Assignment: Regression Analysis and Machine Learning

Syed Furquan Bokhari

Q1: Regression output:

```
if(!require(pacman)){
  install.packages("pacman")
}

pacman::p_load(tidyverse, janitor, tidymodels, randomForest, rpart,
               rpart.plot, gam, caTools, e1071, class, GGally, ggthemes)


theme_set(theme_minimal())
```

This question uses the IMDB movies ratings data from class 13 (with some observations removed at random). Below is R output from a regression of IMDbScore on Budget, DirectorFacebookLikes, CastFacebookLikes, AvgDirectorScore, and AvgActorScore. Use this output to answer the questions below.

a. What is the SE on DirectorFacebookLikes?

The t-value is the estimate divided by standard error (SE).

In this case:

```
0.489321/5.73
```

```
[1] 0.08539634
```

```
Call:
lm(formula = IMDbScore ~ Budget + DirectorFacebookLikes + CastTotalFacebookLikes +
    AvgDirectorScore + AvgActorScore, data = Movies)

Residuals:
    Min      1Q  Median      3Q     Max
-2.5690 -0.4149  0.0171  0.4348  1.6878

Coefficients:
                      Estimate Std. Error t value    Pr(>|t|)
(Intercept)           1.584016   0.870113    1.82     0.06975 .
Budget               -0.000288   0.000746   -0.39
DirectorFacebookLikes  0.489321                5.73 0.000000026 ***
CastTotalFacebookLikes 0.009656   0.022158    0.44     0.66333
AvgDirectorScore       0.286065   0.067569          0.000031209 ***
AvgActorScore          0.456769   0.120003    3.81     0.00017 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.692 on 280 degrees of freedom
Multiple R-squared:  0.307,     Adjusted R-squared:  0.295
F-statistic: 24.8 on 5 and 280 DF,  p-value: <2e-16
```

Figure 1: Regression Output

b. What is the t-value on AvgDirectorScore?

Again, the t-value is the estimate divided by standard error (SE).

```
0.286065/0.067569
```

```
[1] 4.233672
```

c. What is the p-value on Budget?

This is a 2 tailed test, hence we multiply the probability by two and toggle the `lower.tail` argument to false.

```
2 * pt(0.39, df = 280, lower.tail = FALSE)
```

```
[1] 0.696833
```

Q2: A/B Testing

The file AdSmartAB.csv contains data from an experiment done by an advertising agency in which users were randomized into either being shown a particular ad, or not, and then their behavior was tracked and recorded. The data has 8077 rows and 9 columns. The only two columns you need to know about are:

- experiment: Factor with two values: "control" & "exposed". "exposed" means user was shown the ad. "control" means that user was not shown the ad.
- yes: 0-1 variable that equals 1 if the user pressed the "Yes" button after seeing the ad, and 0 otherwise

Is there strong evidence in this data that seeing the ad led more users to click "Yes"? What is the p-value from your hypothesis test? In words, what does the p-value mean in this test? What do you conclude?

Given that we have two groups, that are categorical then the he chi-square test of independence will suffice. We start by reading the data and preview the first few rows of the data.

```
ads <- read_csv("AdSmartAB(1).csv", show_col_types = FALSE)

head(ads)
```

```
# A tibble: 6 x 9
  auction_id             exper~1 date   hour devic~2 platf~3 browser   yes    no
  <chr>                  <chr>   <chr> <dbl> <chr>     <dbl> <chr>   <dbl> <dbl>
1 0008ef63-77a7-448b-bd~ exposed 7/10~     8 Generi~       6 Chrome~     0     0
2 000eabc5-17ce-4137-8e~ exposed 7/7/~    10 Generi~       6 Chrome~     0     0
3 0016d14a-ae18-4a02-a2~ exposed 7/5/~     2 E5823         6 Chrome~     0     1
4 00187412-2932-4542-a8~ control 7/3/~    15 Samsun~       6 Facebo~     0     0
5 001a7785-d3fe-4e11-a3~ control 7/3/~    15 Generi~       6 Chrome~     0     0
6 0027ce48-d3c6-4935-bb~ control 7/3/~    15 Samsun~       6 Facebo~     0     0
# ... with abbreviated variable names 1: experiment, 2: device_make,
#   3: platform_os
```

We then do a chi-square test.

```
chisq.test(ads$yes, ads$experiment)
```

```
    Pearson's Chi-squared test with Yates' continuity correction
```

```
data:  ads$yes and ads$experiment
X-squared = 4.2639, df = 1, p-value = 0.03893
```

At 1% confidence level, the output means that showing the advert has no effect on whether or not the users click "Yes". This observation means that adverts have no effect on consumer choice. However, at 5% and beyond, we can safely conclude that adverts do affect the likelihood that a consumer clicks "Yes".

