NOVA IMS
Information Management School

NATIONAL TOURISM PROMOTION

Professor Nuno António
15 January 2023

Paula Catalán
20221048
Vitória Simões
20210879

Bruna Kluwe
20220273
Katia Ferreira
20220237

Index

Introduction

The COVID-19 pandemic had a big impact on the tourism industry all around the globe and many organizations started a process of adapting to the market. This project is an exercise as a National Tourism Board Organizations (NTBO) consultant.

Using the CRISP-DM (Cross Industry Standard Process for Data Mining) process model to characterize and describe patterns of visitors to Portuguese attractions and compare with other countries/attractions, we will navigate through its six phases to guide NTBO on its study of the UGC (users generated content), to improve the tourism industry.

We have used three different modelling types to gain a thorough understanding of the data. Market Basket Analysis helps us understand the relationships between different attractions and identify which ones are frequently visited together. Data Similarity modelling helps us identify which attractions are similar to each other based on visitor demographics and behavior. Finally, RFM (Recency, Frequency, Monetary) modelling helps us understand how recently, frequently, and how much money visitors spend on different attractions.

This project is part of the Data Science for Marketing curriculum for NOVA Information Management School.

1. Business Understanding

In order to properly use the CRISP-DM methodology, we must start by defining the business objectives and user needs.

As a consultant, we must consider the NTBO objective: to study through UGC (Users Generated Content), categorizing and analyzing patterns of visitors to Portuguese attractions, and then comparing them with other attractions to give a solid solution on how to improve tourism in Portugal.

We can speculate about how COVID-19 may have affected many industries and lives, but in tourism, according to the World Tourism Organization (UNWTO), in 2021, the tourism percentage worldwide increased by 4% comparatively with the year before, but this number is still 72% less than in 2019[1].

Our goal is to dynamize the perception and to give solid solutions to validate the tourism levels in Portugal for the next few years - having in mind the dataset provided.

For this project, we are using Microsoft Excel to clean our data and Python to run the code.

2. Data Understanding

As the second step of CRISP-DM analysis, we must be familiar with the dataset, clean it, and organize it in a functional manner.

The initial data analysis was made through Microsoft Excel, as part of the process of cleaning the provided dataset. In terms of organization, we identified that the dataset could have been organized in a different way - there are many blank spaces and incomplete information, such as unuseful data - 6.04% of the data we have are not being considered when performing the analysis.

3. Data Preparation

3.1. Dataset structure

- The dataset is **92.120 rows** and **16 columns**.
- There is a total of **92.120 reviews**.
- **6.03%** of the total (5.559 reviews) are reviews from the Portuguese attractions.
- There are 97 attractions and the most visited one was the **Basilica of the Sagrada Familia** (Barcelona, Spain);

[1] Turismo em Portugal (6th May 2022) <http://www.turismodeportugal.pt/pt/Turismo_Portugal/visao_geral/Paginas/default.aspx>

- There are **65.785** different users.
- The most frequent user is "Malgorzata@Margo7850p" which is **present in 31 reviews**.
- Most of the users came from **London, UK**.
- All reviews are in **English**.
- The main reviews rating is **4.58/5**.
- **Couples** represent the most frequent trip type, 31.702 of trips were made by them.

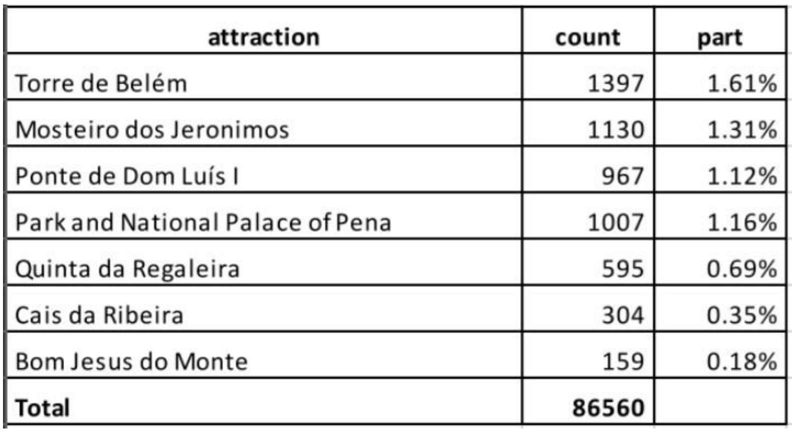
3.2. Considerations on data quality

user Location and trip Type have some Nan values that we will replace later.

4. Data Visualization

Since the Portuguese attractions are the focus of this project, we cleared the data with Excel Pivot in order to specify some points.

Before merging the data using Python, we counted the percentage of Portuguese attractions on the dataset available - so we would have a better understanding of how Portugal is positioned compared to the competitors.



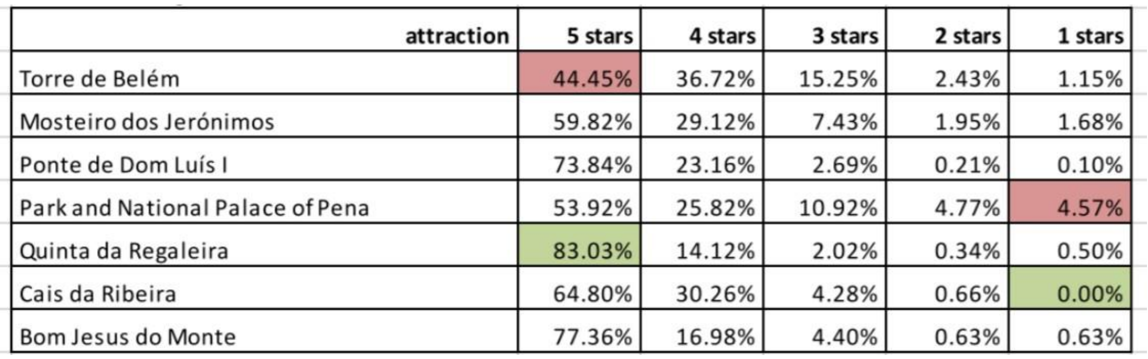
attraction	count	part
Torre de Belém	1397	1.61%
Mosteiro dos Jeronimos	1130	1.31%
Ponte de Dom Luís I	967	1.12%
Park and National Palace of Pena	1007	1.16%
Quinta da Regaleira	595	0.69%
Cais da Ribeira	304	0.35%
Bom Jesus do Monte	159	0.18%
Total	**86560**	

Fig 1. overview Reviews of Portuguese attractions

After understanding the proportions of Portuguese attractions within the 97 attractions list, we made some conclusions on which attractions visitors prefer.

The best-rated attraction was Quinta da Regaleira, with more than 80% of its ratings on 5 stars. The worst was Park and National Palace of Pena, with the most “1 star” rating - about 5% of its total ratings.

Bom Jesus do Monte is also a well-rated attraction, with most 5 and 4 stars.
The only attraction with a zero “1-star” rating is Cais da Ribeira - with 65% of its visitors giving 5 stars rating.



attraction	5 stars	4 stars	3 stars	2 stars	1 stars
Torre de Belém	44.45%	36.72%	15.25%	2.43%	1.15%
Mosteiro dos Jerónimos	59.82%	29.12%	7.43%	1.95%	1.68%
Ponte de Dom Luís I	73.84%	23.16%	2.69%	0.21%	0.10%
Park and National Palace of Pena	53.92%	25.82%	10.92%	4.77%	4.57%
Quinta da Regaleira	83.03%	14.12%	2.02%	0.34%	0.50%
Cais da Ribeira	64.80%	30.26%	4.28%	0.66%	0.00%
Bom Jesus do Monte	77.36%	16.98%	4.40%	0.63%	0.63%

Fig 2. overviewRatings, Portuguese classification

It is important to point out that we did consider only 93.96% of the dataset - 6.04% was unclear and/or unnecessary data.

