EDA: Housing Dominican Republic

Introduction

For this analysis, we will examine the effect that area (measured in squared meters) has on the value of apartment, for residential use, in Santo Domingo, Domincan Republic. Prices and apartment's characteristics are collect via web scraping. Specifically, data were retrieved on 22nd of march, 2022, from supercasas.com, a beacon on the online dominican real estate market.

Loading libraries and data

```
rm(list = ls())

options(scipen = 999)

library(robustbase)
library(tidyverse)
library(caret)

set.seed(1234)
```

Data on every listing available on several dates at supercasas.com were retrieved. Here, we load and filter said data so we only have information on apartments for residential use on the date of interest. Then, we split the dataset into training (70%) and testing (30%).

```
path <- "../1_data/0_raw/housing price/"
housing_files <- list.files(path)
housing <- read_csv(paste0(path, housing_files))
```

```
## Rows: 24575 Columns: 25
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (8): id, currency, seller, neighborhood, city, province, status, usage
## dbl  (7): parking, bathrooms, bedrooms, price, area, story, price.usd
## lgl  (9): planta, lift, pool, pozo, terraza, lobby, balcon, jacuzzi, gimnasio
## date (1): date
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
housing <- housing %>%
  filter(date ==  "2022-03-22",
         usage == "Residencial",
```

```
        province %in% c("Santo Domingo", "Santo Domingo Centro (D.N.)")) %>%
  rename(location = neighborhood) %>%
  select(-c(date, usage, city, province))

inTrain <- createDataPartition(housing$price.usd, p = 0.7, list = FALSE)

training <- housing[inTrain, ]
testing <- housing[-inTrain, ]
```

Data cleaning

First, let's see what's on the dataset:

```
glimpse(training)
```

```
## Rows: 4,154
## Columns: 21
## $ id        <chr> "/apartamentos-venta-piantini/1265273/", "/apartamentos-vent~
## $ parking   <dbl> 3, 2, 2, 3, NA, 3, 2, 2, 2, 1, 3, 1, 3, 3, 3, 1, 2, 1, 2, 2,~
## $ bathrooms <dbl> 3.0, 2.5, 2.5, 3.5, 3.5, 3.5, 3.5, 3.5, 2.5, 2.0, 3.0, 1.5, ~
## $ bedrooms  <dbl> 2, 2, 2, 3, 3, 3, 3, 3, 2, 1, 2, 1, 3, 3, 3, 1, 1, 1, 2, 3, ~
## $ currency  <chr> "US$", "US$", "US$", "US$", "US$", "US$", "US$", "US$", "US$~
## $ price     <dbl> 258000, 220000, 296250, 408825, 390000, 370000, 275000, 3000~
## $ seller    <chr> "BAEZ MUESES INMOBILIARIA", "Premium Real Estate", "Algonovo~
## $ location  <chr> "Piantini", "Piantini", "Piantini", "Piantini", "Piantini", ~
## $ status    <chr> "Segundo Uso", "Segundo Uso", "En Construcción", "En Constru~
## $ area      <dbl> 180, 100, 153, 185, NA, 217, NA, NA, 142, NA, 225, 70, 175, ~
## $ story     <dbl> 2, 2, 4, 3, NA, NA, NA, NA, NA, NA, NA, NA, 3, 13, NA, NA, N~
## $ planta    <lgl> TRUE, FALSE, TRUE, TRUE, FALSE, FALSE, TRUE, TRUE, TRUE, TRU~
## $ lift      <lgl> TRUE, TRUE, TRUE, TRUE, FALSE, TRUE, TRUE, TRUE, TRUE, TRUE,~
## $ pool      <lgl> TRUE, TRUE, TRUE, TRUE, TRUE, FALSE, FALSE, FALSE, FALSE, FA~
## $ pozo      <lgl> TRUE, FALSE, TRUE, TRUE, FALSE, FALSE, FALSE, FALSE, FALSE, ~
## $ terraza   <lgl> TRUE, TRUE, TRUE, TRUE, FALSE, TRUE, FALSE, FALSE, FALSE, TR~
## $ lobby     <lgl> TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, FALSE, FALSE, FALSE, FAL~
## $ balcon    <lgl> TRUE, TRUE, TRUE, TRUE, FALSE, FALSE, FALSE, FALSE, FALSE, F~
## $ jacuzzi   <lgl> FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALS~
## $ gimnasio  <lgl> TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, FALSE, FALSE, FALSE, FAL~
## $ price.usd <dbl> 258000, 220000, 296250, 408825, 390000, 370000, 275000, 3000~
```

Our training dataset contains 4,154 observations and 21 variables. Of them, we can highlight `price`, `currency` and `price.usd`. `price` and `currency` is the actual price shown in the listing's site. They can be in local currency or US dollars, depending on the seller's preference. `price.usd` is a user-made feature of prices in US dollars. Hence, if prices were stated in local currency, they were converted into US dollars. Otherwise, they stay the same. We'll drop `currency` and `price` and keep `price.usd`. Then, and for simplicity's sake, we'll rename `price.usd` as `price`.

```
training <- training %>%
  select(-c(currency, price)) %>%
  rename(price = price.usd)
```

When looking at the proportion of NAs are present per variable, over 50% of listings did not provide information regarding the floor the apartment is located at. The proportion of missing values for all other variables is acceptable. And so, we removed that variable.

```
apply(training, 2,
      \(x) {
        n <- length(x)
        na <- x %>%
          is.na() %>%
          sum()
        prop.na <- na / n * 100
        })
```

```
##         id    parking  bathrooms   bedrooms     seller   location     status
##  0.0000000  6.7645643  5.0072220  2.4795378  0.0000000  0.4573905  6.6441984
##       area      story     planta       lift       pool       pozo    terraza
## 10.7125662 51.4443909  0.0000000  0.0000000  0.0000000  0.0000000  0.0000000
##      lobby     balcon    jacuzzi   gimnasio      price
##  0.0000000  0.0000000  0.0000000  0.0000000  0.0000000
```

```
training <- training %>%
  select(-c(story))
```

