Università degli studi di Milano

Data Science for Economics

Statistical Learning Project

Price prediction in Restaurant awarded by Michelin Guide

Student: Chiara Anni

Registration number: 28503A

Academic Year: 2022/2023

Contents

1. Abstract

The aim of this project is to predict the medium price for restaurants that have been awarded by Michelin Guide in 2019 and 2021. The dataset has its main informations in the location of the restaurants around the word, expressed by latitude and longitude, and the number of Michelin's stars received. It is built by the cleaning and union of four datasets. This is done because of the curiosity also in the analysis about Italian restaurants.
The final dataset has a lot of observations, but really few features.

2. The Dataset

The final dataset is built on the union of four.
The first one (*ds1*) can be downloaded at the following Kaggle's link:

https://www.kaggle.com/datasets/dimitrisangelide/michelin-star-restaurants-2021

It is the most consistent in terms of number of rows, with 3155 observations. The entire dataset presents the following 14 columns:

- Restaurant: name of the restaurant.
- Link
- Michelin guide: the national Michelin guide who judged the restaurant.
- Address: complete address of the structure, with street, city, cap and nation.
- Price: price range that you could spend in this restaurant.
- Cuisine: type of cuisine they prepare.
- Michelin Guide point of view: brief comment on the Guide's experience.
- Facilities & services: services that the structure offers.
- Contact number
- Website
- Opening hours
- Michelin stars: number of stars obtained by the structure.
- Location_lat: latitude of the restaurant.
- Location_long: longitude of the restaurant.

The second (*star1*), third (*star2*) and fourth (*star3*) datasets could be downloaded at the following Kaggle's link:

https://www.kaggle.com/datasets/jackywang529/michelin-restaurants

They have all the same structure; they are divided only based on the number of their Michelin stars. They have respectively 549, 110 and 36 observations. Their features are less but similar to the one's before:

- Name: name of the restaurant.
- Year: year in which the star is assigned.
- Latitude: latitude of the restaurant.
- Longitude: longitude of the restaurant.
- City: city of the restaurant.
- Region: region of the restaurant.
- Zipcode
- Cuisine: type of cuisine they prepare.
- Price: a quotation from $ to $$$$$ relative to the price range of the restaurant.

- Url: link to the Michelin Guide's site with the restaurant's page.

The column *zipcode* and *url* can be dropped from these three datasets, and then we compute the union.
To merge also the first dataset, we need to do an intense cleaning of this one.
First of all, we renamed the common column with the same name (*name*, *latitude* and *longitude).* In order to obtain the *region* column, we can extract the information from splitting the *Michelin guide* one's. The same operation could be done for the *city* column.
The first dataset represent all the Michelin stars assigned in 2021, so we could create the *year* column.
All the other features not presented in the other dataset could be dropped out.
We now need to work on the *price*. In *ds1* it is expressed as a range, so we substitute it with the mean. To maintain the same balance, we drop out all the not euro value. This could be done without too much struggle because we are working with a lot of value.
In order to make a regression on the price, we need to also change the column of the second dataset, because it is expressed in terms on $. This estimate could be done searching the interception between the two dataframe. Seeing the correlation between $ and the relative price in the *ds1*, we substitute an estimation of these values in the second dataset.
We are finally ready to merge all the values, changing only the name of *Michelin stars* to avoid problems with the space character, and delating the NA values.

We are here arrived to a reduced set of features:

- Name
- Year
- Latitude
- Longitude
- City
- Region
- Cuisine
- Price
- Stars

The column *city* and *region* are categorical, but they can be left apart in the analysis because are already expressed through *latitude* and *longitude*.
The categorical variable *cuisine* instead can be divide using dummy variables. It contains a very huge number of different categories, so we summarise into the relative continental type of cuisine and few others.

This dataset has the following shape:

- Name
- Year
- Latitude
- Longitude
- City
- Region
- Cuisine
- Price
- Stars
- Cuisine American
- Cuisine Asian
- Cuisine Australian
- Cuisine Country
- Cuisine European
- Cuisine Innovative
- Cuisine International
- Cuisine Meats
- Cuisine Regional
- Cuisine Seafood
- Cuisine Seasonal
- Cuisine Traditional
- Cuisine Vegan

All this work done on the *price* column surely has a bad impact on the analysis's results. This fact, combined with the really few features of *MG*, will take us to model that could not success in fitting perfectly the data.

3. Exploratory Analysis

3.1 Introduction

Drawing a map of the analysed restaurant around the world, we could notice that most of them are locate in Europe.

