Finding Hotel Deals in Vienna using Machine Learning

Bruno Helmeczy

12/02/2021

Executive Summary

This report looks to find the best "Hotel Deal" in Vienna, during holidays in December 2018. I look to answer "Which 5 hotels offer the best 'Deal' during 2018 year-end holiday period ?" by applying a Linear Regression, Classification & Regression Tree, & a Random Forest model, on data obtained from Trip Advisor. I define the 'best deals' as those observations with largest negative residuals vs their expected price according to my selected model.

Introduction

Vienna is a popular destination during the (pre)-christmas period, famous for its christmas market, & beautiful scenery, glamorous reminders of a glorious age called the Austria-Hungarian Monarchy. As such a conveniently distanced, & suitable destination for couples' & families' from neighbouring countries, to relax. I look to answer "Which 5 hotels offer the best 'Deal' during 2018 year-end holiday period ?" by applying a Linear Regression, Classification & Regression Tree, & a Random Forest model, on data obtained from Trip Advisor. I define 'Best Deal' as the hotel offering the most-value for money. In this context, after modelling hotels' prices based on variables related to hotel quality, distance from the city centre & guest reviews, I define the 'best deals' as those observations with largest negative residuals vs their expected price according to my selected model.

Data Cleaning, Association Patterns, & Variable Transformations

Cleaning: My raw data comprises the hotels europe dataset available by clicking here. This dataset includes hotel prices & various other features from 46 european cities on 10 different dates, & is scraped from a price comparison website. I filtered the dataset for 2018, December, holiay noted period, netting 708 observations. Note that I included both 1-night & 4-night deals, as they reporesent different offerings, possibly with different value for money, even if offered by the same hotel. Furthermore, it is a realistic scenario for traveller to decide on the holidays' length, based on the deals that they would find most to their liking. To avoid comparing apples & oranges however, I calculated the price per night for every deal, which represents my target variable. I cleaned the data 1st by Converting distance strings, & review ratings to numeric vectors; then 2nd by Imputing missing Review Scores for a hotel with the datasets' median & adding flag variable columns to note hotels without reviews. With that, I am ready to observe variables' distributions.

Distributions & Association Patterns:

Vars	min	median	mean	max	sd	skewness
price	30.0	262.0	359.1	1546	300.3	1.4
pricepernight	30.0	128.0	147.4	386	67.8	1.2
Nrnights	1.0	1.0	2.4	4	1.5	0.2
starrating	1.0	3.5	3.4	5	0.7	-0.3
center1distance	0.0	1.5	1.7	13	1.6	3.1
center2distance	0.5	3.5	3.7	13	1.6	1.5
rating2_ta	2.5	4.0	4.0	5	0.4	-0.7

Vars	min	median	mean	max	sd	skewness
rating2_ta_reviewcount	0.0	233.0	471.6	3262	613.5	1.9
guestreviewsrating	1.0	4.0	4.0	5	0.5	-1.3

Modelling

	Rsq	MAE	RMSE	RMSE_norm
OLS	0.520	30.855	43.738	0.297
CART	0.505	24.472	35.336	0.240
Random_forest	0.668	16.295	24.777	0.168

Final Model:

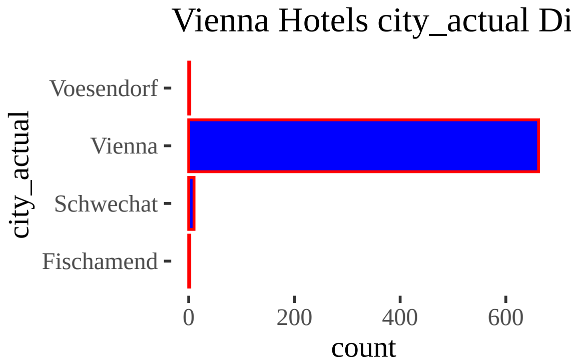
modelname	Rsq	MAE	RMSE	RMSE_norm
RF Final Model	0.683	16.179	24.45	0.166

The 5 Best Deals:

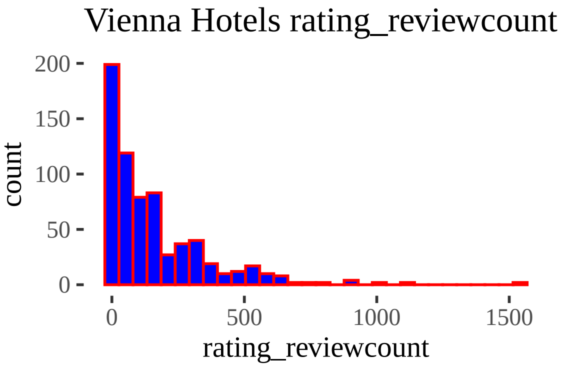
	Hotel_id	Stars	Avg.Price	Resid.	Nights	Type	Where?	Miles fr Center	TA_Rating	Nr.Ratings
1	21905	3.5	70	-54.4	1	Apartment	Alsergrund	1.2	4.0	0
2	21980	4.0	146	-47.3	1	Hotel	Innere Stadt	0.1	4.0	714
3	21982	3.0	123	-46.4	1	Hotel	Innere Stadt	0.5	4.5	985
4	22360	3.0	104	-45.2	4	Apartment	Vienna	2.8	4.0	0
5	21939	3.0	95	-44.6	1	Apartment	Favoriten	2.5	4.0	0
65	22184	3.0	85	-18.6	1	Hotel	Leopoldstadt	0.7	4.0	827
79	21975	4.0	197	-17.2	1	Hotel	Innere Stadt	0.1	4.5	211
84	22344	3.0	65	-16.7	4	Hotel	Vienna	3.9	4.0	12
106	22344	3.0	57	-14.8	1	Hotel	Vienna	3.9	4.0	12
150	21912	4.0	95	-10.7	1	Hotel	Alsergrund	1.1	4.0	359
151	22080	3.0	65	-10.7	1	Hotel	Josefstadt	1.1	3.0	85
588	21975	4.0	264	28.0	4	Hotel	Innere Stadt	0.1	4.5	211

Appendices

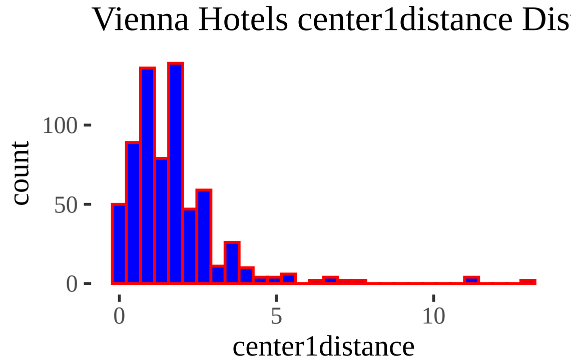
1) Variable Histograms



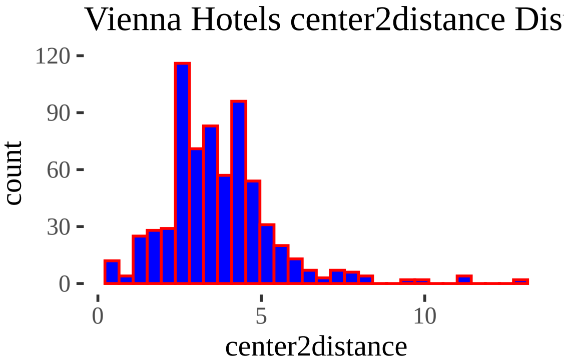
[[1]]



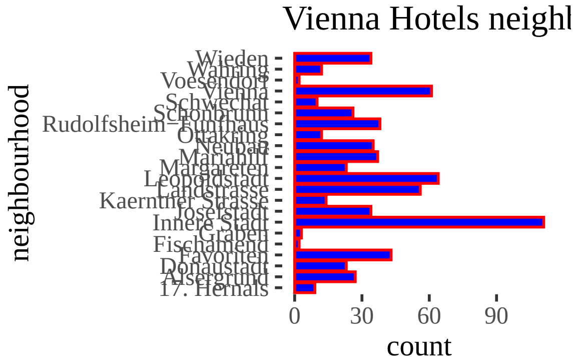
[[2]]



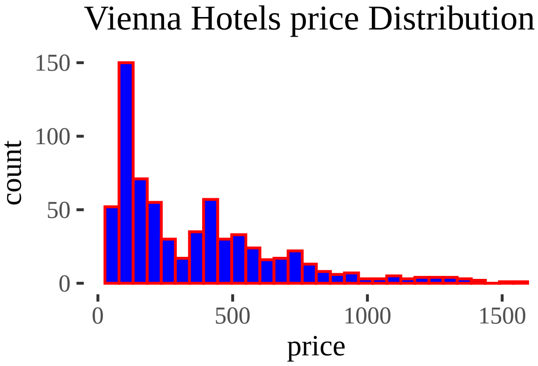
[[3]]



