Airbnb

Loaded data and performed sanity checks. Observed null values in bedrooms column.

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
airbnb_all <- read.delim("airbnb-vancouver-bc-listings.csv.bz2", sep=",")
#head(airbnb_all)
airbnb <- airbnb_all %>% select(price, bedrooms, room_type, accommodates)
head(airbnb)
```

```
##     price bedrooms       room_type accommodates
## 1 $158.00        2 Entire home/apt            5
## 2 $150.00       NA Entire home/apt            4
## 3  $85.00        1 Entire home/apt            2
## 4 $149.00        1 Entire home/apt            2
## 5 $150.00        1 Entire home/apt            4
## 6 $350.00        2 Entire home/apt            4
```

```
dim(airbnb)
```

```
## [1] 4448    4
```

```
any(is.na(airbnb))
```

```
## [1] TRUE
```

```
any(is.na(airbnb$price))
```

```
## [1] FALSE
```

```
any(is.na(airbnb$bedrooms))
```

```
## [1] TRUE
```

```
any(is.na(airbnb$room_type))
```

```
## [1] FALSE
```

```
any(is.na(airbnb$accommodates))
```

```
## [1] FALSE
```

Cleaning Data

Converted price to numeric.

```
library(stringr)
airbnb <- airbnb %>%
  mutate(price = str_replace(price, "[\\$]", "")) %>%
  mutate(price = str_replace(price, ",", ""))
airbnb$price <- as.numeric(airbnb$price)
class(airbnb$price)
```

```
## [1] "numeric"
```

```
head(airbnb)
```

```
##   price bedrooms       room_type accommodates
## 1   158        2 Entire home/apt            5
## 2   150       NA Entire home/apt            4
## 3    85        1 Entire home/apt            2
## 4   149        1 Entire home/apt            2
## 5   150        1 Entire home/apt            4
## 6   350        2 Entire home/apt            4
```

Removed entries with missing or invalid price, bedrooms, and other variables

```
mean(airbnb$bedrooms, na.rm = TRUE)
```

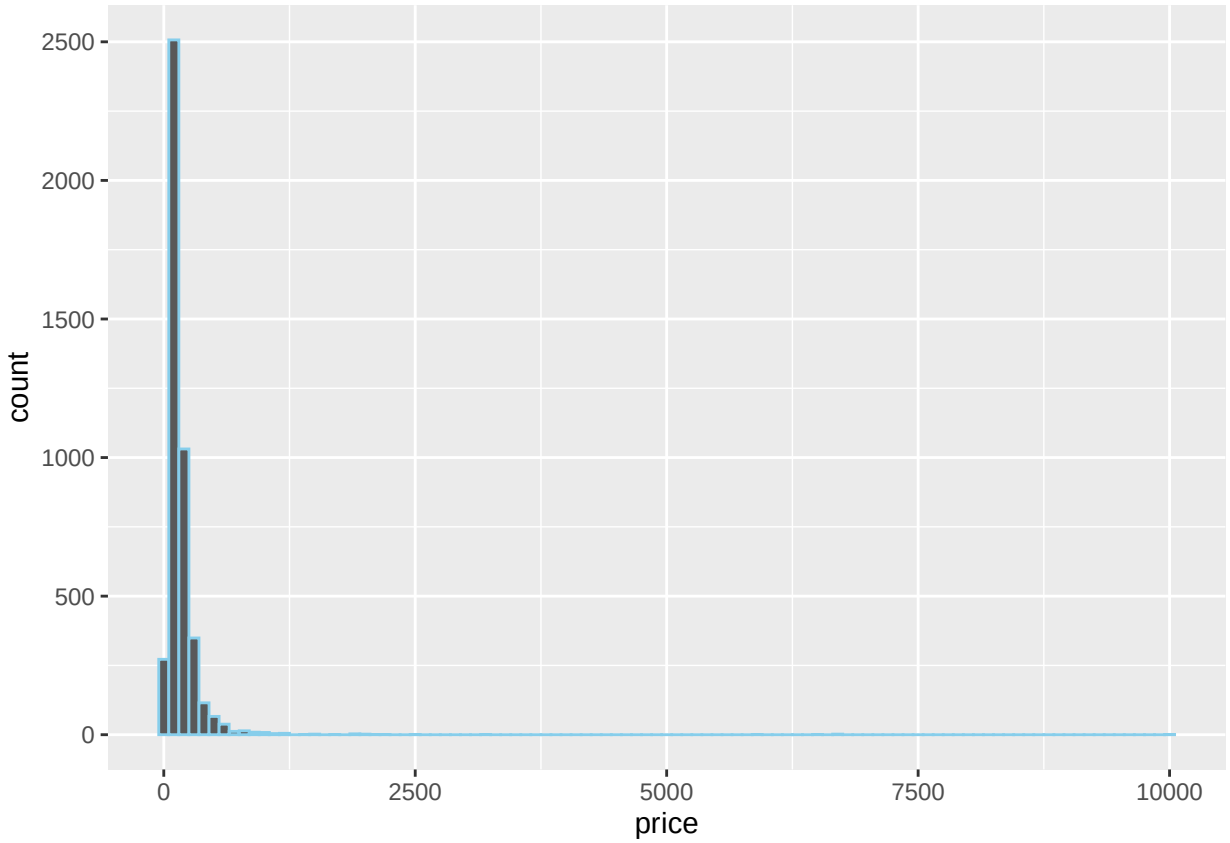
```
## [1] 1.601064
```

```
#since mean is 1.6, rounding it off to 2 and replacing missing values with 2
airbnb$bedrooms[is.na(airbnb$bedrooms)] <- 2
head(airbnb)
```

```
##   price bedrooms       room_type accommodates
## 1   158        2 Entire home/apt            5
## 2   150        2 Entire home/apt            4
## 3    85        1 Entire home/apt            2
## 4   149        1 Entire home/apt            2
## 5   150        1 Entire home/apt            4
## 6   350        2 Entire home/apt            4
```

Analyzed the distribution of price. It is a normal distribution. We should do a log transformation as price has a lower bound.

```
library(ggplot2)
ggplot(airbnb, aes(x=price)) +
    geom_histogram(binwidth=100, colour="skyblue")
```



Converted the number of bedrooms into another variable with a limited number of categories only, such as 0, 1, 2, 3+, and used these categories in the models below.

```
airbnb <- airbnb %>% mutate(BR= case_when(bedrooms == 0 ~ "0",
                                          bedrooms == 1 ~ "1",
                                          bedrooms == 2 ~ "2",
                                         bedrooms >= 3 ~ "3+"))
tail(airbnb,10)
```

```
##      price bedrooms       room_type accommodates BR
## 4439   630        4 Entire home/apt           12 3+
## 4440   351        3 Entire home/apt            7 3+
## 4441    97        1 Entire home/apt            3  1
## 4442   132        1 Entire home/apt            3  1
## 4443   134        1 Entire home/apt            4  1
## 4444    54        1    Private room            2  1
## 4445    52        1    Private room            2  1
## 4446   228        1 Entire home/apt            1  1
```

```
## 4447    80        1 Entire home/apt            1  1
## 4448   121        2 Entire home/apt            6  2
```

Estimated a linear regression model. The higher the adjusted R square, the better the model. Linear regression of price vs BR1 behaves the best out of the three. Linear regression of price vs BR behaves better than all three.

```
library(fastDummies)
dataForLR <- airbnb %>%
 dummy_cols(select_columns = c('BR'), remove_selected_columns = FALSE)


dataForLR<- dataForLR%>%
       rename("BR_3PLUS" = "BR_3+")

head(dataForLR)
```

```
##   price bedrooms       room_type accommodates BR BR_1 BR_2 BR_3PLUS
## 1   158        2 Entire home/apt            5  2    0    1        0
## 2   150        2 Entire home/apt            4  2    0    1        0
## 3    85        1 Entire home/apt            2  1    1    0        0
## 4   149        1 Entire home/apt            2  1    1    0        0
## 5   150        1 Entire home/apt            4  1    1    0        0
## 6   350        2 Entire home/apt            4  2    0    1        0
```

```
##linear regression of price vs bedrooms together

modelBR <- lm(log1p(price) ~ factor(BR), data=airbnb)
summary(modelBR)
```

```
##
## Call:
## lm(formula = log1p(price) ~ factor(BR), data = airbnb)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.4062 -0.3269 -0.0425  0.3399  4.5527
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.65766    0.01112  418.70   <2e-16 ***
## factor(BR)2   0.41405    0.01817   22.78   <2e-16 ***
## factor(BR)3+  1.00664    0.02601   38.70   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5495 on 4445 degrees of freedom
## Multiple R-squared:  0.273,  Adjusted R-squared:  0.2727
## F-statistic: 834.7 on 2 and 4445 DF,  p-value: < 2.2e-16
```

```
##linear regression for each type of bedroom, to compare the models

modelBR1 <- lm(log1p(price) ~ factor(BR_1), data=dataForLR)
summary(modelBR1)
```

```
##
## Call:
## lm(formula = log1p(price) ~ factor(BR_1), data = dataForLR)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.0973 -0.3576 -0.0425  0.3439  4.5527
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    5.23284    0.01288  406.19   <2e-16 ***
## factor(BR_1)1 -0.57519    0.01739  -33.07   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5773 on 4446 degrees of freedom
## Multiple R-squared:  0.1974, Adjusted R-squared:  0.1972
## F-statistic:  1093 on 1 and 4446 DF,  p-value: < 2.2e-16
```

```
modelBR2 <- lm(log1p(price) ~ factor(BR_2), data=dataForLR)
summary(modelBR2)
```

```
##
## Call:
## lm(formula = log1p(price) ~ factor(BR_2), data = dataForLR)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.2027 -0.3874 -0.0544  0.3512  4.3686
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    4.84172    0.01163  416.48   <2e-16 ***
## factor(BR_2)1  0.22999    0.02028   11.34   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6353 on 4446 degrees of freedom
## Multiple R-squared:  0.02812,    Adjusted R-squared:  0.0279
## F-statistic: 128.6 on 1 and 4446 DF,  p-value: < 2.2e-16
```

```
modelBR3 <- lm(log1p(price) ~ factor(BR_3PLUS), data=dataForLR)
summary(modelBR3)
```

```
##
## Call:
## lm(formula = log1p(price) ~ factor(BR_3PLUS), data = dataForLR)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.4062 -0.3469 -0.0170  0.3347  4.3975
##
## Coefficients:
```

```
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       4.812794   0.009295   517.8   <2e-16 ***
## factor(BR_3PLUS)1 0.851506   0.026530    32.1   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5806 on 4446 degrees of freedom
## Multiple R-squared:  0.1881, Adjusted R-squared:  0.1879
## F-statistic:  1030 on 1 and 4446 DF,  p-value: < 2.2e-16
```

Included two further variables into the model: room type and accommodates. Room type only contains a few values, but accommodates contains many different categories.

Answer:

room_type

```
table(dataForLR$room_type)
```

```
##
## Entire home/apt      Hotel room    Private room     Shared room
##            3582               4             854               8
```

accommodates

```
table(dataForLR$accommodates)
```

```
##
##    1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16
##  288 1699  462 1108  247  412   61  101    8   27    5   14    3    7    1    5
```

Converted the room type into 3 categories: Entire home/apt, Private room, Other; and re-coded accommodates into 3 categories: "1", "2", "3 or more"

```
dataForLR <- dataForLR %>% mutate(room_type = case_when(room_type == "Entire home/apt" ~ "Entire home/a

dataForLR <- dataForLR %>% mutate(accommodates = case_when(accommodates == 1 ~ "1", accommodates == 2 ~

head(dataForLR)
```

```
##   price bedrooms       room_type accommodates BR BR_1 BR_2 BR_3PLUS
## 1   158        2 Entire home/apt    3 or more  2    0    1        0
## 2   150        2 Entire home/apt    3 or more  2    0    1        0
## 3    85        1 Entire home/apt            2  1    1    0        0
## 4   149        1 Entire home/apt            2  1    1    0        0
## 5   150        1 Entire home/apt    3 or more  1    1    0        0
## 6   350        2 Entire home/apt    3 or more  2    0    1        0
```

```
table(dataForLR$room_type)
```

```
##
## Entire home/apt           Other    Private room
##            3582              12             854
```

```
table(dataForLR$accommodates)
```

```
##
##         1         2 3 or more
##       288      1699      2461
```

Amended my previous model with these two variables. Room type - Other is not statistically significant. Rest of the categories are statistically significant. For each category a baseline is selected, for eg for accommodates, R finds the statistical significance of accommodates 2 with relation to accommodates 1. This is why we see 6 coefficients instead of 9. The adjusted R square before was 0.27. When we consider all 3 variables it is 0.399, which means this is a better model than price vs BR.

