DATA PROJECT

RIO AIRBNB

PREDICTIVE MODEL AND INTERPRETABILITY

ALEX
DAUCOURT

alexdaucourt.com

INTRODUCTION / EDA

This project was one of my final exam in my M2 year. It is based on a dataset available in the github link, which contains information about apartements available for rent in the website Airbnb, in Rio de Janeiro.

This dataset contains the room type, the neighbourhood, the number of rooms, bathrooms and beds, the maximum number of person allowed and the title chose by the owner. Finally, we have the price of the apartment to train our model.

We will proceed with class, depending on quantiles of the airbnb's price.
Here are our results :

Quantile	Price
20 %	129 $
50 %	300 $
80 %	790 $

Thus, we have our **class 0** for apartments **lower than 129$**, **class 1** for **129 - 300 $** range, **class 2** for **300 - 790 $** and finally **class 3** for apartments **higher than 790 $**.

Then, I handled missing values.

Variable	Number of missing values
Bedrooms	24
Bathrooms	69
Beds	49

INTRODUCTION / EDA

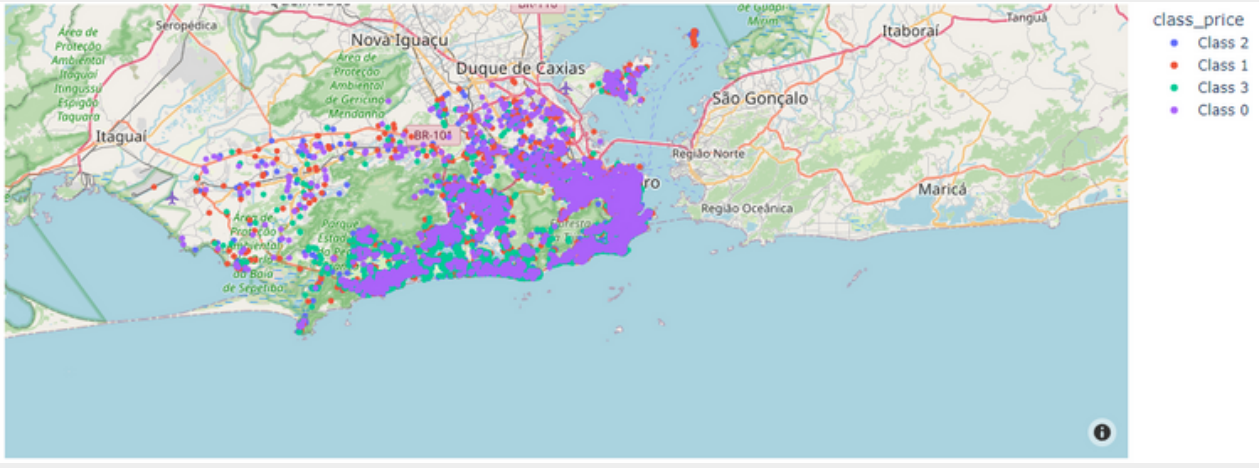
I decided to replace these missing values with the respective median of each variables. Moreover, our basic analysis of the dataset shows that in average, an airbnb in Rio has **2 rooms**, **3 beds** and **2 bathrooms** for a **maximal capacity** of **4 people**.

We also can have a look of all the room types :

Room type	Count
Entire home/apt	24 929
Private room	9 819
Shared room	854

Let's focus on localisation now. For 3030 rows, we do have a missing value for the neighbourhood. However, we dispose of latitude and longitude. so I tried to access the neighbourhood based on these datas. For performance issues, and because sometimes the neighbourhood we found was not the same than in the dataframe, I canceled this idea. Finally, I decided not to replace those missing values and to drop these apartments from our dataframe.

