UDACITY

PROJECT

Predicting Boston Housing Prices

A part of the Machine Learning Engineer Nanodegree Program

PROJECT REVIEW

CODE REVIEW

NOTES

SHARE YOUR ACCOMPLISHMENT!

Meets Specifications

Congratulations for meeting all specifications! You made lots of improvements on what was already a strong report. Be sure to check the comments below, and keep up the good work!

Data Exploration

All requested statistics for the Boston Housing dataset are accurately calculated. Student correctly leverages NumPy functionality to obtain these results.

I'm following the previous reviewer and considering you met specifications here, but note you should use numpy to calculate all the requested statistics, including the minimum and maximum. You could have used `np.amax(prices)` or `prices.max()` to calculate the maximum value, for instance.

Student correctly justifies how each feature correlates with an increase or decrease in the target variable.

Developing a Model

Student correctly identifies whether the hypothetical model successfully captures the variation of the target variable based on the model's R^2 score. The performance metric is correctly implemented in code.

> This model is considered successfully captured the variation of the target variable because the R^2 score is high which indicates the 92.3% of of the response variable variation that is explained by the model.

Yes, a high R^2 indicates the model captures most of the variation of the target variable. In this toy example you can see for yourself that this is the case by comparing actual and predicted values for all data points.

Student provides a valid reason for why a dataset is split into training and testing subsets for a model. Training and testing split is correctly implemented in code.

Your answer is quite short here, but I'm deferring to the previous reviewer and considering you met specifications. I also encourage you to check out the previous reviewer's comments and links pertaining to this item if you haven't done so - this bit is particularly illuminating:

> Test sets in Machine Learning are created in order to test the trained model's ability to generalise beyond to predict other points which it didn't encounter in the training set.

Analyzing Model Performance

Student correctly identifies the trend of both the training and testing curves from the graph as more training points are added. Discussion is made as to whether additional training points would benefit the model.

Nice revision! You are right, adding more data points would not benefit the model, since testing and training scores already converged - indicating that to increase the model's testing score we would need to reduce bias (i.e., come up with a more flexible model), not variance. It may also be that this model is "as good as it gets", and that the error we see is irreducible (see section 1.3 here).

Student correctly identifies whether the model at a max depth of 1 and a max depth of 10 suffer from either high bias or high variance, with justification using the complexity curves graph.

Student picks a best-guess optimal model with reasonable justification using the model complexity graph.

Evaluating Model Performance

Student correctly describes the grid search technique and how it can be applied to a learning algorithm.

Student correctly describes the k-fold cross-validation technique and discusses the benefits of its application when used with grid search when optimizing a model.

> Cross-validation training could prevent the overfiting for grid search when optimizing a model.

This for me is the key point here. If you try out several models, with different sets of parameters, and check each model's results against the *same* testing set, then you are letting information about your testing set blend into your model. K-fold cross-validation, by creating several testing sets and taking the average value for all of them, helps you avoid this problem (although it's still a good idea to have a final, *really* held-out testing set to check your model once parameter tuning is finished!). K-fold also lets you use a all your data for both training and testing, which can be nice especially when you don't have a lot of data to work with.

Student correctly implements the `fit_model` function in code.

Student reports the optimal model and compares this model to the one they chose earlier.

Student reports the predicted selling price for the three clients listed in the provided table. Discussion is made as to whether these prices are reasonable given the data and the earlier calculated descriptive statistics.

Really great answer, congrats!

Student thoroughly discusses whether the model should or should not be used in a real-world setting.

DOWNLOAD PROJECT

Have a question about your review? Email us at review-support@udacity.com and include the link to this review.

RETURN TO PATH

Rate this review

Student FAQ