MLR_in_R

Dima Mikhaylov

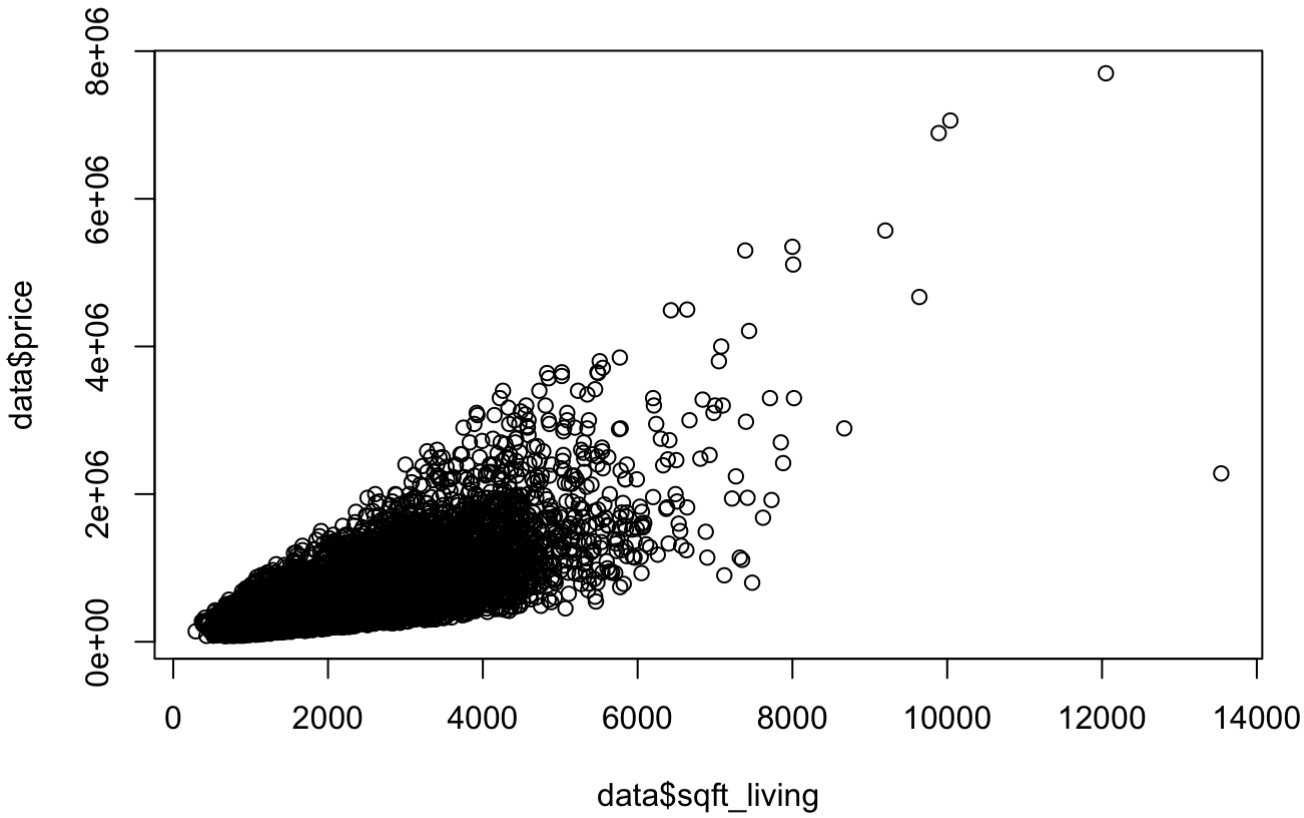
1/4/2022

```
data <- read.csv("house_ins.csv")
head(data)
```

```
##     price bedrooms bathrooms sqft_living sqft_lot floors waterfront view
## 1  221900        3      1.00        1180     5650      1          0    0
## 2  538000        3      2.25        2570     7242      2          0    0
## 3  180000        2      1.00         770    10000      1          0    0
## 4  604000        4      3.00        1960     5000      1          0    0
## 5  510000        3      2.00        1680     8080      1          0    0
## 6 1230000        4      4.50        5420   101930      1          0    0
##   condition grade sqft_above sqft_basement yr_built yr_renovated sqft_living15
## 1         3     7       1180             0     1955            0          1340
## 2         3     7       2170           400     1951         1991          1690
## 3         3     6        770             0     1933            0          2720
## 4         5     7       1050           910     1965            0          1360
## 5         3     8       1680             0     1987            0          1800
## 6         3    11       3890          1530     2001            0          4760
##   sqft_lot15
## 1       5650
## 2       7639
## 3       8062
## 4       5000
## 5       7503
## 6     101930
```

What is regression analysis? Any regression is a form of predictive modeling that investigates the relationship between a dependent, response or target) and independent variable(s), aka predictor or input.

```
plot(data$sqft_living, data$price)
```



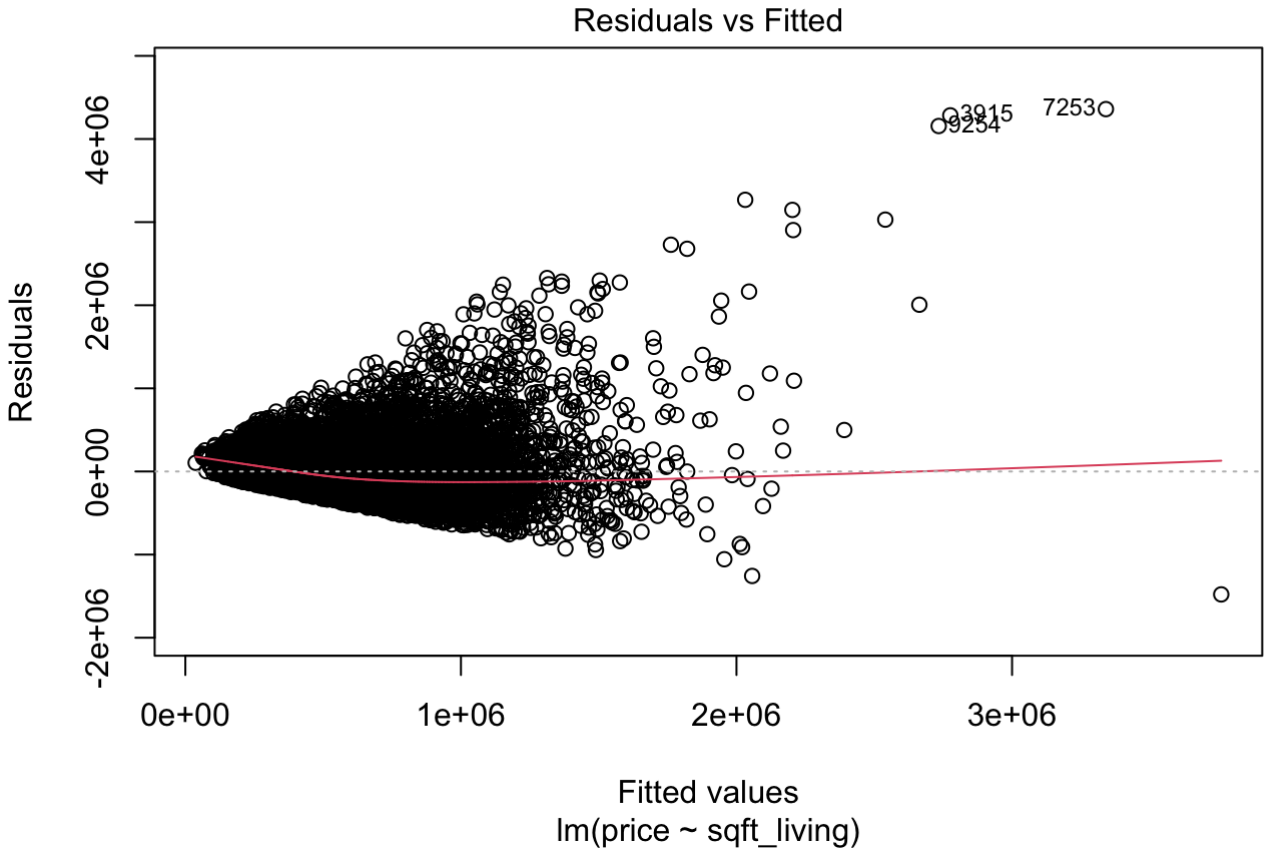
There are many possible types of regressions - a good classification will take into account 1) number of independent predictors, 2) shape of the regression line, 3) type of dependent variable.

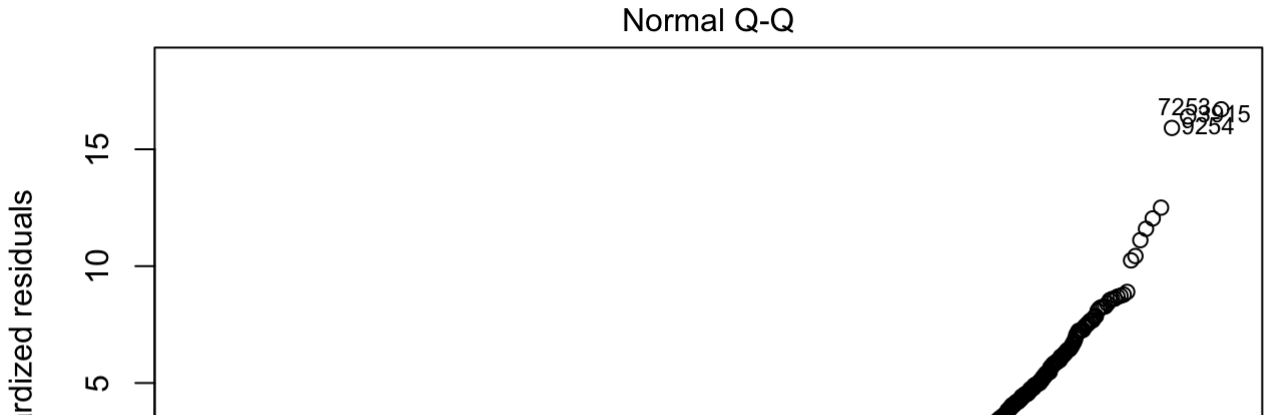
```
basic_model <- lm(price~sqft_living, data=data)
summary(basic_model)
```

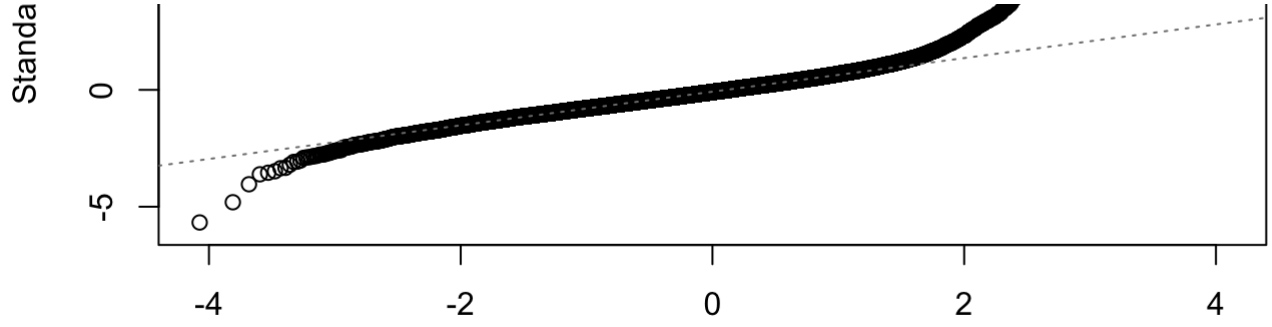
```
##
## Call:
## lm(formula = price ~ sqft_living, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1478973  -147600   -24142   106322  4359517
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -43953.635   4406.184  -9.975   <2e-16 ***
## sqft_living    280.866      1.938 144.910   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 261600 on 21607 degrees of freedom
## Multiple R-squared:  0.4929, Adjusted R-squared:  0.4928
## F-statistic: 2.1e+04 on 1 and 21607 DF,  p-value: < 2.2e-16
```

