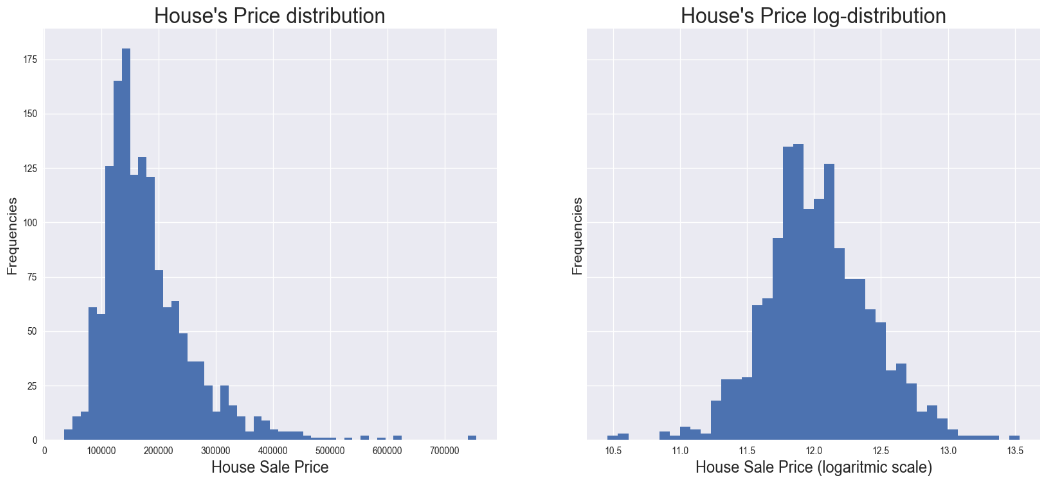
kaggle

House Prices: Advanced Regression Techniques

Project by Roberta Parisi

Introduction: The aim of this project was to predict the price of the houses sold between 2006 and 2010, having 79 variable and 1459 houses (our trainset had 1460 houses) to use for the test. Here i will explaine the steps that i follow to have my prediction. The first thing that i did firts of all was to look at the target value that, as is usual for price or income variables, has a asymmetric distribution (**Graph 1**) and since i want to have a distribution more similar to the Normal I applied the log-transformation.



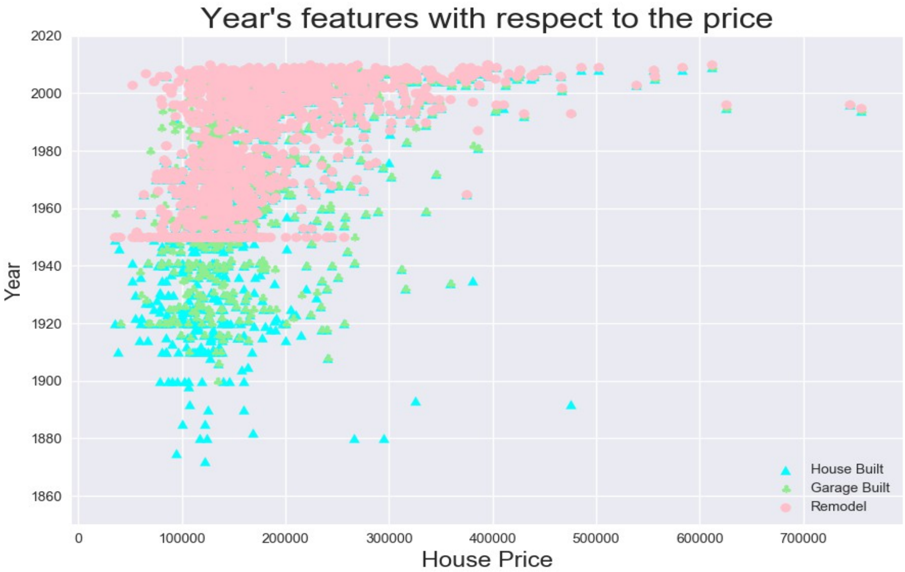
Graph 1: *Price and log+1(Price) distribution compared*

Pre-processing :

1.1 Fill NA value: In the pdf in wich are explained the features I found out that for some variables there were NAs value because particular feature is not present in the house (for exemple the garage), so i treated this missing value like *"false NA"* and i fill it specifying the absence of that feature (e.g. *"NoGarage"*). I handle all the other features with missing values imputing them with the mode for categorical variables and with the mean for the numerical, exept for the variable *"LotFrontage",* that represent the linear feet of street connected to the property, which had a lot of NAs(more than the 10%) so, considering also the small importance (trascurable in my opinion) of this feature, i decided to drop it from the dataset.

