Regressão Simples

Davi Guerra

16/03/2022

```
setwd('D:/pibic')

options(scipen=999)

pacman::p_load('tidyverse')

#Retirando essas variáveis dos bancos de dados pois elas apresentavam muitos valores faltantes
#na hora das análises e os valores que restavam não influenciava muito na variável resposta

df_treino = read.csv('data/train.csv') %>%
  dplyr::select(!c(Alley,PoolQC,Fence,MiscFeature))

df_test = read.csv('data/test.csv') %>%
  dplyr::select(!c(Alley,PoolQC,Fence,MiscFeature))

y = df_treino$SalePrice

df_treino = df_treino %>%
  dplyr::select(!c(SalePrice))
```

Separando as variáveis numéricas e categóricas

```
col_num = sapply(df_treino, typeof) == "integer"
col_char = sapply(df_treino, typeof) == "character"

numericas = df_treino[col_num]
categoricas = df_treino[col_char]

length(numericas)+length(categoricas)
```

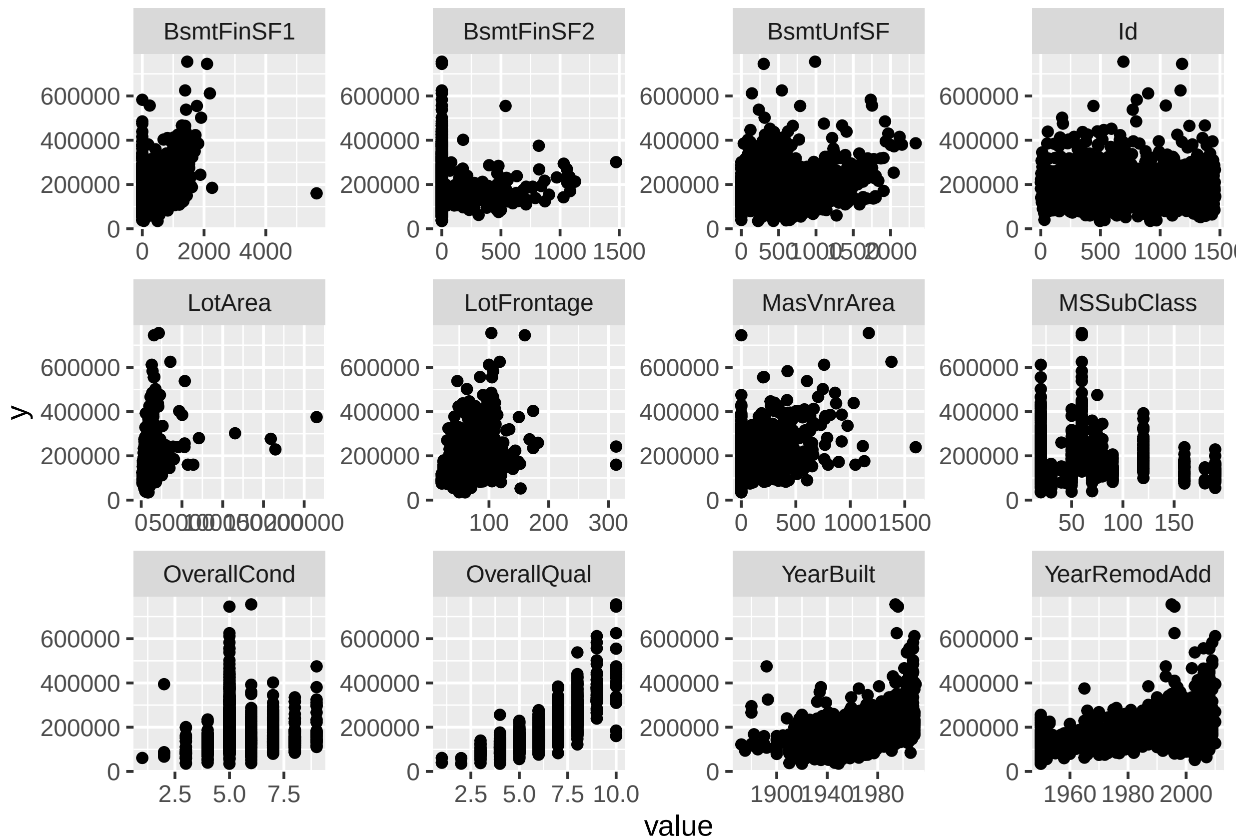
```
## [1] 76
```

Gráfico de Dispersão das covariáveis numéricas pela variável resposta

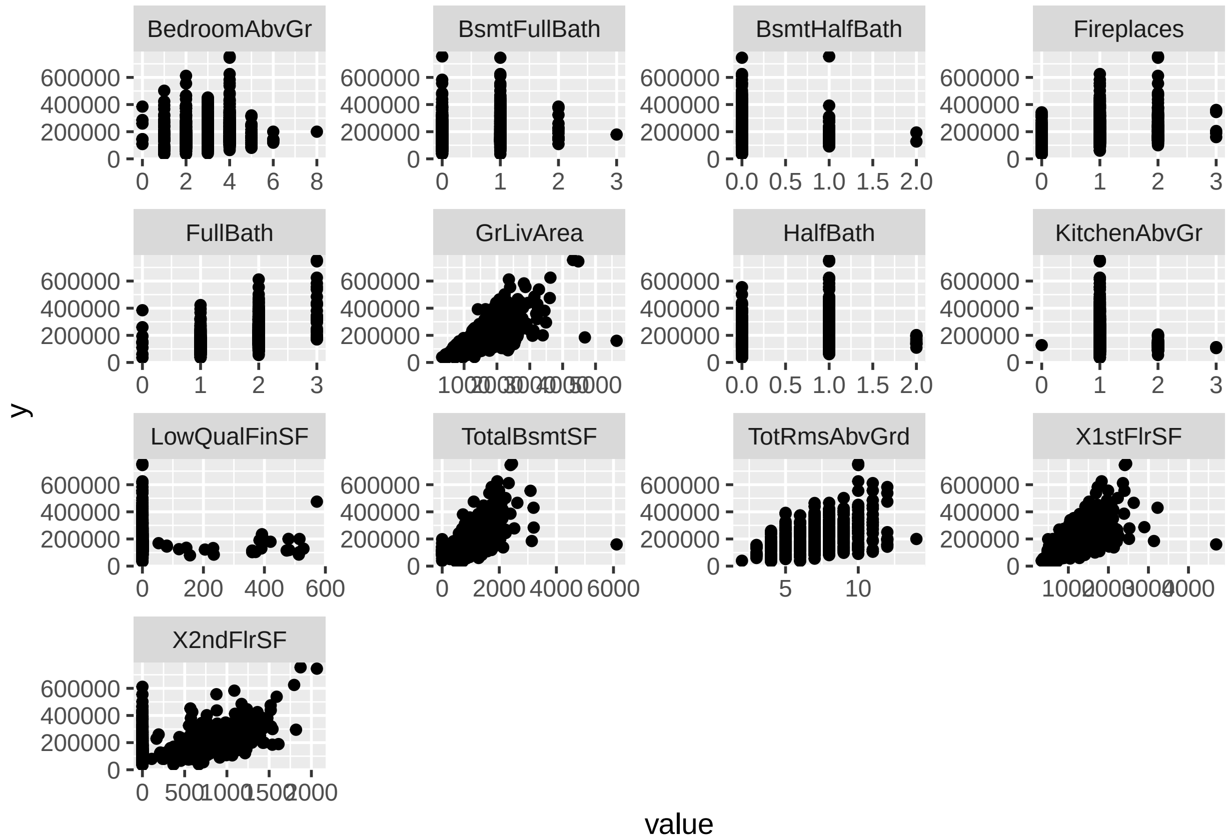
```
cbind(numericas[1:12],y) %>%
  pivot_longer(-y) %>%
  ggplot(aes(x = value, y = y))+
```

```
    geom_point()+
    facet_wrap(~name, scales = 'free')
```

```
## Warning: Removed 267 rows containing missing values (geom_point).
```

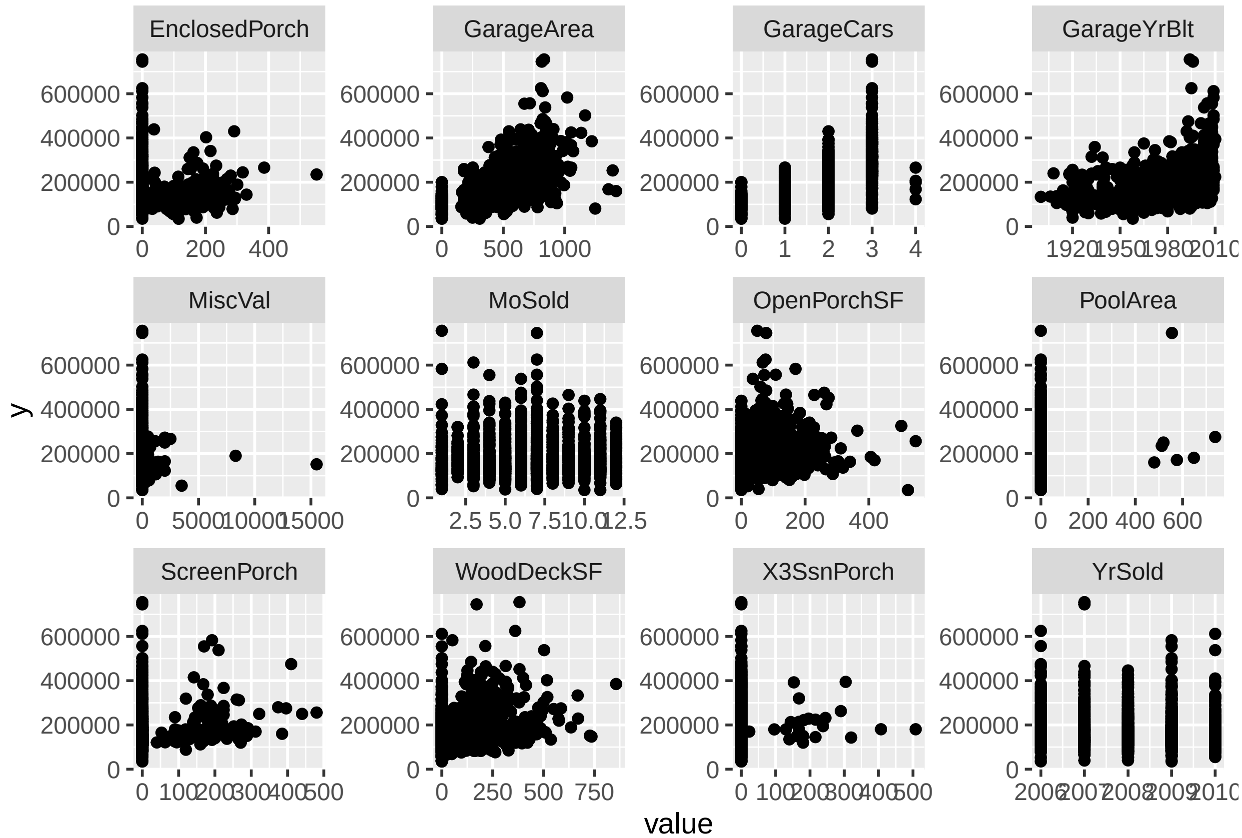



```
cbind(numericas[13:25],y) %>%
  pivot_longer(-y) %>%
  ggplot(aes(x = value, y = y))+
    geom_point()+
    facet_wrap(~name, scales = 'free')
```




```
cbind(numericas[26:37],y) %>%
  pivot_longer(-y) %>%
  ggplot(aes(x = value, y = y))+
    geom_point()+
    facet_wrap(~name, scales = 'free')
```

```
## Warning: Removed 81 rows containing missing values (geom_point).
```



