Assignment 2

Sára Vargha & Benjámin Salga

Introduction

The aim of this analysis to uncover the probability that a hotel in Barcelona is getting a high rating for various explanatory variables, such as the stars of the hotel and its distance from the city centre. For the sake of the analysis, the hotels-europe dateset was used that includes information on hotels in 46 European cities.

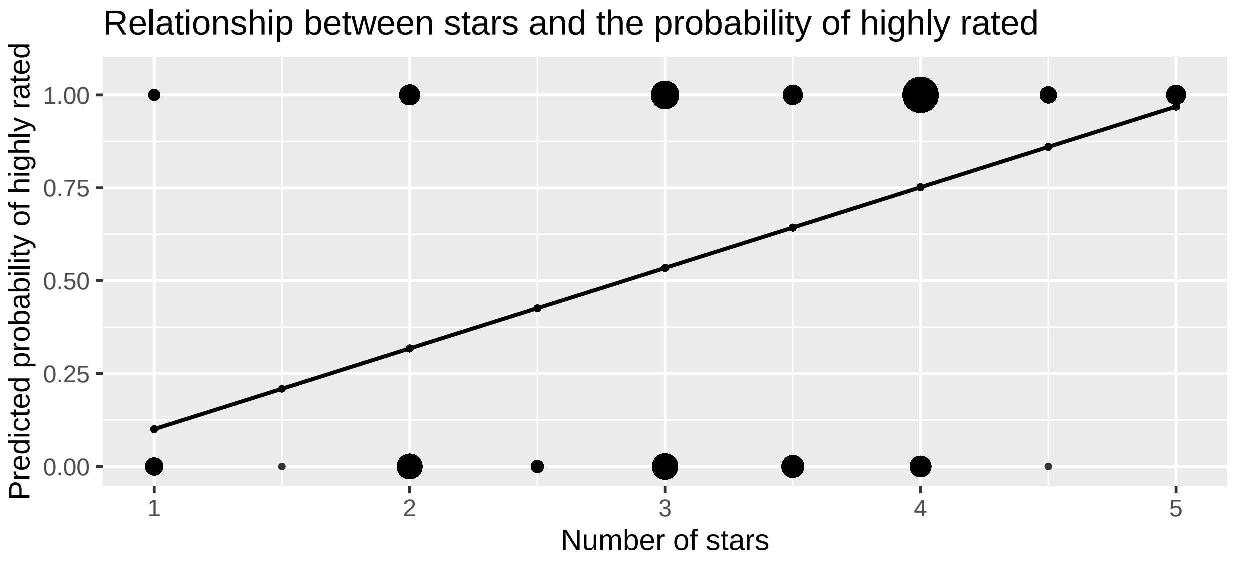
Data cleaning

After the data has been loaded from osf.io, we first checked the summary of the variables (e.g. ratings, Trip Advisor ratings, number of ratings, starts, distance) we intended to use in our analysis. As the first step, we substituted missing values of rating, rating_reviewcount, ratingta, ratingta_count and stars with the average values, accordingly. Second, we filtered the dataset for Barcelona as the actual city and created a binary variable for highly rated hotels (highly_rated=1 above for >= 4 stars, highly_rated=0 otherwise). Finally, we applied a filter to set a minimum number of reviews and dropped observations with less than 20 reviews. Looking at the summary of our highly_rated variable, the median value is 0.594, meaning that 59.4% of hotels got at least a rating of 4 in our data.