Regarding the competitors, we can see the top 10 most visited attractions in Europe. Although Portugal has a good rating (as we could see above), it is still not part of this shortlist.

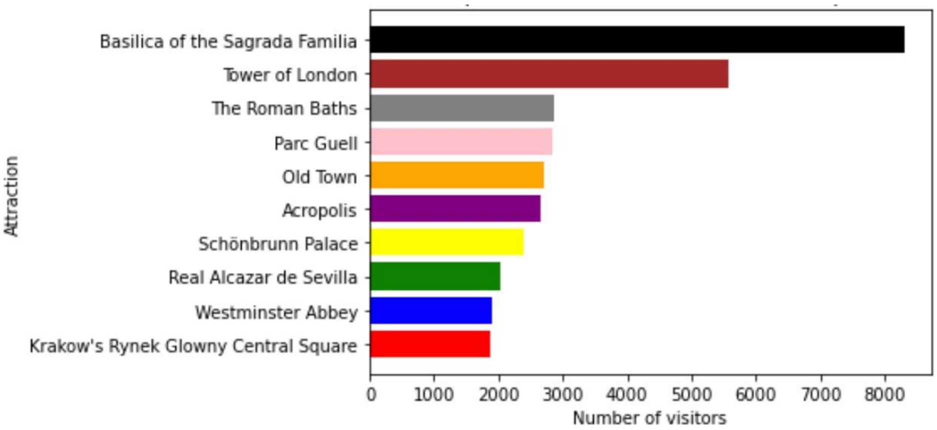


Fig. 3 Top 10 Attractions more visited in Europe (2019-2021)

Although Portugal's attractions are not in the Top 10, we can see Torre de Belém (Lisbon, Portugal) in the Top 20.

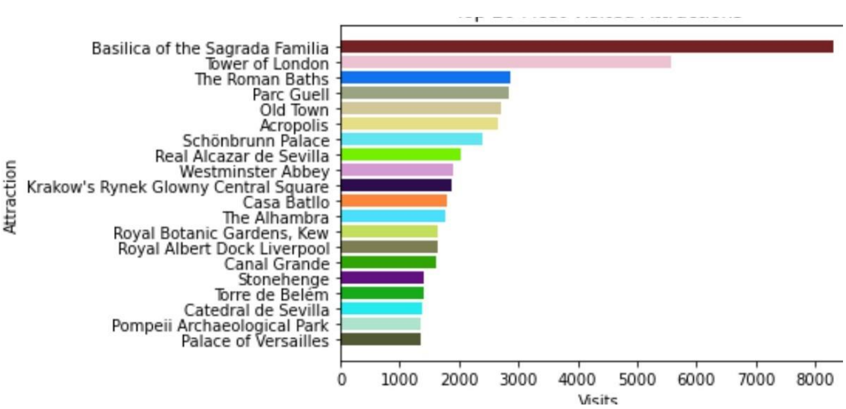


Fig. 4 Top 20 Most Visited Attractions (2019-2021)

5. Modeling

5.1. Data Similarity Approach

Similarity refers to the numeric measure of how alike two data objects are. We decided to explore this model to give the most complete analysis report to the NTBO. The value will be higher if the objects are more similar to each other. It often falls in the range [0,1], meaning that the most similar measures will be when the range is closer to 1 and the dissimilarity will show when the values are closer to 0.

To focus on visitors to Portuguese attractions we first did an analysis using a specific pattern of data that was more accurate for it. After merging the data from the Reviews data framework and the Attractions data framework we decided to filter the resulting DataFrame to only include rows where the "Country" column was equal to “Portugal”. As a result, we would have just Portuguese attractions to start the analysis.

After this step, we created a table to visualize the data we were going to use, and it showed that the most visited attraction in Portugal was ‘’Torre de Belém’’ with 1.397 reviews. Nonetheless, our main analysis consisted in **generating the matrix** that would show us the similarities between the Attraction’s name, the userName (meaning the user who visited the attractions) and the reviewRating.

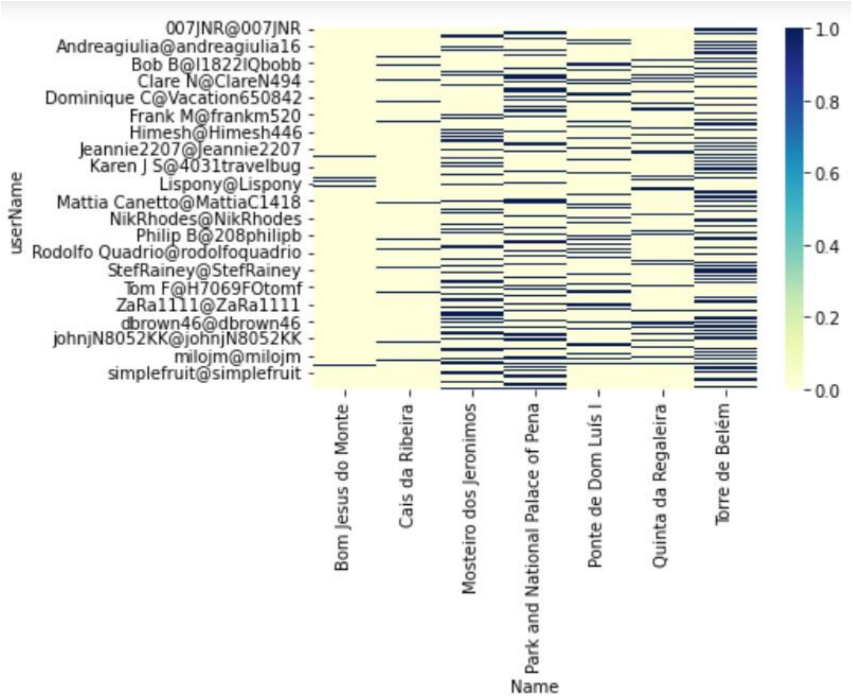


Fig. 6 userNames that gone to one or more attraction with review

This heatmap shows all the userNames that have gone to one or more attractions and have ratted with a review. The study shows that after the filtering, 3.957 userNames were left and seven Portuguese Attractions.

We then generated a **matrix** to explain the **userName** to **userName** patrons so we **similarity** would compare the similar users who prefer alike attractions with alike ratings as the matrix shows.

