Decision Tree

Liang Liang

Decision Tree Example

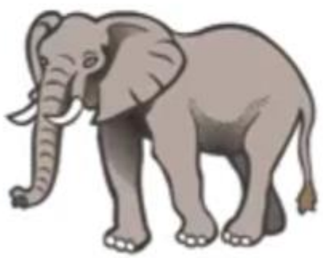
Assume a person only knows the following 6 objects:

Car

Bus

Airplane

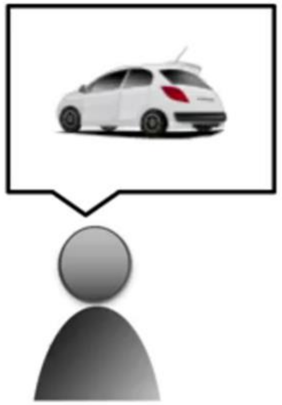
Bird



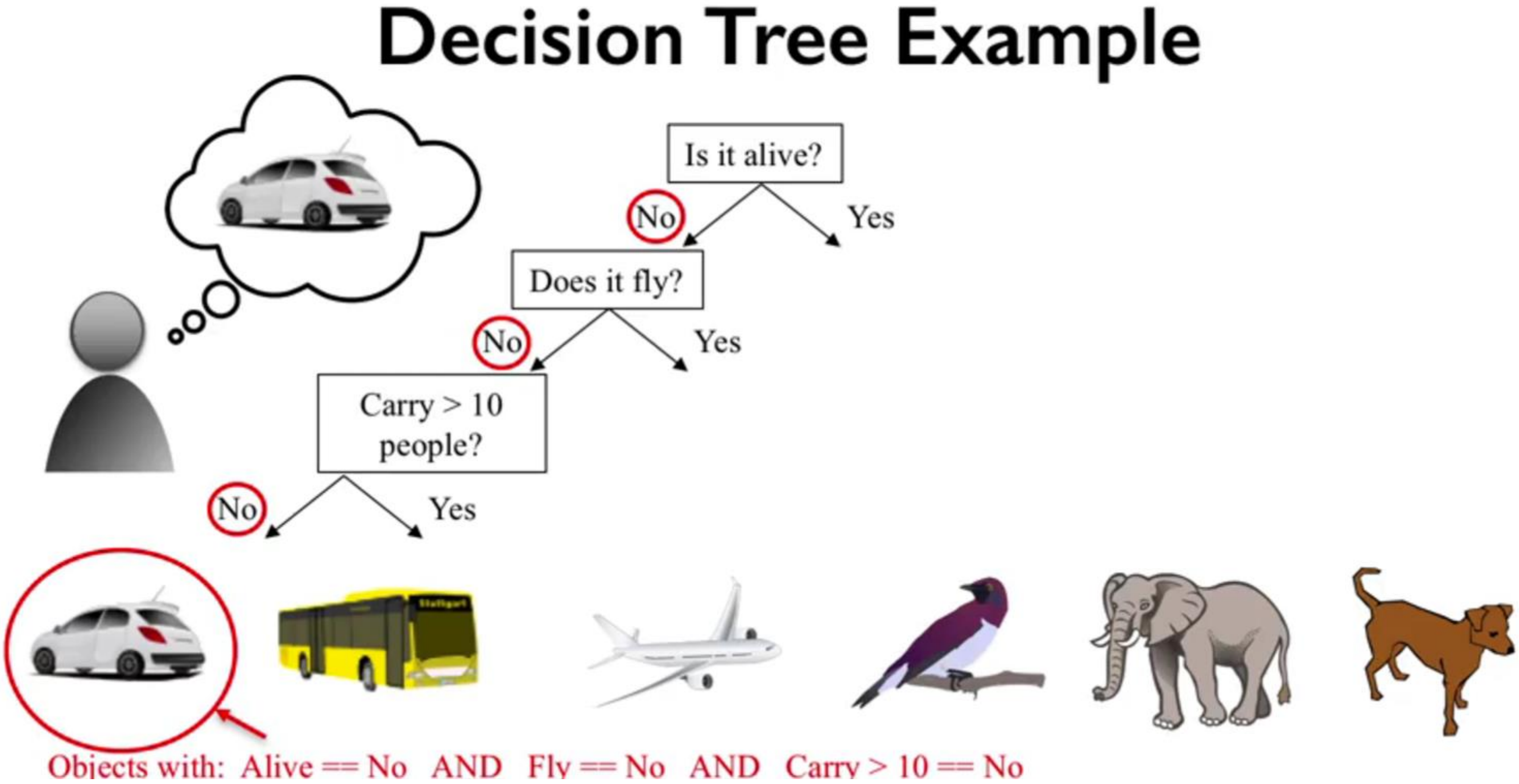
Elephant

Dog

One day, a person saw a picture of an object, then this guy tried to figure out the name of the object using a decision tree.



The person asked many yes-no questions



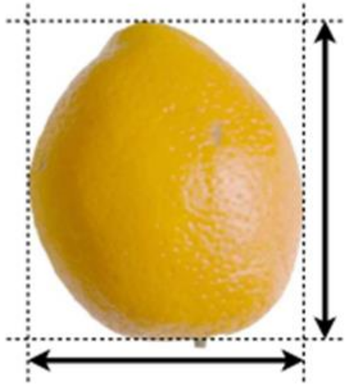
A data sample has three binary features: [alive, fly, carry_more_than_10]

Decision Tree Example

Feature Vector x

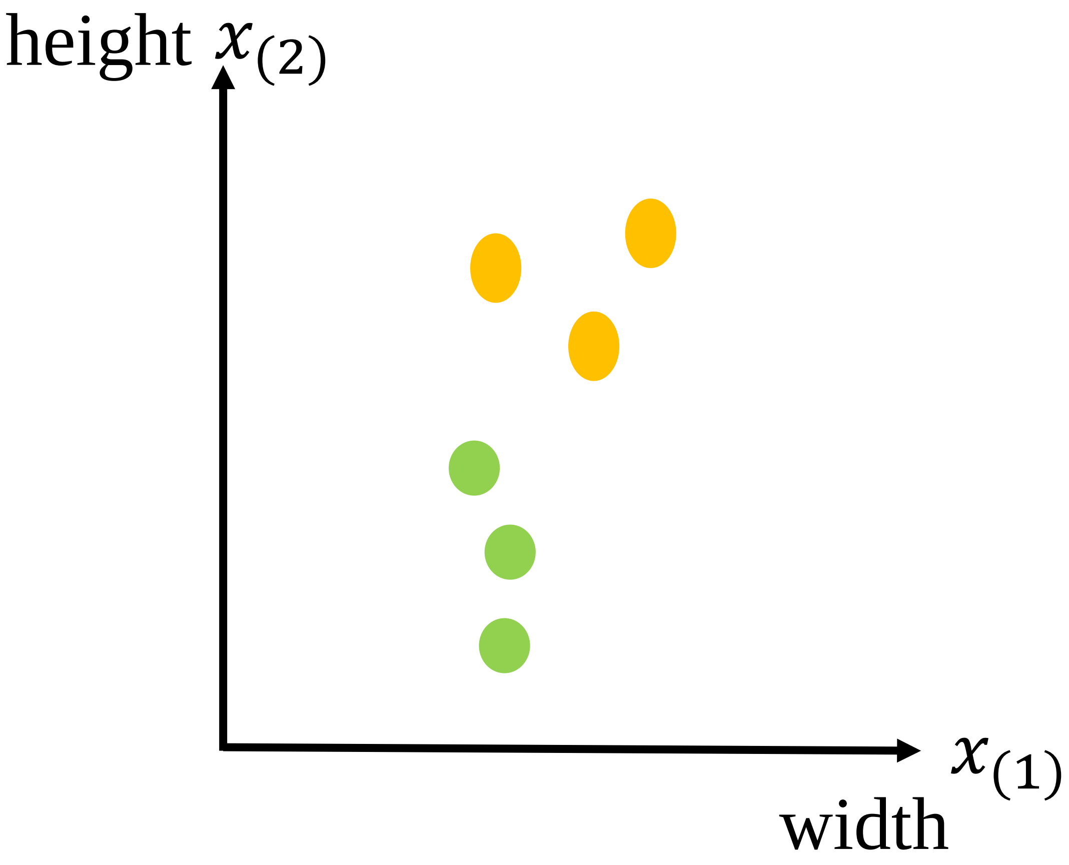
$$x = \begin{bmatrix} x_{(1)} \\ x_{(2)} \end{bmatrix} \qquad \begin{matrix} \text{width} \\ \text{height} \end{matrix}$$

$y \quad = 1$ It is an apple

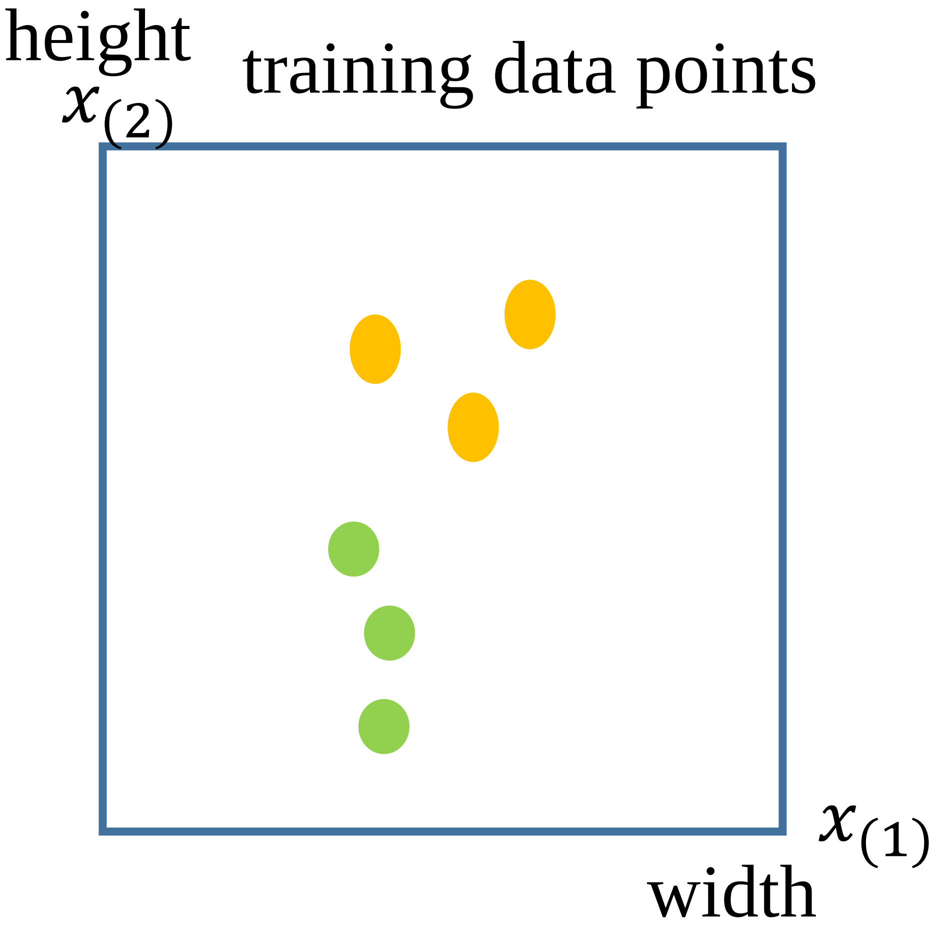


$y \quad = 0$ It is a lemon

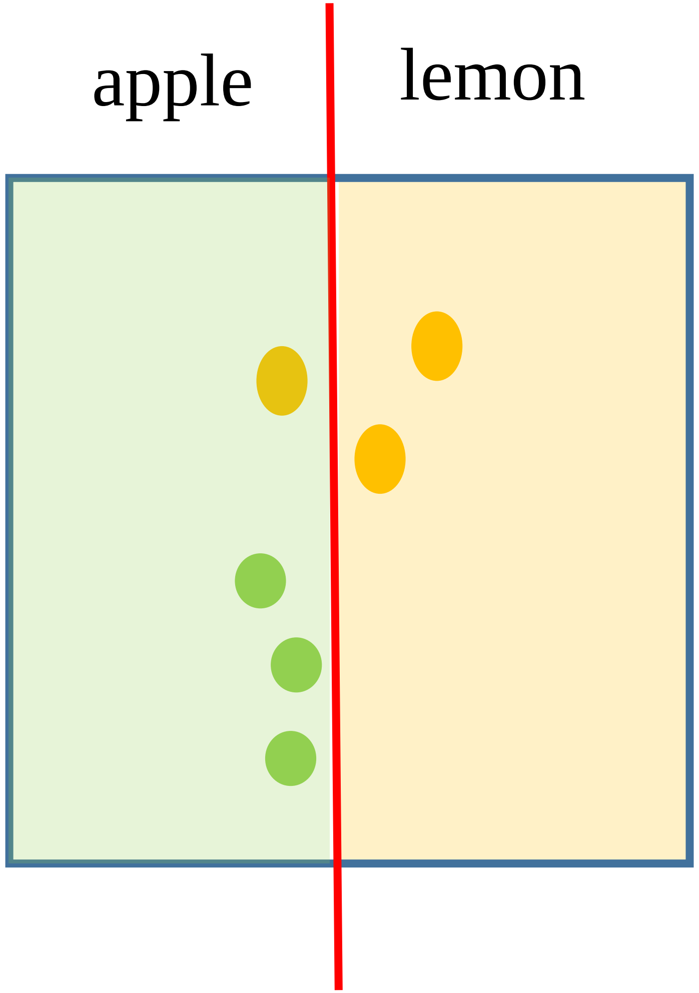
Feature Space (Input Space)



