Aprendizaje computacional aplicado en mercados inmobiliarios

Adam Lemkhanter Boubri

Resumen – La tecnologıa continua transformando nuestro mundo, cada dia hay mas y mejores formas de recolectar informacion, estos datos pueden provenir de usuarios a traves de la interaccion con sistemas informaticos, estadısticas de compra de bienes, estudios de mercado, etc. Esta gran cantidad de informacion que ha surgido en los últimos anñs ha derivado en una gran demanda de profesionales especializados en el sector del Analisis y ciencia de Datos. El Data Science and Engineering Club de la Universidad Autonoma de Barcelona pretende ayudar a suplir esta demanda´ de expertos a partir de una plataforma web en la cual los alumnos tienen a su acceso distintos problemas ya resueltos mediante Machine Learning y Deep Learning. El objetivo principal de este proyecto es estudiar y resolver varios problemas utilizando distintas metodologıas mostrando en todo momento el proceso que se ha seguido, las tecnicas utilizadas y el resultado final con el fin de ampliar el numero de articulos del Data Science Club.

Palabras clave – Aprendizaje automatico, Aprendizaje Computacional, Mercado inmobiliario, Ciencia de Datos, Raspeado web

Abstract–Technology keeps transforming our world, every day there are more and better ways of collecting information, this data may come from users through interaction with computer systems, statistics on user purchase on goods, market studies, etc. This vast amount of information that has emerged in recent years has resulted in a great demand of professionals specialized in the Data Science sector. The Data Science and Engineering Club of the Autonomous University of Barcelona aims to help meet this demand for experts through a web platform in which students have access to different solved problems that use techniques such as Machine Learning and Deep Learning. The main objective of this project is to study and solve various problems using different methodologies, showing at all times the process that has been followed, the techniques used and the final result in order to expand the number of articles in the Data Science Club.

Keywords– Artificial intelligence,Real-estate market,Data Science, Machine Learning, Web scraping.

1 INTRODUCTION

KNOWLEDGE through vast amounts of data is becoming a very popular way to solve complex problems, specially in business related scenarios, where nowadays even the small-est businesses want some sort of machine learning implementation in their business model.

As more companies have joined this trend, the demand of skilled and resourceful data professionals has skyrocketed. This led to the creation of the "Data Science and Engineering Club UAB"[1]. On the one hand, this club aims to develop interest and ease the learning pro-cess for new young students through intelligible, practical already resolved real-life data problems where new pupils can learn and develop their skills. The members can either develop their skills looking through the published posts, or challenge themselves to solve new ones and post their own results and conclusions.

On the other hand, once the members have developed their skills and gained knowledge, this club also helps to put the students in the spotlight for companies that are looking for this set of skills. The published works posted can serve as proof for these companies that you have the knowledge they're looking for and can boost chances in joining a reputable firm.

The main objective of this TFG is to expand the number

• E-mail de contacte: 1531322@uab.cat
• Menció realitzada: Computació
• Treball tutoritzat per: Jordi Gonzalez Sabaté (Ciències de la Computació)
• Curs 2021/22

of contributions in order to provide a greater amount of knowledge through the resolution of different Kaggle datasets, specifically datasets that have been selected to participate in competitions on the platform itself.
In order to provide this knowledge, I've taken a real dataset from a competition in Kaggle, tried to solve it through different techniques, use these techniques in a new dataset created using web scraping to then be published on the club's GitHub. These articles will cover different machine learning techniques and will be sup-ported with the code that has been written.

2 Dataset Selection

The first task to be carried out has been the selection of databases on which to work and apply computation-al learning algorithms. In order to obtain the databases, Kaggle has been used, a web page with an extensive catalogue of datasets. After deliberation with the tutor, it has been decided to choose those datasets that correspond to competitions, which allows obtaining reliable datasets with already established evaluation methods, in this way the result can be compared with the rest of the people who tried to solve the same dataset.

2.1 House price prediction Dataset

For the first regression problem, the "House Prices"dataset [3] has been selected, which is made up of 1460 rows and 79 columns, where these columns represent variables that describe the different aspects that can be taken into account when buying a property. house in the city of Ames, United States. The objective is to be able to predict the price of a house in Ames using a regression model.

