```r
# Math 564 project
library(ggplot2)
library(reshape2)
library(MASS)
```

```
## Warning: package 'MASS' was built under R version 3.4.4
```

```r
library(leaps)
```

```
## Warning: package 'leaps' was built under R version 3.4.4
```

```r
################################################
# helper functions
# get lower triangle of the matrix
get_lower_tri<-function(cormat){
    cormat[upper.tri(cormat)] <- NA
    return(cormat)
}

# reorder the correlation values
reorder_cormat <- function(cormat){
    # Use correlation between variables as distance
    dd <- as.dist((1-cormat)/2)
    hc <- hclust(dd)
    cormat <-cormat[hc$order, hc$order]
}

# select best model by adj R^2
best_adjr2 <- function(model,...)
{
    subsets <- regsubsets(formula(model), model.frame(model), ...)
    subsets <- with(summary(subsets),
                    cbind(p = as.numeric(rownames(which)), which, adjr2))

    return(subsets)
}
# select best model by mallow Cp
best_cp <- function(model,...)
{
    subsets <- regsubsets(formula(model), model.frame(model), ...)
    subsets <- with(summary(subsets),
                    cbind(p = as.numeric(rownames(which)), which, cp))

    return(subsets)
}
################################################


##set up
# setwd("C:/Users/Jin/Desktop/MATH564Project")
myD<-read.csv("./kc_house_data.csv",stringsAsFactors = FALSE)
summary(myD)
```

```
##        id                 date               price            bedrooms
##  Min.   :1.000e+06   Length:21613       Min.   :  75000   Min.   : 0.000
```

```
##  1st Qu.:2.123e+09   Class :character   1st Qu.: 321950   1st Qu.: 3.000
##  Median :3.905e+09   Mode  :character   Median : 450000   Median : 3.000
##  Mean   :4.580e+09                      Mean   : 540088   Mean   : 3.371
##  3rd Qu.:7.309e+09                      3rd Qu.: 645000   3rd Qu.: 4.000
##  Max.   :9.900e+09                      Max.   :7700000   Max.   :33.000
##    bathrooms      sqft_living       sqft_lot           floors
##  Min.   :0.000   Min.   :  290   Min.   :    520   Min.   :1.000
##  1st Qu.:1.750   1st Qu.: 1427   1st Qu.:   5040   1st Qu.:1.000
##  Median :2.250   Median : 1910   Median :   7618   Median :1.500
##  Mean   :2.115   Mean   : 2080   Mean   :  15107   Mean   :1.494
##  3rd Qu.:2.500   3rd Qu.: 2550   3rd Qu.:  10688   3rd Qu.:2.000
##  Max.   :8.000   Max.   :13540   Max.   :1651359   Max.   :3.500
##    waterfront           view          condition         grade
##  Min.   :0.000000   Min.   :0.0000   Min.   :1.000   Min.   : 1.000
##  1st Qu.:0.000000   1st Qu.:0.0000   1st Qu.:3.000   1st Qu.: 7.000
##  Median :0.000000   Median :0.0000   Median :3.000   Median : 7.000
##  Mean   :0.007542   Mean   :0.2343   Mean   :3.409   Mean   : 7.657
##  3rd Qu.:0.000000   3rd Qu.:0.0000   3rd Qu.:4.000   3rd Qu.: 8.000
##  Max.   :1.000000   Max.   :4.0000   Max.   :5.000   Max.   :13.000
##    sqft_above   sqft_basement       yr_built     yr_renovated
##  Min.   : 290   Min.   :   0.0   Min.   :1900   Min.   :   0.0
##  1st Qu.:1190   1st Qu.:   0.0   1st Qu.:1951   1st Qu.:   0.0
##  Median :1560   Median :   0.0   Median :1975   Median :   0.0
##  Mean   :1788   Mean   : 291.5   Mean   :1971   Mean   :  84.4
##  3rd Qu.:2210   3rd Qu.: 560.0   3rd Qu.:1997   3rd Qu.:   0.0
##  Max.   :9410   Max.   :4820.0   Max.   :2015   Max.   :2015.0
##     zipcode           lat             long        sqft_living15
##  Min.   :98001   Min.   :47.16   Min.   :-122.5   Min.   : 399
##  1st Qu.:98033   1st Qu.:47.47   1st Qu.:-122.3   1st Qu.:1490
##  Median :98065   Median :47.57   Median :-122.2   Median :1840
##  Mean   :98078   Mean   :47.56   Mean   :-122.2   Mean   :1987
##  3rd Qu.:98118   3rd Qu.:47.68   3rd Qu.:-122.1   3rd Qu.:2360
##  Max.   :98199   Max.   :47.78   Max.   :-121.3   Max.   :6210
##    sqft_lot15
##  Min.   :   651
##  1st Qu.:  5100
##  Median :  7620
##  Mean   : 12768
##  3rd Qu.: 10083
##  Max.   :871200
```

```
str(myD)
```

```
## 'data.frame':    21613 obs. of  21 variables:
##  $ id           : num  7.13e+09 6.41e+09 5.63e+09 2.49e+09 1.95e+09 ...
##  $ date         : chr  "20141013T000000" "20141209T000000" "20150225T000000" "20141209T000000" ...
##  $ 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 ...
##  $ waterfront   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ 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 ...
##  $ zipcode      : int  98178 98125 98028 98136 98074 98053 98003 98198 98146 98038 ...
##  $ lat          : num  47.5 47.7 47.7 47.5 47.6 ...
##  $ long         : num  -122 -122 -122 -122 -122 ...
##  $ 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 ...
```

```
sum(is.na(myD))
```

```
## [1] 0
```

```
#the data is pretty clean. No NAs and most of them are int/num.
#we can just drop the first two columns, id and dates.