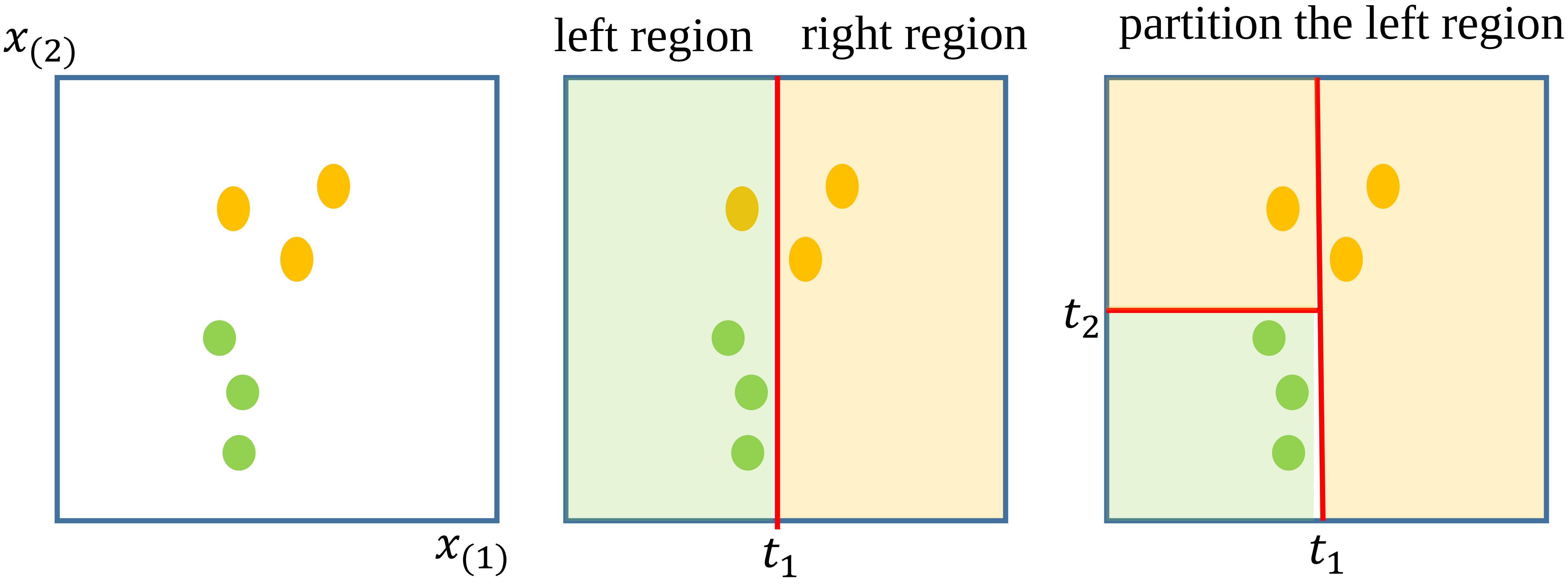
Decision Tree Example



split the feature space



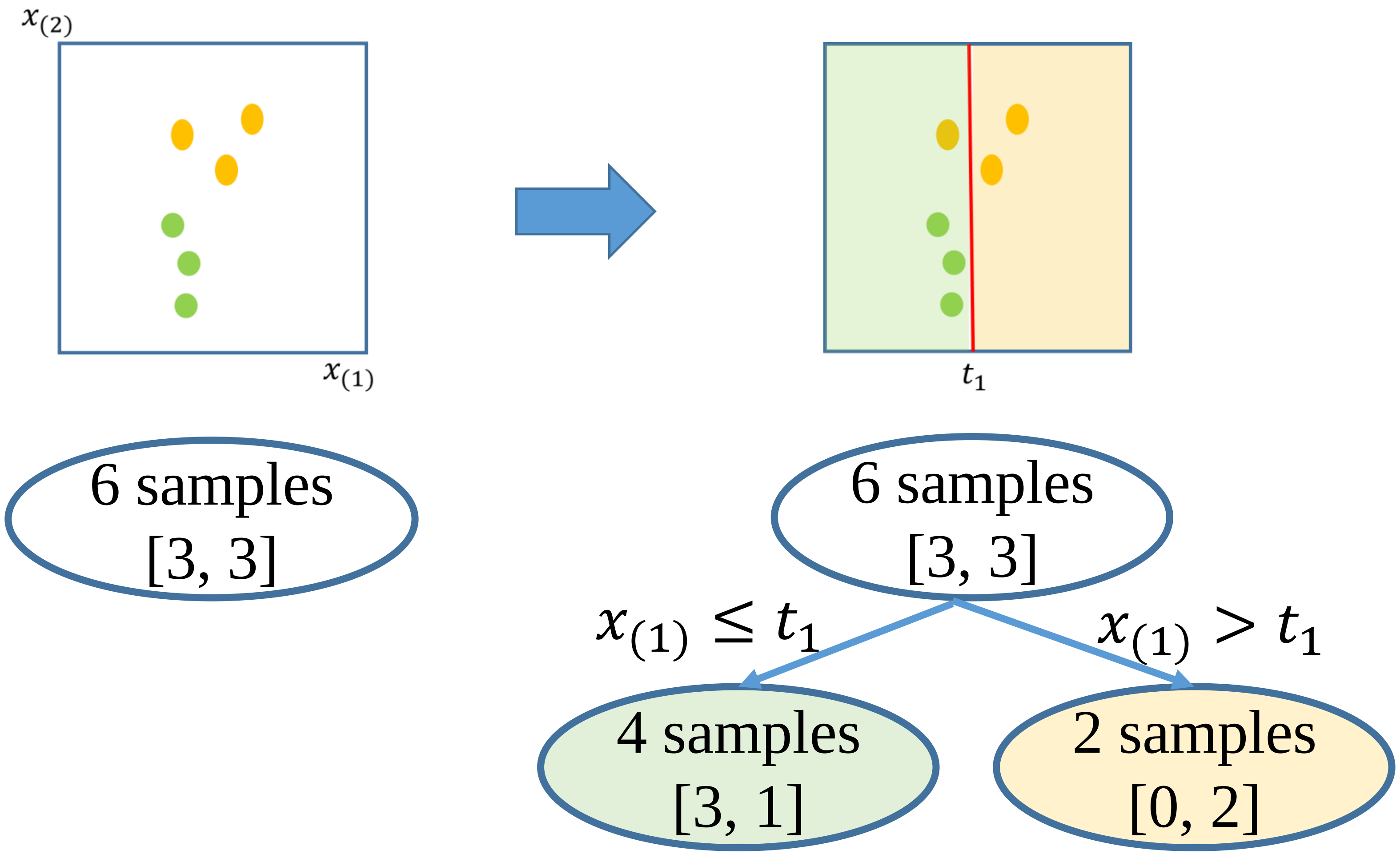
$x_{(1)} \leq t_1$: apple (majority voting)
because there are 3 apples and 1 lemon



a sample has two features $x_{(1)}$ and $x_{(2)}$

$x_{(1)} \leq t_1$: apple
$x_{(1)} > t_1$: lemon

Given $x_{(1)} \leq t_1$,
$x_{(2)} \leq t_2$: apple
$x_{(2)} > t_2$: lemon



6 samples
[3, 3]

6 samples
[3, 3]

$x_{(1)} \leq t_1$

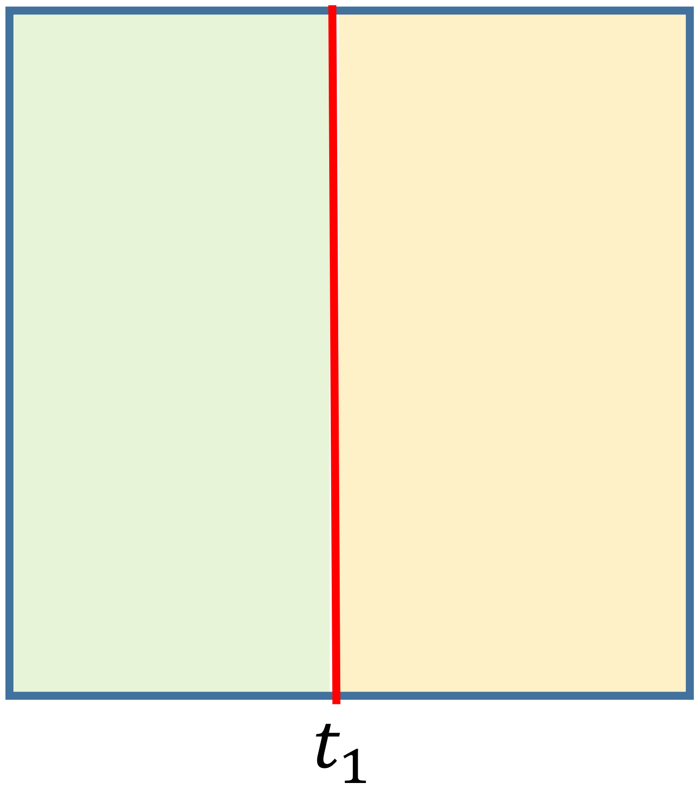
$x_{(1)} > t_1$

4 samples
[3, 1]

