

# CRITERION & MAX\_LEAF\_NODES

Classification

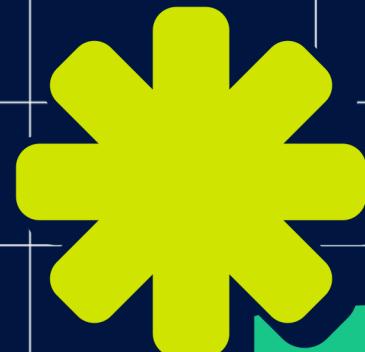
กลุ่มกลุ่ม

หอยหลอดดกรูป

# กลุ่ม กลุ่ม X หอยหลอดด

## Member

|             |                         |
|-------------|-------------------------|
| 643020501-6 | นายตะวัน เบ้าหล่อเพชร   |
| 643021260-7 | นางสาวกิติลักษณ์ ลาดโภน |
| 643021261-5 | นางสาวจารุพร การร้อย    |
| 643021263-1 | นางสาวชนมนาก อั้งคุระเม |
| 643021265-7 | นายธนาธิป อันตรคีรี     |
| 643021266-5 | นางสาวธิติพร ใจเอื้อ    |
| 643021268-1 | นายพุทธิพงศ์ ย่างนอก    |
| 643021273-8 | นายศตวรรษ มูลสันเทียะ   |



Tawan Industries

# CRITERION

**gini**

**entropy**

**log\_loss**

ฟังก์ชันที่ใช้วัดคุณภาพ  
ของการ split  
โดย default = ‘gini’





# CRITERION

## GINI



Define

```
[ ] Dtree_gini = DecisionTreeClassifier(random_state=0, criterion='gini' )
```

Train

```
▶ Dtree_gini.fit(X_train,y_train)
```

```
▶ DecisionTreeClassifier
```

```
DecisionTreeClassifier(random_state=0)
```

Test

```
[ ] y_predict_gini = Dtree_gini.predict(X_test)
```

```
[ ] data1_score = accuracy_score(y_test, y_predict_gini)  
data1_score
```

```
0.8387978142076503
```

$x[0] \leq 0.5$

gini = 0.497

samples = 324

value = [175, 149]

# CRITERION

## ENTROPY

### Define

```
[ ] Dtree_entropy = DecisionTreeClassifier(random_state=0, criterion='entropy')
```

### Train

```
▶ Dtree_entropy.fit(X_train,y_train)
```

```
DecisionTreeClassifier  
DecisionTreeClassifier(criterion='entropy', random_state=0)
```

```
[ ] _, ax = plt.subplots(figsize=(15,10))  
tree.plot_tree(Dtree_entropy, ax=ax);
```

### Test

```
[ ] y_predict_entropy = Dtree_entropy.predict(X_test)
```

```
[ ] data1_score = accuracy_score(y_test, y_predict_entropy)  
data1_score
```

```
0.8387978142076503
```

$x[0] \leq 0.5$

entropy = 0.995

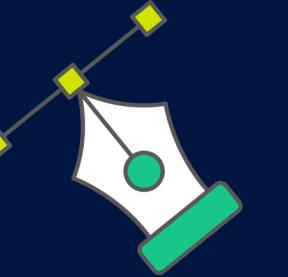
samples = 324

value = [175, 149]



# CRITERION

# LOG LOSS



## Define

```
[ ] Dtree_log_loss = DecisionTreeClassifier(random_state=0, criterion='log_loss')
```

### Train

```
▶ Dtree_log_loss.fit(X_train,y_train)
```

```
DecisionTreeClassifier  
DecisionTreeClassifier(criterion='log_loss', random_state=0)
```

```
[ ] _, ax = plt.subplots(figsize=(15,10))  
tree.plot_tree(Dtree_log_loss, ax=ax);
```

### Test

```
[ ] y_predict_gini = Dtree_log_loss.predict(X_test)
```

```
▶ accuracy_score(y_test, y_predict_gini)
```

```
0.8387978142076503
```

