# Data Mining Project - House Prices

Raul Marinău

June 12, 2021

## Introduction

The dataset we are going to analyze in this report is House Prices - Advanced Regression Techniques (https://www.kaggle.com/c/house-prices-advanced-regression-techniques/overview). The dataset contains 79 variables describing almost every possible aspect of residential homes in Ames, Iowa (USA).

This dataset was used in a prediction competition on Kaggle, with the main challenge being predicting the final price of each home.The main skills meant to be practiced are creative feature engineering and regression techniques.

There are 4 files available: a file that contains the descriptions of all the variables, test and train data in two separate csv files and a submission sample. The test and train datasets are of similar size, about 1460 entries. The test csv file doesn't contain the target variable *SalePrice*.

## Data preprocessing

We are going to look at the processing steps that were applied to the train dataset, similar ones were then done to the test dataset.

### *SalePrice* exploration

In this part we are mainly going to explore our data and the possible relationships it has with the target. We start off by plotting the distribution of our house prices:

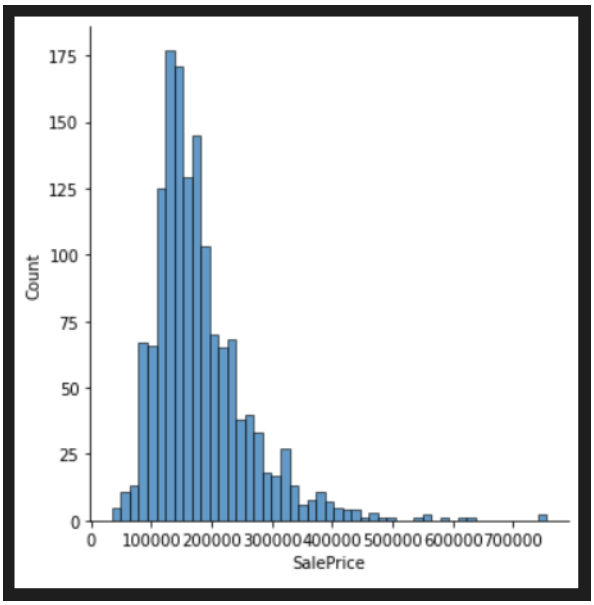


Figure 1: Sale Price Distribution

There are already two things we can observe by doing this: our target resembles a normal distribution and that it has a pretty large positive skewness.

Next, we are going to analyze a correlation matrix of:

