UDACITY

Logout

PROJECT

Predicting Boston Housing Prices

A part of the Machine Learning Engineer Nanodegree Program

PROJECT REVIEW | CODE REVIEW | NOTES

Requires Changes

3 SPECIFICATIONS REQUIRE CHANGES

SHARE YOUR ACCOMPLISHMENT

Congratulations for a very strong submission! You just need to beef up some of your answers to meet all specifications, but I'm confident this won't be much of a challenge. 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.**

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

Rate this review

☆☆☆☆☆

Suggestion

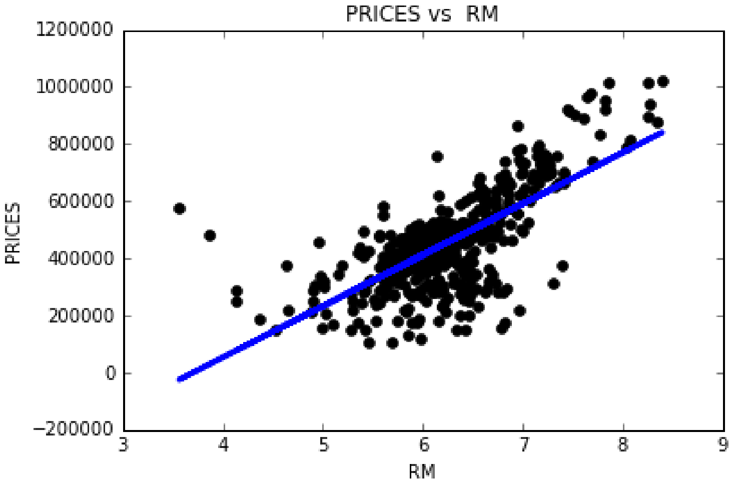
You could add some lines of code to check the scatterplots between the prices and the other variables:

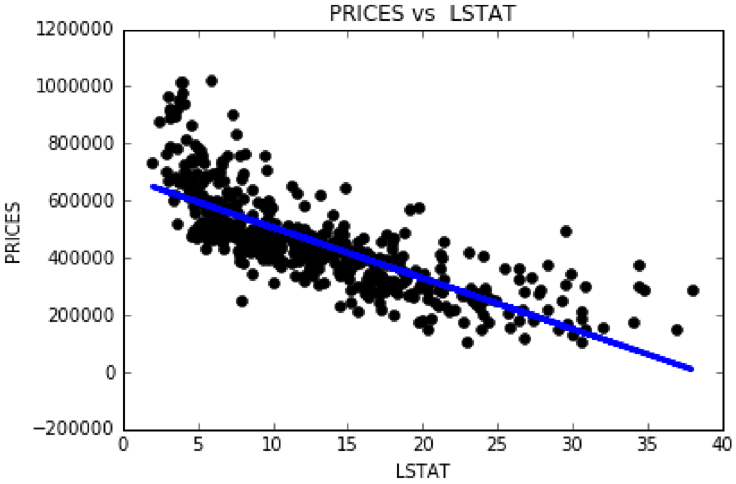
```
import matplotlib.pyplot as plt
import seaborn as sns

for var in ['RM', 'LSTAT', 'PTRATIO']:
    sns.regplot(data[var], prices)
    plt.show()
```

Which would yield something like this:

Rate this review

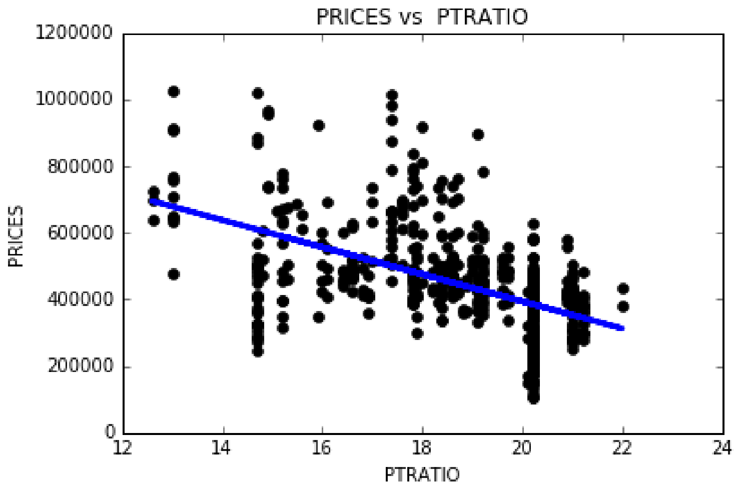
☆☆☆☆☆





Rate this review

☆☆☆☆☆



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.

