AirBnb Analysis in New York

Chaoqun Yin

12/3/2018

# 1 Abstract

Airbnb is changing people’s habits for accommodation when traveling. And there rises an problem how customers can find the cost-effective accommodation and how owners set the the proper price for their properties.

To address this problem, I set New York as example to analyze what factors have a influence on the prices of the airbnb accommodation. The report are consisted of the parts of **introduction, data preparing and modeling, results, discussion and other additional information**. I mainly utlized multilevel models to conduct regression modeling.

# 2 Introduction

## 2.1 Background

Several days ago when I first came to New York City for traveling, I chose Airbnb as the platform to reserve the accommodation. As for why to choose Airbnb, it is because that shared economy industry are changing the whole world, that is, it provides premium vacation rentals from certified owners - boutique hotels, homes, rooms, and more. Airbnb has 5+ million unique listings worldwide.

However, I am also puzzled that how to choose a reasonable and cost-efficient residence when facing thousands of residences in New York. Therefore, I will conduct a series of analysis to dig into the Airbnb residences to help me and other people that want to find a residence in Airbnb in New York make a decision to choose the proper living place.

## 2.2 Previous Work

The datasets of airbnb are open and were analyzed several times before about the rating and prices of accommodation.

# 3 Data Preparing and Modeling

## 3.1 Data Source

I got the data from the site tomslee.net/airbnb-data-collection-get-the-data that is public to everyone. The data contains much information about the residences on some specific days in New York. Here is the data description:

* oom\_id: A unique number identifying an Airbnb listing.
* host\_id: A unique number identifying an Airbnb host.
* room\_type: One of “Entire home/apt”, “Private room”, or “Shared room”
* borough: A subregion of the city or search area for which the survey is carried out. The borough is taken from a shapefile of the city that is obtained independently of the Airbnb web site.
* neighborhood: As with borough: a subregion of the city or search area for which the survey is carried out.
* reviews: The number of reviews that a listing has received. Airbnb has said that 70% of visits end up with a review, so the number of reviews can be used to estimate the number of visits.
* overall\_satisfaction: The average rating (out of five) that the listing has received from those visitors who left a review.
* accommodates: The number of guests a listing can accommodate.
* bedrooms: The number of bedrooms a listing offers.
* price: The price (in $US) for a night stay. In early surveys, there may be some values that were recorded by month.
* minstay: The minimum stay for a visit, as posted by the host.
* latitude and longitude: The latitude and longitude of the listing as posted on the Airbnb site: this may be off by a few hundred metres. I do not have a way to track individual listing locations with
* last\_modified: the date and time that the values were read from the Airbnb web site.

## 3.2 Overall of the data

In this section, I will load the data and check the basic information of the data. Then I will use some visualization to show how the residences’ features distribute. Based on the findings observing the visualization, I will choose the proper models to conduct analysis.

### Prepare the data

During this part, I load the data and it has 430783 rows and 16 columns.

0 ### Data Structure During this part, I check the data structure and data type of each variables using str().

str(ny)

## 'data.frame': 430783 obs. of 16 variables:  
## $ room\_id : int 3557831 7780413 1788989 9680791 8204392 7455635 1742471 2045660 9603803 3162817 ...  
## $ host\_id : num 17911953 4887492 7607092 3201897 11268108 ...  
## $ room\_type : Factor w/ 4 levels "","Entire home/apt",..: 3 4 2 3 3 2 2 3 3 2 ...  
## $ borough : Factor w/ 5 levels "Bronx","Brooklyn",..: 3 3 3 2 2 3 4 2 2 2 ...  
## $ neighborhood : Factor w/ 240 levels "Allerton","Arden Heights",..: 65 199 180 162 217 34 191 154 13 217 ...  
## $ reviews : int 43 4 9 1 0 11 17 7 1 2 ...  
## $ overall\_satisfaction: num 4.5 5 5 4 NA 4.5 5 4.5 3 5 ...  
## $ accommodates : num 2 4 4 2 2 2 2 2 1 4 ...  
## $ bedrooms : num 1 1 1 1 1 0 1 1 1 2 ...  
## $ price : num 115 69 225 75 500 110 120 55 50 350 ...  
## $ minstay : num 2 2 3 3 1 2 3 1 1 2 ...  
## $ latitude : num 40.7 40.7 40.7 40.7 40.7 ...  
## $ longitude : num -74 -74 -74 -74 -73.9 ...  
## $ last\_modified : Factor w/ 430798 levels "2016-01-21 07:02:47.986871",..: 36003 36001 35999 35997 35996 35995 35994 35992 35991 35988 ...  
## $ Date : Date, format: "2016-01-01" "2016-01-01" ...  
## $ pricepp : num 57.5 17.2 56.2 37.5 250 ...