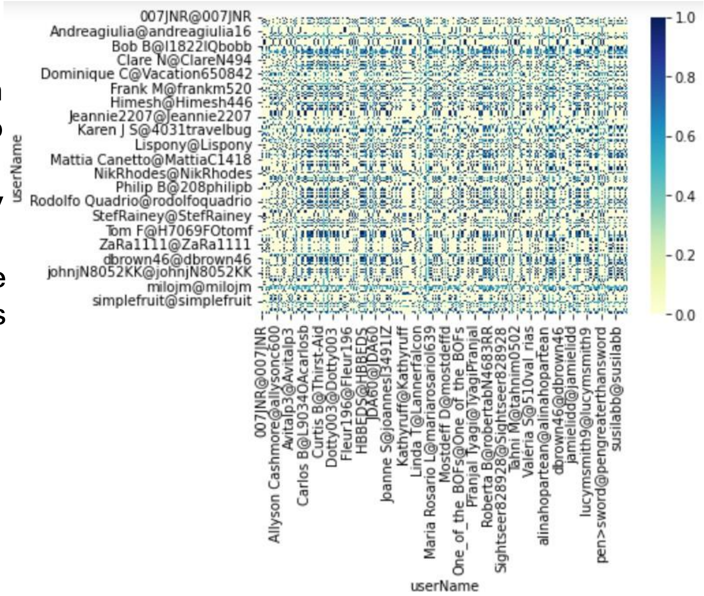
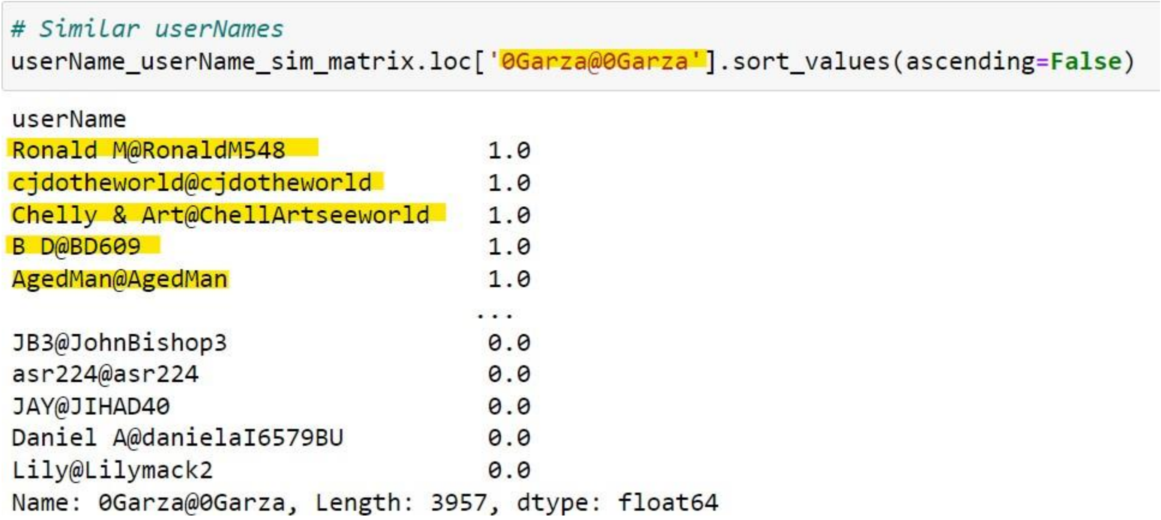


Fig. 7 Data Similarity results - overview

So the more ''blue'', meaning the closer to one, a userName will be with another, the more similar their preference for that specific Attraction and Rating.

The following analysis was to pick a specific userName and find within the Matrix, which other userNames were more similar in terms of Attraction visited and rating.

The data showed the ones marked in yellow, as the top 5. With this information, we are able to recommend to visitors, some attractions that they have not visited but other userNames with high similarity to them visited.



```
# Similar userNames
userName_userName_sim_matrix.loc['0Garza@0Garza'].sort_values(ascending=False)
```

```
userName
Ronald M@RonaldM548                1.0
cjdotheworld@cjdotheworld          1.0
Chelly & Art@ChellArtseeworld      1.0
B D@BD609                          1.0
AgedMan@AgedMan                    1.0
                                  ...
JB3@JohnBishop3                    0.0
asr224@asr224                      0.0
JAY@JIHAD40                        0.0
Daniel A@danielaI6579BU            0.0
Lily@Lilymack2                     0.0
Name: 0Garza@0Garza, Length: 3957, dtype: float64
```

Fig. 8 Similar userNames

After we got concrete data about our analysis, we decided to generate another Matrix that would find the similarities patrons between the Attractions.

It shows which Attractions are more similar to one another based on the rating that the users have given them. So, to userNames who have visited for example ''Torre de Belém'' will be most likely visiting ''Mosteiro dos Jerónimos'' if we offer it to them.

This information is indispensable to share with our client because they will know which audience to target and what Attraction to offer them to reactive tourism in Portugal.

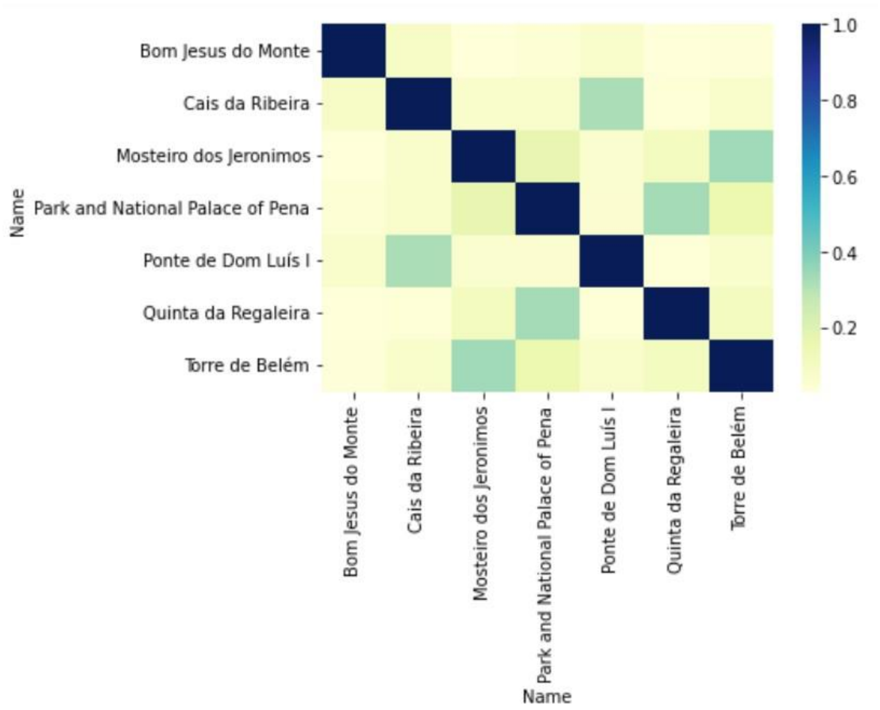


Fig. 9 Similar attractions based on rating.

Since the analysis before was focused just on Portugal Attractions, we decided to analyze the data, taking a different approach. After merging the data this time, we drop all the Attractions in which the global rating was not 5. So, we took just the Attraction with the highest global rating score.

As a result, we realized that ''Mezquita Cathedral de Cordoba '' had 1.050 visits with a global rating of 5. While the country with the greatest number of visits was Spain with
1.296 visitors.

After creating the Matrix with these characteristics, it resulted in a Matrix of 4.084 userNames and 10 global Attractions ratted with the score 5. And we picked up a heatmap to represent it.