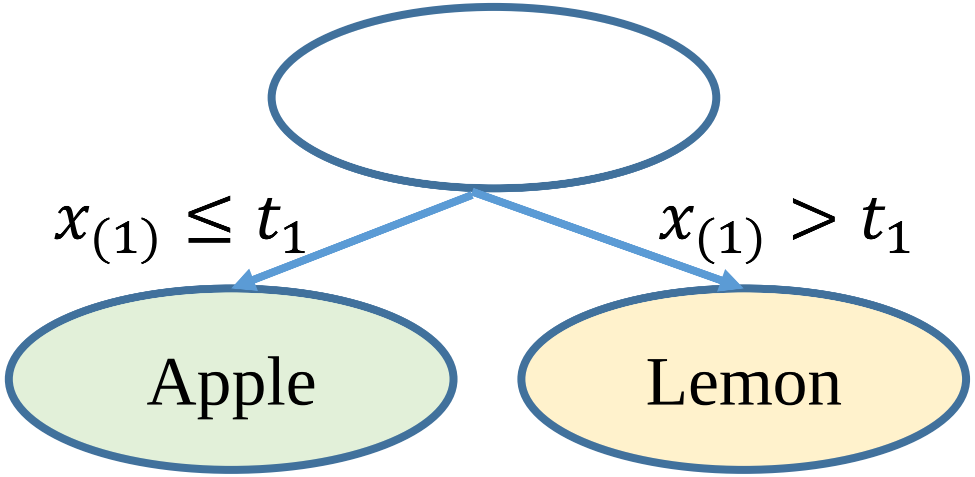
2 samples
[0, 2]

a tree is growing
during training

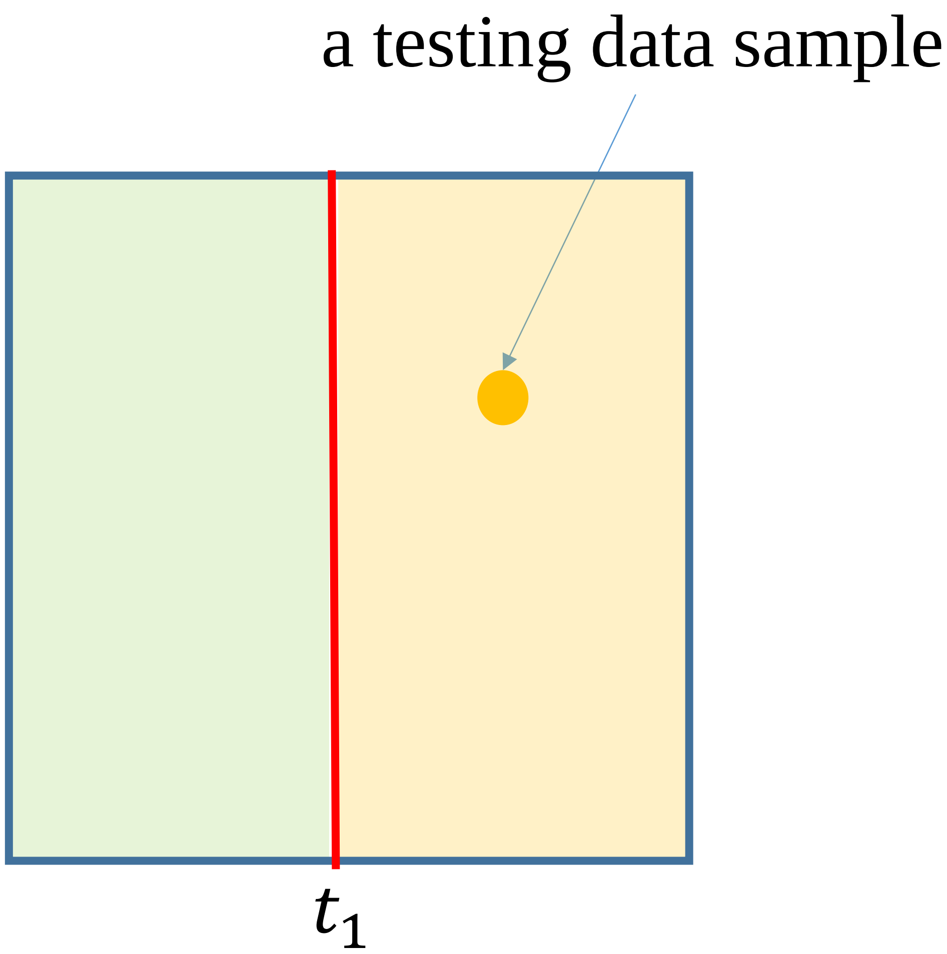
partition of input space

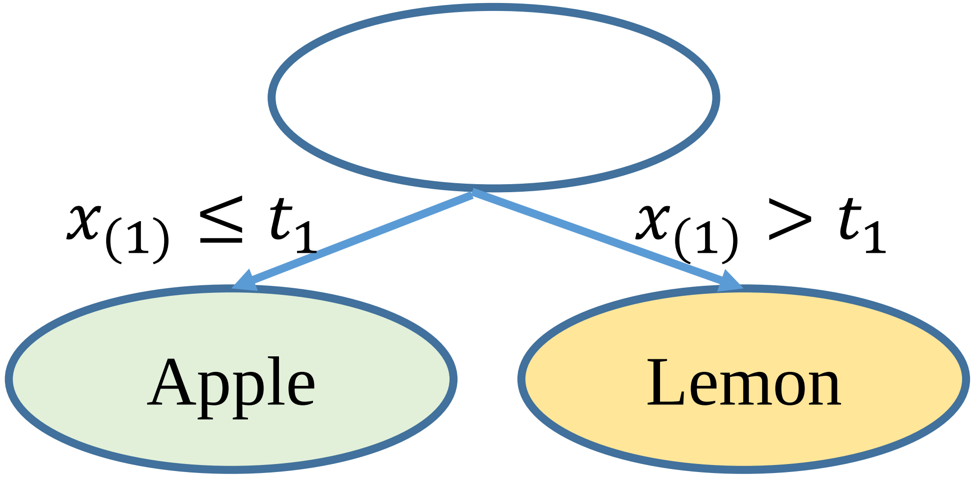


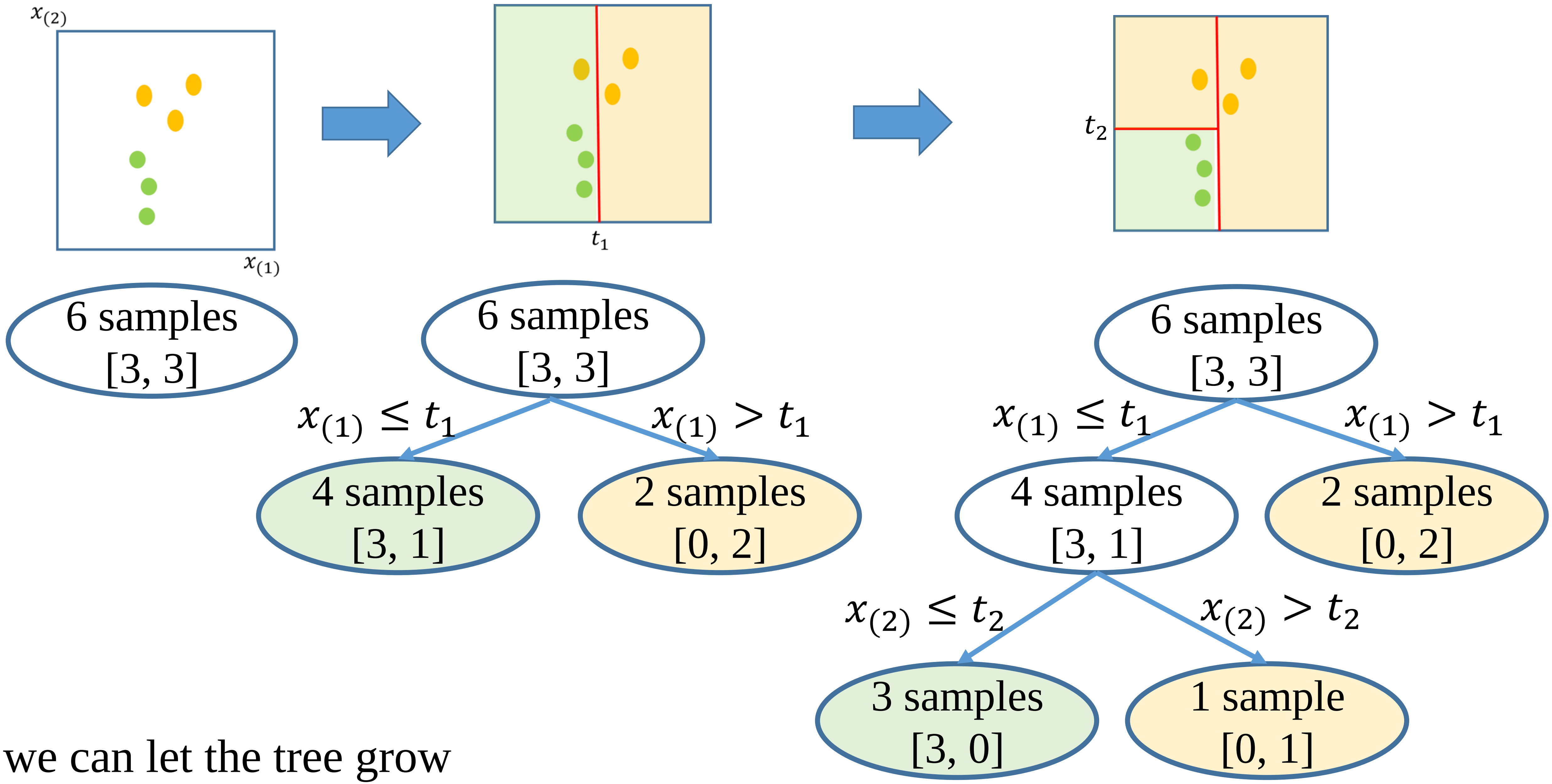
the decision tree T1



test the decision tree T1







$x_{(2)}$

$x_{(1)}$

t_1

t_2

6 samples
[3, 3]

6 samples
[3, 3]

$x_{(1)} \leq t_1$

$x_{(1)} > t_1$

4 samples
[3, 1]

2 samples
[0, 2]

6 samples
[3, 3]

$x_{(1)} \leq t_1$

$x_{(1)} > t_1$

4 samples
[3, 1]

2 samples
[0, 2]

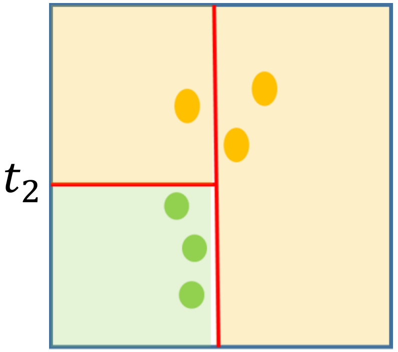
$x_{(2)} \leq t_2$

$x_{(2)} > t_2$

3 samples
[3, 0]

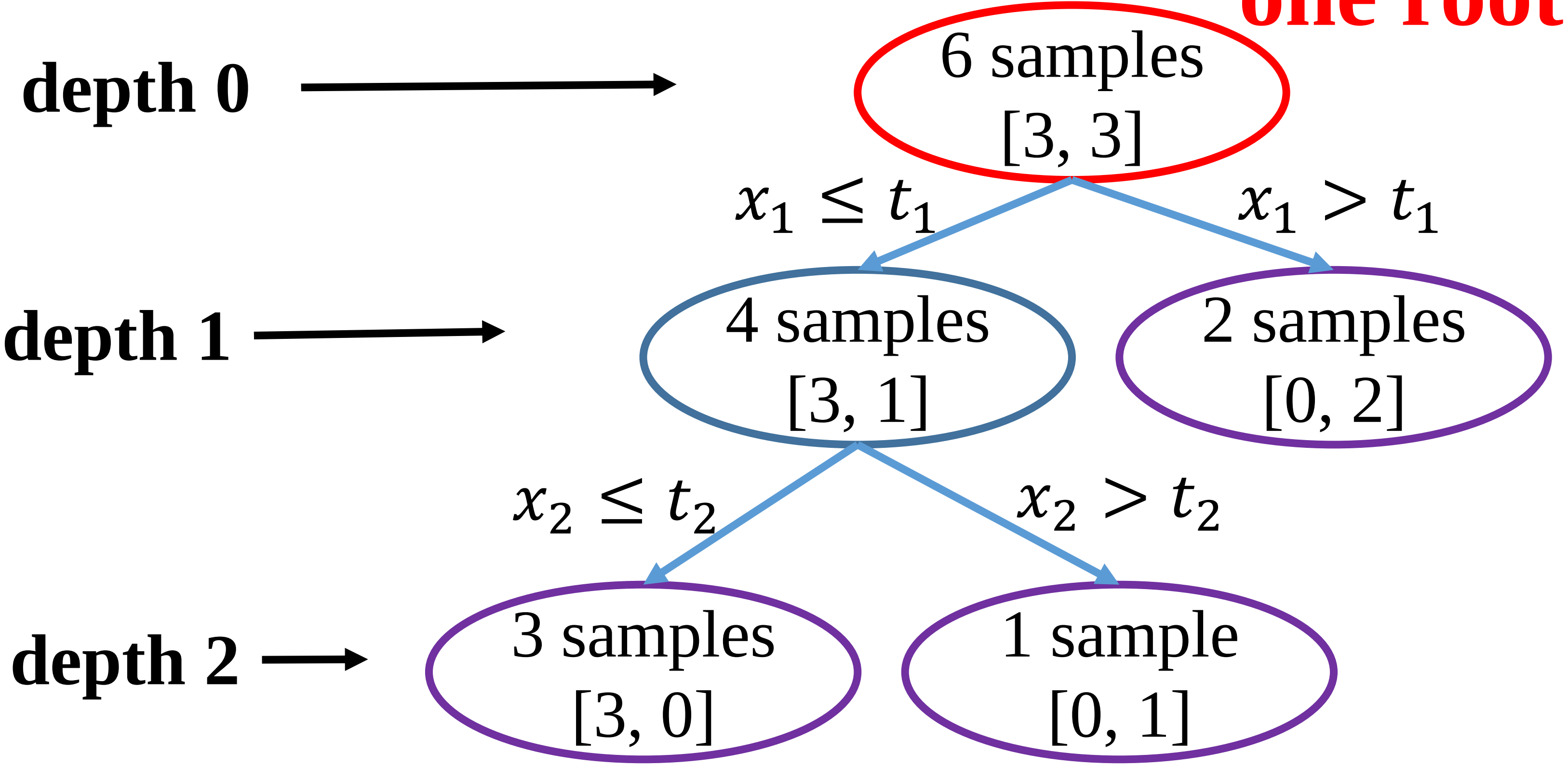
1 sample
[0, 1]

we can let the tree grow another layer



We can further expand this tree such that every leaf node only contains one data sample

one root node



depth 0

6 samples
[3, 3]