$x[0] \leq 0.5$

log loss = 0.995

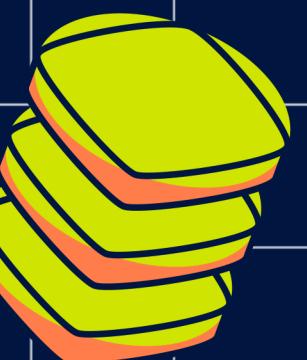
samples = 324

value = [175, 149]

**MAX**

**LEAF**

**NODES**



```
Dtree3_4 = DecisionTreeClassifier(max_leaf_nodes=2)
```

```
Dtree3_4.fit(X_train,y_train)
```

```
DecisionTreeClassifier
```

```
DecisionTreeClassifier(max_leaf_nodes=2)
```

```
tree.plot_tree(Dtree3_4);
```

x[0] <= 0.5  
gini = 0.497  
samples = 324  
value = [175, 149]

gini = 0.141  
samples = 144  
value = [133, 11]

gini = 0.358  
samples = 180  
value = [42, 138]

```
y_predict3_4 = Dtree3_4.predict(X_test)
```

```
accuracy_score(y_test, y_predict3_4)
```

0.8715846994535519

# อธิบาย



ในการ Define และเราจะกำหนดพารามิเตอร์ อยู่ที่ 2  
จากนั้นเมื่อกำหนดแล้วเราจะนำมาเทรน โดยการเรียกใช้ `.fit` ซึ่งจะเป็นการเทรนโมเดลเพื่อให้  
สามารถคำนวณค่าที่ถูกต้อง และปรับพารามิเตอร์ต่างๆให้จำแนกหรือได้ข้อมูลที่ดีที่สุด เมื่อเทรน  
ข้อมูลแล้วเราจะเรียกดู `tree.plot` ได้เลย

จากนั้นเราจะ Test ดูนะครับ โดยเราจะดูที่ค่า `accuracy_score` ซึ่งตัว `accuracy_score` จะ  
ใช้สำหรับการประเมินประสิทธิภาพของโมเดลที่ถูกเทรน ซึ่งเลข 2 ที่เรา define ไว้

เมื่อเทสแล้วดูที่ค่า `accuracy_score` จะได้เท่ากับ 0.8715846994535519  
หมายถึงโมเดล Decision Tree Classifier ที่ถูกสร้างขึ้นมีความแม่นยำในการคำนวณชุด  
ข้อมูลทดสอบประมาณ 87.15%

(คนแรกพูดเปิดหัวข้อด้วย )



```
Dtree3_3 = DecisionTreeClassifier(max_leaf_nodes=5)
```

```
Dtree3_3.fit(X_train,y_train)
```

```
DecisionTreeClassifier
```

```
DecisionTreeClassifier(max_leaf_nodes=5)
```

