Analysis of Air BnB data

Will Harrison

First set up the required packages and load in the data.

```r
library(ggplot2)
library(dplyr)
library(boot)
library(patchwork)
library(MASS)
```

```r
bar_wday <- read.csv("barcelona_weekdays.csv") # read in the data
bar_wend <- read.csv("barcelona_weekends.csv")
lon_wday <- read.csv("london_weekdays.csv")
lon_wend <- read.csv("london_weekends.csv")
```

Analysing the Airbnb Price Data in European Cities

Task 1: *Are there any missing data in any of the datasets? Comment on if there is/are any variable/variables that may not be useful for further analysis. Calculate the average listing price per person per night for each room type for the weekdays data in Barcelona.* [3 marks]

```r
summary(bar_wday) # summary of one of the datasets
```

```
##        X             realSum         room_type          room_shared       
##  Min.   :   0.0   Min.   :  69.59   Length:1555        Length:1555       
##  1st Qu.: 388.5   1st Qu.: 161.98   Class :character   Class :character  
##  Median : 777.0   Median : 208.53   Mode  :character   Mode  :character  
##  Mean   : 777.0   Mean   : 288.39                                        
##  3rd Qu.:1165.5   3rd Qu.: 335.37                                        
##  Max.   :1554.0   Max.   :6943.70                                        
##  room_private       person_capacity host_is_superhost      multi       
##  Length:1555        Min.   :2.000   Length:1555        Min.   :0.0000  
##  Class :character   1st Qu.:2.000   Class :character   1st Qu.:0.0000  
##  Mode  :character   Median :2.000   Mode  :character   Median :0.0000  
##                     Mean   :2.756                      Mean   :0.3768  
##                      3rd Qu.:3.000                      3rd Qu.:1.0000
##                      Max.   :6.000                      Max.   :1.0000
##       biz         cleanliness_rating guest_satisfaction_overall    bedrooms
##  Min.   :0.0000   Min.   : 2.000     Min.   : 20.00             Min.   :0.000
##  1st Qu.:0.0000   1st Qu.: 9.000     1st Qu.: 88.00             1st Qu.:1.000
##  Median :0.0000   Median :10.000     Median : 93.00             Median :1.000
##  Mean   :0.3505   Mean   : 9.286     Mean   : 90.93             Mean   :1.217
##  3rd Qu.:1.0000   3rd Qu.:10.000     3rd Qu.: 97.00             3rd Qu.:1.000
##  Max.   :1.0000   Max.   :10.000     Max.   :100.00             Max.   :6.000
##       dist          metro_dist       attr_index      attr_index_norm
##  Min.   :0.1199   Min.   :0.0130   Min.   :  93.82   Min.   :  3.198
##  1st Qu.:1.0906   1st Qu.:0.2521   1st Qu.: 282.77   1st Qu.:  9.637
##  Median :1.7518   Median :0.3705   Median : 389.20   Median : 13.265
##  Mean   :2.1173   Mean   :0.4349   Mean   : 464.37   Mean   : 15.827
##  3rd Qu.:2.9492   3rd Qu.:0.5542   3rd Qu.: 591.59   3rd Qu.: 20.162
##  Max.   :8.4440   Max.   :2.4028   Max.   :2934.13   Max.   :100.000
##    rest_index     rest_index_norm        lng             lat
##  Min.   : 159.8   Min.   :  3.518   Min.   :2.105   Min.   :41.35
##  1st Qu.: 494.4   1st Qu.: 10.883   1st Qu.:2.156   1st Qu.:41.38
##  Median : 801.8   Median : 17.650   Median :2.171   Median :41.39
##  Mean   : 877.7   Mean   : 19.320   Mean   :2.169   Mean   :41.39
##  3rd Qu.:1211.3   3rd Qu.: 26.663   3rd Qu.:2.179   3rd Qu.:41.40
##  Max.   :4542.8   Max.   :100.000   Max.   :2.226   Max.   :41.46
```

- Looking at the summary of the Barcelona weekdays data, there are no missing values in any of the variables. No missing data is found upon inspection of the summaries of the other datasets either.
- The variable `X` is just the observation number, so won't be useful in any further analysis.
- From the data descriptions, `attr_index_norm` and `rest_index_norm` look to be better than their counterparts `attr_index` and `rest_index` - if comparisons are to be made between Barcelona and London, these metrics to be on the same scale to draw any meaningful comparisons or predictions.
- `room_shared` and `room_private` are just dummy variables whose information is captured in `room_type`, these two variables won't really be useful in analysis or modelling.
- `lat` and `lng` are specific to each city and so won't be useful for comparisons between the two cities.

`realSum` is the price for two nights for two people, so the price of one night for two people is `realSum/2`, and the price of one night for one person is `realSum/4`. Take the average of this across each room type in the Barcelona weekdays data.

```
bar_wday %>%
  group_by(room_type) %>%
  summarise(price = mean(realSum/4))
```

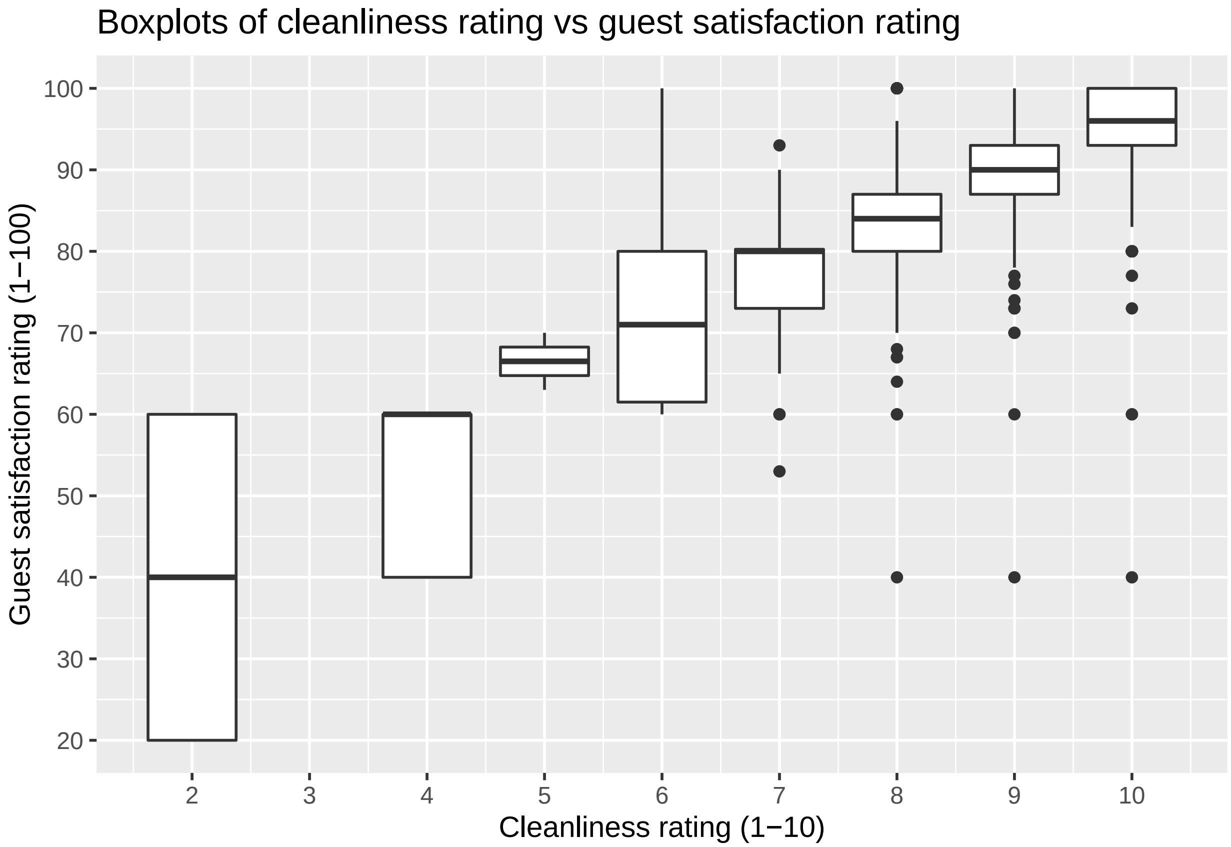
```
## # A tibble: 3 x 2
##   room_type       price
##   <chr>           <dbl>
## 1 Entire home/apt 143.
## 2 Private room     50.7
## 3 Shared room      28.7
```

- It is found that renting an entire house/apartment is a lot more expensive than just one room, and a private room on average costs more than a shared room.

