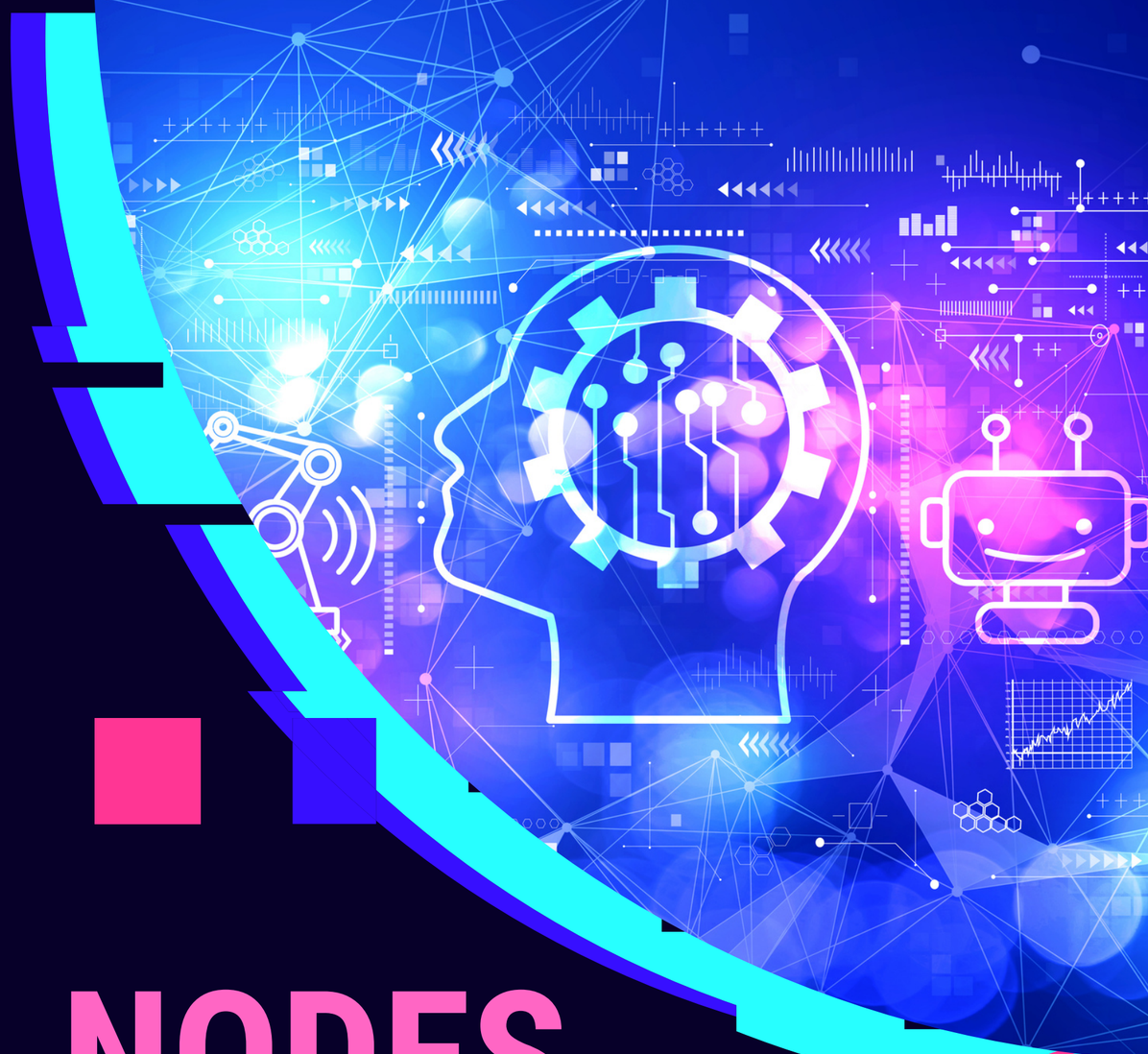


CRITERION AND MAX_LEAF_NODES

กลุ่ม กลุ้มXหอยหลอดด

กลุ่ม กลุ้มXหอยหลอดด

643020501-6	นายตะวัน เบ้าหล่อเพชร
643021260-7	นางสาวกิติลักษณ์ ลาดโฮม
643021261-5	นางสาวจารุพร การร้อย
643021263-1	นางสาวชนม์ชนก อังคุระษี
643021265-7	นายธนาธิป อินทรคีรี
643021266-5	นางสาวธิติพร ใจเอื้อ
643021268-1	นายพุทธิพงศ์ ยางนอก
643021273-8	นายศตวรรษ มูลสันเทียะ

CRITERION

ค่าวัดในการ Split โดยตัดสินว่าข้อมูลตัวอย่างนั้นจะถูกจัดประเภทไปยังกลุ่มใดโดยค่านี้มีให้เลือกหลากหลายเช่น gini entropy logloss

MAX_LEAFT_NODES

การกำหนดจำนวนสูงสุดของโหนดใบ ในต้นไม้ตัดสินใจ โดยทำให้การ Splitting หยุดลงเมื่อจำนวนโหนดใบถึงขีดจำกัดที่กำหนดไว้

CRITERION

GINI

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

Gini Impurity เป็นตัววัดความ "ไม่บริสุทธิ์" ของข้อมูลในโมเดล Decision Tree หมายถึงจำนวนข้อมูลที่มี class label ผสมกันอยู่มากน้อยเพียงใด คะแนน Gini ที่ต่ำแสดงถึงกลุ่มข้อมูลที่ "บริสุทธิ์" มากกว่า และเป็นการแบ่งกลุ่มที่ดีกว่า