We can get from the output that these variables like **room\_type, borough, neiborhood** are factors having different level. Therefore, I can put these variables as category into my model. Then as for numerical variables such as **reviews, overall\_satisfaction, accommodates, bedrooms**, they are discrete data rather than consecutive data. So they also can be category variables in my model.

### Data overview

During this part, I use summary() to get the overview of the dataset.

summary(ny)

## room\_id host\_id room\_type   
## Min. : 105 Min. : 43 : 0   
## 1st Qu.: 3907848 1st Qu.: 4071690 Entire home/apt:220697   
## Median : 7708956 Median : 13972066 Private room :197044   
## Mean : 7488534 Mean : 22168318 Shared room : 13042   
## 3rd Qu.:10686406 3rd Qu.: 35322092   
## Max. :16544971 Max. :108817370   
##   
## borough neighborhood reviews   
## Bronx : 6342 Williamsburg : 38146 Min. : 0.00   
## Brooklyn :174455 Bedford-Stuyvesant: 27952 1st Qu.: 1.00   
## Manhattan :209213 Harlem : 25489 Median : 3.00   
## Queens : 38367 Upper West Side : 20398 Mean : 13.75   
## Staten Island: 2406 East Village : 19989 3rd Qu.: 15.00   
## Bushwick : 19643 Max. :387.00   
## (Other) :279166   
## overall\_satisfaction accommodates bedrooms price   
## Min. :0.00 Min. : 1.000 Min. : 0.00 Min. : 10.0   
## 1st Qu.:4.50 1st Qu.: 2.000 1st Qu.: 1.00 1st Qu.: 70.0   
## Median :4.50 Median : 2.000 Median : 1.00 Median : 110.0   
## Mean :4.37 Mean : 2.738 Mean : 1.14 Mean : 147.4   
## 3rd Qu.:5.00 3rd Qu.: 4.000 3rd Qu.: 1.00 3rd Qu.: 175.0   
## Max. :5.00 Max. :16.000 Max. :10.00 Max. :10000.0   
## NA's :154955 NA's :8417 NA's :38053   
## minstay latitude longitude   
## Min. : 1 Min. :40.50 Min. :-74.24   
## 1st Qu.: 1 1st Qu.:40.69 1st Qu.:-73.98   
## Median : 2 Median :40.72 Median :-73.96   
## Mean : 3 Mean :40.73 Mean :-73.96   
## 3rd Qu.: 3 3rd Qu.:40.76 3rd Qu.:-73.94   
## Max. :1250 Max. :40.91 Max. :-73.71   
## NA's :47723   
## last\_modified Date   
## 2016-01-21 07:02:47.986871: 1 Min. :2016-01-01   
## 2016-01-21 07:02:53.241423: 1 1st Qu.:2016-04-01   
## 2016-01-21 07:03:21.182491: 1 Median :2016-07-01   
## 2016-01-21 07:03:23.417401: 1 Mean :2016-06-19   
## 2016-01-21 07:03:25.318650: 1 3rd Qu.:2016-10-01   
## 2016-01-21 07:03:32.555722: 1 Max. :2016-12-01   
## (Other) :430777   
## pricepp   
## Min. : 0.625   
## 1st Qu.: 32.500   
## Median : 47.500   
## Mean : 57.293   
## 3rd Qu.: 67.500   
## Max. :9998.000   
## NA's :8417

We can see from the output that these variables may have correlation with the price of each listing: **room\_type, borough, neighborhood, reviews, overall\_satisfaction, accommodates, bedrooms**. It is a suppose and I will explore the correlation of variables mainly with the prices in the exploratory data analysis and based the outcome to help choose proper model to analyze the price.