Task 2: *Using appropriate exploratory tools such as tables/graphs/summary statistics comment on the relationship between cleanliness and guest satisfaction in the weekdays data in Barcelona. Also comment on the relationship between superhost and guest satisfaction using exploratory analysis on the weekdays data in London.* [4 marks]

To look at the relationship between cleanliness and guest satisfaction, use a boxplot to examine how cleanliness rating affects the distribution of guest satisfaction.

```
ggplot(data = bar_wday, aes(x = cleanliness_rating,
                            y = guest_satisfaction_overall,
                            group = cleanliness_rating)) +
  geom_boxplot() +
  labs(title = "Boxplots of cleanliness rating vs guest satisfaction rating",
       x = "Cleanliness rating (1-10)",
       y = "Guest satisfaction rating (1-100)") +
  scale_x_continuous(breaks = (0:10)) + # add more breaks for interpretation
  scale_y_continuous(breaks = (0:10)*10)
```



- No BnBs considered in the Barcelona weekday data had cleanliness ratings of 0, 1 or 3.
- Higher cleanliness ratings are associated with higher median guest satisfaction. The guest satisfaction distribution of higher cleanliness ratings also tends to be more condensed, this could be explained a far smaller samples (so larger variance) of BnBs with cleanliness ratings lower than 6. This suggests that there is a positive correlation between cleanliness and guest satisfaction.

Also examine the correlation between the two:

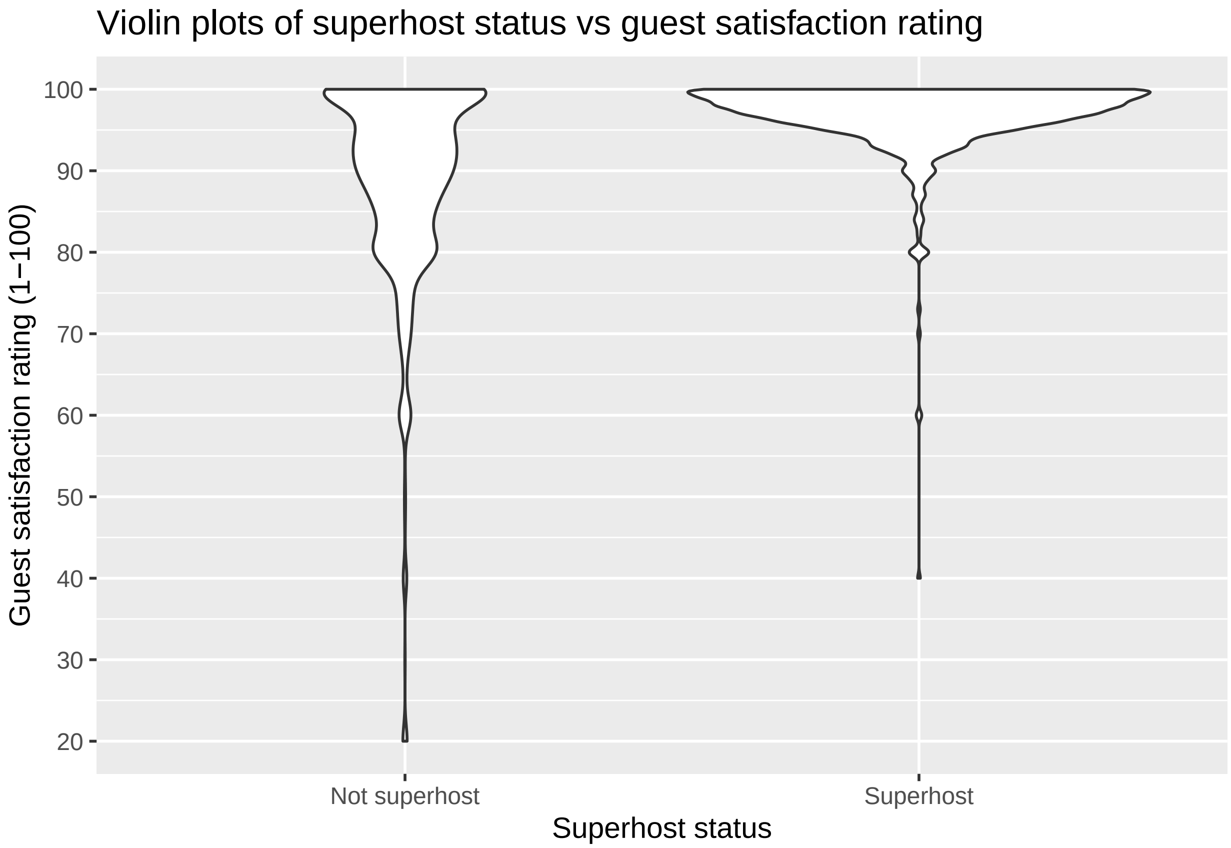
```
cor(bar_wday$cleanliness_rating, bar_wday$guest_satisfaction_overall)
```

```
## [1] 0.7201852
```

- This reinforces that cleanliness and guest satisfaction are strongly positively correlated.

To examine the relationship between superhost status and guest satisfaction, create a violin plot to determine how superhost status affects the distribution of guest satisfaction ratings.

```
ggplot(data = lon_wday, aes(x = host_is_superhost,
                            y = guest_satisfaction_overall,
                            group = host_is_superhost)) +
  geom_violin() +
  labs(title = "Violin plots of superhost status vs guest satisfaction rating",
       x = "Superhost status",
       y = "Guest satisfaction rating (1-100)") +
  scale_y_continuous(breaks = (0:10)*10) +
  scale_x_discrete(labels = c(False = "Not superhost",
                              True = "Superhost"))
```



- From the violin plots it can be seen that there does seem to be a relationship between superhost status and guest satisfaction. The distribution of guest satisfaction for superhosts is quite a bit more skewed towards perfect guest satisfaction than BnBs without the superhost status. The average guest satisfaction for superhosts is higher than non-superhosts.
- Superhosts also had a higher minimum satisfaction than non-superhosts (~40 vs ~20).

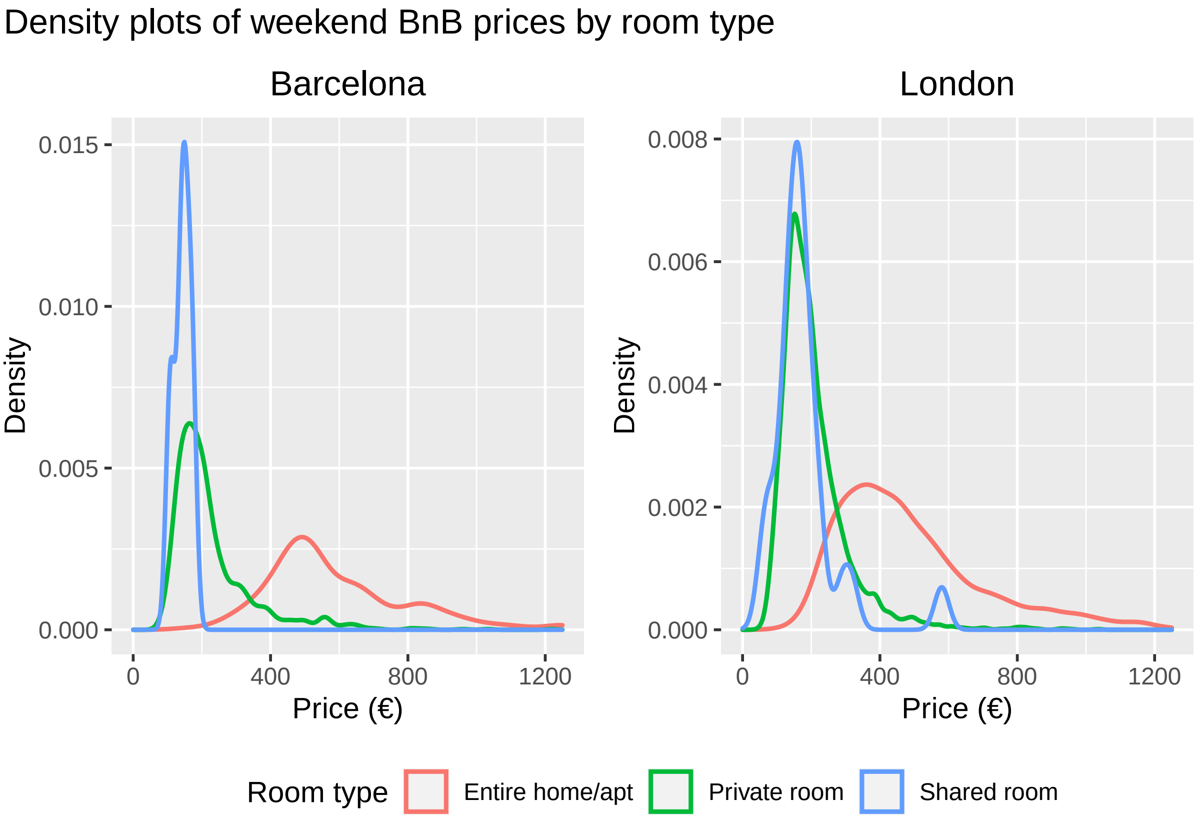
Task 3: *Use an appropriate plot to illustrate the distribution of listed room prices per room type for the Barcelona and London weekends datasets. You should provide separate plots for each city. Comment on what you observe.* [3 marks]

