

HOMEWORK 3

DECISION TREE CLASSIFIER

sklearn.tree

▼ HW3

ให้แต่ละกลุ่ม สร้างต้นไม้ของตัวเอง โดยปรับเปลี่ยน parameters ให้ และ มาอธิบาย parameter

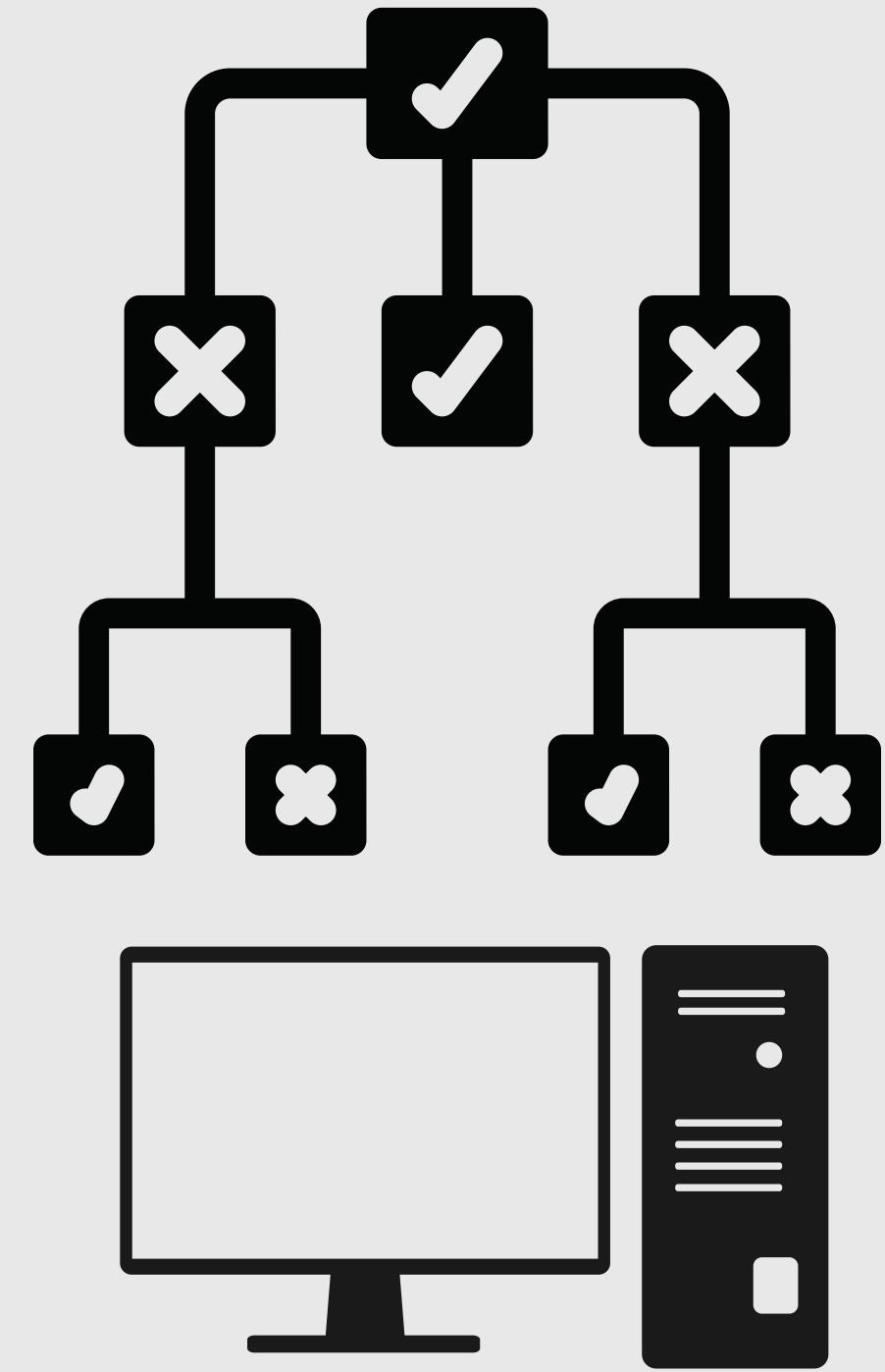
- แคล 1 criterion, max_leaf_nodes
- แคล 2 splitter, min_samples_split
- แคล 3 min_samples_leaf, max_features
- แคล 4 min_impurity_decrease
- แคล 5 min_weight_fraction_leaf