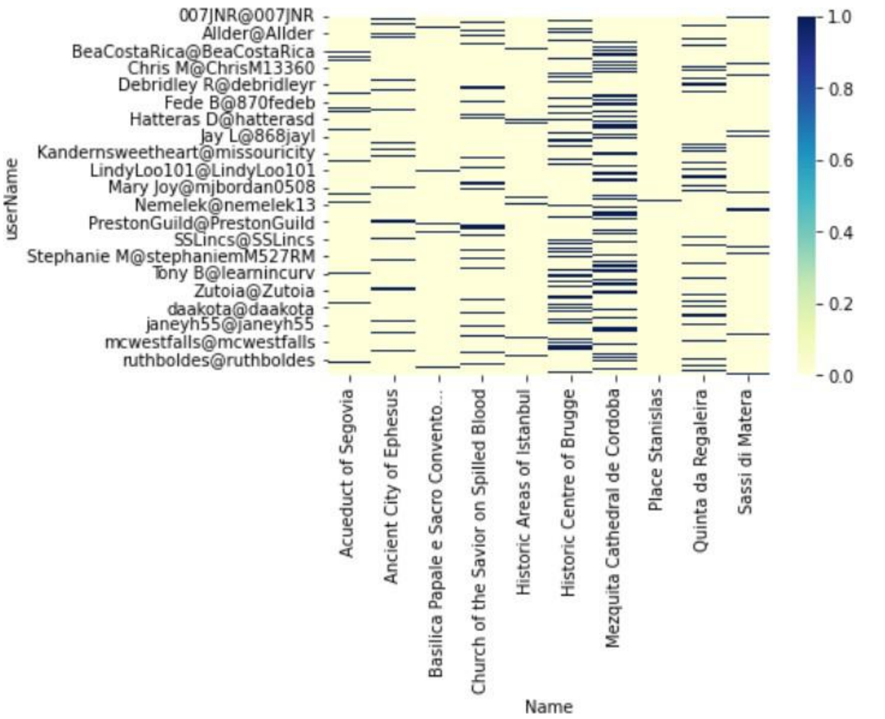


Fig. 10 Heatmap - Representation of Matrix

We wanted then to analyse the userNames attractions visits, so we created a userName to userName similarity matrix, which shows us how alike are to one of the Attractions that were scored with a 5 in the review rating.

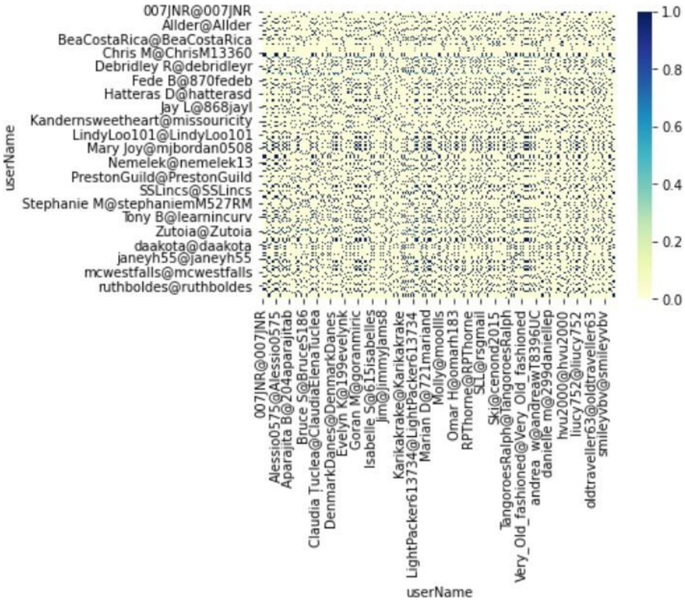


Fig. 11 Heatmap - Similarity Matrix on Attractions with 5 rating review.

We did the same analysis based on Attraction Name Similarity to know which Attractions in the global rating were more similar to other that also were rating with a 5.

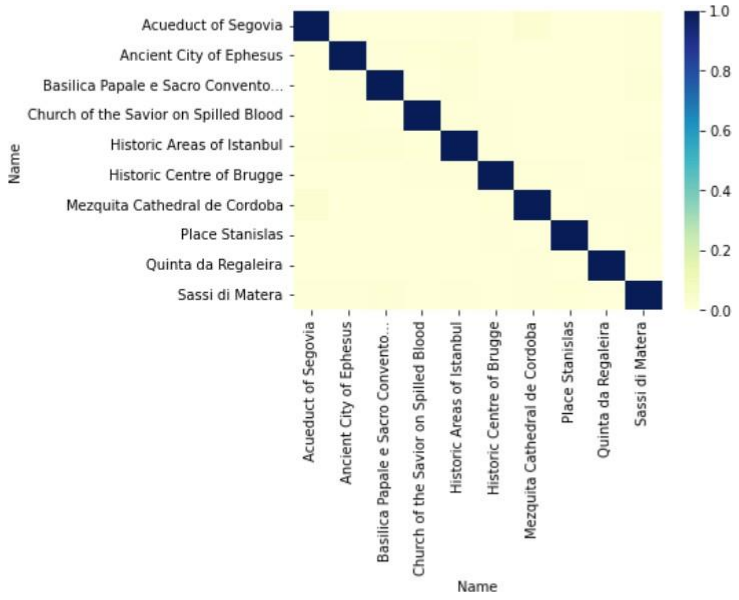


Fig. 12 Heatmap - Attraction Name Similarity

5.2. Market Basket Analysis

The Market Basket Analysis is a Data Mining technique used most frequently by retailers, to boost revenue. This is done by using certain algorithms to analyze large sets of data to understand your consumer's behavior when it comes to purchasing trends or patterns.

By doing so, you are then able to determine the advantages of grouping certain products together, as they are most likely to be purchased together.

In this report we focused on the Apriori algorithm, we used this algorithm to understand the attractions which are most frequently visited together to create tourist packages and understand which demographic to target. Since we are providing recommendations to the Portuguese tourism board, in our market basket analysis we focused solely on the Portugal data.

To do this we started by filtering our data set to show only the Portugal data by using the following code:

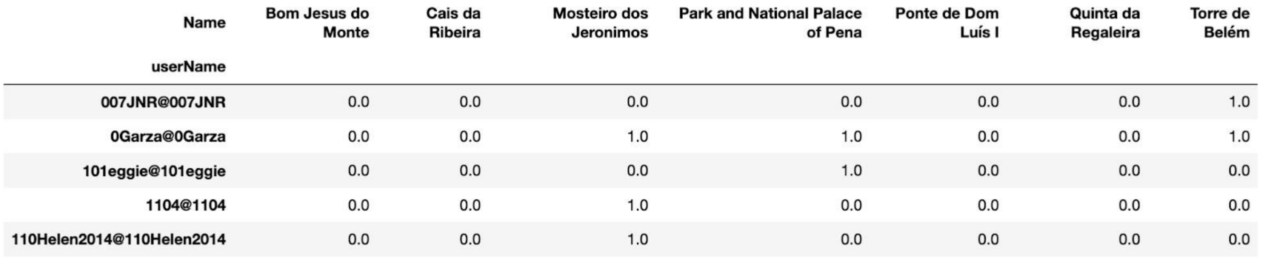
```
countries = ['Portugal']

PT = ds_com[ds_com.Country.isin(countries)]
```

We then needed to create a pivot table, where we needed to decide which items to analyze together and apply the algorithm. In the first analysis we decided to use the users and attractions, and created the following table:

```
pt_table = pd.pivot_table(PT[['userName','Name']],
                    index= 'userName',
                     columns= 'Name',
                    aggfunc=lambda x: 1 if len(x)>0 else 0).fillna(0)
pt_table.head()
```

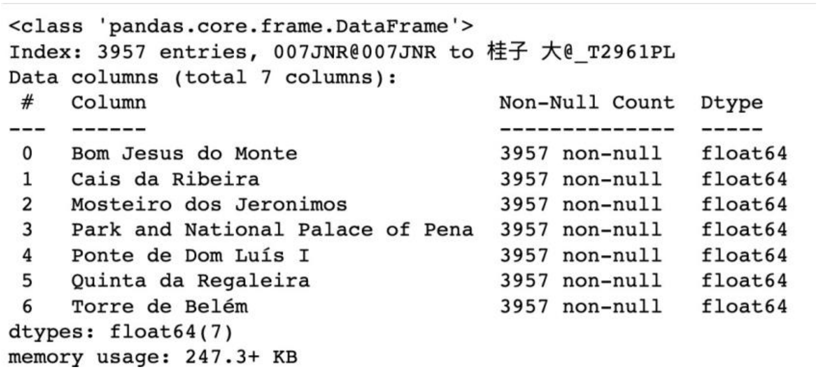
Once we ran the code above this is the table we got:



Name	Bom Jesus do Monte	Cais da Ribeira	Mosteiro dos Jeronimos	Park and National Palace of Pena	Ponte de Dom Luís I	Quinta da Regaleira	Torre de Belém
userName							
007JNR@007JNR	0.0	0.0	0.0	0.0	0.0	0.0	1.0
0Garza@0Garza	0.0	0.0	1.0	1.0	0.0	0.0	1.0
101eggie@101eggie	0.0	0.0	0.0	1.0	0.0	0.0	0.0
1104@1104	0.0	0.0	1.0	0.0	0.0	0.0	0.0
110Helen2014@110Helen2014	0.0	0.0	1.0	0.0	0.0	0.0	0.0

Fig. 13 Results when running the code

pt_table.info()



```
<class 'pandas.core.frame.DataFrame'>
Index: 3957 entries, 007JNR@007JNR to 桂子 大@_T2961PL
Data columns (total 7 columns):
 #   Column                            Non-Null Count  Dtype  
---  ------                            --------------  -----  
 0   Bom Jesus do Monte                3957 non-null   float64
 1   Cais da Ribeira                   3957 non-null   float64
 2   Mosteiro dos Jeronimos            3957 non-null   float64
 3   Park and National Palace of Pena  3957 non-null   float64
 4   Ponte de Dom Luís I               3957 non-null   float64
 5   Quinta da Regaleira               3957 non-null   float64
 6   Torre de Belém                    3957 non-null   float64
dtypes: float64(7)
memory usage: 247.3+ KB
```

Fig. 14 Index

From the information above we can see that the table will have 3.957 entries of users and attractions, and 7 columns for the attractions.

We then needed to run the Apari algorithm to start running the analysis, using the following code:

```
# Rules supported in at least 5% of the transactions
frequent_itemsets = apriori(pt_table, min_support=0.03, use_colnames=True)
```

Once we have applied the algorithm and created a dataset with the algorithm for frequent item sets, we can apply various combinations and group them by association rules.

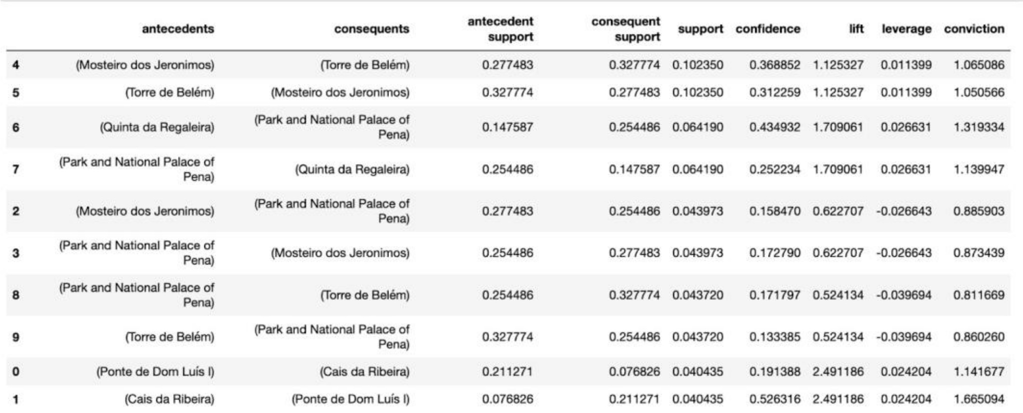
We started with generating an association rule by support. We first tried setting the minimum threshold to 0.10, however, we did not get enough outputs to generate a decent analysis. Therefore, we lowered the minimum threshold to 0.03, we did this using the following code:

```
rulesSupport = association_rules(frequent_itemsets, metric="support",
          min_threshold=0.03)

          rulesSupport.sort_values(by='support', ascending=False, inplace=True)

          rulesSupport.head(10)
```

By running this code this is the combinations we achieved:



	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
4	(Mosteiro dos Jeronimos)	(Torre de Belém)	0.277483	0.327774	0.102350	0.368852	1.125327	0.011399	1.065086
5	(Torre de Belém)	(Mosteiro dos Jeronimos)	0.327774	0.277483	0.102350	0.312259	1.125327	0.011399	1.050566
6	(Quinta da Regaleira)	(Park and National Palace of Pena)	0.147587	0.254486	0.064190	0.434932	1.709061	0.026631	1.319334
7	(Park and National Palace of Pena)	(Quinta da Regaleira)	0.254486	0.147587	0.064190	0.252234	1.709061	0.026631	1.139947
2	(Mosteiro dos Jeronimos)	(Park and National Palace of Pena)	0.277483	0.254486	0.043973	0.158470	0.622707	-0.026643	0.885903
3	(Park and National Palace of Pena)	(Mosteiro dos Jeronimos)	0.254486	0.277483	0.043973	0.172790	0.622707	-0.026643	0.873439
8	(Park and National Palace of Pena)	(Torre de Belém)	0.254486	0.327774	0.043720	0.171797	0.524134	-0.039694	0.811669
9	(Torre de Belém)	(Park and National Palace of Pena)	0.327774	0.254486	0.043720	0.133385	0.524134	-0.039694	0.860260
0	(Ponte de Dom Luís I)	(Cais da Ribeira)	0.211271	0.076826	0.040435	0.191388	2.491186	0.024204	1.141677
1	(Cais da Ribeira)	(Ponte de Dom Luís I)	0.076826	0.211271	0.040435	0.526316	2.491186	0.024204	1.665094

Fig. 15 Combinations between attractions

To understand better the results in a more visually appealing way, we decided to illustrate the results using a scatter graph.

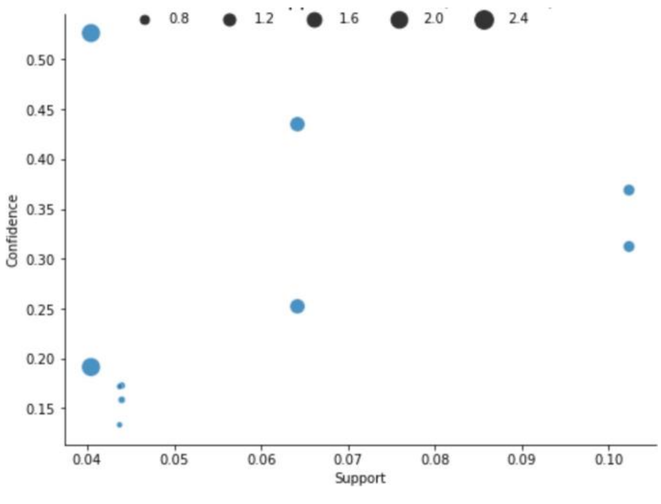


Fig. 16 Rules with support above 3%

From this graph we can see that the confidence is high and the support is also matching the confidence.