2.1.1 Kaggle competition: House Prices - Advanced Regression Techniques

Kaggle competitions are machine learning tasks made by the same company or others like Google or WHO. And in some cases, you can win real money prizes and otherwise leading place in a competition is a great addition to your machine learning engineer resume.
Competitions range in types of problems and complexity, in our case is a getting started prediction competition, these are standard Kaggle competitions where the people access data, build a model, make a submission and then, the results are checked by the hosts of the competition and everyone is attributed a score on the leaderboard
The Ames Housing dataset was compiled by Dean De Cock for use in data science education. It's an incredible alternative for data scientists looking for a modernized and expanded version of the often cited Boston Housing dataset, And now is being used for a competition

About how is gone be evaluated the competition:
The Submissions are evaluated on Root-Mean-Squared-Error (RMSE) between the logarithm of the predicted value and the logarithm of the observed sales price. (Taking logs means that errors in predicting expensive houses and cheap houses will affect the result equally.)

2.2 Fotocasa Barcelona Houses Dataset

This dataset has been created by collecting data from different houses displayed for sale on the real estate web portal Fotocasa[2]. Specifically, data on cases from different neighbourhoods of the City of Barcelona have been collected: Sarria, Sants, Nou Barris, Sant Andreu and Gracia. The attributes to be analyzed are: square meters, number of rooms, number of bathrooms, neighbourhood and finally the objective attribute to predict, the price of the house.
The objective of creating this dataset is to experiment with real and current data how well our machine learning model can work if we change the real estate market.
It contains 3,729 entries that include houses and flats for sale on Fotocasa during the month of January, of which 36% come from Nou Barris, 26 % from Gracia, 23% from Sant Andreu, 9% from Sants and 6 % of Sarria

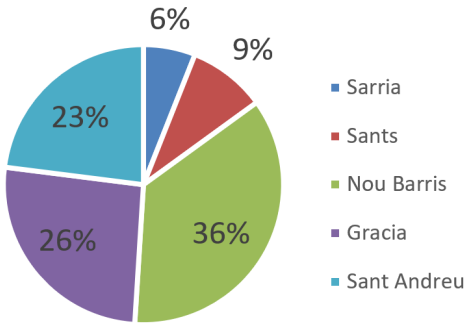

Fig. 1: Houses for sale in Fotocasa in January

3 Blog Posts

Since one of the objectives of this project is to bring forth intelligible solutions to new students and Data Science and Engineering Club members a post for each problem has been uploaded. Therefore, spreading the use and raising the popularity of the blog so more students and companies are aware of it. These posts are written using Python 3.8 in an interactive data science environment called Jupyter Notebook. This environment is used be-cause it is an open-source web application that allows its users to create and share documents that contain live code, equations, visualizations and narrative text. Those characteristics are the ones that make Jupyter Notebook such a great tool for these posts, it allows the displaying of small snippets of code alongside theoretical descriptions or explanations which makes it easier for the reader to understand the theory behind the code that is being executed and allows the reader to easily follow the steps that construct the solution of the problem. These posts also follow a specific format to make it easier to understand, the format is the following:

1. EDA
2. Preprocessing
3. Model Creation
4. Results

3.1 House price prediction

Ask a home buyer to describe their dream house, and they probably won't begin with the height of the basement ceiling or the proximity to an east-west railroad. But this Kaggle competition's dataset proves that much more influences price negotiations than the number of bedrooms or a white-picket fence.
With 79 explanatory variables describing (almost) every aspect of residential homes in Ames, Iowa, this competition challenges to predict the final price of each home.

3.1.1 EDA

For this part of the Exploratory Data Analysis, we are going to first analyse the target attribute "SalePrice"

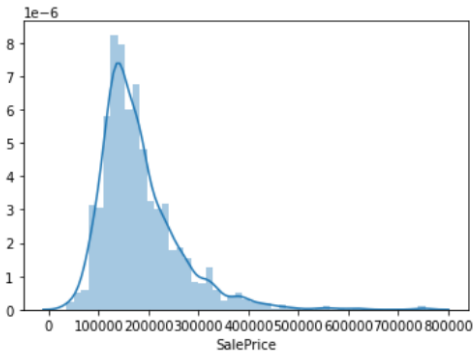

Fig. 2: Distribution of "SalePrice"target variables

As we can observe, our target shows a deviate from the normal distribution, have appreciable positive skewness and shows peakedness.