GINI

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

เป็นค่า gini ของ feature ยิ่งน้อยยิ่งดี

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

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 หรือมีความสุ่มต่ำ

Entropy

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

เป็นค่า Entropy ของ feature ยิ่งน้อยยิ่งดี

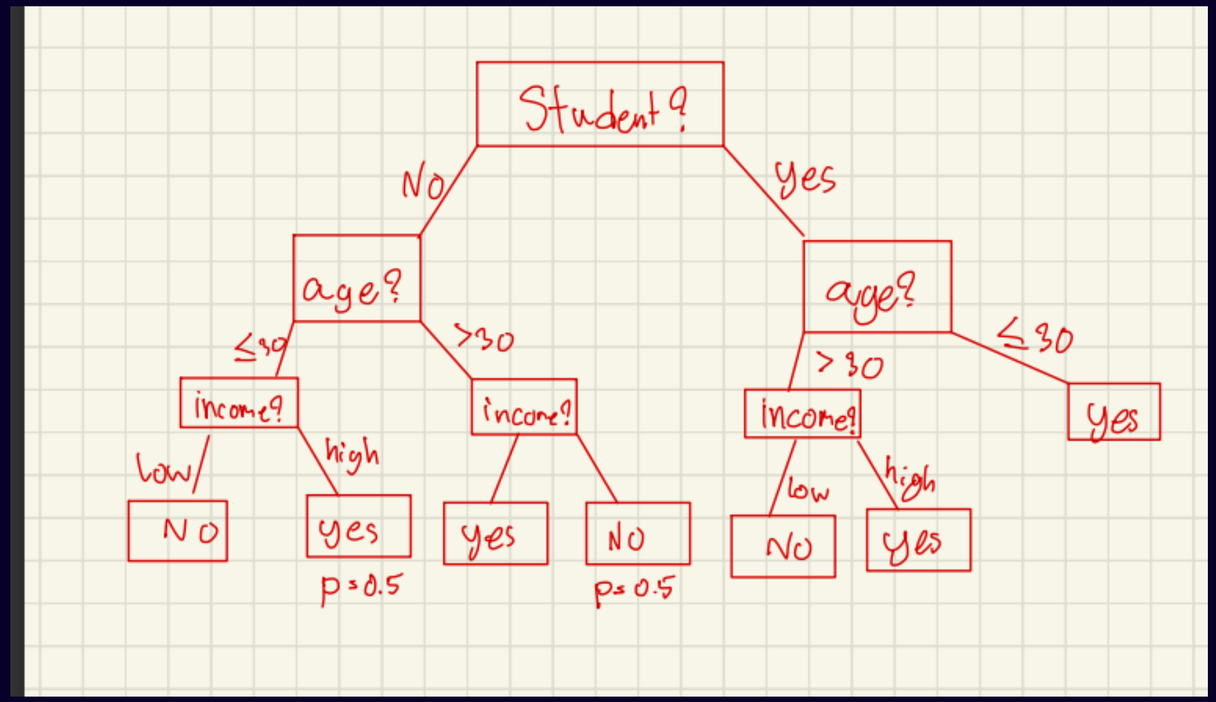
$$Gain(A) = Info(D) - Info_A(D)$$

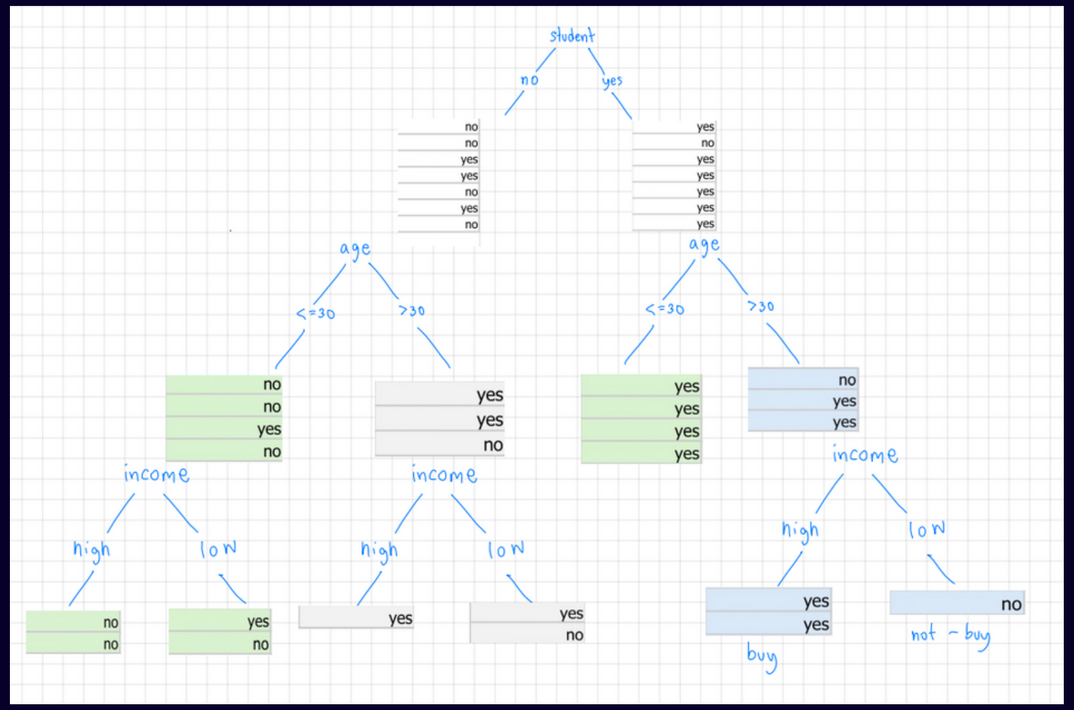
Gain คำนวณจาก ความแตกต่าง ของ ค่าความไม่แน่นอน ของ node หลัก (parent node) กับ ค่าความไม่แน่นอน ของ node ย่อย (child nodes)

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





$gini(D) = 0.459$

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

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

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

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

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

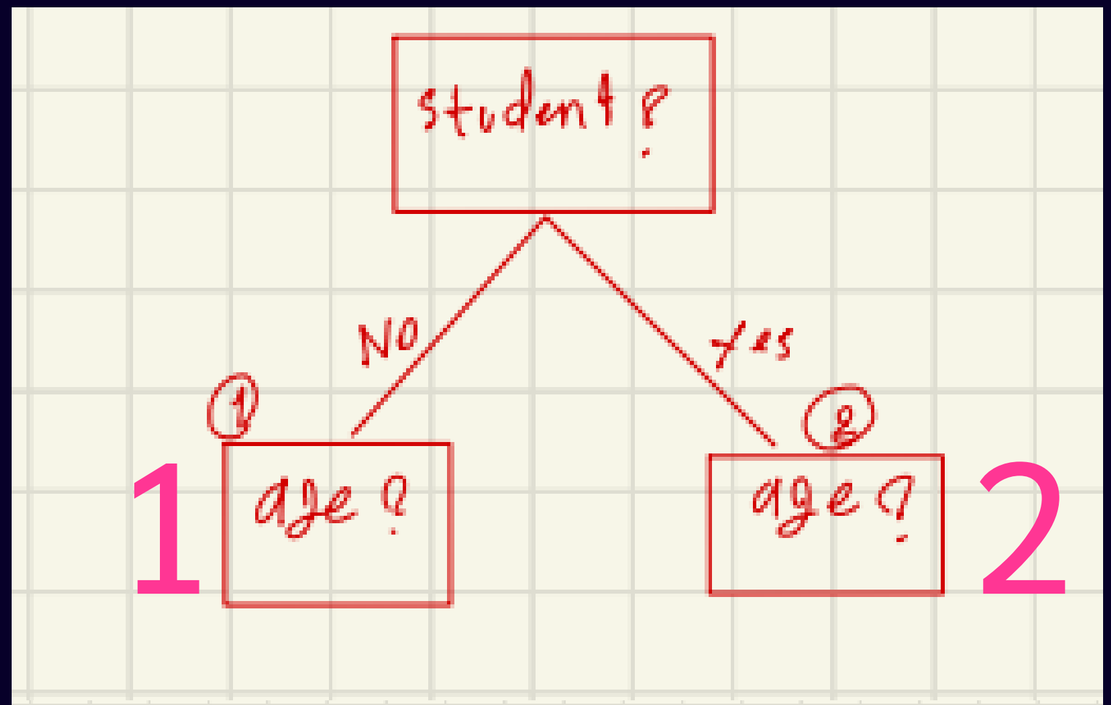
$Gain(income) = Info(D) - Info_{income}(D) = 0.940 - 0.938 = 0.002$

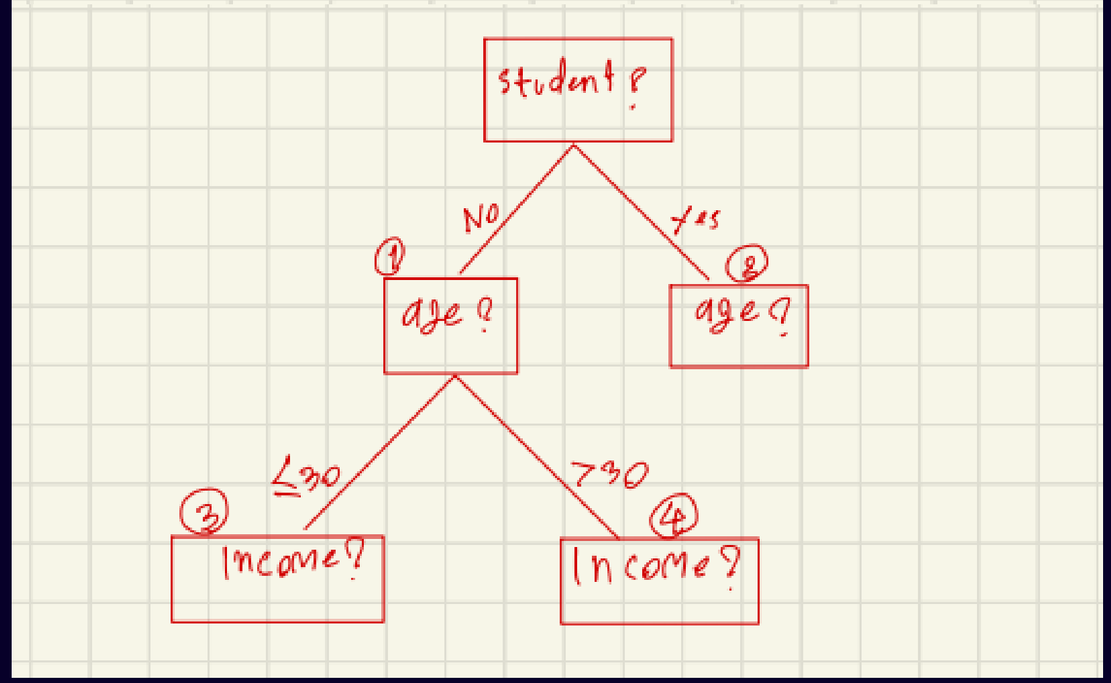
$Gain(student) = Info(D) - Info_{student}(D) = 0.940 - 0.788 = 0.152$