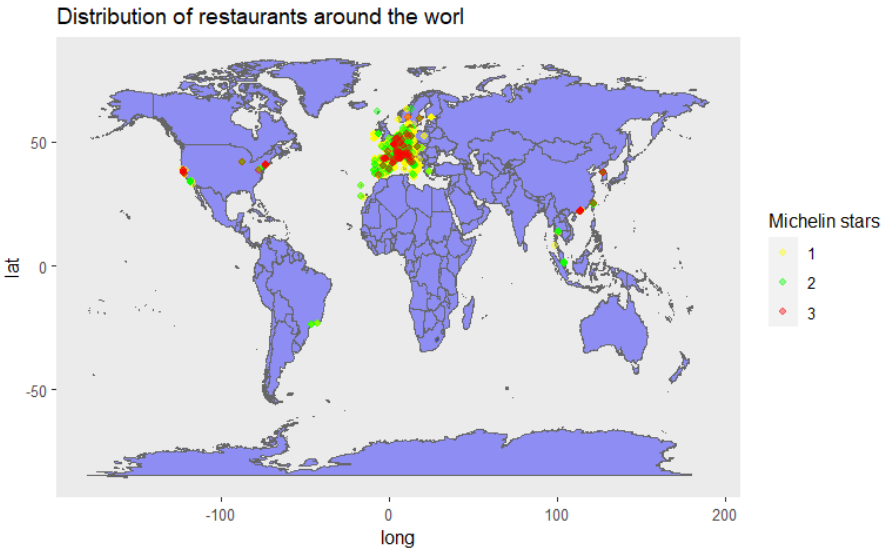

Figure 1

This is showed also throw the boxplot of *latitude* and *longitude.*

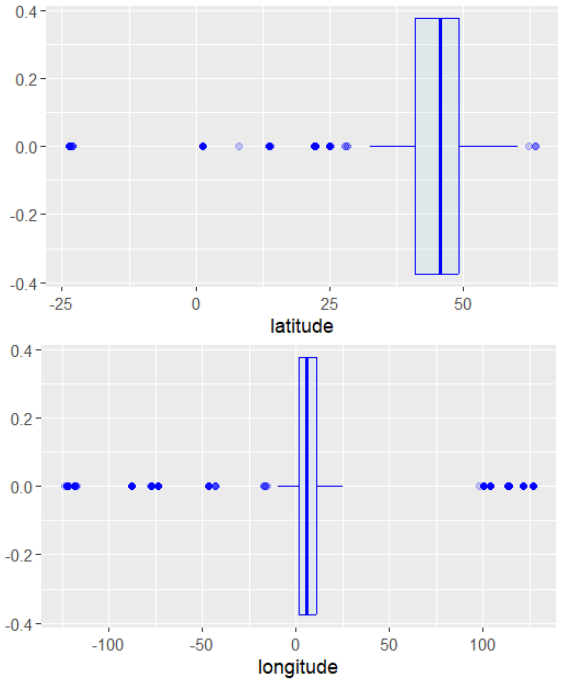

Figure 2

The point representing restaurants in America are in fact identified as some outliers. To establish also if different continentals could influence the medium price, we will study both the case with and without this points, knowing already that the second analysis will probably behaves better.

The prices are distributed in a range of 60 to more than 400 euros.

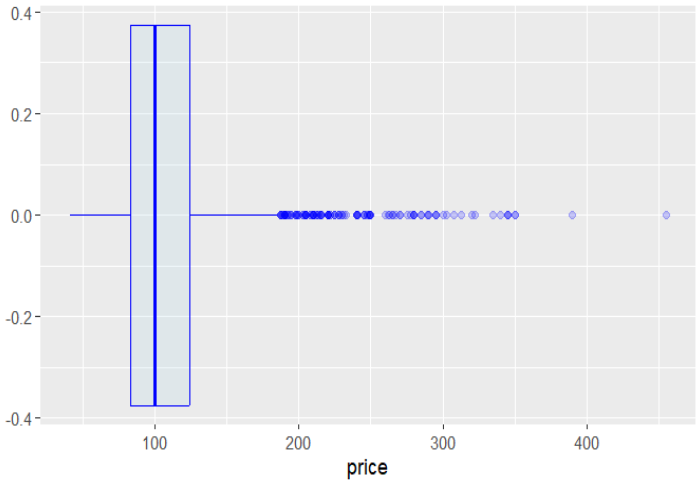

Figure 3

Also here, the higher prices, outside the range of 60 to 160 euros, could be seen as outliers, but they are realistic and their presence could be relevant in order to their correlation with the number of Michelin stars and locations, so they won't be removed.

The distribution of the number of stars and years are the following:

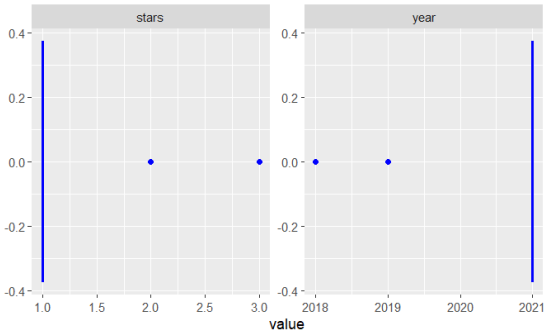

Figure 4

All the first dataset is relative to the 2021, so is the most frequent value. The total for the restaurants with two and three stars is

respectively 318 and 96, really low respect to the 1946 of the one-star places.

The features' distributions could also be seen using histograms.

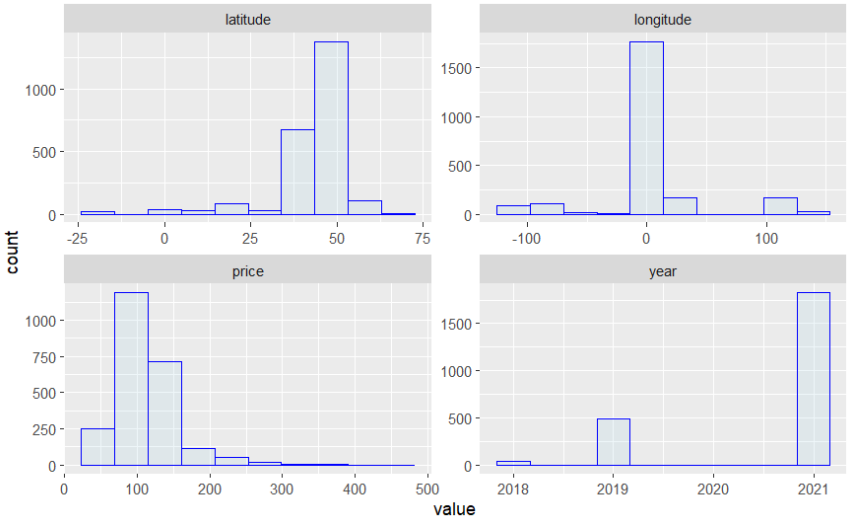

Figure 5

The categorical variables *city, country* and *cuisine* can be plotted through barplots.

