DATA MINING

Group 6 - Hotelbook

D4 - Final delivery

2023-2024 Q1

23/10/2023

Team Members: Gerard Álvarez
Tomás Calaf
Natalia Dai
Adrià Espinoza
Mario Martin

Index

1 - Motivation of the work

In this first part of the course, we were asked to find a database to analyze and practice the concepts explained in class. From finding the data and preprocessing it, to organizing it into clusters and doing a PCA analysis, we will cover all the basic points to analyze and prepare the data, as we would in a real case scenario. The aim is to learn the good practices and procedures that lead to good data and, therefore, good results.

As for our dataset, we used one about hotel bookings, with the objective of detecting the trends among the tourists, and determining the most common types of customers depending on the season. To sum up, our intention is to categorize all bookings into groups to provide us a better understanding about the different types of tourists present in our dataset.

2 - Data Source presentation

We obtained the data directly from the Kaggle website, but its original source was the article ***Hotel Booking Demand Datasets,*** by Nuno Antonio, Ana Almeida, and Luis Nunes that was published in the journal *Data in Brief* in February 2019. It was cleaned later on by Thomas Mock and Antoine Bichat. The URL is the following: https://www.kaggle.com/datasets/jessemostipak/hotel-booking-demand?select=hotel_bookings.csv

3 - Metadata

Our dataset consists of 32 columns. At the start of the study, each row has all the following variables mentioned in the metadata table. However, in the preprocessing process we got rid of some ones (explained in 5 - Preprocessing).

Variable Name	Variable Short Name	Meaning	Type	Measuring unit	Range	Role
Hotel		Describes the type of hotel	Cat			Explanatory
Is_canceled	can	Indicates whether the reservation was canceled	Binary			Response
lead_time	lt	The number of days elapsed between the making of the reservation and the arrival at the hotel	Num	days		Explanatory
arrival_date_year	arr_y	The year of the arrival date	Cat		[2015,2017]	Explanatory
arrival_date_month	arr_m	The month of the arrival date	Cat			Explanatory
arrival_date_week_number	arr_wn	The week number of a year of the arrival	Cat		[1,53]	Explanatory
arrival_date_day_of_month	arr_dm	The day of the month of the arrival	Cat		[1,31]	Explanatory
stays_in_week_nights	s_wn	Number of weekday nights booked or stayed	Num	nights		Explanatory
adults		Number of adults in the reservation	Num			Explanatory
children		Number of children in the reservation	Num			Explanatory
babies		Number of babies in the reservation	Num			Explanatory
meal		Type of meal chosen by the guest	Cat			Explanatory
country		Country of origin of the guest	Cat			Explanatory
market_segment	mkt_s	Market segment designation	Cat			Explanatory
distribution_channel	dst_ch	Booking distribution channel	Cat			Explanatory

previous_cancellations	prev_c	Number of previous cancellations by the guest	Num			Explanatory
previous_bookings_not_canceled	pb_nc	Number of previous bookings not canceled by the guest	Num			Explanatory
reserved_room_type	res_rt	Code of room type reserved	Cat			Explanatory
assigned_room_type	as_rt	Code for the type of room assigned to the booking	Cat			Explanatory
booking_changes	book_ch	Number of changes/amendments	Num			Explanatory
deposit_type	dp_t	Indication on if the customer made a deposit	Cat			Explanatory
agent		ID of the travel agency that made the booking	Cat			Explanatory
company		ID of the company/entity that made the booking or responsible for paying the booking	Cat			Explanatory
days_in_waiting_list	d_wl	Number of days the booking was in the waiting list before it was confirmed to the customer	Num			Explanatory
customer_type	ct	Type of booking	Cat			Explanatory
adr		Average Daily Rate	Num	daily rate		Explanatory
required_car_parking_spaces	prk_s	Number of car parking spaces required by the customer	Num			Explanatory
total_of_special_requests	sp_req	Number of special requests made by the customer	Num			Explanatory
reservation_status	res_s	Reservation last status	Cat			Explanatory
reservation_status_date	res_sd	Date at which the last status was set.	Date			Explanatory

4 - Data Mining process

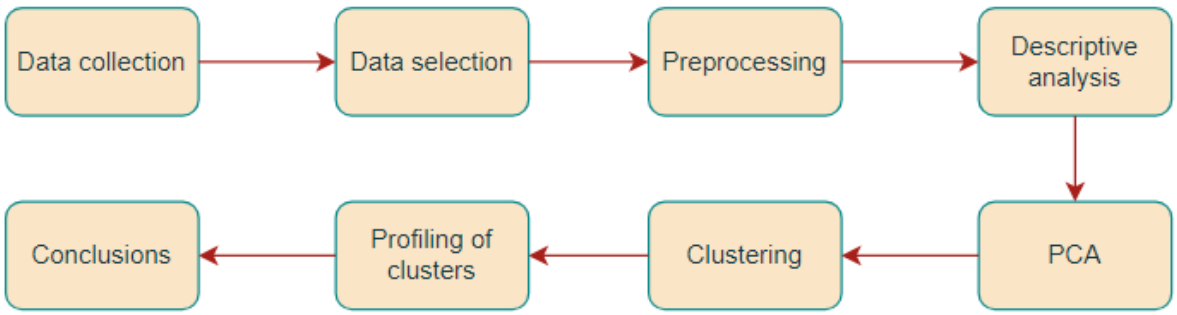
Our project's data mining process starts by selecting a dataset from Kaggle which fits the criteria of this project.

Once the database has been chosen, we proceed to visualize its contents, which have 32 columns and more than 119k rows. We then opted by taking a sample from it of only 5k individuals. In the process of data cleaning and preprocessing we decided to simplify the database and consider only the 17 most significant variables (detailed reasons explained in section 5.2). Since we have not got any missing data, we managed to create them in order to apply the techniques studied in class.

Once the dataset preprocessing is complete, we conducted a descriptive analysis (univariate and bivariate) of the variables and performed the PCA process, in order to assess data separability.

Finally, we did the hierarchical clustering process of our individuals to determine the potential groupings and the profiling of them, so that we could analyze them individually.

This entire process is summarized in the following workflow:



5 - Preprocessing

5.1. Formatting issues

We did not get any issue regarding the format of our database. We used the command *read.csv()* in order for RStudio to be able to read it properly.

5.2. Determining working matrix, filtering

Since the original database had more than 119.000 rows, with the help of RStudio we managed to get a sample of 5000 individuals only:

```
database <- read.csv("C:/Users/Natal/OneDrive/Documents/MD/hotel_bookings.csv",header=T);
random <- database[sample(nrow(database)), ]
dd <- random[1:5000, ]
write.csv(dd, file = "C:/Users/Natal/OneDrive/Documents/MD/dd.csv", row.names = FALSE)
```

The next step is to extract the unwanted columns with more than 70% NULL data. In this case we had a variable named “company”, being the ID of the company/entity that made the booking or responsible for paying the booking, which had 94% of NULL data. It is reasonable for a company to not give more information than necessary so we ruled it out as a non-random missing value. For the “agent” variable we also had some NULL values, and the categories that were not NULL didn’t give us much information as they were very generic to protect the identity and private information of those agents, so we decided to rule it out as well.

There are other variables that do not seem to contribute much information and were just creating more noise so we erased them and kept this dataset with 17 variables in total. Moreover, those variables were not being really explicative about what they were representing, so it was not understandable. Also, it reduces the complexity of the academic exercise.

5.3. Creation of the missing values

Firstly our dataset did not have any missing values, so we have decided to create them for 6 different variables. We have added them in 3 categorical variables and 3 numericals with a representation of 5% of the total rows (250 missing values per column).

The columns that we have chosen to create missing values for are *arrival_date_week_number* , *meal* and *country* for categoricals and lead_time , *days_in_waiting_list and adr* for the numericals.

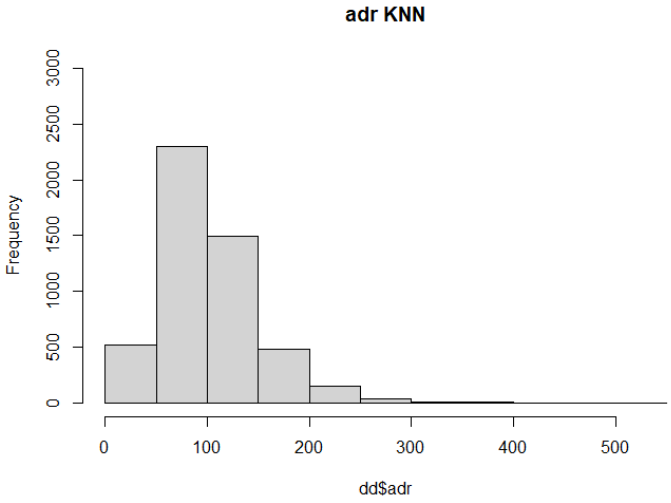
5.4. Identification and treatment of missing data

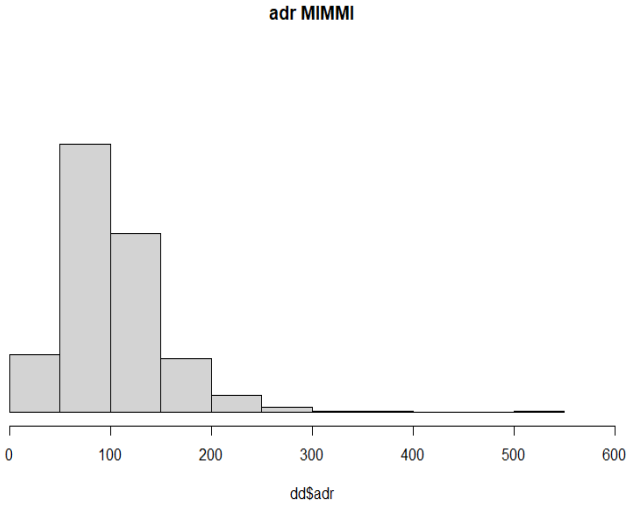
Column “arrival_date_week_number” had some missing values that were non-random and could be deduced by applying logical criteria, so we developed a script to fix them by calculating the number based on columns “arrival_date_month”, “arrival_date_year” and “arrival_date_day_of_month”.

For columns “country” and “meal” we also had some missing values marked as NA, and we had to create a new category “Unknown”, since we are unable to apply any logical reasoning to deduce the origin or preferred meal of the guests.

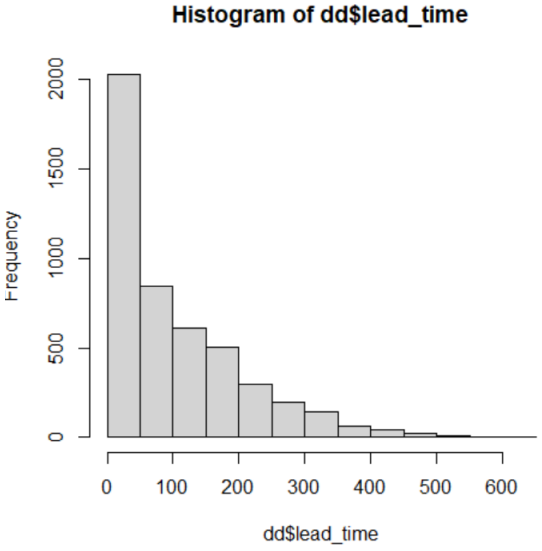
For the numerical values, we used the imputation methods seen in class. As we don’t know beforehand which one will work the best, we started by plotting the histograms for the 3 variables containing the missing values. We then applied both the KNN and MIMMI methods to calculate the missing values, and plotted again the histograms to compare which method better follows the same trend than the original values. After plotting the histograms we can see that for *adr* the best method is KNN, while for *lead_time* and *days_in_waiting_list* is MIMMI, so we generated the final version of our dataset with those methods to obtain a dataset with no missing values.

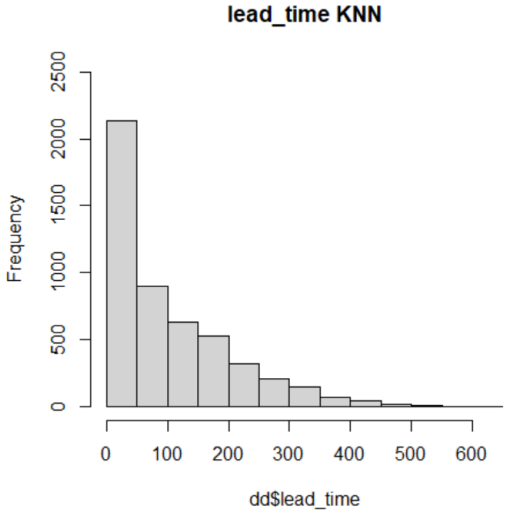
- adr -> KNN

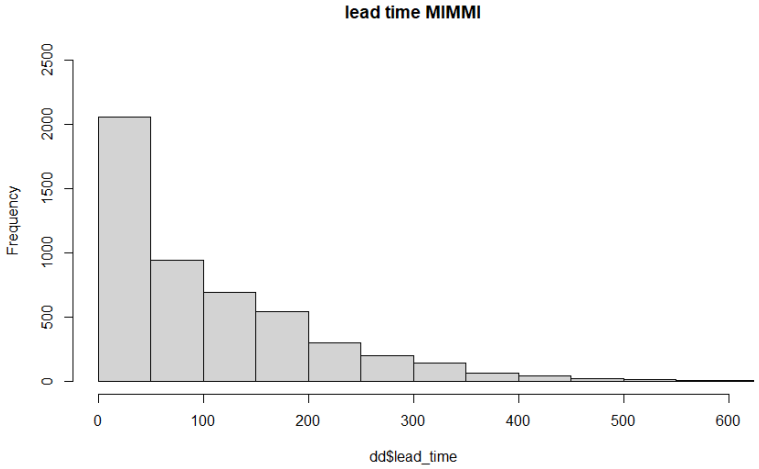




- lead_time -> MIMMI

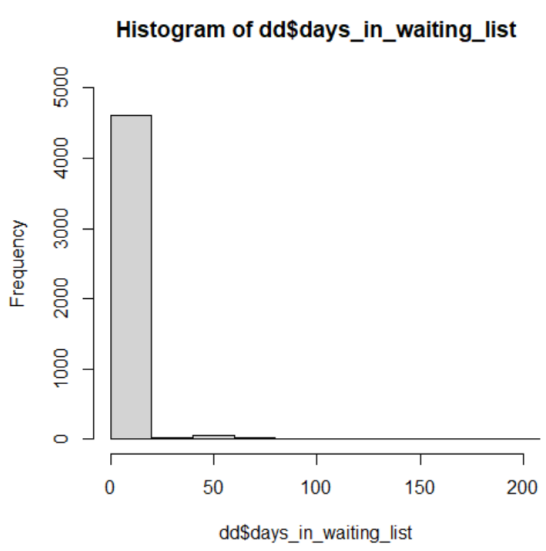


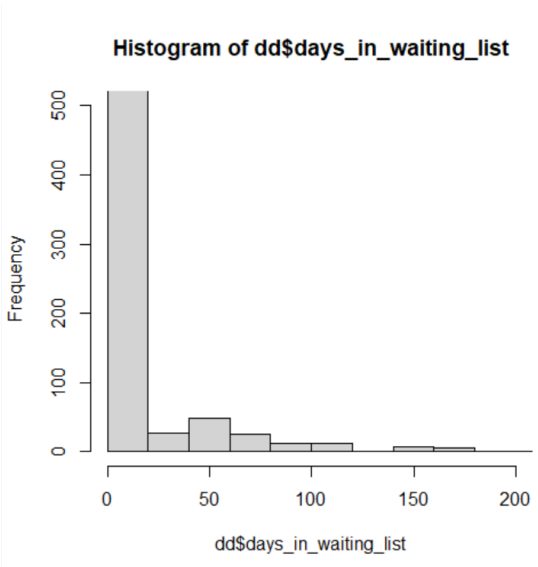


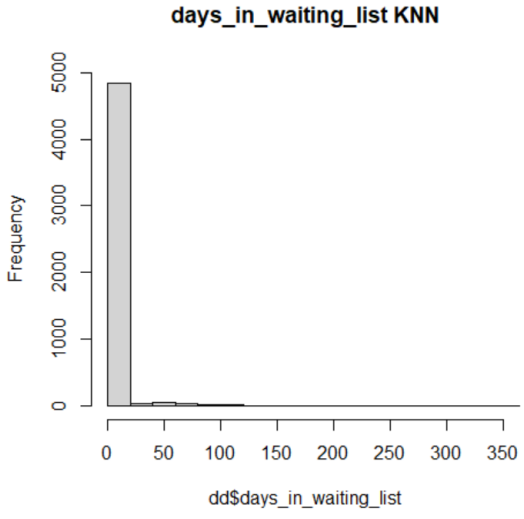


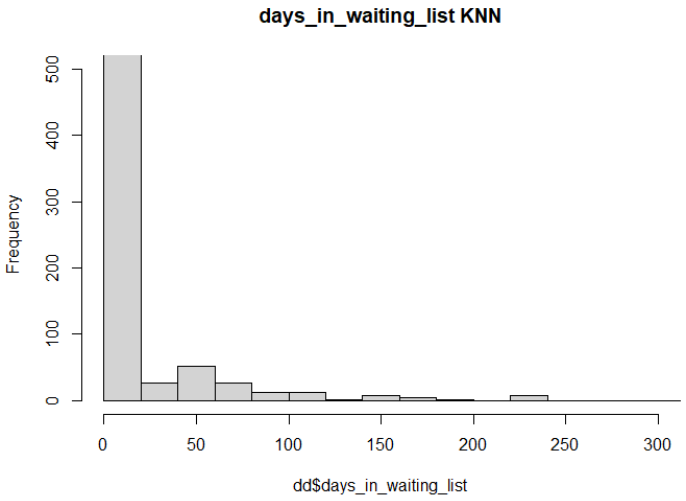
- days_in_waiting_list -> MIMMI

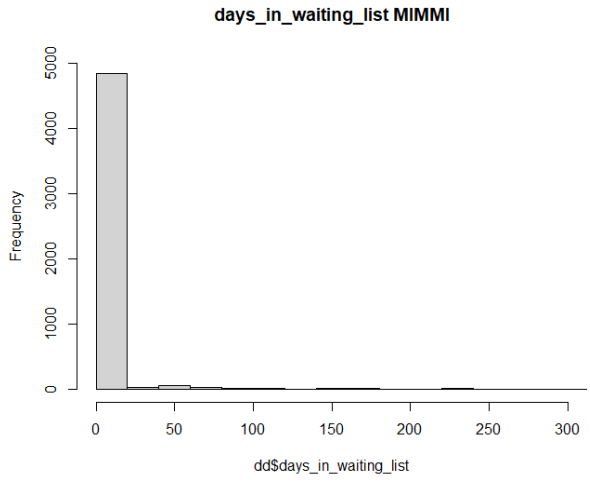
After viewing in the first histogram that the other values are not that visible compared to the first bar, we made a zoomed in version so that we can take them more into account.

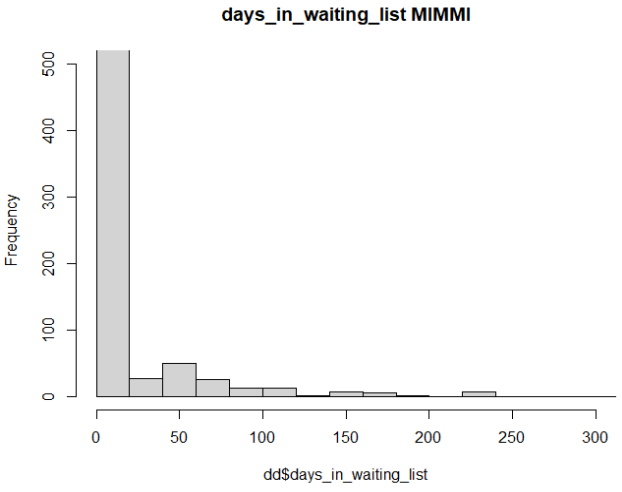












5.5. Identification and treatment of outliers

We have detected a few outliers during our univariate analysis. The first one is in the variable *stays_in_weekend_days* where we encountered a client that stayed 16 weekend nights, while all other customers didn't stay more than one or two weekends. We will leave this value as it could just be a customer that had to stay for a longer period of time for whatever reason.

In *days_in_waiting_list* we have found another very distinct outlier, since there are always almost no days of waiting to book a hotel, but one has a value of 391. We believe, even if it is an extremely high value compared to the others, that it could be a rare case and, therefore, it could add some value to our analysis. For this reason, we will keep it in the data.

Finally, in the *adr* variable, we have another outlier at 508, that could be due to a higher number of customers or price of the rooms as mentioned before. Therefore, we will not remove it and assume that it is just another value of the population.

5.6. Identification and treatment of errors

The identification of the errors and their treatment is so important to not make our analysis useless. So after reviewing the different variables we have found that one booking has stayed 391 days in the waiting list.

From our point of view, being on the waitlist for more than one year when you are booking a hotel does not make sense. However, if we delete that instance our database would be modified. In the next table we can see that the instance mentioned has a high repercussion on the mean, making that elimination a significant change on the mean if we take in mind that it is only one instance out of 5000.

BEFORE REMOVING

MIN	**1st Qu.**	**Median**	**Mean**	**3rd Qu.**	**MAX**
0	0	0	2.557	0	391

AFTER REMOVING

MIN	1st Qu.	Median	Mean	3rd Qu.	MAX
0	0	0	2.479	0	236

In conclusion, if we delete the outliers, our database would lose valuable and potential information for our final analysis, which would result in a reduction of the reliability of our results.

5.7. Instance selection

As described in section *5.6. Identification and treatment of errors* we have decided not to remove the instance errors since we would lose information that would be useful for our final analysis.

We can consider that our data is now perfect, so we will not apply sampling that could introduce unnecessary randomness and may lead to loss of information. We are confident of the quality of our dataset and we decided to work with our full dataset to maintain its integrity and minimize the risk of introducing bias or distorsions.

5.8. Data transformation

While we haven't really transformed the data so far, we have had to add artificial missing values as our database didn't contain any, so that we could practice the imputation techniques. Also, we have shortened the names of the variables for them to not to be that long, for the visibility of our graphs.

5.9. Derivation of new variables

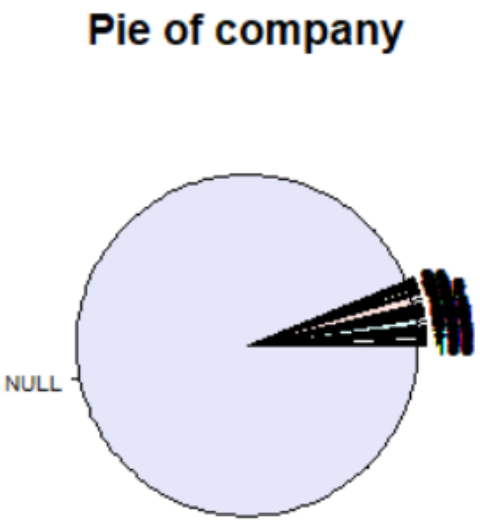
We have not added any new values, unless we count the ones generated by MIMMI (the index and the cluster number). But none meaningful to our analysis.

6 - Basic statistical descriptive analysis

6.1. Univariate descriptive statistics

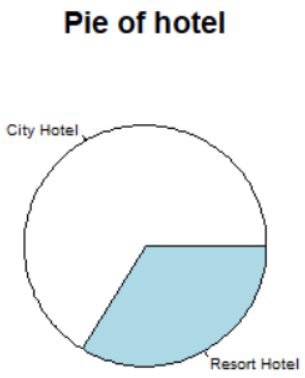
With the script provided on the web we managed to get the following results:

Variable company



We displayed this variable as well to show that there is indeed a lot of missing information. Here we have 107 modalities and hence why the table of frequencies and proportions is simply too big to be of importance. However, we can see that almost all the data is NULL (94%).

