Author: Caroline Moraes da Cruz

Seazone Challenge for Junior Data Scientist

Seazone is a company that sells nightly stays in apartments, houses and hotels, which are booked through online travel agencies as well as the company's website. Seazone provided two datasets, one containing information about the listings available and another containing information regarding daily revenue for each property. The objective of this case was to analyze the data provided and construct models to predict behaviors of different features.

For this project, it was used Python, since this is a well-established and disseminated language, with easily found documentation and a large number of users. Also, Python is a versatile language, which allows the use of functions to improve code running.

The project uses a cookiecutter structure, since this is used in a large number of projects of data science. Organization throughout the development of the project was made using git and GitHub.

Data Cleaning

The first step into analyzing the datasets received was importing the data. It was noticed that the types of some of the data were not befitting to the nature of the variable, such as numerical variables listed as string. In order to adjust this information, cleaning functions were made for both datasets, which can be found in the "make_dataset.py" file. In addition, the dataset was limited to the data until March 15th, 2022, that was the date in which the dataset was received.

Features Engineering

Several building features were implemented throughout data analysis and modelling, all of them can be found in "build_features.py" module. Firstly, it was implemented "build_daily_features" in order to determine the number of days between booking and check-in, adding a column to the "df_daily_revenue" dataframe called "reservation_advance_days". The objective was to analyze the pattern of reservation advance.

Here, the “creation_date” was considered as the date of booking, as it differs from the variable “Data Inicial do contrato” in the listings table and it is only present when the listing is occupied (“occupancy” = 1). The advance days number was obtained through the subtraction of “date” from “creation_date” and transforming it into days. Cases in which “creation_date” was greater than “date” were considered to be invalid and turned into “NaN”.

The function “build_listings_features” was implemented for the listings dataset, in which properties were categorized using numbers from 1 to 5, where 1 was attributed to properties whose quality were considered as “SIM”, and “MASTER” quality was considered as 5. From the string in the “Categoria” column, the number of rooms was obtained and a column “Quartos” was added to “df_listings” in order to model prices and revenues for properties with similar quality, but different number of rooms.

The function “build_date_features” uses functions from “commons.py” in order to decompose the date, add a column containing the day of the week and turn it into a one-hot encoding structure, as well as add another column indicating whether the date is a holiday. First, the date was decomposed into separate columns, namely “year”, “month” and “day”. In order to do so, timestamp dates listed in the dataframe were converted into datetime format for each feature.

The column labeled as “day_of_week’’ was added to the dataframe using the attribute dayofweek and, by the use of a dictionary, labelling each day accordingly (0 is “Monday” and 6 is “Sunday”). This categorical variable, then, was converted using one hot encoding, therefore preparing it to be used when modelling to evaluate the effect of the day of the week in the revenue.

Finally, a column was added to daily_revenue dataframe to indicate whether a day is a holiday in Brazil. The function “is_holiday” was implemented using the holiday library. This is useful to help determine the influence of holidays on the revenue obtained in a period.

In case it is necessary to recompose the year, month and day into a single date column, a function “get_date_from_ymd” was also added to “commons.py”.

Results

1) What is the expected price and revenue for a listing tagged as JUR MASTER2Q in March?

After due cleaning and adjusting of both “listings” and “daily_revenue” datasets, they were merged into a new dataset named “data”. Before the merging, the listings dataset already contained the number of rooms of each property, as well as numerically encoded features indicating its category, as explained in previous section. From the “data” dataset, it was observed that there were no listings that corresponded to the “MASTER2Q” category.

Thus, in order to obtain a model which can predict price and revenue based on category and number of rooms, for an explicit period of time (March), it was necessary to explicit these features in the dataset. The column “date” was decomposed into year, month and day. Both the day_of_week and localization were treated through one-hot encoding. At the end, a dataframe contained information on date, category and number of rooms, and two series had the last offered price and company revenue, which are the variables of interest for modelling.

Finally, the dataframe was preprocessed in order to fulfill missing data using mean as strategy and also scaled using MinMaxScaler. An instance of the XGBoost regressor was trained using 70% of data, and then the model was applied to test set, using mean average error (MAE) as an evaluation metric. It was also considered different regressors, such as Linear and RandomForest, however XGBoost was the best fit and chosen for this evaluation.