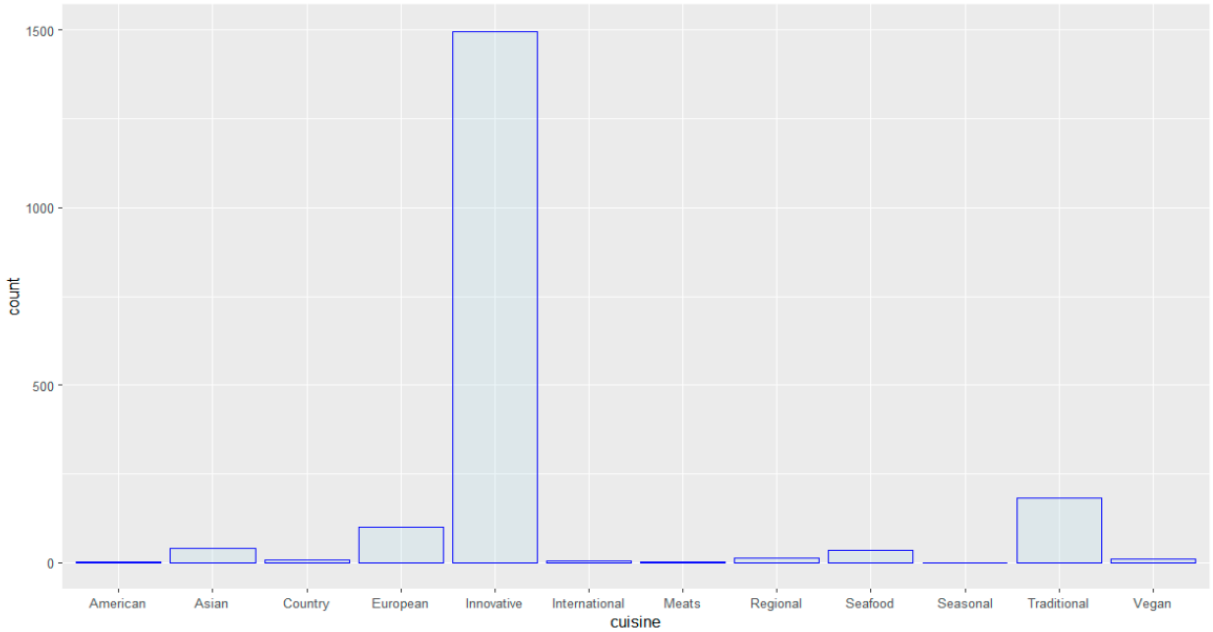

Figure 6

Some interest observations are present in *Figure 7* on the best type of cuisine, city and region that were awarded by the Guide.

Figure 7

To see the correlation of the numeric features with the output variable *price*, is possible to do the plots in *Figure 8*. We could observe that in *price~stars* representation, some visible values outside the common value are present.

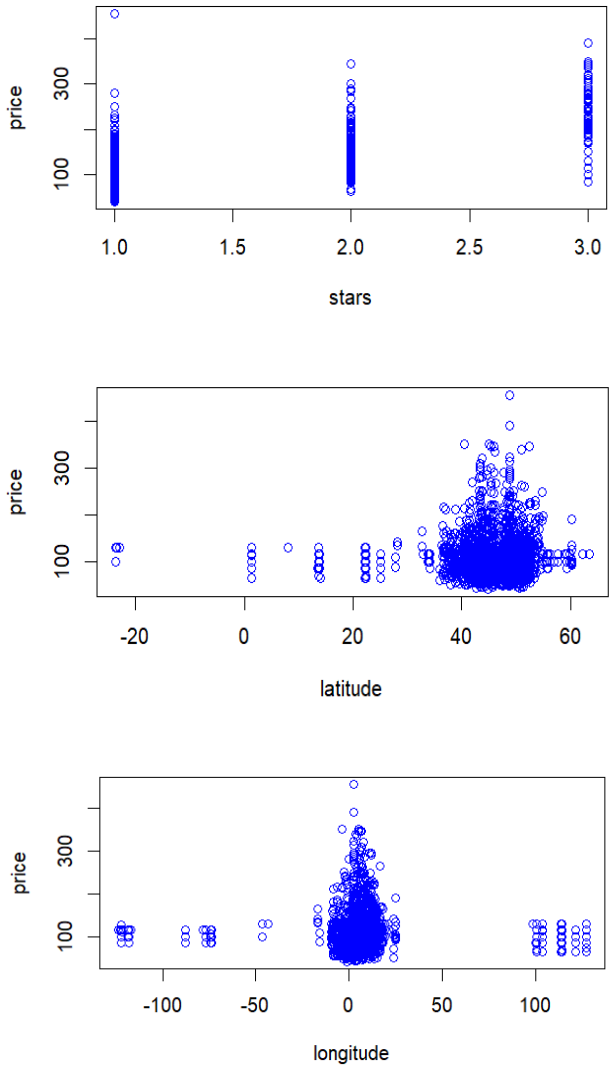

Figure 8

3.2 Outliers

In a general study, is appropriate to drop the outliers from the model before starting with all the analysis. In this case, our variables have a strongly real significance, and their value are not some measures that can be wrong. For example, the outliers of the *longitude* are all the restaurant out of the range of Europe. This allows us to study two different situations: Europe and world.
As we said before, in *price~stars* plot there is a value that is significantly out of the range, and other three slightly upper their mean.

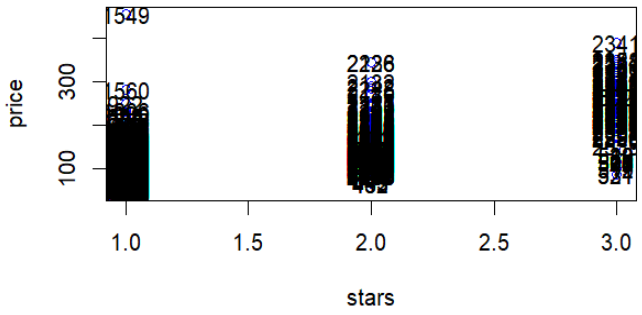

Figure 9

Giving that we have a high number of rows and foreseeing that our set will not behave optimally, we decide to drop all these values.
Let's now modify it to also have the dataset without the values for *longitude* and *latitude* that we will call outliers. The features *start* is left unchanged because, as it is possible to see making this changed, is very influent in the changes in price.
Doing this, 462 values are lost.
The barplots and histograms for our new model are in *Figure 10* and *11*.