myD <- myD[, -c(1,2)]

##correlations
# pair matrix without dates
cormat <- cor(myD)
round(cormat, 2)
```

```
##               price bedrooms bathrooms sqft_living sqft_lot floors
## price          1.00     0.31      0.53        0.70     0.09   0.26
## bedrooms       0.31     1.00      0.52        0.58     0.03   0.18
## bathrooms      0.53     0.52      1.00        0.75     0.09   0.50
## sqft_living    0.70     0.58      0.75        1.00     0.17   0.35
## sqft_lot       0.09     0.03      0.09        0.17     1.00  -0.01
## floors         0.26     0.18      0.50        0.35    -0.01   1.00
## waterfront     0.27    -0.01      0.06        0.10     0.02   0.02
## view           0.40     0.08      0.19        0.28     0.07   0.03
## condition      0.04     0.03     -0.12       -0.06    -0.01  -0.26
## grade          0.67     0.36      0.66        0.76     0.11   0.46
## sqft_above     0.61     0.48      0.69        0.88     0.18   0.52
## sqft_basement  0.32     0.30      0.28        0.44     0.02  -0.25
## yr_built       0.05     0.15      0.51        0.32     0.05   0.49
## yr_renovated   0.13     0.02      0.05        0.06     0.01   0.01
## zipcode       -0.05    -0.15     -0.20       -0.20    -0.13  -0.06
## lat            0.31    -0.01      0.02        0.05    -0.09   0.05
## long           0.02     0.13      0.22        0.24     0.23   0.13
## sqft_living15  0.59     0.39      0.57        0.76     0.14   0.28
## sqft_lot15     0.08     0.03      0.09        0.18     0.72  -0.01
##               waterfront  view condition grade sqft_above sqft_basement
## price               0.27  0.40      0.04  0.67       0.61          0.32
## bedrooms           -0.01  0.08      0.03  0.36       0.48          0.30
## bathrooms           0.06  0.19     -0.12  0.66       0.69          0.28
## sqft_living         0.10  0.28     -0.06  0.76       0.88          0.44
## sqft_lot            0.02  0.07     -0.01  0.11       0.18          0.02
## floors              0.02  0.03     -0.26  0.46       0.52         -0.25
## waterfront          1.00  0.40      0.02  0.08       0.07          0.08
## view                0.40  1.00      0.05  0.25       0.17          0.28
## condition           0.02  0.05      1.00 -0.14      -0.16          0.17
## grade               0.08  0.25     -0.14  1.00       0.76          0.17
```

```
## sqft_above          0.07  0.17     -0.16  0.76       1.00         -0.05
## sqft_basement       0.08  0.28      0.17  0.17      -0.05          1.00
## yr_built           -0.03 -0.05     -0.36  0.45       0.42         -0.13
## yr_renovated        0.09  0.10     -0.06  0.01       0.02          0.07
## zipcode             0.03  0.08      0.00 -0.18      -0.26          0.07
## lat                -0.01  0.01     -0.01  0.11       0.00          0.11
## long               -0.04 -0.08     -0.11  0.20       0.34         -0.14
## sqft_living15       0.09  0.28     -0.09  0.71       0.73          0.20
## sqft_lot15          0.03  0.07      0.00  0.12       0.19          0.02
##               yr_built yr_renovated zipcode   lat  long sqft_living15
## price             0.05         0.13   -0.05  0.31  0.02          0.59
## bedrooms          0.15         0.02   -0.15 -0.01  0.13          0.39
## bathrooms         0.51         0.05   -0.20  0.02  0.22          0.57
## sqft_living       0.32         0.06   -0.20  0.05  0.24          0.76
## sqft_lot          0.05         0.01   -0.13 -0.09  0.23          0.14
## floors            0.49         0.01   -0.06  0.05  0.13          0.28
## waterfront       -0.03         0.09    0.03 -0.01 -0.04          0.09
## view             -0.05         0.10    0.08  0.01 -0.08          0.28
## condition        -0.36        -0.06    0.00 -0.01 -0.11         -0.09
## grade             0.45         0.01   -0.18  0.11  0.20          0.71
## sqft_above        0.42         0.02   -0.26  0.00  0.34          0.73
## sqft_basement    -0.13         0.07    0.07  0.11 -0.14          0.20
## yr_built          1.00        -0.22   -0.35 -0.15  0.41          0.33
## yr_renovated     -0.22         1.00    0.06  0.03 -0.07          0.00
## zipcode          -0.35         0.06    1.00  0.27 -0.56         -0.28
## lat              -0.15         0.03    0.27  1.00 -0.14          0.05
## long              0.41        -0.07   -0.56 -0.14  1.00          0.33
## sqft_living15     0.33         0.00   -0.28  0.05  0.33          1.00
## sqft_lot15        0.07         0.01   -0.15 -0.09  0.25          0.18
##               sqft_lot15
## price               0.08
## bedrooms            0.03
## bathrooms           0.09
## sqft_living         0.18
## sqft_lot            0.72
## floors             -0.01
## waterfront          0.03
## view                0.07
## condition           0.00
## grade               0.12
## sqft_above          0.19
## sqft_basement       0.02
## yr_built            0.07
## yr_renovated        0.01
## zipcode            -0.15
## lat                -0.09
## long                0.25
## sqft_living15       0.18
## sqft_lot15          1.00
```

```
melted_cormat <- melt(cormat)
cormat <- reorder_cormat(cormat)
upper_tri <- get_lower_tri(cormat)
melted_cormat <- melt(upper_tri, na.rm = TRUE)
```

```r
# create correlation heatmap
ggheatmap <- ggplot(data = melted_cormat, aes(Var2, Var1, fill = value))+
    geom_tile(color = "white")+
    scale_fill_gradient2(low = "blue", high = "red", mid = "white",
                         midpoint = 0, limit = c(-1,1), space = "Lab",
                         name="Pearson\nCorrelation") +
    theme_minimal()+
    theme(axis.text.x = element_text(angle = 45, vjust = 1,
                                     size = 12, hjust = 1))+
    coord_fixed()


# sqft_living, sqft_above and grades are highly related.
# price has high correlation with sqft_living, grade, sqft_above, sqft_living15, and bathrooms.
melted_cormat_byValue <- melted_cormat[order(abs(melted_cormat$value), decreasing = T),]
melted_cormat_byValue_price <- melted_cormat_byValue[melted_cormat_byValue$Var1=='price' |
                                                       melted_cormat_byValue$Var2=='price',]




## full linear model
full.model <- lm(price~., data = myD)
summary(full.model)
```

```
##
## Call:
## lm(formula = price ~ ., data = myD)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1291725   -99229    -9739    77583  4333222
##
## Coefficients: (1 not defined because of singularities)
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    6.690e+06  2.931e+06   2.282  0.02249 *
