

# Trabalho de Probabilidade e Estatística

Brainer Sueverti de Campos - 790829 | Thiago Martins - 790964

20/04/2022

```
#install.packages("readr");  
#install.packages("ggplot2");  
#install.packages("dplyr");  
#install.packages("hrbrthemes");  
#install.packages("tidyverse");  
#library(readr)  
#library(ggplot2)  
#library(dplyr)  
#library(hrbrthemes)  
#library(tidyverse)
```

As análises Univariadas foram feitas nas 5 variáveis (SalePrice, OverallQual, OverallCond, YearBuilt e GarageCars). Calculamos média, mediana, intervalo, distância interquartilica, variância, desvio padrão, tabela de frequência, boxplot, histograma e gráfico de densidade.

OverallQual (índice de qualidade geral)

A variável OverallQual está levemente à direita ao analisar o gráfico de densidade, histograma e o boxplot. Analisando numericamente vemos que os valores à direita na tabela de frequência estão maiores também.

```
cat("Média: ", mean(overallqual), "\n")
```

```
## Média: 6.099315
```

```
cat("Mediana:", median(overallqual), "\n")
```

```
## Mediana: 6
```

```
cat("Intervalo:", max(overallqual) - min(overallqual), "\n")
```

```
## Intervalo: 9
```

```
cat("Distancia Interquartilica:",IQR(overallqual),'\n')
```

```
## Distancia Interquartilica: 2
```

```
cat('Variância:',var(overallqual),'\n')
```

```
## Variância: 1.912679
```

```
cat('Desvio Padrão:',sd(overallqual),'\n')
```

```
## Desvio Padrão: 1.382997
```

```
cat('\nTabela de Frequência')
```

```
##
```

```
## Tabela de Frequência
```

```
table(overallqual)
```

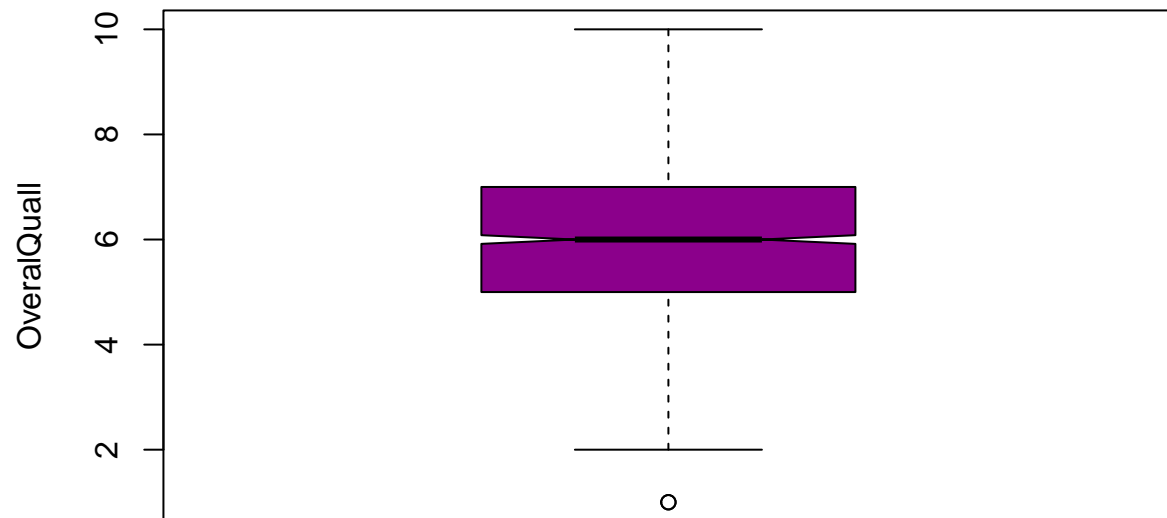
```
## overallqual
```

```
##   1   2   3   4   5   6   7   8   9  10
```

```
##   2   3  20 116 397 374 319 168  43  18
```

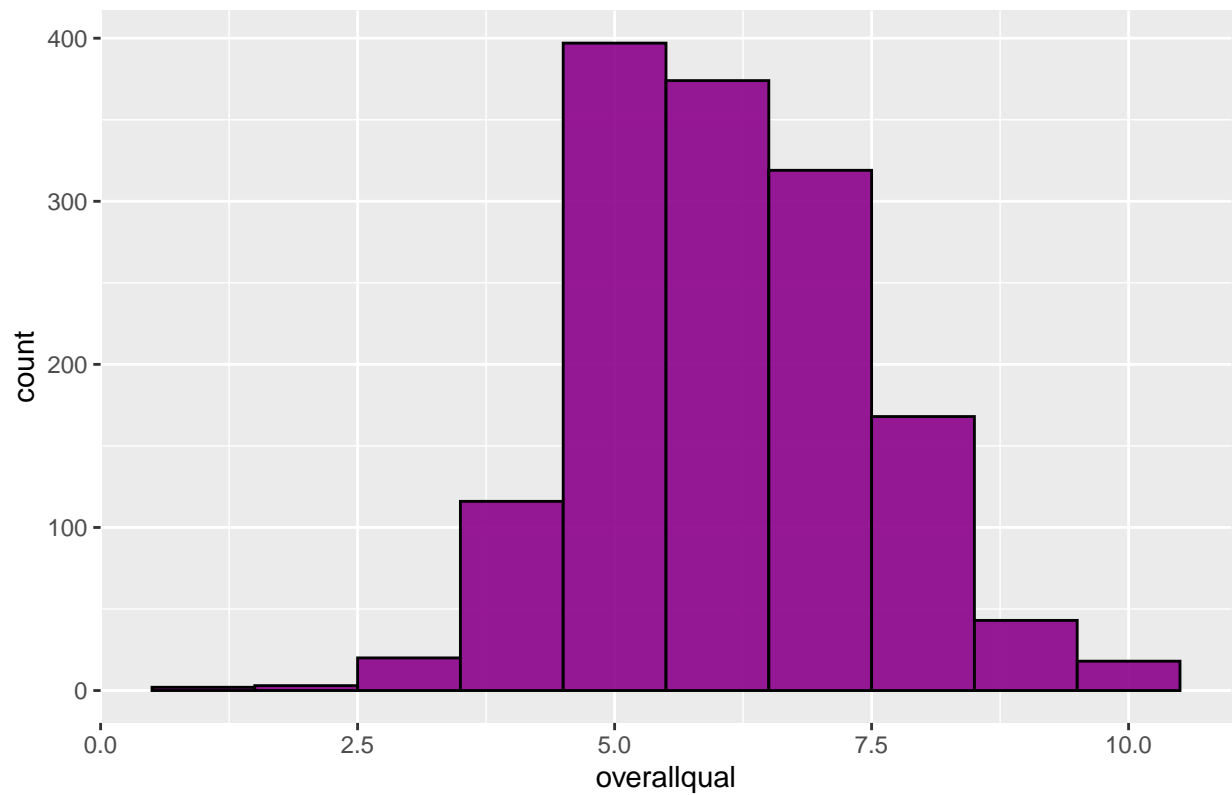
```
boxplot(overallqual,  
main = "BoxPlot - OVERALLQUALL",  
ylab = "OveralQual1",  
col = "#8b008b",  
border = "black",  
notch = TRUE  
)
```

## BoxPlot – OVERALLQUALL

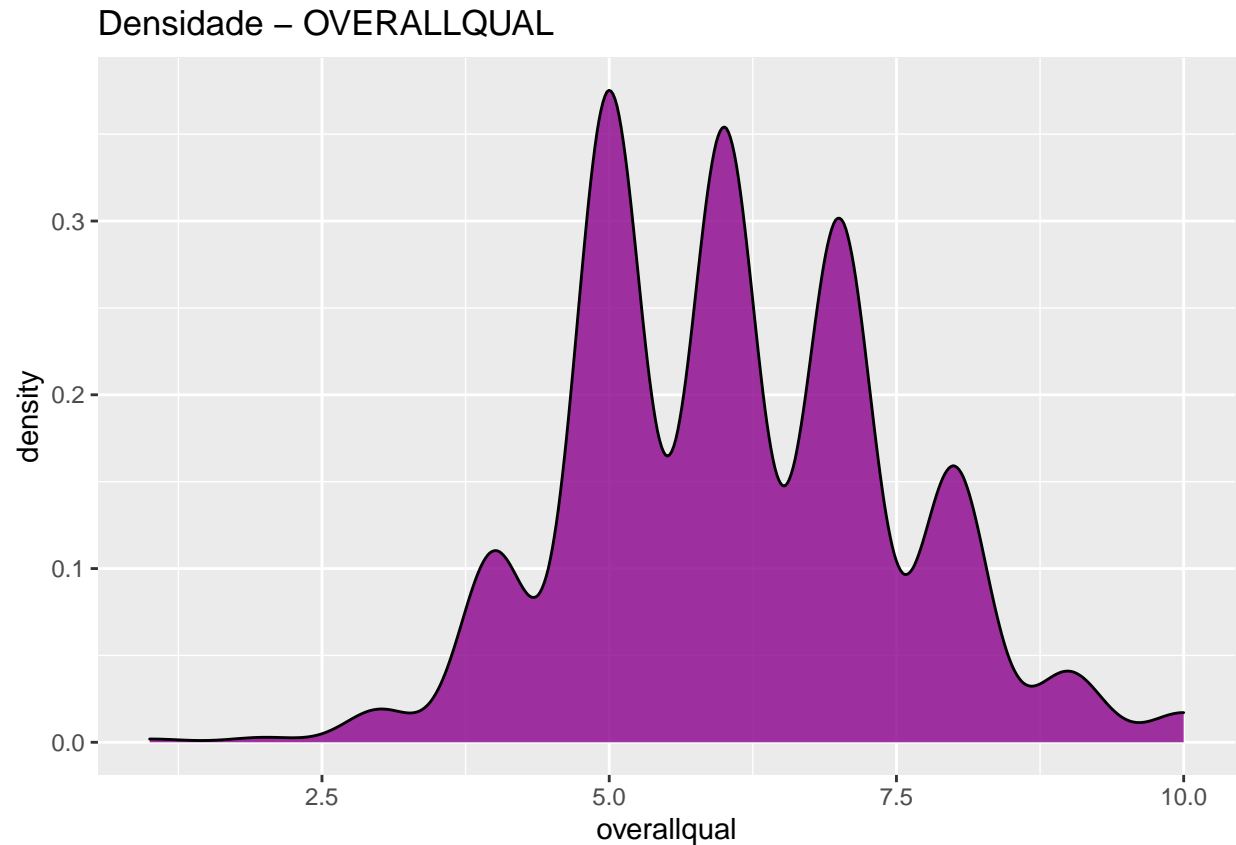


```
p <- data %>%  
  ggplot( aes(x=overallqual)) +  
    geom_histogram( binwidth=1, fill="#8b008b", color="#000000", alpha=0.9) +  
    ggtitle('Histograma - OVERALLQUAL') +  
    theme(  
      plot.title = element_text(size=15)  
    )  
p
```

Histograma – OVERALLQUAL



```
data %>%  
  ggplot( aes(x=overallqual)) +  
    geom_density(fill="#8b008b", color="#000000", alpha=0.8) +  
    ggtitle("Densidade - OVERALLQUAL")
```



OverrralCond(índice do estado do imóvel)

A variável OverrralCond segue a mesma tendência da Variável OverrralQual com um leve deslocamento à direita, indicado uma leve tendência a imóveis em bom estado.

```
cat("Média: ", mean(overallcond), "\n")
```

```
## Média: 5.575342
```

```
cat("Mediana:", median(overallcond), "\n")
```

```
## Mediana: 5
```

```
cat("Intervalo:", max(overallcond) - min(overallcond), "\n")
```

```
## Intervalo: 8
```

```
cat("Distancia Interquartilica:", IQR(overallcond), '\n')
```

```
## Distancia Interquartilica: 1
```

```
cat('Variância:',var(overallcond),'\n')
```

```
## Variância: 1.238322
```

```
cat('Desvio Padrão:',sd(overallcond),'\n')
```

```
## Desvio Padrão: 1.112799
```

```
cat('\nTabela de Frequência')
```

```
##
```

```
## Tabela de Frequência
```

```
table(overallcond)
```

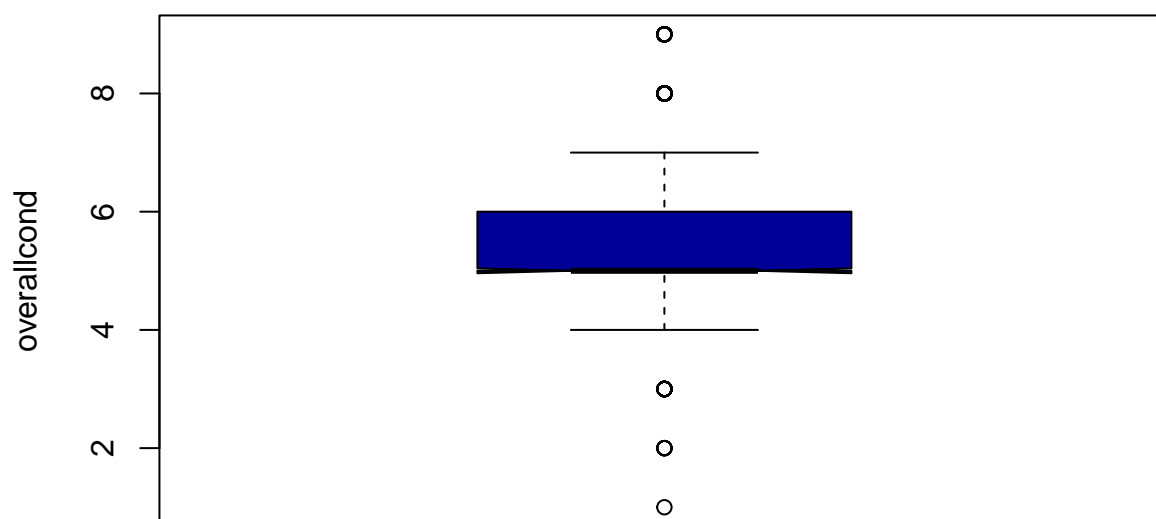
```
## overallcond
```

```
##  1  2  3  4  5  6  7  8  9
```

```
##  1  5 25 57 821 252 205 72 22
```

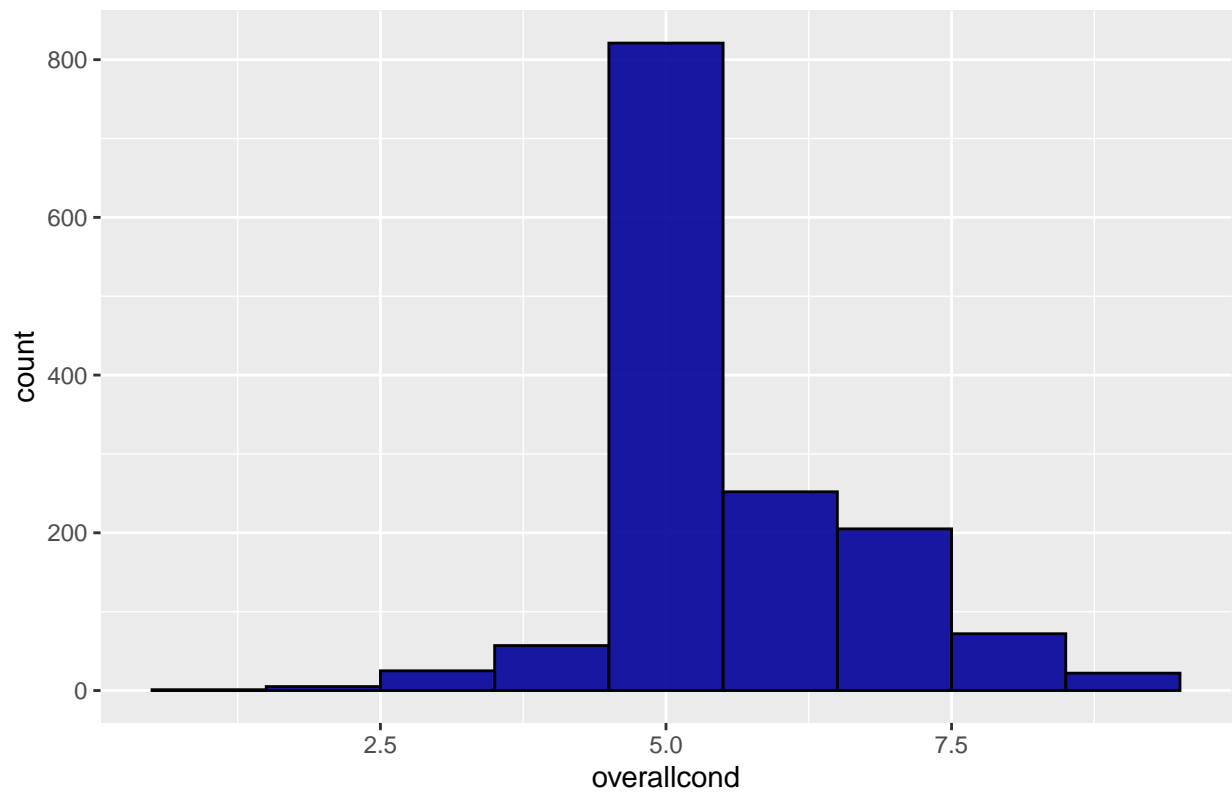
```
boxplot(overallcond,  
main = "BoxPlot - OVERALLCOND",  
ylab = "overallcond",  
col = "#000099",  
border = "black",  
notch = TRUE  
)
```