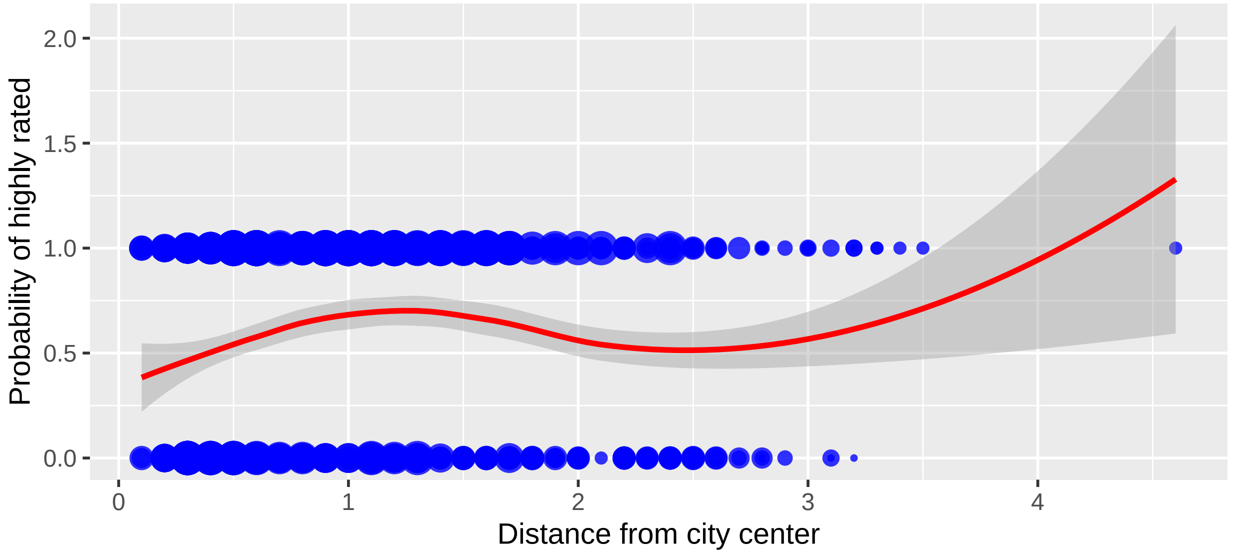
Analysis

In the first step of our analysis, we examined the relationship between the number of stars of a hotel and the probability of getting a higher rating (lpm1).Based on our LPM model, we can conclude that hotels with one additional star are 21.7% points more likely to get highly rated. average. Next, we investigated the relationship between the number of reviews (both on the aggregrator website from where the data was originally scraped and on TripAdvisor) and the likelihood of getting a higher rating. First, we started with the number of ratings on the website (lpm2). Here we decided to add an extra binary variable to split the observations into two groups based on the number of ratings (rating_count_group=1 for observations with >= 180 reviews, rating_count=0 for observations with less than 180 reviews). Based on the outcome of our LPM model, hotels from the group with at least 180 reviews are 31.4 percentage points more likely to get at least a rating of 4. As we also had the count of Trip Advisor reviews in our data, we repeated the same steps with reviewta_count, splitting the hotels in two groups, with at least 480 reviews and less than 480 reviews (we chose a higher limit because hotels seemed to have significantly higher number of reviews on TripAdvisor). Similarly to the results of our second LPM model (lpm2), we can conclude that hotels belonging to the group of at least 480 TA reviews are 34.9 percentage points more likely to get highly rated (lpm4). In the next step, we enriched our LPM model with linear splines for distance, and accommodation types as factors on top of stars and rating_count_group (lpm4). From this model, only stars and rating_count_group lead to significant results. Ceteris paribus, hotels with one additional star are 20.5% points more likely to get highly rated, wheres ceteris paribus hotels with at least 180 reviews on the website are 17.9% points more likely to get highly rated. In the final part of our analysis we used the same set of explanatory variables as in case of lpm 4 to run logit and probit probability models. THe summary of the outcomes of these models (including the marginal differences of our logit and probit models) were stored in Table 1. From the results, we can indeed see that LPM, logit and probit lead to very similar results, with stars and rating_count_group being the only significant explanatory variables at 1% and LPM having the smallest standard errors. Looking at our histogram where we plotted one histogram for observations with actual y=1 (highly rated) and one for observations with actual y=0 (not highly rated) based on our rich LPM model (lpm4), we can see that the fit of the prediction is far from perfect. Although the two distributions overlap to some extent, a larger part of the distribution covers higher predicted values among hotels that got highly rated, as it should.