$Gain(credit\_rating) = Info(D) - Info_{credit\_rating}(D)$

$= 0.940 - 0.892 = 0.048$

∴ Root Node คือ student





การโตของต้นไม้

Entropy vs gini

node 1 Δgini(Student=no, age) = 0.0854

node 2 Δgini(Student=yes, age) = 0.0547

แบ่ง node 1 (ทางซ้าย) ก่อน

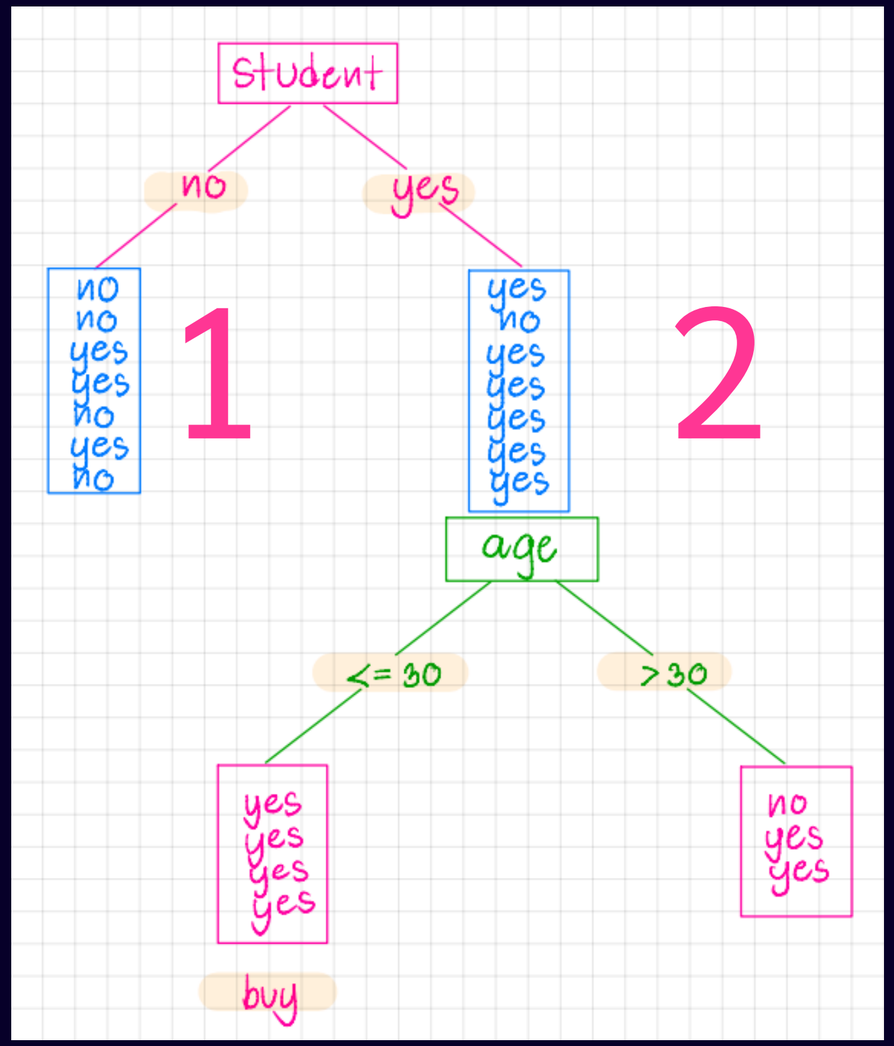
การโตของต้นไม้

Entropy vs gini

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

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

แบ่ง node 2 (ทางขวา) ก่อน

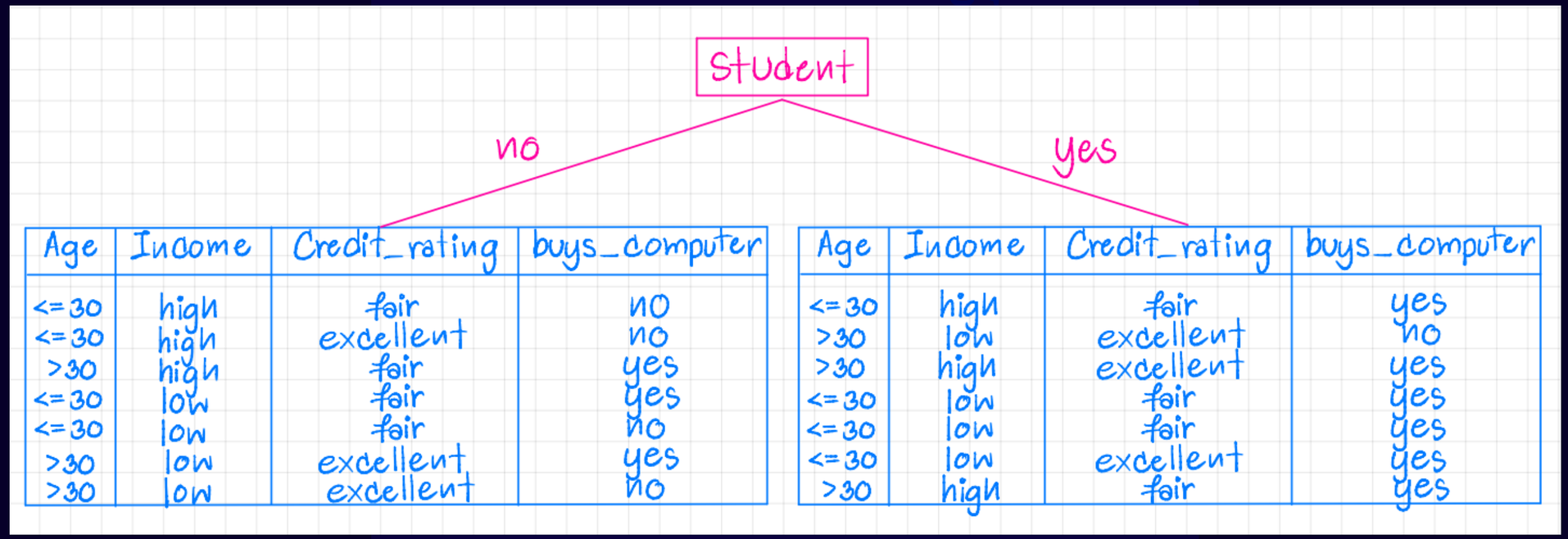


MAX_LEAF_NODES

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

MAX_LEAF_NODES = 2



Age	Income	Credit_rating	buys_computer
<=30	high	fair	no
<=30	high	excellent	no
>30	high	fair	yes
<=30	low	fair	yes
<=30	low	fair	no
>30	low	excellent	yes
>30	low	excellent	no

Age	Income	Credit_rating	buys_computer
<=30	high	fair	yes
>30	low	excellent	no
>30	high	excellent	yes
<=30	low	fair	yes
<=30	low	fair	yes
<=30	low	excellent	yes
>30	high	fair	yes

INFORMATION GAINED

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

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

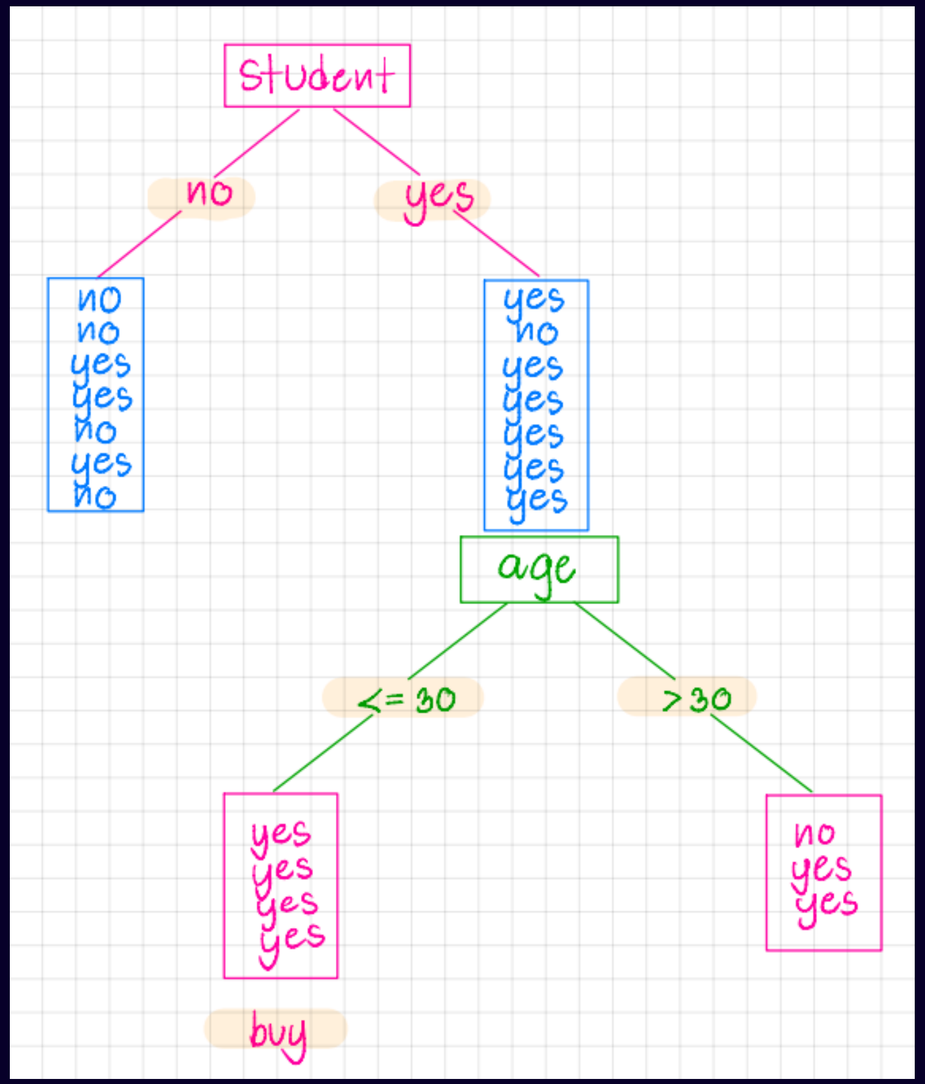
$$Gain(age) = Info(D) - Info_{age}(D) = 0.940 - 0.938 = 0.002$$

