# housing_data

Stephen Smitshoek

06/05/2022

```
## Warning: package ’readxl’ was built under R version 4.1.3
```

```
##
## Call:
## lm(formula = sale_price ~ sq_ft_lot, data = housing_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2016064  -194842   -63293    91565  3735109
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 6.418e+05  3.800e+03  168.90   <2e-16 ***
## sq_ft_lot   8.510e-01  6.217e-02   13.69   <2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 401500 on 12863 degrees of freedom
## Multiple R-squared:  0.01435,    Adjusted R-squared:  0.01428
## F-statistic: 187.3 on 1 and 12863 DF,  p-value: < 2.2e-16
```

```
##
## Call:
## lm(formula = sale_price ~ sq_ft_lot + year_built + sale_year +
##     bath_total + bedrooms, data = housing_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3289836  -143540   -46434    61109  3681894
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.881e+07  2.068e+06  -9.096  < 2e-16 ***
## sq_ft_lot    8.339e-01  5.873e-02  14.200  < 2e-16 ***
## year_built   3.986e+03  2.168e+02  18.390  < 2e-16 ***
## sale_year    5.483e+03  1.003e+03   5.468 4.63e-08 ***
## bath_total   1.310e+05  6.245e+03  20.984  < 2e-16 ***
## bedrooms     4.543e+04  4.512e+03  10.069  < 2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 370800 on 12859 degrees of freedom
## Multiple R-squared:  0.1597, Adjusted R-squared:  0.1593
## F-statistic: 488.6 on 5 and 12859 DF,  p-value: < 2.2e-16
```

## Sale Price vs. Square Foot Lot

- Rˆ2 = 0.01435
- Adjusted R2 = 0.01428 These R2 values suggest that the square footage of a lot accounts for approximately 1.4% of the variation in sales price.

## Sale Price vs. Square Foot Lot + Year Built + Sale Year + Bathrooms + Bedrooms

- Rˆ2 = 0.1597
- Adjusted R2 = 0.1593

These R2 values suggest that the predictors used account for approximately 16% of the variation in sales price. This is a substantial increase over the 1.4% predicted by just the square footage and indicates that this model is a better fit for predicting the sale price of a home.

## Beta Values

```
##
## Call:
## lm(formula = sale_price ~ sq_ft_lot + year_built + sale_year +
##     bath_total + bedrooms, data = housing_data)
##
## Standardized Coefficients::
## (Intercept)   sq_ft_lot  year_built   sale_year  bath_total    bedrooms
##  0.00000000  0.11741061  0.16976054  0.04426981  0.22528554  0.09843756
```

- sq_ft_lot = 0.12
- year_built = 0.17
- sale_year = 0.04
- bath_total = 0.23
- bedrooms = 0.10

These standardized beta values represent the impact that a one standard deviation change in the respective variable will have on the sale price. For example an increase of one standard deviation in square footage will increase sale price by 0.11 standard deviations. If everything else is held constant.

The standardized beta values also show the relative importance of each variable. Square footage and number of bedrooms can be said to have a similar impact, where number of bathrooms has a siginifcantly larger impact on the sale price.

## Confidence Intervals

```
##                     2.5 %        97.5 %
## (Intercept) -2.286640e+07 -1.475864e+07
## sq_ft_lot    7.188174e-01  9.490514e-01
## year_built   3.561580e+03  4.411391e+03
## sale_year    3.517740e+03  7.448835e+03
## bath_total   1.188039e+05  1.432862e+05
## bedrooms     3.658943e+04  5.427930e+04
```

These confidence intervals show the range of where the possible beta values may lay with a 95% confidence level. The smaller the range the more likely our models beta value is representative of the greater population. The fact that none of the confidence intervals cross zero indicates that each factor listed is significant.