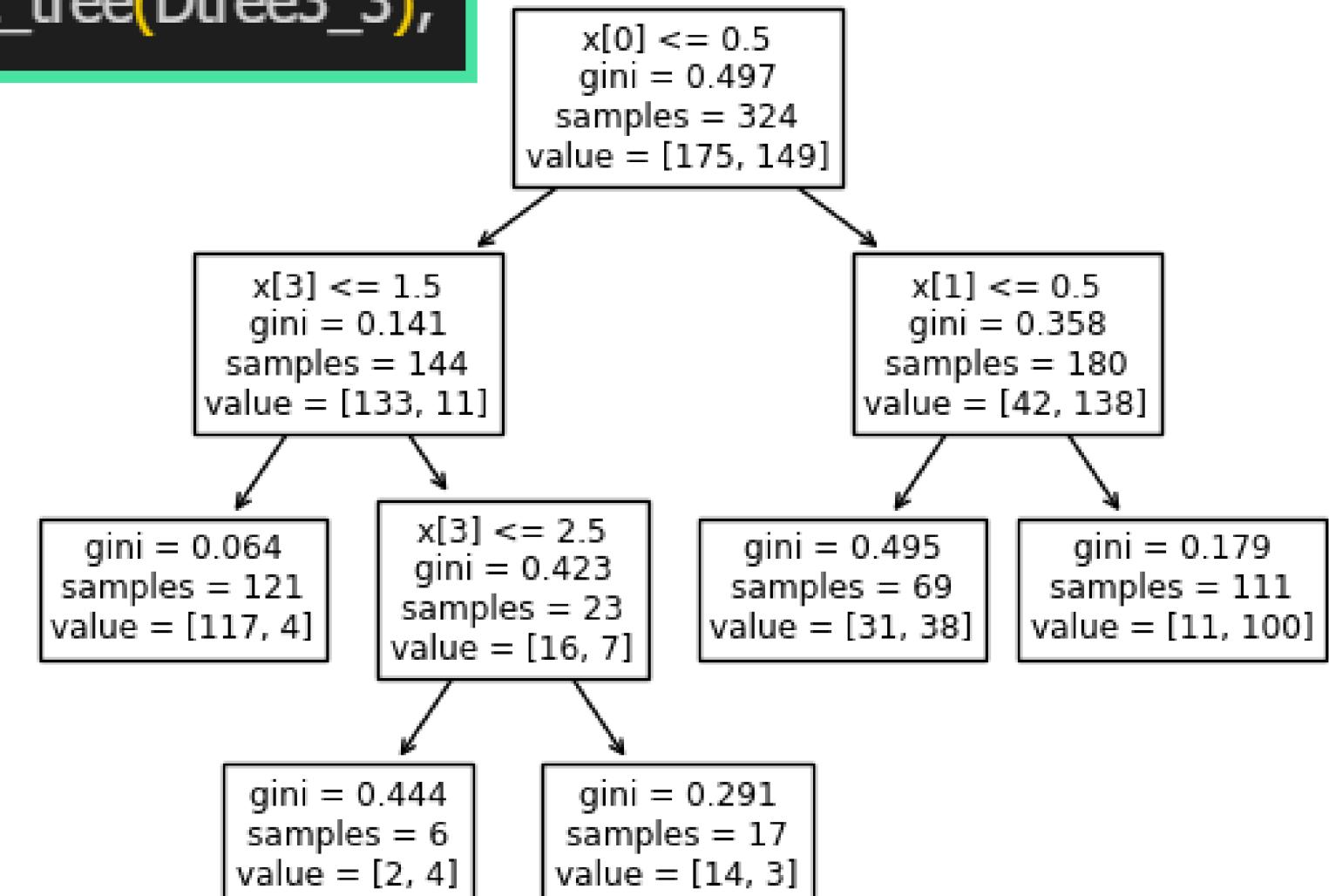
```
y_predict3_3 = Dtree3_3.predict(X_test)
```

```
accuracy_score(y_test, y_predict3_3)
```

```
0.8715846994535519
```

```
(max_leaf_nodes=5)
```

```
tree.plot_tree(Dtree3_3);
```



# อธิบาย



ถัดมา Define กี่ 2 เราจะกำหนดพารามิเตอร์ อยู่ที่ 5 จากนั้นจะการเรียกใช้ .fit เช่นเดิม เพื่อ เทคนโมเดล และปรับพารามิเตอร์ต่างๆให้จำแนกหรือได้ข้อมูลที่ดีที่สุด เมื่อเทวนข้อมูลแล้วเราจะเรียกดู tree.plot

ในตัว treeplot Simpleตัวแรกจะเท่ากับ 324 ชั่ง จะเป็นจำนวนข้อมูลที่เข้ากันได้กับ node นั้น ดังนั้นเมื่อการตัดสินใจลงเรื่อยๆตามความลึกของต้นไม้ จำนวน rsample ของ node ในแต่ละชั้นก็จะมีแนวโน้มลดลงเรื่อยๆเช่นเดียวกัน

ในส่วนของ gini = 0.497: หมายถึงค่า Gini impurity ที่ถูกคำนวณขึ้นสำหรับโหนดนี้และมีค่าเท่ากับ 0.497 ซึ่งแสดงถึงความไม่สมดุลในการแบ่งกลุ่ม