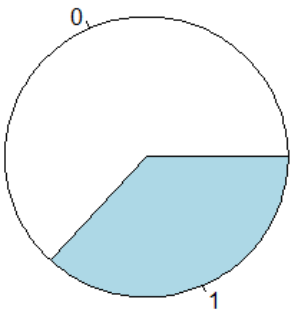
Variable 1: hotel



We can see that there are 2 modalities, the two different types of hotels. Also, with the table above we can deduce that practically 2/3 of the hotels are located in the cities while the other ones (1/3) are resort hotels.

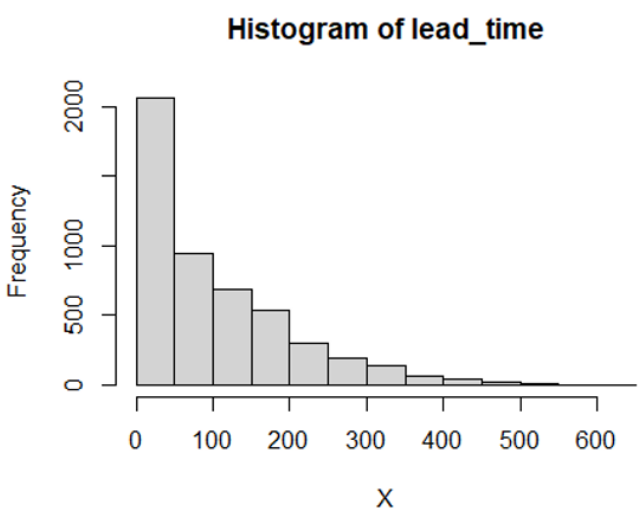
We have almost twice the number of bookings on city hotels than in resort areas. Maybe it could indicate that most of the people who have booked the hotels preferred to go to the city than to the resort.

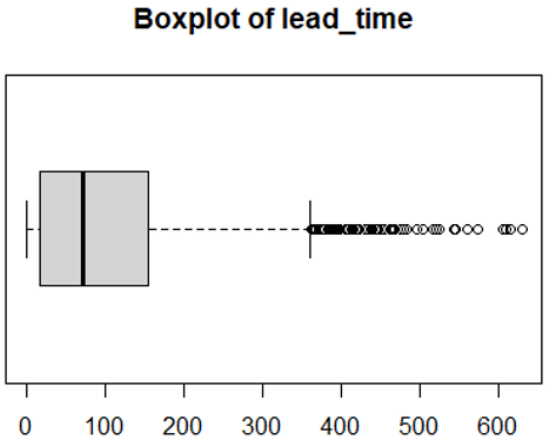
Variable 2: is_canceled



We can see that the amount of bookings canceled is high. About 36% of the bookings are cancellations. In the pie plot we can deduce that the not canceled bookings represent the “0” and the canceled ones are represented with the “1”.

Variable 3: lead_time



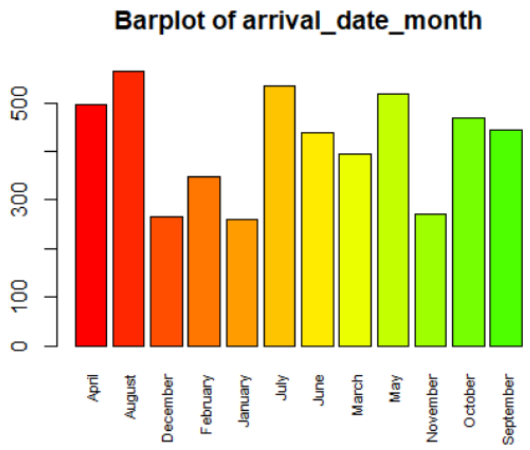


Variable	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	sd	vc
lead_time	0.00	18.00	72.63	101.30	155.00	629.00	101.47	1.00

One of our top interesting variables is the lead time, the time passed during a person books the hotel and he or she finally arrives. The average is 100 days, and most bookings are done between 18 and 155 days before the arrival to the hotel. It makes a large interval of practically 4 months and a half where people usually booked their hotel.

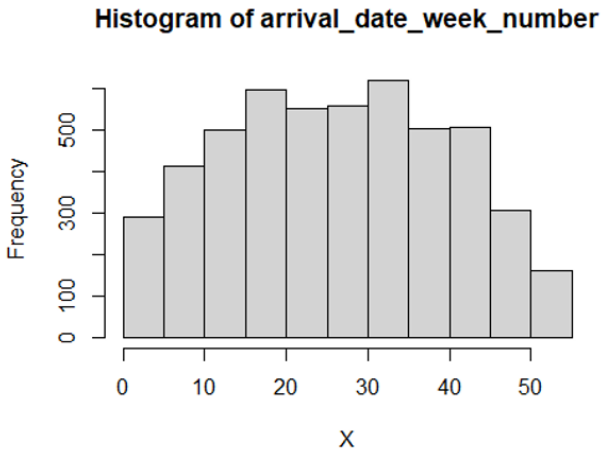
Talking about the histogram we can see that a lot of people tend to book about 50 days before their arrival. Also there is the other side that if we focus on the boxplot we watch some of bookings done before one year of the arrival, that outliers makes the mean being about 100 days.

Variable 4: arrival_date_month



With these statistics we can see in which month of the year the most people decide to book. As expected, the months with the most bookings are the two during the holidays: August and July. Something to take into account is that their predecessors are May and April. We have thought that is due to the fact that they are the spring months and usually the weather improves.

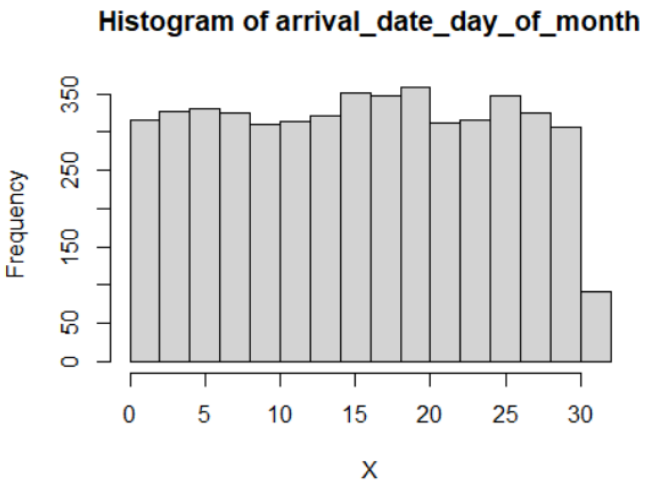
Variable 5: arrival_date_week_number



Another interesting plot, also related with the periods of the holidays, are the week number of the year. Looking at the histogram we can see that most of the bookings are done during the weeks between the holidays' days. Another point to be taken into account is that at the start and the final of the year people do not go to the hotels. We have thought that the reason would be for Christmas and the low temperatures during the winter.

In the box plot we can see clearly when most of the bookings are done, between 16 weeks and 38 weeks, which represents May, June, July, August, September and October.

Variable 6: arrival_date_day_of_month

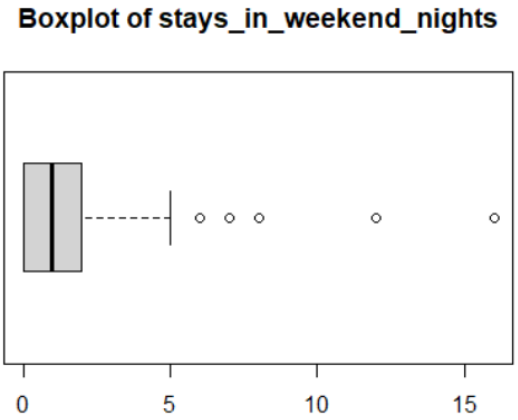


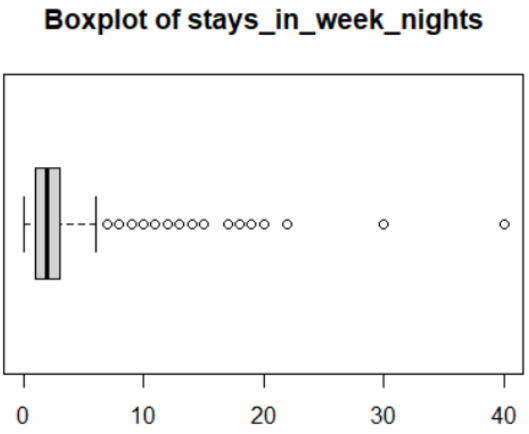
Variable	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	sd	vc
arrival_date_day_of_month	1.00	8.00	16.00	15.81	23.00	31.00	8.7476	0.5531

In that histogram we can see that the days of the month booked are normally distributed, indicating that the day of the month (the number) is irrelevant for most of the people when deciding to book a hotel.

We have to clarify that in the histogram, after the day of the month 30 th frequency is way lower. That is due to the fact that there are 5 months of the year that do not have 31 days.

Variable 7: stays_in_weekend_nights; Variable 8: stays_in_week_nights



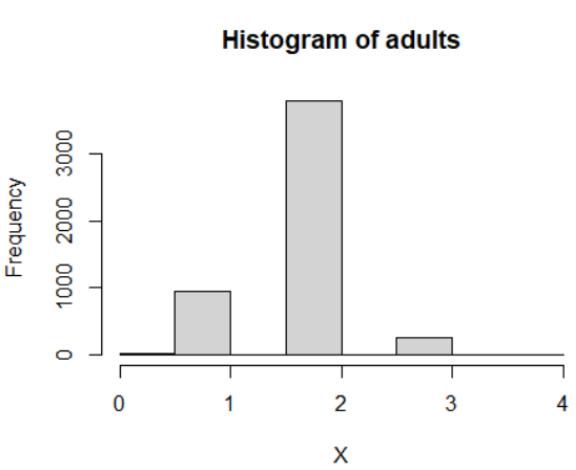


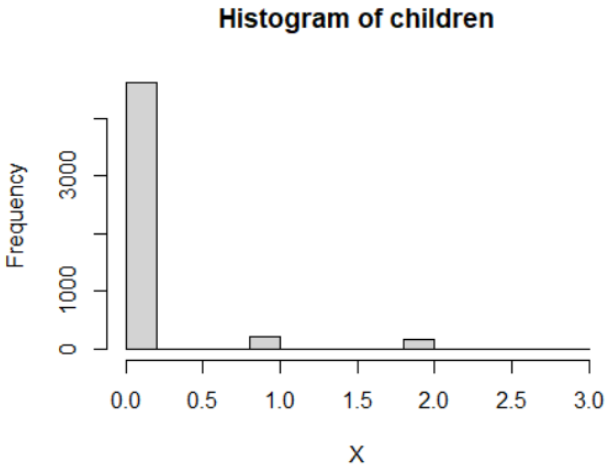
Variable	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	sd	vc
stays_in_weekend_nights	0.0000	0.0000	1.0000	0.9094	2.0000	16.0000	1.0067	1.1070
stays_in_week_nights	0.000	1.000	2.000	2.491	3.000	40.000	1.946	0.781

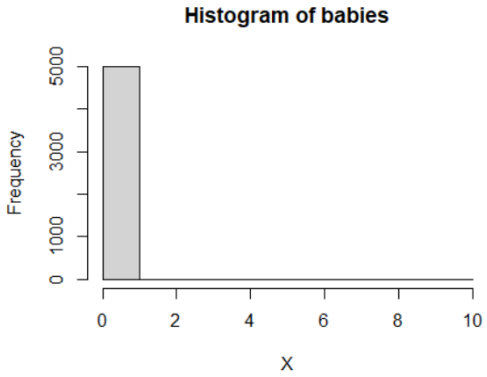
We can look at the weekend nights histogram and deduce that there are not lots of bookings made for more than 2 nights of the weekend. That is caused by the fact that most people cannot afford being on holidays, or working abroad for more than a week.
Also we have to add that there is an outlier that has stayed for more than 16 weekend days which at least means 8 weeks. But as we have mentioned, there are punctual cases.

In the histogram of the week nights we can see that there is more variety because the standard variation has increased. Now, that value is indicating that the major part of the bookings are planned between 1 and 4 days of the week.

Variable 9: adults; Variable 10: children; Variable 11: babies





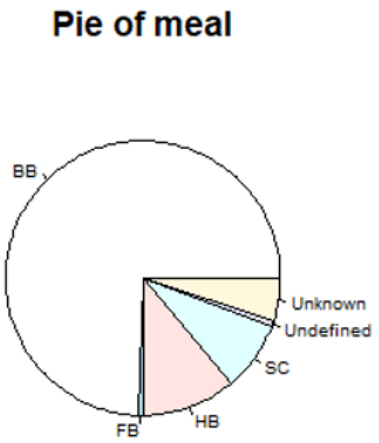


Variable	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	sd	vc
adults	0.000	2.000	2.000	1.856	2.000	4.000	0.481	0.259
children	0.0000	0.0000	0.0000	0.1082	0.0000	3.0000	0.4066	3.7579
babies	0.0000	0.0000	0.0000	0.0092	0.0000	10.0000	0.1647	17.9004

We have decided to compare these 3 variables because we can extract some conclusions from these three segments of people. Firstly, most of the reservations done are for 2 adults, 0 children and 0 babies.

We can see that the mean of children and babies are 0, and the standard deviation also indicates that it is very common not to make a booking that has children or babies.

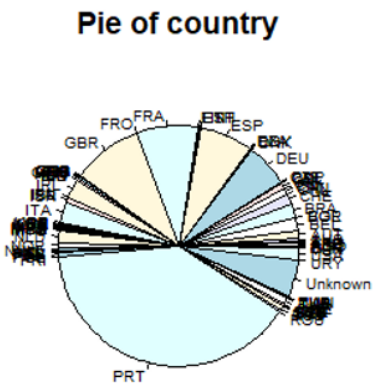
Variable 12: meal



Modalities	Frequency	Proportions
BB	3719	0.7438
FB	37	0.0074
HB	542	0.1084
SC	414	0.0828
Undefined	38	0.0076
Unknown	250	0.0500

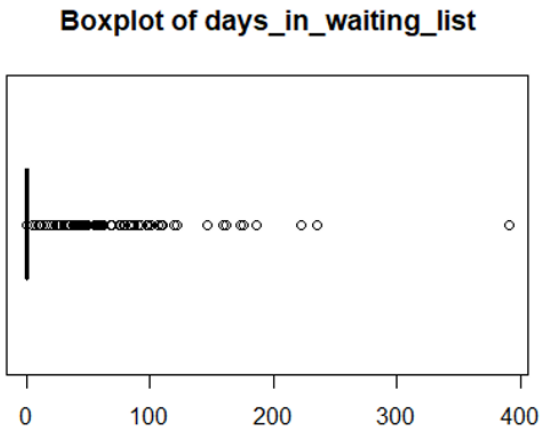
In this pie chart we can see how the most popular meal type by far is the Bed and Breakfast option, with almost 75% of the people choosing it. The second most popular is Half Board and the remaining guests choose between getting all the meals or none at all (or they are missing values).

Variable 13: country



Here we have 90 modalities and there is a lot of noise. However, from the sorted frequency table we can see that most of the customers come from PRT, GBR, FRA, ESP, DEU and Unknown (we cannot know by logic where the customers come from), being that 1960 (39,2%), 476 (9,52%), 403 (8,06%), 355 (7,1%), 281 (5,62%) and 250 (5%) respectively.

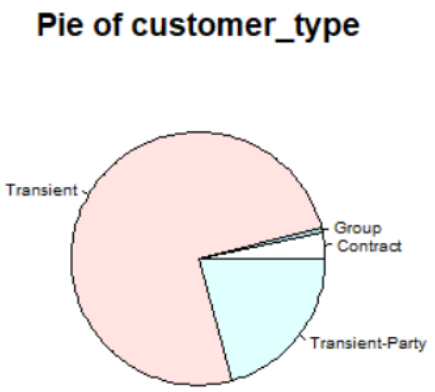
Variable 14: days_in_waiting_list



Variable	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	sd	vc
days_in_waiting_list	0.000	0.000	0.000	2.557	0.000	391.000	16.78	6.564

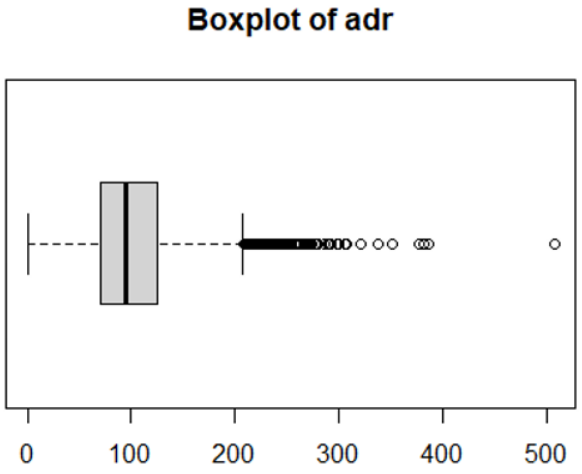
In this plot we can clearly see that, even if the majority of people spend between 0 and 3 days in the waiting list, we have some outliers that spend more than 100 days, or even an extreme case of 391 waiting. This last case could be a typo when collecting the data or it could be a rare case from a hotel with a big demand for rooms.

Variable 15: customer_type



In this pie chart, we can clearly see that most of the bookings are not associated to a group or a contract nor related to another transient booking. Also, *group* values are almost negligible due to the fact that they represent a really small part.

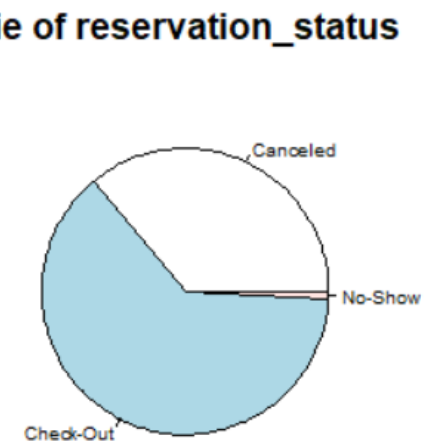
Variable 16: adr



Variable	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	sd	vc
adr	0.0	70.0	95.0	101.7	125.2	508.0	48.00	0.47

For the ADR we can see that the average is at around 100. That being said, we can also see that there are many hotels with an average rate higher than 200, and a clear outlier at 508. This could be due to the price of the hotel or the amount of people that book it, giving it a higher daily rate.

Variable 17: reservation_status



Modalities	Frequency	Proportions
Canceles	1799	0.3598
Check-Out	3157	0.6314
No-Show	44	0.0088

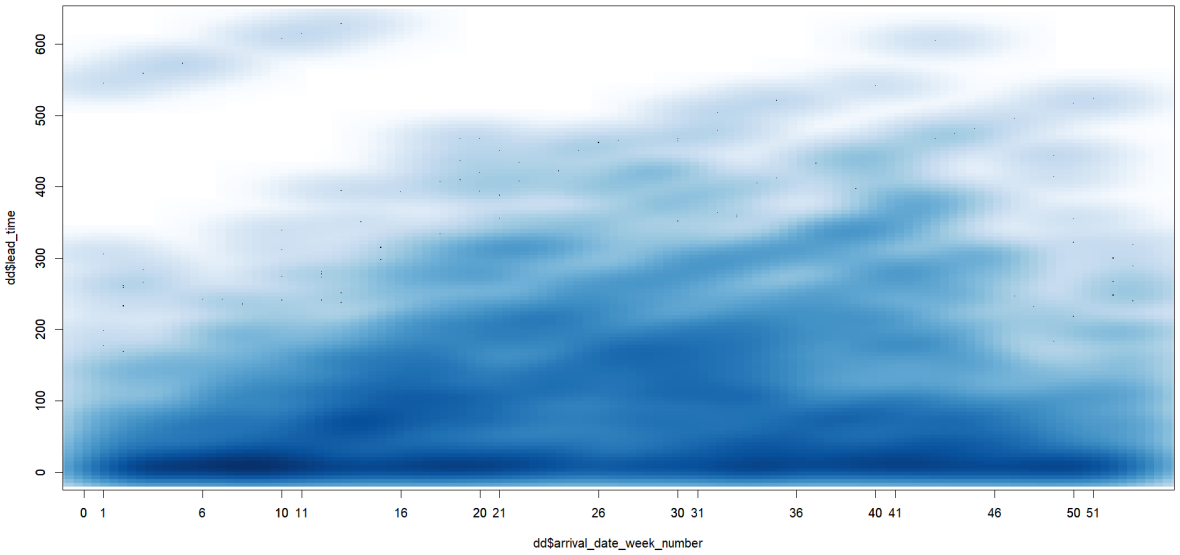
Here we can see that the majority of reservations made are confirmed at Check-Out (around 63%). What seems worrying is that a relatively high percentage happens to be canceled by the client, while a practically insignificant part does not check-in.

6.2. Bivariate descriptive statistics

Bivariate refers to a statistical analysis or data analysis that involves two variables. It is crucial to choose two variables which are the most reliable ones. In this section we have compared different variables which we believe their analysis will give us some information.

6.2.1. Lead Time vs Arrival Date Week Number

Two variables that we have analyzed are the relation between the *lead_time,* (the time between the booking and the arrival) with the *arrival_date_week_number* (the week number of the year).



If we observe the scatter plot, it provides us with information about how much time tourists book their hotel depending on the period of the year. We can notice that between weeks 1 to 10, there are a lot of bookings made in the previous days, which is indicated by that dark zone, close to the X-axis.

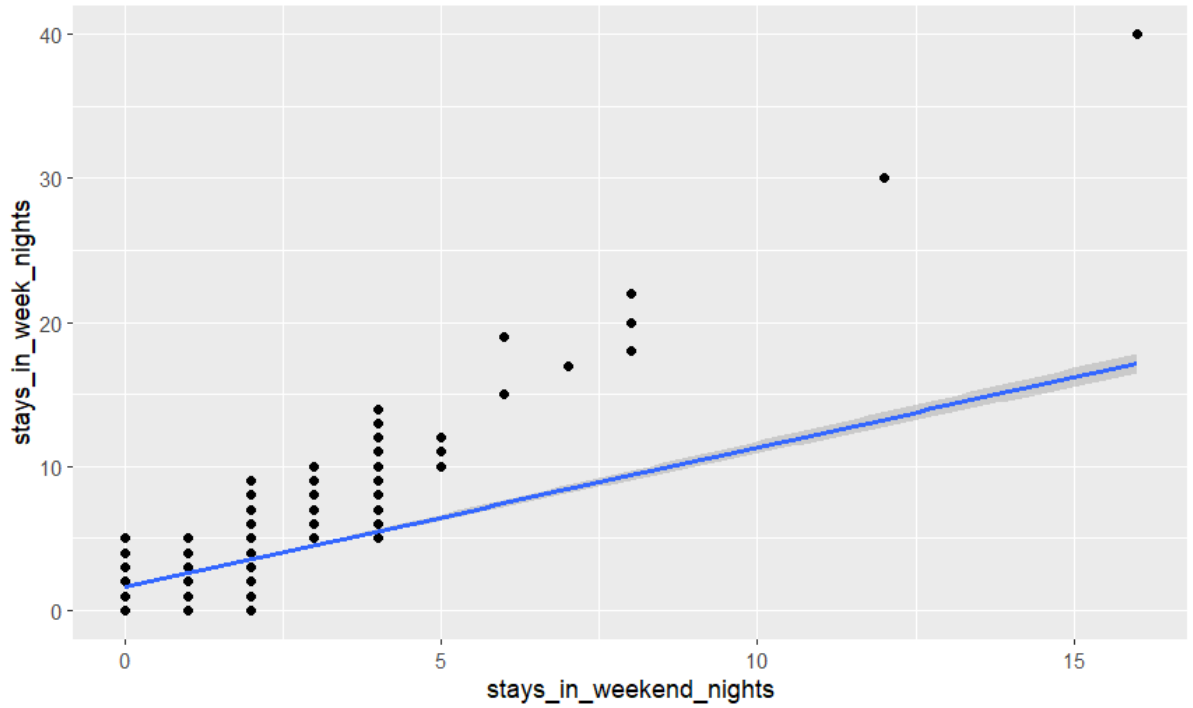
Another interesting point is that most of the bookings are made by more provident tourists between weeks 26 to 33, which are also the weeks of the summer vacation.

