Trabalho Estatistica FIAP Trabalho Final

Rodrigo de Miranda Videira

Contexto:

Este estudo é um caso de aplicação do método dos valores hedônicos, para valorar benefícios ambientais associados à proximidade a áreas verdes, existência de vista panorâmica e a localização da propriedade em rua com ou sem poluição sonora, relacionados a preços de apartamentos. O objetivo é contribuir aos estudos de valoração econômica do meio ambiente, propondo, para a análise em questão, a formulação de um modelo desenvolvido a partir de conceitos da engenharia de avaliações e associado ao meio ambiente, através de pesquisa na variação dos valores imobiliários.

Fonte: Marlene Salete Uberti;Norberto Hochheim. Valoração Ambiental: Estudo de Caso no Centro de Florianópolis.

```
#install.packages('caret', dependencies = TRUE)
#install.packages('gower', dependencies = TRUE)
#install.packages('parallelly', dependencies = TRUE)
#install.packages('psych', dependencies = TRUE)
```

Bibliotecas utilizadas

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.5     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.0.2     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(readr)
library(ggcorrplot)
library(ModelMetrics)
```

```
## Warning: package 'ModelMetrics' was built under R version 4.1.2
```

```
##
## Attaching package: 'ModelMetrics'
```

```
## The following object is masked from 'package:base':
##
##     kappa
```

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 4.1.2
```

```
## Carregando pacotes exigidos: lattice
```

```
##
## Attaching package: 'caret'
```

```
## The following objects are masked from 'package:ModelMetrics':
##
##     confusionMatrix, precision, recall, sensitivity, specificity
```

```
## The following object is masked from 'package:purrr':
##
##     lift
```

```
library(psych)
```

```
##
## Attaching package: 'psych'
```

```
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
library(readxl)
```

Carregando a base de dados para análise

```
df <- read_delim("Arquivo_Valorizacao_Ambiental_2.csv",
    delim = ";", escape_double = FALSE, trim_ws = TRUE, show_col_types = FALSE)
```

Análisando as variáveis presentes no dataset

```
names(df)
```

```
##  [1] "Ordem"    "Valor"    "Area"     "IA"       "Andar"    "Suites"
##  [7] "Vista"    "DistBM"   "Semruido" "AV100m"
```

```
str(df)
```

```
## spec_tbl_df [172 x 10] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ Ordem   : num [1:172] 1 2 3 4 5 6 7 8 9 10 ...
##  $ Valor   : num [1:172] 160000 67000 190000 110000 70000 75000 95000 135000 110000 115000
...
##  $ Area    : num [1:172] 168 129 218 180 120 160 155 165 150 185 ...
##  $ IA      : num [1:172] 1 1 1 12 15 18 5 1 10 15 ...
##  $ Andar   : num [1:172] 5 6 8 4 3 2 3 2 4 5 ...
##  $ Suites  : num [1:172] 1 0 1 1 1 0 1 1 1 1 ...
##  $ Vista   : num [1:172] 1 0 0 0 0 1 0 1 0 0 ...
##  $ DistBM  : num [1:172] 294 1505 251 245 956 ...
##  $ Semruido: num [1:172] 1 1 0 0 1 0 1 0 0 0 ...
##  $ AV100m  : num [1:172] 0 0 1 0 0 1 0 1 0 0 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   Ordem = col_double(),
##   ..   Valor = col_double(),
##   ..   Area = col_double(),
##   ..   IA = col_double(),
##   ..   Andar = col_double(),
##   ..   Suites = col_double(),
##   ..   Vista = col_double(),
##   ..   DistBM = col_double(),
##   ..   Semruido = col_double(),
##   ..   AV100m = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
#Ordem -> ID
#Valor -> Quantitativa discreta
#Area -> Quantitativa discreta
#IA -> Quantitativa discreta
#Andar -> Categórica ordinal
#Suites ->  Quantitativa discreta
#Vista -> Categórica Nominal
#DistBM -> Quantitativa discreta
#Semruido -> Categórica Nominal
#AV100m -> Categórica Nominal
```