Teste de correlação de pearson

```
#Pvalor
result_cor_test = sapply(numericas, function(x) round(cor.test(x,y)$p.value,5))
result_cor_test
```

```
##            Id     MSSubClass    LotFrontage        LotArea    OverallQual
##       0.40269        0.00127        0.00000        0.00000        0.00000
##   OverallCond      YearBuilt   YearRemodAdd     MasVnrArea     BsmtFinSF1
##       0.00291        0.00000        0.00000        0.00000        0.00000
##    BsmtFinSF2      BsmtUnfSF    TotalBsmtSF      X1stFlrSF      X2ndFlrSF
##       0.66400        0.00000        0.00000        0.00000        0.00000
##  LowQualFinSF      GrLivArea   BsmtFullBath   BsmtHalfBath       FullBath
##       0.32821        0.00000        0.00000        0.52015        0.00000
##      HalfBath   BedroomAbvGr   KitchenAbvGr   TotRmsAbvGrd     Fireplaces
##       0.00000        0.00000        0.00000        0.00000        0.00000
##   GarageYrBlt     GarageCars     GarageArea     WoodDeckSF    OpenPorchSF
##       0.00000        0.00000        0.00000        0.00000        0.00000
## EnclosedPorch     X3SsnPorch    ScreenPorch       PoolArea        MiscVal
##       0.00000        0.08858        0.00002        0.00041        0.41849
##        MoSold         YrSold
##       0.07613        0.26941
```

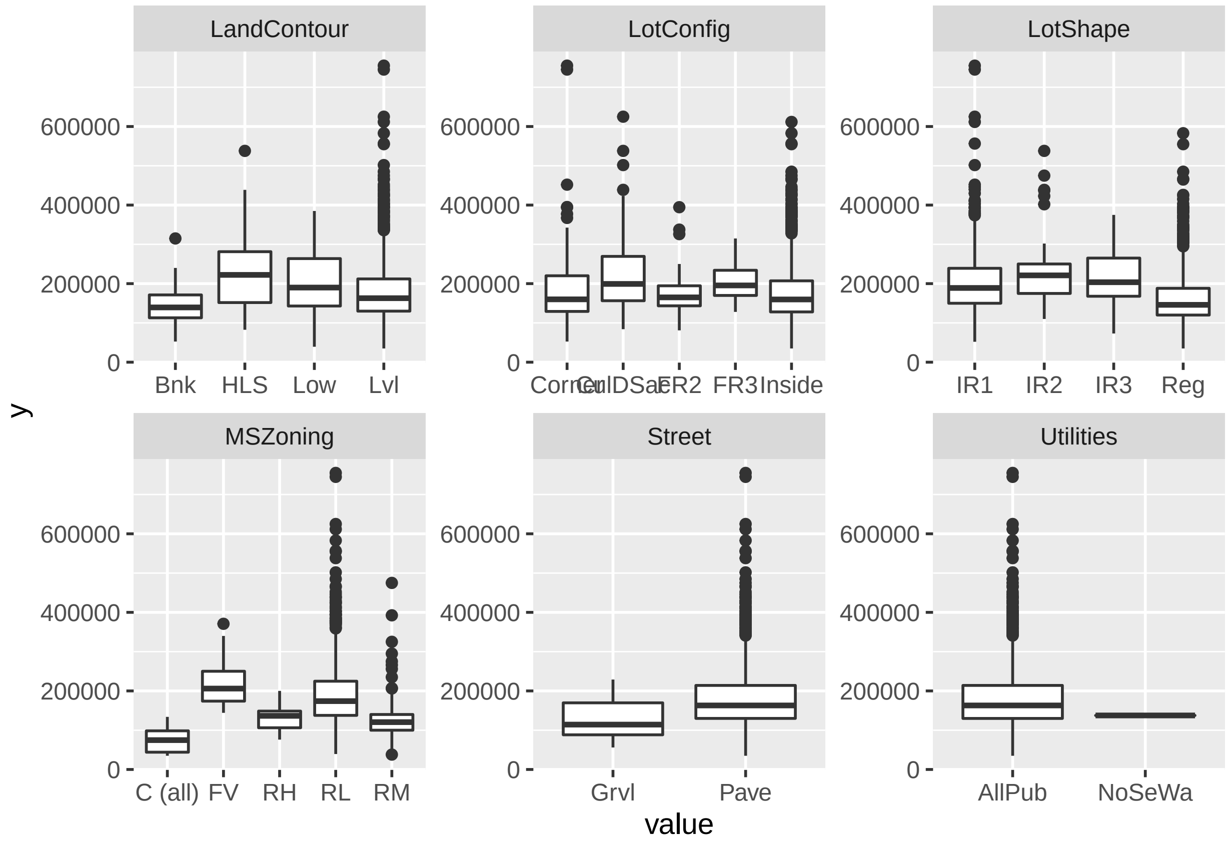
```r
signif_num_vars = numericas[result_cor_test < 0.05]
signif_num_vars %>% head()
```

```
##   MSSubClass LotFrontage LotArea OverallQual OverallCond YearBuilt YearRemodAdd
## 1         60          65    8450           7           5      2003         2003
## 2         20          80    9600           6           8      1976         1976
## 3         60          68   11250           7           5      2001         2002
## 4         70          60    9550           7           5      1915         1970
## 5         60          84   14260           8           5      2000         2000
## 6         50          85   14115           5           5      1993         1995
##   MasVnrArea BsmtFinSF1 BsmtUnfSF TotalBsmtSF X1stFlrSF X2ndFlrSF GrLivArea
## 1        196        706       150         856       856       854      1710
## 2          0        978       284        1262      1262         0      1262
## 3        162        486       434         920       920       866      1786
## 4          0        216       540         756       961       756      1717
## 5        350        655       490        1145      1145      1053      2198
## 6          0        732        64         796       796       566      1362
##   BsmtFullBath FullBath HalfBath BedroomAbvGr KitchenAbvGr TotRmsAbvGrd
## 1            1        2        1            3            1            8
## 2            0        2        0            3            1            6
## 3            1        2        1            3            1            6
## 4            1        1        0            3            1            7
## 5            1        2        1            4            1            9
## 6            1        1        1            1            1            5
##   Fireplaces GarageYrBlt GarageCars GarageArea WoodDeckSF OpenPorchSF
## 1          0        2003          2        548          0          61
## 2          1        1976          2        460        298           0
## 3          1        2001          2        608          0          42
## 4          1        1998          3        642          0          35
## 5          1        2000          3        836        192          84
## 6          0        1993          2        480         40          30
##   EnclosedPorch ScreenPorch PoolArea
## 1             0           0        0
## 2             0           0        0
## 3             0           0        0
## 4           272           0        0
## 5             0           0        0
## 6             0           0        0
```