After those weeks, we see a shift as tourists tend to reserve with less anticipation. Before that, we can also notice that during the final two weeks, around Christmas, most people tend to make their bookings some time in advance.

6.2.2. Nights on Week Days vs Nights on Weekend Days

Another two variables which we have noticed that might be correlated are the nights on weekend days and the nights on weekdays.

The next graphic shows the correlation between those two variables:



We can notice that as we increase the number of week nights it also increases the number of weekend nights. Although, the blue line represents the average, that indicates the tendency of our data.

Firstly we should think that for every 2 weekend days we would have 5 week nights. Nevertheless, our data does not show the same. If we observe where the average line crosses the Y-axis we can notice that for 10 week nights, our population stays between 6-8 days, which represents a high deviation of what we thought.

The reason for that data is the average of week and weekend nights booked. We can notice that most of our data are for people who book between 2 and 3 week nights and an average of 1 weekend night, showed in the section:

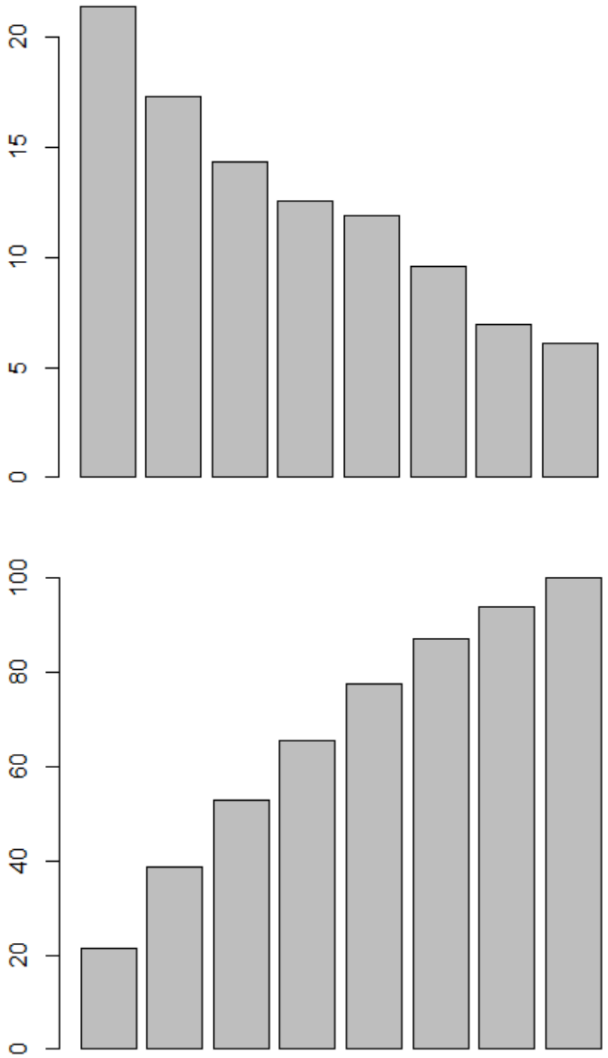
Variable 7: stays_in_weekend_nights; Variable 8: stays_in_week_nights

7 - PCA

7.1. Factorial maps

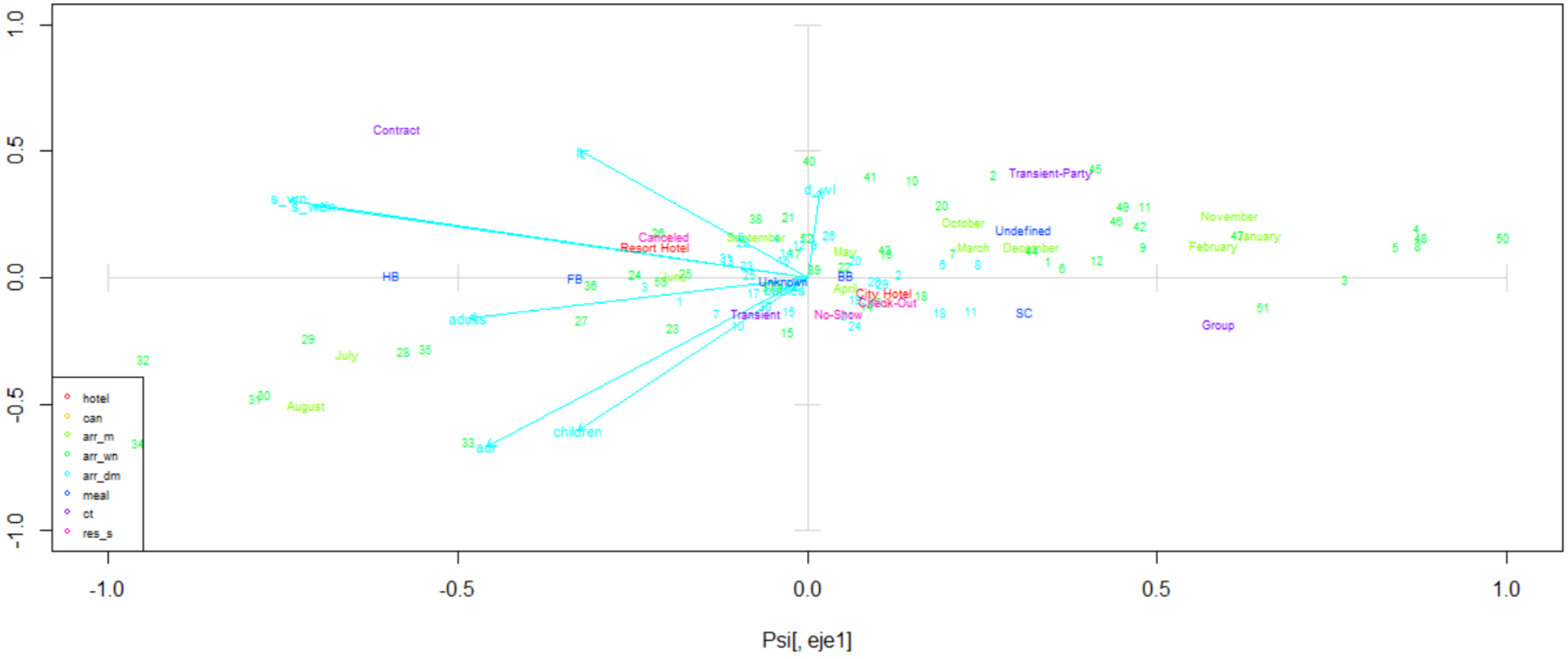
In order to reduce and summarize the amount of data we have, PCA has been applied.
Since it has to represent a good quantity (75%-80%) of all the information of the original database, we have to know how many linear combinations we need in order to reach the 80%.

In the first plot we can see the information provided by every PCx and in the following cumulative plot we can see that we get to the acceptable percentage in the PC6.

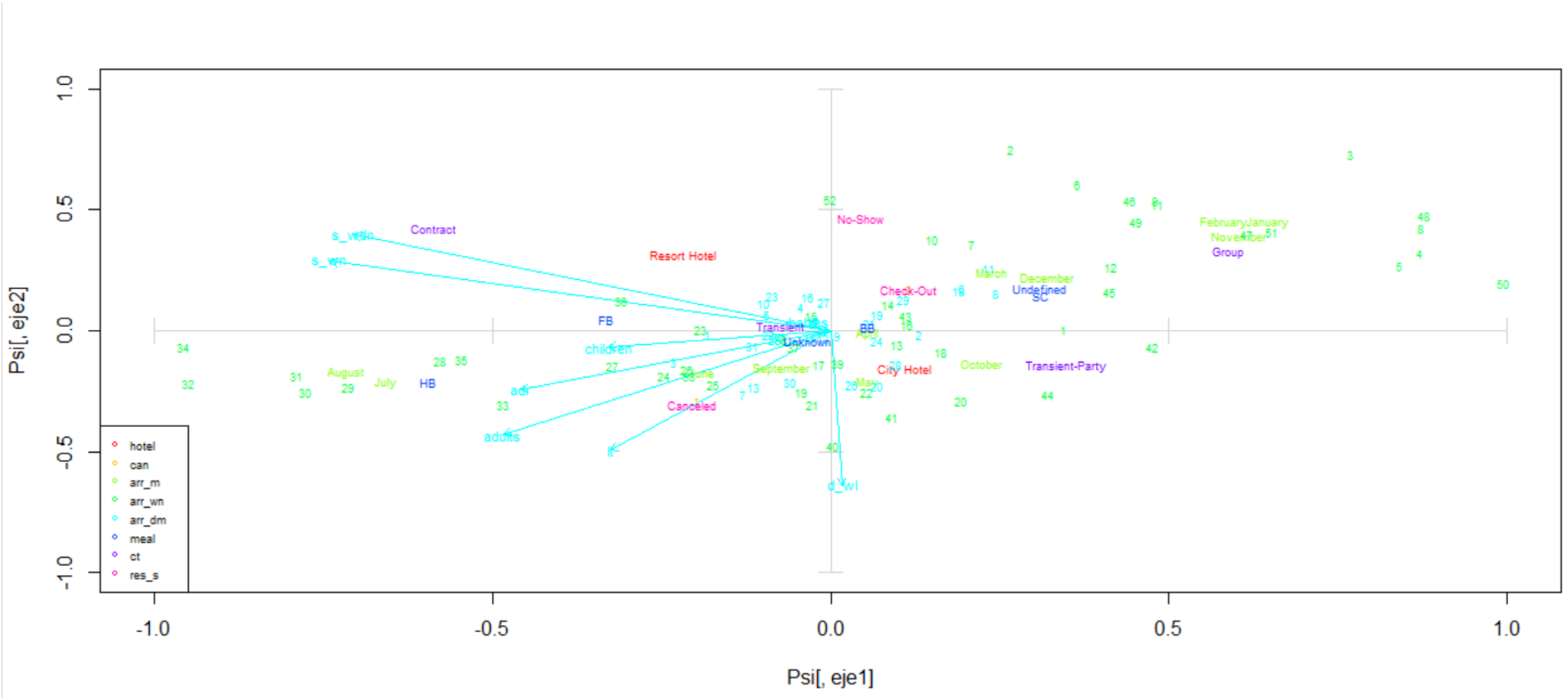


Since 6 dimensions is a lot, we will do our analysis with the first 3 components, trying all combinations between them to find which one works the best for us.

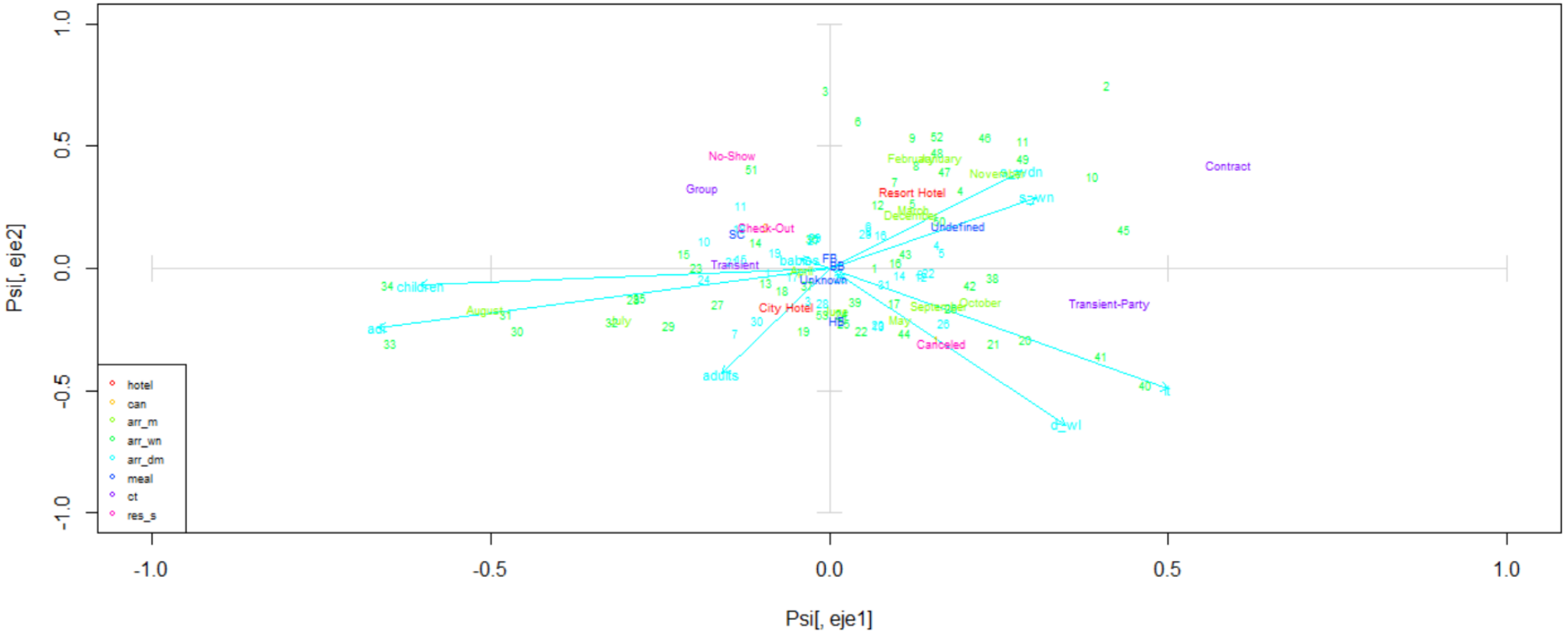
There is the combination of PC1 and PC2:



There is the combination of PC1 and PC3:



And finally, there is the combination of PC2 and PC3:



7.2. Interpretations

After getting the previous PCA plots, we can see that the first two are the ones with the most information, since they contain PC1, and that the third one isn't very clear which vectors contribute the most to the Y axis. Therefore, we will focus on the former ones to get our interpretations.

PC1 vs PC2

In this graph we can see that for the X axis, the most important vectors are *adults*, *stays in week day nights* and *stays in weekend nights*, so we will label it as *Accomodation Demand*. As for the Y axis, the only clear contributor is *days in waiting list* so that will be what we label it. Both *stays in week day nights* and *stays in weekend nights*'s vectors are almost on top of each other, which indicates that they are highly correlated. As for the other vectors, they are closer to a 45º angle with both axes, so we won't take them into account when labeling them.

Since the vectors for the X axis are pointing to the left and the one for the Y axis is going upwards, we know that the more to the left we go, the more nights reserved by adults, and the more upwards, the more days in the waiting list. With that in mind, we can start extracting some conclusions.

Starting with the type of hotel, we can see that the Resort is the preferred option. It has more nights booked by adults than the City option. As per the meal type, we see that people staying at City hotels usually choose a Bed and Breakfast option, whereas in the Resort they prefer the Full Board if they don't stay many nights, and switch to a Half Board when they stay longer. This would make sense in order to accommodate the price.

If we take a look at the months, it doesn't come as a surprise that from June to August is when adults spend more nights in a Resort, with the highest values on August (coinciding with the usual vacation period of adults). April and May are the months closer to the City hotel, which tells us that that's the most common time to stay there, and the rest of the months of the year have the lowest number of nights, probably due to the fact that most people are working.

An interesting fact is that, when we look at the reservation status, the Resort bookings are more likely to get canceled by the customer, as opposed to the City ones where the customers get to check out or simply don't show up.

Finally, observing the *customer type*, we can tell that customers that have placed a booking with some kind of contract usually spend more nights, but they are also more days on the waiting list. Transient customers, probably customers on vacations, spend less days and have to wait less to get a booking, and groups are the less popular type of booking, with the least amount of nights spent by this kind of reservation.

PC1 vs PC3

In this other graph we can see that for the X axis, the most important vectors are *Children*, *Adr* and the same ones from the previous graph, so we will label it as *Accommodation Demand and expenses*. As for the Y axis, the main contributor is once again *Days in waiting list,* even though this time it is inverse to the previous one, so we will label the Y axis *Days in waiting list* again. We can see that obviously both graphs are very similar to each other since both of them contain the PC with the most information, which is PC1, so in the following section we will try to describe the situations we have not been able to see (differently or clearly) in PC1 vs PC2.

Lead time refers to the time that elapses from when a person books a hotel until this person arrives. On the other hand, *Canceled* refers to the action of canceling a hotel booking. Although it may not be a very clear conclusion since the vector "ld" is at 45 degrees at the left bottom part of the graph, we can see that it is really attached to "canceled". This is due the fact that *Lead time* can influence the likelihood of a hotel booking being canceled. The longer the *Lead time*, the more time the person has to decide whether to cancel the reservation or not.

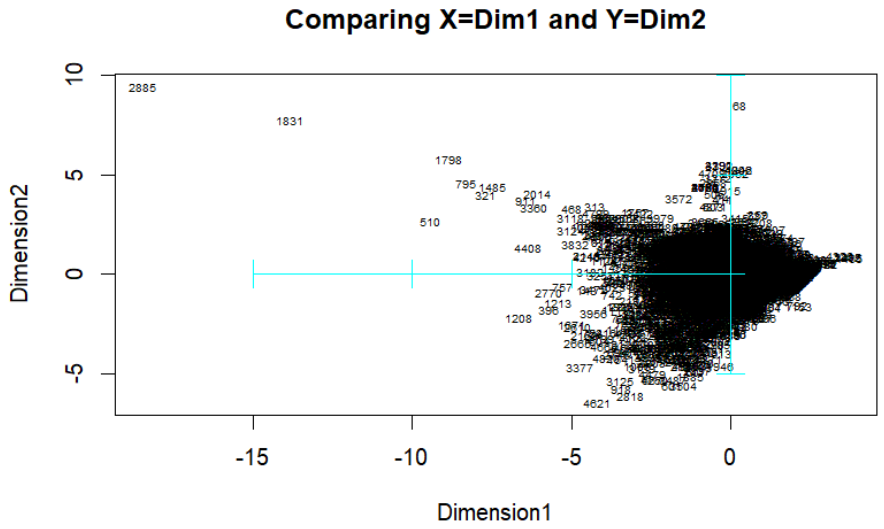
On the other hand, as we can also notice in PC1 vs PC2, there is a decrease in the probability of the reservation status *No-Show* as the *Days in the waiting list* increases. We have concluded that this may be the situation of the guests who may have initially been on the waiting list due to lack of availability, but as their stay date approaches, they are more likely to be moved to confirmed reservations, which in turn reduces the likelihood of *No-show*s.

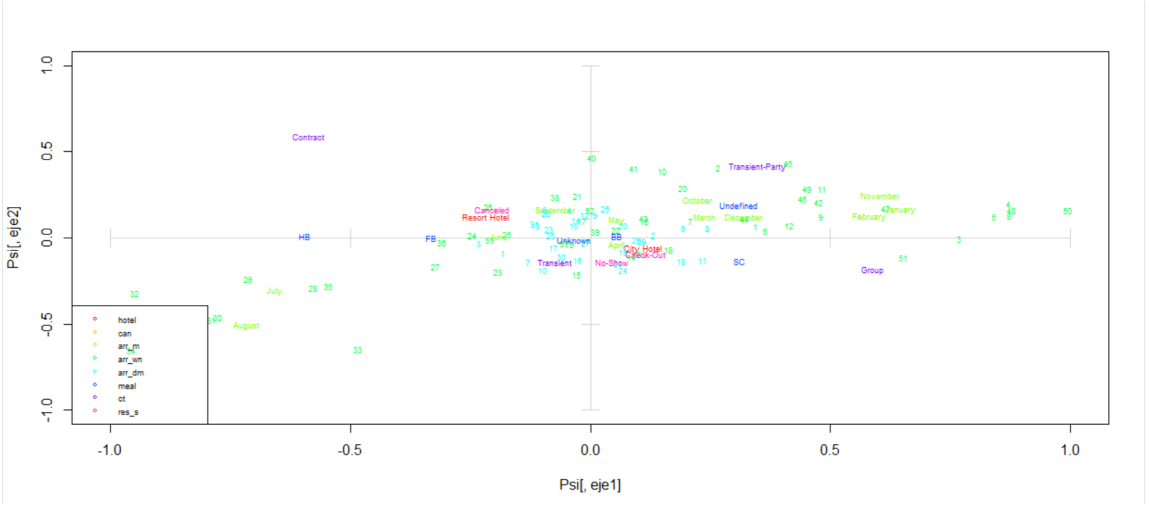
Unlike PC1 vs PC2, we can observe that the number of *children* increases significantly during the summer months of August and July due to summer vacations.

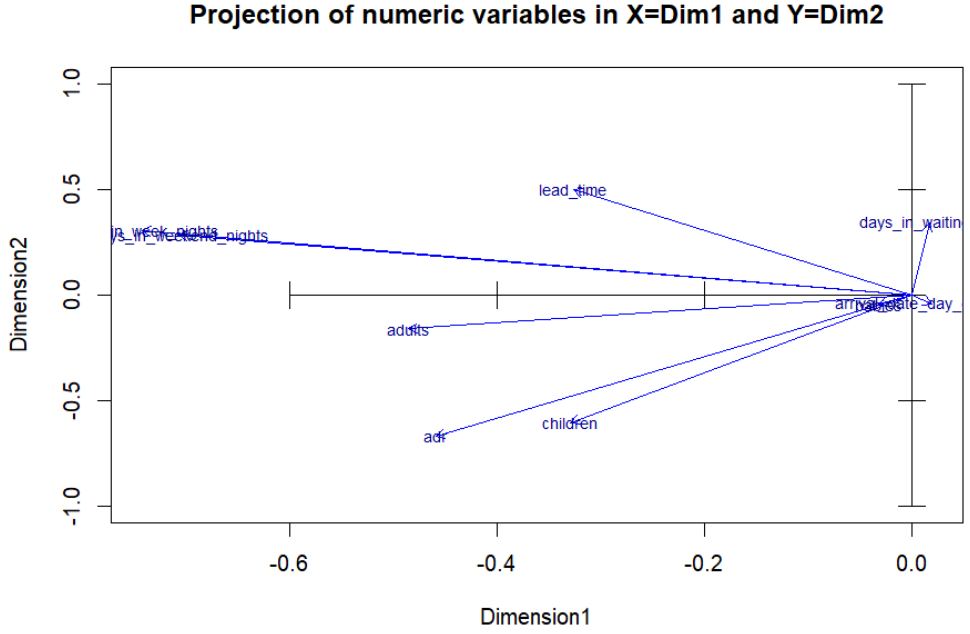
Moreover, the positioning of *children* along the x-axis, near *FB*, suggests a correlation between the number of children in the reservation and the overall expenses. It appears that when there are more children, there is a higher likelihood of opting for "Full Board" as the meal plan, contributing to increased spending per person per night.