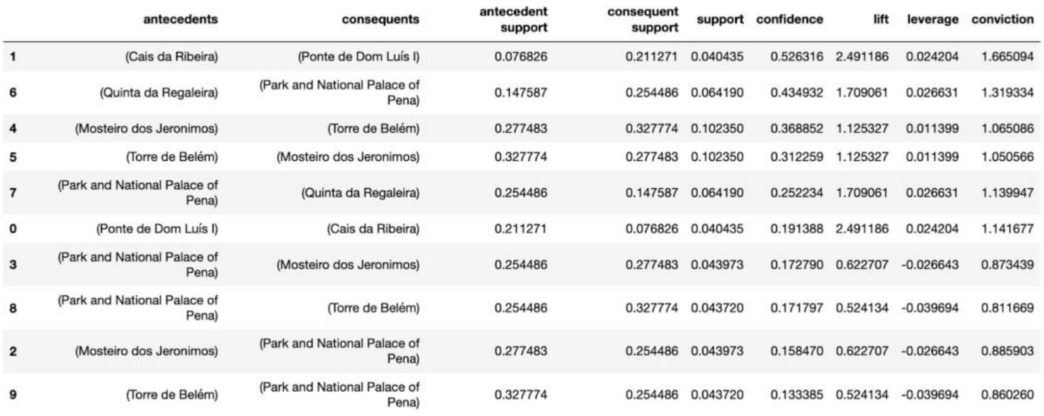
We then decided to group the data by association rule of confidence, setting the minimum threshold to once again 0.03. We did so by using the following code:

```
rulesConfidence = association_rules(frequent_itemsets, metric="confidence",
            min_threshold=0.03)

            rulesConfidence.sort_values(by='confidence', ascending=False,
inplace=True)

            rulesConfidence.head(10)
```

These are the results we got:



	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
1	(Cais da Ribeira)	(Ponte de Dom Luís I)	0.076826	0.211271	0.040435	0.526316	2.491186	0.024204	1.665094
6	(Quinta da Regaleira)	(Park and National Palace of Pena)	0.147587	0.254486	0.064190	0.434932	1.709061	0.026631	1.319334
4	(Mosteiro dos Jeronimos)	(Torre de Belém)	0.277483	0.327774	0.102350	0.368852	1.125327	0.011399	1.065086
5	(Torre de Belém)	(Mosteiro dos Jeronimos)	0.327774	0.277483	0.102350	0.312259	1.125327	0.011399	1.050566
7	(Park and National Palace of Pena)	(Quinta da Regaleira)	0.254486	0.147587	0.064190	0.252234	1.709061	0.026631	1.139947
0	(Ponte de Dom Luís I)	(Cais da Ribeira)	0.211271	0.076826	0.040435	0.191388	2.491186	0.024204	1.141677
3	(Park and National Palace of Pena)	(Mosteiro dos Jeronimos)	0.254486	0.277483	0.043973	0.172790	0.622707	-0.026643	0.873439
8	(Park and National Palace of Pena)	(Torre de Belém)	0.254486	0.327774	0.043720	0.171797	0.524134	-0.039694	0.811669
2	(Mosteiro dos Jeronimos)	(Park and National Palace of Pena)	0.277483	0.254486	0.043973	0.158470	0.622707	-0.026643	0.885903
9	(Torre de Belém)	(Park and National Palace of Pena)	0.327774	0.254486	0.043720	0.133385	0.524134	-0.039694	0.860260

Fig. 17

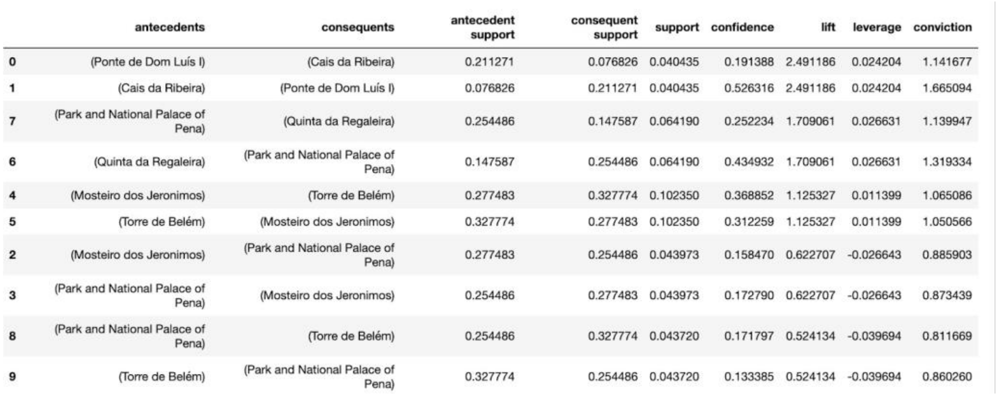
We then dedicated to run the code of the associating by the lift rule. We used the following code:

```
rulesLift = association_rules(frequent_itemsets, metric="lift", min_threshold=0.5)

            rulesLift.sort_values(by='lift', ascending=False, inplace=True)

            rulesLift.head(10)
```

After we ran this code, we got the following results:



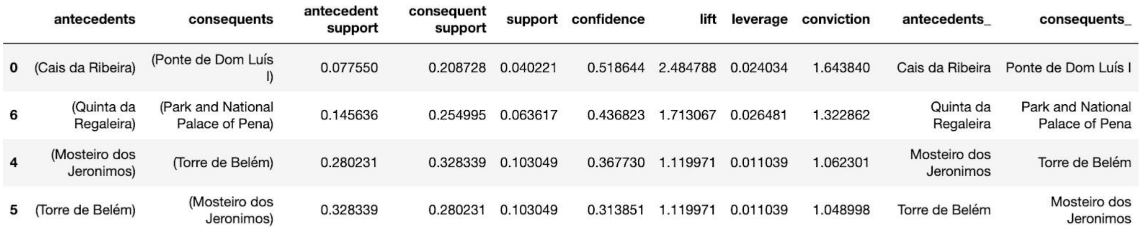
	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
0	(Ponte de Dom Luís I)	(Cais da Ribeira)	0.211271	0.076826	0.040435	0.191388	2.491186	0.024204	1.141677
1	(Cais da Ribeira)	(Ponte de Dom Luís I)	0.076826	0.211271	0.040435	0.526316	2.491186	0.024204	1.665094
7	(Park and National Palace of Pena)	(Quinta da Regaleira)	0.254486	0.147587	0.064190	0.252234	1.709061	0.026631	1.139947
6	(Quinta da Regaleira)	(Park and National Palace of Pena)	0.147587	0.254486	0.064190	0.434932	1.709061	0.026631	1.319334
4	(Mosteiro dos Jeronimos)	(Torre de Belém)	0.277483	0.327774	0.102350	0.368852	1.125327	0.011399	1.065086
5	(Torre de Belém)	(Mosteiro dos Jeronimos)	0.327774	0.277483	0.102350	0.312259	1.125327	0.011399	1.050566
2	(Mosteiro dos Jeronimos)	(Park and National Palace of Pena)	0.277483	0.254486	0.043973	0.158470	0.622707	-0.026643	0.885903
3	(Park and National Palace of Pena)	(Mosteiro dos Jeronimos)	0.254486	0.277483	0.043973	0.172790	0.622707	-0.026643	0.873439
8	(Park and National Palace of Pena)	(Torre de Belém)	0.254486	0.327774	0.043720	0.171797	0.524134	-0.039694	0.811669
9	(Torre de Belém)	(Park and National Palace of Pena)	0.327774	0.254486	0.043720	0.133385	0.524134	-0.039694	0.860260

Fig. 18

In order to better understand the combinations and know which attractions could potentially be grouped together in packages, we decided to group the data by a high lift and high confidence. To do so we used the following code:

```
rulesConfidence[(rulesConfidence['confidence'] >= 0.3) &  (rulesConfidence['lift']  >=
          1)]
```

After running this code these are the results we saw:



	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction	antecedents_	consequents_
0	(Cais da Ribeira)	(Ponte de Dom Luís I)	0.077550	0.208728	0.040221	0.518644	2.484788	0.024034	1.643840	Cais da Ribeira	Ponte de Dom Luís I
6	(Quinta da Regaleira)	(Park and National Palace of Pena)	0.145636	0.254995	0.063617	0.436823	1.713067	0.026481	1.322862	Quinta da Regaleira	Park and National Palace of Pena
4	(Mosteiro dos Jeronimos)	(Torre de Belém)	0.280231	0.328339	0.103049	0.367730	1.119971	0.011039	1.062301	Mosteiro dos Jeronimos	Torre de Belém
5	(Torre de Belém)	(Mosteiro dos Jeronimos)	0.328339	0.280231	0.103049	0.313851	1.119971	0.011039	1.048998	Torre de Belém	Mosteiro dos Jeronimos

Fig. 19

From this we were able to understand that tourists who come to Portugal will most likely visit the following attractions together:

- Cais da Ribeira and Ponte de Dom Luís I, for this analysis we will call Porto.
- Quinta da Regaleira and Park and National Palace of Pena, Sintra.
- Mosteiro dos Jerónimos and Torre de Belém, Belém.