Consecutively, of the 79 variables given, I have selected the ones that can have the most influence on the price of a house and analysed their correlation with the objective variable:

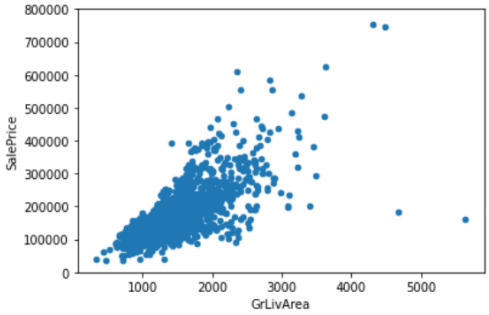

Fig. 3: scatter plot grlivarea/saleprice

GrLivArea: Above grade (ground) living area square feet, seems to show a certain linear relationship and at the same time we can observe the existence of outliers that will be treated later

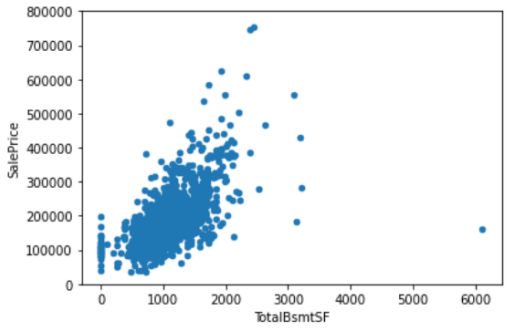

Fig. 4: scatter plot totalbsmtsf/saleprice

TotalBsmtSF: Total square feet of basement area, shows strong relation with "SalePrice", almost exponential relation

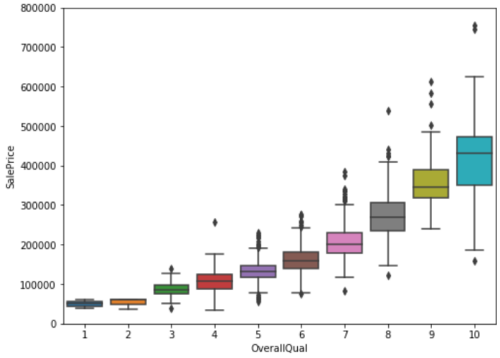

Fig. 5: box plot overallqual/saleprice

OverallQual: Rates the overall material and finish of the house and shows also a linear relation with our target

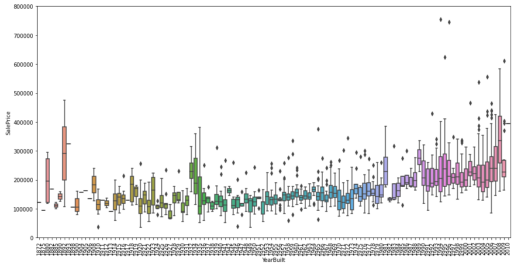

Fig. 6: box plot YearBuilt/saleprice

YearBuilt: Original construction date, does not show a strong relation, and also we should take on count that we don't know if 'SalePrice' is in constant prices. Constant prices try to remove the effect of inflation. If 'SalePrice' is not in constant prices, it should be, so that prices are comparable over the years.

In summary,we can conclude that:

'GrLivArea' and 'TotalBsmtSF' seem to be linearly related with 'SalePrice'. Both relationships are positive, which means that as one variable increases, the other also increases. In the case of 'TotalBsmtSF', we can see that the slope of the linear relationship is particularly high. 'OverallQual' and 'YearBuilt' also seem to be related with 'SalePrice'. The relationship seems to be stronger in the

case of 'OverallQual', where the box plot shows how sales prices increase with the overall quality.
We just analysed four variables, but there are many others that we should analyse. The trick here seems to be the choice of the right features (feature selection) and not the definition of complex relationships between them (feature engineering).