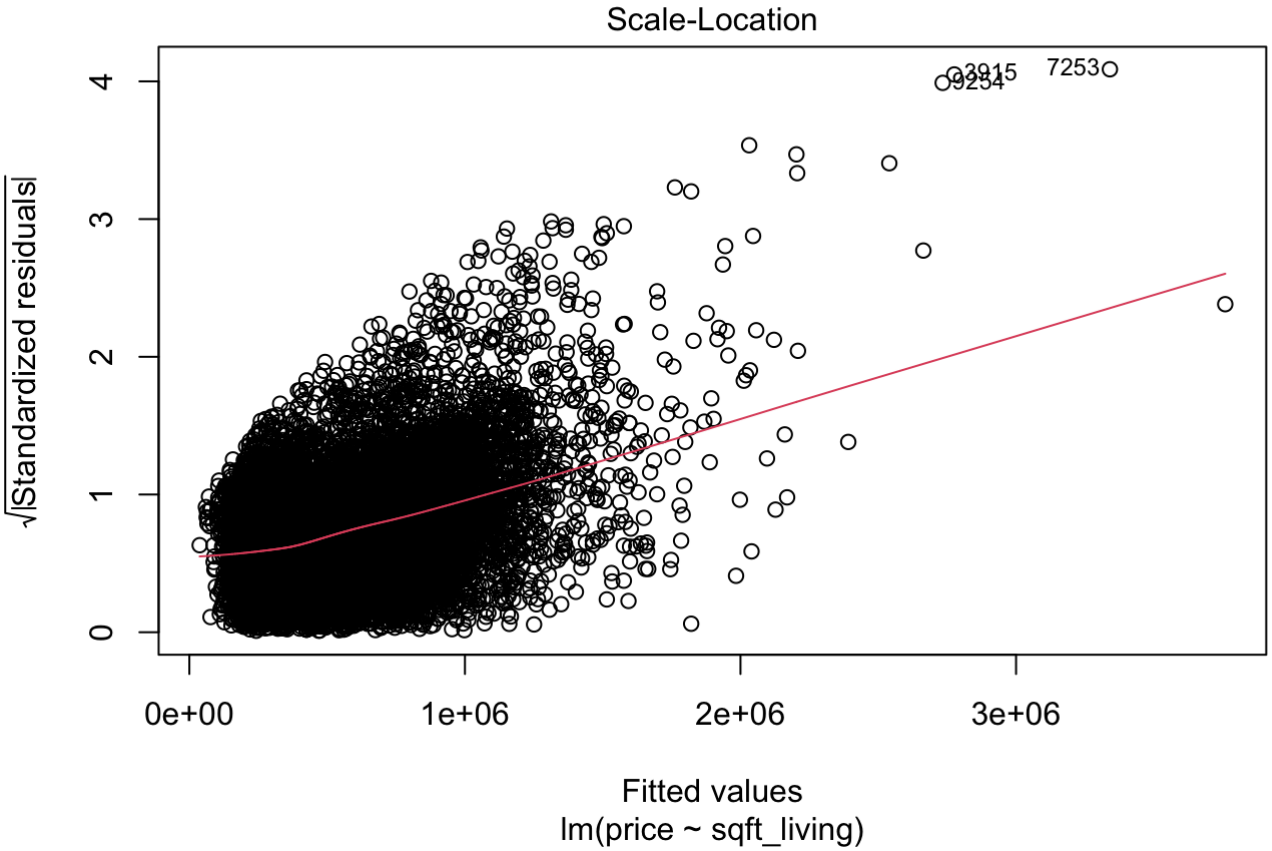
Linear regression can be used when we have linear (in coefficients) relationship between the variables and this is assured by two fundamental assumptions for the residuals: zero mean and constant variance of residuals agains fitted values. Important consideration include multicollinearity, autocorrelation and heteroskedasticity.

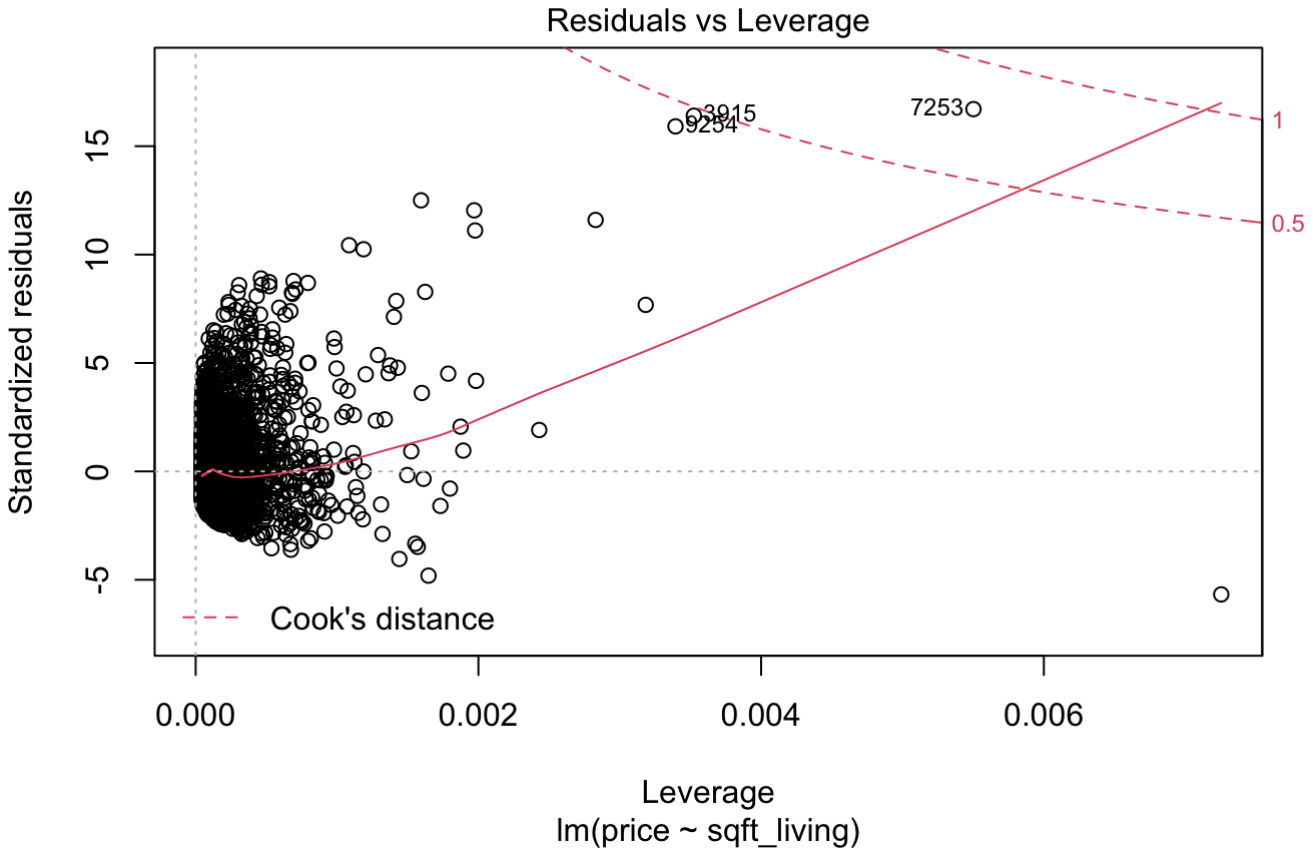
```
plot(basic_model)
```


Residuals vs Fitted
Residuals
4e+06
2e+06
0e+00
-2e+06
3915
7253
9254
0e+00
1e+06
2e+06
3e+06
Fitted values
lm(price ~ sqft_living)


Normal Q-Q
rdized residuals
15
10
5
7253
3915
9254


Standa
0
-5
-4
-2
0
2
4
Theoretical Quantiles
lm(price ~ sqft_living)

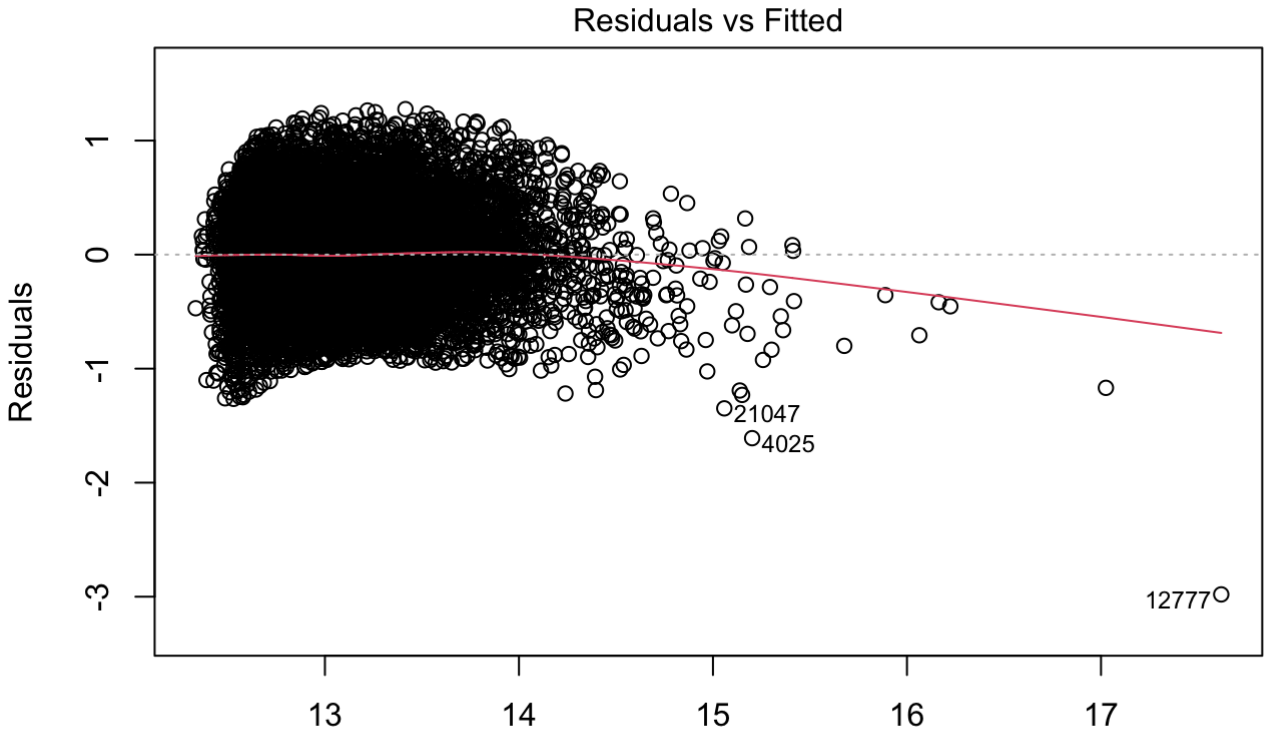

Scale-Location
3915
7253
9254
√|Standardized residuals|
0
1
2
3
4
0e+00
1e+06
2e+06
3e+06
Fitted values
lm(price ~ sqft_living)