Therefore, we can assume that these are good combinations to sell day trips for, create joint tickets and advertise together for people who are visiting other locations and did not have one of the other attractions planned for their trip.

However, to suggest other attractions to tourists it's important to know what connection lies between the two packs, so to do this we decided to run one more analysis to show us the network of these attractions.

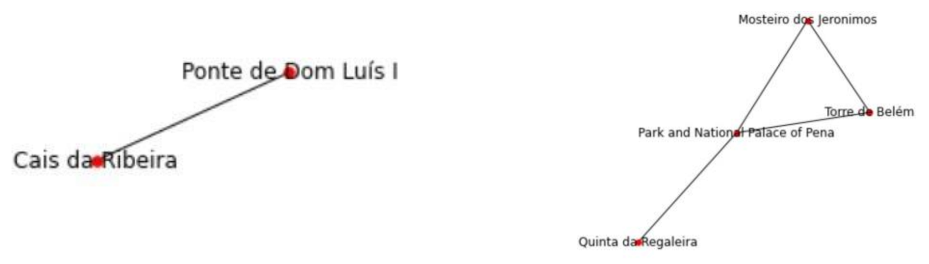
We ran the network analysis with the following code:

```
# Create a copy of the rules and transform the frozensets to strings
rulesToPlot = rulesConfidence.copy(deep=True)
rulesToPlot['LHS'] = [','.join(list(x)) for x in rulesToPlot['antecedents']]
rulesToPlot['RHS'] = [','.join(list(x)) for x in rulesToPlot[‘consequents’]]


# Remove duplicate if reversed rules
rulesToPlot['sortedRow'] = [sorted([a,b]) for a,b in zip(rulesToPlot.LHS,
          rulesToPlot.RHS)]
rulesToPlot['sortedRow'] = rulesToPlot['sortedRow'].astype(str)
rulesToPlot.drop_duplicates(subset=['sortedRow'], inplace=True)


# Plot
rulesToPlot=rulesToPlot[:5]
fig = plt.figure(figsize=(15, 15))
G = nx.from_pandas_edgelist(rulesToPlot, 'LHS', 'RHS')
np.random.seed(1234)
nx.draw(G, with_labels=True, node_size=30, node_color="red",
          pos=nx.spring_layout(G))
plt.axis('equal')
plt.show()
```

After running the code we got the following network:



After analyzing this network we can see that despite not having a direct correlation between the Belém combination and the Sintra combination, there is a link between the Sintra nodes and Belém nodes. Meaning that a percentage of tourists that go to Belém will also to Sintra, therefore you can create discounts such as when you go to

Belém you get a discount for when you go visit Sintra, so that tourists are inclined to go visit Sintra, because it will cost them less.

Another option would be to create a package which creates a link with Porto, therefore, a 4-day package could be created to include all 3 combinations. For instance, you could create a package which would give you tours to Belém and Sintra, on two consecutive days and then a night in Porto for a day trip to the attractions in the Porto combination.

5.3. RFM Analysis

After understanding and preparing the data it's time to identify which metrics we will work with to deliver the RFM Analysis. Our objective in this analysis is to identify segments of users and to impact them at different moments: some are totally active so we can focus on repurchases(new travels), others are at risk so we need to retain them and the worst scenarios are the ones we have to reactivate, so there is a lot of time that passed since they travelled, they evaluate negatively the trips and they hardly ever travel.

Recency = reviewVisited - date - date when the customer visited the attraction. The day is always 1 because Tripadvisor only asks users to describe the year and the month, not the day.

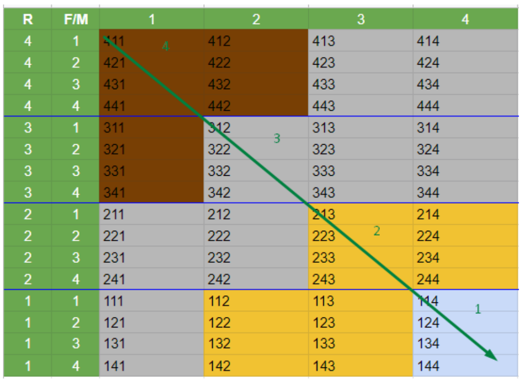
Frequency = userContributions - numeric - how many reviews has the user written in TripAdvisor at the time of the extraction of the review.

Monetary = reviewRating - numeric - quantitative rating assigned by the user (1 star - bad to 5 stars - excellent).
userName - string - user name of the TripAdvisor user who posted the review. = to identify and group the metrics by it.

We have created different segments to analyze and compare them using three types of data visualisation. So, we can get the similarities and differences about the users.

PT = attractions from Portugal
Basket Analysis 1 = which filters attractions from the first set of combinations in the results of the market Basket Analysis, results named Porto.
Basket Analysis 2 = which filter attractions from the second set of combinations from the results of the market basket analysis, results named Belém.
Basket Analysis 3 = which filter attractions from the third set of combinations from the results of the market basket analysis, results named Sintra.

The first chart is the TreeMap, which helps us understand the relation between the metrics (RFM) and how relevant each segment created by RFM is, by the number of users. To define the colors we used this reference to classify the segments:

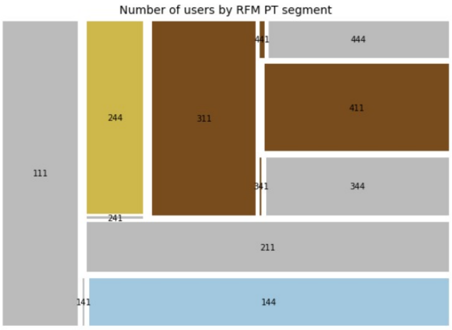


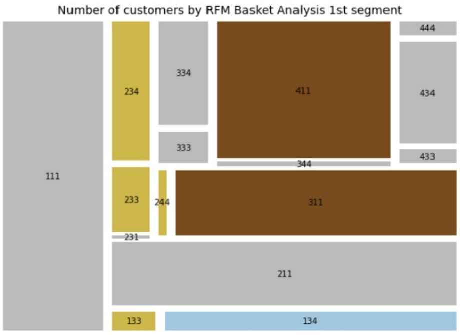
R	F/M	1	2	3	4
4	1	411	412	413	414
4	2	421	422	423	424
4	3	431	432	433	434
4	4	441	442	443	444
3	1	311	312	313	314
3	2	321	322	323	324
3	3	331	332	333	334
3	4	341	342	343	344
2	1	211	212	213	214
2	2	221	222	223	224
2	3	231	232	233	234
2	4	241	242	243	244
1	1	111	112	113	114
1	2	121	122	123	124
1	3	131	132	133	134
1	4	141	142	143	144

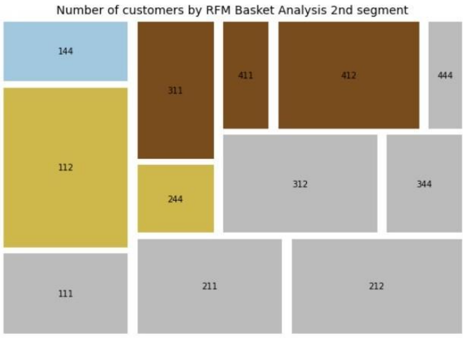
1./2. They are the most loyal ones (best customers): users take a few days between the trips, frequently give reviews, and give great grades to places visited. We can impact them more with general places in the PT segment, the ones most visited would be a good offer as they are more likely to travel compared to other segments. In addition, in Basket Analysis, which focuses on Portugal's attractions, pairs are great segments to explore to promote a "cross-visit" to the one who visited one of the places of the basket analyzed.