Q3: Earnings regression:

The nls.csv dataset contains 929 rows where each row is a worker in the US. The data has the following columns:

- luwe = log weekly wages
- educ = years of eduction
- exper = job-market experience in years

Using this dataset, answer the following questions:

```
nls <- read_csv("nls(1).csv")
head(nls)
```

```
# A tibble: 6 x 3
   luwe  educ exper
  <dbl> <dbl> <dbl>
1  6.00    18    13
2  5.80    14    13
3  5.56    12    14
4  5.42    11    17
5  6.33    16    13
6  5.48    10    14
```

a. What are the average years of education and average years of job-market experience in this dataset? The average years of education are 13.4725511 while the average experience is 13.6243272 years.

```
mean(nls$educ, na.rm = TRUE)
```

```
[1] 13.47255
```

```
mean(nls$exper, na.rm = TRUE)
```

```
[1] 13.62433
```

b. Run a linear regression of luwe on exper. What is the interpretation of the coefficient on exper in terms of the relationship of job-market experience to weekly wages?

The model shows a negative relationship between wages and experience. Specifically, a unit increase in experience reduces wages by an average of 0.007115 USD, all else remaining the same. However, at 1% and 5% significance level, the relationship is not significant meaning that experience has no material impact on wages.

```
job_reg <- lm(luwe ~ exper, data = nls)
summary(job_reg)
```

```
Call:
lm(formula = luwe ~ exper, data = nls)

Residuals:
     Min       1Q   Median       3Q      Max
-1.90816 -0.27459  0.02052  0.27675  1.67271

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.042330   0.053663 112.598   <2e-16 ***
exper       -0.007115   0.003792  -1.876   0.0609 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4424 on 927 degrees of freedom
Multiple R-squared:  0.003784,  Adjusted R-squared:  0.002709
F-statistic: 3.521 on 1 and 927 DF,  p-value: 0.06092
```

c. Now run a linear regression of luwe on educ and exper. Provide a 95% confidence interval for the effect of exper on luwe computed from this regression.

We compute the confidence interval using the confint function in R.

```r
job_reg_ext <- lm(luwe ~ educ + exper, data = nls)
summary(job_reg_ext)
```

```
Call:
lm(formula = luwe ~ educ + exper, data = nls)

Residuals:
     Min       1Q   Median       3Q      Max
-1.69358 -0.25911  0.02664  0.27649  1.71365

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 4.397318   0.145631  30.195  < 2e-16 ***
educ        0.091112   0.007577  12.025  < 2e-16 ***
exper       0.023529   0.004353   5.406 8.21e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4117 on 926 degrees of freedom
Multiple R-squared:  0.1383,    Adjusted R-squared:  0.1365
F-statistic: 74.33 on 2 and 926 DF,  p-value: < 2.2e-16
```

```r
confint(job_reg_ext, level = 0.95)
```

```
                2.5 %     97.5 %
(Intercept) 4.11151249 4.68312393
educ        0.07624181 0.10598232
exper       0.01498666 0.03207057
```

d. What is the correlation of exper and educ in the data? Suggest an explanation for the direction of this correlation.

The correlation between education and experience is -0.5854988. This is a high level of correlation that could result in collinearity in the regression. Specifically, because education affects experience. The more years spent in school implies less years of work experience. Again, there is a circular relationship between wages, experience, and education that makes the model unstable.

```r
cor(nls$educ, nls$exper)
```

```
[1] -0.5854988
```

e. Explain the difference between the estimated coefficient on exper in the regression in (b) versus that in the regression in (c).