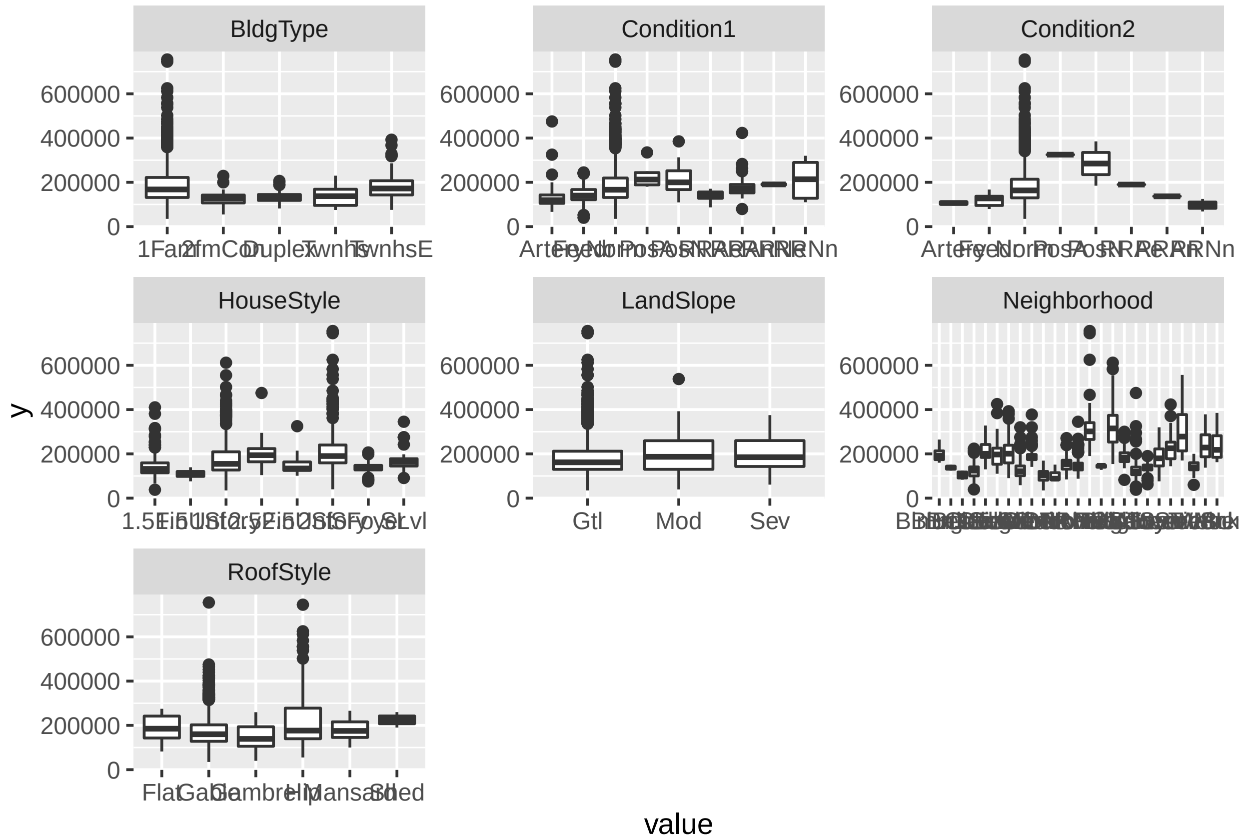
Boxplot das covariáveis categóricas em relação à variável resposta

Como o intuito é verificar mais se dentro das covariáveis alguma varíavel apresenta maior influência que as outras, os nomes dentro das variáveis ficou corrompido, por isso, caso haja necessidade de ver algum covariável com mais detalhe posso criar um gráfico só pra ela.

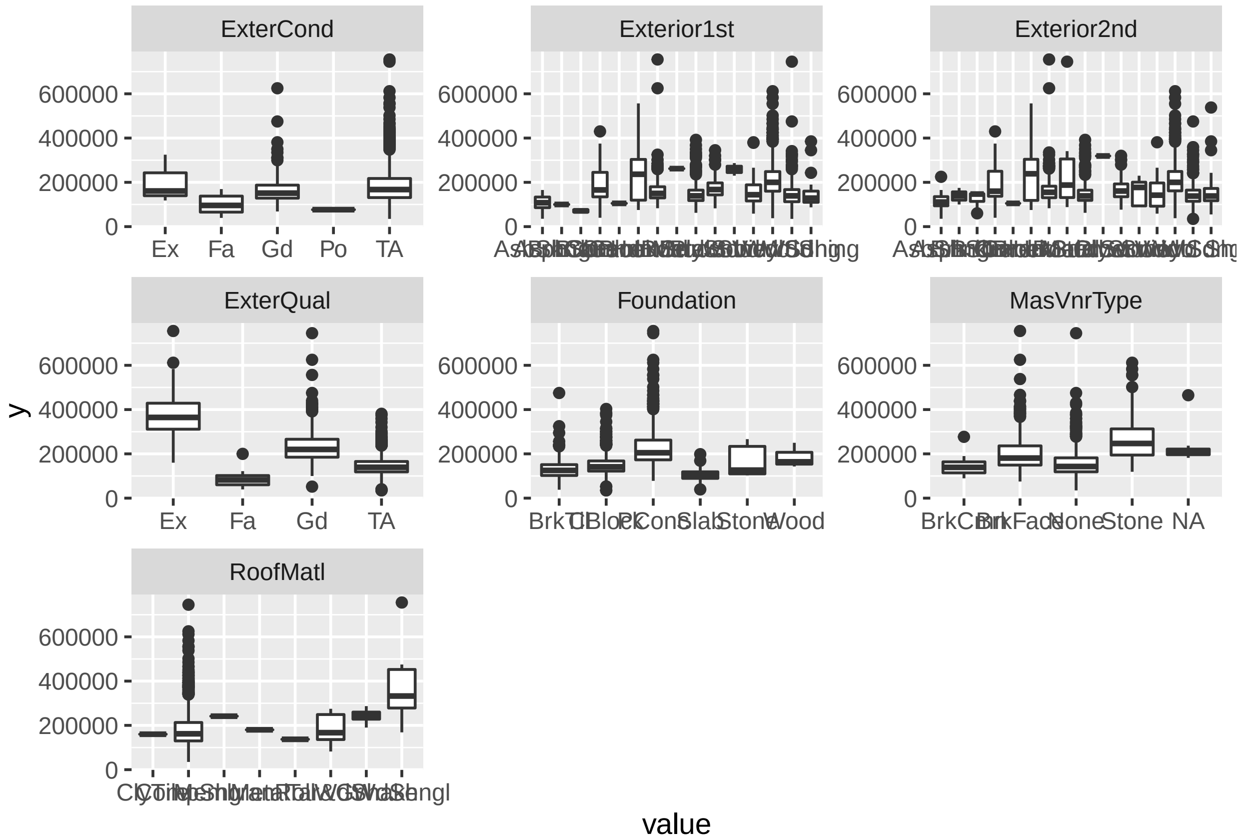
```r
cbind(categoricas[1:6],y) %>%
  pivot_longer(-y) %>%
  ggplot(aes(x = value, y = y))+
    geom_boxplot()+
    facet_wrap(~name, scales = 'free')
```