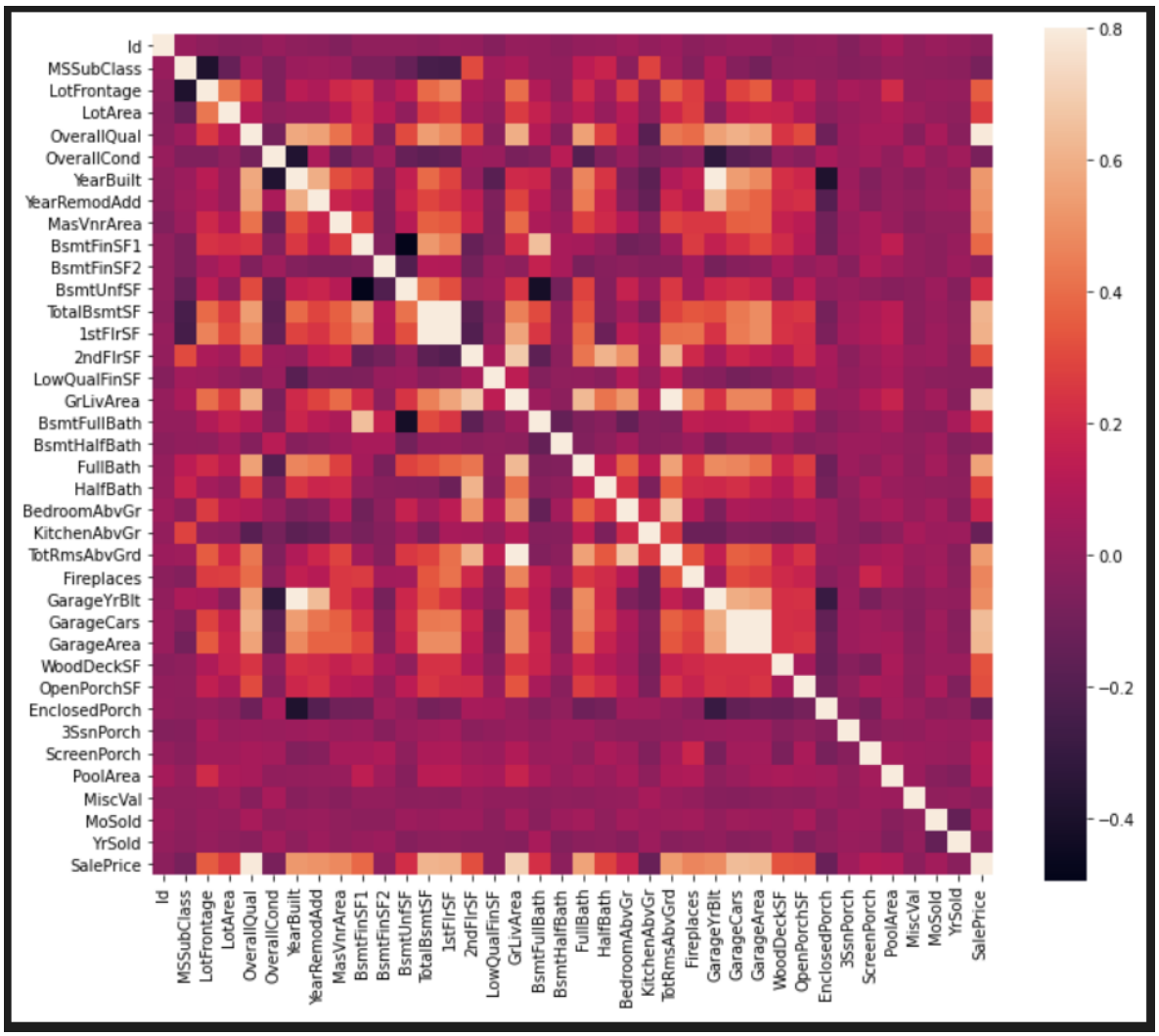


Figure 2: Correlation matrix (heatmap)

In this figure we are mainly looking at the very last column (or row) in order to see which variables have the highest values (the brightest color). Unsurprisingly, *OverallQual* (overall material and finish quality) has one of the highest correlation to the sale price; closely followed by:

- TotalBsmtSF: Total square feet of basement area
- 1stFlrSF: First Floor square feet
- GrLivArea: Above grade (ground) living area square feet

## Missing Data

We are now going to look at the missing data. The variables with very high percentage of missing data (pool quality, miscellaneous features, alley access, fence quality) can be seen as optional stuff, or thing people don't really consider to be a priority when they want to purchase a house.

The other variables are either related to garage or basement. Because from our correlation matrix, we obtained high values for *GarageCars* and *TotalBsmtSF* it would be safe to assume that dropping these variables won't negatively impact our model. For the *Electrical* variable, we are only going do remove the row with missing data.

| | Total | Percent |
|---|---|---|
| PoolQC | 1453 | 0.995205 |
| MiscFeature | 1406 | 0.963014 |
| Alley | 1369 | 0.937671 |
| Fence | 1179 | 0.807534 |
| FireplaceQu | 690 | 0.472603 |
| LotFrontage | 259 | 0.177397 |
| GarageYrBlt | 81 | 0.055479 |
| GarageCond | 81 | 0.055479 |
| GarageType | 81 | 0.055479 |
| GarageFinish | 81 | 0.055479 |
| GarageQual | 81 | 0.055479 |
| BsmtFinType2 | 38 | 0.026027 |
| BsmtExposure | 38 | 0.026027 |
| BsmtQual | 37 | 0.025342 |
| BsmtCond | 37 | 0.025342 |
| BsmtFinType1 | 37 | 0.025342 |
| MasVnrArea | 8 | 0.005479 |
| MasVnrType | 8 | 0.005479 |
| Electrical | 1 | 0.000685 |
| Id | 0 | 0.000000 |

Figure 3: Missing data

## Data normalization

In order to ensure normality for our data, in case of a positive skewness, the simplest trick we can apply is a logarithm transformation. Remember our first histogram (1), and let's see what happens after applying the transformation in figure 4.

We then applied the same technique to *GrLivArea* and *TotalBsmtSF*.

