REPORT

House Prices

Weslley Caldas

Correspondence:
weslleylc@gmail.com
Department of Computer, UFC,
Avenida Humberto Monte s/n ,
Fortaleza, BR

Introduction

This report provides technical and theoretical support to House Prices application. The rest of this report is described as follows: section 1 presents the results of Univariate and Multivariate Linear Regression that implement the Least Squares problem. Section 2 provides analyses of the chosen solution and preprocessing of the data. Section 3 presents the solution to avoid the problem of overfitting. Section 4 summarizes the selected algorithms Lasso, Ridge Regression and Gradient Boosting to solve the regression problem. Finally, Section 5 presents the final model.

1 First Analyze

There are several approaches to solve the a linear regression. The most popular are Ordinary Least Squares(OLS) and Gradient Descendant(GD). While GD uses an interactive approach and needs to choose learning rate α, OLS can produce a similar result without any parameter. GD is faster than OLS for a large database (rows¿10^4). The House Prices dataset has fewer instances than 2000, so I choose OLS to resolve both regression problems for simplicity.

I got the following results(RMSE) using k-folds validation:

1 Multivariate Linear Regression score: 0.1854 (0.0193).

2 Univariate Linear Regression score: 0.3204 (0.0280).

OBS:I add the intercept(vector of ones) to univariate model.

2 Improve the model

2.1 Short solution

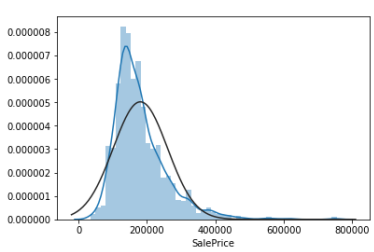
In a nutshell, the previous results were not the best. There are many problems in using all variables, and a certain pre-processing can help with the accuracy of the model.

The choice of the variable was made based on the correlation between the variables. To evaluate the models I chose to use cross-validation and RMSE as metrics. Finally, the choice of algorithms for comparison(Lasso, Ridge Regression and Gradient Boosting) has been made with the intention of minimizing the problem of overfitting.

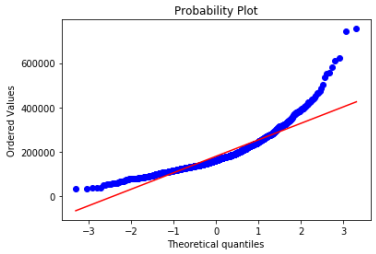
A complete solution can be seen in the rest of the section.

2.2 Target Variable

The first step to analyze the data is to verify the target variable SalePrice. Figure 1 shows the distribution for SalePrice, that deviate from the normal distribution



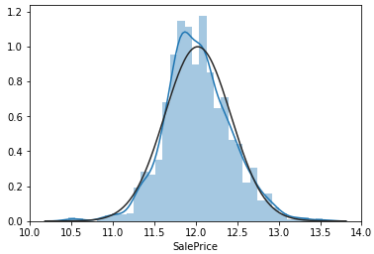
(a) shape of distribution



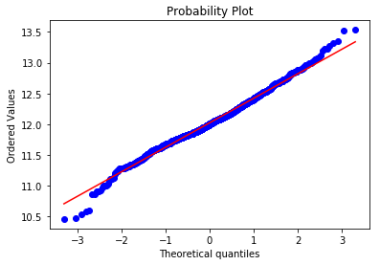
(b) probplot

Figure 1: Investigation of the normality of the target variable.

(Skewness[1]: 1.882876, Kurtosis[2]: 6.536282) with positive skewness and present peakedness . A data normalization (transform the target variable to follows a Normal distribution) can provide us some benefits like Homoscedasticity. It is possible to ensure the difference between what is predicted $\hat{Y}$ and the true values Y should be constant. In other words, the linear regression does not make small errors for low values of X and big errors for higher values of X.



(a) shape of distribution



(b) probplot

Figure 2: Target variable after normalization.

A common way to change a skewness Distribution into a Normal Distribution is using the log function. Another advantage using log normalization is because the linear regression typically minimizes $\|\hat{y} - y\|^2$, so the estimator is going to be very concerned about minimizing some big outliers. The log transformation minimizes this problem.

Figure 2 shows the new plot after the log normalizations, they have Skewness: 0.121347, Kurtosis: 0.809519, and approaches more than one normal distribution.

2.3 How to choose the variable for Univariate Regression?

We have 79 features, so how to find the best feature to fit univariate regression model? A common approach is using the correlation between variables. The correlation is one way to measure the dependence for two discrete random variables.