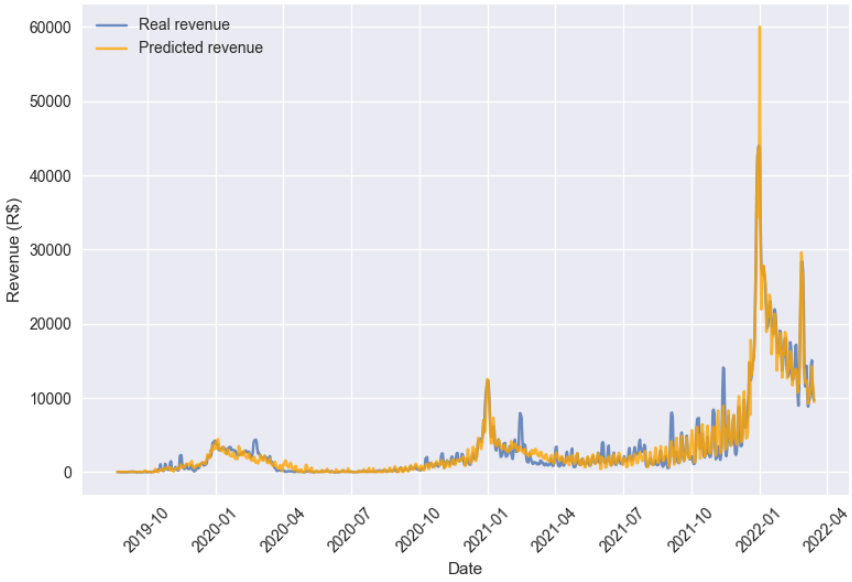
The trained model was applied in a synthetic dataset of listings in category = 5 (MASTER), with 2 rooms for all days of March. The price was estimated through average and revenue was estimated using sum. The **expected price** for a listing tagged as JUR MASTER2Q in March was found to be **R$ 483.77**. In a similar way, the **revenue expected** for this kind of listing is **R$ 4,127.13**.

2) What is Seazone's expected revenue for 2022? Why?

In a procedure similar to question 01, the datasets "listings" and "daily_revenue" were cleaned, adjusted and merged into a dataframe called "data". To this dataframe it was added a column of "company_revenue", which is a result of revenue multiplied by commission. The column "date" was decomposed into year, month and day and the day_of_week treated through one-hot encoding. It was obtained a dataframe containing information on date and a series with company revenue, which are the variables of interest for modelling.

The dataframe was preprocessed in a similar manner, that is, fulfilling missing data with strategy = mean and scaling using MinMaxScaler prior to modelling. In this case, however, it was used an instance of the MLP Regressor, since the data presents an extremally high level of non-linearity and seasonal peaks throughout the year. The model was also trained using 70% of data and applied to test set to determine mean average error. It is possible to see in Figure 1 the comparison of predicted values generated by the model and real data.

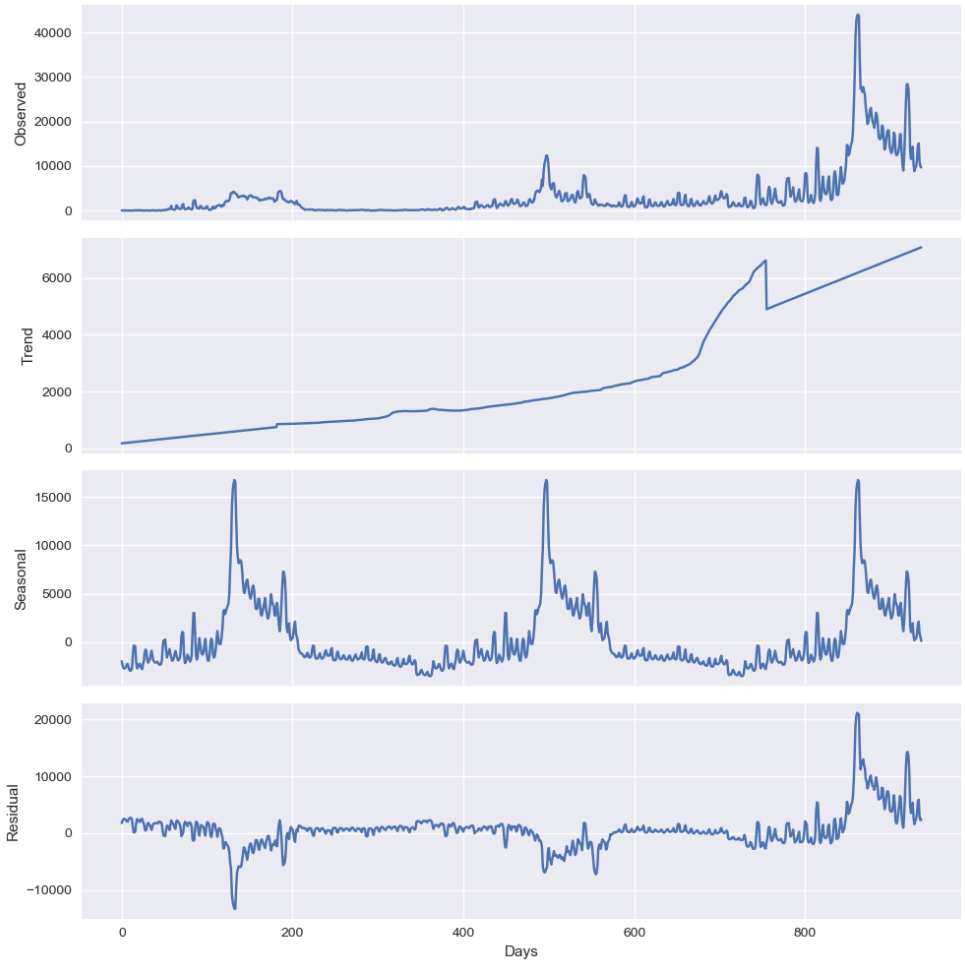
Figure 1 - Predicted and real revenue per date