Create a density plot for room price for each room type using either cities day datasets, the right limit for price on the plots has been set to 1250. Beyond this, there are a few values but the densities are all approximately 0. This zooms the plot in on the interesting part.

```r
p1 <- ggplot(data = bar_wend, aes(x = realSum,
                            colour = room_type)) +
  geom_density(size = 0.8) +
  labs(title = "Barcelona",
       x = "Price (€)",
       y = "Density",
       colour = "Room type") +
    theme(plot.title = element_text(hjust = 0.5)) +
  xlim(0,1250)

p2 <- ggplot(data = lon_wend, aes(x = realSum,
                            colour = room_type)) +
  geom_density(size = 0.8) +
  labs(title = "London",
       x = "Price (€)",
       y = "Density",
       colour = "Room type") +
    theme(plot.title = element_text(hjust = 0.5)) +
  xlim(0,1250)

plt <- p1 + p2 & theme(legend.position = "bottom") # put common key
plt + plot_layout(guides = "collect") +            # at bottom of the plot
  plot_annotation(title =
                    "Density plots of weekend BnB prices by room type")
```



- In both cities on weekends the distribution of prices for entire home/apartments is higher than private rooms and shared rooms, however in London, the price distributions of private and shared rooms is much more similar than in Barcelona.
- Comparing the distributions of each room type by city, the distributions seem to be centred around similar values for private/shared rooms. For entire home/apartment, Barcelona's distribution is more symmetric, whilst London's is right skewed.
- London's distributions look to be more heavily right skewed, suggesting that they may cost more on average.

Task 4: *Perform an appropriate statistical test to compare the listed room price in London vs Barcelona on weekends. You may combine the data across room types to perform the statistical test. Discuss what assumptions you make to perform the test. Comment on the appropriateness of the assumptions made.* [6 marks]

Test the hypothesis that the mean Air BnB room price in London are more expensive than Barcelona on weekends. Let μ_L and μ_B be the mean price of rooms in London and Barcelona on weekends respectively.

The two hypotheses being tested are then:

$H_0 : \mu_L \leq \mu_B$

$H_1 : \mu_L > \mu_B$.

Use an unpaired t-test as the test statistic, considering data from all room types together for this test. By omitting the `var.equal = TRUE` command in R, a Welch's t-test can be used and the usual assumption of a t-test that the variances of the samples are equal is dropped. There is still the assumption that the mean room price of the samples are normally distributed - the central limit theorem means this assumption is OK.

Proceed with the test:

```
t.test(lon_wend$realSum, bar_wend$realSum, alternative = "greater")
```

```
##
##  Welch Two Sample t-test
##
## data:  lon_wend$realSum and bar_wend$realSum
## t = 5.1232, df = 2096.6, p-value = 1.639e-07
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  43.51931      Inf
## sample estimates:
## mean of x mean of y
##  364.3898  300.2775
```

Thus we reject the null hypothesis at the 5% level and conclude that the average price for a BnB on weekends in London is higher than in Barcelona.

Task 5: *Use a generalised linear model (GLM) to study the differences between the listed room prices on weekdays and weekends for Barcelona. Check your modelling assumptions.* [10 marks]

Convert the columns containing categorical variables to factors - in the original data, `multi` and `biz` are numerical variables with value 0 or 1 and so need to be converted so that they are properly treated. Also do this for the London data now, as it may be needed for any models we fit later.

```
cols <- c("biz", "multi", "host_is_superhost") # select columns to convert
bar_wday[cols] <- lapply(bar_wday[cols], as.factor) # convert to factor
bar_wend[cols] <- lapply(bar_wend[cols], as.factor) # in all individual
lon_wday[cols] <- lapply(lon_wday[cols], as.factor) # dataframes
lon_wend[cols] <- lapply(lon_wend[cols], as.factor)
```

Make a new dataset by combining the Barcelona weekday and day datasets, include a new column that indicates whether the data is from the day, this will be used as a factor in the model.

```
barc <- bind_rows(bar_wend, bar_wday, .id = 'day') # stack datasets together
barc$day[barc$day==1] <- "Weekend"
barc$day[barc$day==2] <- "Weekday"
```

The new data frame includes a column `day` which indicates with a character (either `"Weekend"` or `"Weekday"`) whether the observation was originally part of the weekday or weekend dataset.

Start with a linear model with all relevant predictors (exclude variables mentioned in Task 1). Use the step function to remove statistically insignificant parameters. Always include the factor `day` as this will be how to study the difference between weekdays and weekends.

In Task 2, it was found that cleanliness rating and guest satisfaction exhibit a linear relationship, due to collinearity only one of these terms should be included. Task 2 also suggested there may be an interaction between guest satisfaction and superhost status. Choose to drop cleanliness rating and include the interaction between guest satisfaction and superhost status.

```
lm_bar_full <- lm(realSum ~ room_type + person_capacity + multi + biz +
                 bedrooms + dist + metro_dist + attr_index_norm +
                 rest_index_norm + host_is_superhost*guest_satisfaction_overall
                 + day,
               data = barc)
```

Use the step function to drop statistically insignificant parameters, the scope argument ensures that day is never dropped.

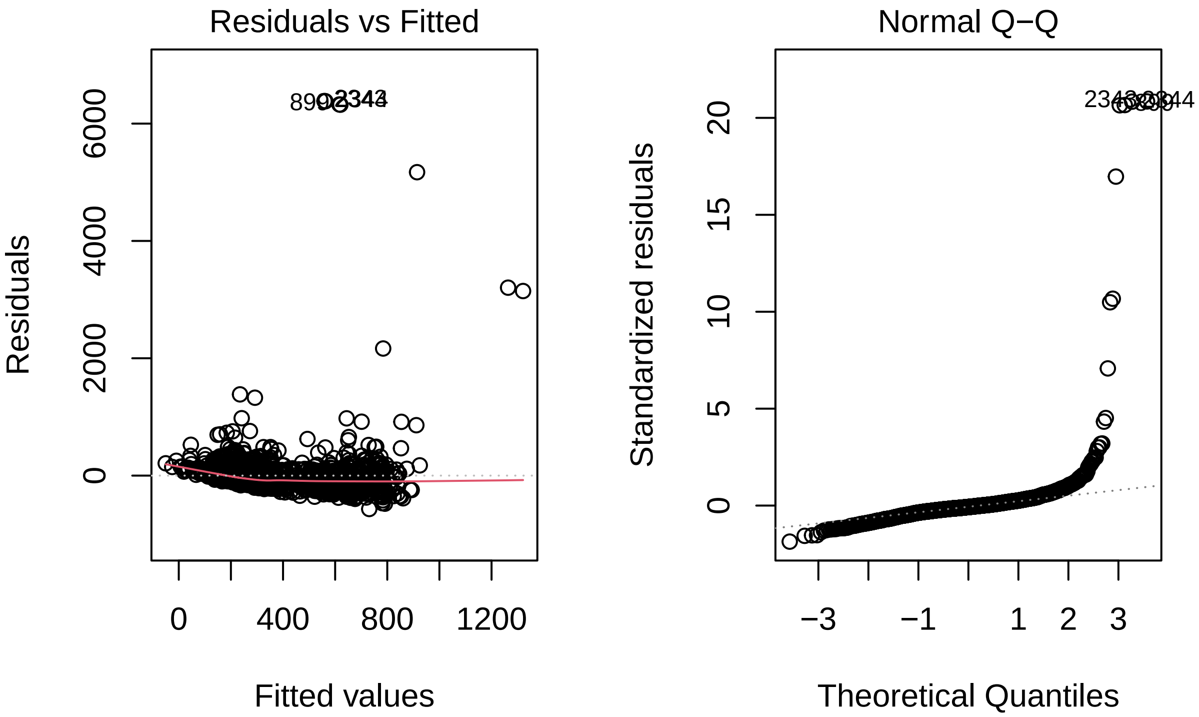
```
lm_bar_step <- step(lm_bar_full, trace = 0,
                 scope = list(lower = ~ day + room_type))
summary(lm_bar_step)
```