Realizando a correção dos tipos categoricos

```
df$Andar = as.factor(df$Andar)
df$Vista = as.factor(df$Vista)
df$Semruido = as.factor(df$Semruido)
df$AV100m = as.factor(df$AV100m)

df$Ordem <- NULL

str(df)
```

```
## spec_tbl_df [172 x 9] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ Valor   : num [1:172] 160000 67000 190000 110000 70000 75000 95000 135000 110000 115000
...
##  $ Area    : num [1:172] 168 129 218 180 120 160 155 165 150 185 ...
##  $ IA      : num [1:172] 1 1 1 12 15 18 5 1 10 15 ...
##  $ Andar   : Factor w/ 12 levels "1","2","3","4",..: 5 6 8 4 3 2 3 2 4 5 ...
##  $ Suites  : num [1:172] 1 0 1 1 1 0 1 1 1 1 ...
##  $ Vista   : Factor w/ 2 levels "0","1": 2 1 1 1 1 2 1 2 1 1 ...
##  $ DistBM  : num [1:172] 294 1505 251 245 956 ...
##  $ Semruido: Factor w/ 2 levels "0","1": 2 2 1 1 2 1 2 1 1 1 ...
##  $ AV100m  : Factor w/ 2 levels "0","1": 1 1 2 1 1 2 1 2 1 1 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   Ordem = col_double(),
##   ..   Valor = col_double(),
##   ..   Area = col_double(),
##   ..   IA = col_double(),
##   ..   Andar = col_double(),
##   ..   Suites = col_double(),
##   ..   Vista = col_double(),
##   ..   DistBM = col_double(),
##   ..   Semruido = col_double(),
##   ..   AV100m = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

Realizando análises estátisticas das variáveis:

- Area (Quantitativa)

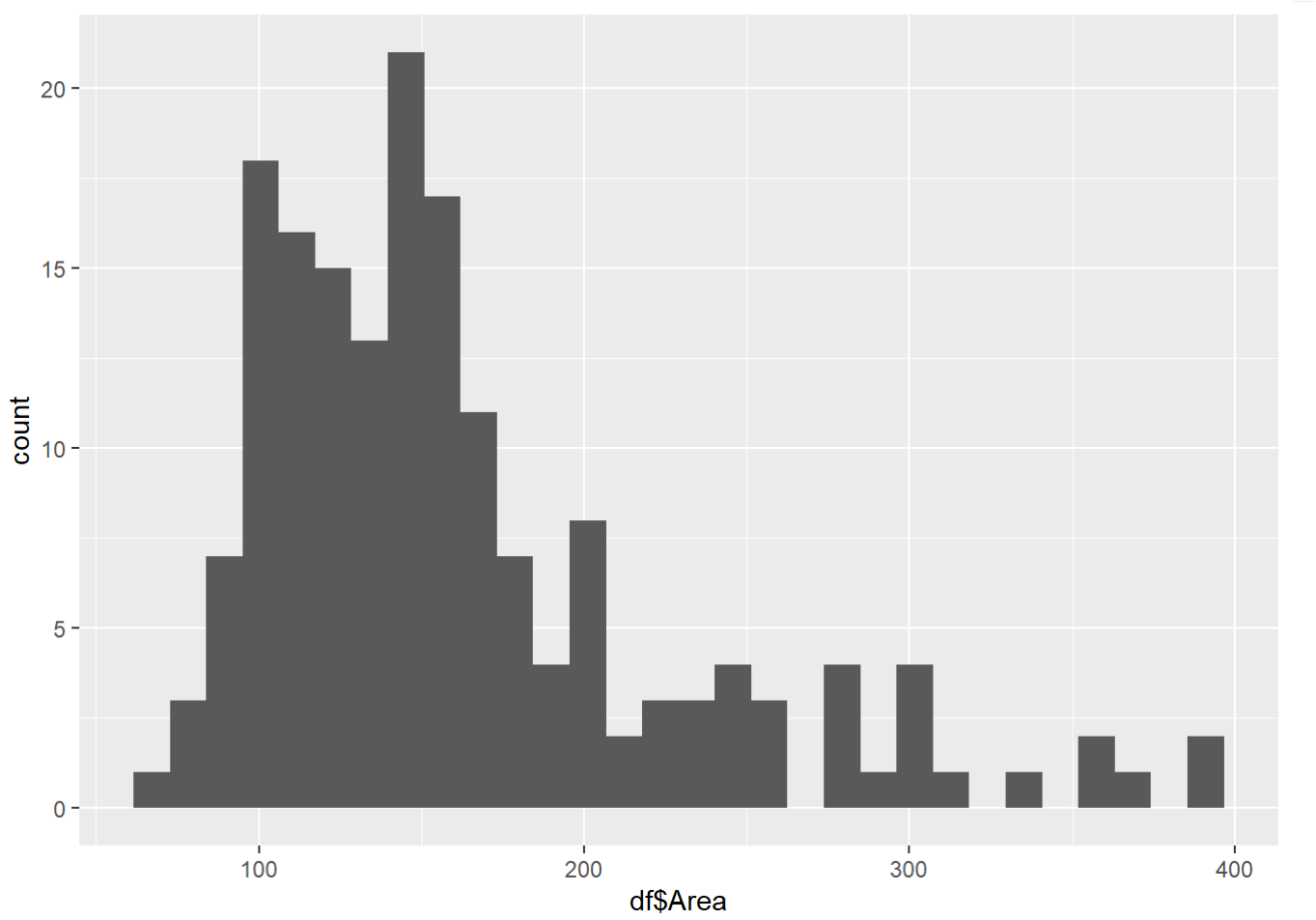
```
summary(df$Area)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    69.0   117.0   145.0   163.2   182.0   393.0
```

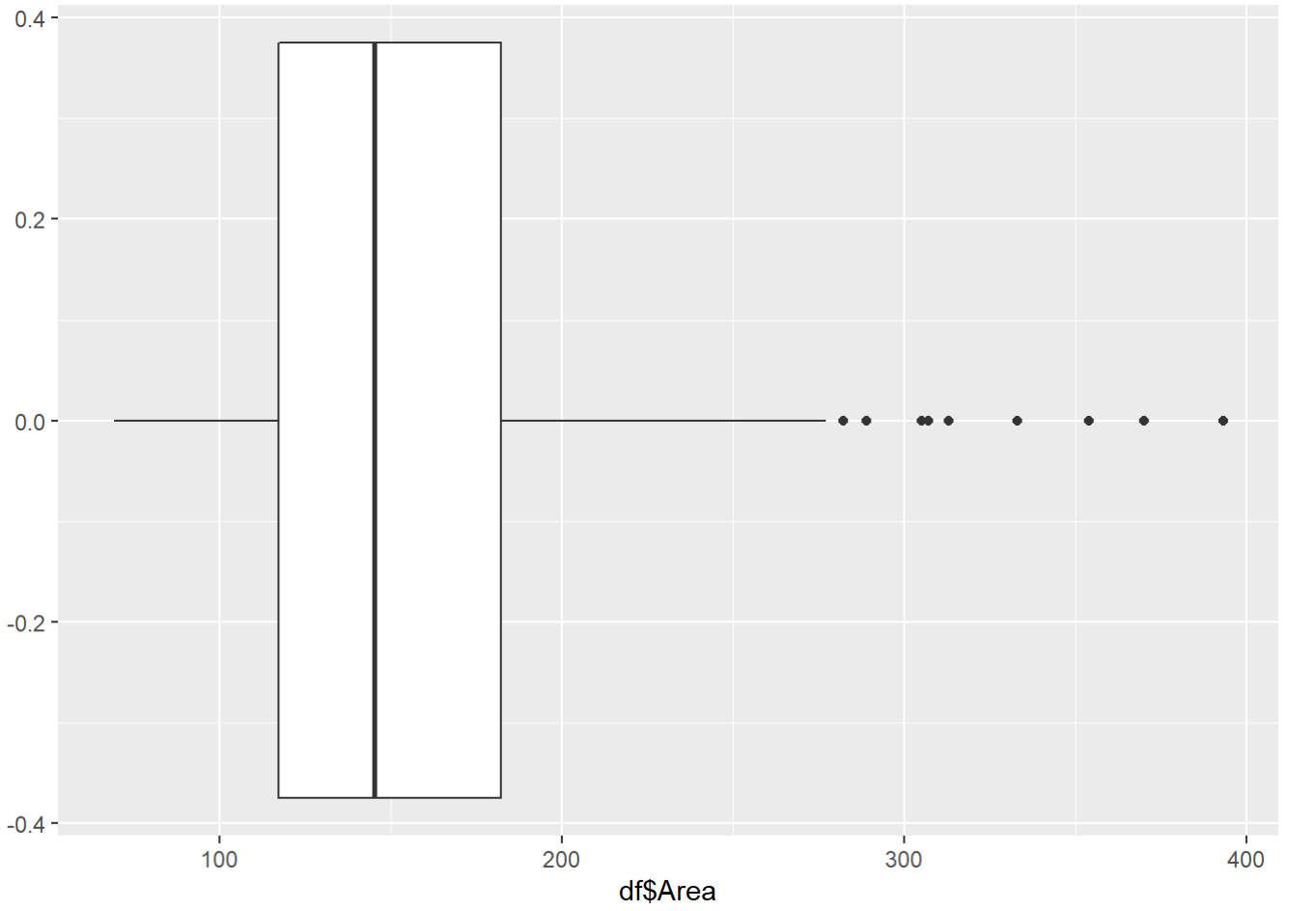
```
# Tirando os quartils Q1 e Q3 para análise de outliers
Area_Q1 = quantile(df$Area, 0.25)
Area_Q3 = quantile(df$Area, 0.75)
Area_IQR = Area_Q3 - Area_Q1
```

Gráficos Area

```
ggplot(df, mapping = aes(x = df$Area)) +
  geom_histogram(bins = 30)
```




```
ggplot(df, mapping = aes(x = df$Area)) +
  geom_boxplot()
```



Pelos gráficos e valores apurados, a variável “Area” possui:

Média: 163,2 Mediana: 145

Como a média é maior que a mediana, e também pelo histograma os dados possuem assimetria a direita
Também pelo gráfico de boxplot verificamos alguns outliers à direita, dados com (Area_Q3 + 1.5 * Area_IQR)

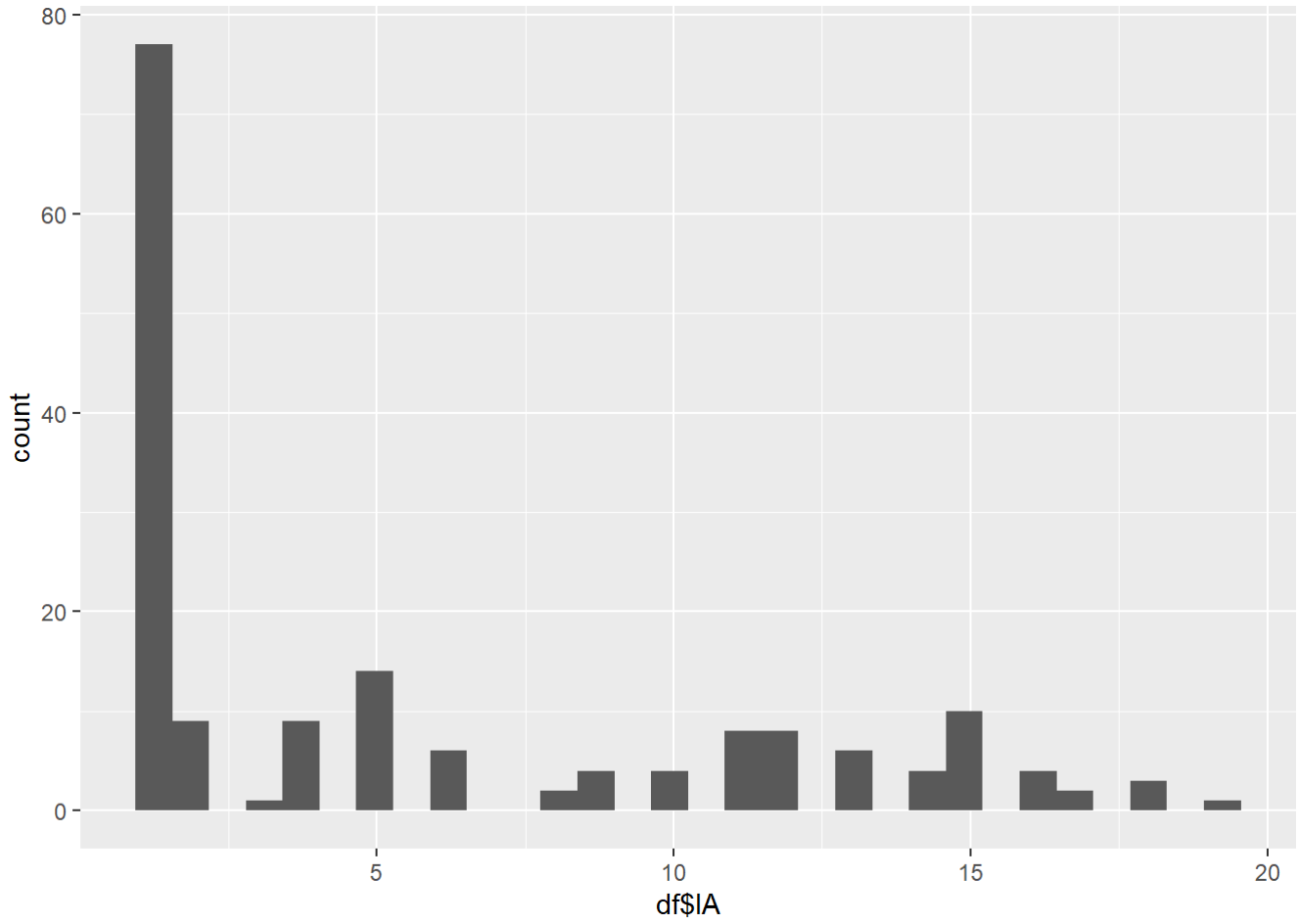
- IA (Quantitativa)

```
summary(df$IA)
```

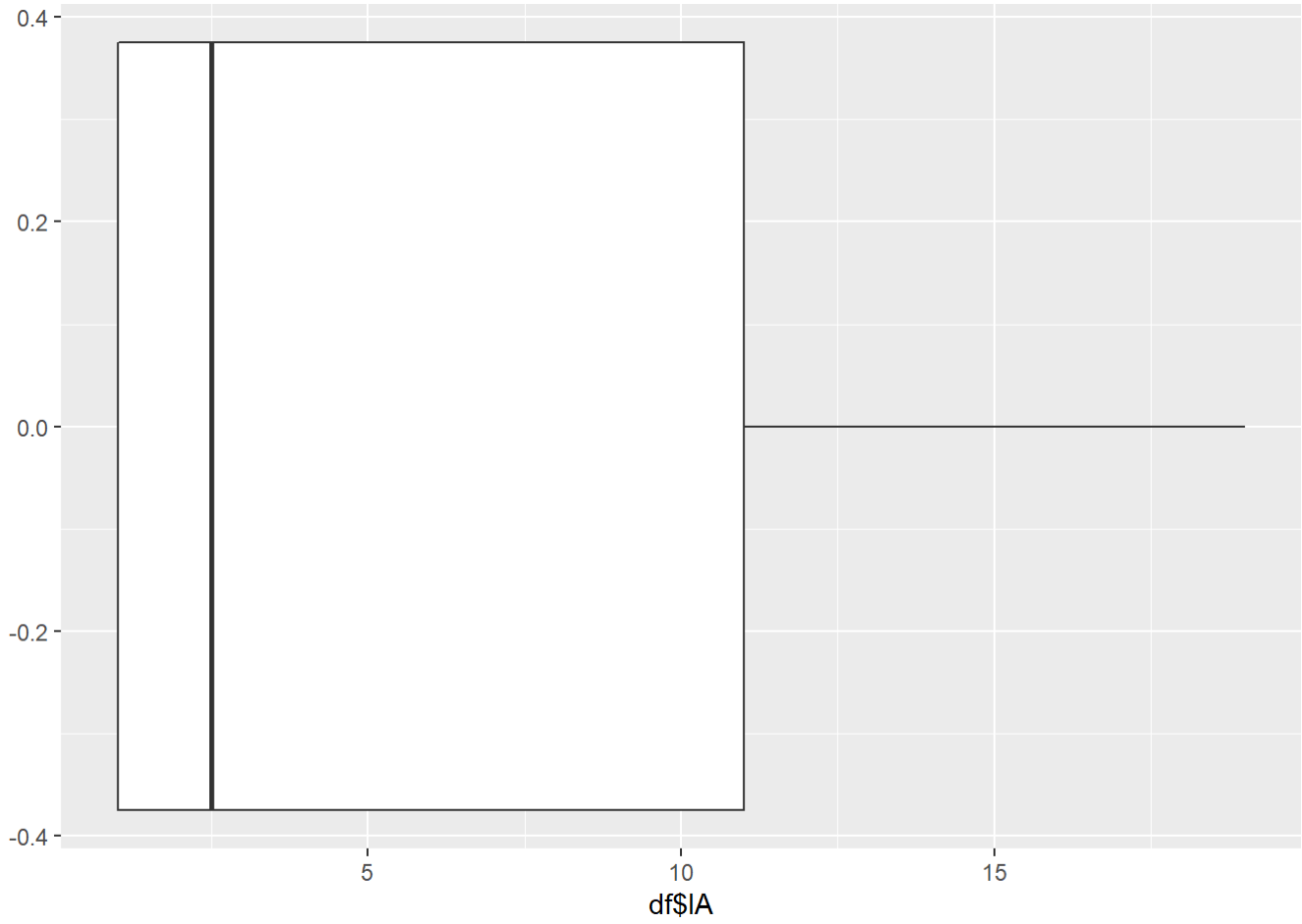
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   1.000   2.500   5.645  11.000  19.000
```

Gráficos IA

```
ggplot(df, mapping = aes(x = df$IA)) +
  geom_histogram(bins = 30)
```




```
ggplot(df, mapping = aes(x = df$IA)) +
  geom_boxplot()
```



Pelos gráficos e valores apurados, a variável “IA” possui:

Média: 5,65 Mediana: 2,5

Como a média é maior que a mediana, e também pelo histograma os dados possuem assimetria a direita Não identificamos outliers, os apartamentos são considerados novos com idades entre 0 a 19 anos.

- DistBM (Quantitativa)

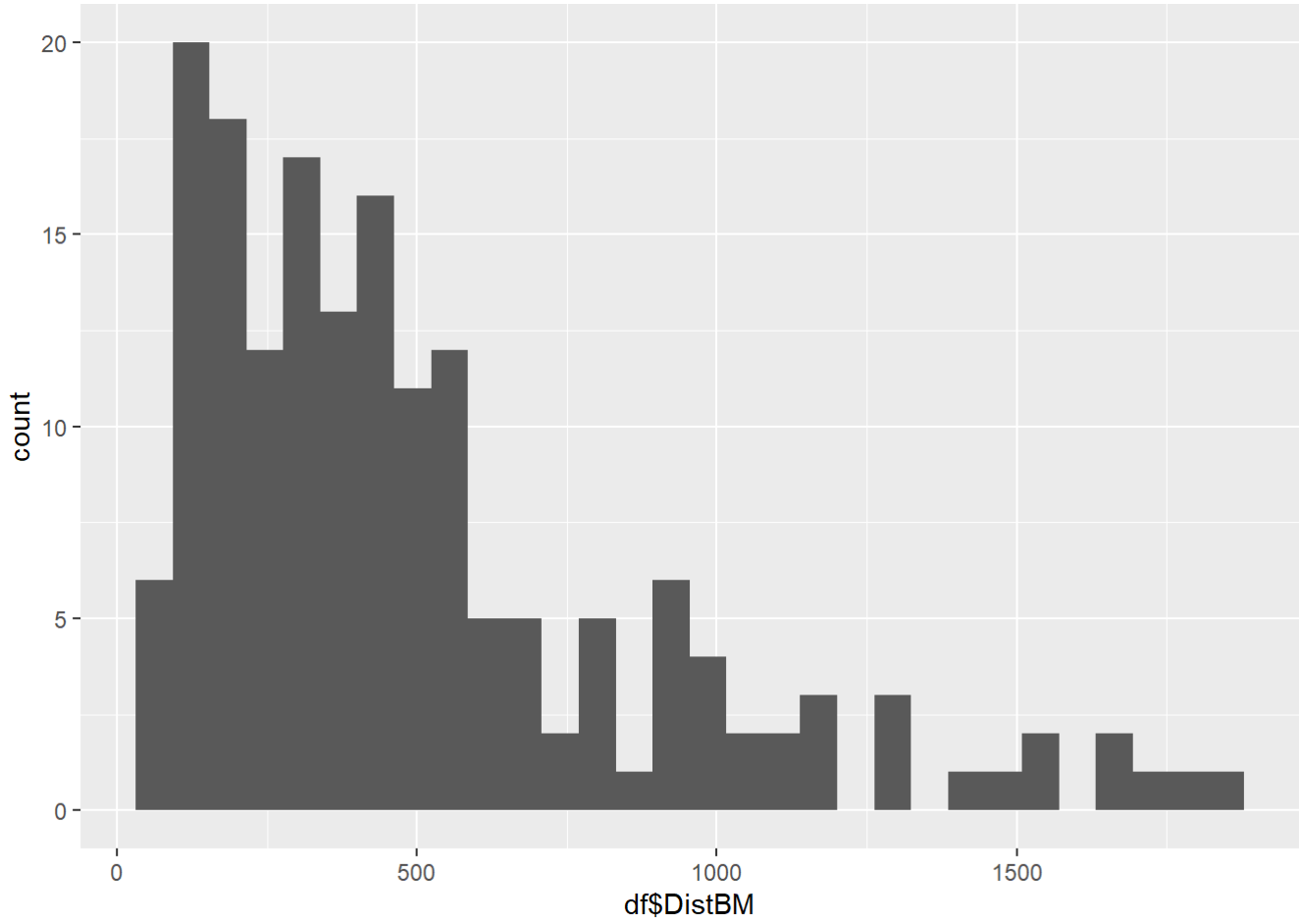
```
summary(df$DistBM)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    73.0   214.8   402.5   505.9   638.0  1859.0
```

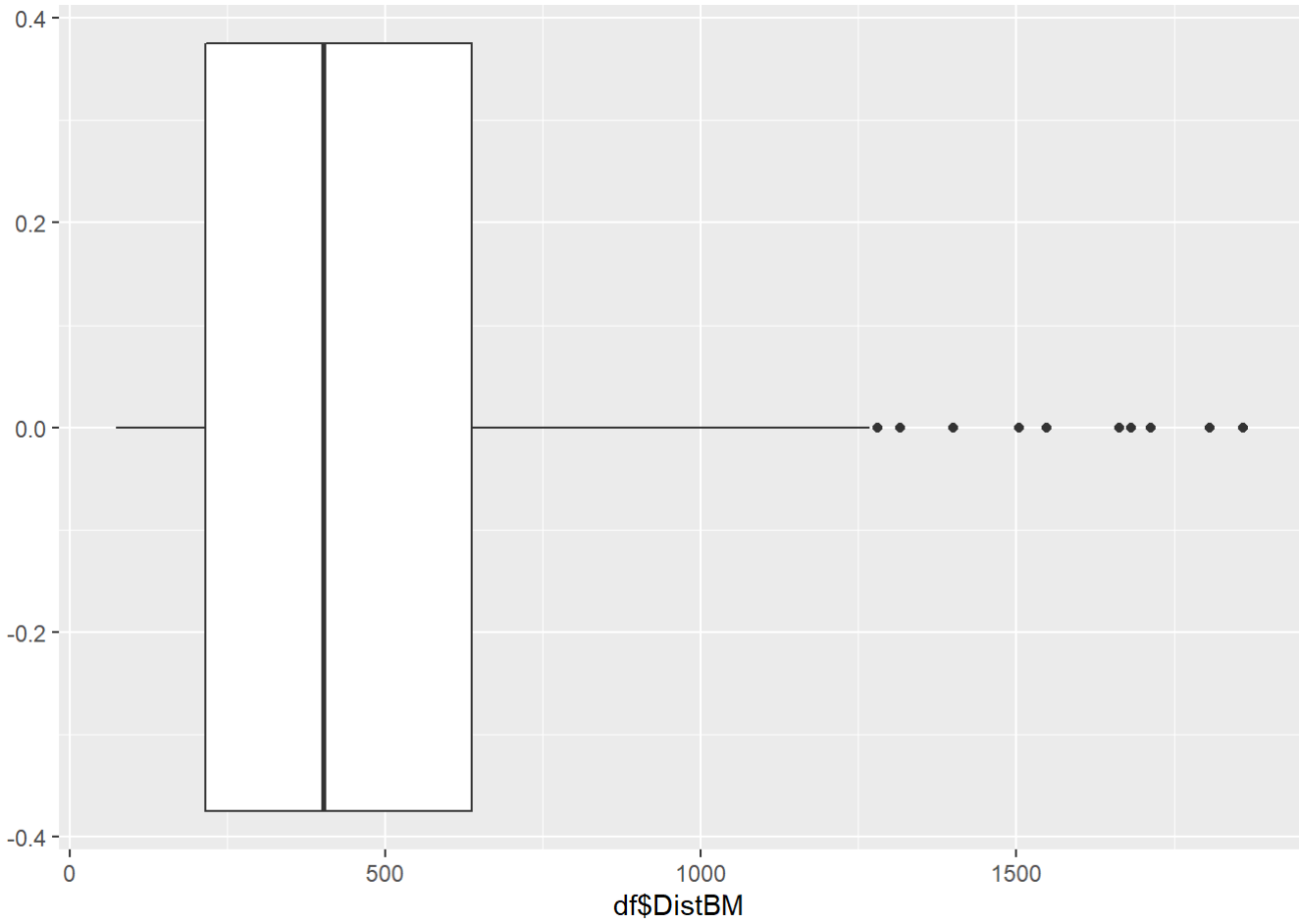
```
# Tirando os quartils Q1 e Q3 para análise de outliers
DistBM_Q1 = quantile(df$DistBM, 0.25)
DistBM_Q3 = quantile(df$DistBM, 0.75)
DistBM_IQR = DistBM_Q3 - DistBM_Q1
```

Gráficos DistBM

```
ggplot(df, mapping = aes(x = df$DistBM)) +
  geom_histogram(bins = 30)
```




```
ggplot(df, mapping = aes(x = df$DistBM)) +
  geom_boxplot()
```



Pelos gráficos e valores apurados, a variável “DistBM” possui:

Média: 505,90 Mediana: 402,5

Como a média é maior que a mediana, e também pelo histograma os dados possuem assimetria a direita Também pelo gráfico de boxplot verificamos alguns outliers à direita, dados com (DistBM_Q3 + 1.5 * DistBM_IQR)

Análisando algumas variáveis qualitativas

- Vista
- Semruido
- AV100m

```
vista_tabela <- table(df$Vista);vista_tabela
```

```
##
##   0   1
## 148  24
```

```
perc_sem_vista <- vista_tabela[1] / sum(vista_tabela) * 100; perc_sem_vista
```

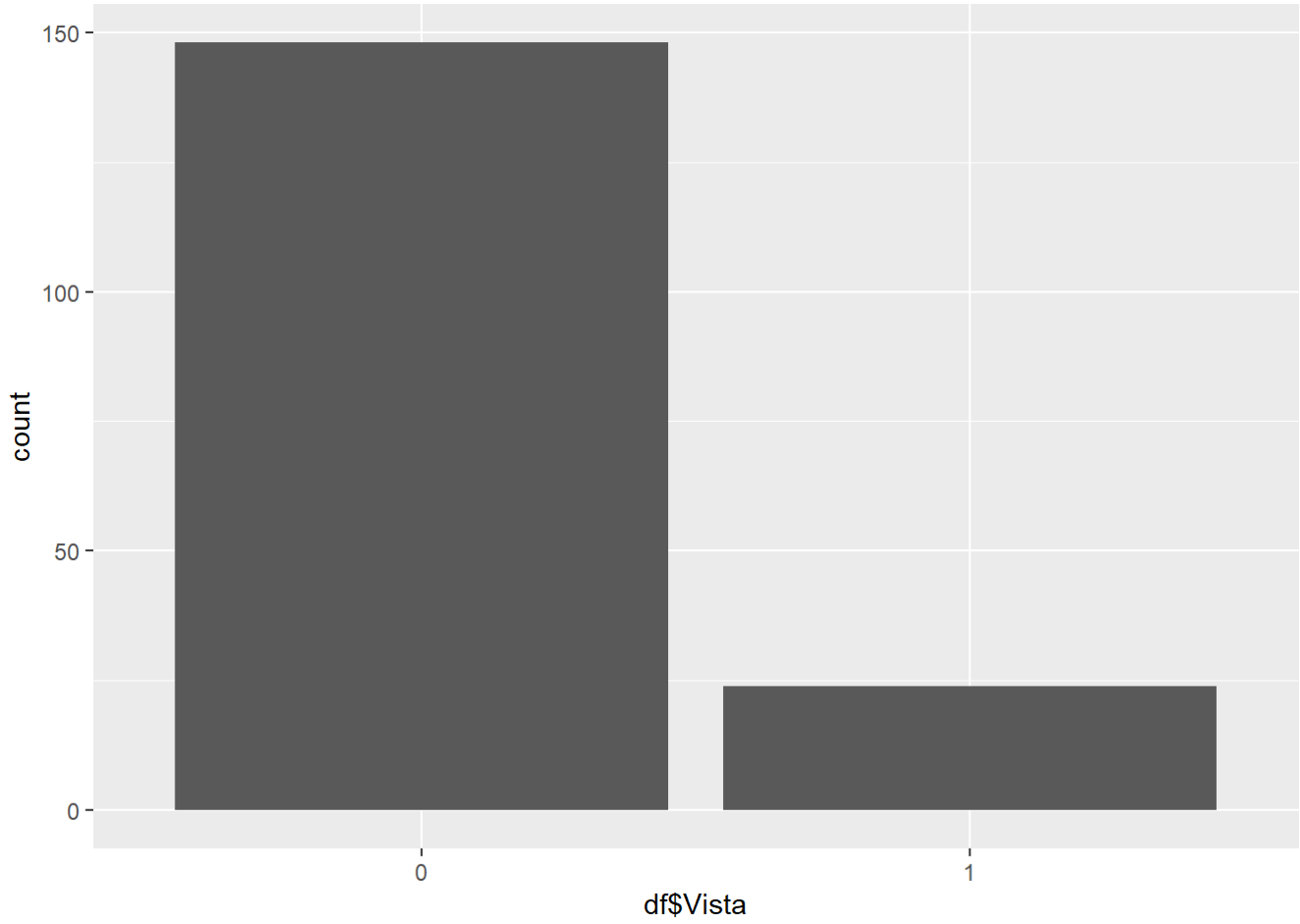
```
##        0
## 86.04651
```

```
perc_com_vista <- 100 - perc_sem_vista;perc_com_vista
```

```
##        0
## 13.95349
```

- 86,04% de nossos apartamentos não possuem vista panoramica e 13,95% possuem.

```
ggplot(df, mapping = aes(x = df$Vista)) +
  geom_bar()
```




```
ruido_tabela <- table(df$Semruido);ruido_tabela
```

```
##
##   0   1
##  72 100
```

```
perc_sem_ruido <- ruido_tabela[1] / sum(ruido_tabela) * 100; perc_sem_ruido
```

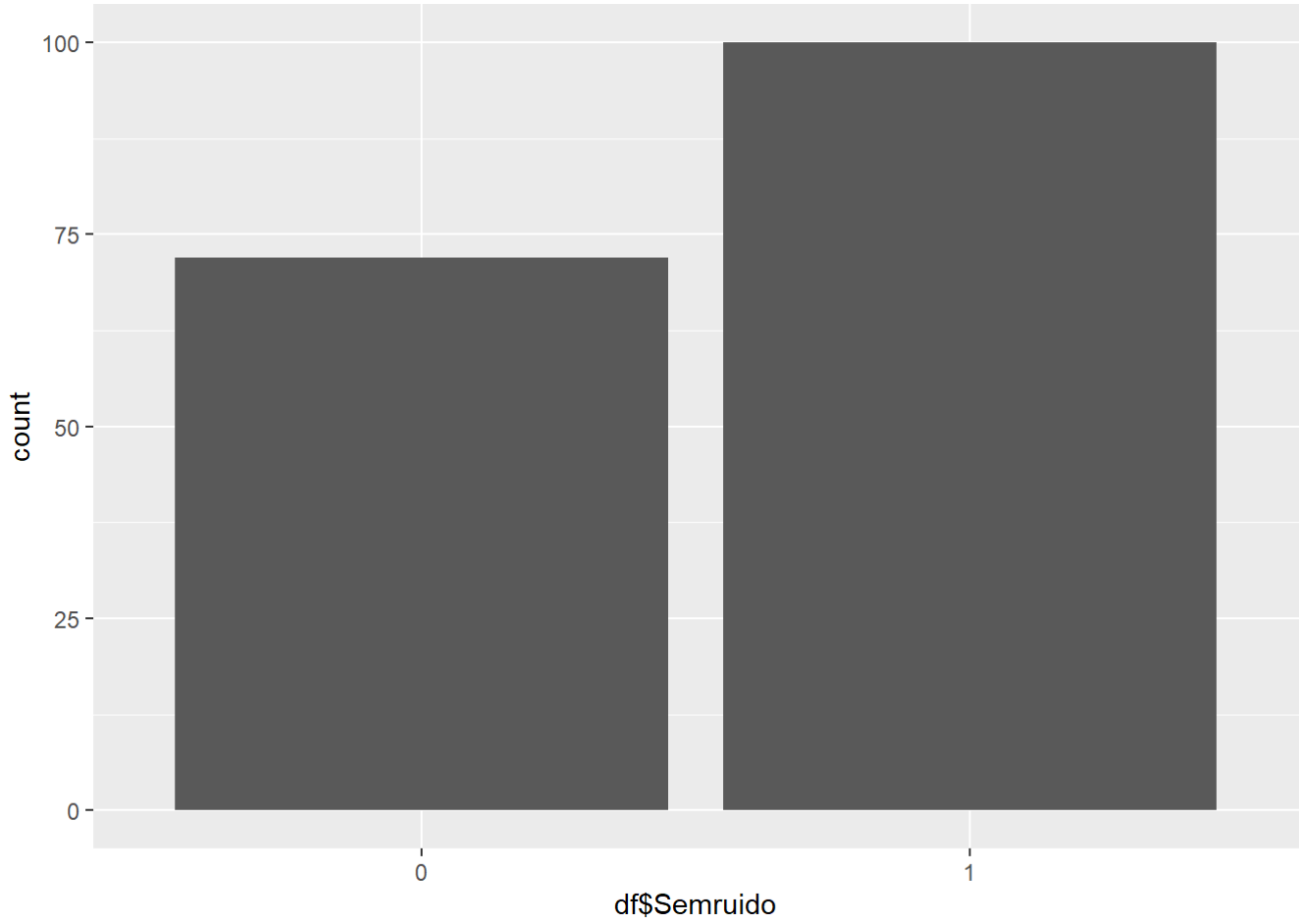
```
##        0
## 41.86047
```

```
perc_com_ruido <- 100 - perc_sem_ruido;perc_com_ruido
```

```
##        0
## 58.13953
```

- 41,86% de nossos apartamentos estão localizados em ruas que possuem muito ruido e 58,13% estão em áreas mais tranquilas

```
ggplot(df, mapping = aes(x = df$Semruido)) +
  geom_bar()
```




```
av100m_tabela <- table(df$AV100m);av100m_tabela
```

```
##
##   0   1
## 112  60
```

```
perc_prox_area_verde <- av100m_tabela[1] / sum(av100m_tabela) * 100; perc_prox_area_verde
```

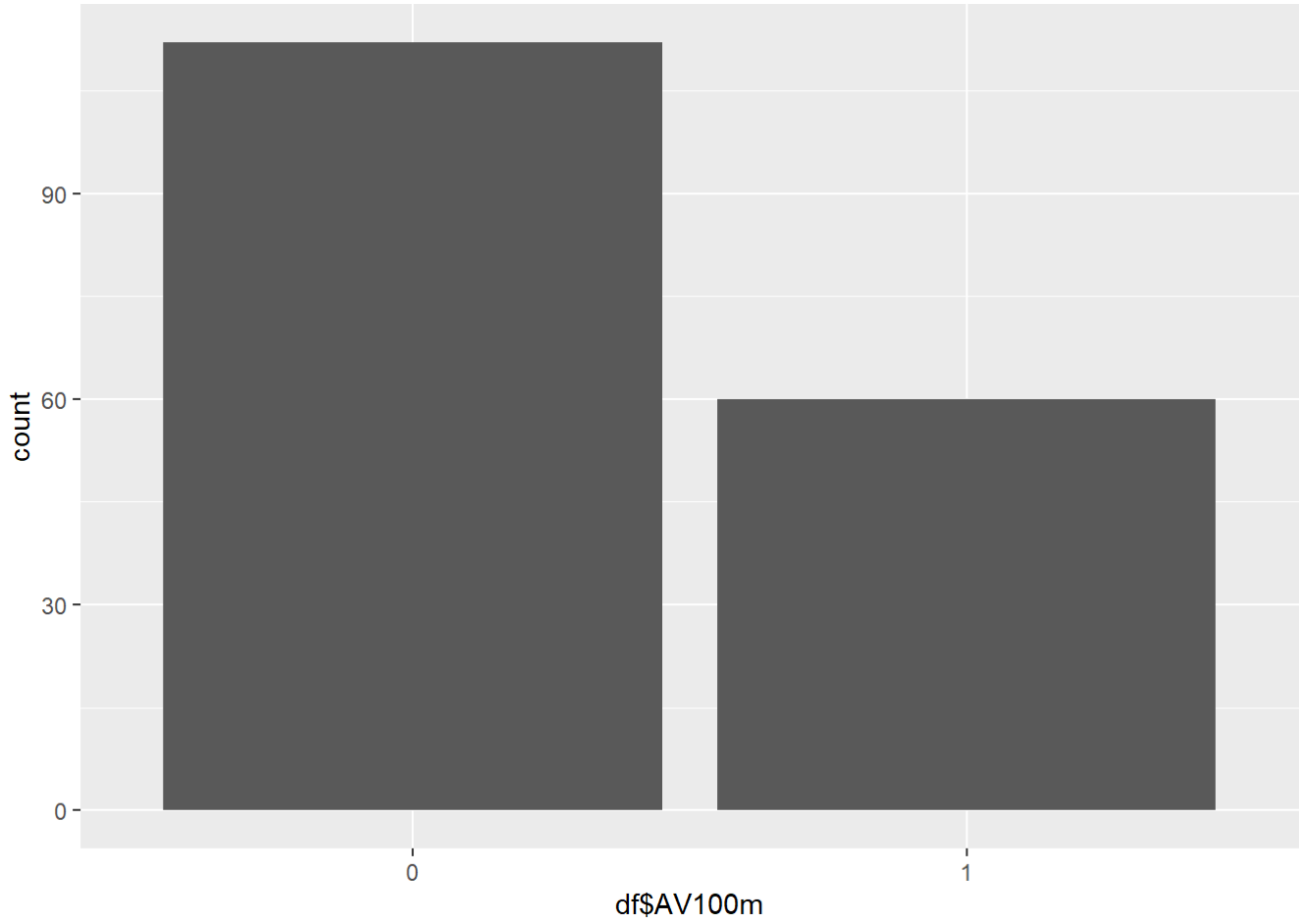
```
##        0
## 65.11628
```

```
perc_longe_area_verde <- 100 - perc_prox_area_verde;perc_longe_area_verde
```

```
##        0
## 34.88372
```

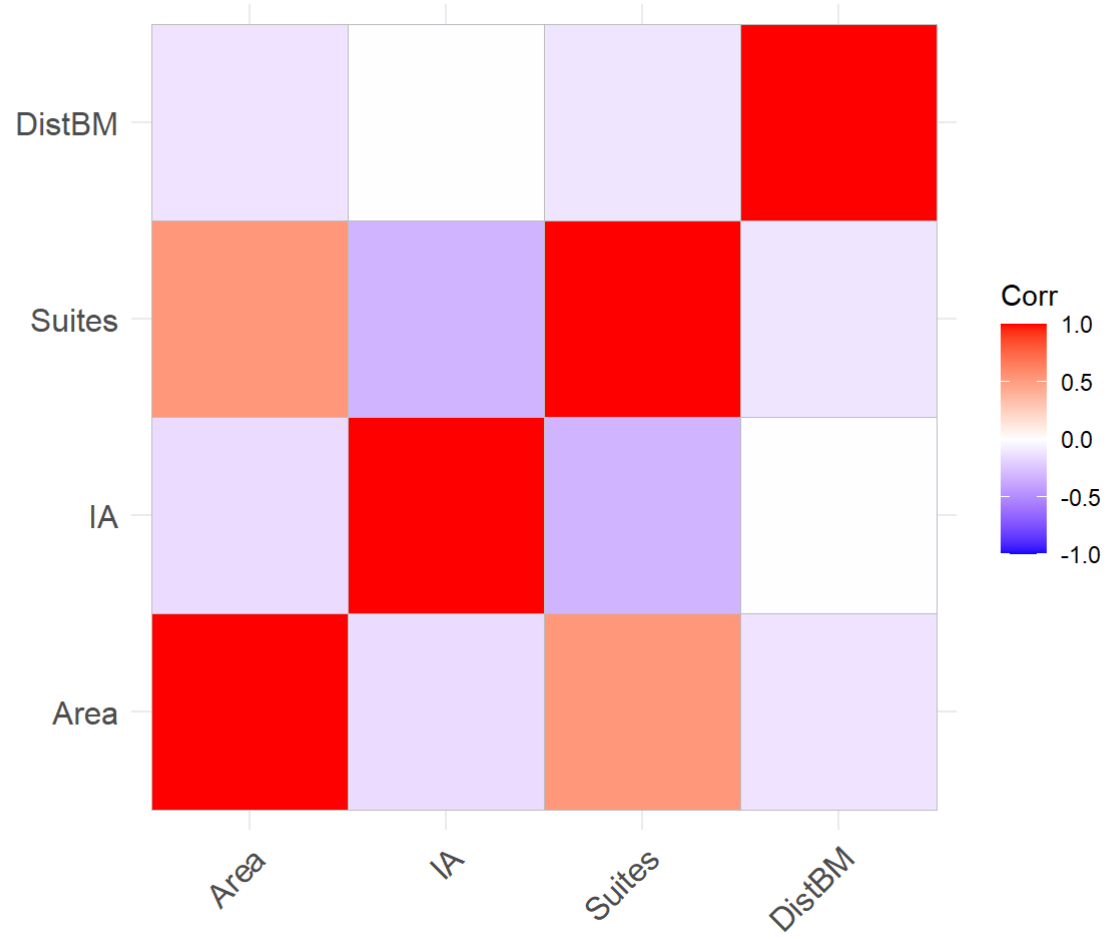
- 65,11% de nossos apartamentos estão localizados a areas verdes como por exemplo de praças e 34,88% estão em áreas que não possuem áres verdes proximas.

```
ggplot(df, mapping = aes(x = df$AV100m)) +
  geom_bar()
```



Realizando análise de correlações das variáveis quantitativas

```
df_numericos <- select_if(df, is.numeric)
df_numericos$Valor <- NULL
correl <-cor(df_numericos)
ggcorrplot(correl)
```



Pelo gráfico as variáveis não possuem uma correlação muito forte. Não estão fortemente corelacionadas.

```
cor(df_numericos)
```

```
##              Area           IA     Suites       DistBM
## Area    1.0000000 -0.153888982  0.5277768 -0.123760011
## IA     -0.1538890  1.000000000 -0.3194795 -0.009441705
## Suites  0.5277768 -0.319479493  1.0000000 -0.113931139
## DistBM -0.1237600 -0.009441705 -0.1139311  1.000000000
```

Filtrando os dados para tirar outliers

```
df_sem_outlier <- filter(df,
                         df$Area < (Area_Q3 + 1.5 * Area_IQR),
                        df$DistBM < (DistBM_Q3 + 1.5 * DistBM_IQR)
                         )
```

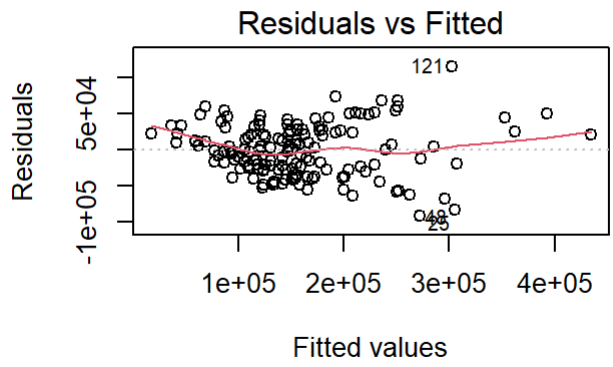
Criando o modelo de regressão linear

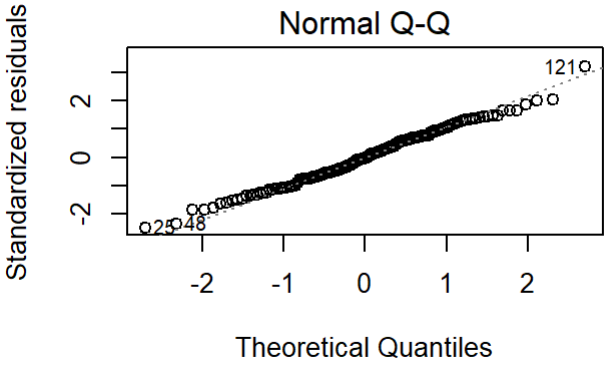
```
modelo_1 <- lm(Valor ~ ., data = df_sem_outlier)
```

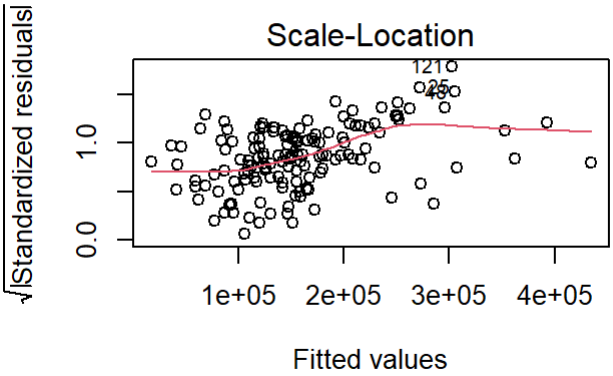
Análisando a performance do modelo

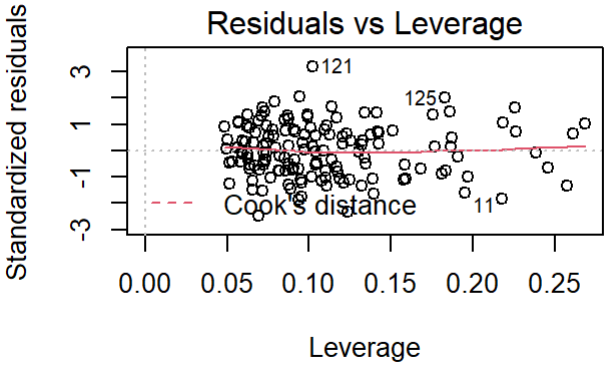
```
par(mfrow=c(2,2))
plot(modelo_1)
```

```
## Warning: not plotting observations with leverage one:
##   12, 18, 132
```









Testando a normalidade dos resíduos.

Ho: distribuição dos dados = normal -> p > 0.05 H1: distribuição dos dados <> normal -> p < 0.05

```
shapiro.test(modelo_1$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  modelo_1$residuals
## W = 0.99277, p-value = 0.6616
```

Escolhendo variáveis atráves do stepAIC

```
mod.simples  <- lm(Valor ~ 1, data = df_sem_outlier)
stepAIC(modelo_1, scope = list(upper = modelo_1,
                               lower = mod.simples, direction = "backward"))
```

```
## Start:  AIC=3141.27
## Valor ~ Area + IA + Andar + Suites + Vista + DistBM + Semruido +
##     AV100m
##
##            Df  Sum of Sq        RSS    AIC
## - Andar    11 1.9403e+10 2.0845e+11 3133.7
## - AV100m    1 3.2833e+08 1.8937e+11 3139.5
## - DistBM    1 5.5131e+08 1.8960e+11 3139.7
## <none>                   1.8904e+11 3141.3
## - IA        1 4.0166e+09 1.9306e+11 3142.4
## - Semruido  1 6.0041e+09 1.9505e+11 3143.9
## - Vista     1 1.1979e+10 2.0102e+11 3148.4
## - Suites    1 1.0442e+11 2.9347e+11 3204.4
## - Area      1 1.1163e+11 3.0067e+11 3207.9
##
## Step:  AIC=3133.73
## Valor ~ Area + IA + Suites + Vista + DistBM + Semruido + AV100m
##
##            Df  Sum of Sq        RSS    AIC
## - AV100m    1 1.2662e+09 2.0971e+11 3132.6
## <none>                   2.0845e+11 3133.7
## - Semruido  1 2.9904e+09 2.1144e+11 3133.8
## - DistBM    1 3.7282e+09 2.1218e+11 3134.4
## - IA        1 1.2313e+10 2.2076e+11 3140.2
## - Vista     1 1.3225e+10 2.2167e+11 3140.8
## + Andar    11 1.9403e+10 1.8904e+11 3141.3
## - Area      1 1.2087e+11 3.2932e+11 3199.4
## - Suites    1 1.5057e+11 3.5902e+11 3212.2
##
## Step:  AIC=3132.63
## Valor ~ Area + IA + Suites + Vista + DistBM + Semruido
##
##            Df  Sum of Sq        RSS    AIC
## <none>                   2.0971e+11 3132.6
## - DistBM    1 3.9061e+09 2.1362e+11 3133.4
## - Semruido  1 4.0246e+09 2.1374e+11 3133.4
## + AV100m    1 1.2662e+09 2.0845e+11 3133.7
## - IA        1 1.2044e+10 2.2176e+11 3138.9
## + Andar    11 2.0341e+10 1.8937e+11 3139.5
## - Vista     1 1.8500e+10 2.2821e+11 3143.1
## - Area      1 1.3584e+11 3.4556e+11 3204.5
## - Suites    1 1.5394e+11 3.6366e+11 3212.1
```

```
##
## Call:
## lm(formula = Valor ~ Area + IA + Suites + Vista + DistBM + Semruido,
##     data = df_sem_outlier)
##
## Coefficients:
## (Intercept)         Area           IA       Suites       Vista1       DistBM
##   -14256.31       772.63     -1811.48     36654.73     33742.30        19.49
##   Semruido1
##    11422.69
```

Criando o modelo final com as variáveis selecionadas pelo metodo stepAIC

```
modelo_2  <- lm(formula = Valor ~ Area + IA + Suites + Vista + DistBM + Semruido,
                data = df_sem_outlier)
```

Comparando os modelos

```
summary(modelo_1)
```

```
##
## Call:
## lm(formula = Valor ~ ., data = df_sem_outlier)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -91771 -26235    -74  25913 115859
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -34592.400  19263.219  -1.796  0.07487 .
## Area           740.684     84.867   8.728 1.14e-14 ***
## IA           -1207.887    729.598  -1.656  0.10024
## Andar2       17224.471  13597.137   1.267  0.20752
## Andar3       30195.466  13604.420   2.220  0.02820 *
## Andar4       38737.878  15517.220   2.496  0.01380 *
## Andar5       29154.541  15586.073   1.871  0.06367 .
## Andar6       42464.314  16427.823   2.585  0.01085 *
## Andar7       18172.681  19739.690   0.921  0.35897
## Andar8       55839.742  21748.793   2.567  0.01138 *
## Andar9       34121.692  21173.492   1.612  0.10951
## Andar10       7427.186  40380.350   0.184  0.85436
## Andar11      49922.340  42783.975   1.167  0.24542
## Andar12      -4021.770  41603.002  -0.097  0.92314
## Suites       33175.834   3930.169   8.441 5.56e-14 ***
## Vista1       30215.943  10568.441   2.859  0.00496 **
## DistBM           8.351     13.616   0.613  0.54072
## Semruido1    15252.181   7535.175   2.024  0.04502 *
## AV100m1       4116.069   8695.897   0.473  0.63677
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 38280 on 129 degrees of freedom
## Multiple R-squared:  0.7948, Adjusted R-squared:  0.7661
## F-statistic: 27.75 on 18 and 129 DF,  p-value: < 2.2e-16
```

```
summary(modelo_2)
```

```
##
## Call:
## lm(formula = Valor ~ Area + IA + Suites + Vista + DistBM + Semruido,
##     data = df_sem_outlier)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -83001 -26524  -2515  26681 116486
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -14256.31   14623.38  -0.975 0.331280
## Area           772.63      80.85   9.557  < 2e-16 ***
## IA           -1811.48     636.59  -2.846 0.005094 **
## Suites       36654.73    3602.89  10.174  < 2e-16 ***
## Vista1       33742.30    9567.27   3.527 0.000568 ***
## DistBM          19.49      12.03   1.621 0.107342
## Semruido1    11422.69    6944.03   1.645 0.102205
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 38570 on 141 degrees of freedom
## Multiple R-squared:  0.7723, Adjusted R-squared:  0.7626
## F-statistic: 79.72 on 6 and 141 DF,  p-value: < 2.2e-16
```

O “modelo_2” foi escolhido por apresentar um R-squared ajustado próximo ao do modelo 1, só que possui menos variáves

```
modelo_2
```

```
##
## Call:
## lm(formula = Valor ~ Area + IA + Suites + Vista + DistBM + Semruido,
##     data = df_sem_outlier)
##
## Coefficients:
## (Intercept)         Area           IA       Suites       Vista1       DistBM
##   -14256.31       772.63     -1811.48     36654.73     33742.30        19.49
##   Semruido1
##    11422.69
```

Prevendo valores da base “Estimar_Valor_Imoveis”

```
# Carregando os dados
ds_estimar_valores <- read_excel("Estimar_Valor_Imoveis.xlsx")
ds_estimar_valores
```

Apartamento <dbl>	Área (m2) <dbl>	IA <dbl>	An… <dbl>	Suítes <dbl>	Vista <chr>	Dist. BM <dbl>	ruído aceitável <chr>	AV 100m <chr>	Valo
1	168	2	9	0	Não	150	NÃO	Sim	
2	130	4	6	1	Sim	2000	SIM	Sim	

Apartamento <dbl>	Área (m2) <dbl>	IA <dbl>	An… <dbl>	Suítes <dbl>	Vista <chr>	Dist. BM <dbl>	ruído aceitável <chr>	AV 100m <chr>	Valo
3	218	1	8	4	Não	251	NÃO	Sim	
4	180	2	4	0	Não	245	NÃO	Não	
5	120	4	3	0	Sim	956	SIM	Não	
6	160	2	2	1	Não	85	NÃO	Não	
7	155	5	3	0	Sim	1600	NÃO	Sim	
8	165	1	2	1	Não	148	SIM	Não	
9	150	10	4	1	Sim	143	SIM	Sim	

9 rows

```
#Alterando nome das colunas
names(ds_estimar_valores) <- c("Ordem","Area","IA","Andar","Suites","Vista","DistBM","Semruid
o","AV100m","Valor_Predito")

#Alterando as colunas de Sim ou não para 0 e 1
ds_estimar_valores$Vista <- ifelse(ds_estimar_valores$Vista == "Sim", 1,0)
ds_estimar_valores$Semruido <- ifelse(ds_estimar_valores$Semruido == "SIM", 1,0)
ds_estimar_valores$AV100m <- ifelse(ds_estimar_valores$AV100m == "Sim", 1,0)

#Ajustando as colunas para fator
ds_estimar_valores$Andar = as.factor(ds_estimar_valores$Andar)
ds_estimar_valores$Vista = as.factor(ds_estimar_valores$Vista)
ds_estimar_valores$Semruido = as.factor(ds_estimar_valores$Semruido)
ds_estimar_valores$AV100m = as.factor(ds_estimar_valores$AV100m)

#Prevendo o valor do apartamento utilizando o modelo_2
ds_estimar_valores$Valor_Predito <- predict(modelo_2, ds_estimar_valores, type = 'response');


write_excel_csv2(ds_estimar_valores, file="Estimar_Valor_Imoveis_Com_Valor_Predito.csv")
view(ds_estimar_valores)
```