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

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

$$Gain(credit\_rating) = Info(D) - Info_{credit\_rating}(D) = 0.940 - 0.892 = 0.048$$

MAX_LEAF_NODES = 3

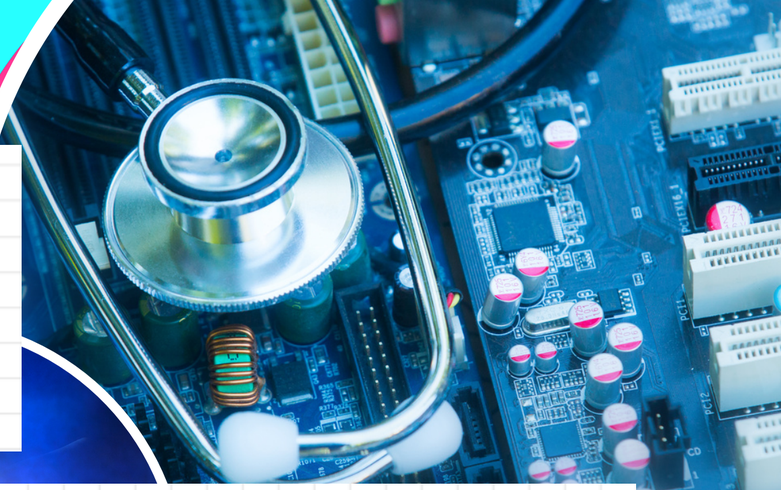


$Gain(age) = Info_{student:yes}(D) - Info_{age}(D) = 0.592 - 0.393 = 0.199$ ※

$Gain(income) = Info_{student:yes}(D) - Info_{income}(D) = 0.592 - 0.463 = 0.129$

$Gain(credit\_rating) = Info_{student:yes}(D) - Info_{credit\_rating}(D)$

$= 0.592 - 0.393 = 0.199$ ※



$Gain(student:no, age) = Info_{student:no}(D) - Info_{age}(D)$

$= 0.985 - 0.128 = 0.128$ ※

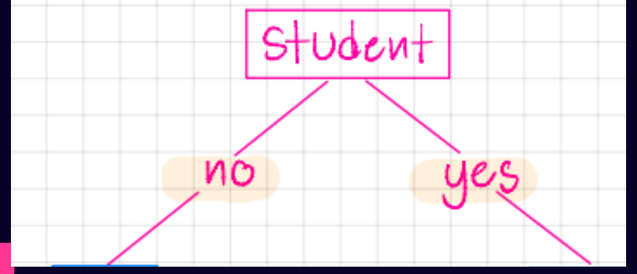
$Gain(student:no, income) = Info_{student:no}(D) - Info_{income}(D)$

$= 0.985 - 0.965 = 0.02$

$Gain(student:no, credit\_rating) = Info_{student:no}(D) - Info_{credit\_rating}(D)$

$= 0.985 - 0.965 = 0.02$

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



$$Gain(age) = Info_{student:yes}(D) - Info_{age}(D) = 0.592 - 0.393 = 0.199 \text{ ✳}$$

$$Gain(income) = Info_{student:yes}(D) - Info_{income}(D) = 0.592 - 0.463 = 0.129$$

$$Gain(credit\_rating) = Info_{student:yes}(D) - Info_{credit\_rating}(D)$$

$$= 0.592 - 0.393 = 0.199 \text{ ✳}$$

$$Gain(student: yes, age > 30, income) = Info_{student: yes, age > 30}(D) - Info_{student: yes, age > 30, income}(D)$$
$$= 0.918 - 0 = 0.918$$

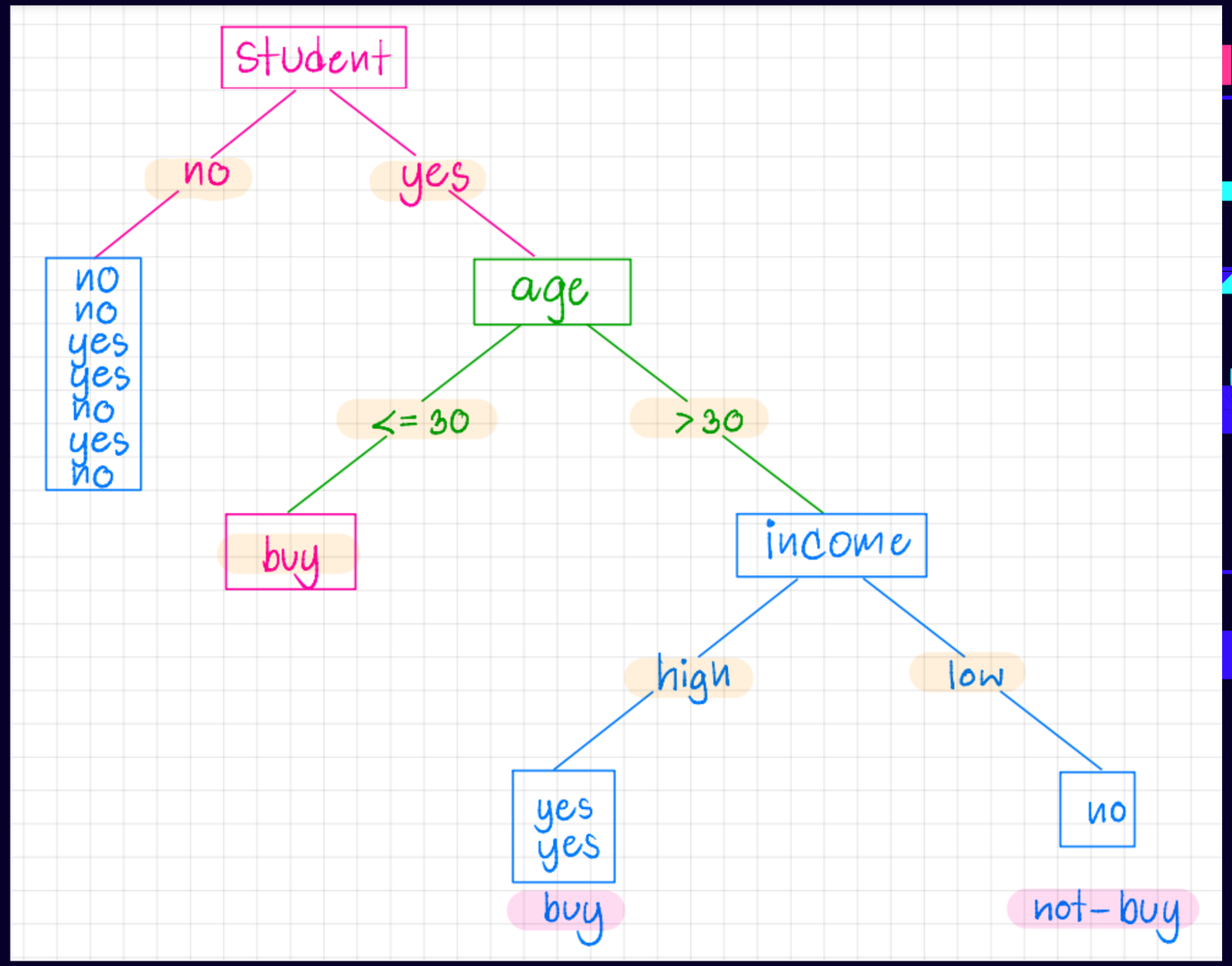
$$Gain(student: yes, age > 30, credit\_rating) = Info_{student: yes, age > 30}(D) - Info_{student: yes, age > 30, credit\_rating}(D)$$
$$= 0.918 - 0.667 = 0.251$$

$$Gain(s: yes, c = excellent, age) = Info_{s: yes, c = excellent}(D) - Info_{s: yes, c = excellent, age}(D)$$
$$= 0.918 - 0.667 = 0.251$$

$$Gain(s: yes, c = excellent, income) = Info_{s: yes, c = excellent}(D) - Info_{s: yes, c = excellent, income}(D)$$
$$= 0.918 - 0.667 = 0.251$$