## bedrooms      -3.577e+04  1.892e+03 -18.906  < 2e-16 ***
## bathrooms      4.114e+04  3.254e+03  12.645  < 2e-16 ***
## sqft_living    1.501e+02  4.385e+00  34.227  < 2e-16 ***
## sqft_lot       1.286e-01  4.792e-02   2.683  0.00729 **
## floors         6.690e+03  3.596e+03   1.860  0.06285 .
## waterfront     5.830e+05  1.736e+04  33.580  < 2e-16 ***
## view           5.287e+04  2.140e+03  24.705  < 2e-16 ***
## condition      2.639e+04  2.351e+03  11.221  < 2e-16 ***
## grade          9.589e+04  2.153e+03  44.542  < 2e-16 ***
## sqft_above     3.113e+01  4.360e+00   7.139 9.71e-13 ***
## sqft_basement         NA         NA      NA       NA
## yr_built      -2.620e+03  7.266e+01 -36.062  < 2e-16 ***
## yr_renovated   1.981e+01  3.656e+00   5.420 6.03e-08 ***
## zipcode       -5.824e+02  3.299e+01 -17.657  < 2e-16 ***
## lat            6.027e+05  1.073e+04  56.149  < 2e-16 ***
## long          -2.147e+05  1.313e+04 -16.349  < 2e-16 ***
## sqft_living15  2.168e+01  3.448e+00   6.289 3.26e-10 ***
```

```
## sqft_lot15    -3.826e-01  7.327e-02  -5.222 1.78e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 201200 on 21595 degrees of freedom
## Multiple R-squared:  0.6997, Adjusted R-squared:  0.6995
## F-statistic:  2960 on 17 and 21595 DF,  p-value: < 2.2e-16
```

```
# coef of sqft_basement is NA, and p-value of floor > 0.05
# R^2 = 0.6997, adR^2 = 0.6995
fit_drop_basement_floors <- lm(price~.-sqft_basement-floors, data = myD)
summary(fit_drop_basement_floors)
```

```
##
## Call:
## lm(formula = price ~ . - sqft_basement - floors, data = myD)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1292272   -99268    -9849    77605  4331513
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    5.741e+06  2.887e+06   1.989  0.04674 *
## bedrooms      -3.586e+04  1.891e+03 -18.962  < 2e-16 ***
## bathrooms      4.272e+04  3.142e+03  13.596  < 2e-16 ***
## sqft_living    1.476e+02  4.175e+00  35.352  < 2e-16 ***
## sqft_lot       1.266e-01  4.791e-02   2.643  0.00822 **
## waterfront     5.831e+05  1.736e+04  33.585  < 2e-16 ***
## view           5.297e+04  2.140e+03  24.756  < 2e-16 ***
## condition      2.614e+04  2.348e+03  11.133  < 2e-16 ***
## grade          9.624e+04  2.145e+03  44.878  < 2e-16 ***
## sqft_above     3.472e+01  3.910e+00   8.878  < 2e-16 ***
## yr_built      -2.591e+03  7.092e+01 -36.531  < 2e-16 ***
## yr_renovated   2.017e+01  3.651e+00   5.526 3.32e-08 ***
## zipcode       -5.767e+02  3.284e+01 -17.559  < 2e-16 ***
## lat            6.044e+05  1.070e+04  56.494  < 2e-16 ***
## long          -2.168e+05  1.309e+04 -16.568  < 2e-16 ***
## sqft_living15  2.097e+01  3.426e+00   6.119 9.57e-10 ***
## sqft_lot15    -3.874e-01  7.323e-02  -5.291 1.23e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 201300 on 21596 degrees of freedom
## Multiple R-squared:  0.6997, Adjusted R-squared:  0.6995
## F-statistic:  3145 on 16 and 21596 DF,  p-value: < 2.2e-16
```

```
# R^2 = 0.6997, adR^2 = 0.6995

## simplified model
sim_fit <- lm(price~sqft_living+grade+sqft_above+sqft_living15+bathrooms, data = myD)
summary(sim_fit)
```

```
##
## Call:
```

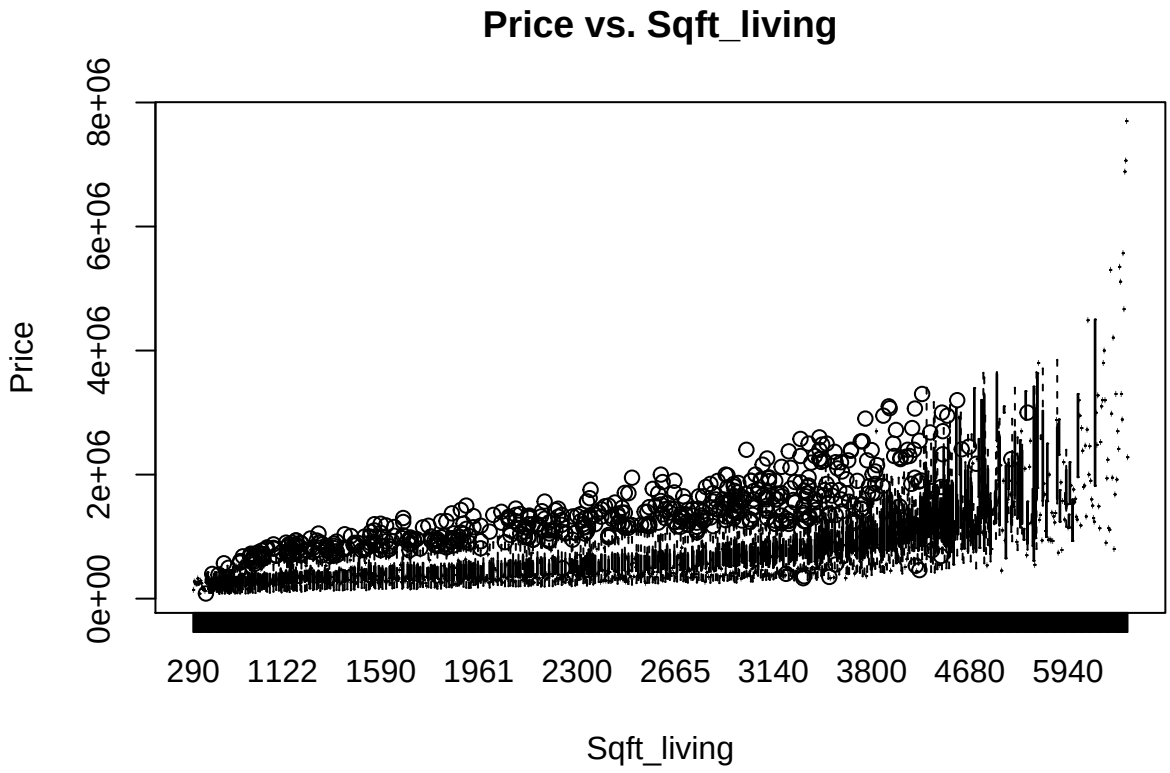
```
## lm(formula = price ~ sqft_living + grade + sqft_above + sqft_living15 +
##     bathrooms, data = myD)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1026038  -135316   -22098    98701  4829774
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -6.469e+05  1.351e+04 -47.870  < 2e-16 ***
## sqft_living    2.454e+02  4.524e+00  54.251  < 2e-16 ***
## grade          1.110e+05  2.462e+03  45.090  < 2e-16 ***
## sqft_above    -8.048e+01  4.455e+00 -18.067  < 2e-16 ***
## sqft_living15  2.282e+01  4.027e+00   5.667 1.47e-08 ***
## bathrooms     -3.546e+04  3.426e+03 -10.353  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 247900 on 21607 degrees of freedom
## Multiple R-squared:  0.5442, Adjusted R-squared:  0.5441
## F-statistic:  5160 on 5 and 21607 DF,  p-value: < 2.2e-16
```

```
# R^2 = 0.5442, adjR^2 = 0.5441
# select model by adjusted R^2
result_sim_fit <- as.data.frame(round(best_adjr2(fit_drop_basement_floors, nbest = 3), 4))
result_sim_fit <- result_sim_fit[order(-abs(result_sim_fit$adjr2)),]
head(result_sim_fit)
```

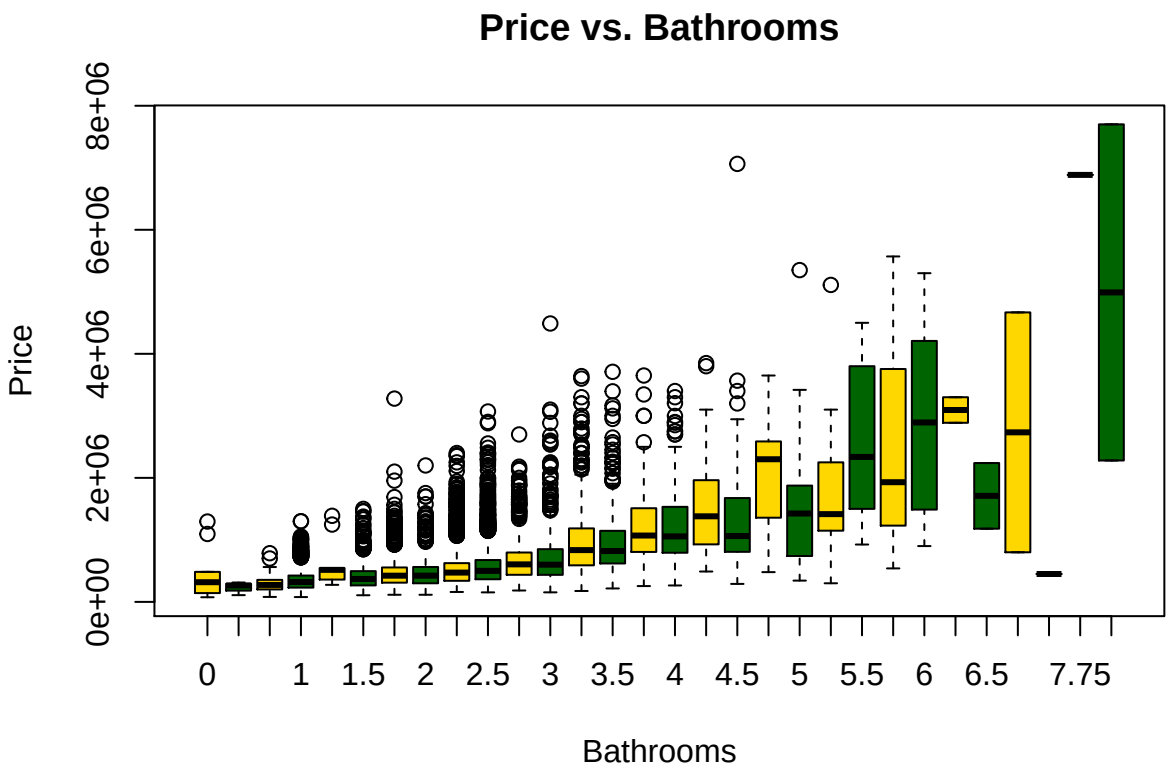
```
##     p (Intercept) bedrooms bathrooms sqft_living sqft_lot waterfront view
## 8   8           1        1         1           1        0          1    1
## 8.1 8           1        0         0           1        0          1    1
## 8.2 8           1        1         0           1        0          1    1
## 7   7           1        1         0           1        0          1    1
## 7.1 7           1        0         0           1        0          1    1
## 7.2 7           1        0         1           1        0          1    1
##     condition grade sqft_above yr_built yr_renovated zipcode lat long
## 8           0     1          0        1            0       0   1    0
## 8.1         0     1          0        1            0       1   1    1
## 8.2         0     1          0        1            0       1   1    0
## 7           0     1          0        1            0       0   1    0
## 7.1         0     1          0        1            0       1   1    0
## 7.2         0     1          0        1            0       0   1    0
##     sqft_living15 sqft_lot15  adjr2
## 8               0          0 0.6900
## 8.1             0          0 0.6894
## 8.2             0          0 0.6892
## 7               0          0 0.6869
## 7.1             0          0 0.6860
## 7.2             0          0 0.6856
```

```
# So best model by adjusted R^2 is
# price ~ bedrooms + bathrooms + sqft_living + waterfront + view + grade + yr_built + lat
# top models by adjusted R^2 should contain
# sqft_living, waterfront, view, grade, yr_built and lat.
```

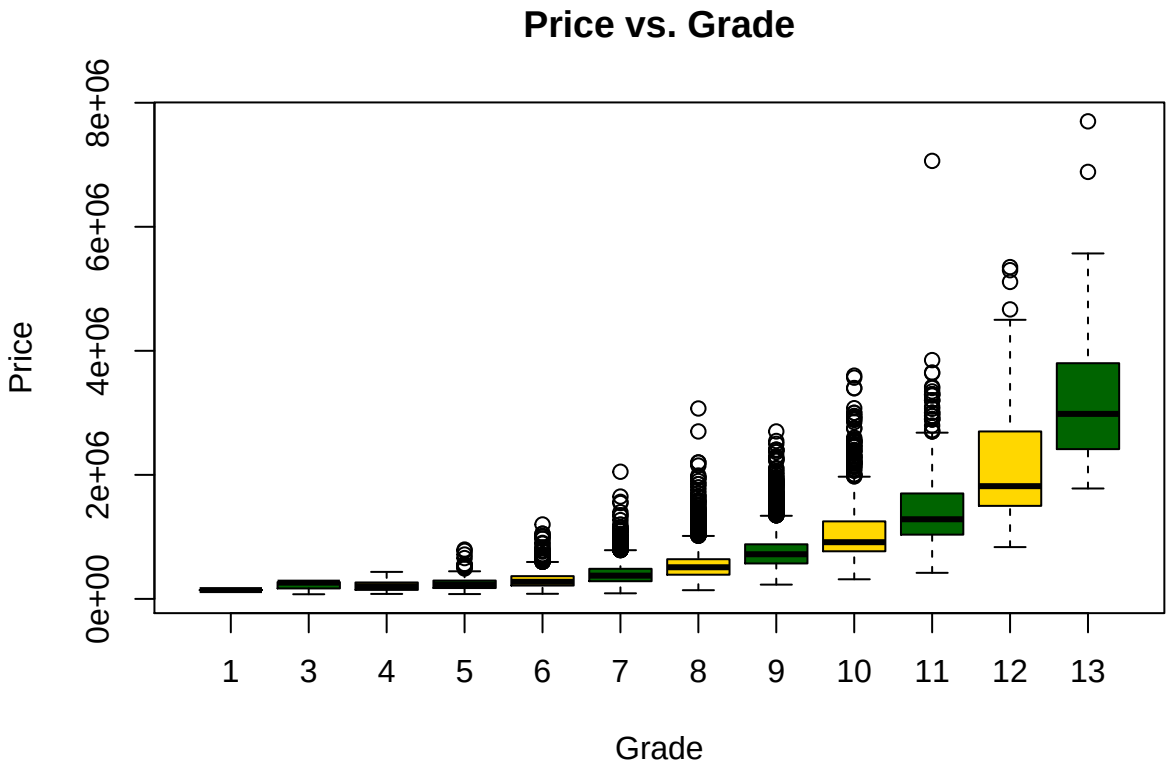
```r
## boxplots for some high-correlated variables and price
boxplot_p_sl <- boxplot(price~sqft_living, data=myD,
                        main="Price vs. Sqft_living", xlab="Sqft_living", ylab="Price")
```



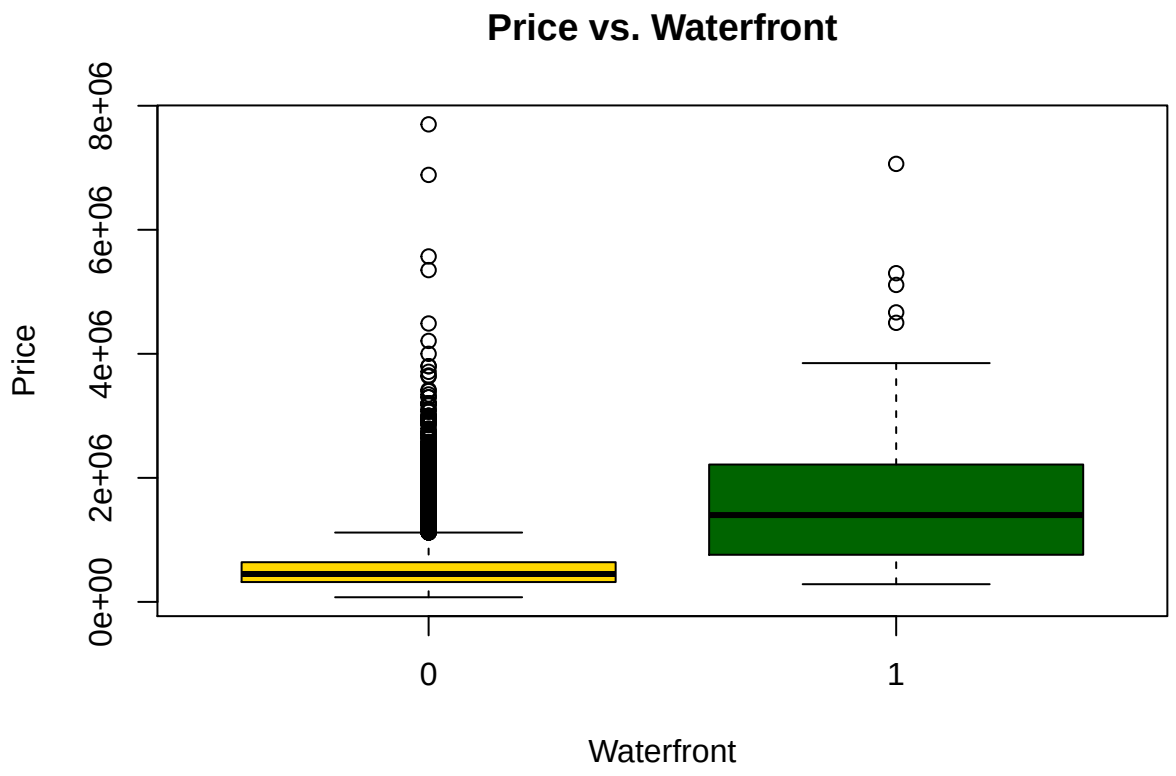

```r
boxplot_p_b <- boxplot(price~bathrooms, data=myD,
                       col=(c("gold","darkgreen")),
                       main="Price vs. Bathrooms", xlab="Bathrooms", ylab="Price")
```



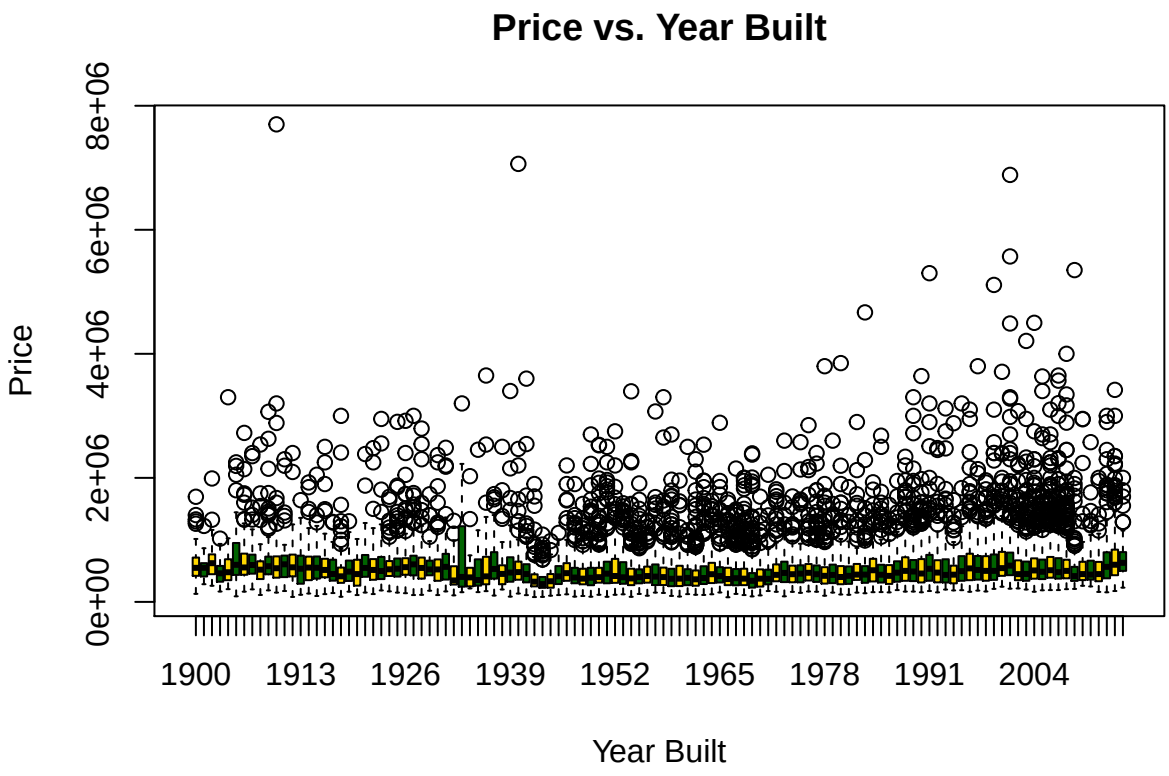

```
boxplot_p_g <- boxplot(price~grade, data=myD,
                       col=(c("gold","darkgreen")),
                       main="Price vs. Grade", xlab="Grade", ylab="Price")
```



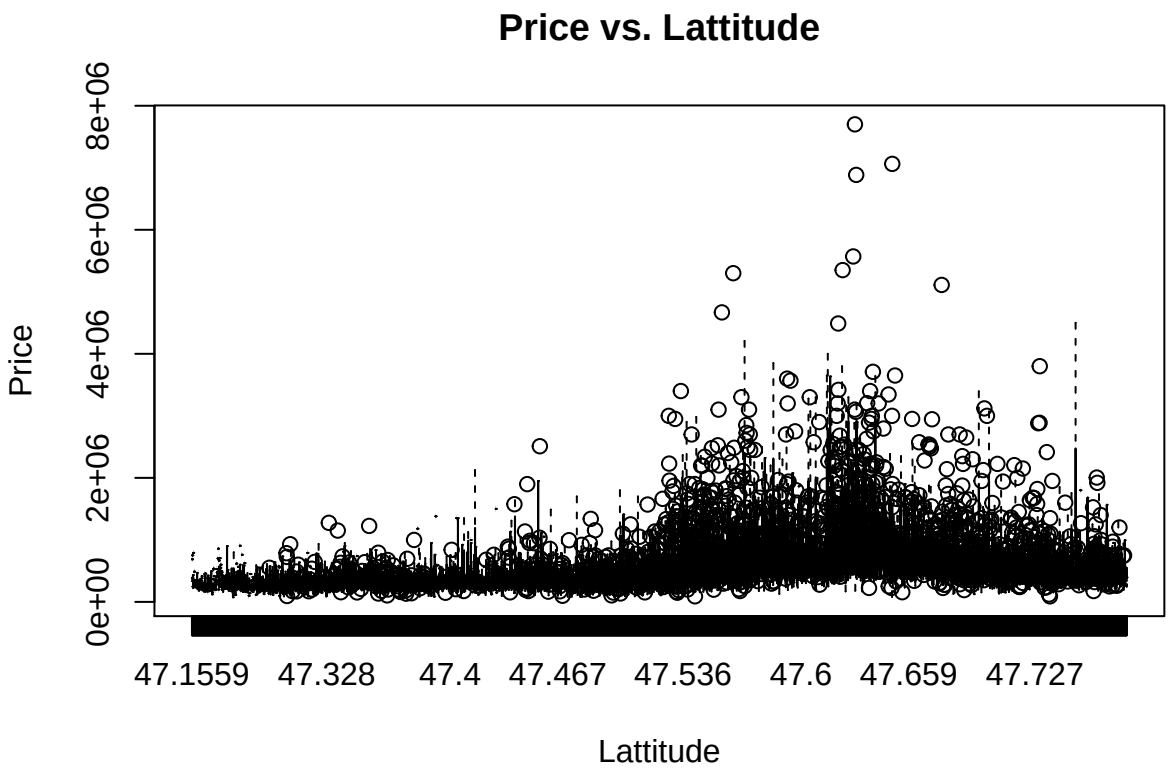

```
boxplot_p_wf <- boxplot(price~waterfront, data=myD,
                        col=(c("gold","darkgreen")),
                        main="Price vs. Waterfront", xlab="Waterfront", ylab="Price")
```




```
boxplot_p_yb <- boxplot(price~yr_built, data=myD,
                        col=(c("gold","darkgreen")),
                        main="Price vs. Year Built", xlab="Year Built", ylab="Price")
```




```
boxplot_p_lat <- boxplot(price~lat, data=myD,
                         col=(c("gold","darkgreen")),
                         main="Price vs. Lattitude", xlab="Lattitude", ylab="Price")
```




```
#The relationship looks a bit non-linear here

## log price
myD$logP <- log(myD$price)
myD <- myD[,-1]
log_price_model <- lm(logP~.-sqft_basement, data = myD)
summary(log_price_model)
```

```
##
## Call:
## lm(formula = logP ~ . - sqft_basement, data = myD)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.78817 -0.16139  0.00316  0.15887  1.19290
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -5.073e+00  3.677e+00  -1.379  0.16776