However, I created two more variables : distance from the "**Christ the Redeemer**" and from the "**Maracana stadium**", which are two attractive points in Rio.
We also can observe a map of Rio, with all our Airbnb with their respectives class :



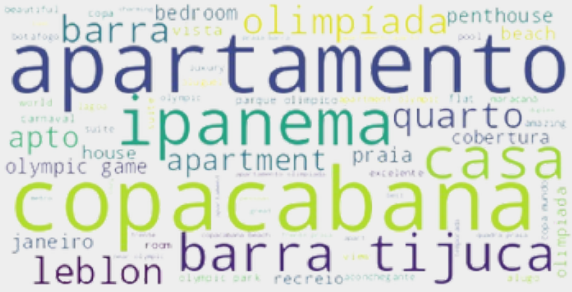
PREDICTIVE MODEL

First, I will try to predict our price class with text mining. Indeed, I will focus on airbnb's title, considering word clouds for each class to see if there are differences.
Here are the four clouds :

class 0

class 1

class 2



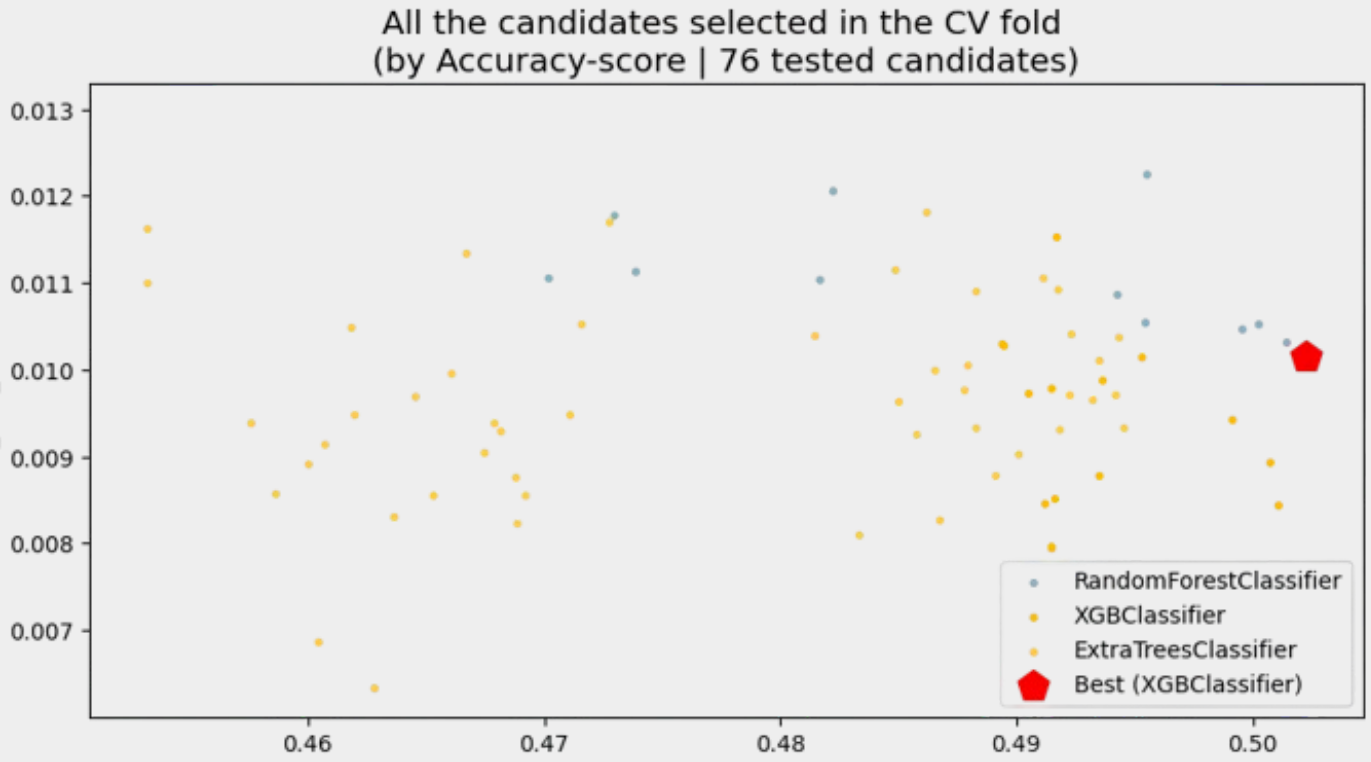
class 3

We can clearly see that word clouds are basically all the sames. Thus, it doesnt **not seem relevant** to create a predictive model based on text mining, because it would not be efficient. However, this may be interesting to delete the most commons words to see if there are differences among classes. Finally, maybe the title isn't big enough to provide us informations, but the description could be.