Individuals:

PC1 vs PC2

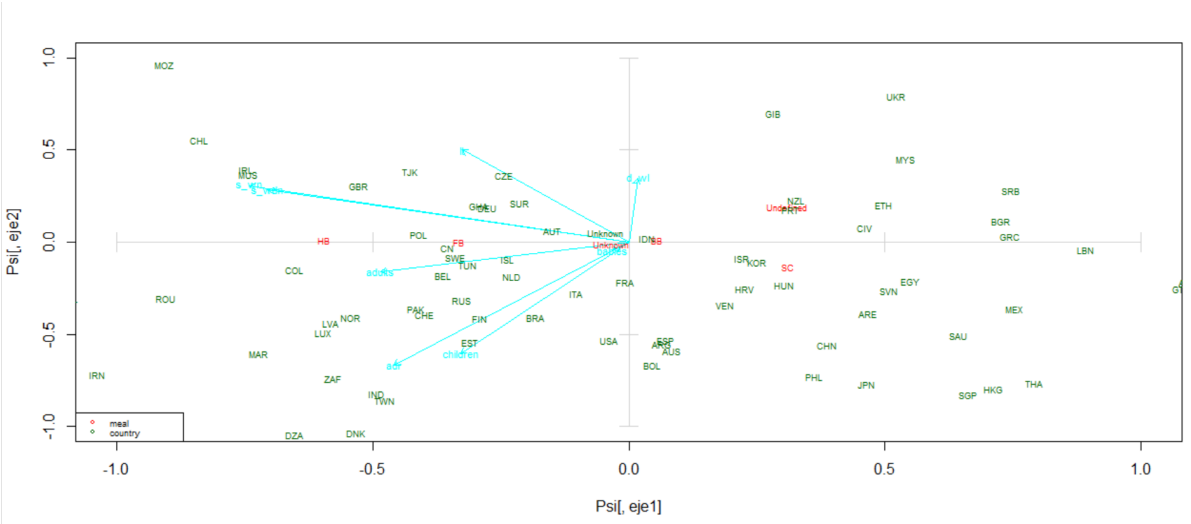






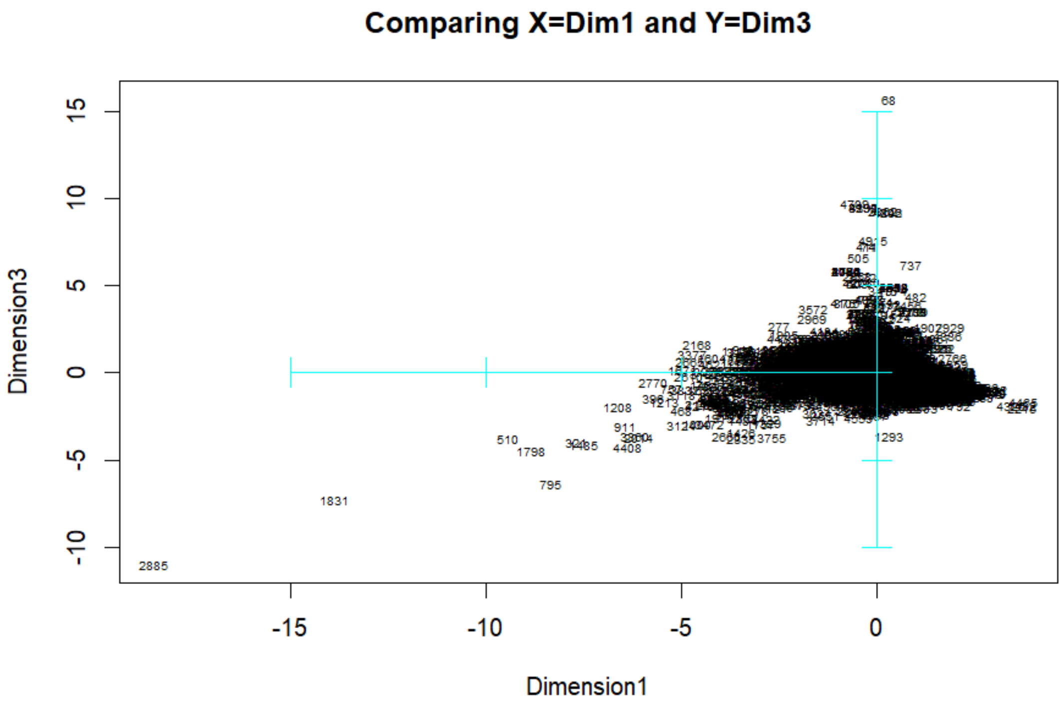
If we plot the individuals for PC1 and PC2, we can see that they concentrate mostly around the (0, 0), extending a bit towards the third quadrant and more pronouncedly towards the second.

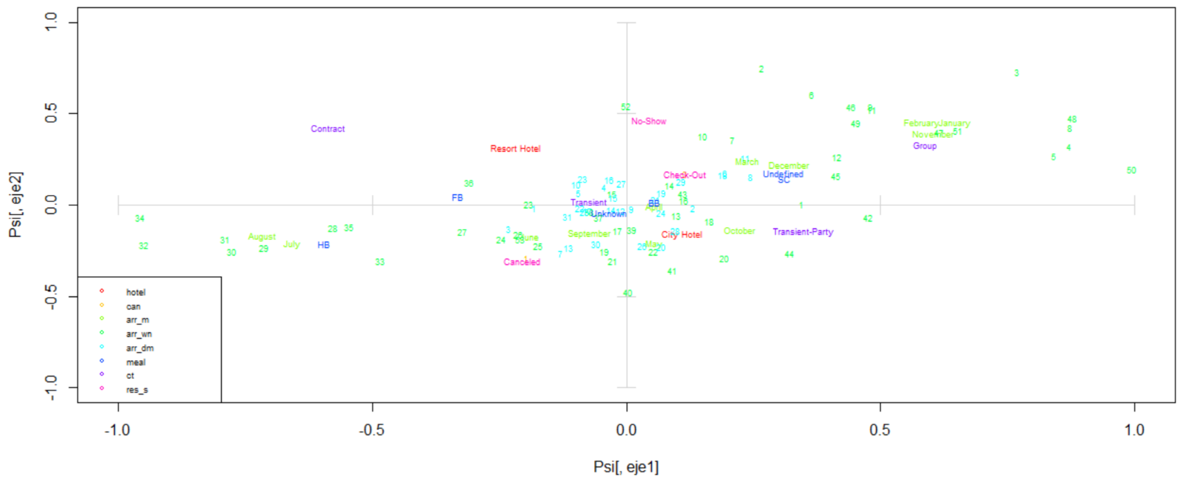
If we overlap this with the graphs of the categorical and numerical variables, we can deduce that the majority of the individuals choose the central values, such as Bed and Breakfast options for the meal, that are mostly transient or have contracts, and that they tend to go more for the Resort hotel. As for the vectors, we can say that the individuals tend to have more nights and more lead time. We can also identify some individuals as outliers compared to the rest (2885, 1831…).

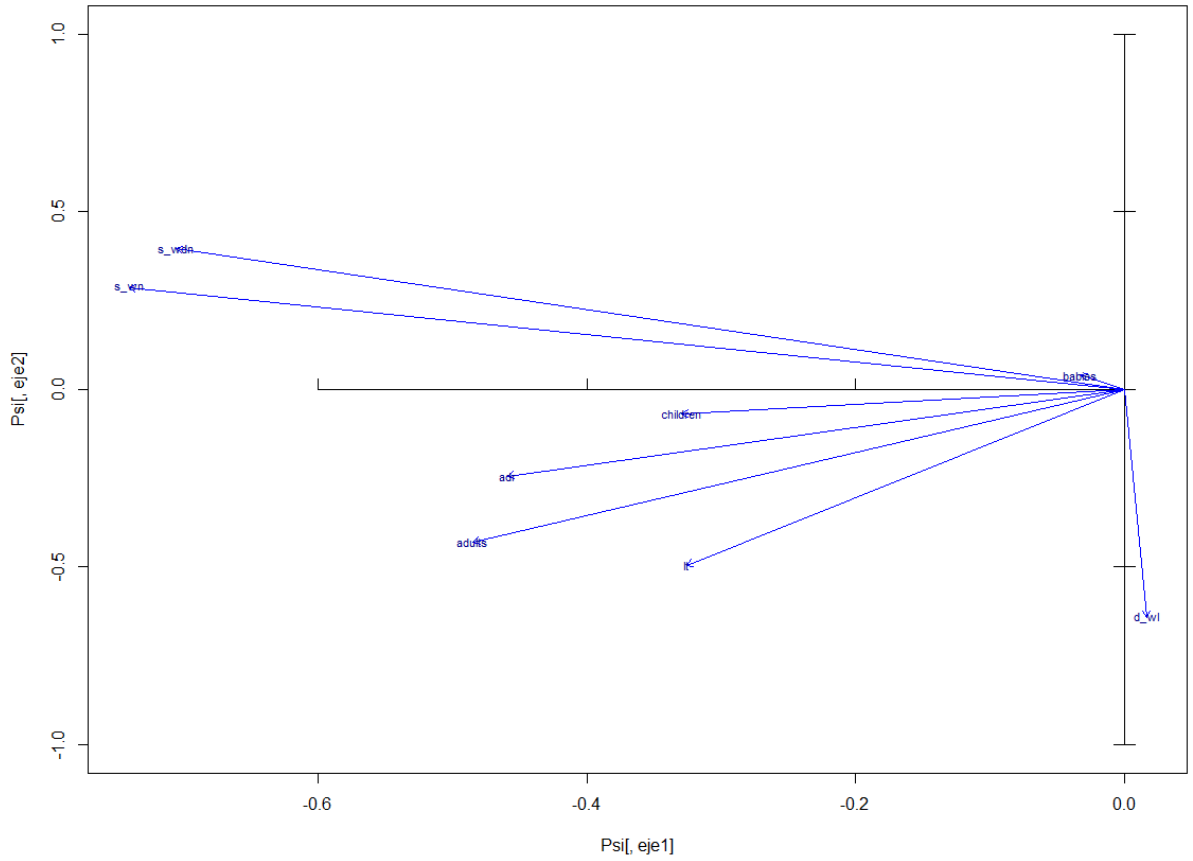


Finally, taking a look at the countries of origin of the customers, we can find some more interesting information, such as New Zealand and Portugal being the countries that are more likely to not specify a meal type, as opposed to Colombia being the most likely to choose a Half Board, or China and Sweden to choose a Full Board. We also see Iran, Romania, Mozambique and Chile at the top of countries that spend more nights in the hotel, with the difference that the first ones spend less days on the waiting list than the latter ones.

PC1 vs PC3


Comparing X=Dim1 and Y=Dim3
Dimension3
Dimension1
15
10
5
0
-5
-10
-15
-10
-5
0

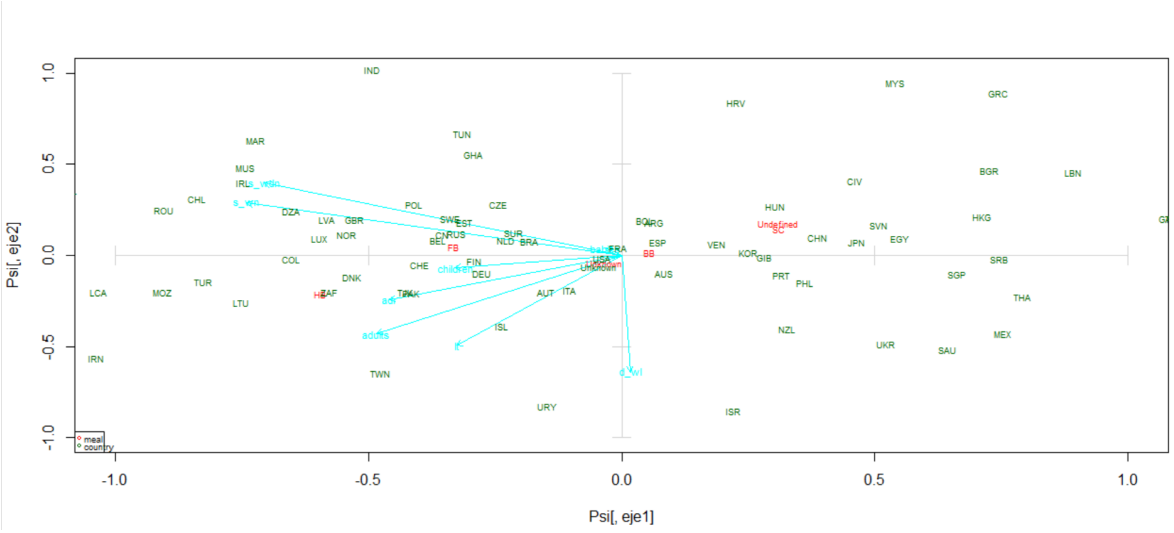

Psi[, eje2]
Psi[, eje1]
1.0
0.5
0.0
-0.5
-1.0
-1.0
-0.5
0.0
0.5
1.0
hotel
can
arr_m
arr_wn
arr_dm
meal
ct
res_s



In this case, we can see the cloud of individuals also centered at (0, 0), but this time spreading especially towards the left upwards.

By overlapping it with the categorical variables, we can see in this case that the individuals lead more towards no show and canceled bookings, though it still predominates the trend of Transient guests with the Bed and Breakfast meal plan. We can also see that the trend of booking during summer months stays true, as the cloud spreads more towards the left than the right.

By looking at the numerical variables, we see that the individuals still tend towards more lead time, and that the tendency is to have lower days on the waiting list. The outliers identified in the previous plots (2885, 1831) are still far from the main cloud of individuals in these ones.



Finally, looking at the plot of the countries, we can see that in this scenario it is Russia, Belgium and still China that go for the Full Board menu, while South Africa is the most likely to choose a Half Board. Iran and Saint Lucia are the countries that book more nights, while Lebanon and Thailand are on the opposite side, with the least nights.

India is the one with less days on the waiting list, while Uruguay and Israel are the ones with the most days of waiting. It also appears that countries like Colombia, Turkey or Luxemburg travel more with their kids.

7.3. Conclusions

After having analyzed our PCA, and even if our data didn't lead to some good results, we can extract some conclusions by combining all the data from all the different graphs we had.

We can see very clearly that the months where more people travel are the summer ones, being with children or just adults, from June to August the number of nights booked are increasing.

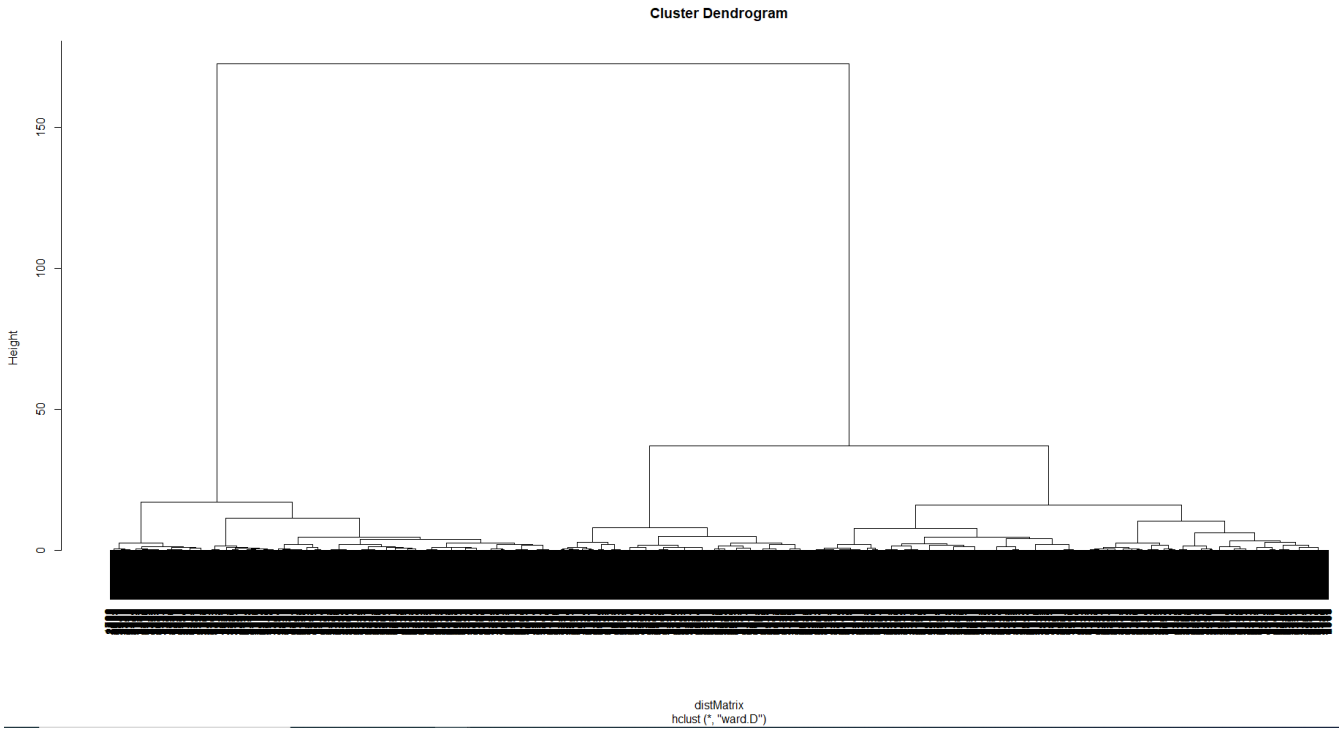
The number of weekday nights and weekend nights also seems to be related with the type of customer, which makes sense since every customer that books a night has to have a method to do so.

We can observe that there is a pattern concerning the variable *customer_type*: the most picked one from most amount to the least is "contract", followed by "transient", then by "transient-party" and finally, "group".

8 - Clustering

In this section we are going to analyze the grouping of the different variables to build data subsets known as “Clusters”. We are going to do the clustering with gower distances.

In order to know how many clusters do we have we build the following dendrogram

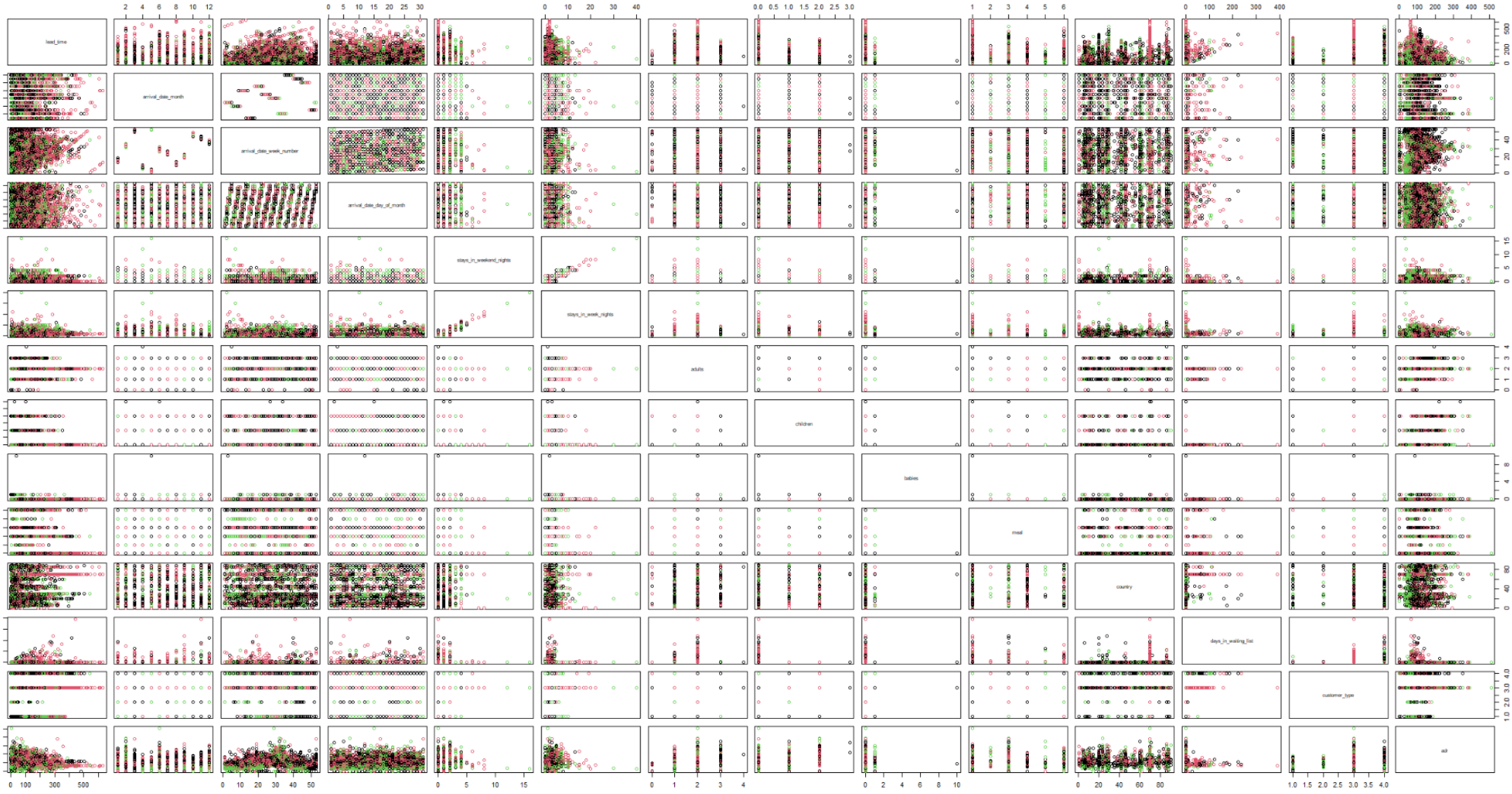


Looking at the maximum distance we get three clusters, because the maximum distance is between ~25 and ~175. Knowing this we can get the following pairplot:

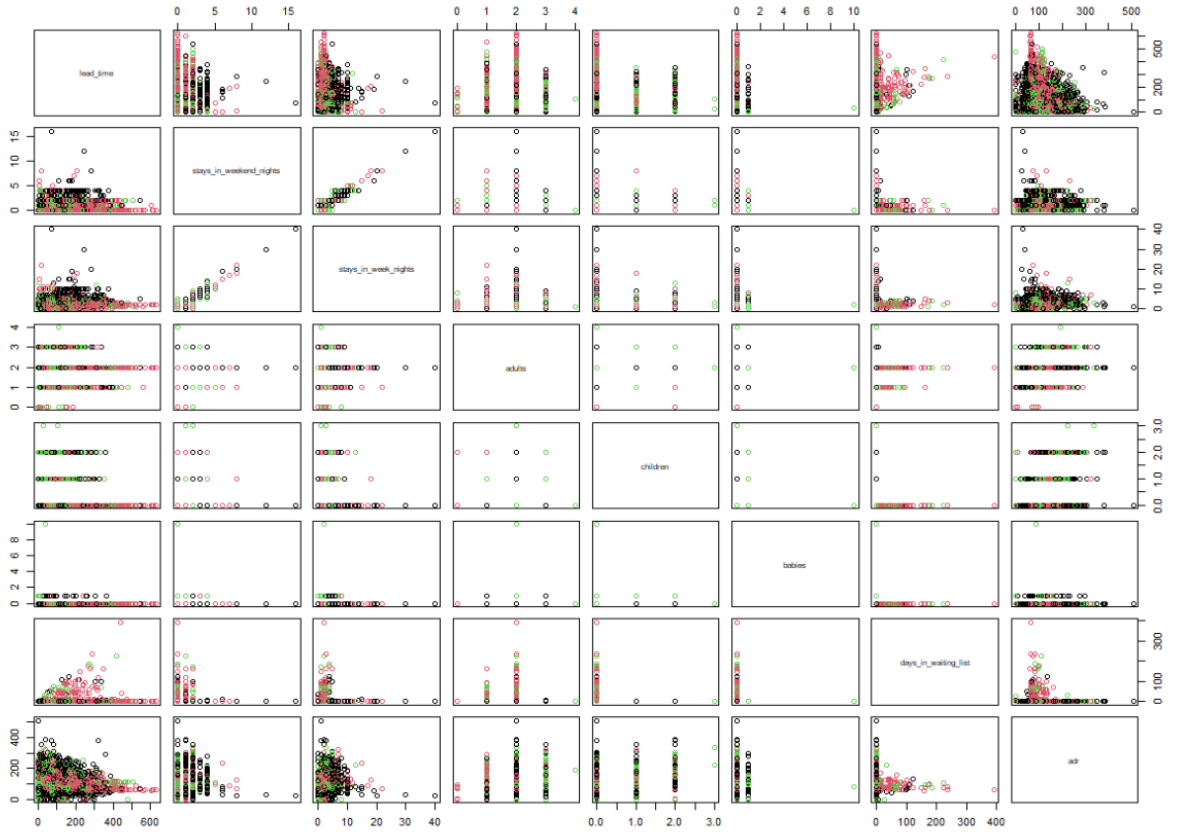

hotel
is_canceled
lead_time
arrival_date_month
arrival_date_week_number
arrival_date_day_of_month
stays_in_weekend_nights
stays_in_week_nights
adults
children
babies
meal
country
days_in_waiting_list
customer_type
adr
reservation_status

As we can see, we can barely extract clear information about the vast majority of graphs. The only thing we can highlight is that in those where we see more information, they are the ones in which the individuals in each cluster are more mixed.

