Homework 2

Ghandilyan Lilit

10/6/2018

We will be using the housing dataset. You are provided with the description of the dataset.

(2 point)Load the housing.csv and check whether the data types are correct, if not, make appropriate corrections assigning labels to each level according to the data description, so that it will be easy to interpret the model results during the next steps.

Pay attention to the variable grade. You can use function cut() here or something like it.

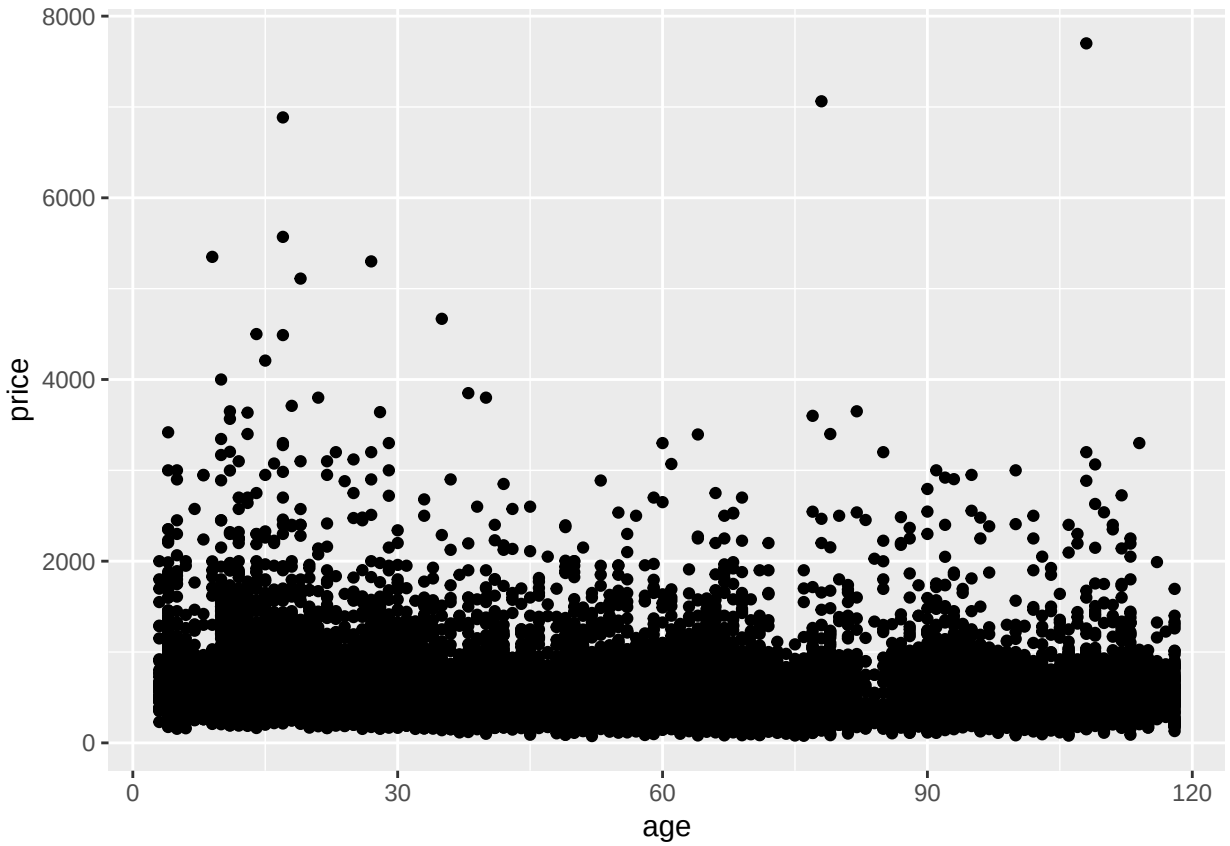
```
df <- read.csv("housing.csv")
str(df)
```

```
## 'data.frame':    21613 obs. of  21 variables:
##  $ id           : num  7.13e+09 1.95e+09 1.18e+09 7.14e+09 5.10e+09 ...
##  $ date         : Factor w/ 372 levels "20140502T000000",..: 165 284 306 63 54 76 138 283 173 32 ...
##  $ price        : num  221900 510000 530000 285000 438000 ...
##  $ bedrooms     : int  3 3 5 5 3 4 3 5 3 3 ...
##  $ bathrooms    : num  1 2 2 2.5 1.75 2.5 1 2.5 1.75 1 ...
##  $ sqft_living  : int  1180 1680 1810 2270 1520 2290 1190 3150 2519 960 ...
##  $ sqft_lot     : int  5650 8080 4850 6300 6380 13416 9199 9134 8690 6634 ...
##  $ floors       : num  1 1 1.5 2 1 2 1 1 2 1 ...
##  $ 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 3 3 4 3 4 5 3 ...
##  $ grade        : int  7 8 7 8 7 9 7 8 8 6 ...
##  $ sqft_above   : int  1180 1680 1810 2270 790 2290 1190 1640 2519 960 ...
##  $ sqft_basement: int  0 0 0 0 730 0 0 1510 0 0 ...
##  $ yr_built     : int  1955 1987 1900 1995 1948 1981 1955 1966 1973 1952 ...
##  $ yr_renovated : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ zipcode      : int  98178 98074 98107 98092 98115 98007 98148 98056 98166 98125 ...
##  $ lat          : num  47.5 47.6 47.7 47.3 47.7 ...
##  $ long         : num  -122 -122 -122 -122 -122 ...
##  $ sqft_living15: int  1340 1800 1360 2240 1520 2680 1190 1990 2500 1570 ...
##  $ sqft_lot15   : int  5650 7503 4850 7005 6235 13685 9364 9133 9500 7203 ...
```

```
df <- df %>% separate(date, "date", sep = "T000000")
df$date <- as.Date(df$date, "%Y%m%d")
df$price <- df$price/1000
df$waterfront  <- as.factor(df$waterfront)
df$view <- as.factor(df$view)
df$condition <- as.factor(df$condition)
df$zipcode <- as.factor(df$zipcode)
df$grade <- cut(df$grade, breaks = 12, labels = c(1:12))
```

(2 points) Create a variable with building's age (The data is collected at 2018). visualize the relationship between the newly created variable with the price and comment whether it can be significant predictor for the price.

```r
df$age <- 2018 - df$yr_built
ggplot(aes(age, price), data = df) + geom_point()
```




```r
#The plot does not show any clear relationship.
t.test(df$age, df$price)
```

```
##
##  Welch Two Sample t-test
##
## data:  df$age and df$price
## t = -196.83, df = 21889, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -498.0037 -488.1829
## sample estimates:
## mean of x mean of y
##  46.99486 540.08814
```

```r
#The p value is smaller than 0.05, so the relationship is significant.
#To avoid multicolinearity, I am going to delete the yr_built column.
df<-subset(df, select = -c(yr_built))
```

(2 points) Our goal in this analysis will be building a model to predict the price of the houses as accurately as possible. First, write a code to check what variables are highly correlated with the price variable. Hint: use function ?cor()