$x_1 \leq t_1$

$x_1 > t_1$

depth 1

4 samples
[3, 1]

2 samples
[0, 2]

$x_2 \leq t_2$

$x_2 > t_2$

depth 2

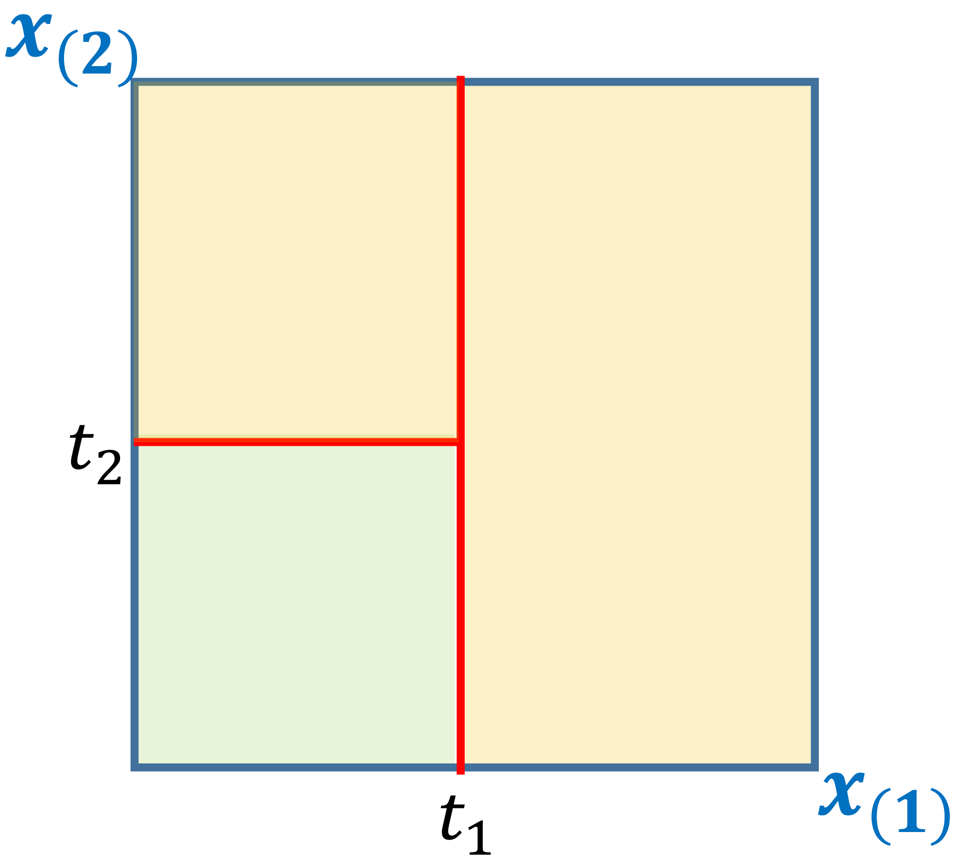
3 samples
[3, 0]

1 sample
[0, 1]

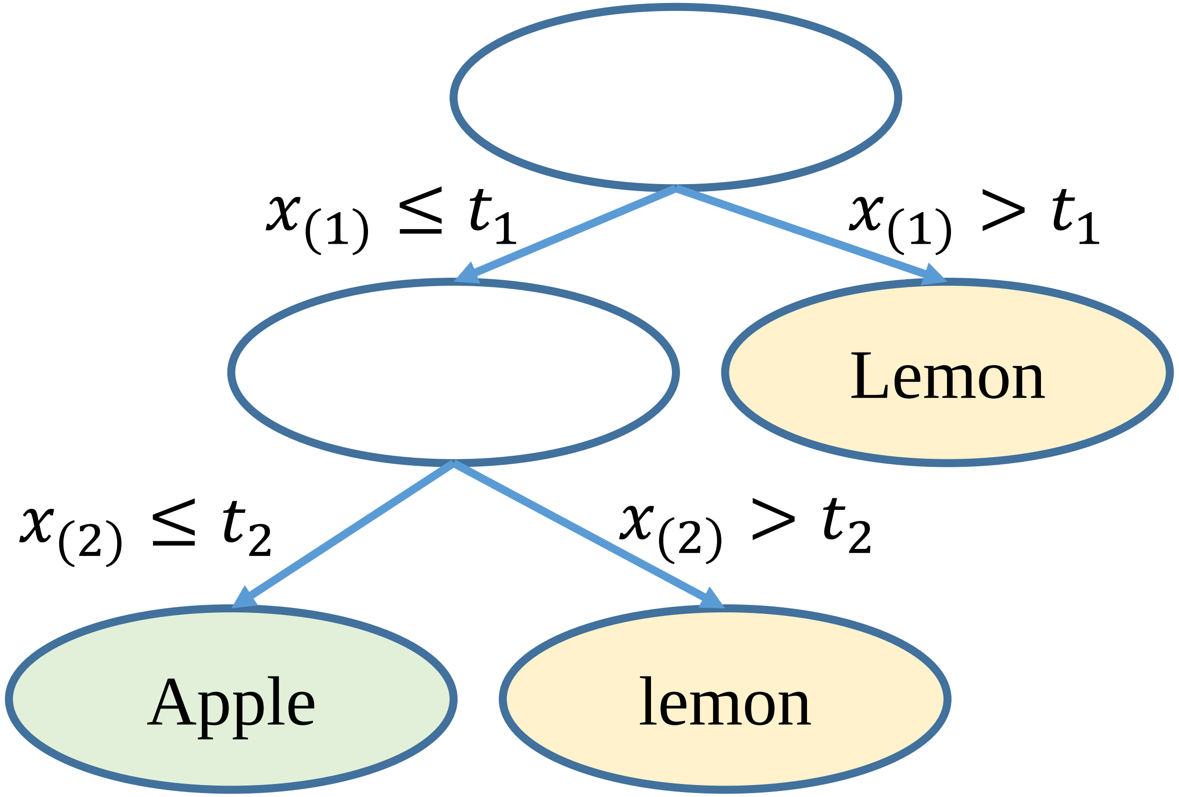
3 leaf (terminal) nodes do not have child nodes