## BoxPlot – OVERALLCOND



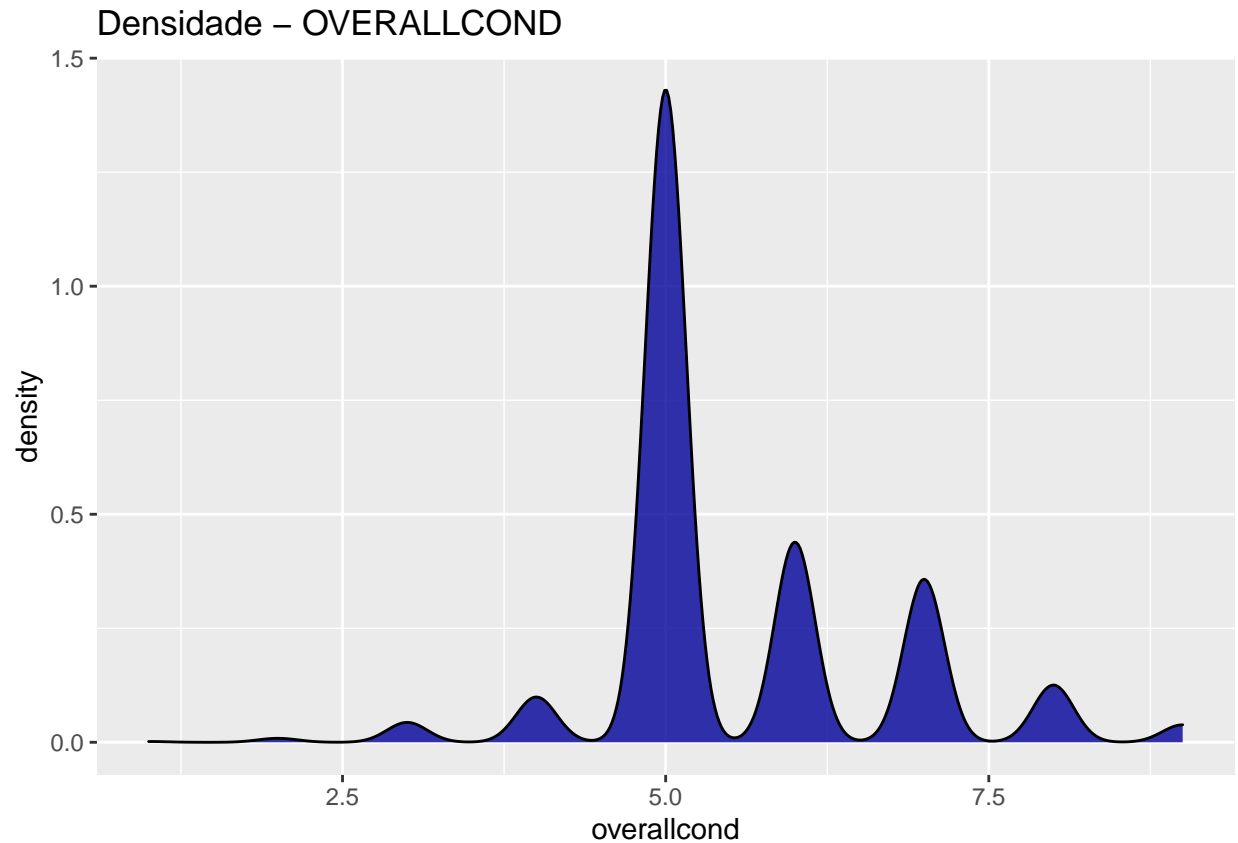
```
p <- data %>%
  ggplot( aes(x=overallcond)) +
    geom_histogram( binwidth=1, fill="#000099", color="#000000", alpha=0.9) +
    ggtitle('Histograma - OVERALLCOND') +
    theme(
      plot.title = element_text(size=15)
    )
p
```

Histograma – OVERALLCOND



```
data %>%  
  ggplot( aes(x=overallcond)) +  
    geom_density(fill="#000099", color="#000000", alpha=0.8) +  
    ggtitle("Densidade - OVERALLCOND")
```





**YearBuilt(anos em que casa foi construída)**

A variável YearBuilt tem um comportamento diferente, pois ela aumenta sua frequência ao aumentar o ano. Além disso, os gráficos mostram uma tendência total para direita com um comportamento crescente, prova disso numericamente são a média e a mediana (1971.268, 1973) em um intervalo de 1872 e 2010.

```
cat("Média: ", mean(yearbuilt), "\n")
```

```
## Média: 1971.268
```

```
cat("Mediana:", median(yearbuilt), "\n")
```

```
## Mediana: 1973
```

```
cat("Intervalo:", max(yearbuilt) - min(yearbuilt), "\n")
```

```
## Intervalo: 138
```

```
cat("Distancia Interquartilica:",IQR(yearbuilt),'\n')
```

```
## Distancia Interquartilica: 46
```

```
cat('Variância:',var(yearbuilt),'\n')
```

```
## Variância: 912.2154
```

```
cat('Desvio Padrão:',sd(yearbuilt),'\n')
```

```
## Desvio Padrão: 30.2029
```

```
cat('\nTabela de Frequência')
```

```
##
```

```
## Tabela de Frequência
```

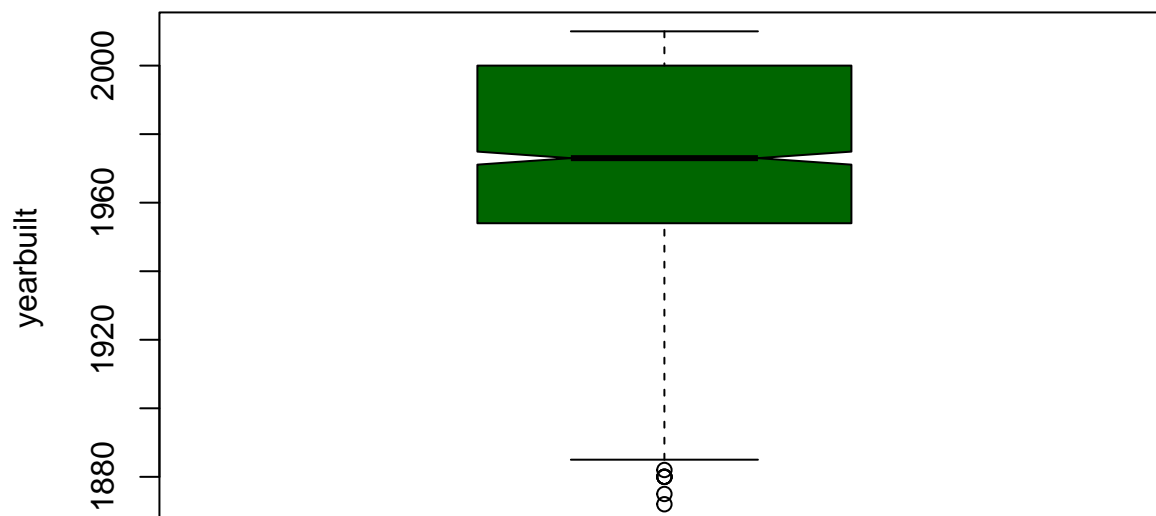
```
table(yearbuilt)
```

```
## yearbuilt
```

```
## 1872 1875 1880 1882 1885 1890 1892 1893 1898 1900 1904 1905 1906 1908 1910 1911
##    1    1    4    1    2    2    2    1    1   10    1    1    1    2   17    1
## 1912 1913 1914 1915 1916 1917 1918 1919 1920 1921 1922 1923 1924 1925 1926 1927
##    3    1    7   10    8    1    7    3   30    6    8    7    7   16    9    3
## 1928 1929 1930 1931 1932 1934 1935 1936 1937 1938 1939 1940 1941 1942 1945 1946
##    7    4    9    6    4    3    6    9    5    4    8   18   15    2    6    7
## 1947 1948 1949 1950 1951 1952 1953 1954 1955 1956 1957 1958 1959 1960 1961 1962
##    5   14   12   20    6    5   12   24   16   14   20   24   26   17   14   19
## 1963 1964 1965 1966 1967 1968 1969 1970 1971 1972 1973 1974 1975 1976 1977 1978
##   16   15   24   18   16   22   14   24   22   23   11   10    8   33   32   16
## 1979 1980 1981 1982 1983 1984 1985 1986 1987 1988 1989 1990 1991 1992 1993 1994
##    9   10    5    6    4    9    5    5    3   11    3   12    5   13   17   19
## 1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010
##   18   15   14   25   25   24   20   23   45   54   64   67   49   23   18    1
```

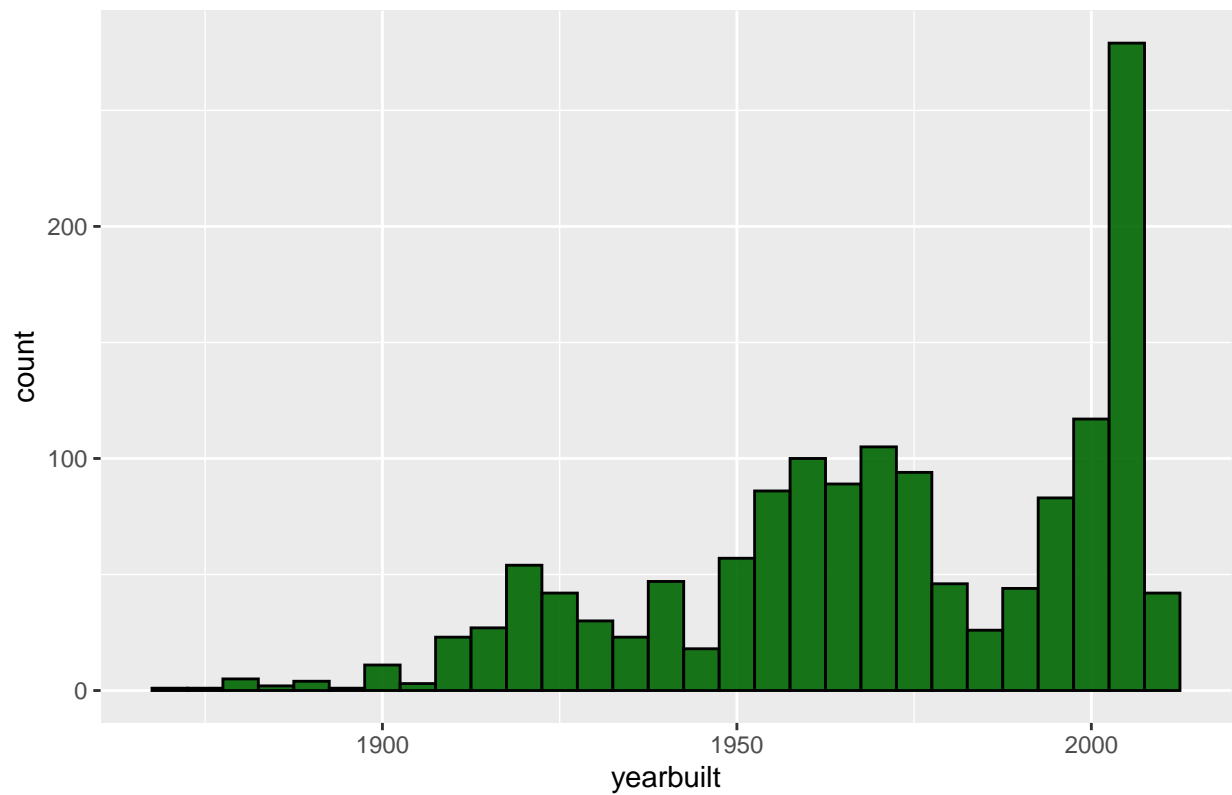
```
boxplot(yearbuilt,
main = "BoxPlot - YEARBUILT",
ylab = "yearbuilt",
col = "#006600",
border = "black",
notch = TRUE
)
```

## BoxPlot – YEARBUILT

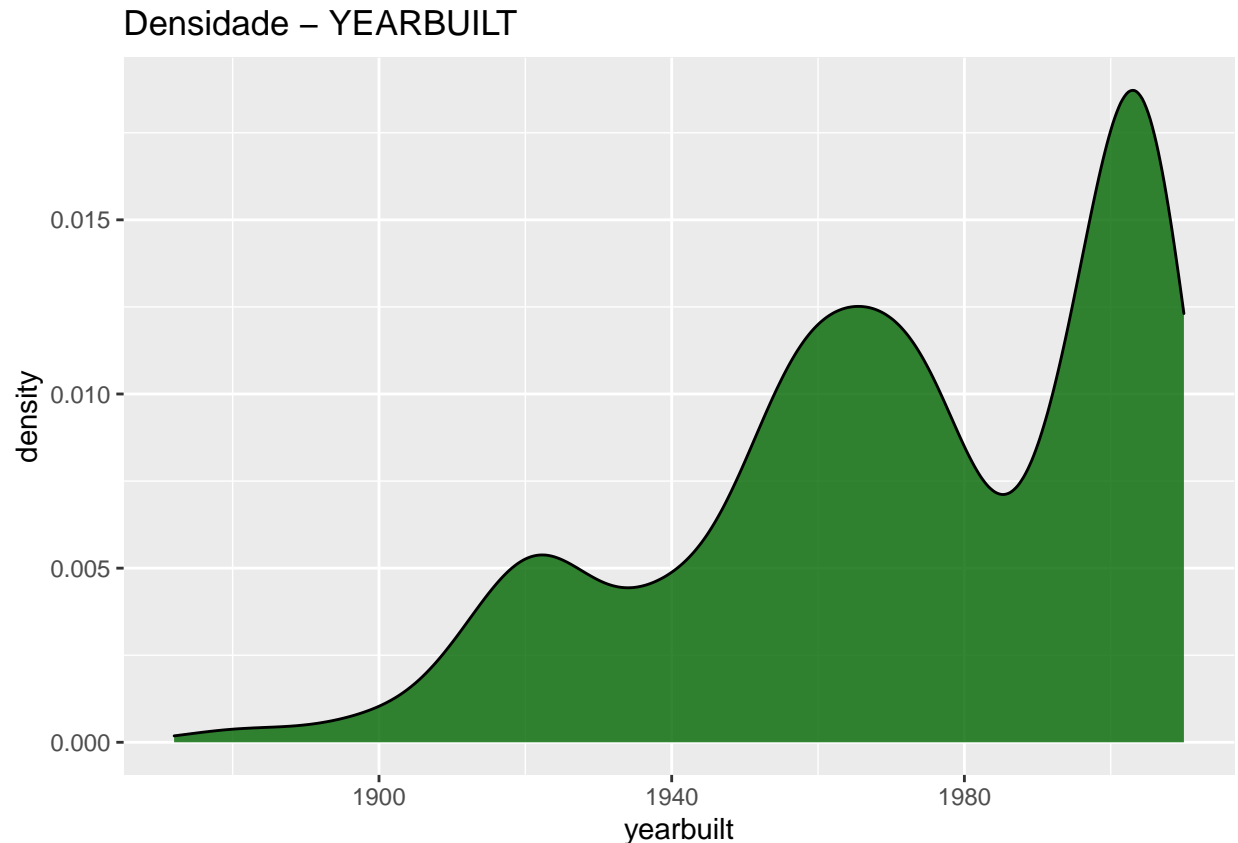


```
p <- data %>%  
  ggplot( aes(x=yearbuilt)) +  
    geom_histogram( binwidth=5, fill="#006600", color="#000000", alpha=0.9) +  
    ggtitle('Histograma - YEARBUILT') +  
    theme(  
      plot.title = element_text(size=15)  
    )  
p
```

## Histograma – YEARBUILT



```
data %>%  
  ggplot( aes(x=yearbuilt)) +  
    geom_density(fill="#006600", color="#000000", alpha=0.8) +  
    ggtitle("Densidade - YEARBUILT")
```



GarageCars(quantidades de carro que cabem na garagem da casa)

A variável GarageCars segue um comportamento contrário às outras variáveis ao analisar os gráficos, visto que ela tem uma tendência levemente para a esquerda, ou seja, há uma frequência um pouco maior para garagens que cabem menos carros. Contudo, a mediana mostra que uma centralidade dos valores.

```
cat("Média: ", mean(GarageCars), "\n")
```

```
## Média: 1.767123
```

```
cat("Mediana:", median(GarageCars), "\n")
```

```
## Mediana: 2
```

```
cat("Intervalo:", max(GarageCars) - min(GarageCars), "\n")
```

```
## Intervalo: 4
```

```
cat("Distancia Interquartilica:", IQR(GarageCars), '\n')
```

```
## Distancia Interquartilica: 1
```

```
cat('Variância:', var(GarageCars), '\n')
```

```
## Variância: 0.5584797
```

```
cat('Desvio Padrão:', sd(GarageCars), '\n')
```

```
## Desvio Padrão: 0.747315
```

```
cat('\nTabela de Frequência')
```

```
##
```

```
## Tabela de Frequência
```

```
table(GarageCars)
```

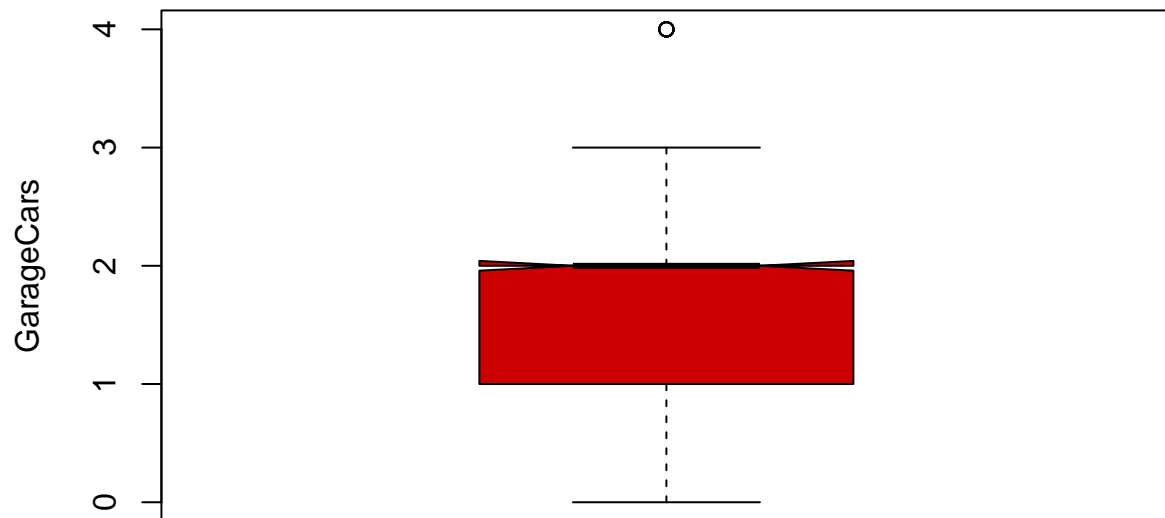
```
## GarageCars
```

```
##    0    1    2    3    4
```

```
##  81 369 824 181    5
```