## Analysis of Variance

```
## Analysis of Variance Table
##
## Model 1: sale_price ~ sq_ft_lot
## Model 2: sale_price ~ sq_ft_lot + year_built + sale_year + bath_total +
##     bedrooms
##   Res.Df        RSS Df  Sum of Sq      F    Pr(>F)
## 1  12863 2.0734e+15
## 2  12859 1.7677e+15  4 3.0566e+14 555.88 < 2.2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

The additional variable made a significant increase in the models accuracy since there is an F ratio of 556 with a P < 0.001.

## Outliers

## Standardized Residuals

## Number of Large Residuals

```
## [1] 345
```

## Variables with Large Residuals

```
## # A tibble: 345 x 28
##    sale_date  sale_price sale_reason sale_instrument sale_warning sitetype
##    <date>          <dbl>       <dbl>           <dbl> <chr>        <chr>
##  1 2006-01-11     265000           1               3 <NA>         R1
##  2 2006-02-01    1900000           1               3 15 52        R1
##  3 2006-02-13    1520000          18               3 52           R1
##  4 2006-02-15    1390000           1               3 <NA>         R1
##  5 2006-03-20    1588359           1               3 <NA>         R1
##  6 2006-03-21    1450000           1               3 <NA>         R1
##  7 2006-03-21    1450000           1               3 <NA>         R1
##  8 2006-03-27     163000           1               3 49           R3
##  9 2006-03-28     270000           1               3 <NA>         R1
## 10 2006-03-29     200000           1               3 <NA>         R1
## # ... with 335 more rows, and 22 more variables: addr_full <chr>, zip5 <dbl>,
## #   ctyname <chr>, postalctyn <chr>, lon <dbl>, lat <dbl>,
## #   building_grade <dbl>, square_feet_total_living <dbl>, bedrooms <dbl>,
## #   bath_full_count <dbl>, bath_half_count <dbl>, bath_3qtr_count <dbl>,
## #   year_built <dbl>, year_renovated <dbl>, current_zoning <chr>,
## #   sq_ft_lot <dbl>, prop_type <chr>, present_use <dbl>, sale_year <dbl>,
## #   bath_total <dbl>, residuals <dbl>, std.resid <lgl>
```

## Leverage, Cooks Distance, and Covariance Ratios

```
## # A tibble: 298 x 5
##    sale_date  std.resid    cooks leverage cov.ratio
##    <date>     <lgl>        <dbl>    <dbl>     <dbl>
##  1 2006-02-01 TRUE      0.00127  0.00103      0.998
##  2 2006-02-13 TRUE      0.00134  0.00195      1.00
##  3 2006-02-15 TRUE      0.00405  0.00279      0.999
##  4 2006-03-20 TRUE      0.000657 0.000561     0.998
##  5 2006-03-21 TRUE      0.000520 0.000500     0.998
##  6 2006-03-21 TRUE      0.00469  0.00201      0.996
##  7 2006-03-28 TRUE      1.99     0.118        1.09
##  8 2006-03-29 TRUE      0.00337  0.00252      0.999
##  9 2006-04-06 TRUE      0.00912  0.00861      1.01
## 10 2006-04-06 TRUE      0.00912  0.00861      1.01
## # ... with 288 more rows
```

```
## # A tibble: 8 x 31
##   sale_date  sale_price sale_reason sale_instrument sale_warning sitetype
##   <date>          <dbl>       <dbl>           <dbl> <chr>        <chr>
## 1 2006-03-28     855990           1               3 <NA>         R1
## 2 2006-03-28     832950           1               3 <NA>         R1
## 3 2006-03-28     632900           1               3 <NA>         R1
## 4 2006-03-28     624900           1               3 <NA>         R1
## 5 2006-03-28     575000           1               3 <NA>         R1
## 6 2006-03-28     378000           1               3 <NA>         R1
## 7 2006-03-28     295000           1               3 <NA>         R1
## 8 2006-03-28     270000           1               3 <NA>         R1
## # ... with 25 more variables: addr_full <chr>, zip5 <dbl>, ctyname <chr>,
## #   postalctyn <chr>, lon <dbl>, lat <dbl>, building_grade <dbl>,
## #   square_feet_total_living <dbl>, bedrooms <dbl>, bath_full_count <dbl>,
## #   bath_half_count <dbl>, bath_3qtr_count <dbl>, year_built <dbl>,
## #   year_renovated <dbl>, current_zoning <chr>, sq_ft_lot <dbl>,
## #   prop_type <chr>, present_use <dbl>, sale_year <dbl>, bath_total <dbl>,
## #   residuals <dbl>, std.resid <lgl>, leverage <dbl>, cooks <dbl>, ...
```

There are 298 rows which could be classified as outliers however all except one have a cooks distance < 1 so they are likely not influencing the model. The one row which is providing influence and is an outlier is a home sold on appears to have sold for $270,000 but has 23 bathrooms and has a lot size of 89734 square feet.

## Assumption of Independence