After that, I decided to test a lot of differents models with a CV search grid. In fact, I tested **random forests**, **XGB** and **ExtraTrees**, each of them with differents hyperparameters. Finally, I have 100 tested candidates, which I compare with the **score accuracy** metric.

PREDICTIVE MODEL

Here are our results :

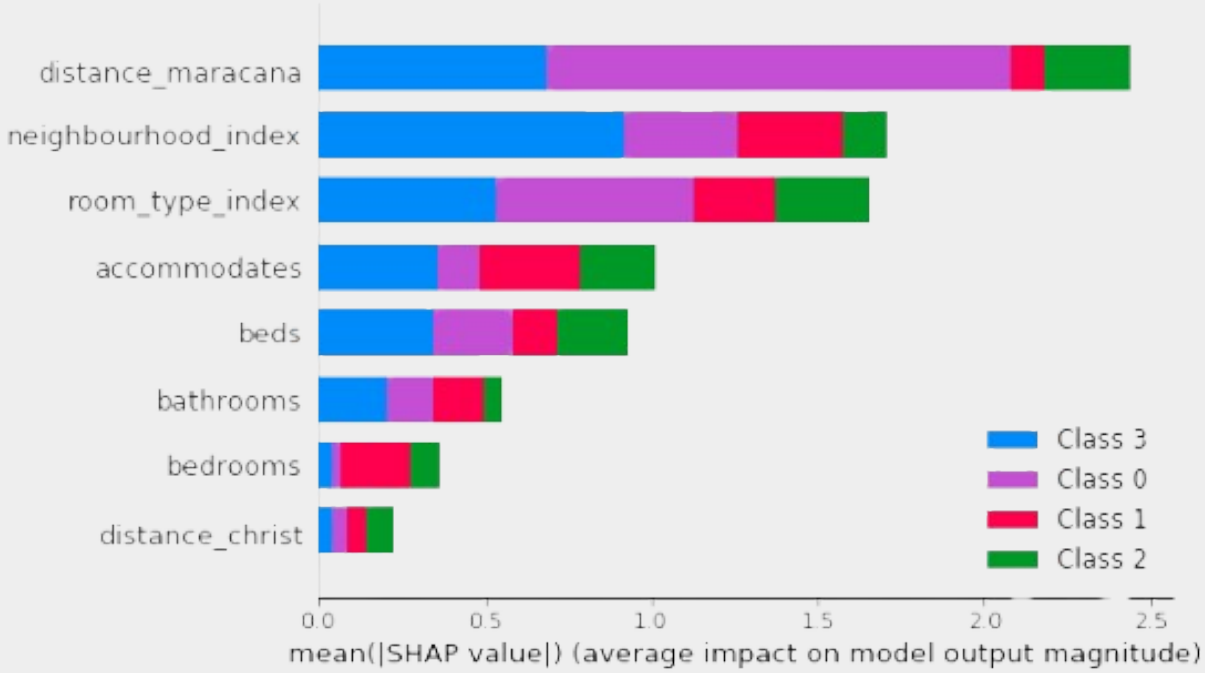


The best model is a boosting gradient one. We will do our predictions with a XGB model, with a **learning rate** of **0,1**, a **max depth** of **7** and **50** for the **n_estimators**.

Let's interprete our best model with the Shap library. However, the library changed the way **explainer.shap_values** calculates SHAP values for XGBoost, and the code is not runnable anymore. The analysis of interpretability is still relevant.