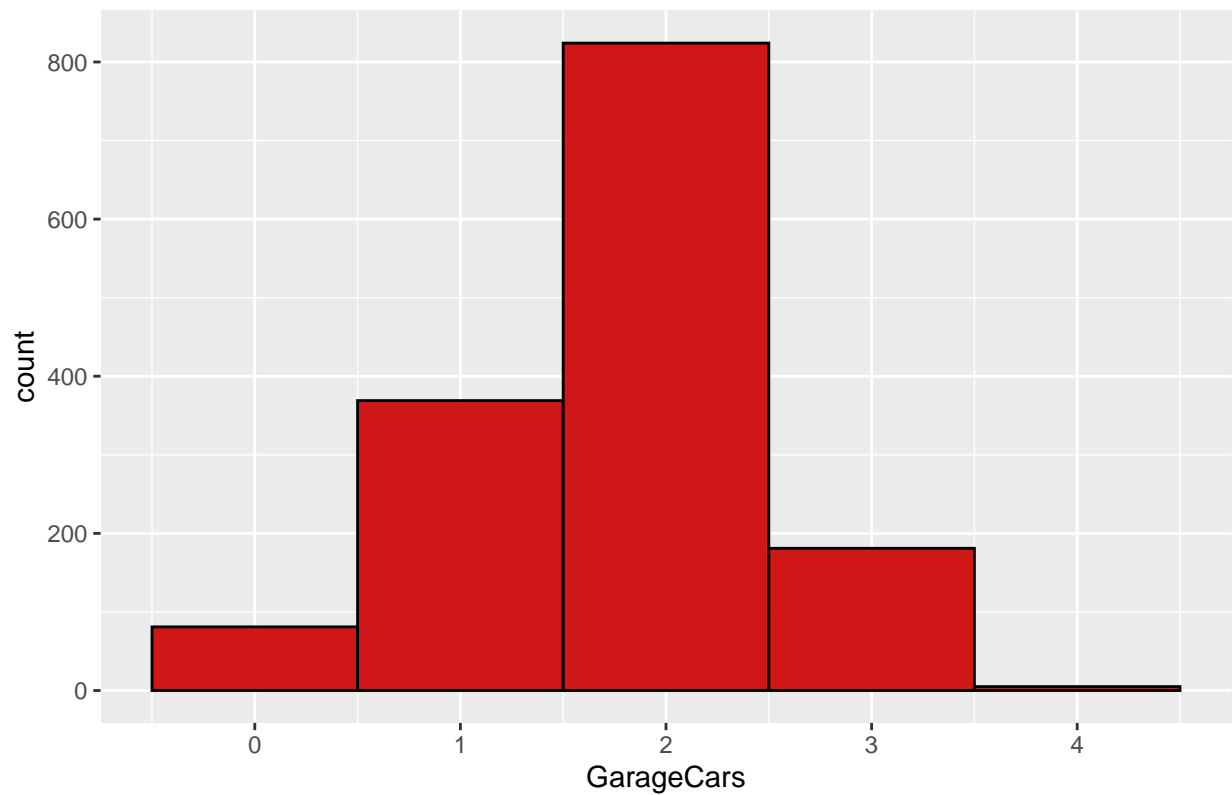
```
boxplot(GarageCars,  
main = "BoxPlot - GARAGECARS",  
ylab = "GarageCars",  
col = "#cc0000",  
border = "black",  
notch = TRUE  
)
```

## BoxPlot – GARAGECARS



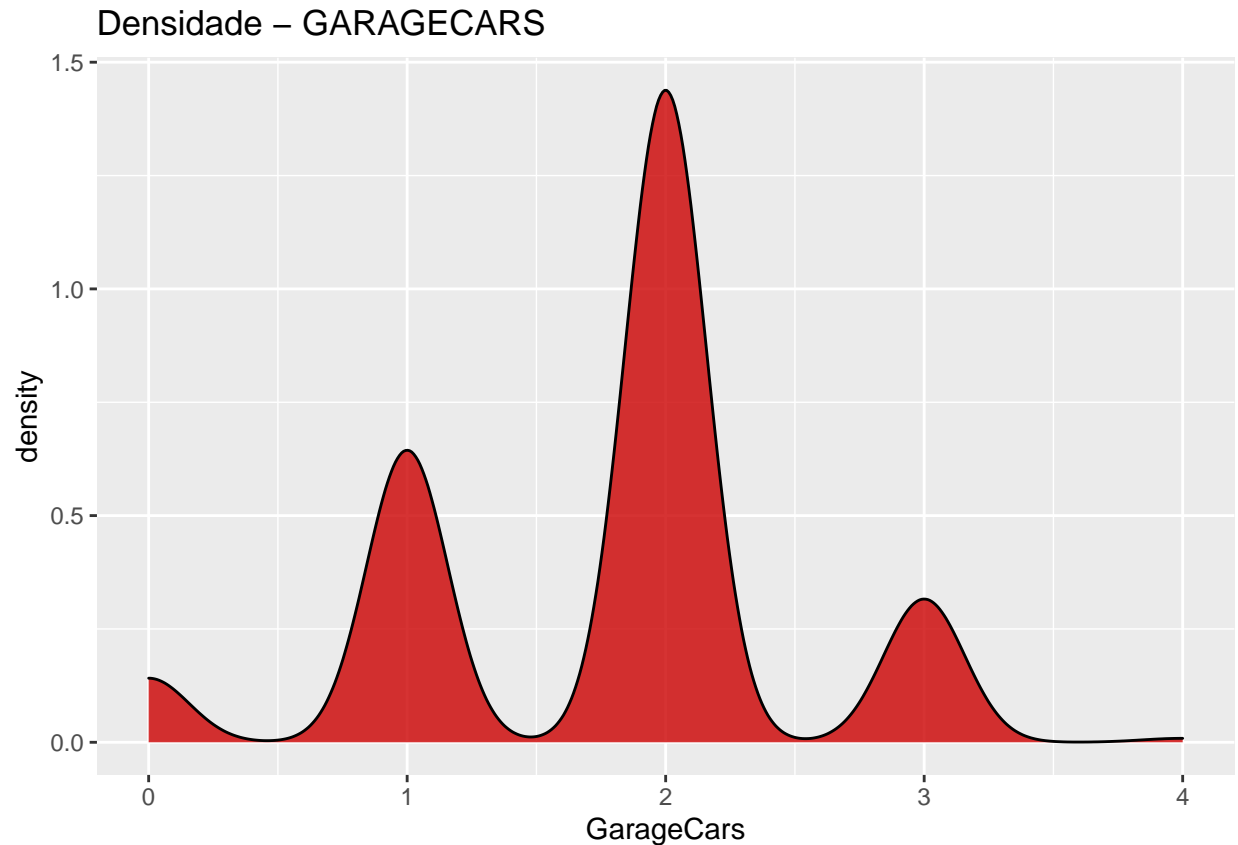
```
p <- data %>%  
  ggplot( aes(x=GarageCars)) +  
    geom_histogram( binwidth=1, fill="#cc0000", color="#000000", alpha=0.9) +  
    ggtitle('Histograma - GARAGECARS') +  
    theme(  
      plot.title = element_text(size=15)  
    )  
p
```

## Histograma – GARAGECARS



```
data %>%  
  ggplot( aes(x=GarageCars)) +  
    geom_density(fill="#cc0000", color="#000000", alpha=0.8) +  
    ggtitle("Densidade - GARAGECARS")
```





### SalePrice(Preço de venda)

A variável SalePrice segue um comportamento parecido com a variável GarageCars ao analisar os gráficos, visto que ela tem uma tendência fortemente para a esquerda, ou seja, há uma frequência maior para a venda de casas com preços menores.

```
cat("Média: ", mean(SalePrice), "\n")
```

```
## Média: 180921.2
```

```
cat("Mediana:", median(SalePrice), "\n")
```

```
## Mediana: 163000
```

```
cat("Intervalo:", max(SalePrice) - min(SalePrice), "\n")
```

```
## Intervalo: 720100
```

```
cat("Distancia Interquartilica:",IQR(SalePrice),'\n')
```

```
## Distancia Interquartilica: 84025
```

```
cat('Variância:',var(SalePrice),'\n')
```

```
## Variância: 6311111264
```

```
cat('Desvio Padrão:',sd(SalePrice),'\n')
```

```
## Desvio Padrão: 79442.5
```

```
cat('\nTabela de Frequência')
```

```
##
```

```
## Tabela de Frequência
```

```
table(SalePrice)
```

```
## SalePrice
```

```
## 34900 35311 37900 39300 40000 52000 52500 55000 55993 58500 60000
##      1      1      1      1      1      1      1      2      1      1      3
## 61000 62383 64500 66500 67000 68400 68500 72500 73000 75000 75500
##      1      1      1      1      2      1      1      1      1      1      1
## 76000 76500 78000 79000 79500 79900 80000 80500 81000 82000 82500
##      1      1      1      3      1      2      4      1      3      3      3
## 83000 83500 84000 84500 84900 85000 85400 85500 86000 87000 87500
##      2      1      1      3      1      4      1      1      3      4      1
## 88000 89000 89471 89500 90000 90350 91000 91300 91500 92000 92900
##      4      1      1      2      3      1      3      1      2      1      1
## 93000 93500 94000 94500 94750 95000 96500 97000 97500 98000 98300
##      3      2      1      1      1      2      2      3      1      3      1
## 98600 99500 99900 1e+05 101000 101800 102000 102776 103000 103200 103600
##      1      1      1      9      1      1      3      1      1      1      1
## 104000 104900 105000 105500 105900 106000 106250 106500 107000 107400 107500
##      1      2      5      1      1      3      1      2      3      1      3
## 107900 108000 108480 108500 108959 109000 109008 109500 109900 110000 110500
##      1      6      1      1      1      2      1      4      3      13      2
## 111000 111250 112000 112500 113000 114500 114504 115000 116000 116050 116500
##      1      1      7      2      6      2      1      12      3      1      1
## 116900 117000 117500 118000 118400 118500 118858 118964 119000 119200 119500
##      1      4      2      6      1      4      1      1      7      1      4
## 119750 119900 120000 120500 121000 121500 121600 122000 122500 122900 123000
##      1      2      7      4      1      1      1      4      2      1      4
## 123500 123600 124000 124500 124900 125000 125500 126000 126175 126500 127000
##      1      1      6      4      2      10      4      3      1      1      9
## 127500 128000 128200 128500 128900 128950 129000 129500 129900 130000 130250
##      6      7      1      4      1      1      8      4      4      11      1
## 130500 131000 131400 131500 132000 132250 132500 133000 133500 133700 133900
##      3      3      1      3      6      1      6      6      1      1      2
```

|    |        |        |        |        |        |        |        |        |        |        |        |
|----|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| ## | 134000 | 134432 | 134450 | 134500 | 134800 | 134900 | 135000 | 135500 | 135750 | 135900 | 135960 |
| ## | 3      | 1      | 1      | 2      | 1      | 2      | 17     | 2      | 1      | 1      | 1      |
| ## | 136000 | 136500 | 136900 | 136905 | 137000 | 137450 | 137500 | 137900 | 138000 | 138500 | 138800 |
| ## | 3      | 5      | 1      | 1      | 5      | 1      | 6      | 1      | 3      | 2      | 1      |
| ## | 138887 | 139000 | 139400 | 139500 | 139600 | 139900 | 139950 | 140000 | 140200 | 141000 | 141500 |
| ## | 1      | 11     | 2      | 1      | 1      | 1      | 1      | 20     | 1      | 8      | 1      |
| ## | 142000 | 142125 | 142500 | 142600 | 142953 | 143000 | 143250 | 143500 | 143750 | 143900 | 144000 |
| ## | 4      | 1      | 3      | 1      | 1      | 10     | 1      | 2      | 1      | 1      | 10     |
| ## | 144152 | 144500 | 144900 | 145000 | 145250 | 145500 | 145900 | 146000 | 146500 | 146800 | 147000 |
| ## | 1      | 2      | 1      | 14     | 1      | 1      | 1      | 3      | 1      | 1      | 9      |
| ## | 147400 | 147500 | 148000 | 148500 | 148800 | 149000 | 149300 | 149350 | 149500 | 149700 | 149900 |
| ## | 1      | 1      | 7      | 3      | 1      | 4      | 1      | 1      | 2      | 1      | 4      |
| ## | 150000 | 150500 | 150750 | 150900 | 151000 | 151400 | 151500 | 152000 | 153000 | 153337 | 153500 |
| ## | 4      | 1      | 1      | 1      | 5      | 1      | 1      | 6      | 3      | 1      | 3      |
| ## | 153575 | 153900 | 154000 | 154300 | 154500 | 154900 | 155000 | 155835 | 155900 | 156000 | 156500 |
| ## | 1      | 2      | 5      | 1      | 1      | 1      | 14     | 1      | 1      | 4      | 1      |
| ## | 156932 | 157000 | 157500 | 157900 | 158000 | 158500 | 158900 | 159000 | 159434 | 159500 | 159895 |
| ## | 1      | 6      | 2      | 2      | 6      | 1      | 1      | 4      | 1      | 3      | 1      |
| ## | 159950 | 160000 | 160200 | 161000 | 161500 | 161750 | 162000 | 162500 | 162900 | 163000 | 163500 |
| ## | 1      | 12     | 1      | 2      | 2      | 1      | 4      | 1      | 2      | 4      | 2      |
| ## | 163900 | 163990 | 164000 | 164500 | 164700 | 164900 | 164990 | 165000 | 165150 | 165400 | 165500 |
| ## | 1      | 1      | 3      | 2      | 1      | 1      | 1      | 8      | 1      | 1      | 3      |
| ## | 165600 | 166000 | 167000 | 167240 | 167500 | 167900 | 168000 | 168500 | 169000 | 169500 | 169900 |
| ## | 1      | 2      | 4      | 1      | 3      | 2      | 4      | 3      | 3      | 2      | 1      |
| ## | 169990 | 170000 | 171000 | 171500 | 171750 | 171900 | 172000 | 172400 | 172500 | 172785 | 173000 |
| ## | 1      | 8      | 5      | 1      | 1      | 1      | 1      | 1      | 5      | 1      | 7      |
| ## | 173500 | 173733 | 173900 | 174000 | 174500 | 174900 | 175000 | 175500 | 175900 | 176000 | 176432 |
| ## | 1      | 1      | 1      | 7      | 1      | 1      | 9      | 3      | 2      | 8      | 1      |
| ## | 176485 | 176500 | 177000 | 177500 | 178000 | 178400 | 178740 | 178900 | 179000 | 179200 | 179400 |
| ## | 1      | 2      | 5      | 3      | 7      | 1      | 1      | 1      | 3      | 2      | 1      |
| ## | 179500 | 179540 | 179600 | 179665 | 179900 | 180000 | 180500 | 181000 | 181134 | 181500 | 181900 |
| ## | 1      | 1      | 1      | 1      | 5      | 10     | 4      | 7      | 1      | 1      | 1      |
| ## | 182000 | 182900 | 183000 | 183200 | 183500 | 183900 | 184000 | 184100 | 184750 | 184900 | 185000 |
| ## | 1      | 1      | 1      | 1      | 1      | 1      | 4      | 1      | 1      | 1      | 10     |
| ## | 185500 | 185750 | 185850 | 185900 | 186000 | 186500 | 186700 | 187000 | 187100 | 187500 | 187750 |
| ## | 1      | 1      | 1      | 1      | 1      | 2      | 1      | 3      | 1      | 6      | 1      |
| ## | 188000 | 188500 | 188700 | 189000 | 189950 | 190000 | 191000 | 192000 | 192140 | 192500 | 193000 |
| ## | 3      | 1      | 1      | 6      | 1      | 13     | 4      | 5      | 1      | 2      | 3      |
| ## | 193500 | 193879 | 194000 | 194201 | 194500 | 194700 | 195000 | 195400 | 196000 | 196500 | 197000 |
| ## | 2      | 1      | 3      | 1      | 3      | 1      | 3      | 1      | 3      | 2      | 4      |
| ## | 197500 | 197900 | 198500 | 198900 | 199900 | 2e+05  | 200100 | 200141 | 200500 | 200624 | 201000 |
| ## | 2      | 3      | 1      | 1      | 2      | 8      | 1      | 1      | 2      | 1      | 3      |
| ## | 201800 | 202500 | 202665 | 202900 | 203000 | 204000 | 204750 | 204900 | 205000 | 205950 | 206000 |
| ## | 1      | 3      | 1      | 1      | 2      | 2      | 1      | 1      | 6      | 1      | 1      |
| ## | 206300 | 206900 | 207000 | 207500 | 208300 | 208500 | 208900 | 209500 | 210000 | 211000 | 212000 |
| ## | 1      | 1      | 2      | 5      | 1      | 1      | 2      | 1      | 5      | 2      | 3      |
| ## | 212900 | 213000 | 213250 | 213490 | 213500 | 214000 | 214500 | 214900 | 215000 | 215200 | 216000 |
| ## | 1      | 3      | 1      | 1      | 2      | 5      | 1      | 1      | 8      | 1      | 1      |
| ## | 216500 | 216837 | 217000 | 217500 | 218000 | 219210 | 219500 | 220000 | 221000 | 221500 | 222000 |
| ## | 1      | 1      | 2      | 1      | 1      | 1      | 3      | 5      | 2      | 1      | 3      |
| ## | 222500 | 223000 | 223500 | 224000 | 224500 | 224900 | 225000 | 226000 | 226700 | 227000 | 227680 |
| ## | 2      | 1      | 2      | 2      | 1      | 2      | 6      | 3      | 1      | 3      | 1      |
| ## | 227875 | 228000 | 228500 | 228950 | 229000 | 229456 | 230000 | 230500 | 231500 | 232000 | 232600 |
| ## | 1      | 2      | 2      | 1      | 1      | 1      | 8      | 1      | 2      | 3      | 1      |