## 3.3 Exploratory Data Analysis

#### The correlation plot

# create the subset  
set.seed(2018)  
ny %>%  
 dplyr::select(Date, price, room\_type, borough, reviews, overall\_satisfaction, neighborhood, accommodates, bedrooms) %>%  
 na.omit(accommodates, bedrooms, room\_type, neighborhood, borough) %>%  
 sample\_n(0.1 \* nrow(ny))-> ny.m  
  
ny.m %>%  
 dplyr::select(reviews, overall\_satisfaction, accommodates, bedrooms, price) ->cor.p  
library("PerformanceAnalytics")

## Loading required package: xts

## Loading required package: zoo

##   
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':  
##   
## as.Date, as.Date.numeric

##   
## Attaching package: 'xts'

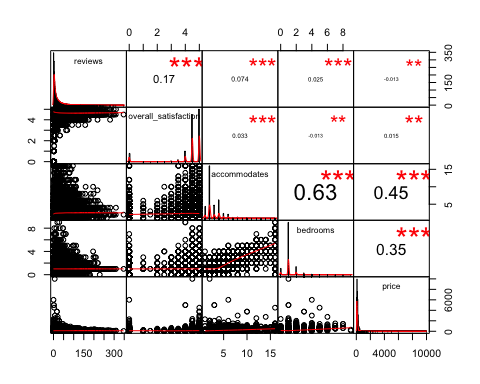
## The following objects are masked from 'package:dplyr':  
##   
## first, last

##   
## Attaching package: 'PerformanceAnalytics'

## The following object is masked from 'package:wordcloud':  
##   
## textplot

## The following object is masked from 'package:graphics':  
##   
## legend

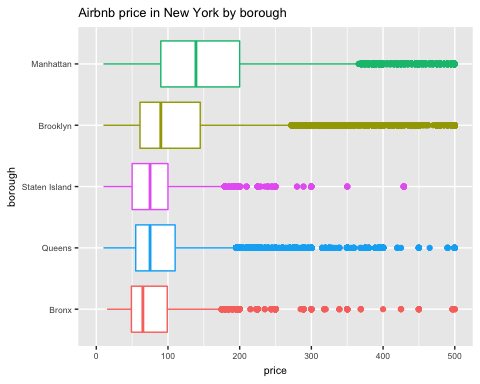
chart.Correlation(cor.p, histogram=TRUE, pch=19)

 From the correlation plot, firstly we can get the distribution of the five variables on the diagonal. Then we can see the bivariate scatter plots with a fitted line on the bottom of the diagonal that there are correlation between these variables. Finally, we can see the numbers on the top of the diagonal that the correlation and the significant level for price are high. Therefore, I consider to put these variables into my model to predict the accommodation price.

#### The boxplot of borough and the prices

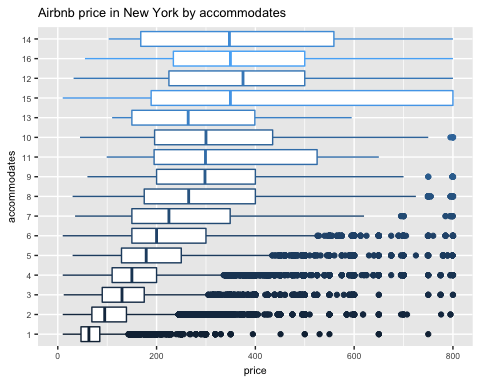
ggplot(ny, aes(x=reorder(borough, price, median),y = price,group = borough,   
 colour = borough)) +  
 geom\_boxplot() +  
 ylim(0,500) +  
 theme(text=element\_text(size = 8),legend.position = "none") +  
 labs(title="Airbnb price in New York by borough", x="borough") +  
 coord\_flip()

## Warning: Removed 7705 rows containing non-finite values (stat\_boxplot).

 We can see from the boxplot that the median price of each borough has *obvious difference*. Manhattan borough has the highest median price rather than other borough while Bronx has the lowest median price. So we can infer from the plot that *borough* is a feature that influences the price of one property.

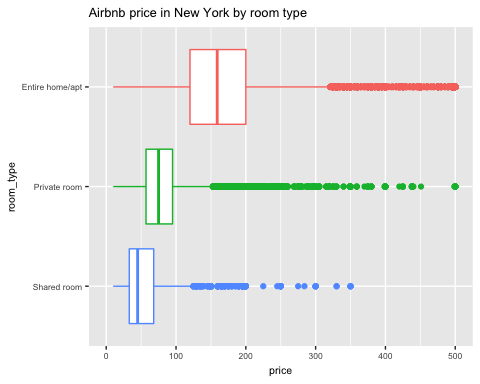
#### Boxplot of accommodates and the prices

ny %>%  
 na.omit(ny$accommodates) %>%  
ggplot(aes(x=reorder(accommodates, price, median),y = price,group = accommodates,   
 colour = accommodates)) +  
 geom\_boxplot() +  
 ylim(0,800) +  
 theme(text=element\_text(size = 8),legend.position = "none") +  
 labs(title="Airbnb price in New York by accommodates", x="accommodates") +  
 coord\_flip()

 Similarly, we can see from the boxplot that the median price of each number of accommodates has *obvious difference*. The accommodates of 14 has the highest median price rather than other number while accommodation of 1 has the lowest median price. So we can infer from the plot that *accommodation* is a feature that influences the price of one property.