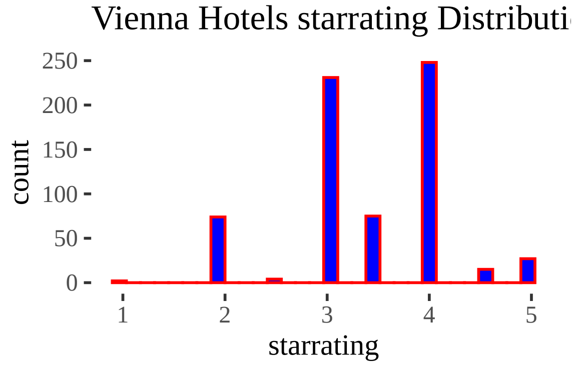
[[4]]



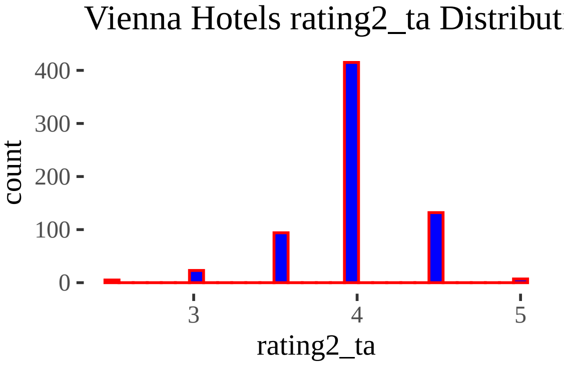
[[5]]



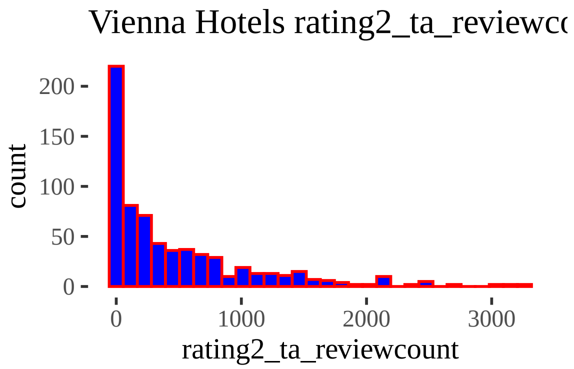
[[6]]



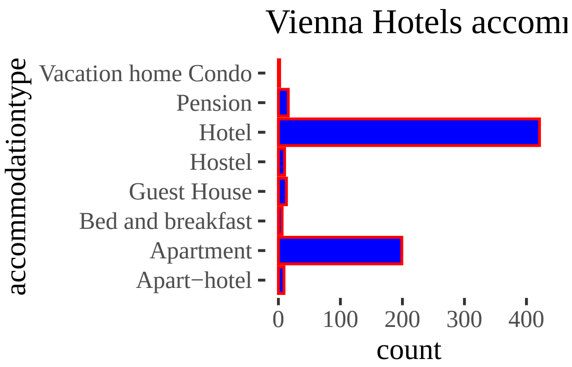
[[7]]



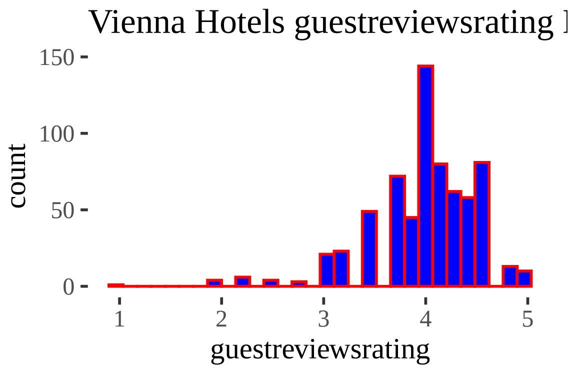
[[8]]



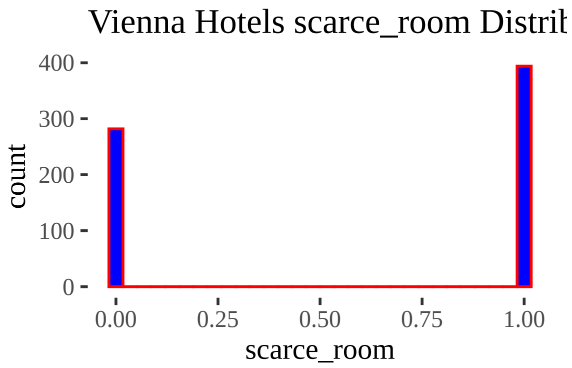
[[9]]