This model was then applied to dates ranging from January 01st, 2022 to December 31st, and the **expected revenue** obtained was **R$ 4,447,642.69**.

In Figure 2, the company revenue series was decomposed into trend, seasonal and residual parts. The data shows that there is a trend of growth, which began with a sudden rise in revenue around August 2021. Revenue also presents a seasonal behavior, with peaks of interest in the December and January, probably due to Christmas and New Year. A smaller peak is also noticeable right after, around February/March, which is probably due to Carnival. In the same manner as observed in Figure 1, there is a period of low revenue starting in March 2020, which can be interpreted as a direct impact of the measures taken to prevent the spread of Covid-19.

Figure 2 - Seasonal decompose of company revenue

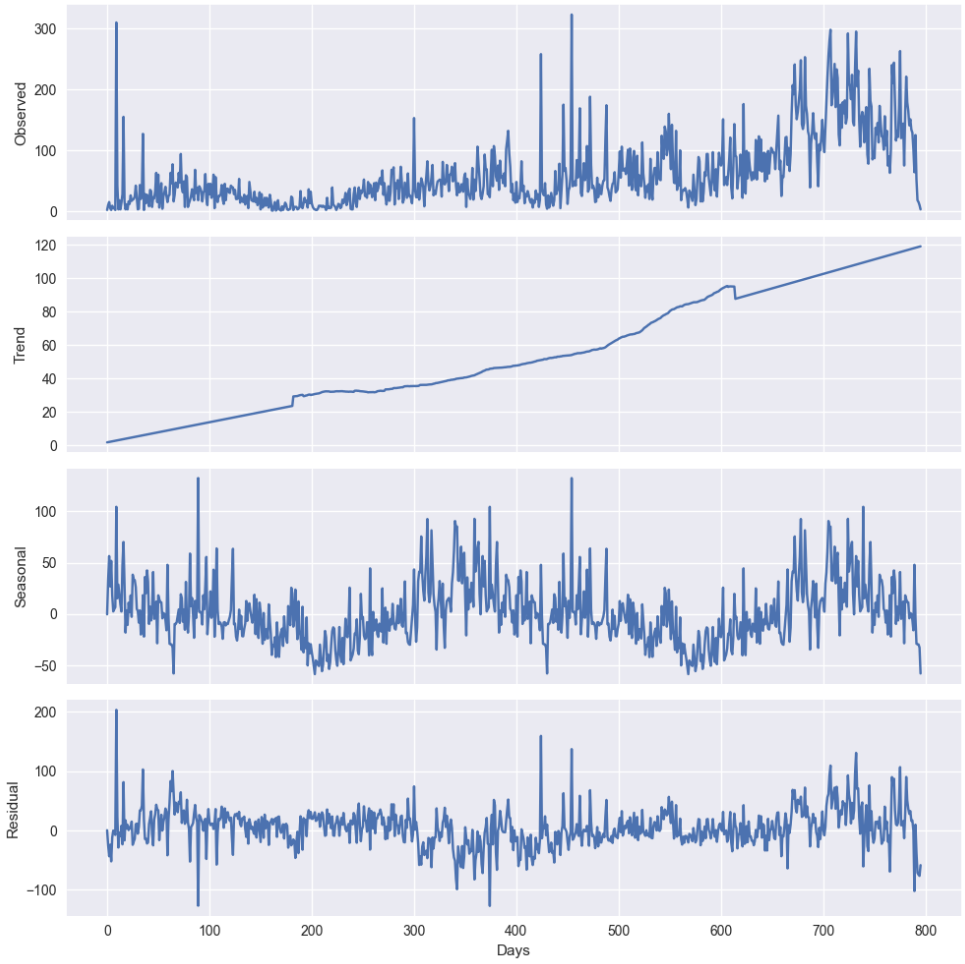


3) How many reservations should we expect to sell per day? Why?

To answer this question, it was used the “daily_revenue” dataset, which was filtered to consider rented listings, that is, with “occupancy” = 1 and “blocked” = 0. Reservations were grouped by day and counted, obtaining a dataset containing information on date and quantity of reservations. This dataframe was then treated similarly to the previous dataframes, by decomposing date into year, month and day, adding information on holidays and day of the week, and the latter was also treated through one-hot encoding.

Seasonal decompose of the quantity of reservations showed that there was a lot of residual noise, as can be seen in Figure 3. Thus, it was considered only the trend and seasonal variation for modelling.

Figure 3 - Seasonal decompose of quantity of reservations



Using the same procedure, the dataframe was preprocessed in order to fulfill missing data using mean as strategy and also scaled using MinMaxScaler. An instance of the XGBoost regressor was trained using 70% of data, and then the model was applied to test set, using mean average error (MAE) as an evaluation metric.

