	2.69	0
7676	-1.514654	...	0.247998	0.771679	0.909412	-0.689281	-0.327642	-0.139097	-0.055353	-0.059752	378.66	0
7436	-1.387024	...	-0.108300	0.005274	-0.190321	-1.175575	0.647376	-0.221929	0.062723	0.061458	123.50	0
0533	0.817739	...	-0.009431	0.798278	-0.137458	0.141267	-0.206010	0.502292	0.219422	0.215153	69.99	0
0314	-0.568671	...	-0.208254	-0.559825	-0.026398	-0.371427	-0.232794	0.105915	0.253844	0.081080	3.67	0
1213	0.464960	...	-0.167716	-0.270710	-0.154104	-0.780055	0.750137	-0.257237	0.034507	0.005168	4.99	0
7864	0.615375	...	1.943465	-1.015455	0.057504	-0.649709	-0.415267	-0.051634	-1.206921	-1.085339	40.80	0

```
# load dataset
creditdata_df = pd.read_csv("/content/drive/MyDrive/AIO2024/creditcard.csv")
print(f"Dataset Shape :-")
print (creditdata_df.shape)
```

```
[11] X = creditdata_df.drop('Class', axis=1)
     y = creditdata_df['Class']
```

```
[12] X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
     classifier = DecisionTreeClassifier()
     classifier.fit(X_train, y_train)
```

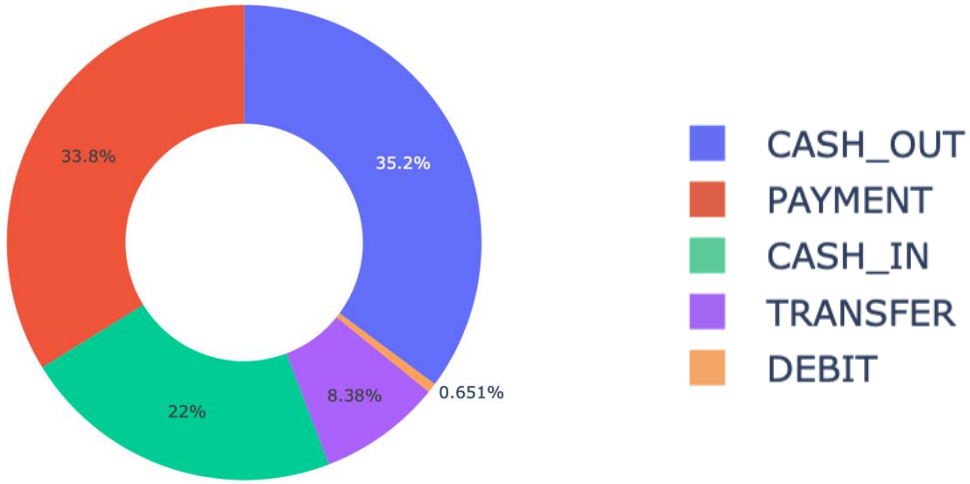
AI VIET NAM
@aivietnam.edu.vn

Credit Card Fraud Detection: 02

```
[17] print(data.isnull().sum())
```

```
step              0
type              0
amount            0
nameOrig          0
oldbalanceOrg     0
newbalanceOrig    0
nameDest          0
oldbalanceDest    0
newbalanceDest    0
isFraud           0
isFlaggedFraud    0
dtype: int64
```

```
type = data["type"].value_counts()
transactions = type.index
quantity = type.values

import plotly.express as px
figure = px.pie(data,
                values=quantity,
                names=transactions,hole = 0.5,
                title="Distribution of Transaction Type")
figure.show()
```



Credit Card Fraud Detection: 02

```
# training a machine learning model
from sklearn.tree import DecisionTreeClassifier
xtrain, xtest, ytrain, ytest = train_test_split(x, y, test_size=0.10, random_state=42)
model = DecisionTreeClassifier()
model.fit(xtrain, ytrain)
print(model.score(xtest, ytest))
```

```
0.9997375295082843
```

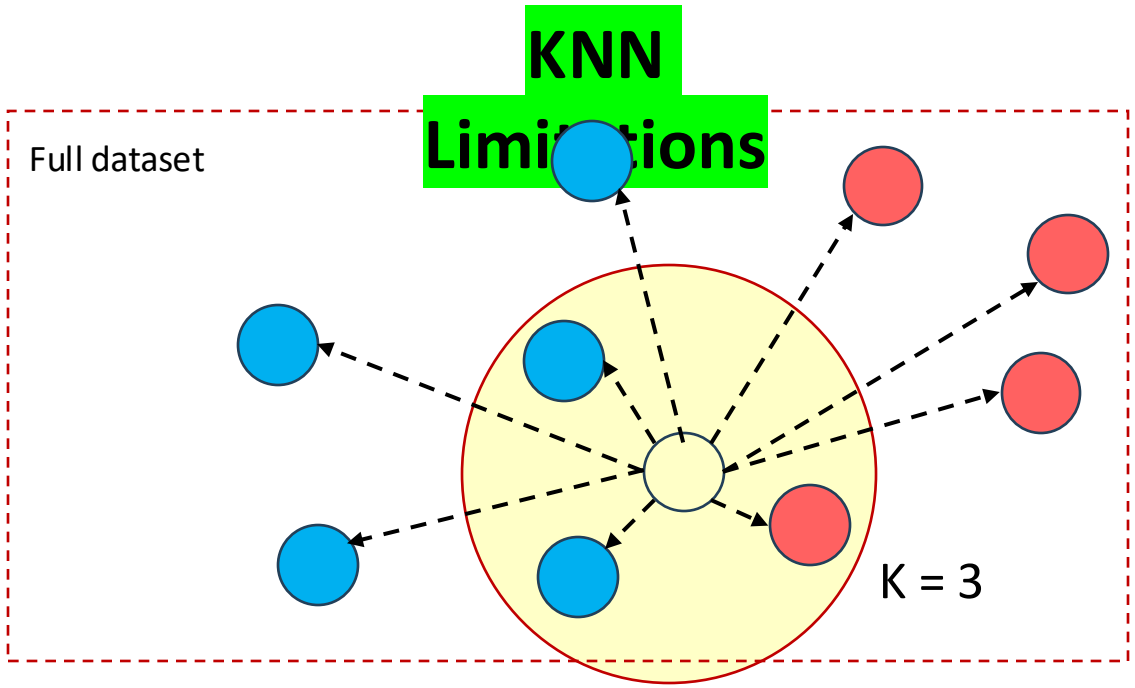
```
# prediction
#features = [type, amount, oldbalanceOrg, newbalanceOrig]
features = np.array([[4, 9000.60, 9000.60, 0.0]])
print(model.predict(features))
```

```
['Fraud']
```

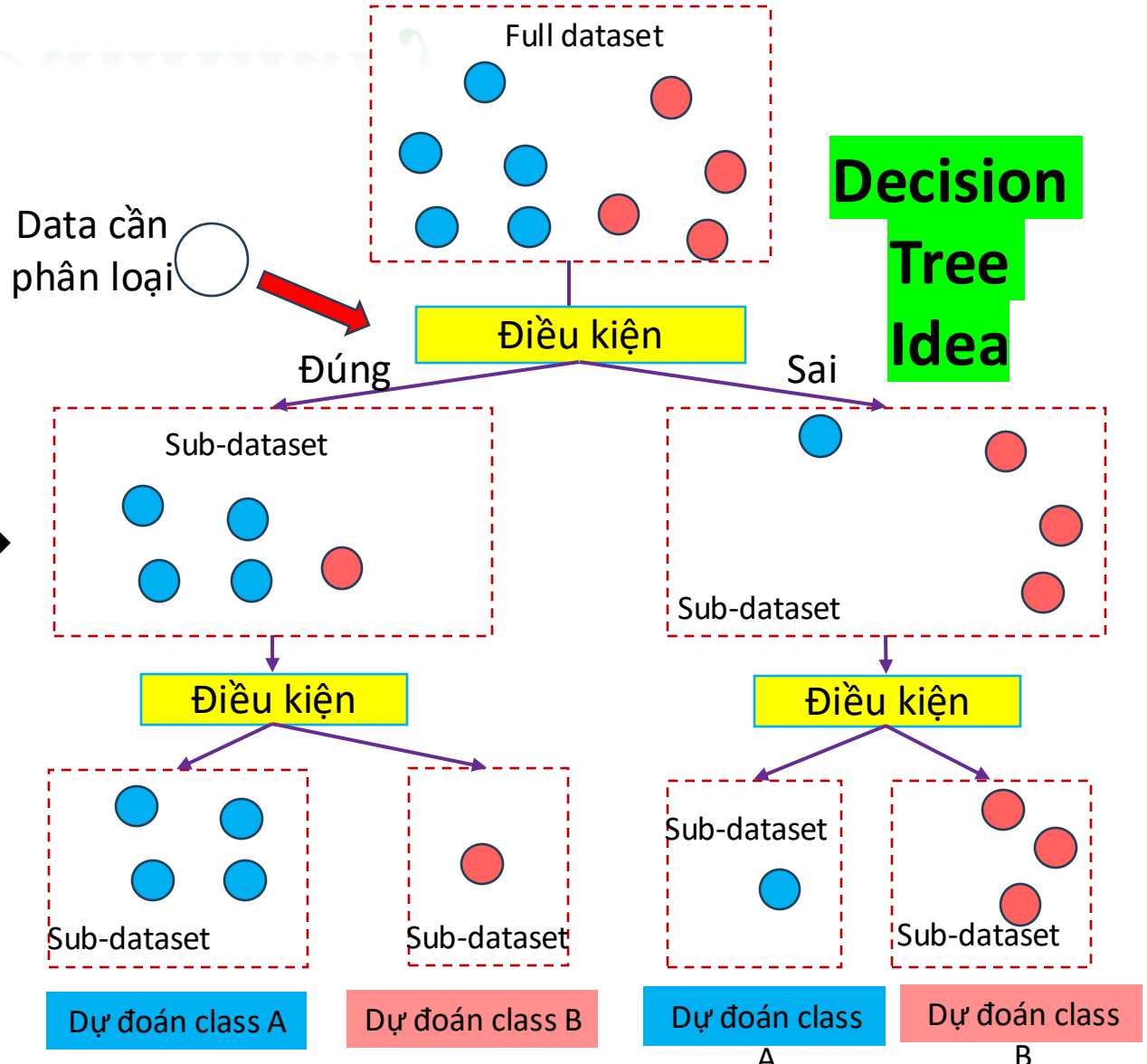
Outline

- Introduction to Decision Tree
- Classification Tree with GINI
- Classification Tree with Entropy
- Examples
- Summary

Summary



KNN has some drawbacks and challenges, such as computational expense, slow speed, for large datasets




Thank
you