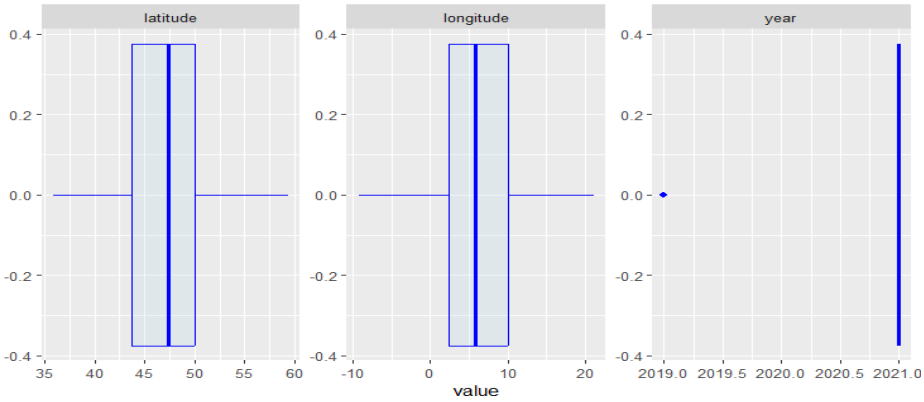

Figure 10

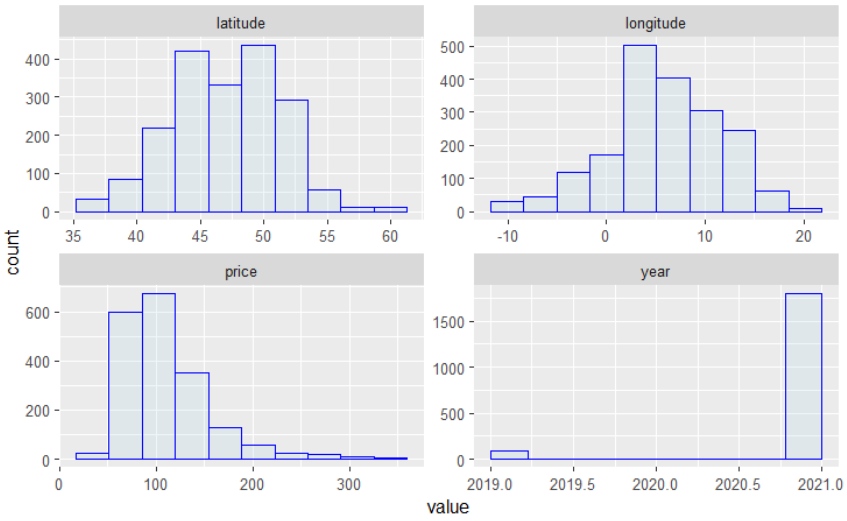

Figure 11

The new plots for the relation with *price* are:

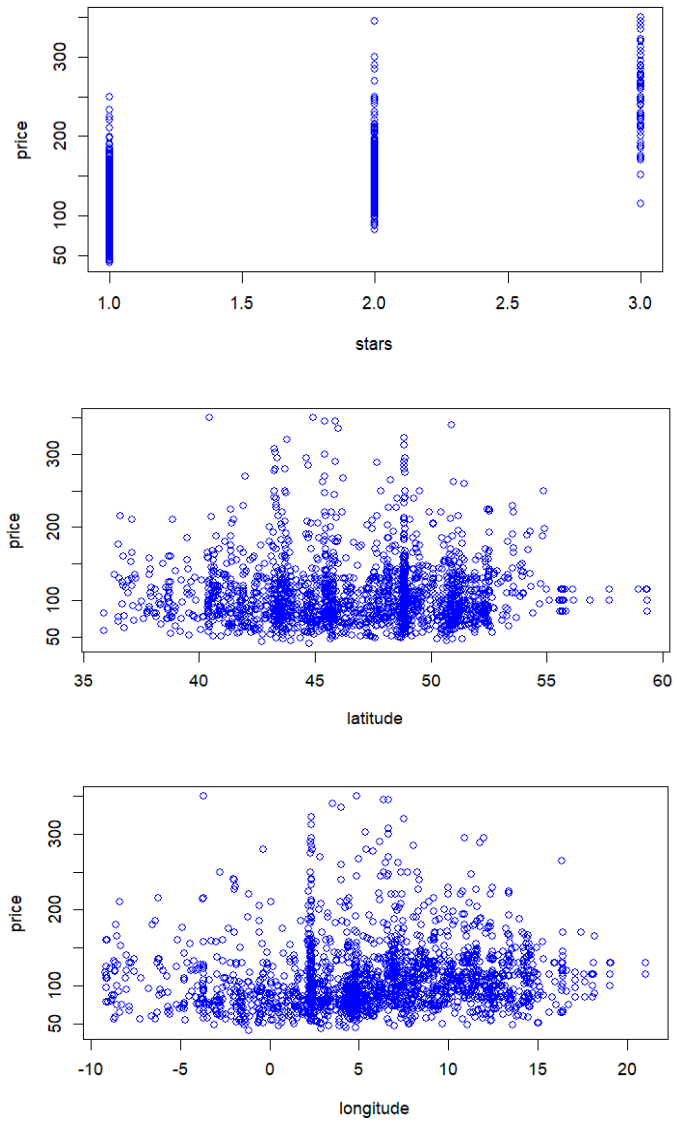

Figure 12

The geographical distribution of this second set of values is in *Figure 13.*

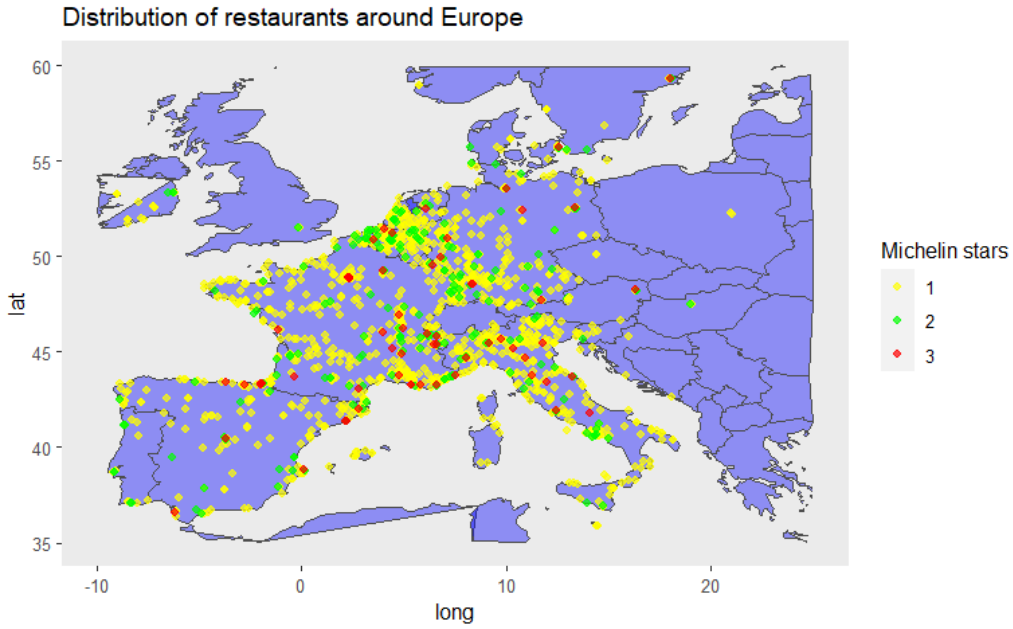

Figure 13

In general, we will expect better results from this model because, in terms of number, all the features geographically far from the Europe can be consider outliers for the main numerical feature *longitude*.

3.3 Correlation

The correlation between the features can be seen through the Correlation matrix.