```
dataForLR1 <- dataForLR %>%
 dummy_cols(select_columns = c('accommodates','room_type'), remove_selected_columns = FALSE)
head(dataForLR)
```

```
##   price bedrooms       room_type accommodates BR BR_1 BR_2 BR_3PLUS
## 1   158        2 Entire home/apt    3 or more  2    0    1        0
## 2   150        2 Entire home/apt    3 or more  2    0    1        0
## 3    85        1 Entire home/apt            2  1    1    0        0
## 4   149        1 Entire home/apt            2  1    1    0        0
## 5   150        1 Entire home/apt    3 or more  1    1    0        0
## 6   350        2 Entire home/apt    3 or more  2    0    1        0
```

```
## creating a separate model for each category of 3 different variables would be a lot of iterations, p

modelCombined <- lm(log1p(price) ~+factor(room_type)+factor(BR)+factor(accommodates), data=dataForLR1)
summary(modelCombined)
```

```
##
## Call:
## lm(formula = log1p(price) ~ +factor(room_type) + factor(BR) +
##     factor(accommodates), data = dataForLR1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.4184 -0.3145 -0.0505  0.2748  4.8458
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     4.43542    0.03605 123.048   <2e-16 ***
## factor(room_type)Other         -0.08896    0.14575  -0.610    0.542
## factor(room_type)Private room -0.37182    0.02304 -16.139   <2e-16 ***
## factor(BR)2                    0.17963    0.01887   9.520   <2e-16 ***
## factor(BR)3+                   0.71332    0.02674  26.674   <2e-16 ***
## factor(accommodates)2          0.30092    0.03458   8.701   <2e-16 ***
## factor(accommodates)3 or more  0.52773    0.03758  14.042   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4994 on 4441 degrees of freedom
```

```
## Multiple R-squared:    0.4,  Adjusted R-squared:  0.3992
## F-statistic: 493.4 on 6 and 4441 DF,  p-value: < 2.2e-16
```

Type "Other" is not statistically significant.The data doesn't provide evidence of an effect; it doesn't mean that such an effect cannot exist. I think it is coming as insignificant as the sample size is too small for Other type.

Now we use the model above to predict (log) price for each listing in our data.

```
dataForLR2 <- dataForLR1 %>% select(BR ,room_type, accommodates,price)
head(dataForLR2)
```

```
##   BR       room_type accommodates price
## 1  2 Entire home/apt    3 or more   158
## 2  2 Entire home/apt    3 or more   150
## 3  1 Entire home/apt            2    85
## 4  1 Entire home/apt            2   149
## 5  1 Entire home/apt    3 or more   150
## 6  2 Entire home/apt    3 or more   350
```

```
data <- dataForLR2 %>% mutate(predictedprice = predict(modelCombined, dataForLR2))
#The price is logged here
head(data)
```

```
##   BR       room_type accommodates price predictedprice
## 1  2 Entire home/apt    3 or more   158       5.142781
## 2  2 Entire home/apt    3 or more   150       5.142781
## 3  1 Entire home/apt            2    85       4.736333
## 4  1 Entire home/apt            2   149       4.736333
## 5  1 Entire home/apt    3 or more   150       4.963148
## 6  2 Entire home/apt    3 or more   350       5.142781
```

Computing root-mean-squared-error (RMSE) of our predictions.

```
sqrt(mean((log(data$price) - data$predictedprice)^2))
```

```
## [1] 0.5028288
```

Using our model to predict log price for a 2-bedroom apartment that accommodates 4 (i.e., a full 2BR apartment).

The price of a 2 bedroom apartment that accommodates 4 would be 171 dollars.

```
testdata <-  data.frame("BR" = "2",
                        "room_type" = "Entire home/apt",
                        "accommodates" = "3 or more")
predict(modelCombined, testdata)
```

```
##        1
## 5.142781
```

```
exp(predict(modelCombined, testdata))
```

```
##        1
## 171.1911
```