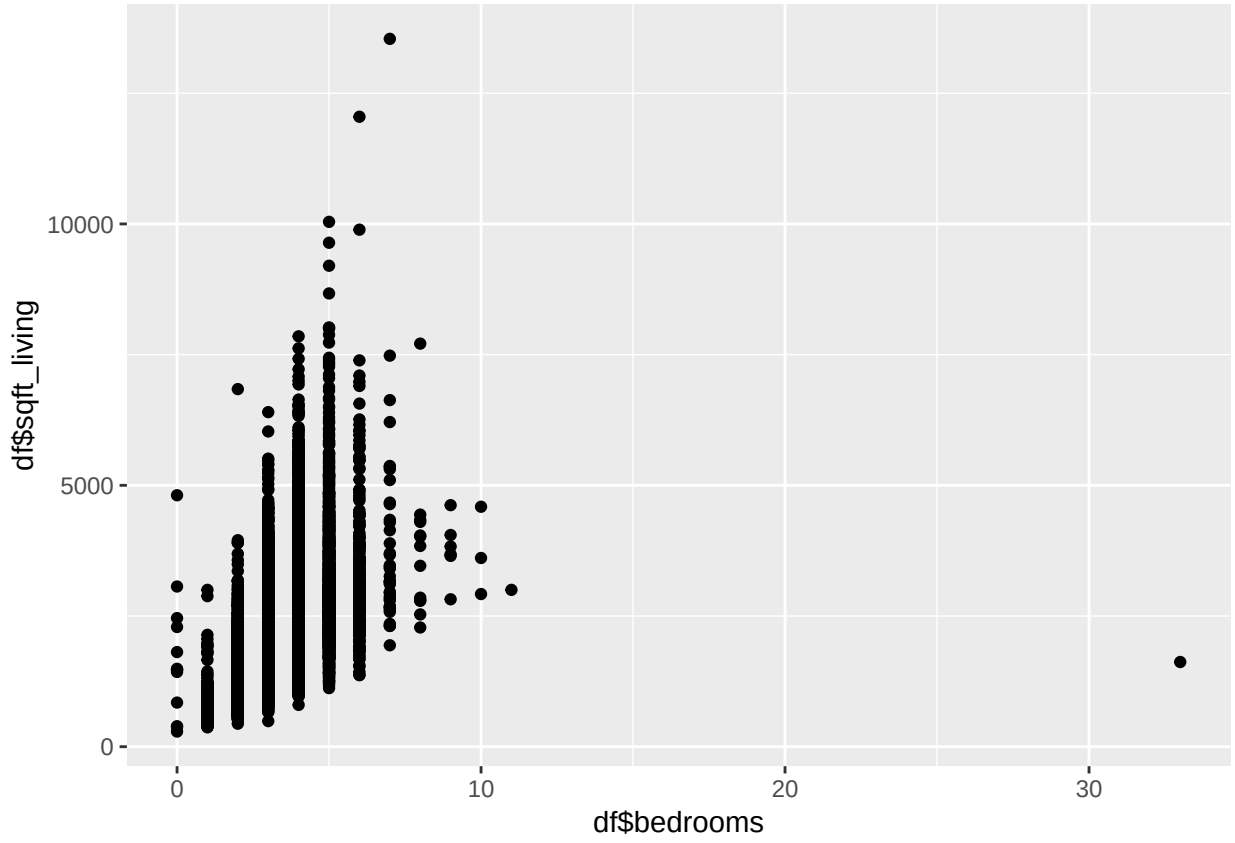
```
cor(df$price, df[, c(4:7, 13:15, 19:21)])
```

```
##       bedrooms bathrooms sqft_living   sqft_lot sqft_above sqft_basement
## [1,] 0.3083496 0.5251375   0.7020351 0.08966086  0.6055673      0.323816
##      yr_renovated sqft_living15 sqft_lot15         age
## [1,]    0.1264338     0.5853789 0.08244715 -0.05401153
```

(3 points) Visualize the relationship between the independent variables. In case you see it might cause multicolinearity during the modeling, also print correlation coeficients and make a note to act accordingly during modeling. Hint: usually variables having more than 0.7 correlation coeficients might cause multicolinearity.

1. the relationship between number of bedrooms and living area in sqft

```
ggplot(aes(df$bedrooms, df$sqft_living), data = df)+geom_point()
```

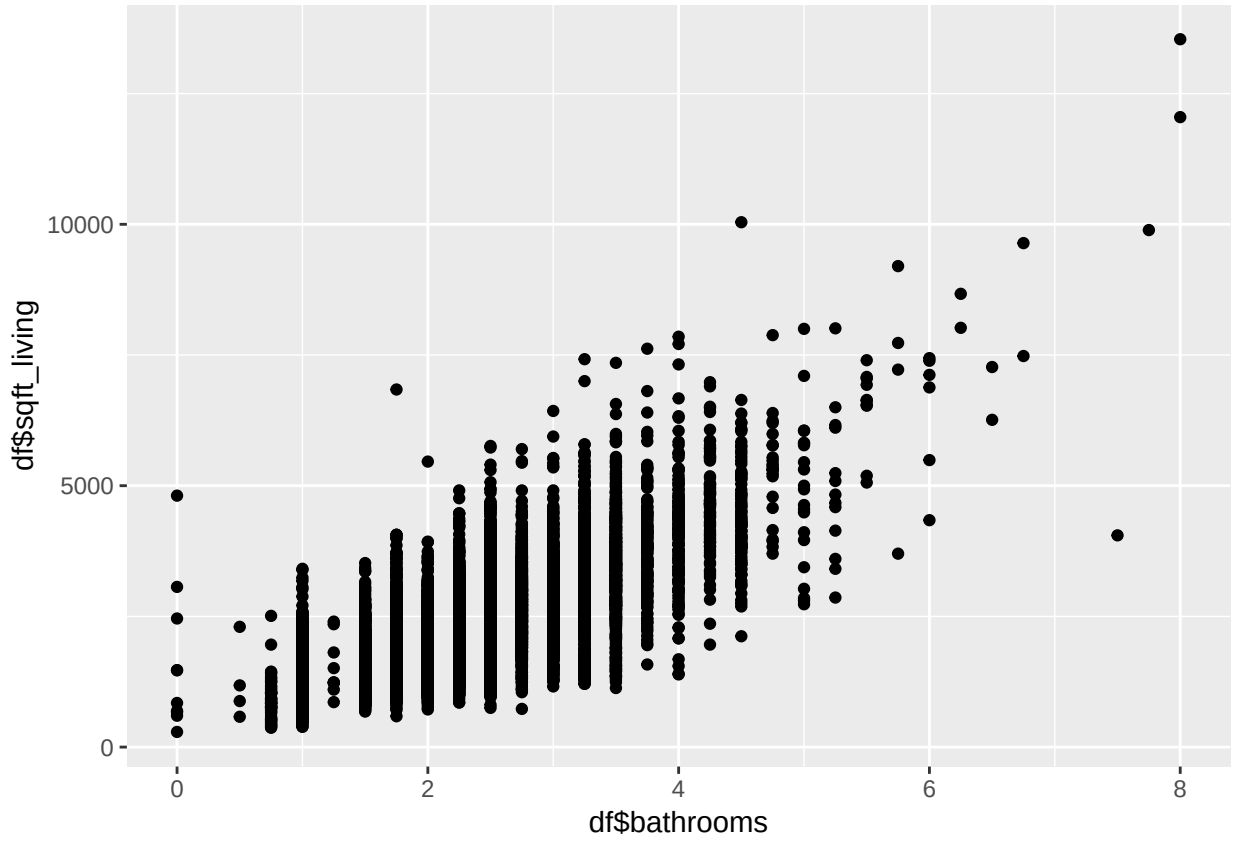



```
cor(df$bedrooms, df$sqft_living)
```

```
## [1] 0.5766707
```

2. the relationship betweeen the living area in sqft and the number of bathrooms

```
ggplot(aes(df$bathrooms, df$sqft_living), data = df)+geom_point()
```

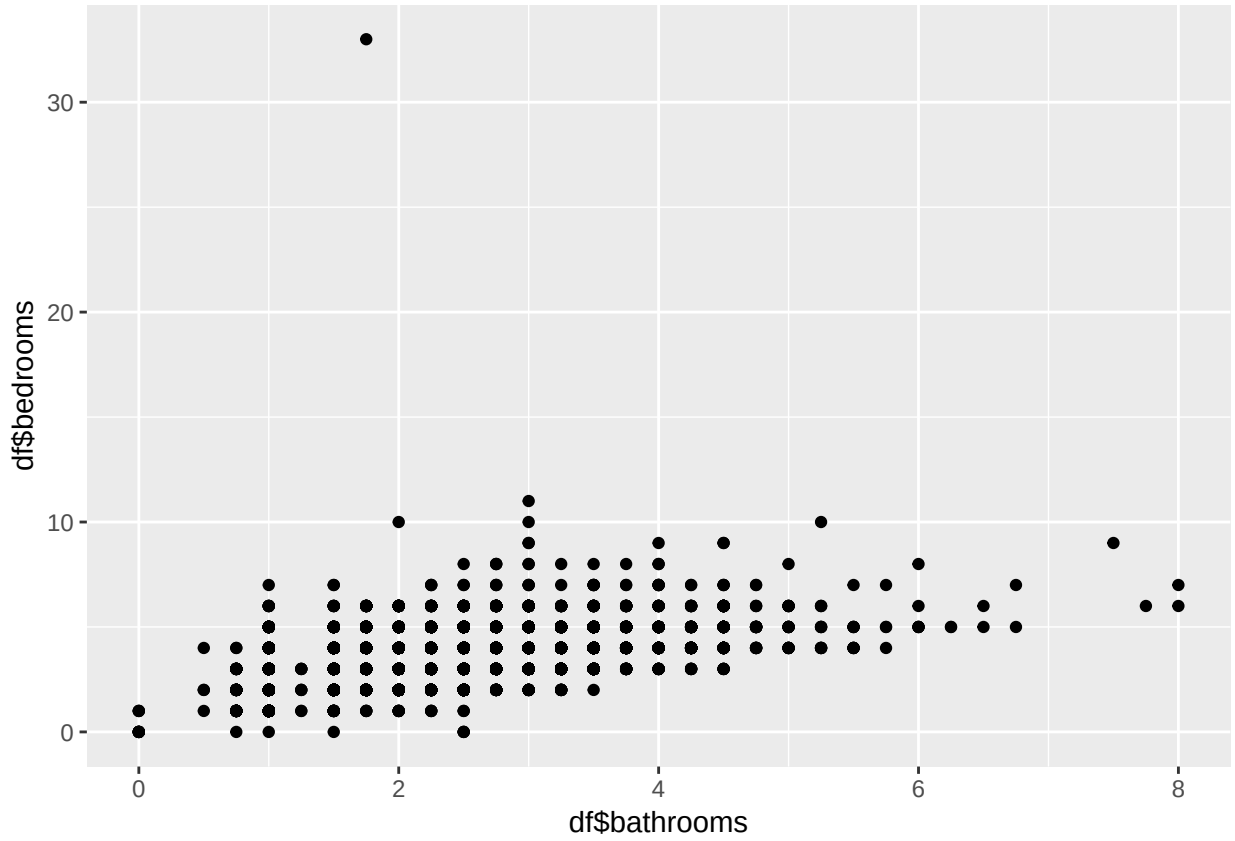



```
cor(df$sqft_living, df$bathrooms)
```

```
## [1] 0.7546653
```

