João Carlos Pinto Santos

Data Science Task and Data Engineering Challenge

February 2, 2021

Contents

1 Introduction

Work carried out within the scope of the application for Data Inter Position. In the next sections, the decisions made during the elaboration of the challenges will be explained. The tools used and their reason will also be explained.

The code developed to carry out the challenges is duly commented in order to facilitate its interpretation and analysis.

Absolute path for the dataset needs to be changed into the python file!

airbnb_dataset_jcps.py

2 Data Science Task - The AirBnB dataset

2.1 Tools used

- **Python3** - No introductions needed!
- **Pandas** - Pandas is the most popular python library that is used for data analysis. It provides highly optimized performance with back-end source code is purely written in C or Python. Pandas seems to be the best option when we need to work with datasets smaller than 1GB.
- **matplotlib.pyplot** - Mainly intended for interactive plots and simple cases of programmatic plot generation.
- **seaborn** - Provides an API on top of Matplotlib that offers sane choices for plot style and color defaults, defines simple high-level functions for common statistical plot types, and integrates with the functionality provided by Pandas DataFrames. Resuming: Simpler and full of features.

requirements.txt attached with versions

2.2 Exercise Resolution

The dataset airbnb_dataset.csv presents data about some US airbnb properties. Work with this dataset to answer the following questions:

- **1. How many features are represented in the dataset? In terms of statistics, what type of variables are "host_since", "host_is_superhost", "room_type", "bedrooms" and "price"?**

A feature is a measurable property of the object we're trying to analyze. In datasets, features appear as columns. The quality of the features in the dataset has a major

impact on the quality of the insights we will gain when we use that dataset for analysis.

The dataset has 158284 rows and 16 columns.

"host_since" - In a practical way it represents the date of beginning of rental or first rental. In some cases, the measurement scale for data is ordinal, but the variable is treated as continuous. Statistically it represents a continuous numeric variable because it can obtain infinite values (1 year = 31 556 926 seconds) being always lower than the current date.

This feature appears as object data type, with class 'str' 158255 values and class 'float' 29 values. Pandas supports DateTime data types, we know then that it will have to be cleaned so that it can be used.

"host_is_superhost" - Indicates whether the host has a superhost category or not, and can only obtain a "Boolean" value of "t" or "f", so it is considered a "Categorical" variable. Again the feature appears with different data types (str and float), so, this parameter must be worked in order to be represented as a Boolean data type True or False.

"room_type" - It represents the type of accommodation and is therefore considered a categorical feature type. With string data type it presents 4 possible options ['Entire home / apt', 'Private room', 'Hotel room', 'Shared room'].

"bedrooms" - Represents the number of rooms in the accommodation, the values range from 1 room to 50, thus being considered a numerical / measurement feature type with datatype float64.

"price" - Price per night in accommodation, falls into the quantitative / numerical feature type because that includes any variable that can be counted, or has a numerical value associated with it. the datatype is int64 and the values vary between 10$ and 25,000$ per night.

All the cleaning processes made explained inside the code file!

- **2. What are the two most correlated variables?**

Correlation can be useful in data analysis and modeling to better understand the relationships between variables. The statistical relationship between two variables is referred to as their correlation.

Looking at the dataset without a graphical analysis and only with a critical perspective, we could normally associate variables such as accommodates and price, as the more accommodates the higher the price and vice versa or the more accommodates

would imply the greater number of rooms. However, these values must be analyzed very carefully, as this linearity does not always exist.

Use of the corr() method referring to the panda library with the standard "person" correlation method in order to obtain a coefficient that allows us to assess the degree of correlation between the different columns / variables.