ถัดมา จะเป็น value = [175,149]: หมายถึงค่าของตัวแปรตามดัชนีที่มีความน่าจะเป็นสูงสุด ซึ่งถูกแบ่งเป็นกลุ่มตามคลาสหรือค่าที่กำหนดได้ ซึ่งในกรณีนี้มีค่าเท่ากับ [175,149] ซึ่งหมายถึง มีจำนวนตัวอย่างที่มีคลาสหรือค่าที่กำหนดได้เท่ากับ 175 ตัวอย่างและ 149 ตัวอย่าง นั่นหมายความว่ามีการแบ่งกลุ่มแบบนี้ในโหนดนี้โดยตรง

เมื่อทดสอบแล้วดูที่ค่า accuracy\_score จะได้เท่ากับ 0.8715846994535519  
หมายถึงโมเดล Decision Tree Classifier ที่ถูกสร้างขึ้นมีความแม่นยำในการคำนายนัด  
ข้อมูลทดสอบประมาณ 87.15%



```
Dtree3_5 = DecisionTreeClassifier(random_state=0,max_leaf_nodes=6)
```

```
Dtree3_5.fit(X_train,y_train)
```

```
DecisionTreeClassifier  
DecisionTreeClassifier(max_leaf_nodes=6, random_state=0)
```

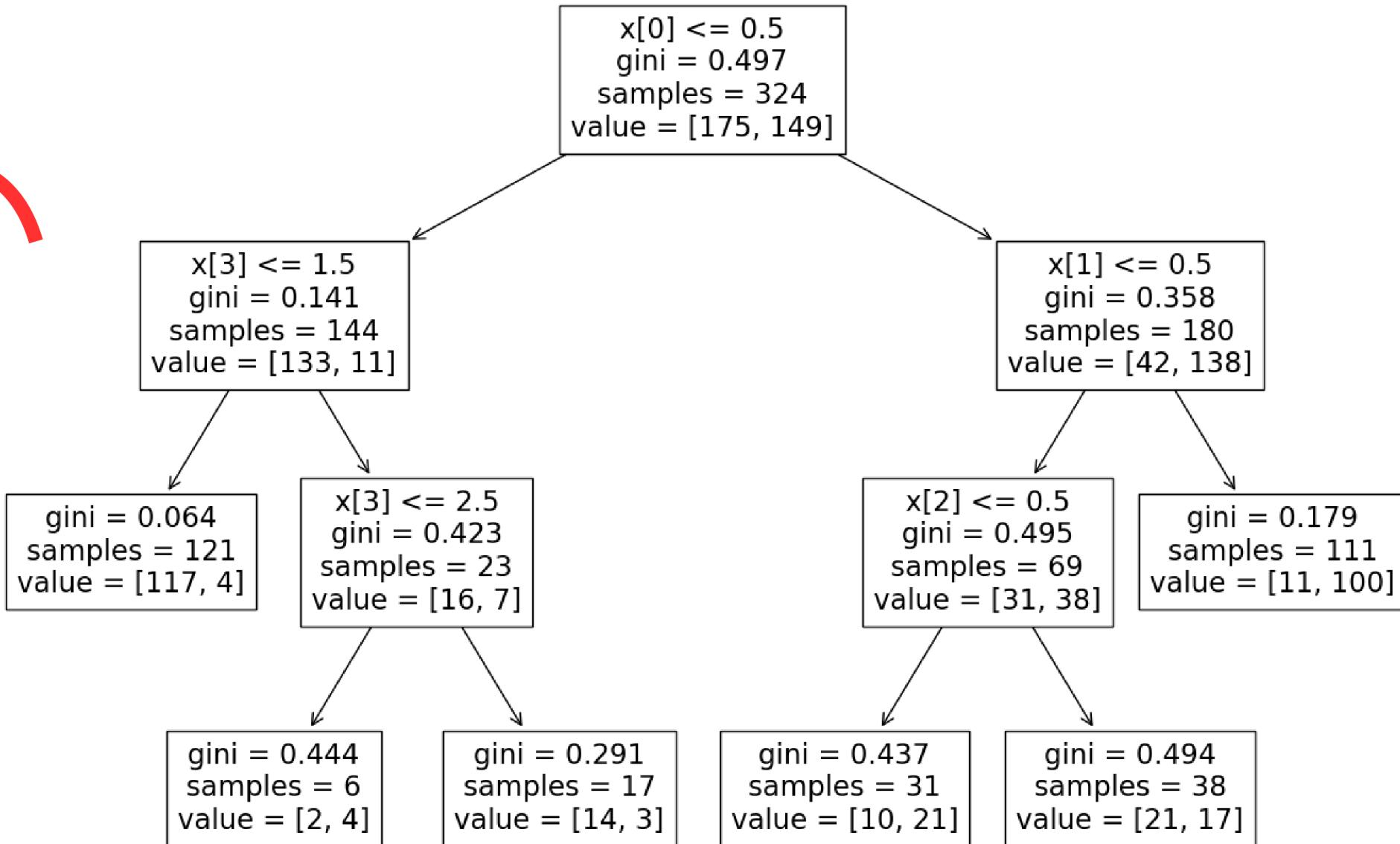
```
y_predict3_5 = Dtree3_5.predict(X_test)
```

```
accuracy_score(y_test, y_predict3_5)
```

```
0.8387978142076503
```