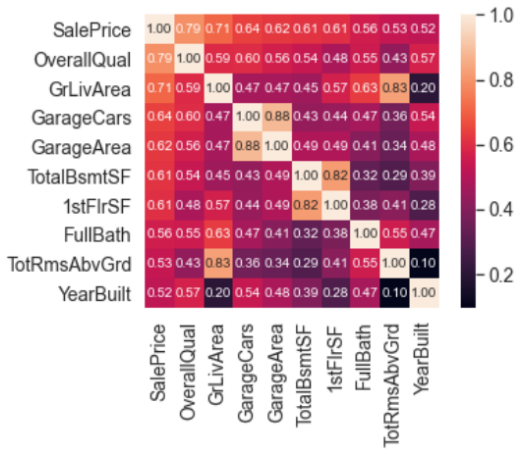

Fig. 7: SalePrice distribution

According to this matrix, these are the variables most correlated with 'SalePrice'. Analysing it:

As I thought, 'OverallQual', 'GrLivArea' and 'TotalBsmtSF' are strongly correlated with 'SalePrice'.
'GarageCars' and 'GarageArea' are also some of the most strongly correlated variables. However, the number of cars that fit into the garage is a consequence of the garage area. 'GarageCars' and 'GarageArea' are describing the same characteristic, just with different metrics. Therefore, we just need one of these variables in our analysis (so maybe we should keep 'GarageCars' since its correlation with 'SalePrice' is higher).
'TotalBsmtSF' and '1stFlrSF' that describes square feet of basement area also seem to be twin brothers

3.1.2 Preprocessing

Taking account, the information analysed in the previous section, now we are going to see the different transformation that I applied to the data to achieve a good prediction of the sale price, trying to take in count how many variables better.
Notice that in the data exploration analysis, I commented that are some outliers and some variables so similar to others that maybe should be removed, but finally I didn't remove them because it gave a worse score
First of all let's see how many missing values contains the dataset:

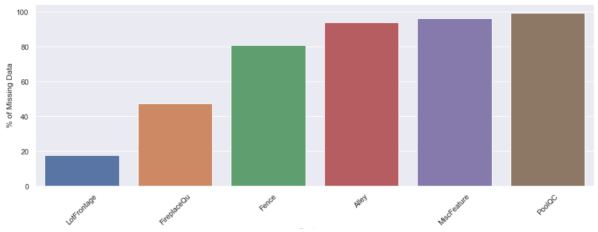

Fig. 8: Features with more than 10 % missing values

Here we clearly see on where I had to focus the work of preprocessing. There are a lot of variables with missing values, and most of the highest percentages of missing values are just describing that does not exist in this attribute, for example, the houses without pool, the variable "PoolQC"that represents Pool quality uses missing values instead of other category for the non-existence of pool.
Removing missing values:
In order to retain most of the data/information of the dataset, I've decides to use the imputation transformers [6] functions of sklearn libraries. Those functions facilitate the task of substitution of an estimated value that is as realistic as possible for a missing or problematic data item.
For numerical columns with missing values, I've decides to use the mean, and for categorical, the most frequented.
In regard to encode the categorical features, I've used the OrdinalEncoder [7], that encode categorical features as an integer array of 0 to NumCategories - 1
And to finalize with the preprocessing, I've applied a scale numeric values through log1p, a method used to normalize the range of independent variables or features of data.

3.1.3 Model Creation

After preprocessing the date I've had to choose one linear regression model of machine learning to predict the sale price of Ames Houses, so instead of chose one and improve it I've decided to take different models and make predictions with them to be sure that I'm choosing the model that makes better predictions with my data, to later improve it by trying different values for its hyperparameters

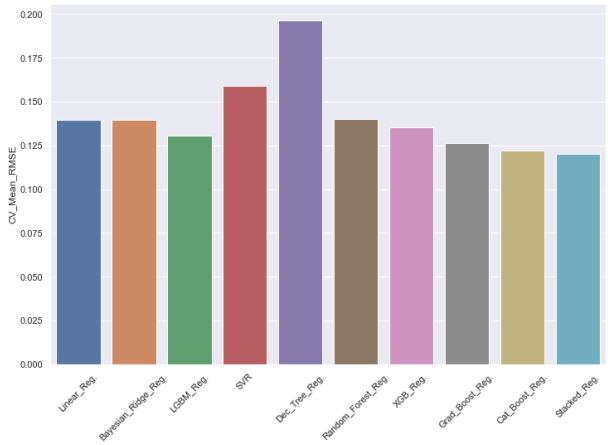

Fig. 9: Regressors

As we can see, seems that Stacked regression is the model that predicts better the sale prices, but after testing different models with similar RMSE, Cat Boost Regressor

is the one which really achieves the best score:

Cat-Boost Regressor

This model is build upon the theory of decision trees and gradient boosting. The main idea of boosting is to sequentially combine many weak models (a model performing slightly better than random chance) and thus through greedy search create a strong competitive predictive model. Because gradient boosting fits the decision trees sequentially, the fitted trees will learn from the mistakes of former trees and hence reduce the errors. This process of adding a new function to existing ones is continued until the selected loss function is no longer minimized.
In the growing procedure of the decision trees, CatBoost does not follow similar gradient boosting models. Instead, CatBoost grows oblivious trees, which means that the trees are grown by imposing the rule that all nodes at the same level, test the same predictor with the same condition, and hence an index of a leaf can be calculated with bitwise operations. The oblivious tree procedure allows for a simple fitting scheme and efficiency on CPUs, while the tree structure operates as a regularization to find an optimal solution and avoid overfitting.

Once selected, the model I've decided to look for the 20 most important variable 10 for our model, It will help to see further insight about the functioning of the algorithm and how and which data it uses most to arrive at the final prediction

To finalize the setting of the model, I've decided to use random grid search to optimize the model. This is a technique where random combinations of the hyperparameters are used to find the best solution for the built model

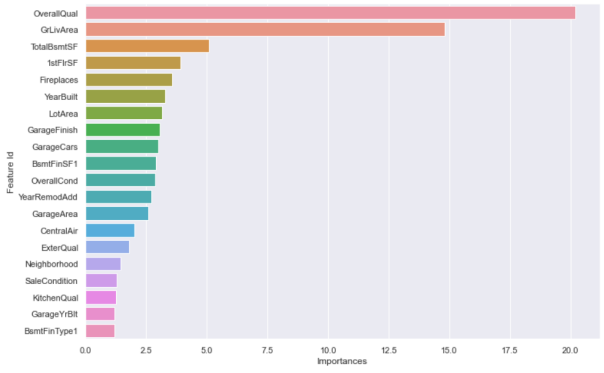

Fig. 10: plot of the 20 most important variables

Observing the plot, we can confirm that the overall material and finish quality seems to be the feature that more influences the price of the house even more than the square meters

3.1.4 Results

After amounts of different predictions, trying to found the hyperparameters that make the Cat Boost predict better and tasting deferents ways of preprocessing the data, finally I've achieved 0.113 mean square error. That lead me to the position of 263 in the leaderboard 11 of the Kaggle competition with an MSE of 0.12027, that means my score is on the top of 5% of the people who participates in this competition(4486 users)

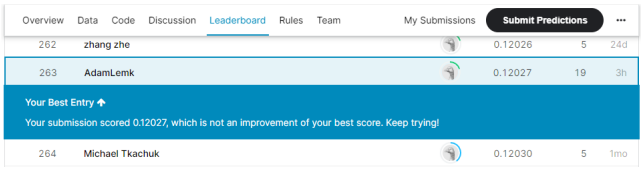

Fig. 11: Submission of House price competition

3.2 Houses for sale in Barcelona

3.2.1 Fotocasa web Scraping EDA

This dataset has been created using web scraping, which is a technique used by software programs to extract information from websites. Usually, these programs simulate a human browsing the World Wide Web either by using the HTTP protocol manually, or by embedding a browser in an application.
In my case, what I have done has been to simulate the navigation of a human being through the Fotocasa real estate portal, and extract the data of the different houses and flats for sale in different neighbourhoods of Barcelona. For this task I have helped myself from libraries such as Selenium [4] and Beautiful Soup[5], the combination of both libraries will do the job of dynamic scraping. Selenium automates web browser interaction from python. Hence, the data rendered by JavaScript links can be made available by automating the button clicks with Selenium and then can be extracted by Beautiful Soup.
Finally, the variables that I could be able to collect are just the Sale price, number of rooms, baths, square meters and the neighbourhood, because the rest of variables were not included on the website as labels and a lot of them were just wrote in the description, which could greatly complicate the automatic data extraction process.

3.2.2 EDA

For this exploration data analysis we will see our target variables and her relation with all the other variables.