The trained model was applied in a synthetic dataset and the number of reservations per day was estimated through average. The **expected reservations per day for 2022** was determined as **63**, considering the historical data.

4) At what time of the year should we expect to have sold 10% of our new year's nights? And 50%? And 80?

- How can this information be useful for pricing our listings?

In this case it was also used the "daily_revenue" dataset, which was filtered to consider reservations for New Year's Eve. Through historical data provided, it was possible to determine the date in which a given percent of reservations for New Year's Eve was achieved.

Data shows that, historically, occupation rate of available listings on New Year's Eve is about 99%. Of these listings, it can be expected that **10%** of reservations will be sold by **October 23rd**. **50%** of new year's nights should be sold by **December 14th**, and **80%** should be sold by **December 24th**.

This result states that most people wait until December to make reservations. By November 30th, for instance, only about 35% of total listings are reserved. In this date, most people receive the first half of 13th salary, so it would be interesting to study a sale in order to sell the remaining listings and increase occupation on new year's night.

OPTIONAL: On the impact of the COVID-19 pandemic:

- Can we estimate Seazone's revenue loss due to the pandemic? How?

- Has the industry recovered?

- If Yes, when can we state that we came back from pre-pandemic levels of sales/revenue?

- If No, when do you expect this recovery to happen?

The datasets "listings" and "daily_revenue" were cleaned, adjusted and merged into a dataframe called "data". To this dataframe it was added a column of "company_revenue", which is a result of revenue multiplied by commission.