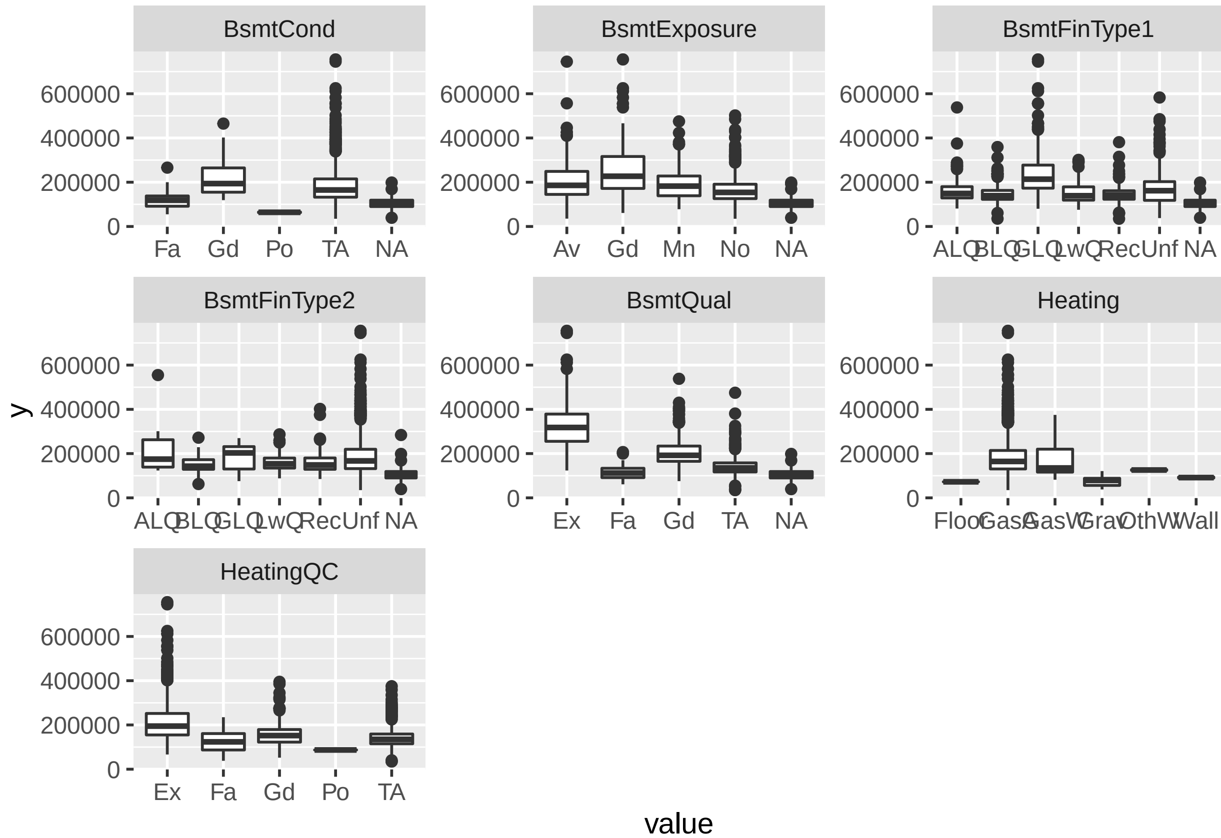

```
cbind(categoricas[7:13],y) %>%
  pivot_longer(-y) %>%
  ggplot(aes(x = value, y = y))+
    geom_boxplot()+
    facet_wrap(~name, scales = 'free')
```