max\_leaf\_nodes=6

```
, ax = plt.subplots(figsize=(15,10))  
tree.plot_tree(Dtree3_5, ax = ax);
```



# อธิบาย

ถัดมา Define ที่ 3 เราจะกำหนดพารามิเตอร์ ออยก์ที่ 6 นะครับ ทำการเรียกใช้ .fit เช่นเดิม เพื่อเทรนโมเดล เมื่อเทรนข้อมูลแล้วเราก็เรียกดู tree.plot ได้เลย

ในตัว treeplot ของโมเดลนี้นะครับ Sample จะเท่ากับ 324 ซึ่งจะเห็นได้ว่าค่อนข้างคล้ายกับโมเดลที่ define ที่ 3

ดังนั้นเมื่อการตัดสินใจลงเรื่อยๆตามความลึกของต้นไม้ จำนวน sample ของ node ในแต่ละชั้นก็จะมี แนวโน้มลดลงเรื่อยๆเช่นเดียวกันและจะเห็นได้ว่า ในความลึกของต้นไม้ จะมีความลึกมากกว่าที่กำหนดพารามิเตอร์ ออยก์ที่ 5

gini หรือค่า Gini impurity = 0.497 ซึ่งแสดงถึงความไม่สมดุลในการแบ่งกลุ่ม

ถัดมาจะเป็น value = [175, 149]: หมายถึงมีจำนวนตัวอย่างที่มีคลาสหรือค่าที่กำหนดได้เท่ากับ 175 ตัวอย่างและ 149 ตัวอย่าง นั่นหมายความว่ามีการแบ่งกลุ่มแบบนี้ในโหนดนี้โดยตรงเช่นเดียวกันนั่นเอง

เมื่อทดสอบแล้วดูที่ค่า accuracy\_score จะได้เท่ากับ 0.8387978142076503

หมายถึงโมเดล Decision Tree Classifier ที่ถูกสร้างขึ้นมีความแม่นยำในการทำนายบนชุดข้อมูล กดสอบประมาณ 83.87%



```
[54] Dtree3_2 = DecisionTreeClassifier(max_leaf_nodes=13)
```

```
[55] Dtree3_2.fit(X_train,y_train)
```

```
DecisionTreeClassifier  
DecisionTreeClassifier(max_leaf_nodes=13)
```