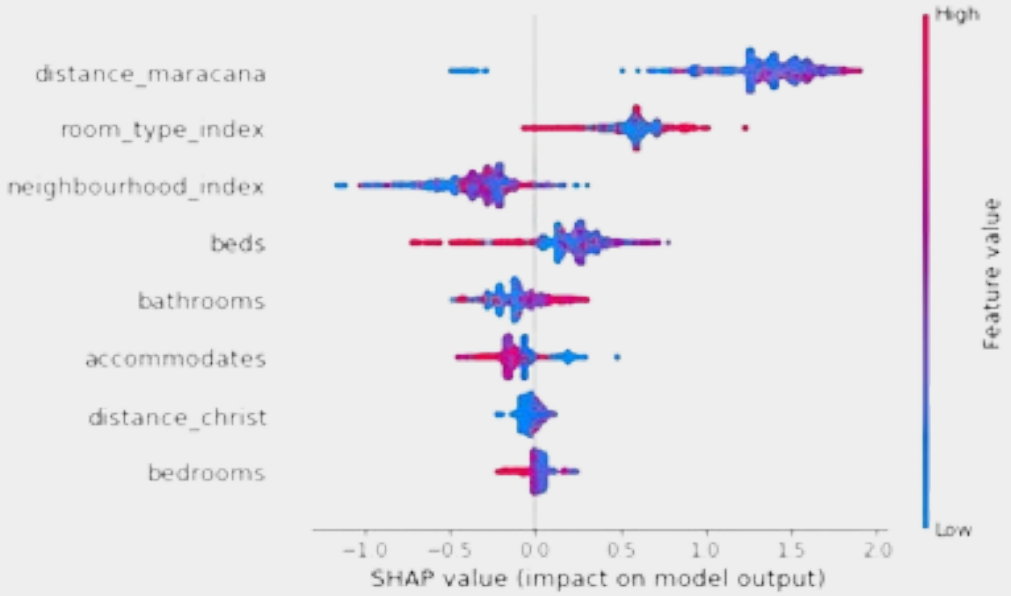
INTERPRETABILITY

Let's first have a look about the **most impactful variables in our predictions**, and in which way they can predict the airbnb's price :



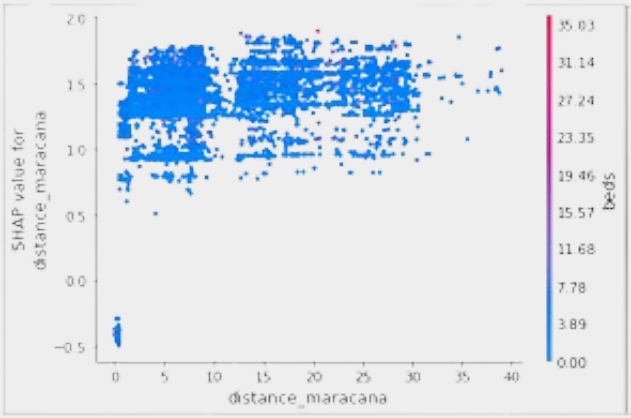
In this figure, we see the importance of the variables for each class. For this model, for example, the variable that has the **most impact** for class 0 is the **distance to Maracanã**. For class 3, it is the "**neighbourhood**" variable corresponding to the district. The type of property and the maximum accommodation capacity are also important parameters to determine the price of a property for all classes.

If we want more details about a particular class, we can display it, as shown in this graph for class 0 :