[1] The skewness measures the symmetry of a distribution. The standard normal distribution has a skewness of zero, and therefore, it is said to be symmetric.

[2] The kurtosis measures the tail ends of a distribution and whether the distribution of a dataset has skinny tails or fat tails in relation to the normal distribution.

Among all types of correlation, I adopted Pearson correlation coefficient (PCC), that is a measure of the linear correlation between two variables X and Y.

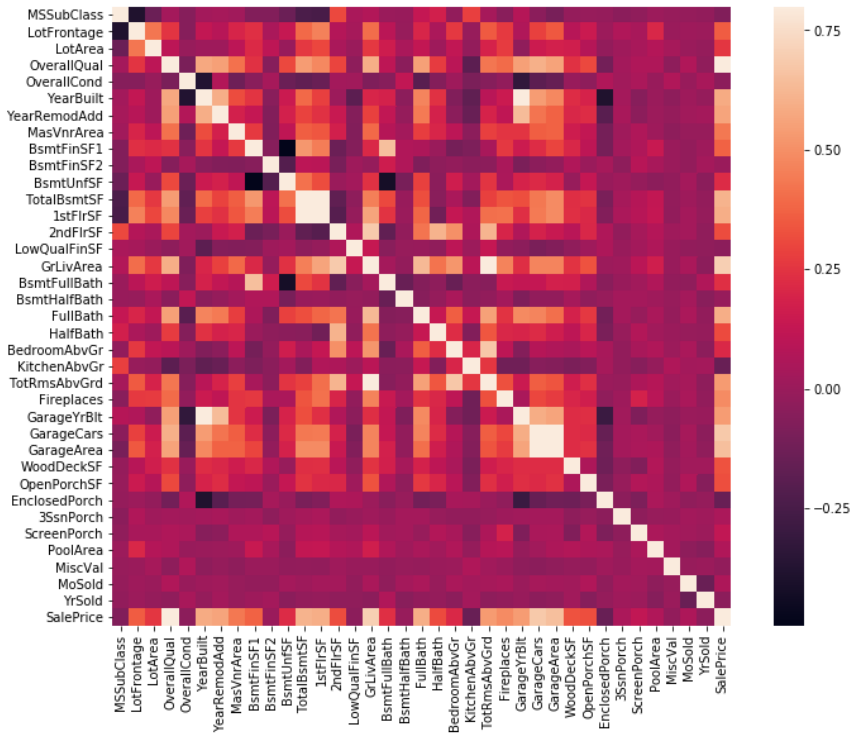


Figure 3: Correlation map between the features.

The correlation map in Figure 3 shows the correlation between all features and the target variable. The best four variables are OverallQual, YearBuilt, TotalBsmtSF, and GrLivArea. I chose OverallQual to fit my linear regression model, because it makes sense that the quality of a house (OverallQual) increases together with the price. The same idea can be said about the total area of the house (TotalBsmtSF), but this variable have a strong correlation with 1stFlrSF and 2ndFlrSF and this can produce a multicollinearity problem.

The year the house was built (YearBuilt) and above grade (ground) living area square feet (GrLivArea), have a strong correlation with SalePrice, but is lower than OverallQual. And in a general way, the quality of the house looks like more important than other features.

2.4 Missing Data

Figure 4 shows that some variables have missing values. We need to trait this problem inputting data or removing these variables. I separated the types of missing values into a couple of cases as below:

1 I decided to drop MiscFeature, Fence, PoolQC, FireplaceQu and Alley, because these features has 50% or more of missing data.

2 For the Linear feet of street connected to property (LotFrontage) I filled missing values using the mean of Linear feet of street of the Neighborhood.