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

MAX_LEAF_NODES = 4



Step 1 : หา expected info ที่จะ
Classify ก่อน

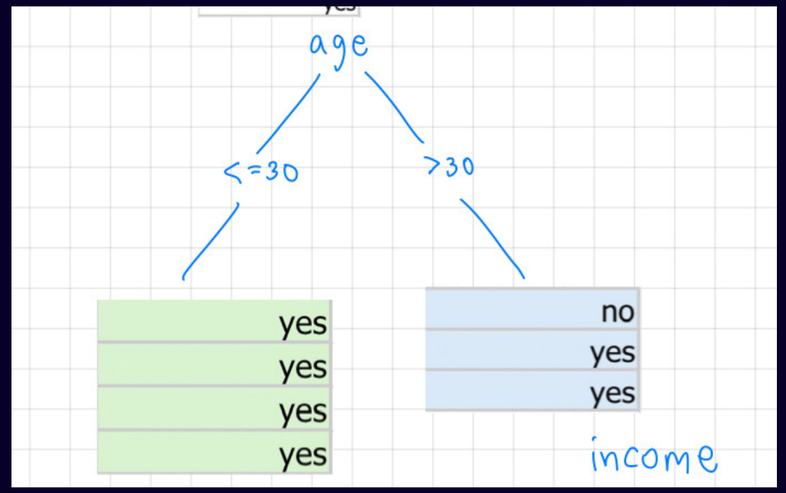
✧ Class P : buys_computer = "yes" → 4

✧ Class N : buys_computer = "no" → 0

$Info_{student:yes,age\leq 30}(D)$ ระบุไม่ได้

$= -\frac{4}{4}\log_2\left(\frac{4}{4}\right) - \frac{0}{4}\log_2\left(\frac{0}{4}\right)$

$= 0$



AGE <= 30

AGE > 30

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

$= Info_{student:yes,age>30}(D) - Info_{student:yes,age>30,income}(D)$

$= 0.918 - 0 = 0.918$

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

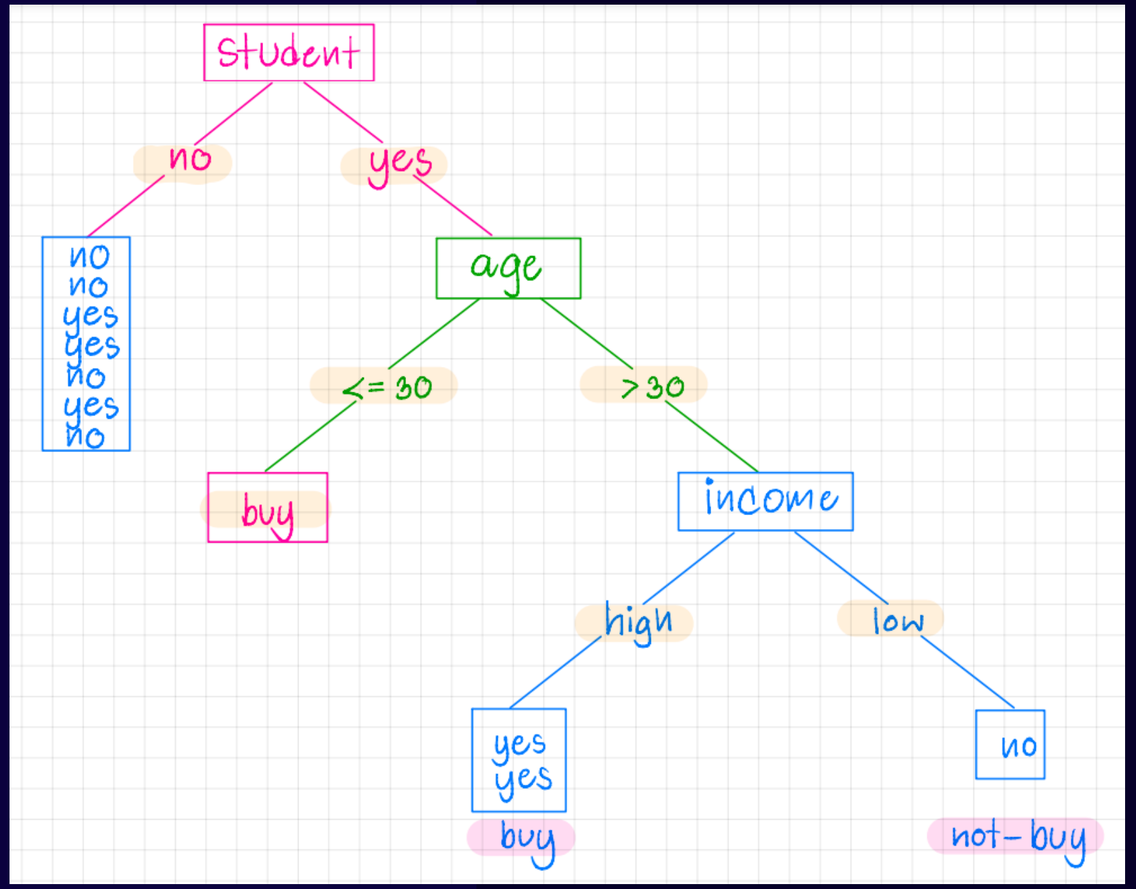
$= Info_{student:yes,age>30}(D) - Info_{student:yes,age>30,credit\_rating}(D)$

$= 0.918 - 0.667 = 0.251$

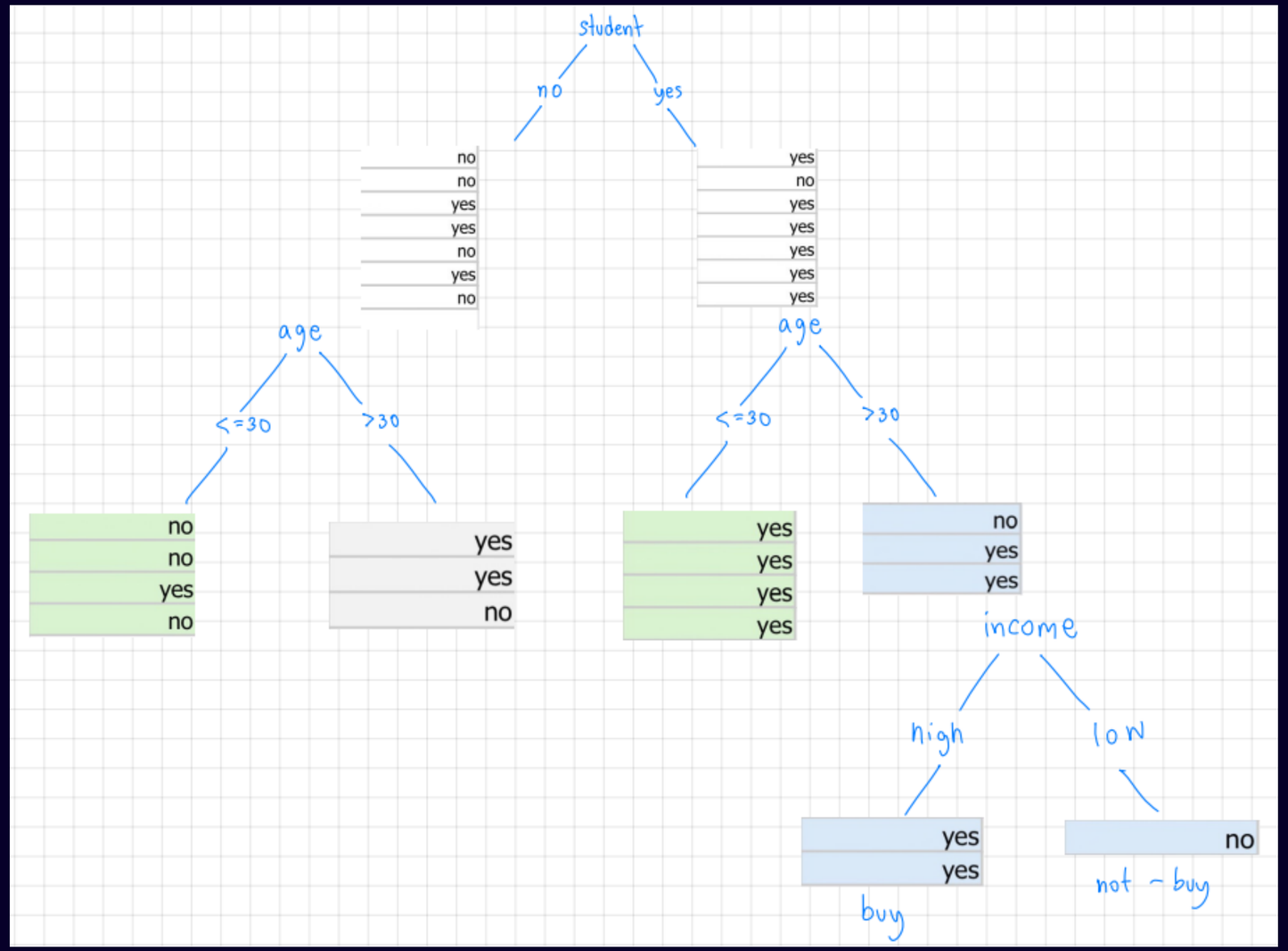
Gain(student : yes, age > 30, income)	0.918
Gain(student : yes, age > 30, credit_rating)	0.251