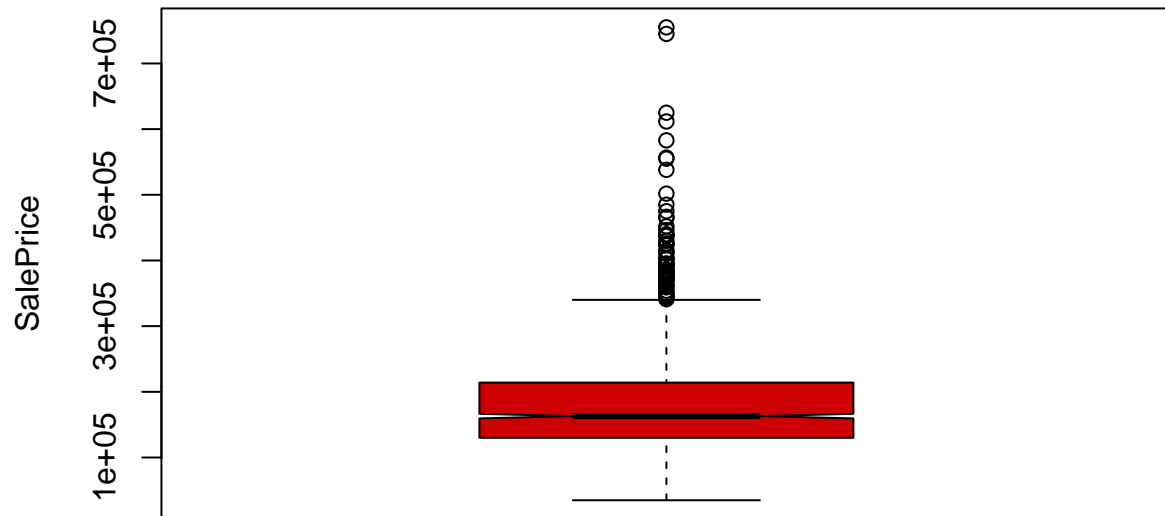
|    |        |        |        |        |        |        |        |        |        |        |        |
|----|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| ## | 233000 | 233170 | 233230 | 234000 | 235000 | 235128 | 236000 | 236500 | 237000 | 237500 | 238000 |
| ## | 1      | 1      | 1      | 2      | 7      | 1      | 2      | 2      | 2      | 1      | 1      |
| ## | 239000 | 239500 | 239686 | 239799 | 239900 | 240000 | 241000 | 241500 | 242000 | 243000 | 244000 |
| ## | 6      | 1      | 1      | 1      | 1      | 6      | 1      | 2      | 2      | 1      | 3      |
| ## | 244400 | 244600 | 245000 | 245350 | 245500 | 246578 | 248000 | 248328 | 248900 | 249700 | 250000 |
| ## | 1      | 1      | 1      | 1      | 1      | 1      | 2      | 1      | 1      | 1      | 8      |
| ## | 250580 | 251000 | 252000 | 252678 | 253000 | 253293 | 254000 | 254900 | 255000 | 255500 | 255900 |
| ## | 1      | 1      | 1      | 1      | 1      | 1      | 1      | 1      | 2      | 1      | 1      |
| ## | 256000 | 256300 | 257000 | 257500 | 258000 | 259000 | 259500 | 260000 | 260400 | 261500 | 262000 |
| ## | 2      | 1      | 1      | 1      | 1      | 1      | 1      | 6      | 1      | 1      | 1      |
| ## | 262280 | 262500 | 263000 | 263435 | 264132 | 264561 | 265000 | 265900 | 265979 | 266000 | 266500 |
| ## | 1      | 2      | 1      | 1      | 1      | 1      | 1      | 1      | 1      | 2      | 1      |
| ## | 267000 | 268000 | 269500 | 269790 | 270000 | 271000 | 271900 | 272000 | 274000 | 274300 | 274725 |
| ## | 1      | 2      | 1      | 1      | 3      | 2      | 1      | 2      | 1      | 1      | 1      |
| ## | 274900 | 274970 | 275000 | 275500 | 276000 | 277000 | 277500 | 278000 | 279500 | 280000 | 281000 |
| ## | 1      | 1      | 5      | 1      | 1      | 2      | 1      | 2      | 1      | 4      | 1      |
| ## | 281213 | 282922 | 283463 | 284000 | 285000 | 286000 | 287000 | 287090 | 289000 | 290000 | 293077 |
| ## | 1      | 1      | 1      | 2      | 3      | 1      | 2      | 1      | 1      | 5      | 1      |
| ## | 294000 | 295000 | 295493 | 297000 | 299800 | 301000 | 301500 | 302000 | 303477 | 305000 | 305900 |
| ## | 1      | 1      | 1      | 1      | 1      | 1      | 1      | 2      | 1      | 1      | 1      |
| ## | 306000 | 307000 | 309000 | 310000 | 311500 | 311872 | 312500 | 313000 | 314813 | 315000 | 315500 |
| ## | 1      | 1      | 1      | 2      | 1      | 1      | 1      | 1      | 1      | 4      | 1      |
| ## | 315750 | 316600 | 317000 | 318000 | 318061 | 319000 | 319900 | 320000 | 324000 | 325000 | 325300 |
| ## | 1      | 1      | 1      | 2      | 1      | 1      | 1      | 4      | 1      | 4      | 1      |
| ## | 325624 | 326000 | 328000 | 328900 | 333168 | 335000 | 336000 | 337000 | 337500 | 339750 | 340000 |
| ## | 1      | 1      | 1      | 1      | 1      | 3      | 1      | 1      | 1      | 1      | 2      |
| ## | 341000 | 342643 | 345000 | 348000 | 350000 | 354000 | 359100 | 360000 | 361919 | 367294 | 369900 |
| ## | 1      | 1      | 2      | 1      | 2      | 1      | 1      | 1      | 1      | 1      | 1      |
| ## | 370878 | 372402 | 372500 | 374000 | 375000 | 377426 | 377500 | 378500 | 380000 | 381000 | 383970 |
| ## | 1      | 1      | 1      | 1      | 1      | 1      | 1      | 1      | 1      | 1      | 1      |
| ## | 385000 | 386250 | 392000 | 392500 | 394432 | 394617 | 395000 | 395192 | 402000 | 402861 | 403000 |
| ## | 2      | 1      | 1      | 1      | 1      | 1      | 1      | 1      | 1      | 1      | 1      |
| ## | 410000 | 412500 | 415298 | 423000 | 424870 | 426000 | 430000 | 437154 | 438780 | 440000 | 446261 |
| ## | 1      | 1      | 1      | 1      | 1      | 1      | 1      | 1      | 1      | 1      | 1      |
| ## | 451950 | 465000 | 466500 | 475000 | 485000 | 501837 | 538000 | 555000 | 556581 | 582933 | 611657 |
| ## | 1      | 1      | 1      | 1      | 1      | 1      | 1      | 1      | 1      | 1      | 1      |
| ## | 625000 | 745000 | 755000 |        |        |        |        |        |        |        |        |
| ## | 1      | 1      | 1      |        |        |        |        |        |        |        |        |

```

boxplot(SalePrice,
main = "BoxPlot - SALEPRICE",
ylab = "SalePrice",
col = "#cc0000",
border = "black",
notch = TRUE
)

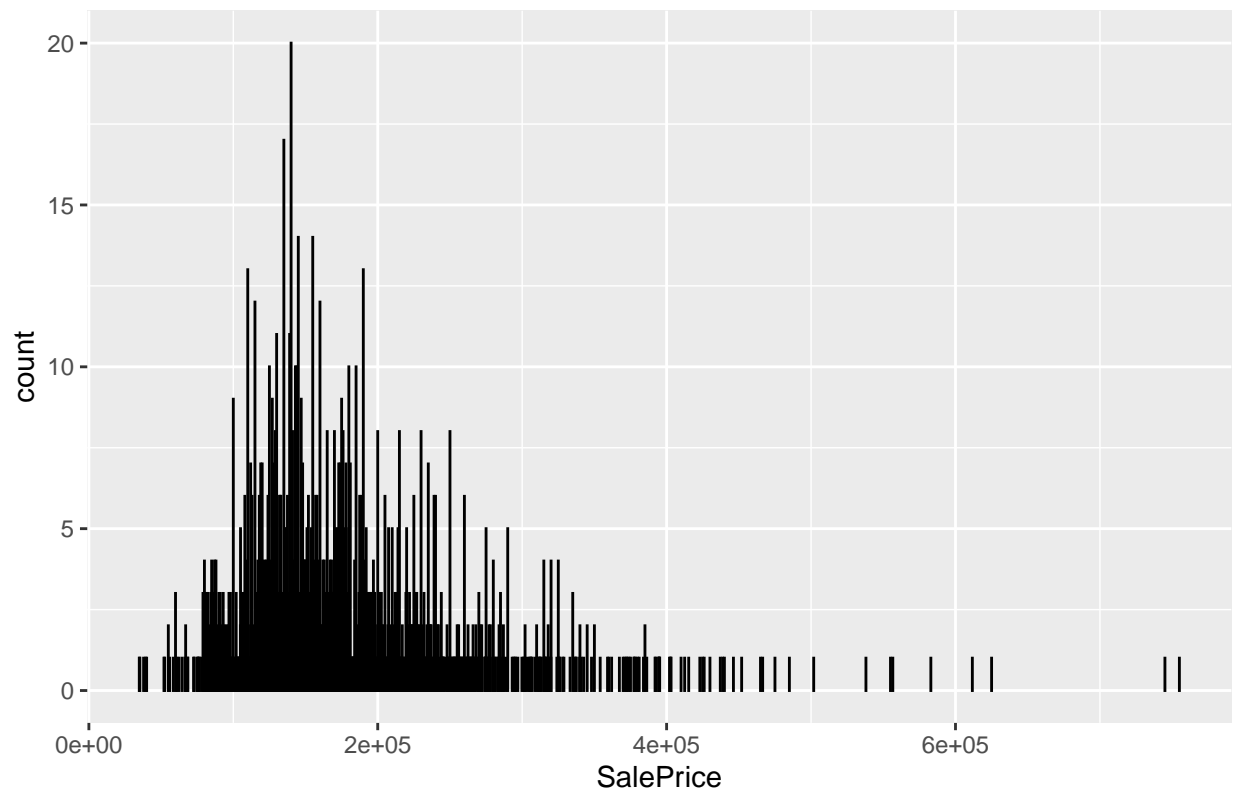
```

## BoxPlot – SALEPRICE

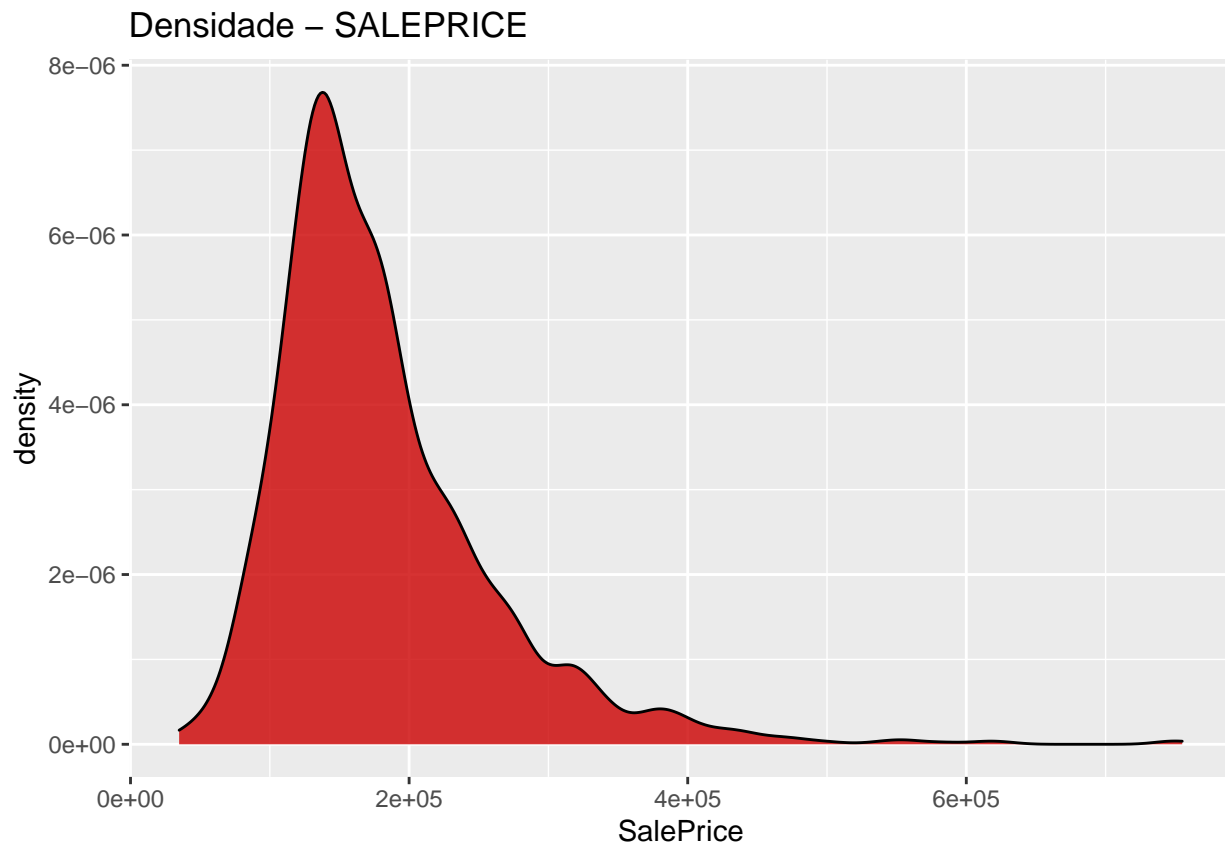


```
p <- data %>%  
  ggplot( aes(x=SalePrice)) +  
    geom_histogram( binwidth=1, fill="#cc0000", color="#000000", alpha=0.9) +  
    ggtitle('Histograma - SALEPRICE') +  
    theme(  
      plot.title = element_text(size=15)  
    )  
p
```

## Histograma – SALEPRICE



```
data %>%  
  ggplot( aes(x=SalePrice)) +  
    geom_density(fill="#cc0000", color="#000000", alpha=0.8) +  
    ggtitle("Densidade - SALEPRICE")
```



As análises bivariadas foram feitas entre a variável Saleprice e as demais variáveis. Calculamos a correlação e o gráfico de dispersão. As correlações mostram a relação entre duas variáveis se forem positivas possuem um comportamento crescente, se forem negativas possuem um comportamento decrescente e se for próximo de zero possuem um comportamento constante. Os gráficos de dispersão relacionam as duas variáveis em um plano cartesiano (demais variáveis, SalePrice).

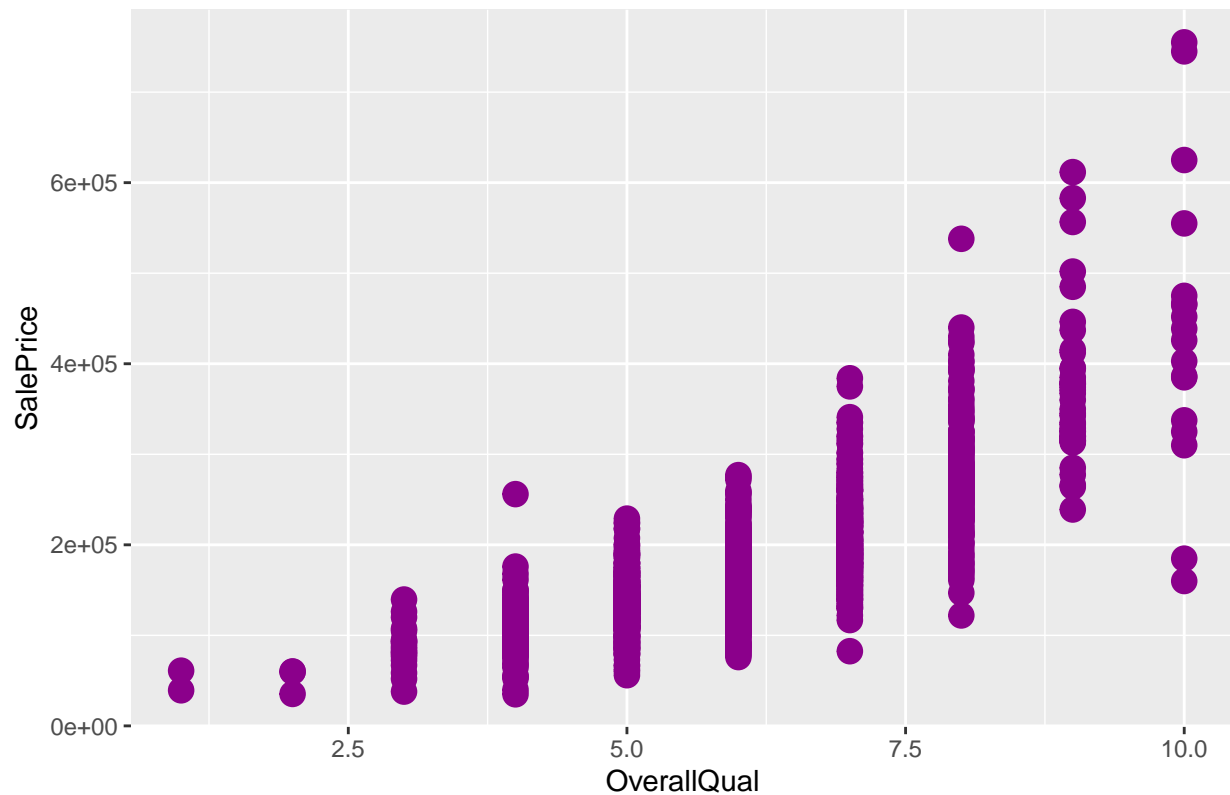
A Correlação entre OverallQual e SalePrice é positiva e próxima de 1, isto quer dizer que possui um comportamento crescente forte. Prova disso visualmente, é o gráfico de dispersão entre as duas variáveis que é crescente. Ademais, isso quer dizer que quanto melhor o índice de qualidade do imóvel maior é seu valor de venda.

```
cat("Correlação OverallQual: ", cor(data$OverallQual,data$SalePrice),"\n")
```

```
## Correlação OverallQual: 0.7909816
```

```
ggplot(data, aes(x= OverallQual, y=SalePrice)) +  
  ggtitle("Gráfico de Dispersão - OverallQual") +  
  geom_point(size=4, color="#8b008b")
```

Grafico de Dispersão – OverallQual



A Correlação entre OverallCond e SalePrice é -0.077 quer dizer que é próxima de zero, ou seja, o comportamento é constante. Prova disso visualmente, é o gráfico de dispersão entre as duas variáveis é relativamente constante, porém há um leve decremento a partir do meio do gráfico (por isso é negativo). Ademais, isso mostra que o Índice da condição de estado do imóvel é relativamente constante ao valor de venda.

