Data Science Project: To what extend can we predict the average price per square-meter of Paris apartment using only external factors ?

Hortense Vallat, Stellan Wea

January 2024

Contents

1 Introduction

Ever wondered which factors influence the price of a Parisian apartment? While current approaches to pricing apartments have traditionally centered around tangible features like square footage and room count, the impact of external variables has, for the most parts, been relegated to the periphery. The aim of our project is to clarify the relative significance of various external factors on the pricing structure of Parisian real estate. Specifically, we seek to investigate how elements such as transportation infrastructure, proximity to essential amenities and schools, the presence of green spaces, and the distance to the central hub of Paris collectively contribute to the comprehensive valuation of an apartment.

To what extend can we predict the average price per square-meter of Paris apartment using only external factors ?

2 Global data management

2.1 Data Collection

Data relative to apartments comes from the API CEREMA. This comprehensive source provides details on apartment prices, square footage, and the sale date. While specific geolocation data for apartments is not available, we leverage information about the parcel[1]. By associating the identification of the parcels with a geojson on parcels from Data Gouv, we derive a single point for each apartment, represented by the center of the polygon and it enables us to have a precise idea of the housing localisation. Our focus spans all apartment sales (called "mutations" in French) in Paris since 2020, encompassing a substantial dataset of over 100,000 transactions. We store the results in a CSV called *mutations.csv* and a HTML to have a map representation of sales.

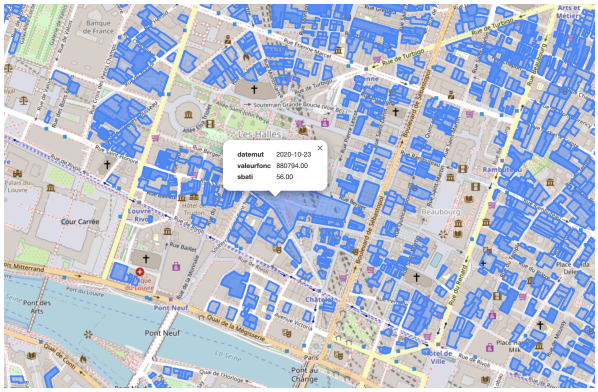

Figure 1: Paris mutations html

The rest of our data are CSVs available on open sources. Specifically:

- Data Gouv: shops (food market, fashion shops, culture shops, catering)
- Opendata Paris: parks and velib stations
- Data Ile de France: subway stations, touristic sites, cinema, live performance sites and schools

It is important to note than while the sales are only within the department of Paris, the rest of our variables are in Ile-de-France scope. We keep 4 departments in our datasets, surrounding Paris (75, 92, 93 and 94).

[1]A parcel is a contiguous piece of land of uniform nature, marked out for ownership purposes, and belonging to a single owner.

2.2 Data processing and distance computation

In our project, the first challenge we encountered involved calculating distances between apartment parcels and our various variables. Considering that we have collected data from over a hundred thousand apartment sales since 2020, a straightforward computation of distances for each apartment with all associated variables would be excessively time-consuming. To illustrate, let's consider the case of Velib stations. Paris has over 1,400 Velib stations, and if we were to loop through each apartment sale and each Velib station, the program would iterate over 100 million times whereas we only needed to know how many were close or not. Initially, our approach involved calculating distances between variables within the same district. However, this method presented a potential issue for apartments situated at the extremities of an Parisian district. This is also the reason why we chose to keep some information about cities surrounding Paris.

To address this computational challenge, we implemented certain conditions. Specifically, a Velib station would only be considered if its latitude and longitude were within a distance of less than 1 km, enabling us to compute the closest station (within a square of 1 km). When no suitable stations were found within the defined perimeter, we set the distance to 1400 meters. We generalized the same calculation for the rest of our variables.

Then, we analyze the average minimal distance to each variable and created a new variable called *nb_variable*. Taking the example of Velib again, the variable nb_station_velib represents the number of Velib stations within the perimeter defined by the calculated average distance. This approach allows us to streamline the computation process and extract meaningful insights from the vast dataset, contributing to a more efficient and targeted analysis of our project objectives. Although the initial code execution time was relatively long, by optimizing vectorized operations, we have significantly reduced its runtime.