This log transformation also helped solved the problem of homoscedasticity - the assumption that dependent variable exhibit equal levels of variance across the range of predictor variable(s). Basically, we can draw a line that goes within range of most values (figure 6).

## Categorical data

For categorical data in our dataset we only performed a simple step, namely we used *get_dummies()* of our pandas dataframe.

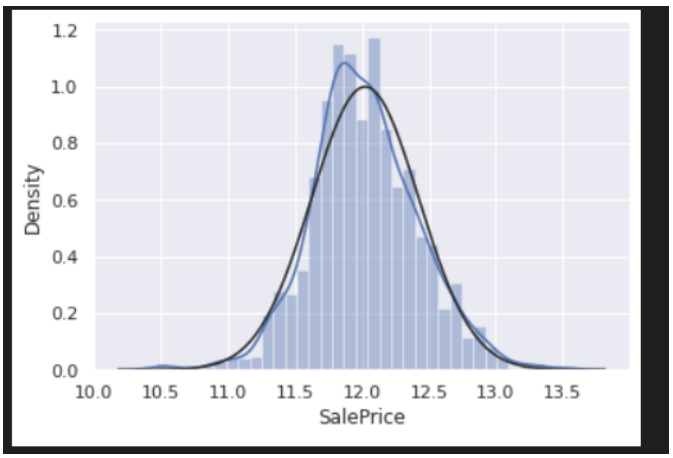


Figure 4: Histogram for log*SalePrice*

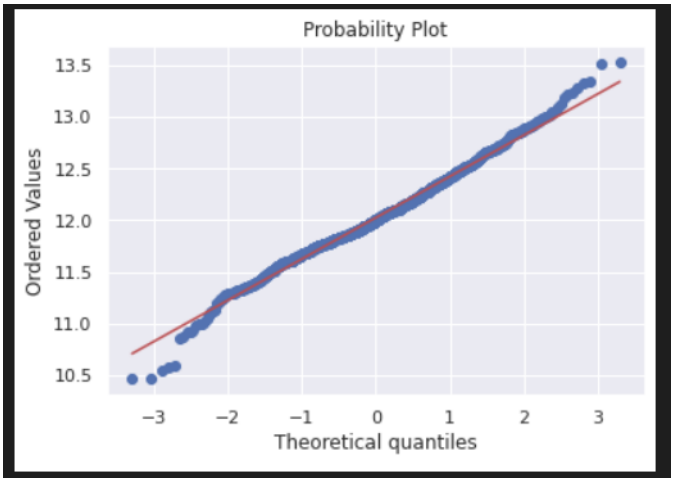


Figure 5: Quantiles plot

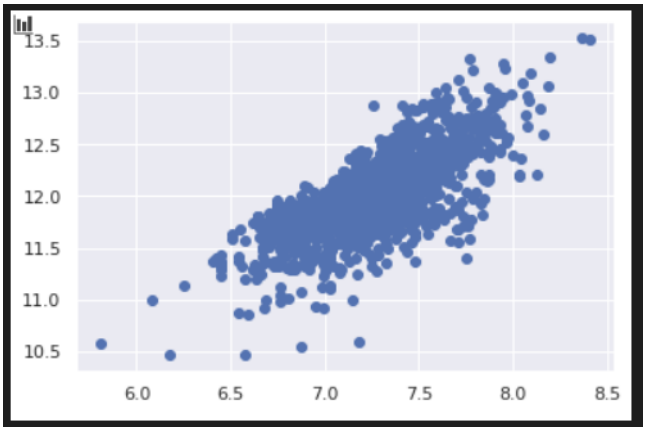


Figure 6: *SalePrice* and *GrLivArea* scatter plot

## Applying linear models to our data

In order to measure the performance of our linear models, two metrics were used: we computed the mean squared errors (MSE) of our predictions and the coefficient of determination ($R^2$). The scores were computed by predicted the price for the training dataset, since our task was to produce the sale price for the test dataset, as mentioned before, the target was missing.

The models used and their respective scores are as follows:

- MSE=0.0087 and $R^2$=0.9456 scored by LinearRegression

- MSE=0.0000 and $R^2$=1.0000 scored by DecisionTreeRegressor
- MSE=0.0410 and $R^2$=0.7437 scored by SVR
- MSE=0.0379 and $R^2$=0.7629 scored by KNeighborsRegressor

Although DecisionTreeRegressor obtained the best scores, it can be due to overfitting of our data.