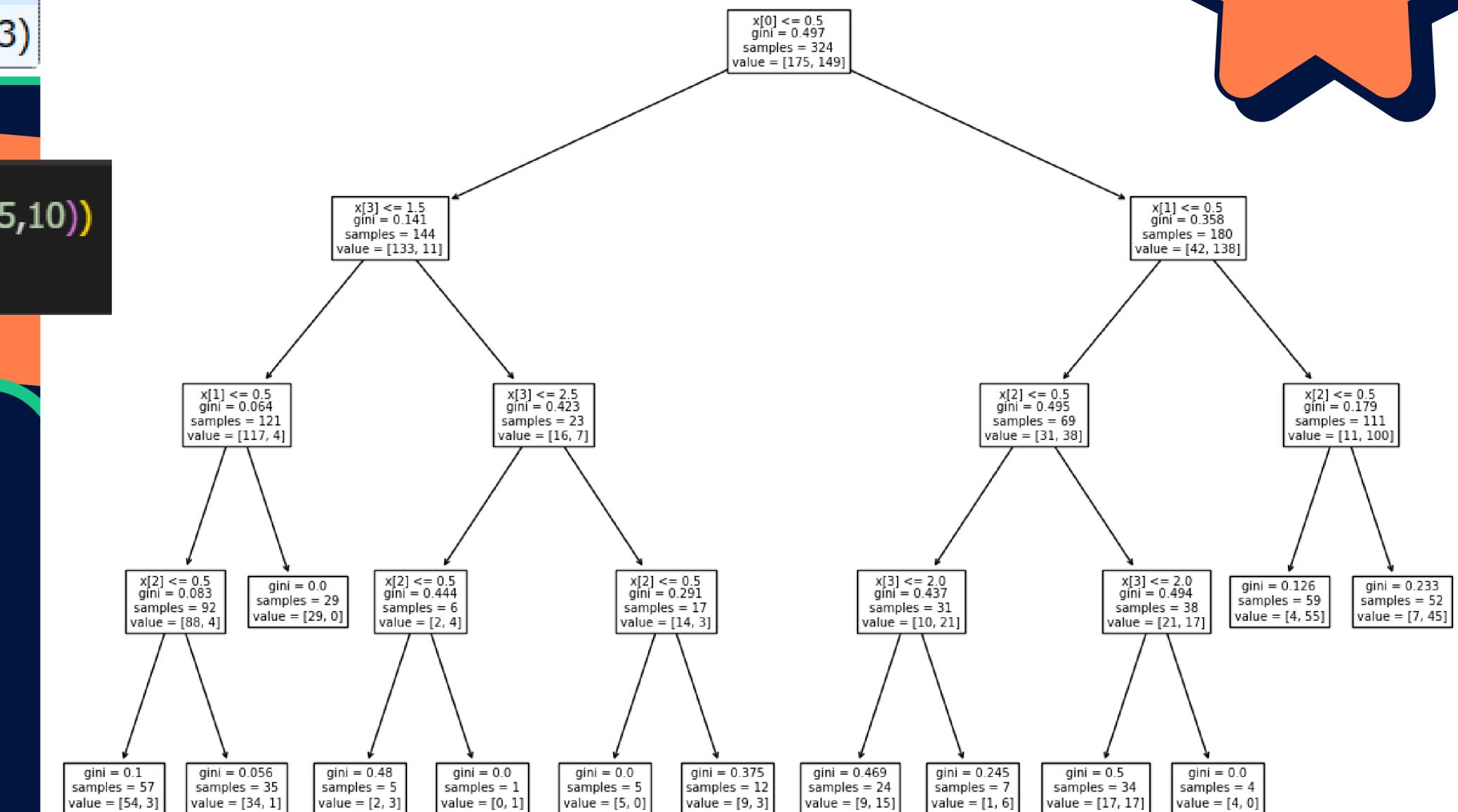
```
_> ax = plt.subplots(figsize=(15,10))  
tree.plot_tree(Dtree3_2);
```

```
y_predict3_2 = Dtree3_2.predict(X_test)
```

```
accuracy_score(y_test, y_predict3_2)
```

```
0.8387978142076503
```

```
er(max_leaf_nodes=13)
```



# อธิบาย

วิจัยวิจัย

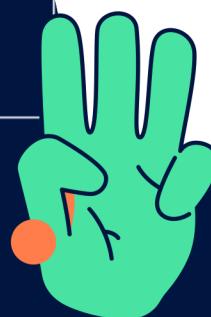


ถัดมาตัวอย่าง Define สุดท้ายนะครับ เรากำหนดพารามิเตอร์อยู่ที่ 13 จากนั้นเทรนเช่นเดียวกันกับพารามิเตอร์ตัวอื่นๆเลยนะครับ