```
## Warning: package 'car' was built under R version 4.1.3
```

```
## Loading required package: carData
```

```
## Warning: package 'carData' was built under R version 4.1.3
##  lag Autocorrelation D-W Statistic p-value
##    1       0.6632046     0.6735824       0
##  Alternative hypothesis: rho != 0
```

Because the D-W Statistic is less than 1 and the p value is less than 0.05 it can be assumed that there is autocorrelation in the variables in the model.

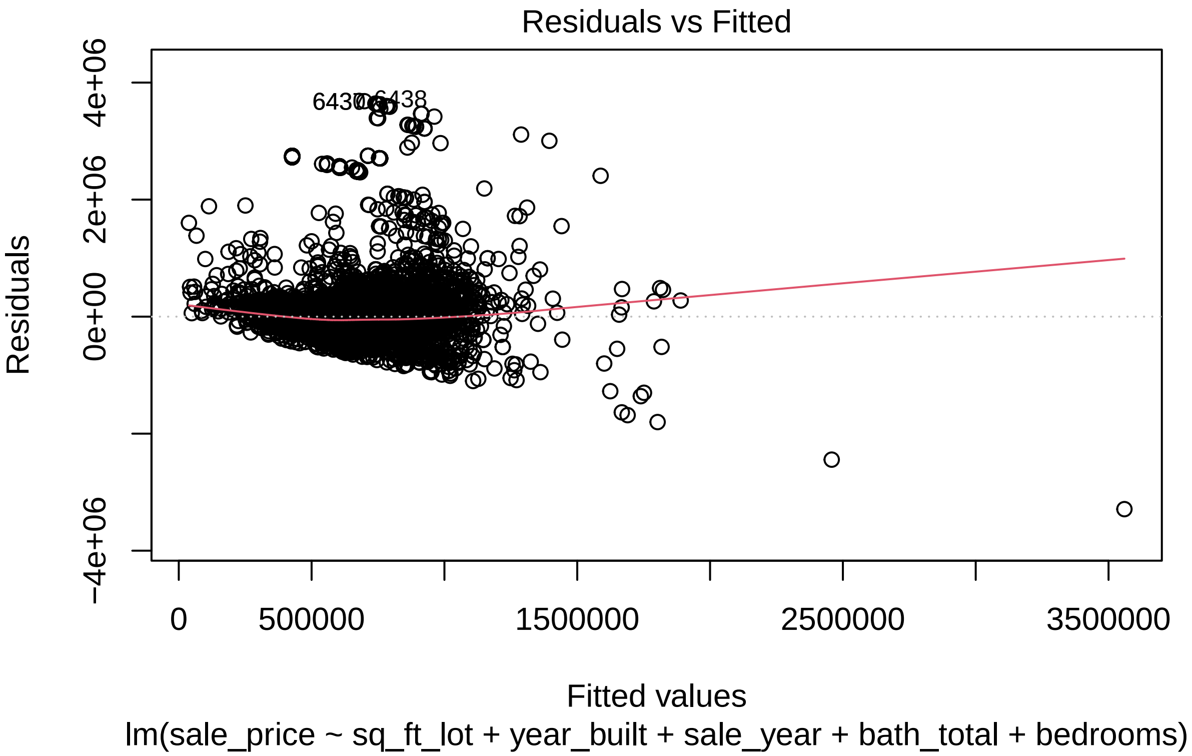
## Assumption of No Multicollinearity

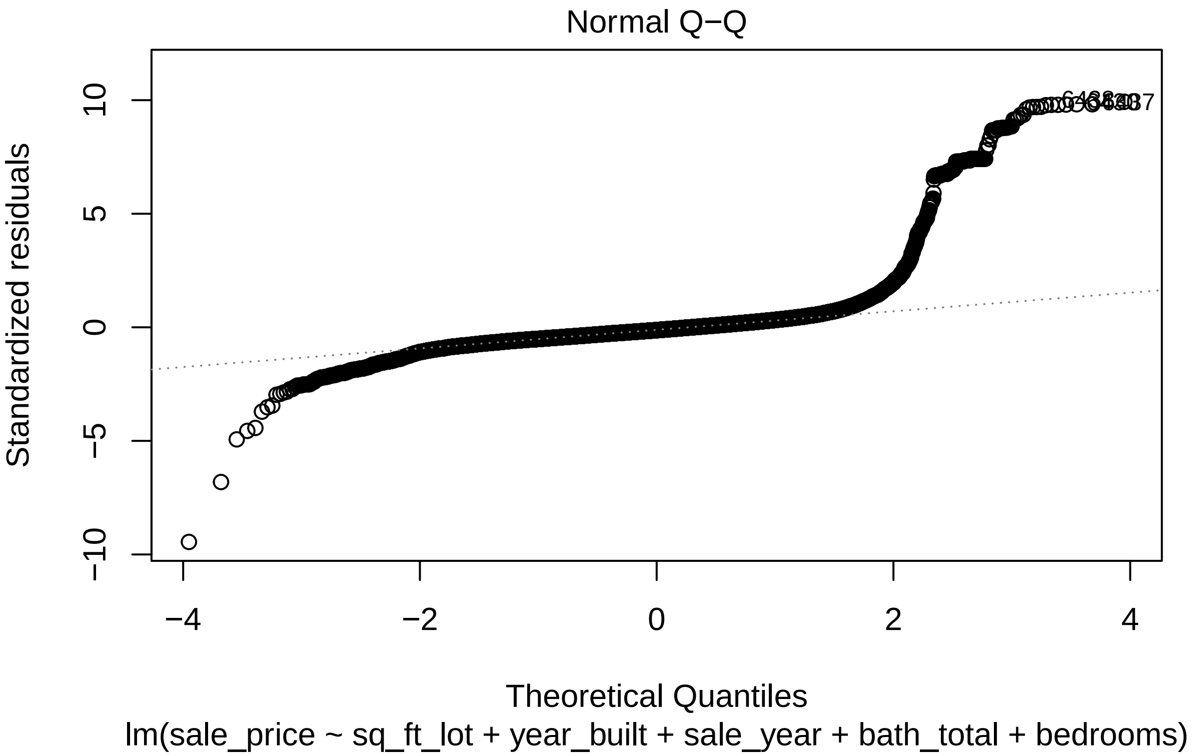
```