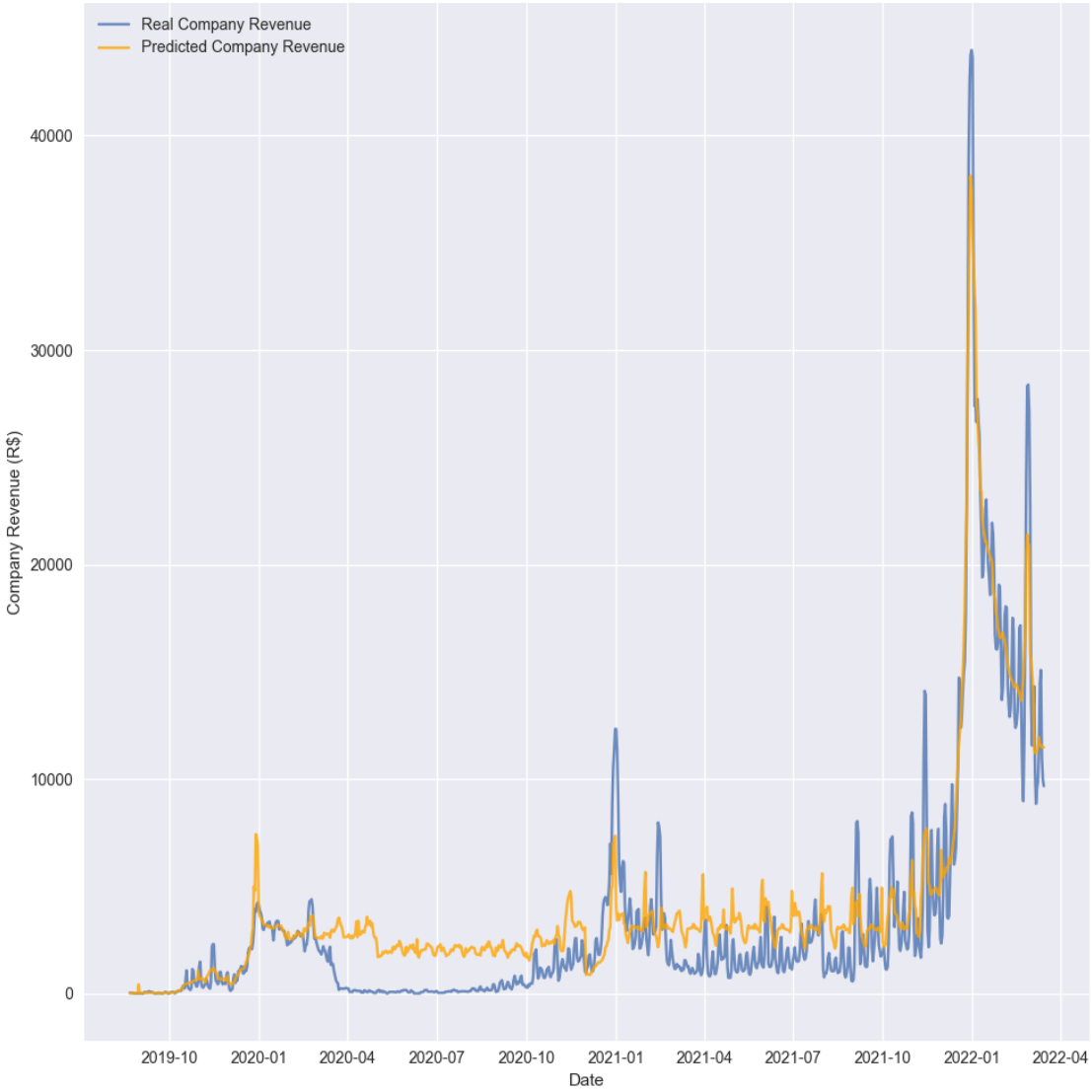
This dataframe was then filtered in order to remove all data from February 29th, 2020 to August 31st, 2021, which was considered to be the peak of Covid-19 pandemic. This was done in order to remove pandemic effect on data used for modelling. The column "date" was decomposed into year, month and day and the day_of_week treated through one-hot encoding. It was obtained a dataframe containing information on date and a series with company revenue, which are the variables of interest for modelling.

The dataframe was preprocessed in a similar manner, that is, fulfilling missing data with strategy = mean and scaling using MinMaxScaler prior to modelling. In this case, however, it was used an instance of the RandomForest Regressor, which presented lower mean average error when compared to other regressors. The model was also trained using 70% of data and applied to test set to determine mean average error.

The trained model was applied in a synthetic dataset and predicted company revenue was estimated using sum. This value was compared to real company revenue in order to determine the **estimated loss due to Covid-19 pandemic**, which was found to be **R$ 714,788.93**. It is possible to see in Figure 4 (next page) the comparison of predicted values generated by the model and real data.

In this graph, it can also be seen that **the industry has recovered**, and the level of revenue reaches the predicted value (based on pre- and post-pandemic data) around the end of **August, 2021**.