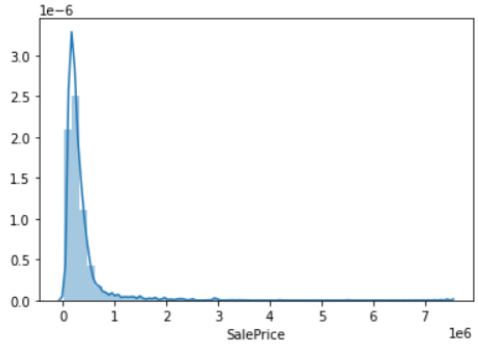

Fig. 12: SalePrice distribution

Observing the histogram, it seems that our target variable shows a deviate from the normal distribution, have huge positive skewness and shows peakedness.

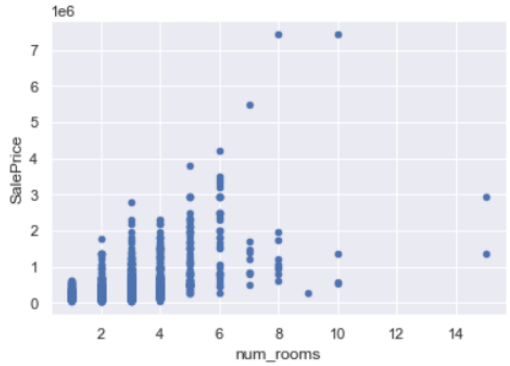

Fig. 13: scatter plot number rooms/sale price

concerning about number of rooms, it seems there is a certain linear correlation, showing that when more rooms more high is the sale price.

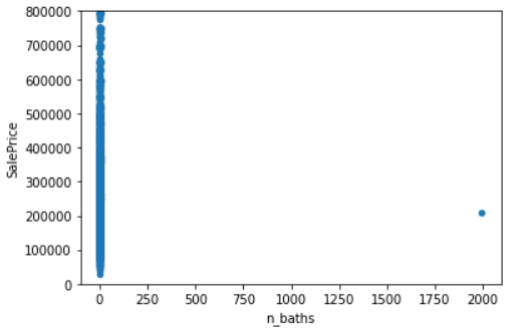

Fig. 14: scatter plot number baths/sale price

According to the number of baths, we can't realize if there is any correlation between sale price and number of baths because there is a big outlier that should be removed.

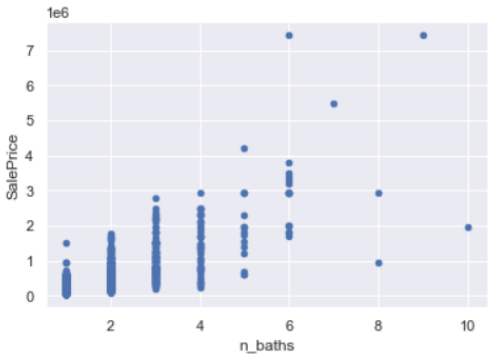

Fig. 15: scatter plot number baths/sale price

Once removed the outlier, finally we can see some type of linear relation with the sale price.

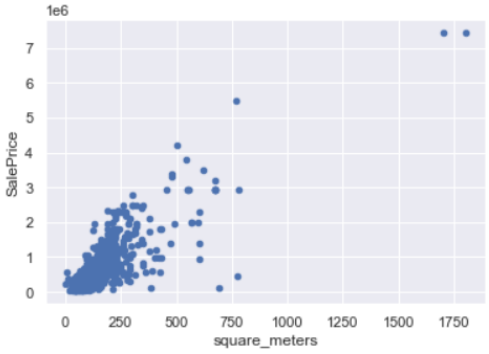

Fig. 16: scatter plot square meters/sale price

Square meters seem to have linear relation with the sale price, but there are also some outliers, let's remove them and see if the changes.

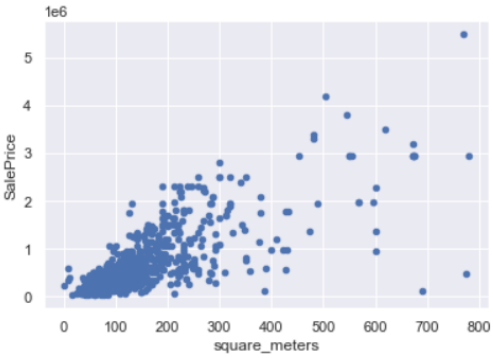

Fig. 17: Prices per each neighbourhood

After removing the outliers the linear relation seems stronger with sale prices.