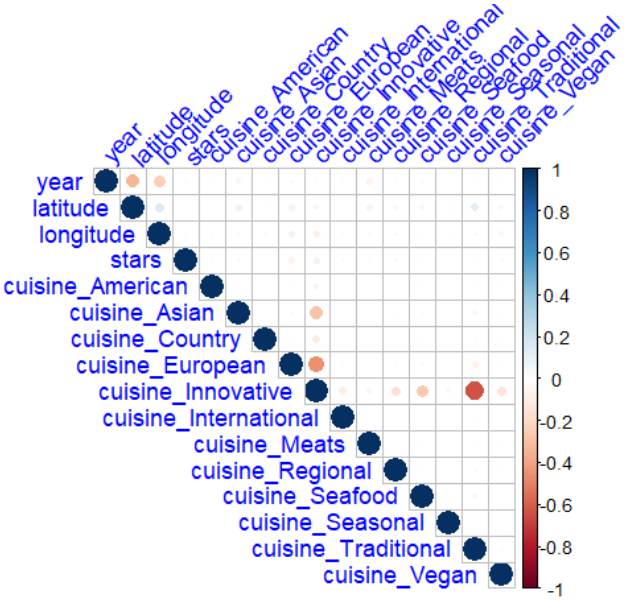

Figure 14

As this plot shows, there no significant correlation between the variables, if not for *cuisine Innovative* and *cuisine Traditional*.
This variable will be excluded when we will run method that suffer from collinearities.

3.4 Subset selection

To see the relevance of all the variables in the model, is possible to study the best subset selection. Applying the '*foreward'* method, we could see that *stars* is the first one chosen, as we can expect. It is followed by *year*, *longitude*, *cuisine Regional* and *cuisine European*. The *latitude* is not considered because his values are really close one to each other.
Based on the *Cp* index, the optimal number of features of the model is four.

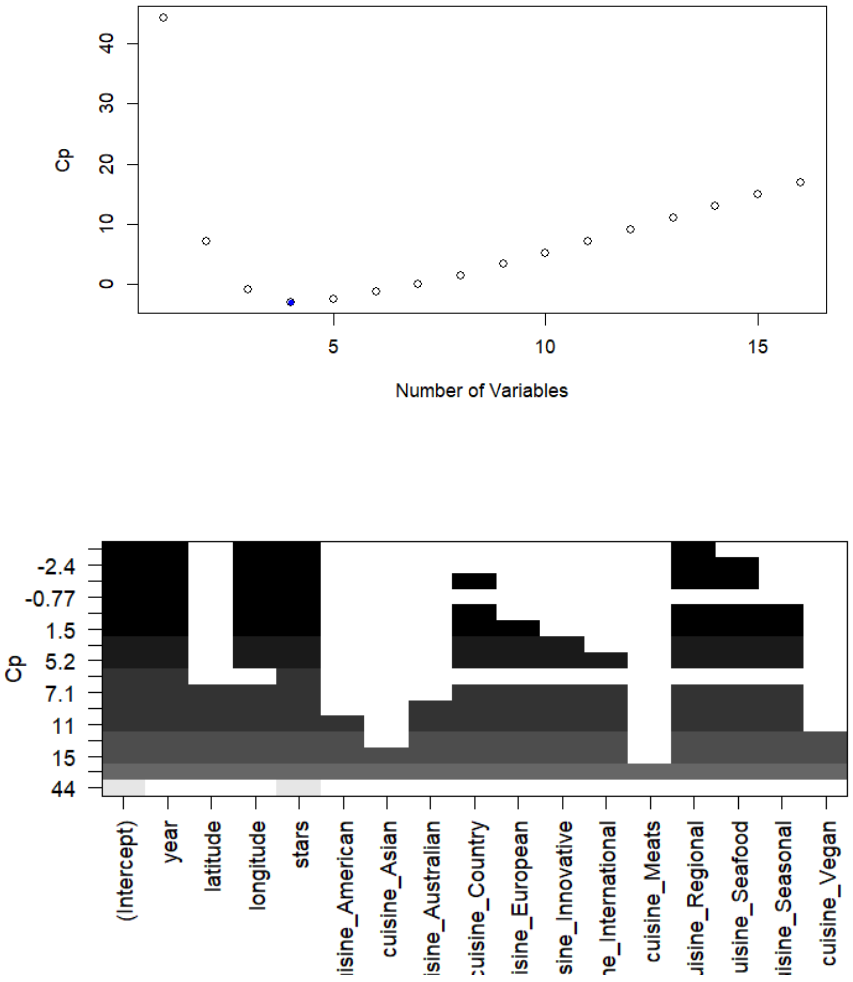

Figure 15

4. Supervised learning

To analyse the model and predict the *price* column, we choose three supervised learning techniques: linear regression, robust linear regression and random forest.
The dataset that we are studying present really few numerical columns, and two of these are *year* and *stars*, taking just two values. The *cuisine* feature is divided into 12 dummies, that we will see, are not particularly significant.
The output feature *price* has been modified with the mean and the estimation of the symbol '*$*' through relative few examples belonging in the intersection of the two sets.
All these premises make us believe that the model probably will not be able to fit the data perfectly. Conscious of that, we will search the ones that behaves better.

4.1 Linear Regression

Starting from the most basic model, we analyse the application of Linear regression model.
In general, an index used to measure the goodness of the fitting of the model is R^2, a statistical measure that represents the proportion of the variance for a dependent variable that is explained by an independent variable in a regression model.
In the linear regression made in Europe, this index is 0.5, which means that the percentage of explained variance is approximately 50%.
As we seen in the subset selection, also there the most significant features are still *longitude* and *stars*, with a little influence of the dummy *cuisine Asian.*

Running the same code for the dataset with still the outliers, the R^2 is 0.38. Its most significant features are *year, longitude, stars* and, for a little, *cuisine Regional.*

As expected, the values of American restaurants behaves as distortion for the *longitude*, and this is suffered from the linear regression, that works worst in this situation.
All the main variables are those most important for the dataset, and the two dummies are two of the most numerous.

But why in general these results are not so good? Linear regression is an analysis that assesses whether one or more predictor variables explain the dependent (criterion) variable. Some assumptions and

good criteria that the features must satisfy to have a good linear fitting are:

- **Linear relationship**: as it is possible to see both in *Figure 8* and *Figure 12*, the distributions of the features are never linear.

- **No multicollinearity**: this is satisfied from the Correlation matrix in *Figure 14.* Also checking the *vif* we obtain that our features are not multicollinear.