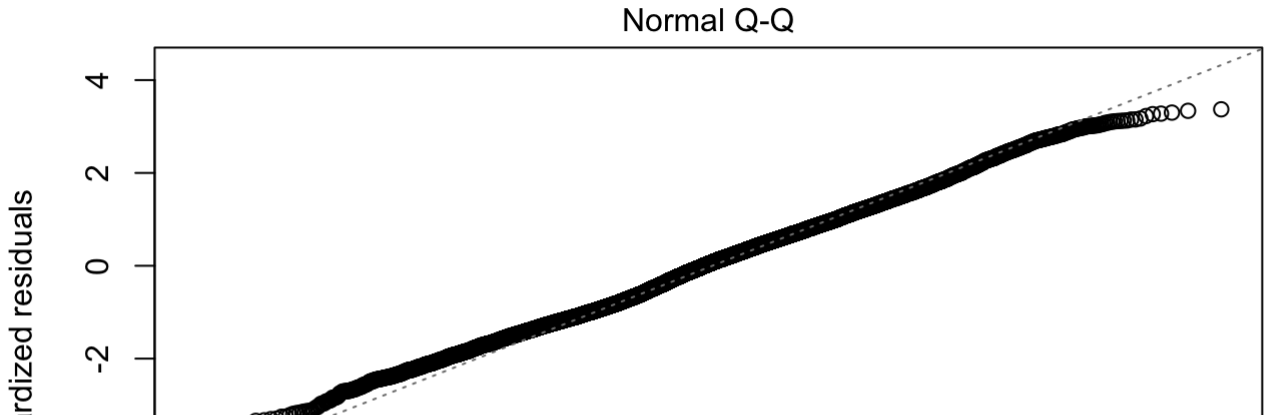


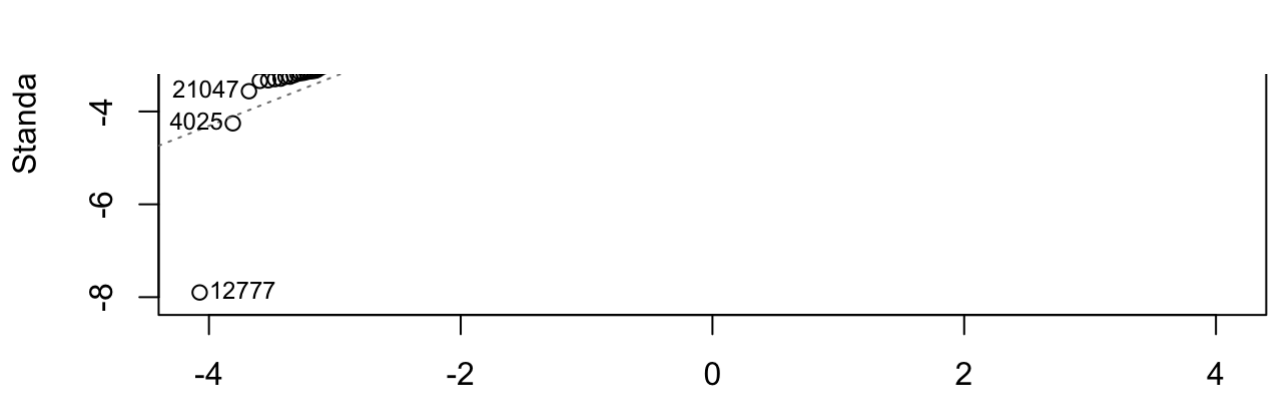

```
log_model <- lm(log(price)~sqft_living, data=data)
summary(log_model)
```

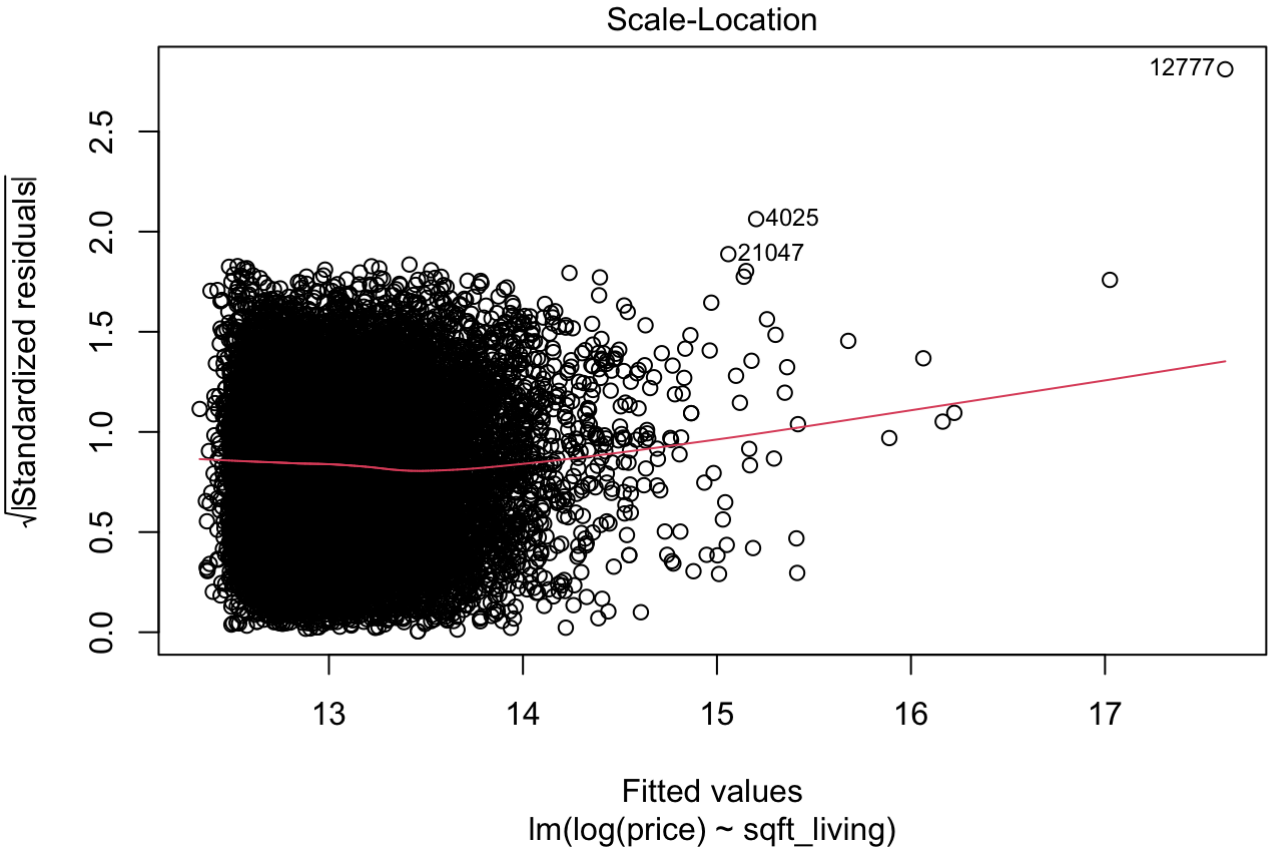
```
##
## Call:
## lm(formula = log(price) ~ sqft_living, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.97990 -0.28561  0.01473  0.26058  1.27602
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.222e+01  6.377e-03  1916.0   <2e-16 ***
## sqft_living 3.989e-04  2.805e-06   142.2   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3786 on 21607 degrees of freedom
## Multiple R-squared:  0.4835, Adjusted R-squared:  0.4835
## F-statistic: 2.023e+04 on 1 and 21607 DF,  p-value: < 2.2e-16
```

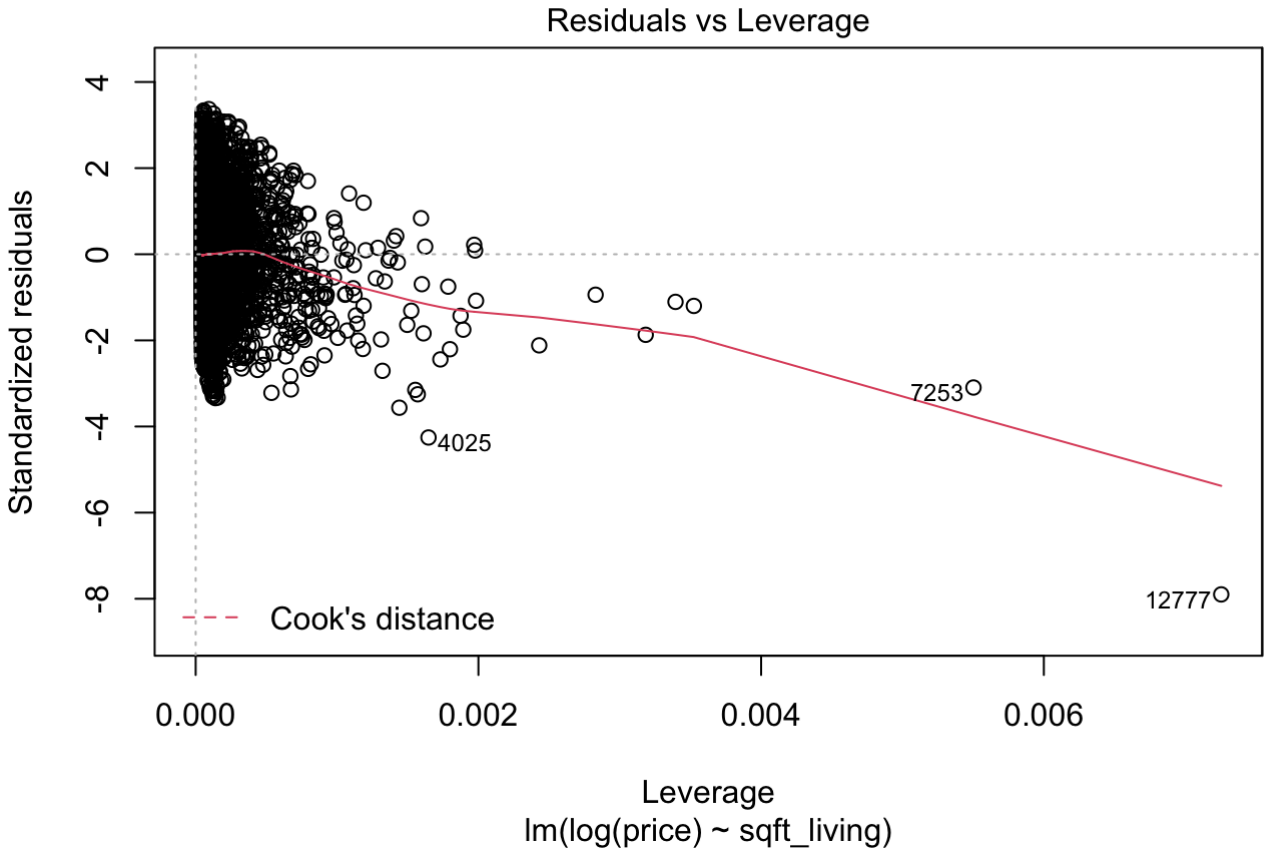
```
plot(log_model)
```


Residuals vs Fitted
Residuals
1
0
-1
-2
-3
21047
4025
12777
13
14
15
16
17
Fitted values
lm(log(price) ~ sqft_living)


Normal Q-Q
rdized residuals
4
2
0
-2


Standa
-4
-6
-8
21047
4025
12777
-4
-2
0
2
4
Theoretical Quantiles
lm(log(price) ~ sqft_living)


Scale-Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
2.5
12777
4025
21047
13
14
15
16
17
Fitted values
lm(log(price) ~ sqft_living)



Categorical variables, binary for example, may cause interractions, meaning their respective lines are not parralled and can intersect at some point.

```
data$waterfront <- factor(data$waterfront, ordered = TRUE, levels = c(0, 1))
```