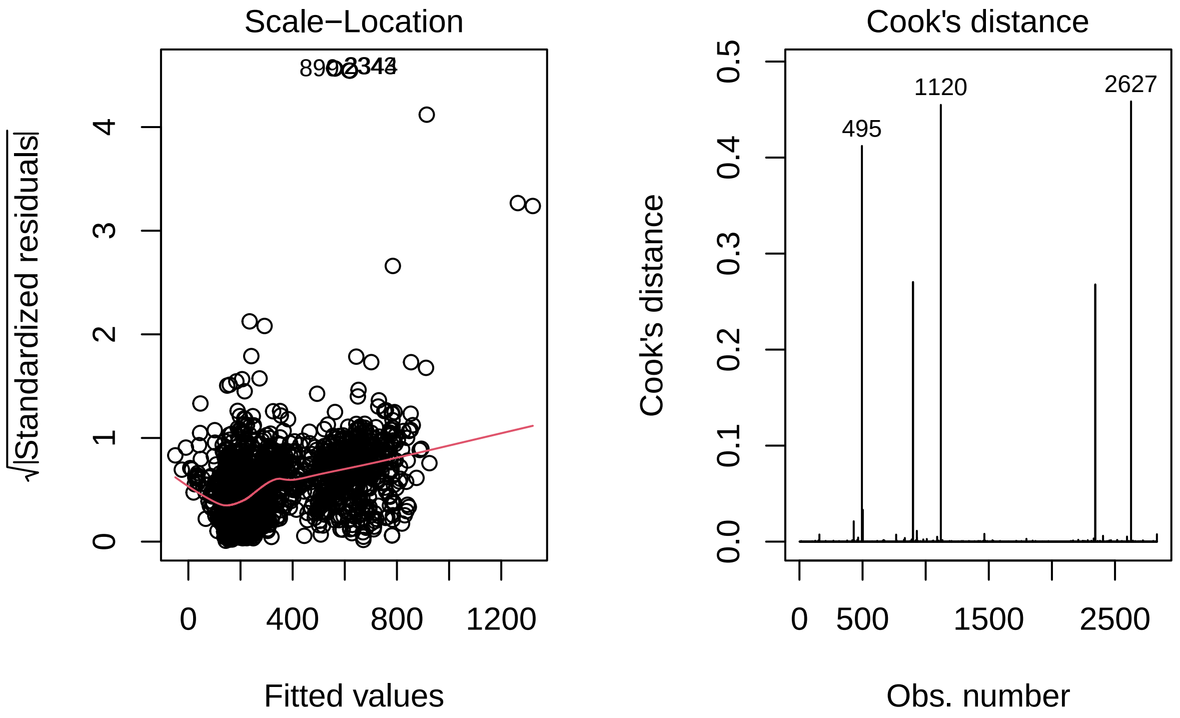
```
##
## Call:
## lm(formula = realSum ~ room_type + person_capacity + biz + bedrooms +
##     dist + rest_index_norm + host_is_superhost + guest_satisfaction_overall +
##     day + host_is_superhost:guest_satisfaction_overall, data = barc)
##
## Residuals:
##    Min      1Q Median      3Q    Max
## -568.8  -77.9  -22.4   40.7 6384.8
##
## Coefficients:
##                                                     Estimate Std. Error t value
## (Intercept)                                         174.9786    81.1283   2.157
## room_typePrivate room                             -315.2687    21.7729 -14.480
## room_typeShared room                              -431.5785    91.0570  -4.740
## person_capacity                                    -13.6567     8.6062  -1.587
## biz1                                                76.2294    13.9776   5.454
## bedrooms                                           146.4871    16.0731   9.114
## dist                                                20.0226     7.5669   2.646
## rest_index_norm                                      3.2765     1.0087   3.248
## host_is_superhostTrue                            -1378.7852   479.6076  -2.875
## guest_satisfaction_overall                           0.8995     0.7173   1.254
## dayWeekend                                          57.4507    11.7243   4.900
## host_is_superhostTrue:guest_satisfaction_overall    14.3735     4.9779   2.887
##                                                  Pr(>|t|)    
## (Intercept)                                       0.03110 *  
## room_typePrivate room                             < 2e-16 ***
## room_typeShared room                             2.25e-06 ***
## person_capacity                                   0.11266    
## biz1                                             5.36e-08 ***
## bedrooms                                          < 2e-16 ***
## dist                                              0.00819 ** 
## rest_index_norm                                   0.00117 ** 
## host_is_superhostTrue                             0.00407 ** 
## guest_satisfaction_overall                        0.20993    
## dayWeekend                                       1.01e-06 ***
## host_is_superhostTrue:guest_satisfaction_overall  0.00391 ** 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 307.4 on 2821 degrees of freedom
## Multiple R-squared:  0.2552, Adjusted R-squared:  0.2523 
## F-statistic: 87.89 on 11 and 2821 DF,  p-value: < 2.2e-16
```

- A look at the summary shows that the `day` factor is highly significant, suggesting that there is a difference between weekday and weekend pricing.

Look at the model diagnostics:

```r
par(mfrow = c(1,2))
plot(lm_bar_step, 1:4)
```


Residuals vs Fitted
Residuals
6000
4000
2000
0
8992
2343
2344
0
400
800
1200
Fitted values
Normal Q−Q
Standardized residuals
20
15
10
5
0
2343
8992
2344
−3
−1
1
2
3
Theoretical Quantiles



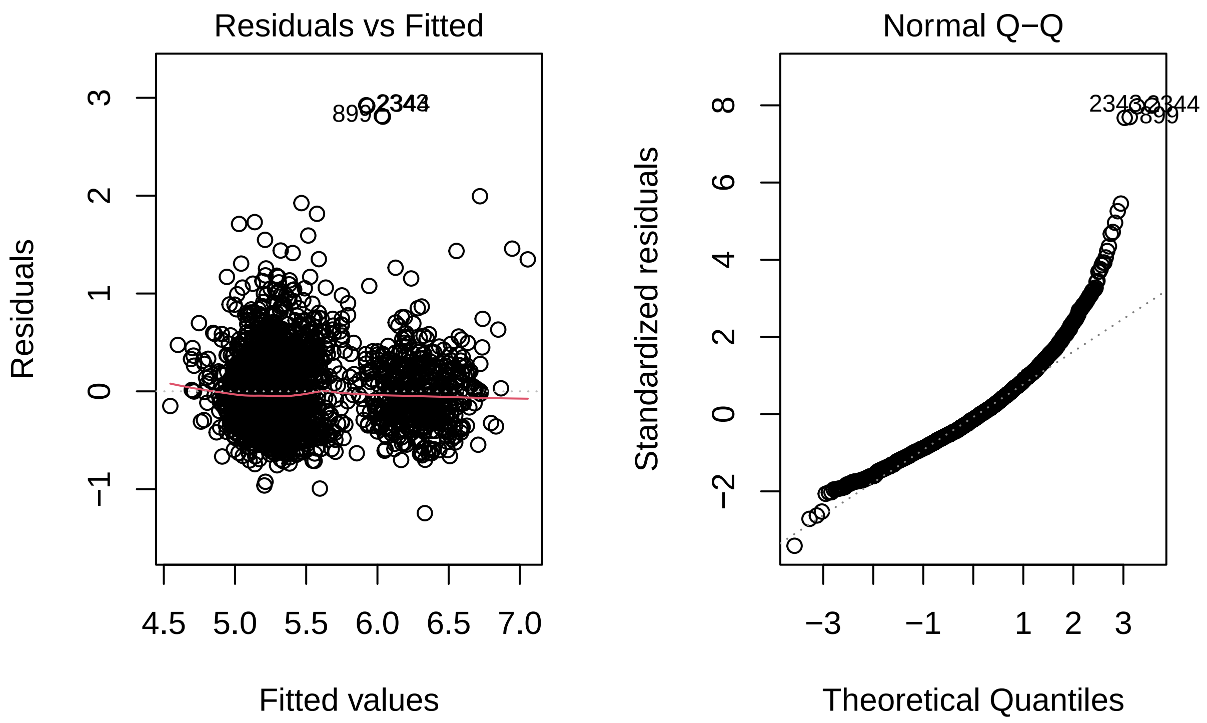
- Residual vs fitted: Looks OK, the points are somewhat evenly distributed above and below 0. There are some observations with especially high residuals, and in general the residuals are on quite a large scale. There isn't evidence of non-linearity.