3. the relationship between number of bedrooms and number of bathrooms

```
ggplot(aes(df$bathrooms, df$bedrooms), data = df)+geom_point()
```

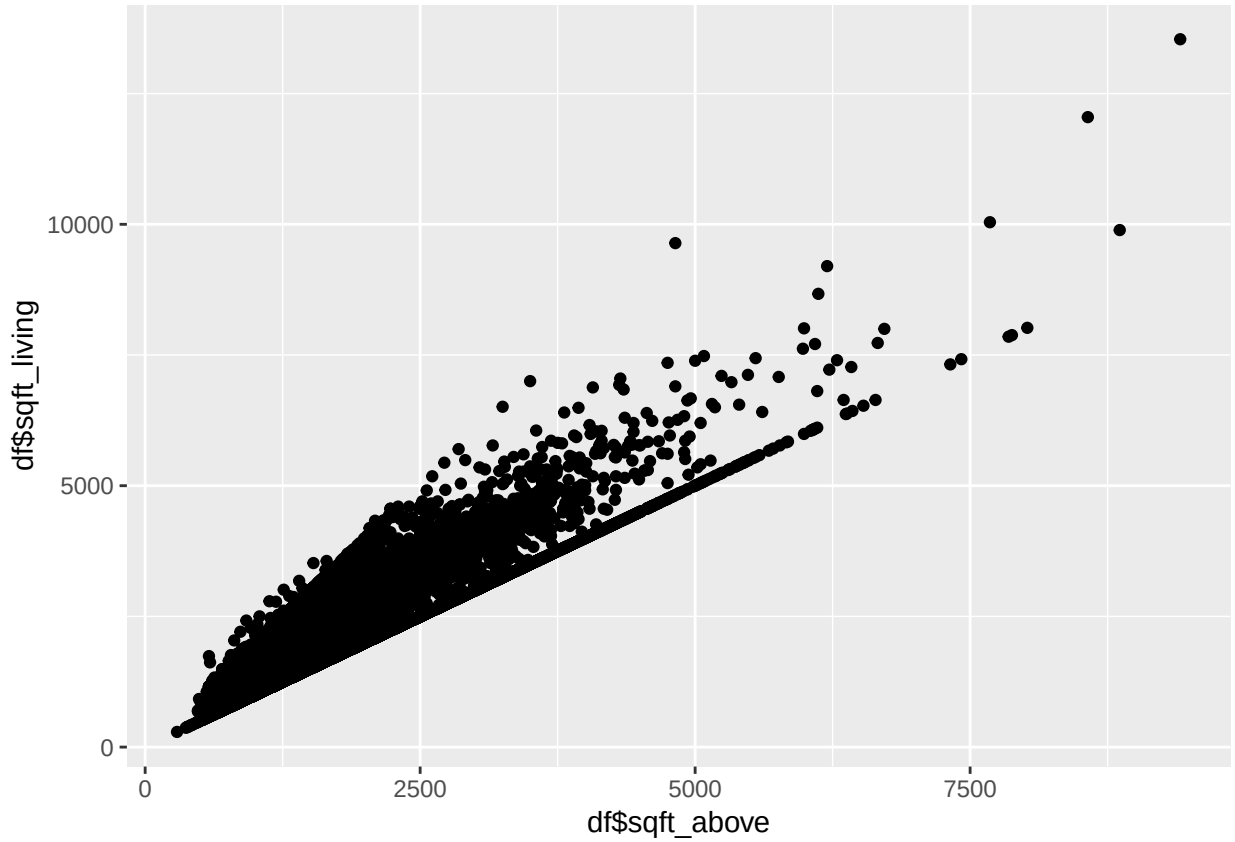



```
cor(df$bedrooms, df$bathrooms)
```

```
## [1] 0.5158836
```

4. the relationship betwen sqft_living and sqft_above

```
ggplot(aes(df$sqft_above, df$sqft_living), data = df)+geom_point()
```

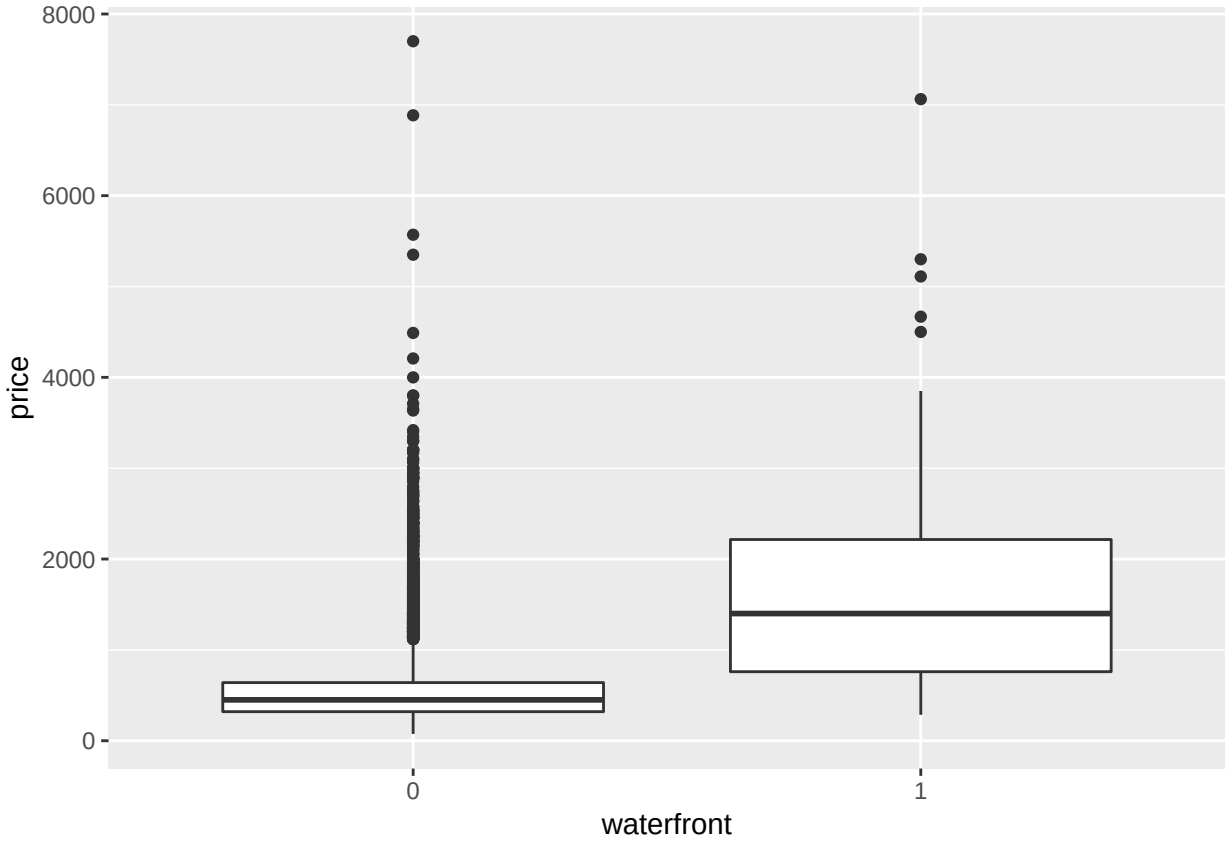



```
cor(df$sqft_living, df$sqft_above)
```

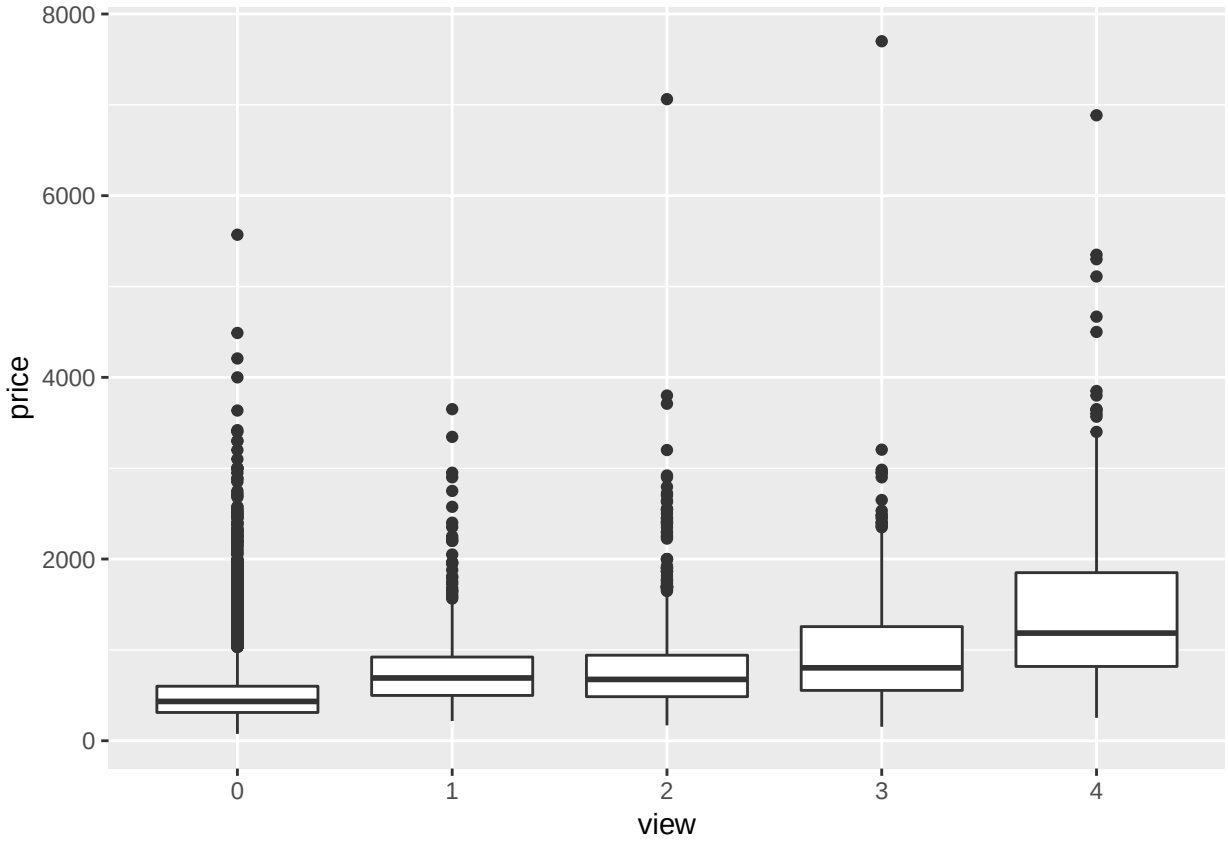
```
## [1] 0.8765966
```

(5 point) Usig ggplot visualizations explore the relationships between categorical variables and price. Also try to visualize whether the relationship between price and other numeric variables differ based on categorical variables such as waterfront, view, condition and grade.

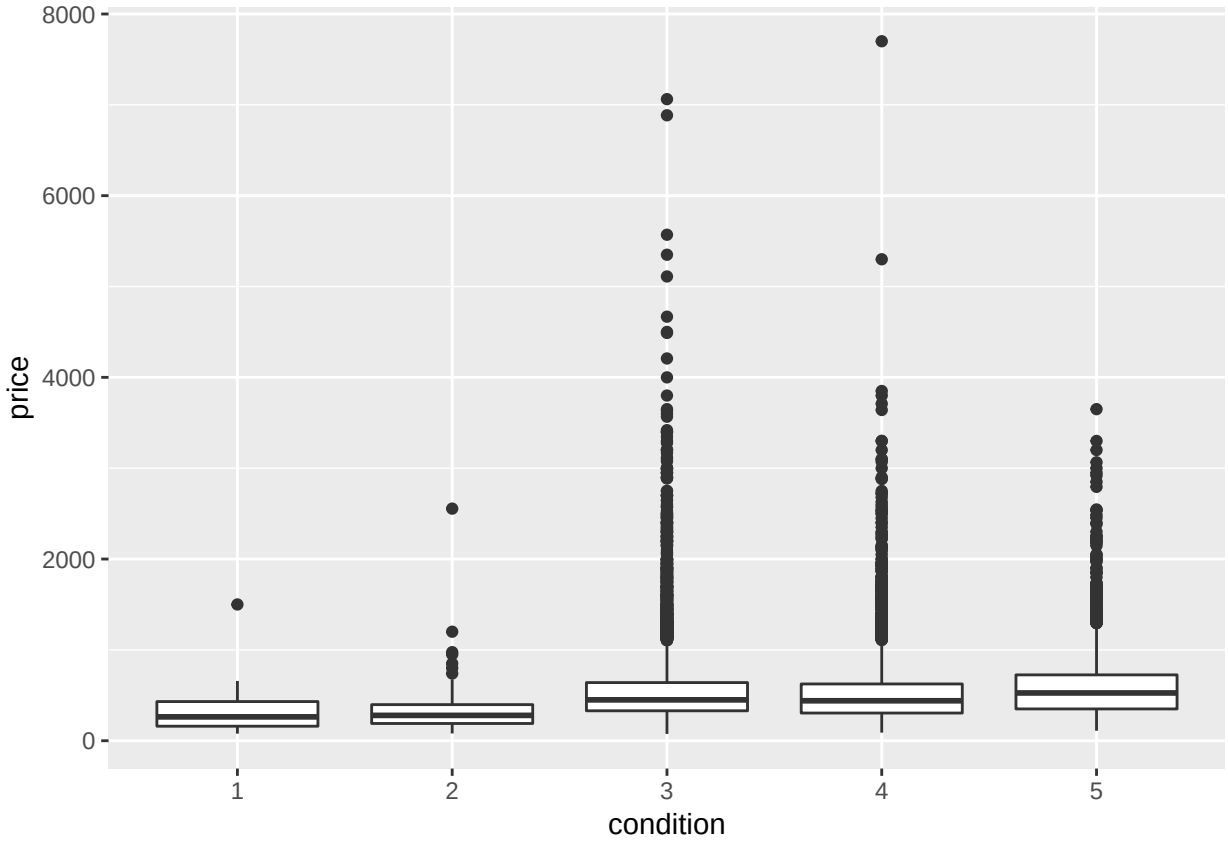
```
ggplot(df, aes(waterfront, price)) + geom_boxplot()
```