- **Multivariate normality**: The dependent value *price* does not present a normal distribution, as it is possible to see in *Figure 5, Figure 11* and *16*, but to be sure we could go in more details.

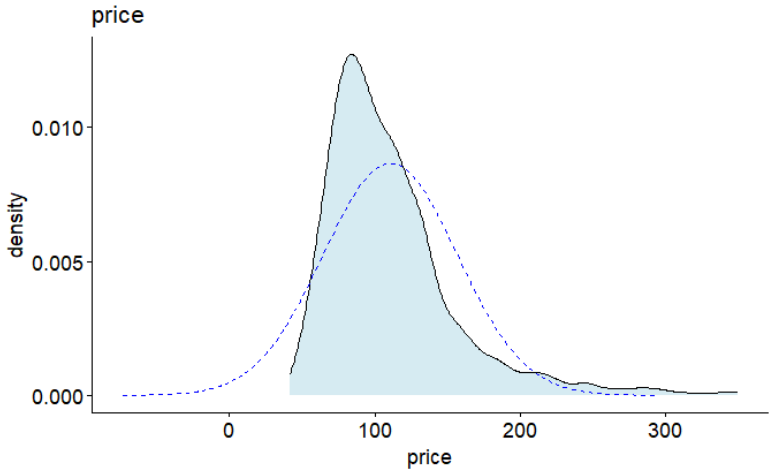

Figure 16

Using the *qqPlot* function like in *Figure 15*, we observe how most of the values fall out of the blue range.

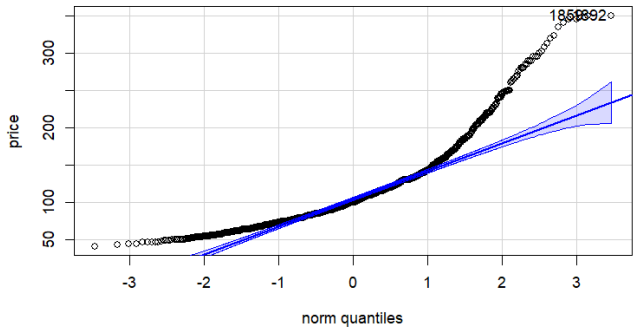

Figure 17

With a non-graphical approach, the Shapiro Test can be evaluated, given a very low *pvalue*<2.2e-16, which make us to reject the null hypothesis of normality.

Also, no one of our independent variables are normally distributed. This could also be showed throw the matrix in *Figure 18*.

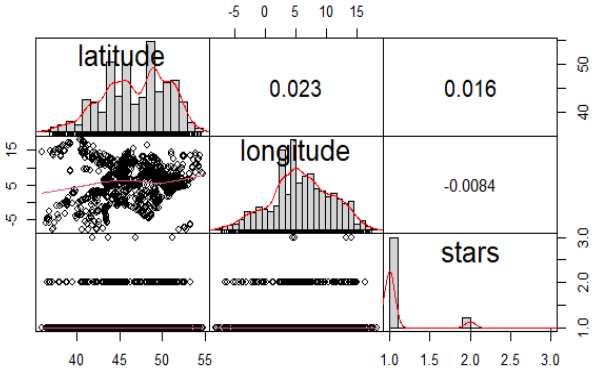

Figure 18

- **Normal distribution of residuals**: This assumption could be checked in different ways. Plotting the histogram of the residuals, as in *Figure 17* their non-normality is immediately clear.

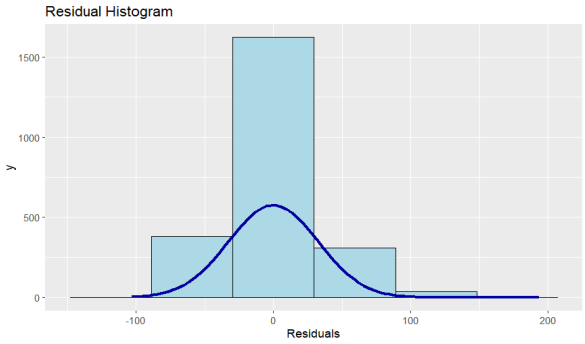

Figure 17

Another way is look at *residual~fitted values* plot. For the normality assumption to hold, the residuals should spread randomly around zero and form a horizontal band. If the red trend line is approximately flat and close to zero, then one can assume that the residuals are normally distributed. In *Figure* 18 this doesn't happens. It is so possible state that our residuals from linear regression are not normal.

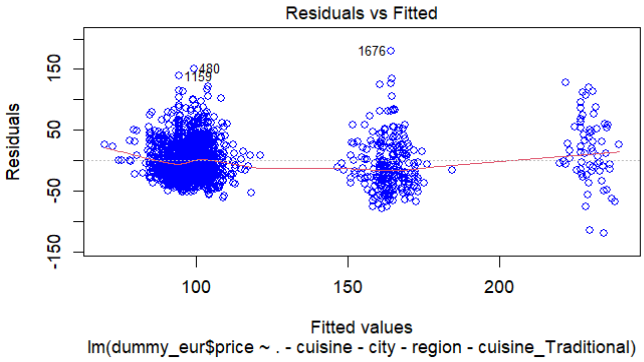

Figure 18

- **Homoscedasticity**: The Breusch-Pagan test is used to test the null hypothesis of homoscedasticity. It regresses the residuals on the fitted values or predictors and checks whether they can explain any of the residual variance. A small p-value, then, indicates that residual variance is non-constant (heteroscedastic). On our data, we obtain a *pvalue* smaller than 0.05, therefore we can reject the null hypothesis that the variance of the residuals is constant and infer that heteroscedasticity is indeed present.

After all these observations, we have now justified the so wrong linear fitting to our data.

4.2 Robust regression

The ordinary least squares estimated for linear regression are optimal when all of the regression assumptions are valid. When some of these assumptions are invalid, least squares regression can perform poorly. Robust regression methods provide an alternative to least squares regression by requiring less restrictive assumptions. These methods attempt to dampen the influence of outlying cases in order to provide a better fit to the majority of the data.

Applying this method, we obtain a R^2 equal to 0.50 for the European model, so nothing changed between the linear model. But there are two main advantages now: all the coefficients are significative, and the residual standard error improved from 32 to 28.
The same conclusions are obtained for the world dataset.