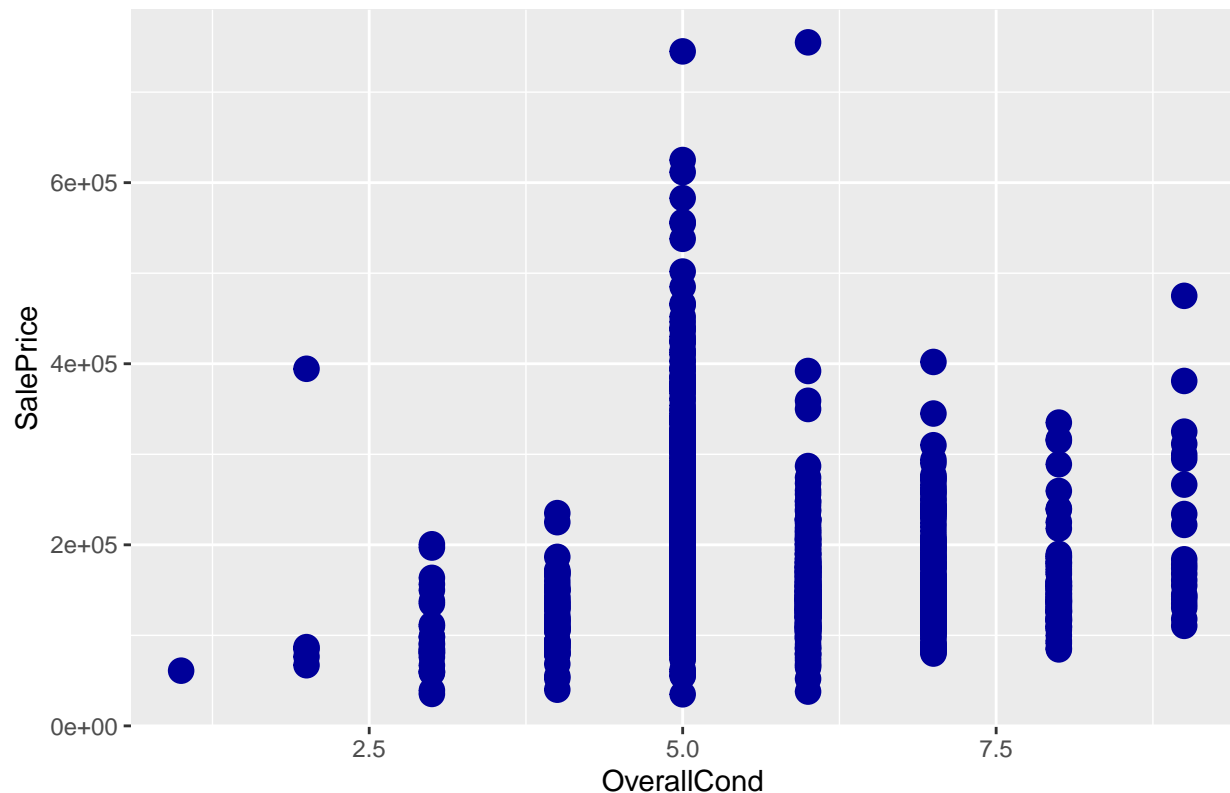
```
cat("Correlação OverallCond: ", cor(data$OverallCond,data$SalePrice),"\n")
```

```
## Correlação OverallCond: -0.07785589
```

```
ggplot(data, aes(x= OverallCond, y=SalePrice)) +  
  ggtitle("Grafico de Dispersão - OverallCond") +  
  geom_point(size=4, color="#000099")
```



Grafico de Dispersão – OverallCond



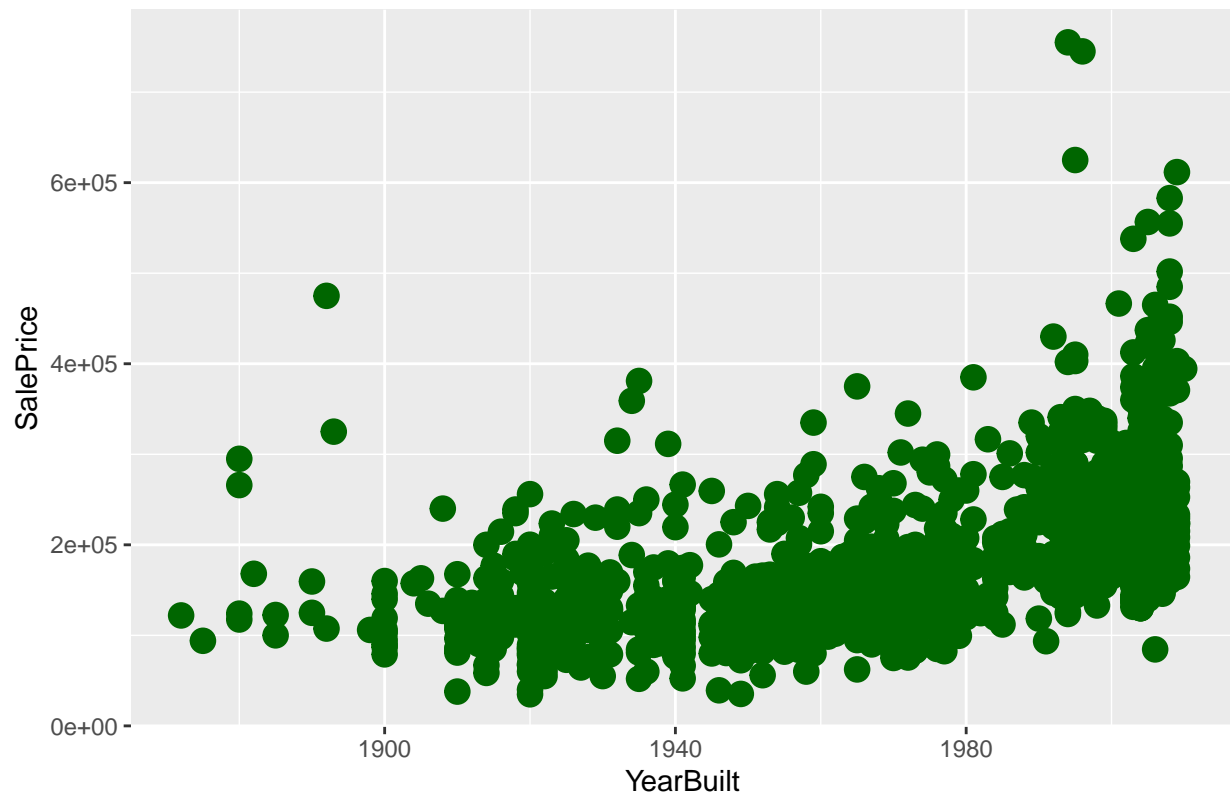
A Correlação entre YearBuilt e SalePrice é de 0.523, isto quer dizer que é crescente e mais próxima de 1, ou seja, é um leve comportamento crescente. Prova disso visualmente, é o gráfico de dispersão entre as duas variáveis que é levemente crescente. Ademais, isso quer dizer que quanto mais nova a casa maior seu valor, mas é uma proporção relativamente pequena.

```
cat("Correlação YearBuilt: ", cor(data$YearBuilt,data$SalePrice),"\n")
```

```
## Correlação YearBuilt: 0.5228973
```

```
ggplot(data, aes(x= YearBuilt, y=SalePrice)) +  
  ggtitle("Grafico de Dispersão - YearBuilt") +  
  geom_point(size=4, color="#006600")
```

Grafico de Dispersão – YearBuilt



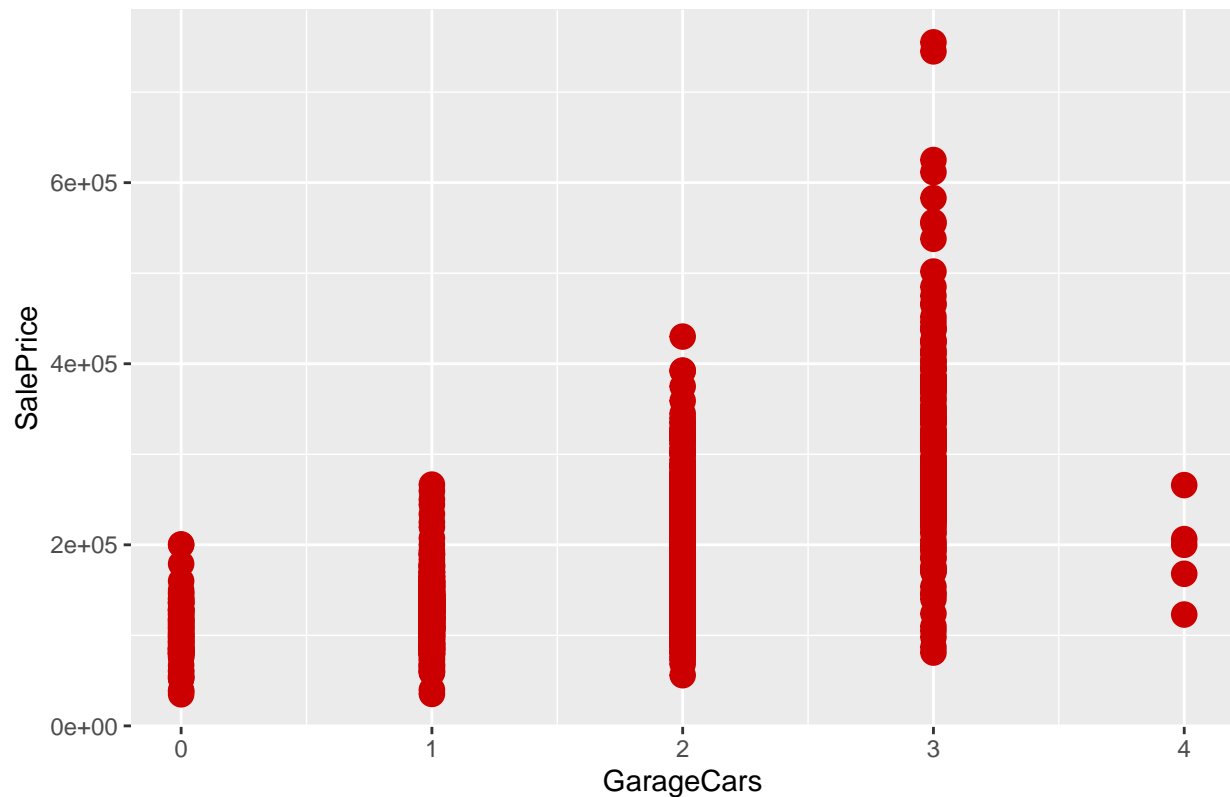
A Correlação entre GarageCars e SalePrice é 0.64, isto quer dizer que a relação possui um comportamento crescente. Prova disso visualmente, é o gráfico de dispersão entre as duas variáveis que é crescente. Ademais, isso quer dizer que quanto mais carro que couber na garage maior é o valor do imóvel.

```
cat("Correlação GarageCars: ", cor(data$GarageCars,data$SalePrice),"\n")
```

```
## Correlação GarageCars: 0.6404092
```

```
ggplot(data, aes(x= GarageCars, y=SalePrice)) +  
  ggtitle("Grafico de Dispersão - GarageCars") +  
  geom_point(size=4, color="#cc0000")
```

Grafico de Dispersão – GarageCars

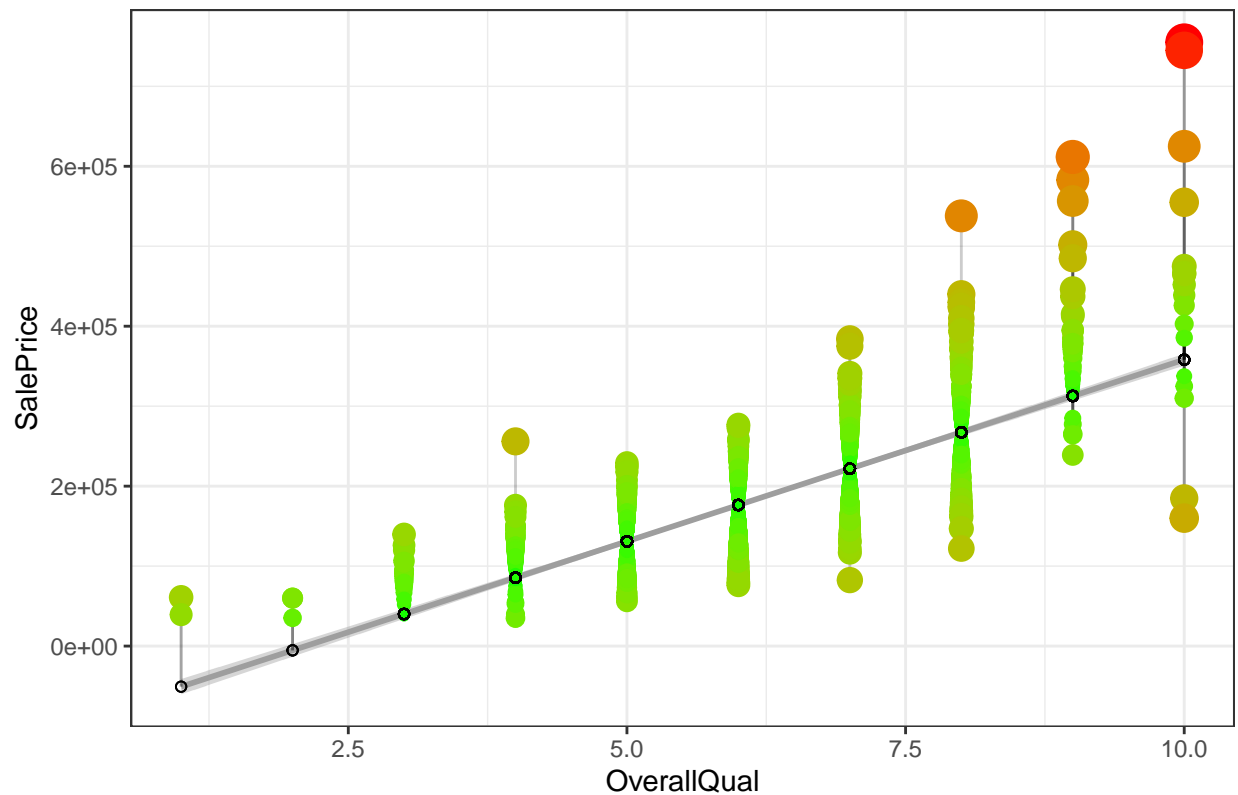


REGRESSÕES LINEARES com as análises de resíduos.

Regressão Linear para OverallQual sem ajustes.

```
fit1 <- lm(SalePrice ~ OverallQual, data = data)
data$predicted <- predict(fit1)
data$residuals <- residuals(fit1)
ggplot(data, aes(x = OverallQual, y = SalePrice)) +
  geom_smooth(formula = y ~ x, method = "lm", se = TRUE, color = "8b008b") +
  geom_segment(aes(xend = OverallQual, yend = predicted), alpha = .2) +
  geom_point(aes(color = abs(residuals), size = abs(residuals))) +
  scale_color_continuous(low = "green", high = "red") +
  guides(color = FALSE, size = FALSE) +
  geom_point(aes(y = predicted), shape = 1) +
  ggtitle("Regressão Linear sem ajustes - OverallQual")+
  theme_bw()
```

## Regressão Linear sem ajustes – OverallQual

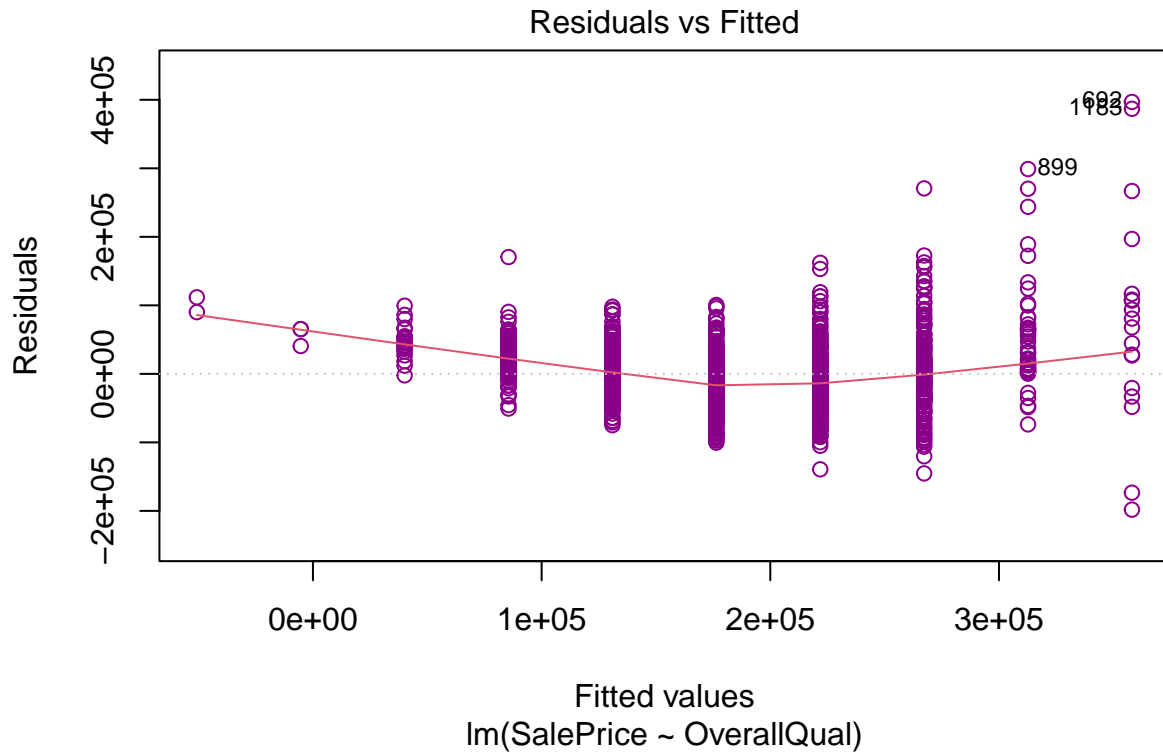


Ao analisar a normalidade dos resíduos com o `shapiro.test()` verificamos que o p-value é muito próximo de zero o que quer dizer que eles não seguem uma normalidade e que os intervalos de confiança estão comprometidos.

```
shapiro.test(residuals(fit1))
```

```
##  
##  Shapiro-Wilk normality test  
##  
## data:  residuals(fit1)  
## W = 0.90582, p-value < 2.2e-16
```

