Predicting Boston Housing Prices

Model Evaluation & Validation Project

Project Description

You want to be the best real estate agent out there. In order to compete with other agents in your area, you decide to use machine learning. You are going to use various statistical analysis tools to build the best model to predict the value of a given house. Your task is to find the best price your client can sell their house at. The best guess from a model is one that best generalizes the data.

For this assignment your client has a house with the following feature set: [11.95, 0.00, 18.100, 0, 0.6590, 5.6090, 90.00, 1.385, 24, 680.0, 20.20, 332.09, 12.13]. To get started, use the example scikit implementation. You will have to modify the code slightly to get the file up and running.

When you are done implementing the code please answer the following questions in a report with the appropriate sections provided.

Language and libraries

For this project, you will need to have the following software installed:

- Python 2.7
- NumPy
- scikit-learn

Template code

Download the `boston_housing.py` template file from the Downloadables section below. Follow the instructions to complete each step, and answer the questions in your report.

Deliverables

- Report in PDF format
- Fully implemented Boston Housing Python code as `boston_housing.py`

You can package these two files as a single zip and submit it.

Questions and Report Structure

1) Statistical Analysis and Data Exploration

- Number of data points (houses)?
- Number of features?
- Minimum and maximum housing prices?
- Mean and median Boston housing prices?
- Standard deviation?

2) Evaluating Model Performance

- Which measure of model performance is best to use for predicting Boston housing data and analyzing the errors? Why do you think this measurement most appropriate? Why might the other measurements not be appropriate here?
- Why is it important to split the Boston housing data into training and testing data? What happens if you do not do this?
- What does grid search do and why might you want to use it?
- Why is cross validation useful and why might we use it with grid search?

3) Analyzing Model Performance

- Look at all learning curve graphs provided. What is the general trend of training and testing error as training size increases?
- Look at the learning curves for the decision tree regressor with max depth 1 and 10 (first and last learning curve graphs). When the model is fully trained does it suffer from either high bias/underfitting or high variance/overfitting?
- Look at the model complexity graph. How do the training and test error relate to increasing model complexity? Based on this relationship, which model (max depth) best generalizes the dataset and why?

4) Model Prediction

- Model makes predicted housing price with detailed model parameters (max depth) reported using grid search. Note due to the small randomization of the code it is recommended to run the program several times to identify the most common/reasonable price/model complexity.
- Compare prediction to earlier statistics and make a case if you think it is a valid model.

Evaluation

Your project will be reviewed by a Udacity reviewer against **this rubric**. Be sure to review it thoroughly before you submit. All criteria must "meet specifications" in order to pass.

Submission

When you're ready to submit your project go back to your Udacity Home, click on Project 1, and we'll walk you through the rest of the submission process.

If you are having any problems submitting your project or wish to check on the status of your submission, please email us at **machine-support@udacity.com** or visit us in the discussion forums.

What's Next?

You will get an email as soon as your reviewer has feedback for you. In the meantime, review your next project and feel free to get started on it or the courses supporting it!