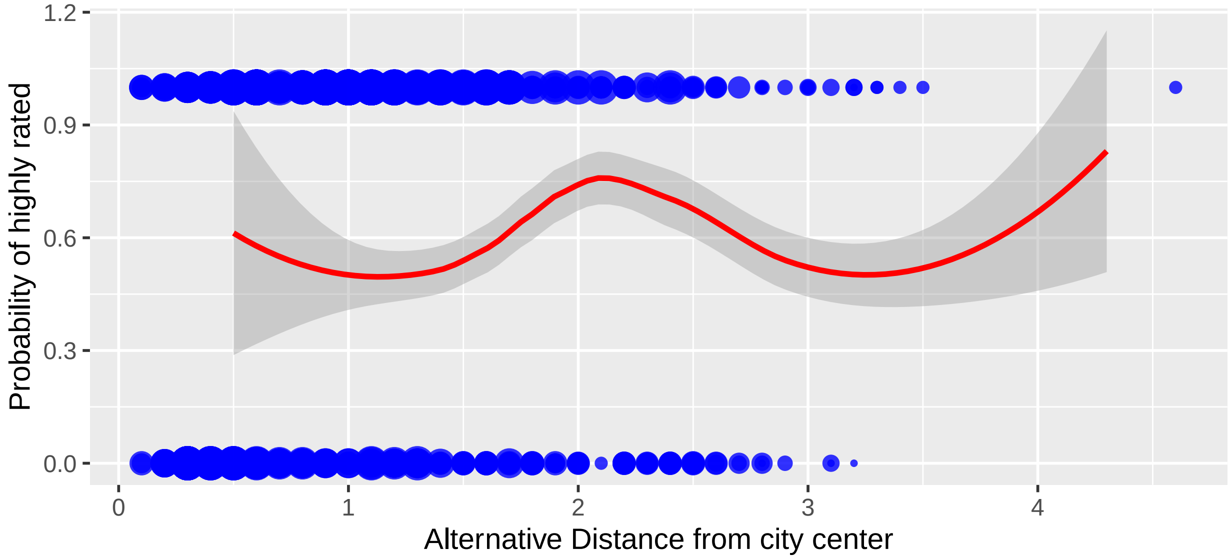
Charts

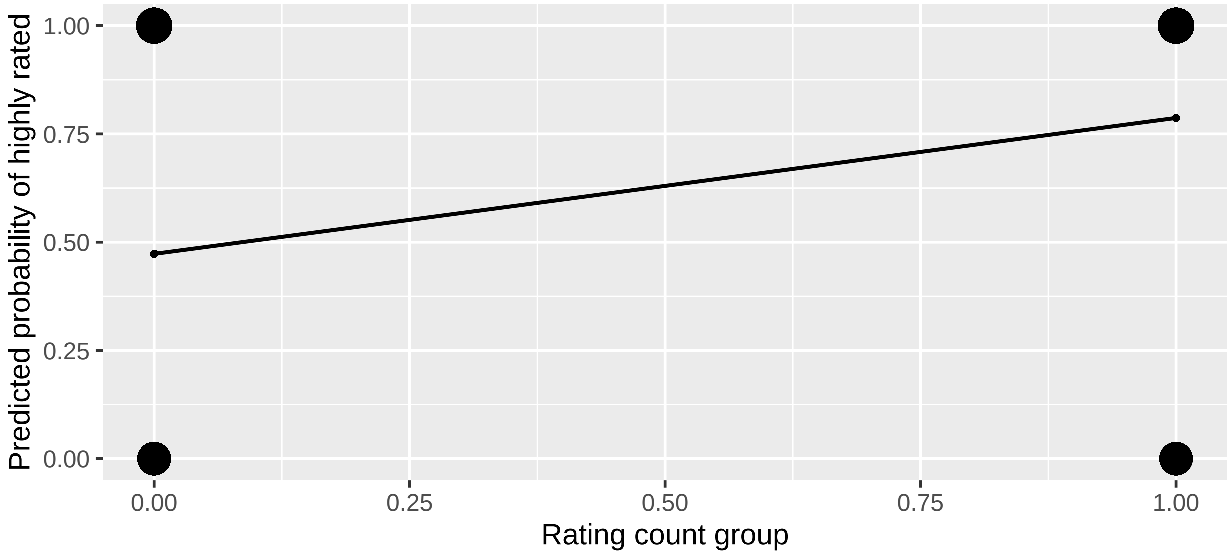