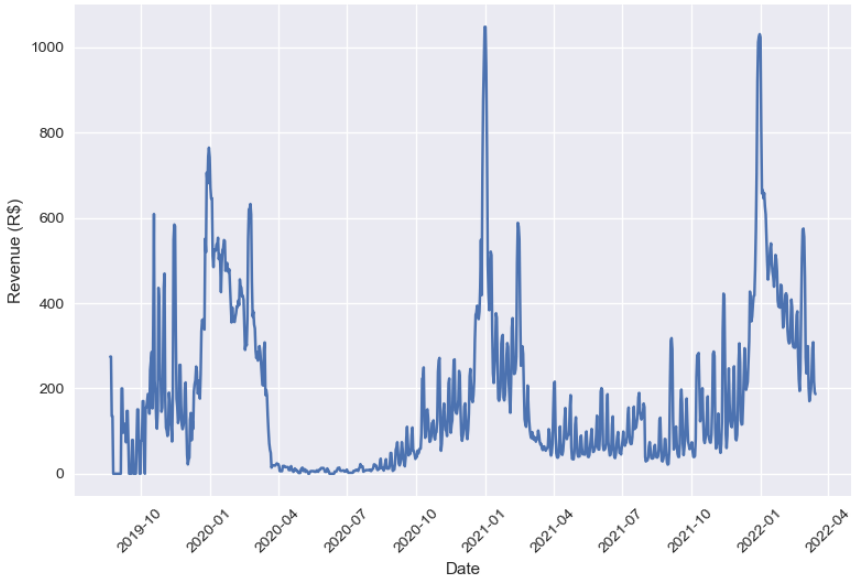
Figure 4 - Estimated loss on company revenue due to Covid-19 panemic



Complementary Data Analysis

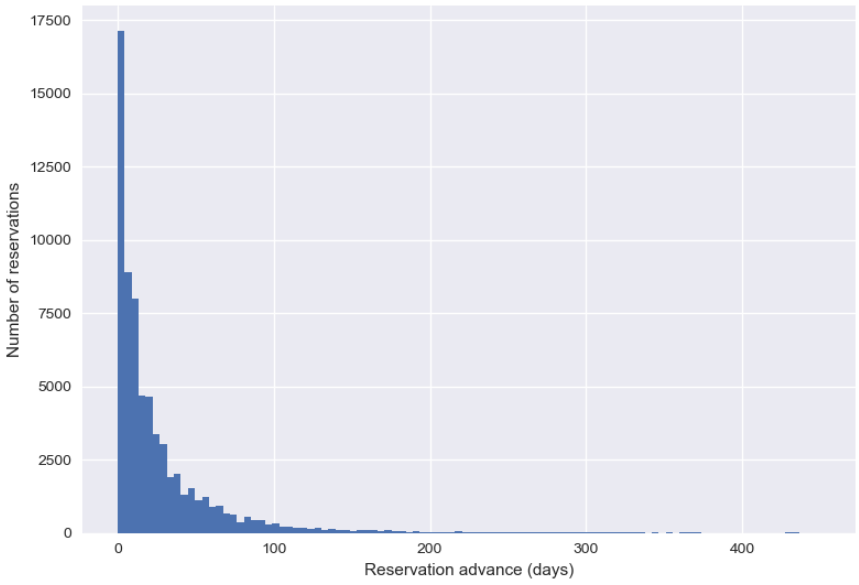
According to the data analyzed, the revenue obtained throughout the year is seasonal, with peaks of interest at the end of each year as can be observed in the graph in Figure 5. It can also be observed that there was a sudden reduction on revenue around March, 2020, which can be explained by the measures taken in order to slow down the spread of Covid-19 pandemic.

Figure 5 - Average revenue per date



Regarding the behavior on reservation advance, it was analyzed the distribution of advance days using the histogram shown in Figure 6. According to this data, approximately 35% of the customers book the listing less than a week before check-in. Similarly, 73% of customers book listings with an advance of up to a month, and only approximately 1.6% of customers do so with advance greater than 6 months.

Figure 6 - Histogram of reservation advance in days



Final Considerations

Through the analysis of the data provided, it was clear that the company is growing, which was explicit from the seasonal decompose graphs. It was also noticeable that seasonality is a factor with huge influence, as the number of reservations at the beginning and end of the year increases considerably.

It was detected a marketing opportunity to encourage bookings in advance for New Year's Eve. A suitable time would be by the end of November, as people could use the first half of 13th salary which is normally received around this date.

This project was really important, as it was necessary to deal with preparing, cleaning and working with real data, all of which are routine in data science. In this opportunity I was able to develop a full project in Python, that required a lot of research on good practices for Python programming, git usage and data modelling in order to answer all proposed topics. Further improvement to this project would comprise:

- Improve encoding method for category feature;
- Implement optimization method for model hyperparameters.