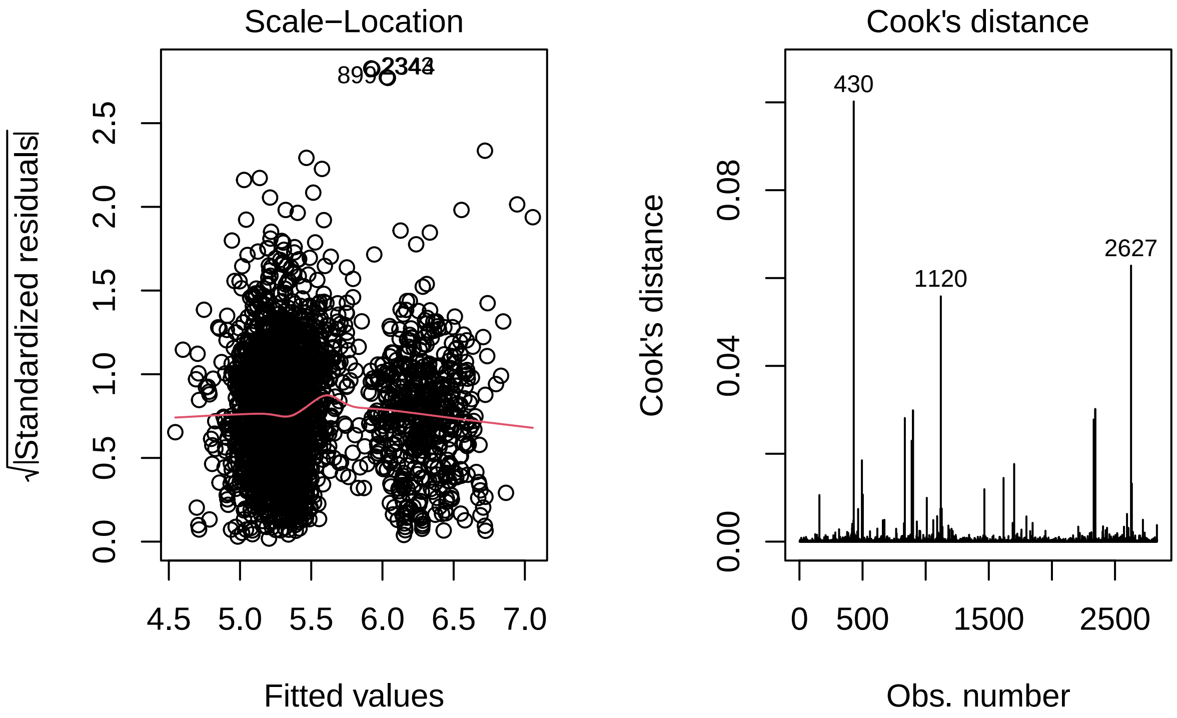
- Normal Q-Q: shows the data is right-skewed, this is be expected as the distribution of price is right skewed.
- Scale-location: there is some evidence of variance increasing with the mean. The points also show some clustering, which is to be expected from the observations made when examining the relationship between room type and price.
- Cooks distance: there are some outliers that have larger cooks distance, refitting the model without these observations doesn't change the model that much, so include them for now.

A check of partial residual plots of each variables against the residuals does not suggest that any new transformations or interactions should be made to the explanatory variable.

Fit the model with the response logged as the distribution of price is right skewed and examine if this helps rectify some of the violations in the modelling assumptions.

```
lm_bar_log <- lm(log(realSum+0.1) ~ room_type + biz + bedrooms + dist +
                rest_index_norm + host_is_superhost*guest_satisfaction_overall
                + day,
                data = barc)
par(mfrow = c(1,2))
plot(lm_bar_log,1:4)
```





- The residuals vs fitted and scale location look better for this model. The Q-Q plot still shows some violation of the normality but looks better than before.

Look at the model summary:

```
summary(lm_bar_log)
```

```
##
## Call:
## lm(formula = log(realSum + 0.1) ~ room_type + biz + bedrooms +
##     dist + rest_index_norm + host_is_superhost * guest_satisfaction_overall +
##     day, data = barc)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.24469 -0.23661 -0.05019  0.18241  2.92445
##
## Coefficients:
##                                                 Estimate Std. Error t value
## (Intercept)                                    5.4135159  0.0932900  58.029
## room_typePrivate room                         -0.7790264  0.0225810 -34.499
## room_typeShared room                          -1.1984485  0.1083963 -11.056
## biz1                                              0.1684753  0.0165514  10.179
## bedrooms                                          0.2143395  0.0161328  13.286
## dist                                             -0.0483898  0.0090270  -5.361
## rest_index_norm                                   0.0043946  0.0012028   3.654
## host_is_superhostTrue                            -2.9726601  0.5723539  -5.194
## guest_satisfaction_overall                        0.0037194  0.0008556   4.347
## dayWeekend                                        0.1096736  0.0139791   7.846
## host_is_superhostTrue:guest_satisfaction_overall  0.0316500  0.0059405   5.328
##                                                  Pr(>|t|)
## (Intercept)                                       < 2e-16 ***
## room_typePrivate room                             < 2e-16 ***
## room_typeShared room                              < 2e-16 ***
## biz1                                              < 2e-16 ***
## bedrooms                                          < 2e-16 ***
## dist                                             8.97e-08 ***
## rest_index_norm                                  0.000263 ***
## host_is_superhostTrue                            2.21e-07 ***
## guest_satisfaction_overall                       1.43e-05 ***
## dayWeekend                                       6.07e-15 ***
## host_is_superhostTrue:guest_satisfaction_overall 1.07e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3668 on 2822 degrees of freedom
## Multiple R-squared:  0.5846, Adjusted R-squared:  0.5831
## F-statistic: 397.1 on 10 and 2822 DF,  p-value: < 2.2e-16
```

To study the difference in price between room prices in Barcelona on weekends and weekdays, look at the `dayWeekend` coefficient. This corresponds to the increase in the log of the room price on weekends compared to weekdays. Its exponential is therefore the multiplicative increase in price on weekends.

```
exp(lm_bar_log$coefficients["dayWeekend"])
```

```
## dayWeekend
##   1.115914
```

- A typical BnB room in Barcelona's price will increase by around 11.6% on days when controlling for other factors.

What variability is there around this estimate?

Construct a 95% confidence interval of the price difference:

```
sapply(confint(lm_bar_log)["dayWeekend",], exp)
```

```
##    2.5 %   97.5 %
## 1.085742 1.146924
```

- This means that whilst controlling for other factors that influence price, the model suggests that a typical BnB in Barcelona increases by around 8.6-14.7%.

Task 6: *Fit a GLM to the listed room price on weekdays in Barcelona. Use this model to predict the listed room prices for Barcelona on weekends. Calculate the prediction error and the cross validation error (perform 10-fold cross validation). Comment on your findings.* [5 marks]

Proceed with the previous model (without day factor now):

```
lm_barwday <- glm(log(realSum+0.1) ~ room_type + biz + bedrooms + dist +
             rest_index_norm + lat +
             host_is_superhost*guest_satisfaction_overall,
           data = bar_wday)
```

- The diagnostic plots look very similar to the model fitted to the model to the full dataset, so won't be shown here.

Now make predictions on the Barcelona weekend dataset.

```
pred_price_barday <- predict(lm_barwday, newdata = bar_wend)
head(pred_price_barday, 3)
```

```
##        1        2        3
## 6.136299 4.964657 5.280983
```

Note: room price has been logged, hence the predictions for the actual room prices are:

```
head(exp(pred_price_barday),3)
```

```
##        1        2        3
## 462.3393 143.2594 196.5630
```

Now calculate the prediction and cross-validation error, since the response in the model is logged, transform back to the original scale.

Calculate the prediction error on the training dataset (weekdays):

```
err.train <- mean((bar_wday$realSum - exp(lm_barwday$fitted.values))^2)
err.train
```

```
## [1] 76671.93
```

```
sqrt(err.train)
```

```
## [1] 276.897
```

And also the prediction error on the test dataset (weekends):

```
err.test <- mean((bar_wend$realSum - exp(pred_price_barday))^2)
err.test
```

```
## [1] 125952.2
```

```
sqrt(err.test)
```

```
## [1] 354.8974
```

The cross validation on the training dataset using K = 10 folds is:

```
cost <- function(obs, fit) mean( (obs - exp(fit))^2 )
err.cv <- cv.glm(data = bar_wday, cost, glmfit = lm_barwday, K = 10)$delta[1]
err.cv
```

```
## [1] 89108.56
```

```
sqrt(err.cv)
```

```
## [1] 298.5106
```

- The prediction error on the test dataset is a lot larger than the training dataset, and the cross-validation error is only slightly larger than the prediction error on the training dataset. This can be explained by our findings in task 6 - we found that there was quite a substantial increase in price on weekends, so it is expected that the model underestimates room price on the weekend dataset. In the cross-validation error, only weekday data is being used, so no such problem occurs and its thus its magnitude is more comparable to the prediction error on the weekday data.