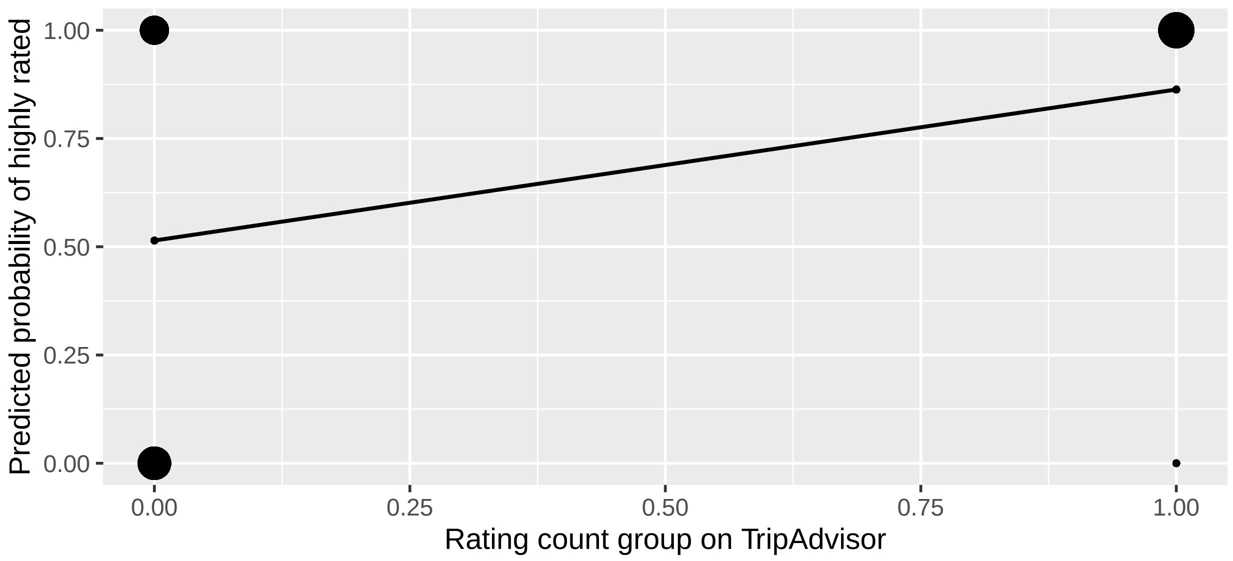
```
## `geom_smooth()` using formula 'y ~ x'
```

