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

Data Exploration

✓

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

Well done! You have shown a good understanding of making use of `numpy` to get the descriptive statistics of the dataset.

✓

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

Well done for identifying the relevant features' correlation.

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.

- Great work done here fore reporting the R2 score correctly.
- The r2 score measures the goodness of how fit the model is, the greater the r2 score is, the better the model is fitting.
- Please note that for the performance metrics, we have several different ways to measure the model performance, Explained Variance Score and R2 Score are metrics to measure how well the model fits the data, explains the variability in predictions. The other possible options are mean squared error and mean absolute error etc.
- Please look at here for the list of the performance metrics.

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.

SEEDING YOUR ALGORITHMS:

- In order to remove randomness of your algorithms, and to make sure your results don't differ at each run, please consider to always use a random seed to seed your algorithms.
- A standard practice I've come across is to define a random seed as a global variable in your work, and to use it throughout all the algorithms/methods which require random number generation (splitting data, decision tree initialisation, neural network weight initialisation etc).
- In sklearn, as far as I know, random seeds are provided to methods and functions using the parameter `random_state`. Please seed all of your algorithms in the future if you haven't been doing so yet

Analyzing Model Performance

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 correctly identifies significant qualities of the training and testing errors as the training set size increases.
Alternatively, 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.

- To give more context, when the training set is small, the trained model can essentially "memorize" all of the training data. As the training set gets larger, the model won't be able to fit all of the training data exactly.
- The opposite is happening with the test set. When the training set is small, then it's more likely the model hasn't seen similar data before. As the training set gets larger, it becomes more likely that the

model has seen similar data before.

✓

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

- Please note that with the increasing model complexity, the model goes through two stages from underfitting to overfitting - Please consider to include this in your report
- The first phase is where the model is underfitted and the training error is extremely low.
- The second phase is where the model is overfitted and the difference between testing and training score is high.
- The optimal model is where the turning point at, which the training score is high and testing score is at global maximum.

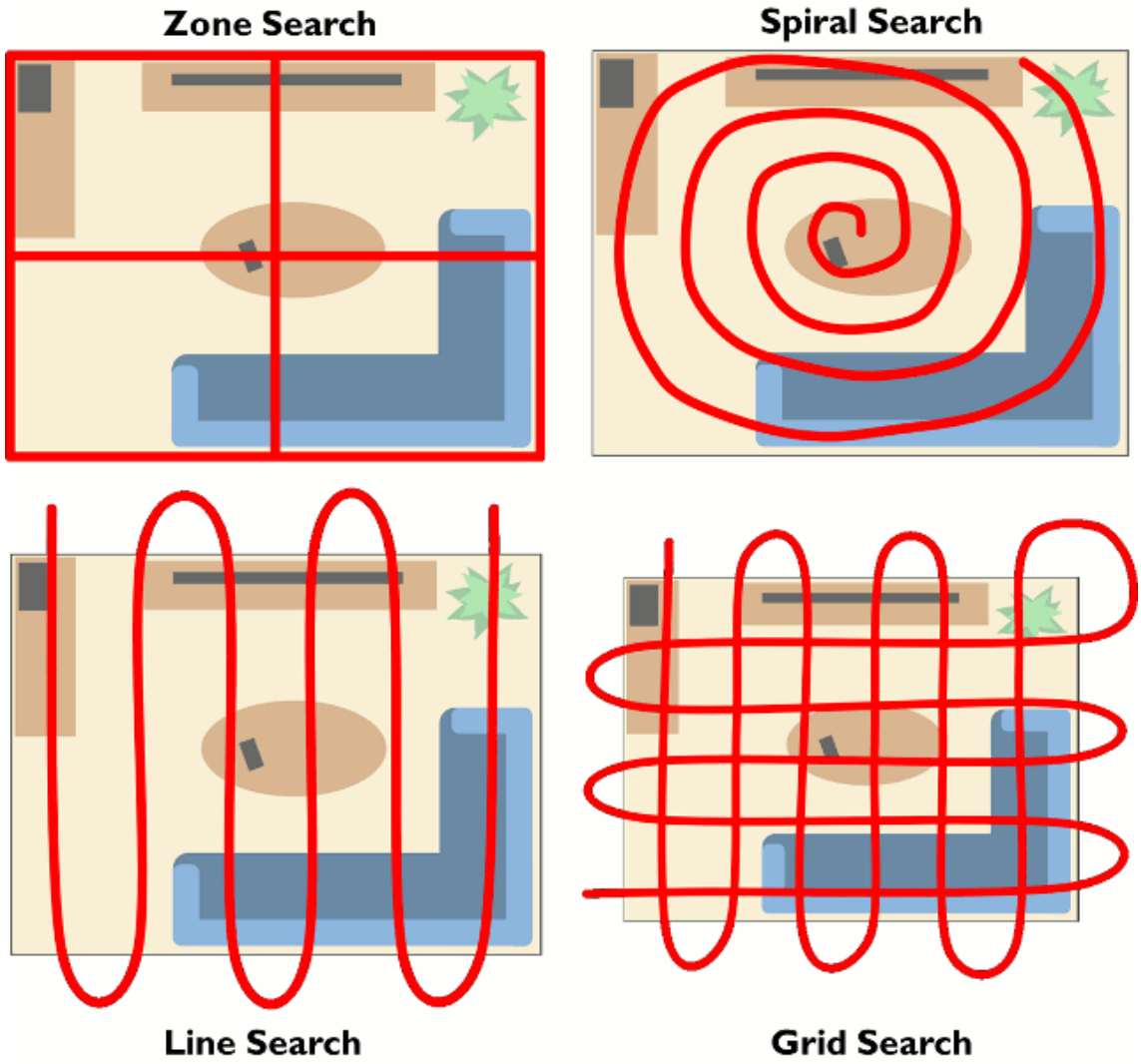
Evaluating Model Performance

✓

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

- Similar to what you have included regarding to grid search algorithm which is simply an exhaustive searching through a manually specified subset of the hyperparameter space of a learning algorithm. A grid search algorithm must be guided by some performance metric, typically measured by cross-validation on the training set
- It would be worth mentioning about fine tuning a learning algorithm for a more successful learning/testing performance in terms of the application for grid search.

Please look at the following comparison for different space search:



✓

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 is useful because it maximize both the training and testing data so that the data we can use to provide best learning result and best validation - this is extremely useful when the dataset is limited in size as Grid Search will allow an extensive exploitation of available data.
- If we limit grid search to single testing set, we may accidentally over fit our model if the split testing set is imbalanced. Using cross validation the parameters will be optimized on the entire data set and any random anomalies due to random splitting will be removed.
- Grid Search with Cross-Validation Scheme would make the brute force parameter search better in the way that each model would be trained with best learning result and best validation.

✓

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

✓

Student reports a valid predicted selling price for the client's data and adequately justifies the prediction using the earlier calculated statistics.
Alternatively, if three clients are listed, discussion is made for each client as to whether these prices are reasonable given the data and the earlier calculated statistics.

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

Student FAQ