- These errors are quite high in magnitude, and their square roots give a notion of the average ‘distance’ from each point. Hence our model may not be able to predict all AirBnB prices with a good degree of accuracy. Experiments with a Gamma glm did not improve much either.

Is the model completely useless? Not entirely - if we restrict the data to discount higher prices, e.g. all those over, say, €1000 and recalculate the prediction errors, they are a lot lower. For example the training error is

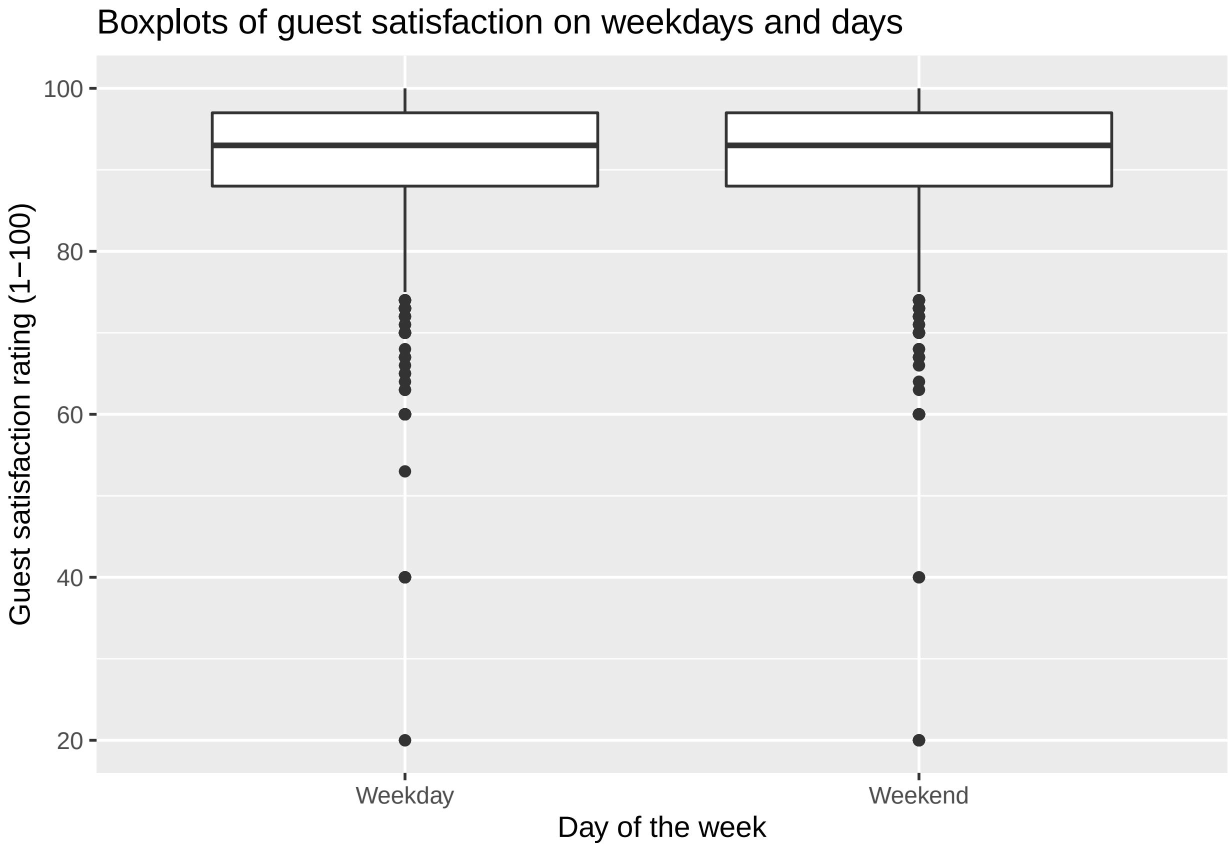
```
bar_wday1000 <- bar_wday[-which(bar_wday$realSum > 1000),]
pred_barwday1000 <- lm_barwday$fitted.values[-which(bar_wday$realSum > 1000)]
sqrt( mean( (bar_wday1000$realSum - exp(pred_barwday1000))^2 ) )
```

```
## [1] 97.08066
```

So the model is OK at predicting prices in the €0 - €1000 range (this error is still a bit high), the larger prediction error on the full data is partly due to the model’s poor performance in predicting higher valued listings.

Task 7: *Use plots or a statistical test to comment on whether the guest satisfaction varies between the weekdays and the weekends in Barcelona. Further, define a GLM that may be used to predict guest satisfaction.* [6 marks]

```
ggplot(data = barc, aes(x = day, y = guest_satisfaction_overall)) +
  geom_boxplot() +
  labs(title = "Boxplots of guest satisfaction on weekdays and days",
       x = "Day of the week",
       y = "Guest satisfaction rating (1-100)")
```



- The distributions of guest satisfaction looks almost identical, therefore conclude that it does not vary between weekdays and weekends in Barcelona.

Guest satisfaction is a count variable, so fit a poisson GLM.

In Task 2, a strong linear relationship between cleanliness rating and guest satisfaction was found. It was also found that superhost status had an influence on guest satisfaction motivating inclusion of these variables in the model. Use the previously created dataset `barc` containing both weekend and weekday entries.

Since the poisson distribution is right skewed, and guest satisfaction is left skewed, instead model `100 - guest_satisfaction_overall`. This quantity will still be on the scale of 1-100, but is now also right skewed. To derive the predicted guest satisfaction score, simply calculate 100 - predicted value.

```
guest_full <- glm(100 - guest_satisfaction_overall ~ realSum + room_type +
                    person_capacity + host_is_superhost + cleanliness_rating +
                    multi + biz + bedrooms + dist + metro_dist +
                    attr_index_norm + rest_index_norm + lng + lat + day,
                  family = poisson,
                  data = barc)
```

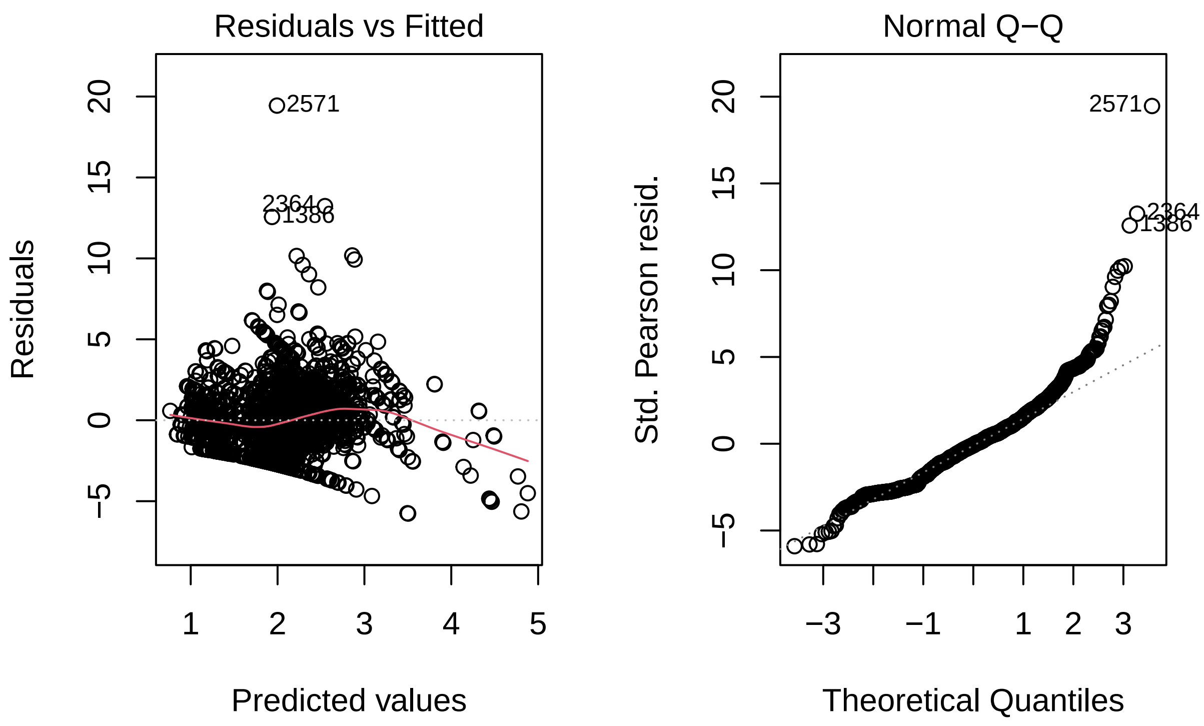
Use the step function to reduce the model:

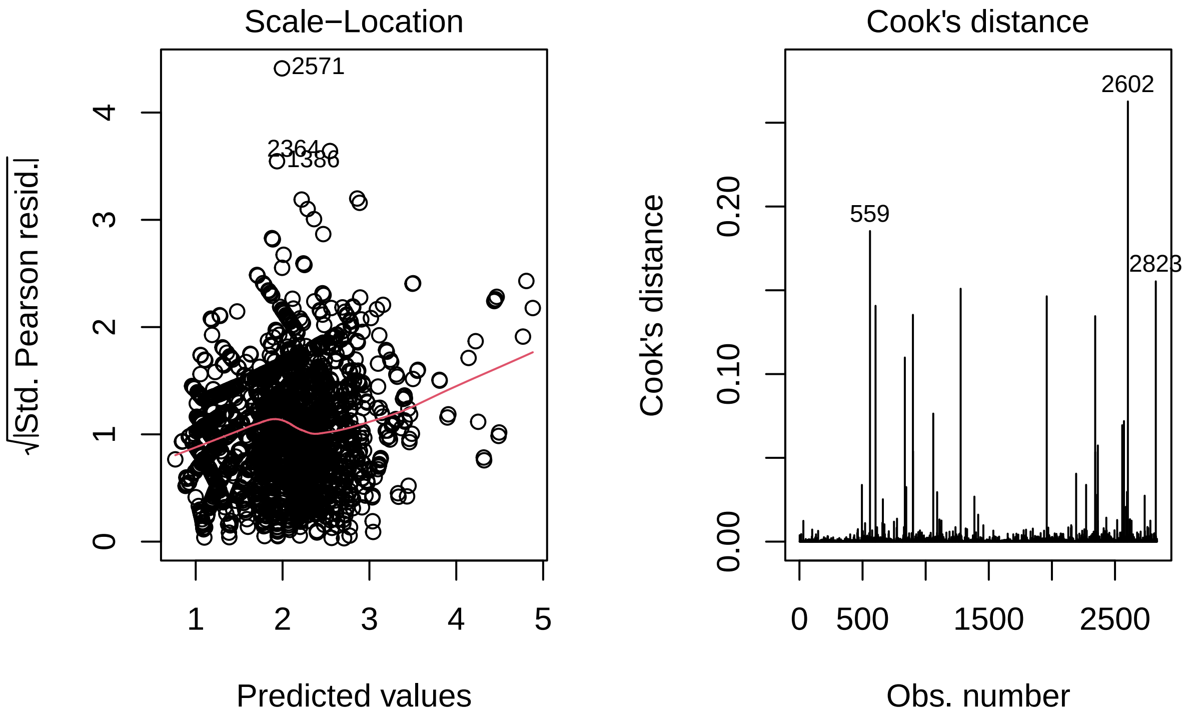
```
guest_step <- step(guest_full, trace = 0,
                   score = list(lower  =
                                  ~ cleanliness_rating + host_is_superhost))
summary(guest_step)
```

```
##
## Call:
## glm(formula = 100 - guest_satisfaction_overall ~ realSum + room_type +
##     host_is_superhost + cleanliness_rating + multi + biz + metro_dist +
##     attr_index_norm + rest_index_norm + lng + lat, family = poisson,
##     data = barc)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -8.1514  -1.2554  -0.0828   0.8843  12.1166
##
## Coefficients:
##                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)           -7.118e+01  2.221e+01  -3.205 0.001351 **
## realSum               -7.263e-05  2.167e-05  -3.352 0.000803 ***
## room_typePrivate room -1.128e-01  1.865e-02  -6.051 1.44e-09 ***
## room_typeShared room  -1.026e-01  9.070e-02  -1.132 0.257782
## host_is_superhostTrue -6.989e-01  2.472e-02 -28.278  < 2e-16 ***
## cleanliness_rating    -3.398e-01  3.532e-03 -96.196  < 2e-16 ***
## multi1                 1.454e-01  1.673e-02   8.691  < 2e-16 ***
## biz1                   2.896e-01  1.784e-02  16.235  < 2e-16 ***
## metro_dist            -1.363e-01  2.883e-02  -4.729 2.26e-06 ***
## attr_index_norm       -4.148e-03  1.060e-03  -3.914 9.07e-05 ***
## rest_index_norm        2.545e-03  1.069e-03   2.381 0.017271 *
## lng                   -1.562e+00  3.894e-01  -4.011 6.04e-05 ***
## lat                    1.931e+00  5.438e-01   3.551 0.000384 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 20961  on 2832  degrees of freedom
## Residual deviance: 11444  on 2820  degrees of freedom
## AIC: 20963
##
## Number of Fisher Scoring iterations: 5
```

Take a look at the diagnostics:

```r
par(mfrow = c(1,2))
plot(guest_step, 1:4)
```





- Residuals vs fitted shows heteroscedasticity, the variance is increasing with the mean.
- Normal Q-Q looks OK, the left tail of the distribution looks good (this is actually better than when trying to model guest satisfaction instead). The right tail shows some departure from the poisson distribution.
- Scale location again shows heteroscedasticity. The fact that there is some pattern at low predicted values is due to the fact that we are using count data and so is not cause for concern.
- Cooks distance indicates a few outliers. Attempting fits without the outliers do not change the model much, so continue with them included.

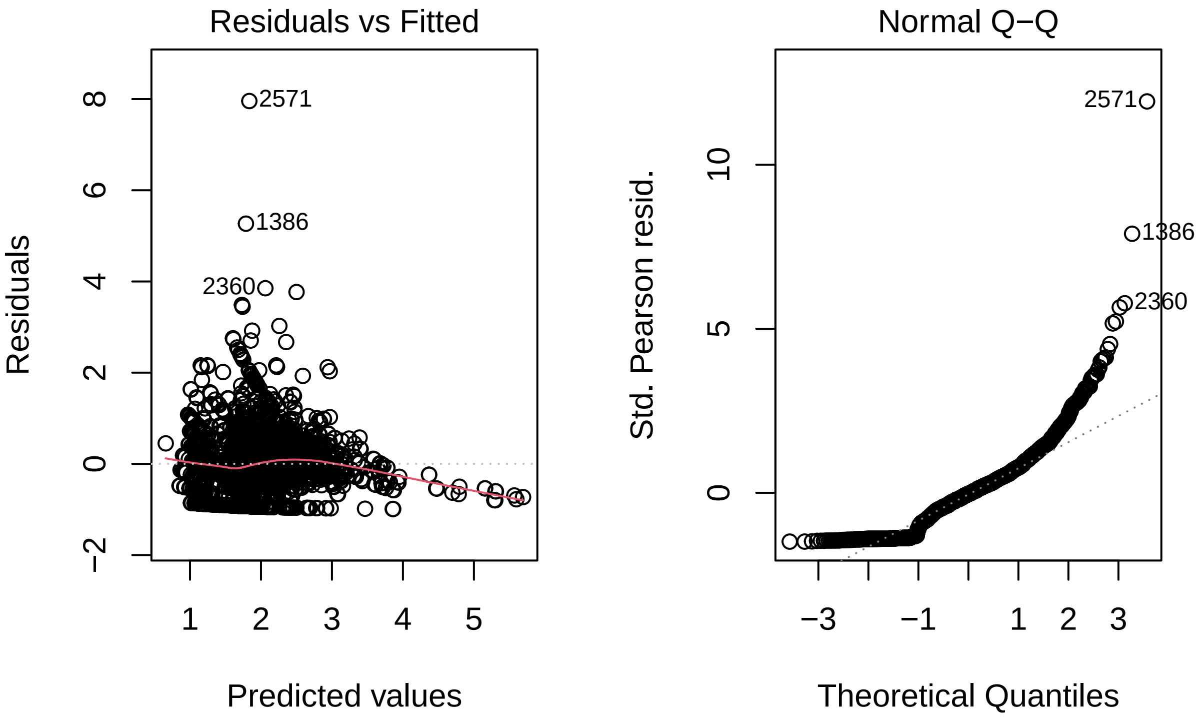
Fitting a quasipoisson model with the same parameters shows there is also overdispersion (the dispersion parameter is estimated to be around 3.5), try fitting a negative binomial model to account for this:

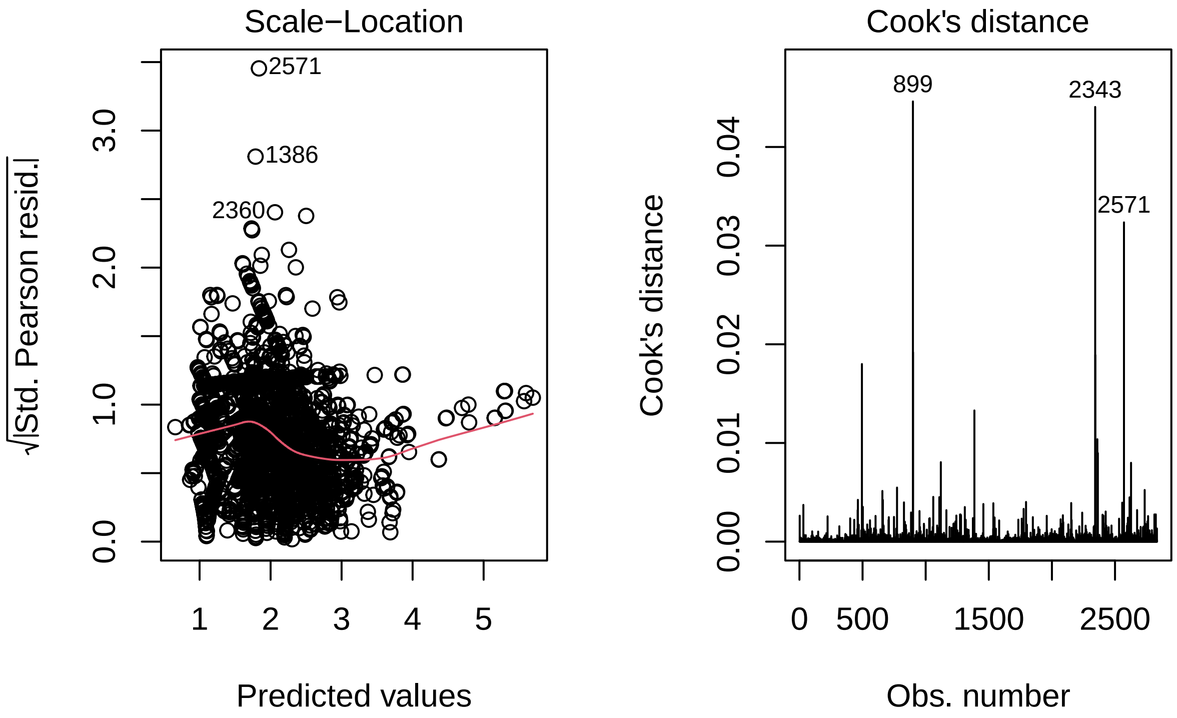
```
guest_nb <- glm(formula = (100 - guest_satisfaction_overall) ~ realSum +
                    room_type + host_is_superhost + cleanliness_rating + multi +
                    biz + metro_dist + attr_index_norm + rest_index_norm + lng +
                    lat, family = negative.binomial(theta = 1),
    data = barc)

summary(guest_nb)
```

```
## 
## Call:
## glm(formula = (100 - guest_satisfaction_overall) ~ realSum + 
##     room_type + host_is_superhost + cleanliness_rating + multi + 
##     biz + metro_dist + attr_index_norm + rest_index_norm + lng + 
##     lat, family = negative.binomial(theta = 1), data = barc)
## 
## Deviance Residuals: 
##     Min       1Q   Median       3Q      Max  
## -2.7870  -0.4588  -0.0220   0.2950   3.3938  
## 
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)    
## (Intercept)             -2.076e+01  4.663e+01  -0.445 0.656111    
## realSum                 -8.091e-05  4.305e-05  -1.880 0.060278 .  
## room_typePrivate room -1.257e-01  4.095e-02  -3.070 0.002160 ** 
## room_typeShared room  -1.227e-01  2.080e-01  -0.590 0.555262    
## host_is_superhostTrue -5.923e-01  3.789e-02 -15.633  < 2e-16 ***
## cleanliness_rating    -4.570e-01  1.322e-02 -34.567  < 2e-16 ***
## multi1                 1.183e-01  3.353e-02   3.529 0.000424 ***
## biz1                   2.530e-01  3.738e-02   6.769 1.57e-11 ***
## metro_dist            -1.435e-01  5.765e-02  -2.489 0.012850 *  
## attr_index_norm       -4.012e-03  2.089e-03  -1.921 0.054890 .  
## rest_index_norm        1.915e-03  2.218e-03   0.864 0.387903    
## lng                   -9.447e-01  8.104e-01  -1.166 0.243812    
## lat                    7.069e-01  1.142e+00   0.619 0.535960    
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
## 
## (Dispersion parameter for Negative Binomial(1) family taken to be 0.44614)
## 
##     Null deviance: 3158.4  on 2832  degrees of freedom
## Residual deviance: 2238.7  on 2820  degrees of freedom
## AIC: 17460
## 
## Number of Fisher Scoring iterations: 9
```

```
par(mfrow = c(1,2))
plot(guest_nb, 1:4)
```


Residuals vs Fitted
Residuals
Predicted values
-2
0
2
4
6
8
1
2
3
4
5
2571
1386
2360
Normal Q−Q
Std. Pearson resid.
Theoretical Quantiles
0
5
10
−3
−1
1
2
3
2571
1386
2360



- The negative binomial model diagnostics do look a little better, in particular residuals look to be smaller in magnitude, giving the model better predictive performance. The heteroscedasticity is still present, but is a quite a bit less pronounced.

Settle on this model as the glm we can use to predict guest satisfaction.

Task 8: *Predict the London weekend prices for different room types based on the weekends price model for Barcelona. Calculate the prediction error and comment on what you observe.* [5 marks]

Recall the price model for Barcelona and fit this to the weekend data in Barcelona. We drop the latitude term, as the values for this will be different in London.

```
lm_barwend <- lm(log(realSum+0.1) ~ room_type + biz + bedrooms +
             dist + rest_index_norm  +
             host_is_superhost*guest_satisfaction_overall,
           data = bar_wend)
```

- Again, diagnostic plots are very similar to the log model in Task 5.

Now use this model to predict weekend prices for different room types. Make predictions for each observation in the London weekend data. Average over each room type to get obtain a predicted price for different room types.

```
pred_price_lonwend <- predict(lm_barwend, newdata = lon_wend)
head(exp(pred_price_lonwend))   # since this is a log model, must exponentiate
```

```
##        1        2        3        4        5        6
## 136.9907 159.9681 161.4351 188.8318 324.4783 358.9928
```

```
cbind.data.frame(room_type = lon_wend$room_type, # label each prediction with
  prediction = exp(pred_price_lonwend)) %>%      # room type
  group_by(room_type) %>%
  summarise(price = mean(prediction))            # average of each room type
```

```
## # A tibble: 3 x 2
##   room_type       price
##   <chr>           <dbl>
## 1 Entire home/apt  441.
## 2 Private room     169.
## 3 Shared room      126.
```

- Looking back at the density plot for London in Task 3, these look like quite reasonable predictions.

Calculate the prediction error:

```
err.lon <- mean( (lon_wend$realSum - exp(pred_price_lonwend))^2 )
err.lon
```

```
## [1] 152288.1
```

```
sqrt(err.lon)
```

```
## [1] 390.2411
```

The prediction error is quite high, so whilst the average across each room price looks very reasonable, the individual predictions aren't always great. This is expected as the same model didn't perform well in predicting higher priced listings before on the full Barcelona dataset.

Task 9: *Provide a non-scientific summary of your analysis in Task 5 (300 words maximum).* [4 marks]

The price of an AirBnB is influenced by a range of different factors [1] : guest capacity, whether the listing is a private room or an entire accommodation, and its proximity to the city centre to name a few. Our analysis focused on the question of whether there was a difference in price between weekdays and weekends.

The data studied contains data from AirBnBs in Barcelona and is a subset of the data investigated by Gyódi, K., & Nawaro, Ł. (2021).[2] A generalised linear model was fitted to this data to predict each AirBnB's listing price based on relevant variables in the data, including a variable that determined whether the listing was for a weekday or for the weekend.

It was found that the variable indicating whether the listing was on a weekday or a weekend was significant, suggesting that it does indeed have a role in predicting price. We further estimated that the increase in listing price on a weekend is within the range of 8.6% - 14.7% after controlling for other factors such as guest satisfaction or room type that were found to explain price; with our specific model's estimate being 11.6%.

It is worth mentioning that this is specific to AirBnBs in Barcelona; whether the same magnitude of change is seen in other locations is an avenue for future work. There may also be additional factors not present in the data that need consideration, for example recently listed BnB's may charge a lower price to attract initial customers.

1. Toader, V., Negrușa, A.L., Bode, O.R. and Rus, R.V., 2022. Analysis of price determinants in the case of Airbnb listings. *Economic Research-Ekonomska Istraživanja, 35(1)*, pp.2493-2509.

2. Gyódi, K. and Nawaro, Ł., 2021. Determinants of Airbnb prices in European cities: A spatial econometrics approach. *Tourism Management*, 86, p.104319.