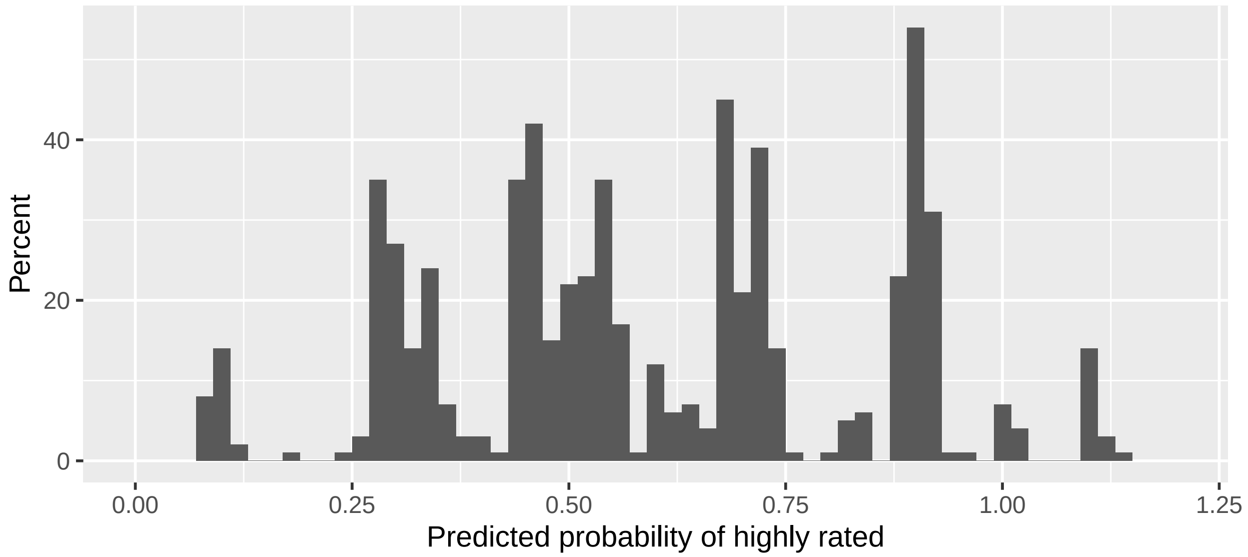


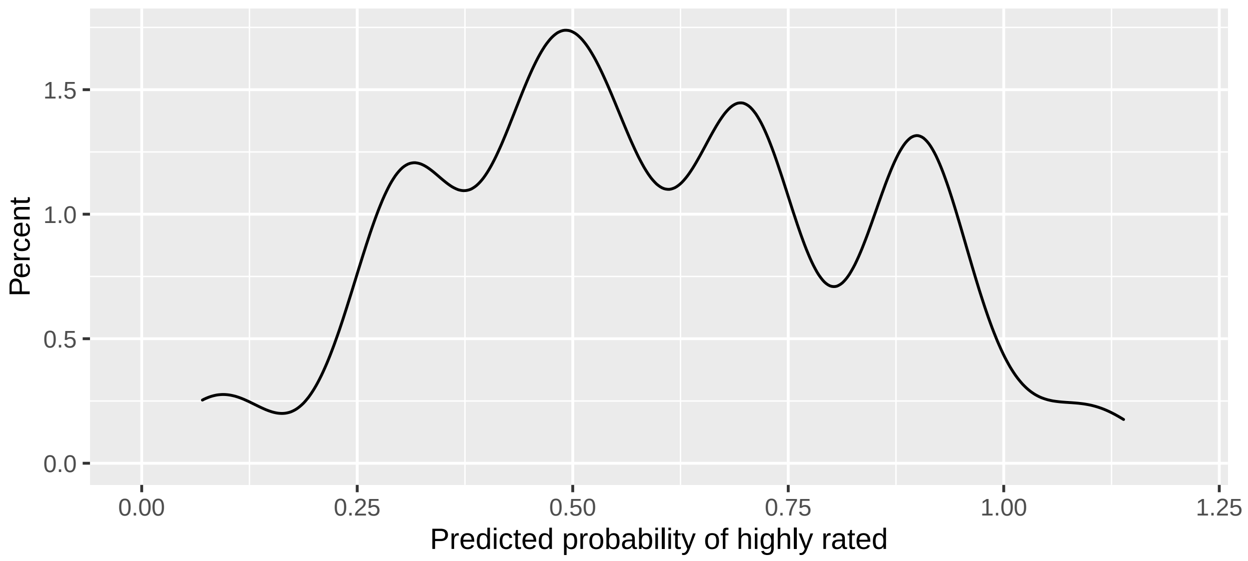
```
## `geom_smooth()` using formula 'y ~ x'
```

