MSDS 422 Practical Machine Learning
John Moderwell
Assignment 4

Summary

The Boston Housing Study was a study of 506 census tracts in the Boston metropolitan area. The purpose of the study was to analyze the effect of air pollution on housing prices. However, the data can also be used to calculate valuations for the residential real estate market. MSDS 422 students evaluated several machine learning regression models as they were applied to the dataset. The models included in this study were Elastic Net, Lasso, Linear and Ridge. In the Module 4 assignment, students were expected evaluate various tree methods. The models included in this analysis were the RandomTreeRegressor, GradientBoostingRegressor and DecisionTreeRegressor.

Research Design

The dataset consisted of 506 observations and 14 variables. These variables represented factors that would influence the valuation of a residential house as well as descriptive characteristics. They include: name of the Boston neighborhood (neighborhood), median value of home in 1970 dollars (mv), air pollution (nox), crime rate (crim), percent of land zone for lots (zn), percent of business that is industrial (indus), proximity to Charles River (chas), average number of rooms per home (rooms), percentage of homes built before 1940 (age), distance to employment centers (dis), accessibility to highways (rad), tax rate (tax), student/teacher ratio (ptratio) and percentage of population of lower socio-economic status (lstat). The jumpstart code provided to students dropped the neighborhood variable, standardized scores across all variables and arranged the data into a numpy array. As part of the exploratory data analysis, histogram and density plots were created. Descriptive statistics were calculated as well. To apply machine learning methods to the data, median value of the house (mv) was used as the response variable and the other 12

variables were used as the explanatory variables. A 10-fold cross validation was employed and then the average of the folds was used to calculate RMSE. The root mean squared error (RMSE) is an evaluation metric that calculates prediction error and was used to compare the regression models. This same process was repeated for this week's assignment, this time used to evaluate decision tree and random forest models.

Programming Overview

For this assignment, the programming language Python was used and executed in the Jupyter Notebook environment. Several packages were utilized throughout the project including: Pandas and Numpy for data handling/manipulation. This included the loading of the 'Boston.csv' file and conversion to a Numpy array. Descriptive statistics were calculated using Numpy built-in functions. The package Seaborn was used to visualize data including histogram/density plots as well as a correlation matrix. The machine learning portion of the assignment was done entirely using the Sci-Kit Learn package. This included the regression methods (LinearRegresion, Ridge, Lasso, ElasticNet), metrics to evaluate the models (mean squared error, R2 score) and KFold for cross validation. The sqrt function was imported from the math package to calculate root mean-squared error. Additional Sci-kit Learn packages were used in the Module 4 assignment. These packages included the tree method 'DecisionTreeRegressor' and ensemble methods 'RandomForestRegressor' and 'GradientBoostingRegressor'.

Recommendations

In comparison to the regression models used in Week 3, on average tree methods performed better. Of the three methods employed, it appears that the Gradient Boosting Regressor (max depth = 2 and number of estimators = 100) is the most suitable method for this project. This

model used a max depth of 2 to avoid overfitting. Models were tested on the original data with no transformations. Each method was evaluated using RMSE (root mean-squared error) as an index for prediction error. Using 10-fold cross-validation, Gradient Boosting achieved an average RMSE of .42166, superior to the RandomForestRegressor (.47454) and DecisionTreeRegressor (.60855). Since the Gradient Boosting model performed better than the regression methods analyzed in Week 3, it is recommended that management uses the Gradient Boosting method with small depth to avoid overfitting.