We should have used OneHotEncoder (as explained here) instead of encoding the labels as integers (the *get_dummies()* method of pandas dataframe) in order to obtain accurate scores for the DecisionTreeRegressor.

## Trying to improve our models

Based on the results up to now, we can be fairly confident in our linear regression model. So let's try improving it by:

- applying log transformations to all features that present skewness
- using regularized linear models ridge and lasso

We still transformed our categorical data to dummies and then filled all $NA$ values with the mean of the respective column.

We used Ridge and Lasso with built-in cross validation, which are regularized linear models that add some sort of penalty to coefficients when trying to minimize the cost function to our linear model. Lasso has the added benefit of helping us in feature selection.

Let's take a quick look at the features selected by our Lasso model.

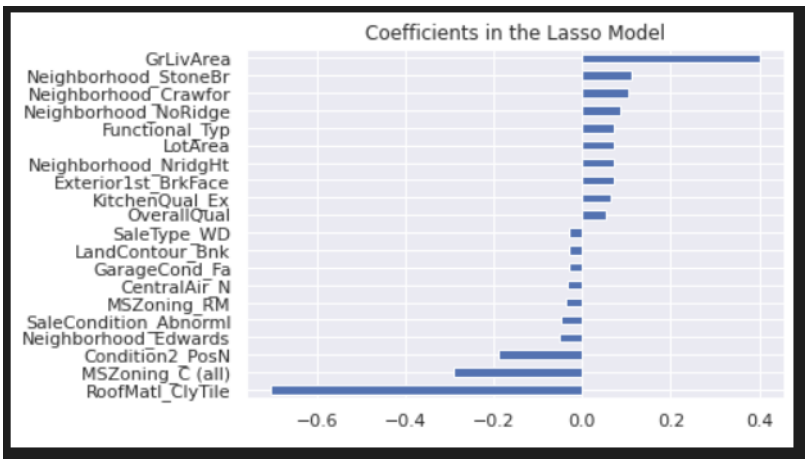


Figure 7: Top 10 best/worst Lasso coefficients

It looks like our lasso model also gave a pretty high importance to *GrLivArea* (Above grade (ground) living area square feet) which makes total sense, as we have also discovered in our data exploration step. Next few features selected are related to the location $NeighborhoodX$, which is such an obvious thing, clearly the location of a house can greatly influence it's price. This goes to show how impressive these mathematical models are.

The models used and their respective scores:

- MSE=0.0112 and $R^2$=0.9300 scored by RidgeCV
- MSE=0.0110 and $R^2$=0.9308 scored by LassoCV

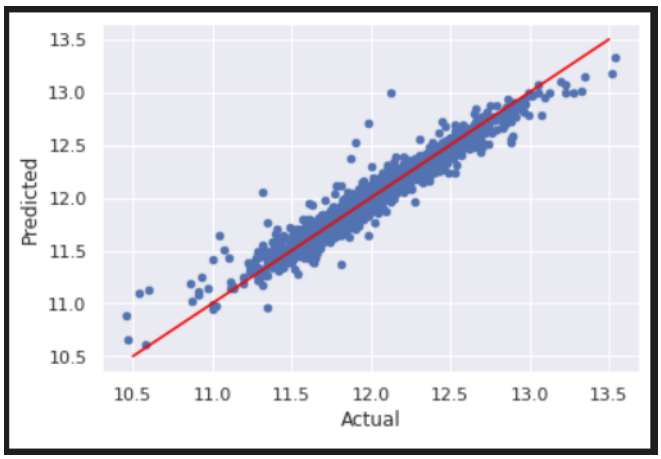


Figure 8: Lasso prediction vs actual

## Conclusions

Although we received slightly better scores in our simple linear regression model, it used 222 features while our Lasso used only 110, thus reducing our model's complexity and multicollinearity (high correlations between our independant variables).

Throughout this paper we discussed about the relationships that influence our target. We discovered different variables that influence the sale price of a house, which make a lot of sense in practical terms and also saw how a more "autonomous" mathematical model tries to discover influential coefficients in our data.

There are a lot of things one can do in order to observe patterns in the data. The exploration part can be one of the most fun parts when interacting with a dataset and finding meaningful information is a reward in of itself.