There is a problem of **collinearity** between education and experience. Specifically, collinearity results in unstable coefficients or coefficients that are non-intuitive that may even go against theory. Specifically, because education affects experience, there is a circular relationship between wages, experience, and education that makes the model unstable. Hence the change in the sign of the coefficient.

f. Create a new variable called expersq that equals the square of exper. Run a linear regression of luwe on educ, exper, and expersq. Test the null hypothesis that the true coefficient on expersq is zero. Report the p-value. Do you reject the null hypothesis at the 1% significance level?

At the 1% level, the coefficient of expersq is zero given the p-value of 0.03057 is greater than 0.01 (1%) significance level. This observation means that expersq is not a significant driver of wages.

```r
nls$expersq = nls$exper^2

final_reg <- lm(luwe ~ educ + exper + expersq, data = nls)

summary(final_reg)
```

```
Call:
lm(formula = luwe ~ educ + exper + expersq, data = nls)

Residuals:
     Min       1Q   Median       3Q      Max
-1.70143 -0.25933  0.02076  0.27497  1.68847

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.0306137  0.2231344  18.064  < 2e-16 ***
```

```
educ         0.0924668  0.0075879  12.186  < 2e-16 ***
exper        0.0769025  0.0250225   3.073  0.00218 **
expersq     -0.0018911  0.0008731  -2.166  0.03057 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4108 on 925 degrees of freedom
Multiple R-squared:  0.1427,    Adjusted R-squared:  0.1399
F-statistic: 51.31 on 3 and 925 DF,  p-value: < 2.2e-16
```

Q4: AirBnB

The dataset airbnb.csv contains data on airbnb listings in several major cities in the US. We will use this dataset to create a prediction model for the variable price. (NOTE: be sure to load in the data with the "stringsAsFactors=TRUE" option!)

There are 57129 rows and 19 columns in the data. The columns are:

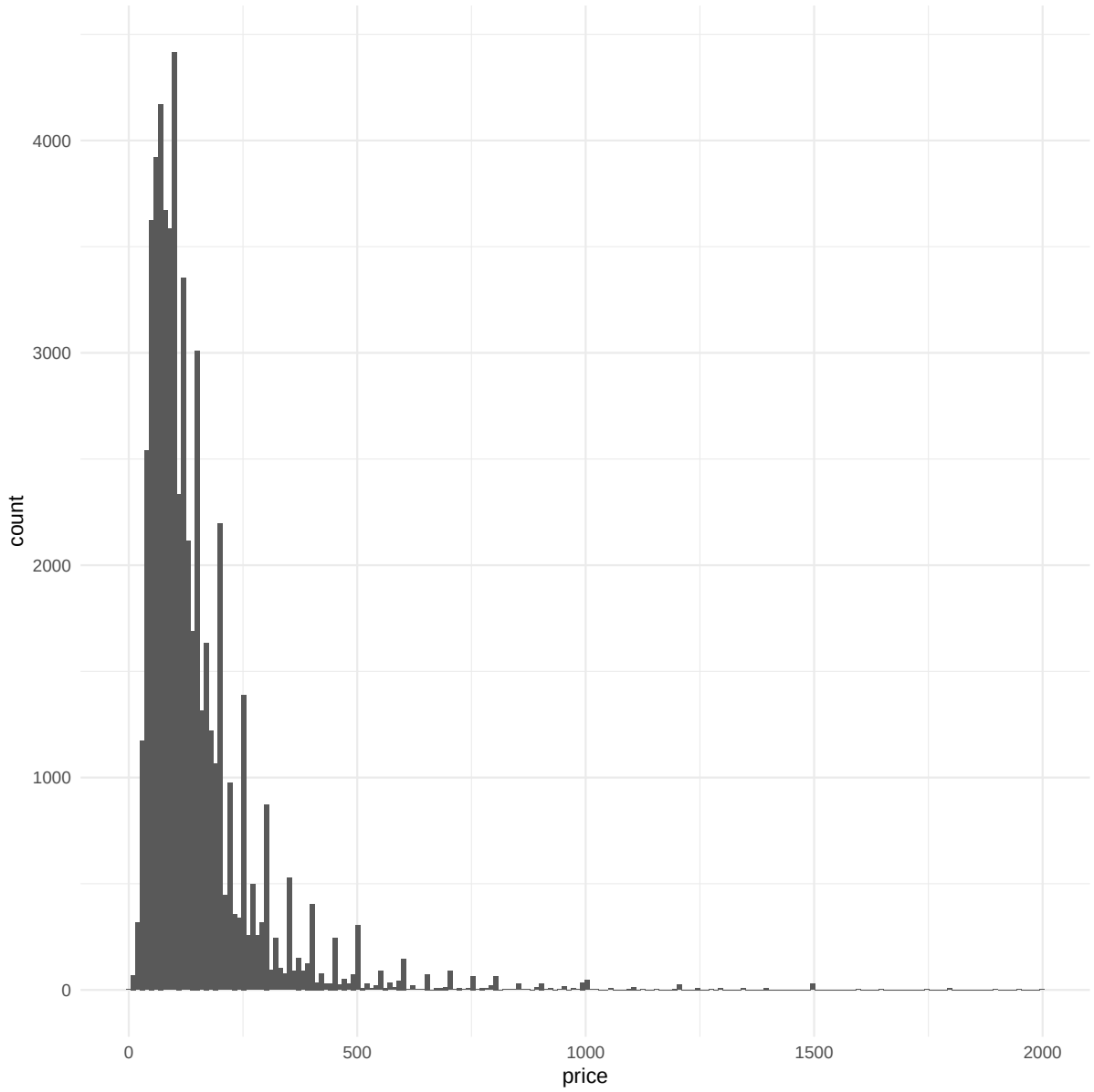
- price: price per night in dollars
- property type: Factor noting property type (e.g., "Apartment")
- room type: Factor noting room type (e.g., "Entire home/apt")
- accomodates: Number of people that the property accomodates
- bathrooms: Number of bathrooms
- bed type: Factor noting bed type (e.g. "Airbed")
- cancellation policy: Factor noting cancellation policy (e.g., "flexible")
- cleaning fee: True/False
- city: Factor noting the city (e.g. "Boston")
- host has profile pic: True/False/Unknown
- host identity verified: True/False/Unknown
- host response rate: Factor (e.g., "10%")
- instant bookable: True/False
- neighbourhood: Factor noting neighborhood (e.g., "16th Street Heights")
- number of reviews: Number of reviews
- review scores rating: Avg review rating (0-100)
- zipcode: Factor with 716 levels (e.g., "02108")