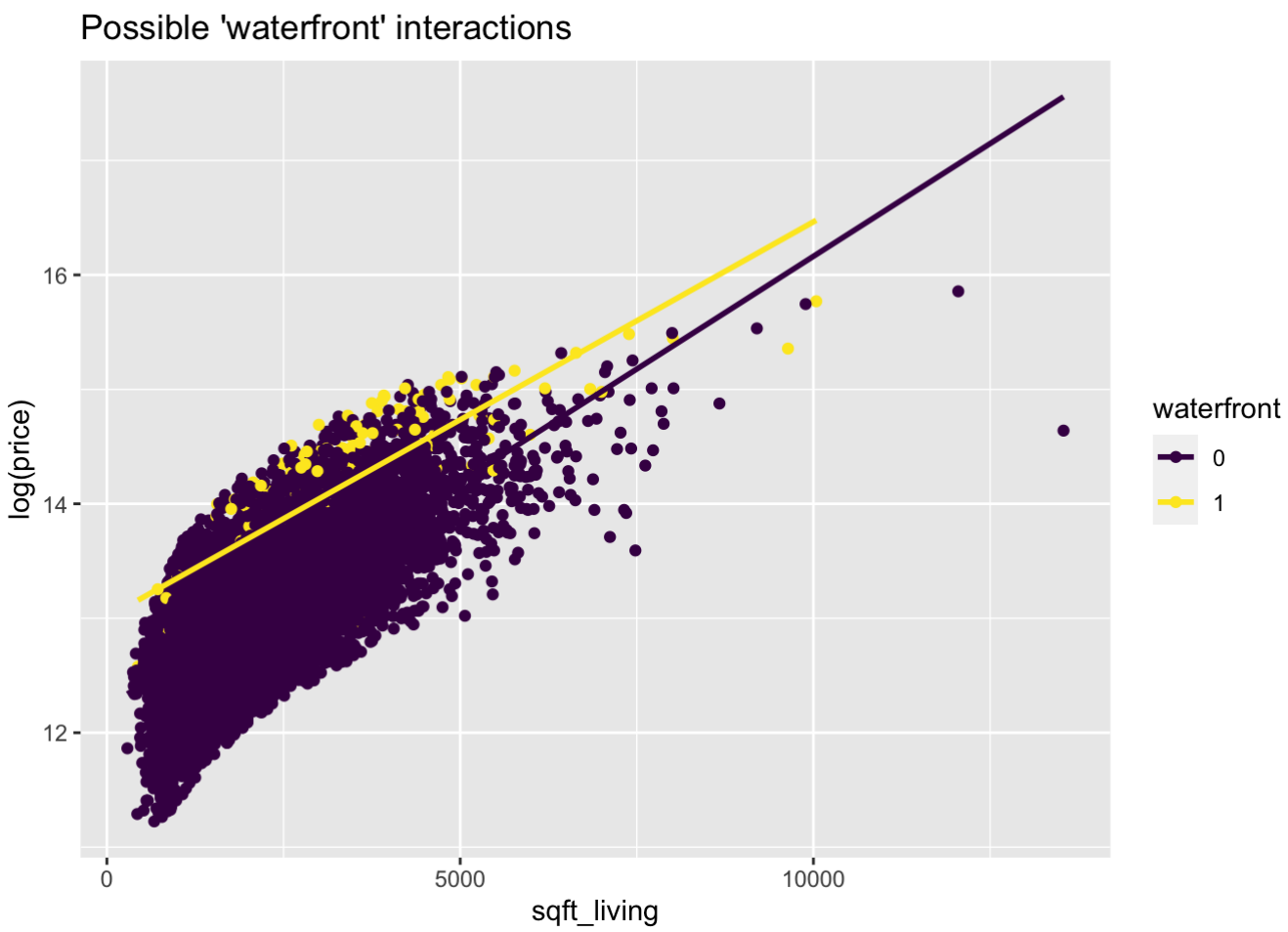
```
library(ggplot2)
ggplot(data, aes(x=sqft_living, y=price, color=waterfront))+
  geom_point()+
  geom_smooth(method = "lm", se=FALSE)+
  labs(x="sqft_living",
       y="price",
       title="Possible 'waterfront' interactions")
```

```
## `geom_smooth()` using formula 'y ~ x'
```

Log transformation also helps with this:

```
ggplot(data, aes(x=sqft_living, y=log(price), color=waterfront))+
  geom_point()+
  geom_smooth(method = "lm", se=FALSE)+
  labs(x="sqft_living",
       y="log(price)",
       title="Possible 'waterfront' interactions")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Important consideration include multicollinearity, autocorrelation and heteroskedasticity.

```
quant_vars <- subset(data, select=-waterfront)
str(quant_vars)
```

```
## 'data.frame':    21609 obs. of  15 variables:
##  $ price        : num  221900 538000 180000 604000 510000 ...
##  $ bedrooms     : int  3 3 2 4 3 4 3 3 3 3 ...
##  $ bathrooms    : num  1 2.25 1 3 2 4.5 2.25 1.5 1 2.5 ...
##  $ sqft_living  : int  1180 2570 770 1960 1680 5420 1715 1060 1780 1890 ...
##  $ sqft_lot     : int  5650 7242 10000 5000 8080 101930 6819 9711 7470 6560 ...
##  $ floors       : num  1 2 1 1 1 1 2 1 1 2 ...
##  $ view         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ condition    : int  3 3 3 5 3 3 3 3 3 3 ...
##  $ grade        : int  7 7 6 7 8 11 7 7 7 7 ...
##  $ sqft_above   : int  1180 2170 770 1050 1680 3890 1715 1060 1050 1890 ...
##  $ sqft_basement: int  0 400 0 910 0 1530 0 0 730 0 ...
##  $ yr_built     : int  1955 1951 1933 1965 1987 2001 1995 1963 1960 2003 ...
##  $ yr_renovated : int  0 1991 0 0 0 0 0 0 0 0 ...
##  $ sqft_living15: int  1340 1690 2720 1360 1800 4760 2238 1650 1780 2390 ...
##  $ sqft_lot15   : int  5650 7639 8062 5000 7503 101930 6819 9711 8113 7570 ...
```

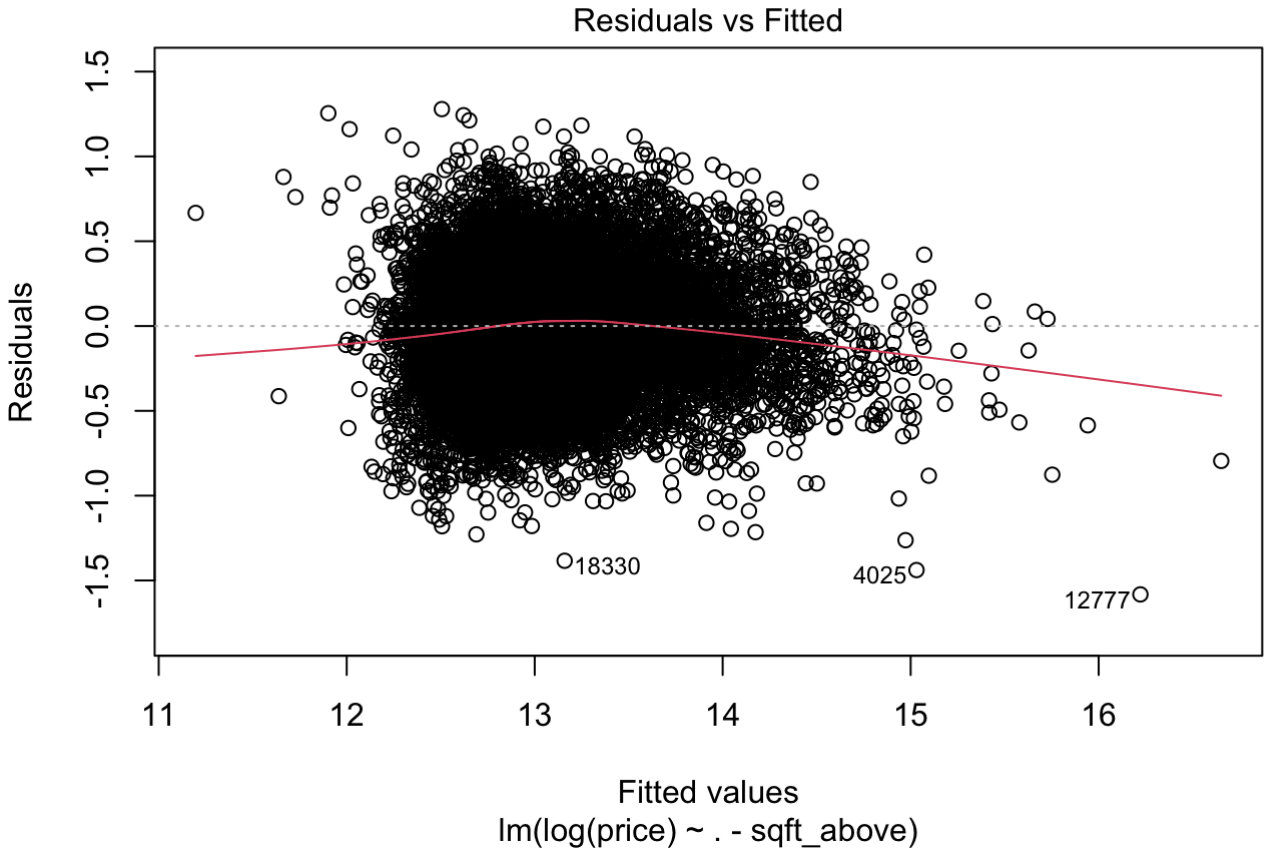
```
library(ggcorrplot)
corr <- round(cor(quant_vars), 1)
ggcorrplot(corr,
           method = "circle",
           lab = TRUE,
           type = "lower",
)
```

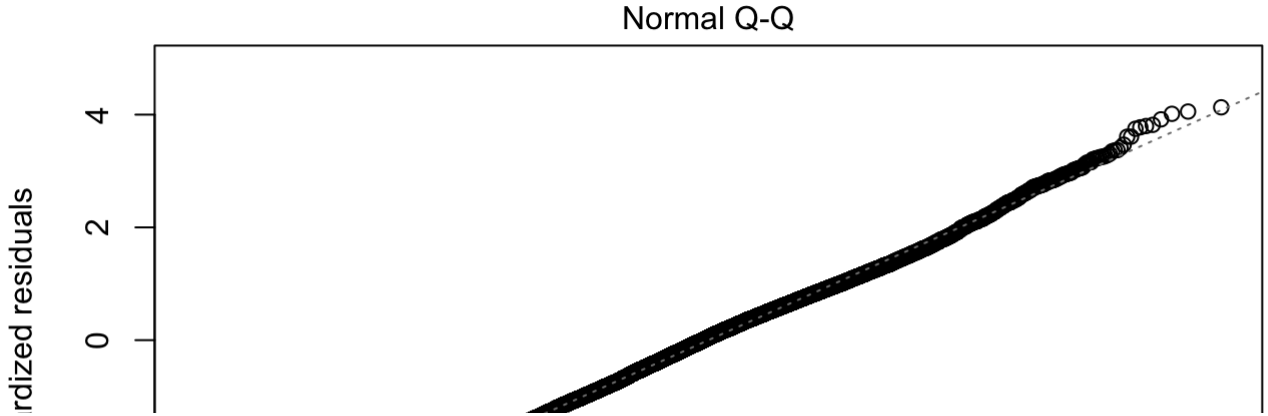
```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```

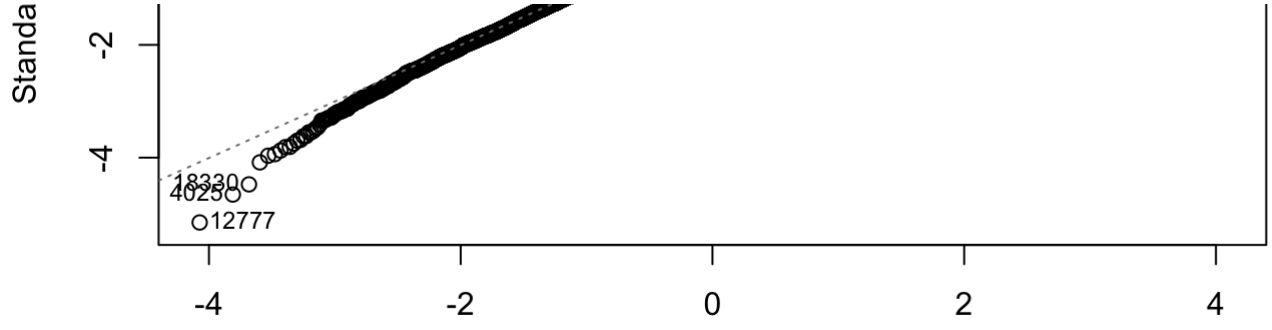

```
full_model <- lm(log(price) ~  . - sqft_above, data=data)
summary(full_model)
```

```
##
## Call:
## lm(formula = log(price) ~ . - sqft_above, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.58223 -0.20902  0.01461  0.20929  1.27886
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.150e+01  1.990e-01 108.023  < 2e-16 ***
## bedrooms      -2.265e-02  2.921e-03  -7.755 9.23e-15 ***
## bathrooms      7.793e-02  5.007e-03  15.562  < 2e-16 ***
## sqft_living    9.718e-05  5.593e-06  17.377  < 2e-16 ***
## sqft_lot       2.287e-07  7.350e-08   3.111  0.00187 **
## floors         1.218e-01  5.425e-03  22.446  < 2e-16 ***
## waterfront.L   2.483e-01  1.889e-02  13.149  < 2e-16 ***
## view           3.793e-02  3.258e-03  11.641  < 2e-16 ***
## condition      4.273e-02  3.579e-03  11.941  < 2e-16 ***
## grade          2.080e-01  3.224e-03  64.532  < 2e-16 ***
## sqft_basement  9.026e-05  6.514e-06  13.857  < 2e-16 ***
## yr_built      -5.453e-03  1.018e-04 -53.583  < 2e-16 ***
## yr_renovated   1.504e-05  5.612e-06   2.679  0.00738 **
## sqft_living15  1.129e-04  5.160e-06  21.888  < 2e-16 ***
## sqft_lot15    -5.289e-07  1.123e-07  -4.710 2.50e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3097 on 21594 degrees of freedom
## Multiple R-squared:  0.6546, Adjusted R-squared:  0.6544
## F-statistic:  2923 on 14 and 21594 DF,  p-value: < 2.2e-16
```