## bedrooms      -1.221e-02  2.373e-03  -5.144 2.71e-07 ***
## bathrooms      6.912e-02  4.081e-03  16.936  < 2e-16 ***
## sqft_living    1.512e-04  5.501e-06  27.494  < 2e-16 ***
## sqft_lot       4.712e-07  6.011e-08   7.838 4.78e-15 ***
## floors         7.515e-02  4.511e-03  16.661  < 2e-16 ***
## waterfront     3.712e-01  2.178e-02  17.046  < 2e-16 ***
## view           6.040e-02  2.684e-03  22.501  < 2e-16 ***
```

```
## condition      6.264e-02  2.950e-03  21.235  < 2e-16 ***
## grade          1.589e-01  2.700e-03  58.855  < 2e-16 ***
## sqft_above    -1.529e-05  5.470e-06  -2.795  0.00520 **
## yr_built      -3.411e-03  9.114e-05 -37.419  < 2e-16 ***
## yr_renovated   3.659e-05  4.586e-06   7.979 1.54e-15 ***
## zipcode       -6.459e-04  4.138e-05 -15.610  < 2e-16 ***
## lat            1.400e+00  1.347e-02 103.968  < 2e-16 ***
## long          -1.592e-01  1.648e-02  -9.660  < 2e-16 ***
## sqft_living15  9.857e-05  4.325e-06  22.791  < 2e-16 ***
## sqft_lot15    -2.610e-07  9.191e-08  -2.840  0.00452 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2524 on 21595 degrees of freedom
## Multiple R-squared:  0.7704, Adjusted R-squared:  0.7703
## F-statistic:  4263 on 17 and 21595 DF,  p-value: < 2.2e-16
```

```
# R^2 = 0.7704, adR^2 = 0.7703
result_log <- as.data.frame(round(best_adjr2(log_price_model, nbest = 3), 4))
result_log <- result_log[order(-abs(result_log$adjr2)),]
head(result_log)
```

```
##     p (Intercept) bedrooms bathrooms sqft_living sqft_lot floors
## 8   8           1        0         1           1        0      0
## 8.1 8           1        0         0           1        0      1
## 8.2 8           1        0         1           1        0      0
## 7   7           1        0         1           1        0      0
## 7.1 7           1        0         1           1        0      0
## 7.2 7           1        0         0           1        0      0
##     waterfront view condition grade sqft_above yr_built yr_renovated
## 8            0    1         1     1          0        1            0
## 8.1          0    1         1     1          0        1            0
## 8.2          1    1         0     1          0        1            0
## 7            0    1         0     1          0        1            0
## 7.1          0    1         1     1          0        1            0
## 7.2          0    1         1     1          0        1            0
##     zipcode lat long sqft_living15 sqft_lot15  adjr2
## 8         0   1    0             1          0 0.7595
## 8.1       0   1    0             1          0 0.7589
## 8.2       0   1    0             1          0 0.7588
## 7         0   1    0             1          0 0.7553
## 7.1       0   1    0             0          0 0.7544
## 7.2       0   1    0             1          0 0.7539
```