Before implementing some formal procedure for outlier removal, let's analyse the data at hand. First, we can see that area has some outstanding observations: a minimum value of 30 and a maximum of 650,000.

```
summary(training)
```

```
##       id                parking          bathrooms        bedrooms
##  Length:4154        Min.   : 1.000   Min.   :1.000   Min.   :1.000
##  Class :character   1st Qu.: 2.000   1st Qu.:2.000   1st Qu.:2.000
##  Mode  :character   Median : 2.000   Median :2.500   Median :3.000
##                     Mean   : 2.012   Mean   :2.719   Mean   :2.554
##                     3rd Qu.: 2.000   3rd Qu.:3.500   3rd Qu.:3.000
##                     Max.   :25.000   Max.   :6.500   Max.   :6.000
##                     NA's   :281      NA's   :208     NA's   :103
##     seller            location            status               area
##  Length:4154        Length:4154        Length:4154        Min.   :    30.0
##  Class :character   Class :character   Class :character   1st Qu.:   100.0
##  Mode  :character   Mode  :character   Mode  :character   Median :   151.0
##                                                           Mean   :   582.8
##                                                           3rd Qu.:   220.0
##                                                           Max.   :650000.0
##                                                           NA's   :445
##    planta           lift            pool            pozo
##  Mode :logical   Mode :logical   Mode :logical   Mode :logical
##  FALSE:1516      FALSE:1283      FALSE:2732      FALSE:2962
##  TRUE :2638      TRUE :2871      TRUE :1422      TRUE :1192
##
##
##
##
```

```
##   terraza          lobby           balcon         jacuzzi
##  Mode :logical   Mode :logical   Mode :logical   Mode :logical
##  FALSE:2424      FALSE:1665      FALSE:1307      FALSE:3372
##  TRUE :1730      TRUE :2489      TRUE :2847      TRUE :782
##
##
##
##
##   gimnasio           price
##  Mode :logical   Min.   :        1
##  FALSE:1991      1st Qu.:   138000
##  TRUE :2163      Median :   210000
##                  Mean   :   496043
##                  3rd Qu.:   300000
##                  Max.   :257500000
##
```

By viewing the "largest apartments", some things become apparent. First, the are repeated observations on this small sample: see the third and fourth rows, for instance. Second, the first four rows are obviously typos: the seller typed 650,000 instead of 650 squared meters, to cited the first case. Third, the apartment listed on the fifth row is no longer available rising some doubts on its veracity. Last, the sixth "apartment" is actually a house. When analysing the "smallest apartments", everything seems in order.

```
training %>%
  arrange(desc(area)) %>%
  head(10)
```

```
## # A tibble: 10 x 18
##    id      parking bathrooms bedrooms seller location status   area planta lift
##    <chr>     <dbl>     <dbl>    <dbl> <chr>  <chr>    <chr>   <dbl> <lgl>  <lgl>
##  1 /apart~       5       3.5        3 Flavi~ Piantini Segun~ 650000 TRUE   TRUE
##  2 /apart~       3       3.5        3 Flavi~ Piantini Segun~ 400000 TRUE   TRUE
##  3 /apart~       2       3.5        3 Premi~ Alma Ro~ En Co~ 220000 TRUE   TRUE
##  4 /apart~       2       3.5        3 Premi~ Alma Ro~ En Co~ 220000 TRUE   TRUE
##  5 /apart~       3       3.5        3 Gineb~ Los Cac~ Segun~   2016 FALSE  FALSE
##  6 /apart~      NA       5.5        5 Vícto~ Cuesta ~ Segun~   1431 TRUE   FALSE
##  7 /apart~      NA       5          4 Patri~ Paraiso  Segun~    952 TRUE   TRUE
##  8 /apart~       4       6          4 Paez ~ La Espe~ Segun~    890 FALSE  FALSE
##  9 /apart~      NA      NA          4 Paez ~ Anacaona <NA>      890 FALSE  FALSE
## 10 /apart~       5       4.5        4 Flavi~ Anacaona Segun~    854 FALSE  TRUE
## # ... with 8 more variables: pool <lgl>, pozo <lgl>, terraza <lgl>,
## #   lobby <lgl>, balcon <lgl>, jacuzzi <lgl>, gimnasio <lgl>, price <dbl>
```

So, to fix these (1) we eliminate duplicates, (2) we divide by 1,000 the area of those apartments with over 10,000 squared meters of area, (3) remove those apartments that are obviously not of interest.

```
training <- training %>%
  filter(id != "/apartamentos-venta-cuesta-hermosa-ii/1236477/",
         id != "/apartamentos-venta-los-cacicazgos/1272251/") %>%
  mutate(area = ifelse(area > 10000, area / 1000, area)) %>%
  unique()
```

Let's do the same with `price`. Viewing `price` alone might be misleading as a apartment with 30 squared meters could be worth 20,000 dollars, but one with 280 squared meters could hardly be worth $10,256. Price per squared meter could tell us more about how extreme of a value it is.

```
summary(training$price)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
##         1    142000    215000    528651    315000 257500000
```

```
training <- training %>%
  mutate(price_per_m2 = price / area,
         area_per_br = area / bedrooms)
```

Anything below $200 seems rather dubious, right? Let's filter them out.But let's use something less subjective.

```
(cutoff <- adjboxStats(training$price_per_m2)$fence)
```

```
## The default of 'doScale' is FALSE now for stability;
##   set options(mc_doScale_quiet=TRUE) to suppress this (once per session) message
```

```
## [1]  226.7685 3270.8771
```

```
training <- training %>%
  filter(between(price_per_m2, cutoff[1], cutoff[2]))

summary(training$price_per_m2)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   268.9  1100.0  1456.3  1505.9  1847.7  3219.2
```