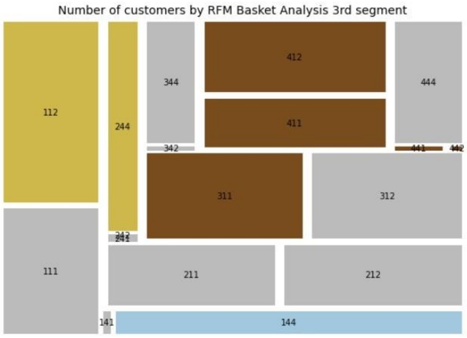
3. They're the ones who can contribute more frequently to reviews but take more time to travel or are not evaluating very well. Or otherwise, they comment less, so we work to retain them and pay attention to which metrics we can improve. They are the most common ones in the data analyzed, we suggest focusing on the metric that is lower (FM) or higher in Recency to promote this improvement and balance the experience.

4. They are the risk ones, they take a lot of time to visit new attractions, it is not usual for them to give reviews or/and evaluate badly the attractions where they went. Our main recommendation here is to work with aggressive promotions and make sure that all the experience will be great. In the end, invite them to evaluate the trip and maybe give a gift for that. In all segments the 3 metrics must be improved, as focusing on one of them as recency which is "3" in most of them is a good beginning, in the ones in which F or M are 2 is another option because these metrics are less affected in the segment.

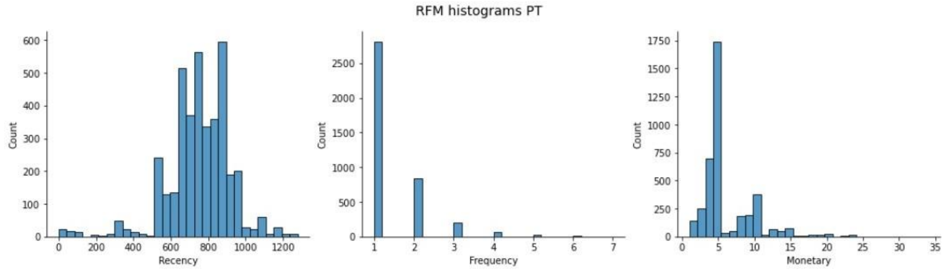


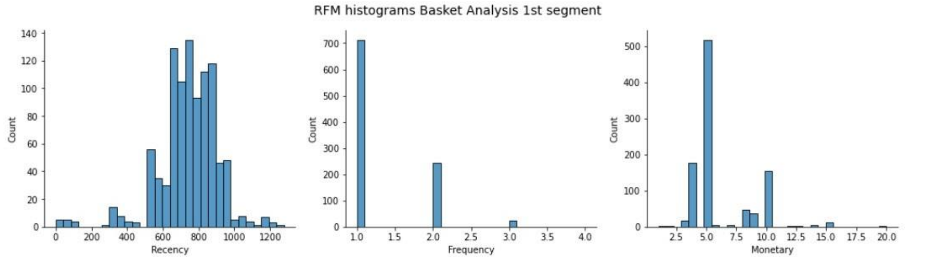


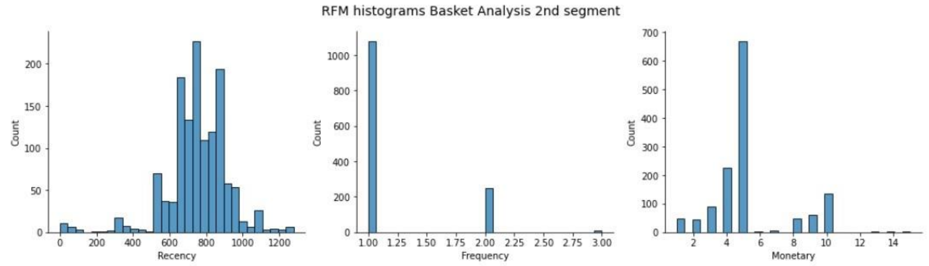


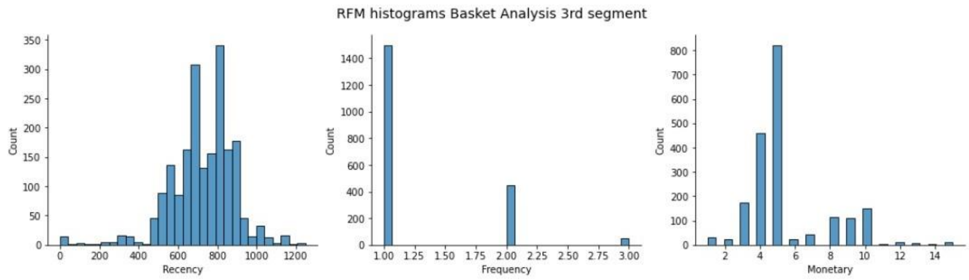


In this visual, we can observe that in general attractions from Portugal present more diversity in frequency of comments in reviews than the segments from the market basket analysis ones, most of the users commented only once. About the recency, this metric is more similar to all of them, the third market basket analysis presents the best recency as the users take less days, in addition they give better scores in monetary (amount of rating reviews). So, the third market basket analysis segment is an excellent segment to impact with cross-selling or up-selling with these attractions. The second market basket analysis shows a good distribution of frequency compared to the others, in the same way as monetary, so we could work to retain and explore them more with the attractions of the market basket analysis too, maybe with a campaign with a similar period before one of the travel baskets is done.









6. Evaluation and Deployment

All three methodologies showed different approaches to the same issue.

The Similarity Method showed that the most relevant analysis was the one in which we filtered the data, and we got the Attractions of Portugal compared with other similar Portuguese Attractions in a matrix - this is exactly the data NTBO will need to reactivate tourism after the pandemic.

Regarding the second analysis of the Similarity Method, it gave us a wider overview of Portugal's competitors so our client would know which attractions are better rated and will also have similar attractions with the highest ranges.

With this Similarity Method, we are able to suggest user's new destinations, based on similar tastes.

The Market Basket Analysis allowed us to identify three strong combinations of attractions which tourists visit during the same period in Portugal.

This analysis was the most challenging to perform, considering the tourism in Portugal and the dataset provided. Portugal has many different attractions but, unfortunately, TripAdvisor (the source of our dataset) does not have it all listed, making the Market Basket Analysis more limiting.

We conclude that this analysis is better suited for retail in terms of products sold as the combinations are higher (due to the quantity of data available), making it easier to identify patterns and trends, and able to provide better recommendations for revenue growth.

Perhaps if the data provided had a wider array of attractions, and more data on visitors to support the various attractions, we would be able to have more diverse combinations and identify more trends.