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



MAX_LEAF_NODES = 5

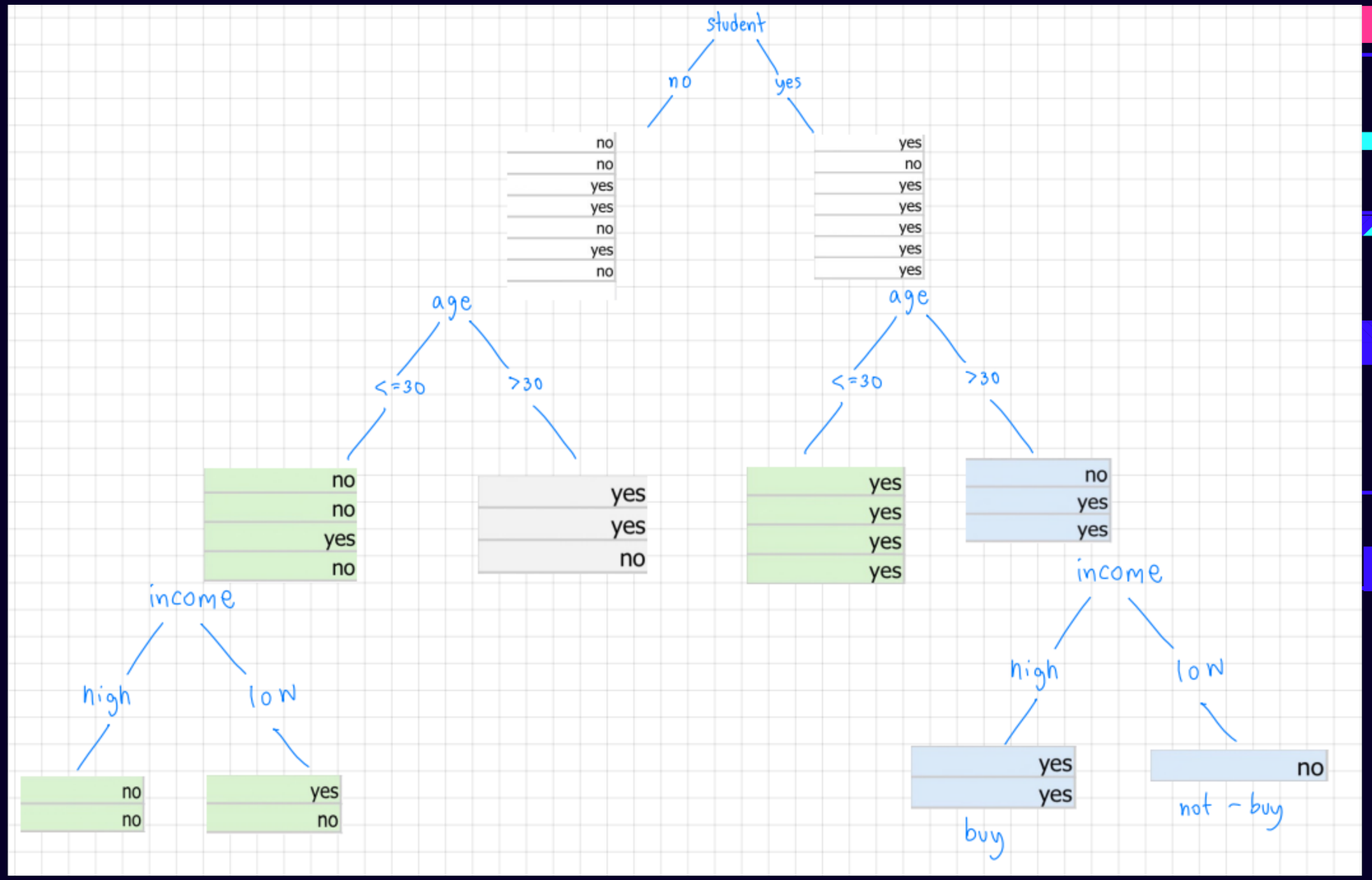


$$Gain(age) = Info_{student:no}(D) - Info_{age}(D) = 0.9852 - 0.8571 = 0.1281$$ ✓

$$Gain(income) = Info_{student:no}(D) - Info_{income}(D) = 0.9852 - 0.9649 = 0.0203$$

$$Gain(credit\_rating) = Info_{student:no}(D) - Info_{credit\_rating}(D) = 0.9852 - 0.9649 = 0.0203$$

MAX_LEAF_NODES = 6



Gain (student: no, age <= 30, income) = Info$_{\text{student: no, age<=30}}$(D) − Info$_{\text{income}}$(D) = 0.8113 − 0.5 = 0.3113 ✓

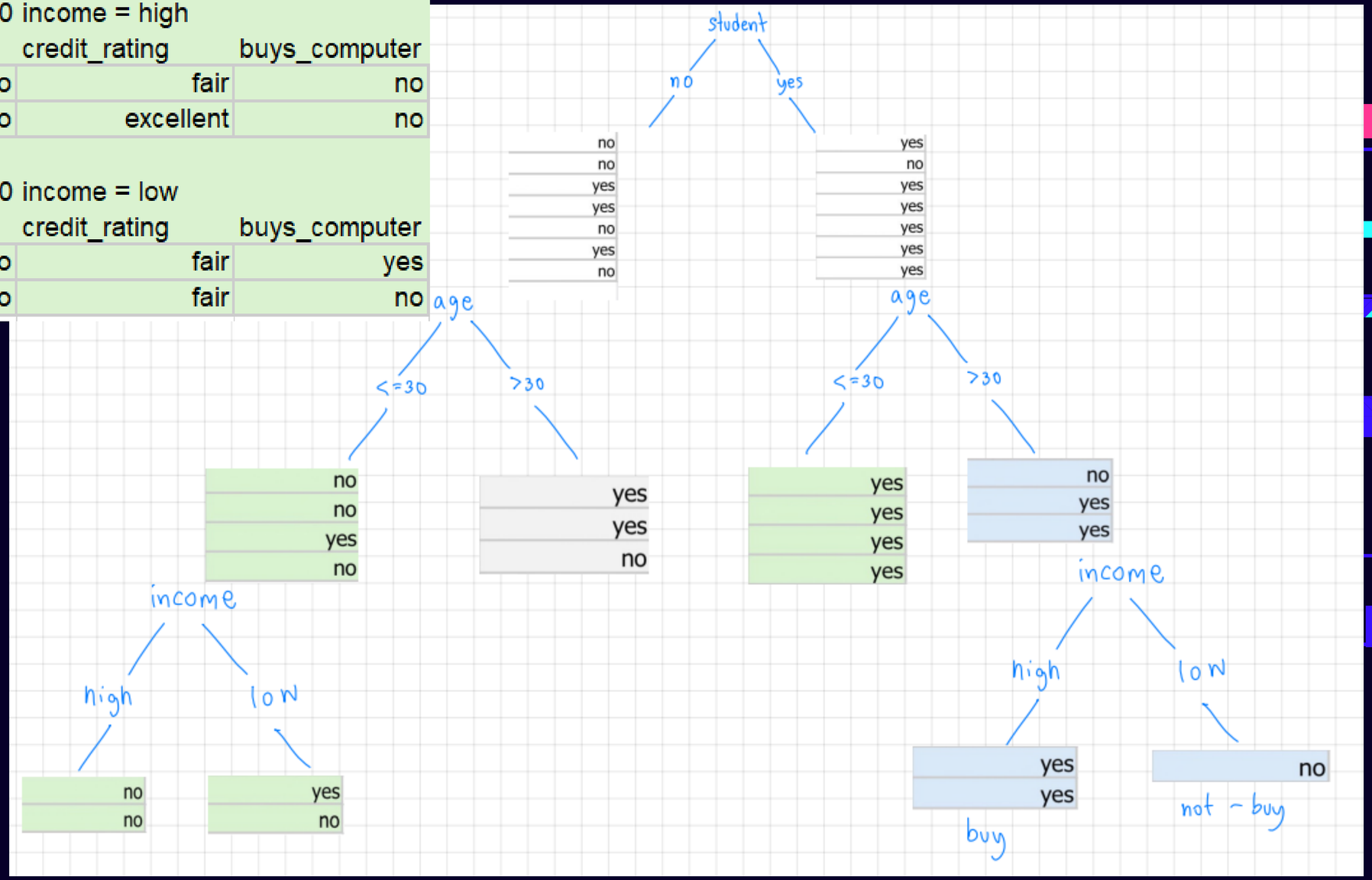
Gain (student: no, age <= 30, credit_rating) = Info$_{\text{student: no, age<=30}}$(D) − Info$_{\text{credit_rating}}$(D) = 0.8113 − 0.6887 = 0.1226