It's worth noticing that two approaches of distance can be used : how many do I have around me or where is my closest. Consequently, we chose to build 3 datasets, the first one considers for instance the number of Velib stations around a parcel, the second one takes into account the minimal distance from a parcel to a station and the third one gathers the two priors.

3 Descriptive Analysis

3.1 Summary of the variables

Table 1 provides details on all explanatory variables of our model. Our final dataset is consequently composed of 114 361 rows which correspond to apartment sales and 30 explanatory variables. The target variable is as a result the price per square-meters.

Variable Type	Variable Name	Description
Apartment	libtypbien	One apartment, two apartments or undetermined apartment
Arrondissment	arrondissment	arrondissement of the sales
School	nb_mat_elem	Number of primary or elementary school in perimeter
	nb_mat_elem_prive	Number of private primary or elementary schools in perimeter
	nb_mat_elem_public	Number of public primary or elementary schools in perimeter
	min_dist_mat_elem	Distance (m) to the closest school
	nb_coll_lycee	Number of middle schools and high-schools in perimeter
	nb_coll_lycee_prive	Number of private middle school and high-schools in perimeter
	nb_coll_lycee_public	Number of public middle school and high-schools in perimeter
	min_dist_coll_lycee	Distance (m) to the closest school
Transport	nb_station_velib	Number of Velib stations in perimeter
	nb_velib	Sum of Velib capacity stations in perimeter
	min_dist_velib	Distance (m) to the closest Velib station
	nb_gare_unique	Number of metro/RER stations in perimeter
	nb_lignes_unique	Number of unique lines of metro/RER in perimeter
	min_dist_gare	Distance (m) to the closest metro/RER station
Cinema and Show	nb_cine	Number of cinemas in perimeter
	nb_cine_plus500	Number of cinemas with more than 500 seats in perimeter
	min_dist_cine	Distance (m) to the closest cinema
	nb_spectacle	Number of live performance sites in perimeter
	min_dist_spectacle	Distance (m) to the closest live performance sites
Parks	nb_parks	Number of parks in perimeter
	min_dist_parks	Distance (m) to the closest parks
Touristic sites	nb_site_tour	Number of touristic sites in perimeter
	min_dist_site_tour	Distance (m) to the closest touristic site
Shops	nb_food_market	Number of food markets in perimeter (supermarket, wholesale, bakery, beverages...)
	nb_fashion_shop	Number of fashion shops in perimeter (bag, boutique, clothes, fabric, shoes...)
	nb_culture_shop	Number of culture shops in perimeter (art, camera, collector, music, video games...)
	nb_catering	Number of catering in perimeter (bar, cafe, fast food, food court, pub, restaurant)
Center of Paris	distance_to_center	Distance (m) to center
Target	pricem2	Price per square-meters

Table 1: Variables explanation

3.2 Statistical study of variables

Now that we succeeded to scrap and gather all information computed with the mutations locations, we lead a statistical study to get familiar with our explanatory variables and the target.

Looking at the *prixm2* target in Figure 2, we observe extreme values for the last quantile. The dataset contains apartments with a very high fare while most housings have an average price below 15,000€ per square-meters. We consequently change our visualisation to a histogram displayed in Figure 3. To create that graph, we removed sales where the price was above €20,000, considering them as outliers. The target almost follows a gaussian distribution with a mean of 10,000€. Still, we noticed some outliers at the left part of the gaussian curve where the price is very low.

Therefore we decided to remove from our study outliers (left and right) because they might damage the model performance. Housings with a price between 6,000€ and 15,000€ stick more with reality prices so we keep this scope. Outliers represented 15% of our dataset.