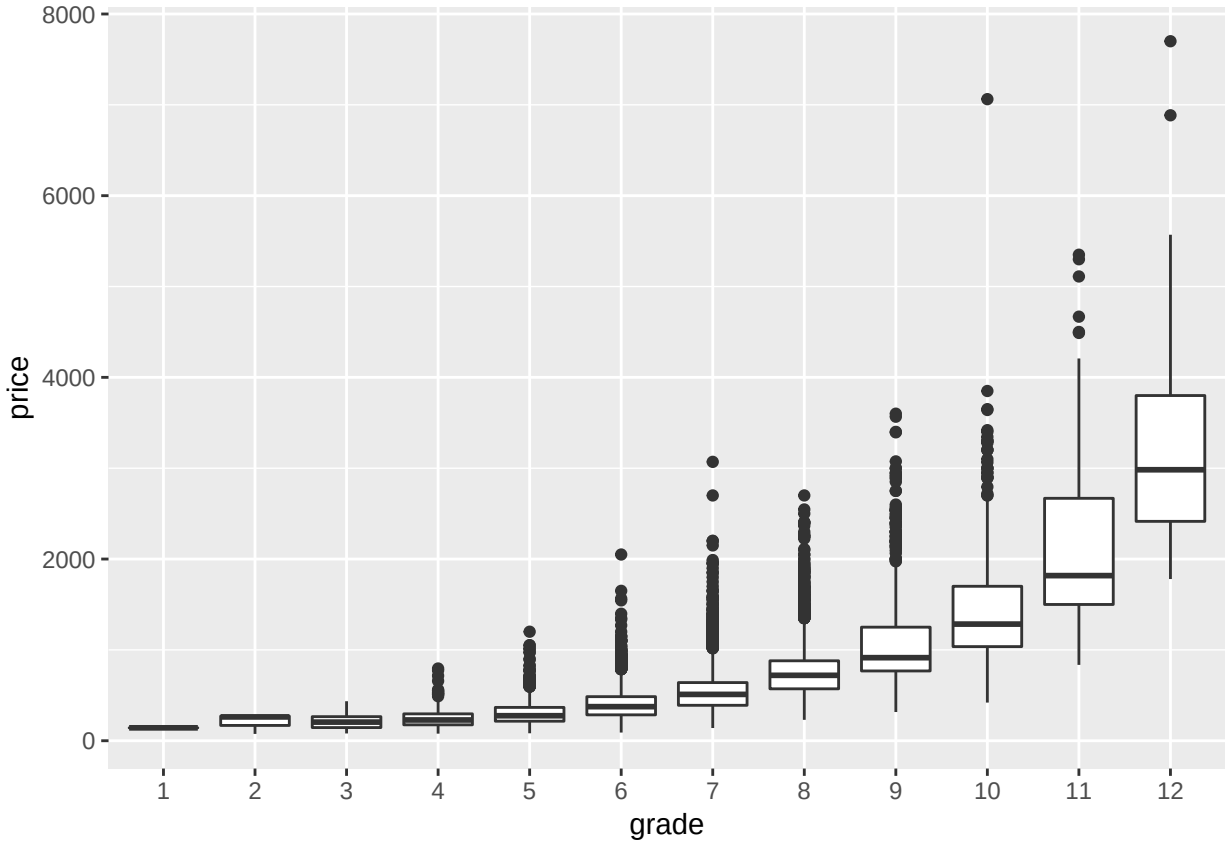

```
ggplot(df, aes(view, price)) + geom_boxplot()
```



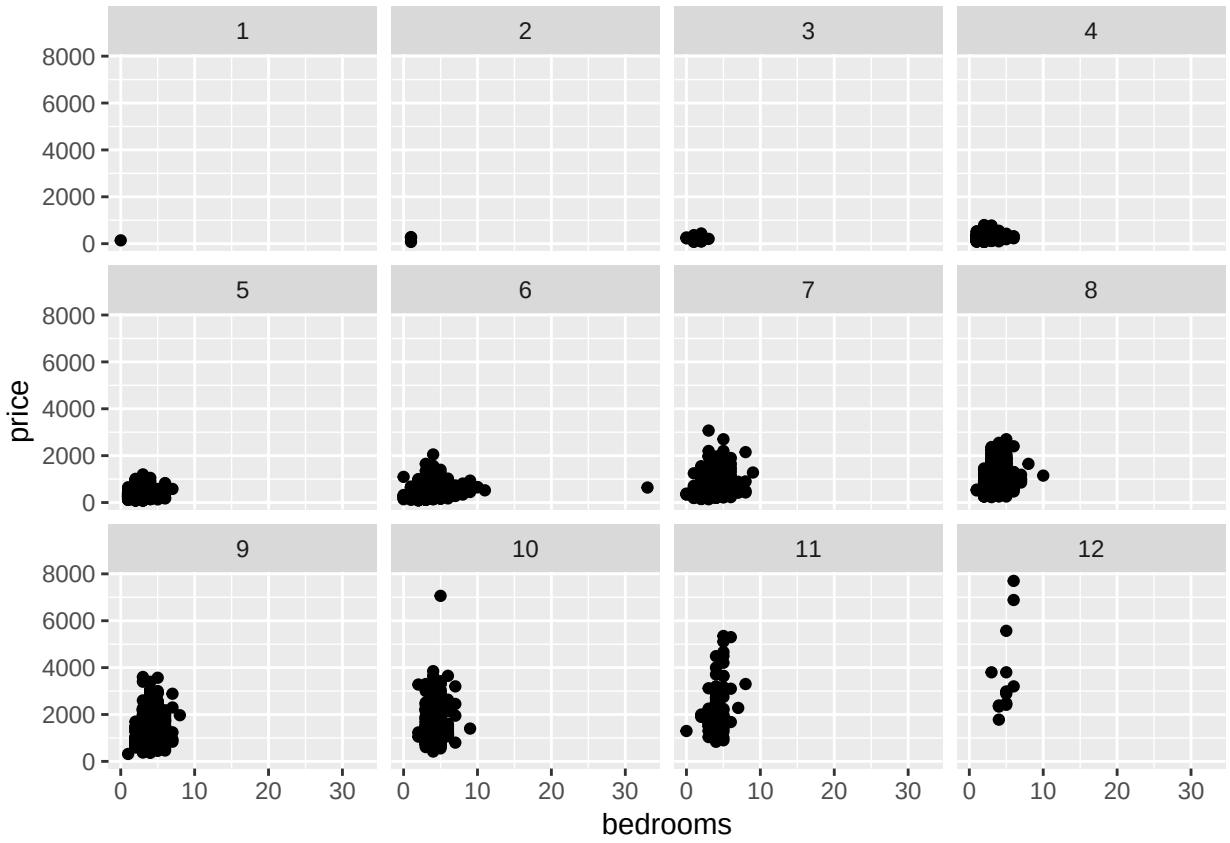

```
ggplot(df, aes(condition, price)) + geom_boxplot()
```



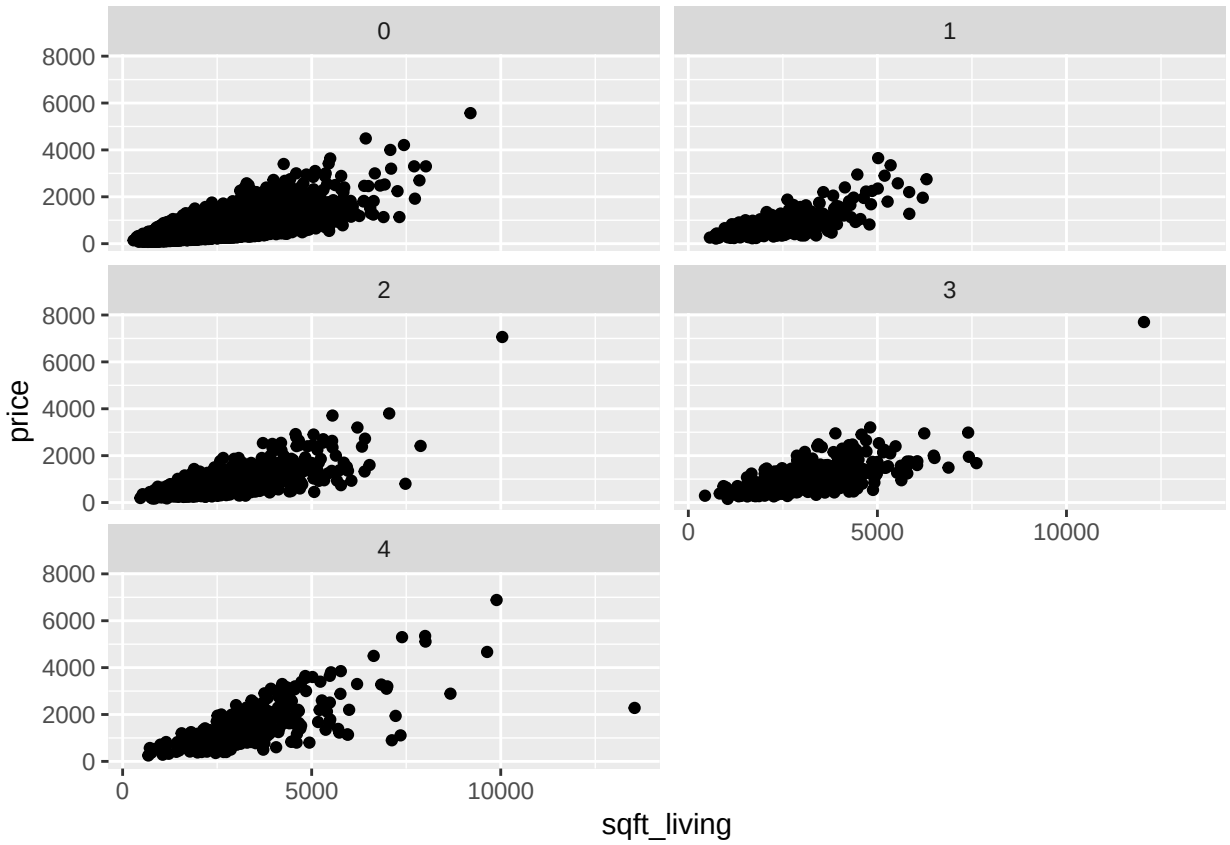

```
ggplot(df, aes(grade, price)) + geom_boxplot()
```