Anything below $200 seems rather dubious, right? Let's filter them out.But let's use something less subjective.

```
(cutoff <- adjboxStats(training$area_per_br)$fence)
```

```
## [1]  27.28568 153.12550
```

```
training <- training %>%
  filter(between(area_per_br, cutoff[1], cutoff[2]))

summary(training$area_per_br)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   27.33   51.67   64.00   68.33   80.00  150.00
```

```
glimpse(training)
```

```
## Rows: 2,851
## Columns: 20
## $ id           <chr> "/apartamentos-venta-piantini/1265273/", "/apartamentos-v~
## $ parking      <dbl> 3, 2, 2, 3, 3, 2, 3, 3, 3, 3, 1, 2, 1, 2, 2, 3, 2, 3, 3, ~
```

```
## $ bathrooms    <dbl> 3.0, 2.5, 2.5, 3.5, 3.5, 2.5, 3.0, 3.5, 3.5, 3.5, 1.5, 1.~
## $ bedrooms     <dbl> 2, 2, 2, 3, 3, 2, 2, 3, 3, 3, 1, 1, 1, 3, 3, 3, 2, 3, 3, ~
## $ seller       <chr> "BAEZ MUESES INMOBILIARIA", "Premium Real Estate", "Algon~
## $ location     <chr> "Piantini", "Piantini", "Piantini", "Piantini", "Piantini~
## $ status       <chr> "Segundo Uso", "Segundo Uso", "En Construcción", "En Cons~
## $ area         <dbl> 180, 100, 153, 185, 217, 142, 225, 175, 200, 340, 67, 103~
## $ planta       <lgl> TRUE, FALSE, TRUE, TRUE, FALSE, TRUE, FALSE, TRUE, TRUE, ~
## $ lift         <lgl> TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, FALSE, TR~
## $ pool         <lgl> TRUE, TRUE, TRUE, TRUE, FALSE, FALSE, FALSE, FALSE, TRUE,~
## $ pozo         <lgl> TRUE, FALSE, TRUE, TRUE, FALSE, FALSE, FALSE, FALSE, FALS~
## $ terraza      <lgl> TRUE, TRUE, TRUE, TRUE, TRUE, FALSE, FALSE, FALSE, FALSE,~
## $ lobby        <lgl> TRUE, TRUE, TRUE, TRUE, TRUE, FALSE, FALSE, TRUE, TRUE, T~
## $ balcon       <lgl> TRUE, TRUE, TRUE, TRUE, FALSE, FALSE, FALSE, FALSE, TRUE,~
## $ jacuzzi      <lgl> FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, TRUE, FA~
## $ gimnasio     <lgl> TRUE, TRUE, TRUE, TRUE, TRUE, FALSE, FALSE, TRUE, TRUE, T~
## $ price        <dbl> 258000, 220000, 296250, 408825, 370000, 155000, 280000, 3~
## $ price_per_m2 <dbl> 1433.333, 2200.000, 1936.275, 2209.865, 1705.069, 1091.54~
## $ area_per_br  <dbl> 90.00000, 50.00000, 76.50000, 61.66667, 72.33333, 71.0000~
```

Our data frame is now 20 columns wide and 2,851 rows long.

Many times, the same apartment is listed by different sellers. So, eliminating duplicates is not enough to remove confliting. Some seller are not as rigourous as to list all the amaneties, so we are going to keep the most complete listing:

Some feature engineering

Now, let's do some feature engineering to help the analysis:

```
glimpse(training)
```

```
## Rows: 2,851
## Columns: 20
## $ id           <chr> "/apartamentos-venta-piantini/1265273/", "/apartamentos-v~
## $ parking      <dbl> 3, 2, 2, 3, 3, 2, 3, 3, 3, 3, 1, 2, 1, 2, 2, 3, 2, 3, 3, ~
## $ bathrooms    <dbl> 3.0, 2.5, 2.5, 3.5, 3.5, 2.5, 3.0, 3.5, 3.5, 3.5, 1.5, 1.~
## $ bedrooms     <dbl> 2, 2, 2, 3, 3, 2, 2, 3, 3, 3, 1, 1, 1, 3, 3, 3, 2, 3, 3, ~
## $ seller       <chr> "BAEZ MUESES INMOBILIARIA", "Premium Real Estate", "Algon~
## $ location     <chr> "Piantini", "Piantini", "Piantini", "Piantini", "Piantini~
## $ status       <chr> "Segundo Uso", "Segundo Uso", "En Construcción", "En Cons~
## $ area         <dbl> 180, 100, 153, 185, 217, 142, 225, 175, 200, 340, 67, 103~
## $ planta       <lgl> TRUE, FALSE, TRUE, TRUE, FALSE, TRUE, FALSE, TRUE, TRUE, ~
## $ lift         <lgl> TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, FALSE, TR~
## $ pool         <lgl> TRUE, TRUE, TRUE, TRUE, FALSE, FALSE, FALSE, FALSE, TRUE,~
## $ pozo         <lgl> TRUE, FALSE, TRUE, TRUE, FALSE, FALSE, FALSE, FALSE, FALS~
## $ terraza      <lgl> TRUE, TRUE, TRUE, TRUE, TRUE, FALSE, FALSE, FALSE, FALSE,~
## $ lobby        <lgl> TRUE, TRUE, TRUE, TRUE, TRUE, FALSE, FALSE, TRUE, TRUE, T~
## $ balcon       <lgl> TRUE, TRUE, TRUE, TRUE, FALSE, FALSE, FALSE, FALSE, TRUE,~
## $ jacuzzi      <lgl> FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, TRUE, FA~
## $ gimnasio     <lgl> TRUE, TRUE, TRUE, TRUE, TRUE, FALSE, FALSE, TRUE, TRUE, T~
## $ price        <dbl> 258000, 220000, 296250, 408825, 370000, 155000, 280000, 3~
## $ price_per_m2 <dbl> 1433.333, 2200.000, 1936.275, 2209.865, 1705.069, 1091.54~
## $ area_per_br  <dbl> 90.00000, 50.00000, 76.50000, 61.66667, 72.33333, 71.0000~
```

```
summary(training)
```

```
##       id                parking         bathrooms         bedrooms
##  Length:2851        Min.   :1.000   Min.   :1.000   Min.   :1.000
##  Class :character   1st Qu.:2.000   1st Qu.:2.000   1st Qu.:2.000
##  Mode  :character   Median :2.000   Median :2.500   Median :3.000
##                     Mean   :2.013   Mean   :2.751   Mean   :2.552
##                     3rd Qu.:2.000   3rd Qu.:3.500   3rd Qu.:3.000
##                     Max.   :6.000   Max.   :6.500   Max.   :6.000
##                     NA's   :101     NA's   :61
##     seller            location            status               area
##  Length:2851        Length:2851        Length:2851        Min.   : 30.0
##  Class :character   Class :character   Class :character   1st Qu.:104.0
##  Mode  :character   Mode  :character   Mode  :character   Median :155.0
##                                                           Mean   :177.3
##                                                           3rd Qu.:222.0
##                                                           Max.   :720.0
##
##    planta            lift             pool             pozo
##  Mode :logical   Mode :logical   Mode :logical   Mode :logical
##  FALSE:933       FALSE:787       FALSE:1881      FALSE:1980
##  TRUE :1918      TRUE :2064      TRUE :970       TRUE :871
##
##
##
##
##   terraza           lobby            balcon          jacuzzi
##  Mode :logical   Mode :logical   Mode :logical   Mode :logical
##  FALSE:1661      FALSE:1077      FALSE:828       FALSE:2312
##  TRUE :1190      TRUE :1774      TRUE :2023      TRUE :539
##
##
##
##
##   gimnasio            price           price_per_m2     area_per_br
##  Mode :logical   Min.   :  20000   Min.   : 425.3   Min.   : 27.33
##  FALSE:1317      1st Qu.: 145250   1st Qu.:1107.6   1st Qu.: 51.67
##  TRUE :1534      Median : 217650   Median :1459.5   Median : 64.00
##                  Mean   : 262042   Mean   :1502.3   Mean   : 68.33
##                  3rd Qu.: 315000   3rd Qu.:1843.8   3rd Qu.: 80.00
##                  Max.   :1800000   Max.   :3219.2   Max.   :150.00
##
```

```
training <- training %>%
  na.omit()

nonOutlier <- adjOutlyingness(training)
nonOutlier <- nonOutlier$nonOut

training <- training[nonOutlier, ]

glimpse(training)
```

```
## Rows: 2,467
## Columns: 20
## $ id           <chr> "/apartamentos-venta-piantini/1265273/", "/apartamentos-v~
## $ parking      <dbl> 3, 2, 2, 3, 3, 3, 3, 1, 1, 2, 2, 3, 2, 3, 3, 2, 1, 3, 5, ~
## $ bathrooms    <dbl> 3.0, 2.5, 2.5, 3.5, 3.5, 3.5, 3.5, 1.5, 1.5, 3.5, 2.5, 3.~
## $ bedrooms     <dbl> 2, 2, 2, 3, 3, 3, 3, 1, 1, 3, 3, 3, 2, 3, 3, 3, 1, 3, 3, ~
## $ seller       <chr> "BAEZ MUESES INMOBILIARIA", "Premium Real Estate", "Algon~
## $ location     <chr> "Piantini", "Piantini", "Piantini", "Piantini", "Piantini~
## $ status       <chr> "Segundo Uso", "Segundo Uso", "En Construcción", "En Cons~
## $ area         <dbl> 180, 100, 153, 185, 175, 200, 340, 67, 65, 170, 192, 221,~
## $ planta       <lgl> TRUE, FALSE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TR~
## $ lift         <lgl> TRUE, TRUE, TRUE, TRUE, TRUE, FALSE, TRUE, TRUE, TRUE, TR~
## $ pool         <lgl> TRUE, TRUE, TRUE, TRUE, FALSE, TRUE, TRUE, TRUE, TRUE, FA~
## $ pozo         <lgl> TRUE, FALSE, TRUE, TRUE, FALSE, FALSE, FALSE, TRUE, TRUE,~
## $ terraza      <lgl> TRUE, TRUE, TRUE, TRUE, FALSE, FALSE, TRUE, TRUE, TRUE, T~
## $ lobby        <lgl> TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRU~
## $ balcon       <lgl> TRUE, TRUE, TRUE, TRUE, FALSE, TRUE, TRUE, TRUE, TRUE, TR~
## $ jacuzzi      <lgl> FALSE, FALSE, FALSE, FALSE, TRUE, FALSE, TRUE, TRUE, FALS~
## $ gimnasio     <lgl> TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRU~
## $ price        <dbl> 258000, 220000, 296250, 408825, 358750, 370000, 695000, 1~
## $ price_per_m2 <dbl> 1433.333, 2200.000, 1936.275, 2209.865, 2050.000, 1850.00~
## $ area_per_br  <dbl> 90.00000, 50.00000, 76.50000, 61.66667, 58.33333, 66.6666~
```

```
summary(training)
```

```
##       id                parking        bathrooms        bedrooms
##  Length:2467        Min.   :1.000   Min.   :1.000   Min.   :1.000
##  Class :character   1st Qu.:2.000   1st Qu.:2.000   1st Qu.:2.000
##  Mode  :character   Median :2.000   Median :2.500   Median :3.000
##                     Mean   :1.988   Mean   :2.731   Mean   :2.533
##                     3rd Qu.:2.000   3rd Qu.:3.500   3rd Qu.:3.000
##                     Max.   :6.000   Max.   :6.000   Max.   :5.000
##     seller            location            status              area
##  Length:2467        Length:2467        Length:2467        Min.   : 36.0
##  Class :character   Class :character   Class :character   1st Qu.:105.0
##  Mode  :character   Mode  :character   Mode  :character   Median :154.0
##                                                           Mean   :171.6
##                                                           3rd Qu.:220.0
##                                                           Max.   :554.0
##    planta           lift            pool            pozo
##  Mode :logical   Mode :logical   Mode :logical   Mode :logical
##  FALSE:741       FALSE:636       FALSE:1621      FALSE:1665
##  TRUE :1726      TRUE :1831      TRUE :846       TRUE :802
##
##
##
##   terraza          lobby           balcon         jacuzzi
##  Mode :logical   Mode :logical   Mode :logical   Mode :logical
##  FALSE:1411      FALSE:880       FALSE:650       FALSE:1997
##  TRUE :1056      TRUE :1587      TRUE :1817      TRUE :470
##
##
##
##   gimnasio           price          price_per_m2     area_per_br
```