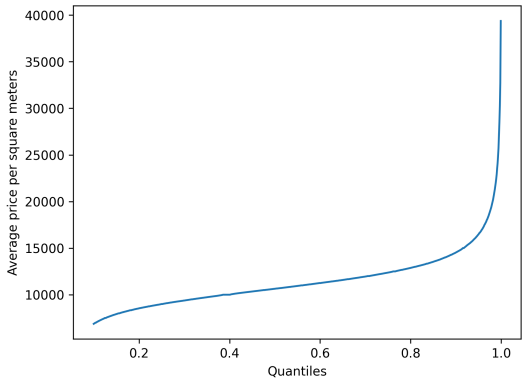

Figure 2: Average retail price per quantile of sales

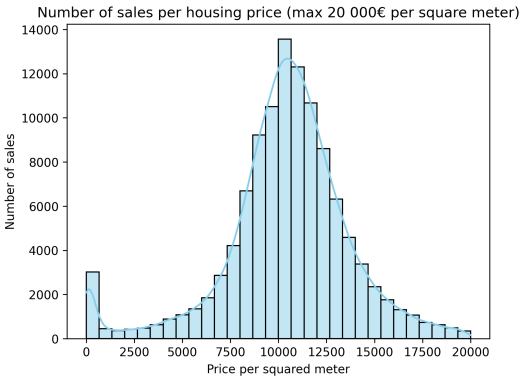

Figure 3: Number of sales per housing price

Then looking at explanatory variables, we noticed that most of the numeric variables are left-skewed. These variables are i a way positively correlated with neighborhood's activity since some new restaurants make a place more jovial. However, the impact of this activity on prices is not so clear though. For instance, some parents may be indifferent in their distance to private schools since they prefer public education while other would value it more.

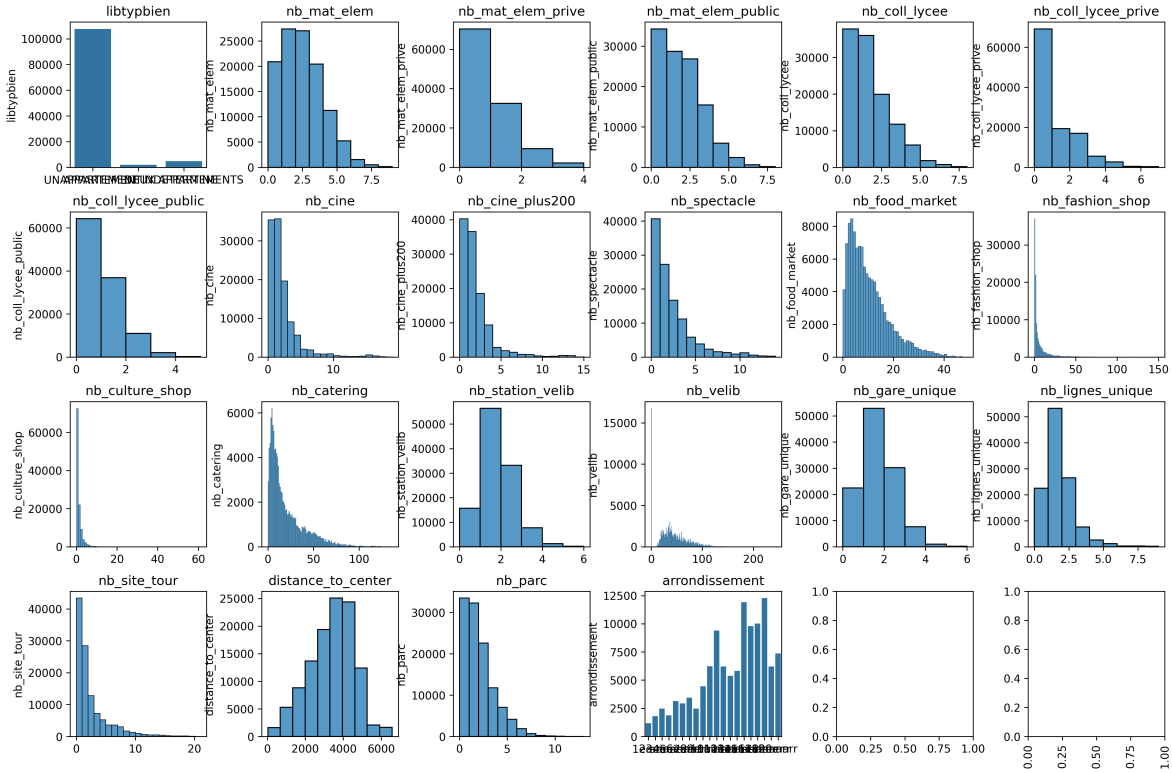

Figure 4: Description of the explanatory variables

In our study, we are interested in correlations between our variables as we know that correlations can damage the model prediction. We thus created the correlation matrix in Figure 5. We consequently see that the number of Velib stations and the bike capacity per Velib stations nearby lead to a remarkable positive correlation. All outcomes from the correlation matrix follow a business sense and we don't detect aberrant values between our features.One interesting finding is that distance to center