So, let's see if removing the first two variables *hotel* and *is_canceled* and the last one this plot gives us more information:


lead_time
arrival_date_month
arrival_date_week_number
arrival_date_day_of_month
stays_in_weekend_nights
stays_in_week_nights
adults
children
babies
meal
country
days_in_waiting_list
customer_type
adr

If we only show the numerical variables:



These plots show us the comparison between variables and the values in clusters. The first impression is that there's no clear separation between clusters among any of the variables.

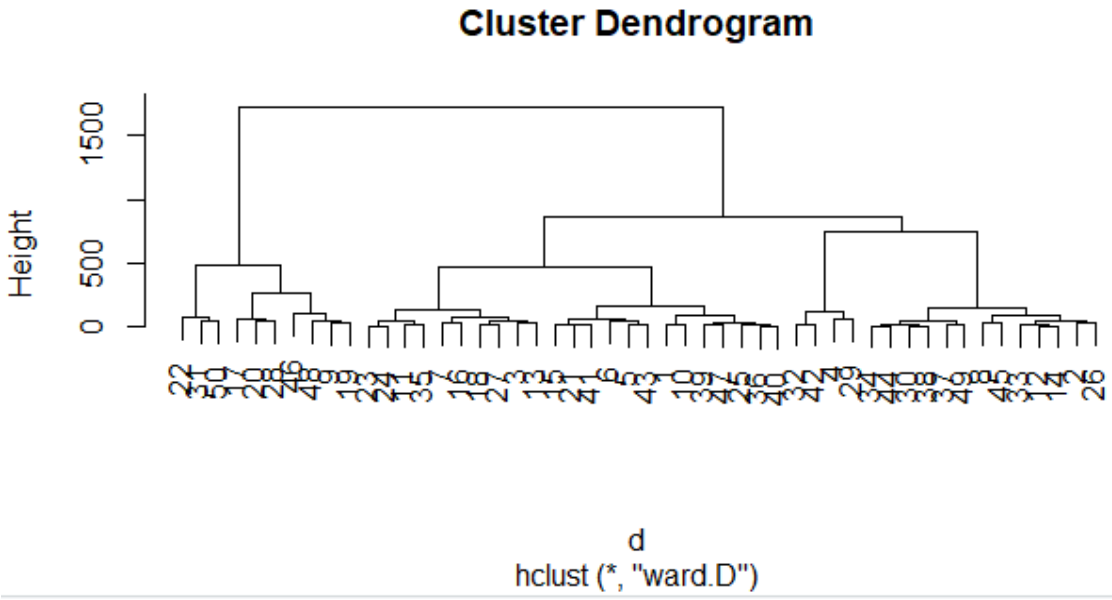
Despite this fact, we can still see some things. First of all, it seems that the variable "lead_time" is best to separate into clusters. Although not clearly separated, we can observe that when compared to "days_in_waiting_list", between 400 and 600 lead_time, it is mainly the pink cluster. We can still see this behavior in the comparisons with "adr" and "stays_in_week_nights".

Another variable that can give us information is "adr". As we can see, when compared to "days_in_waiting_list", the pink cluster is basically values from ~25 days_in_waiting_list, while the other two clusters get all other values. This means that people that have been waiting for an abnormal amount of time all are among similar "adr" ratings.

Lastly, the repartition of the clusters is pretty good, we have 1601 values in cluster 1, 1717 values in cluster 2 and 1682 values in cluster 3.

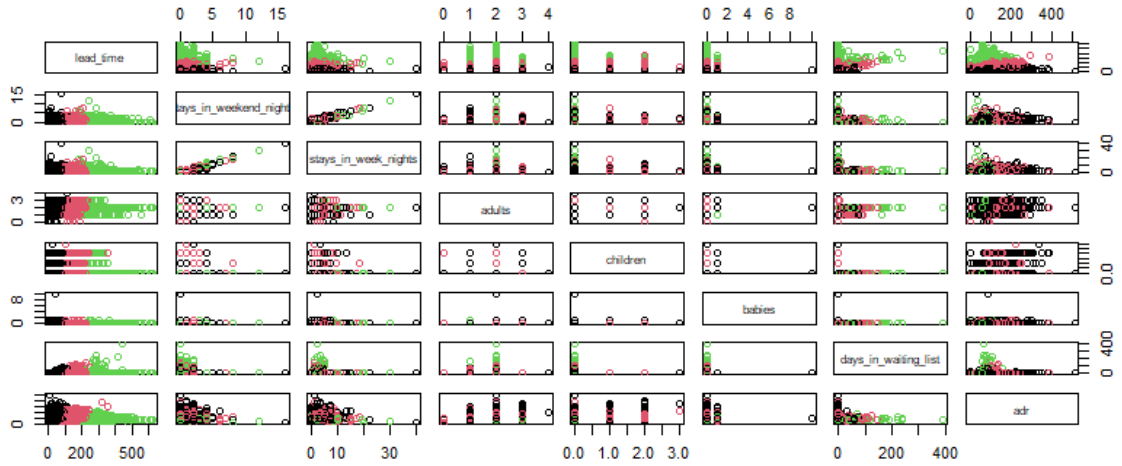
Apart from these two variables, which gave us not much information, we cannot see much more, apart from the fact that there is no clear separation between clusters. For this reason, we are now going to analyze our data with hierarchical clustering.

We are only going to work with numerical variables due to the Euclidean. To know the number of clusters we have, first we build the following endogram using the *euclidean* distance.



As we see, the highest height difference is between 500 and 1500+, so we will have to cut on 2 clusters.

Furthermore, we printed the following plot of the cleaned data to know a bit more about the clusters:



As we can see in the plot, the variable that most helps us to clearly separate the data in the two clusters is *lead_time*. In the horizontal axis for the variable, we can see that clusters are differentiated without difficulties, although in some of the graphs they are very slightly mixed. We have one side that ranges from 0 to 100 approximately with black color, another side from 100 to 250, and another last side from 250 to 650 approximately. These three colors represent the clusters that we have.

Thus, the majority of individuals belong to the third cluster.

Another interesting fact that can be seen is that if we look at the graphs in which lead_time belongs to the X axis, we can clearly see the difference between all three clusters. In fact, as the lead_time becomes larger, the more the other variables tend toward 0, except for adults, especially if we look at the green cluster. From this we can deduce that the longer the arrival time at a hotel, the more likely it is that there will be between one and two adults involved. Also, we observe that the decreasing trend is maintained from completely from the beginning to the end when we relate it to days_in_weekend_nights, days_in_week_nights and adr.

The last thing we want to highlight is that when lead_time is on the Y axis, the individuals belonging to the black cluster are diffuse, without being able to generate a solid structure of their color, so that the red and green clusters occupy the majority. When days_in_weekend_nights and days_in_week_nights have larger values, there is less lead_time, finding the maximum peak of this in the smallest values of the first two variables. Furthermore, we found a pyramid structure between lead_time and adults, suggesting that the peak lead_time is reached when there are one or two adults. When there are babies instead of adults, the range of values is significantly low, that is, there is only a significant waiting time when one or two babies are present.

In the case that adr is on the X axis, we see that a semi-pyramid is formed: between 0 and 100 the maximum is reached and subsequently the “lead_time” decreases. This may mean that at higher lead_time, the adr rating is worse.

Regarding the distribution among clusters, we have a little differentiation between clusters, with cluster 1 having 2867 values, 1437 values in cluster 2 and 696 values in cluster 3.

9 - Profiling of clusters

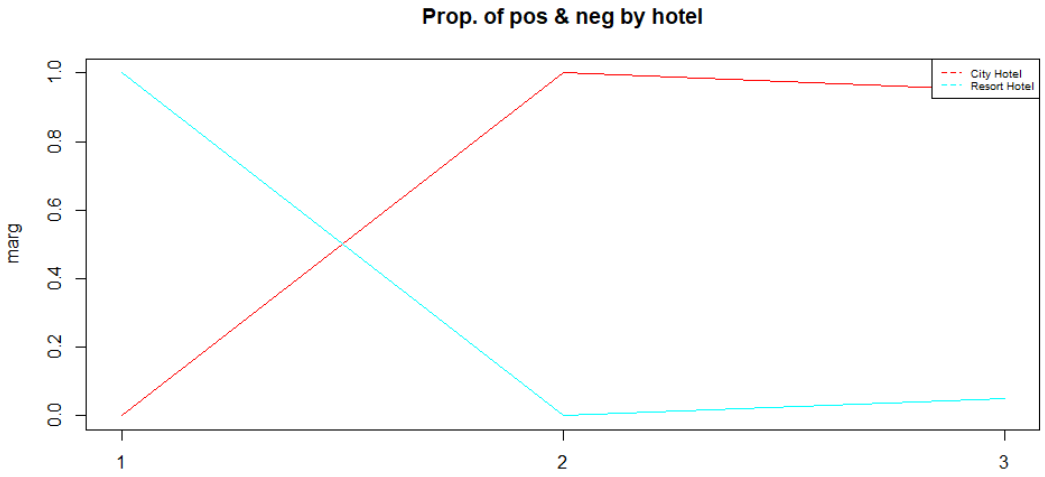
Now we are going to analyze the information contained in our data . For each variable we will perform an analysis of the relationships between the data, the structure, and its information. As a result of clustering we have our data separated into three clusters.

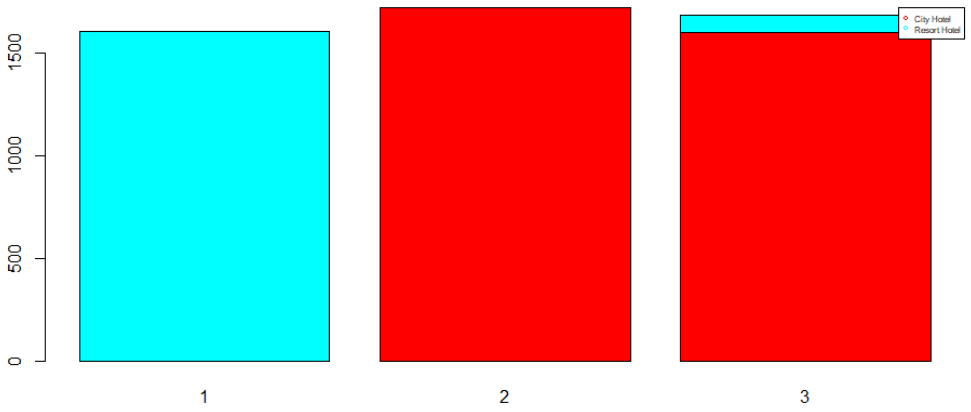
For numerical variables there would be boxplots and means, and for categorical variables, barplots.

9.1. Interpretations

1: HOTEL

The first variable to analyze is *hotel*, which represents whether a hotel type is a city hotel or a resort hotel. A first sight of the barplot suggests that clusters 2 and 3 have almost the same number of city hotels and are equally distributed. Besides, the first cluster has most of its members in resort hotels. The third one also has some.

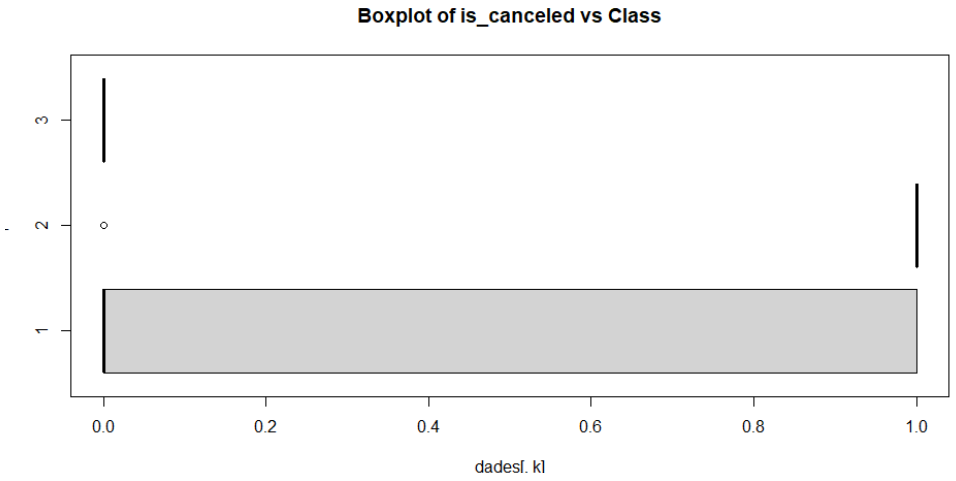


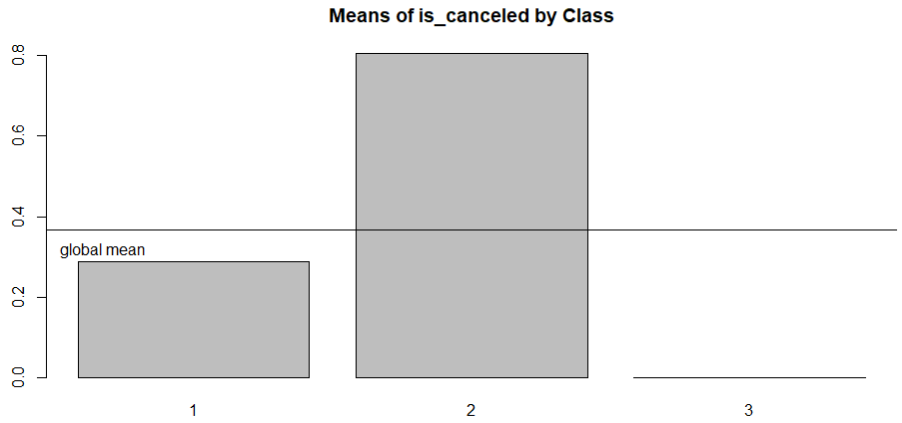


2: IS_CANCELLED

In the analysis of this variable, we can see that cluster one goes from 0.0 to 1.0, while the other clusters don't have any range. The second has its values at 1.0 and an outlier with value zero, and the last cluster has its values at 0.0 without outliers.

Then, looking at the global mean plot, we see that cluster number 2 is far upper than the global mean because of its members at 1.0, while cluster one is under because due to the fact that most of its members have lower values.



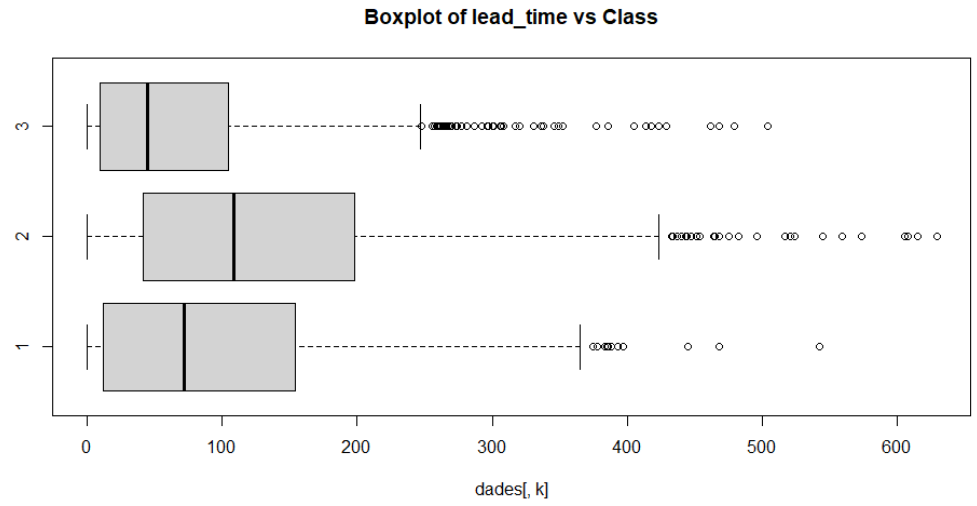


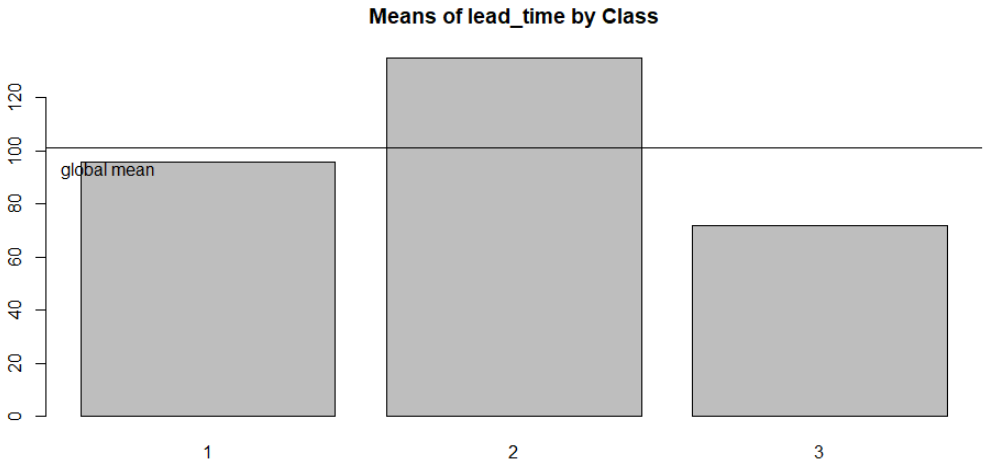
3: LEAD TIME

We can observe that the individuals in the second cluster are more likely to have a higher lead time, between 45 to 200 approximately. The individuals in the first cluster are also very distributed, but between lower values than the second group, from 10 to 155. The individuals of the third cluster are less distributed and generally include smaller values.

The difference between clusters one and three is slightly high (about 25 points), but if we compare the highest of these two with the second cluster, we find a huge difference of even more points, about 35+.

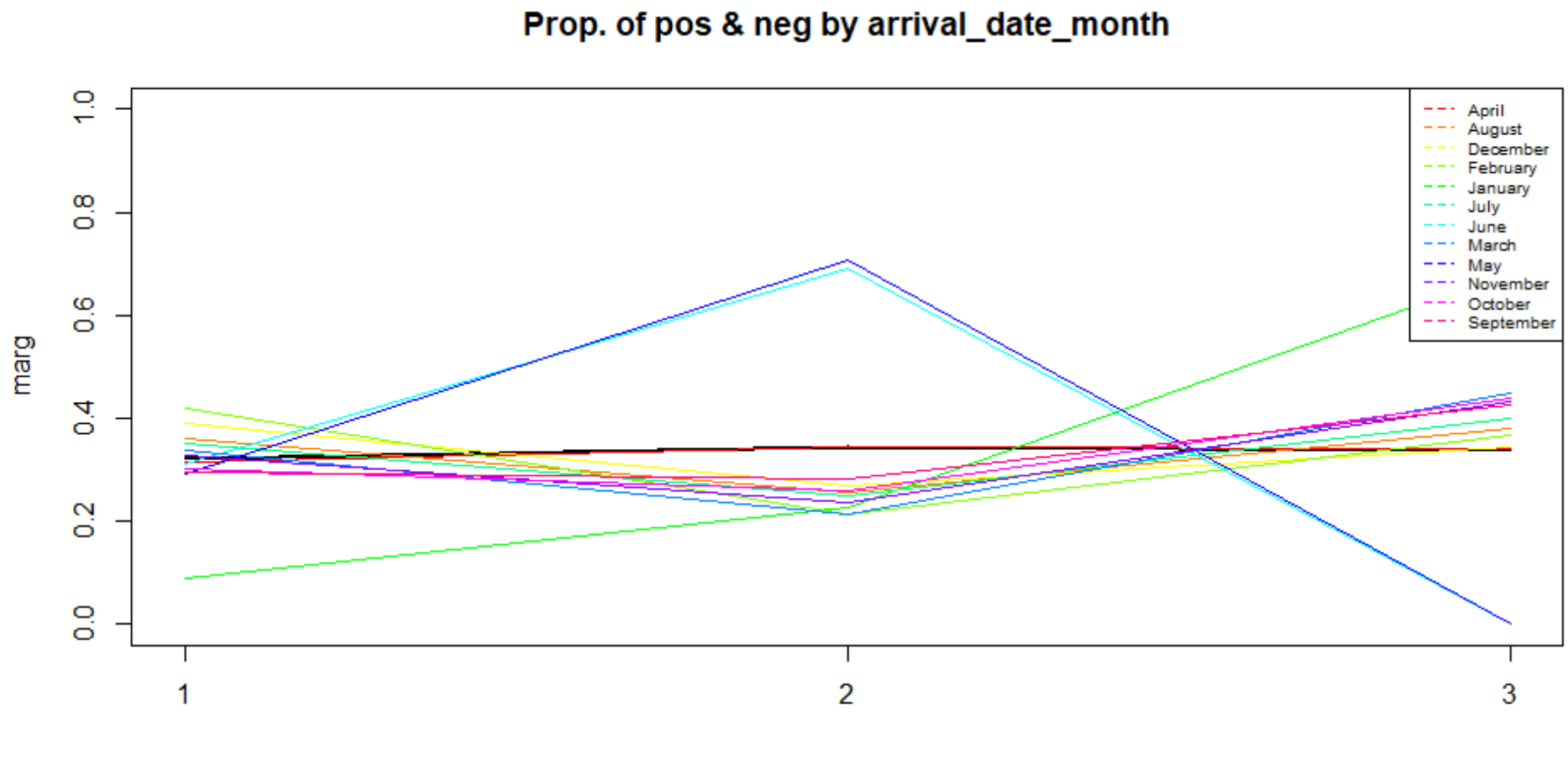
We can conclude that members from cluster 2 are way prone to have really high lead time.