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

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

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

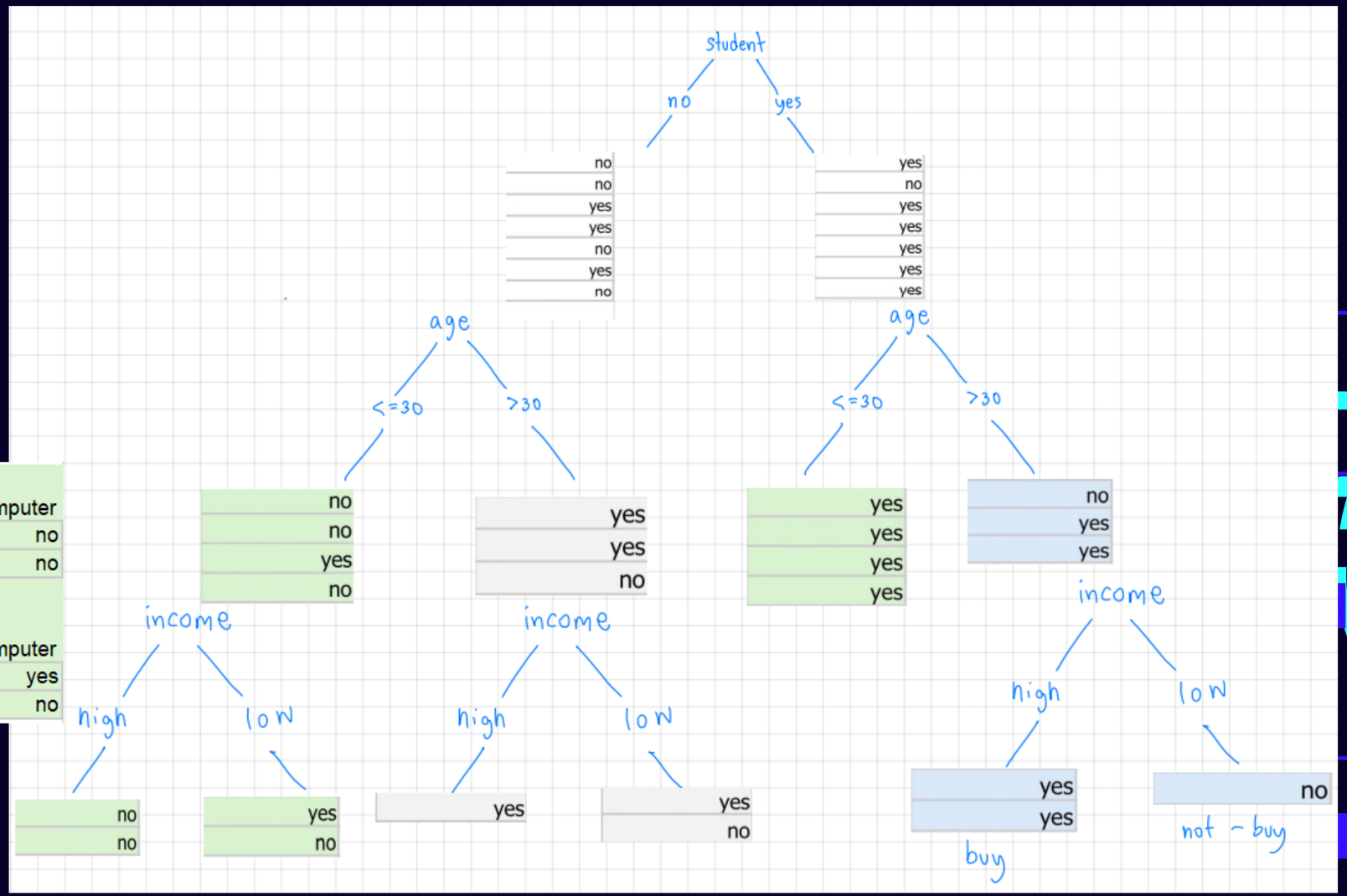
student	no	age <= 30	income = low	
age	income	student	credit_rating	buys_computer
<=30	low	no	fair	yes
<=30	low	no	fair	no



MAX_LEAF_NODES = 7

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



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

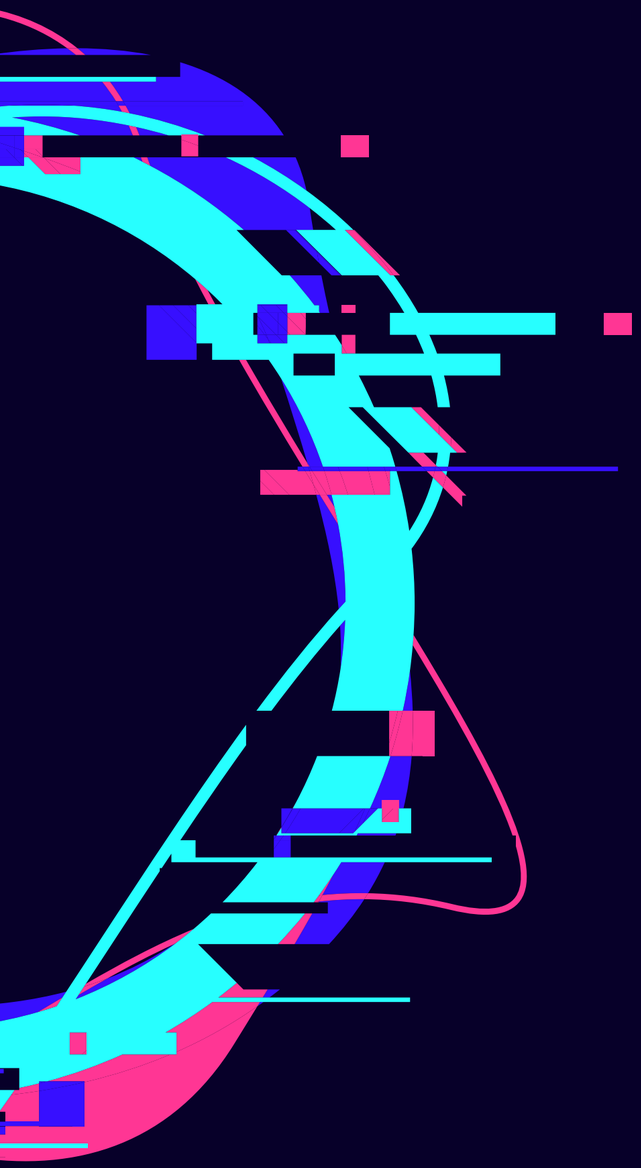
student age	no income	age > 30 student	income = low credit_rating	buys_computer
>30	low	no	excellent	yes
>30	low	no	excellent	no

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

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

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

$$Gain(student: no, age>30, credit\_rating) = Info_{student: no,\ age>30: no(D)} - Info_{credit\_rating}(D) = 0.918 - 0.667 = 0.251$$

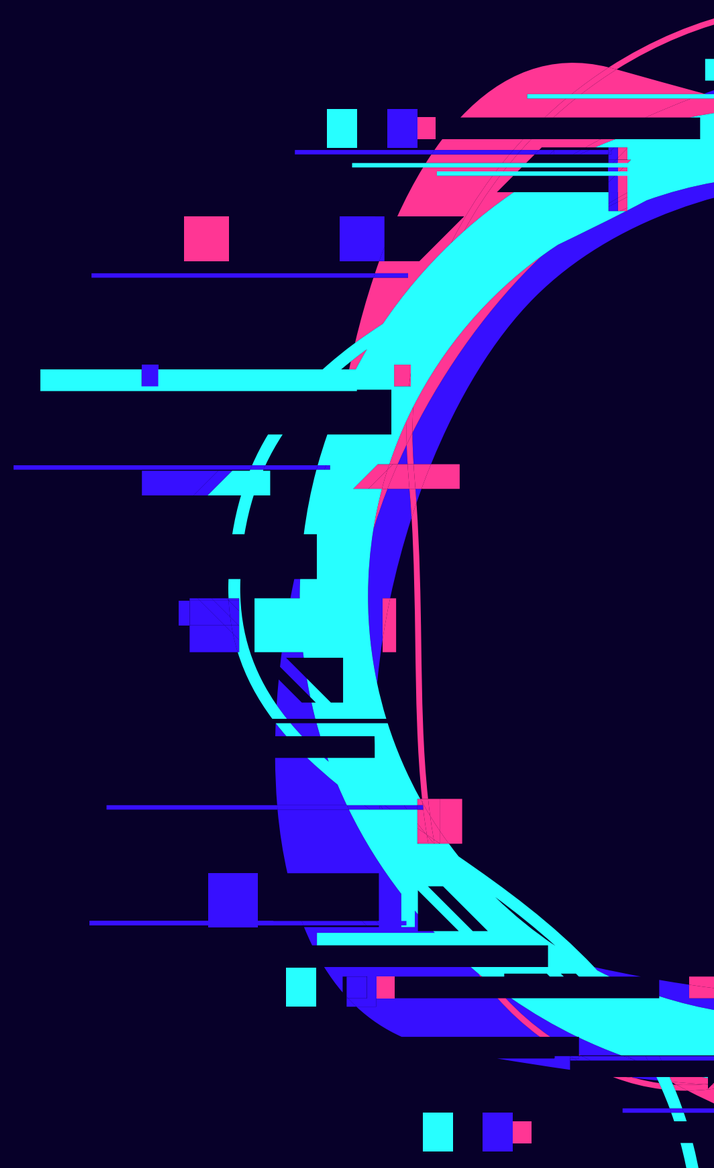


student	no	age > 30	income = high	
age	income	student	credit_rating	buys_computer
>30	high	no	fair	yes