of Paris is negatively correlated with all variables but schools. We can conclude that schools are not in the same areas than places with many activities, in the center of Paris.

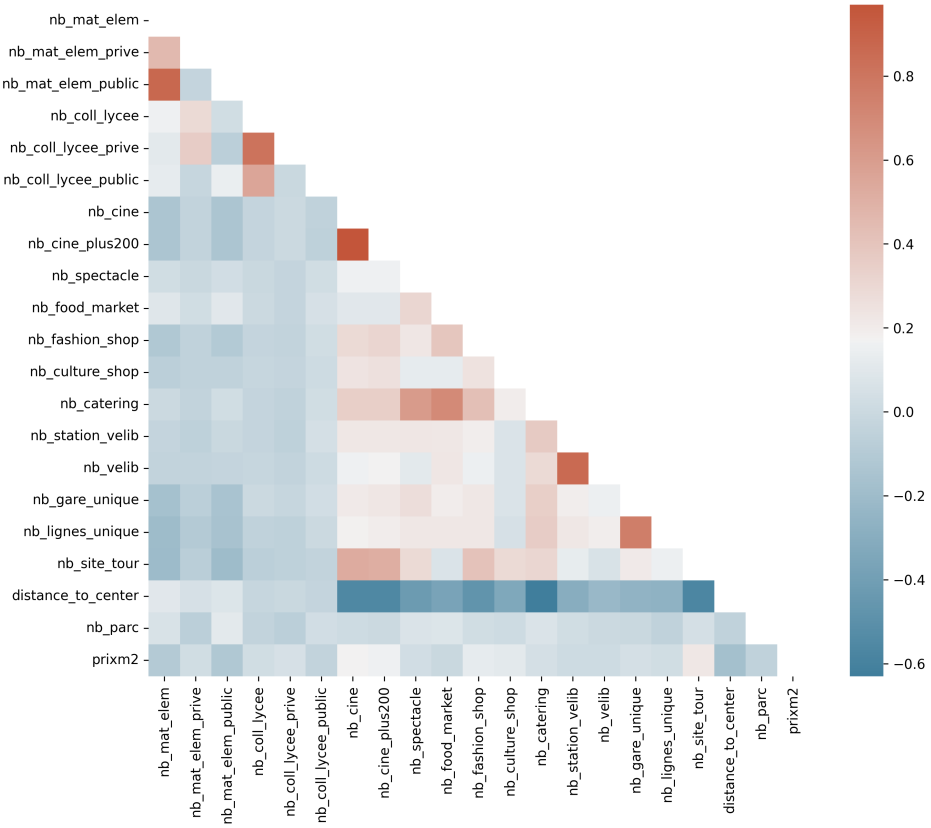

Figure 5: Correlation matrix of the explanatory variables

4 Predictions

4.1 Models comparison on a subsample

The final goal of this study is to obtain a tool able to determine the average real estate price of apartments based on the variables described above. Two of our explanatory variables are categorical (*arrondissement* and *libtypbien*) so we decided to turn them into dummies with One-Hot-Encoding. As for the 28 other numeric features, they are all scaled in order to balance each variable importance if they don't share the same variance or are calculated with a different unit.

