Descriptive and predictive analysis (Formatted and Exploitation Zones)

Big Data Management

Abstract This project, for the Big Data Management course, integrates and analyzes data related to Barcelona's housing market and economy. Using Apache Spark, we cleaned and reconciled multiple datasets in the Formatted Zone, ensuring efficient processing of large volumes of data. The unified data was stored in MongoDB, leveraging its scalability and flexible schema for rapid retrieval and analysis. This setup facilitated the calculation of key performance indicators (KPIs), training of machine learning models, and real-time predictions, supporting both descriptive and predictive analytics objectives.

Keywords: *Formatted Zone, Explotation Zone, Barcelona Housing Price,*

1 Introduction

This project aims to do a descriptive and a predictive analysis using real data from Idealista, on apartments and houses from the different neighborhoods of the city, supplemented by social and economic data sourced from OpenData Barcelona. The 3 sources are the following:

- **Idealista Source:** Data in multiple Parquet Files
- **OpenData Barcelona Source:**
  - Income per neighborhood. (JSON file)
  - Votes per party per neighborhood. (CSV file)
- **Lookup Tables**
  - Lookup for Rent (per neighborhood), Rent (per district)
  - Lookup for Income (per neighborhood) and Income (per district). Also used for the 3rd source `Elections` because they are from the same source therefore with the same data governance.

Here in Figure 1 we can take a look at the processes and environments we used in the developing phase of the project.

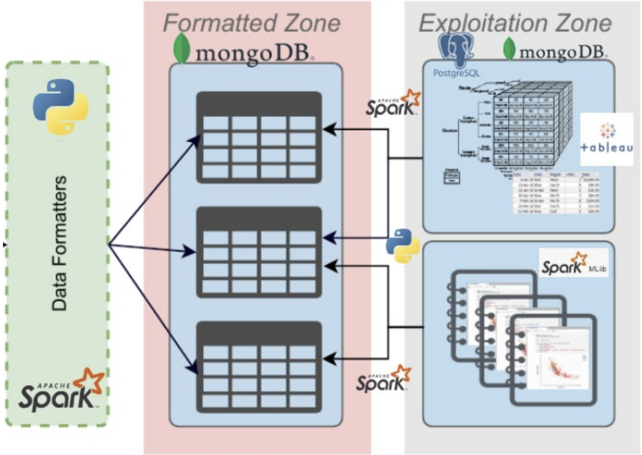


Figure 1: Graphic BPMN flow for the different zones.

2 Formatting Zone

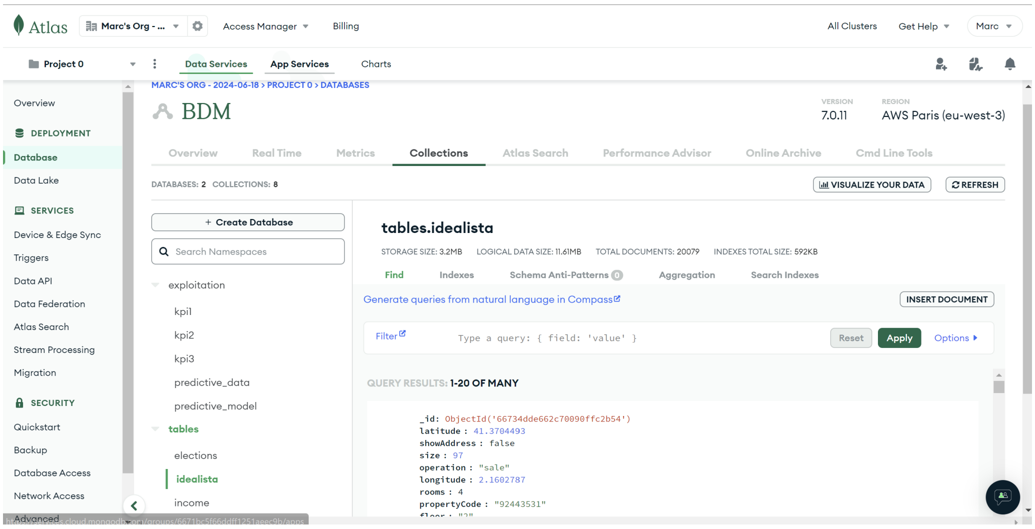
In the Formatting Zone we dealt with data integration and reconciliation. Our approach was to join the corresponding main tables (idealista, elections, income) with his respective lookup table, in order to get 3 main ground of truth sources for our predictive and descriptive analysis that is being done in the exploitation zone.

