Riley Carlin

6 Dec 2023

Final Project Summary

For my project, I wanted to find out how much impact the age of a house has on the price of the house. To do this, I planned on finding a well-fitting linear model– if such a thing could exist– and then examining how said model used the age of a house to predict the sale price. In my ultimately unsuccessful attempt to find a trustworthy linear model, I got to explore various model selection methods, as well as compare the performance/ results of different sized models.

I chose this topic because I love looking at old houses for sale but have a hard time judging their prices out of context. Without looking at other houses in the same area, it's hard to tell if older houses are cheaper because they're old (maybe it's harder to sell a house that requires more upkeep) or because of location/ other aspects of the house. Thankfully, I was able to find a dataset of houses that were sold in the same metropolitan area around the same time to help minimize the effect of location on the patterns in prices.

I used Eric Pierce's "Austin, TX House Listings" dataset on Kaggle because it had plenty of observations (15,171) and variables (47), including yearBuilt (house age was surprisingly rare in the housing datasets I looked through). This data came from Zillow's own API, which I figured was a reliable source for house listings; additionally, Pierce had done some cleaning to get the API output down from 736 variables. To ensure that the clean dataset was still trustworthy, I did a little digging to see how Pierce cleaned the data. The majority of changes were from removing mostly empty columns or condensing very similar columns into one (i.e., we don't need 5 columns about the number of bathrooms). However, a little over 1000 rows were also dropped, for reasons I never figured out.

Before I made my models, I did some preliminary data exploration and cleaned up columns as necessary. A histogram of yearBuilt suggested that some of the older years may have too few houses, so I made a decadeBuilt column to try to alleviate that (however, performance between models with yearBuilt and decadeBuilt did not differ by much). My correlation heat map made me realize that parkingSpaces and garageSpaces had a 0.9997 correlation (so I removed the latter); while there were other strongly correlated variables (e.g., avgSchoolRating and MedianStudentsPerTeacher), I chose not to delete them because they carried different information and interacted with other variables differently. I made zipcode a categorical variable, removed the column for photos, and removed any listing not for a single-family house. Finally, I condensed or removed any categorical variable that had over 50 unique values, depending on how many unique values there were. For example, I edited latestPriceSource since 95% of the data had one of two values, but I deleted streetAddress since all but five values were only seen once.

I attempted 5 sizes of linear models. The first was a simple linear regression of latestSalePrice over yearBuilt (or decadeBuilt). While I was happy to see that R thought yearBuilt's coefficient (and some belonging to the decades) was statistically significant, the adjusted R^2 left much to be desired. This was not an unexpected outcome, as a house's price is obviously not mainly determined by its age– this model most likely suffered from omitted variable bias. The full model produced a similar, significant beta for yearBuilt with an adjusted R^2 of over 50%. However, I did not believe that all 41 covariates were that useful, so I turned to stepwise and lasso regression to build me a smaller (and hopefully more effective) model. Since these regressions needed to both pick variables *and* fit a model, I broke the data into a 70/30 train/test split to check that the chosen variables weren't overfitting the data. While both the AIC models and BIC models selected yearBuilt and lasso gave yearBuilt a non-zero coefficient– points in favor of house age being useful for predicting price– the outputs and/ or performance for/ on the testing data was often surprisingly different from that of the training data (or even the naive full model). These startling discrepancies made it hard for me to say with confidence what kind of impact age has on price (see table at end).

The two biggest challenges I faced involved cleaning the data and getting the variable selection algorithms to run. It turned out that these issues were related. My program kept timing out when I tried to fit the full model; at first, I thought this was simply because I had too many variables– a good argument for variable selection. But then I realized that the built-in stepwise regression functions needed a full model as input. I spent an embarrassingly long amount of time debugging, trying to see why the stepAIC function took one step every five hours… and then I realized I still had a categorical variable in the dataset that practically had one unique value per house (so my model had to find more than 14,000 betas for one concept). I think knowing that the dataset had previously been cleaned made me a little careless when it came time for me to do my own cleaning. Overall, this project allowed me to get more comfortable with stepwise and lasso regression in R, as well as taught me to be more critical about which variables I consider for a linear model.

	Simple	Full Model	AIC Train	AIC Test	BIC Train	BIC Test	Lasso Train	Lasso Test
yearBuilt Coefficient	1209.2	1069	1179	557.5	1196	568	-655	-655 (uses Train's coefficients)
Significance	3.31e-12	7.07e-9	2.08e-10	0.135	1.16e-10	0.128	NA	NA
Adjusted R^2	0.00333	0.5552	0.6357	0.5475	0.6348	0.5469	0.560	0.118

Output and Performance Summary for Models using yearBuilt.