```
In [2]: import pandas as pd
         data = pd.read_csv("Boston_housing_modified.csv", header=0)
         data.head()
Out[7]:
                                                                                                                             property
                                                                                                                                         teacl
                                  crime pollution (nitrous oxide) avr rooms per house %houses built <1940 distance to employment offices
            area_number Longitude
                                                                                                                                 tax student_ra
         0
                     O
                          -70.955 0.00632
                                                        0.538
                                                                          6.575
                                                                                             65.2
                                                                                                                      4.0900
                                                                                                                                 296
         1
                          -70.950 0.02731
                                                        0.469
                                                                          6.421
                                                                                             78.9
                                                                                                                      4.9671
                                                                                                                                 242
                     1
         2
                          -70.936 0.02729
                                                        0.469
                                                                          7.185
                                                                                             61.1
                                                                                                                       4.9671
                                                                                                                                 242
         3
                     2
                          -70.928 0.03237
                                                        0.458
                                                                          6.998
                                                                                             45.8
                                                                                                                      6.0622
                                                                                                                                 222
                                                        0.458
                                                                                                                      6.0622
                     2
                          -70.922 0.06905
                                                                          7.147
                                                                                             54.2
                                                                                                                                 222
In [8]: import pandas as pd
         import sklearn
         from sklearn.tree import DecisionTreeRegressor
         from sklearn.tree import export_graphviz
         from six import StringIO
         from IPython.display import Image
         import pydotplus
         data = pd.read_csv("Boston_housing_modified.csv", header=0)
         predictors = data.drop(['median_price'],axis=1) # Features
         target = data['median_price'] # Target variable
         from sklearn.model_selection import train_test_split
         predictors_teach, predictors_test, target_teach, target_test = sklearn.model_selection.train_test_split(predictors,
         target, test_size=0.3, random_state=1) # 70% training and 30% test
         decision_tree = DecisionTreeRegressor(min_impurity_decrease=0.02, max_depth=4 , min_samples_leaf=20)
         decision_tree = decision_tree.fit(predictors_teach, target_teach)
         dot_data = StringIO()
         export_graphviz(decision_tree, out_file=dot_data,
                           filled=True, rounded=True, impurity=False, proportion=True, precision=2,
                           special_characters=True, feature_names = predictors.columns,class_names=['survived','died'])
         graph = pydotplus.graph_from_dot_data(dot_data.getvalue())
         Image(graph.create_png())
Out[8]:
                                                               poor_residents \leq 9.55
                                                                samples = 100.0\%
                                                                  value = 22.34
                                                             True
                                                                               False
                                              avr_{rooms_per_house} \le 7.41
                                                                            poor_residents ≤ 19.26
                                                   samples = 38.4\%
                                                                               samples = 61.6\%
                                                    value = 29.99
                                                                                value = 17.57
                            poor_residents ≤ 5.14`
                                                                              Longitude ≤ -71.07
                                                                                                   pollution_(nitrous_oxide) \le 0.67
                                                    samples = 5.6\%
                              samples = 32.8\%
                                                                              samples = 44.4\%
                                                                                                         samples = 17.2\%
                                                    value = 43.74
                               value = 27.62
                                                                                                          value = 12.75
                                                                                value = 19.44
                             Longitude ≤ -71.02
                                                               property tax ≤ 318.5
                                                                                        crime ≤ 0.56
                                                                                                          samples = 7.3\%
                                                                                                                            samples = 9.9\%
          samples = 8.8\%
                              samples = 24.0\%
                                                                 samples = 14.7\%
                                                                                      samples = 29.7\%
           value = 33.16
                                                                                                          value = 15.81
                                                                                                                             value = 10.47
                                value = 25.6
                                                                  value = 21.92
                                                                                       value = 18.21
                              samples = 8.2\%
                                                                   samples = 7.1\%
                                                                                      samples = 14.4\%
                                                                                                          samples = 15.3\%
           samples = 15.8\%
                                                 samples = 7.6\%
             value = 27.3
                                value = 22.3
                                                  value = 23.69
                                                                     value = 20.0
                                                                                       value = 19.76
                                                                                                           value = 16.75
In [9]: feature_importances = pd.DataFrame({'predictor': predictors.columns,
                              'importance': decision_tree.feature_importances_}).\
                               sort_values('importance', ascending = False)
         feature_importances.head()
Out[9]:
                        predictor importance
         10
                    poor_residents
                                  0.722157
          4
              avr_rooms_per_house
                                  0.197858
                                  0.042606
                       Longitude
          3 pollution_(nitrous_oxide)
                                  0.018924
                           crime
                                  0.010577
         prediction = decision_tree.predict(predictors_test)
         from sklearn.metrics import r2_score
         accuracy = r2_score(target_test, prediction)
         print("r2_score:", accuracy)
         r2_score: 0.7866802906225296
In [ ]:
```