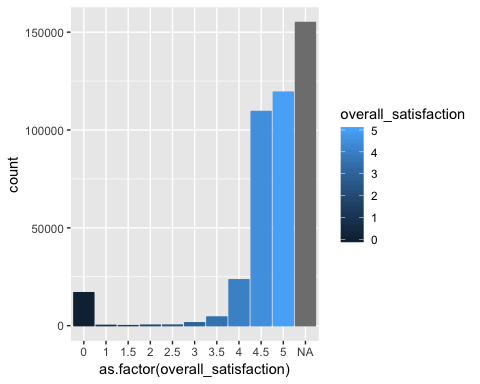
#### Boxplot of the room types and the price

ny %>%  
 na.omit(ny$room\_type) %>%  
ggplot(aes(x=reorder(room\_type, price, median),y = price,group = room\_type,   
 colour = room\_type)) +  
 geom\_boxplot() +  
 ylim(0,500) +  
 theme(text=element\_text(size = 8),legend.position = "none") +  
 labs(title="Airbnb price in New York by room type", x="room\_type") +  
 coord\_flip()

 Similarly, we can see from the boxplot that the median price of each type of room has *obvious difference*. The entire apartment room type has the highest median price rather than other types while the shared room has the lowest median price. So we can infer from the plot that *room\_type* is a feature that influences the price of one property.

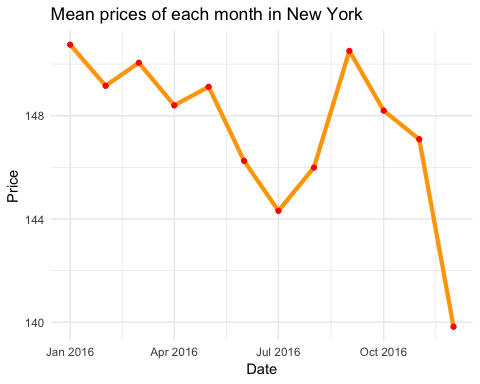
#### Diagram of reviews

ggplot(data = ny, aes(x = as.factor(overall\_satisfaction), col = overall\_satisfaction, fill = overall\_satisfaction)) +  
 geom\_histogram(stat = "count")



#### Mean prices of each month in New York

#Calculate the mean of home values  
MHV <- aggregate(price ~ Date, ny, mean)  
  
# Plot it  
#Calculate the mean of home values  
MHV <- aggregate(price ~ Date, ny, mean)  
# Plot it  
ggplot(data = MHV, aes(x = Date, y = price)) +  
 geom\_line(col = "orange", size = 1.5) +  
 geom\_point(col = "red") +  
 labs(y = "Price", title = "Mean prices of each month in New York")+  
 theme\_minimal()



## 3.4 Model Used

**The multilevel model is chosen to analyze the prices.**

* At the very beginning, I check whether the month is a factor influencing the prices.

# Transfer the type of the Date  
#ny$Date = factor(format(Date, format = "%B"), levels = month.name)  
  
# Fit linear regression  
fit.0 <- lm(data = ny.m, price ~ Date)  
summary(fit.0)

##   
## Call:  
## lm(formula = price ~ Date, data = ny.m)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -135.2 -65.5 -29.1 33.2 9857.5   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -5.361e+02 9.730e+01 -5.510 3.61e-08 \*\*\*  
## Date 3.975e-02 5.735e-03 6.932 4.21e-12 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 132.7 on 43076 degrees of freedom  
## Multiple R-squared: 0.001114, Adjusted R-squared: 0.001091   
## F-statistic: 48.05 on 1 and 43076 DF, p-value: 4.212e-12

* Regress **price** on **accommodates and bedrooms** and treat all other intercepts as random.

fit.1 <- lmer(data = ny.m, price ~ accommodates + bedrooms + (1 | Date) + (1 | room\_type)  
 + (1 | borough), REML = FALSE)  
display(fit.1)