The first challenge we faced on this step is the prediction of extreme values. Indeed, apartments with extremely high or low prices damage the model performance. Therefore, we decide to conduct several models on a subsample of apartments (10,000 rows) with prices per square meters between €6,000 and €15,000, as determined above. We opt for linear models powered by **Lasso** and **Ridge** regularization and for a **Random Forest** model. The reason we choose these models is that they can manage correlation between variables by reducing the effect of certain features. Aligned to our interpretability wish, we decided to only try a Random Forest as black box algorithm

We partitioned our sub-sample between a train (80%) and a test set and get the results of the R2 and the Mean Square Error on the test set. Despite using hyper parameters tuning, all our models have quite bad performances since none reach more than 23% of R2 on the test sample. Comparing the results we obtained in our sub-sample (see Table 2), we decided to run a LASSO model on the entire dataset.

Model	Variables	R2	MSE
LASSO	Closest distance	18.92%	1746
	Number in perimeter	20.24%	1754
	All variables	22.28%	1459
RIDGE	Closest distance	18.95%	1745
	Number in perimeter	20.24%	1754
	All variables	22.27%	1459
Random Forest	Closest distance	17.84%	1757
	Number in perimeter	16.52%	1795
	All variables	17%	1507

Table 2: Results comparison on sub-sample (prices between 6 000 and 15 000 per square-meters)

4.2 LASSO model on the entire dataset

In the python file "Final prediction", we run three times the LASSO model:

- One for the values of prixm2 between €6k and €15k
- One for the extremely low values of prixm2
- And one for the extremely large values of prixm2

For each model we partitioned our data into one train sample containing 80% of the data and a test set.

The best model for the non-extreme values has an R2 of 21% and a MSE equal to 1715. The associated best parameter alpha is 0.951. This quite high parameter is not really surprising in our correlated model. The model promotes the selection of important variables by driving certain coefficients to become exactly zero. The graph in Figure 5, shows the predictions of our model compared to the real values.

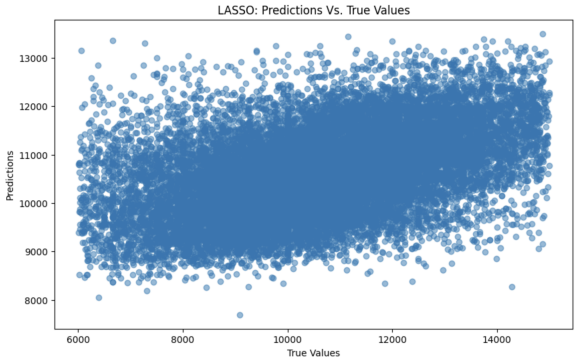

Figure 6: Result of LASSO (prices between 6 000 and 15 000 per square-meters)

For the extreme values, the model is worse than before, reaching an R2 of 11% for low values and less than 4% for high values. We understand the lower performances as the more expensive the apartment, the more it relies on its internal characteristics. Furthermore, we can suppose that rich people buying these flats do not matter so much about the transport location for instance.

5 Conclusion: Evaluation of the results

The main challenge we faced during this project is the data scrapping and processing. Indeed, we manipulated many massive datasets and had to compute some distances based on geolocations points and polygons. This step implied optimizing, as much as possible, vectorized operations. We finally succeeded to find a solution to reduce the calculation times and after some checks, we were delighted to see that our outcomes in terms of features were conclusive and reliable.

Yet, we were quite disappointed when we firstly launched some models because the performance are very low despite our data processing, feature selection, and ML model attempts. The first reason explaining the low performance of our model is that this study is focused on the eternal factors of the apartments. We don't have any ideas on when was the latest wall renovation, equipment such as garden or swimming pool, floor... This lack of information on the internal housing represents a loss of features and it can prevent the model from being more accurate. Our analysis is therefore a good complement to the traditional data science projects whose goal is to determine the price from the flat characteristics.

Another approach of this topic could be to design a recommendation model to give advice on the Parisian area where the user wants to live depending on the input provided regarding his budget, tastes and life priorities. Nevertheless, it would imply to firstly improve our knowledge of Paris through a more consistent database that gathers in one hand details on the flat and on the other hand on the housing environment.

6 Sources

Apartment sales

- https://apidf-preprod.cerema.fr/swagger/

Other variables

- https://cadastre.data.gouv.fr/
- https://opendata.paris.fr/pages/home/
- https://data.iledefrance-mobilites.fr/pages/home/

Link to our GitHub Repository

- https://github.com/stellanweadau/DataScienceProject