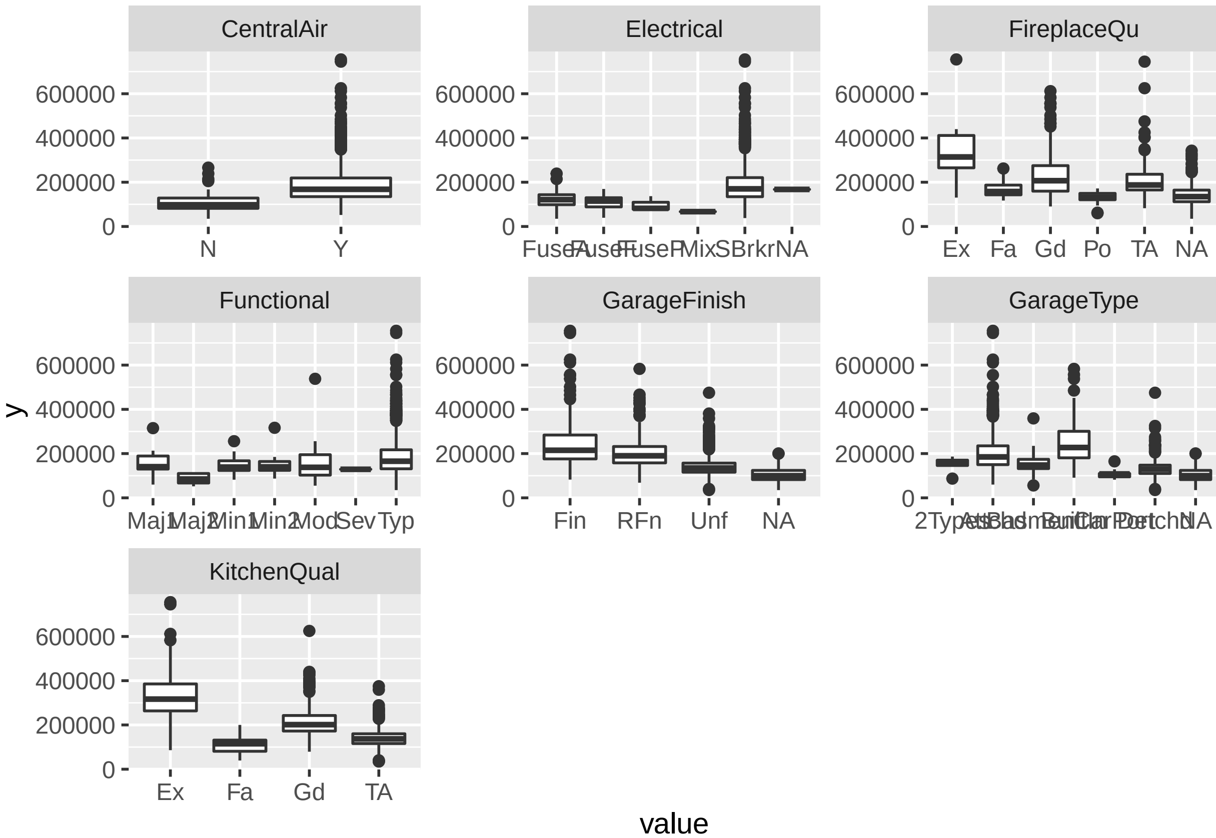

```
cbind(categoricas[14:20],y) %>%
  pivot_longer(-y) %>%
  ggplot(aes(x = value, y = y))+
    geom_boxplot()+
    facet_wrap(~name, scales = 'free')
```



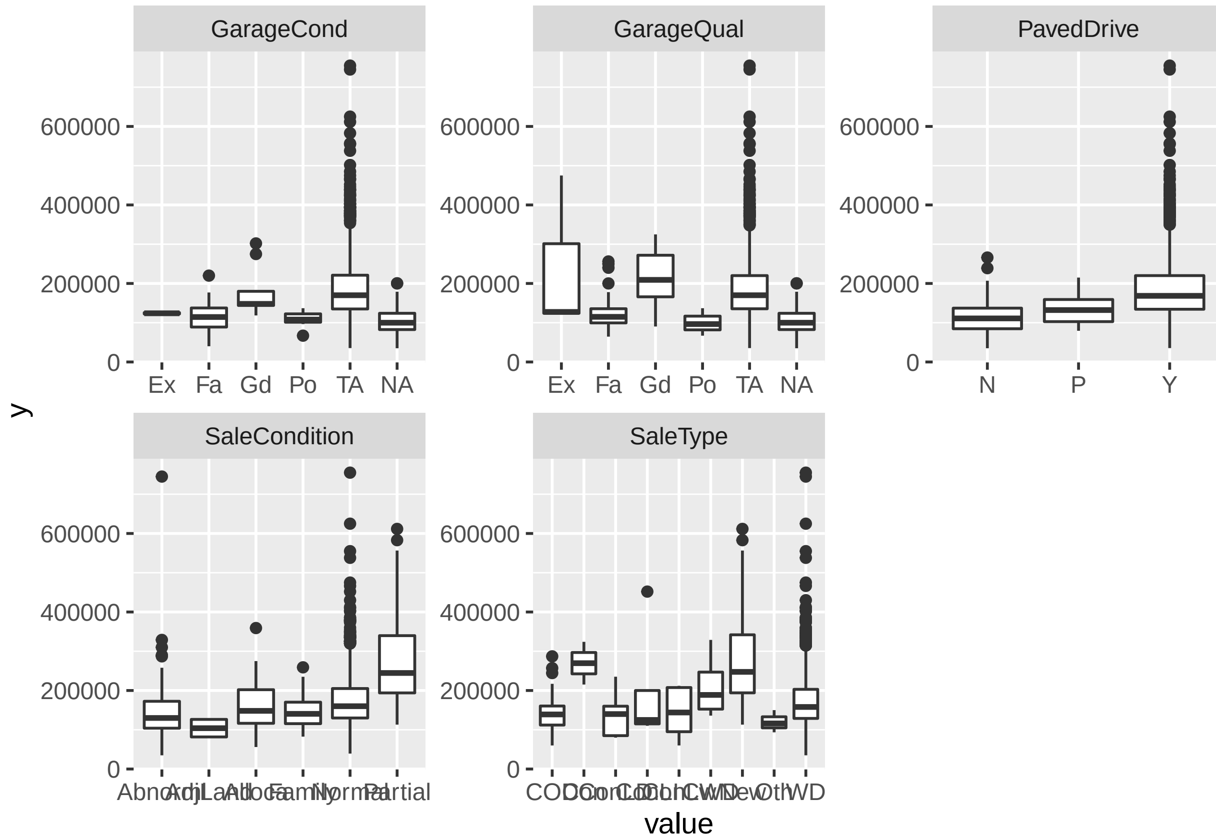

```
cbind(categoricas[21:27],y) %>%
  pivot_longer(-y) %>%
  ggplot(aes(x = value, y = y))+
    geom_boxplot()+
    facet_wrap(~name, scales = 'free')
```




```
cbind(categoricas[28:34],y) %>%
  pivot_longer(-y) %>%
  ggplot(aes(x = value, y = y))+
    geom_boxplot()+
    facet_wrap(~name, scales = 'free')
```




```
cbind(categoricas[35:39],y) %>%
  pivot_longer(-y) %>%
  ggplot(aes(x = value, y = y))+
    geom_boxplot()+
    facet_wrap(~name, scales = 'free')
```