Required

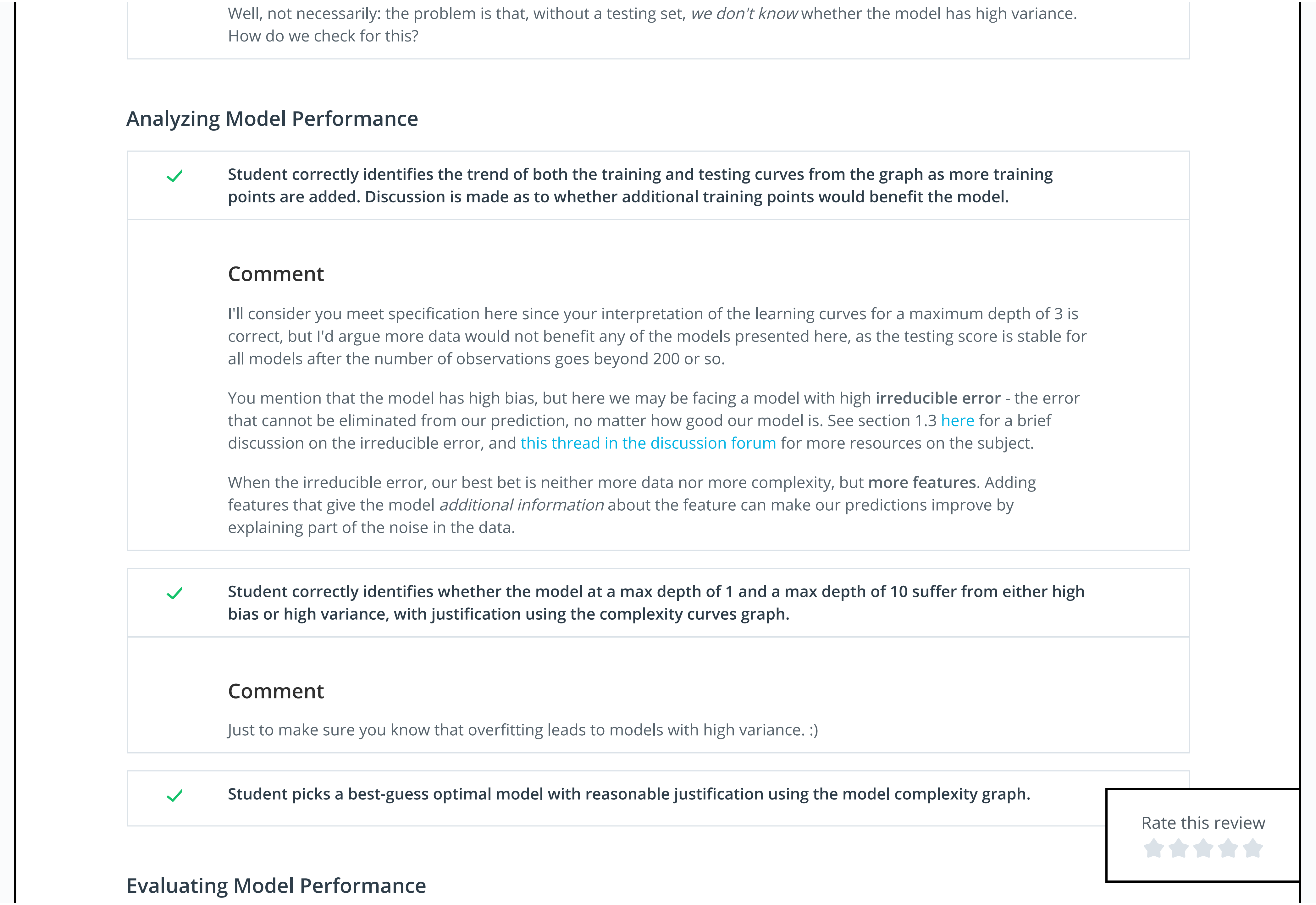
What's missing here is a discussion on whether the prediction captures the variation of the target. You can compare the model's R² to the maximum possible value as a start. :)

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.

Required

> No test data leads to a model with high variance.

Rate this review
☆☆☆☆☆



Well, not necessarily: the problem is that, without a testing set, *we don't know* whether the model has high variance. How do we check for this?

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.**

Comment

I'll consider you meet specification here since your interpretation of the learning curves for a maximum depth of 3 is correct, but I'd argue more data would not benefit any of the models presented here, as the testing score is stable for all models after the number of observations goes beyond 200 or so.

You mention that the model has high bias, but here we may be facing a model with high **irreducible error** - the error that cannot be eliminated from our prediction, no matter how good our model is. See section 1.3 here for a brief discussion on the irreducible error, and this thread in the discussion forum for more resources on the subject.

When the irreducible error, our best bet is neither more data nor more complexity, but **more features**. Adding features that give the model *additional information* about the feature can make our predictions improve by explaining part of the noise in the data.

✓ **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.**

Comment

Just to make sure you know that overfitting leads to models with high variance. :)

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

Rate this review

☆☆☆☆☆

Evaluating Model Performance

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

Awesome

Great idea to analyze grid search through the eyes of a programmer and to mention its time complexity - although I believe you mean, for h parameters with n options each? Also, this is just the number of loops - for the full grid search time complexity we'd have to consider the time it takes to train our model as well!

Comment

One alternative to deal with the cost of implementing full grid search is randomized search, which will train the model over a predefined number of parameter combinations picked at random from the full grid.

