Housing Market Analysis Report

“in this land we dream” team

I - Overview

The US housing market’s value reached $36.2 trillion in 2020, with a $2.5 trillion gain, the most in a single year since 2005 [1]. One major reason for this huge gain is that working remotely during the pandemic prompted many buyers to re-evaluate their housing options. That’s why in such a mutable environment it is mandatory to well study house pricing whether from the vendor’s or the customer’s perspective. On one hand, the wrong price is bad marketing for the vendor, on the other hand it is a bad deal for the customer.
In fact, house pricing prediction has been the subject of a lot of studies, as it depends on a lot of variables.
In this report, we discuss the correlation of a lot of factors to house pricing in Boston, relying on the famous Boston Housing Dataset. This dataset contains information collected by the U.S Census Service concerning housing in the area of Boston Mass.It has two prototasks: **nox**, in which the nitrous oxide level is to be predicted; and **price**, in which the median value of a home is to be predicted.
Since we are going to be working on the pricing, our methodology will be the following. We will first explore and clean the dataset, study the correlation of its features mainly with the pricing and their intercorrelation and then visualize the target’s distribution. After that, we will study a few types of regression algorithms and we will compare their performance before and after scaling and then we will apply feature selection on the best performing one.

II - Previous work

Since the original dataset is highly used and issued from a real world case, it was incorporated in the sklearn library datasets. This is why many previous works treated the same topic as demos for those who want to get started with machine learning main concepts.The work of [2] used the same dataset in order to explore logistic regression and the impact of the selected features on improving the metrics of the model. [3] aimed to predict the selling price of a new home by exploring the data to obtain important features and descriptive statistics about the dataset, splitting the data into testing and training subsets, and determining a suitable performance metric for this problem and analyzing performance graphs for a learning algorithm with varying parameters and training set sizes.

III - Data processing

This is the most important phase of the whole process, since no matter how powerful the machine learning algorithm might be, we will surely have poor results if we didn’t prepare our data. In our case, the dataset (boston_corrected.csv) likely has no missing values and only one categorical feature which is “TOWN”. We decided to drop this column since we already have the “TOWNNO” which is identifies each town.

IV - Methodology and implementation:

1- Clean the dataset - Impute missing data (if any)

Our dataset contains 506 rows and the following 20 columns:

- TOWN a factor with levels given by town names
- TOWNNO a numeric vector corresponding to TOWN
- TRACT a numeric vector of tract ID numbers
- LON a numeric vector of tract point longitudes in decimal degrees
- LAT a numeric vector of tract point latitudes in decimal degrees
- MEDV a numeric vector of median values of owner-occupied housing in USD 1000
- CMEDV a numeric vector of corrected median values of owner-occupied housing in USD 1000
- CRIM a numeric vector of per capita crime
- ZN a numeric vector of proportions of residential land zoned for lots over 25000 sq. ft per town (constant for all Boston tracts)
- INDUS a numeric vector of proportions of non-retail business acres per town (constant for all Boston tracts)
- CHAS a factor with levels 1 if tract borders Charles River; 0 otherwise
- NOX a numeric vector of nitric oxides concentration (parts per 10 million) per town
- RM a numeric vector of average numbers of rooms per dwelling
- AGE a numeric vector of proportions of owner-occupied units built prior to 1940
- DIS a numeric vector of weighted distances to five Boston employment centres
- RAD a numeric vector of an index of accessibility to radial highways per town (constant for all Boston tracts)
- TAX a numeric vector full-value property-tax rate per USD 10,000 per town (constant for all Boston tracts)
- PTRATIO a numeric vector of pupil-teacher ratios per town (constant for all Boston tracts)
- B a numeric vector of 1000*(Bk - 0.63)^2 where Bk is the proportion of blacks
- LSTAT a numeric vector of percentage values of lower status population

We chose to work on the newer version of the dataset, the corrected one, with town names and spatial information. It is also augmented with longitude and latitude of the observations and corrected for the censoring error.

There are no missing values. Thus, TOWN and TOWNNO as well as MEDV and CMEDV are considered duplicated, TOWN column will drop after visualization so we could keep only numeric values for regressions. MEDV will also be removed not to mislead our model, since we chose CMEDV as our target.

2- Visualize linear correlations.

2.1 - Interpreting the heatmap

The heat map shows that there are a few features that are highly correlated to Price (MEDVC): RM, LSTAT, PTRATIO, TAX, INDUS, CRIM. Semantically, it is expected, as people look for comfort, security, good education for their children, number of rooms per dwelling, crime rate and pupil-teacher ratios per town are major factors for house pricing. Besides, the social and intellectual status of a neighborhood decides somehow the quality of housing.

2.2 - Visualizing correlation with and without scaling.

The operation of scaling changes raw feature vectors into a representation that is more suitable for the downstream estimators. In general, learning algorithms benefit from standardization of the data set. If some outliers are present in the set, robust scalers or transformers are more appropriate [3].

For now we will show the importance of scaling in some cases of visualization. Later we will study its impact on regression.

Since nox values vary between 0.38 and 0.87 and DIS's between 2.9 and 100 the first plot of their correlation didn't show much. After the standardization of their values, the plot shows that they are inversely correlated.

3- Visualize target distribution

CMEDV follows the normal distribution (with few outliers)centered in 22.52 with a standard deviation of 90.18.

By observing the CMEDV value distribution by town (graph on the notebook), we found that the highest values are mainly located in Cambridge, Arlington, Belmont and Lexington.

4- Regress/classify on all features

4.1- unscaled data:

The following algorithms were applied on all features unscaled: Linear Regression, Polynomial Linear Regression, Random Forest, Lasso, Ridge, Xgboost. Refer to the metrics table in the notebook to view the different performances of these models. As we can see, Ridge and Lasso were the best performing algorithms in this case according to the R2 score.

4.2 - scaled data:

After scaling, Ridge and Lasso are still the most performing algorithms, their R2 scores though increased from 0.77 to 0.80 and from 0.73 to 0.80 respectively.

5- Feature selection:

We used grid search to extract highly correlated features to the MEDVC and we trained two more models by Ridge and Lasso algorithms. The same table in the notebook shows that R2 score increased. Thus, scaling has no more impact on the performance.

V - Results' interpretation

The results show that Lasso and Ridge are the best fit for our dataset. This is explained by the fact that these techniques are used when the data suffers from multicollinearity. They are a modification of linear regression, where Lasso penalizes the model for the sum of absolute values of the weights and Ridge penalizes it for the sum of their squared values.

As for the scaling, it showed a positive impact on the performance of most algorithms, especially Lasso and Ridge, because by not using standardization, the model might require big absolute values of the coefficients.

Finally, we opted for a feature selection by threshold = 0.45. As a result the models were only trained on highly correlated variables to the house pricing (INDUS, RM, TAX, PTRATIO, LSTAT). This had also a positive impact on the models' performance.

VI - Conclusion:

After running several models,tuning their hyperparameters, applying StandardScaler and doing feature selection, we found that the top 2 models (lasso and regide) give us pretty close results (r2_score ≃ 0.81, MSE ≃ 15.5). These results are very satisfying for our dataset. A better performance might be achieved if we had a bigger dataset.

VII - references:

[1] https://www.forbes.com/sites/brendarichardson/2021/01/26/housing-market-gains-more-value-in-2020-than-in-any-year-since-2005/?sh=5b1dbdb94fe0
[2] https://kgptalkie.com/linear-regression-with-python-machine-learlearning-kgp-talkie/
[3] https://github.com/sushantdhumak/Predicting-Boston-Housing-Prices
[4] sklearn documentation.