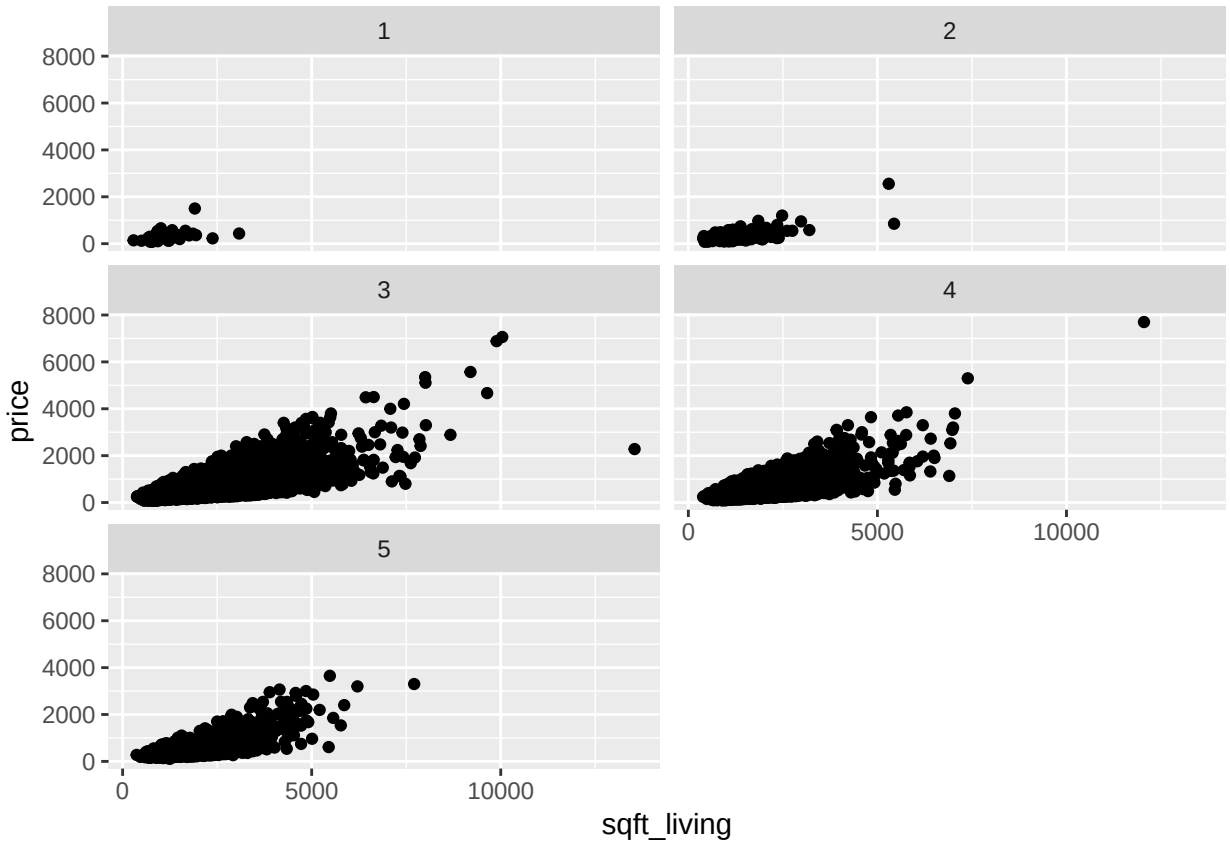

```
plt <- ggplot(df, aes(bedrooms, price)) + geom_point(na.rm = TRUE)
plt + facet_wrap(~grade, ncol = 4)
```



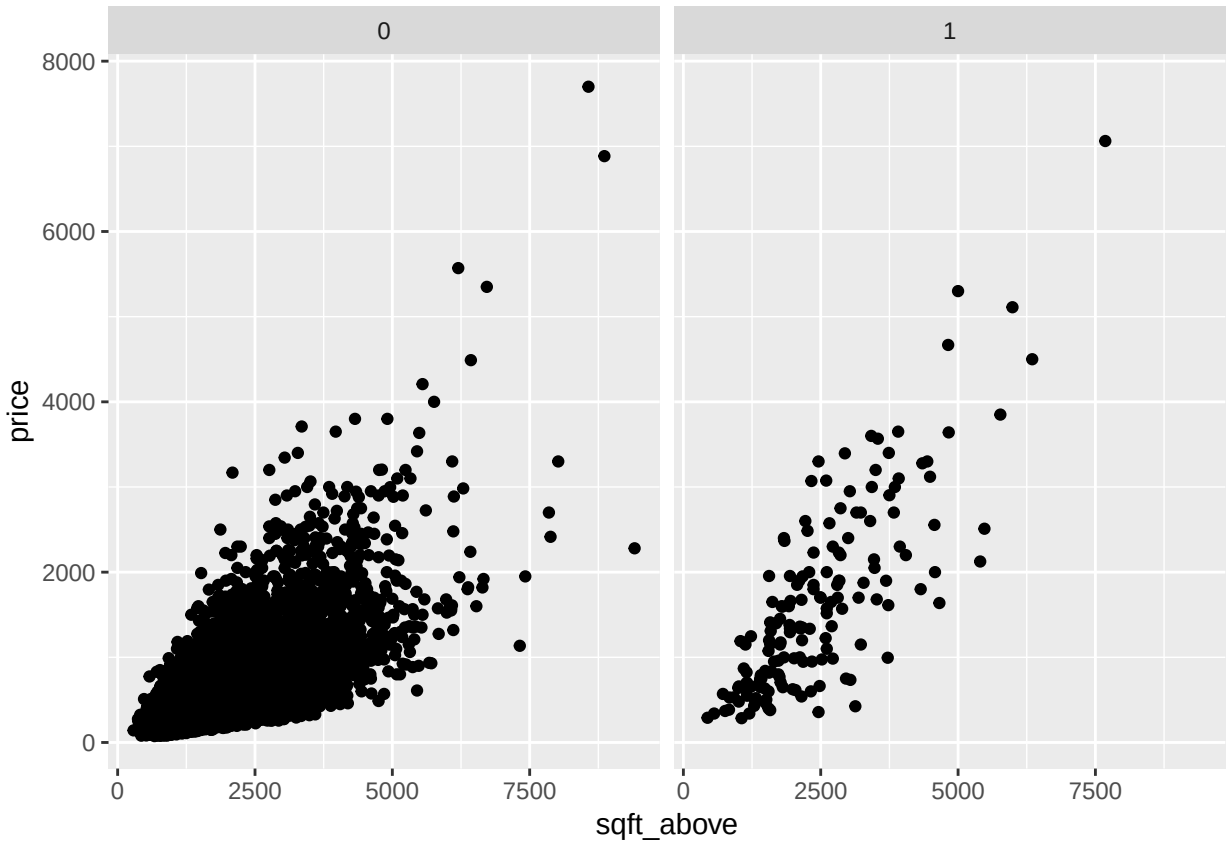

```
plt <- ggplot(df, aes(sqft_living, price)) + geom_point(na.rm = TRUE)
plt + facet_wrap(~view, ncol = 2)
```




```
plt <- ggplot(df, aes(sqft_living, price)) + geom_point(na.rm = TRUE)
plt + facet_wrap(~condition, ncol = 2)
```




```
plt <- ggplot(df, aes(sqft_above, price)) + geom_point(na.rm = TRUE)
plt + facet_wrap(~waterfront, ncol = 2)
```