##  sq_ft_lot year_built  sale_year bath_total   bedrooms
##   1.046177   1.303929   1.002941   1.763784   1.462572
```

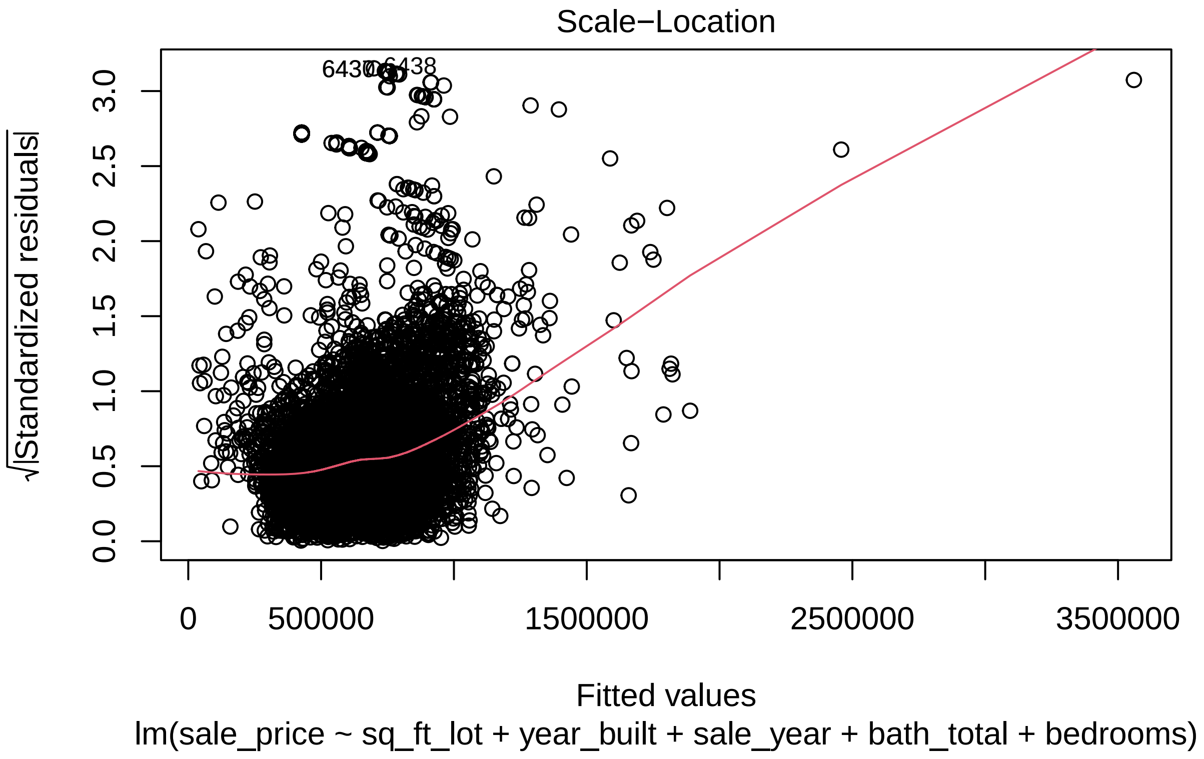
```
## [1] 1.315881
```

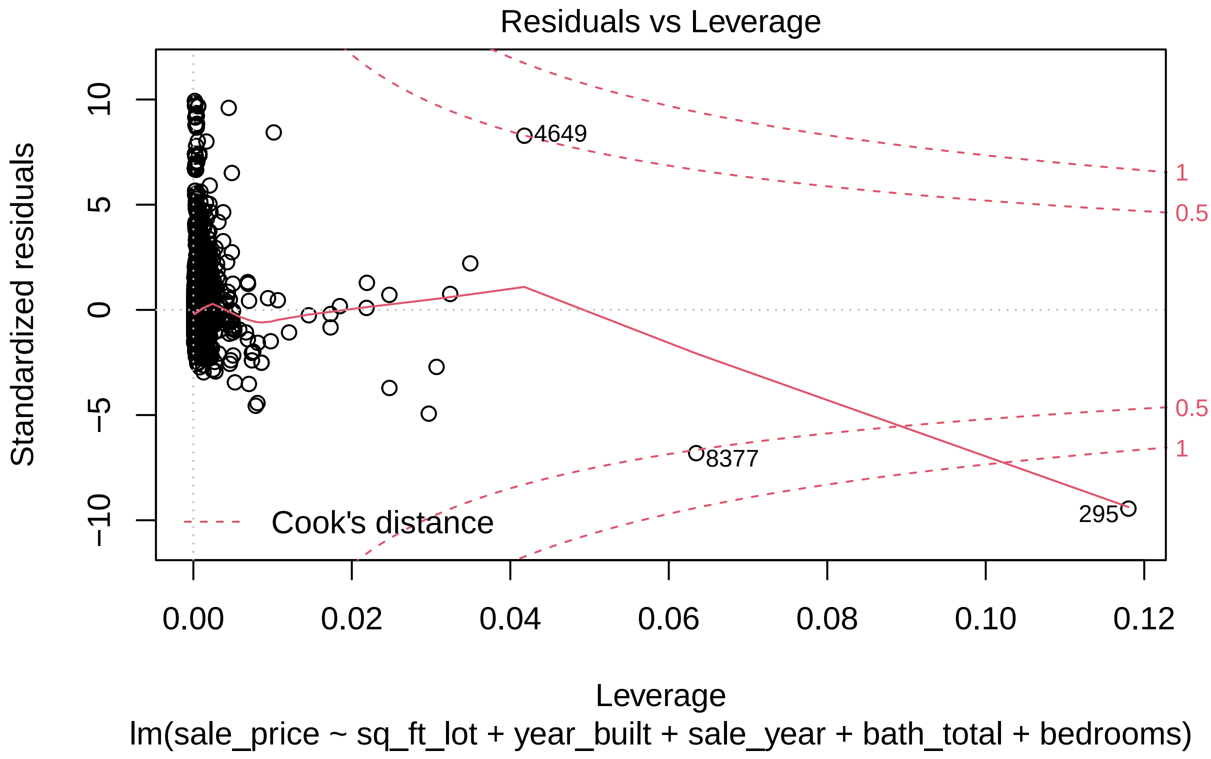
```
##  sq_ft_lot year_built  sale_year bath_total   bedrooms
##  0.9558616  0.7669131  0.9970672  0.5669628  0.6837269
```

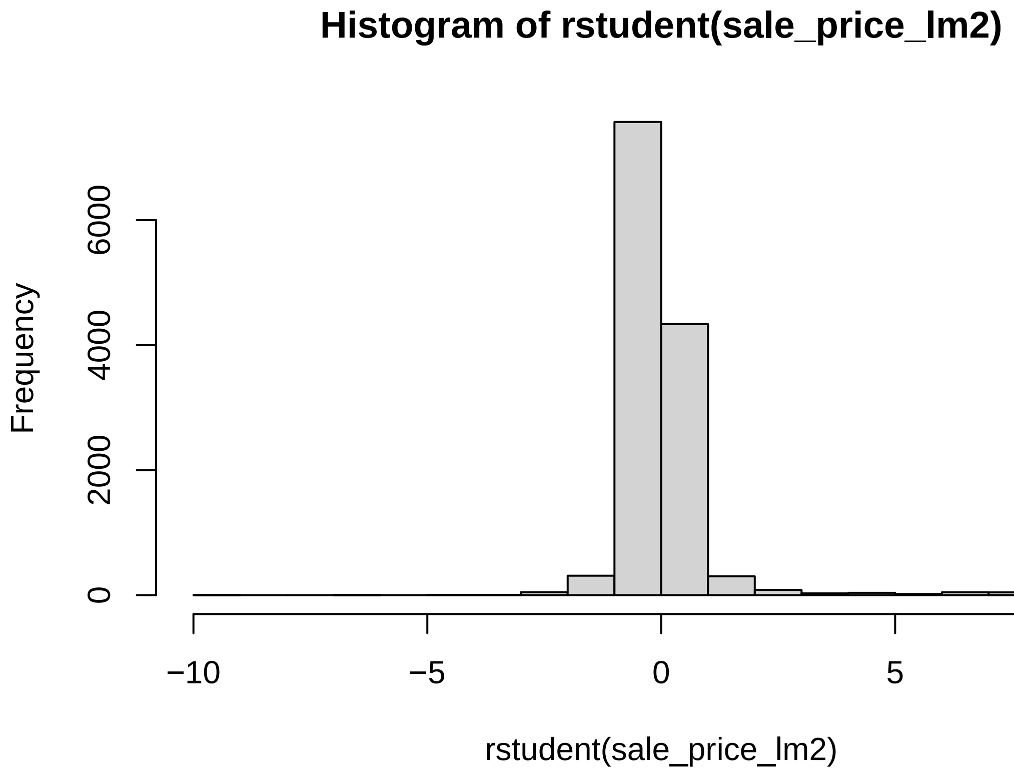
The largest VIF is not greater than 10 and the average VIF is not substantially greater than 1. Additionally none of the tolerances are less than 0.2. With all of this information it can be concluded there is no multicollinearity between the variables used in the model.


Residuals vs Fitted
Residuals
4e+06