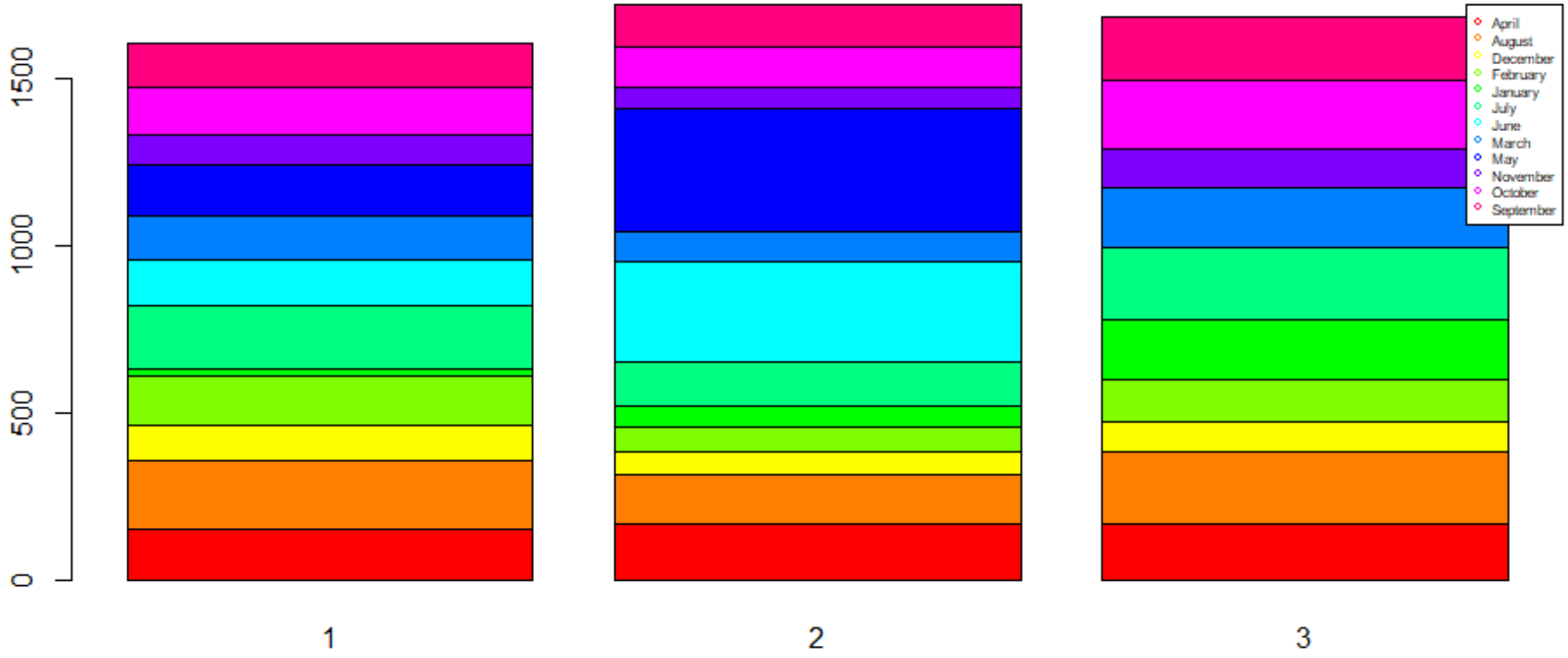




4: ARRIVAL DATE MONTH

In this variable, we can see the distribution of the members of each cluster by the arrival date month. In the first plot, the first thing we notice is that most of the arrivals of the second cluster are distributed between May and June, while in the third one these both are nearly 0. As a consequence of this, the other months have significantly lower values. Aside from this, in general, the first and third clusters are the ones that have the months better distributed. This distribution can be seen better in the bar plot.



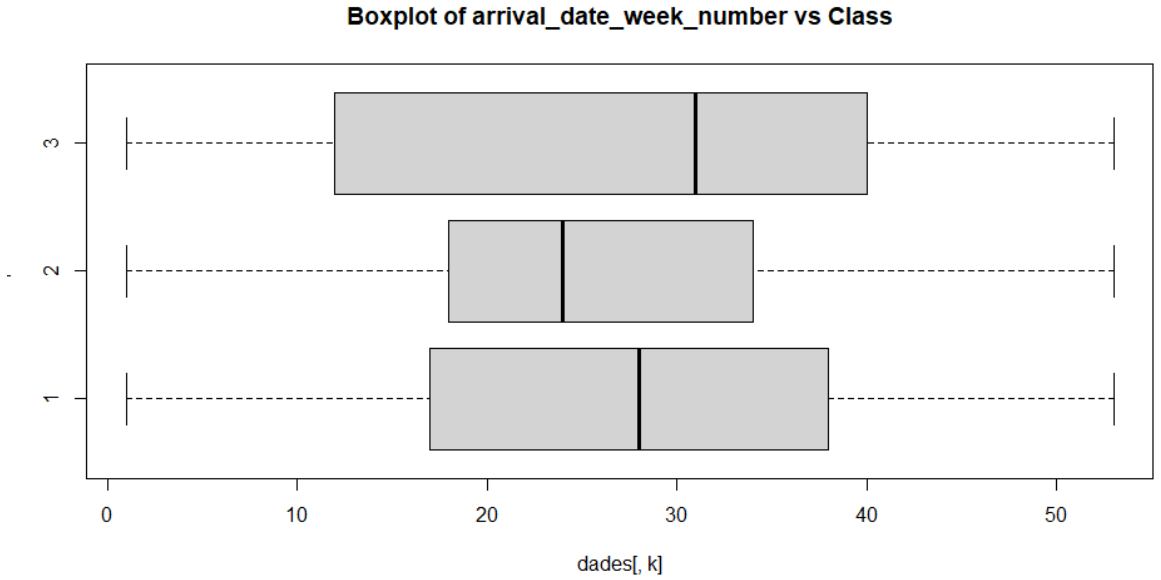

0
500
1000
1500
1
2
3
April
August
December
February
January
July
June
March
May
November
October
September

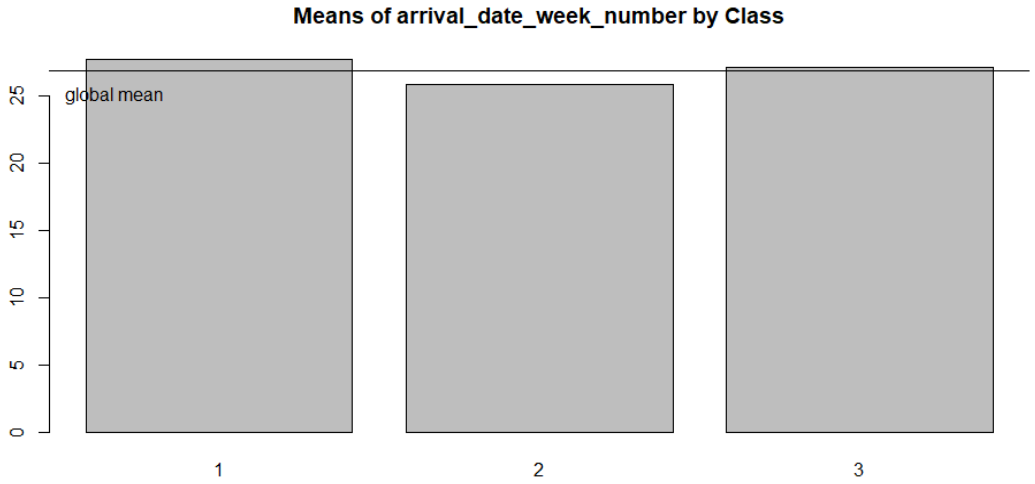
5: ARRIVAL WEEK NUMBER

The next variable to analyze is *arryival_week_number*. In the first plot we can highlight one thing first: the members of cluster 3 are more distributed, covering values between approximately 12 and 41. In addition, it has the highest average values.
Compared to the other clusters, they are less distributed and cover a smaller range of dates, in addition to having a lower average, something notable especially in the 2nd cluster.

In the 2nd graph, we can see that the 2nd cluster does not even touch the global average, while the other two exceed it very slightly.

Since this variable can be expressed in months of the year (seeing each one which week numbers belong to it), these graphs are closely related to those of the previous analysis.

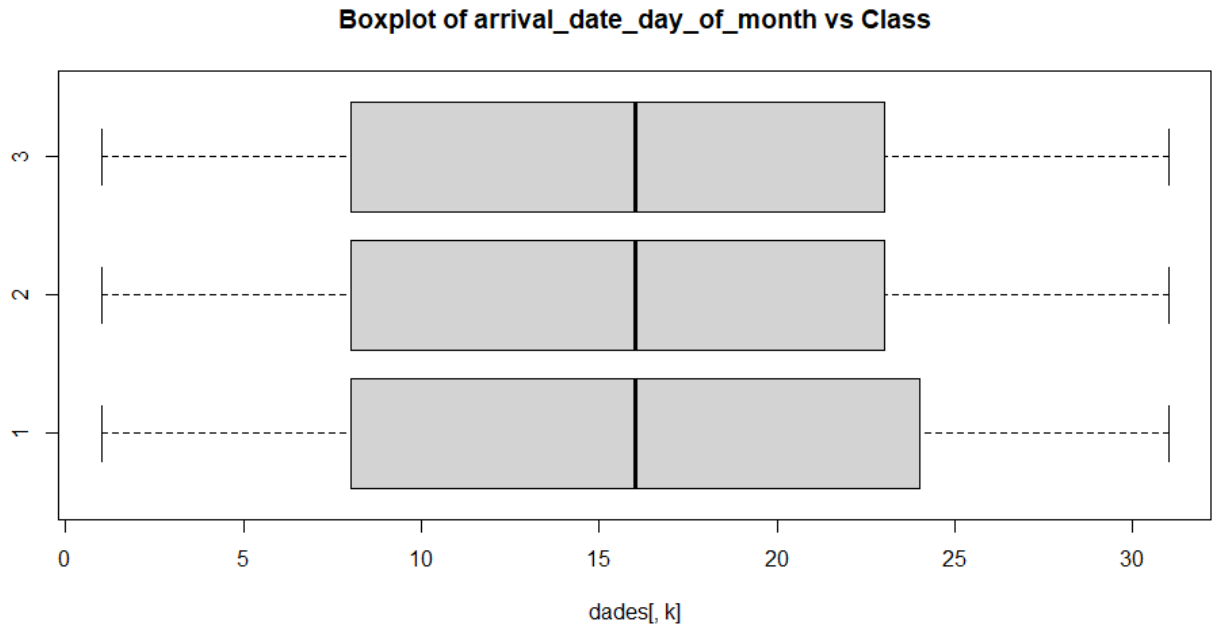


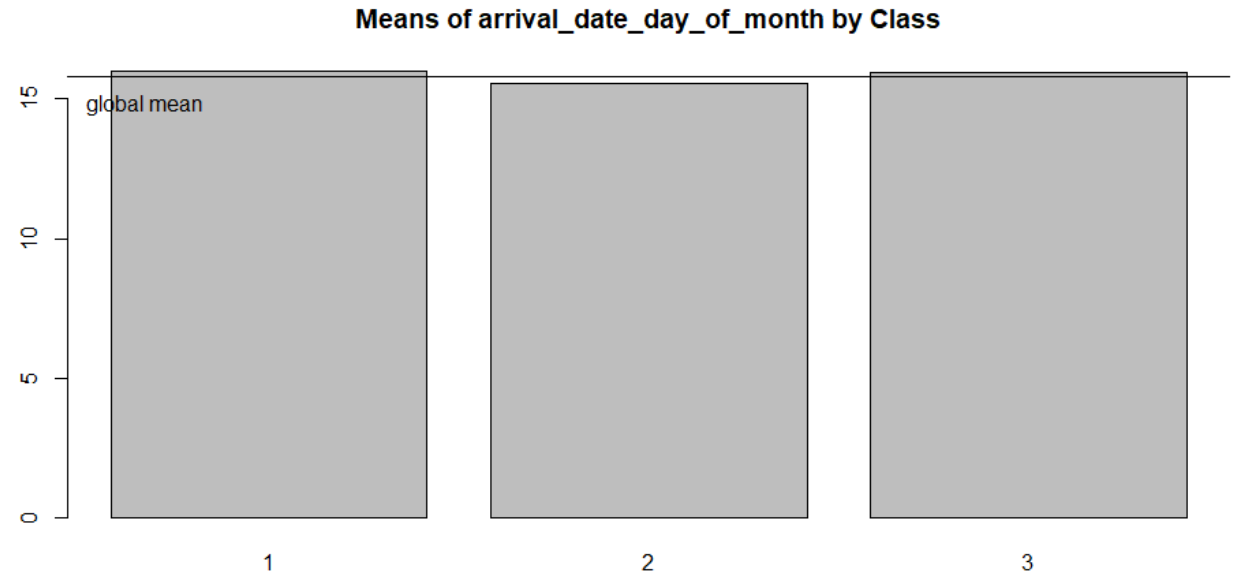


6: ARRIVAL DATE DAY OF MONTH

In the analysis of this variable, the first thing we can highlight is that the members of each and every one of the clusters cover a range of values that are practically the same or very similar. In fact, the 2nd and 3rd cluster are practically identical, covering a range of 6~7 up to 23, while the 1st cluster covers up to 24, a minimal difference. In addition to that, the average of the 1st and 3rd cluster is almost the same, while that of the 2nd cluster does not touch the global average.

It is something interesting because the 2nd and 3rd cluster cover the same range and the 1st of all just a little more.

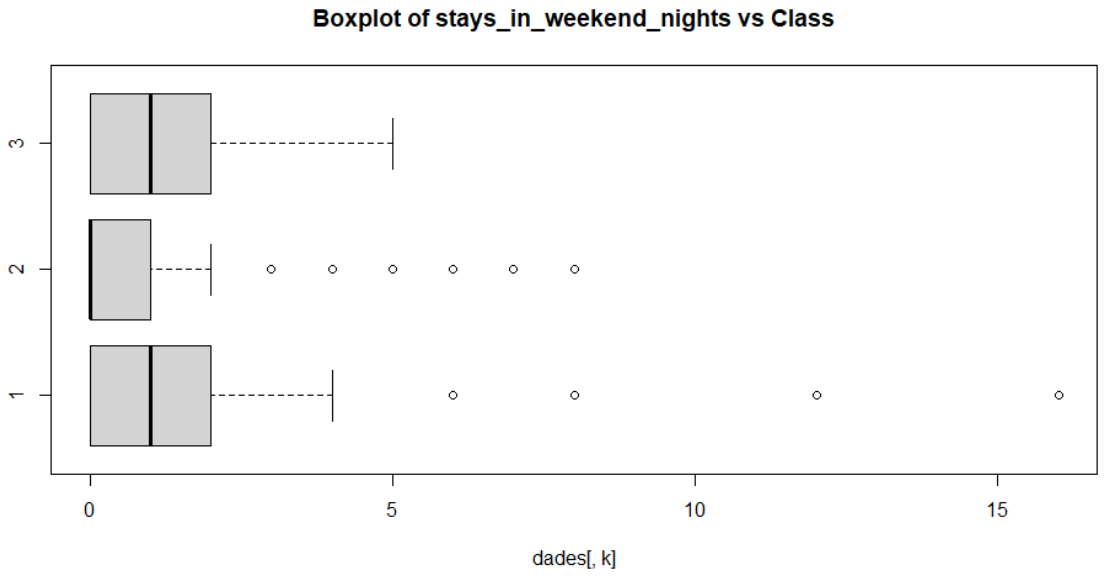


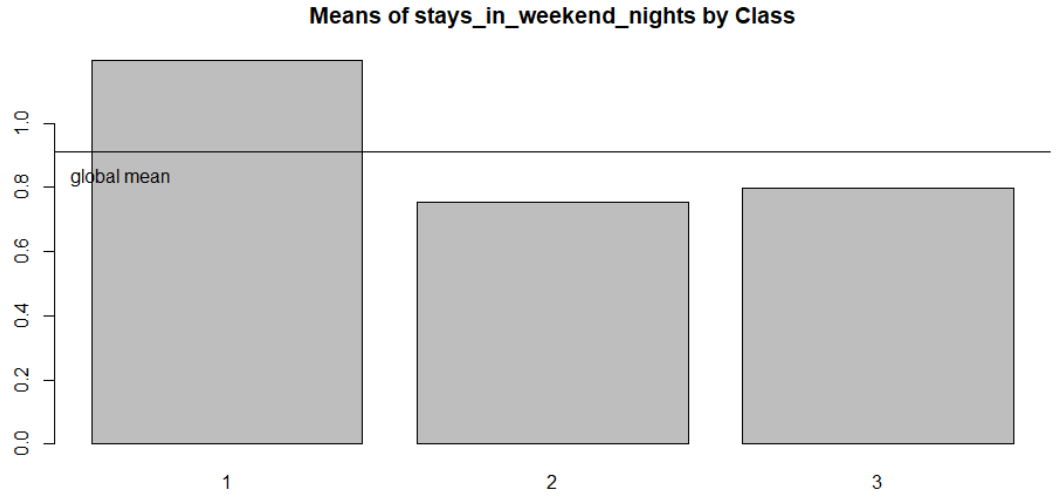


7: STAYS IN WEEKEND NIGHTS

The next variable to analyze is stays_in_weekend_nights. As we have already explained, this variable deals with weekend nights spent in a hotel. So, the first thing we can think is that we are not going to have very high values since people do not usually stay for a very long time. And this is reflected in the graphics. The 1st and 3rd cluster cover a practically equal range of between 0 and 3 nights, while the 2nd only covers 0 to 1. Despite this, in these last two clusters we obtain significant outliers.

Regarding the global averages, we see that the 1st cluster significantly exceeds the global average, while the other two do not even touch the global average, despite what we have observed that the 1st and 3rd groups have a similar boxplot.
We could conclude that individuals in cluster 2 have likely stayed during working days.

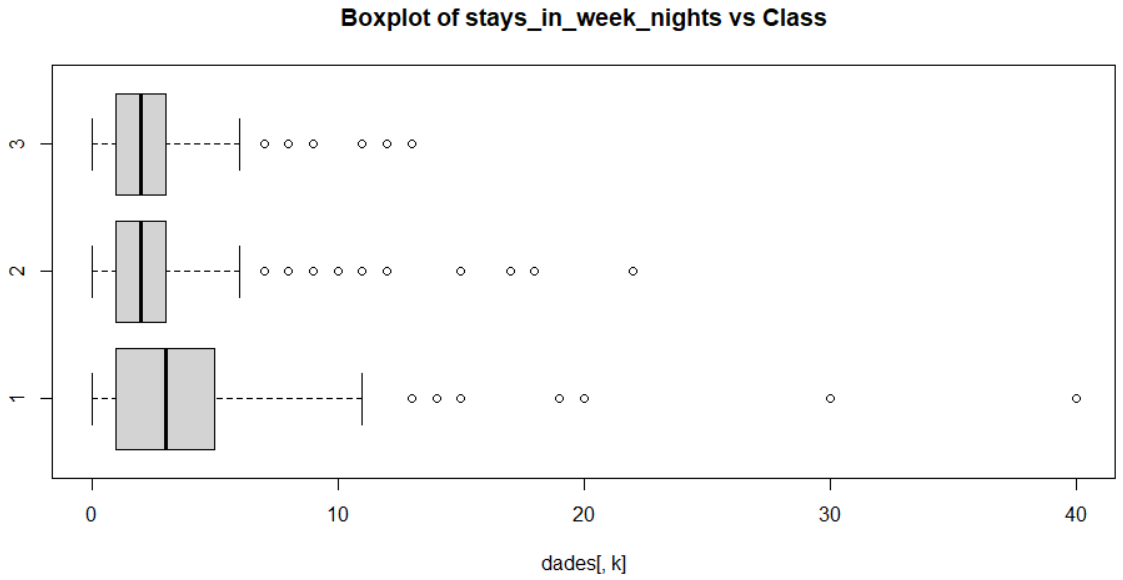




8: STAYS IN WEEK NIGHTS

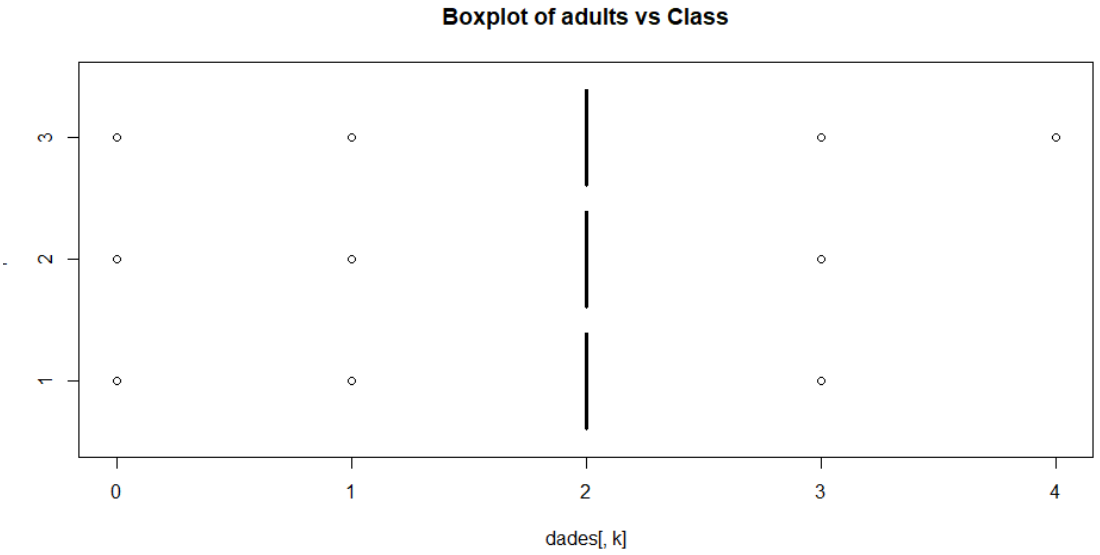
The main difference in the analysis of this variable is that we now deal with nights in general, without specifically referring to weekend nights specifically. The first thing we can observe is that the range covered by all the members of clusters 2 and 3 is practically the same, but both with outliers of different values.
Cluster 1 is the most distributed, although it also has outliers of generally higher values.

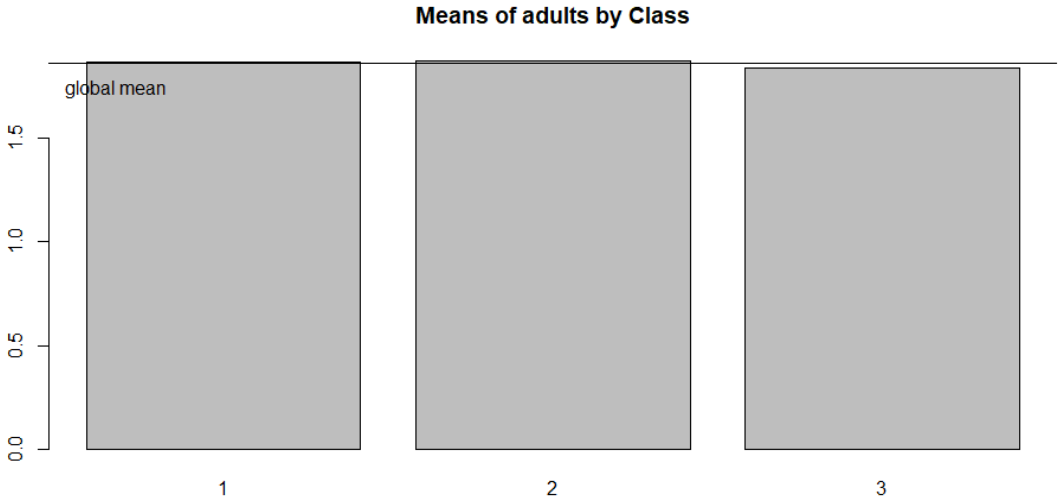
Furthermore, the global means of groups 2 and 3 are somewhat similar and do not reach the global mean, while that of the first cluster clearly exceeds it.



9: ADULTS

For the variable adults, we can see how all the clusters have practically the same shape and cover the same value: 2 adults. Furthermore, they present almost the same outliers: 0, 1, 2 and 3 adults, although in the case of the third cluster, it also has 4. Despite this, all groups have practically the same average, which coincides with the global average: 2.

