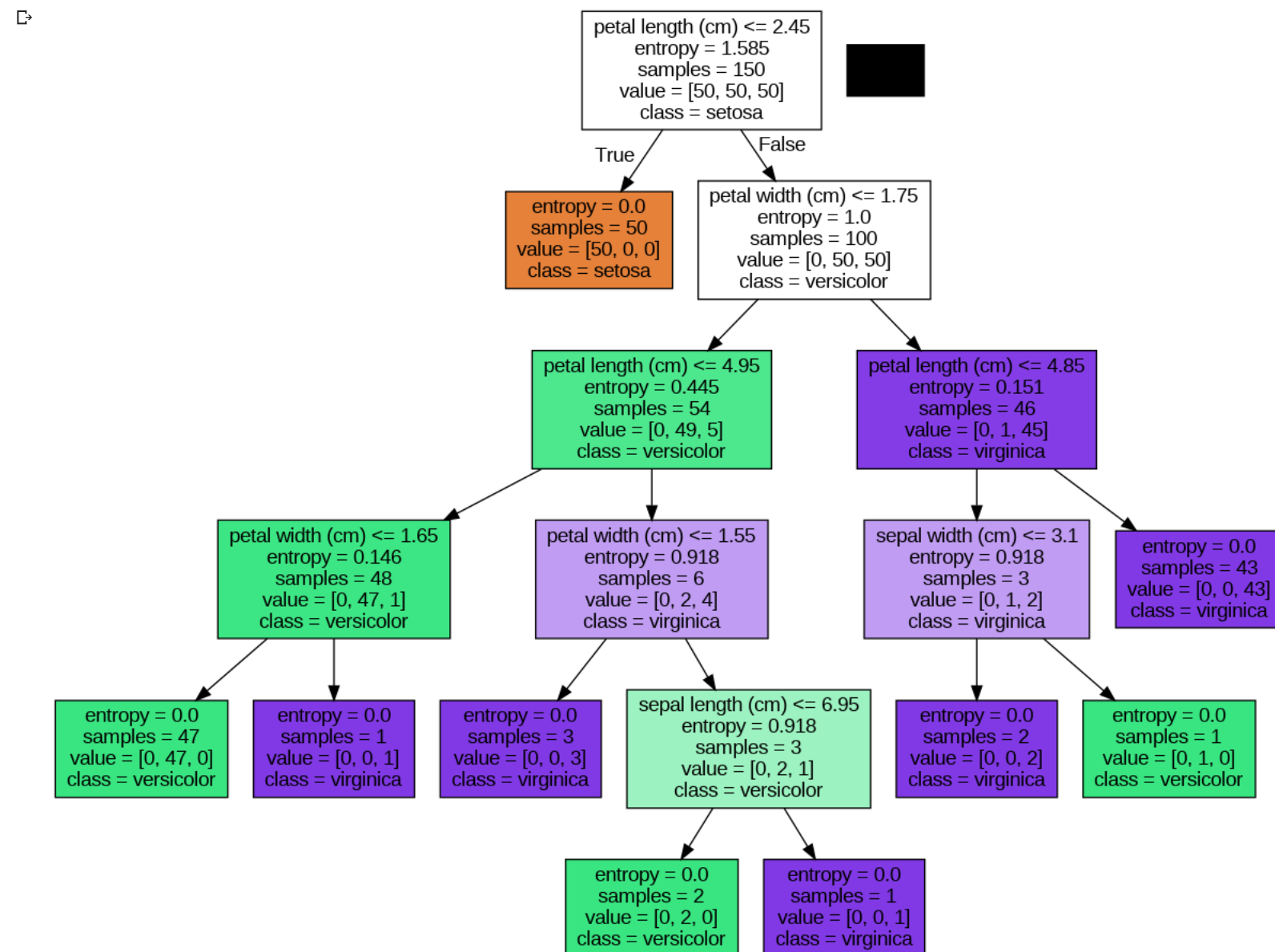


▼ ID3

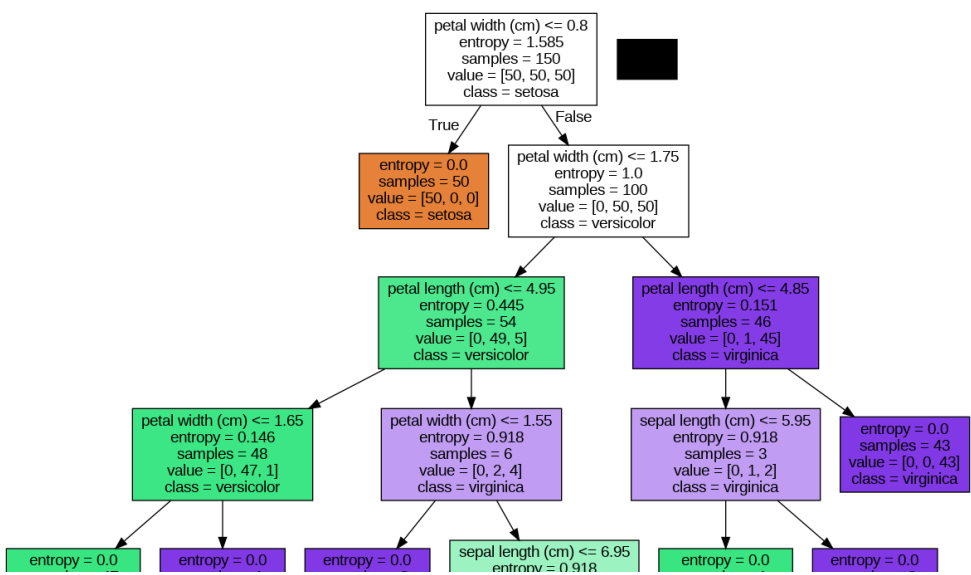
Decision Tree Algorithms: ID3

```
1 import numpy as np
2 from sklearn.datasets import load_iris
3 from sklearn.tree import DecisionTreeClassifier, export_graphviz
4 import pydotplus
5 from IPython.display import Image
6
7 # Loading the Iris dataset
8 iris = load_iris()
9 X = iris.data
10 y = iris.target
11 feature_names = iris.feature_names
12 class_names = iris.target_names
13
14 # Creating a decision tree classifier using the ID3 algorithm
15 dt = DecisionTreeClassifier(criterion='entropy')
16 dt.fit(X, y)
17
18 # Generating the decision tree visualization
19 dot_data = export_graphviz(dt, out_file=None, feature_names=feature_names, class_names=class_names, filled=True)
20 graph = pydotplus.graph_from_dot_data(dot_data)
21 Image(graph.create_png())
```



▼ C4.5

```
1 import numpy as np
2 from sklearn.datasets import load_iris
3 from sklearn.tree import DecisionTreeClassifier, export_graphviz
4 import pydotplus
5 from IPython.display import Image
6
7 # Loading the Iris dataset
8 iris = load_iris()
9 X = iris.data
10 y = iris.target
11 feature_names = iris.feature_names
