Estatística Aplicada II

Primeira Lista de Exercícios

Estudante: Clístenes Grizafis Bento

Com a base de dados "imoveiscwbav" obter os seguintes resultados com o auxílio do "R"

Estimar três modelos (Ridge, Lasso e Elasticnet) para explicar a variável Y (price), as demais variáveis da base de dados são todas variáveis explicativas; particione a base de dados em 80% para treino e 20% para teste; e apresente os resultados:

- O valor ótimo do lambda para os modelos;
- O valor do alpha para o modelo ElasticNet;
- Os valores dos parâmetros para os modelos;
- O R^2 e RMSE dos modelos estimados;
- Apresente os resultados de uma predição proposta por você mesmo para os modelos (valor estimado e intervalos de confiança).

Importando as bibliotecas necessárias e dados

In [8]:
```
library(plyr)
library(readr)
library(dplyr)
library(caret)
library(ggplot2)
library(repr)
library(glmnet)

load("./Arquivos_para_R/imoveiscwbav.RData")
dataset <- imoveiscwbav
glimpse(dataset)
gc
```

```
Observations: 541
Variables: 20
$ price  <dbl> 1100000, 895000, 2513600, 755000, 1099000, 475000, 463900, 1...
$ age    <dbl> 15, 11, 2, 25, 1, 31, 2, 1, 11, 1, 3, 3, 3, 11, 3, 20, 3, 3,...
$ parea  <dbl> 150, 165, 146, 163, 107, 96, 75, 122, 63, 97, 92, 138, 199, ...
$ tarea  <dbl> 190, 210, 275, 238, 189, 124, 90, 227, 87, 180, 130, 253, 40...
$ bath   <dbl> 4, 4, 4, 3, 3, 2, 2, 3, 2, 2, 2, 3, 5, 3, 2, 3, 5, 5, 2, 4, ...
$ ensuit <dbl> 1, 1, 3, 1, 1, 1, 1, 1, 1, 1, 1, 3, 3, 1, 1, 1, 3, 3, 1, 3, ...
$ garag  <dbl> 2, 2, 3, 2, 2, 1, 1, 2, 1, 2, 3, 2, 3, 2, 2, 1, 2, 3, 2, 2, ...
$ plaz   <dbl> 0.08058169, 0.16635098, 0.05607530, 0.32159391, 0.14663511, ...
$ park   <dbl> 0.7132806, 0.6983694, 1.3129824, 2.1099578, 1.0175299, 1.970...
$ trans  <dbl> 2.3862709, 2.2463043, 2.6314112, 2.1387003, 1.7978931, 0.994...
$ kidca  <dbl> 1.4109813, 1.8625914, 1.5914926, 1.6215857, 1.2572430, 1.097...
$ school <dbl> 0.9028108, 0.9355790, 0.4517910, 0.4478709, 0.8841994, 0.391...
$ health <dbl> 0.4146473, 0.2569533, 0.2321598, 0.6848450, 0.2990089, 0.279...
$ bike   <dbl> 0.21319266, 0.23255291, 0.29709268, 0.34714701, 0.77876451, ...
$ barb   <dbl> 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, ...
$ balc   <dbl> 1, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 1, 1, 1, ...
$ elev   <dbl> 0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, ...
$ fitg   <dbl> 1, 0, 1, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, ...
$ party  <dbl> 1, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 1, ...
$ categ  <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, ...
function (verbose = getOption("verbose"), reset = FALSE, full = TRUE)
{
    res <- .Internal(gc(verbose, reset, full))
    res <- matrix(res, 2L, 7L, dimnames = list(c("Ncells", "Vcells"),
        c("used", "(Mb)", "gc trigger", "(Mb)", "limit (Mb)",
            "max used", "(Mb)")))
    if (all(is.na(res[, 5L])))
        res[, -5L]
    else res
}
```

Separando os dados de treino e os de teste

In [9]:

```r
# criando semente pseudo aleatória para verificação futura
set.seed(1)

indices <- createDataPartition(dataset$price, p=0.8, list=F)
treino <- dataset[indices,]
teste <- dataset[-indices,]
```

Alterando escala das variaveis

In [19]:

```r
cols <- c('price', 'age', 'parea', 'tarea', 'bath',
         'ensuit', 'garag', 'plaz','park', 'trans',
         'kidca', 'school', 'health', 'bike')

pre_proc_val <- preProcess(treino[,cols], method = c("center", "scale"))

treino[,cols] <- predict(pre_proc_val, treino[,cols])
teste[,cols] <- predict(pre_proc_val, teste[,cols])

print("valores de treino")
summary(treino)
print("valores de teste")
summary(teste)
```

```
[1] "valores de treino"
```

```
     price              age              parea              tarea         
 Min.   :-1.3512   Min.   :-1.0471   Min.   :-2.17312   Min.   :-1.872522  
 1st Qu.:-0.7324   1st Qu.:-0.8890   1st Qu.:-0.83396   1st Qu.:-0.882164  
 Median :-0.1389   Median :-0.3354   Median :-0.04264   Median : 0.005744  
 Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.00000   Mean   : 0.000000  
 3rd Qu.: 0.4155   3rd Qu.: 0.7716   3rd Qu.: 0.74108   3rd Qu.: 0.782663  
 Max.   : 6.1479   Max.   : 2.9857   Max.   : 2.39220   Max.   : 3.762277  
      bath             ensuit             garag              plaz        
 Min.   :-2.5996   Min.   :-1.5890   Min.   :-2.7684   Min.   :-1.6640  
 1st Qu.:-0.8758   1st Qu.:-0.4978   1st Qu.:-1.2590   1st Qu.:-0.8550  
 Median :-0.0139   Median :-0.4978   Median : 0.2504   Median :-0.1820  
 Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000  
 3rd Qu.: 0.8480   3rd Qu.: 0.5934   3rd Qu.: 0.2504   3rd Qu.: 0.7331  
 Max.   : 2.5718   Max.   : 1.6845   Max.   : 3.2692   Max.   : 3.2624  
      park              trans             kidca             school        
 Min.   :-2.2628   Min.   :-2.4925   Min.   :-3.2493   Min.   :-2.06525  
 1st Qu.:-0.7801   1st Qu.:-0.7636   1st Qu.:-0.6062   1st Qu.:-0.80119  
 Median : 0.2407   Median : 0.2202   Median : 0.2238   Median :-0.01037  
 Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.00000  
 3rd Qu.: 0.8295   3rd Qu.: 0.8089   3rd Qu.: 0.7240   3rd Qu.: 0.62998  
 Max.   : 1.8369   Max.   : 1.4371   Max.   : 2.0784   Max.   : 3.47389  
     health              bike              barb           balc       
 Min.   :-1.7846   Min.   :-1.7295   Min.   :0.00   Min.   :0.0000  
 1st Qu.:-0.7214   1st Qu.:-0.7519   1st Qu.:0.00   1st Qu.:0.0000  
 Median :-0.2081   Median :-0.1259   Median :0.00   Median :0.0000  
 Mean   : 0.0000   Mean   : 0.0000   Mean   :0.47   Mean   :0.4401  
 3rd Qu.: 0.5379   3rd Qu.: 0.7515   3rd Qu.:1.00   3rd Qu.:1.0000  
 Max.   : 3.9310   Max.   : 3.5095   Max.   :1.00   Max.   :1.0000  
      elev             fitg             party            categ       
 Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000  
 1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:1.0000  
 Median :0.0000   Median :0.0000   Median :1.0000   Median :1.0000  
 Mean   :0.2972   Mean   :0.2926   Mean   :0.5369   Mean   :0.9562  
 3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.0000  
 Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000  
[1] "valores de teste"
```

```
     price               age                parea               tarea
 Min.   :-1.2572   Min.   :-1.04711   Min.   :-1.68615   Min.   :-1.68470
 1st Qu.:-0.7310   1st Qu.:-0.88896   1st Qu.:-0.84918   1st Qu.:-0.80533
 Median :-0.1577   Median :-0.49358   Median :-0.04264   Median :-0.07963
 Mean   :-0.0544   Mean   :-0.08712   Mean   :-0.03041   Mean   :-0.06399
 3rd Qu.: 0.3500   3rd Qu.: 0.73209   3rd Qu.: 0.67260   3rd Qu.: 0.64606
 Max.   : 2.3357   Max.   : 2.27406   Max.   : 1.90523   Max.   : 1.91816
      bath              ensuit                garag               plaz
 Min.   :-2.5996   Min.   :-1.588981   Min.   :-2.76839   Min.   :-1.67190
 1st Qu.:-0.8758   1st Qu.:-0.497814   1st Qu.:-1.25899   1st Qu.:-0.77106
 Median :-0.0139   Median :-0.497814   Median : 0.25041   Median :-0.02648
 Mean   :-0.1508   Mean   :-0.008318   Mean   : 0.08113   Mean   : 0.07582
 3rd Qu.: 0.8480   3rd Qu.: 0.593354   3rd Qu.: 0.25041   3rd Qu.: 0.92688
 Max.   : 1.7099   Max.   : 1.684521   Max.   : 3.26920   Max.   : 2.65944
      park               trans               kidca               school
 Min.   :-1.7396   Min.   :-2.19678   Min.   :-3.00038   Min.   :-1.9975
 1st Qu.:-0.4292   1st Qu.:-0.76358   1st Qu.:-0.51831   1st Qu.:-0.7239
 Median : 0.5308   Median : 0.06041   Median : 0.09492   Median :-0.1170
 Mean   : 0.2440   Mean   :-0.01099   Mean   :-0.02816   Mean   :-0.1620
 3rd Qu.: 0.9718   3rd Qu.: 0.77202   3rd Qu.: 0.67673   3rd Qu.: 0.2248
 Max.   : 1.5623   Max.   : 1.43446   Max.   : 1.47962   Max.   : 2.3377
     health              bike               barb              balc
 Min.   :-1.5980   Min.   :-1.7051   Min.   :0.0000   Min.   :0.0000
 1st Qu.:-0.7324   1st Qu.:-0.5777   1st Qu.:0.0000   1st Qu.:0.0000
 Median :-0.1705   Median : 0.1026   Median :1.0000   Median :0.0000
 Mean   : 0.1157   Mean   : 0.2439   Mean   :0.5981   Mean   :0.4766
 3rd Qu.: 0.8431   3rd Qu.: 0.9821   3rd Qu.:1.0000   3rd Qu.:1.0000
 Max.   : 3.9310   Max.   : 3.1196   Max.   :1.0000   Max.   :1.0000
      elev             fitg             party             categ
 Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
 1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:1.0000
 Median :0.0000   Median :0.0000   Median :1.0000   Median :1.0000
 Mean   :0.3551   Mean   :0.3738   Mean   :0.5794   Mean   :0.9533
 3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.0000
 Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
```

Função que calcula e retorna R^2 e RMSE

In [50]:
```
eval_results <- function(true, predicted, df) {
  SSE <- sum((predicted - true)^2)
  SST <- sum((true - mean(true))^2)
  R_square <- 1 - SSE / SST
  RMSE = sqrt(SSE/nrow(df))

  # Model performance metrics
  data.frame(
    RMSE = RMSE,
    Rsquare = R_square
  )
}
```

Regressão Ridge

In [21]:
```
cols_reg <- c('price', 'age', 'parea', 'tarea', 'bath',
         'ensuit', 'garag', 'plaz','park', 'trans',
         'kidca', 'school', 'health', 'bike', 'barb',
         'balc', 'elev', 'fitg', 'party', 'categ')

dummies <- dummyVars(price ~ age+parea+tarea+bath+
                     ensuit+garag+plaz+park+trans+kidca+
                     school+health+bike+barb+balc+elev+fitg+
                     party+categ,
```

```
                     data = dataset[,cols_reg])

train_dummies <- predict(dummies, newdata = treino[,cols_reg])

test_dummies <- predict(dummies, newdata = teste[,cols_reg])

print(dim(train_dummies)); print(dim(test_dummies))
```

```
[1] 434  19
[1] 107  19
```

In [22]:
```
# Dados para o modelo
x = as.matrix(train_dummies)
y_train = treino$price

x_test = as.matrix(test_dummies)
y_test = teste$price
```

In [28]:
```
#Identficando o valor ótimo de lambda
lambdas <- 10^seq(2, -3, by = -.1)
ridge_lamb <- cv.glmnet(x, y_train, alpha = 0,
                      lambda = lambdas)
best_lambda_ridge <- ridge_lamb$lambda.min
best_lambda_ridge
```

0.1

O melhor lambda obtido é 0.1

In [29]:
```
ridge_reg = glmnet(x, y_train, nlambda = 25, alpha = 0,
                   family = 'gaussian',
                   lambda = best_lambda_ridge)

summary(ridge_reg)
```

```
          Length Class     Mode
a0         1     -none-    numeric
beta      19     dgCMatrix S4
df         1     -none-    numeric
dim        2     -none-    numeric
lambda     1     -none-    numeric
dev.ratio  1     -none-    numeric
nulldev    1     -none-    numeric
npasses    1     -none-    numeric
jerr       1     -none-    numeric
offset     1     -none-    logical
call       7     -none-    call
nobs       1     -none-    numeric
```

In [48]:
```
# obtendo valores dos parâmetros
ridge_reg[["beta"]]
```

```
19 x 1 sparse Matrix of class "dgCMatrix"
                s0
age    -0.17994688
parea   0.15907495
tarea   0.21631677
bath    0.04261451
ensuit  0.18954877
garag   0.20506605
plaz    0.04714112
park   -0.05375536
trans   0.03345027
kidca   0.01621329
school -0.00131847
health -0.01054514
bike   -0.04790725
barb   -0.06878716
balc    0.15377909
elev   -0.18946965
fitg    0.23383601
party   0.06497685
categ   0.46258571
```

In [54]:
```
#predição dos dados de Teste
predictions_test <- predict(ridge_reg, s = best_lambda_ridge,
                            newx = x_test)
eval_results(y_test, predictions_test, test)
```

RMSE	Rsquare
0.1792301	0.8487347

Avaliação: O R^2 está próximo de 1, mas não tão próximo, o que é um bom sinal pois significa que tem menos chances de o modelo estar sofrendo overfitting. O RMSE está próximo de zero, o que é bom sinal, significa que poucos erros foram cometidos.

Realizando predição através de variáveis obtidas através da médiana de todo banco de dados

In [69]:
```
# O valor esperado é R$ 880000
price <- (median(dataset$price)-pre_proc_val[["mean"]][["price"]])/pre_proc_val[["s
age <- (median(dataset$age)-pre_proc_val[["mean"]][["age"]])/pre_proc_val[["std"]]
parea <- (median(dataset$parea)-pre_proc_val[["mean"]][["parea"]])/pre_proc_val[["s
tarea <- (median(dataset$tarea)-pre_proc_val[["mean"]][["tarea"]])/pre_proc_val[["s
bath <- (median(dataset$bath)-pre_proc_val[["mean"]][["bath"]])/pre_proc_val[["std
ensuit <- (median(dataset$ensuit)-pre_proc_val[["mean"]][["ensuit"]])/pre_proc_val
garag <- (median(dataset$garag)-pre_proc_val[["mean"]][["garag"]])/pre_proc_val[["s
plaz <- (median(dataset$plaz)-pre_proc_val[["mean"]][["plaz"]])/pre_proc_val[["std
park <- (median(dataset$park)-pre_proc_val[["mean"]][["park"]])/pre_proc_val[["std
trans <- (median(dataset$trans)-pre_proc_val[["mean"]][["trans"]])/pre_proc_val[["s
kidca <- (median(dataset$kidca)-pre_proc_val[["mean"]][["kidca"]])/pre_proc_val[["s
school <- (median(dataset$school)-pre_proc_val[["mean"]][["school"]])/pre_proc_val
health <- (median(dataset$health)-pre_proc_val[["mean"]][["health"]])/pre_proc_val
bike <- (median(dataset$bike)-pre_proc_val[["mean"]][["bike"]])/pre_proc_val[["std
barb <- 0
balc <- 0
elev <- 0
fitg <- 0
party <- 0
categ <- 0

# Constuirndo matriz com dados para predição
our_pred <- as.matrix(data.frame(age=age,
```

```
                                    parea=parea,
                                    tarea=tarea,
                                    bath=bath,
                                    ensuit=ensuit,
                                    garag=garag,
                                    plaz=plaz,
                                    park=park,
                                    trans=trans,
                                    kidca=kidca,
                                    school=school,
                                    health=health,
                                    bike=bike,
                                    barb=barb,
                                    balc=balc,
                                    elev=elev,
                                    fitg=fitg,
                                    party=party,
                                    categ=categ))
```

In [70]:
```
# Fazendo a predição
predict_our_ridge <- predict(ridge_reg, s = best_lambda_ridge,
                             newx = our_pred)
predict_our_ridge
```

1
57.237

In [72]:
```
# Intervalo de confiança para nosso Exemplo
n <- nrow(treino)
m <- predict_our_ridge
s <- pre_proc_val[["std"]][["price"]]
dam <- s/sqrt(n)
CIlwr_ridge <- m + (qnorm(0.025))*dam
CIupr_ridge <- m - (qnorm(0.025))*dam

CIlwr_ridge
CIupr_ridge
```

1
57.14292

1
57.33108

Avaliação: O valor obtido assim como o intervalo de confiança tiveram grande divergência com o valor esperado.

Regressão Lasso

In [73]:
```
# Escolhendo  melhor lambda como parâmetro

lambdas <- 10^seq(2, -3, by = -.1)

# Setting alpha = 1 implements lasso regression
lasso_lamb <- cv.glmnet(x, y_train, alpha = 1,
                        lambda = lambdas,
                        standardize = TRUE, nfolds = 5)
```

```
# Best
best_lambda_lasso <- lasso_lamb$lambda.min
best_lambda_lasso

lasso_model <- glmnet(x, y_train, alpha = 1,
                      lambda = best_lambda_lasso,
                      standardize = TRUE)
```

0.00794328234724281

O melhor lambda obtido é 0.00794328234724281

In [74]:
```
# Visualizando os parâmetros calculados
lasso_model[["beta"]]
```

```
19 x 1 sparse Matrix of class "dgCMatrix"
                 s0
age    -0.177787541
parea   0.160244253
tarea   0.239394524
bath    0.005550922
ensuit  0.215324160
garag   0.214458904
plaz    0.043167047
park   -0.056688502
trans   0.030263567
kidca   0.008624390
school  .
health -0.002502411
bike   -0.038298740
barb   -0.066286730
balc    0.147531615
elev   -0.179246307
fitg    0.240380064
party   0.043609304
categ   0.490774990
```

In [76]:
```
# Fazendo as predições a avaliando o modelo lasso na base teste
predictions_test <- predict(lasso_model, s = best_lambda_lasso,
                            newx = x_test)
eval_results(y_test, predictions_test, teste)
```

RMSE	Rsquare
0.3313211	0.8459347

Avaliação: O R^2 está próximo de 1, mas não tão próximo, o que é um bom sinal pois significa que tem menos chances de o modelo estar sofrendo overfitting. O RMSE está menos próximo de zero do que o modelo de regressão Ridge.

Realizando predição através de variáveis obtidas através da médiana de todo banco de dados

In [79]:
```
# O valor esperado é R$ 880000
predict_our_lasso <- predict(lasso_model, s = best_lambda_lasso,
                             newx = our_pred)
predict_our_lasso
```

1
61.51794

In [81]:
```
# Intervalo de confiança para nosso Exemplo
n <- nrow(treino)
m <- predict_our_lasso
s <- pre_proc_val[["std"]][["price"]]
dam <- s/sqrt(n)
CIlwr_lasso <- m + (qnorm(0.025))*dam
CIupr_lasso <- m - (qnorm(0.025))*dam

CIlwr_lasso
CIupr_lasso
```

1
61.42386

1
61.61202

Avaliação: O valor obtido assim como o intervalo de confiança tiveram grande divergência com o valor esperado, mas o resultado foi melhor que o do modelo Ridge.

Regressão ElasticNet

In [82]:
```
# Ajustando controle de treino
train_cont <- trainControl(method = "repeatedcv",
                           number = 10,
                           repeats = 5,
                           search = "random",
                           verboseIter = TRUE)
```

In [84]:
```
# treinando o modelo
price ~ age+parea+tarea+bath+
                     ensuit+garag+plaz+park+trans+kidca+
                     school+health+bike+barb+balc+elev+fitg+
                     party+categ
```

In [85]:
```
# melhor ajuste de parâmetro
elastic_reg$bestTune
```

	alpha	lambda
10	0.8963074	0.006091447

O melhor alfa obtido foi 0.8963074, e lambda 0.006091447

In [87]:
```
elastic_reg[["finalModel"]][["beta"]]
```

```
[[ suppressing 69 column names 's0', 's1', 's2' ... ]]
```

19 x 69 sparse Matrix of class "dgCMatrix"

```
age     . .          .          .          .          .          .
parea   . .          .          .          .          .          .
tarea   . 0.06221618 0.09989076 0.12772340 0.14929387 0.16906064 0.18712835
bath    . .          .          .          .          .          .
ensuit  . .          .          0.01504467 0.04071750 0.06428228 0.08589871
garag   . .          0.03222980 0.05994032 0.08179667 0.10176534 0.12002760
plaz    . .          .          .          .          .          .
park    . .          .          .          .          .          .
trans   . .          .          .          .          .          .
kidca   . .          .          .          .          .          .
school  . .          .          .          .          .          .
health  . .          .          .          .          .          .
bike    . .          .          .          .          .          .
barb    . .          .          .          .          .          .
balc    . .          .          .          .          .          .
elev    . .          .          .          .          .          .
fitg    . .          .          .          .          .          .
party   . .          .          .          .          .          .
categ   . .          .          .          .          .          .

age     .         .         .         .         -0.001319996 -0.015948635
parea   .         .         .         .          .            0.008033533
tarea   0.2036379 0.2187182 0.2324884 0.2450617  0.256510523  0.262629920
bath    .         .         .         .          .            .
ensuit  0.1057171 0.1238765 0.1405069 0.1557327  0.169321015  0.176382496
garag   0.1367236 0.1519846 0.1659313 0.1786711  0.189911772  0.195037754
plaz    .         .         .         .          .            .
park    .         .         .         .          .            .
trans   .         .         .         .          .            .
kidca   .         .         .         .          .            .
school  .         .         .         .          .            .
health  .         .         .         .          .            .
bike    .         .         .         .          .            .
barb    .         .         .         .          .            .
balc    .         .         .         .          .            .
elev    .         .         .         .          .            .
fitg    .         .         .         .          .            .
party   .         .         .         .          .            .
categ   .         .         .         .          .            .

age     -0.029166231 -0.03999327 -0.04992644 -0.05901133 -0.06735031 -0.07498425
parea    0.017929361  0.02796488  0.03722751  0.04572351  0.05359753  0.06084214
tarea    0.266891447  0.26983617  0.27249177  0.27482479  0.27693159  0.27881146
bath     .            .           .           .           .           .
ensuit   0.181247928  0.18358432  0.18564700  0.18752487  0.18915754  0.19061480
garag    0.198741933  0.20130987  0.20359580  0.20568498  0.20753013  0.20919235
plaz     .            .           .           .           .           .
park     .            .           .           .           .           .
trans    .            .           .           .           .           .
kidca    .            .           .           .           .           .
school   .            .           .           .           .           .
health   .            .           .           .           .           .
bike     .            .           .           .           .           .
barb     .            .           .           .           .           .
balc     .            .           .           .           .           .
elev     .            .           .           .           .           .
fitg     0.009048512  0.03597395  0.06062133  0.08316566  0.10379565  0.12265972
party    .            .           .           .           .           .
categ    .            .           .           .           .           .

age     -0.081144168 -0.08655653 -0.0915995143 -0.09724719 -0.10242773
parea    0.067422767  0.07326105  0.0783940808  0.08226249  0.08594144
```

```
tarea   0.280351651  0.28186735  0.2831354310  0.28222708  0.28145011
bath    .            .           .             .           .
ensuit  0.191919411  0.19332039  0.1946554143  0.19750374  0.19997006
garag   0.210044232  0.21071987  0.2113176206  0.21150551  0.21161258
plaz    .            .           .             .           .
park    .            .          -0.0008610719 -0.00922158 -0.01681923
trans   .            .           .             .           .
kidca   .            .           .             .           .
school  .            .           .             .           .
health  .            .           .             .           .
bike    .            .           .             .           .
barb    .            .           .             .           .
balc    0.008801045  0.01843474  0.0270423875  0.03427803  0.04084007
elev    .            .           .             .           .
fitg    0.137007719  0.14942761  0.1607000827  0.16940873  0.17740768
party   .            .           .             .           .
categ   .            .           .             .           .

age    -0.10750618 -0.11346026 -0.11882739 -0.12370348 -0.12819454 -0.13287809
parea   0.09030636  0.09833596  0.10546571  0.11189032  0.11790761  0.12355372
tarea   0.28015189  0.27632497  0.27296444  0.26988853  0.26701619  0.26418877
bath    .           .           .           .           .           .
ensuit  0.20239267  0.20493887  0.20727597  0.20944969  0.21138309  0.21235905
garag   0.21170484  0.21155492  0.21147173  0.21142489  0.21134282  0.21116069
plaz    .           .           .           .           .           .
park   -0.02342943 -0.02859407 -0.03330871 -0.03762630 -0.04154957 -0.04516227
trans   .           .           .           .           .           .
kidca   .           .           .           .           .           .
school  .           .           .           .           .           .
health  .           .           .           .           .           .
bike    .           .           .           .           .           .
barb    .           .           .           .           .           .
balc    0.04671784  0.05168975  0.05624594  0.06041401  0.06419566  0.07007890
elev    .           .           .           .           .          -0.01132395
fitg    0.18373590  0.18695534  0.18988070  0.19253359  0.19495280  0.19959118
party   .           .           .           .           .           .
categ   0.01238136  0.05716961  0.09787397  0.13495746  0.16893405  0.20127714

age    -0.1373712562 -0.1412039255 -0.145081392 -0.148471707 -0.151405223
parea   0.1287084710  0.1331262925  0.136886140  0.140528017  0.142454026
tarea   0.2616073504  0.2592665924  0.256890887  0.254918090  0.253774710
bath    .             0.0007854225  0.002258168  0.003079957  0.003467332
ensuit  0.2128029981  0.2126959396  0.212331350  0.212177170  0.212757750
garag   0.2109586318  0.2107836383  0.210507059  0.210264878  0.210455337
plaz    .             .             .            0.002730924  0.007615291
park   -0.0482011334 -0.0503680376 -0.052415376 -0.054297271 -0.054754789
trans   .             .             .            0.001137964  0.005008823
kidca   .             .             .            .            .
school  .             .             .            .            .
health  .             .             .            .            .
bike   -0.0008320617 -0.0031424879 -0.005210268 -0.007884779 -0.011160998
barb    .             .             .            .           -0.005039656
balc    0.0769521800  0.0832656885  0.088903960  0.094149279  0.100700481
elev   -0.0279779800 -0.0432195096 -0.057266599 -0.069883945 -0.081077067
fitg    0.2051730861  0.2105059286  0.215237424  0.219727109  0.224411224
party   .             .             .            .            .
categ   0.2312280936  0.2577301734  0.281937283  0.304854753  0.327222287

age    -0.154343606 -0.157137342 -0.159431578 -0.161751565 -0.164012648
parea   0.144040786  0.145956848  0.147150839  0.148679800  0.150413464
tarea   0.252576817  0.251339892  0.250203366  0.248977187  0.247685206
bath    0.004183773  0.004829303  0.004765201  0.005053105  0.005391851
ensuit  0.213132024  0.213216731  0.213972298  0.214305969  0.214352605
garag   0.210788433  0.210990756  0.211599905  0.211901392  0.212098405
```

```
plaz    0.011847899  0.015644212  0.019093419  0.022215137  0.025055097
park   -0.055176890 -0.055492075 -0.055984638 -0.056349362 -0.056501619
trans   0.008356168  0.011401516  0.014229227  0.016781280  0.018909339
kidca   .            .            .            .            0.000587330
school  .            .            .            .            .
health  .            .            .            .            .
bike   -0.013936531 -0.016501308 -0.018869622 -0.021024597 -0.023162967
barb   -0.012247476 -0.018831430 -0.024706588 -0.030154874 -0.035167512
balc    0.107407384  0.112823110  0.117108122  0.120972001  0.124502646
elev   -0.091149349 -0.101101488 -0.111062213 -0.120164498 -0.128506245
fitg    0.228602120  0.231553959  0.232894892  0.234172723  0.235165963
party   .            0.002538709  0.007854756  0.012630910  0.016973130
categ   0.347596566  0.366009674  0.381777552  0.396572408  0.410115943

age    -0.165770732 -0.167537886 -0.169181338 -0.170681654 -0.171976042
parea   0.151259860  0.152396870  0.153531403  0.154577464  0.155385075
tarea   0.246772900  0.245695708  0.244689147  0.243769629  0.243095851
bath    0.005306857  0.005428177  0.005559029  0.005678079  0.005687657
ensuit  0.214810701  0.215014690  0.215139027  0.215245325  0.215423776
garag   0.212587036  0.212896771  0.213151105  0.213380748  0.213651971
plaz    0.027771868  0.030217299  0.032439025  0.034463100  0.036325414
park   -0.056731059 -0.056861012 -0.056968245 -0.057064279 -0.057203340
trans   0.020543614  0.022008527  0.023334470  0.024542294  0.025677655
kidca   0.001881330  0.003076373  0.004161548  0.005150008  0.006001930
school  .            .            .            .            .
health  .            .            .            .            .
bike   -0.025334779 -0.027320699 -0.029126856 -0.030772775 -0.032256713
barb   -0.039703992 -0.043919494 -0.047764028 -0.051268272 -0.054355878
balc    0.127956149  0.131107600  0.133964660  0.136567915  0.138843368
elev   -0.135883150 -0.142640843 -0.148813492 -0.154441442 -0.159596728
fitg    0.235968274  0.236678095  0.237328118  0.237921684  0.238417015
party   0.020799909  0.024282347  0.027458263  0.030353327  0.033047920
categ   0.421313229  0.431865217  0.441561422  0.450408268  0.458371116

age    -0.1731968999 -0.1744159129 -0.175538604 -0.176563504 -0.177497442
parea   0.1561870788  0.1569185504  0.157638291  0.158302339  0.158907994
tarea   0.2423448082  0.2416127882  0.240943994  0.240335891  0.239782025
bath    0.0057657430  0.0058679594  0.005982500  0.006087107  0.006182360
ensuit  0.2155560525  0.2155603201  0.215531824  0.215498579  0.215467500
garag   0.2138656937  0.2140780188  0.214249155  0.214402685  0.214542600
plaz    0.0380100399  0.0395052866  0.040857991  0.042089552  0.043211793
park   -0.0572840067 -0.0571655724 -0.057047547 -0.056936621 -0.056835082
trans   0.0266889940  0.0277169827  0.028676699  0.029551305  0.030348549
kidca   0.0068292836  0.0073396295  0.007811388  0.008240737  0.008631871
school  .             .             .            .            .
health -0.0001389622 -0.0008240572 -0.001436755 -0.001995567 -0.002504879
bike   -0.0336449555 -0.0349802254 -0.036192965 -0.037298304 -0.038305715
barb   -0.0572426723 -0.0597838388 -0.062090096 -0.064192734 -0.066108992
balc    0.1410045357  0.1429026306  0.144591779  0.146131056  0.147534026
elev   -0.1642490603 -0.1684653801 -0.172350347 -0.175892323 -0.179120595
fitg    0.2388821200  0.2392683017  0.239608919  0.239922581  0.240208684
party   0.0354668989  0.0377787189  0.039912323  0.041856072  0.043627584
categ   0.4656460847  0.4723705260  0.478567167  0.484220409  0.489372912

age    -0.178348632 -0.179124497 -0.179831708 -0.180443390 -0.181013336
parea   0.159460128  0.159963713  0.160423078  0.160789410  0.161110959
tarea   0.239276874  0.238816000  0.238395535  0.238120899  0.237796835
bath    0.006269162  0.006348296  0.006420443  0.006454649  0.006503040
ensuit  0.215439048  0.215412952  0.215388990  0.215393101  0.215401813
garag   0.214670202  0.214786498  0.214892460  0.214992637  0.215096404
plaz    0.044234508  0.045166506  0.046015817  0.046807929  0.047516234
park   -0.056742327 -0.056657607 -0.056580236 -0.056581008 -0.056517807
trans   0.031075257  0.031737626  0.032341328  0.032889012  0.033397558
kidca   0.008988307  0.009313122  0.009609115  0.009844299  0.010092145
```

```
school  .            .            .            .            .
health -0.002969078 -0.003392158 -0.003777751 -0.004114290 -0.004434039
bike   -0.039223908 -0.040060761 -0.040823457 -0.041493759 -0.042129984
barb   -0.067855325 -0.069446783 -0.070897082 -0.072144606 -0.073346423
balc    0.148812692  0.149978004  0.151039978  0.151915878  0.152805097
elev   -0.182062963 -0.184744709 -0.187188862 -0.189421773 -0.191451444
fitg    0.240469222  0.240706453  0.240922478  0.241114464  0.241288625
party   0.045242356  0.046714241  0.048055840  0.049291693  0.050409311
categ   0.494068921  0.498349018  0.502250014  0.505748543  0.508944115

age    -0.181544162 -0.182033846 -0.182482349 -0.182891827 -0.183265228
parea   0.161442306  0.161762609  0.162061165  0.162335338  0.162585798
tarea   0.237475116  0.237174255  0.236898197  0.236646252  0.236416580
bath    0.006556024  0.006607560  0.006655808  0.006700329  0.006741149
ensuit  0.215393097  0.215376205  0.215357051  0.215338160  0.215320414
garag   0.215183054  0.215257062  0.215322361  0.215381076  0.215434312
plaz    0.048158753  0.048743188  0.049275321  0.049760057  0.050201706
park   -0.056454810 -0.056396015 -0.056341777 -0.056292043 -0.056246581
trans   0.033856619  0.034273239  0.034652381  0.034997751  0.035312455
kidca   0.010320315  0.010527662  0.010716163  0.010887730  0.011043989
school  .            .            .            .            .
health -0.004726785 -0.004993878 -0.005237391 -0.005459347 -0.005661632
bike   -0.042710415 -0.043238704 -0.043719804 -0.044158102 -0.044557474
barb   -0.074449311 -0.075455365 -0.076372382 -0.077208097 -0.077969663
balc    0.153619614  0.154360027  0.155033769  0.155647368  0.156206393
elev   -0.193299385 -0.194984354 -0.196520418 -0.197920441 -0.199196343
fitg    0.241450592  0.241599181  0.241735096  0.241859167  0.241972285
party   0.051424445  0.052349184  0.053191770  0.053959545  0.054659190
categ   0.511884590  0.514578635  0.517039045  0.519282875  0.521328111

age    -0.183605590 -0.183915791 -0.184198490 -0.184433102 -0.1846642179
parea   0.162814233  0.163022492  0.163212340  0.163378387  0.1634952707
tarea   0.236207233  0.236016402  0.235842443  0.235744941  0.2356276601
bath    0.006778461  0.006812515  0.006843572  0.006880321  0.0069134294
ensuit  0.215304036  0.215289018  0.215275280  0.215260469  0.2152581582
garag   0.215482736  0.215526830  0.215566996  0.215582343  0.2156155982
plaz    0.050604125  0.050970809  0.051304931  0.051599843  0.0518811437
park   -0.056205079 -0.056167216 -0.056132682 -0.056166985 -0.0561456251
trans   0.035599235  0.035860572  0.036098719  0.036299612  0.0365040697
kidca   0.011186346  0.011316053  0.011434240  0.011502216  0.0115940473
school  .            .            .            .           -0.0002255448
health -0.005845978 -0.006013970 -0.006167057 -0.006291547 -0.0065009678
bike   -0.044921398 -0.045253024 -0.045555218 -0.045801573 -0.0460404970
barb   -0.078663638 -0.079296010 -0.079872241 -0.080359739 -0.0809047354
balc    0.156715761  0.157179901  0.157602832  0.157906174  0.1582613724
elev   -0.200359075 -0.201418648 -0.202384199 -0.203242407 -0.2040669603
fitg    0.242075361  0.242169265  0.242254810  0.242387815  0.2425187813
party   0.055296769  0.055877791  0.056407268  0.056880965  0.0573162862
categ   0.523192014  0.524890572  0.526438423  0.527799943  0.5289239427

age    -0.1848692780 -0.185059893 -0.185233400 -0.185363814
parea   0.1637464019  0.163985899  0.164192137  0.164357467
tarea   0.2354010073  0.235217566  0.235059668  0.234972004
bath    0.0068916221  0.006871758  0.006857157  0.006866585
ensuit  0.2152238843  0.215199367  0.215180224  0.215165792
garag   0.2156514621  0.215668808  0.215682240  0.215689148
plaz    0.0523163184  0.052685501  0.053020449  0.053287742
park   -0.0564404845 -0.056726190 -0.056983885 -0.057188877
trans   0.0365630616  0.036612991  0.036664384  0.036734860
kidca   0.0117253710  0.011804460  0.011872399  0.011926985
school -0.0008088907 -0.001310853 -0.001765625 -0.002112836
health -0.0067315320 -0.006938953 -0.007127574 -0.007281728
bike   -0.0461949769 -0.046321085 -0.046435427 -0.046540304
barb   -0.0814971186 -0.082000786 -0.082456479 -0.082838422
```

```
balc     0.1585916988  0.158834549  0.159049977  0.159232123
elev    -0.2048996180 -0.205615243 -0.206271619 -0.206826843
fitg     0.2427353115  0.242969076  0.243180815  0.243366056
party    0.0577534884  0.058133817  0.058483126  0.058763011
categ    0.5295536223  0.530153592  0.530701363  0.531265608
```

Fazendo as predições nas bases de teste e avaliando o modelo

In [88]:
```
predictions_test <- predict(elastic_reg, x_test)
eval_results(y_test, predictions_test, teste)
```

RMSE	Rsquare
0.3326286	0.8447163

Avaliação: O R^2 está próximo de 1, mas não tão próximo, o que é um bom sinal pois significa que tem menos chances de o modelo estar sofrendo overfitting. O RMSE está menos próximo de zero do que o modelo de regressão Ridge.

Realizando predição através de variáveis obtidas através da médiana de todo banco de dados

In [91]:
```
predict_our_elastic <- predict(elastic_reg,our_pred)
predict_our_elastic
```

61.3230193466218

In [92]:
```
# Intervalo de confiança para nosso Exemplo
n <- nrow(treino)
m <- predict_our_elastic
s <- pre_proc_val[["std"]][["price"]]
dam <- s/sqrt(n)
CIlwr_elastic <- m + (qnorm(0.025))*dam
CIupr_elastic <- m - (qnorm(0.025))*dam

CIlwr_elastic
CIupr_elastic
```

61.2289380647147

61.4171006285289

Avaliação: O valor obtido assim como o intervalo de confiança tiveram grande divergência com o valor esperado, mas o resultado foi melhor que o do modelo Ridge.

Avaliação Final: Dentre os modelos testados o que obteve melhor resultado foi o lasso. Todavia em todos modelos o valor obtido assim como o intervalo de confiança tiveram grande divergência com o valor esperado.