NMN & ក្រុងក្រាម

MEMBER

- 01 นางสาวเพ็ญพิชชา วรรณชุมาตร 643020063-4
- 02 นายภัทรรร ก้อนมนี 643020513-9
- 03 นายรัชชานนท์ กิจพิมานพ 643020515-5
- 04 นายวุฒิชัย คำนา 643020521-0
- 05 นางสาววรศรา ปันลา 643020519-7
- 06 นางสาวอนัญญา พูลสวัสดิ์ 643020526-0
- 07 นางสาววิลันดา ทารามาตร 643021271-2



HW3

MIN_SAMPLES_SPLIT

The minimum number of samples required to split an internal node:

SPLITTER

The strategy used to choose the split at each node. Supported strategies are “best” to choose the best split and “random” to choose the best random split.



MIN_SAMPLES_SPLIT

```
DtreeHW2 = DecisionTreeClassifier(random_state=0,max_depth=3,min_samples_split=323)
```

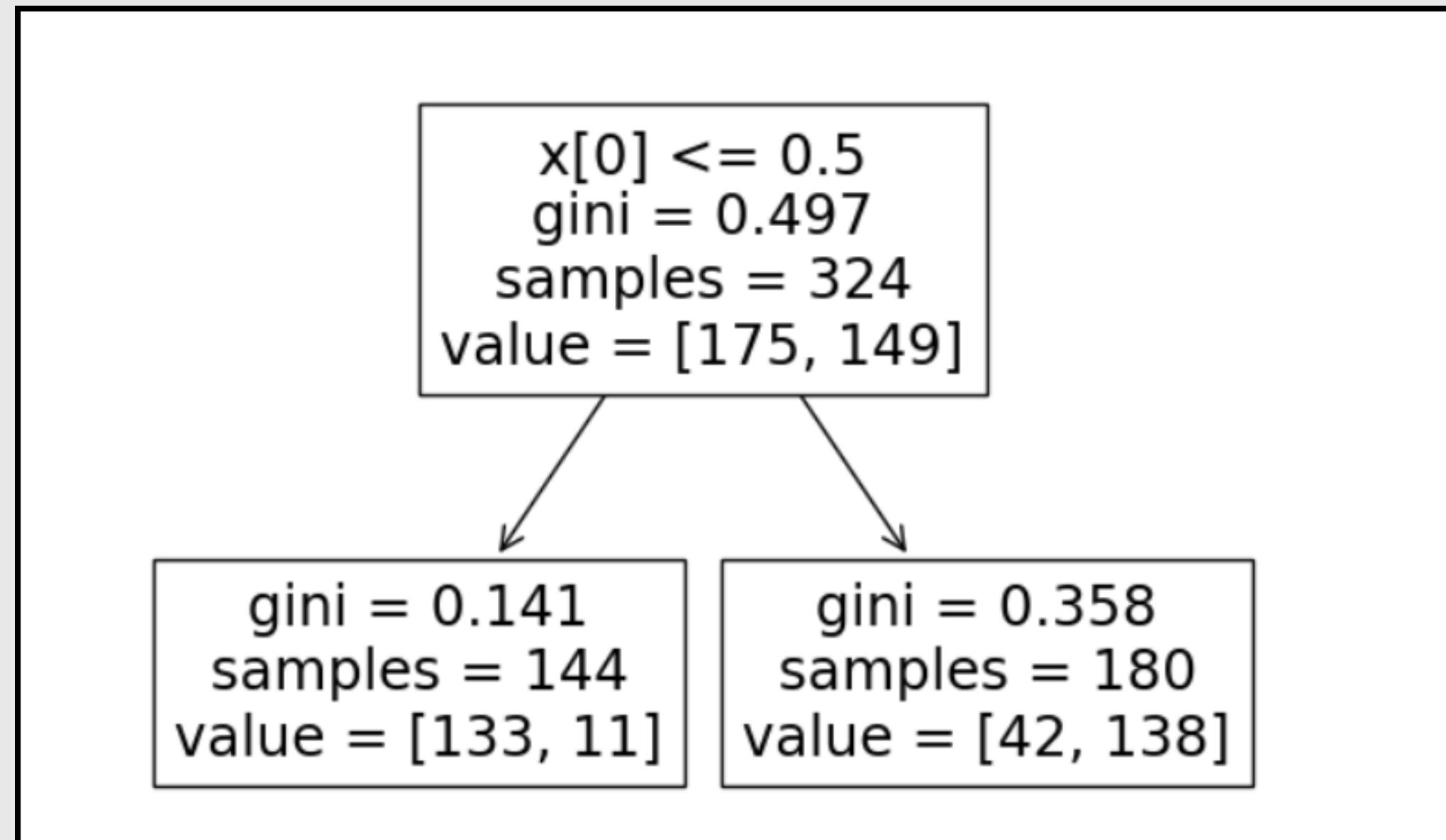
```
[60] DtreeHW2.fit(X_train,y_train)
```

```
▼ DecisionTreeClassifier  
DecisionTreeClassifier(max_depth=2, min_samples_split=323, random_state=0)
```



```
tree.plot_tree(DtreeHW2);
```

DECISIONTREE

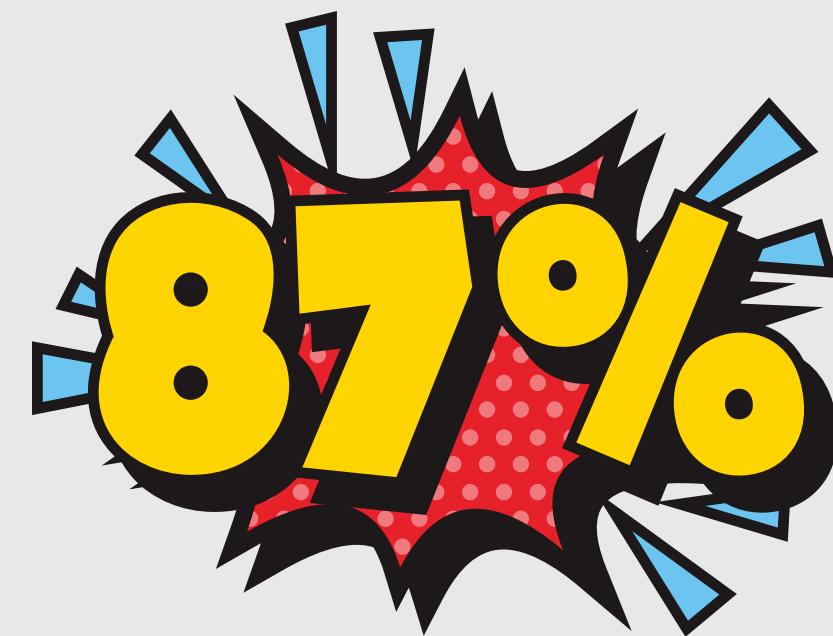


DECISIONTREE

```
✓ 0s [44] y_predictHW2 = DtreeHW2.predict(x_test)
```

```
✓ 0s [45] accuracy_score(y_test, y_predictHW2)
```

0.8715846994535519



SPLITTER "BEST"

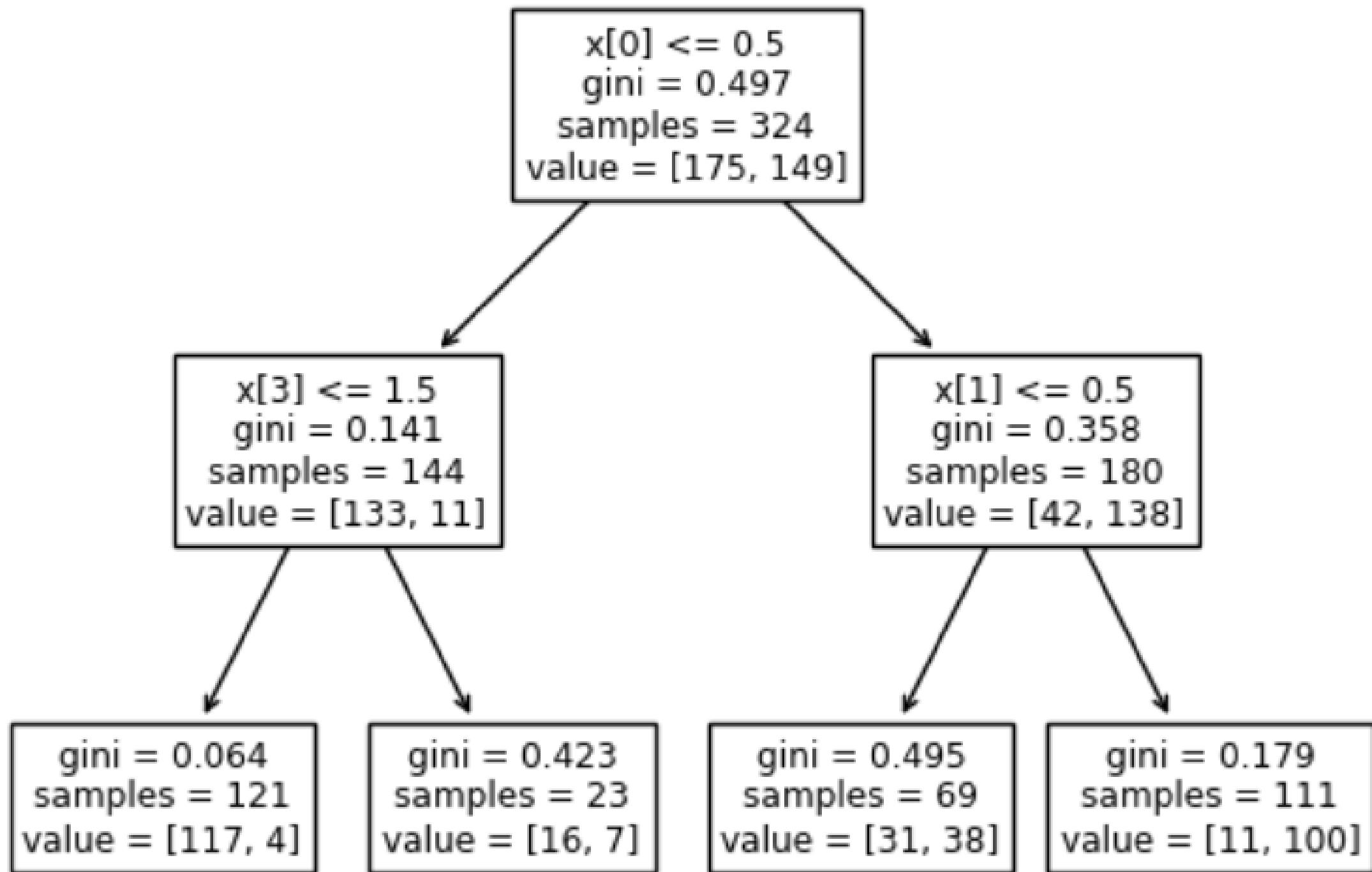
```
DtreeHW2_1 = DecisionTreeClassifier(random_state=0,max_depth=2,splitter="best")
```

```
DtreeHW2_1.fit(X_train,y_train)
```

```
▼      DecisionTreeClassifier  
DecisionTreeClassifier(max_depth=2, random_state=0)
```

DECISION TREE

```
tree.plot_tree(DtreeHW2_1);
```



SPLITTER "BEST"

```
y_predictHW2_1 = DtreeHW2_1.predict(X_test)  
  
accuracy_score(y_test, y_predictHW2_1)  
  
0.8715846994535519
```

SPLITTER "RANDOM"

```
DtreeHW2_1 = DecisionTreeClassifier(random_state=0,max_depth=2,splitter="random")
```

```
DtreeHW2_1.fit(X_train,y_train)
```