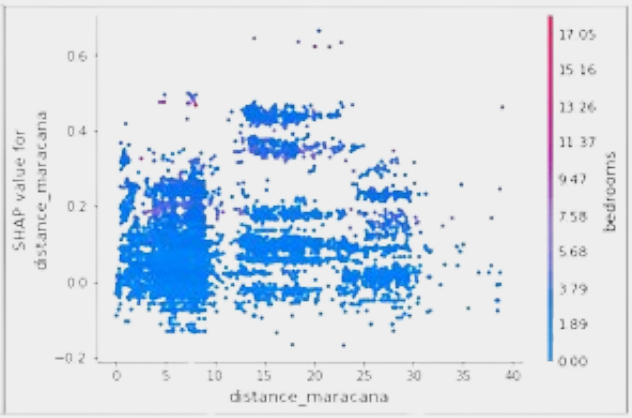


INTERPRETABILITY

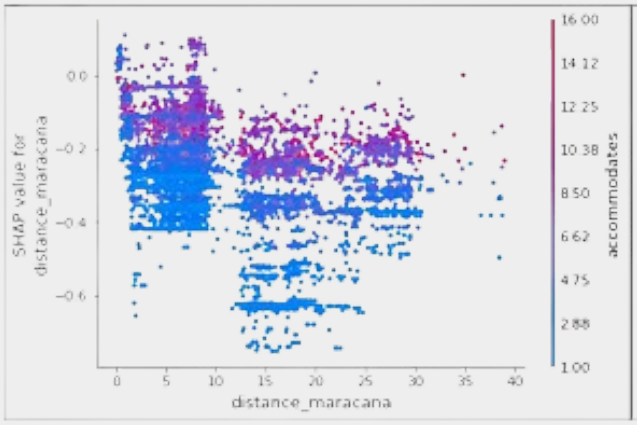
The y-axis lists the explanatory variables in decreasing order of importance, similar to the previous plot.
The x-axis represents the SHAP values : a **negative value** indicates a **negative relationship** and a **positive value** indicates a **positive relationship** of the explanatory variable (here, belonging to class 0).
The color bar on the right indicates whether the variable is high (in red) or low (in blue). For example, we see that for the "beds" variable indicating the number of beds, a high number of beds decreases the probability of belonging to class 0.
Let's now look at the dependence plots, specifically the dependence of the distance to Maracanã variable for the 4 different classes.