Before start talking about the relatione behind our code it is important to say that all our formatted zone was saved in MongoDB, because its document-oriented storage model is highly scalable and flexible, allowing for easy handling of varied data structures without the need for complex schema migrations, this is very useful in our case as the schemas of the different sources can evolve over time. Also, the ability to store data in a JSON-like format makes it an efficient choice for storing the diverse and nested data structures resulting from the joins between datasets. This enables quick retrieval of complex documents, facilitating rapid descriptive analysis. Joining with the next component of our formatted zone, MongoDB integrates well with Big Data tools like Apache Spark, enabling seamless data processing and storage workflows. This integration ensures that data can be processed in Spark and then efficiently stored in MongoDB for subsequent analysis and retrieval.

As said in the last paragraph we used Apache Spark for the Data Integration process. It was chosen for this project due to its capabilities in handling large datasets efficiently and performing complex data transformations. Spark's ability to distribute data processing tasks across multiple nodes allows for efficient handling of large datasets. This capability was essential for processing the extensive data related to Barcelona's housing and economic, elective indicators. Spark's in-memory computing capability significantly speeds up data processing tasks by keeping intermediate data in memory rather than writing it to disk. This feature was important for the iterative operations required in joining and transforming the datasets. For this project, Python was used to leverage Spark's DataFrame API, which simplifies the process of data manipulation and transformation. Finally, as said earlier the integration with MongoDB was particularly important for storing the resulting tables after performing the necessary joins and transformations.

Our reconciliation process of the data to feed the exploitation zone with useful data for the different predictive and descriptive analyses was crucial so they could give us interesting information and insights. The first step was to load the different datasets into Spark DataFrames. Each dataset is stored in multiple files, which are read into separate DataFrames. For each type of dataset (income, elections, Idealista), individual DataFrames are concatenated into a single DataFrame. This step ensures that all data points from the multiple files are combined into a unified dataset for further processing. The core of the reconciliation process involves joining the datasets with lookup tables to standardize district and neighbourhood names. This ensures consistency across different datasets, facilitating accurate analysis. For all of the three DataFrames we get the join with the respective lookup table in order to get the neighbourhood and district correctly. For some of the neighbourhoods and districts typos were corrected such as punctuation.

For the idealista dataset some columns contained nested data structures, so these columns were flattened to simplify the DataFrame structure and facilitate easier analysis. Finally, for this dataset, for being highly diverse, it required handling differences in schema across files. This was achieved by ensuring all DataFrames have a consistent set of columns, adding missing columns with null values where necessary. In Figure 2 it can be seen the final MongoDB tables are uploaded after the whole process taking into account that the collections in the Exploitation Zone are uploaded in further steps. The database 'tables' is the one that has the 3 data sources generated by the Formatted Zone.



DATABASES: 2 COLLECTIONS: 8

- exploitation
  - kpi1
  - kpi2
  - kpi3
  - predictive_data
  - predictive_model
- tables
  - elections
  - idealista
  - income

tables.idealista

STORAGE SIZE: 3.2MB LOGICAL DATA SIZE: 11.61MB TOTAL DOCUMENTS: 20079 INDEXES TOTAL SIZE: 592KB

QUERY RESULTS: 1-20 OF MANY

```
_id: ObjectId('66734dde662c70090ffc2b54')
latitude : 41.3704493
showAddress : false
size : 97
operation : "sale"
longitude : 2.1602787
rooms : 4
propertyCode : "92443531"
floor : "2"
```

Figure 2: MongoDB Atlas BDM project databases and their collections

3 Exploitation Zone

For the storage in the exploitation zone, we considered several storage systems, such as PostgreSQL. Ultimately, we chose **MongoDB** due to its superior performance in handling large datasets. The exploitation zone often deals with substantial volumes of data generated for modeling and querying purposes, making MongoDB an ideal choice. Its ability to efficiently manage and process extensive data sets ensures that our storage requirements are met with optimal performance and reliability.