เมื่อเทสแล้วดูที่ค่า accuracy\_score จะได้เท่ากับ 0.8387978142076503 หมายถึงมีความแม่นยำในการคำนายนอนชุดข้อมูลทดสอบประมาณ 83.87% จะเห็นว่า พารามิเตอร์ที่กำหนด 13 มีความแม่นยำในการคำนายนอนเท่ากับพารามิเตอร์ที่กำหนดไว้ที่ 6



# MAX LEAF\_NODES



```
from sklearn.tree import DecisionTreeClassifier
from sklearn.model_selection import GridSearchCV

Dtreett= DecisionTreeClassifier()

param_grid = {'max_leaf_nodes': [2,5,6, 13,]}

grid_search = GridSearchCV(Dtreett, param_grid, cv=11)
```

```
grid_search.fit(X_train, y_train)
```

```
GridSearchCV
estimator: DecisionTreeClassifier
DecisionTreeClassifier
DecisionTreeClassifier()
```

```
best_max_leaf_nodes = grid_search.best_params_['max_leaf_nodes']
```

```
best_max_leaf_nodes
```

```
2
```

# อธิบาย

วิจัยวิจัย



จาก max\_leaf\_nodes ที่เพื่อนๆ พูดมาจะเห็นว่า เมื่อเลือก

max\_leaf nodes ตั้งแต่ 5 ลงมาจนถึง 2 จะได้ค่า accuracy score = 0.8715846994535519 เมื่อันกันทุกค่าและ

max\_leaf nodes ตั้งแต่ 6 ขึ้นไป จะได้ค่า accuracy score = 0.8387978142076503 เมื่อันกันทุกค่าจะ และเพื่อตอกย้ำว่าค่าพารามิเตอร์ที่ 2 เป็นตัวที่เหมาะสม เลี้ยงความเสี่ยงที่จะเกิด Overfitting และลดความซ้ำซ้อนของโมเดล เราเลยใช้อีกโคน้ำเพื่อพิจารณาค่า

นั่นคือ GridSearch ซึ่ง GridSearch ใช้ในการค้นหาค่าพารามิเตอร์ที่ดีที่สุดสำหรับโมเดลเฉพาะ โดยทำการทดลองค่าพารามิเตอร์ต่าง ๆ ในช่วงที่กำหนด และเลือกค่าที่ให้ประสิทธิภาพที่ดีที่สุดโดยใช้วิธี การทดสอบที่กำหนด ซึ่งตัวที่เรากำหนดให้ GridSearch ทดสอบ ก็คือที่ตัวที่เพื่อนๆ ได้พูดไปค่ะ นั่นคือ 2 5 6 และ 13 ซึ่งเมื่อเทียบอุปกรณ์แล้วจะพบว่า GridSearch ก็แสดงผลให้เห็นว่า พารามิเตอร์ที่ถูกกำหนดที่ 2 เป็น best\_max\_leaf\_nodes ซึ่งเป็นการ define ที่ดีที่สุด

### Note

gini มีไว้เพื่อปั่งชี้ความบริสุทธิ์ของโหนด หรือพูดง่ายๆคือ โหนดนั้นคือตัวเดียวที่กันกั้งโหนด ถ้า gini เท่ากับ 0 คือข้อมูลทุกรายการในโหนดนั้นอยู่ในคลาสเดียวกันทุกไฟล์ ถ้า giniig เท่ากับ 0.5 ข้อมูลอยู่ใน 2 คลาสเท่าๆกันโดยจะแสดง ผ่าน Value เช่น Value = 12 , 18 แปลว่า จากข้อมูล 30 รายการ จะอยู่โหนดฝั่งขวา 12 รายการ และโหนดฝั่งซ้าย 18 รายการ

TW Industries



THANK YOU  
SO MUCH

กลุ่มกลุ่ม

Classification



หอยหลอดดกรูป