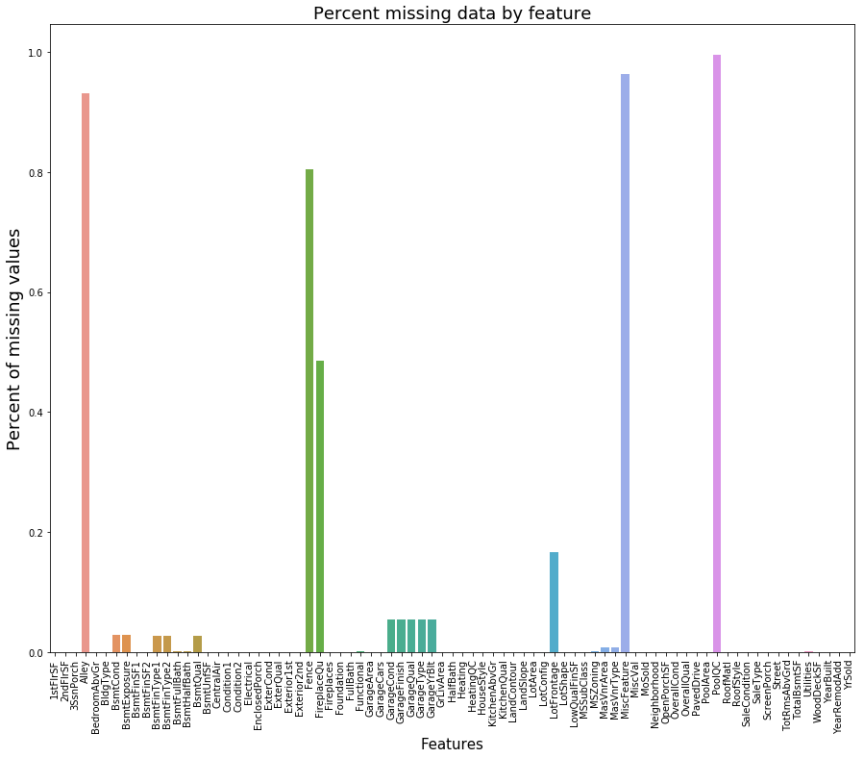


Figure 4: Percent of missing data for all variables

3 Drop Utilities, since basically all instances are AllPub.

4 Electrical,KitchenQual,Exterior1st,Exterior2nd,SaleType,MSZoning has a little bit of missing data instances, so I just replaced the missing values for the most frequent category.

5 GarageYrBlt, GarageArea, GarageCars,BsmtFinSF1, BsmtFinSF2, BsmtUnfSF,TotalBsmtSF, BsmtFullBath, BsmtHalfBath,GarageType, GarageFinish, GarageQual, GarageCond,BsmtQual, BsmtCond, BsmtExposure, BsmtFinType1, BsmtFinType2,MasVnrType,MSSubClass,MasVnrArea, most part of these features has missing values, that probability represent the absence of something (eg basement,garage area). I replaced 0 for numerical and 'none' for categorical features.

6 From the data description "Assume typical unless deductions are warranted",so I replaced missing values in Functional with 'type'.

2.5 Data preprocessing

1 Label Encoding: some variables are categorical, but have integer values. To solve this, I applied One hot encoding.

2 Skewness distribution: for all numeric variables, after an analyze about them skewness, all variables with a skew value greater than 0.75 (The skewness for zero distribution is zero) passed for a log transformation similar to the target variable.

2.6 Multicollinearity Problem

The results showed in Section 1 do not present the better results as possible. Forcing the use of all variables is not a good idea, since some variables can produce noise, and other variables have the same information, breaking the assumption of multicollinearity.

The standard errors of the coefficients increase on the presence of multicollinearity. This means, that coefficients for some independent variables may not being significantly different from 0. In some cases, some variables are statistically insignificant when they should be significant because of the multicollinearity of the variables. Without multicollinearity (and thus, with lower standard errors), those coefficients might be significant. In the Section 3 I will show how to avoid this problem.

2.6.1 Create a new Feature

Recall that TotalBsmtSF ,1stFlrSF,2ndFlrSF are correlated. We can combine these features into a new feature called TotalSF, that's is the sum of the first floor(1stFlrSF) and second floor(2ndFlrSF) of the house.

2.7 Validation

For machine learning problems, a common technique for validation is k-folds. In k-fold cross-validation, the original data set is randomly separated into k equal-sized subsets. For a total of k iterations, a single subsample is selected as the validation data for testing the model, and the remaining k - 1 subsets are used as training data. This technique provides a better degree of confidence than the way that the model will behave in a real scenario. By the way, it is possible to infer some statistical properties like the stand deviation of error for a group of models. I applied a 5-folds cross-validation.