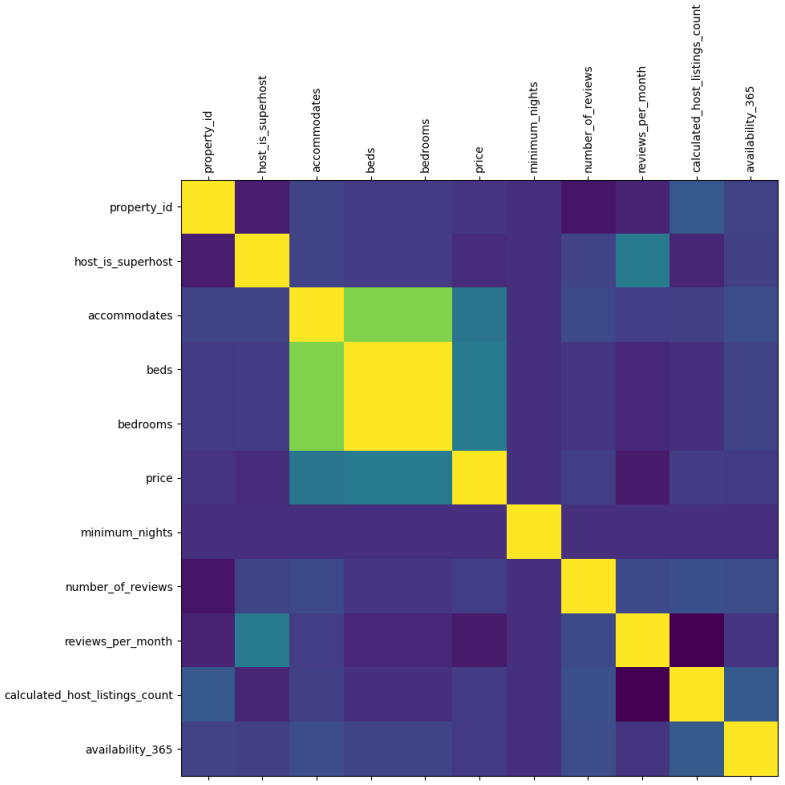
Pearson's correlation is understood to be strong the closer its value is to the edges. The P interval is between [-1.1], where the highest absolute value corresponds to the highest correlation between two variables. In this case, with a correlation value = 0.999977, the variables "beds" and "bedrooms" have the highest degree of correlation, indicating an almost perfect positive correlation.

- **3. What are the two least correlated variables?**

We can take as a "metric" that the closer to 0 the correlation between two variables the weaker it will be. If the correlation is equal to zero it means that the two variables do not depend on each other.

The two least correlated variables are "price" and "minimum_nights" with a correlation value = -0.000687

- **4. Create a plot for the correlation matrix. According to this matrix, what are the most relevant attributes? How can you use these findings for feature selection?**



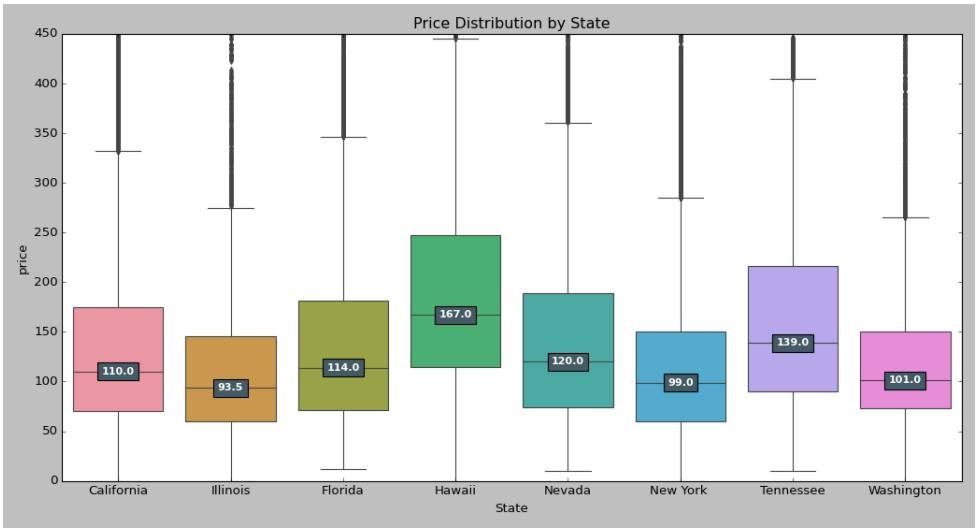
Feature selection and Data cleaning should be the first and most important step of model designing because irrelevant or partially relevant features can negatively impact model performance.