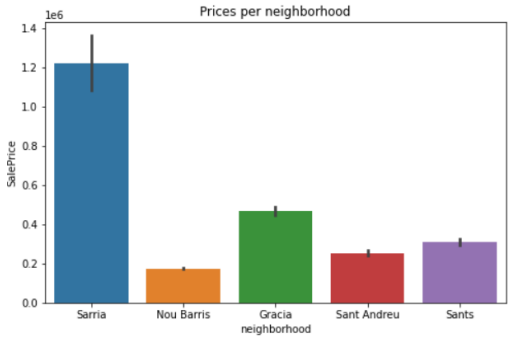

Fig. 18: Prices per each neighbourhood

Here we can see the sale prices for each neighbourhood, and observe clearly that Sarria is the richest and Nou Barris the poorest neighbourhoods. In addition, if we compare this plot with this figure 1, we can see that the neighbourhood with fewer houses for sale has more expensive prices and the same vice versa, which in some way could demonstrate the law of supply and demand

Fig. 19: SalePrice correlation matrix

Finally, this correlation matrix verifies the previously found linear relation, where it seems that in matters of numerical attributes, the square meters and the number of rooms are the variables that most influence the sale price.

3.2.3 Preprocessing

For the preprocessing of the data, the first thing I have done has been to look for the null values, which come from empty fields from the web scrapping process. Given that less than 5% were nulls, I've decided to remove them from the data set.
According to outliers and as we saw in the last point, it has been deleted 3 entry, one from number of baths and the other for square meters. Then I've applied a scale of numeric values through log1p, a method used to normalize the range of independent variables or features of data And to end, I've taken the categorical neighbourhood variable and converted to numerical using the pandas.factorize function.

3.2.4 Model Creation

As I've commented previously, the objective of this dataset is try to use the same machine learning model used for the real-estate market of Ames(Iowa) to the actual real-estate market of Barcelona, and since the model that gave me the best score with the Ames dataset was the Cat-Boost Regressor, I have applied this same model to predict the price of houses in Barcelona.

But as we can see in the plot below, it gives us a very high mean square error, so I thought it would be better to try another model, but after tried with other models and as we can see in the figure below, although the score is not good, this model is the one that gives the best score among all other tested models.

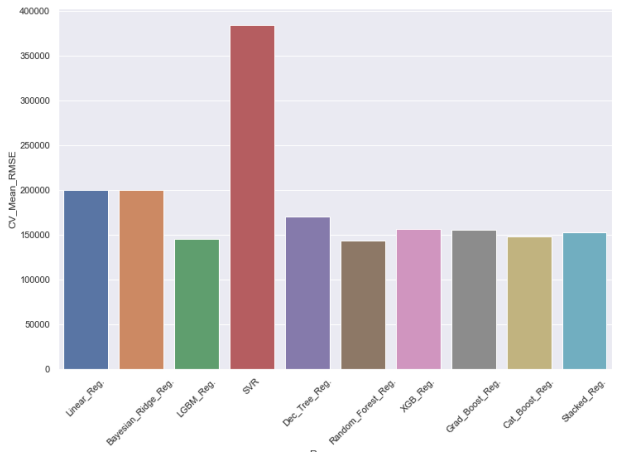

Fig. 20: Regressors

As I did with the Ames houses dataset, the model I've decided to look for the 20 most important variable 10 for our model, because it allows to see further insight about the functioning of the algorithm and how and which data it uses most to arrive at the final prediction

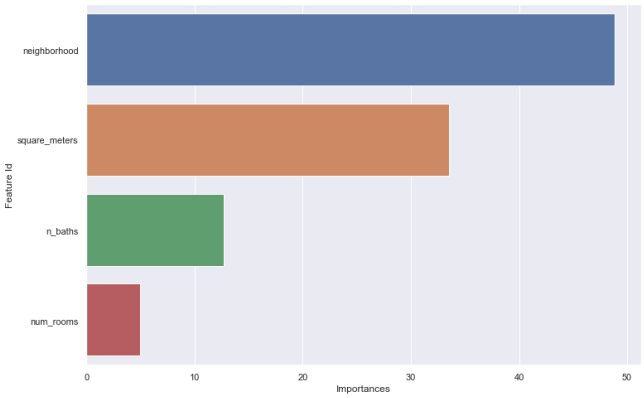

Fig. 21: Feature importances plot

As is commonly known and as is shown in the plot, the neighbourhood the feature that most influences the prices of a house followed by square meters, but also notice that seems more important for the cat boost model the number of baths than the number of rooms.