12 class_names = iris.target_names
13
14 # Creating a decision tree classifier using the C4.5 algorithm
15 dt = DecisionTreeClassifier(criterion='entropy')
16 dt.fit(X, y)
17
18 # Generating the decision tree visualization
19 dot_data = export_graphviz(dt, out_file=None, feature_names=feature_names, class_names=class_names, filled=True)
20 graph = pydotplus.graph_from_dot_data(dot_data)
21 Image(graph.create_png())
22
```



CART ALGORITHM

```
1 import numpy as np
2 from sklearn.datasets import load_iris
3 from sklearn.tree import DecisionTreeClassifier, export_graphviz
4 import pydotplus
5 from IPython.display import Image
6
7 # Load the Iris dataset
8 iris = load_iris()
9 X = iris.data
10 y = iris.target
11 feature_names = iris.feature_names
12 class_names = iris.target_names
13
14 # Create a decision tree classifier using the CART algorithm
15 dt = DecisionTreeClassifier(criterion='gini')
16 dt.fit(X, y)
17
18 # Generate the decision tree visualization
19 dot_data = export_graphviz(dt, out_file=None, feature_names=feature_names, class_names=class_names, filled=True)
20 graph = pydotplus.graph_from_dot_data(dot_data)
21 Image(graph.create_png())
22
```