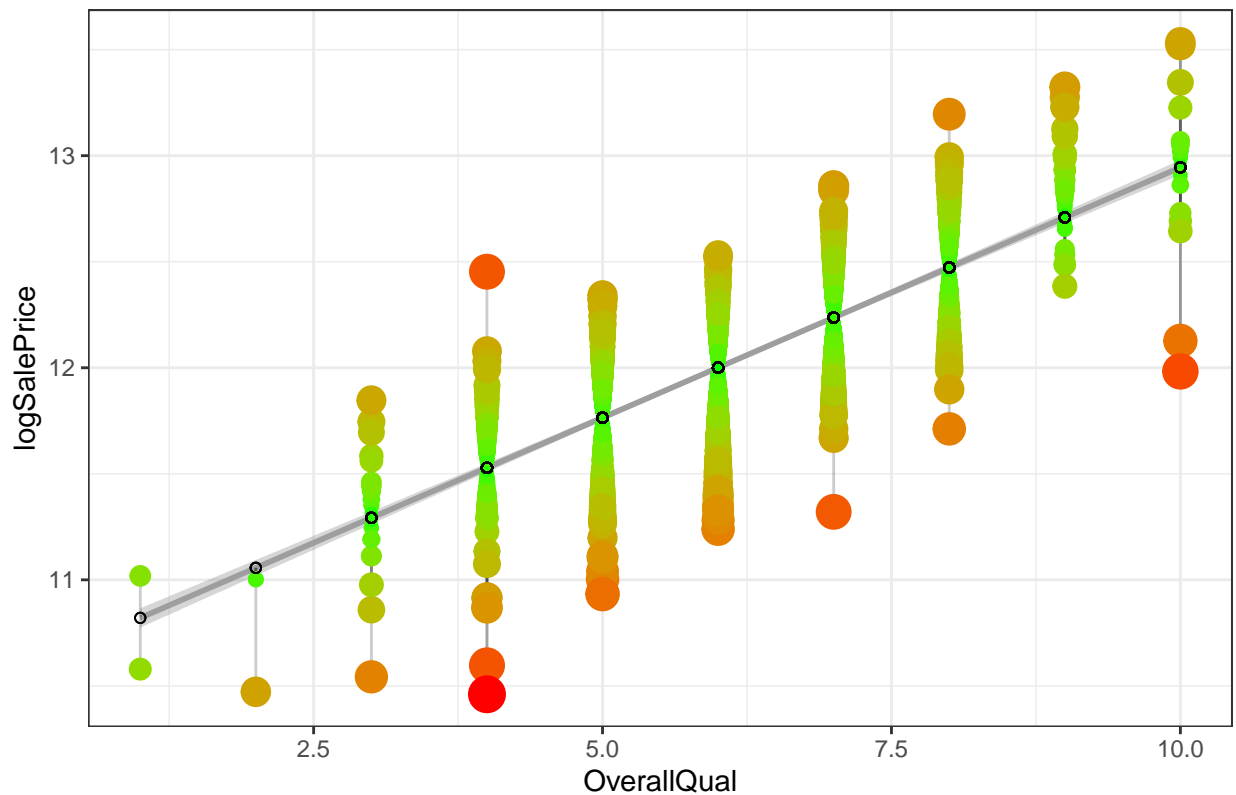
```
plot(fit1, which = 1, col=c("#8b008b"))
```



Para tentar chegar em uma normalidade aplicamos um log na variável SalePrice.

```
logSalePrice <- log(data$SalePrice)
fit <- lm(logSalePrice ~ OverallQual, data = data)
data$predicted <- predict(fit)
data$residuals <- residuals(fit)
ggplot(data, aes(x = OverallQual, y = logSalePrice)) +
  geom_smooth(formula = y ~ x, method = "lm", se = TRUE, color = "8b008b") +
  geom_segment(aes(xend = OverallQual, yend = predicted), alpha = .2) +
  geom_point(aes(color = abs(residuals), size = abs(residuals))) +
  scale_color_continuous(low = "green", high = "red") +
  guides(color = FALSE, size = FALSE) +
  geom_point(aes(y = predicted), shape = 1) +
  ggtitle("Regressão Lienar com ajustes - OverallQual")+
  theme_bw()
```

## Regressão Linear com ajustes – OverallQual

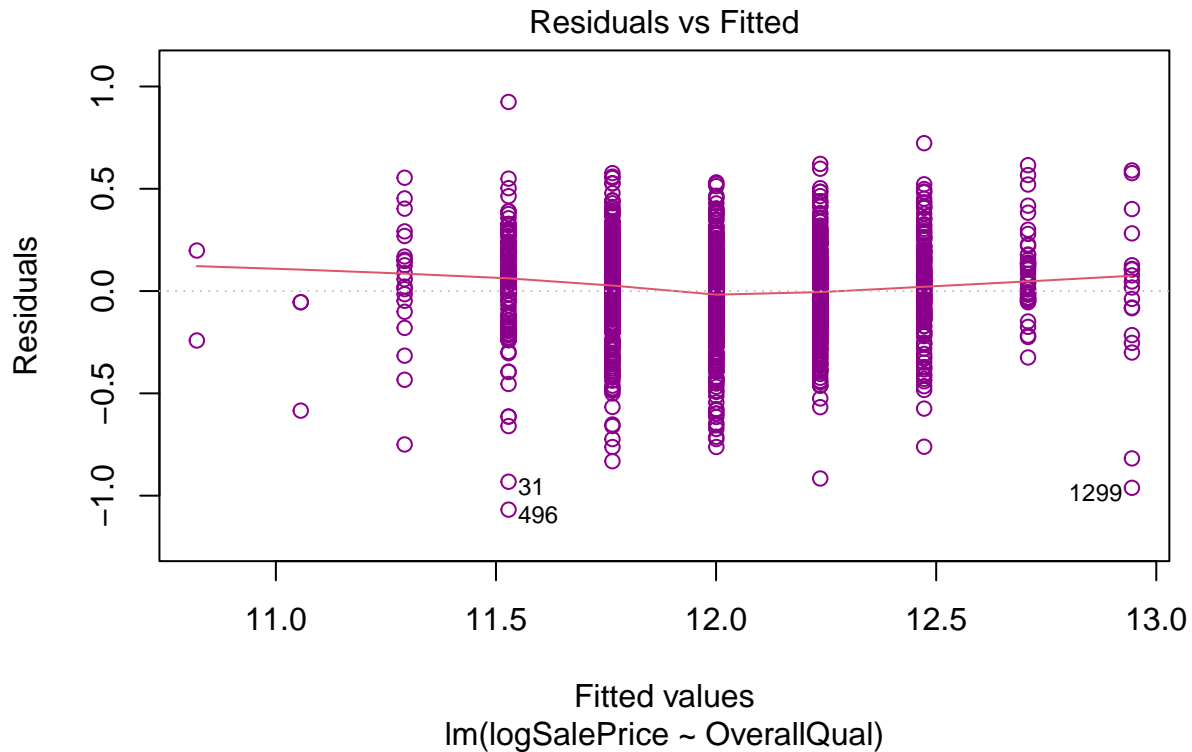


Verificamos o p-value. Contudo, ainda encontramos um valor muito próximo de zero que faz com que não cheguemos em uma normalidade.

```
shapiro.test(residuals(fit))
```

```
##  
##  Shapiro-Wilk normality test  
##  
## data:  residuals(fit)  
## W = 0.98216, p-value = 1.811e-12
```

```
plot(fit,which = 1,col=c("#8b008b"))
```

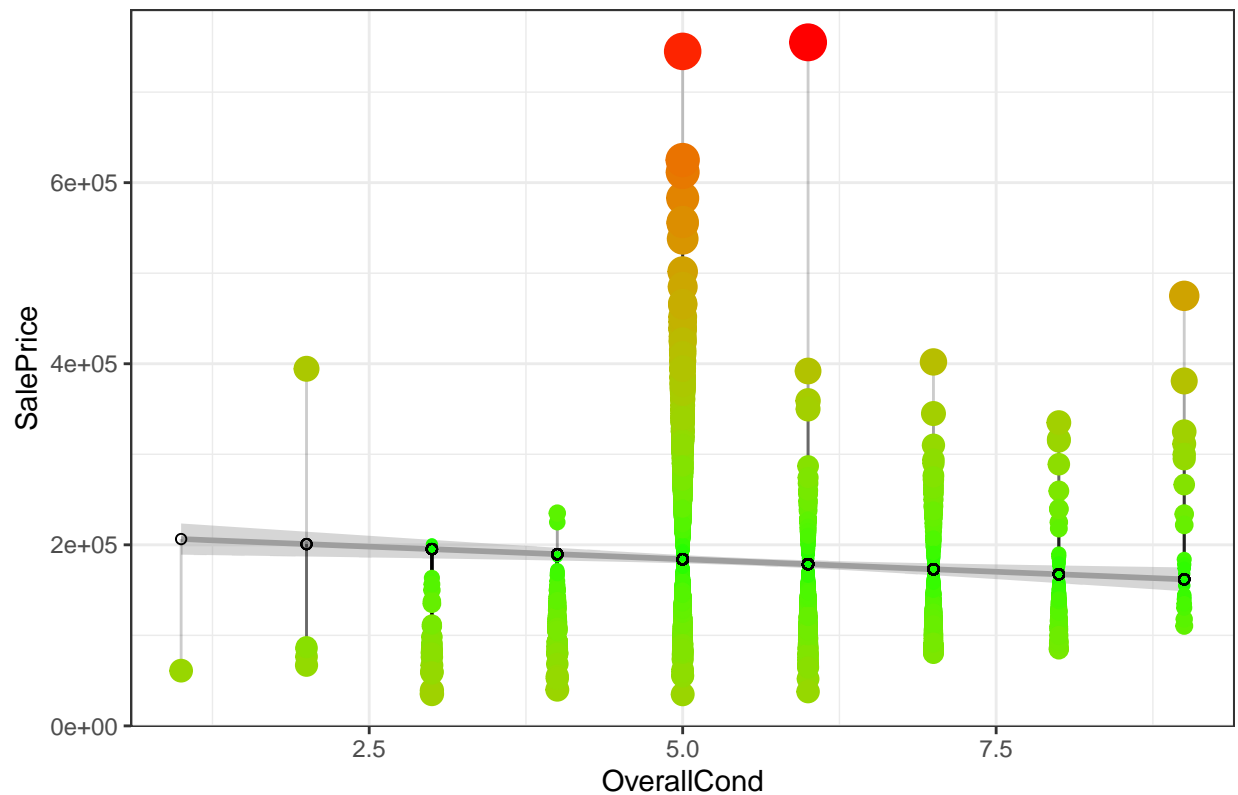


Fizemos o mesmo processo da variável OverallQual nas demais variáveis para tentar atingir a normalidade.

Regressão Linear para OverallCond sem ajustes.

```
fit2 <- lm(SalePrice ~ OverallCond, data = data)
data$predicted <- predict(fit2)
data$residuals <- residuals(fit2)
ggplot(data, aes(x = OverallCond, y = SalePrice)) +
  geom_smooth(formula = y ~ x, method = "lm", se = TRUE, color = "8b008b") +
  geom_segment(aes(xend = OverallCond, yend = predicted), alpha = .2) +
  geom_point(aes(color = abs(residuals), size = abs(residuals))) +
  scale_color_continuous(low = "green", high = "red") +
  guides(color = FALSE, size = FALSE) +
  geom_point(aes(y = predicted), shape = 1) +
  ggtitle("Regressão Linear sem ajustes - OverallCond") +
  theme_bw()
```

### Regressão Linear sem ajustes – OverallCond



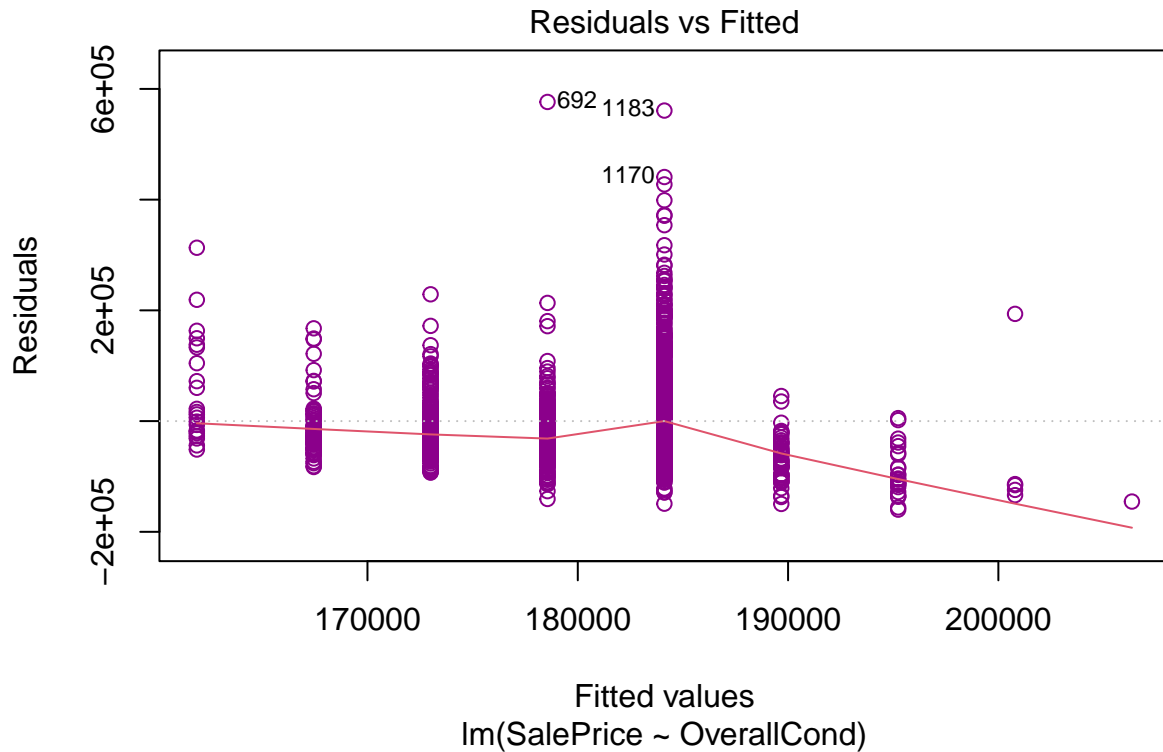
Ao analisar a normalidade dos resíduos com o `shapiro.test()` verificamos que o p-value é muito próximo de zero o que quer dizer que eles não seguem uma normalidade e que os intervalos de confiança estão comprometidos.

```
shapiro.test(residuals(fit2))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(fit2)
## W = 0.87642, p-value < 2.2e-16
```

```
plot(fit2,which = 1,col=c("#8b008b"))
```

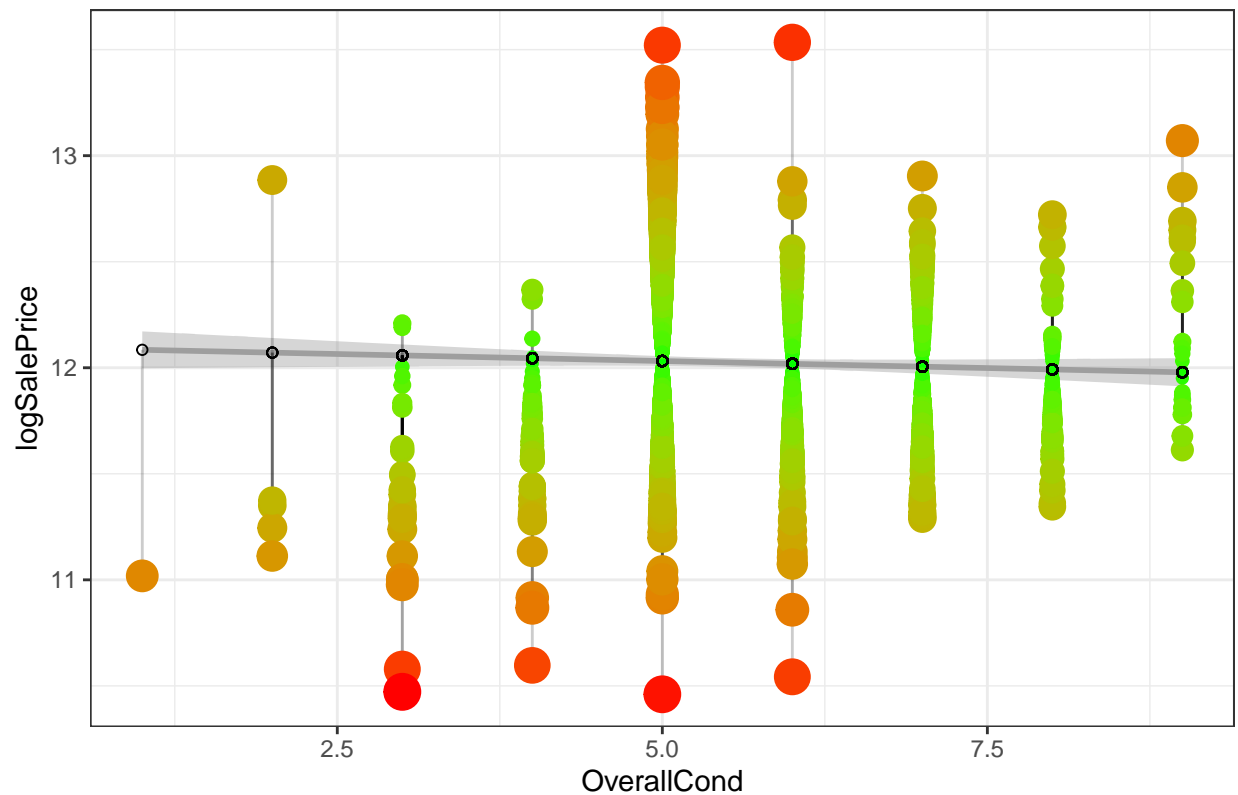




Para tentar chegar em uma normalidade aplicamos um log na variável SalePrice.

```
logSalePrice <- log(data$SalePrice)
fit <- lm(logSalePrice ~ OverallCond, data = data)
data$predicted <- predict(fit)
data$residuals <- residuals(fit)
ggplot(data, aes(x = OverallCond, y = logSalePrice)) +
  geom_smooth(formula = y ~ x, method = "lm", se = TRUE, color = "8b008b") +
  geom_segment(aes(xend = OverallCond, yend = predicted), alpha = .2) +
  geom_point(aes(color = abs(residuals), size = abs(residuals))) +
  scale_color_continuous(low = "green", high = "red") +
  guides(color = FALSE, size = FALSE) +
  geom_point(aes(y = predicted), shape = 1) +
  ggtitle("Regressão Lienar com ajustes - OverallCond")+
  theme_bw()
```