```
# So best model by adjusted R^2 is
# logP ~ bathrooms + sqft_living + view + condition + grade + yr_built + lat + sqft_living15
# top models by adjusted R^2 should contain
# sqft_living, view, grade, yr_built, lat, and sqft_living15.

## select with Cp
result_log_cp <- as.data.frame(round(best_cp(log_price_model, nbest = 3), 4))
result_log_cp <- result_log_cp[order(abs(result_log_cp$cp)),]
head(result_log_cp)
```

```
##     p (Intercept) bedrooms bathrooms sqft_living sqft_lot floors
```

```
## 8   8           1        0         1           1        0      0
## 8.1 8           1        0         0           1        0      1
## 8.2 8           1        0         1           1        0      0
## 7   7           1        0         1           1        0      0
## 7.1 7           1        0         1           1        0      0
## 7.2 7           1        0         0           1        0      0
##     waterfront view condition grade sqft_above yr_built yr_renovated
## 8            0    1         1     1          0        1            0
## 8.1          0    1         1     1          0        1            0
## 8.2          1    1         0     1          0        1            0
## 7            0    1         0     1          0        1            0
## 7.1          0    1         1     1          0        1            0
## 7.2          0    1         1     1          0        1            0
##     zipcode lat long sqft_living15 sqft_lot15       cp
## 8         0   1    0             1          0 1025.864
## 8.1       0   1    0             1          0 1074.810
## 8.2       0   1    0             1          0 1086.717
## 7         0   1    0             1          0 1414.986
## 7.1       0   1    0             0          0 1502.692
## 7.2       0   1    0             1          0 1542.836
```

```
## wald test for view
fit.coef = summary(log_price_model)$coef
alpha = 0.05
zStar = fit.coef[8,1]/fit.coef[8,2]
zStar <= qnorm(1-alpha/2)
```

```
## [1] FALSE
```

```
fit.coef[4,4]
```

```
## [1] 1.354671e-163
```

```
# Ho: beta=0
# Ha: beta!=0
# zstar > qnorm, therefore we conclude H_a that beta of view is not zero. the p-value of this test is v

## AIC selection
model.null <- lm(logP~.-sqft_basement, data = myD)
log_price_model.AIC <- stepAIC(log_price_model, scope = list(upper = log_price_model, lower = model.null
log_price_model.AIC$formula
```

```
## logP ~ (bedrooms + bathrooms + sqft_living + sqft_lot + floors +
##     waterfront + view + condition + grade + sqft_above + sqft_basement +
##     yr_built + yr_renovated + zipcode + lat + long + sqft_living15 +
##     sqft_lot15) - sqft_basement
## attr(,"variables")
## list(logP, bedrooms, bathrooms, sqft_living, sqft_lot, floors,
##     waterfront, view, condition, grade, sqft_above, sqft_basement,
##     yr_built, yr_renovated, zipcode, lat, long, sqft_living15,
##     sqft_lot15)
## attr(,"factors")
##               bedrooms bathrooms sqft_living sqft_lot floors waterfront
## logP                 0         0           0       0      0          0
## bedrooms             1         0           0       0      0          0
## bathrooms            0         1           0       0      0          0
```

```
## sqft_living          0         0           1        0      0           0
## sqft_lot             0         0           0        1      0           0
## floors               0         0           0        0      1           0
## waterfront           0         0           0        0      0           1
## view                 0         0           0        0      0           0
## condition            0         0           0        0      0           0
## grade                0         0           0        0      0           0
## sqft_above           0         0           0        0      0           0
## sqft_basement        0         0           0        0      0           0
## yr_built             0         0           0        0      0           0
## yr_renovated         0         0           0        0      0           0
## zipcode              0         0           0        0      0           0
## lat                  0         0           0        0      0           0
## long                 0         0           0        0      0           0
## sqft_living15        0         0           0        0      0           0
## sqft_lot15           0         0           0        0      0           0
##               view condition grade sqft_above yr_built yr_renovated
## logP             0         0     0          0        0            0
## bedrooms         0         0     0          0        0            0
## bathrooms        0         0     0          0        0            0
## sqft_living      0         0     0          0        0            0
## sqft_lot         0         0     0          0        0            0
## floors           0         0     0          0        0            0
## waterfront       0         0     0          0        0            0
## view             1         0     0          0        0            0
## condition        0         1     0          0        0            0
## grade            0         0     1          0        0            0
## sqft_above       0         0     0          1        0            0
## sqft_basement    0         0     0          0        0            0
## yr_built         0         0     0          0        1            0
## yr_renovated     0         0     0          0        0            1
## zipcode          0         0     0          0        0            0
## lat              0         0     0          0        0            0
## long             0         0     0          0        0            0
## sqft_living15    0         0     0          0        0            0
## sqft_lot15       0         0     0          0        0            0
##               zipcode lat long sqft_living15 sqft_lot15
## logP                0   0    0             0          0
## bedrooms            0   0    0             0          0
## bathrooms           0   0    0             0          0
## sqft_living         0   0    0             0          0
## sqft_lot            0   0    0             0          0
## floors              0   0    0             0          0
## waterfront          0   0    0             0          0
## view                0   0    0             0          0
## condition           0   0    0             0          0
## grade               0   0    0             0          0
## sqft_above          0   0    0             0          0
## sqft_basement       0   0    0             0          0
## yr_built            0   0    0             0          0
## yr_renovated        0   0    0             0          0
## zipcode             1   0    0             0          0
## lat                 0   1    0             0          0
## long                0   0    1             0          0
```

```
## sqft_living15       0   0    0             1          0
## sqft_lot15          0   0    0             0          1
## attr(,"term.labels")
##  [1] "bedrooms"      "bathrooms"     "sqft_living"   "sqft_lot"
##  [5] "floors"        "waterfront"    "view"          "condition"
##  [9] "grade"         "sqft_above"    "yr_built"      "yr_renovated"
## [13] "zipcode"       "lat"           "long"          "sqft_living15"
## [17] "sqft_lot15"
## attr(,"order")
##  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
## attr(,"intercept")
## [1] 1
## attr(,"response")
## [1] 1
## attr(,".Environment")
## <environment: R_GlobalEnv>
## attr(,"predvars")
## list(logP, bedrooms, bathrooms, sqft_living, sqft_lot, floors,
##     waterfront, view, condition, grade, sqft_above, sqft_basement,
##     yr_built, yr_renovated, zipcode, lat, long, sqft_living15,
##     sqft_lot15)
## attr(,"dataClasses")
##          logP      bedrooms     bathrooms   sqft_living      sqft_lot
##     "numeric"     "numeric"     "numeric"     "numeric"     "numeric"
##        floors    waterfront          view     condition         grade
##     "numeric"     "numeric"     "numeric"     "numeric"     "numeric"
##    sqft_above sqft_basement      yr_built  yr_renovated       zipcode
##     "numeric"     "numeric"     "numeric"     "numeric"     "numeric"
##           lat          long sqft_living15    sqft_lot15
##     "numeric"     "numeric"     "numeric"     "numeric"
```