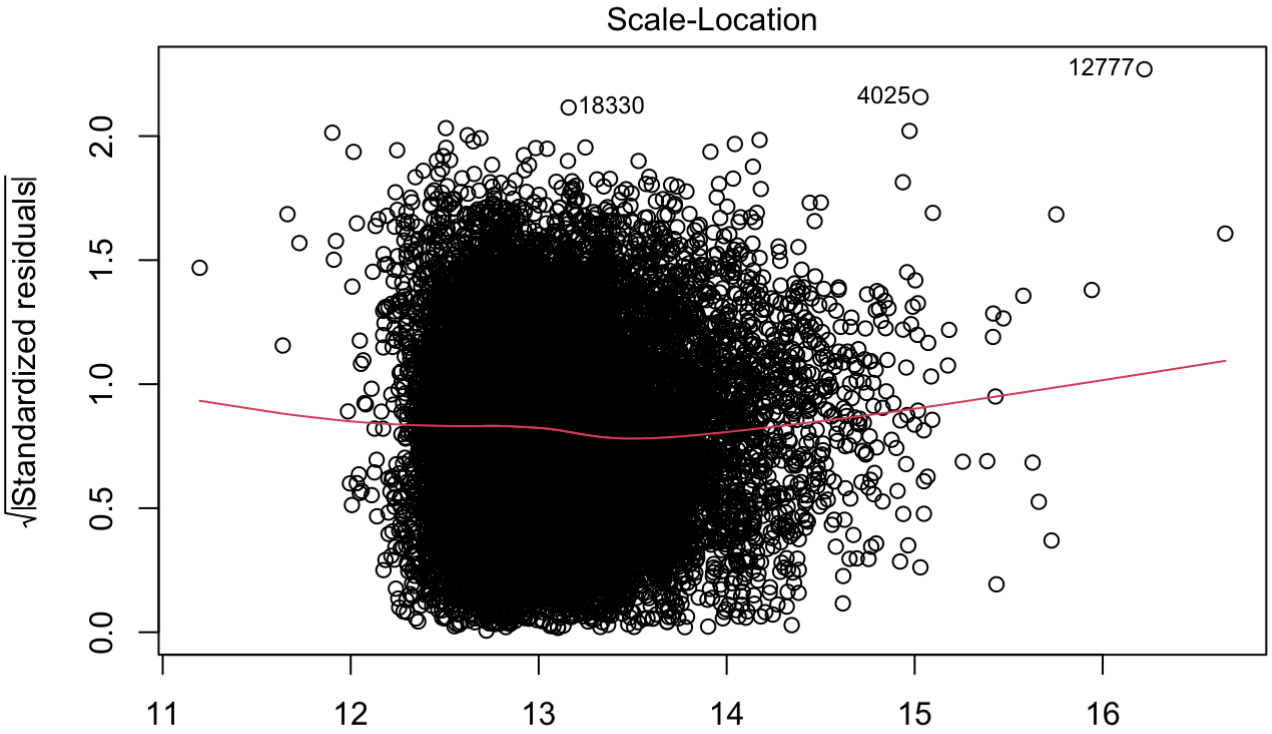
```
plot(full_model)
```


Residuals vs Fitted
Residuals
1.5
1.0
0.5
0.0
-0.5
-1.0
-1.5
18330
4025
12777
11
12
13
14
15
16
Fitted values
lm(log(price) ~ . - sqft_above)

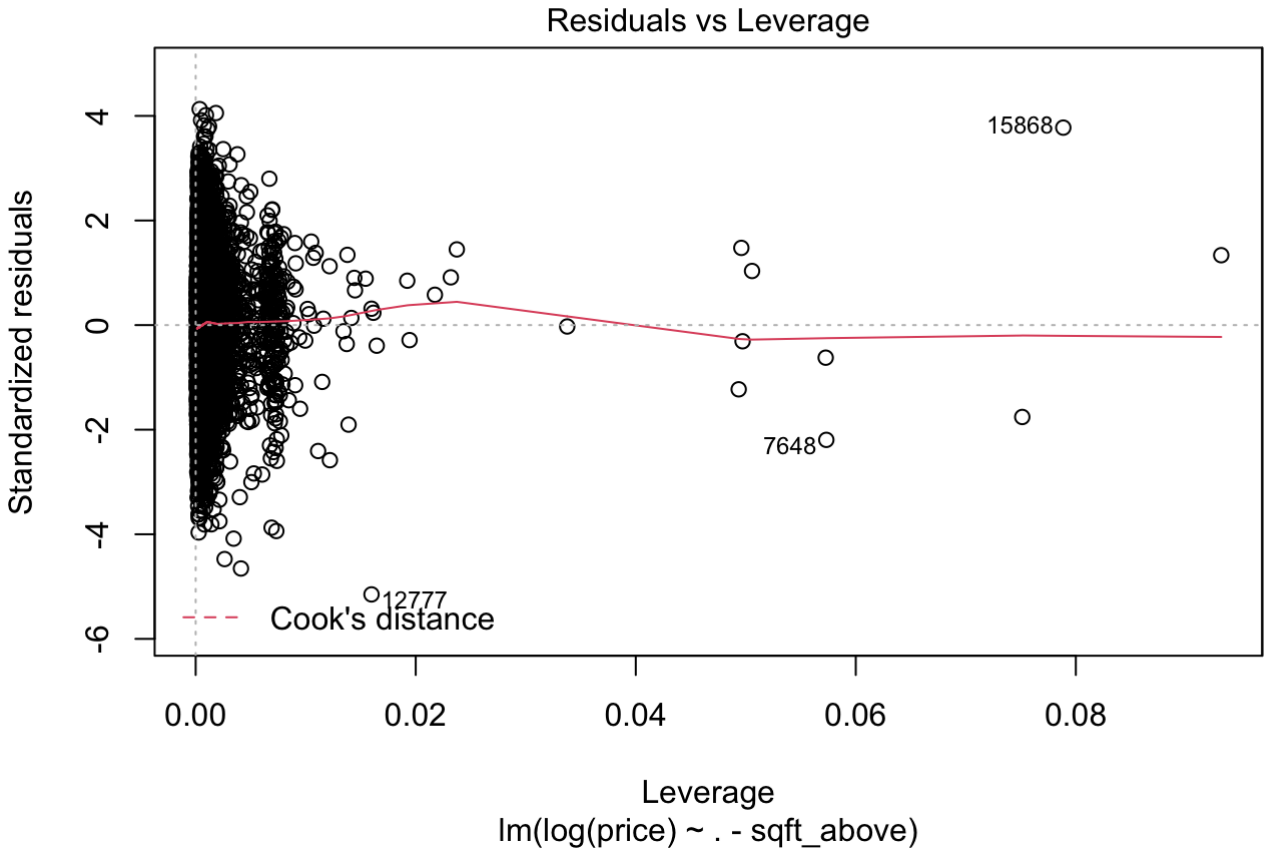

Normal Q-Q
rdized residuals
4
2
0



Theoretical Quantiles
lm(log(price) ~ . - sqft_above)



Fitted values
lm(log(price) ~ . - sqft_above)



Noteworthy is sensitivity to outliers - high-leverage vs influential observations. Not every outlier turns out to be an influential point that changes the slop of regression.

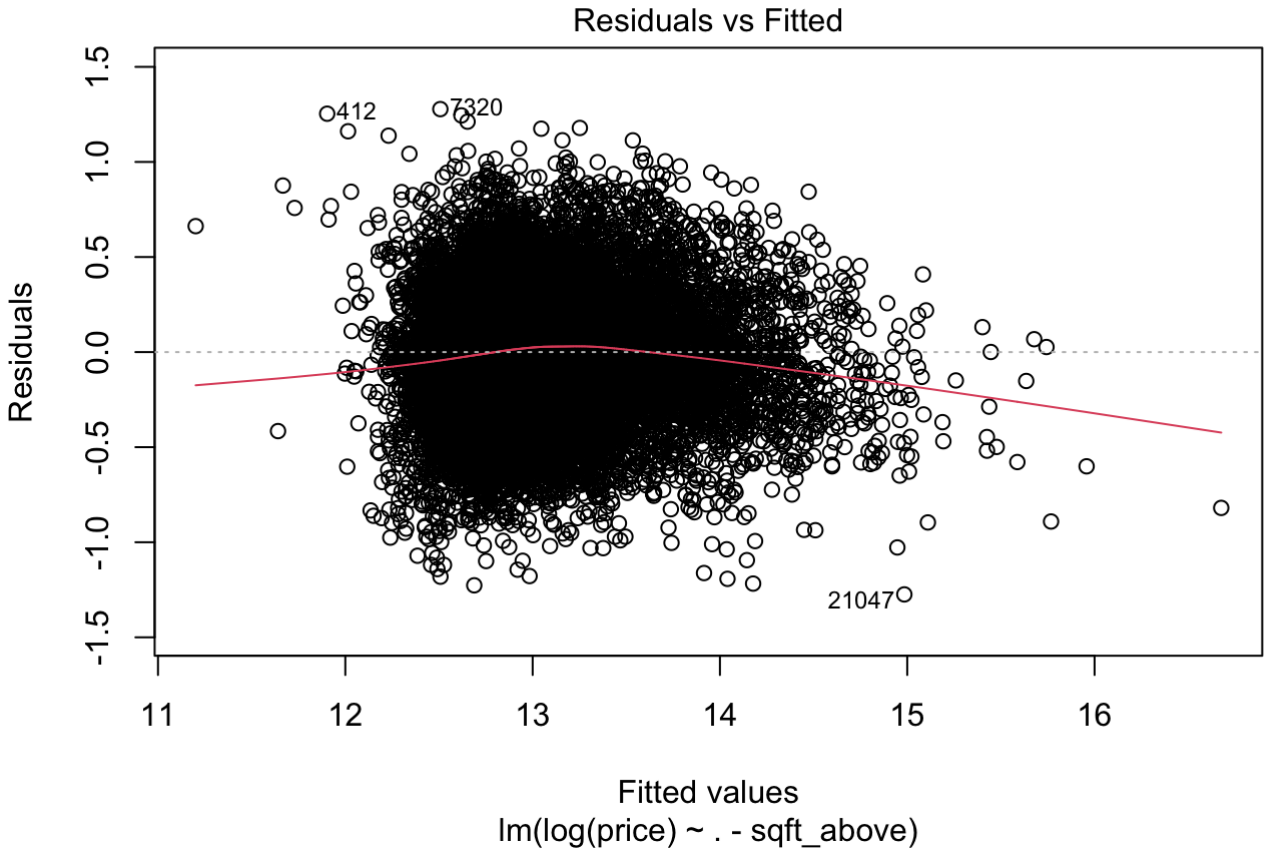
```
data[c(12777, 4025, 18330), ]
```

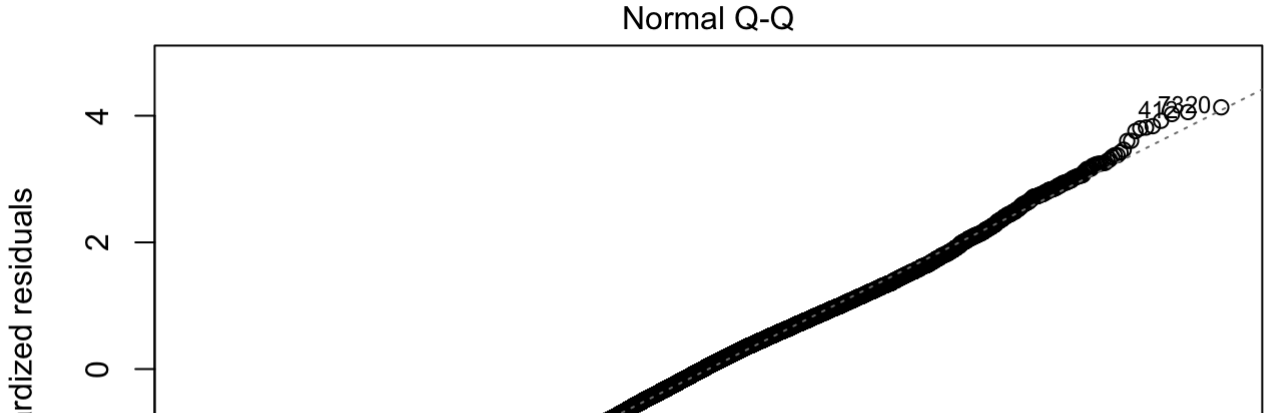
```
##         price bedrooms bathrooms sqft_living sqft_lot floors waterfront view
## 12777 2280000        7      8.00       13540   307752      3          0    4
## 4025   800000        7      6.75        7480    41664      2          0    2
## 18330  130000        3      1.00        1200     7000      2          0    0
##       condition grade sqft_above sqft_basement yr_built yr_renovated
## 12777         3    12       9410          4130     1999            0
## 4025          3    11       5080          2400     1953            0
## 18330         1     7       1200             0     1908            0
##       sqft_living15 sqft_lot15
## 12777          4850     217800
## 4025           2810      33190
## 18330          3290       6000
```