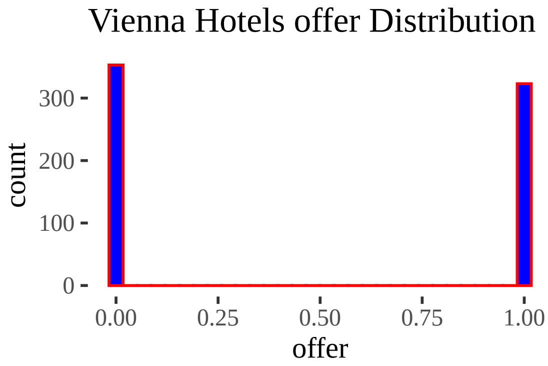
[[10]]



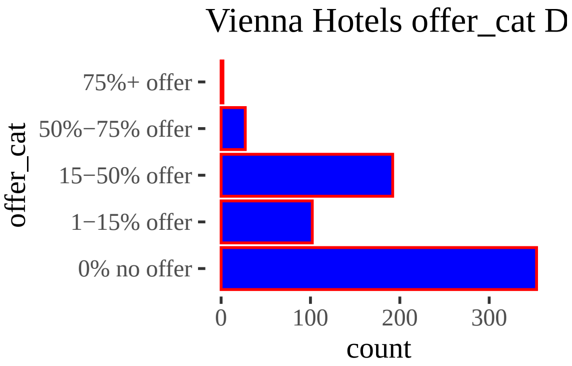
[[11]]



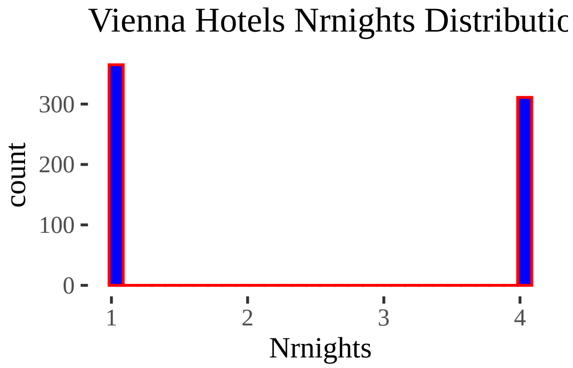
[[12]]



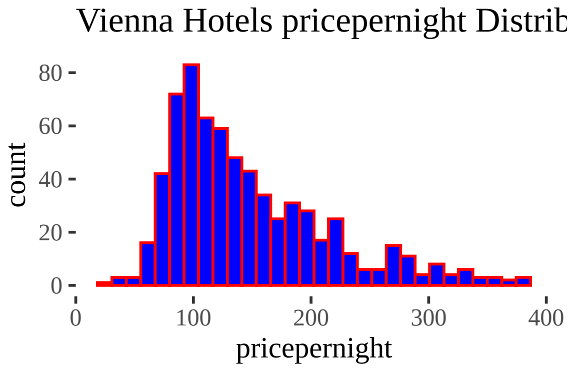
[[13]]



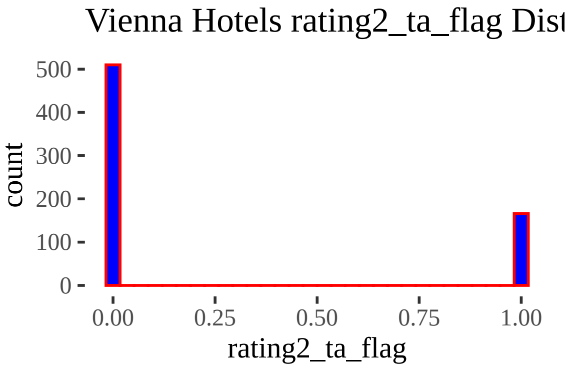
[[14]]



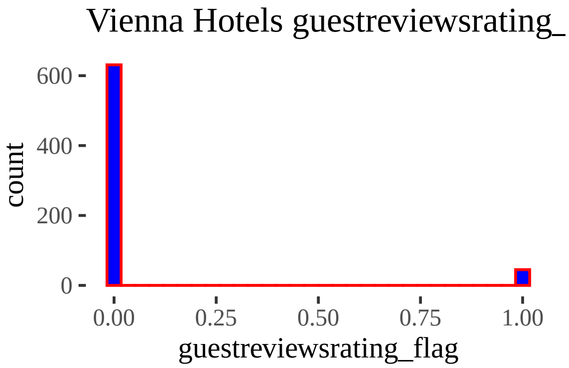
[[15]]