The most relevant attributes are directly associated with attributes with a strong degree of correlation. Attributes such as "beds", "accommodates", "bedrooms" and price are strongly associated and must be taken into account as decisive elements.

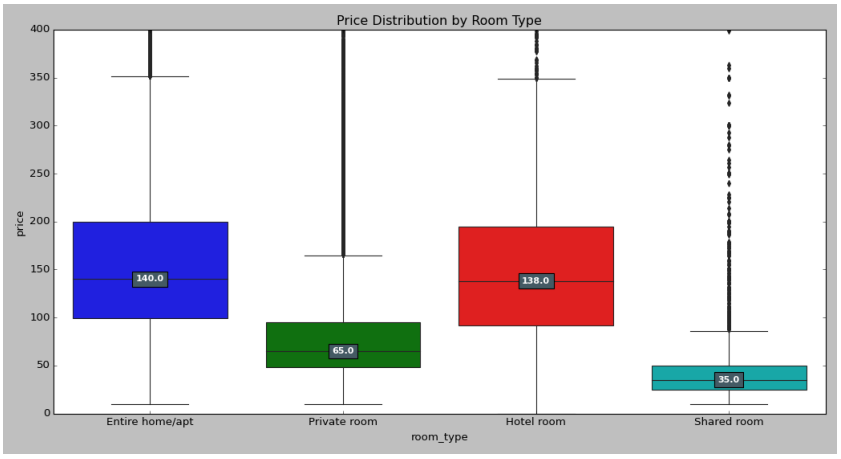
We can also notice that the number of reviews per month has a good association with whether or not the host is a superhost. It will probably be a defined criterion for obtaining this status.

Variables with a global degree of correlation close to zero should be dosed for feature selection. For example the property_id variable makes no sense to be included in our model, it will only delay computing and data analysis.

- **5. How does the price distribution vary by state? And by room type? Create a few slides summarizing your findings about this.**



With analysis of the graph we can see that the state of Hawaii has the largest price variance with an average of 167$ per night. On the contrary, the average price paid in the state of Illinois is 93.5$ per night. We can also observe that the state of Washington has the lowest rate of price change relative to the states analyzed.



When comparing the distribution of prices with the type of room we observe a small difference between the two most expensive types, however, the type of room "Entire

home / apt" surpasses the others with an average price of 140$ per night. With the lowest variance and also the lowest average we have the "shared room" with an average price per night of 35$.

3 Data Engineering Challenge - The Native Wines dataset

3.1 Tools used

- **Python3** - No introductions needed!
- **Pandas** - Pandas is the most popular python library that is used for data analysis. It provides highly optimized performance with back-end source code is purely written in C or Python. Pandas seems to be the best option when we need to work with datasets smaller than 1GB.
- **sqlite3** - Small, fast, self-contained, high-reliability, full-featured, SQL database engine. The use of sqlite3 module, allows to create a connection object that represents the database and then we can create a cursor object, which will help executing all the SQL statements.

requirements.txt attached with versions

3.2 Exercise Resolution

The code developed to carry out the challenges is duly commented in order to facilitate its interpretation and analysis.

- **How many rows are in the staging_wines table?**

The best way to obtain the number of rows in the table, since our dataset is not very long, is to execute the query SELECT Count(*) FROM staging_wines"

We can see that the table has 24997 rows.