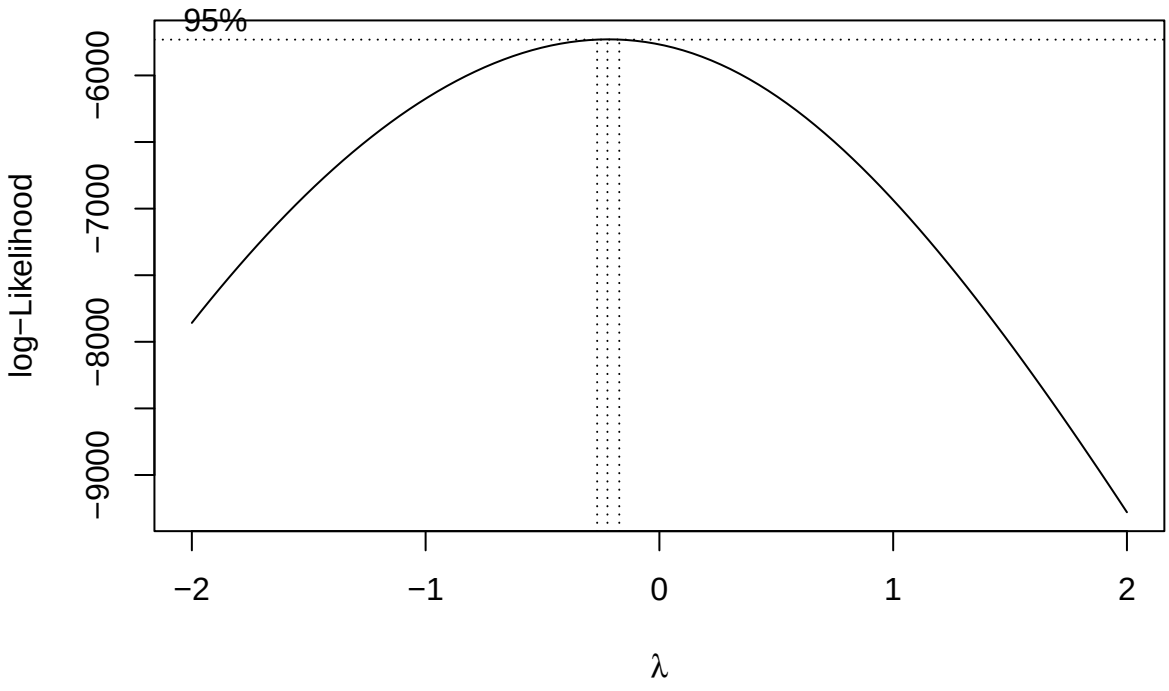
```
##  Mode :logical   Min.   :  30462   Min.   : 425.3   Min.   : 27.33
##  FALSE:1107      1st Qu.: 146726   1st Qu.:1117.7   1st Qu.: 51.67
##  TRUE :1360      Median : 217000   Median :1475.4   Median : 63.33
##                  Mean   : 253166   Mean   :1506.6   Mean   : 67.06
##                  3rd Qu.: 301000   3rd Qu.:1847.4   3rd Qu.: 78.67
##                  Max.   :1100000   Max.   :3219.2   Max.   :150.00
```

```
#training <- training[complete.cases(training), ]
#dim(training)
```

Now, our data.frame has 2,467 observations and 20 variables.

Final removal of outliers

```
trn <- training %>%
  mutate(no_amenities = rowSums(across(planta:gimnasio))) %>%
  arrange(desc(no_amenities)) %>%
  distinct(bathrooms, bedrooms, area, price,
           .keep_all = TRUE)

# Box-Cox tranform
bc <- with(training,
           MASS::boxcox(price ~ area * location))
```

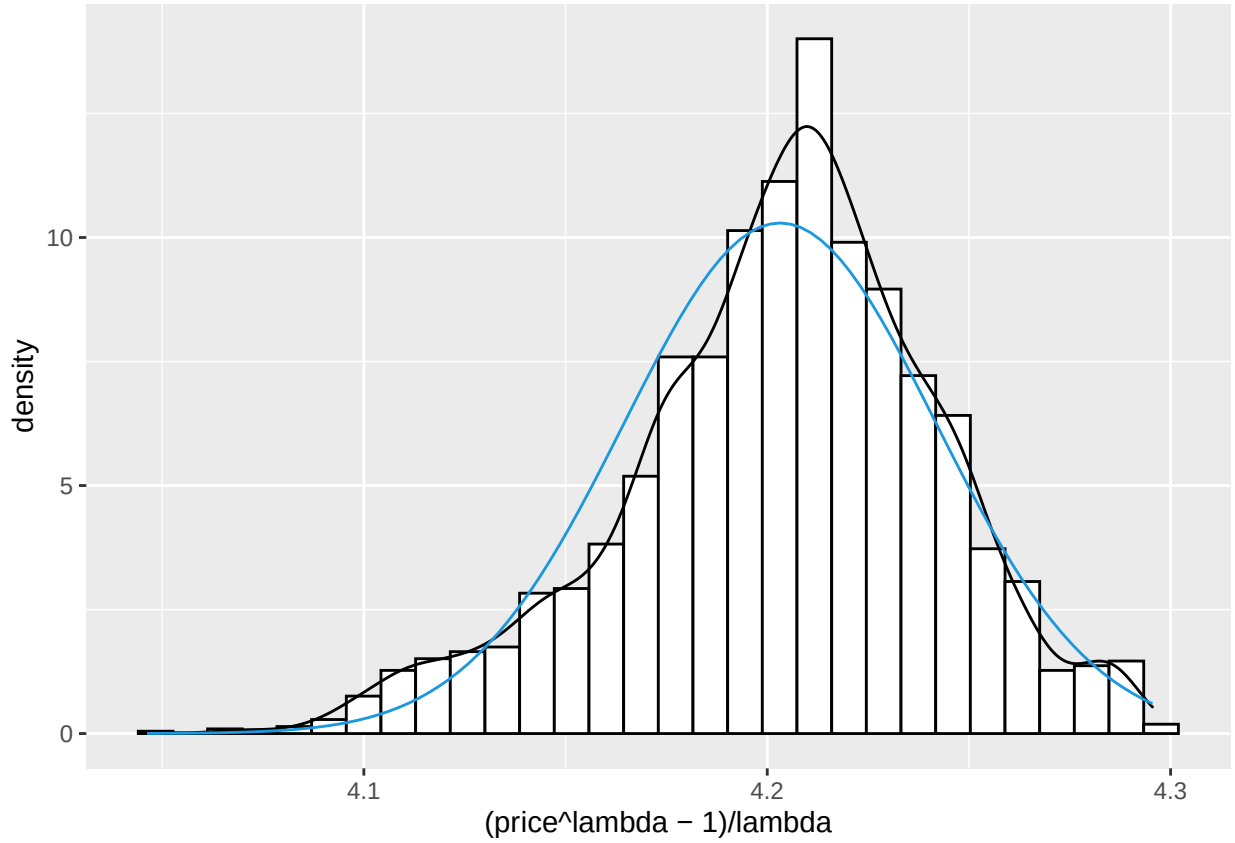


```
(lambda <- bc$x[which.max(bc$y)])
```

```
## [1] -0.2222222
```