↻ **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.**

Required

> Combined with grid search, we can ensure that a model has better bias than if it had been trained with the whole set.

Ok, but does k-fold ensure this? Think of it this way: if we use a single testing set to check the results of each instance of the model, with different sets of parameters, how sure are we that the model generalizes well, instead of simply being good *for that specific testing set*?

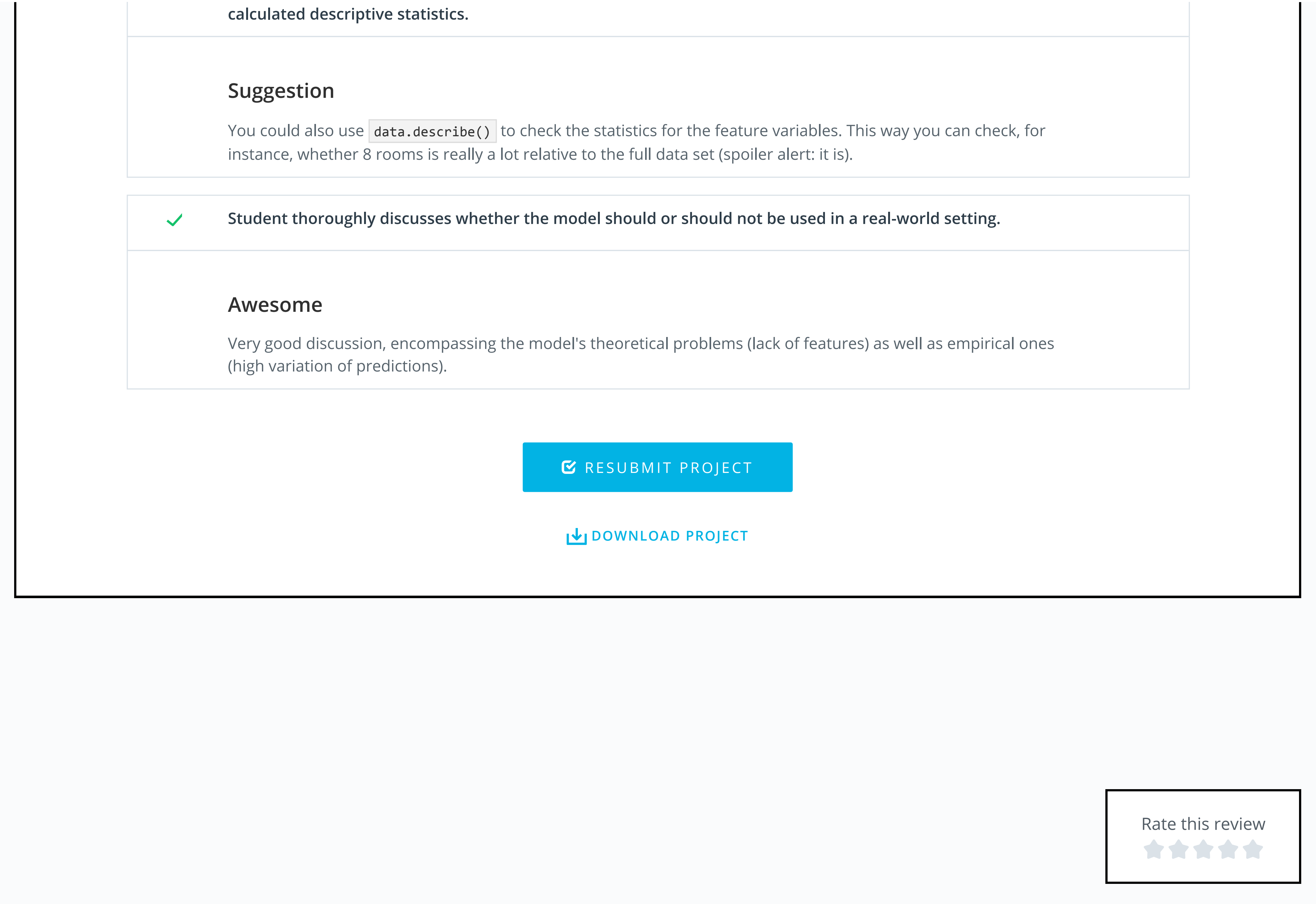
✓ **Student correctly implements the `fit_model` function in code.**

✓ **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 for each of the three predictions as to whether these prices are reasonable given the data and the earlier**

Rate this review

☆☆☆☆☆



calculated descriptive statistics.

Suggestion

You could also use `data.describe()` to check the statistics for the feature variables. This way you can check, for instance, whether 8 rooms is really a lot relative to the full data set (spoiler alert: it is).

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

Awesome

Very good discussion, encompassing the model's theoretical problems (lack of features) as well as empirical ones (high variation of predictions).

Best practices for your project resubmission

Ben shares 5 helpful tips to get you through revising and resubmitting your project.

▶ Watch Video (3:01)

Rate this review
☆☆☆☆☆

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