The depth of the tree is 2

partition of input space



the decision tree T2



$x_{(1)} \leq t_1$

$x_{(1)} > t_1$

Lemon

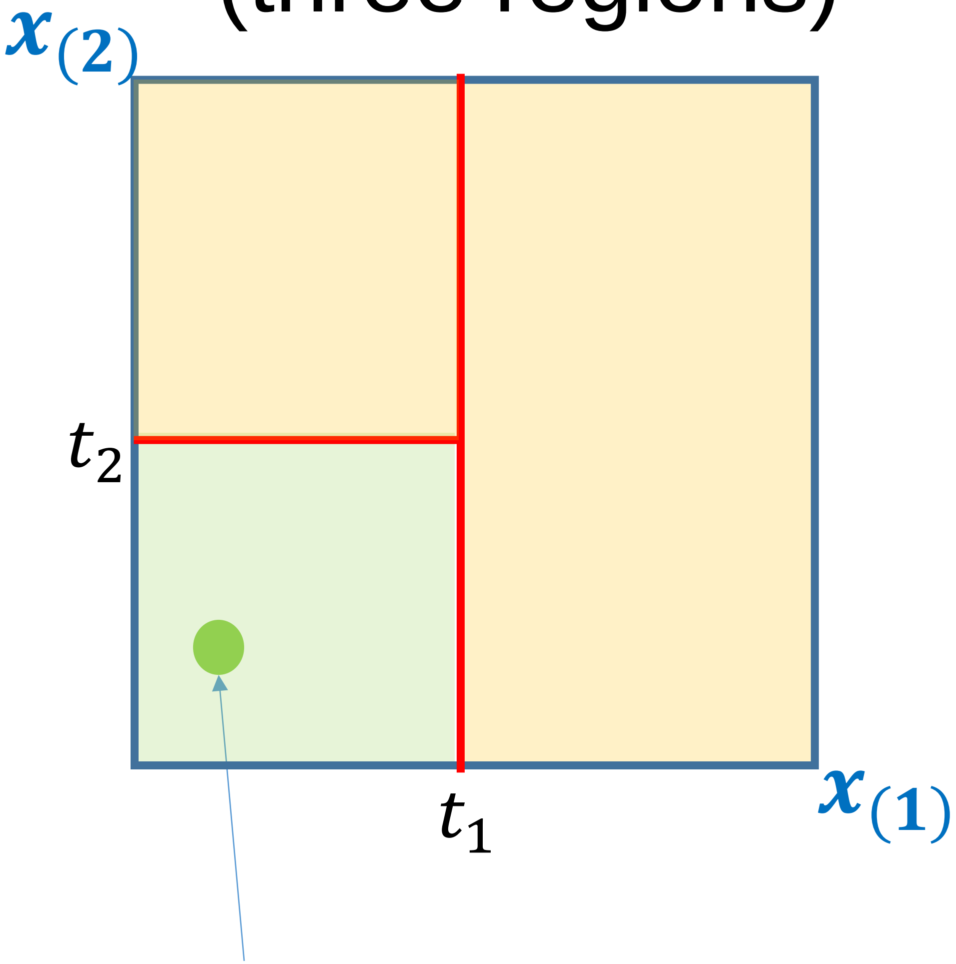
$x_{(2)} \leq t_2$

$x_{(2)} > t_2$

Apple

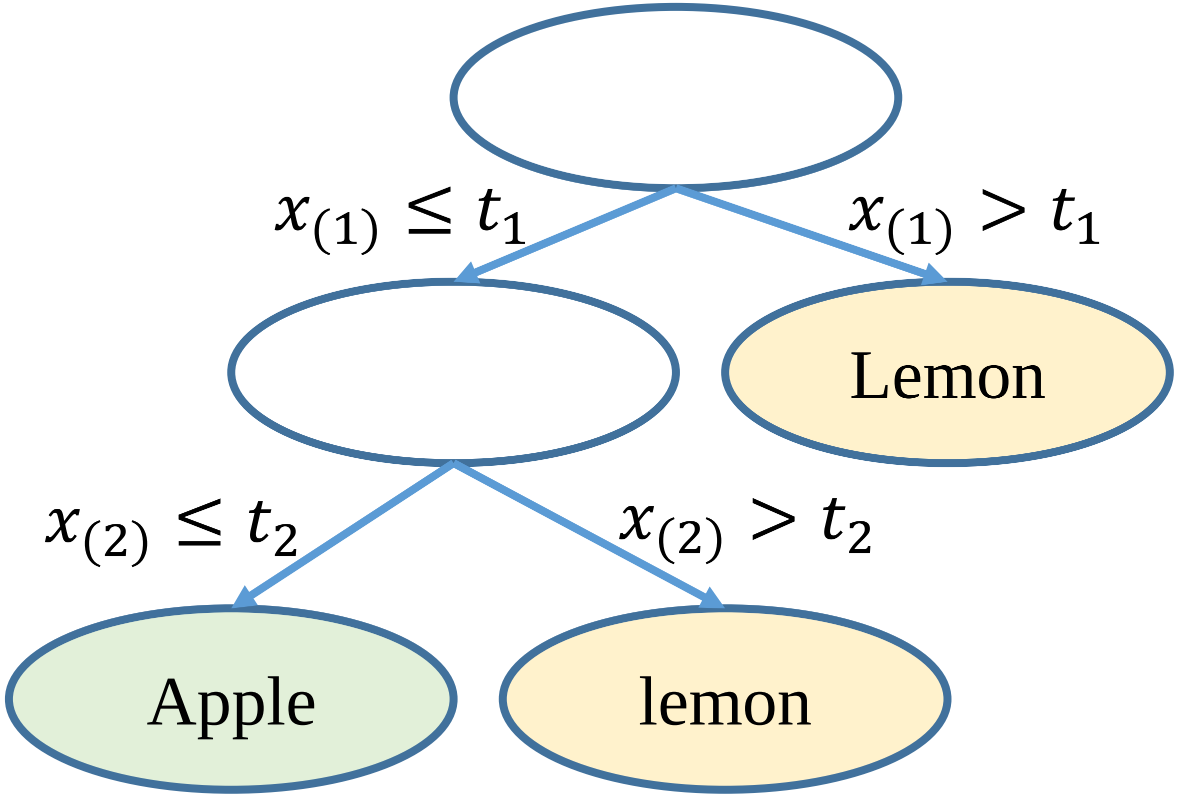
lemon

partition of input space (three regions)



a new data sample x

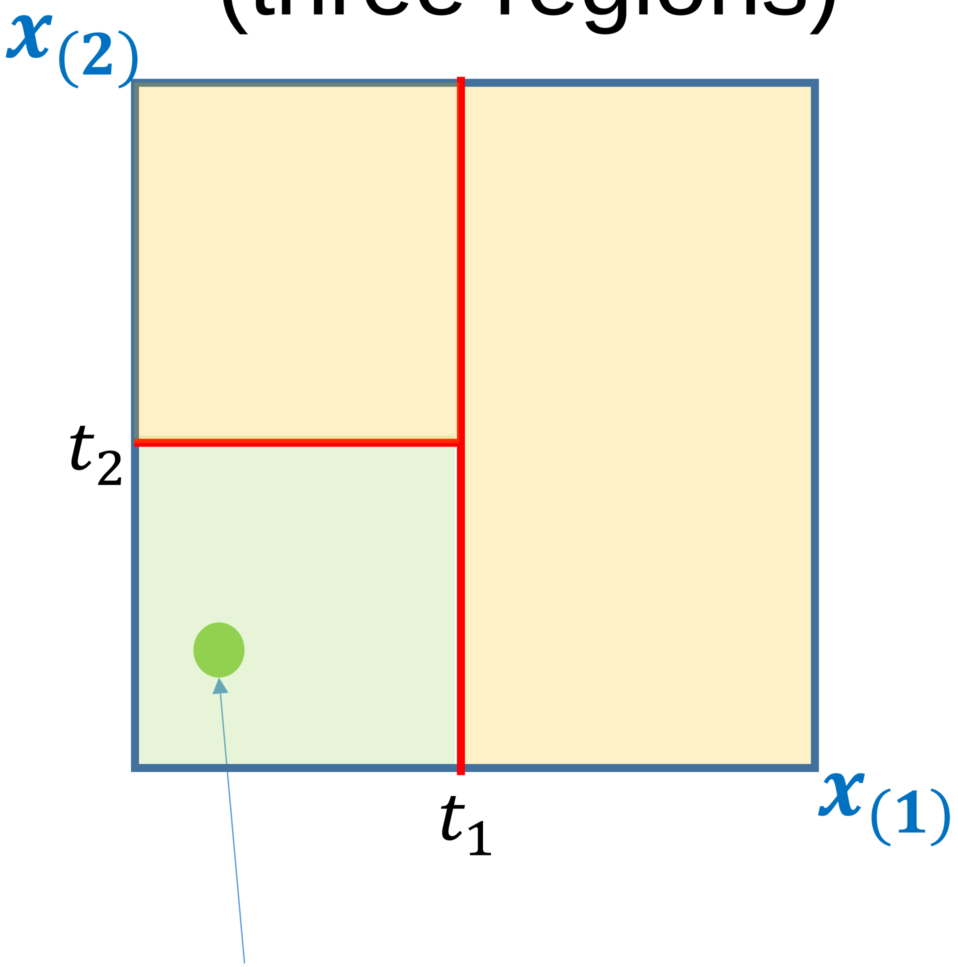
use the decision tree T2

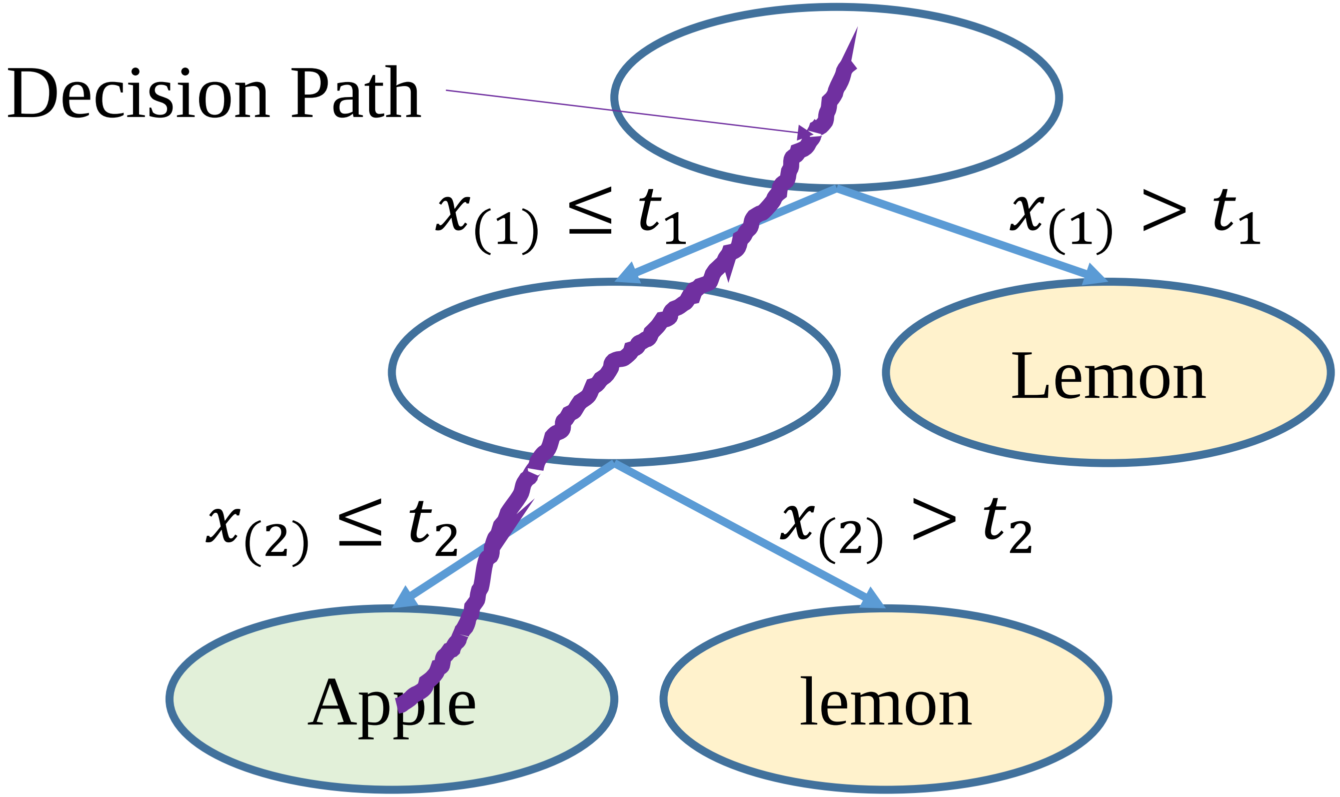


a region corresponds to a leaf note in the tree

partition of input space (three regions)

use the decision tree T2

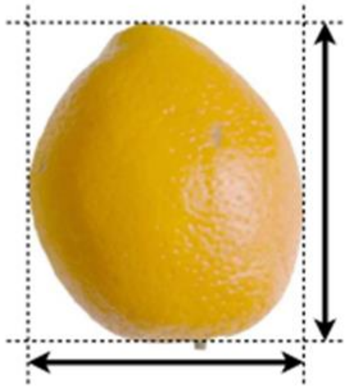




a new data sample x

The data sample x falls into a region/node

Apply Decision Tree to the fruit dataset



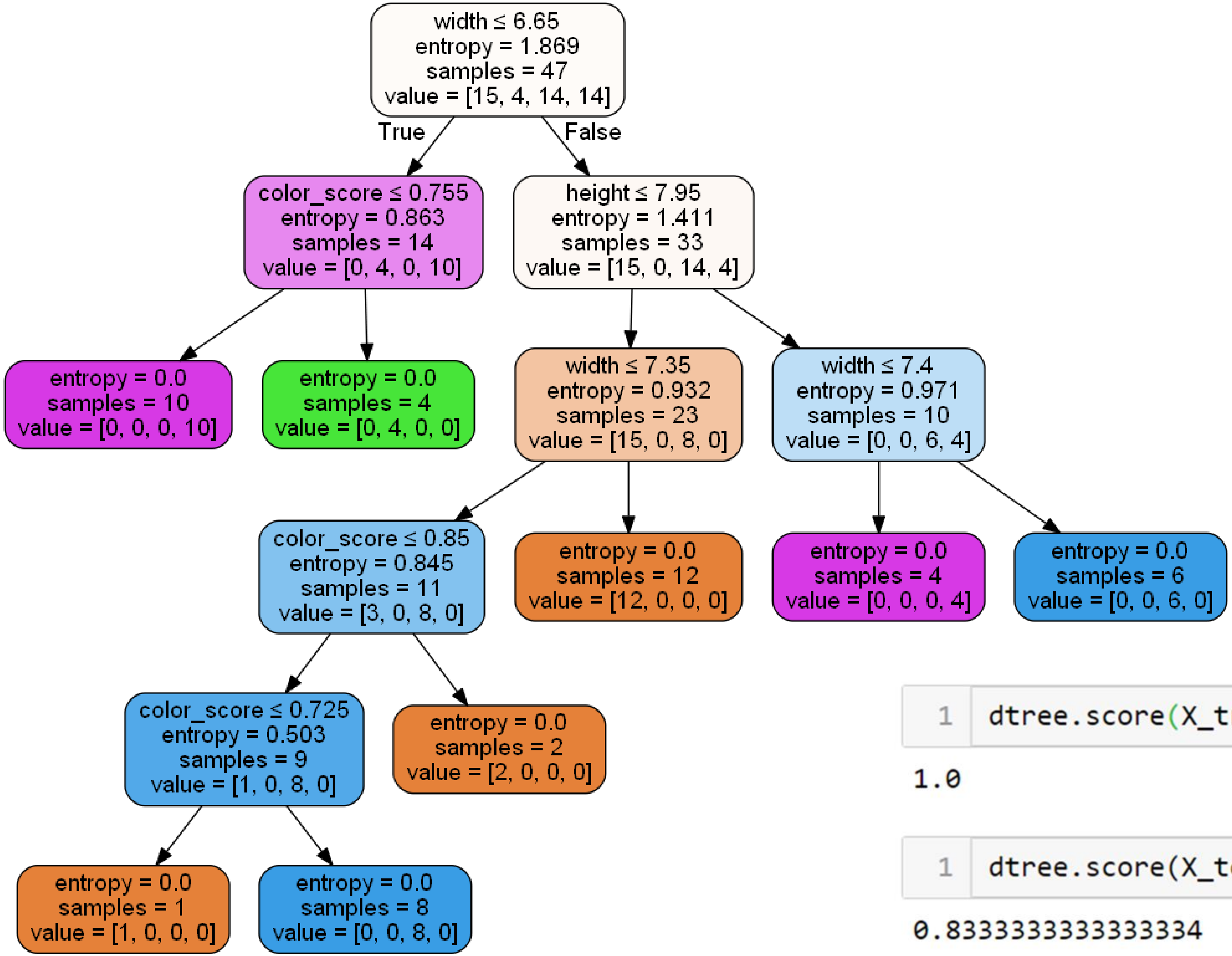
The fruits dataset (a large table)
Each row contains the information of a fruit sample/instance

fruit label	fruit_name	subtype	mass (g)	width (cm)	height (cm)	color_score
1	apple	granny_smith	192	8.4	7.3	0.55
4	lemon	spanish_belsan	194	7.2	10.3	0.70

http://usapple.org/the-industry/apple-varieties/

The feature vector of a fruit sample: [width, height, color_score]

