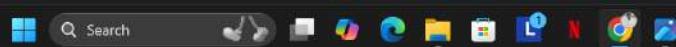


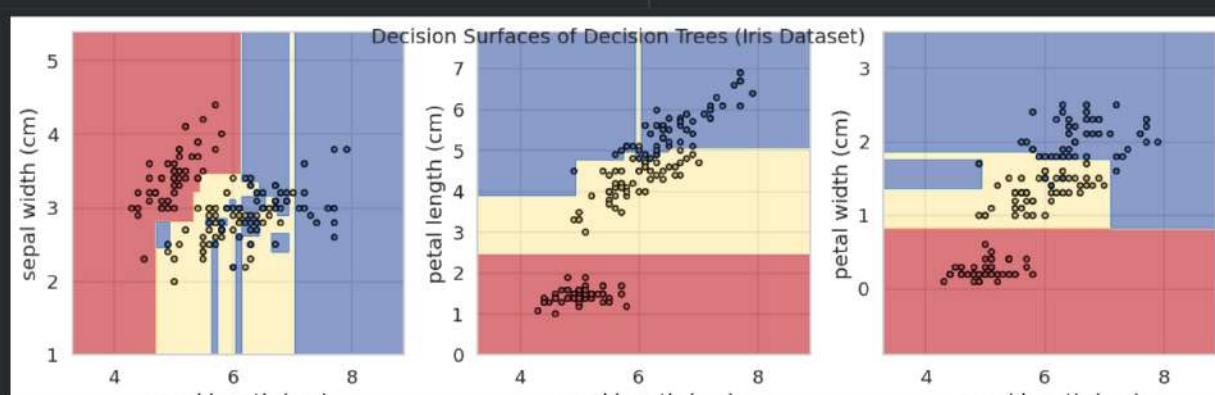
```
from sklearn.datasets import load_iris
from sklearn.tree import DecisionTreeClassifier, plot_tree
import numpy as np
import matplotlib.pyplot as plt
iris = load_iris()
X = iris.data
y = iris.target
n_classes = 3
plot_colors = "ryb"
plot_step = 0.02
plt.figure(figsize=(12, 8))
for pairidx, pair in enumerate([[0, 1], [0, 2], [0, 3],
                               [1, 2], [1, 3], [2, 3]]):
    X_pair = X[:, pair]
    clf = DecisionTreeClassifier().fit(X_pair, y)
    x_min, x_max = X_pair[:, 0].min() - 1, X_pair[:, 0].max() + 1
    y_min, y_max = X_pair[:, 1].min() - 1, X_pair[:, 1].max() + 1
    xx, yy = np.meshgrid(np.arange(x_min, x_max, plot_step),
                          np.arange(y_min, y_max, plot_step))
    z = clf.predict(np.c_[xx.ravel(), yy.ravel()])
    z = z.reshape(xx.shape)
    plt.subplot(2, 3, pairidx + 1)
    plt.contourf(xx, yy, z, cmap=plt.cm.RdYlBu, alpha=0.6)
    for i, color in zip(range(n_classes), plot_colors):
        idx = np.where(y == i)
        plt.scatter(X_pair[idx, 0], X_pair[idx, 1], c=color,
                    label=iris.target_names[i], edgecolor='black', s=15)
    plt.xlabel(iris.feature_names[pair[0]])
    plt.ylabel(iris.feature_names[pair[1]])
    plt.tight_layout()
plt.suptitle("Decision Surfaces of Decision Trees (Iris Dataset)", fontsize=14)
plt.legend(loc="lower right", borderpad=0.2, handletextpad=0.2)
```

Variables Terminal



15:46

18-11-2025



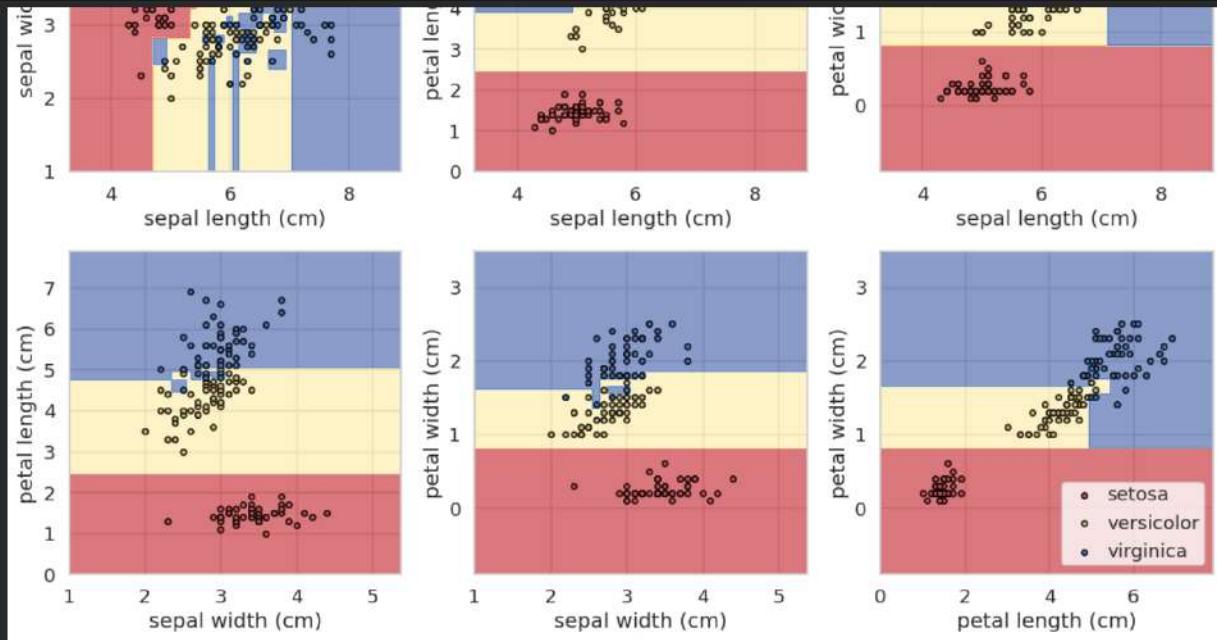
Variables

Terminal



15:46

18-11-2025



Variables Terminal



ENG IN 18-11-2025 15:47