Fazendo o teste da ANOVA

```r
data_cat = cbind(categoricas,y)
result_anova = summary(aov(y ~ ., data = data_cat))
```

```r
# pegando somente as variáveis que tiveram um pvalor abaixo de 0,05
result_anova = result_anova[[1]]
signif_cat_vars = result_anova[result_anova$`Pr(>F)` < 0.05,]

#variáveis categóricas significantes
signif_cat_vars
```

```
##               Df        Sum Sq     Mean Sq F value                Pr(>F)
## MSZoning       3   247982781161 82660927054 51.1904 < 0.0000000000000022 ***
## LotShape       3   209563189418 69854396473 43.2596 < 0.0000000000000022 ***
## LandContour    3   113643260011 37881086670 23.4591  0.0000000000025110 ***
## Utilities      1     8695058870  8695058870  5.3847            0.0206675 *
## LotConfig      4    35566747349  8891686837  5.5065            0.0002355 ***
## LandSlope      2     9784324847  4892162423  3.0296            0.0491155 *
## Neighborhood  24 2258901058219 94120877426 58.2873 < 0.0000000000000022 ***
## Condition1     7    26356121924  3765160275  2.3317            0.0236741 *
## Condition2     4    34932009471  8733002368  5.4082            0.0002802 ***
## BldgType       4   269236101153 67309025288 41.6833 < 0.0000000000000022 ***
```

```
## HouseStyle     7   58436841231  8348120176  5.1698  0.000010097313630804 ***
## RoofStyle      5  176488094453 35297618891 21.8592 < 0.00000000000000022 ***
## RoofMatl       6  179337222622 29889537104 18.5100 < 0.00000000000000022 ***
## Exterior1st   11  188238664474 17112605861 10.5975 < 0.00000000000000022 ***
## Exterior2nd   14   79894901666  5706778690  3.5341  0.000013467699172397 ***
## MasVnrType     3  102172368685 34057456228 21.0912  0.000000000000575464 ***
## ExterQual      3  180701269991 60233756664 37.3017 < 0.00000000000000022 ***
## BsmtQual       3  118553690374 39517896791 24.4727  0.000000000000006632 ***
## BsmtExposure   3  110670296940 36890098980 22.8454  0.000000000000056377 ***
## BsmtFinType1   5   34814719546  6962943909  4.3120             0.0007372 ***
## KitchenQual    3   71540598180 23846866060 14.7679  0.000000002852314055 ***
## SaleType       7   24721413902  3531630557  2.1871             0.0338436 *
## SaleCondition  4   41007437442 10251859361  6.3488  0.000053002355155008 ***
## NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
#Pegando o nome das variáveis numéricas e categóricas
names_signif_vars_cat = signif_cat_vars %>% row.names %>% str_trim()
names_signif_vars_cat = names_signif_vars_cat[names_signif_vars_cat!="NA"]

names_signif_vars_num = signif_num_vars %>% colnames

signif_vars = c(names_signif_vars_cat, names_signif_vars_num)

#Covariáveis significantes
signif_vars
```

```
##  [1] "MSZoning"      "LotShape"      "LandContour"   "Utilities"
##  [5] "LotConfig"     "LandSlope"     "Neighborhood"  "Condition1"
##  [9] "Condition2"    "BldgType"      "HouseStyle"    "RoofStyle"
## [13] "RoofMatl"      "Exterior1st"   "Exterior2nd"   "MasVnrType"
## [17] "ExterQual"     "BsmtQual"      "BsmtExposure"  "BsmtFinType1"
## [21] "KitchenQual"   "SaleType"      "SaleCondition" "MSSubClass"
## [25] "LotFrontage"   "LotArea"       "OverallQual"   "OverallCond"
## [29] "YearBuilt"     "YearRemodAdd"  "MasVnrArea"    "BsmtFinSF1"
## [33] "BsmtUnfSF"     "TotalBsmtSF"   "X1stFlrSF"     "X2ndFlrSF"
## [37] "GrLivArea"     "BsmtFullBath"  "FullBath"      "HalfBath"
## [41] "BedroomAbvGr"  "KitchenAbvGr"  "TotRmsAbvGrd"  "Fireplaces"
## [45] "GarageYrBlt"   "GarageCars"    "GarageArea"    "WoodDeckSF"
## [49] "OpenPorchSF"   "EnclosedPorch" "ScreenPorch"   "PoolArea"
```

Fazendo os testes de correlação e da anova, conseguiu-se reduzir o número de variáveis de 80 para 53.

Com isso, para o restante das análises serão utilizadas essas variáveis.

Criação das variáveis dummies

```r
library(fastDummies)

n_treino = nrow(df_treino)
n_test = nrow(df_test)
```

```r
df_geral = rbind(df_treino, df_test)


#aplicando a técnica no dataframe de treino e teste juntos
df_geral = df_geral[signif_vars]
df_geral = dummy_cols(df_geral, select_columns = names_signif_vars_cat,
                      remove_first_dummy = T)

treino = df_geral[1:n_treino, ]
teste = df_geral[n_treino:nrow(df_geral),]


treino = treino %>%
  mutate(y = y) %>%
  na.omit %>%
  dplyr::select(!names_signif_vars_cat)


#dimensão do dataframe de treino depois de todo o processo de avaliação e criação de variáveis
dim(treino)
```

```
## [1] 1096  181
```

Transformando as variáveis categóricas em variáveis dummies aumentamos o número de variáveis do modelo de 53 para 181 variáveis

Seleção do modelo

Usando a fórmula do backward usando o critério de aic chegamos nos seguintes resultados:

```r
library(MASS)


model = lm(y~., data = treino)
best_model = stepAIC(model, direction = 'backward')
```

Número de variáveis selecionadas:

```r
best_model$coefficients %>% names %>% length
```

```
## [1] 87
```

Variáveis selecionadas:

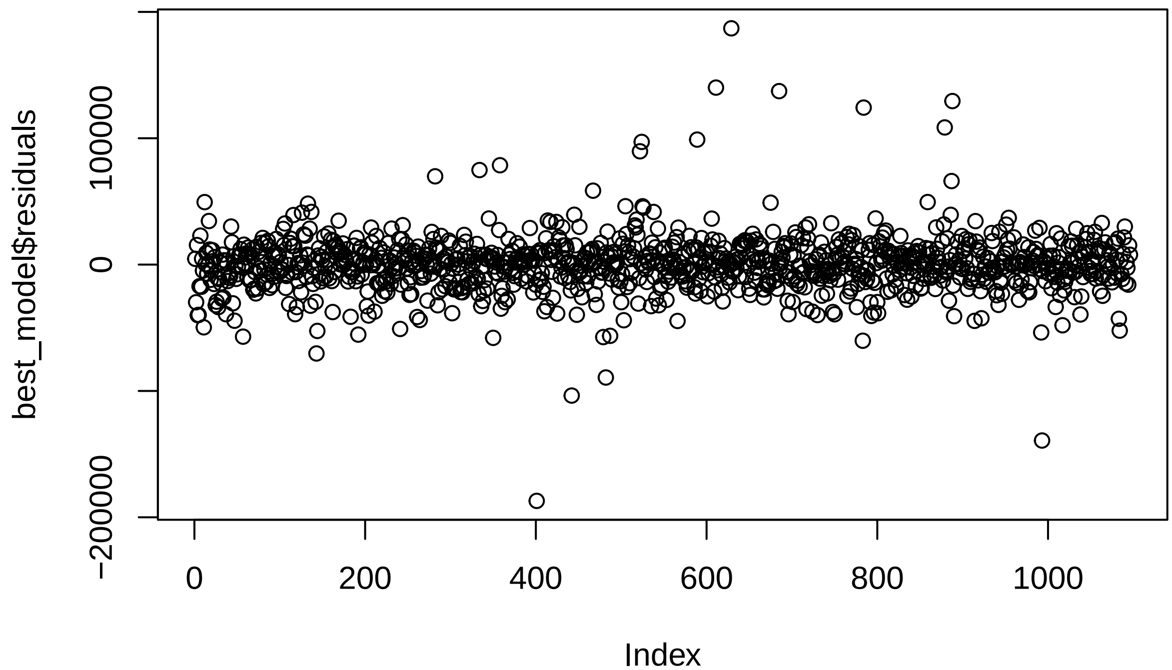
```r
best_model$coefficients %>% names
```

```
##  [1] "(Intercept)"           "LotFrontage"           "LotArea"
##  [4] "OverallQual"           "OverallCond"           "YearBuilt"
##  [7] "MasVnrArea"            "BsmtFinSF1"            "TotalBsmtSF"
## [10] "X2ndFlrSF"             "GrLivArea"             "FullBath"
```

```
## [13] "BedroomAbvGr"          "KitchenAbvGr"          "Fireplaces"
## [16] "GarageYrBlt"           "GarageCars"            "ScreenPorch"
## [19] "PoolArea"              "MSZoning_FV"           "MSZoning_RH"
## [22] "MSZoning_RL"           "MSZoning_RM"           "LotShape_IR2"
## [25] "LotShape_Reg"          "LandContour_HLS"       "LandContour_Low"
## [28] "LandContour_Lvl"       "LotConfig_CulDSac"     "LotConfig_FR2"
## [31] "LandSlope_Sev"         "Neighborhood_ClearCr"  "Neighborhood_CollgCr"
## [34] "Neighborhood_Crawfor"  "Neighborhood_Edwards"  "Neighborhood_Gilbert"
## [37] "Neighborhood_Mitchel"  "Neighborhood_NAmes"    "Neighborhood_NoRidge"
## [40] "Neighborhood_NridgHt"  "Neighborhood_NWAmes"   "Neighborhood_OldTown"
## [43] "Neighborhood_StoneBr"  "Neighborhood_Timber"   "Condition1_Norm"
## [46] "Condition1_RRAe"       "Condition2_PosA"       "Condition2_PosN"
## [49] "BldgType_Duplex"       "BldgType_Twnhs"        "BldgType_TwnhsE"
## [52] "HouseStyle_1Story"     "RoofStyle_Gable"       "RoofStyle_Gambrel"
## [55] "RoofStyle_Hip"         "RoofStyle_Mansard"     "RoofMatl_CompShg"
## [58] "RoofMatl_Membran"      "RoofMatl_Roll"         "`RoofMatl_Tar&Grv`"
## [61] "RoofMatl_WdShake"      "RoofMatl_WdShngl"      "Exterior1st_BrkComm"
## [64] "Exterior1st_BrkFace"   "Exterior1st_HdBoard"   "Exterior1st_ImStucc"
## [67] "Exterior1st_Plywood"   "`Exterior2nd_Brk Cmn`" "Exterior2nd_ImStucc"
## [70] "Exterior2nd_MetalSd"   "MasVnrType_None"       "MasVnrType_Stone"
## [73] "ExterQual_Fa"          "ExterQual_Gd"          "ExterQual_TA"
## [76] "BsmtQual_Fa"           "BsmtQual_Gd"           "BsmtQual_TA"
## [79] "BsmtExposure_Gd"       "BsmtExposure_Mn"       "BsmtExposure_No"
## [82] "BsmtFinType1_GLQ"      "KitchenQual_Fa"        "KitchenQual_Gd"
## [85] "KitchenQual_TA"        "SaleType_Con"          "SaleCondition_Partial"
```

Gráfico de resíduos:

```r
plot(best_model$residuals)
```



Preparando os dados de teste para a predição:

```
teste = teste %>%
  dplyr::select(!names_signif_vars_cat)
```

Fazendo a avaliação do modelo:

```
library(forecast)

y_true = y
y_pred = predict(best_model, treino)

accuracy(y_true, y_pred)
```

```
##                  ME     RMSE      MAE       MPE     MAPE
## Test set 5720.299 113491.1 83389.32 -13.57612 48.12655
```