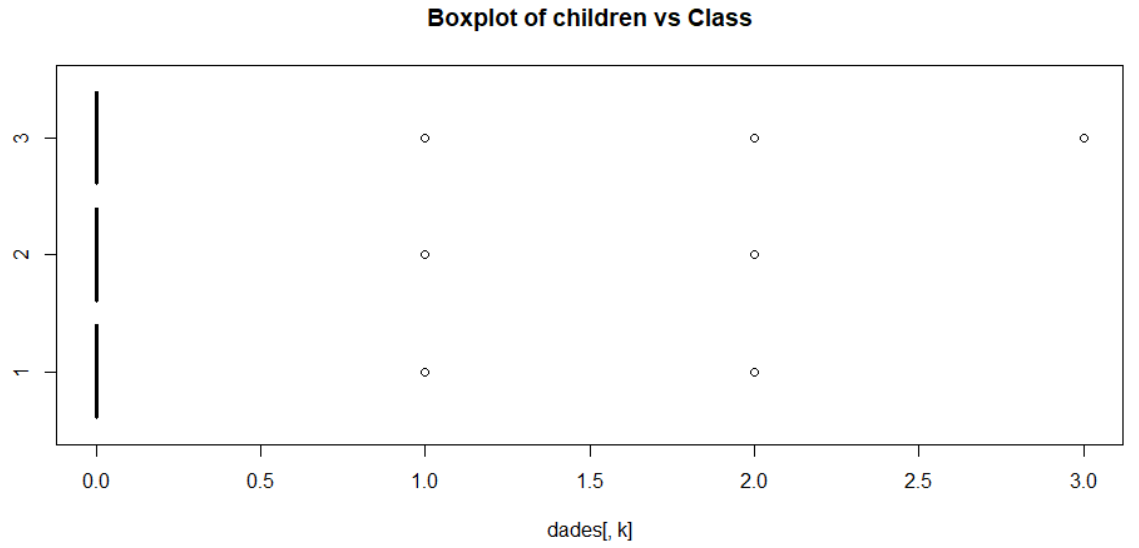


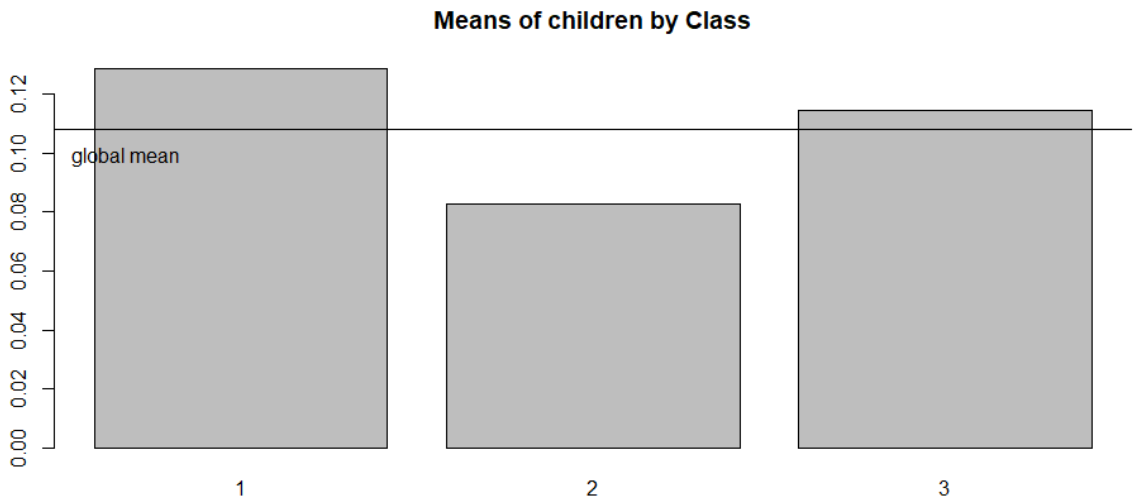


10: CHILDREN

If we now analyze the children variable, we can realize that practically the same thing happens between clusters as the previous variable: they all cover the same value and also present the same outliers, with the exception that the third cluster has one more in 3 childrens . However, we see that the averages are quite different: now they do not all have the same value.

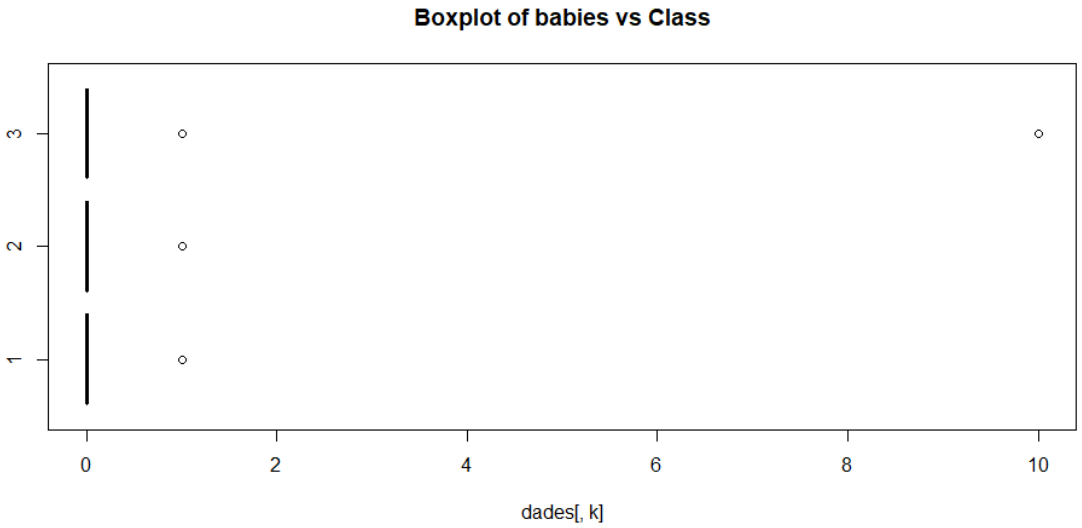
In this case, the averages are quite different from each other. That of the first cluster notably exceeds the global average, while the third slightly exceeds it and the second group barely reaches it.





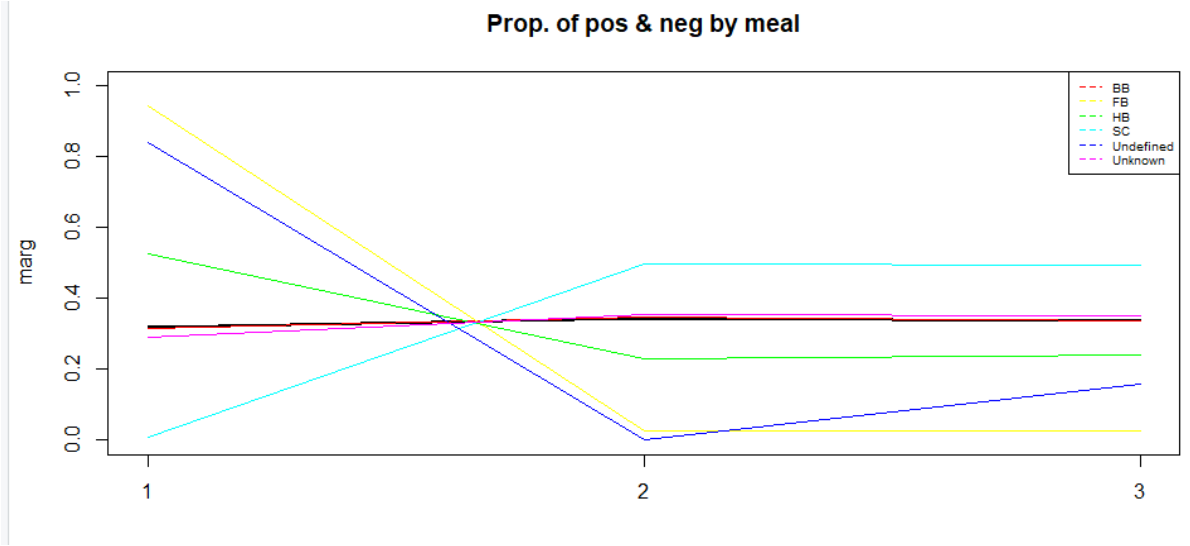
11: BABIES

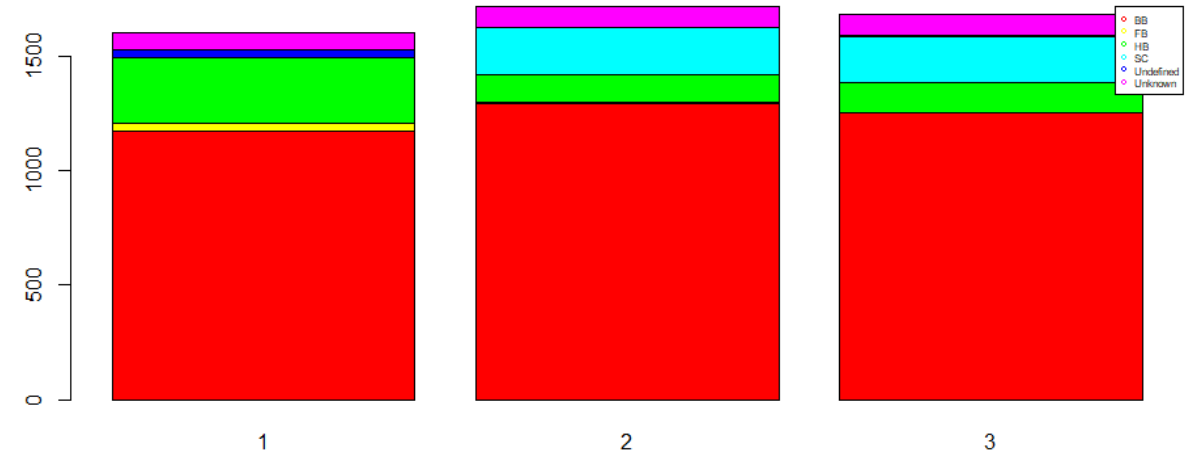
This variable is similar to the previous two, it only ranges from 0 to 1. Because of this the difference between clusters is very low, although, we can see that cluster 2 is the one who has the fewest individuals with babies. Still, the difference is very low as the difference in means is very little.



12: MEAL

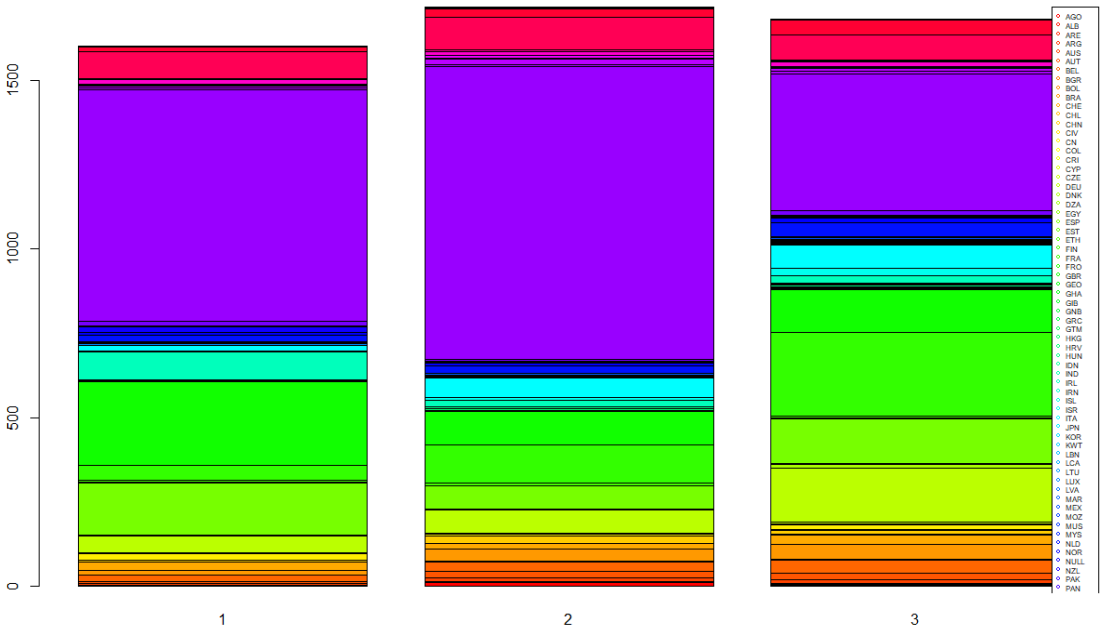
With this variable we can see the distribution of the types of meal among clusters. The majority of values from this variable are “BB”, which are distributed among the three clusters. Despite this we can see that the majority of “undefined” meals are in cluster 1. “FB” meal types are also in cluster 1 in majority, while “SC” meal types are all found in clusters 2 and 3. We could conclude that although there are some visible slopes, those are for those types of meals with less individuals.

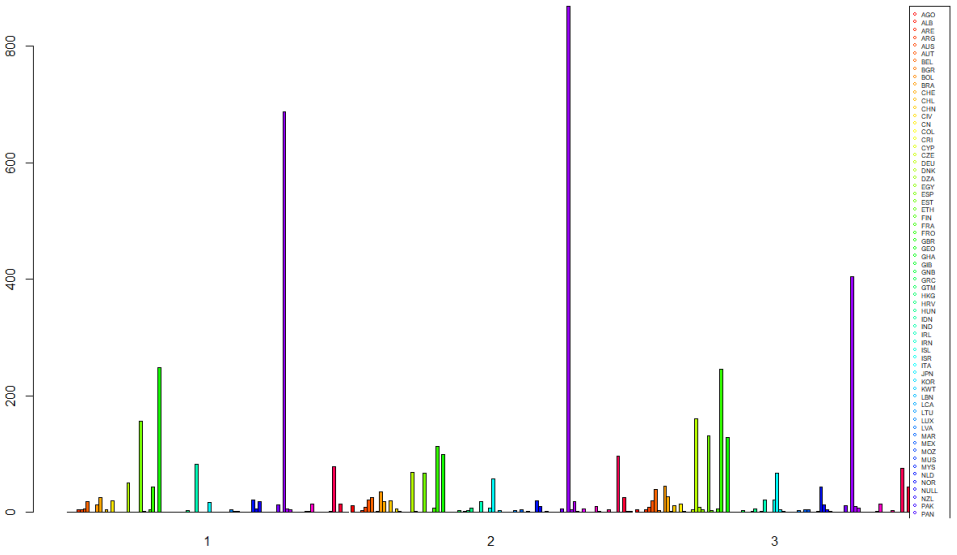




13: COUNTRY

As mentioned in Variable 13: country there are over 90 countries so that is why the plots below have so many modalities. We can see that they are almost quite balancelly separated in the 3 clusters, although the amount of the colour purple (Portugal) is much more significant in the second cluster than the third one, but this is countered with the amount of green section (Great Britain). In the first one there is a smaller percentage of warm colours. However, this does not make much of a difference since the little contribution of them in the graphs.

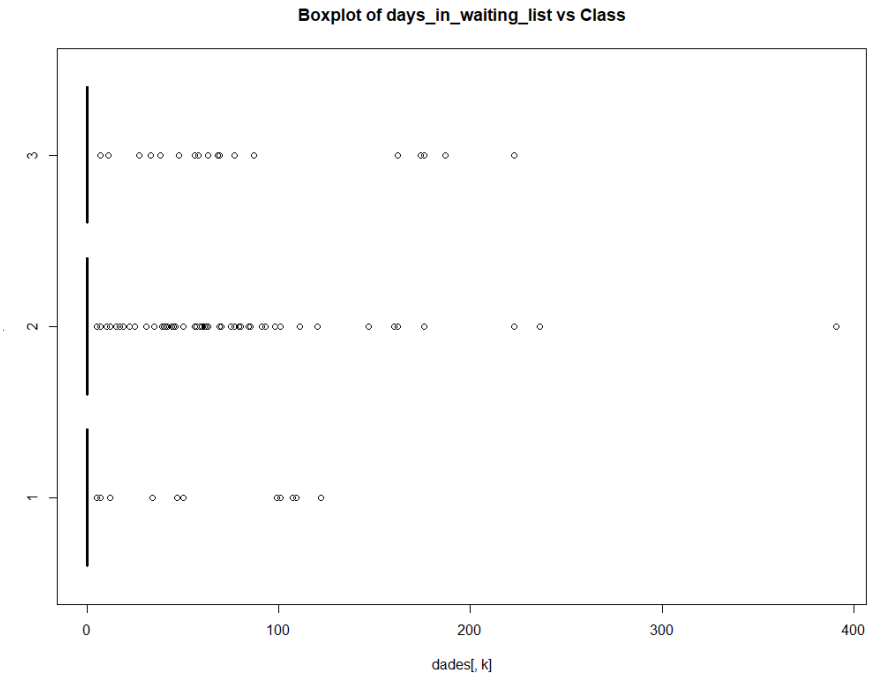


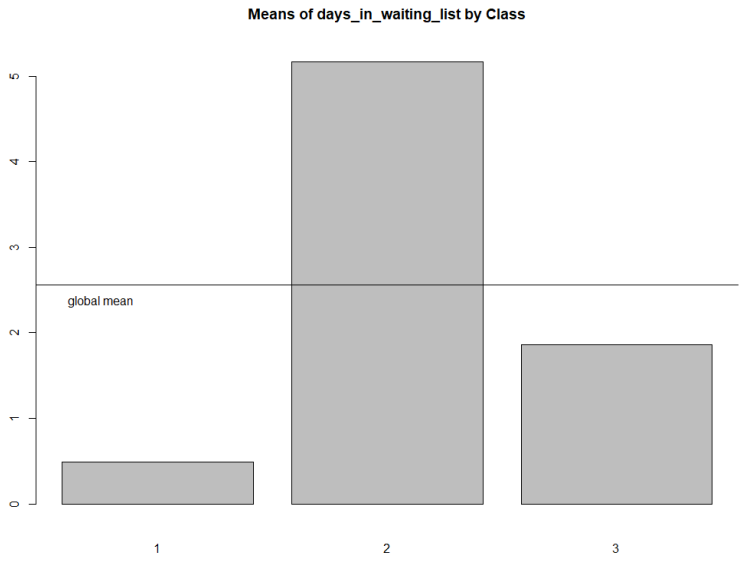


14: DAYS IN WAITING LIST

Now we are going to analyze “days_in_waiting_list”. From first sight we cannot see much with the boxplot, because we have a lot of entries with 0 waiting days. The means can tell us a little bit more, we can see that the majority of individuals with high waiting days belong to cluster 2, while cluster 1 and 3 contain much more individuals with low waiting days. Despite cluster 1 having less median than cluster 3, we can observe in the boxplot that cluster 1 contains higher values than cluster 3. This could mean that cluster 1 contains more individuals and a lot of them with 0 or low waiting days.

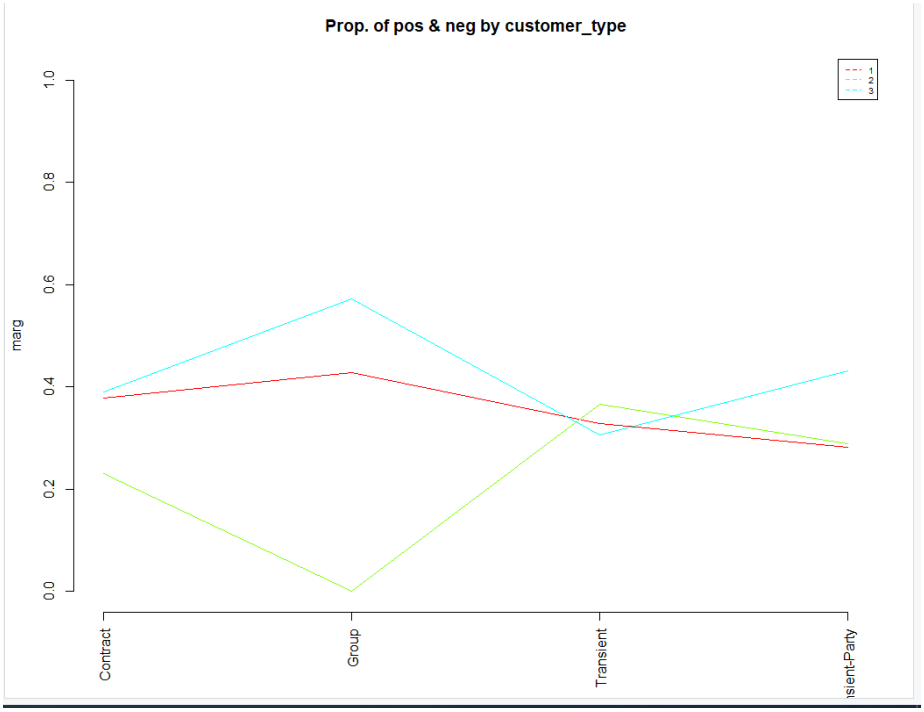
We could conclude that individuals in cluster one and three haven’t had a long waiting time, while cluster 2 contains individuals with higher waiting times.

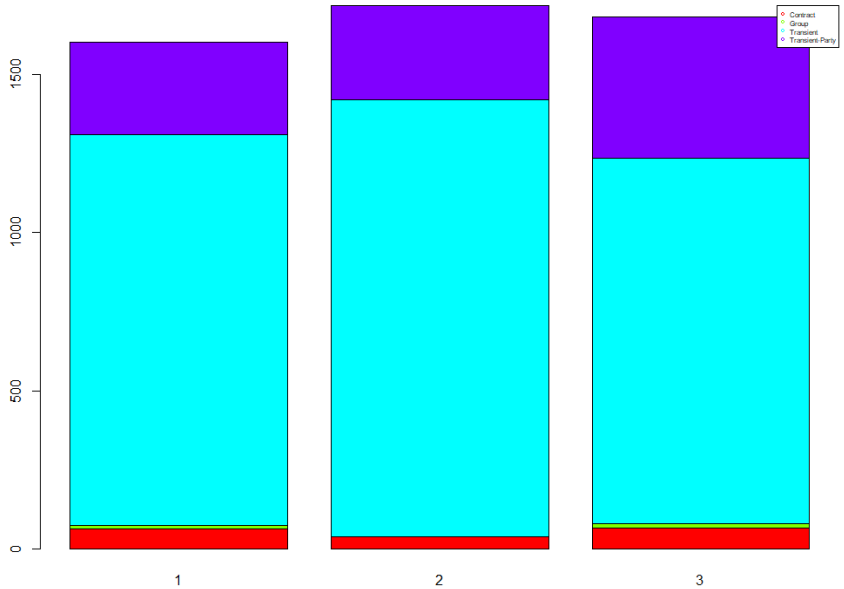




15: CUSTOMER TYPE

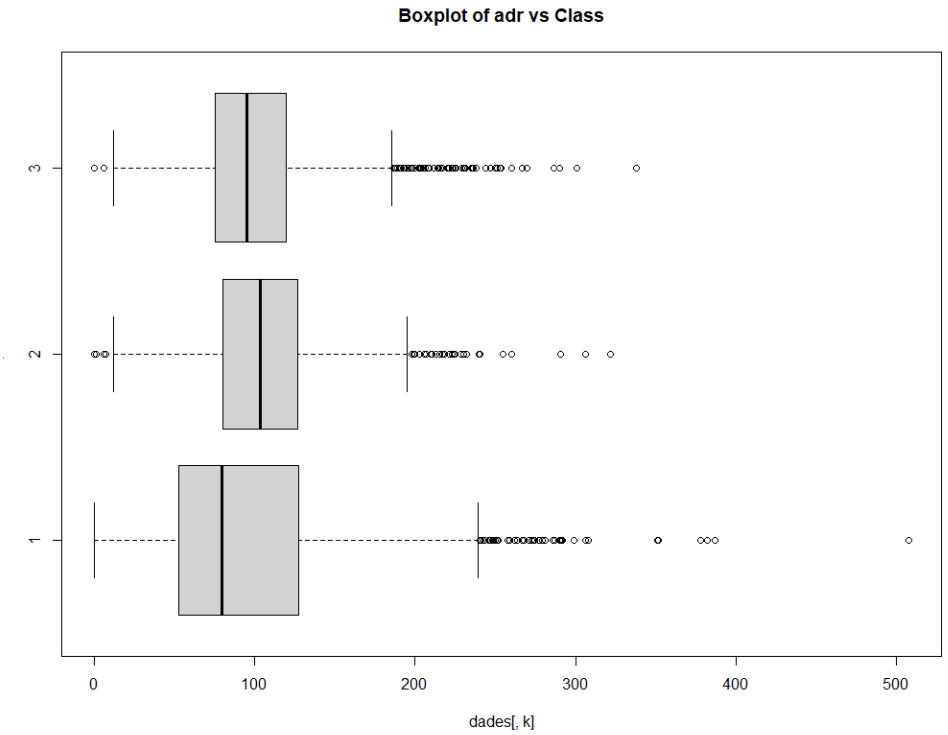
For this variable there's almost no difference between clusters. The only appreciable detail is the inexistence of type "group" of customer on cluster 2. For the other two clusters, the distribution of values is pretty similar. So the difference between clusters is not significant.