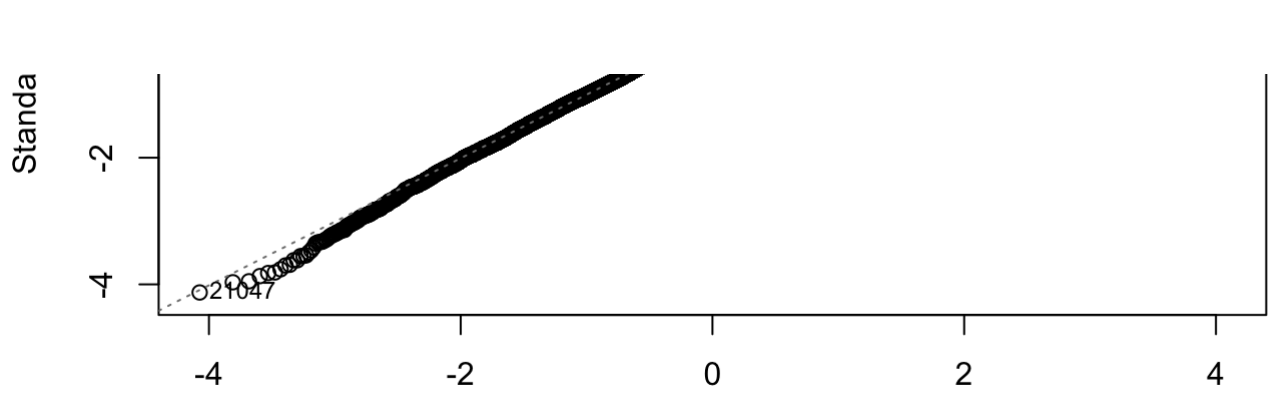
```
data_no <- data[-c(12777, 4025, 18330), ]
linear_model <- lm(log(price) ~ . -sqft_above, data=data_no)
summary(linear_model)
```

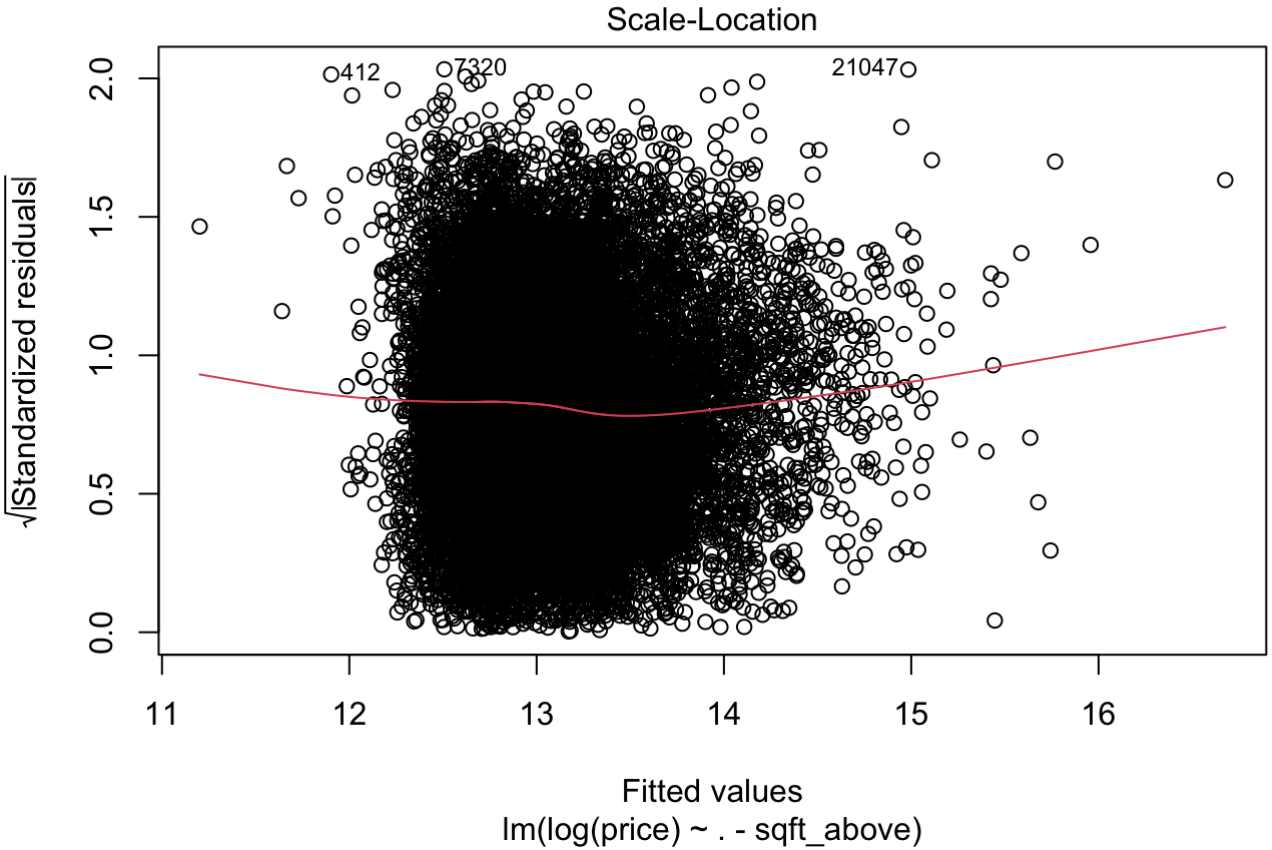
```
##
## Call:
## lm(formula = log(price) ~ . - sqft_above, data = data_no)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.27437 -0.20931  0.01438  0.20920  1.27783
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.156e+01  1.989e-01 108.395  < 2e-16 ***
## bedrooms      -2.315e-02  2.918e-03  -7.934 2.23e-15 ***
## bathrooms      7.849e-02  5.002e-03  15.692  < 2e-16 ***
## sqft_living    9.984e-05  5.607e-06  17.807  < 2e-16 ***
## sqft_lot       2.314e-07  7.339e-08   3.152  0.00162 **
## floors         1.223e-01  5.418e-03  22.578  < 2e-16 ***
## waterfront.L   2.462e-01  1.886e-02  13.051  < 2e-16 ***
## view           3.826e-02  3.254e-03  11.758  < 2e-16 ***
## condition      4.180e-02  3.576e-03  11.689  < 2e-16 ***
## grade          2.071e-01  3.222e-03  64.279  < 2e-16 ***
## sqft_basement  9.083e-05  6.504e-06  13.965  < 2e-16 ***
## yr_built      -5.479e-03  1.017e-04 -53.890  < 2e-16 ***
## yr_renovated   1.400e-05  5.605e-06   2.498  0.01249 *
## sqft_living15  1.119e-04  5.161e-06  21.684  < 2e-16 ***
## sqft_lot15    -5.163e-07  1.122e-07  -4.603 4.18e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3093 on 21591 degrees of freedom
## Multiple R-squared:  0.6554, Adjusted R-squared:  0.6552
## F-statistic:  2934 on 14 and 21591 DF,  p-value: < 2.2e-16
```

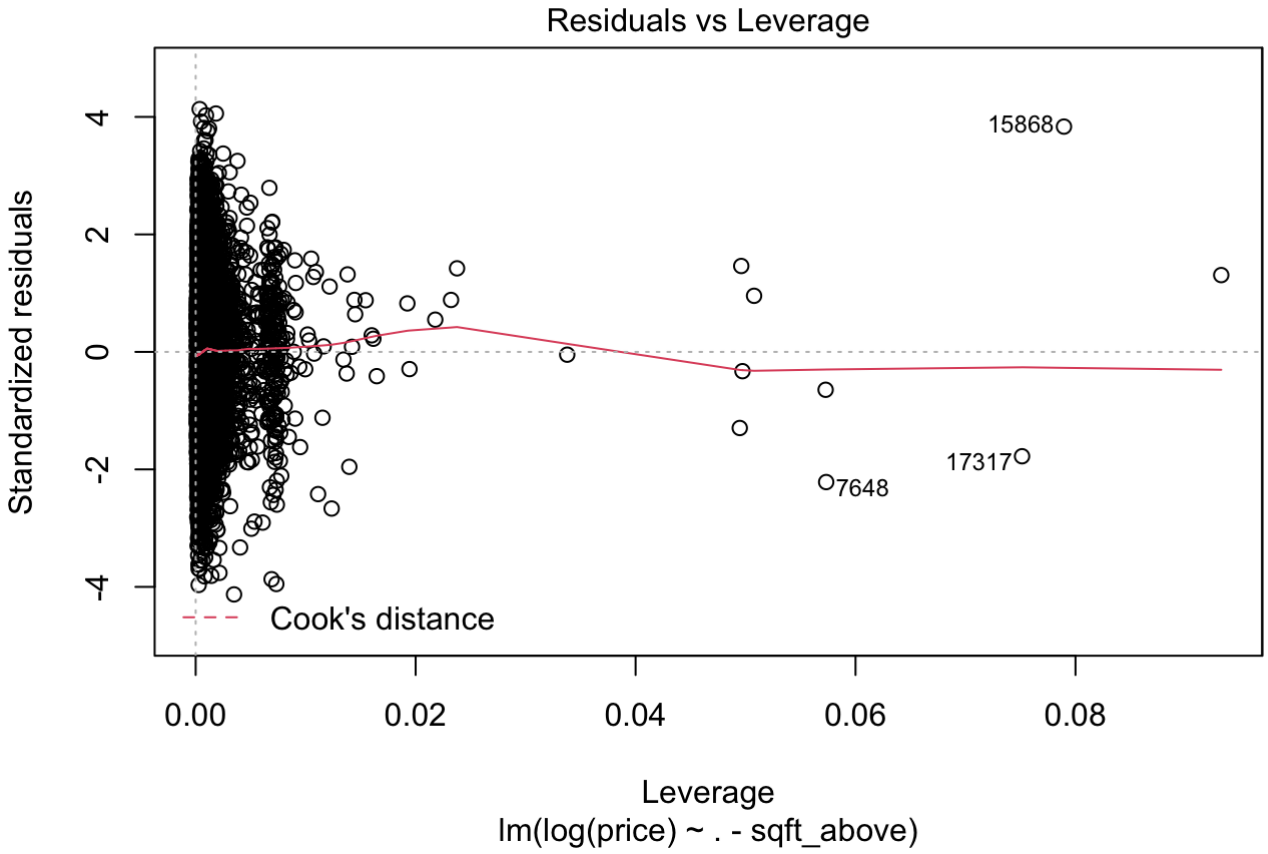
```
plot(linear_model)
```


Residuals vs Fitted
Residuals
Fitted values
lm(log(price) ~ . - sqft_above)
412
7320
21047
-1.5
-1.0
-0.5
0.0
0.5
1.0
1.5
11
12
13
14
15
16


Normal Q-Q
rdized residuals
0
2
4


Standa
-2
-4
21047
-4
-2
0
2
4
Theoretical Quantiles
lm(log(price) ~ . - sqft_above)