4.3 Random Forest

Random Forest is a supervised machine learning algorithm made up of decision trees. From the theory, we know that the tree methods, and this one in particular, are the those who work better. Especially, they improve good results also in datasets with outliers.
This method is able to treat also with categorical variables, so *city*, *region* and *cuisine* can be considered in our model. Having three columns more, native of our downloaded datasets, is another reason why this model will bring us to the best results.

For the data where the outliers have been removed, the random forest method has a good fitting (for our standard). In fact, the R^2 on the training set is 0.55 and on test set is 0.53. To estimate the goodness of the prediction, we could consider the root mean squared error between the observation of the test set and the one predicted by the method. In our case, this has a value of 32, that is pretty high. In *Figure 19* is possible observe this relation, noting the very spread points, not so close to the straight line.

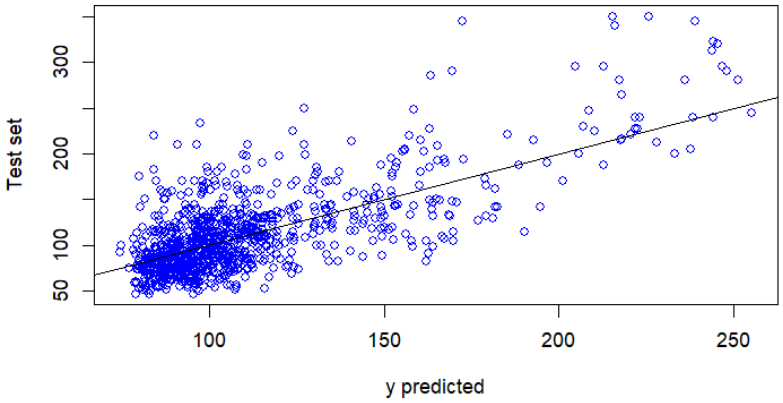

Figure 19

The optimal tree model follows the structure in *Figure 20* to predict the data.

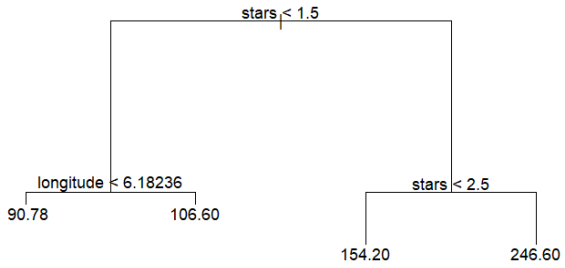

Figure 20

The importance of the variables used to build the tree is:

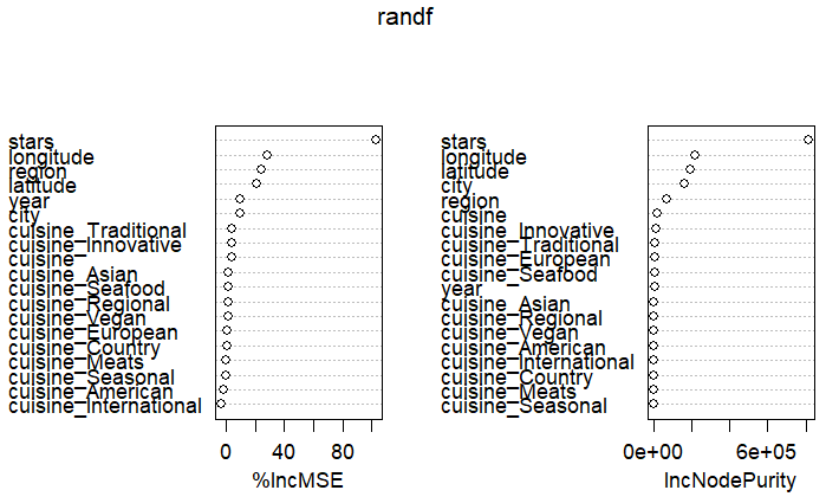

Figure 21

In *Figure 21* we could read that the variable *stars,* if removed, would increase the means square error by more than 90%. Is in fact the first one that splits the tree on the root.

Pruning trees in a random forest is not recommended because it can negatively impact the overall performance of the model. The idea behind a random forest is that it is a collection of decision trees, each of which is trained on a different subset of the data. By pruning trees, you are removing some of the diversity of the model and potentially reducing its accuracy. Additionally, random forests use a technique called "bagging" to reduce overfitting, so pruning trees would also remove this benefit. It is generally better to let the algorithm determine the optimal number and size of the trees.

Let's now consider the dataset where the outliers have not been removed. Repeating the same analysis, the results are optimally improved compared to the linear regression. The R^2, in fact, is now 0.54 for the training and 0.52 for the test set. The root mean squared error is now 28. In *Figure 13* there is the distribution of the real and predicted value.

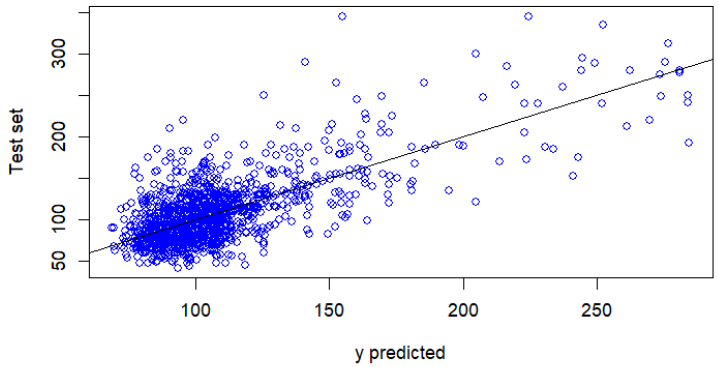

Figure 22

Now more points could be predicted by the model, as we could see by the leaf of the tree in Figure 14 (best predictor tree).