[[16]]

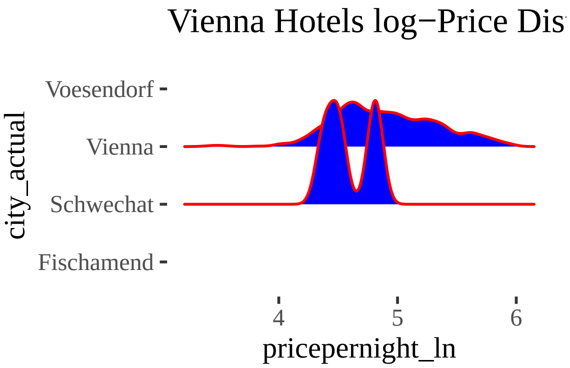


[[17]]

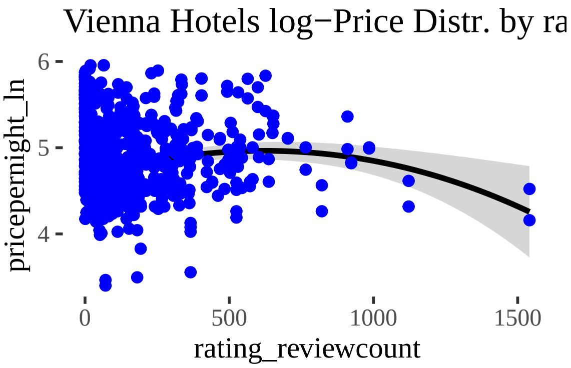


[[18]]

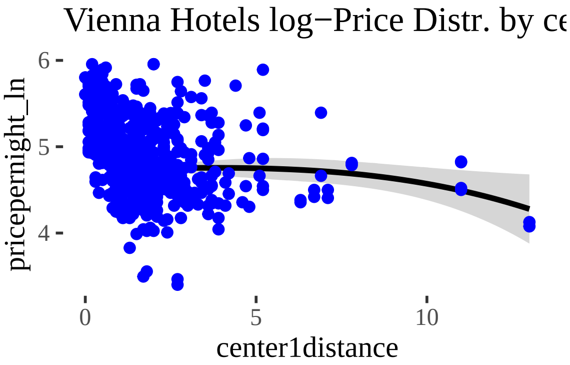
2) Scatterplot Associations & Ridge Distributions



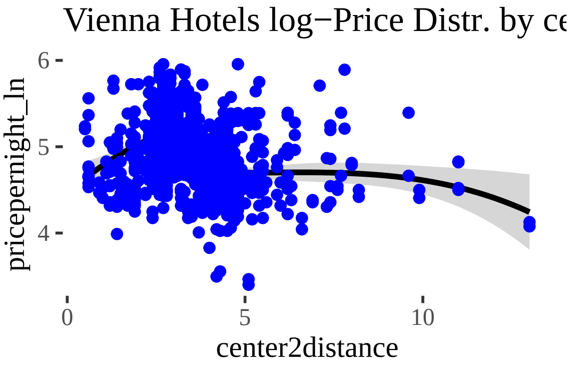
[[1]]



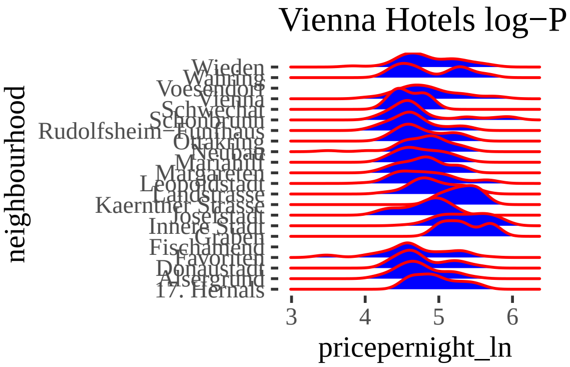
[[2]]



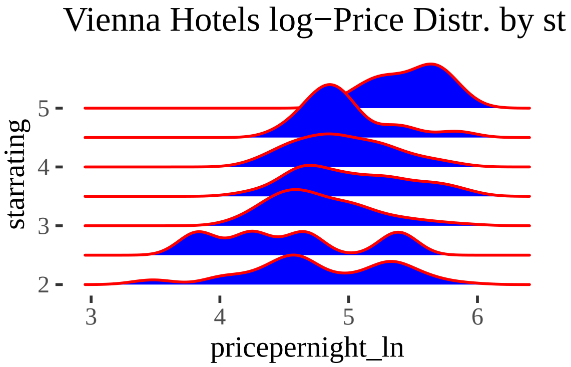
[[3]]