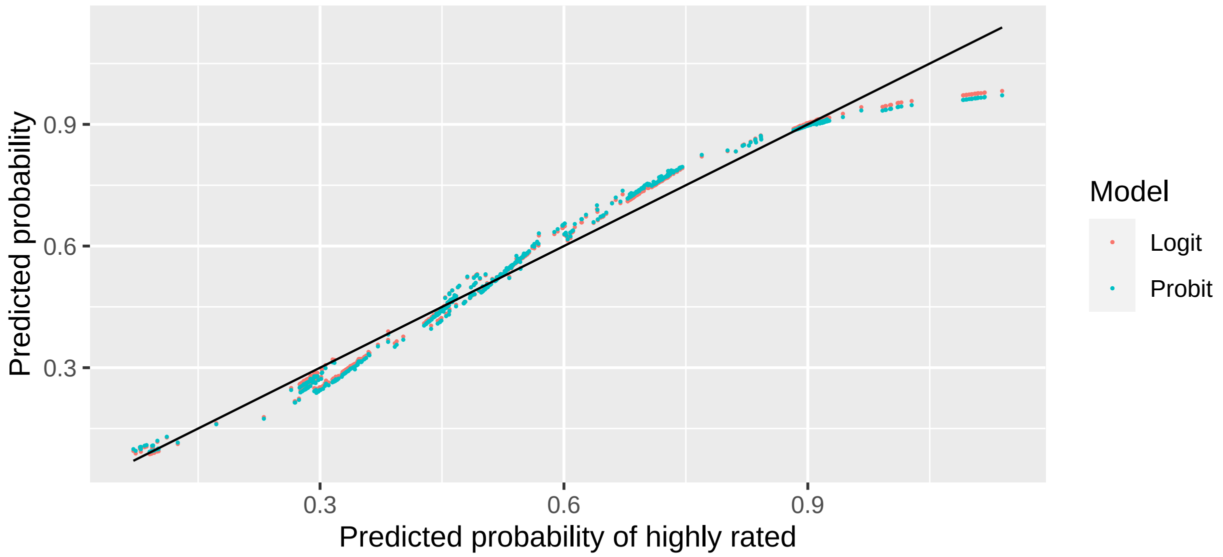

Probability of highly rated
1.2
0.9
0.6
0.3
0.0
0
1
2
3
4
Alternative Distance from city center

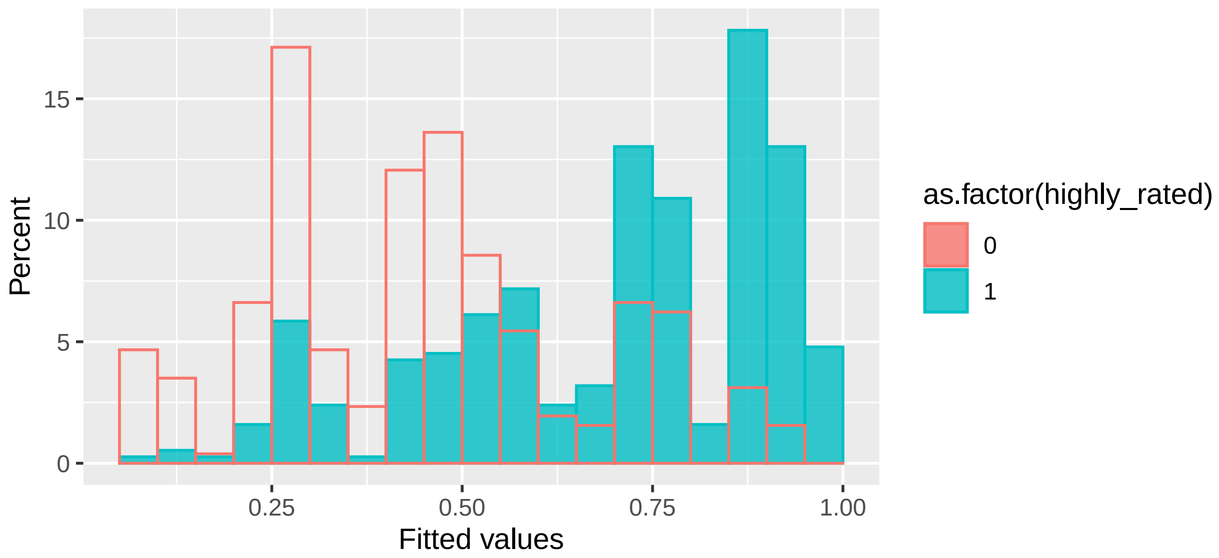

Predicted probability of highly rated
1.00
0.75
0.50
0.25
0.00
0.00
0.25
0.50
0.75
1.00
Rating count group


Predicted probability of highly rated
1.00
0.75
0.50
0.25
0.00
0.00
0.25
0.50
0.75
1.00
Rating count group on TripAdvisor


Percent
40
20
0
0.00
0.25
0.50
0.75
1.00
1.25
Predicted probability of highly rated


Percent
1.5
1.0
0.5
0.0
0.00
0.25
0.50
0.75
1.00
1.25
Predicted probability of highly rated


Predicted probability
0.9
0.6
0.3
0.3
0.6
0.9
Predicted probability of highly rated
Model
● Logit
● Probit


Percent
15
10
5
0
0.25
0.50
0.75
1.00
Fitted values
as.factor(highly_rated)
0
1