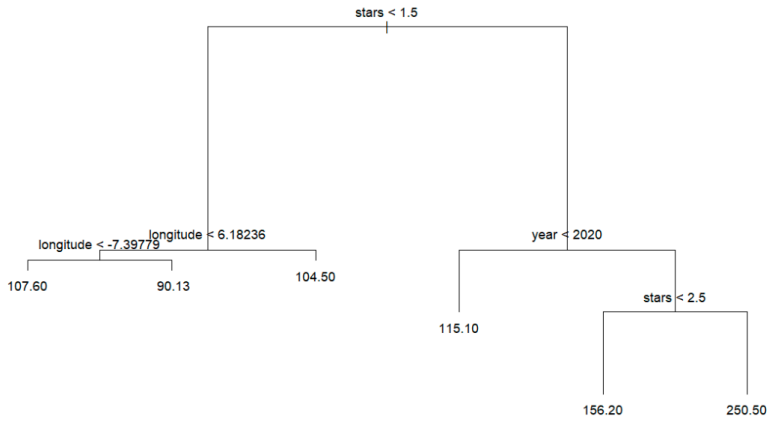

Figure 23

The importance of the variables used to build the tree is:

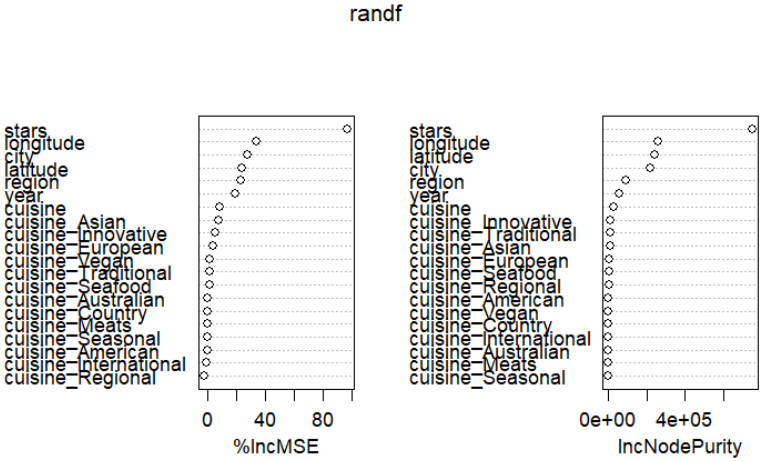

Figure 24

From F*igure 23* and *24* we could notice that the main features used to build the tree are also the one's most significant from our initial subset selection.

5. Unsupervised learning

5.1 Hierarchical Clustering

Clustering is an unsupervised statistical learning technique. Often, clustering involves sorting observations into groups without any prior idea about what the groups are. These groups are delineated so that members of a group should be more similar to each other, more than respect to other groups. Throughout data science, and particularly in geographic data science, clustering is widely used to provide insights on the (geographic) structure of complex multivariate (spatial) data.

Having there the geographic coordinates, we are interested in created the geographical clusters for prices ranges.

First of all, we need to convert the *latitude* and *longitude* to spatial objects and adding these coordinates to the training set. We will not add the predicted variable *price* in our model because is an unsupervised method. Then we evaluate the distances matrix and apply the hierarchical clustering method, obtaining the dendrogram in *Figure 25.*

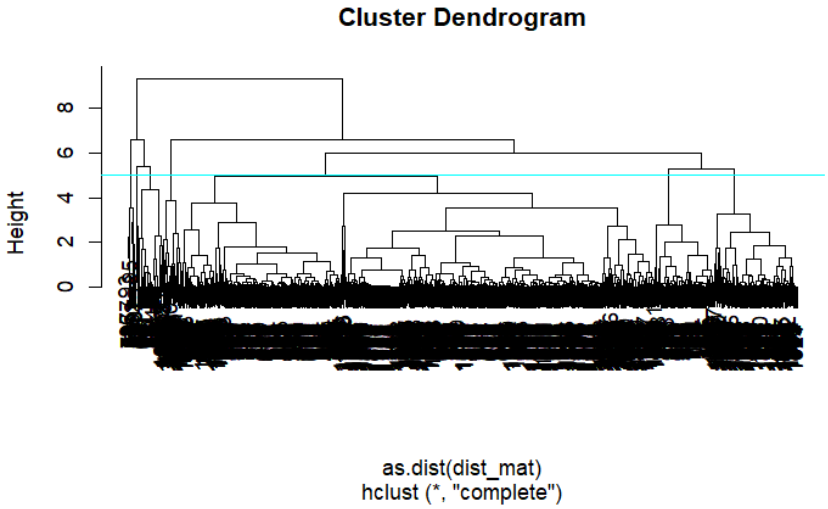

Figure 25

Cutting the tree where we find seven cluster (cyan line in *Figure 25*), we obtain the final groups with the divided point based on all their characteristics. Plotting these points into the world map we find the following clusters.

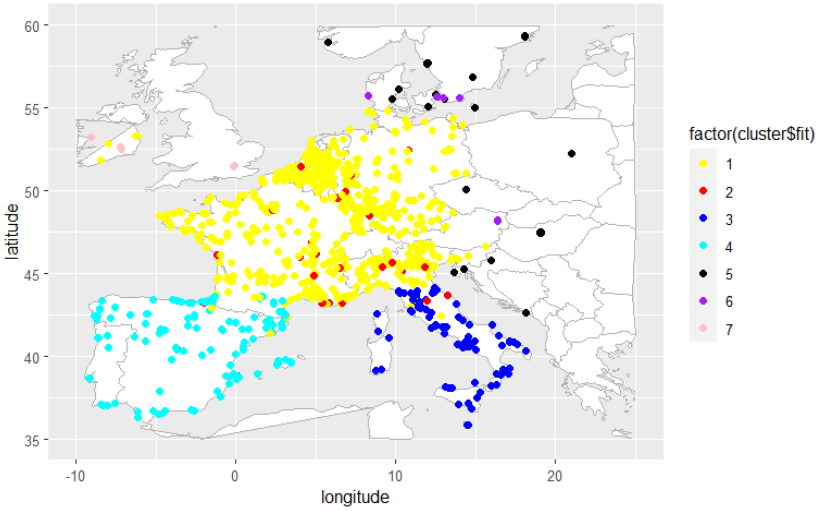

Figure 26

The cluster are distributed as follows:

cluster	*n*
1	892
2	37
3	112
4	156
5	44
6	15
7	6

This was the evaluation of the method on the training set. In order to test how it works, let's try on the simple linear model. The R^2 has increase his value, from 0.50 for the supervised method to 0.54 now. This could be possible because this model works also with categorical variables, so our three features *cuisine, city* and *region* could be included. Two of the seven cluster are also significant for the analysis.

6. Conclusion

As it was said at the beginning, the results of the analysis are good but not completely perfect.

The models couldn't be able to fit the data with such few numeric features. Also, the predicted column was constructed by approximation of the mean and the symbols *'$'*, so partial synthetic.

The methods that finally work better are Random Forest for supervised learning and the unsupervised hierarchical clustering, according to the comparison of all the R^2 indexes.

This result is in according to what we predict at the beginning, because both methods are able to work with categorical features, and in our case is fundamental because are three of the few columns that we have.