1:apple,
2:mandarin
3:orange
4:lemon



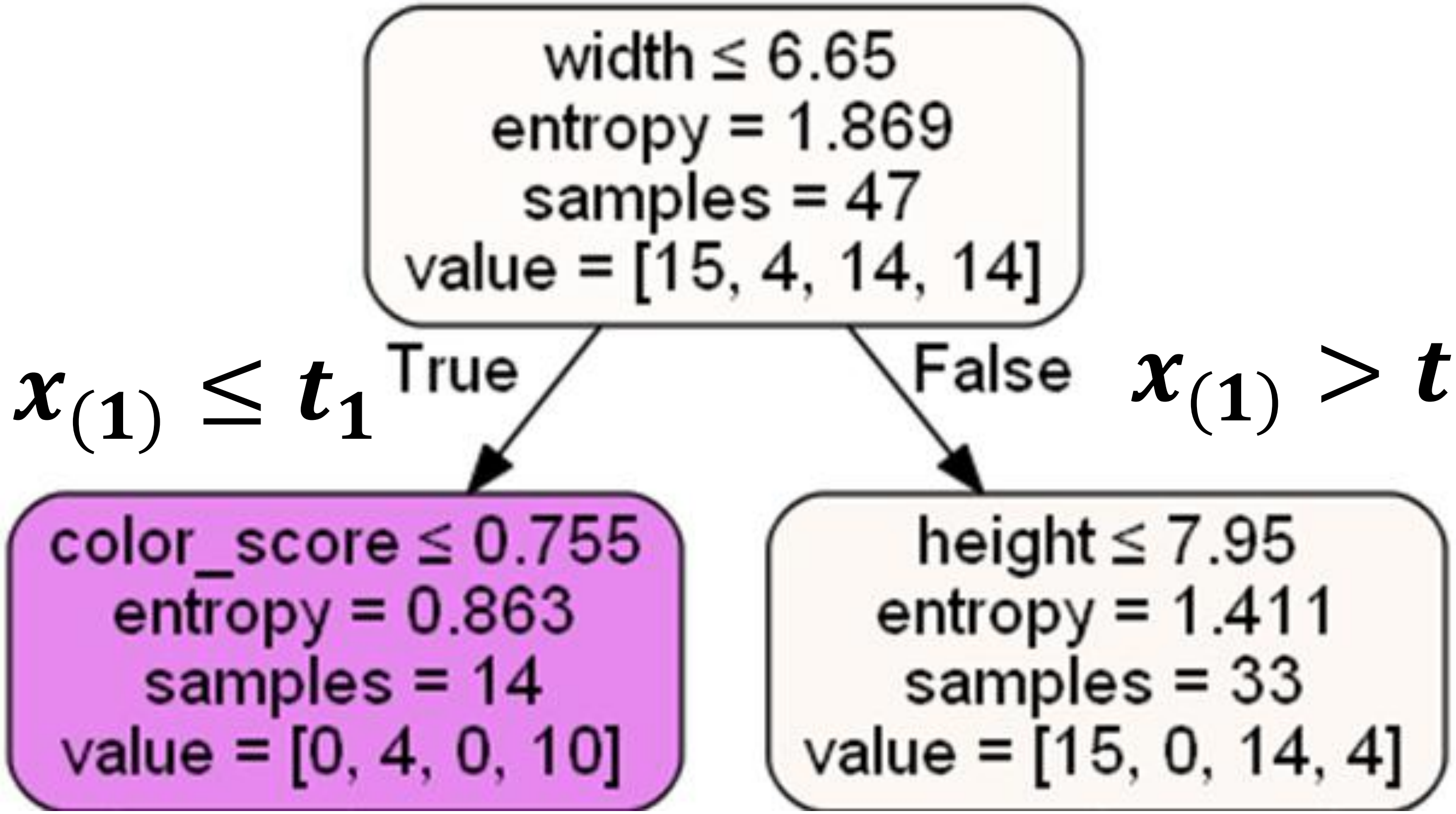
```
dtree.score(X_train, Y_train)
```

```
1.0
```

```
dtree.score(X_test, Y_test)
```

```
0.8333333333333334
```