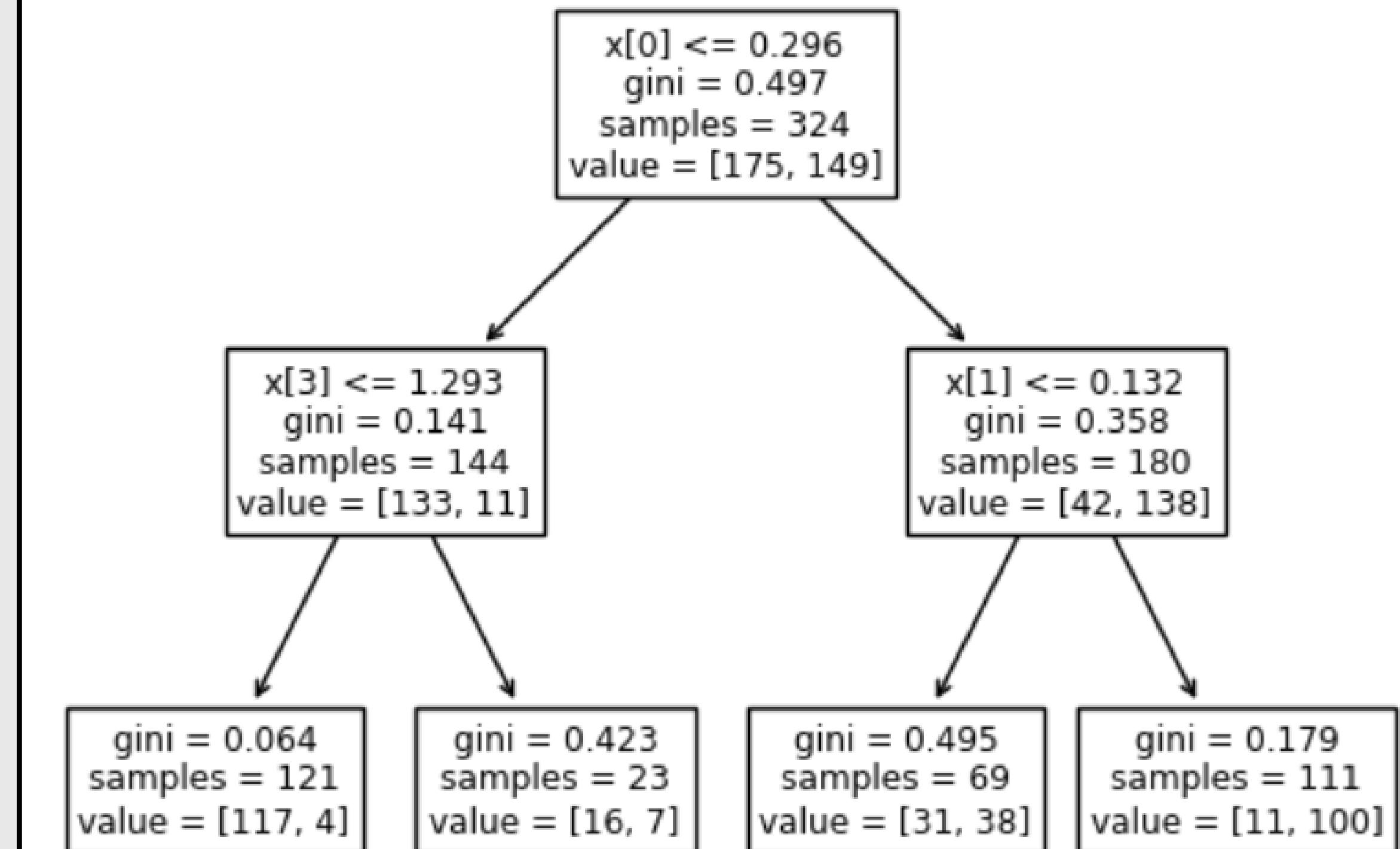
```
▼
```

```
DecisionTreeClassifier
```

```
DecisionTreeClassifier(max_depth=2, random_state=0, splitter='random')
```

DECISION TREE

```
tree.plot_tree(DtreeHW2_1);
```



SPLITTER "RANDOM"

```
y_predictHW2_1 = DtreeHW2_1.predict(X_test)
```

```
accuracy_score(y_test, y_predictHW2_1)
```

```
0.8715846994535519
```



THANK YOU