(1 point) divide the dataframe into Train and Test including in the Train dataset 80% of the observations and 20%, respectively, in Test dataset.

```
set.seed(18)
sample <- sample(nrow(df), floor(nrow(df)) * 0.8)

Train <- df[sample,]
Test <- df[-sample,]
```

(4 points) Build an initial model on Training dataset including as predictors all possible variables and comment on the model performance based on R square and R square Adjusted (which one will you use in this case).

```
model1 <- lm(price~., data = Train)
summary(model1)
```

```
##
## Call:
## lm(formula = price ~ ., data = Train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1552.9   -60.9     2.4    57.0  3594.4
##
## Coefficients: (1 not defined because of singularities)
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -3.505e+04  6.387e+03  -5.488 4.12e-08 ***
## id            -2.559e-10  4.188e-10  -0.611 0.541101
## date           1.176e-01  1.025e-02  11.476  < 2e-16 ***
```

```
## bedrooms      -1.283e+01  1.634e+00  -7.854 4.25e-15 ***
## bathrooms      2.287e+01  2.786e+00   8.207 2.42e-16 ***
## sqft_living    1.182e-01  3.736e-03  31.637  < 2e-16 ***
## sqft_lot       1.949e-04  4.117e-05   4.733 2.23e-06 ***
## floors        -2.712e+01  3.387e+00  -8.006 1.26e-15 ***
## waterfront1    5.846e+02  1.635e+01  35.752  < 2e-16 ***
## view1          8.067e+01  9.652e+00   8.357  < 2e-16 ***
## view2          6.741e+01  5.895e+00  11.434  < 2e-16 ***
## view3          1.557e+02  8.054e+00  19.335  < 2e-16 ***
## view4          2.995e+02  1.189e+01  25.191  < 2e-16 ***
## condition2     8.542e+01  3.377e+01   2.530 0.011430 *
## condition3     9.407e+01  3.148e+01   2.988 0.002809 **
## condition4     1.228e+02  3.150e+01   3.900 9.67e-05 ***
## condition5     1.716e+02  3.169e+01   5.415 6.22e-08 ***
## grade2         4.165e+00  1.791e+02   0.023 0.981451
## grade3        -4.790e+01  1.586e+02  -0.302 0.762568
## grade4        -1.023e+02  1.561e+02  -0.655 0.512384
## grade5        -1.028e+02  1.561e+02  -0.658 0.510266
## grade6        -9.918e+01  1.561e+02  -0.635 0.525197
## grade7        -7.494e+01  1.561e+02  -0.480 0.631264
## grade8        -1.525e+00  1.562e+02  -0.010 0.992212
## grade9         1.257e+02  1.563e+02   0.804 0.421468
## grade10        3.222e+02  1.566e+02   2.057 0.039672 *
## grade11        6.851e+02  1.576e+02   4.347 1.39e-05 ***
## grade12        1.816e+03  1.626e+02  11.170  < 2e-16 ***
## sqft_above     4.960e-02  3.872e-03  12.811  < 2e-16 ***
## sqft_basement         NA         NA      NA       NA
## yr_renovated   3.176e-02  3.102e-03  10.236  < 2e-16 ***
## zipcode98002   1.271e+01  1.498e+01   0.848 0.396290
## zipcode98003  -1.495e+01  1.375e+01  -1.087 0.277080
## zipcode98004   7.171e+02  2.487e+01  28.834  < 2e-16 ***
## zipcode98005   2.488e+02  2.652e+01   9.382  < 2e-16 ***
## zipcode98006   2.110e+02  2.170e+01   9.722  < 2e-16 ***
## zipcode98007   2.223e+02  2.772e+01   8.021 1.12e-15 ***
## zipcode98008   2.316e+02  2.600e+01   8.906  < 2e-16 ***
## zipcode98010   9.145e+01  2.340e+01   3.909 9.32e-05 ***
## zipcode98011   5.540e+01  3.392e+01   1.633 0.102432
## zipcode98014   9.997e+01  3.707e+01   2.697 0.007007 **
## zipcode98019   6.383e+01  3.666e+01   1.741 0.081675 .
## zipcode98022   5.963e+01  2.004e+01   2.975 0.002930 **
## zipcode98023  -4.932e+01  1.242e+01  -3.972 7.15e-05 ***
## zipcode98024   1.675e+02  3.236e+01   5.176 2.29e-07 ***
## zipcode98027   1.585e+02  2.210e+01   7.172 7.70e-13 ***
## zipcode98028   4.349e+01  3.291e+01   1.322 0.186251
## zipcode98029   2.202e+02  2.536e+01   8.686  < 2e-16 ***
## zipcode98030   1.091e+01  1.528e+01   0.714 0.475057
## zipcode98031   1.200e+01  1.576e+01   0.761 0.446377
## zipcode98032  -1.229e+01  1.791e+01  -0.686 0.492518
## zipcode98033   3.005e+02  2.825e+01  10.635  < 2e-16 ***
## zipcode98034   1.265e+02  3.032e+01   4.172 3.04e-05 ***
## zipcode98038   6.652e+01  1.668e+01   3.987 6.72e-05 ***
## zipcode98039   1.160e+03  3.286e+01  35.290  < 2e-16 ***
## zipcode98040   4.577e+02  2.198e+01  20.827  < 2e-16 ***
## zipcode98042   1.875e+01  1.437e+01   1.305 0.192072
```

```
## zipcode98045   1.653e+02  3.081e+01   5.366 8.18e-08 ***
## zipcode98052   1.896e+02  2.885e+01   6.573 5.07e-11 ***
## zipcode98053   1.724e+02  3.082e+01   5.592 2.28e-08 ***
## zipcode98055   2.351e+01  1.748e+01   1.345 0.178592
## zipcode98056   6.075e+01  1.892e+01   3.211 0.001324 **
## zipcode98058   2.572e+01  1.645e+01   1.563 0.118005
## zipcode98059   6.280e+01  1.858e+01   3.380 0.000727 ***
## zipcode98065   1.259e+02  2.846e+01   4.422 9.84e-06 ***
## zipcode98070  -5.958e+01  2.192e+01  -2.719 0.006558 **
## zipcode98072   8.914e+01  3.371e+01   2.644 0.008195 **
## zipcode98074   1.595e+02  2.717e+01   5.872 4.39e-09 ***
## zipcode98075   1.574e+02  2.614e+01   6.023 1.75e-09 ***
## zipcode98077   5.803e+01  3.500e+01   1.658 0.097332 .
## zipcode98092  -7.067e+00  1.352e+01  -0.523 0.601216
## zipcode98102   4.196e+02  2.914e+01  14.403  < 2e-16 ***
## zipcode98103   2.549e+02  2.733e+01   9.326  < 2e-16 ***
## zipcode98105   4.021e+02  2.801e+01  14.355  < 2e-16 ***
## zipcode98106   6.435e+01  2.020e+01   3.186 0.001443 **
## zipcode98107   2.575e+02  2.821e+01   9.127  < 2e-16 ***
## zipcode98108   6.798e+01  2.237e+01   3.038 0.002381 **
## zipcode98109   4.486e+02  2.925e+01  15.335  < 2e-16 ***
## zipcode98112   5.632e+02  2.582e+01  21.812  < 2e-16 ***
## zipcode98115   2.558e+02  2.778e+01   9.208  < 2e-16 ***
## zipcode98116   2.111e+02  2.256e+01   9.357  < 2e-16 ***
## zipcode98117   2.257e+02  2.813e+01   8.025 1.08e-15 ***
## zipcode98118   1.134e+02  1.976e+01   5.739 9.71e-09 ***
## zipcode98119   4.177e+02  2.721e+01  15.353  < 2e-16 ***
## zipcode98122   2.875e+02  2.441e+01  11.778  < 2e-16 ***
## zipcode98125   1.129e+02  3.000e+01   3.765 0.000167 ***
## zipcode98126   1.271e+02  2.102e+01   6.047 1.51e-09 ***
## zipcode98133   6.220e+01  3.101e+01   2.006 0.044888 *
## zipcode98136   1.704e+02  2.140e+01   7.961 1.82e-15 ***
## zipcode98144   2.229e+02  2.277e+01   9.792  < 2e-16 ***
## zipcode98146   3.264e+01  1.903e+01   1.715 0.086344 .
## zipcode98148   3.907e+01  2.641e+01   1.479 0.139042
## zipcode98155   4.661e+01  3.223e+01   1.446 0.148197
## zipcode98166   1.735e+01  1.736e+01   0.999 0.317630
## zipcode98168   7.592e+00  1.830e+01   0.415 0.678323
## zipcode98177   1.128e+02  3.226e+01   3.496 0.000474 ***
## zipcode98178  -8.859e+00  1.904e+01  -0.465 0.641688
## zipcode98188  -5.637e+00  1.967e+01  -0.287 0.774422
## zipcode98198  -2.375e+01  1.465e+01  -1.622 0.104881
## zipcode98199   2.912e+02  2.668e+01  10.916  < 2e-16 ***
## lat            2.116e+02  6.703e+01   3.157 0.001598 **
## long          -1.886e+02  4.713e+01  -4.002 6.31e-05 ***
## sqft_living15  1.367e-02  3.073e-03   4.448 8.71e-06 ***
## sqft_lot15    -5.997e-05  6.512e-05  -0.921 0.357060
## age            2.978e-01  6.996e-02   4.256 2.09e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 151.6 on 17186 degrees of freedom
## Multiple R-squared:  0.8367, Adjusted R-squared:  0.8357
## F-statistic: 854.9 on 103 and 17186 DF,  p-value: < 2.2e-16
```

```
#R-squared adjusted compensates for the additional new variables.
#We should take the R-squared adjusted in this case as we have many variables.
```