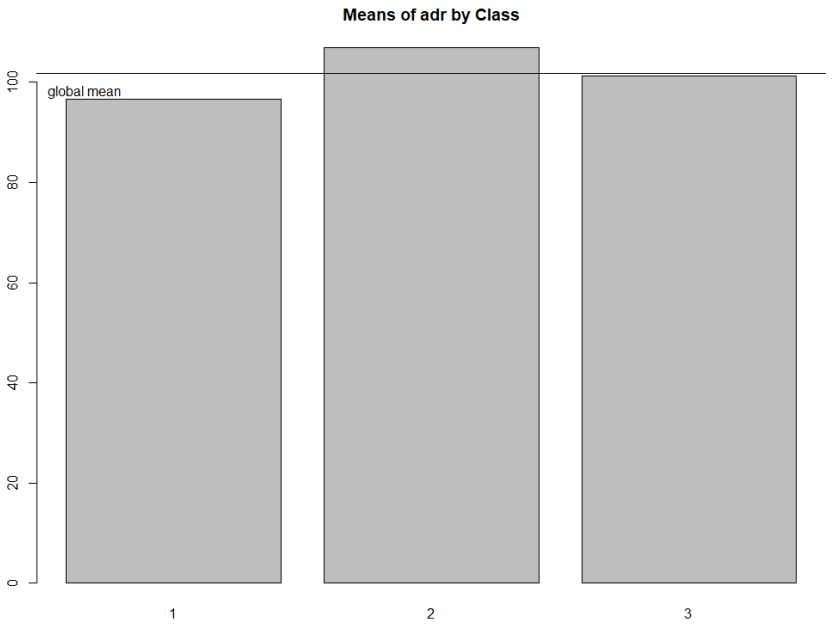




16: ADR

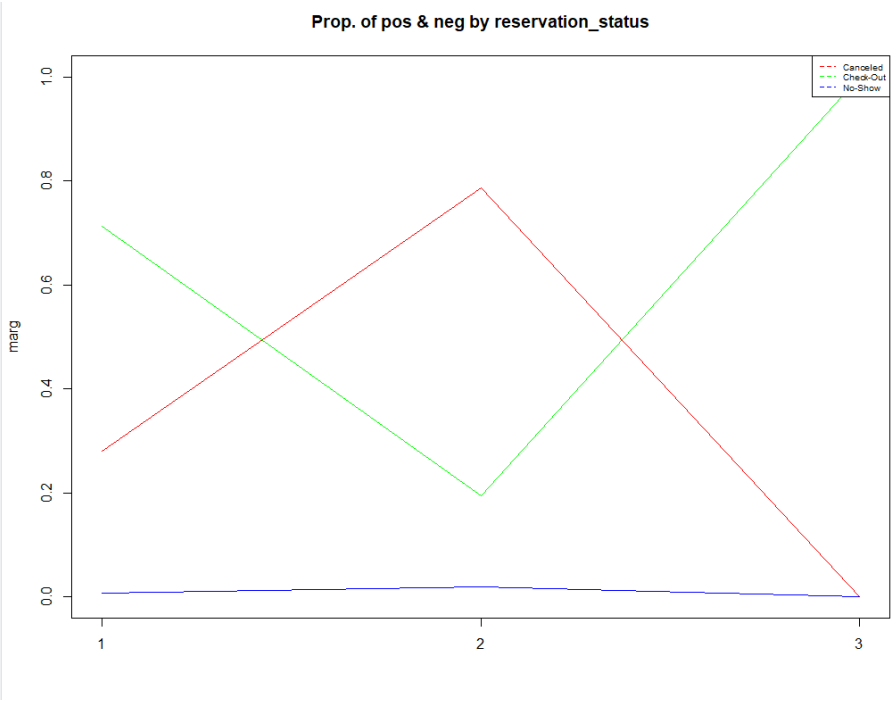
In the clustering section we said that this variable could give us information. Now if we analyze we cannot get any good results. The ranges of the three clusters are very similar, although the third cluster is likely to be in a bigger and lower range. Their medians are all similar too, going from 95 to 105. Then, the difference between clusters is not significant.

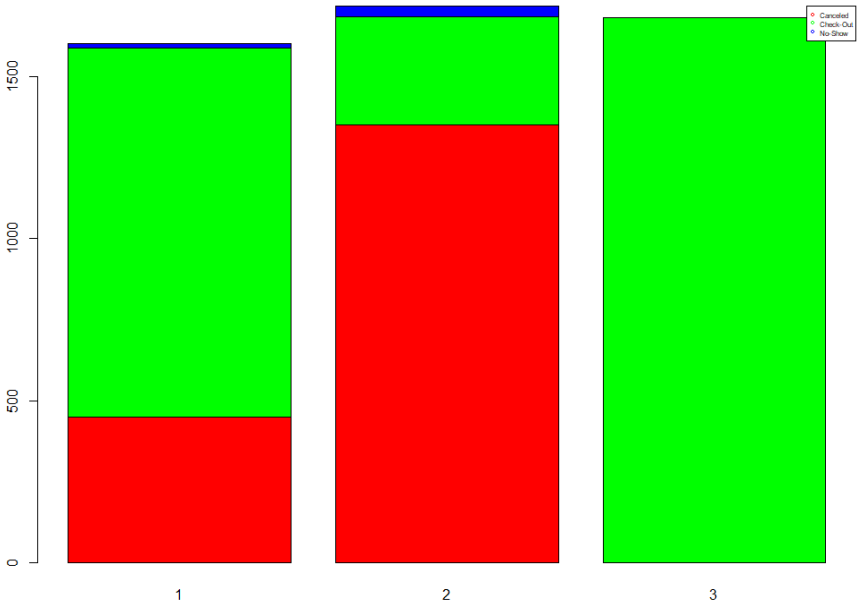




17: RESERVATION_STATUS

If we analyze this variable we can observe in plot one that cluster 3 is fully dedicated to “check-out” values, although these values are on cluster 1 and 2 in a major way too as it can be seen in the bar plot. Apart from this fact, “canceled” values tend to be in cluster 2 more than in cluster 1.





9.2. Conclusions

We used this method in order to separate individuals into different groups with similar characteristics. To know how many groups we could have, we used the gower method to cluster our data. We concluded that we had three clusters, but visualizing the data separately we couldn't see much differentiation, meaning that our data hadn't much relation between individuals. In order to try to extract some more information we did the analysis with hierarchical clustering, using only numerical values. We could see some information with this method regarding some of our variables. To know if our data could be separated into clusters with similar characteristics we had to do the profiling of our variables using the clusters found. From this process we couldn't extract much information, but some observations can be done.

From variables lead_time and is_cancelled, we can conclude that those individuals with high lead_time values are likely to cancel its reservation, or they tend to do it more.

Following this observation of lead_time, if we look at the variable "arrival_date_of_month", we can conclude that those reservations in summer are likely to be reserved with higher anticipation. A similar case happens with "days_in_waiting_list", it is likely to be more days in the waiting list if you make a reservation for summer.

In conclusion, we cannot say much about clustering, the relation between individuals is not big enough to extract real conclusions. The only perceptible relation is that those reservations around summer have higher waiting days and anticipation on reservations. This is reasonable as the summer is the most profitable time of the year for a hotel.

10 - Conclusions of the project

After reviewing the analysis of our dataset, we can draw some conclusions. Following the analysis of our process, we can categorize our results into different sections. The three main components of our work have been preprocessing, PCA, and Clustering.

PREPROCESSING PROCESS CONCLUSION

The first part of our project involved preprocessing, which included the extraction of metadata used in our analysis. Initially, we encountered no issues with the format and selecting the working matrix, eliminating redundant or irrelevant information, and reducing the initial 32 variables to 17, which provided us with more informative data.

One of our project's objectives was to address missing values, so we intentionally introduced them. After their creation, we employed two different methods for filling these null values: KNN and MIMMI. We tested both methods and determined which one provided better results by comparing them to our original dataset.

The final step in the preprocessing phase involved identifying outliers and errors. We encountered no errors, and we were able to clearly identify the outliers. Besides, we decided to retain all instances in the dataset because we had confidence in the accuracy of our data.

Furthermore, we need to compare the two methods we used for analyzing our dataset: PCA and Clustering.

PCA vs CLUSTERING

For this part, as we mentioned multiple times in the former sections, there are no clear conclusions for each method. Since we have chosen a poor database, we could not see much relationship between the individuals. The unsatisfactory results could be the one thing that we have most in common for both techniques. Neither of these methods has provided us with a relevant understanding of the habits or preferences of the tourists.

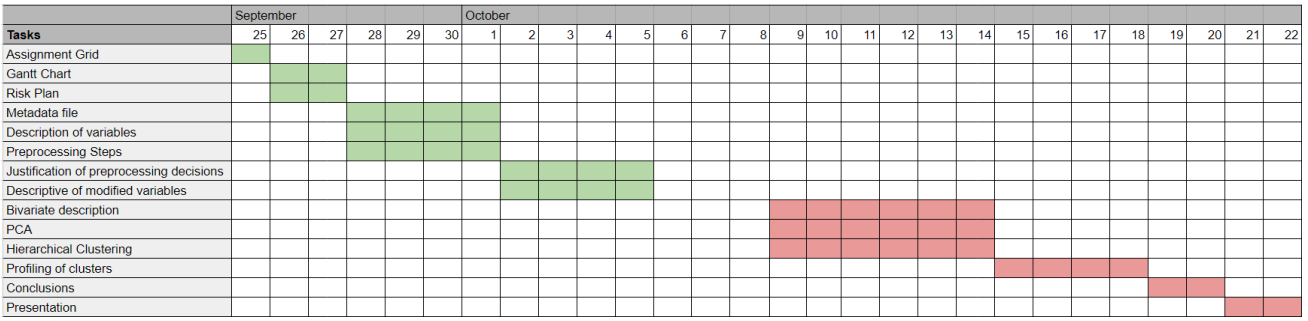
However, in the analysis, PCA helped the most in understanding travel patterns, customer types, lead time, and seasonal reservations. Clustering, on the other hand, only revealed a potential relation between reservations made around summer and higher waiting days and anticipation. While both have assisted us and helped us into knowing more about our

database, the conclusions show that PCA has been more effective in extracting meaningful insights (compared to our Clustering interpretations) whereas the latter one had limited utility for drawing clearer conclusions.

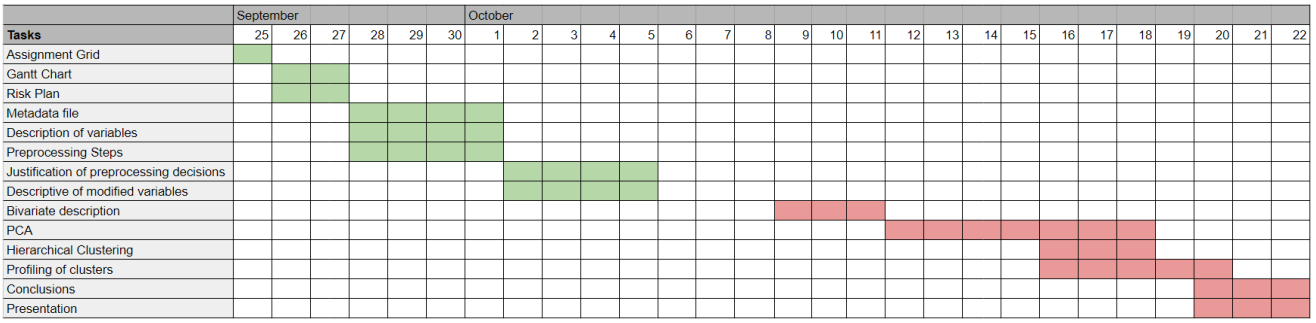
11 - Working plan

11.1. Initial and final Gantt

The initial Gantt diagram we prepared is the following:



Finally, the real one has been:



11.2. Final tasks assignment grid

This is how all the tasks have been assigned for deliveries D3 (Gantt - Additional descriptive statistics) and D4 (Data Mining process - Presentation):

Members / Tasks	Natalia	Gerard	Mario	Adrià	Tomàs
Gantt			x		
Division of tasks	x				
Contingency plan					x
Metadata file		x	x	x	
Univariate descriptive	x	x		x	
Preprocessing steps			x	x	
Preprocessing steps justification	x	x			x
Additional descriptive statistics	-				
Data Mining process	x				
Bivariate description				x	
PCA	x	x		x	
Clustering			x		x
Profiling of clusters	x	x	x		x
Conclusion	x	x	x	x	x
Presentation	x	x	x	x	x

11.3. Deviances and risks avoided

We did not encounter any issues regarding risks presented in the contingency plan.

Annex: Code fragments

Creating Missing Values

Python

```
dd <- read.csv("path",header=T)

names(dd)
dim(dd)

# Lista de columnas en las que deseas aplicar cambios
columnas_a_modificar <- c("lead_time", "arrival_date_week_number", "meal",
"country", "days_in_waiting_list", "adr")


porcentaje <- 5
num_valores_a_convertir <- ceiling((porcentaje / 100) * nrow(dd))


for (col in columnas_a_modificar) {
 indices_a_convertir <- sample(1:nrow(dd), num_valores_a_convertir)
 dd[indices_a_convertir, col] <- NA
}


write.csv(dd, "path", row.names = FALSE)

#Para comprobar que hay 250 missing values en las columnas especificadas:

install.packages("dplyr")

library(dplyr)


missing_values <- dd %>%
 summarise_all(funs(sum(is.na(.))))


print(missing_values)
```

Filling Missing Categorical Values

Python

```
num_rows <- nrow(dd)

for (i in 1:num_rows) {
 if (is.na(dd[i, "arrival_date_week_number"])) {
       y = dd[i, "arrival_date_year"]
       month = dd[i, "arrival_date_month"]
       m = match(month, month.name)
       d = dd[i, "arrival_date_day_of_month"]
       date = sprintf("%s-%s-%s", y, m, d)
       week_num = strftime(date, format="%V")
       dd[i, "arrival_date_week_number"] <- week_num
 }
}

for (i in 1:num_rows) {
 if (is.na(dd[i, "country"])) {
       dd[i, "country"] <- "Unknown"
 }
}

for (i in 1:num_rows) {
 if (is.na(dd[i, "meal"])) {
       dd[i, "meal"] <- "Unknown"
 }
}

write.csv(dd, "path", row.names = FALSE)
```

PCA

Python

```
#Numerical variables
numeriques<-c(3,7,8,9,10,11,14,16)
numeriques

dcon<-dd[,numeriques]

# PRINCIPAL COMPONENT ANALYSIS OF dcon
pc1 <- prcomp(dcon, scale=TRUE)
class(pc1)
attributes(pc1)

#[...]
#numnericals and categoricals
plot(Psi[,eje1],Psi[,eje2],type="n",xlim=c(-1,1),
ylim=c(-1,1))axis(side=1, pos= 0, labels = F, col="lightgray")
axis(side=3, pos= 0, labels = F, col="lightgray")
axis(side=2, pos= 0, labels = F, col="lightgray")
axis(side=4, pos= 0, labels = F, col="lightgray")

arrows(ze, ze, X, Y, length = 0.07,col="cyan")
text(X,Y,labels=etiq,col="cyan", cex=0.7)

c<-1
for(k in dcat){
 seguentColor<-colors[c]

 fdic1 = tapply(Psi[,eje1],dd[,k],mean)
 fdic2 = tapply(Psi[,eje2],dd[,k],mean)
text(fdic1,fdic2,labels=levels(factor(dd[,k])),col=seguentColo
r, cex=0.6)
 c<-c+1
}
legend("bottomleft",names(dd)[dcat],pch=1,col=colors, cex=0.6)
```

Clustering and profiling script

Python

```
# HIERARCHICAL CLUSTERING

d <- dist(dcon[1:50,])
h1 <- hclust(d,method="ward.D") # NOTICE THE COST
plot(h1)

d <- dist(dcon)
h1 <- hclust(d,method="ward") # NOTICE THE COST
plot(h1)

# BUT WE ONLY NEED WHERE THERE ARE THE LEAPS OF THE HEIGHT

# WHERE ARE THER THE LEAPS? WHERE WILL YOU CUT THE DENDREOGRAM?, HOW MANY
CLASSES WILL YOU OBTAIN?

nc = 3

c1 <- cutree(h1,nc)

# c1 dice a que cluster pertenece cada variable
c1


table(c1)


cdg <- aggregate(as.data.frame(dcon),list(c1),mean)
cdg

plot(cdg[,1], cdg[,7])

pairs(dcon[,1:8], col=c1)

# QUALITY OF THE HIERARCHICAL PARTITION
Bss <- sum(rowSums(cdg^2)*as.numeric(table(c1)))

Ib4 <- 100*Bss/Tss
Ib4

# GOWER

library(cluster)
# dcon <- data.frame
(hotel,lead_time,arrival_date_month,arrival_date_week_number,arrival_date_da
y_of_month,stays_in_weekend_nights,stays_in_week_nights,adults,children,babi
es,meal,country,days_in_waiting_list,customer_type,adr,reservation_status)
dim(dcon)
```

```r
#dissimilarity matrix

actives<-c(1:8)
dissimMatrix <- daisy(dd[,actives], metric = "gower", stand=TRUE)

distMatrix<-dissimMatrix^2

h1 <- hclust(distMatrix,method="ward.D")  # NOTICE THE COST
#versions noves "ward.D" i abans de plot: par(mar=rep(2,4)) si se quejara de
los margenes del plot

plot(h1)

c2 <- cutree(h1,3)

#class sizes
table(c2)

#comparing with other partitions
table(c1,c2)

names(dd)

pairs(dcon[,1:8], col=c2)

cdg <- aggregate(as.data.frame(dcon),list(c2),mean)
cdg

potencials<-c(1,2,3,4,5,6,7,8)
pairs(dcon[,potencials],col=c2)

# PROFILING

#Calcula els valor test de la variable Xnum per totes les modalitats del
factor P
ValorTestXnum <- function(Xnum,P){
  #freq dis of fac
  nk <- as.vector(table(P));
  n <- sum(nk);
  #mitjanes x grups
  xk <- tapply(Xnum,P,mean);
  #valors test
  txk <- (xk-mean(Xnum))/(sd(Xnum)*sqrt((n-nk)/(n*nk)));
  #p-values
  pxk <- pt(txk,n-1,lower.tail=F);
  for(c in 1:length(levels(as.factor(P)))){if
(pxk[c]>0.5){pxk[c]<-1-pxk[c]}}
  return (pxk)
```

```
}

ValorTestXquali <- function(P,Xquali){
  taula <- table(P,Xquali);
  n <- sum(taula);
  pk <- apply(taula,1,sum)/n;
  pj <- apply(taula,2,sum)/n;
  pf <- taula/(n*pk);
  pjm <- matrix(data=pj,nrow=dim(pf)[1],ncol=dim(pf)[2], byrow=TRUE);
  dpf <- pf - pjm;
  dvt <- sqrt(((1-pk)/(n*pk))%*%t(pj*(1-pj)));
  #i hi ha divisions iguals a 0 dona NA i no funciona
  zkj <- dpf
  zkj[dpf!=0]<-dpf[dpf!=0]/dvt[dpf!=0];
  pzkj <- pnorm(zkj,lower.tail=F);
  for(c in 1:length(levels(as.factor(P)))){for (s in
1:length(levels(Xquali))){if (pzkj[c,s]> 0.5){pzkj[c,s]<-1- pzkj[c,s]}}}
  return (list(rowpf=pf,vtest=zkj,pval=pzkj))
}

dades<-dd
K<-dim(dades)[2]
par(ask=TRUE)


#P must contain the class variable
P<-c2
nameP<-"classe"

nc<-length(levels(factor(P)))
nc
pvalk <- matrix(data=0,nrow=nc,ncol=K,
dimnames=list(levels(P),names(dades)))
nameP<-"Class"
n<-dim(dades)[1]

for(k in 1:K){
  if (is.numeric(dades[,k])){
    print(paste("Anàlisi per classes de la Variable:", names(dades)[k]))

    boxplot(dades[,k]~P, main=paste("Boxplot of", names(dades)[k], "vs",
nameP ), horizontal=TRUE)

    barplot(tapply(dades[[k]], P, mean),main=paste("Means of",
names(dades)[k], "by", nameP ))
    abline(h=mean(dades[[k]]))
    legend(0,mean(dades[[k]]),"global mean",bty="n")
    print("Estadístics per groups:")
```

```
    for(s in levels(as.factor(P))) {print(summary(dades[P==s,k]))}
    o<-oneway.test(dades[,k]~P)
    print(paste("p-valueANOVA:", o$p.value))
    kw<-kruskal.test(dades[,k]~P)
    print(paste("p-value Kruskal-Wallis:", kw$p.value))
    pvalk[,k]<-ValorTestXnum(dades[,k], P)
    print("p-values ValorsTest: ")
    print(pvalk[,k])
  }else{
    if(class(dd[,k])=="Date"){
      print(summary(dd[,k]))
      print(sd(dd[,k]))
      #decide breaks: weeks, months, quarters...
      hist(dd[,k],breaks="weeks")
    }else{
      #qualitatives
      print(paste("Variable", names(dades)[k]))
      table<-table(P,dades[,k])
      rowperc<-prop.table(table,1)

      colperc<-prop.table(table,2)

      dades[,k]<-as.factor(dades[,k])

      marg <- table(as.factor(P))/n
      print(append("Categories=",levels(as.factor(dades[,k]))))

      #from next plots, select one of them according to your practical case
      plot(marg,type="l",ylim=c(0,1),main=paste("Prop. of pos & neg
by",names(dades)[k]))
      paleta<-rainbow(length(levels(dades[,k])))
      for(c in 1:length(levels(dades[,k]))){lines(colperc[,c],col=paleta[c])
}
      legend("topright", levels(dades[,k]), col=paleta, lty=2, cex=0.6)

      #condicionades a classes
      plot(marg,type="n",ylim=c(0,1),main=paste("Prop. of pos & neg
by",names(dades)[k]))
      paleta<-rainbow(length(levels(dades[,k])))
      for(c in 1:length(levels(dades[,k]))){lines(rowperc[,c],col=paleta[c])
}
      legend("topright", levels(dades[,k]), col=paleta, lty=2, cex=0.6)

      #amb variable en eix d'abcisses
      marg <-table(dades[,k])/n
      print(append("Categories=",levels(dades[,k])))
      plot(marg,type="l",ylim=c(0,1),main=paste("Prop. of pos & neg
by",names(dades)[k]), las=3)
```

```
      for(c in
1:length(levels(as.factor(P)))){lines(rowperc[c,],col=paleta[c])}
      legend("topright", levels(as.factor(P)), col=paleta, lty=2, cex=0.6)

      #condicionades a columna
      plot(marg,type="n",ylim=c(0,1),main=paste("Prop. of pos & neg
by",names(dades)[k]), las=3)
      for(c in
1:length(levels(as.factor(P)))){lines(colperc[c,],col=paleta[c])}
      legend("topright", levels(as.factor(P)), col=paleta, lty=2, cex=0.6)

      table<-table(dades[,k],P)
      print("Cross Table:")
      print(table)
      print("Distribucions condicionades a columnes:")
      print(colperc)

      #diagrames de barres apilades
      paleta<-rainbow(length(levels(dades[,k])))
      barplot(table(dades[,k], as.factor(P)), beside=FALSE,col=paleta )
      legend("topright",levels(as.factor(dades[,k])),pch=1,cex=0.5,
col=paleta)

      #diagrames de barres adosades
      barplot(table(dades[,k], as.factor(P)), beside=TRUE,col=paleta)
      legend("topright",levels(as.factor(dades[,k])),pch=1,cex=0.5,
col=paleta)

      print("Test Chi quadrat: ")
      print(chisq.test(dades[,k], as.factor(P)))

      print("valorsTest:")
      print( ValorTestXquali(P,dades[,k]))
      #calcular els pvalues de les quali
    }
  }
}#endfor
```