- **What is the average price of a bottle of wine?**

SELECT price FROM staging_wines WHERE price IS NOT NULL - total / len

It only makes sense to average prices for bottles that display the price. Therefore, in the average calculation, wines without price are discarded. Average Price = 35.47$

- **How much is the most expensive wine?**

SELECT price FROM staging_wines WHERE price IS NOT NULL - MAX

The price of the most expensive bottle of wine is 2500$. It will certainly be good! :)

- **How many rows are there in the FactWine table?**

After all operations (cleaning and population), we can see that the FactWine table has 20809 rows.

- **What would happen if you deleted a record from the DimWinery table?**

The word "delete" and "dim table" should not be in the same sentence! We should not delete from the dim table. Instead, we should update changed rows, and insert new rows.

if you deleted a record from a dimension you will be left with less information for the corresponding record in the factual, holes will appear with the removal of records in the dim table.

In more severe cases, if all data is erased and the table is truncated, the primary key gets changed from the dimension table. All the columns are having new primary keys.

- **How could you prevent that deletion from occurring?**

Transactions as concepts are extremely important in the database. If we really need to carry out a removal operation on a dim table we must carry out the plan using a transaction, in the event of any step failing the rollback occurs and we are not damaged by our operation.

On the other hand, when we really do not want a delete operation to be performed on any specific table, it is convenient to use triggers. A trigger is a special type of stored procedure that automatically runs when an event occurs in the database server, for this reason it is one of the best ways to prevent someone from deleting or any other unwanted action in the database

- **What sort of process do you imagine could be used to automate the weekly task of updating The Wine Mart from the new file?**

To automate the weekly task of updating The Wine Mart from the new file it is convenient to implement a set of transactions in our data model to deal with new data that will be introduced. New data may lead to errors in the existing database and the deformation of the different tables. After implementing these sql statements, it would be advantageous to use any type of task scheduler. The task scheduler

will execute a process at one or more regular intervals, beginning and ending at a specific date and time thus allowing to update our wines according to the new wines presented by our supplier.

- **What sort of software tools would be most appropriate for this process?**

This task is possible to run locally with the local scheduler, either with the operating system IOS, linux or windows. For larger quantities of operations, which involve more security and data integrity measures, it is convenient to use software available on the market.

Celery, being a distributed task queue, is a good example of software that can be integrated in order to streamline and increase the performance of these transactions. Celery is an asynchronous task queue/job queue based on distributed message passing. It is focused on real-time operation, but supports scheduling as well!

- **What changes could you make to improve the The Wine Mart design to provide an audit trail of when different wines are added and/or updated?**

In simpler cases, the implementation of triggers associated with tables can play an important role when it is necessary to roll back operations. Triggers that are triggered when manipulating data in tables and increase logs with information regarding what was done.

Introducing the concept of slow changing dimensions, it could be a very advantageous alternative and would facilitate the role of any necessary audit. A Slowly Changing Dimension (SCD) is a dimension that stores and manages both current and historical data over time in a data warehouse. It is considered and implemented as one of the most critical tasks in tracking the history of dimension records.

4 References

References

[1] https://towardsdatascience.com/the-search-for-categorical-correlation-a1cf7f1888c9

[2] https://stackoverflow.com/questions/38649501/labeling-boxplot-in-seaborn-with-median-value

[3] https://pbpython.com/pandas_dtypes.html

[4] https://www.kaggle.com/adityasswami/airbnb-sydney-an-exploratory-data-analysis

[5] https://medium.com/@outside2SDs/an-overview-of-correlation-measures-between-categorical-and-continuous-variables-4c7f85610365

[6] https://stackshare.io/stackups/celery-vs-sqlite

[7] https://www.codeproject.com/Articles/5269227/Cleaning-Data-in-a-Pandas-DataFrame/

[8] https://sqlite.org/datatype3.html

[9] https://www.nuwavesolutions.com/slowly-changing-dimensions