In the context provided, the Exploitation Zone involves using the integrated and cleaned data to perform advanced analytics and derive actionable insights. Spark was chosen for KPI calculations due to its robust capability in efficiently processing large volumes of data through parallel processing. We provided the following meaningfull KPIs for our data context.

3.1 Descriptive analysis KPIs

We performed several descriptive indicators for the wealth-social economy of the district / neighbourhood such as:

- Average housing price per neighbourhood.
- Average housing price per district.
- Percentage of elections participation per district.
- Correlation between housing prices and election participation.
- Correlation between housing price and family income per district.
- Rank of the top parties per district.

In the apendix we display the visualization of some extra ones that we decided that were less important, informative or visual.

Our descriptive and predictive analyses were conducted using notebooks, which proved instrumental due to their interactive nature and seamless integration of code with documentation. This environment enabled us to iteratively explore data, visualize trends, and generate insights efficiently. By combining code cells with markdown for explanations, notebooks facilitated clear and comprehensive documentation of our analysis process, enhancing our understanding of complex data patterns and supporting collaborative efforts. The data was read from MongoDB and the computed KPIs and outputs were also loaded.

3.1.1 Average housing prices per district

To visualize this KPI, we opted for a map as the most effective graphic. To create it, we needed to integrate an additional data source that includes the polygons for each district [1]. This process was straightforward given that Barcelona comprises only nine districts. Initially, we contemplated using neighborhood-level data, but this would have necessitated integrating another dataset containing geometries right from the outset, extending the initial data reconciliation process.

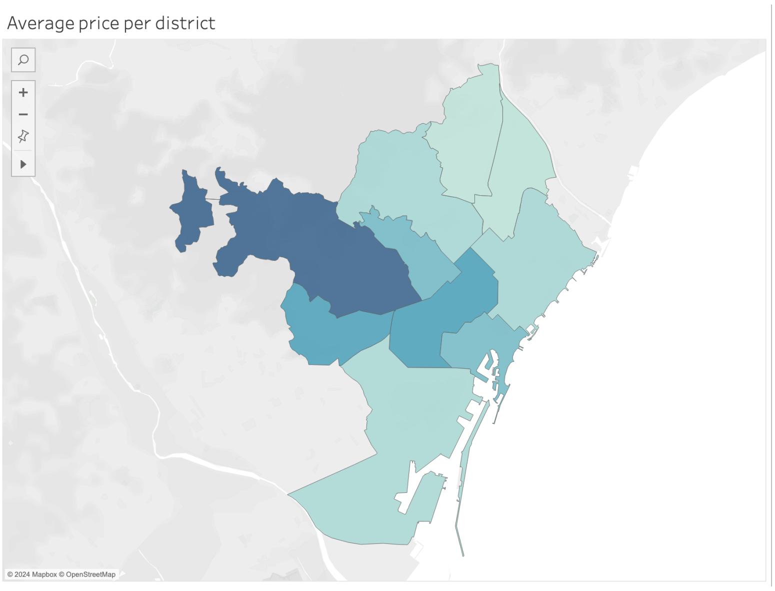


Figure 3: Average housing prices per district

We observe that average prices are notably higher in the `Sarrià-Sant Gervasi` district, located in the western area, compared to other regions.

3.1.2 Correlation of sale price and family income per district

This KPI measures the relationship between housing prices and district-level income. A high correlation indicates strong dependence of housing prices on the district's income levels (RFD). We decided to create a barplot to grouped by district.