- bedrooms: Number of bedrooms
- beds: Number of beds

a. Plot a histogram of price with 200 bins. Does price look normally distributed? Why or why not?

The histogram is shown in figure 1 below.

```
air <- read_csv("airbnb(2).csv")

# ggplot(data = air, aes(x = price)) + geom_histogram(bins = 200)

air %>%
  ggplot(mapping = aes(x = price)) +
  geom_histogram(bins = 200)
```



b. The variable host response rate is stored incorrectly as a factor instead of a number (eg., "10%" or "100%" vs 10 or 100). Clean this variable so that host response rate is stored as a number from 0-100. What is the mean of the cleaned variable? For how many rows is host response rate missing?

In this case, I use the str_remove variable in the package `stringr` which is part of the tidyverse. The package has the functions `str_remove` and `str_remove_all` that take in an input and a pattern to be removed.

```r
## str_remove is from the tidyverse--stringr
air <- air %>%
  ## Remove the pattern %
  mutate(host_response_rate = str_remove(host_response_rate, "%"),
         ## Convert to numeric
         host_response_rate = as.numeric(host_response_rate))
head(air)
```

```
# A tibble: 6 x 19
  proper~1 room_~2 accom~3 bathr~4 bed_t~5 cance~6 clean~7 city  host_~8 host_~9
  <chr>    <chr>     <dbl>   <dbl> <chr>   <chr>   <lgl>   <chr> <lgl>   <lgl>
1 Apartme~ Entire~       3       1 Real B~ strict  TRUE    NYC   TRUE    TRUE
2 Apartme~ Entire~       7       1 Real B~ strict  TRUE    NYC   TRUE    FALSE
3 Apartme~ Entire~       5       1 Real B~ modera~ TRUE    NYC   TRUE    TRUE
4 Apartme~ Entire~       2       1 Real B~ modera~ TRUE    DC    TRUE    TRUE
5 Apartme~ Privat~       2       1 Real B~ strict  TRUE    SF    TRUE    TRUE
6 Apartme~ Entire~       3       1 Real B~ modera~ TRUE    LA    TRUE    FALSE
# ... with 9 more variables: host_response_rate <dbl>, instant_bookable <lgl>,
#   neighbourhood <chr>, number_of_reviews <dbl>, review_scores_rating <dbl>,
#   zipcode <dbl>, bedrooms <dbl>, beds <dbl>, price <dbl>, and abbreviated
#   variable names 1: property_type, 2: room_type, 3: accommodates,
#   4: bathrooms, 5: bed_type, 6: cancellation_policy, 7: cleaning_fee,
#   8: host_has_profile_pic, 9: host_identity_verified
```

c. Set your random seed to 2022 and then randomly partition the data into a 80% training data set and a 20% test data set. Compute the mean and standard deviation of the variable price in both the training set and the test set.

```r
set.seed(2022)

## I use tidymodels to split

air_split <- initial_split(air, prop = 0.8, strata = price)

air_training <- air_split %>% training() ## training(air_split)

air_testing <- air_split %>% testing() ## testing(air_split)
```

d. Run a regression ("Model 1") on the training data set to predict price using bedrooms, beds, bathrooms, number of reviews, review scores rating, and city and report the output. What is the test set RMSE for Model 1?

The RMSE on the test set is 158.1632.

```
air_train_model <- lm(price ~ bedrooms + beds + bathrooms + number_of_reviews + review_sco

summary(air_train_model)
```

```
Call:
lm(formula = price ~ bedrooms + beds + bathrooms + number_of_reviews +
    review_scores_rating + city, data = air_training)

Residuals:
    Min      1Q  Median      3Q     Max
-695.29  -54.13  -16.20   35.27 1895.07

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)         -104.68357    6.51310 -16.073  < 2e-16 ***
bedrooms              46.31638    0.89538  51.728  < 2e-16 ***
beds                  18.51697    0.57114  32.421  < 2e-16 ***
bathrooms             56.33219    1.10244  51.098  < 2e-16 ***
number_of_reviews     -0.11359    0.01225  -9.271  < 2e-16 ***
review_scores_rating   1.11793    0.06405  17.453  < 2e-16 ***
cityChicago          -46.84653    3.07461 -15.237  < 2e-16 ***
cityDC               -14.25452    2.92425  -4.875 1.09e-06 ***
cityLA               -22.72478    2.42670  -9.364  < 2e-16 ***
cityNYC               -5.30122    2.37178  -2.235   0.0254 *
citySF                41.97921    2.80801  14.950  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 106.6 on 45690 degrees of freedom
Multiple R-squared:  0.3773,    Adjusted R-squared:  0.3771
F-statistic:  2768 on 10 and 45690 DF,  p-value: < 2.2e-16
```

```
## RMSE- Root mean squared error ----
sqrt(mean((air$price - predict(air_train_model, newdata = air_testing))^2))
```

```
[1] 158.1632
```

e. What is the interpretation of the coefficient on bedrooms in Model 1?

The number of bedrooms are a significant driver of the price of housing. Specifically, all other variables held constant a unit increase in bedrooms raises the price of a house by 46.31638 USD on average.

f. What is the interpretation of the coefficient on "cityChicago" in Model 1?

The interpretation here is in reference to the city of Boston. Holding all other factors constant, a house in chicago is, on average 46.84653 USD cheaper than an equivalent house in Boston.

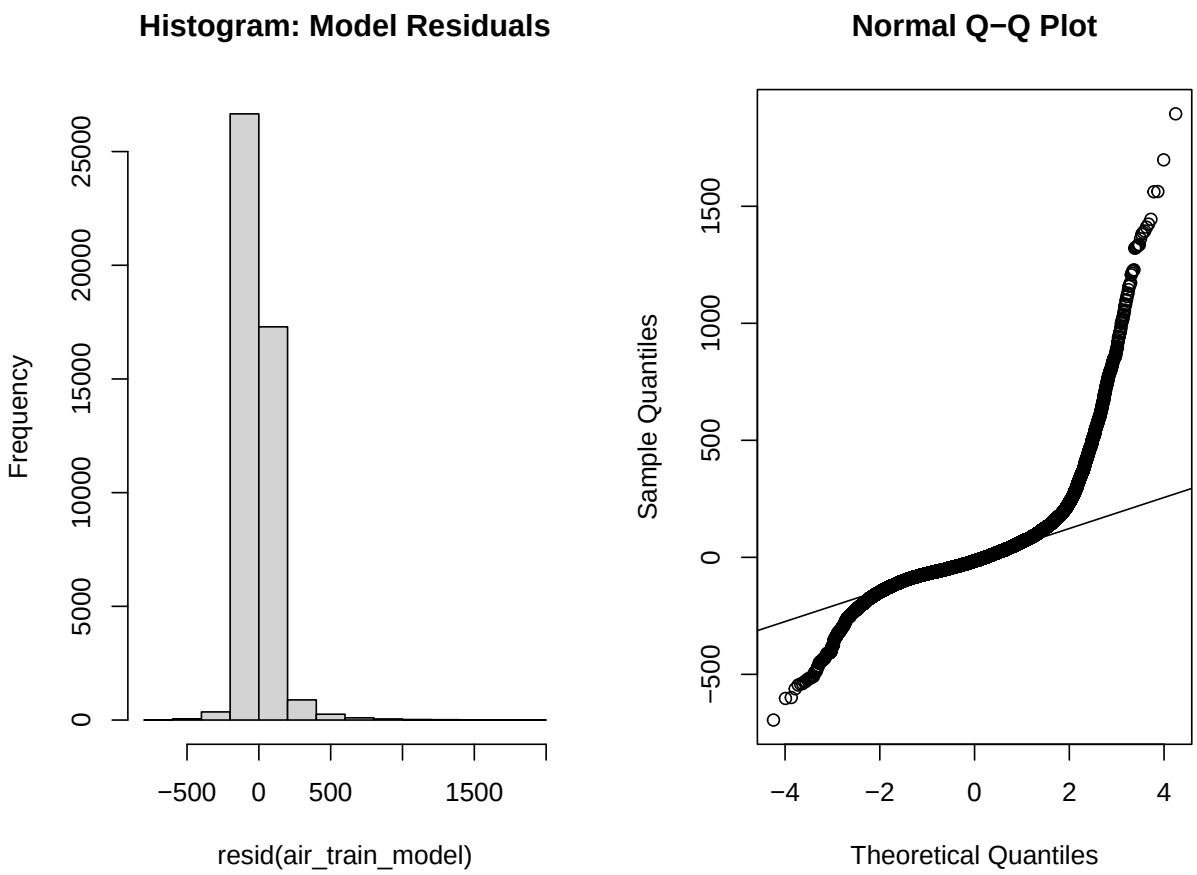
g. What is the interpretation of the coefficient on number of reviews in Model 1? Provide a possible explanation for why this coefficient has the sign that it does.

The number of reviews is inversely related to the price of a house. It is likely that houses with most reviews happen to have the most complaints from customers. Hence, more reviews could signal poor quality and hence the lower price.

h. Plot the residuals from Model 1 and comment on any patterns you see.

The model residuals deviate significantly from normality. This means that the model may not be very reliable for forecasting. Again, the outliers show a significant existence of outliers in the data that affect model performance.

```r
## R Markdown
## Quarto
par(mfrow = c(1, 2))
hist(resid(air_train_model), main = "Histogram: Model Residuals")

#create Q-Q plot for residuals
qqnorm(resid(air_train_model))

#add a straight diagonal line to the plot
qqline(resid(air_train_model))
```




```
par(mfrow = c(1, 1))
```

i. Try running the same regression as Model 1 using the log of price instead of price. Plot the residuals. Are the residuals from this model better, worse, or equally good/bad to those in part (h)?