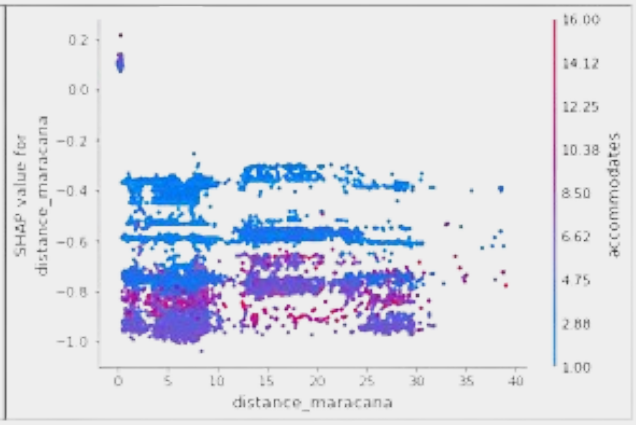
class 0



class 1



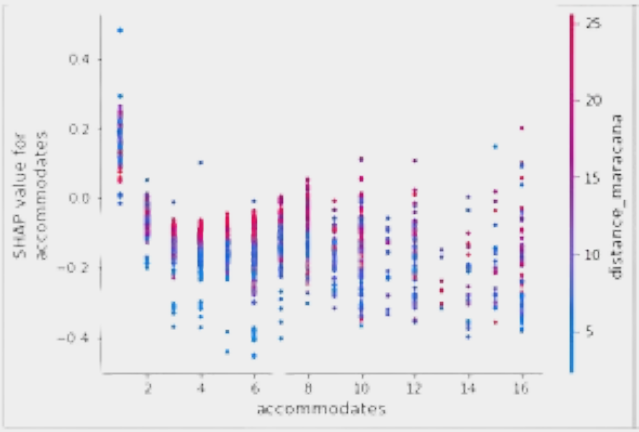
class 2



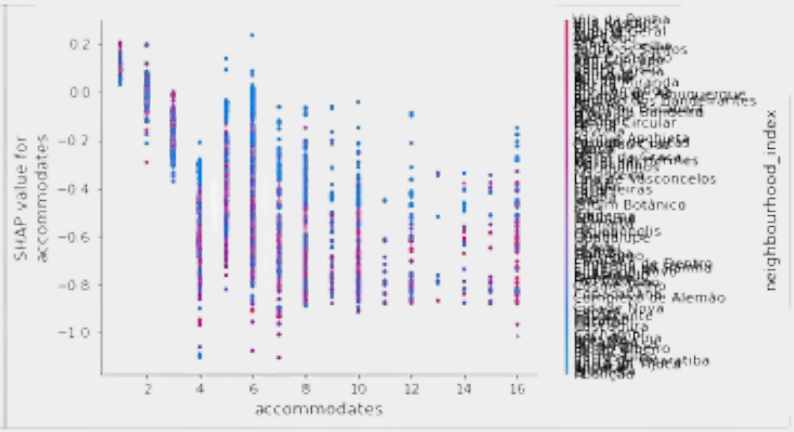
class 3

The y-axis represents the SHAP values of the chosen variable, while the x-axis represents the initial values of that same variable.
The color bar on the right represents a **scale for the values of the variable** that would most interact with the 'distance to Maracanã' variable. The points within the plot represent the **combined overall impact** of the two variables in predicting class membership. For example, for class 0, we see that a **distance of less than ~2km from Maracanã decreases the probability of belonging to the class** (as the SHAP is below 0). Conversely, beyond 2km, the probability of belonging to this class increases.
Additionally, the chosen interaction variable ('beds') does not seem to have a strong interaction that can be interpreted, as almost all the points are blue.

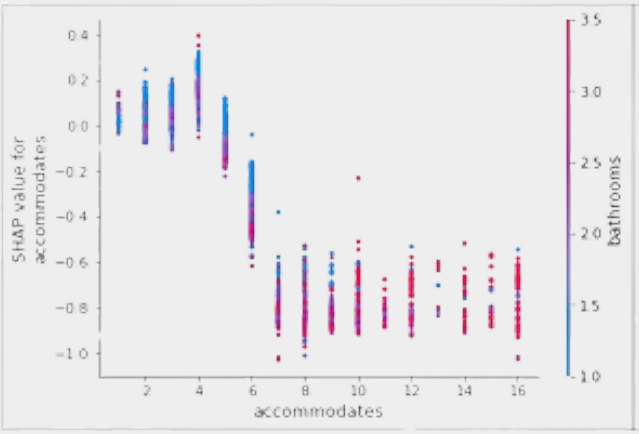
INTERPRETABILITY



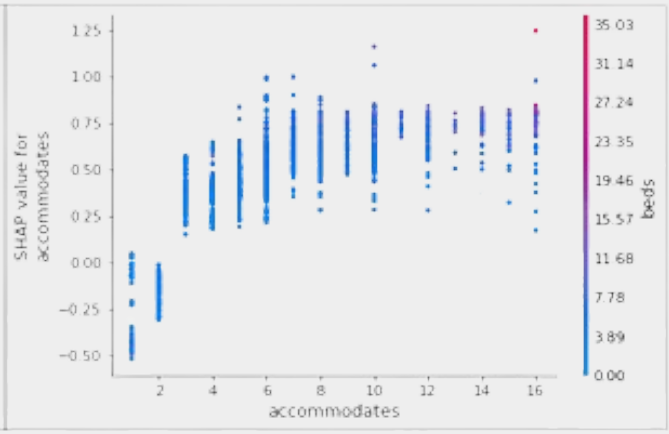
class 0



class 1



class 2



class 3

Let's now move on to the **dependence plots** for the 'accommodates' variable.
For class 0, we notice that only a capacity of 1 person has a positive SHAP value. This means that a **low accommodation capacity increases the chances of being in class 0**.
In the contrary, from 2 people onwards, the probability of being in class 0 decreases.