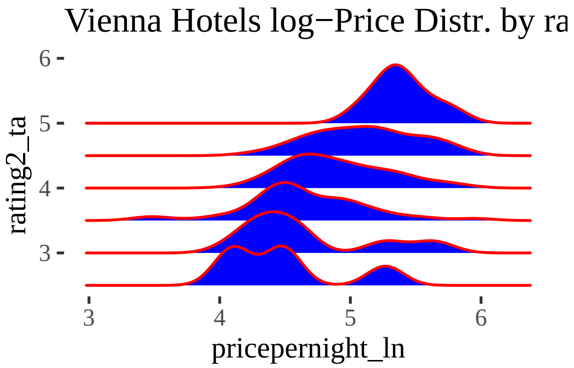
[[4]]



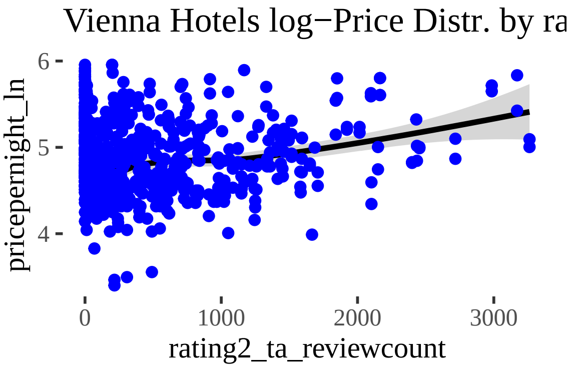
[[5]]



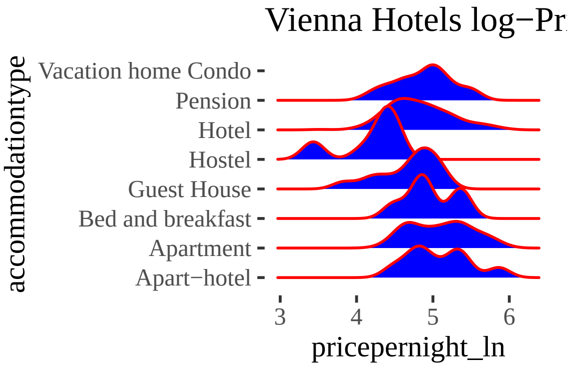
[[6]]



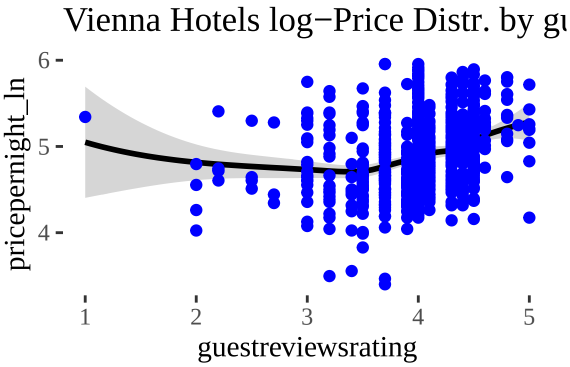
[[7]]



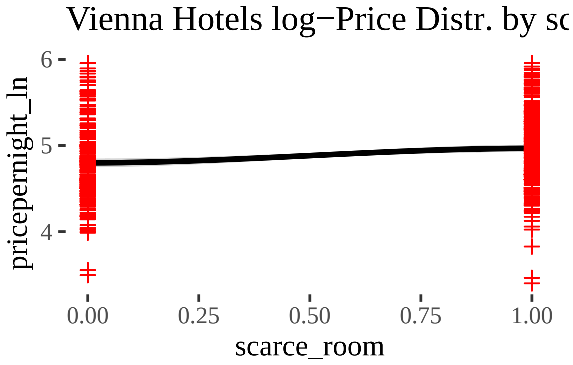
[[8]]



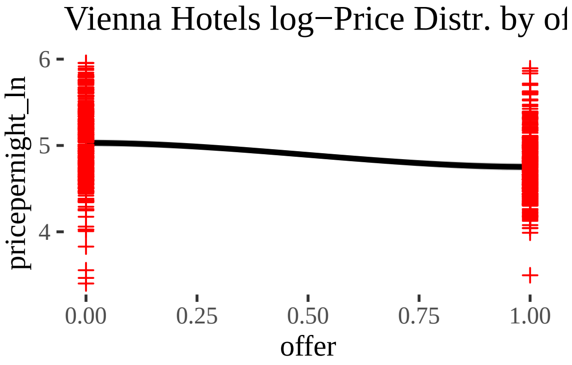
[[9]]