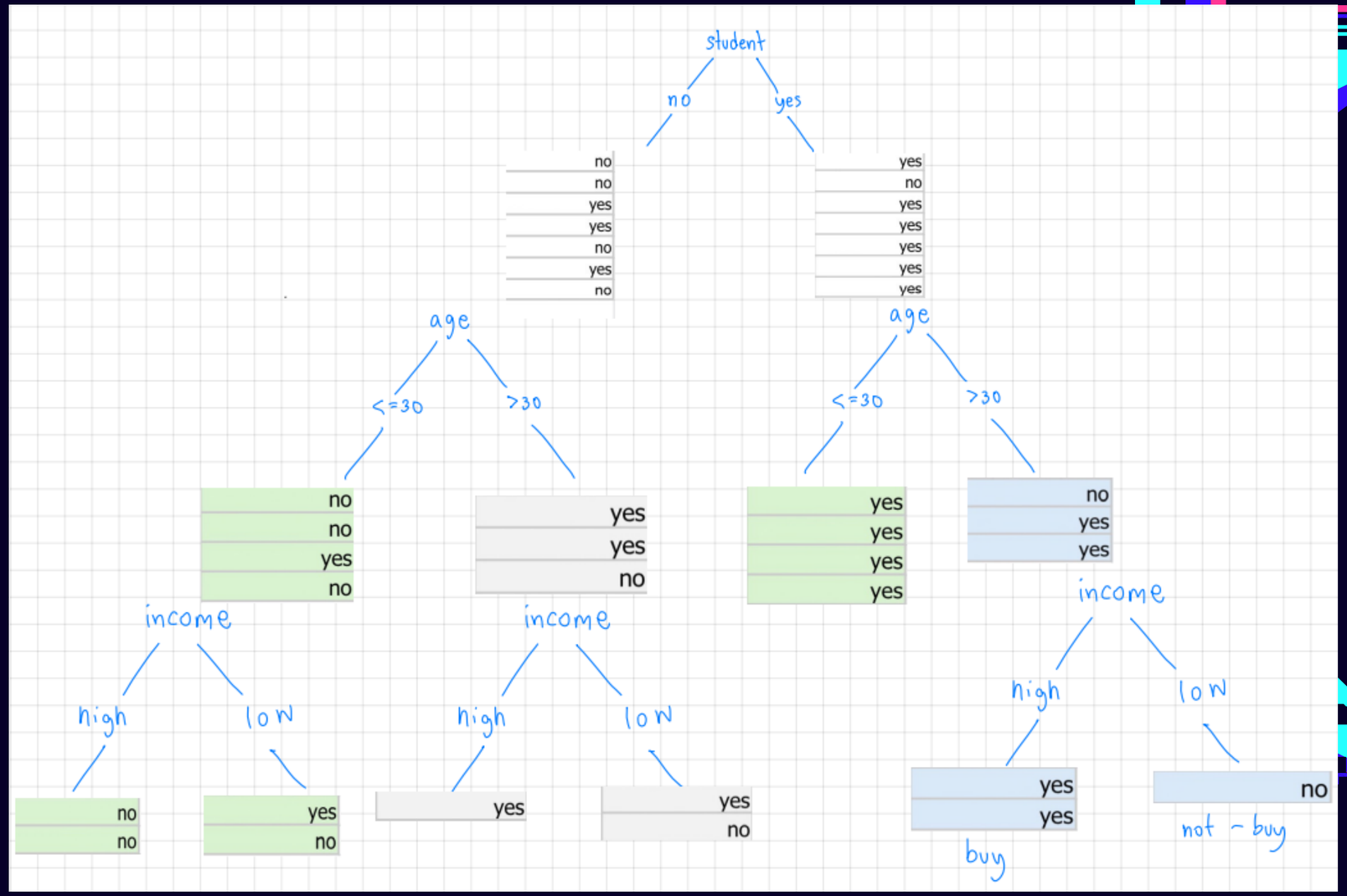
student	no	age > 30	income = low	
age	income	student	credit_rating	buys_computer
>30	low	no	excellent	yes
>30	low	no	excellent	no

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

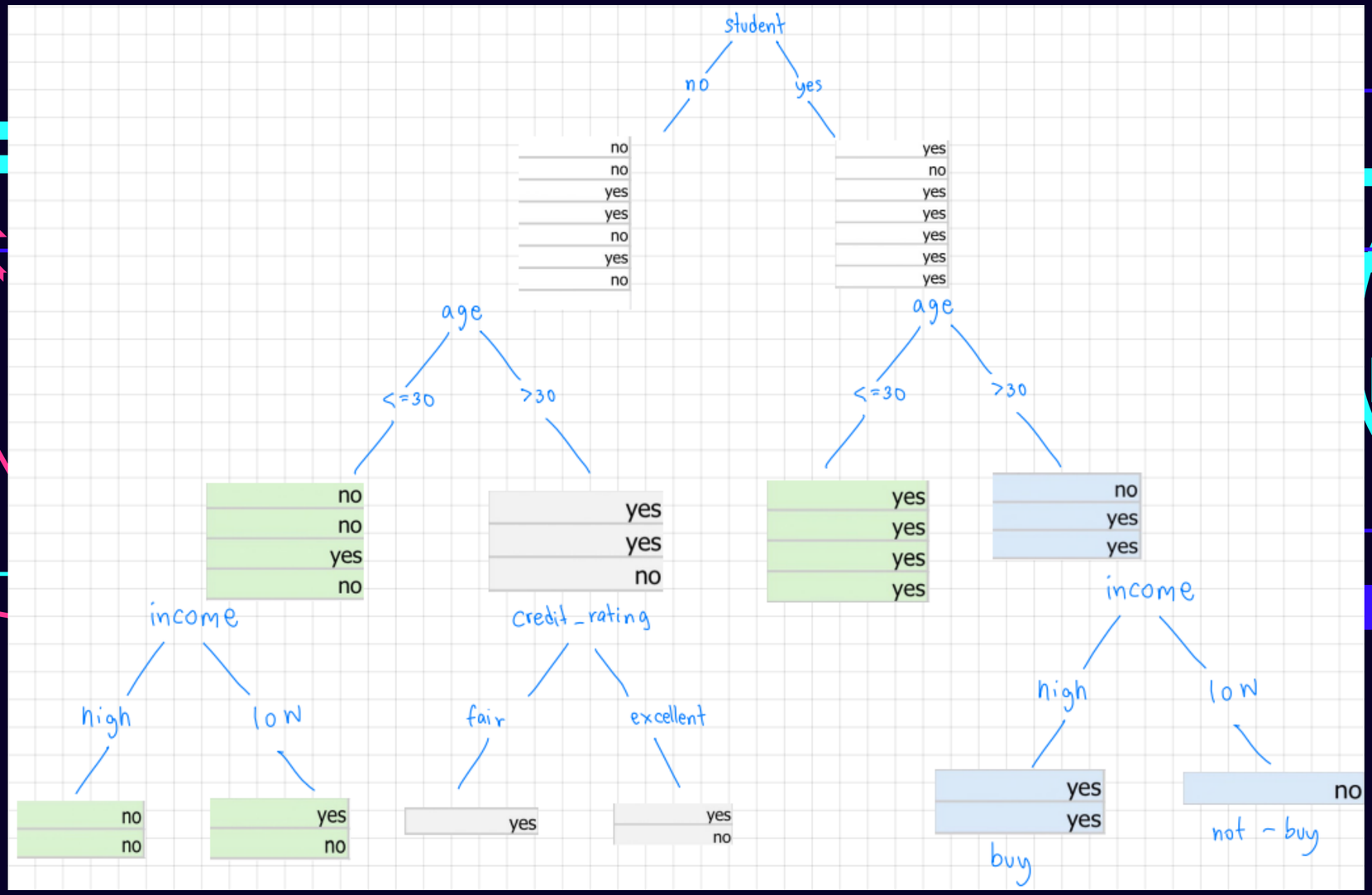
student	no	age > 30	credit = excellent	
age	income	student	credit_rating	buys_computer
>30	low	no	excellent	yes
>30	low	no	excellent	no



แบบที่ 1 INCOME เป็น DECISION NODE



แบบที่ 2 CREDIT_RATING เป็น DECISION NODE



CONCLUSION

Max_leaf_nodes เมื่อต้องการดูว่าต้นไม้จะโตไปทางไหน ให้ดูที่ค่า Gain ของแต่ละ features หาก Gain ของ features ใด มีค่ามาก การตัดสินใจที่จะโตของต้นไม้จะเลือกโตไปใน features นั้น ทั้งนี้ในการโตของต้นไม้จะต้องเทียบไปแต่ละ features เพื่อเลือกทิศทางในการโต และในการโตของต้นไม้นั้นจะโตได้ไม่เกินที่ กำหนด 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