## lmer(formula = price ~ accommodates + bedrooms + (1 | Date) +   
## (1 | room\_type) + (1 | borough), data = ny.m, REML = FALSE)  
## coef.est coef.se  
## (Intercept) -0.25 27.15   
## accommodates 15.87 0.46   
## bedrooms 41.68 1.09   
##   
## Error terms:  
## Groups Name Std.Dev.  
## Date (Intercept) 1.88   
## borough (Intercept) 31.60   
## room\_type (Intercept) 39.96   
## Residual 109.81   
## ---  
## number of obs: 43078, groups: Date, 11; borough, 5; room\_type, 3  
## AIC = 527143, DIC = 527128.7  
## deviance = 527128.7

summary(fit.1)

## Linear mixed model fit by maximum likelihood ['lmerMod']  
## Formula: price ~ accommodates + bedrooms + (1 | Date) + (1 | room\_type) +   
## (1 | borough)  
## Data: ny.m  
##   
## AIC BIC logLik deviance df.resid   
## 527142.7 527203.4 -263564.3 527128.7 43071   
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -5.046 -0.338 -0.055 0.197 90.005   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## Date (Intercept) 3.543 1.882   
## borough (Intercept) 998.480 31.599   
## room\_type (Intercept) 1597.126 39.964   
## Residual 12059.171 109.814   
## Number of obs: 43078, groups: Date, 11; borough, 5; room\_type, 3  
##   
## Fixed effects:  
## Estimate Std. Error t value  
## (Intercept) -0.2453 27.1550 -0.009  
## accommodates 15.8744 0.4629 34.293  
## bedrooms 41.6809 1.0875 38.328  
##   
## Correlation of Fixed Effects:  
## (Intr) accmmd  
## accommodats -0.017   
## bedrooms -0.016 -0.637

ranef(fit.1)

## $Date  
## (Intercept)  
## 2016-01-01 0.5139702  
## 2016-02-01 -0.9564560  
## 2016-03-01 -1.5507774  
## 2016-04-01 -0.8619676  
## 2016-05-01 -0.6411910  
## 2016-06-01 -1.0860412  
## 2016-07-01 -0.5594195  
## 2016-08-01 0.4097307  
## 2016-09-01 2.0693076  
## 2016-10-01 -0.1619777  
## 2016-12-01 2.8248217  
##   
## $borough  
## (Intercept)  
## Bronx -20.441928  
## Brooklyn 4.076987  
## Manhattan 54.277402  
## Queens -11.645150  
## Staten Island -26.267311  
##   
## $room\_type  
## (Intercept)  
## Entire home/apt 52.08633  
## Private room -14.37363  
## Shared room -37.71270

* Based on model 1, include a between-group correlation between accommodates and borough.

fit.2 <- lmer(data = ny.m, price ~ accommodates + room\_type + (1 | Date) +   
 (1 | room\_type) + (1 + accommodates | borough), REML = FALSE)

## singular fit

display(fit.2)

## lmer(formula = price ~ accommodates + room\_type + (1 | Date) +   
## (1 | room\_type) + (1 + accommodates | borough), data = ny.m,   
## REML = FALSE)  
## coef.est coef.se  
## (Intercept) 88.57 2.44   
## accommodates 17.91 4.72   
## room\_typePrivate room -58.19 1.24   
## room\_typeShared room -78.61 3.31   
##   
## Error terms:  
## Groups Name Std.Dev. Corr   
## Date (Intercept) 2.39   
## borough (Intercept) 3.14   
## accommodates 10.45 -0.51   
## room\_type (Intercept) 0.00   
## Residual 110.59   
## ---  
## number of obs: 43078, groups: Date, 11; borough, 5; room\_type, 3  
## AIC = 527750, DIC = 527730.4  
## deviance = 527730.4

summary(fit.2)

## Linear mixed model fit by maximum likelihood ['lmerMod']  
## Formula:   
## price ~ accommodates + room\_type + (1 | Date) + (1 | room\_type) +   
## (1 + accommodates | borough)  
## Data: ny.m  
##   
## AIC BIC logLik deviance df.resid   
## 527750.4 527837.1 -263865.2 527730.4 43068   
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -4.997 -0.315 -0.063 0.184 89.464   
##   
## Random effects:  
## Groups Name Variance Std.Dev. Corr   
## Date (Intercept) 5.715 2.391   
## borough (Intercept) 9.831 3.135   
## accommodates 109.196 10.450 -0.51  
## room\_type (Intercept) 0.000 0.000   
## Residual 12230.964 110.594   
## Number of obs: 43078, groups: Date, 11; borough, 5; room\_type, 3  
##   
## Fixed effects:  
## Estimate Std. Error t value  
## (Intercept) 88.574 2.436 36.362  
## accommodates 17.914 4.717 3.798  
## room\_typePrivate room -58.187 1.241 -46.885  
## room\_typeShared room -78.606 3.311 -23.741  
##   
## Correlation of Fixed Effects:  
## (Intr) accmmd rm\_tPr  
## accommodats -0.357   
## rm\_typPrvtr -0.444 0.030   
## rm\_typShrdr -0.181 0.014 0.226  
## convergence code: 0  
## singular fit