2e+06
0e+00
−4e+06
6437
6438
0
500000
1500000
2500000
3500000
Fitted values
lm(sale_price ~ sq_ft_lot + year_built + sale_year + bath_total + bedrooms)


Normal Q−Q
Standardized residuals
10
5
0
−5
−10
6438
6437
−4
−2
0
2
4
Theoretical Quantiles
lm(sale_price ~ sq_ft_lot + year_built + sale_year + bath_total + bedrooms)


Scale−Location
6437
6438
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
2.5
3.0
0
500000
1500000
2500000
3500000
Fitted values
lm(sale_price ~ sq_ft_lot + year_built + sale_year + bath_total + bedrooms)


Residuals vs Leverage
Standardized residuals
10
5
0
−5
−10
0.00
0.02
0.04
0.06
0.08
0.10
0.12
Leverage
lm(sale_price ~ sq_ft_lot + year_built + sale_year + bath_total + bedrooms)
Cook's distance
4649
8377
295
1
0.5
0.5
1



- Residuals vs Fitted
  - For the most part the show a normal distribution, it is possible there is some heteroscedasticity as there are a number of residuals over 3E6 but only for fitted values between 500,000 and 1,000,000.
- Normal Q-Q
  - This plot suggests we have leptokurtic kurtosis and so the data is not normally distributed
- Histogram Studentized Residuals
  - The histogram of the studentized residuals shows a very strong leptokurtic kurtosis and agrees with the Q-Q plot that the data is not normally distributed.

# Code