$\boldsymbol{x}_{(\mathbf{1})}$ is width, $\boldsymbol{t}_{\mathbf{1}} = 6.65$



$\boldsymbol{x}_{(\mathbf{1})} \le \boldsymbol{t}_{\mathbf{1}}$

$\boldsymbol{x}_{(\mathbf{1})} > \boldsymbol{t}_{\mathbf{1}}$

$\boldsymbol{x}_{(\mathbf{3})}$ is color_score

$\boldsymbol{t}_{\mathbf{3}} = 0.755$

$\boldsymbol{x}_{(\mathbf{2})}$ is height

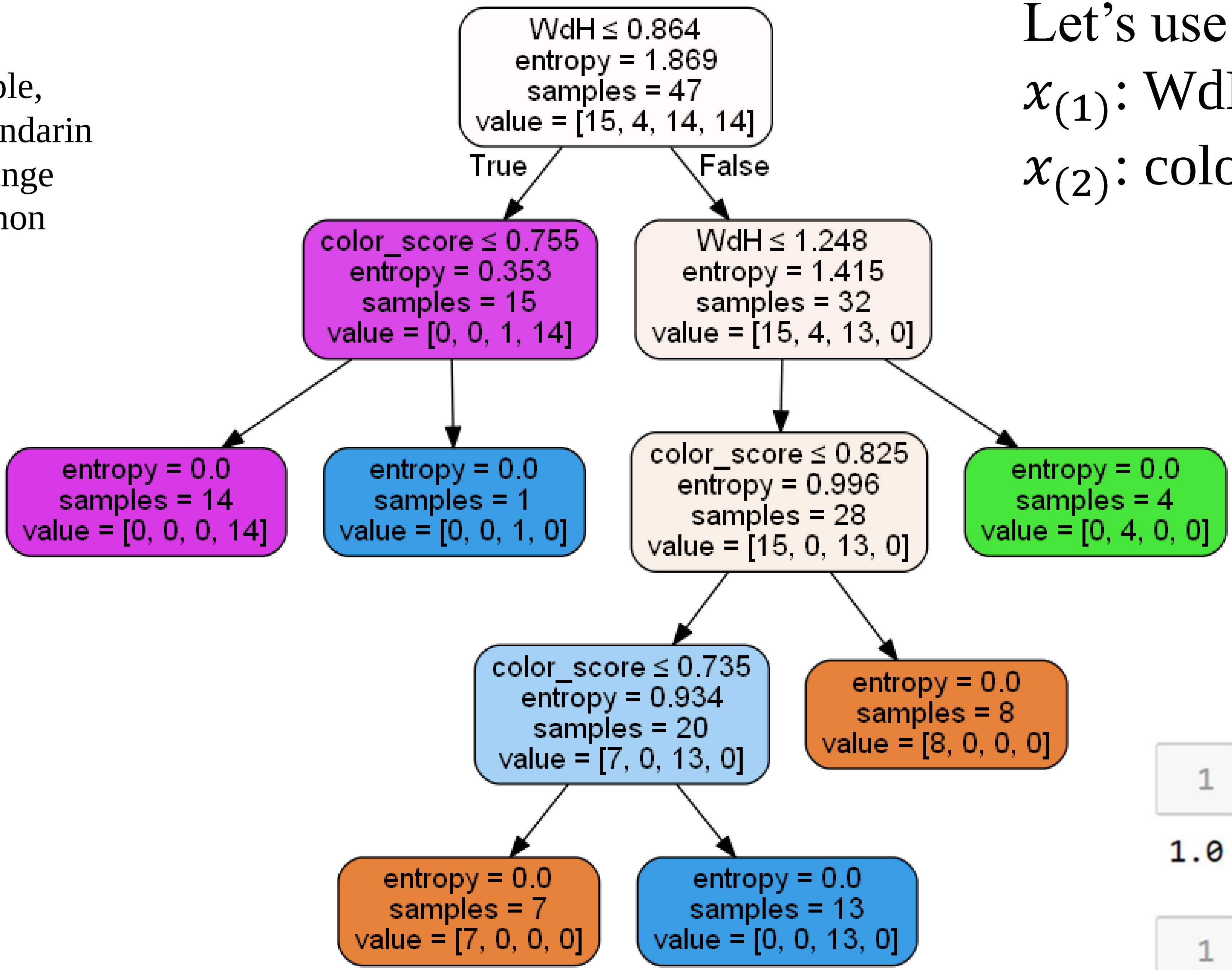
$\boldsymbol{t}_{\mathbf{2}} = 7.95$

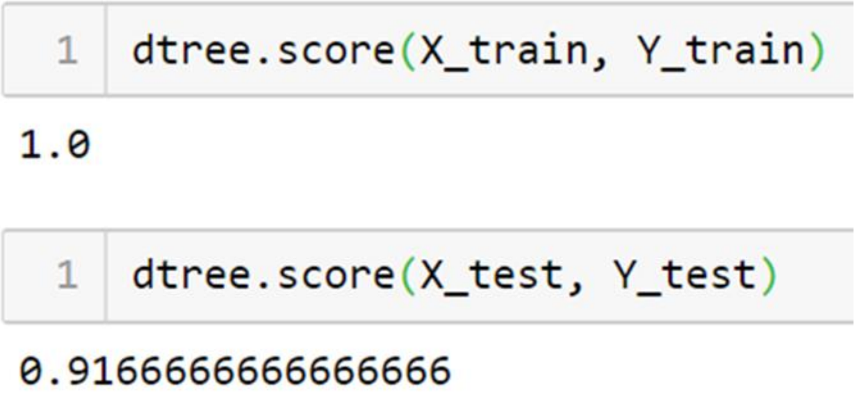
Let’s use two features per sample

$x_{(1)}$: WdH= width/height

$x_{(2)}$: color_score

1:apple,
2:mandarin
3:orange
4:lemon





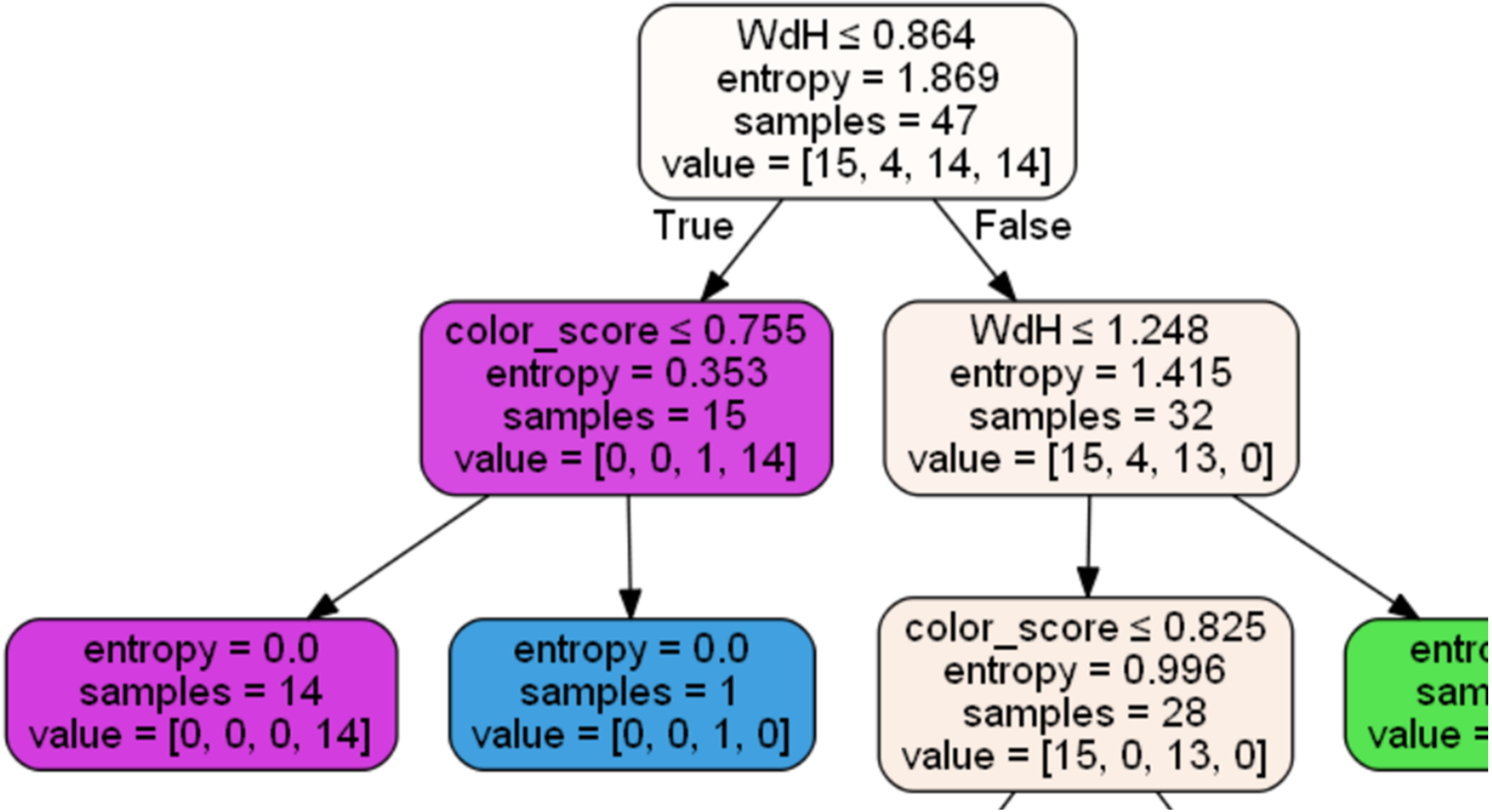
```
dtree.score(X_train, Y_train)
```

```
1.0
```

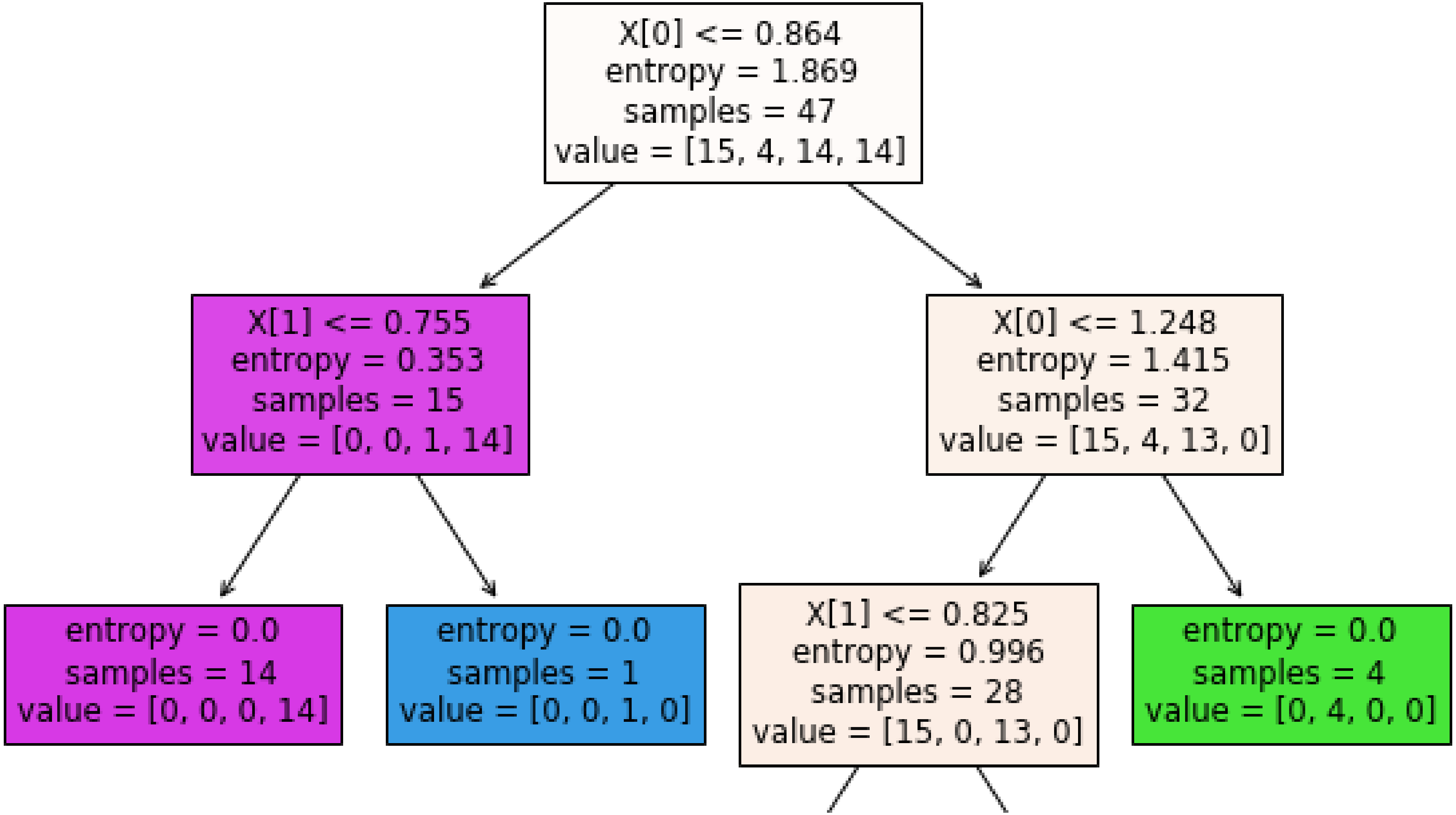
```
dtree.score(X_test, Y_test)
```

```
0.9166666666666666
```

visualize a decision tree: using graphviz



visualize a decision tree: using plot_tree in sk-learn



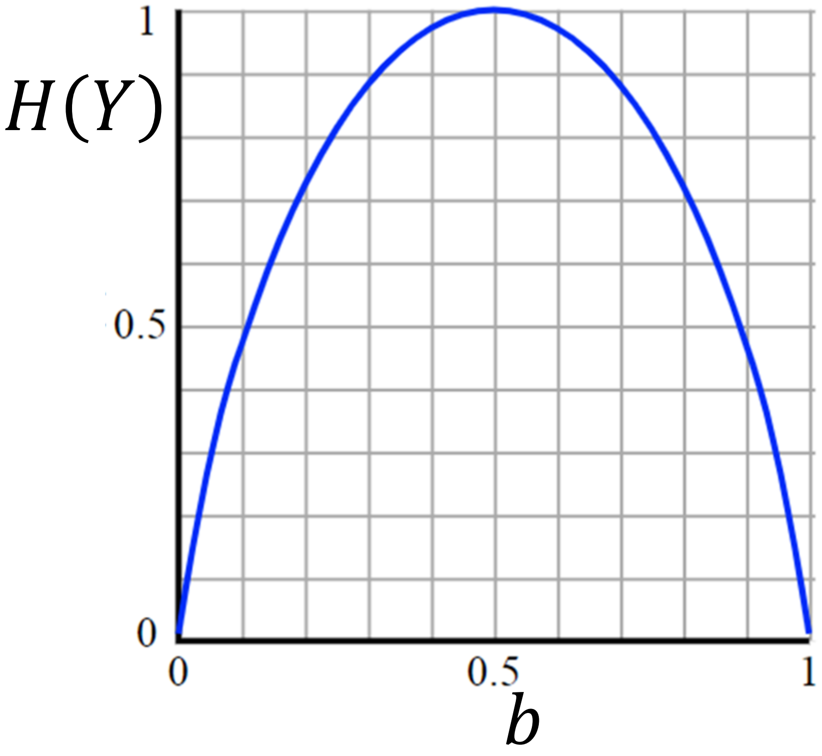
Entropy of a discrete random variable

- a discrete random variable Y

  the value of Y could be $a_1, a_2, \ldots., a_K$
- it has a probability mass function (PMF): $p(y = a_k) = p_k$
- the entropy is defined to be

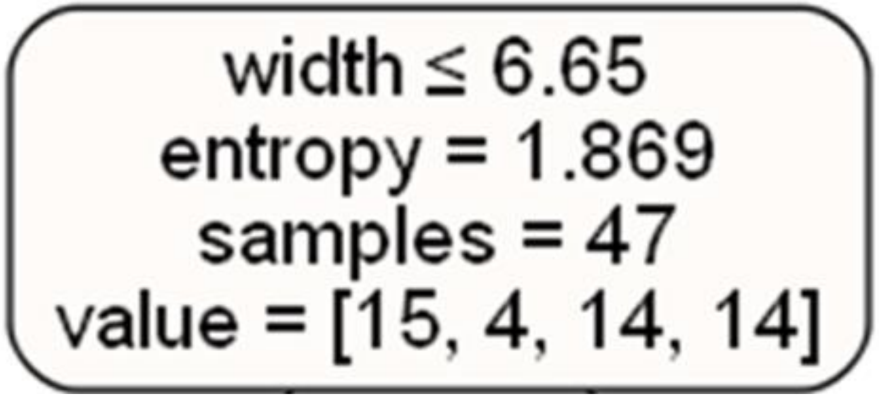
$$H(Y) = -\sum_{k=1}^{K} p_k log(p_k) \geq 0$$

$$H(Y) \equiv H(p)$$



- Example: $P(Y = 1) = b$ and $P(Y = 0) = 1 - b$

  $H(Y) = -blog(b) - (1 - b)log(1 - b)$



width ≤ 6.65

entropy = 1.869

samples = 47

value = [15, 4, 14, 14]

On this node:

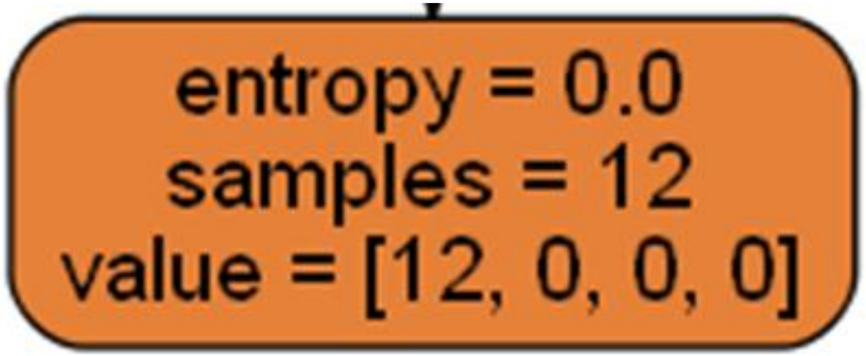
the distribution (PMF) over the 4 classes is

$$[p_1 \quad p_2 \quad p_3 \quad p_4]$$

$$p_1 = \frac{15}{47}, p_2 = \frac{4}{47}, p_3 = \frac{14}{47}, p_4 = \frac{14}{47}$$

PMF: probability mass function

Entropy $H(p) = -\sum_{k=1}^{4} p_k \, log_2(p_k) = 1.869$



entropy = 0.0
samples = 12
value = [12, 0, 0, 0]

On this node:
the distribution (PMF) over the 4 classes is

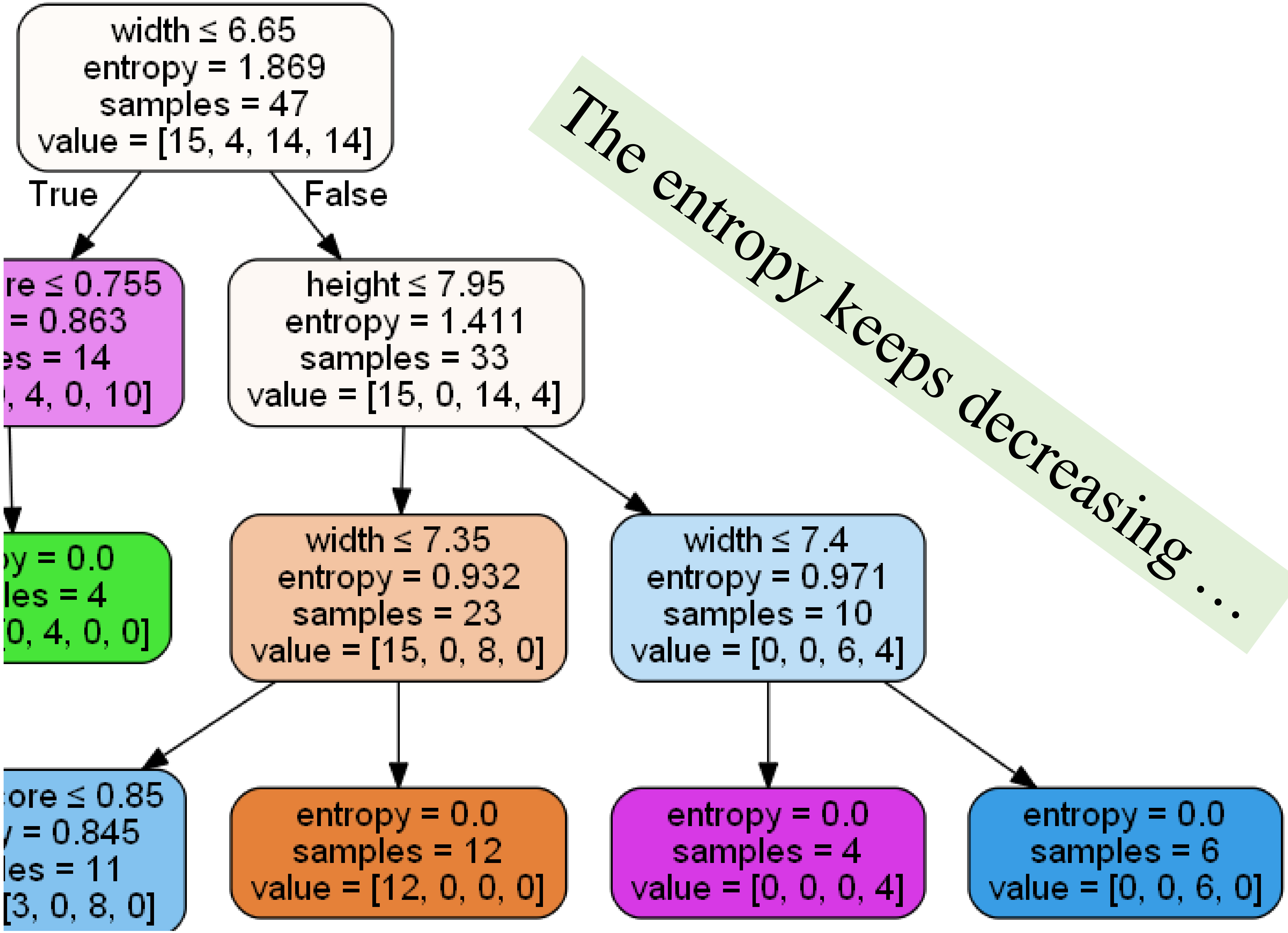
$$[p_1 \quad p_2 \quad p_3 \quad p_4]$$

$$p_1 = \frac{12}{12} = 1 \,,\, p_2 = 0 \,,\, p_3 = 0 \,,\, p_4 = 0$$

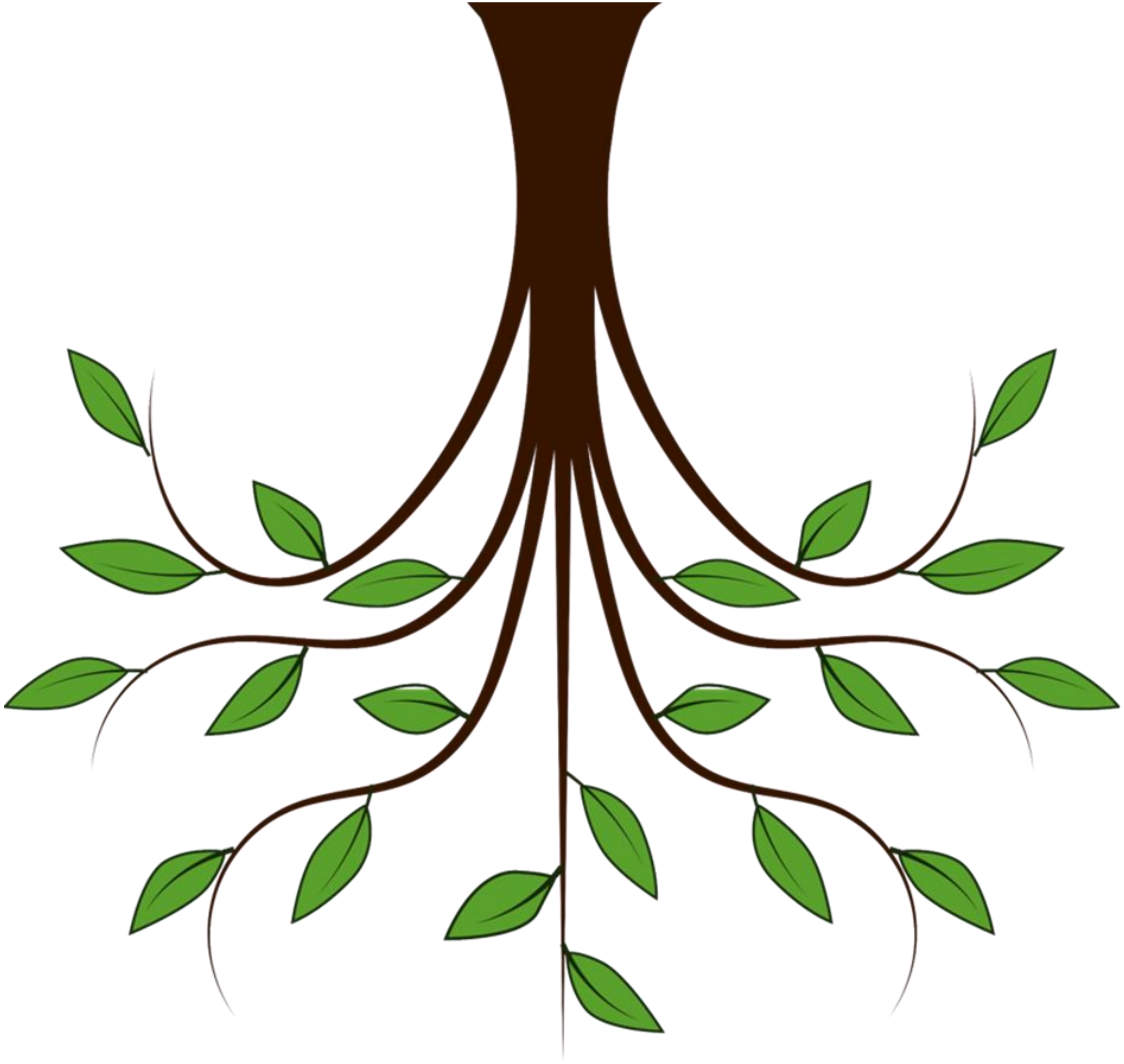
Entropy $H(p) = -\sum_{k=1}^{4} p_k \, log_2(p_k) = 0$

Define: $0 \, log_2(0) \equiv 0$

This node only contains apples ($p_1 = 1$).
It is a pure node (of apples)



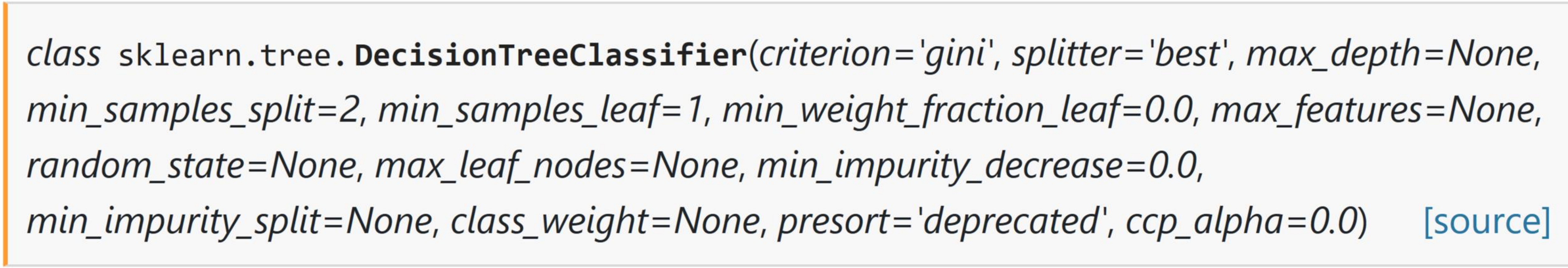
The entropy keeps decreasing …



Use a trained decision tree for classification

- Assume that a tree has been trained (fitted) on the training set.
- A new sample x needs to be classified
- The sample x will eventually fall into one of the leaf nodes.
- The majority class of the training samples in the leaf node is C.
- The predicted class label of the new sample x is C

DecisionTreeClassifier



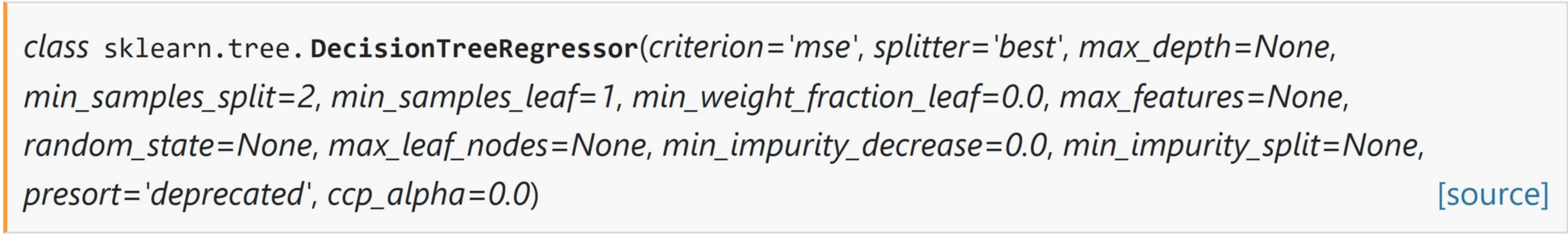
class sklearn.tree.**DecisionTreeClassifier**(criterion='gini', splitter='best', max_depth=None, min_samples_split=2, min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features=None, random_state=None, max_leaf_nodes=None, min_impurity_decrease=0.0, min_impurity_split=None, class_weight=None, presort='deprecated', ccp_alpha=0.0) [source]

Decision_Tree_friut_classification.ipynb

Use a trained decision tree for regression

- Assume that a tree has been trained (fitted) on the training set.
- A new sample x needs to be classified
- The sample x will eventually fall into one of the leaf nodes.
- The average target value of the training samples in the leaf node is C.
- The predicted target value (e.g. price) of the new sample x is C

DecisionTreeRegressor



```
class sklearn.tree.DecisionTreeRegressor(criterion='mse', splitter='best', max_depth=None,
min_samples_split=2, min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features=None,
random_state=None, max_leaf_nodes=None, min_impurity_decrease=0.0, min_impurity_split=None,
presort='deprecated', ccp_alpha=0.0)                                                   [source]
```