#confint(fit.2)

* Then, I regress price on accomodates and bedrooms while including one between-group and other random effect.

fit.3 <- lmer(data = ny.m, price ~ accommodates + bedrooms + (1 | room\_type)  
 + (1 | Date) + (1 + bedrooms | borough), REML = FALSE)  
 display(fit.3)

## lmer(formula = price ~ accommodates + bedrooms + (1 | room\_type) +   
## (1 | Date) + (1 + bedrooms | borough), data = ny.m, REML = FALSE)  
## coef.est coef.se  
## (Intercept) 24.42 22.79   
## accommodates 16.27 0.46   
## bedrooms 19.86 10.28   
##   
## Error terms:  
## Groups Name Std.Dev. Corr   
## Date (Intercept) 1.81   
## borough (Intercept) 4.62   
## bedrooms 22.65 0.69   
## room\_type (Intercept) 39.16   
## Residual 109.03   
## ---  
## number of obs: 43078, groups: Date, 11; borough, 5; room\_type, 3  
## AIC = 526532, DIC = 526514.4  
## deviance = 526514.4

# 4 Result

## 4.1 Model Choice

### Intercept plot

dwplot(list(fit.1, fit.2, fit.3), show\_intercept = TRUE)

## Warning in bind\_rows\_(x, .id): binding factor and character vector,  
## coercing into character vector

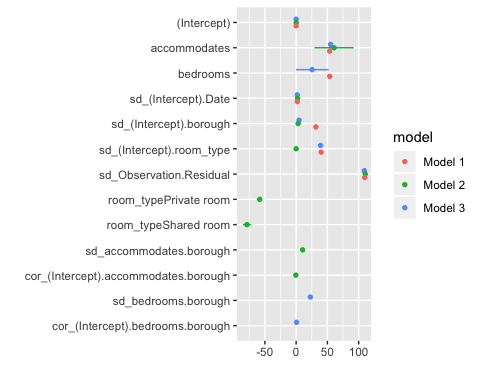
## Warning in bind\_rows\_(x, .id): binding character and factor vector,  
## coercing into character vector

## Warning in bind\_rows\_(x, .id): binding factor and character vector,  
## coercing into character vector

## Warning in bind\_rows\_(x, .id): binding character and factor vector,  
## coercing into character vector

## Warning in bind\_rows\_(x, .id): binding factor and character vector,  
## coercing into character vector

## Warning in bind\_rows\_(x, .id): binding character and factor vector,  
## coercing into character vector



### ANOVA

anova(fit.1)

## Analysis of Variance Table  
## Df Sum Sq Mean Sq F value  
## accommodates 1 69846202 69846202 5792  
## bedrooms 1 17715456 17715456 1469

#chisq.test() #check if the variables improve the model  
anova(fit.1, fit.2, fit.3)

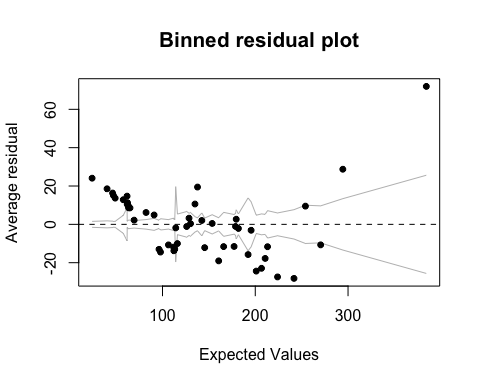
## Data: ny.m  
## Models:  
## fit.1: price ~ accommodates + bedrooms + (1 | Date) + (1 | room\_type) +   
## fit.1: (1 | borough)  
## fit.3: price ~ accommodates + bedrooms + (1 | room\_type) + (1 | Date) +   
## fit.3: (1 + bedrooms | borough)  
## fit.2: price ~ accommodates + room\_type + (1 | Date) + (1 | room\_type) +   
## fit.2: (1 + accommodates | borough)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)   
## fit.1 7 527143 527203 -263564 527129   
## fit.3 9 526532 526610 -263257 526514 614.28 2 <2e-16 \*\*\*  
## fit.2 10 527750 527837 -263865 527730 0.00 1 1   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

