

ndgigliotti / dsc-phase-2-project

forked from [learn-co-curriculum/dsc-phase-2-project](#)


 View license

☆ 0 stars  106 forks

☆ Star

👁 Watch ▾

 Code


 Pull requests

 Actions

 Projects

 Wiki

 Security

 Insights

 Settings

 main ▾

...

This branch is 65 commits ahead of learn-co-curriculum:main.

 Pull request

 Compare



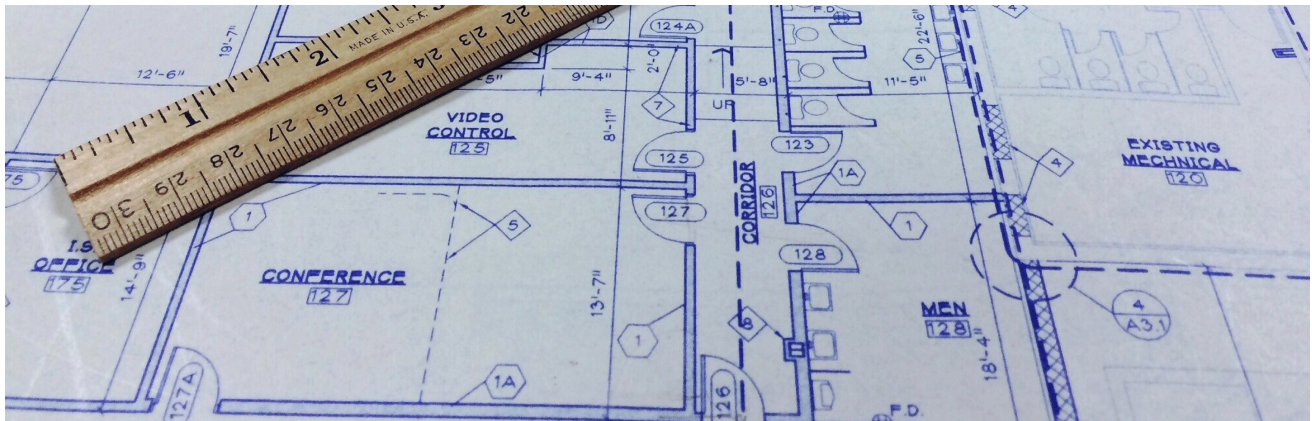
ndgigliotti Update README.md ...

2 hours ago

🕒 74

[View code](#)

☰ README.md



Factors Affecting a Home's Value

- Name: Nicholas Gigliotti
- Email: ndgigliotti@gmail.com

Business Problem

In this notebook, I conduct an analysis of the factors which contribute to a home's value for Data Driven Realty, a company which serves the residents of King County. King County home-owners are looking to renovate their homes to increase their value, and need some advice on where to direct their efforts. Naturally, I prioritize **renovatable features** over other features in my analysis. My central research question is:

What renovatable features have the strongest positive effect on price?

I will attempt to answer this question by developing an ordinary least squares multiple regression model which is both (1) highly interpretable and (2) reasonably accurate. Interpretability is my main priority, since there are many ways to sacrifice interpretability to increase accuracy. Nevertheless, I try to select a model which meets the assumptions of linear regression to a reasonable degree, fits the data well, and deals with renovatable features.

After cleaning the data and dealing with outliers, I run several feature selection sweeps (see [sweeps.ipynb](#)) to guide my model-building process. In other words, I create thousands of models and record their statistics, and then analyze them (in this notebook) in order to decide which features to include in my model.

Data

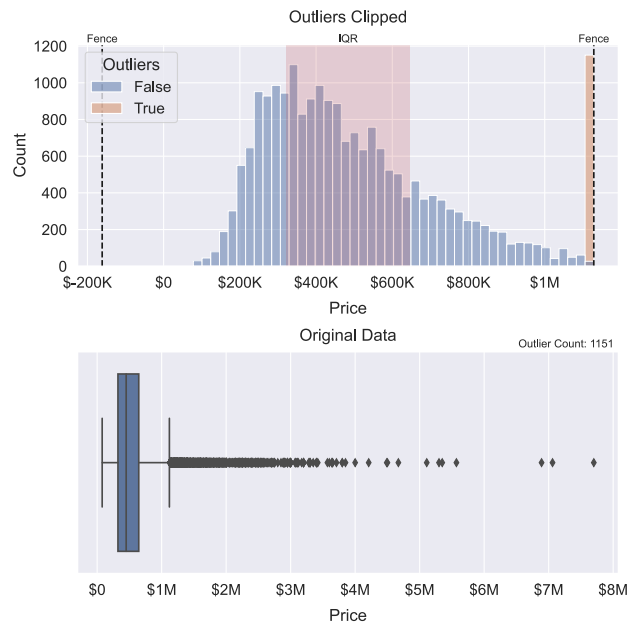
The complete King County real estate dataset includes about 21 features and 21,600 observations. It probably originated with the [King County Department of Assessments](#), but has been passed around a lot amongst data scientists on the internet (e.g. on [Kaggle](#)). My final model involves the features "price", "bedrooms", "bathrooms", "view", and "zipcode".



I also make use of a zipcode dataset downloaded from [unitedstateszipcodes.org](#). It's about 14 features and 42,632 observations. I'm only interested in a small subset of these observations related to the King County zipcodes, namely "primary_city".