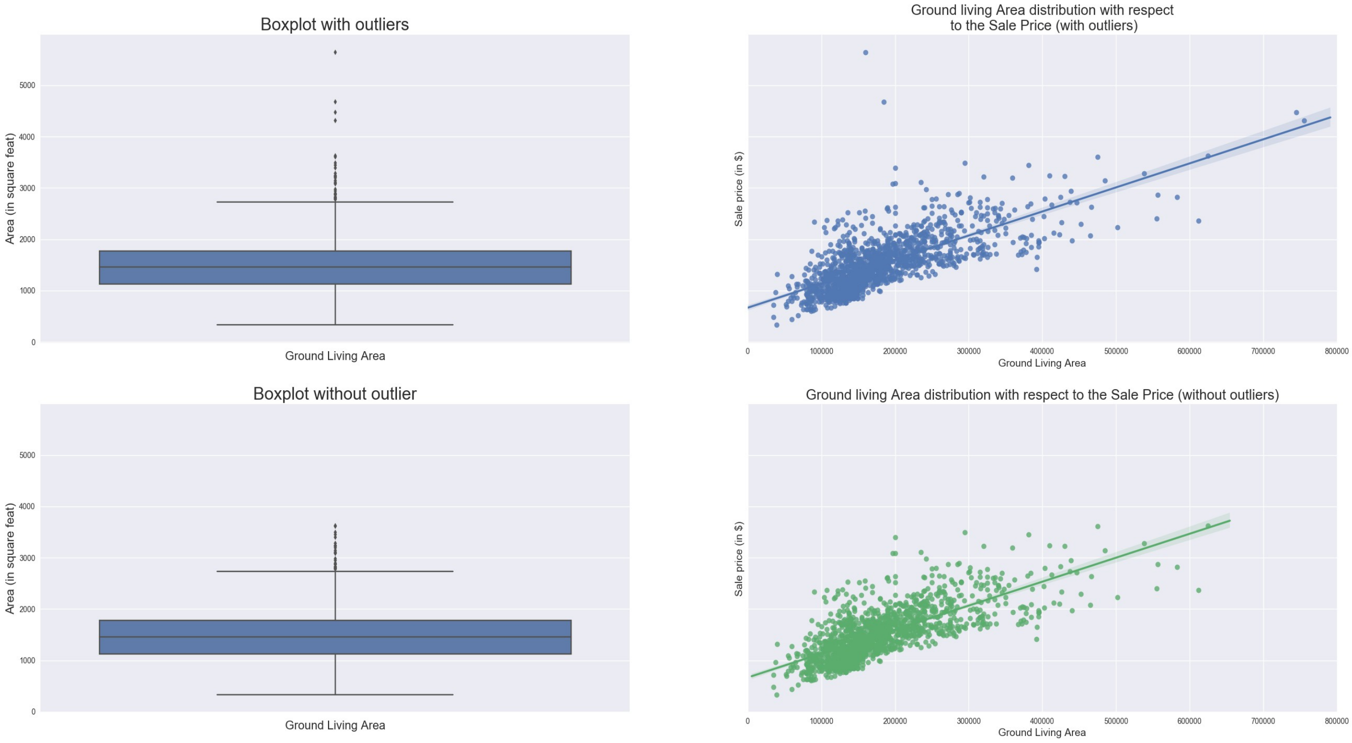
1.2 Delete inconsistency: I found an inconsistency in the variable *"GarageYrBlt"* which has like value 2027, and that i changed with 2007 since it seems to me the more reasonable value.

