decision tree regression

# example code

|  |
| --- |
| import matplotlib.pyplot as plt  from sklearn.datasets import load\_iris  from sklearn.datasets import load\_breast\_cancer  from sklearn.tree import DecisionTreeClassifier  from sklearn.ensemble import RandomForestClassifier  from sklearn.model\_selection import train\_test\_split  import pandas as pd  import numpy as np  from sklearn import tree  data = load\_iris()  df = pd.DataFrame(data.data, columns=data.feature\_names)  df['target'] = data.target  X\_train, X\_test, Y\_train, Y\_test = train\_test\_split(df[data.feature\_names], df['target'], random\_state=0)  # Step 1: Import the model you want to use  # This was already imported earlier in the notebook so commenting out  #from sklearn.tree import DecisionTreeClassifier  # Step 2: Make an instance of the Model  clf = DecisionTreeClassifier(max\_depth = 2,                               random\_state = 0)  # Step 3: Train the model on the data  clf.fit(X\_train, Y\_train)  # Step 4: Predict labels of unseen (test) data  # Not doing this step in the tutorial  # clf.predict(X\_test)  tree.plot\_tree(clf)  fn=['sepal length (cm)','sepal width (cm)','petal length (cm)','petal width (cm)']  cn=['setosa', 'versicolor', 'virginica']  fig, axes = plt.subplots(nrows = 1,ncols = 1,figsize = (4,4), dpi=300)  tree.plot\_tree(clf,                 feature\_names = fn,                 class\_names=cn,                 filled = True)  fig.savefig('imagename.png') |

# testing result

|  |
| --- |
|  |