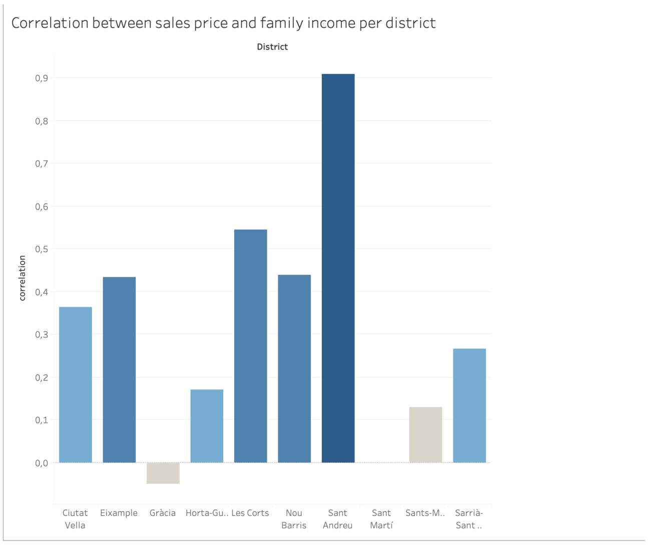


Figure 4: Correlation between sales price and family income per district

It is important to note that the correlation value ranges from -1 to 1. Values close to -1 or 1 indicate a strong correlation between variables, either negative or positive, respectively. In this context, the district of `Sant Andreu` shows a very high correlation, meaning that sale price and family income are strongly positive correlated in this district. Conversely, a correlation close to 0 indicates that housing prices and income are not correlated, and thus, no clear pattern can be discerned in that district.

3.1.3 Most voted party per district

We analyzed the most popular political parties in each district of Barcelona using recent election data. By filtering out non-relevant parties like "Electors", "Votants", and "Vots vàlids", we aggregated votes per district and party using Spark's groupBy and agg functions. Employing window functions with Window and rank(), we efficiently ranked parties based on total votes within each district, revealing insights into the dominant political preferences across the city's districts.

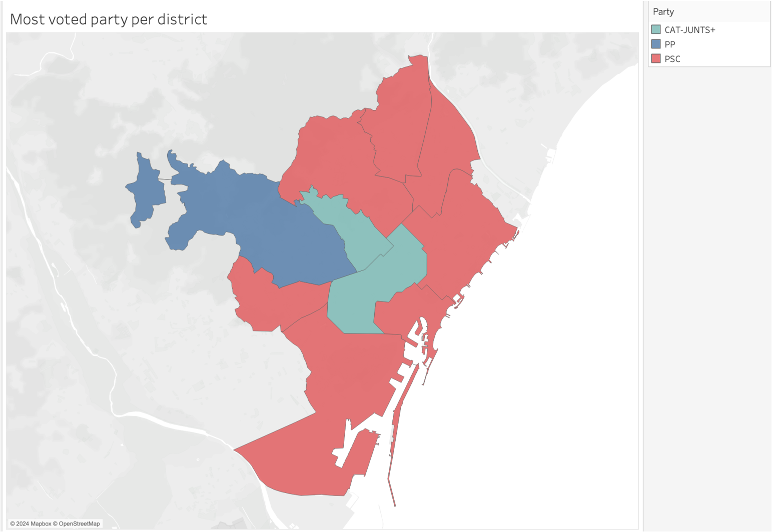


Figure 5: Most voted party per district

3.2 Predictive Analysis KPIs

We utilized real data from Idealista supplemented by social and economic data from OpenData Barcelona to predict housing prices. Leveraging Spark's capabilities, we conducted a predictive analysis employing a multiple variable linear regression model. Initially, we identified variables highly correlated with the "price" variable, which were subsequently included in our prediction model. The features used for prediction were `['neighborhood','RFD', 'price', 'bathrooms', 'size']`. After training the model, we serialized and saved it for future use. Evaluation metrics such as r^2 and mean absolute error were employed to assess the model's performance, ensuring robust predictive accuracy for housing prices in Barcelona. The final error metrics gave a MAE of 303.159 EUROS and a rsquare of 0.371795, this value of rsquare is low because of the quality of the data available, having more quality data or updated would make the model perform better in this parameters. We can observe the error plot in 9, where we can see that some outliers are present and bias the whole linear regression, but if we look at the cluster near the origin axis most of our observations are quite represented with some typical error given by the relation of the data.