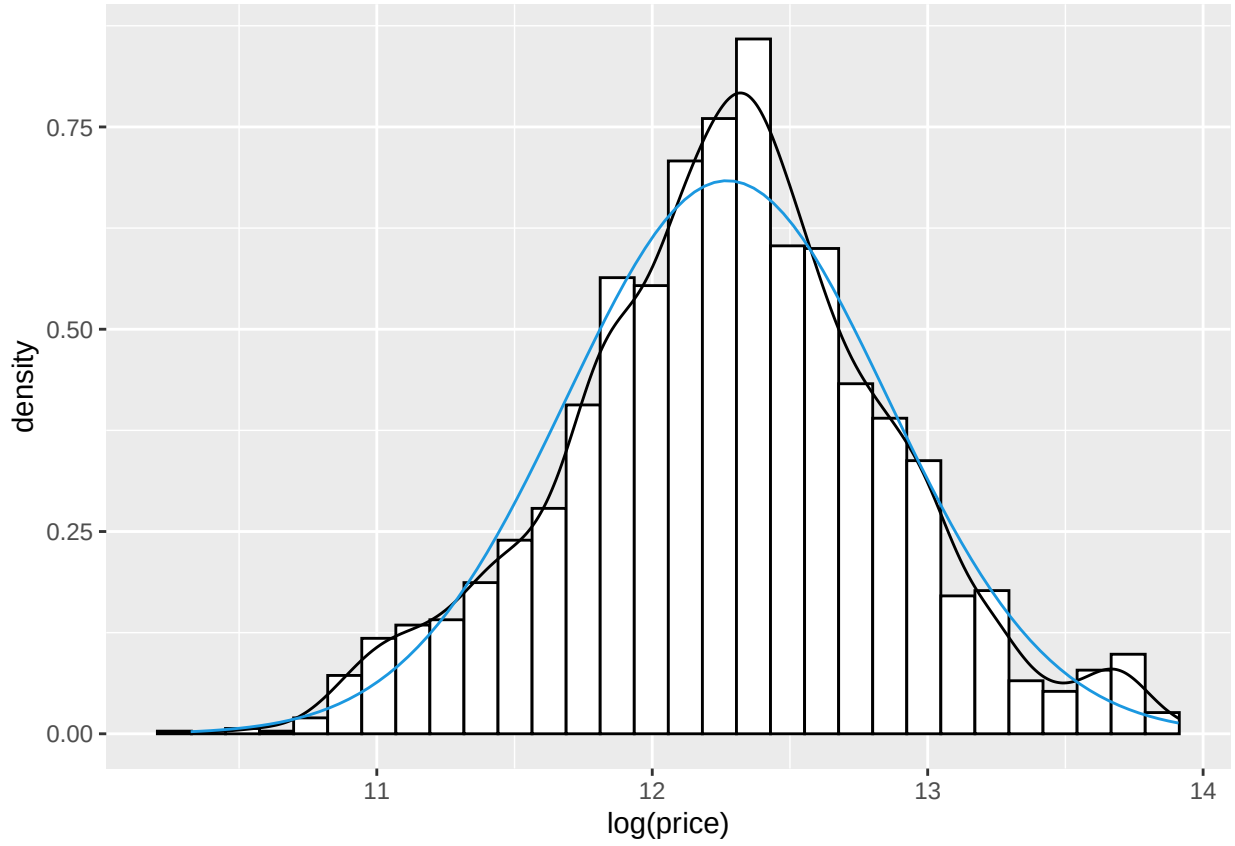
```
new_model <- lm((price ^ lambda - 1)/lambda ~ area, data = training)

ggplot(training, aes(x = (price ^ lambda-1)/lambda)) +
geom_histogram(aes(y = ..density..),
               colour = 1, fill = "white") +
  geom_density() +
  stat_function(fun = dnorm,
                args = list(mean = mean((training$price ^ lambda-1)/lambda),
                            sd = sd((training$price ^ lambda-1)/lambda)),
                col = "#1b98e0")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



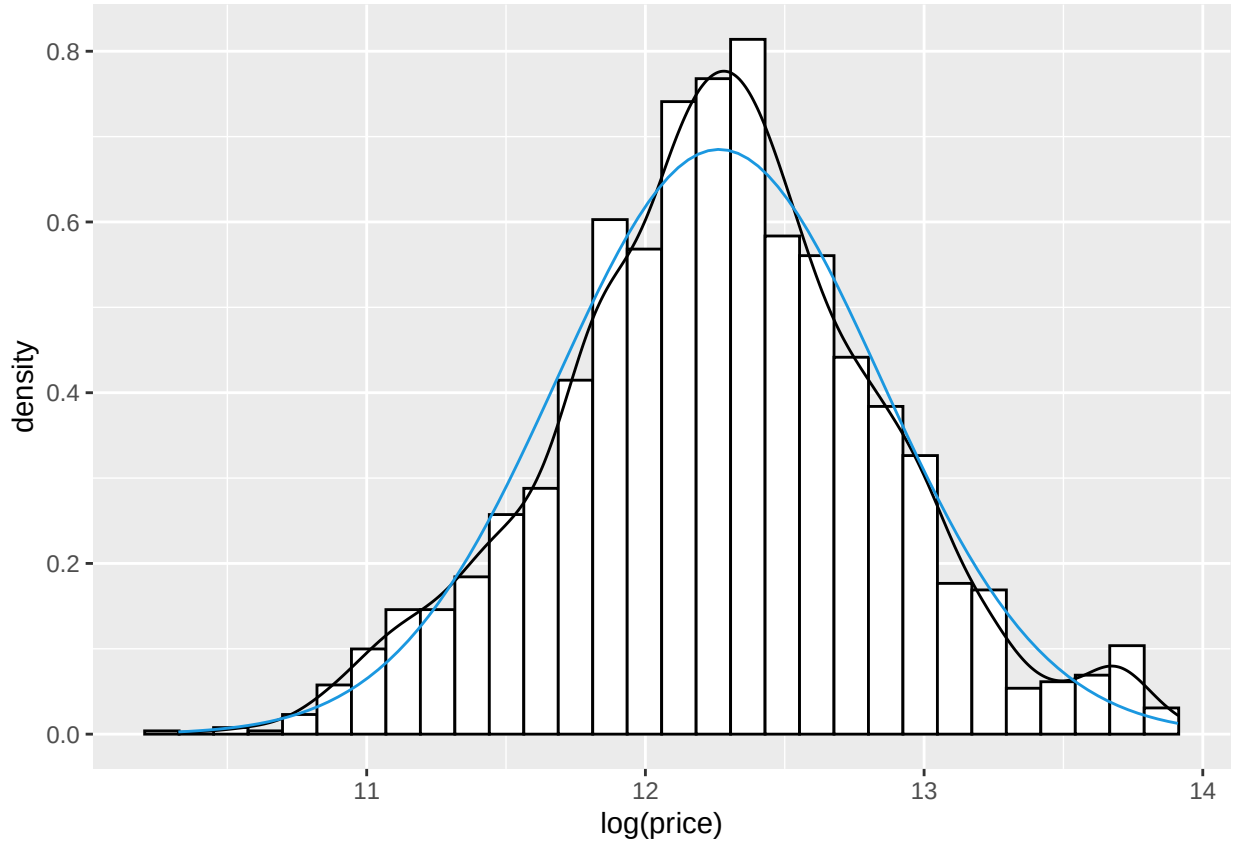