Scale-Location
√|Standardized residuals|
412
7320
21047
0.0
0.5
1.0
1.5
2.0
11
12
13
14
15
16
Fitted values
lm(log(price) ~ . - sqft_above)



In case of multiple predictors, consider either forward selection & backward elimination or step wise regression.

```
library(leaps)
allregs <- regsubsets(log(price) ~ . - sqft_above, data=data, nbest=1)
summary(allregs)
```

```
## Subset selection object
## Call: regsubsets.formula(log(price) ~ . - sqft_above, data = data,
##     nbest = 1)
## 14 Variables  (and intercept)
##               Forced in Forced out
## bedrooms          FALSE      FALSE
## bathrooms         FALSE      FALSE
## sqft_living       FALSE      FALSE
## sqft_lot          FALSE      FALSE
## floors            FALSE      FALSE
## waterfront.L      FALSE      FALSE
## view              FALSE      FALSE
## condition         FALSE      FALSE
## grade             FALSE      FALSE
## sqft_basement     FALSE      FALSE
## yr_built          FALSE      FALSE
## yr_renovated      FALSE      FALSE
## sqft_living15     FALSE      FALSE
## sqft_lot15        FALSE      FALSE
## 1 subsets of each size up to 8
## Selection Algorithm: exhaustive
##          bedrooms bathrooms sqft_living sqft_lot floors waterfront.L view
## 1  ( 1 ) " "      " "       " "         " "      " "    " "          " "
## 2  ( 1 ) " "      " "       " "         " "      " "    " "          " "
## 3  ( 1 ) " "      " "       "*"         " "      " "    " "          " "
## 4  ( 1 ) " "      " "       "*"         " "      " "    " "          "*"
## 5  ( 1 ) " "      "*"       "*"         " "      " "    " "          "*"
## 6  ( 1 ) " "      "*"       "*"         " "      " "    " "          "*"
## 7  ( 1 ) " "      "*"       " "         " "      "*"    "*"          " "
## 8  ( 1 ) " "      "*"       "*"         " "      "*"    "*"          " "
##          condition grade sqft_basement yr_built yr_renovated sqft_living15
## 1  ( 1 ) " "       "*"   " "           " "      " "          " "
## 2  ( 1 ) " "       "*"   " "           "*"      " "          " "
## 3  ( 1 ) " "       "*"   " "           "*"      " "          " "
## 4  ( 1 ) " "       "*"   " "           "*"      " "          " "
## 5  ( 1 ) " "       "*"   " "           "*"      " "          " "
## 6  ( 1 ) " "       "*"   " "           "*"      " "          "*"
## 7  ( 1 ) " "       "*"   "*"           "*"      " "          "*"
## 8  ( 1 ) " "       "*"   "*"           "*"      " "          "*"
```

```
##          sqft_lot15
## 1  ( 1 ) " "
## 2  ( 1 ) " "
## 3  ( 1 ) " "
## 4  ( 1 ) " "
## 5  ( 1 ) " "
## 6  ( 1 ) " "
## 7  ( 1 ) " "
## 8  ( 1 ) " "
```

```
which.max(summary(allregs)$adjr2)
```

```
## [1] 8
```

```
coef(allregs, 8)
```

```
##   (Intercept)     bathrooms   sqft_living        floors  waterfront.L
##  2.262432e+01  7.624836e-02  8.028798e-05  1.217079e-01  3.470974e-01
##         grade sqft_basement      yr_built sqft_living15
##  2.160242e-01  1.096923e-04 -5.969946e-03  1.187650e-04
```

```
max_adjR <- lm(log(price) ~ bathrooms + sqft_living + floors + waterfront + grade + sqft_basement + yr_built + sq
ft_living15, data=data_no)
summary(max_adjR)
```

```
##
## Call:
## lm(formula = log(price) ~ bathrooms + sqft_living + floors +
##     waterfront + grade + sqft_basement + yr_built + sqft_living15,
##     data = data_no)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.24293 -0.21067  0.01291  0.21061  1.27986
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.265e+01  1.754e-01  129.15   <2e-16 ***
## bathrooms      7.643e-02  4.890e-03   15.63   <2e-16 ***
## sqft_living    8.262e-05  5.291e-06   15.62   <2e-16 ***
## floors         1.223e-01  5.391e-03   22.69   <2e-16 ***
## waterfront.L   3.457e-01  1.749e-02   19.77   <2e-16 ***
## grade          2.152e-01  3.196e-03   67.35   <2e-16 ***
## sqft_basement  1.102e-04  6.394e-06   17.23   <2e-16 ***
## yr_built      -5.985e-03  9.251e-05  -64.69   <2e-16 ***
## sqft_living15  1.180e-04  5.140e-06   22.97   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3118 on 21597 degrees of freedom
## Multiple R-squared:  0.6496, Adjusted R-squared:  0.6495
## F-statistic:  5005 on 8 and 21597 DF,  p-value: < 2.2e-16
```

```
regnull <- lm(log(price) ~ 1, data=data_no)
regfull <- lm(log(price) ~ ., data=data_no)
step(regfull, scope=list(lower=regnull, upper=regfull), direction = 'backward')
```

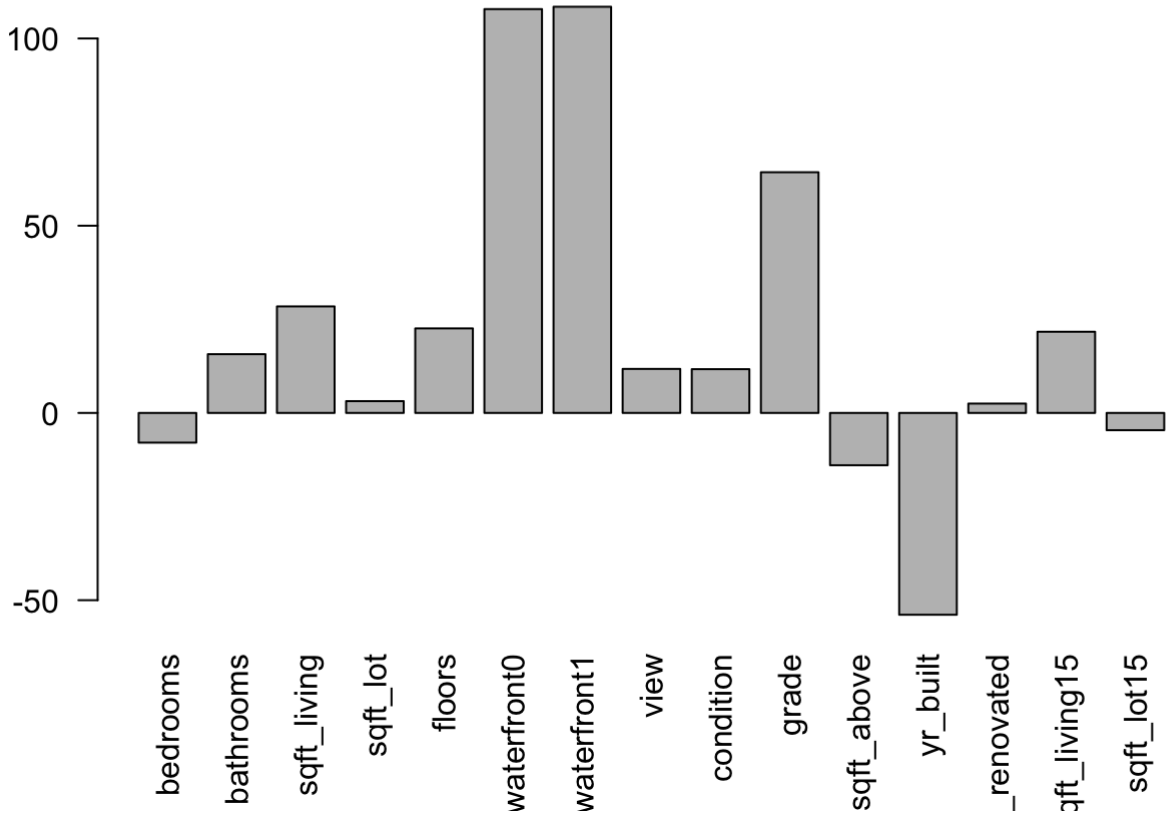
```
## Start:  AIC=-50698.57
## log(price) ~ bedrooms + bathrooms + sqft_living + sqft_lot +
##     floors + waterfront + view + condition + grade + sqft_above +
##     sqft_basement + yr_built + yr_renovated + sqft_living15 +
##     sqft_lot15
##
##
## Step:  AIC=-50698.57
## log(price) ~ bedrooms + bathrooms + sqft_living + sqft_lot +
##     floors + waterfront + view + condition + grade + sqft_above +
##     yr_built + yr_renovated + sqft_living15 + sqft_lot15
##
##                 Df Sum of Sq    RSS    AIC
## <none>                       2064.9 -50699
## - yr_renovated   1      0.60 2065.5 -50694
## - sqft_lot       1      0.95 2065.8 -50691
## - sqft_lot15     1      2.03 2066.9 -50679
## - bedrooms       1      6.02 2070.9 -50638
## - condition      1     13.07 2078.0 -50564
## - view           1     13.22 2078.1 -50563
## - waterfront     1     16.29 2081.2 -50531
## - sqft_above     1     18.65 2083.5 -50506
## - bathrooms      1     23.55 2088.4 -50456
## - sqft_living15  1     44.97 2109.9 -50235
## - floors         1     48.75 2113.7 -50196
## - sqft_living    1     77.41 2142.3 -49905
## - yr_built       1    277.74 2342.6 -47974
## - grade          1    395.15 2460.0 -46917
```

```
##
## Call:
## lm(formula = log(price) ~ bedrooms + bathrooms + sqft_living +
##     sqft_lot + floors + waterfront + view + condition + grade +
##     sqft_above + yr_built + yr_renovated + sqft_living15 + sqft_lot15,
##     data = data_no)
##
## Coefficients:
##   (Intercept)       bedrooms      bathrooms    sqft_living       sqft_lot
##     2.156e+01     -2.315e-02      7.849e-02      1.907e-04      2.314e-07
##        floors   waterfront.L           view      condition          grade
##     1.223e-01      2.462e-01      3.826e-02      4.180e-02      2.071e-01
##    sqft_above       yr_built   yr_renovated  sqft_living15     sqft_lot15
##    -9.083e-05     -5.479e-03      1.400e-05      1.119e-04     -5.163e-07
```

```
best_reg <- lm(log(price) ~ bedrooms + bathrooms + sqft_living +
    sqft_lot + floors + waterfront + view + condition + grade +
    sqft_above + yr_built + yr_renovated + sqft_living15 + sqft_lot15 + 0,
    data = data_no)
summary(best_reg)
```

```
##
## Call:
## lm(formula = log(price) ~ bedrooms + bathrooms + sqft_living +
##     sqft_lot + floors + waterfront + view + condition + grade +
##     sqft_above + yr_built + yr_renovated + sqft_living15 + sqft_lot15 +
##     0, data = data_no)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.27437 -0.20931  0.01438  0.20920  1.27783
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## bedrooms      -2.315e-02  2.918e-03  -7.934 2.23e-15 ***
## bathrooms      7.849e-02  5.002e-03  15.692  < 2e-16 ***
## sqft_living    1.907e-04  6.702e-06  28.451  < 2e-16 ***
## sqft_lot       2.314e-07  7.339e-08   3.152  0.00162 **
## floors         1.223e-01  5.418e-03  22.578  < 2e-16 ***
## waterfront0    2.138e+01  1.983e-01 107.839  < 2e-16 ***
## waterfront1    2.173e+01  2.003e-01 108.464  < 2e-16 ***
## view           3.826e-02  3.254e-03  11.758  < 2e-16 ***
## condition      4.180e-02  3.576e-03  11.689  < 2e-16 ***
## grade          2.071e-01  3.222e-03  64.279  < 2e-16 ***
## sqft_above    -9.083e-05  6.504e-06 -13.965  < 2e-16 ***
## yr_built      -5.479e-03  1.017e-04 -53.890  < 2e-16 ***
## yr_renovated   1.400e-05  5.605e-06   2.498  0.01249 *
## sqft_living15  1.119e-04  5.161e-06  21.684  < 2e-16 ***
## sqft_lot15    -5.163e-07  1.122e-07  -4.603 4.18e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3093 on 21591 degrees of freedom
## Multiple R-squared:  0.9994, Adjusted R-squared:  0.9994
## F-statistic: 2.567e+06 on 15 and 21591 DF,  p-value: < 2.2e-16
```

```
barplot(summary(best_reg)$coefficients[,3], las=2)
```




```
data_oos <- read.csv("house_oos.csv")
data_oos$waterfront <- factor(data_oos$waterfront, ordered = TRUE, levels = c(0, 1))
data_oos
```

```
##     price bedrooms bathrooms sqft_living sqft_lot floors waterfront view
## 1  520000       11      3.00        3000     4960      2          0    0
## 2 1150000       10      5.25        4590    10920      1          0    2
## 3  650000       10      2.00        3610    11914      2          0    0
## 4  660000       10      3.00        2920     3745      2          0    0
##   condition grade sqft_above sqft_basement yr_built yr_renovated sqft_living15
## 1         3     7       2400           600     1918         1999          1420
## 2         3     9       2500          2090     2008            0          2730
## 3         4     7       3010           600     1958            0          2040
## 4         4     7       1860          1060     1913            0          1810
##   sqft_lot15
## 1       4960
## 2      10400
## 3      11914
## 4       3745
```

```
exp(predict(best_reg, data_oos, level=0.95, interval="prediction"))
```

```
##        fit      lwr       upr
## 1 547659.4 298209.5 1005772.3
## 2 890698.2 485284.4 1634801.0
## 3 479753.5 261382.6  880561.3
## 4 631311.9 343960.4 1158722.6
```

```
data_ins <- data[which(data$bedrooms == 33), ]
data_ins
```

```
##        price bedrooms bathrooms sqft_living sqft_lot floors waterfront view
## 15868 640000       33      1.75        1620     6000      1          0    0
##       condition grade sqft_above sqft_basement yr_built yr_renovated
## 15868         5     7       1040           580     1947            0
##       sqft_living15 sqft_lot15
## 15868          1330       4700
```

```
exp(predict(best_reg, data_ins, level=0.95, interval="prediction"))
```

```
##            fit      lwr    upr
## 15868 205057.1 109252.1 384875
```

```
library(quantreg)
```

```
## Loading required package: SparseM
```

```
##
## Attaching package: 'SparseM'
```