### Regressão Linear com ajustes – OverallCond

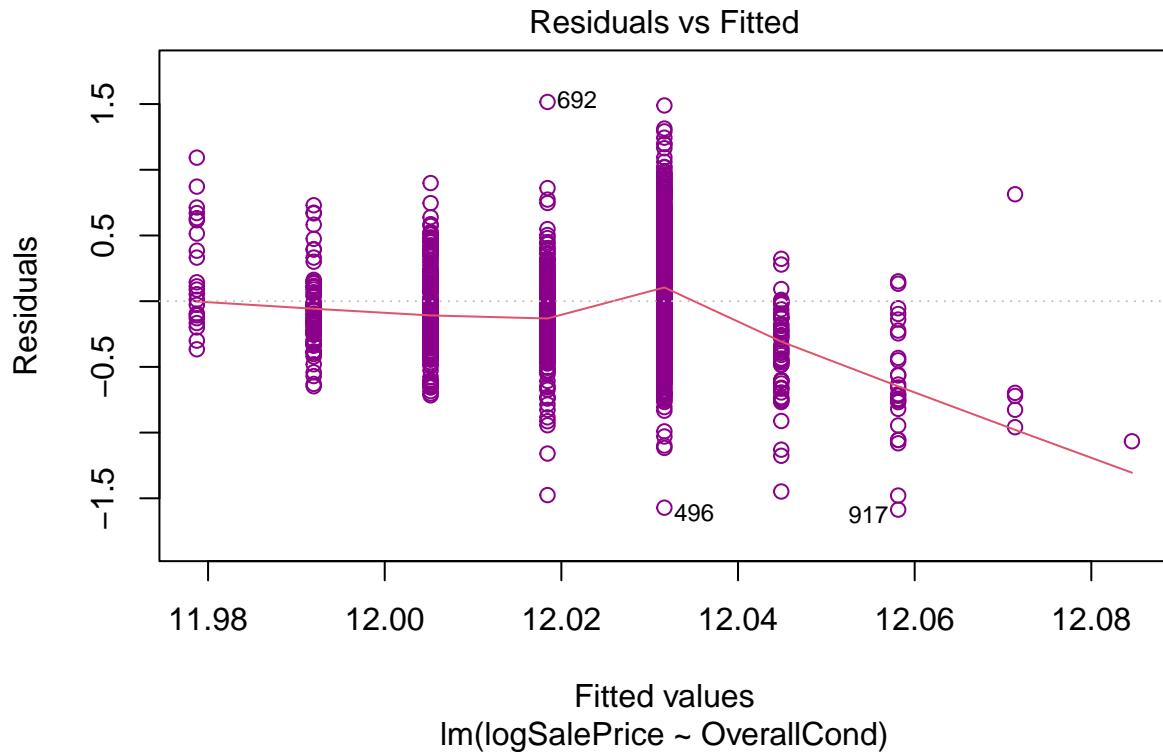


Verificamos o p-value. Contudo, ainda encontramos um valor muito próximo de zero que faz com que não cheguemos em uma normalidade.

```
shapiro.test(residuals(fit))
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: residuals(fit)  
## W = 0.99083, p-value = 6.503e-08
```

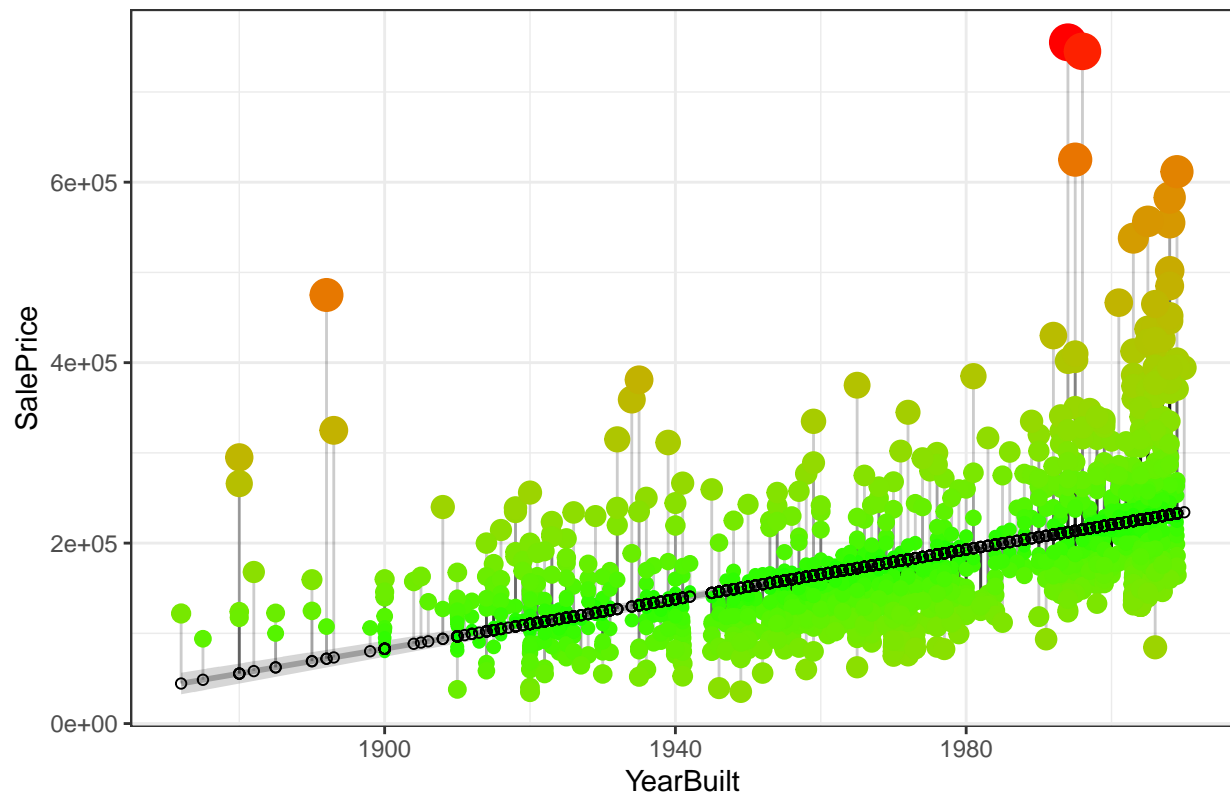
```
plot(fit, which = 1, col=c("#8b008b"))
```



Regressão Linear para YearBuilt sem ajustes.

```
fit3 <- lm(SalePrice ~ YearBuilt, data = data)
data$predicted <- predict(fit3)
data$residuals <- residuals(fit3)
ggplot(data, aes(x = YearBuilt, y = SalePrice)) +
  geom_smooth(formula = y ~ x, method = "lm", se = TRUE, color = "8b008b") +
  geom_segment(aes(xend = YearBuilt, yend = predicted), alpha = .2) +
  geom_point(aes(color = abs(residuals), size = abs(residuals))) +
  scale_color_continuous(low = "green", high = "red") +
  guides(color = FALSE, size = FALSE) +
  geom_point(aes(y = predicted), shape = 1) +
  ggtitle("Regressão Linear sem ajustes - YearBuilt")+
  theme_bw()
```

## Regressão Linear sem ajustes – YearBuilt

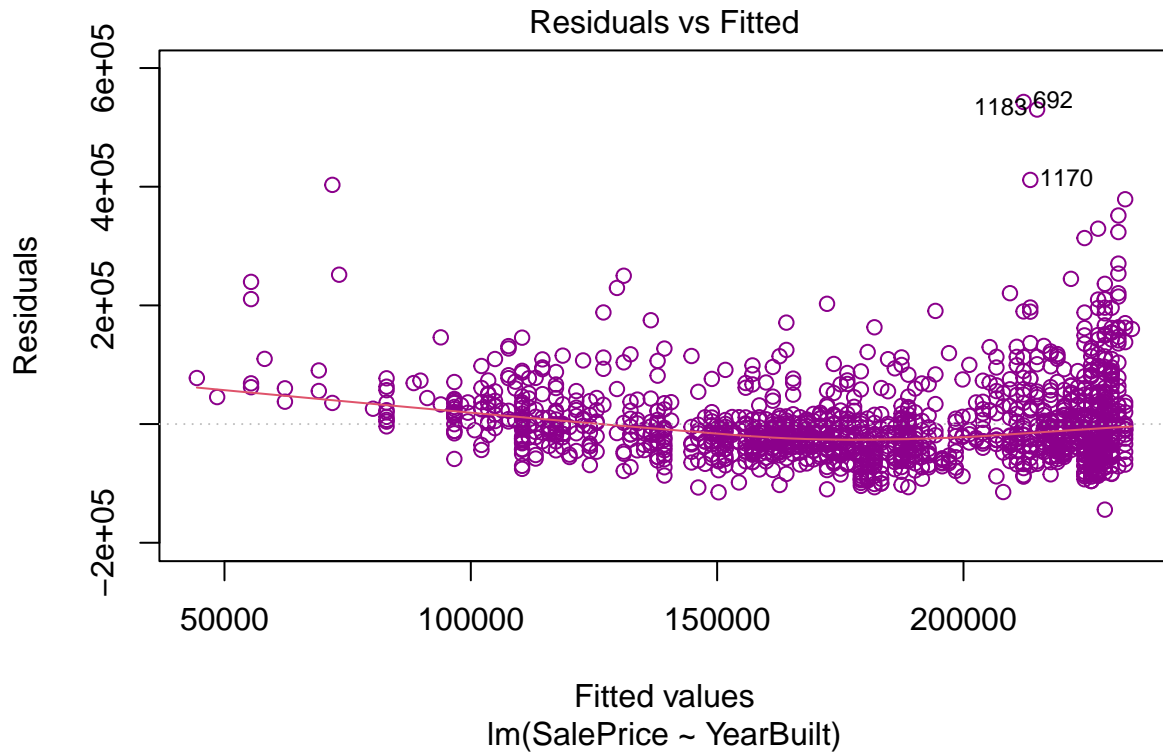


Ao analisar a normalidade dos resíduos com o `shapiro.test()` verificamos que o p-value é muito próximo de zero o que quer dizer que eles não seguem uma normalidade e que os intervalos de confiança estão comprometidos.

```
shapiro.test(residuals(fit3))
```

```
##  
##  Shapiro-Wilk normality test  
##  
## data:  residuals(fit3)  
## W = 0.83785, p-value < 2.2e-16
```

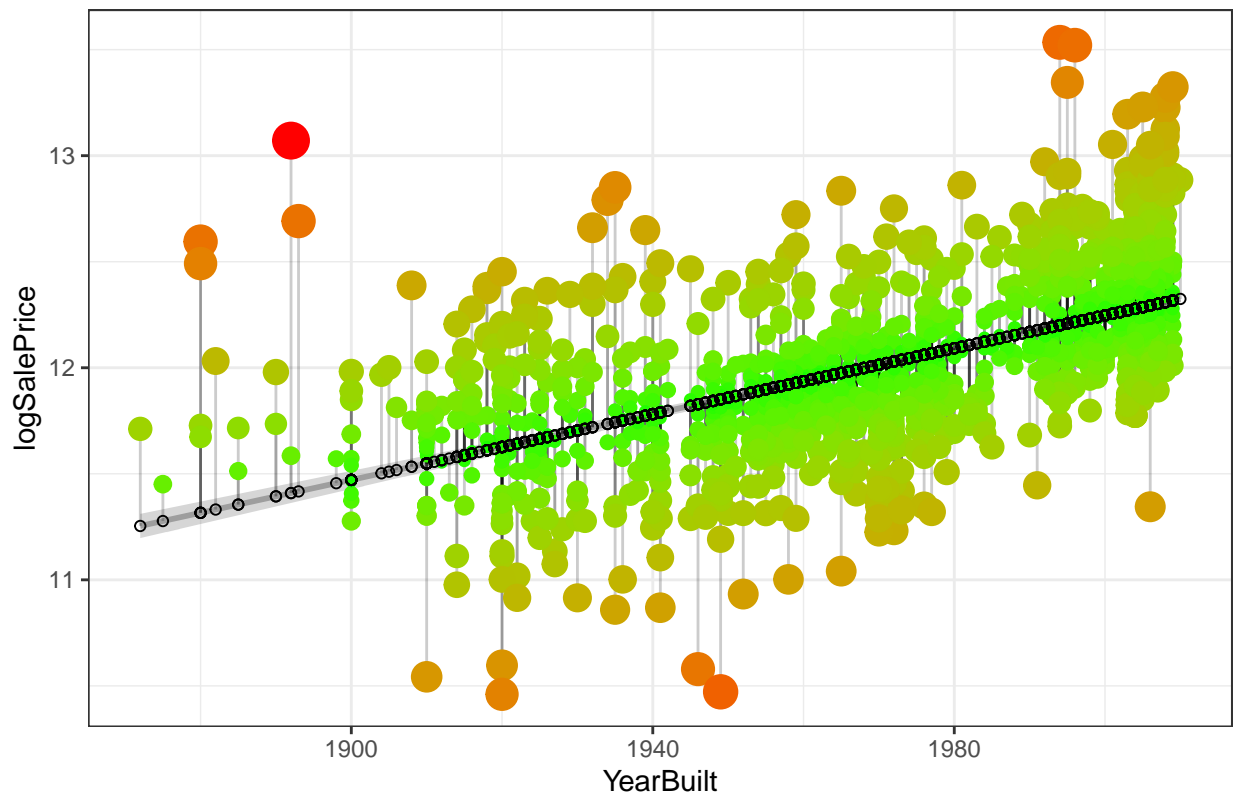
```
plot(fit3,which = 1,col=c("#8b008b"))
```



Para tentar chegar em uma normalidade aplicamos um log na variável SalePrice.

```
logSalePrice <- log(data$SalePrice)
fit <- lm(logSalePrice ~ YearBuilt, data = data)
data$predicted <- predict(fit)
data$residuals <- residuals(fit)
ggplot(data, aes(x = YearBuilt, y = logSalePrice)) +
  geom_smooth(formula = y ~ x, method = "lm", se = TRUE, color = "8b008b") +
  geom_segment(aes(xend = YearBuilt, yend = predicted), alpha = .2) +
  geom_point(aes(color = abs(residuals), size = abs(residuals))) +
  scale_color_continuous(low = "green", high = "red") +
  guides(color = FALSE, size = FALSE) +
  geom_point(aes(y = predicted), shape = 1) +
  ggtitle("Regressão Lienar com ajustes - YearBuilt")+
  theme_bw()
```

## Regressão Linear com ajustes – YearBuilt

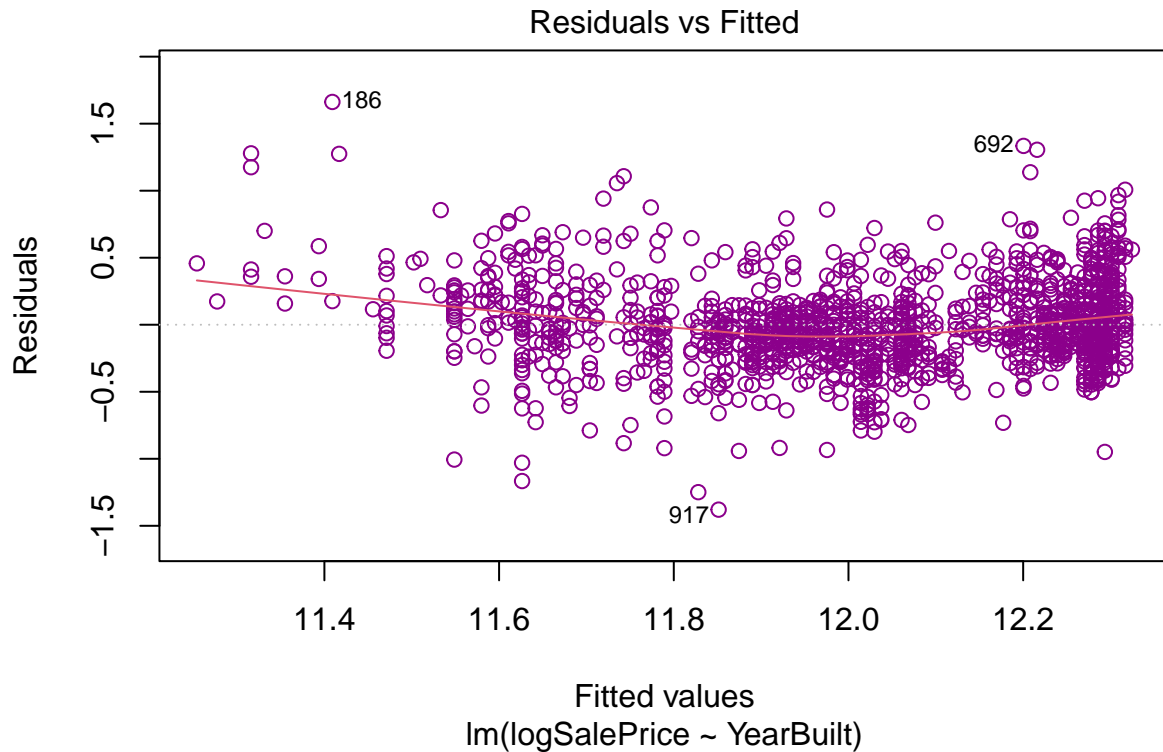


Verificamos o p-value. Contudo, ainda encontramos um valor muito próximo de zero que faz com que não cheguemos em uma normalidade.

```
shapiro.test(residuals(fit))
```

```
##  
##  Shapiro-Wilk normality test  
##  
## data:  residuals(fit)  
## W = 0.97436, p-value = 1.862e-15
```