```
ggplot(training, aes(x = log(price))) +
  geom_histogram(aes(y = ..density..),
                 colour = 1, fill = "white") +
  geom_density() +
  stat_function(fun = dnorm,
                args = list(mean = mean(log(training$price)),
                            sd = sd(log(training$price))),
                col = "#1b98e0")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



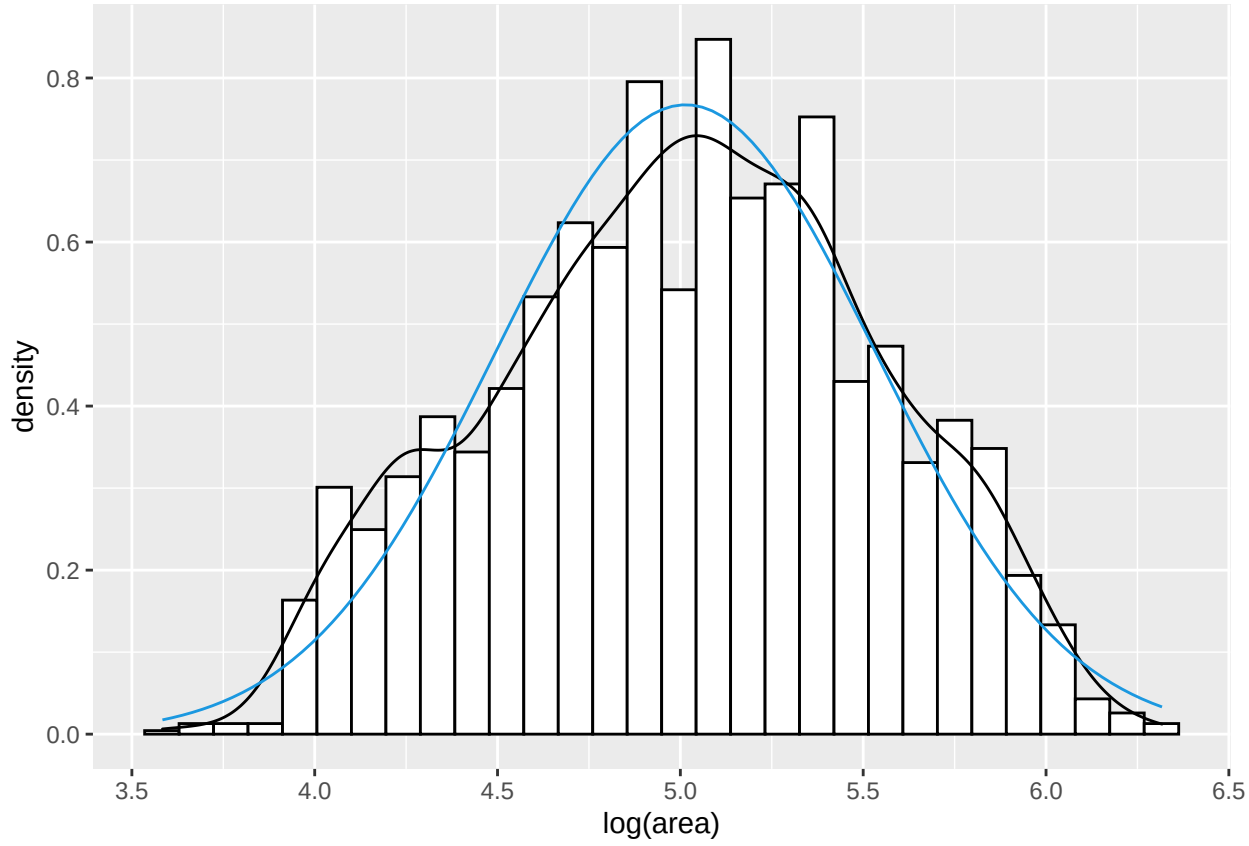

```
ggplot(trn, aes(x = log(price))) +
  geom_histogram(aes(y = ..density..),
                 colour = 1, fill = "white") +
  geom_density() +
  stat_function(fun = dnorm,
                args = list(mean = mean(log(trn$price)),
                            sd = sd(log(trn$price))),
                col = "#1b98e0")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



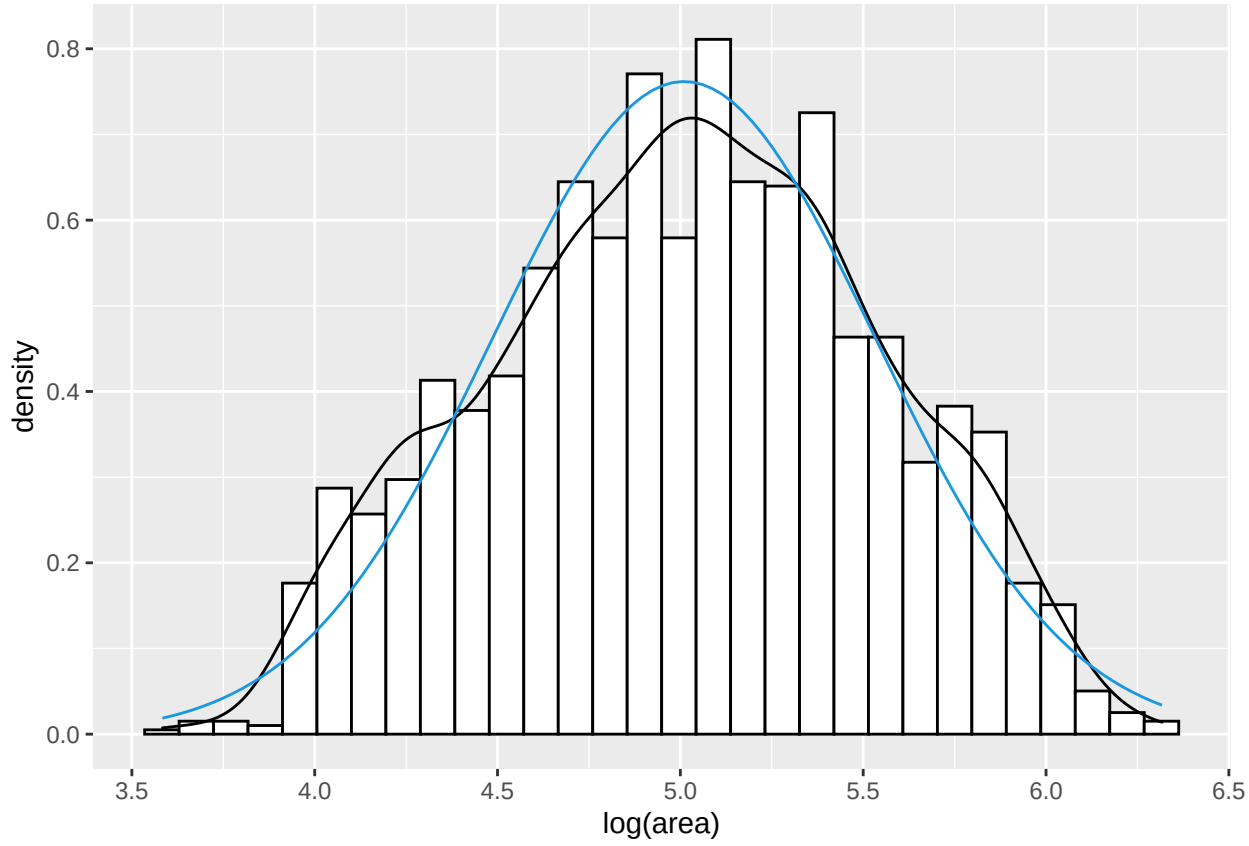