```
# Assignment: Housing Data
# Name: Smitshoek, Stephen
# Date: 2022-05-06

library(readxl)
library(ggplot2)
setwd("C:\\Users\\sksmi\\PeytoAccess\\Personal\\Bellevue\\DSC520\\dsc520")
housing_data <- read_xlsx("data\\week-7-housing.xlsx")
colnames(housing_data)[1:2] <- c("sale_date", "sale_price")

housing_data$sale_year <- as.numeric(format(housing_data$sale_date, format =
"%Y"))
housing_data$sale_date <- as.Date(housing_data$sale_date)
housing_data$bath_total <- with(housing_data,
                                bath_full_count +
                                  bath_half_count * .5 +
                                  bath_3qtr_count * .75)

sale_price_lm1 <- lm(sale_price ~ sq_ft_lot,
                     data=housing_data)

sale_price_lm2 <- lm(sale_price ~ sq_ft_lot + year_built + sale_year +
                       bath_total + bedrooms,
                     data=housing_data)

summary(sale_price_lm1)
summary(sale_price_lm2)

library(lm.beta)
lm.beta(sale_price_lm2)

confint(sale_price_lm2, level=.95)

anova(sale_price_lm1, sale_price_lm2)

housing_data$residuals <- resid(sale_price_lm2)

housing_data$std.resid <- rstandard(sale_price_lm2) > 2 |
rstandard(sale_price_lm2) < -2

sum(housing_data$std.resid)

housing_data[housing_data$std.resid,]

housing_data$leverage <- hatvalues(sale_price_lm2)
housing_data$cooks <- cooks.distance(sale_price_lm2)
housing_data$cov.ratio <- covratio(sale_price_lm2)

avg.lvg <- (5 + 1) / 12865
cvr.pos <- 1 + (3 * (5+1) / 12865)
cvr.neg <- 1 - (3 * (5+1) / 12865)

subset(housing_data, std.resid & (cooks > 1 | leverage > avg.lvg * 3 |
cov.ratio > cvr.pos | cov.ratio < cvr.neg))[,c("sale_date", "std.resid",
"cooks", "leverage", "cov.ratio")]
```

```
housing_data[housing_data$sale_date == "2006-03-28",]

library(car)
durbinWatsonTest(sale_price_lm2)

vif(sale_price_lm2)
mean(vif(sale_price_lm2))
1/vif(sale_price_lm2)

plot(sale_price_lm2)
hist(rstudent(sale_price_lm2))
```