3.2.5 Results

Despite having less data and features than the Ames dataset, it seems that the Cat Boost regressor works so good in a different real state market and even with fewer data and features, achieving then a 0.29 of mean square error and r2 of 0.818

4 Conclusions

Finally, to sum up, I've taken part in a Kaggle competition of predicting the sale price of houses in Ames. To do that, I had to find the machine learning model that works better for this dataset. After that, I achieved a place in 5% of the people in the leaderboard of the competition. For then test this same machine learning model in a new dataset of a different real-estate market, that has been collected through

web scrapping, and as a final result seems that the model Cat Boost Regressors is good for predicting sale prices of houses even in different real-estate markets

ACKNOWLEDGEMENTS

I would like to thank the following people, without whom, I would not have been able to complete this research, and without whom I would not have made it through my final subjects. First of all, my tutor, Jordi Gonzalez, for the trust and guidance he has given me during the last months and for giving me the opportunity to develop this project. I would also like to thank my internship tutor Alfonso Fornell who has been very understanding and flexible with my working hours, allowing me to spend more time on this project.

And last but not least to Jordi Pons, who despite the Virtual Campus crash, he has always kept us up to date with emails, notices and updated information on the next steps of the TFG

REFERÈNCIES

[1] DataUAB Blog. [Online]. Available: https://datauab.github.io/.

[2] Fotocasa. [Online]. Available: https://www.fotocasa.es/es/.

[3] House Prices - Advanced Regression Techniques [Online]. Available: https://www.kaggle.com/c/house-prices-advanced-regression-techniques/overview.

[4] Selenium Documentation [Online]. Available: https://www.selenium.dev/documentation/.

[5] Beautiful Soup Documentation [Online]. Available: https://www.crummy.com/software/BeautifulSoup/bs4/doc/.

[6] sklearn.impute.SimpleImputer [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.impute.SimpleImputer.html.

[7] sklearn.preprocessing.OrdinalEncoder [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.OrdinalEncoder.html

[8] randomized search [Online]. Available: https://catboost.ai/en/docs/concepts/python-reference_catboost_randomized_search

APPENDIX

A.1 Hyperparameters selection

The method used to find the best hyperparameters to optimize the Cat-Boost regressor is the random grid search [8], this is a technique used to find the optimal hyperparameters of a model which results in the most 'accurate' predictions. Below 10 we can see hoy acts the model with the default parameters and in the next figure 23 when the best hyperparameters are setted.

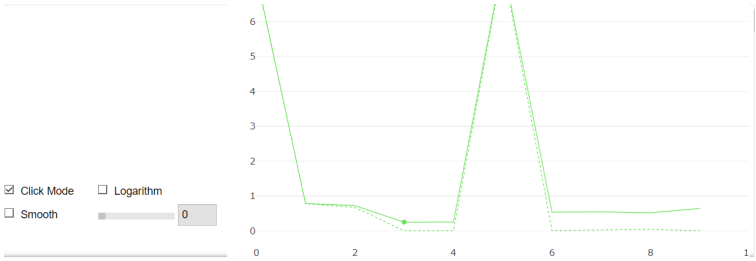

Fig. 22: Cat-Boost Regressor with default parameters

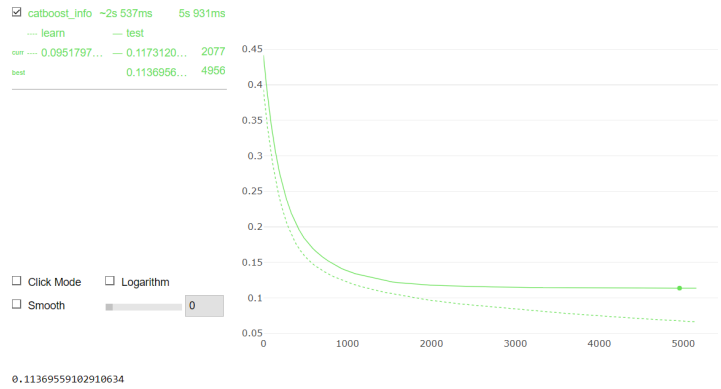

Fig. 23: Cat-Boost Regressor with optimized parameters