```
ggplot(training, aes(x = log(area))) +
  geom_histogram(aes(y = ..density..),
                 colour = 1, fill = "white") +
  geom_density() +
  stat_function(fun = dnorm,
                args = list(mean = mean(log(training$area)),
                            sd = sd(log(training$area))),
                col = "#1b98e0")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
ggplot(trn, aes(x = log(area))) +
  geom_histogram(aes(y = ..density..),
                 colour = 1, fill = "white") +
  geom_density() +
  stat_function(fun = dnorm,
                args = list(mean = mean(log(trn$area)),
                            sd = sd(log(trn$area))),
                col = "#1b98e0")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
dec_training <- training %>%
  mutate(decile = cut(log(price), breaks = seq(10, 20, length = 30))) %>%
  group_by(decile) %>%
  summarise(n = n())

dec_trn <- trn %>%
  mutate(decile = cut(log(price), breaks = seq(10, 20, length = 30))) %>%
  group_by(decile) %>%
  summarise(n = n())

dec_training %>%
  left_join(dec_trn, by = "decile") %>%
  mutate(d = n.x - n.y,
         d_per = (n.y / n.x - 1) * 100)
```

```
## # A tibble: 12 x 5
##    decile        n.x   n.y     d  d_per
##    <fct>       <int> <int> <int>  <dbl>
##  1 (10,10.3]       1     1     0   0
##  2 (10.3,10.7]     3     3     0   0
##  3 (10.7,11]      52    37    15 -28.8
##  4 (11,11.4]     123   108    15 -12.2
##  5 (11.4,11.7]   212   191    21  -9.91
##  6 (11.7,12.1]   448   396    52 -11.6
##  7 (12.1,12.4]   667   571    96 -14.4
```

```
##  8 (12.4,12.8]   481   397    84 -17.5
##  9 (12.8,13.1]   309   258    51 -16.5
## 10 (13.1,13.4]   100    82    18 -18
## 11 (13.4,13.8]    63    54     9 -14.3
## 12 (13.8,14.1]     8     8     0   0
```

```
shapiro.test(log(training$area))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  log(training$area)
## W = 0.99049, p-value = 0.00000000001059
```

Some more feature engineering

Working on location:

```
other_loc <- training %>%
  group_by(location) %>%
  summarise(n = n()) %>%
  filter(n < 10) %>%
  .$location

training <- training %>%
  mutate(location = ifelse(location %in% other_loc, "Other", location),
         location = factor(location))
```

`status` is an ordered categorical variable:

```
training <- training %>%
  mutate(status = factor(status, levels = c("En Planos", "En Construccion",
                                            "Nueva", "Remodelada", "A Remodelar",
                                            "Segundo Uso", "Fideicomiso")))
```

Transforming other character variables into factor variables: