

DecisionTrees

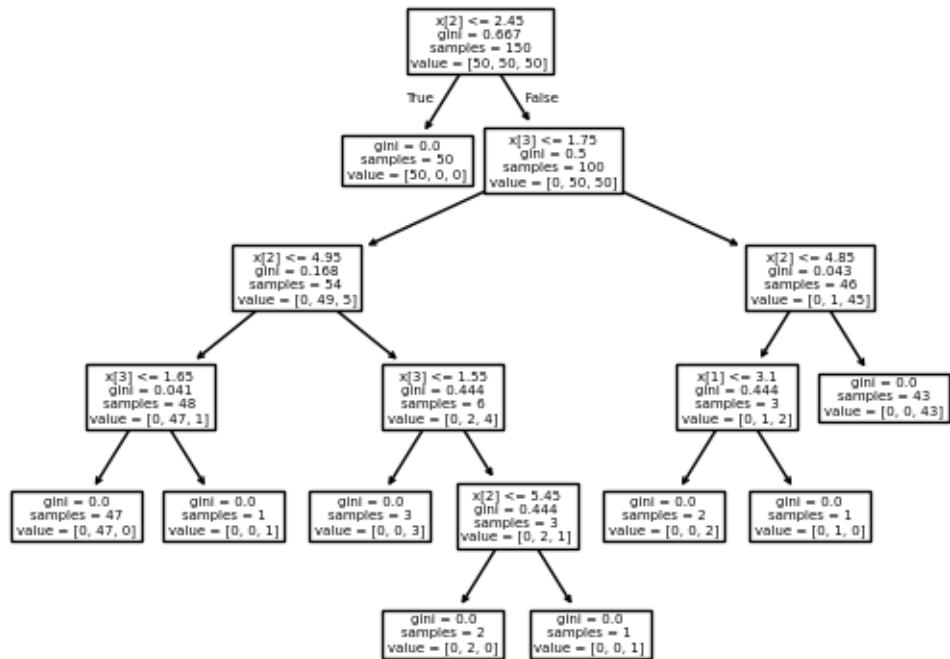
November 27, 2025

```
[1]: from sklearn.datasets import load_iris
from sklearn import tree
iris = load_iris()
X, y = iris.data, iris.target
clf = tree.DecisionTreeClassifier()
clf = clf.fit(X, y)
```

```
[2]: tree.plot_tree(clf)
```

```
[2]: [Text(0.5, 0.9166666666666666, 'x[2] <= 2.45\n gini = 0.667\n samples = 150\n value = [50, 50, 50]'),
Text(0.4230769230769231, 0.75, 'gini = 0.0\n samples = 50\n value = [50, 0, 0]'),
Text(0.46153846153846156, 0.8333333333333333, 'True '),
Text(0.5769230769230769, 0.75, 'x[3] <= 1.75\n gini = 0.5\n samples = 100\n value = [0, 50, 50]'),
Text(0.5384615384615384, 0.8333333333333333, ' False'),
Text(0.3076923076923077, 0.5833333333333334, 'x[2] <= 4.95\n gini = 0.168\n samples = 54\n value = [0, 49, 5]'),
Text(0.15384615384615385, 0.4166666666666667, 'x[3] <= 1.65\n gini = 0.041\n samples = 48\n value = [0, 47, 1]'),
Text(0.07692307692307693, 0.25, 'gini = 0.0\n samples = 47\n value = [0, 47, 0]'),
Text(0.23076923076923078, 0.25, 'gini = 0.0\n samples = 1\n value = [0, 0, 1]'),
Text(0.46153846153846156, 0.4166666666666667, 'x[3] <= 1.55\n gini = 0.444\n samples = 6\n value = [0, 2, 4]'),
Text(0.38461538461538464, 0.25, 'gini = 0.0\n samples = 3\n value = [0, 0, 3]'),
Text(0.5384615384615384, 0.25, 'x[2] <= 5.45\n gini = 0.444\n samples = 3\n value = [0, 2, 1]'),
Text(0.46153846153846156, 0.0833333333333333, 'gini = 0.0\n samples = 2\n value = [0, 2, 0]'),
Text(0.6153846153846154, 0.0833333333333333, 'gini = 0.0\n samples = 1\n value = [0, 0, 1]'),
Text(0.8461538461538461, 0.5833333333333334, 'x[2] <= 4.85\n gini = 0.043\n samples = 46\n value = [0, 1, 45]'),
Text(0.7692307692307693, 0.4166666666666667, 'x[1] <= 3.1\n gini = 0.444\n samples = 3\n value = [0, 1, 2]'),
Text(0.6923076923076923, 0.25, 'gini = 0.0\n samples = 2\n value = [0, 0, 2]'),
Text(0.8461538461538461, 0.25, 'gini = 0.0\n samples = 1\n value = [0, 1, 0]'),
```

```
Text(0.9230769230769231, 0.4166666666666667, 'gini = 0.0\nsamples = 43\nvalue = [0, 0, 43]')
```



```
[3]: iris.feature_names
```

```
[3]: ['sepal length (cm)',  
      'sepal width (cm)',  
      'petal length (cm)',  
      'petal width (cm)']
```

```
[4]: iris.target_names
```

```
[4]: array(['setosa', 'versicolor', 'virginica'], dtype='|<U10')
```

```
[ ]:
```