2.7.1 Evaluation Metrics

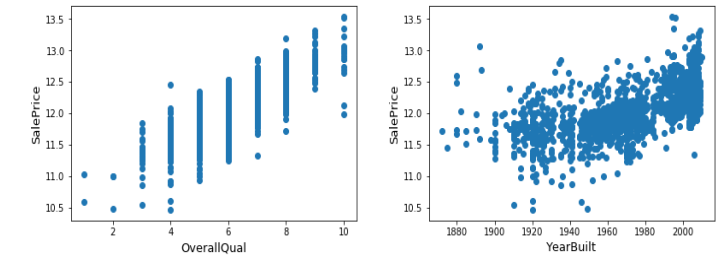
The most common metrics used to measure accuracy for continuous variables is the Root mean squared error (RMSE), that measures the average magnitude of the error. I opted to use this metric to evaluate my results. For a target variable y and a $\hat{y}$ approximated output, RMSE its define as below:

$$RMSE = \frac{1}{n}\sum_{i=1}^{n}\sqrt{(y_i - \hat{y_i})^2} \tag{1}$$

2.8 Outliers

It is possible that exist some outliers, in other words, some instances may cause more problem than offer information. Is important to note, that remove outliers should be a big mistake since only an expert can know if the instance it's a true outlier. Nevertheless, we can explore the most important features, OverallQual, YearBuilt, GrLivArea and TotalBsmtSF .

As seeing in Figure 5a, we don't have any guarantee of outliers for OverallQual. The same assumption for figure 5b and YearBuilt, but in figures 5c and 5d, that represent respectively the GrLivArea and TotalBsmtSF plots among SalePrice, it is possible to see points very far way from the center of the distribution, so I decided



(a) OverallQual and SalePrice. (b) YearBuilt and SalePrice.

(c) GrLivArea and SalePrice. (d) TotalSF and SalePrice.

Figure 5: Relation between the best variables and SalePrice.

to remove this points. Note that, we do not have any guarantee here too, but since the points are well away from the rest of the groups, the chances of being false lower.

3 Avoid Overfitting

As I said, not all variables are important to a multivariate analysis. The problem of multicollinearity imply in data redundancy, and a principal danger of this situation is the overfitting in regression analysis models. We need to remove the redundant variable.

The correlation map on Figure 3 shows two groups of correlated variables. The first group is TotalBsmtSF ,1stFlrSF,2ndFlrSF and we solve this problem combining these variables into a new feature. In the case of GarageYrBlt, GarageArea, GarageCars group, I just removed GarageYrBlt and GarageCars, since GarageArea it's more correlated with SalePrice.

Another way to avoid overfitting is to use regularization models. briefly speaking, regularization penalizes the loss function (the function that we want to minimize to solve a regression problem and find the coefficients w).

A common way to regularization is using L norm regularization. We call Lasso Regression the regression model that uses L1 regularization technique and Ridge Regression the model which uses L2. We'll talk more about these methods in the next section.

Finally, in all experiments I used k-folds cross validation that separate the available data into train (including validation) and test data sets. this methodology is useful to avoid overfitting, since the model is built without knowledge of the data that will be used for evaluation.

4 Lasso, Ride Regression and Gradient Boosting

I choose the bellow methods because of their robustness and simplicity. In addition, all algorithms are well-knowing methods for the academic community.

1 Lasso: LASSO (Least Absolute Shrinkage and Selection Operator) is a linear regression that improves a least-squares estimator, reducing their variance by adding constraints on the value of coefficients. In this case, Lasso penalizes their coefficients with the absolute values (avoiding outlier problem) until some coefficients are nullified (and their features unused). Lasso is resilient against overfitting since they apply an automatic feature selection (avoid multicollinearity) plus that is a robust method because of the regularization.

2 Ridge Regression: it is similar than LASSO, but penalizes the coefficients with a square error instead of absolute value, but it does not enforce them to be zero. That means the variable will not remove but has their impact on the trained model minimize.

3 Gradient Boosting: is an ensemble method that uses many loss functions with tree decisions (weak learners). Their predictions are generated by the gradient descendant combining with a Boosting (combination of multiple weak learners), GB can benefit from different regularization methods to improve the performance of the algorithm by reducing overfitting.

   Scores(RMSE) of all approaches using k-folds:

   (a) Lasso score: 0.1133 (0.0075).
   (b) Multivariate Linear Regression (with Ride regularization) score: 0.1154 (0.0093).
   (c) Univariate Linear Regression (with Ride regularization) score: 0.2704 (0.0087).
   (d) Gradient Boosting score: 0.1154 (0.0055).

5 Final Model

A well-knowing machine learning technique is combining different models into a single model. Intuitively, two brains thing better than one, and two or more models can understand different things about the same data, augmenting the precision of the final model.

My final score submission on Kaglle is 0.11726, that guarantee to me the 490/4523 position (top 11%) on the competition. To achieve this result, I combined the mean of the predictions of Lasso, Ride and Gradient Boosting algorithms.