1.3 Binning: In order to reduce the effects of some minor observation i created some bins for all the year variables (*"YearBuilt", "GarageYrBlt", "YearRemodAdd"*). I created some bins of different length looking at the distribution of each variable (**Graph 2**) and trying to give the right importance at each range (more old year has bigger range bins, more recent year has bins with few modalities) because obviously a modernation made in the 1920 has not the same influence on the price like the ne did in 2001, for example.



Graph 2: *Year of built of the house, of the garage and of the remodernation consider the sold price*

1.4 Outlier analysis: for this section i based my choices on the pdf provided by Dean Decock[1] (the supplier of the dataset), that suggested to remove all the houses bigger than 4000 square feets, and infact I could noticed the presence of some strange values, because despite the greatness of the property, the house was sold at really low price (**Graph 3**). So to handle the outliers, temporarily, I just dropped the outlier of the *"GrLivArea"*, but i will come back later on this topic.



Graph 3: *Distribution of the living area with and without outliers and with respect to the sale price, again with and without outliers*

1.5 Categorical Features analysis: i noticed that some variables had modalities with few observation inside (just like the year's feature that we saw before) and that had the same behaviour (i looked at the boxplot of the feature with respect to our target), so i decided to merge those modalities together. I did the same for *"MSZoning"* (zoning classification) merging the modalities *Residential Medium Density* and *Residential High Density;* for *"MSSubClass"* (type of dwelling) joining *Split Foyer* and Split or Multi-Level; for *"LandSlope"* putting together properties with moderate and severe slope; for *"SaleCondition"* with a particular encoding gained from some descriptive analysis present in my code. After this I made a rating factorization of the variables that express the quality or the conditions of the house (that was expressed before with comment like fair, good and so on). I also created two new variables representing the overall quality and conditions, already inside the dataset, changing their scale from 1:10 to 1:5 to make them similar to the other quality features. The last operation on the categorical features that i made was the conversion of the type of the one that was wrongly setted as numerical.

1.6 Data tranformation: I simply used a log+1 transformation (just like for "SalePrice") for the numeric features with a skewed distribution that allow to attenuate the outlier influence. I thought to use a scale function (that is commented in my file) but at the end I decided to not use it because I chose to use the normalization in the model that i applied to my data.

2. Feature-engineering: In this step I created the dummies variable for all the categorical features and added new variables obtained by the interactions between different features that in my opinion were linked between them, and could explain more the variability of the house's price. Some of the features that i added are about the grade of the various house quality (Garage, exterior, fireplace, ecc..), I did an average of them, and other, that i called scores (e.g. *"PoolScore"*), obtained by multiplying the average grades for the total area, using the size of the quality as weight. An other feature that i added was *"EnoughBath"*, that will assume 1 if there is at least 1 bathroom for room and 0 otherwise. About bathroom I also created two other variables that count the total number of full and half bathroom adding the one of the basement to the one above ground. Another feature that I created was called *"TotFloorSF"* that represent the sum of the square feet of the first and second floor. After I also create a seasonal feature (winter, spring, summer, autumn coded like 1, 2 3, 4) aggregating the month together. Last but not least, looking at the distribution of the area of a

[1] *https://ww2.amstat.org/publications/jse/v19n3/decock.pdf*

porch with respect to my outcome feature (graph in the code), I noticed that the screened, the enclosed and the three season has a similar behaviour, so I choose to use them only added togheter.

3. **Model Learning:** First of all I make a comparison between Lasso and Ridge model looking at the rmse values for each value of alpha, and I found out that Lasso assume lower value, so I started predicting my target variable with it (parameter tuned by LassoCV function), but I obtain a kaggle score above 0.13 althought the mse was 0.01, so I decided to look at another model, Elastic Net. This model use L1 and L2 regularization (Lasso and Ridge) and, since what we saw before, I decided to give a big value to L1 (0.7). Despite the improvement registered in the kaggle score, with the model score computed on python (0.96), i decided to use an Ensemble method merging Elastic Net and Lasso with 0.3 and 0.7 as weight (the one that give my best Kaggle core). Other graphics in the code