UDACITY

Back to Machine Learning Engineer Nanodegree

Predicting Boston Housing Prices

REVIEW

HISTORY

Meets Specifications

Excellent job answering all the questions correctly on your first try! Understanding these fundamental concepts is really important before you embark on your learning journey throughout the next lessons.

Data Exploration

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

Your justifications were thorough, awesome work. Good to see you have included visualizations as well.

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

Good work using numpy functions to calculate the requested statistics. Several numpy functions have a slight difference from Pandas functions; for example, `prices.std()` would use the entire population (`n` instead of `n-1`) by default, while `numpy.std()` uses sample population (`n-1`).

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.

Correct, the model does fit well given this dataset. Please be reminded however that the number of training data may not be statistically significant depending on how many actual population we do have here (i.e. if all these 5 observations were the entire population then the model is indeed good).

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.

Analyzing Model Performance

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

Good choice, either `max_depth` of 3 or 4 is good here. `max_depth` of 4 gives slightly higher validation score which is arguably the most important metric, but less complexity in `max_depth` 3 would have been a good reason to choose it as well.

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.

The answer has demonstrated a good understanding of both high bias and high variance problems. Well done.

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.

Correct. The training score decreases and testing score increases, and both scores converged at around 0.8 after trained with 300 training data. There was no significant increase in score with more data points, so we may safely assume adding data points further will not benefit the model.

Evaluating Model Performance

Student correctly implements the `fit_model` function in code.

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.

Nice explanation of k-fold cross-validation, good work. Note that there are several variations to k-fold cross-validation in sklearn. ShuffleSplit is explained in this project, and another notable one is StratifiedShuffleSplit which is useful when you have greatly unbalanced class allocation, for example in the case of cancer prediction (i.e. there are only very few observations where cancer is positive, compared to the entire population).

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

An accurate description of GridSearchCV is included in the answer, good job.

Note that on larger machine learning projects with a vast hyperparameter space, especially when some of these hyperparameters are continuous, it is recommended to use RandomizedSearchCV due to its ability to search over a huge space. If you're interested in learning more about scikit learn outside of course materials, you may want to check out this workshop by Andreas Mueller, core developer of scikit learn, contains very useful tricks to work with a large dataset.

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 for each of the three predictions as to whether these prices are reasonable given the data and the earlier calculated descriptive statistics.

I think the answer is effective enough in describing the intuition behind your decision, so I qualify this answer as passing the specification. However, I highly recommend that you improve this answer by quantifying the justification.

To correctly decide whether the predictions were reasonable, we can use calculated descriptive statistics above. Firstly, we can compare predicted features with descriptive stats as given by function `features.describe()`, then see how far from the mean their predicted values are. For example, if we see that RM is very low and LSTAT is quite high, then we'd expect predicted price to be really low, as compared to mean of MDEV (one sure way to compare is by seeing how many standard deviations).

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

Good answer. I agree that features from very old data like in this case won't likely result in good model prediction, and more features and more complex models may be required here.

DOWNLOAD PROJECT

RETURN TO PATH

Rate this review

Student FAQ