If we use explainer.expected_value, we can see the base values for each class. We can then determine the average probabilities for an apartment to belong to each class. In fact, we realize that these probabilities logically equal the class distributions in our total sample. We can calculate the probabilities with this basic formula :

P(Classe_i) = exp(expectedvalue_i)/Sum(expectedvalues)

INTERPRETABILITY

	explainer.expected.value	Softmax
Class 0	0,14023	0,17
Class 1	0,74588	0,31
Class 2	0,73057	0,31
Class 3	0.25118	0,19

Now, let's analyze a **force plot**.
To do this, we will randomly select an Airbnb and observe the 4 corresponding force plots for the 4 classes. The values on the axis are the **SHAP values** represented as **ln(odds)**. The base value is the average prediction in our dataset, the one we just calculated. Let's analyze the first force plot, for our class 0 :

The output value is 2.40.
This is the prediction of being in class 0 by our model for Airbnb number 66 in our test sample.
The red color represents the explanatory variables pushing the prediction up : we see that the **number of bathrooms**, **beds**, the **type of Airbnb**, and the **distance from Maracanã increase the probability** of being in **class 0**.
However, the blue color represents the explanatory variables **pushing the prediction down**, such as the **neighborhood** and **maximum capacity**.
We can calculate the probability using this formula: $P(x) = \frac{e^{ln(odds)}}{\sum e^{ln(odds)}}$

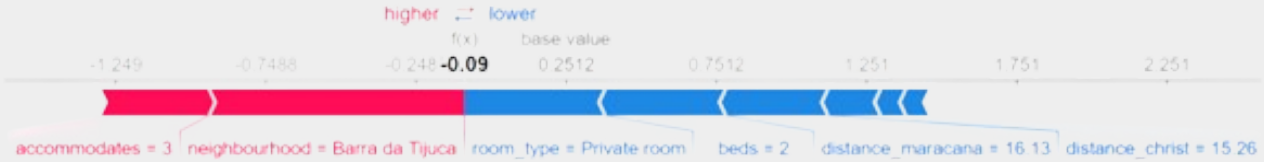
We find a probability of **77.9%** that this apartment belongs to **class 0**.
Similarly, we obtain a probability of 9.7% of belonging to class 1, 5.8% of belonging to class 2, and finally a probability of 6.5% of belonging to class 3.

INTERPRETABILITY

Here are the others force plots :

force plot for class 1

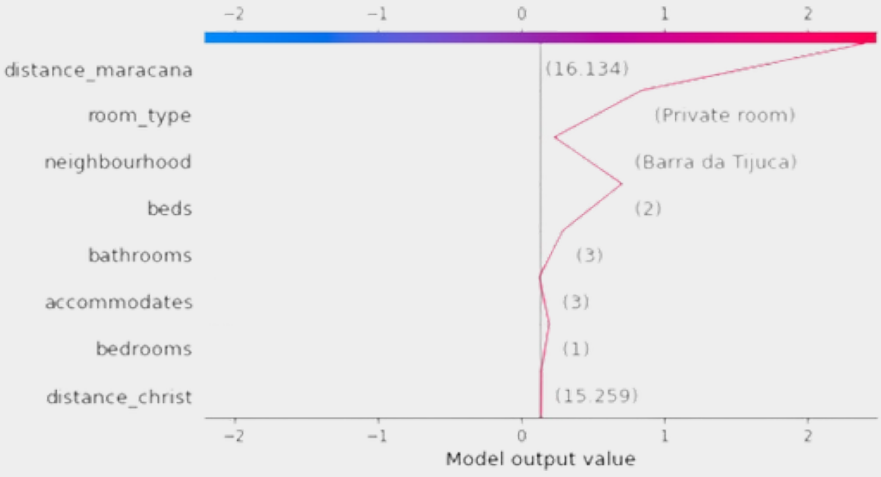
force plot for class 2



force plot for class 3

We can verify the actual class of our apartment by checking it in our base sample, and indeed it is in class 0.

To go further, we can look at the **decision plots**. SHAP decision plots show how complex models arrive at their predictions (i.e., **how models make decisions**).
A decision plot can reveal how predictions change as explanatory variables are taken into account.



For example, still with the same Airbnb, we see that the algorithm starts from the bottom of the graph with the base value **0.2512** and updates its shape value based on the Airbnb's characteristics until it reaches the value of **2.40** at the top of the graph, which we previously saw in the force plot.

Thanks for reading !

Rio Airbnb analysis project

by Alex Daucourt.