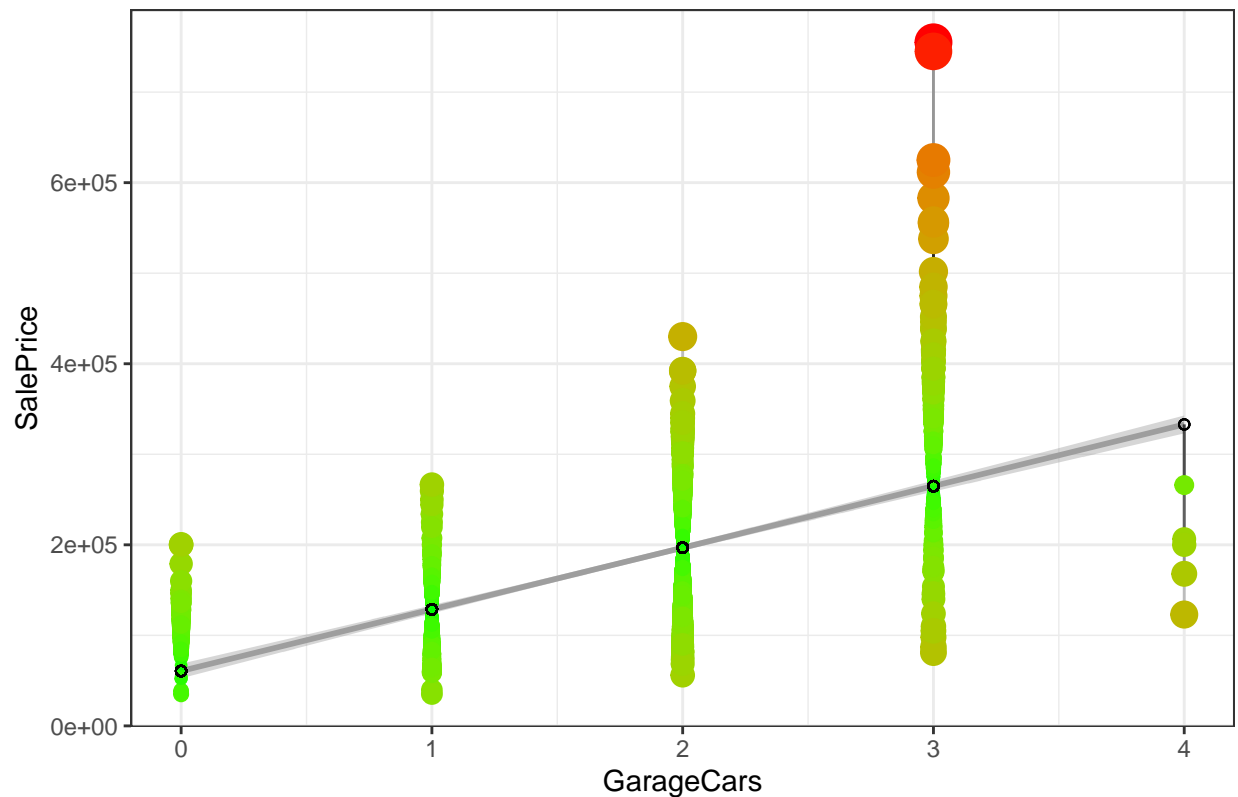
```
plot(fit,which = 1,col=c("#8b008b"))
```



Regressão Linear para GarageCars sem ajustes.

```
fit4 <- lm(SalePrice ~ GarageCars, data = data)
data$predicted <- predict(fit4)
data$residuals <- residuals(fit4)
ggplot(data, aes(x = GarageCars, y = SalePrice)) +
  geom_smooth(formula = y ~ x, method = "lm", se = TRUE, color = "8b008b") +
  geom_segment(aes(xend = GarageCars, yend = predicted), alpha = .2) +
  geom_point(aes(color = abs(residuals), size = abs(residuals))) +
  scale_color_continuous(low = "green", high = "red") +
  guides(color = FALSE, size = FALSE) +
  geom_point(aes(y = predicted), shape = 1) +
  ggtitle("Regressão Linear sem ajustes - GarageCars")+
  theme_bw()
```

## Regressão Linear sem ajustes – GarageCars



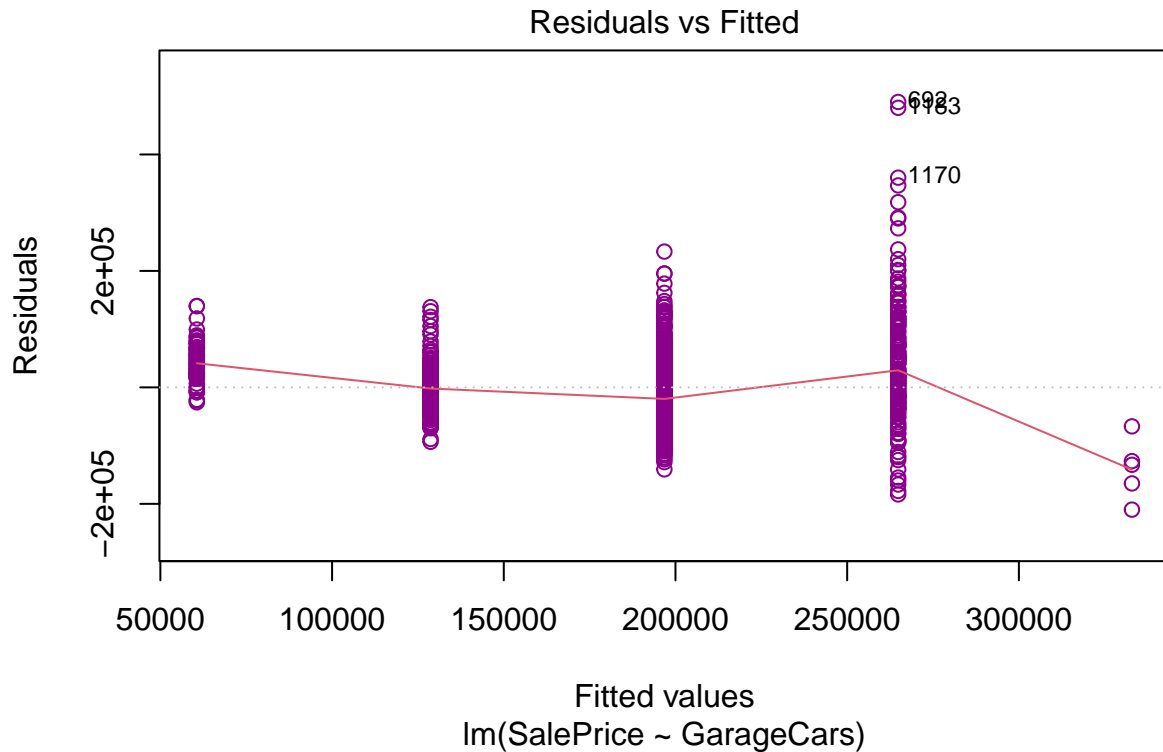
Ao analisar a normalidade dos resíduos com o `shapiro.test()` verificamos que o p-value é muito próximo de zero o que quer dizer que eles não seguem uma normalidade e que os intervalos de confiança estão comprometidos.

```
shapiro.test(residuals(fit4))
```

```
##  
##  Shapiro-Wilk normality test  
##  
## data:  residuals(fit4)  
## W = 0.90407, p-value < 2.2e-16
```

```
plot(fit4,which = 1,col=c("#8b008b"))
```

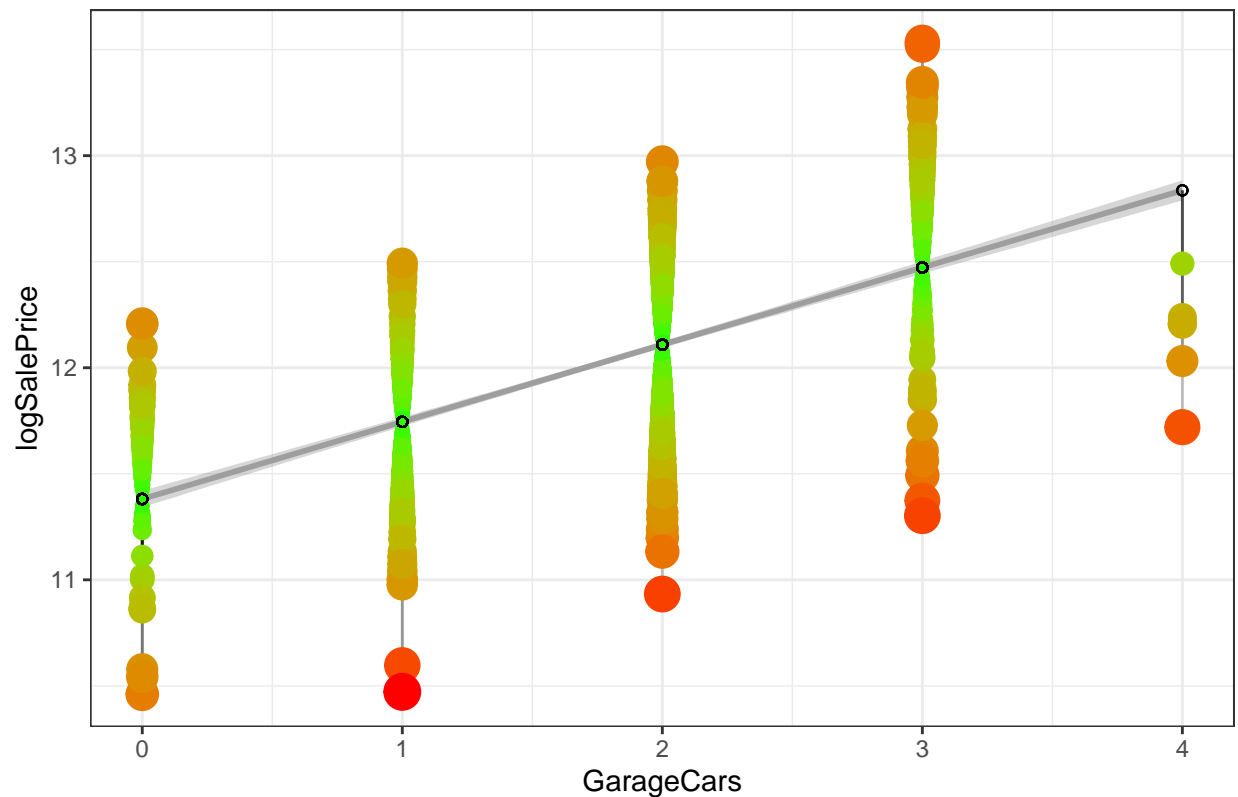




Para tentar chegar em uma normalidade aplicamos um log na variável SalePrice.

```
logSalePrice <- log(data$SalePrice)
fit <- lm(logSalePrice ~ GarageCars, data = data)
data$predicted <- predict(fit)
data$residuals <- residuals(fit)
ggplot(data, aes(x = GarageCars, y = logSalePrice)) +
  geom_smooth(formula = y ~ x, method = "lm", se = TRUE, color = "8b008b") +
  geom_segment(aes(xend = GarageCars, yend = predicted), alpha = .2) +
  geom_point(aes(color = abs(residuals), size = abs(residuals))) +
  scale_color_continuous(low = "green", high = "red") +
  guides(color = FALSE, size = FALSE) +
  geom_point(aes(y = predicted), shape = 1) +
  ggtitle("Regressão Lienar com ajustes - GarageCars")+
  theme_bw()
```

## Regressão Linear com ajustes – GarageCars

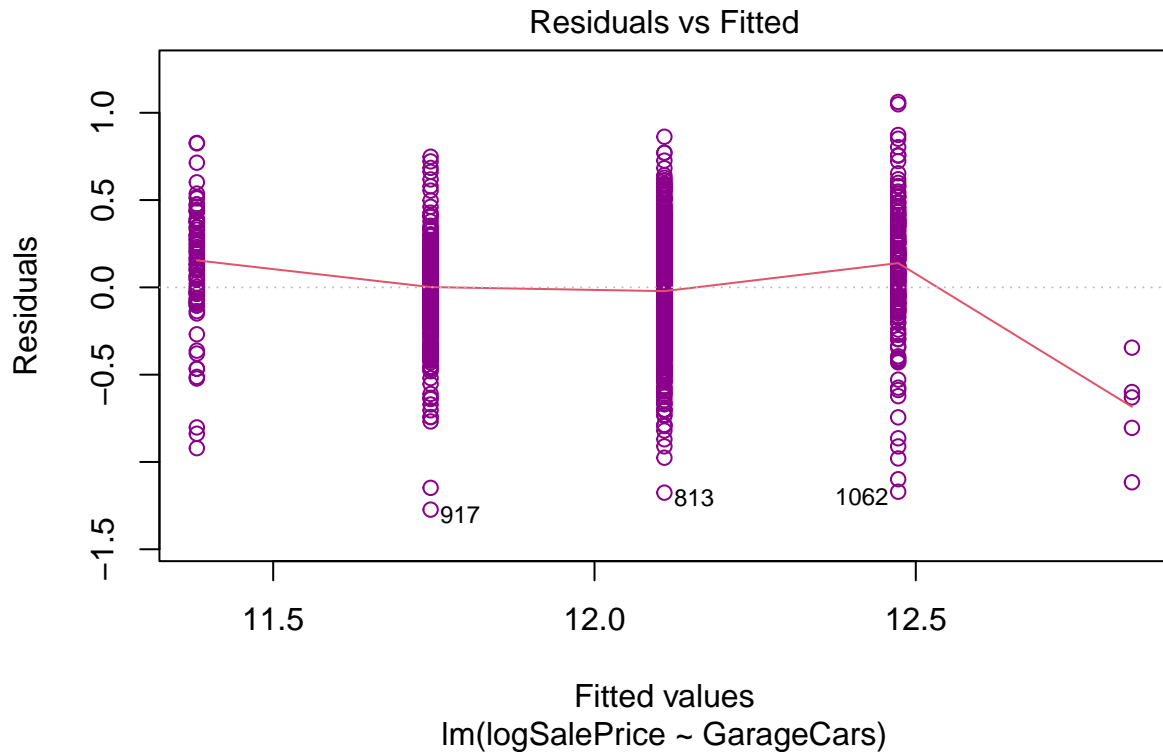


Verificamos o p-value. Contudo, ainda encontramos um valor muito próximo de zero que faz com que não cheguemos em uma normalidade.

```
shapiro.test(residuals(fit))
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: residuals(fit)  
## W = 0.98394, p-value = 1.132e-11
```

```
plot(fit, which = 1, col=c("#8b008b"))
```



Sendo assim, verificamos que não conseguimos definir uma regressão linear bem ajustada, visto que usamos o modelo simples. Contudo, mesmo assim iremos calcular os intervalos e confiança.

Intervalos de Confiança para a Variável OverallQual utilizando a função `confint()`.

```
confint(fit1)
```

```
##                2.5 %    97.5 %
## (Intercept) -107497.80 -84914.35
## OverallQual  43630.29  47241.31
```

Intervalos de Confiança para a Variável OverallCond utilizando a função `confint()`.

```
confint(fit2)
```

```
##                2.5 %    97.5 %
## (Intercept) 191122.470 232696.715
## OverallCond  -9214.449 -1901.782
```

Intervalos de Confiança para a Variável YearBuilt utilizando a função `confint()`.

```
confint(fit3)
```

```
##              2.5 %      97.5 %  
## (Intercept) -2757384.656 -2303231.836  
## YearBuilt    1260.194      1490.553
```

Intervalos de Confiança para a Variável GarageCars utilizando a função `confint()`.

```
confint(fit4)
```

```
##              2.5 %    97.5 %  
## (Intercept) 52572.02 68665.94  
## GarageCars  63883.70 72272.30
```

Concluimos que conseguimos determinar algumas relações e informações acerca da leitura e interpretação dos dados, algumas dela: ‘conforme a qualidade do imóvel aumenta o seu preço de venda aumentará também’, ‘quanto mais carros couber na garagem do imóvel maior será o preço de venda’, frequências etc. Ademais, a normalidade dos resíduos não foi suficiente para gerar um intervalo de confiança para o coeficiente da angular da regressão linear mesmo realizando alguns ajustes, visto que foi feita uma regressão linear simples.

As referências necessárias para a realização desse trabalho estão abaixo em forma de links. Agradecemos a todos os autores pelo conhecimento compartilhado.

- Prof. Afrânio Vieira por todas as aulas
- <https://r-graph-gallery.com/index.html> (Gráficos)
- <https://www.statology.org/univariate-analysis-in-r/>
- <https://www.statology.org/bivariate-analysis-in-r/>
- <https://rpubs.com/iabrady/residual-analysis>
- <https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/shapiro.test>
- [https://cran.r-project.org/web/packages/olsrr/vignettes/residual\\_diagnostics.html](https://cran.r-project.org/web/packages/olsrr/vignettes/residual_diagnostics.html)