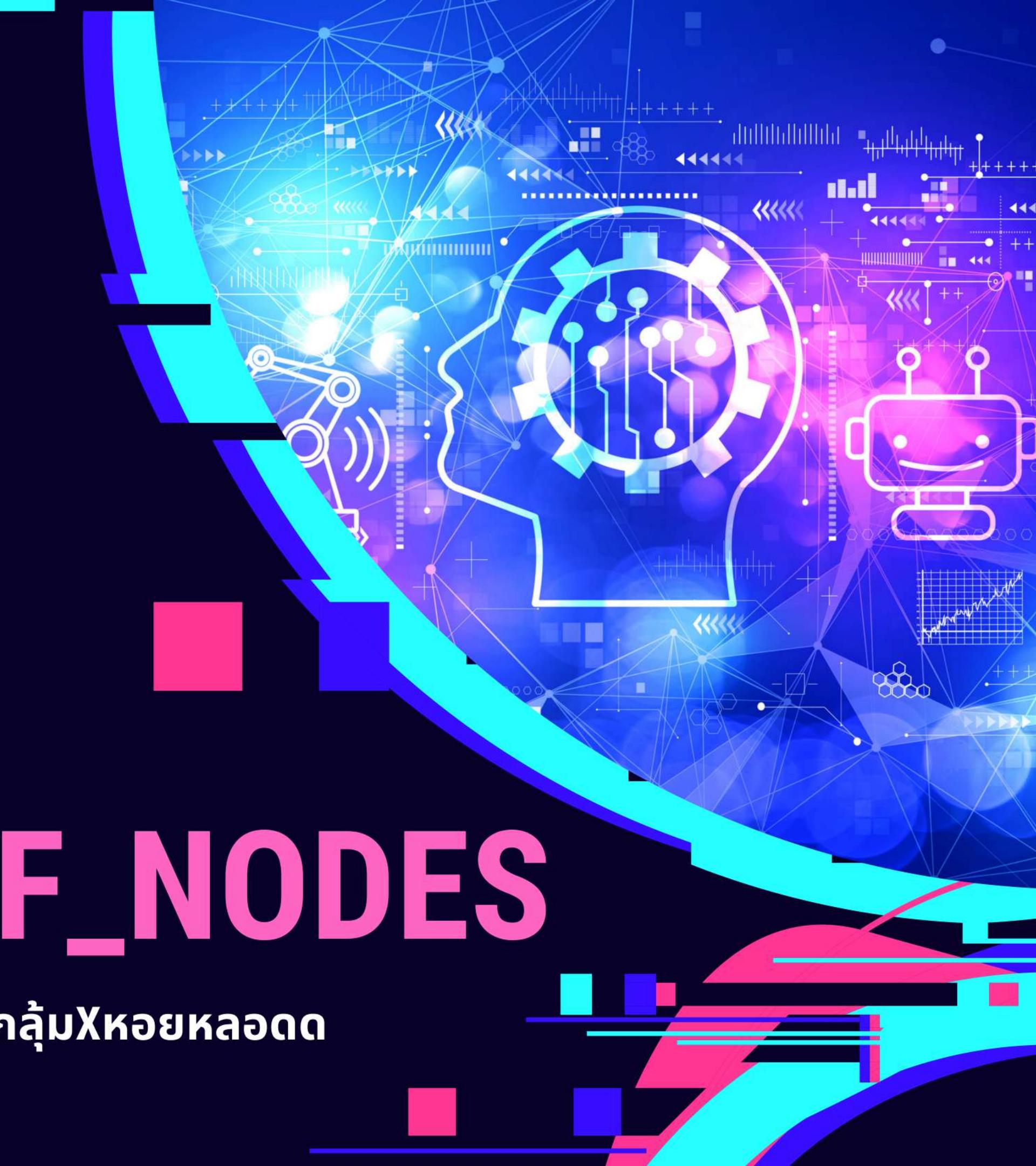


CRITERION AND MAX_LEAF_NODES

กลุ่ม กลุ่ม X หอยหลอดด



กลุ่ม กลุ่ม X หอยหลอดด

- 643020501-6 นายตะวัน เป้าหล่อเพชร
- 643021260-7 นางสาวกิตติลักษณ์ ลาดโอม
- 643021261-5 นางสาวจารุพร การร้อย
- 643021263-1 นางสาวชนเมษนก อั้งคุระ
- 643021265-7 นายธนาธิป อินทรคีรี
- 643021266-5 นางสาวธิตพร ใจเอื้อ
- 643021268-1 นายพุทธิพงศ์ ย่างนอก
- 643021273-8 นายศตวรรษ มูลสันเทียะ

■ CRITERION

ค่าวัดในการ Split โดยตัดสินว่าข้อมูลตัวอย่างนั้นจะถูกจัดประเภทไปยังกลุ่มใดโดยค่าไหนมีให้เลือกหลากหลาย เช่น gini entropy logloss

■ MAX_LEAFT_NODES

การกำหนดจำนวนสูงสุดของโหนดในต้นไม้ตัดสินใจ โดยกำหนดให้การ Splitting หยุดลงเมื่อจำนวนโหนดในถึงขีดจำกัดที่กำหนดไว้

CRITERION



GINI

$$\square gini(D) = 1 - \sum_{j=1}^n p_j^2$$

Gini Impurity เป็นตัววัดความ "ไม่บริสุทธิ์" ของข้อมูลในโโนเดล Decision Tree หมายถึงจำนวนข้อมูลที่มี class label ผสมกันอยู่มากน้อยเพียงใด คะแนน Gini ก็ต่อําแสดงถึงกลุ่มข้อมูลที่ "บริสุทธิ์" มากกว่า และเป็นการแบ่งกลุ่มที่ดีกว่า

GINI

$$\square gini_A(D) = \frac{|D_1|}{|D|} gini(D_1) + \frac{|D_2|}{|D|} gini(D_2)$$

เป็นค่า gini ของ feature ยิ่งน้อยยิ่งดี

$$\square \Delta gini(A) = gini(D) - gini_A(D)$$

Reduction in Impurity คำนวณจาก ความแตกต่าง ของ ค่าความไม่บริสุทธิ์ ของ node หลัก (parent node) กับ ค่าความไม่บริสุทธิ์ ของ node ย่อย (child nodes)

Entropy

$$Info(D) = - \sum_{i=1}^m p_i \log_2(p_i)$$

Entropy ของข้อมูล เป็นตัววัด ความไม่แน่นอน ของข้อมูล หรือพูดอีกนัยหนึ่งคือ เป็นตัววัด ความสุ่ม ของข้อมูล

ค่า Entropy สูง หมายถึง ข้อมูลมีความไม่แน่นอนสูง หรือมีความสุ่มสูง

ค่า Entropy ต่ำ หมายถึง ข้อมูลมีความแน่นอนสูง หรือ มีความสุ่มต่ำ

Entropy

$$Info_A(D) = \sum_{j=1}^v \frac{|D_j|}{|D|} \times Info(D_j)$$

เป็นค่า Entropy ของ feature ยิ่งน้อยยิ่งดี

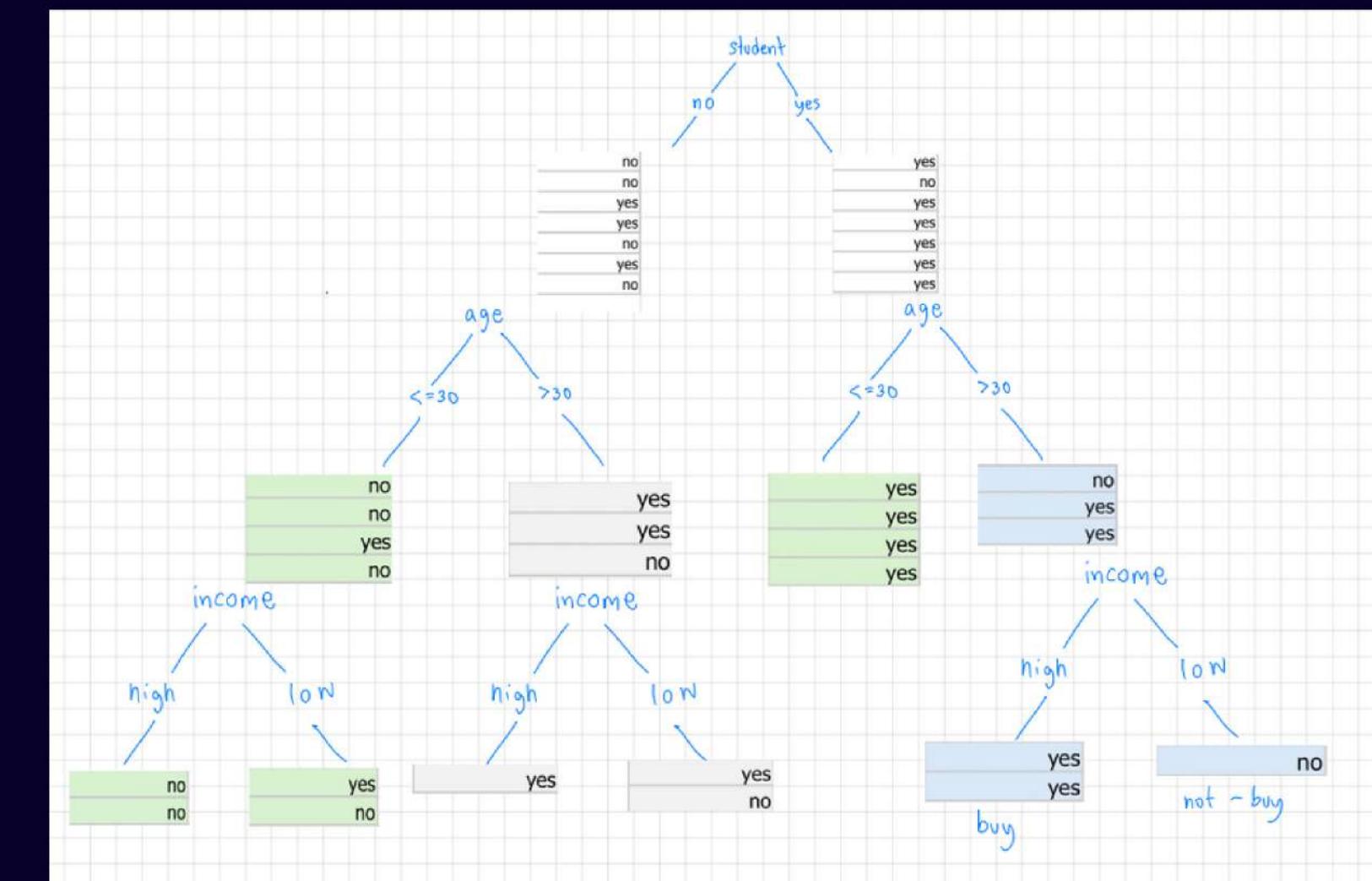
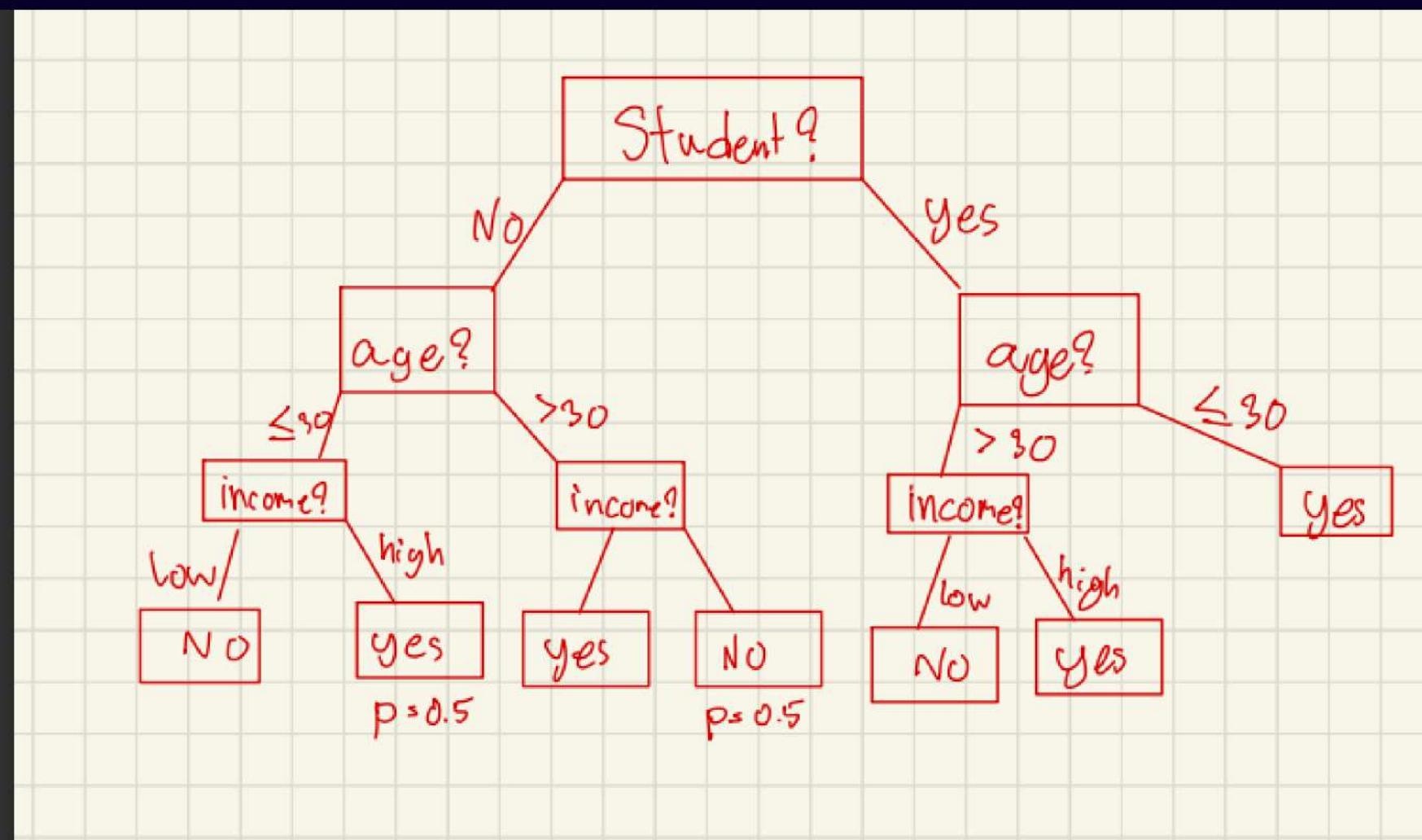
$$Gain(A) = Info(D) - Info_A(D)$$

Gain คำนวณจาก ความแตกต่าง ของ ค่าความไม่แน่นอน ของ node หลัก (parent node) กับ ค่าความไม่แน่นอน ของ node ย่อย (child nodes)

TRAINING DATA SET: WHO BUYS COMPUTER?

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
>30	high	no	fair	yes
<=30	low	no	fair	yes
<=30	high	yes	fair	yes
>30	low	yes	excellent	no
>30	high	yes	excellent	yes
<=30	low	no	fair	no
<=30	low	yes	fair	yes
<=30	low	yes	fair	yes
<=30	low	yes	excellent	yes
>30	low	no	excellent	yes
>30	high	yes	fair	yes
>30	low	no	excellent	no

Full Growth tree



$$gini(D) = 0.459$$

$$gini_{age}(D) = 0.457 ; \Delta gini_{age}(D) > 0.02$$

$$gini_{income}(D) = 0.457 ; \Delta gini_{income}(D) = 0.02$$

$$gini_{credit}(D) = 0.428 ; \Delta gini_{credit}(D) = 0.031$$

$$gini_{student}(D) = 0.366 ; \Delta gini_{student}(D) = 0.093$$

$$Gain(age) = Info(D) - Info_{age}(D) = 0.940 - 0.938 = 0.002$$

$$Gain(income) = Info(D) - Info_{income}(D) = 0.940 - 0.938 = 0.002$$

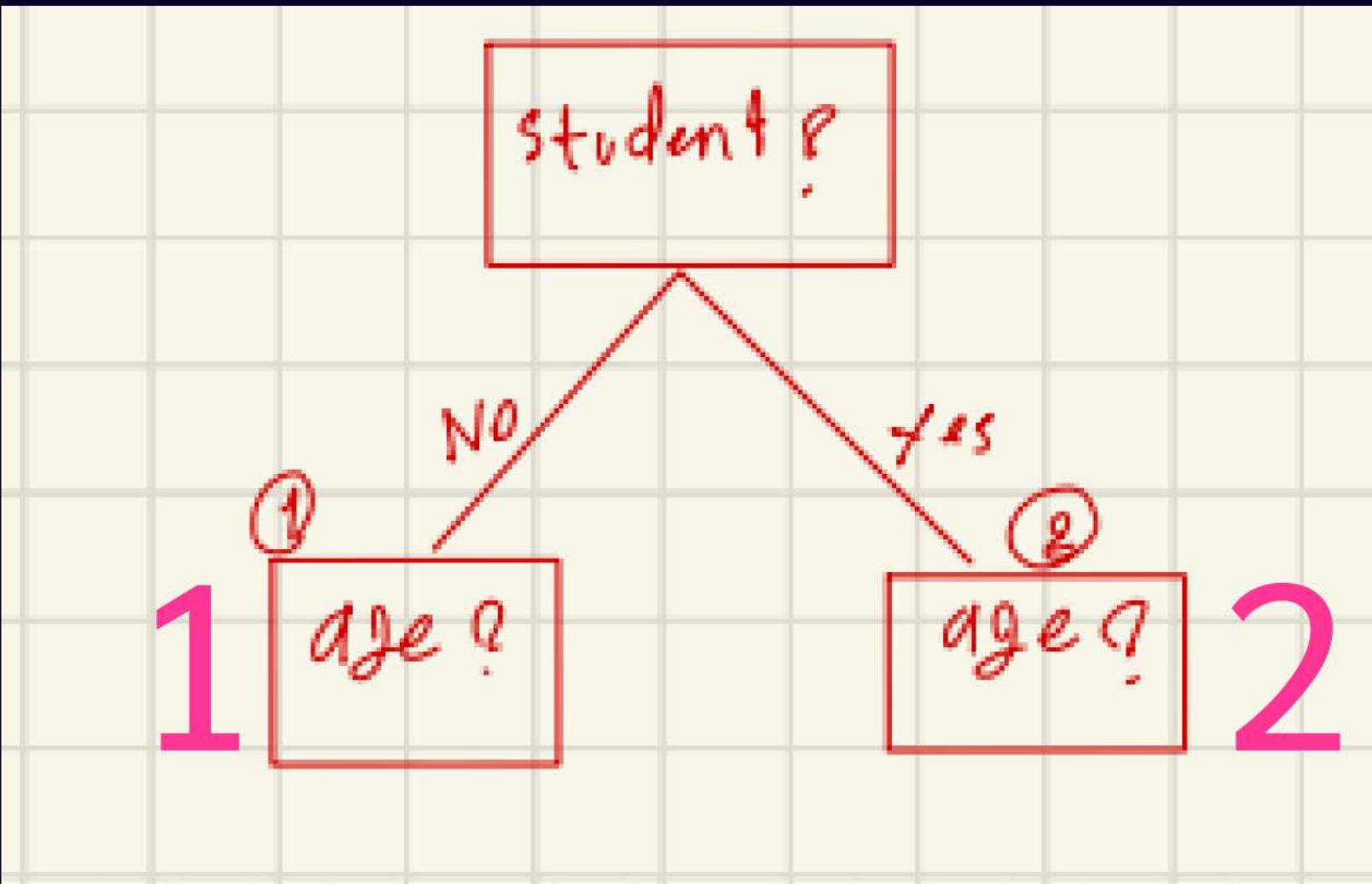
$$Gain(student) = Info(D) - Info_{student}(D) = 0.940 - 0.788 = 0.152$$

$$\begin{aligned} Gain(credit_rating) &= Info(D) - Info_{credit_rating}(D) \\ &= 0.940 - 0.892 = 0.048 \end{aligned}$$

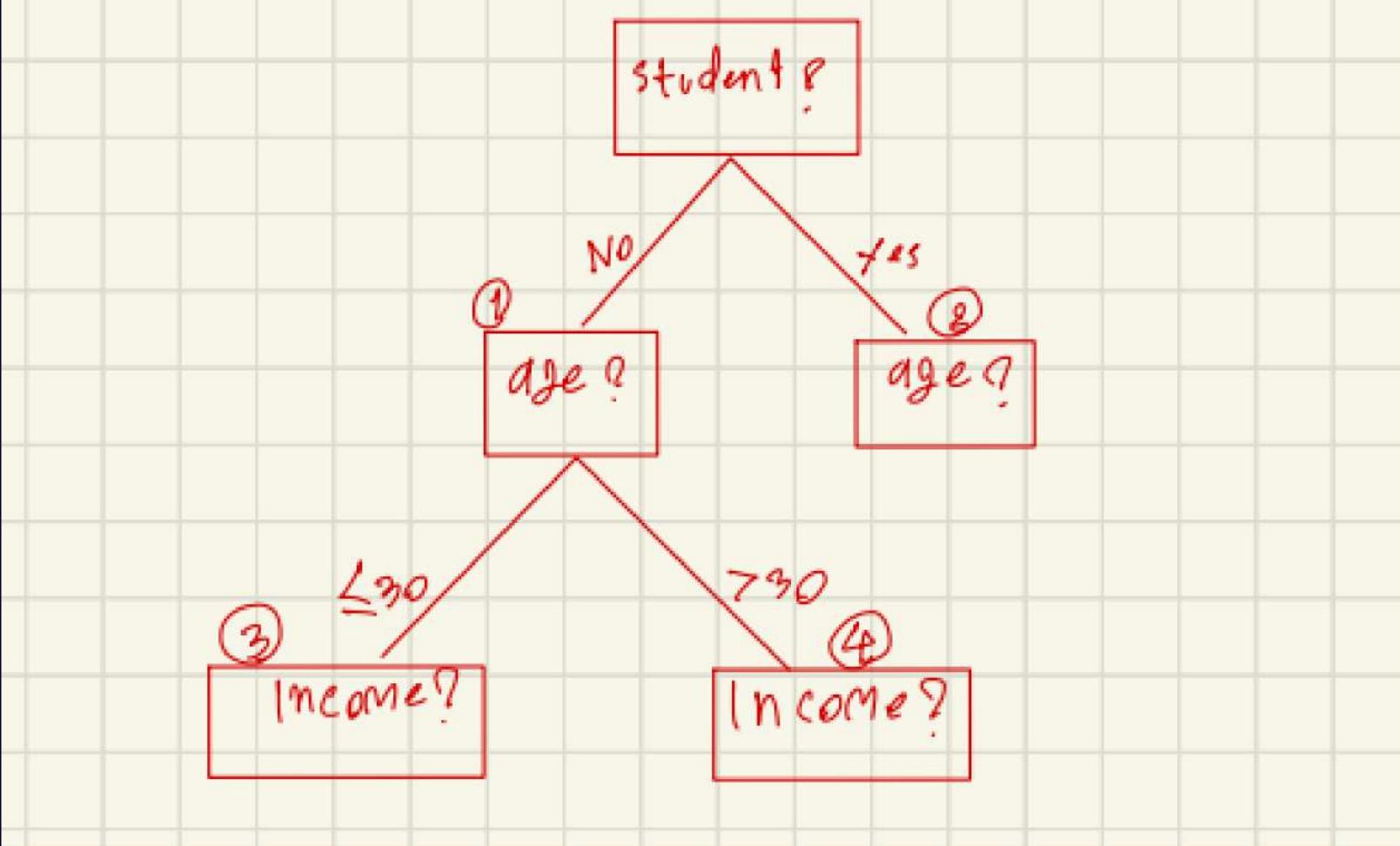
∴ Root Node is student

การโটของต้นไม้

Entropy vs gini



node 1 $\Delta gini(\text{Student}=\text{no}, \text{age}) = 0.0854$



node 2 $\Delta gini(\text{Student}=\text{yes}, \text{age}) = 0.0547$

แบ่ง node 1 (ทางซ้าย) ก่อน

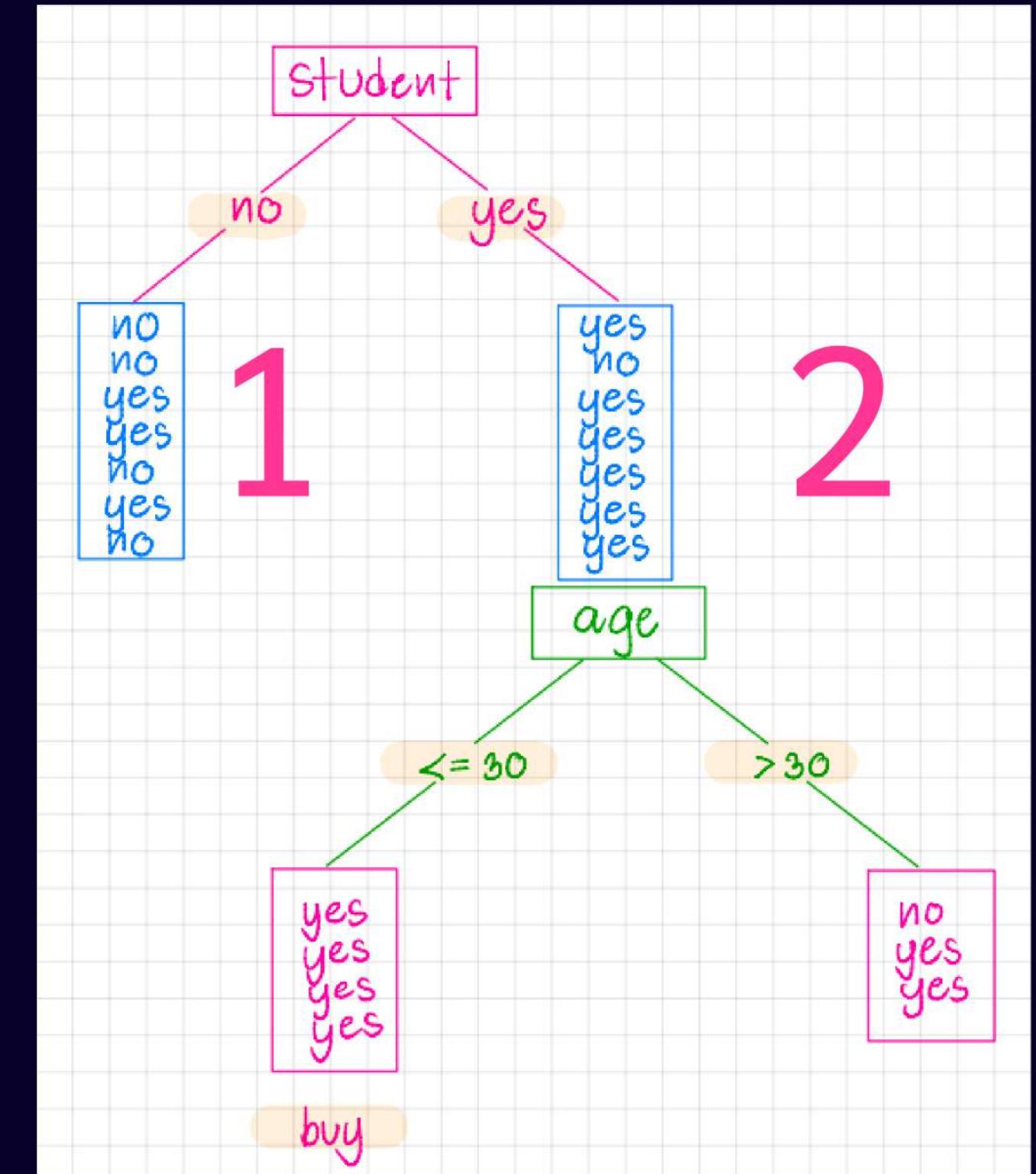
การตัดข้อมูลต้นไม้

Entropy vs gini

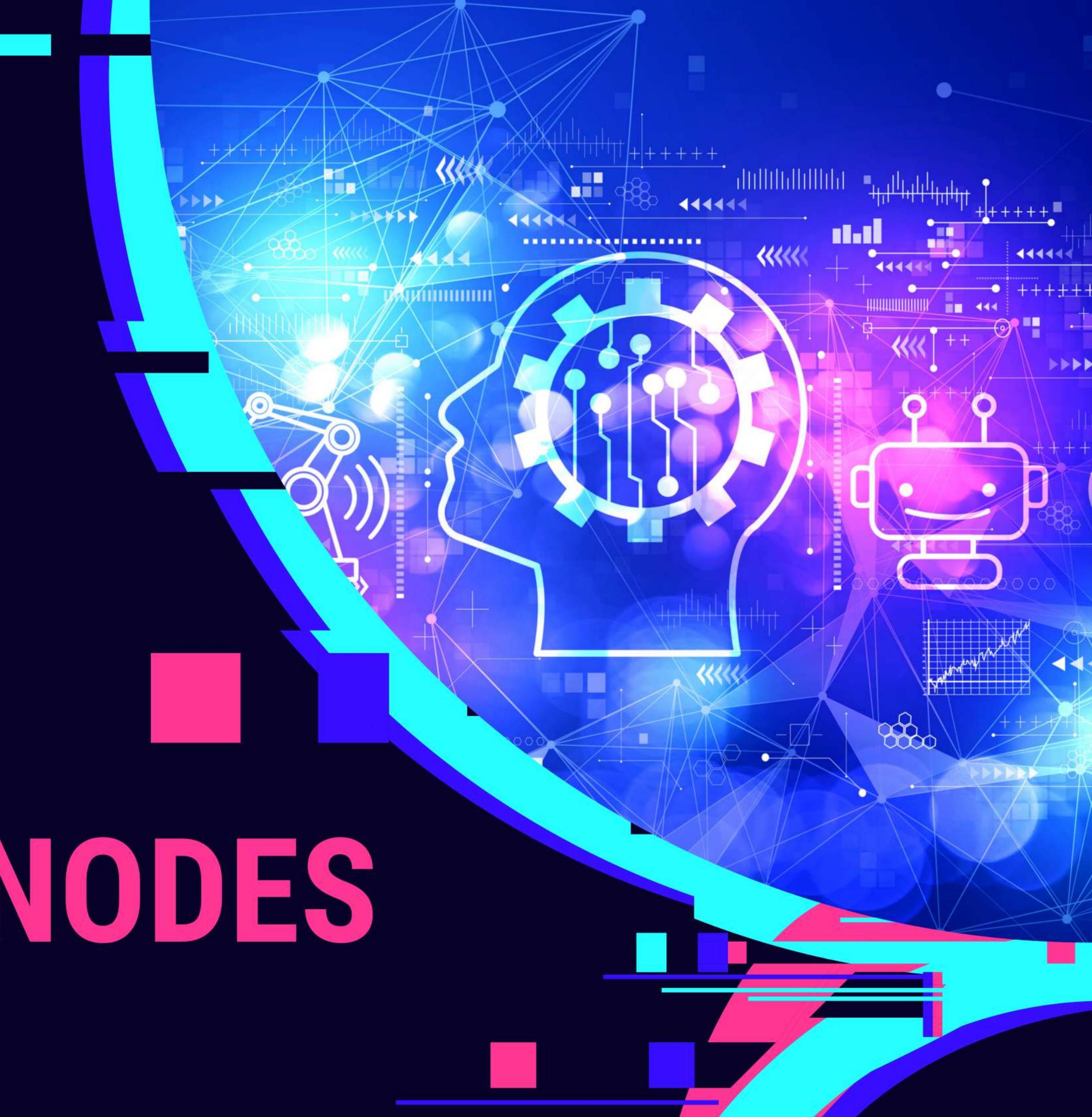
node 1 gain(Student=no, age) = 0.1282

node 2 gain(Student=yes, age) = 0.1986

แบ่ง node 2 (ทางขวา) ก่อน



MAX.LEAF.NODES



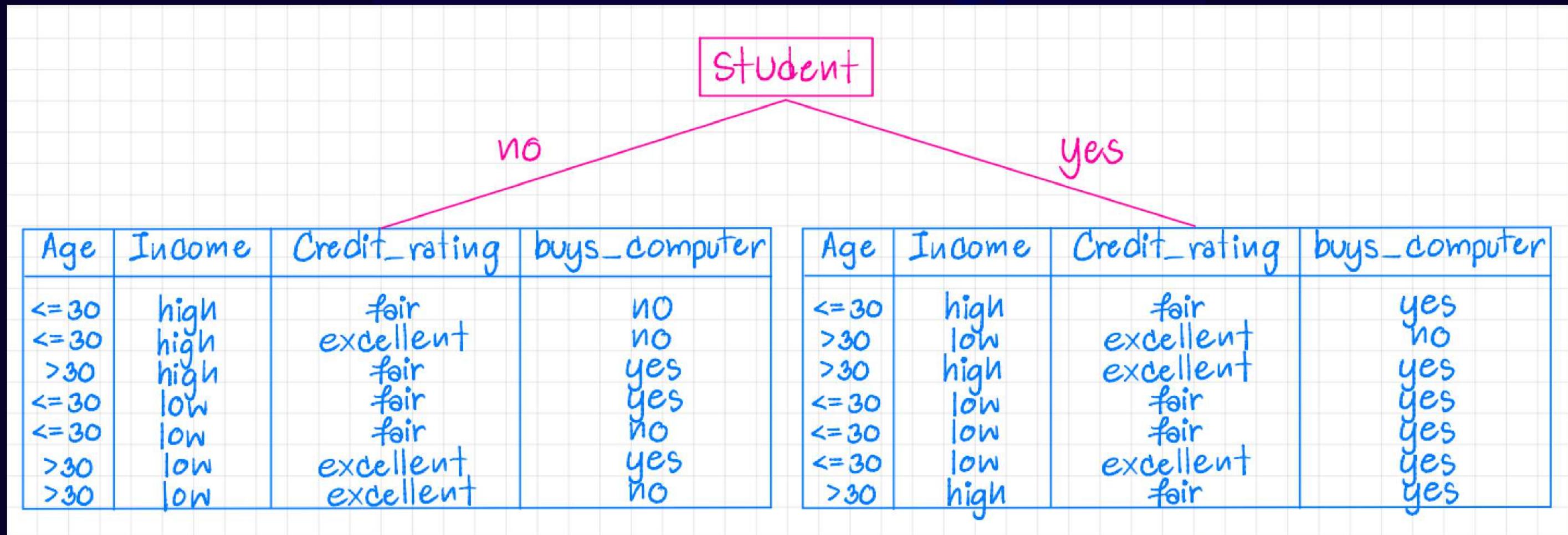
TRAINING DATA SET: WHO BUYS COMPUTER?

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
>30	high	no	fair	yes
<=30	low	no	fair	yes
<=30	high	yes	fair	yes
>30	low	yes	excellent	no
>30	high	yes	excellent	yes
<=30	low	no	fair	no
<=30	low	yes	fair	yes
<=30	low	yes	fair	yes
<=30	low	yes	excellent	yes
>30	low	no	excellent	yes
>30	high	yes	fair	yes
>30	low	no	excellent	no

TRAINING DATA SET: WHO BUYS COMPUTER?

จาก data set เราจะทำการ training data set เพื่อทำนายว่า
บุคคลที่มีลักษณะอย่างไร จึงจะตัดสินใจเลือกซื้อคอมพิวเตอร์

MAX_LEAF_NODES = 2



MAX_LEAF_NODES = 2

เมื่อ กำหนด max leaf nodes = 2 จะได้ decision tree ดังภาพ
โดย root node คือ student
ซึ่ง Student จำแนกเป็น yes กับ no คือ เป็นนักเรียน กับ ไม่เป็น
นักเรียน
ซึ่งคำนวณมากจากค่า gain

INFORMATION GAINED

- Class P: buys_computer = "yes" → 9
- Class N: buys_computer = "no" → 5

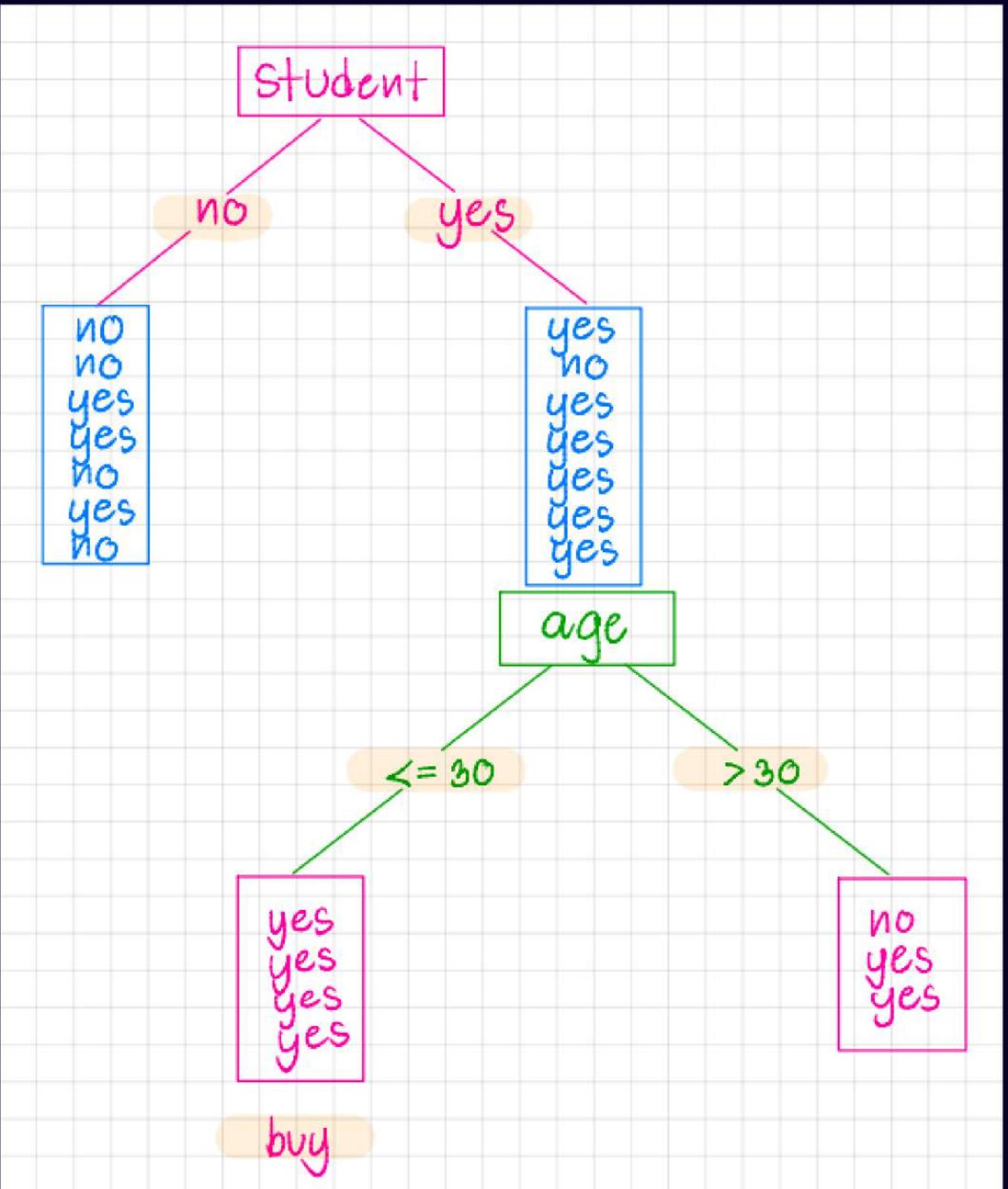
$$\text{Info}(D) = -\frac{9}{14} \log_2 \left(\frac{9}{14} \right) - \frac{5}{14} \log_2 \left(\frac{5}{14} \right) \\ = 0.940$$

$$\begin{aligned}\text{Gain}(\text{age}) &= \text{Info}(D) - \text{Info}_{\text{age}}(D) = 0.940 - 0.938 = 0.002 \\ \text{Gain}(\text{income}) &= \text{Info}(D) - \text{Info}_{\text{income}}(D) = 0.940 - 0.938 = 0.002 \\ \text{Gain}(\text{student}) &= \text{Info}(D) - \text{Info}_{\text{student}}(D) = 0.940 - 0.788 = 0.152 \\ \text{Gain}(\text{credit_rating}) &= \text{Info}(D) - \text{Info}_{\text{credit_rating}}(D) \\ &= 0.940 - 0.892 = 0.048\end{aligned}$$

INFORMATION GAINED

จากการคำนวณ information gained ของแต่ละ features จะได้ information gained ที่มีค่ามากที่สุด เท่ากับ 0.152 ดังนั้น จึงได้ student เป็น root node

MAX_LEAF_NODES = 3



MAX_LEAF_NODES = 3

เมื่อ กำหนด max leaf nodes = 3 จะได้ decision tree ดังภาพ
โดย root node คือ student
และจะเห็นว่า ใน แต่ละเส้นทางด้านขวา นั่นคือทางผ่าน student :
yes ซึ่งแต่ละเส้นทางของ age ดังภาพ โดยเราจะใช้การเก็บ
information gain ของแต่ features ที่มีค่าสูงสุด ทั้งทางผ่าน
student : no และ student : yes มาเก็บกัน ดังนี้

$$\text{Gain}(\text{age}) = \text{Info}_{\text{student:yes}}(D) - \text{Info}_{\text{age}}(D) = 0.592 - 0.393 = 0.199 \times$$

$$\text{Gain}(\text{income}) = \text{Info}_{\text{student:yes}}(D) - \text{Info}_{\text{income}}(D) = 0.592 - 0.463 = 0.129$$

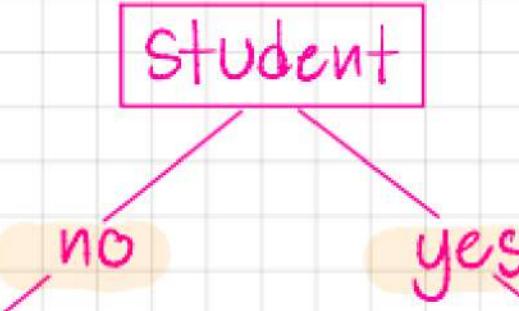
$$\begin{aligned}\text{Gain}(\text{credit_rating}) &= \text{Info}_{\text{student:yes}}(D) - \text{Info}_{\text{credit_rating}}(D) \\ &= 0.592 - 0.393 = 0.199 \times\end{aligned}$$

$$\begin{aligned}\text{Gain}(\text{student: no, age}) &= \text{Info}_{\text{student: no}}(D) - \text{Info}_{\text{age}}(D) \\ &= 0.985 - 0.128 = 0.128 \times\end{aligned}$$

$$\begin{aligned}\text{Gain}(\text{student: no, income}) &= \text{Info}_{\text{student: no}}(D) - \text{Info}_{\text{income}}(D) \\ &= 0.985 - 0.965 = 0.02\end{aligned}$$

$$\begin{aligned}\text{Gain}(\text{student: no, credit rating}) &= \text{Info}_{\text{student: no}}(D) - \text{Info}_{\text{credit rating}}(D) \\ &= 0.985 - 0.965 = 0.02\end{aligned}$$

Gain(student: yes)	0.199
Gain(student: no)	0.128



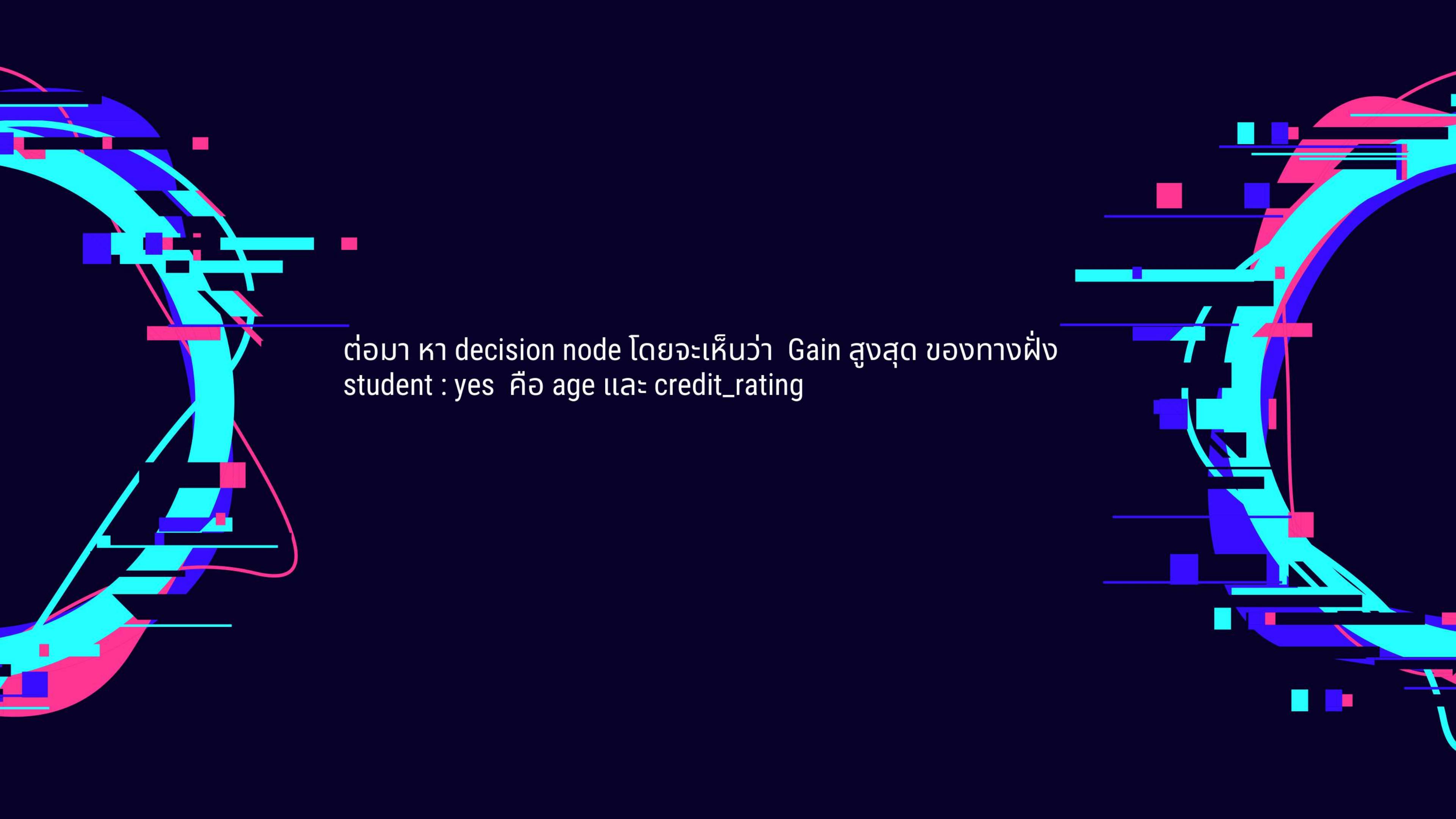
MAX_LEAF_NODES = 3

จะเห็นว่า Gain สูงสุด ของทางผัง student : yes คือ age มีค่า เท่ากับ 0.199 และ Gain สูงสุด ของทางผัง student : no คือ age มีค่าเท่ากับ 0.128 เมื่อนำมาคำ Gain สูงสุดของแต่ละผังมา เทียบกัน จะได้ว่า Gain ผัง student : yes มีค่ามากกว่า จึง ตัดสินใจแตกใบ ผังทางขวา

$$\text{Gain}(\text{age}) = \text{Info}_{\text{student:yes}}(D) - \text{Info}_{\text{age}}(D) = 0.592 - 0.393 = 0.199 \times$$

$$\text{Gain}(\text{income}) = \text{Info}_{\text{student:yes}}(D) - \text{Info}_{\text{income}}(D) = 0.592 - 0.463 = 0.129$$

$$\begin{aligned}\text{Gain}(\text{credit_rating}) &= \text{Info}_{\text{student:yes}}(D) - \text{Info}_{\text{credit_rating}}(D) \\ &= 0.592 - 0.393 = 0.199 \times\end{aligned}$$



ต่อมา หา decision node โดยจะเห็นว่า Gain สูงสุด ของກາງຝັ້ງ
student : yes គື່ອ age ແລະ credit_rating

$\text{Gain}(\text{student: yes}, \text{age} > 30, \text{income})$

$$= \text{Info}_{\text{student: yes}, \text{age} > 30}^{(D)} - \text{Info}_{\text{student: yes}, \text{age} > 30, \text{income}}^{(D)}$$
$$= 0.918 - 0 = 0.918$$

$\text{Gain}(\text{student: yes}, \text{age} > 30, \text{credit rating})$

$$= \text{Info}_{\text{student: yes}, \text{age} > 30}^{(D)} - \text{Info}_{\text{student: yes}, \text{age} > 30, \text{credit_rating}}^{(D)}$$
$$= 0.918 - 0.667 = 0.251$$

$\text{Gain}(s: \text{yes}, c = \text{excellent}, \text{age})$

$$= \text{Info}_{s: \text{yes}, c = \text{excellent}}^{(D)} - \text{Info}_{s: \text{yes}, c = \text{excellent}, \text{age}}^{(D)}$$
$$= 0.918 - 0.667 = 0.251$$

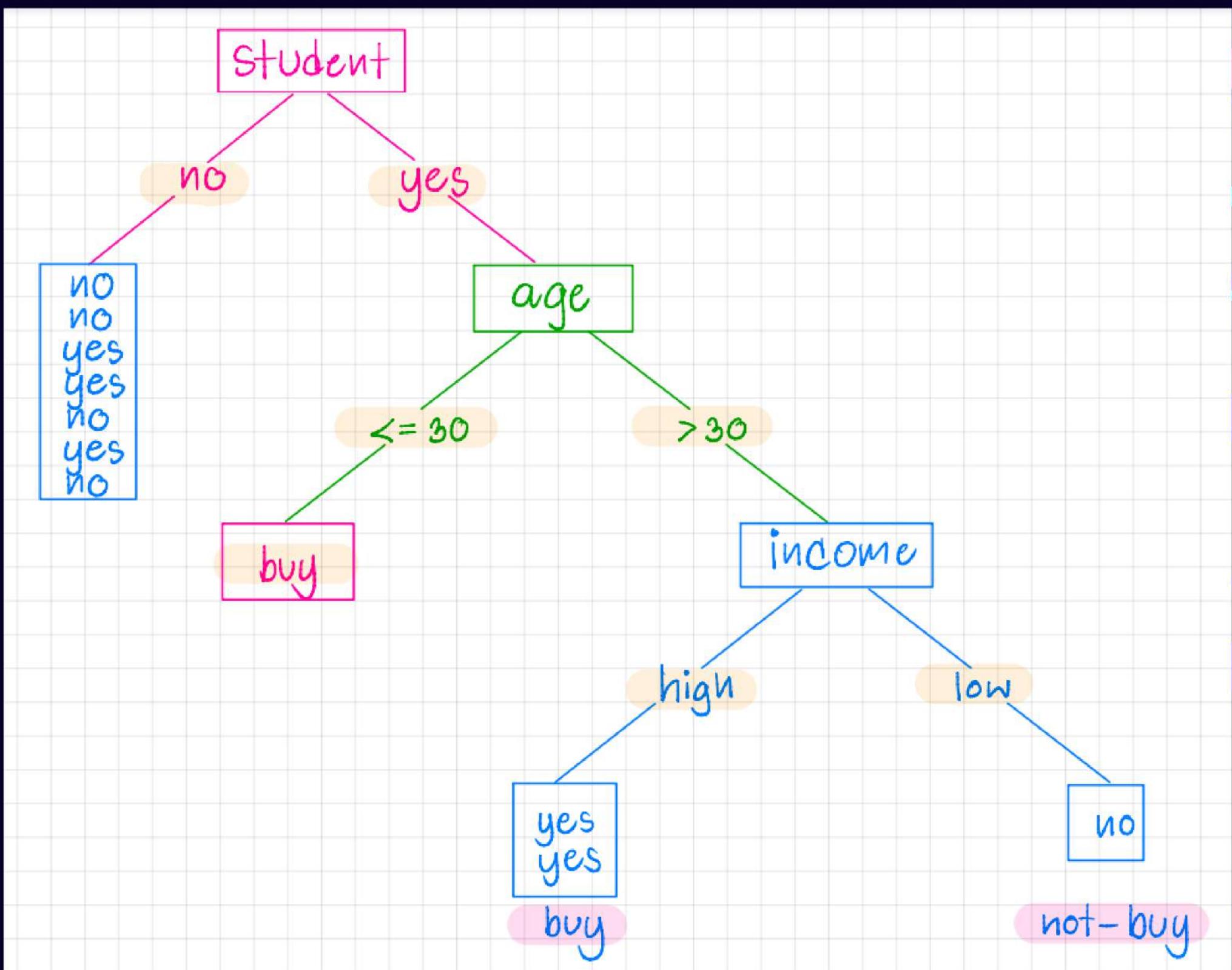
$\text{Gain}(s: \text{yes}, c = \text{excellent}, \text{income})$

$$= \text{Info}_{s: \text{yes}, c = \text{excellent}}^{(D)} - \text{Info}_{s: \text{yes}, c = \text{excellent}, \text{income}}^{(D)}$$
$$= 0.918 - 0.667 = 0.251$$

$\text{Gain}(\text{student: yes}, \text{age} > 30, \text{income})$	0.918
$\text{Gain}(\text{student: yes}, \text{age} > 30, \text{credit rating})$	0.251
$\text{Gain}(s: \text{yes}, c = \text{excellent}, \text{age})$	0.251
$\text{Gain}(s: \text{yes}, c = \text{excellent}, \text{income})$	0.251

จะเห็นว่า Gain สูงสุด คือ age มีค่าเท่ากับ 0.918 และ Gain สูงสุดของ credit_rating มีค่าเท่ากับ 0.251 เมื่อนำมาคำ Gain สูงสุดของแต่ละ features มาเทียบกัน จะได้ว่า Gain ของ age มีค่ามากกว่า จึงได้ age เป็น decision node

MAX_LEAF_NODES = 4



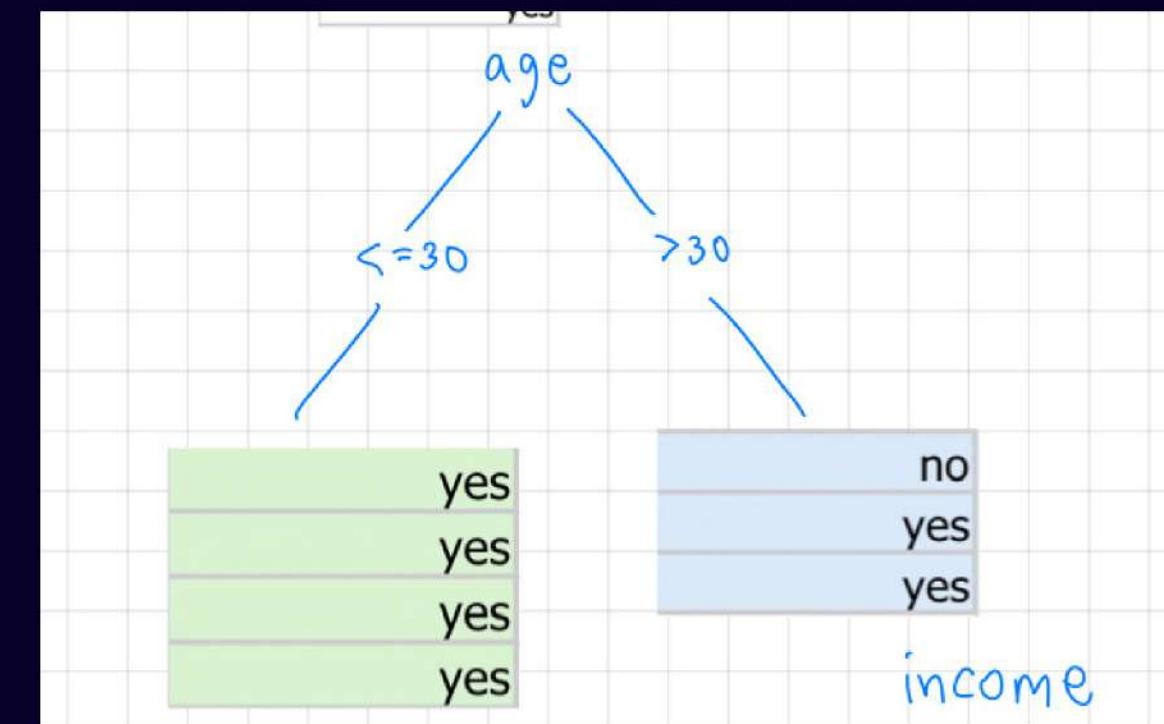
MAX_LEAF_NODES = 4

เมื่อกำหนด max leaf node = 4 จะได้ decision tree ดังภาพ
จะเห็นว่า $age \leq 30$ ไม่สามารถหาต่อได้ เนื่องจากสามารถ^{ทำนายได้แล้ว}
และจะเห็นว่าจะแตกใบ กี่ $age > 30$ ของทางผั้ง student : yes
ซึ่งแตกใบเป็น income เป็น decision node โดยดูจากค่า Gain

Step 1 : an expected info gain
classify them

- Class P: buys_computer = "yes" $\rightarrow 4$
- Class N: buys_computer = "no" $\rightarrow 0$

Info_{student}: yes, age ≤ 30 (D)
 $= -\frac{4}{4} \log_2 \left(\frac{4}{4}\right) - \frac{0}{4} \log_2 \left(\frac{0}{4}\right)$
 $= 0$



AGE ≤ 30

AGE > 30

Gain(student: yes, age ≤ 30 , income)

$$\begin{aligned} &= \text{Info}_{\text{student: yes, age } \leq 30} \text{ (D)} - \text{Info}_{\text{student: yes, age } \leq 30, \text{ income}} \text{ (D)} \\ &= 0.918 - 0 = 0.918 \end{aligned}$$

Gain(student: yes, age > 30 , credit rating)

$$\begin{aligned} &= \text{Info}_{\text{student: yes, age } > 30} \text{ (D)} - \text{Info}_{\text{student: yes, age } > 30, \text{ credit_rating}} \text{ (D)} \\ &= 0.918 - 0.667 = 0.251 \end{aligned}$$

จาก decision node age

เนื่องจากคำนวนค่า expected info ของ $age \leq 30$ มีค่าเท่ากับ 0 จึงไม่สามารถแบ่งต่อได้แล้ว หมายความว่า ที่ $age \leq 30$ สามารถทำนายได้แล้ว คือ คนที่เป็นนักเรียน/นักศึกษา อายุน้อยกว่าหรือเท่ากับ 30 จะตัดสินใจ ซื้อ คอมพิวเตอร์

และ $age > 30$ เพื่อหา decision node ต่อไป โดยการคำนวนค่า gain

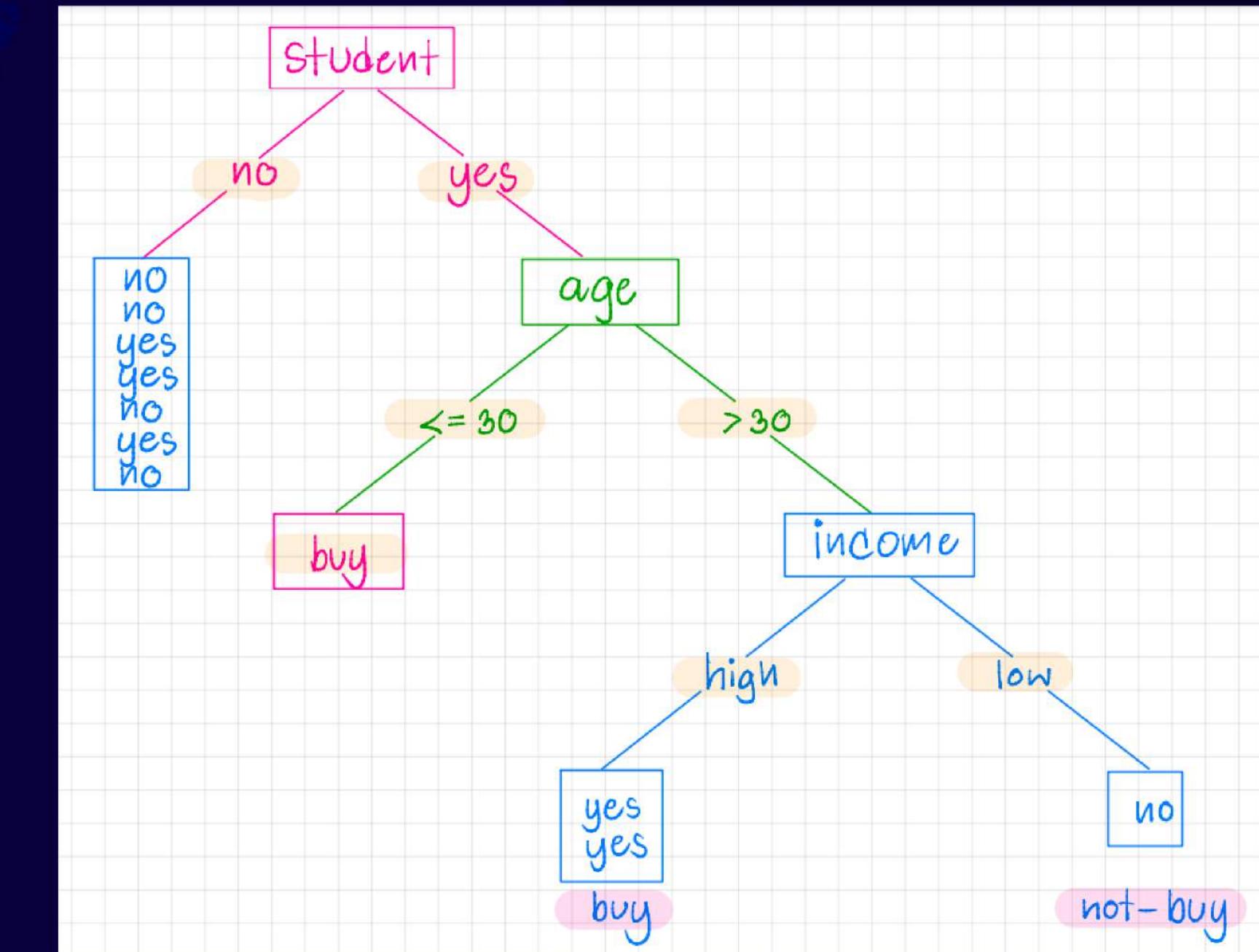
Gain(student: yes, age > 30, income)

0.918

Gain(student: yes, age > 30, credit rating)

0.251

student	yes	age	>30	income = low
age	income	student	credit_rating	buys_computer
>30	low	yes	excellent	no
student	yes	age	>30	income = high
age	income	student	credit_rating	buys_computer
>30	high	yes	excellent	yes
>30	high	yes	fair	yes

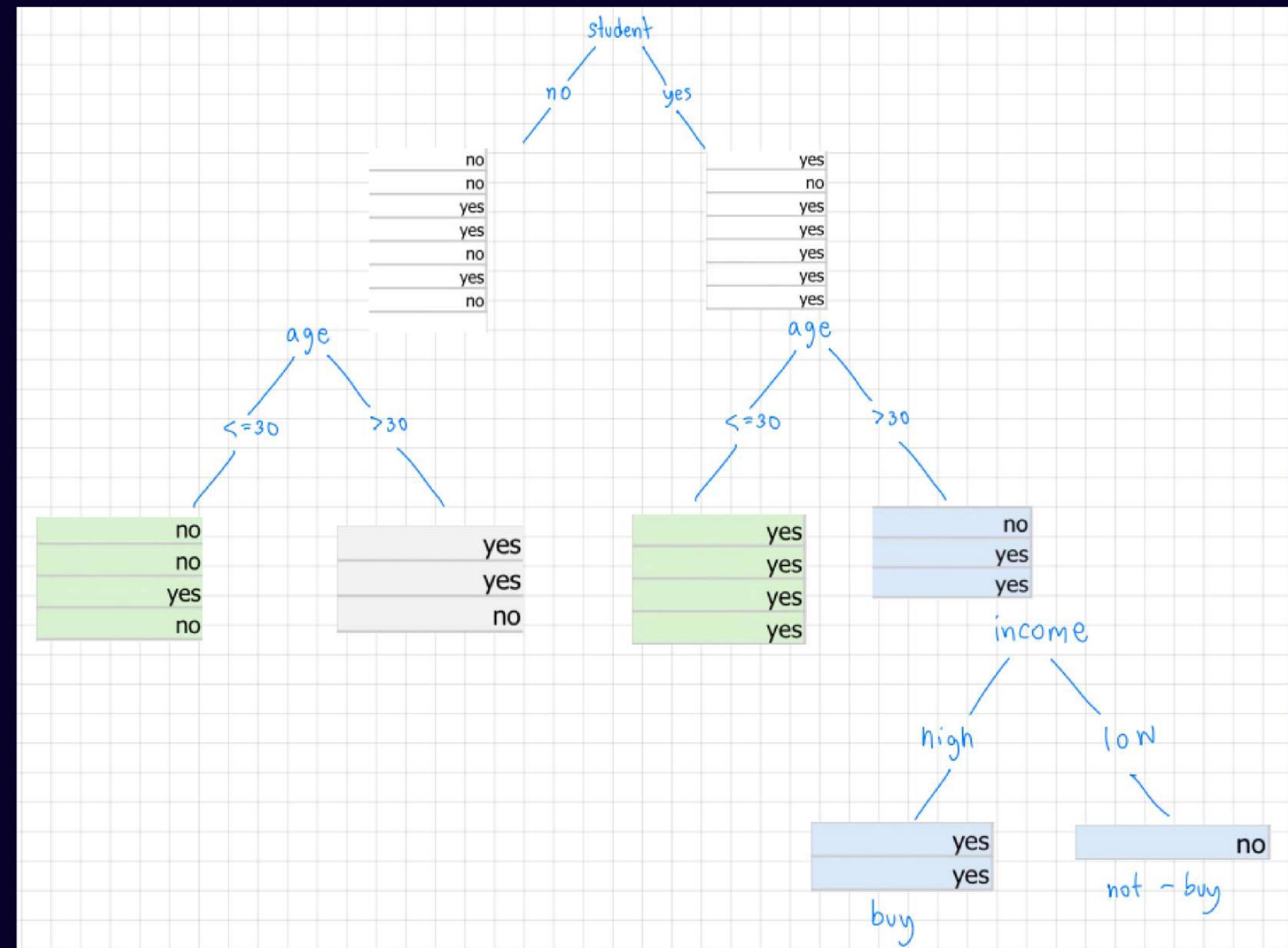


เมื่อนำค่า Gain ของ income กับ credit_rating มาเทียบกัน จะได้ว่า Gain ของ income มีค่ามากกว่า ดังนั้น decision node คือ income และจะเห็นได้ว่า decision node แบ่งคลาสได้แล้ว สามารถทำนายได้แล้ว ต่อ leaf node

income high ตอบ yes คือ buy
และ income low ตอบ no คือ not-buy

คือ คนที่เป็นนักเรียน/นักศึกษา อายุมากกว่า 30 ที่มีรายได้ สูง จะตัดสินใจ ซื้อ คอมพิวเตอร์ และคนที่เป็นนักเรียน/นักศึกษา อายุมากกว่า 30 ที่มีรายได้ น้อย จะตัดสินใจ ไม่ซื้อ คอมพิวเตอร์

MAX_LEAF_NODES = 5



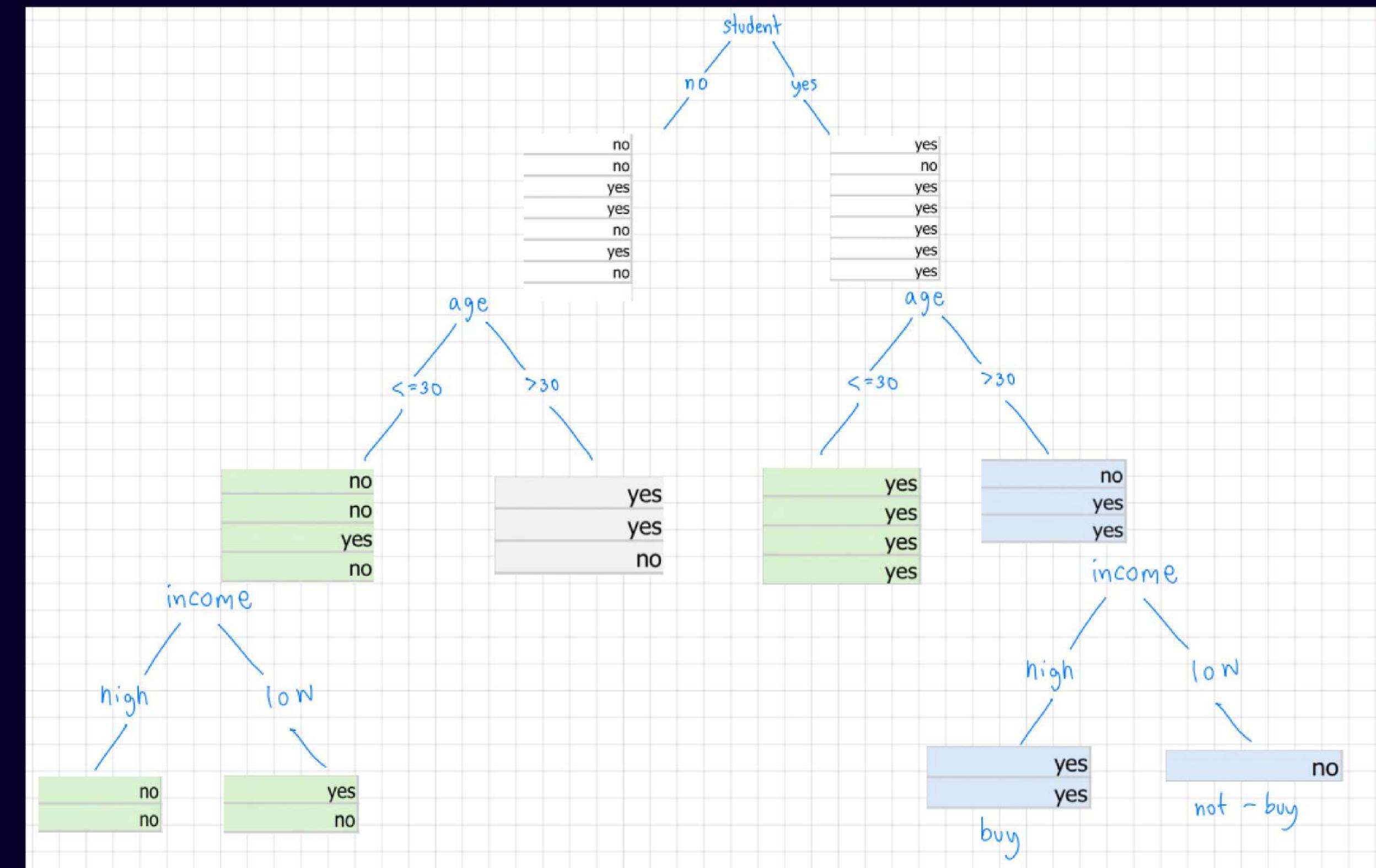
MAX_LEAF_NODES = 5

เมื่อกำหนด max leaf node = 5 จะเห็นว่า ในแต่กมหากางผั้ง student : no เนื่องจากค่า expected info ของ income, high และ income, low มีค่าเท่ากับ 0 จึงไม่สามารถแบ่งต่อได้แล้ว พอกำหนด max leaf node = 5 จึงตัดสินใจแต่กมหากางผั้ง student : no

$$\text{Gain}(\text{age}) = \text{Info}_{\text{Student: no}(D)} - \text{Info}_{\text{age}(D)} = 0.9852 - 0.8571 = 0.1281 \quad \checkmark$$
$$\text{Gain}(\text{income}) = \text{Info}_{\text{Student: no}(D)} - \text{Info}_{\text{income}(D)} = 0.9852 - 0.9649 = 0.0203$$
$$\text{Gain}(\text{credit_rating}) = \text{Info}_{\text{Student: no}(D)} - \text{Info}_{\text{credit_rating}(D)} = 0.9852 - 0.9649 = 0.0203$$

เมื่อคำนวณหาค่า Gain ของ age , income และ credit_rating_wu
ว่า Gain ของ age มีค่าสูงสุด ซึ่งมีค่าเท่ากับ 0.1281 ดังนี้
student : no จึงตัดสินใจแตกรายของ age

MAX_LEAF_NODES = 6



MAX_LEAF_NODES = 6

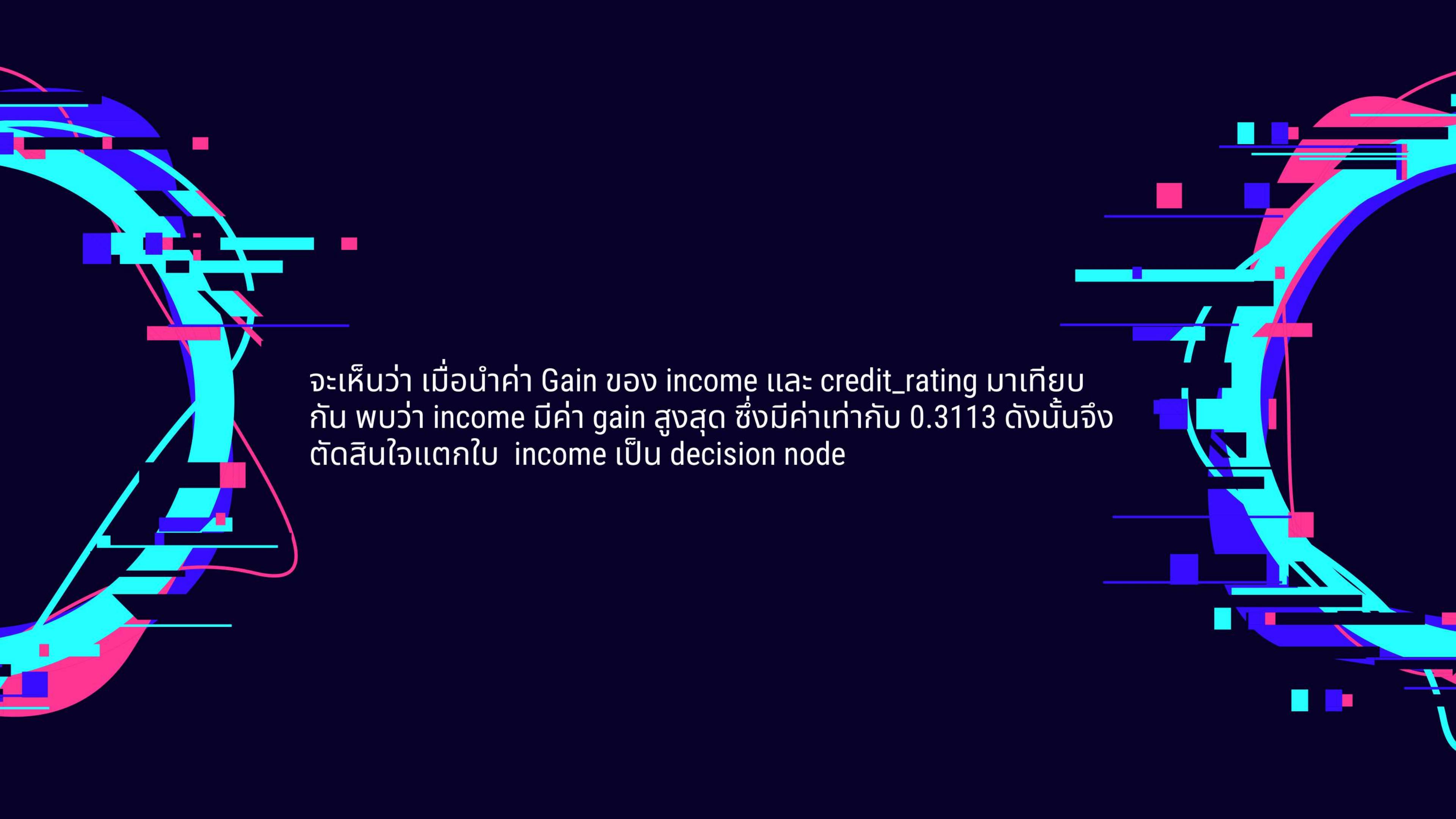
ต่อมาหาก กำหนด max leaf node = 6 จะแตกมาทางฝั่ง $age \leq 30$ ซึ่งตัดสินใจแตกไป จากการเทียบค่า gain ของทั้ง ทางฝั่ง $age \leq 30$ และ ฝั่ง $age > 30$ ดังนี้

$$\text{Gain}(\text{student: no}, \text{age} \leq 30, \text{income}) = \text{Info}_{\text{student: no}, \text{age} \leq 30: \text{no}(D)} - \text{Info}_{\text{income}}(D) = 0.8113 - 0.5 = 0.3113 \quad \checkmark$$

$$\text{Gain}(\text{student: no}, \text{age} \leq 30, \text{credit_rating}) = \text{Info}_{\text{student: no}, \text{age} \leq 30: \text{no}(D)} - \text{Info}_{\text{credit_rating}}(D) = 0.8113 - 0.6887 = 0.1226$$

$$\text{Gain}(\text{student: no}, \text{age} > 30, \text{income}) = \text{Info}_{\text{student: no}, \text{age} > 30: \text{no}(D)} - \text{Info}_{\text{income}}(D) = 0.918 - 0.667 = 0.251$$

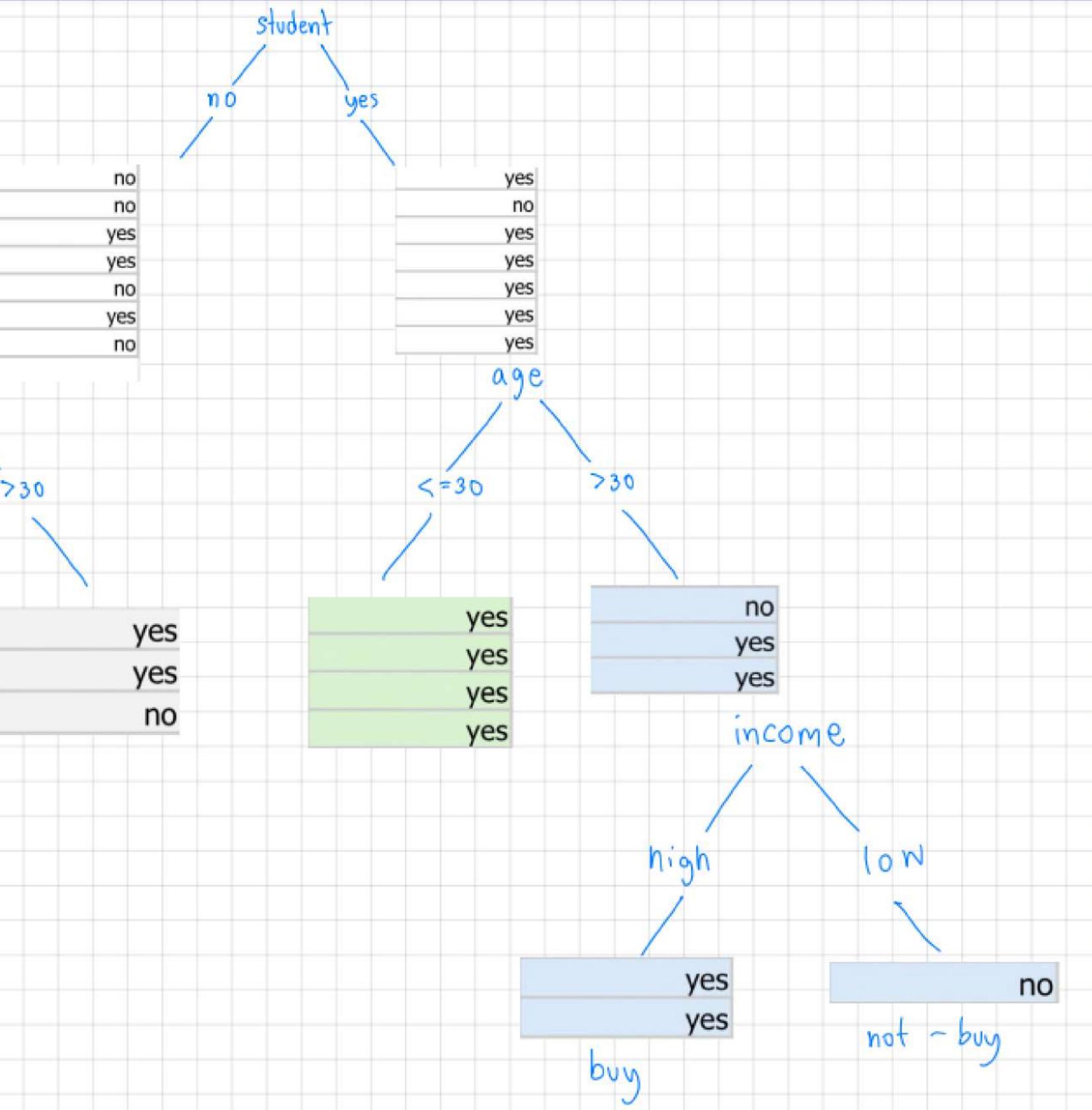
$$\text{Gain}(\text{student: no}, \text{age} > 30, \text{credit_rating}) = \text{Info}_{\text{student: no}, \text{age} > 30: \text{no}(D)} - \text{Info}_{\text{credit_rating}}(D) = 0.918 - 0.667 = 0.251$$

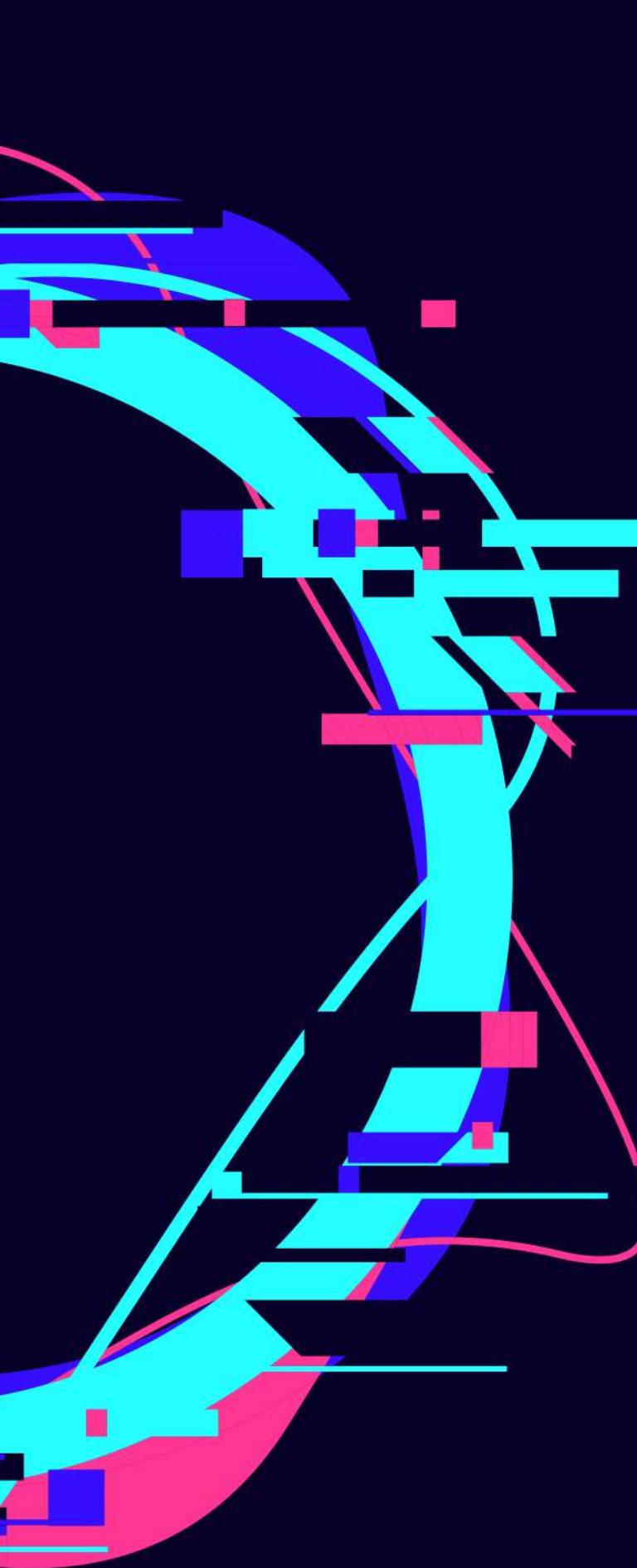


จะเห็นว่า เมื่อนำค่า Gain ของ income และ credit_rating มาเทียบกัน พบร่วมว่า income มีค่า gain สูงสุด ซึ่งมีค่าเท่ากับ 0.3113 ดังนั้นจึงตัดสินใจแตกรอบ income เป็น decision node

student	no	age <= 30	income = high	
age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no

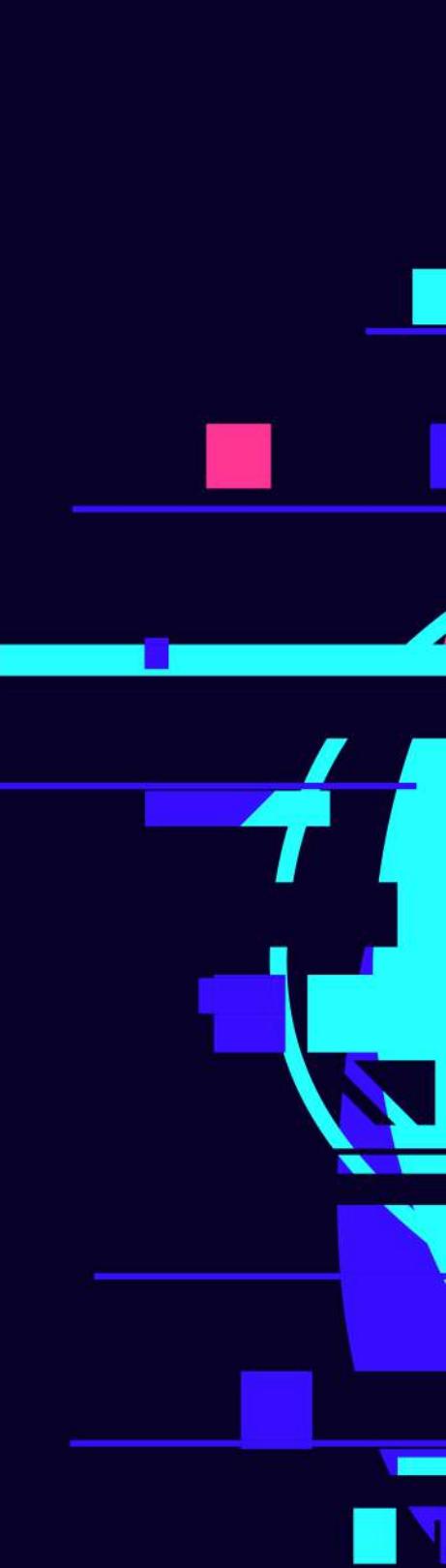
student	no	age <= 30	income = low	
age	income	student	credit_rating	buys_computer
<=30	low	no	fair	yes
<=30	low	no	fair	no





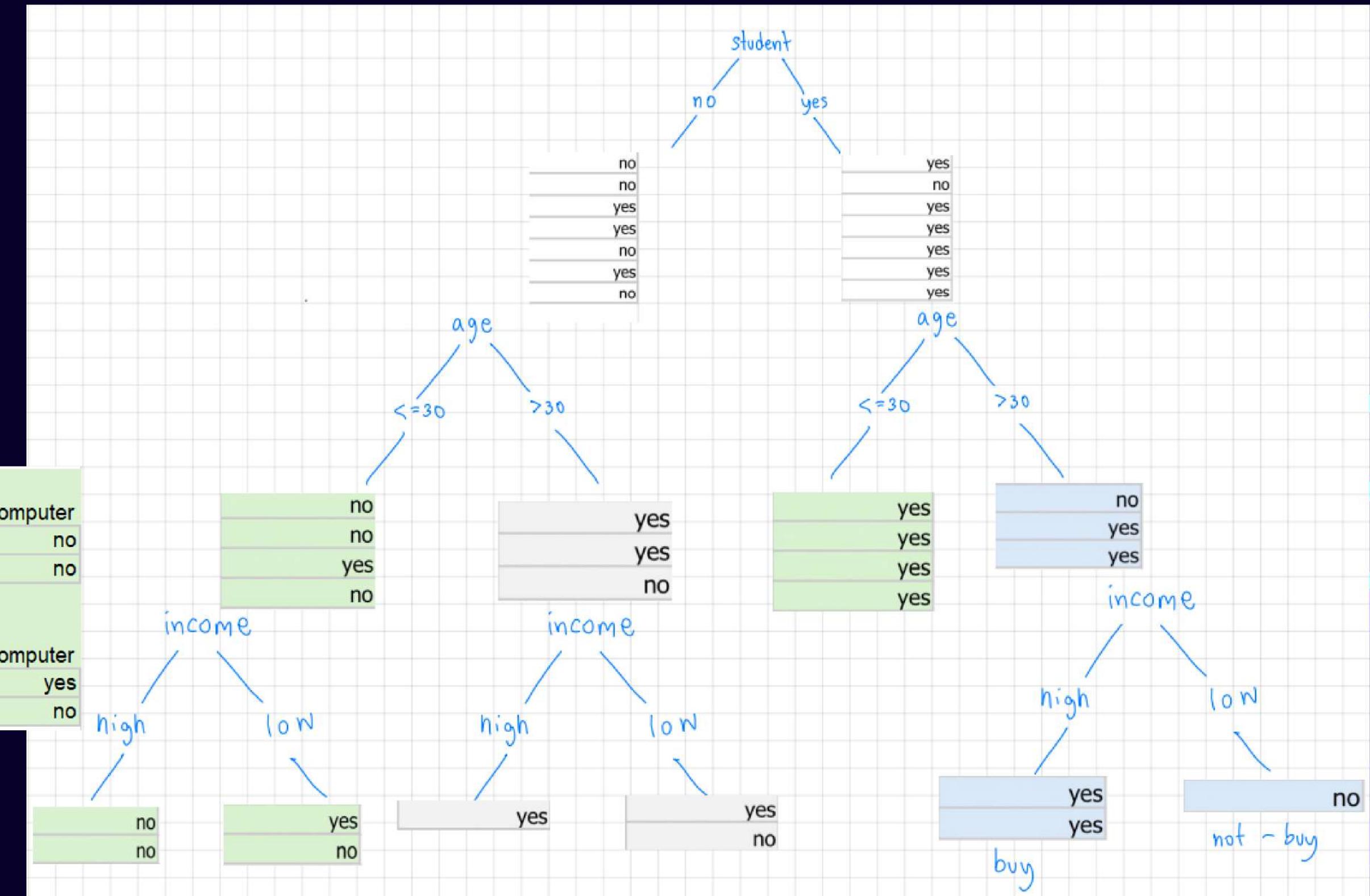
จะเห็นได้ว่า decision node หรือ income high สามารถกำหนดได้แล้ว โดย leaf node income high ตอบ โน คือ not-buy หมายความว่า คนที่ไม่ใช่นักเรียน/นักศึกษา อายุ น้อยกว่าหรือเท่ากับ 30 ที่มีรายได้ สูง จะตัดสินใจ ไมซื้อ คอมพิวเตอร์

และ เมื่องจากค่า expected info ของ income, low มีค่าเท่ากับ 0 และ 1 ตามลำดับ จึงตัดสินใจไม่เป็น เพราะค่า features ตัวอื่นในตาราง เมื่อนอกันหมด



MAX_LEAF_NODES = 7

student	no	age <= 30 income = high
age	income	student credit_rating buys_computer
<=30	high	no fair no
<=30	high	no excellent no
student	no	age <= 30 income = low
age	income	student credit_rating buys_computer
<=30	low	no fair yes
<=30	low	no fair no



MAX_LEAF_NODES = 7

และเมื่อกำหนด max leaf node = 7 จะแตกใบมาทางฝั่ง age > 30 เนื่องจากค่า expected info ของ income, high และ imcome, low ทางฝั่ง age <= 30 มีค่าเท่ากับ 0 และ 1 ตามลำดับ จึงตัดสินใจไม่แบ่ง เพราะค่า features ตัวอื่นในตาราง เมื่อนอนกับหมวด จึงตัดสินใจแตกใบทาง age > 30

student no age > 30 income = high				
age	income	student	credit_rating	buys_computer
>30	high	no	fair	yes

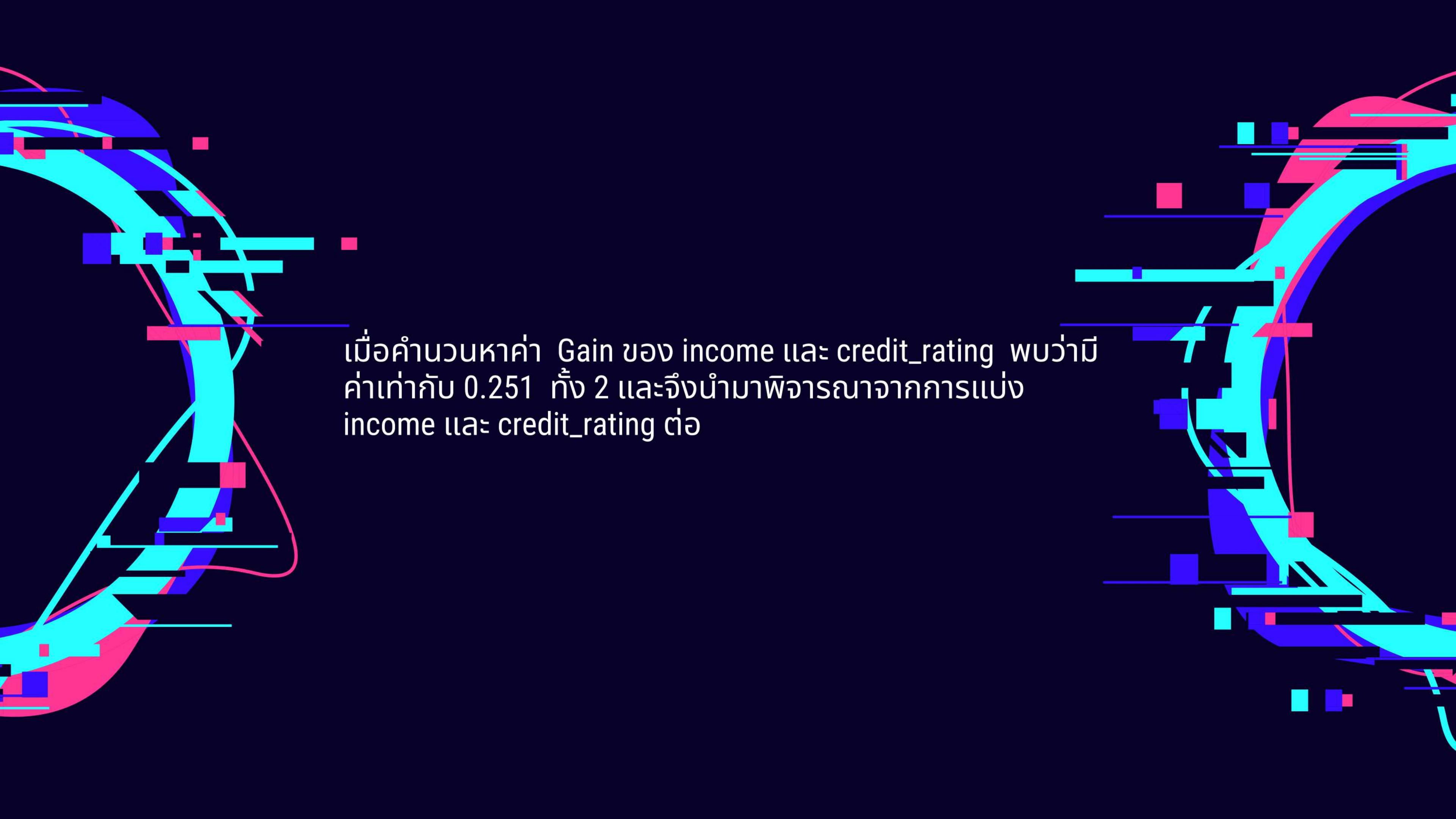
student no age > 30 income = low				
age	income	student	credit_rating	buys_computer
>30	low	no	excellent	yes
>30	low	no	excellent	no

student no age > 30 credit = fair				
age	income	student	credit_rating	buys_computer
>30	high	no	fair	yes

student no age > 30 credit = excellent				
age	income	student	credit_rating	buys_computer
>30	low	no	excellent	yes
>30	low	no	excellent	no

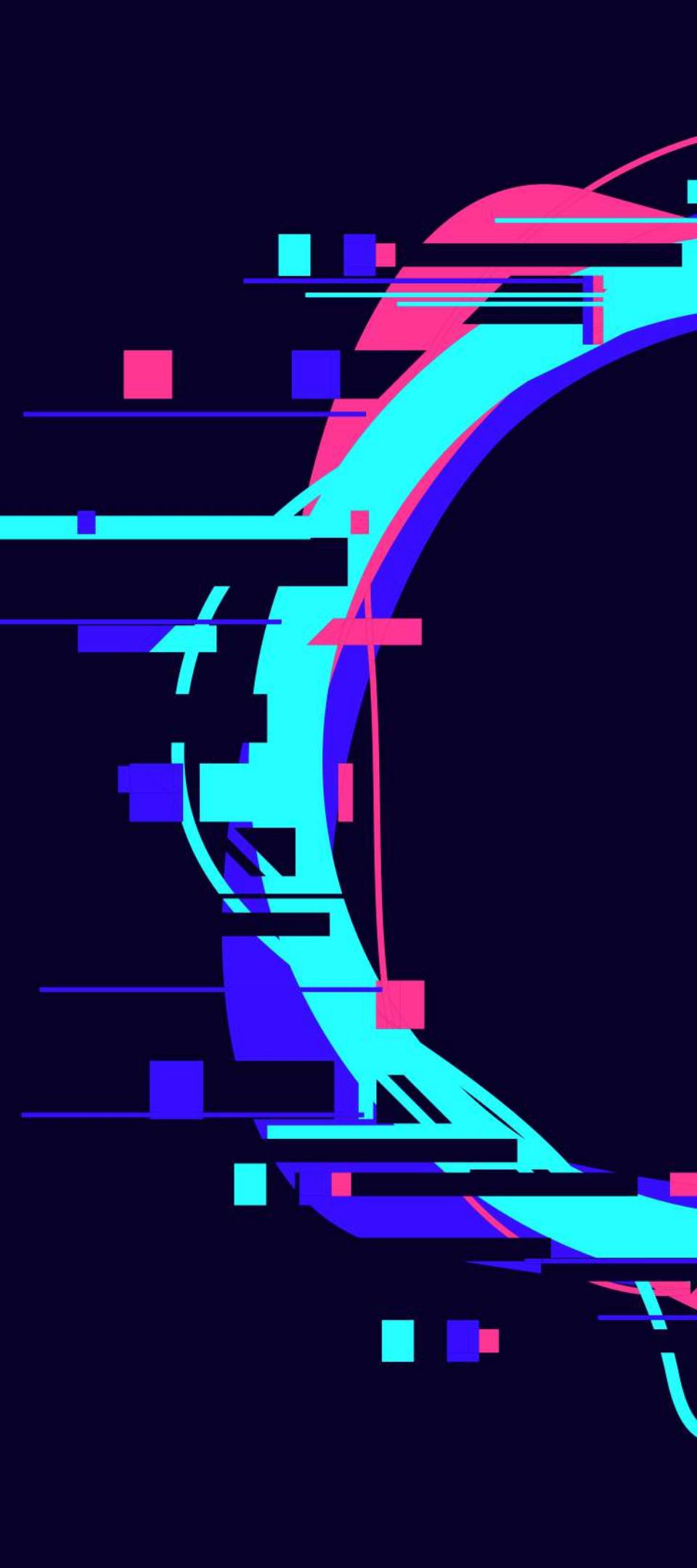
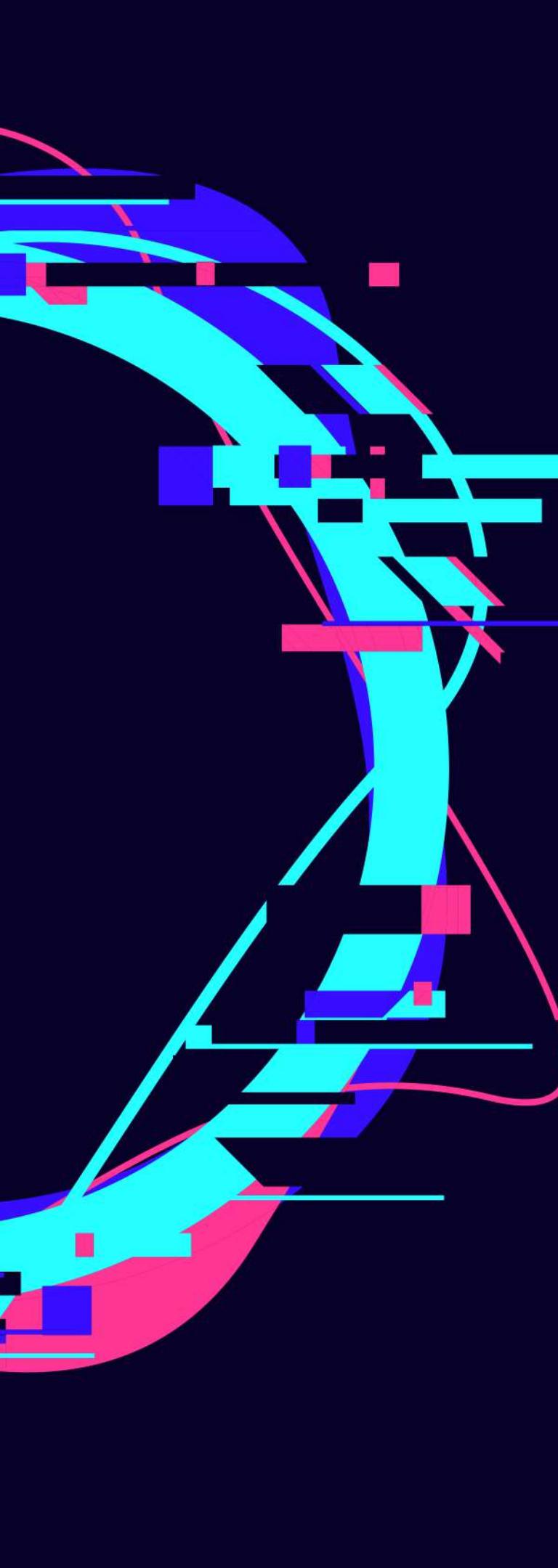
$$\text{Gain}(\text{student: no, age} > 30, \text{income}) = \text{Info}_{\text{student: no, age} > 30: \text{no}(D)} - \text{Info}_{\text{income}(D)} = 0.918 - 0.667 = 0.251$$

$$\text{Gain}(\text{student: no, age} > 30, \text{credit_rating}) = \text{Info}_{\text{student: no, age} > 30: \text{no}(D)} - \text{Info}_{\text{credit_rating}(D)} = 0.918 - 0.667 = 0.251$$



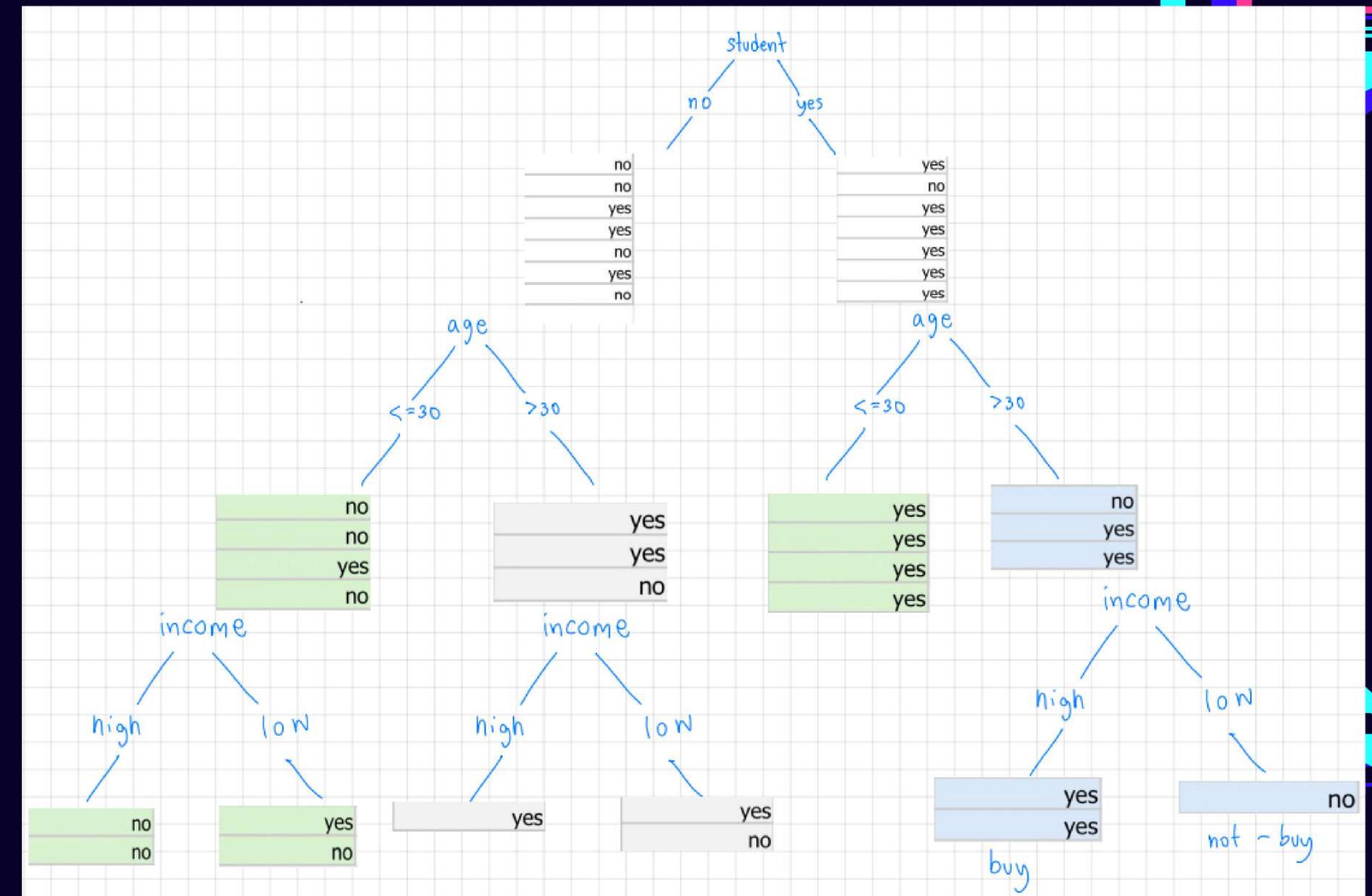
เมื่อคำนวณหาค่า Gain ของ income และ credit_rating พบร่วมค่าเท่ากับ 0.251 กึ่ง 2 และจึงนำมาพิจารณาจากการแบ่ง income และ credit_rating ต่อ

student	no	age > 30	income = high		
age	income	student	credit_rating		buys_computer
	>30	high	no	fair	yes
student	no	age > 30	income = low		
age	income	student	credit_rating		buys_computer
	>30	low	no	excellent	yes
	>30	low	no	excellent	no
student	no	age > 30	credit = fair		
age	income	student	credit_rating		buys_computer
	>30	high	no	fair	yes
student	no	age > 30	credit = excellent		
age	income	student	credit_rating		buys_computer
	>30	low	no	excellent	yes
	>30	low	no	excellent	no

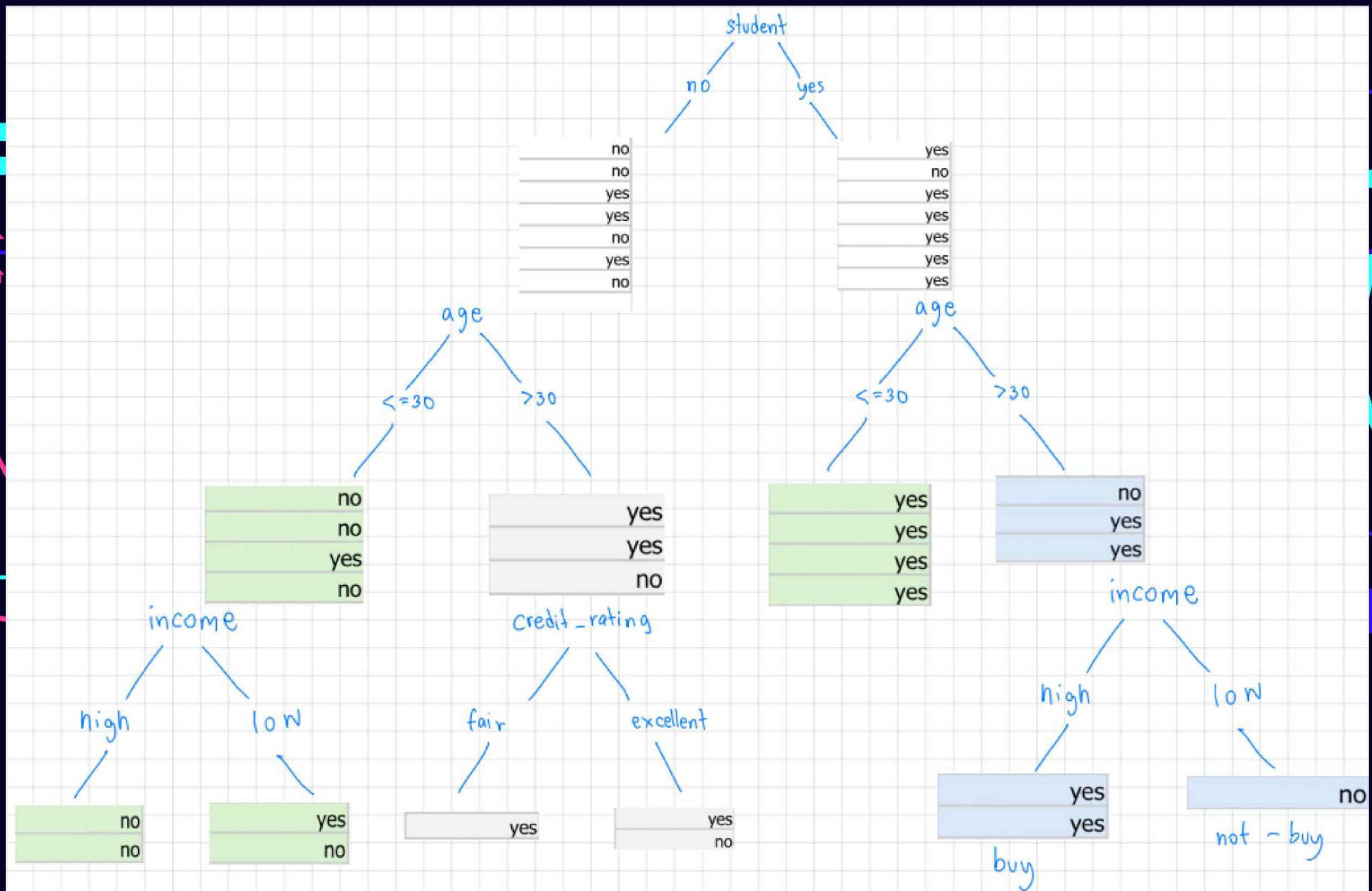


เมื่อแบ่งออกมา ค่า features ของทั้ง income และ credit_rating เหมือนกัน จึงได้ Decision tree ออกมา 2 แบบ

แบบที่ 1 INCOME เป็น DECISION NODE



ແບບທີ 2 CREDIT_RATING ເລື່ອ DECISION NODE



CONCLUSION

- Max_leaf_nodes เมื่อต้องการดูว่าต้นไม้จะโตไปทางไหน ให้ดูที่ค่า Gain ของแต่ละ features หาก Gain ของ features ใด มีค่ามาก การตัดสินใจที่จะโตของต้นไม้จะเลือกโตไปใน features นั้น ทั้งนี้ในการโตของต้นไม้จะต้องเก็บไปแต่ละ features เพื่อเลือกทิศทางในการโต และในการโตของต้นไม้นั้นจะโตได้ไม่เกินที่กำหนด max leaf nodes ไว้

ALGORITHM

```
function BFTree ( $A$ : a set of attributes,  
                 $E$ : the training instances,  
                 $N$ : the number of expansions,  
                 $M$ : the minimal number of instances at a terminal node  
    ) return a decision tree  
    begin  
        If  $E$  is empty, return failure;  
        Calculate the reduction of impurity for each attribute  
                in  $A$  on  $E$  at the root node  $RN$ ;  
        Find the best attribute  $A_b$  in  $A$ ;  
        Initialise an empty list  $NL$  to store nodes;  
        Add  $RN$  (with  $E$  and  $A_b$ ) into  $NL$ ;  
        expandTree( $NL$ ,  $N$ ,  $M$ );  
        return a tree with the root  $RN$ ;  
    end
```

```
expandTree( $NL$ ,  $N$ ,  $M$ )
begin
    If  $NL$  is empty, return;
    Get the first node  $FN$  from  $NL$ ;
    Retrieve training instances  $E$  and the best splitting attribute  $A_b$  of  $FN$ ;
    If  $E$  is empty, return failure;
    If the reduction of impurity of  $FN$  is 0 or  $N$  is reached,
        Make all nodes in  $NL$  into terminal nodes;
        return;
    If the split of  $FN$  on  $A_b$  would result in a successor node
        with less than  $M$  instances,
        Make  $FN$  into the terminal node;
        Remove  $FN$  from  $NL$ ;
        expandTree( $NL$ ,  $N$ ,  $M$ );
    Let  $SN_1$  and  $SN_2$  be the successor nodes generated by
        splitting  $FN$  on  $A_b$  on  $E$ ;
    Increment the number of expansions by one;
    Let  $E_1$  and  $E_2$  be the subsets of instances corresponding to
         $SN_1$  and  $SN_2$ ;
    Find the corresponding best attributes  $A_{b_1}$  for  $SN_1$ ;
    Find the corresponding best attributes  $A_{b_2}$  for  $SN_2$ ;
    Put  $SN_1$  (with  $E_1$  and  $A_{b_1}$ ) and  $SN_2$  (with  $E_2$  and  $A_{b_2}$ )
        into  $NL$  according to the reduction of impurity;
    Remove  $FN$  from  $NL$ ;
    expandTree( $NL$ ,  $N$ ,  $M$ );
end
```

THANK YOU