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

```
#qs <- 1:9/10 # DELETE
q = c(0.5, 0.6, 0.7, 0.8, 0.9)
quant_reg <- rq(log(price) ~ bedrooms + bathrooms + sqft_living +
    sqft_lot + floors + waterfront + view + condition + grade +
    sqft_above + yr_built + yr_renovated + sqft_living15 + sqft_lot15 +0,
    data = data_no, tau = q)
summary(quant_reg)
```

```
## Warning in summary.rq(xi, U = U, ...): 2 non-positive fis

## Warning in summary.rq(xi, U = U, ...): 2 non-positive fis

## Warning in summary.rq(xi, U = U, ...): 2 non-positive fis
```

```
##
## Call: rq(formula = log(price) ~ bedrooms + bathrooms + sqft_living +
##     sqft_lot + floors + waterfront + view + condition + grade +
##     sqft_above + yr_built + yr_renovated + sqft_living15 + sqft_lot15 +
##     0, tau = q, data = data_no)
##
## tau: [1] 0.5
##
## Coefficients:
##               Value     Std. Error t value   Pr(>|t|)
## bedrooms       -0.02663   0.00390   -6.83298   0.00000
## bathrooms       0.08562   0.00637   13.43697   0.00000
## sqft_living     0.00019   0.00001   22.85014   0.00000
## sqft_lot        0.00000   0.00000    2.71259   0.00668
## floors          0.13649   0.00661   20.65371   0.00000
## waterfront0    22.58375   0.23028   98.07150   0.00000
## waterfront1    22.97175   0.23390   98.21038   0.00000
## view            0.03555   0.00434    8.19822   0.00000
## condition       0.04325   0.00448    9.64821   0.00000
## grade           0.21361   0.00408   52.39378   0.00000
## sqft_above     -0.00011   0.00001  -13.59060   0.00000
## yr_built       -0.00610   0.00012  -51.75937   0.00000
## yr_renovated    0.00000   0.00001   -0.26159   0.79364
## sqft_living15   0.00011   0.00001   17.17702   0.00000
## sqft_lot15      0.00000   0.00000   -4.10403   0.00004
##
## Call: rq(formula = log(price) ~ bedrooms + bathrooms + sqft_living +
##     sqft_lot + floors + waterfront + view + condition + grade +
##     sqft_above + yr_built + yr_renovated + sqft_living15 + sqft_lot15 +
##     0, tau = q, data = data_no)
##
## tau: [1] 0.6
##
## Coefficients:
##               Value     Std. Error t value   Pr(>|t|)
## bedrooms       -0.02705   0.00340   -7.95268   0.00000
## bathrooms       0.08244   0.00576   14.31946   0.00000
## sqft_living     0.00019   0.00001   24.16433   0.00000
## sqft_lot        0.00000   0.00000    2.70262   0.00688
```

```
## floors          0.11470   0.00611   18.77026   0.00000
## waterfront0    22.25755   0.21370  104.15236   0.00000
## waterfront1    22.63137   0.21614  104.70575   0.00000
## view            0.03409   0.00397    8.58374   0.00000
## condition       0.03668   0.00403    9.10101   0.00000
## grade           0.21278   0.00372   57.15017   0.00000
## sqft_above     -0.00010   0.00001  -12.61016   0.00000
## yr_built       -0.00586   0.00011  -54.10595   0.00000
## yr_renovated    0.00000   0.00001    0.66320   0.50721
## sqft_living15   0.00010   0.00001   16.29029   0.00000
## sqft_lot15      0.00000   0.00000   -5.06612   0.00000
##
## Call: rq(formula = log(price) ~ bedrooms + bathrooms + sqft_living +
##     sqft_lot + floors + waterfront + view + condition + grade +
##     sqft_above + yr_built + yr_renovated + sqft_living15 + sqft_lot15 +
##     0, tau = q, data = data_no)
##
## tau: [1] 0.7
##
## Coefficients:
##               Value     Std. Error t value   Pr(>|t|)
## bedrooms       -0.02062   0.00345   -5.97849   0.00000
## bathrooms       0.07083   0.00554   12.79695   0.00000
## sqft_living     0.00020   0.00001   26.72441   0.00000
## sqft_lot        0.00000   0.00000   14.88498   0.00000
## floors          0.10504   0.00565   18.59664   0.00000
## waterfront0    21.71025   0.20992  103.42144   0.00000
## waterfront1    22.10137   0.21517  102.71794   0.00000
## view            0.03996   0.00375   10.66720   0.00000
## condition       0.03487   0.00403    8.66247   0.00000
## grade           0.20808   0.00351   59.26061   0.00000
## sqft_above     -0.00009   0.00001  -12.90731   0.00000
## yr_built       -0.00552   0.00011  -51.44633   0.00000
## yr_renovated    0.00002   0.00001    3.13285   0.00173
## sqft_living15   0.00009   0.00001   15.66307   0.00000
## sqft_lot15      0.00000   0.00000   -9.16054   0.00000
##
## Call: rq(formula = log(price) ~ bedrooms + bathrooms + sqft_living +
##     sqft_lot + floors + waterfront + view + condition + grade +
```

```
##     sqft_above + yr_built + yr_renovated + sqft_living15 + sqft_lot15 +
##     0, tau = q, data = data_no)
##
## tau: [1] 0.8
##
## Coefficients:
##               Value     Std. Error t value   Pr(>|t|)
## bedrooms       -0.02299   0.00319   -7.21796   0.00000
## bathrooms       0.07405   0.00605   12.24228   0.00000
## sqft_living     0.00020   0.00001   24.79380   0.00000
## sqft_lot        0.00000   0.00000    5.19633   0.00000
## floors          0.08836   0.00633   13.94881   0.00000
## waterfront0    21.69907   0.23533   92.20620   0.00000
## waterfront1    22.07783   0.23694   93.17934   0.00000
## view            0.04643   0.00443   10.47875   0.00000
## condition       0.03291   0.00409    8.03758   0.00000
## grade           0.20045   0.00393   51.01694   0.00000
## sqft_above     -0.00007   0.00001   -9.17231   0.00000
## yr_built       -0.00544   0.00012  -45.15292   0.00000
## yr_renovated    0.00002   0.00001    2.64798   0.00810
## sqft_living15   0.00008   0.00001   12.94906   0.00000
## sqft_lot15      0.00000   0.00000   -8.42151   0.00000
##
## Call: rq(formula = log(price) ~ bedrooms + bathrooms + sqft_living +
##     sqft_lot + floors + waterfront + view + condition + grade +
##     sqft_above + yr_built + yr_renovated + sqft_living15 + sqft_lot15 +
##     0, tau = q, data = data_no)
##
## tau: [1] 0.9
##
## Coefficients:
##               Value     Std. Error t value   Pr(>|t|)
## bedrooms       -0.02635   0.00435   -6.06045   0.00000
## bathrooms       0.05829   0.00681    8.56300   0.00000
## sqft_living     0.00021   0.00001   20.89920   0.00000
## sqft_lot        0.00000   0.00000    3.72942   0.00019
## floors          0.06962   0.00647   10.76133   0.00000
## waterfront0    21.68966   0.24275   89.35074   0.00000
## waterfront1    22.05254   0.24464   90.14364   0.00000
```

```
## view            0.04349   0.00472    9.21010   0.00000
## condition       0.02098   0.00454    4.61770   0.00000
## grade           0.19456   0.00475   40.98195   0.00000
## sqft_above     -0.00004   0.00001   -4.45489   0.00001
## yr_built       -0.00534   0.00012  -43.11337   0.00000
## yr_renovated    0.00003   0.00001    3.52866   0.00042
## sqft_living15   0.00009   0.00001   11.31905   0.00000
## sqft_lot15      0.00000   0.00000   -8.74695   0.00000
```

```
(coef(quant_reg))
```

```
##                    tau= 0.5      tau= 0.6      tau= 0.7      tau= 0.8
## bedrooms      -2.663374e-02 -2.705325e-02 -2.061549e-02 -2.299131e-02
## bathrooms      8.562012e-02  8.243888e-02  7.083195e-02  7.405117e-02
## sqft_living    1.917111e-04  1.939939e-04  1.981967e-04  2.011387e-04
## sqft_lot       2.828902e-07  3.263768e-07  4.186836e-07  3.541776e-07
## floors         1.364855e-01  1.146951e-01  1.050371e-01  8.836091e-02
## waterfront0    2.258375e+01  2.225755e+01  2.171025e+01  2.169907e+01
## waterfront1    2.297175e+01  2.263137e+01  2.210137e+01  2.207783e+01
## view           3.554883e-02  3.409030e-02  3.995719e-02  4.643100e-02
## condition      4.324892e-02  3.668106e-02  3.486664e-02  3.291090e-02
## grade          2.136080e-01  2.127811e-01  2.080830e-01  2.004518e-01
## sqft_above    -1.133247e-04 -9.763107e-05 -9.290394e-05 -7.276540e-05
## yr_built      -6.104308e-03 -5.861562e-03 -5.517590e-03 -5.436200e-03
## yr_renovated  -1.767474e-06  4.757526e-06  1.962743e-05  2.317524e-05
## sqft_living15  1.133608e-04  9.593063e-05  8.856437e-05  8.159872e-05
## sqft_lot15    -5.086115e-07 -6.614035e-07 -7.375552e-07 -7.224849e-07
##                    tau= 0.9
## bedrooms      -2.635171e-02
## bathrooms      5.828906e-02
## sqft_living    2.147960e-04
## sqft_lot       3.198705e-07
## floors         6.961504e-02
## waterfront0    2.168966e+01
## waterfront1    2.205254e+01
## view           4.348736e-02
## condition      2.097946e-02
## grade          1.945570e-01
## sqft_above    -4.452834e-05
## yr_built      -5.340409e-03
## yr_renovated   2.984237e-05
## sqft_living15  9.377651e-05
## sqft_lot15    -8.719731e-07
```