

# Démonstration KNN 2 paramètres en utilisant la base IRIS

## Importation des Bibliothèques:

```
In [1]: from sklearn import datasets
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn import tree
import matplotlib.pyplot as plt
```

## Step 1: Load the Iris dataset

```
In [2]: iris = datasets.load_iris()
```

## Step 2: Prepare the data

```
In [3]: X = iris.data # Features (sepal length, sepal width, petal length, petal width)
y = iris.target # Target labels (species)
```

## Step 3: Split the dataset into training and test sets

```
In [4]: X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_
```

## Step 4: Train a Decision Tree Classifier

```
In [5]: clf = DecisionTreeClassifier(random_state=42)
clf.fit(X_train, y_train)
```

```
Out[5]: DecisionTreeClassifier
DecisionTreeClassifier(random_state=42)
```

## Step 5: Predict on the test data

```
In [6]: y_pred = clf.predict(X_test)
```

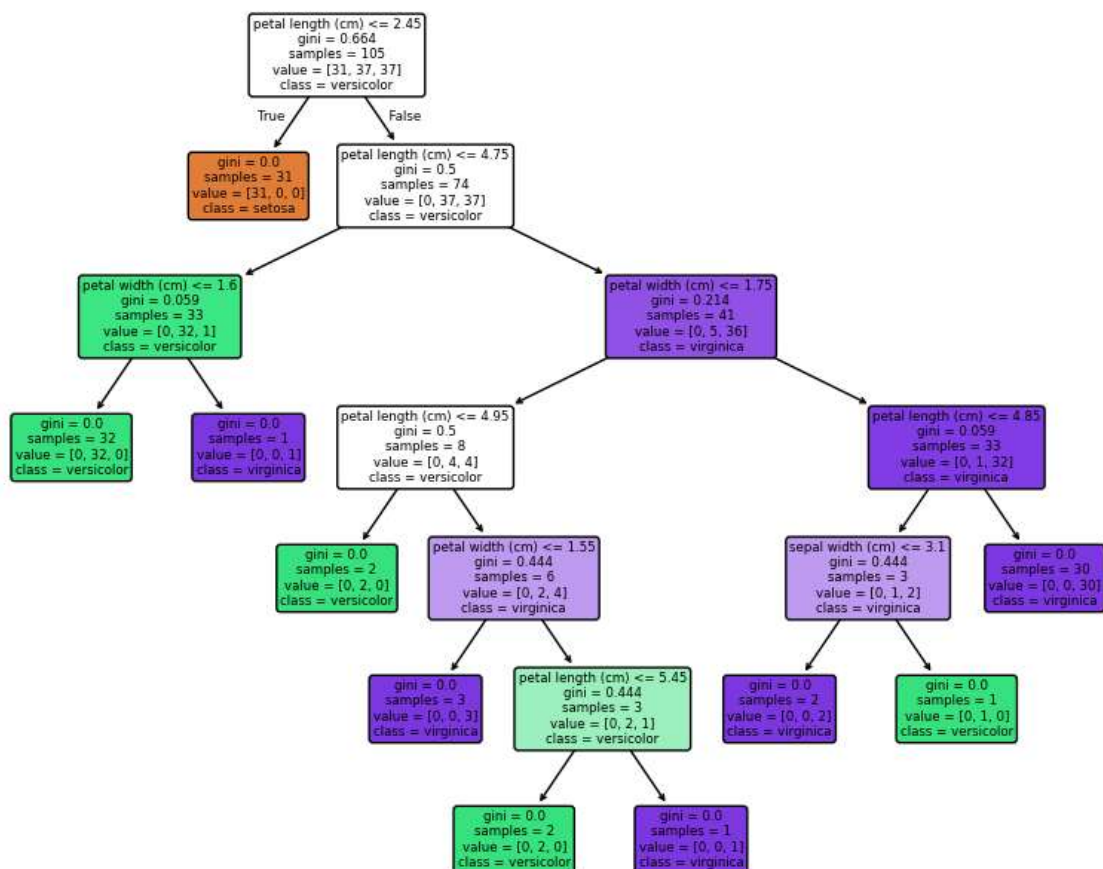
## Step 6: Evaluate the model's performance

```
In [7]: accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy of the Decision Tree Classifier: {accuracy * 100:.2f}%")
```

Accuracy of the Decision Tree Classifier: 100.00%

## Step 7: Visualize the Decision Tree

```
In [8]: plt.figure(figsize=(10, 8))
tree.plot_tree(clf, filled=True, feature_names=iris.feature_names, class_names=iris.target_names)
plt.show()
```



## Step 8: Classification example - Classify a new sample

```
In [9]: # Define a new sample (e.g., new iris flower with specific features)
new_sample = [[5.5, 2.4, 3.8, 1.1]] # Example: sepal length = 5.5, sepal width
```

Predict the class for the new sample

```
In [11]: predicted_class = clf.predict(new_sample)
```

## Print the predicted class and corresponding species name

```
In [12]: print(f"The predicted class for the new sample is: {iris.target_names[predicted_
```

The predicted class for the new sample is: versicolor