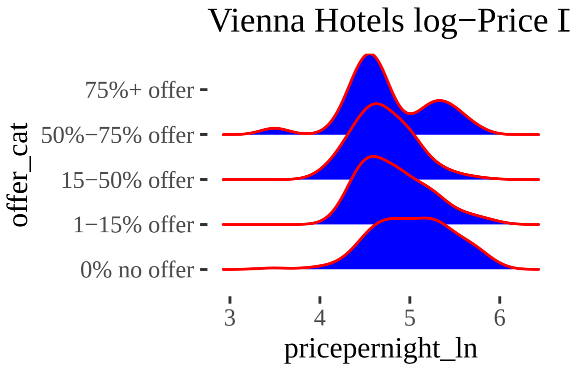
[[10]]



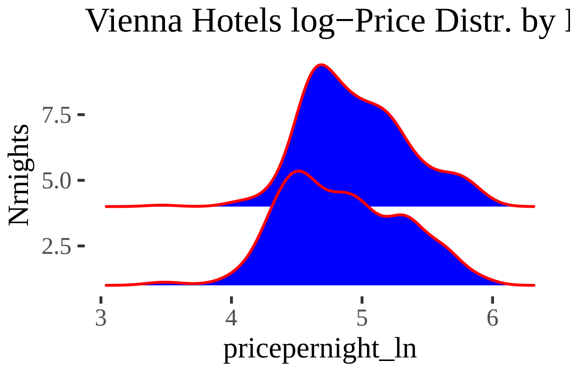
[[11]]



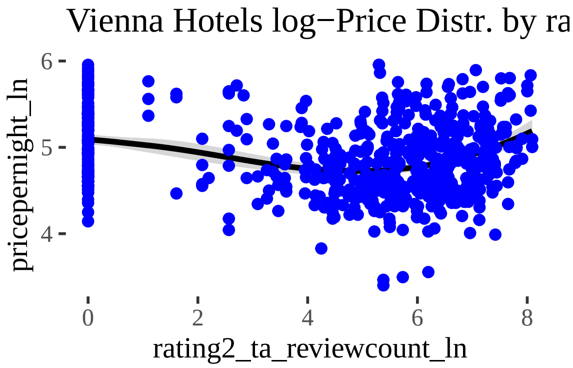
[[12]]



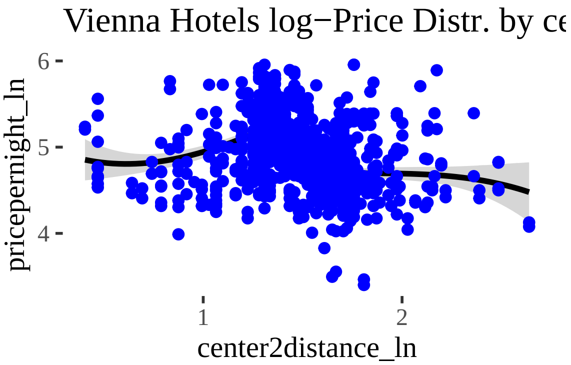
[[13]]



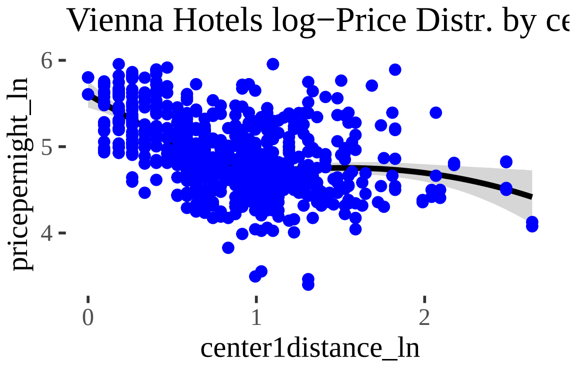
[[14]]


Vienna Hotels log−Price Distr. by ra
pricepernight_ln
6
5
4
0
2
4
6
8
rating2_ta_reviewcount_ln

