Madrid, Dec 24th 2018

Master in Business Analytics and Big Data

Machine Learning I

Section O-2

Group H

Data Gathering, Ingestion and Processing

Madrid Real Estate Project

Data ingestion and processing document:

For this project we didn’t want to use the second-hand data, and using the scraper in Fotocasa yielded very low quality of data. This is why we wanted to leverage current APIs to get the data needed.

We obtained credentials to use Idealista’s API for research purposes so we generated a miner (python code) that would request the information that we needed and write it into a CSV file. The code in question can be found in the Annex.

This code yielded 4450 rows of information and then we proceeded to clean it and transform it to fit our model. The steps we took were:

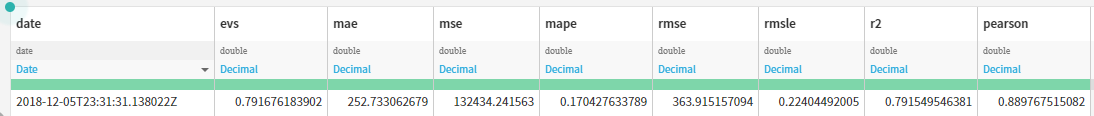
1. Uploaded data into Dataiku
2. Removed about 45 rows that had a problem in the CSV or delimitation and it had shifted.
3. We replaced some values in the floor column:
   1. “en” was though of entrance so we mapped 0 to it
   2. “bj” was though of bajo so we mapped -1 to it
   3. “ss” and “st” we thought of sotano so we mapped -1 to them
   4. Empty floor values were replaced with 0
4. We removed “col\_34” as it was an issue regarding point #2
5. We un-nested detailedType into typology and subtypology and deleted the original column
6. We imputer False into missing values for “hasLift”
7. We unnested the “suggestedText” and “hasParkingspace” and deleted the original columns
8. We imputed the missing values on status with “good” as it was by far the most common
9. We created a geopoint column which was useful to map the places in Dataiku
10. We eliminated apartments less than 400 Euros and more than 4030 Euros as they were outliers. The cheapest option sometimes meant that it was a room in an apartment, so we took those off as it would throw off the model (high number of rooms but cheap price but for only part of the apartment)

This gave us our final dataset for using to the model. However not every variable was still useful for the model. In this case we used Lasso regression which is linear regression with L1 regularization optimized by cross validation.

While creating the model we used the following variables:

Bathrooms, detailedType\_typology, district, exterior, floor, hasLift, neighborhood, hasParkingSpace, propertyType, rooms and size. We decided to take municipality off as it increased the score of the model.

After the model was trained, we got a score of 0.799 R squared and a Mean absolute error of 255. We then tested against a test set the model had never seen and got the following results.



Which means our model is not overfit as the model scores are close between the train and test sets.

We then proceed to score the original dataset with our new model to see what the predicted price would be and then created a couple of new features regarding the results

1. Benefits: This is the difference between the listed price and the predicted price of the listing. This is basically using the residual to find if the listing is over or underpriced. This quantifies how much that difference is
2. GoodDeal: This is a Boolean variable (0,1) that simply denotes if the listing is a good deal or not, the basis is if the benefit is positive it’s a good deal, else it’s a bad deal. This will help the user filter apartments by underpricing
3. Discount%: This is simply (predicted\_price - listed\_price) / predicted\_price to show the % of discount from that listing from the predicted price.

After these columns are created, we can the export the data to then visualize it using Tableau or other visualization software