## 4.2 Interpretation

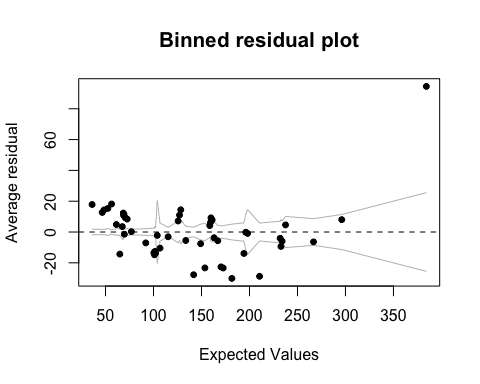
## 4.3 Model Check

### 

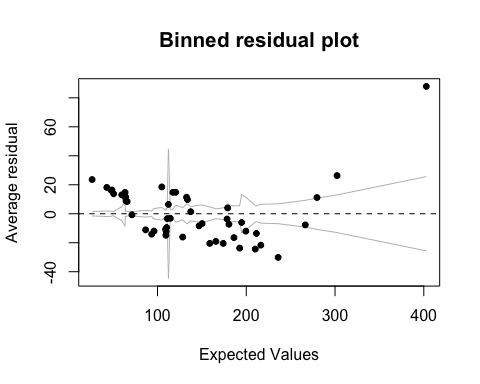
binnedplot(fitted(fit.1), resid(fit.1), cex.main=1.3, main="Binned residual plot", nclass = 50)



binnedplot(fitted(fit.2), resid(fit.1), cex.main=1.3, main="Binned residual plot", nclass = 50)



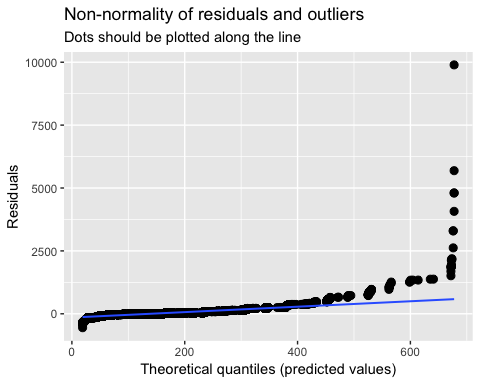
binnedplot(fitted(fit.3), resid(fit.1), cex.main=1.3, main="Binned residual plot", nclass = 50)



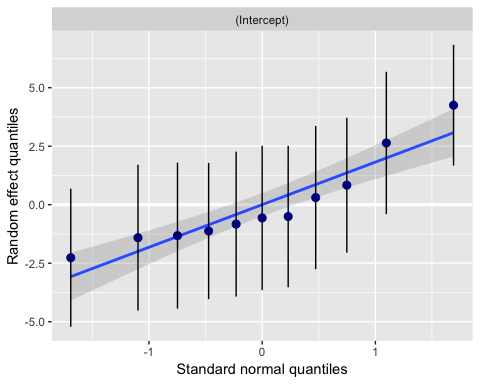
### 

library(sjPlot)  
plot\_model(fit.2, type = "diag", show.values = TRUE, value.offset = .3)

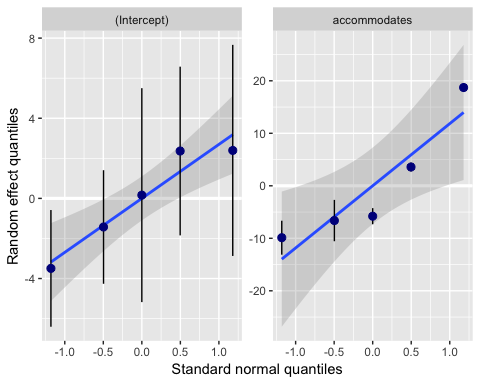
## [[1]]