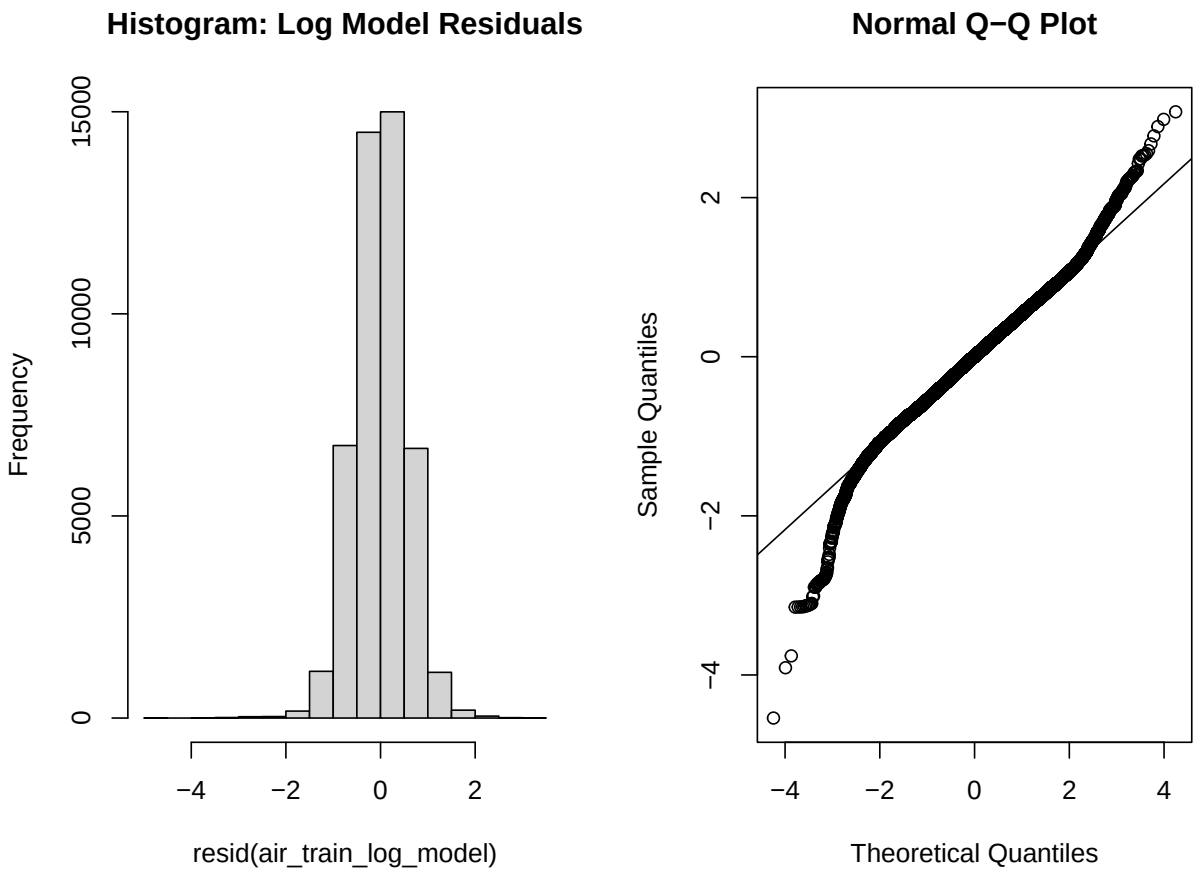
The residuals in this model are better than those of the model in part (h). This means that this model can better predict prices.

```
par(mfrow = c(1, 2))
air_train_log_model <- lm(log(price) ~ bedrooms + beds + bathrooms + number_of_reviews + r

summary(air_train_log_model)
```

```
Call:
```

```
lm(formula = log(price) ~ bedrooms + beds + bathrooms + number_of_reviews +
    review_scores_rating + city, data = air_training)

Residuals:
    Min      1Q  Median      3Q     Max
-4.5413 -0.3662  0.0059  0.3667  3.0779

Coefficients:
                       Estimate Std. Error t value Pr(>|t|)
(Intercept)           3.533e+00  3.392e-02 104.133  < 2e-16 ***
bedrooms              2.199e-01  4.664e-03  47.144  < 2e-16 ***
beds                  1.281e-01  2.975e-03  43.053  < 2e-16 ***
bathrooms             8.443e-02  5.742e-03  14.703  < 2e-16 ***
number_of_reviews    -2.065e-04  6.382e-05  -3.236  0.00121 **
review_scores_rating  7.669e-03  3.336e-04  22.986  < 2e-16 ***
cityChicago          -3.313e-01  1.601e-02 -20.688  < 2e-16 ***
cityDC               -8.443e-02  1.523e-02  -5.543 2.99e-08 ***
cityLA               -1.941e-01  1.264e-02 -15.359  < 2e-16 ***
cityNYC              -9.017e-02  1.235e-02  -7.299 2.94e-13 ***
citySF                2.396e-01  1.463e-02  16.385  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.5551 on 45690 degrees of freedom
Multiple R-squared:  0.312, Adjusted R-squared:  0.3119
F-statistic:  2072 on 10 and 45690 DF,  p-value: < 2.2e-16
```

```r
hist(resid(air_train_log_model), main = "Histogram: Log Model Residuals")


#create Q-Q plot for residuals
qqnorm(resid(air_train_log_model))

#add a straight diagonal line to the plot
qqline(resid(air_train_log_model))
```

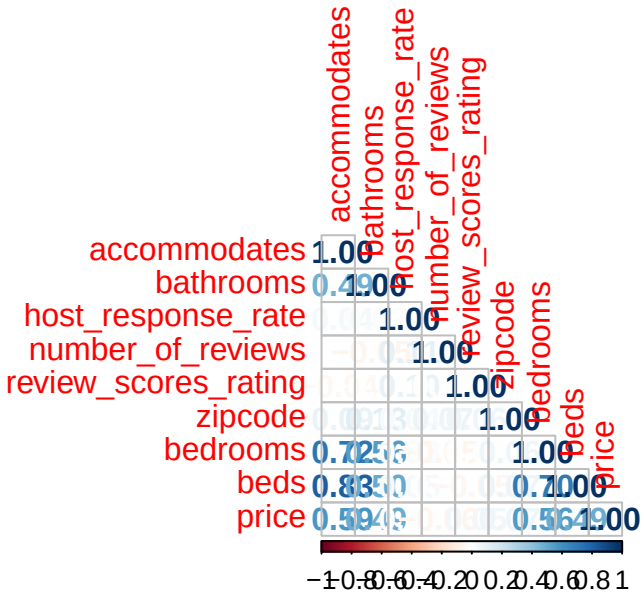



```
par(mfrow = c(1, 1))
```

ii. Find a better prediction model for price (not log of price!) that has a lower test set RMSE than Model 1. Your grade will depend in part on how low the test set RMSE is for your proposed model.

In this section, I train a random forest model and include the variable `host_identity_verified` and `accommodates`. The RMSE is 88.2339 compared to 158.1632 for the linear model.

```
air_training %>%
  drop_na() %>%
  select(where(is.numeric)) %>%
  cor() %>%
  corrplot::corrplot(method = "number", type = "lower")
```




```
set.seed(200)

# head(air_training)
#
# names(air_training)
air %>% select(where(is.character))
```

```
# A tibble: 57,129 x 6
   property_type room_type       bed_type cancellation_policy city  neighbourh~1
   <chr>         <chr>           <chr>    <chr>               <chr> <chr>
 1 Apartment     Entire home/apt Real Bed strict              NYC   Brooklyn He~
 2 Apartment     Entire home/apt Real Bed strict              NYC   Hell's Kitc~
 3 Apartment     Entire home/apt Real Bed moderate            NYC   Harlem
 4 Apartment     Entire home/apt Real Bed moderate            DC    Columbia He~
 5 Apartment     Private room    Real Bed strict              SF    Noe Valley
 6 Apartment     Entire home/apt Real Bed moderate            LA    <NA>
 7 Condominium   Entire home/apt Real Bed moderate            LA    Downtown
 8 House         Private room    Real Bed moderate            SF    Richmond Di~
 9 House         Private room    Real Bed moderate            LA    <NA>
10 Apartment     Private room    Real Bed strict              NYC   Alphabet Ci~
# ... with 57,119 more rows, and abbreviated variable name 1: neighbourhood
```

```r
better_model <- randomForest(price ~ bedrooms + beds + bathrooms + number_of_reviews + rev

sqrt(mean((air_testing$price - predict(better_model, newdata = air_testing))^2, na.rm = TR
```

```
[1] 88.2339
```