```
#BIC
log_price_model.BIC <- step(log_price_model, scope = list(upper = log_price_model, lower = model.null),
```

```
## Start:  AIC=-59341.72
## logP ~ (bedrooms + bathrooms + sqft_living + sqft_lot + floors +
##     waterfront + view + condition + grade + sqft_above + sqft_basement +
##     yr_built + yr_renovated + zipcode + lat + long + sqft_living15 +
##     sqft_lot15) - sqft_basement
```

```
log_price_model.BIC$formula
```

```
## logP ~ (bedrooms + bathrooms + sqft_living + sqft_lot + floors +
##     waterfront + view + condition + grade + sqft_above + sqft_basement +
##     yr_built + yr_renovated + zipcode + lat + long + sqft_living15 +
##     sqft_lot15) - sqft_basement
## attr(,"variables")
## list(logP, bedrooms, bathrooms, sqft_living, sqft_lot, floors,
##     waterfront, view, condition, grade, sqft_above, sqft_basement,
##     yr_built, yr_renovated, zipcode, lat, long, sqft_living15,
##     sqft_lot15)
## attr(,"factors")
##               bedrooms bathrooms sqft_living sqft_lot floors waterfront
## logP                 0         0           0       0      0          0
## bedrooms             1         0           0       0      0          0
```

```
## bathrooms            0         1           0        0      0           0
## sqft_living          0         0           1        0      0           0
## sqft_lot             0         0           0        1      0           0
## floors               0         0           0        0      1           0
## waterfront           0         0           0        0      0           1
## view                 0         0           0        0      0           0
## condition            0         0           0        0      0           0
## grade                0         0           0        0      0           0
## sqft_above           0         0           0        0      0           0
## sqft_basement        0         0           0        0      0           0
## yr_built             0         0           0        0      0           0
## yr_renovated         0         0           0        0      0           0
## zipcode              0         0           0        0      0           0
## lat                  0         0           0        0      0           0
## long                 0         0           0        0      0           0
## sqft_living15        0         0           0        0      0           0
## sqft_lot15           0         0           0        0      0           0
##                view condition grade sqft_above yr_built yr_renovated
## logP              0         0     0          0        0            0
## bedrooms          0         0     0          0        0            0
## bathrooms         0         0     0          0        0            0
## sqft_living       0         0     0          0        0            0
## sqft_lot          0         0     0          0        0            0
## floors            0         0     0          0        0            0
## waterfront        0         0     0          0        0            0
## view              1         0     0          0        0            0
## condition         0         1     0          0        0            0
## grade             0         0     1          0        0            0
## sqft_above        0         0     0          1        0            0
## sqft_basement     0         0     0          0        0            0
## yr_built          0         0     0          0        1            0
## yr_renovated      0         0     0          0        0            1
## zipcode           0         0     0          0        0            0
## lat               0         0     0          0        0            0
## long              0         0     0          0        0            0
## sqft_living15     0         0     0          0        0            0
## sqft_lot15        0         0     0          0        0            0
##                zipcode lat long sqft_living15 sqft_lot15
## logP                 0   0    0             0          0
## bedrooms             0   0    0             0          0
## bathrooms            0   0    0             0          0
## sqft_living          0   0    0             0          0
## sqft_lot             0   0    0             0          0
## floors               0   0    0             0          0
## waterfront           0   0    0             0          0
## view                 0   0    0             0          0
## condition            0   0    0             0          0
## grade                0   0    0             0          0
## sqft_above           0   0    0             0          0
## sqft_basement        0   0    0             0          0
## yr_built             0   0    0             0          0
## yr_renovated         0   0    0             0          0
## zipcode              1   0    0             0          0
## lat                  0   1    0             0          0
```

```
## long                0   0    1             0          0
## sqft_living15       0   0    0             1          0
## sqft_lot15          0   0    0             0          1
## attr(,"term.labels")
##  [1] "bedrooms"      "bathrooms"     "sqft_living"   "sqft_lot"
##  [5] "floors"        "waterfront"    "view"          "condition"
##  [9] "grade"         "sqft_above"    "yr_built"      "yr_renovated"
## [13] "zipcode"       "lat"           "long"          "sqft_living15"
## [17] "sqft_lot15"
## attr(,"order")
##  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
## attr(,"intercept")
## [1] 1
## attr(,"response")
## [1] 1
## attr(,".Environment")
## <environment: R_GlobalEnv>
## attr(,"predvars")
## list(logP, bedrooms, bathrooms, sqft_living, sqft_lot, floors,
##     waterfront, view, condition, grade, sqft_above, sqft_basement,
##     yr_built, yr_renovated, zipcode, lat, long, sqft_living15,
##     sqft_lot15)
## attr(,"dataClasses")
##          logP      bedrooms     bathrooms   sqft_living      sqft_lot
##     "numeric"     "numeric"     "numeric"     "numeric"     "numeric"
##        floors    waterfront          view     condition         grade
##     "numeric"     "numeric"     "numeric"     "numeric"     "numeric"
##    sqft_above sqft_basement      yr_built  yr_renovated       zipcode
##     "numeric"     "numeric"     "numeric"     "numeric"     "numeric"
##           lat          long sqft_living15    sqft_lot15
##     "numeric"     "numeric"     "numeric"     "numeric"
```

```
## VIF
library(car)
vif(log_price_model)
```

```
##      bedrooms     bathrooms   sqft_living      sqft_lot        floors
##      1.652063      3.350793      8.656765      2.102522      2.011907
##    waterfront          view     condition         grade    sqft_above
##      1.203766      1.435160      1.249475      3.417046      6.957060
##      yr_built  yr_renovated       zipcode           lat          long
##      2.430649      1.150554      1.662174      1.180630      1.825579
## sqft_living15    sqft_lot15
##      2.979713      2.135668
```

```
max(vif(log_price_model)) > 10
```

```
## [1] FALSE
```

```
mean(vif(log_price_model))
```

```
## [1] 2.670678
```

```
# there exists multicollinearity in the model.
vif(fit_drop_basement_floors)
```

```
##      bedrooms     bathrooms   sqft_living      sqft_lot    waterfront
```

```
##      1.650836      3.124278      7.846355      2.101503      1.203752
##          view     condition         grade    sqft_above      yr_built
##      1.434339      1.245479      3.390557      5.594685      2.315424
##  yr_renovated       zipcode           lat          long sqft_living15
##      1.147325      1.647680      1.172532      1.812200      2.942773
##    sqft_lot15
##      2.133044
```

```
max(vif(fit_drop_basement_floors)) > 10
```

```
## [1] FALSE
```

```
mean(vif(fit_drop_basement_floors))
```

```
## [1] 2.547673
```