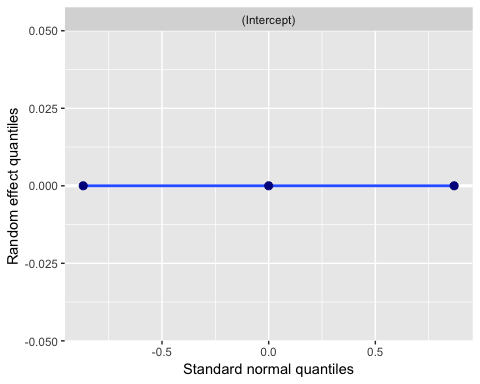
##   
## [[2]]  
## [[2]]$Date



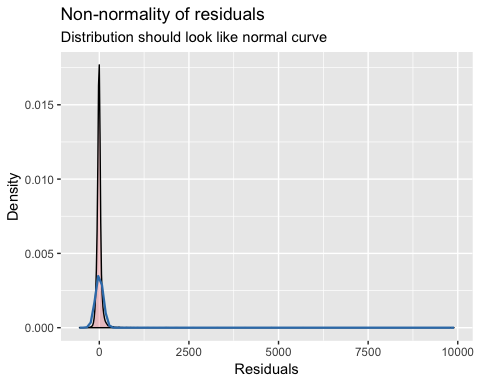
##   
## [[2]]$borough



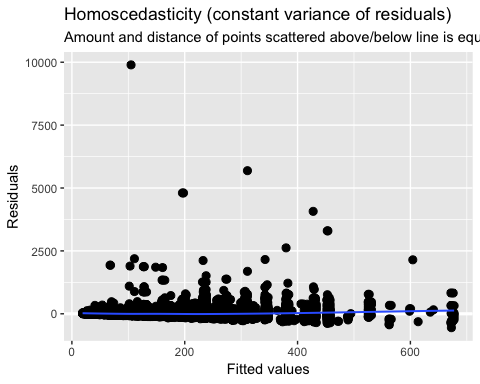
##   
## [[2]]$room\_type



##   
##   
## [[3]]



##   
## [[4]]

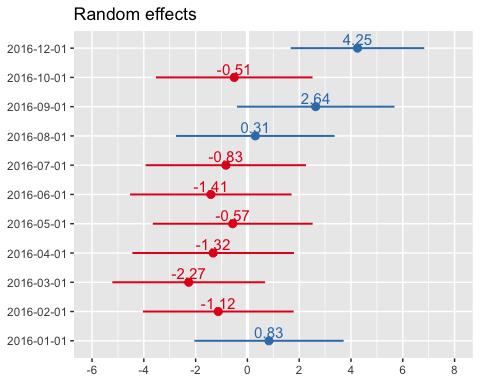


#binned plot HW5

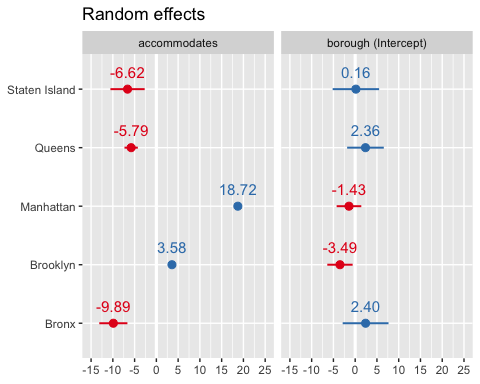
### 

plot\_model(fit.2, type = "re", show.values = TRUE, value.offset = .3)

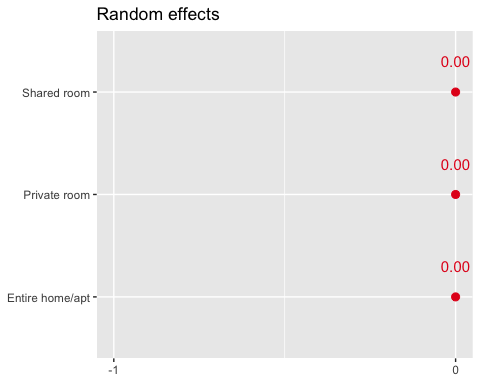
## [[1]]



##   
## [[2]]

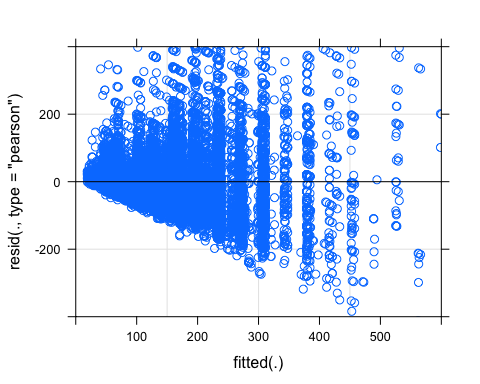


##   
## [[3]]



### 

plot(fit.2, ylim = c(-400,400), xlim = c(0,600))



## Prediction

# 5 Discussion

## 5.1 Implication

## 5.2 Limitation

## 5.3 Future Direction

# 6 Acknowledgement

# 7 Reference

# 8 Appendix