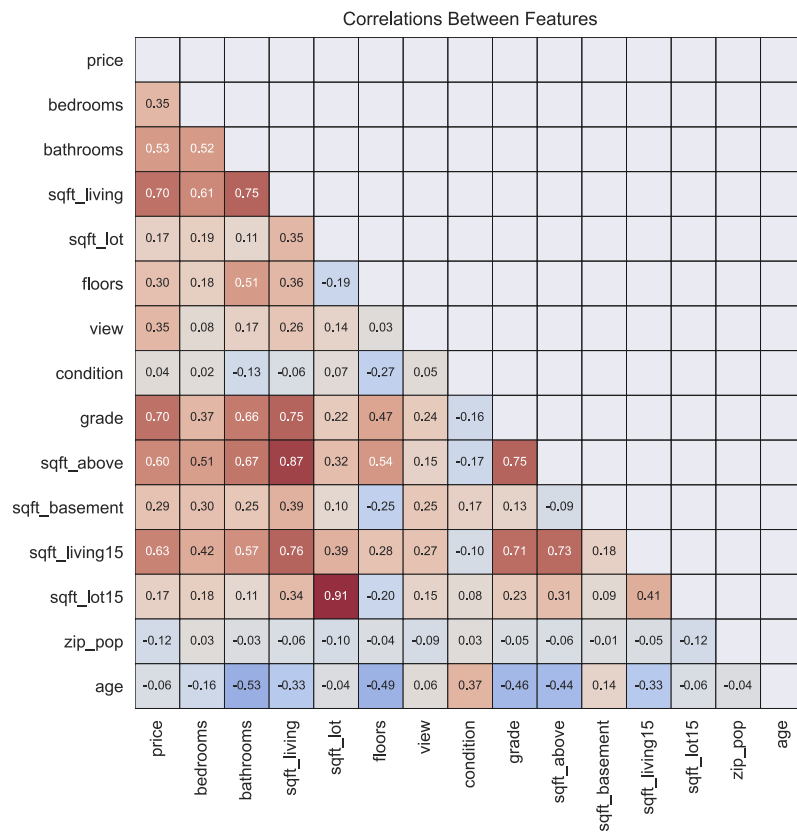
Methods

I clean the data in typical fashion, dealing with NaNs and duplicates and converting categorical variables. I deal with outliers by clipping them (i.e. moving them) to the Tukey fences surrounding the IQR. I use the IQR method because most of the feature distributions are very right-skewed. I clip the outliers to the IQR fences rather than dropping them in order to avoid the collateral damage that comes with dropping outliers from every feature (losing ~18% of the data, in this case). Clipping the outliers also preserves more of the dataset's accuracy, although it often results in a large artificial clusters at the edge of distributions. See the illustration below:



I do some feature engineering before processing all of the outliers. This includes creating categorical versions of numeric variables (e.g. "bedrooms" and "age"), and creating a "nearby_city" feature based on "zipcode".

After preparing the data, I produce numerous heatmaps to explore correlations between features. This is important in order to avoid multicollinearity in my regression models. I also make scatterplots to examine any possible linear relationships with price.



In an auxilliary notebook (sweeps.py), I run feature selection sweeps for OLS multiple regression by building models for nearly every combination of features and recording their statistics. I do this for 1-variable through 5-variable regressions. Then I use this sweep data in the main notebook to iteratively develop a final model which has high R^2_{adj} and many renovatable features.

Conclusion

I arrive at the following recommendations for King County residents:

- Increase your housing grade as much as possible.
- Try to attain a view rating of at least 1 by landscaping, installing windows, or removing junk.
- Aim to have at least 4 bedrooms, and then expect diminishing returns if you keep adding more.
- Add more full, half, and quarter-bathrooms.

Final Model

I'm confident in that these recommendations will help residents increase the value of their homes through renovation. I selected my final model from about 21,700 other models with a wide variety of predictor combinations. I filtered through this massive set of models to find the model with the best combination of R^2_{adj} , relevant predictors, residual normality, and residual homoscedasticity.

My final model has an R^2_{adj} of 0.813, meaning that it explains 81.3% of the variance in price. Its residuals are not perfectly normally distributed, nor are they perfectly homoscedastic, but I have to sacrifice some accuracy in order to maintain maximum interpretability. Based on the diagnostic plots, I'm confident that the coefficients of the model are reasonably accurate.

Future Work

There is much more that could be done for King County homeowners. The first step would be to look at other models I've created and learn about features such as "sqft_living" and "sqft_lot" which didn't make it into `final_model`. There is much more that could be done with data from this very notebook.

Speaking of data from this very notebook, I'd like to run feature-selection sweeps using the normalized datasets, `log_df` and `quantile_df`, I created. I'd be interested in seeing how much better the scaled models are, and what insights they offer at the expense of interpretability.

My final suggestion is to look for more data, ideally data with time information gathered over the course of decades. I'd be interested in investigating the change over time in how various features affect home value. For someone looking to renovate, the best investment would be in features that are on the rise.

Releases

No releases published

[Create a new release](#)

Packages

No packages published
[Publish your first package](#)

Languages

● Jupyter Notebook 100.0%