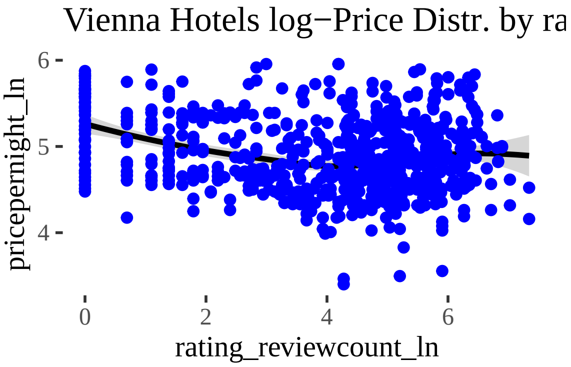
[[15]]


Vienna Hotels log−Price Distr. by ce
pricepernight_ln
6
5
4
1
2
center2distance_ln

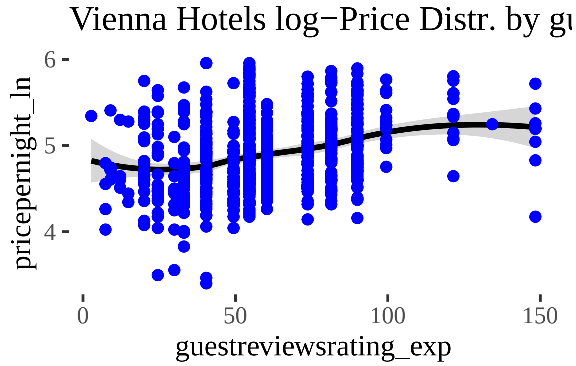
[[16]]


Vienna Hotels log−Price Distr. by ce
pricepernight_ln
6
5
4
0
1
2
center1distance_ln

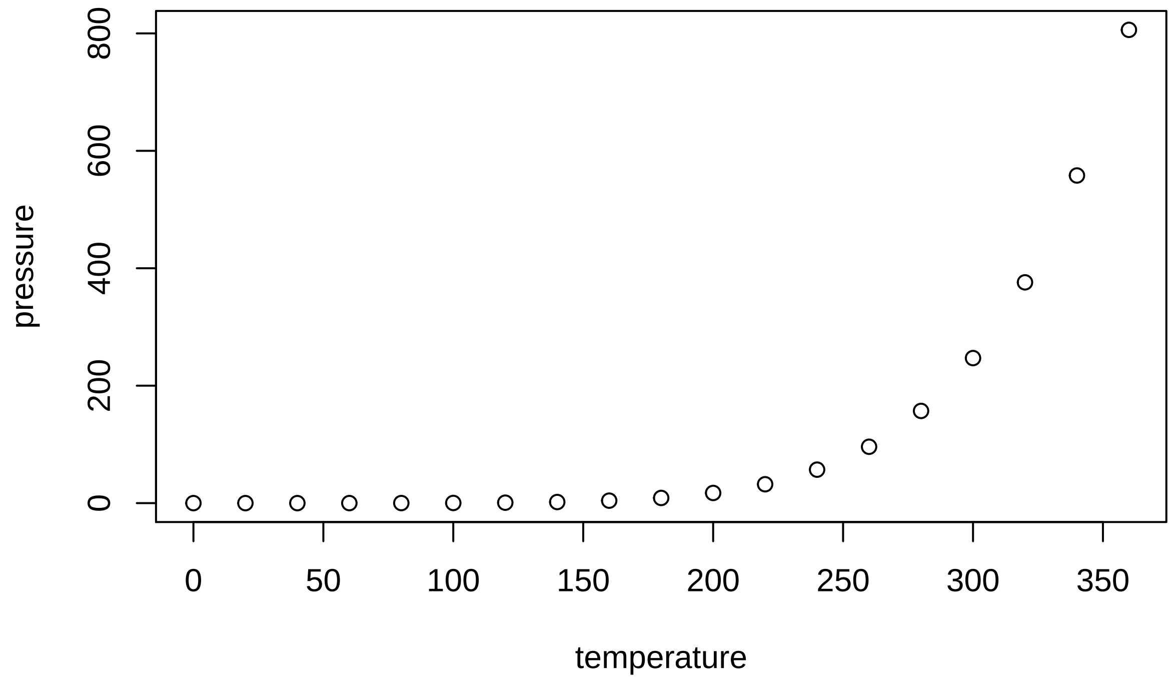
[[17]]


Vienna Hotels log−Price Distr. by ra
pricepernight_ln
6
5
4
0
2
4
6
rating_reviewcount_ln

[[18]]


Vienna Hotels log−Price Distr. by gu
pricepernight_ln
6
5
4
0
50
100
150
guestreviewsrating_exp

[[19]]



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.