Apendix

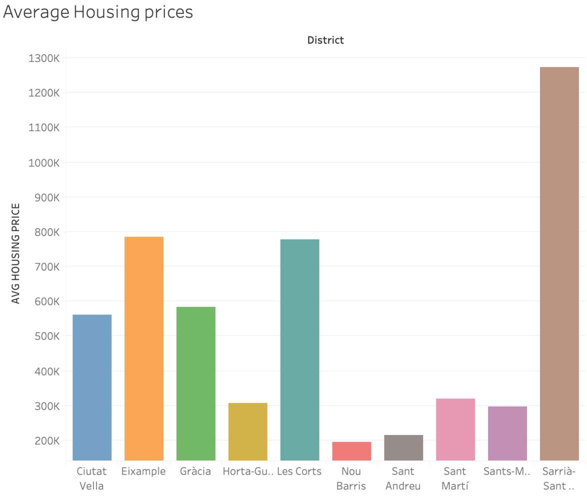


Figure 6: Average housing prices

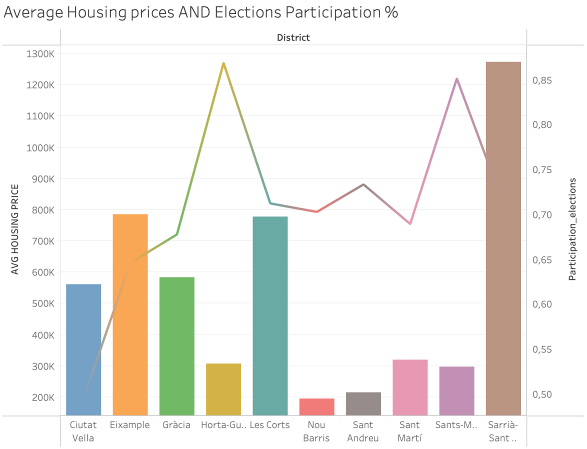


Figure 7: How housing prices correlates with elections participation

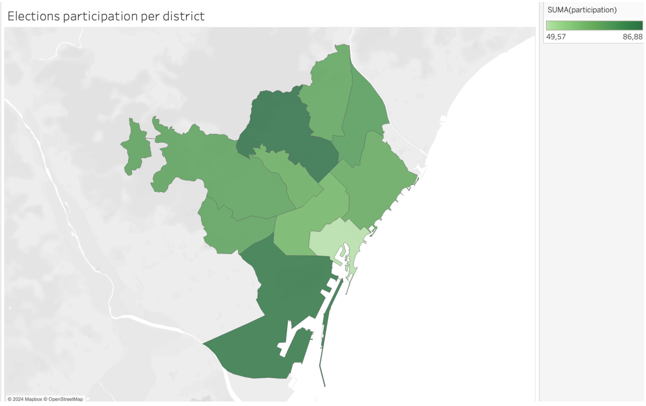

Elections participation per district
SUMA(participation)
49,57
86,88
© 2024 Mapbox © OpenStreetMap

Figure 8: Elections participation per district

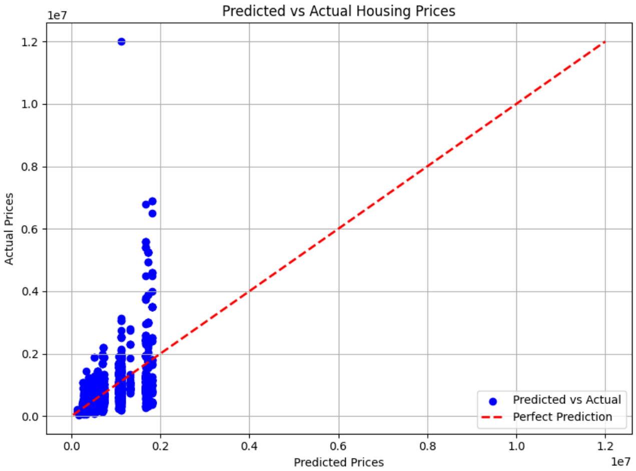


Figure 9: Error plot

References

[1] Ajuntament de Barcelona. Districts and neighborhoods dataset. `https://opendata-ajuntament.barcelona.cat/data/en/dataset/20170706-districtes-barris`, 2017.