(5 points) What variables are signifcant predictors in the model? Comment on the relationships between each independent variable with the dependent variable. (Be attentive in determining the reference group while interpreting the relationships in case of categorical variable)

Date, bedrooms, bathrooms, sqft of living, sqft of land, sqft above, year built, year renovated, latitude, longitude, sqft_living, sqft_lot15 are all significant. Categorical variable is treated as significant if at least one of its categories is significant. All categorical variables: floors, waterfront, view, condition, zipcode and grade are significant as well.

(4 points) Remove the variables you consider might cause multicolinearity, explain the logic how you decide to omit this or that variable from the correlated pairs. Comment on the changes of model performance based on R square and coefficients.

```
cor(df[, c(5:7, 13:15, 20:21)], df[, c(5:7, 13:15, 20:21)]) > 0.7
```

```
##               bathrooms sqft_living sqft_lot sqft_above sqft_basement
## bathrooms          TRUE        TRUE    FALSE      FALSE         FALSE
## sqft_living        TRUE        TRUE    FALSE       TRUE         FALSE
## sqft_lot          FALSE       FALSE     TRUE      FALSE         FALSE
## sqft_above        FALSE        TRUE    FALSE       TRUE         FALSE
## sqft_basement     FALSE       FALSE    FALSE      FALSE          TRUE
## yr_renovated      FALSE       FALSE    FALSE      FALSE         FALSE
## sqft_lot15        FALSE       FALSE     TRUE      FALSE         FALSE
## age               FALSE       FALSE    FALSE      FALSE         FALSE
##               yr_renovated sqft_lot15   age
## bathrooms            FALSE      FALSE FALSE
## sqft_living          FALSE      FALSE FALSE
## sqft_lot             FALSE       TRUE FALSE
## sqft_above           FALSE      FALSE FALSE
## sqft_basement        FALSE      FALSE FALSE
## yr_renovated          TRUE      FALSE FALSE
## sqft_lot15           FALSE       TRUE FALSE
## age                  FALSE      FALSE  TRUE
```

```
#The highly correlated variables were
#1.sqft_living and sqft_above
#2.sqft_living and bathrooms
#3.sqft_living and sqft_living15
#4.sqft_lot and sqft_lot15

cor(df$price, df[, c(5:7, 13:15, 20:21)])
```

```
##      bathrooms sqft_living   sqft_lot sqft_above sqft_basement
## [1,] 0.5251375   0.7020351 0.08966086  0.6055673      0.323816
##      yr_renovated sqft_lot15         age
## [1,]    0.1264338 0.08244715 -0.05401153
```

```
#Sqft_living is just sqft_above+sqft_basement and gives no additional information.
#Sqft_lot15 concerns the neighbors,so it is more logical to omit it
#instead of omitting the land of the house itself.
#Id is not significant.

model2 <- lm(price~.-sqft_living-sqft_lot15-id, data = Train)
summary(model2)
```

```
##
## Call:
## lm(formula = price ~ . - sqft_living - sqft_lot15 - id, data = Train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1552.8   -61.1     2.5    57.0  3593.1
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -3.526e+04  6.379e+03  -5.527 3.30e-08 ***
## date           1.175e-01  1.025e-02  11.469  < 2e-16 ***
## bedrooms      -1.279e+01  1.632e+00  -7.833 5.02e-15 ***
## bathrooms      2.294e+01  2.785e+00   8.237  < 2e-16 ***
## sqft_lot       1.729e-04  3.160e-05   5.472 4.50e-08 ***
## floors        -2.704e+01  3.386e+00  -7.987 1.47e-15 ***
## waterfront1    5.849e+02  1.635e+01  35.778  < 2e-16 ***
## view1          8.076e+01  9.651e+00   8.368  < 2e-16 ***
## view2          6.738e+01  5.894e+00  11.431  < 2e-16 ***
## view3          1.557e+02  8.053e+00  19.334  < 2e-16 ***
## view4          2.992e+02  1.189e+01  25.177  < 2e-16 ***
## condition2     8.543e+01  3.377e+01   2.530 0.011414 *
## condition3     9.390e+01  3.148e+01   2.983 0.002857 **
## condition4     1.226e+02  3.149e+01   3.894 9.88e-05 ***
## condition5     1.714e+02  3.168e+01   5.411 6.37e-08 ***
## grade2         3.529e+00  1.791e+02   0.020 0.984284
## grade3        -4.885e+01  1.585e+02  -0.308 0.758010
## grade4        -1.030e+02  1.561e+02  -0.660 0.509348
## grade5        -1.038e+02  1.561e+02  -0.665 0.505982
## grade6        -1.002e+02  1.561e+02  -0.642 0.520860
## grade7        -7.600e+01  1.561e+02  -0.487 0.626465
## grade8        -2.604e+00  1.562e+02  -0.017 0.986700
## grade9         1.247e+02  1.563e+02   0.798 0.425125
## grade10        3.212e+02  1.566e+02   2.051 0.040269 *
## grade11        6.842e+02  1.576e+02   4.341 1.43e-05 ***
## grade12        1.815e+03  1.625e+02  11.164  < 2e-16 ***
## sqft_above     1.677e-01  3.304e-03  50.763  < 2e-16 ***
## sqft_basement  1.181e-01  3.734e-03  31.625  < 2e-16 ***
## yr_renovated   3.175e-02  3.102e-03  10.238  < 2e-16 ***
## zipcode98002   1.275e+01  1.498e+01   0.851 0.394758
## zipcode98003  -1.501e+01  1.375e+01  -1.091 0.275080
## zipcode98004   7.174e+02  2.487e+01  28.852  < 2e-16 ***
## zipcode98005   2.490e+02  2.652e+01   9.387  < 2e-16 ***
## zipcode98006   2.115e+02  2.170e+01   9.747  < 2e-16 ***
## zipcode98007   2.228e+02  2.771e+01   8.041 9.52e-16 ***
## zipcode98008   2.321e+02  2.600e+01   8.928  < 2e-16 ***
## zipcode98010   9.090e+01  2.339e+01   3.887 0.000102 ***
## zipcode98011   5.597e+01  3.391e+01   1.650 0.098897 .
## zipcode98014   9.909e+01  3.706e+01   2.674 0.007500 **
## zipcode98019   6.401e+01  3.665e+01   1.747 0.080682 .
## zipcode98022   5.917e+01  2.003e+01   2.953 0.003149 **
## zipcode98023  -4.967e+01  1.241e+01  -4.002 6.30e-05 ***
## zipcode98024   1.669e+02  3.234e+01   5.160 2.50e-07 ***
## zipcode98027   1.583e+02  2.210e+01   7.162 8.25e-13 ***
```

```
## zipcode98028   4.385e+01  3.290e+01   1.333 0.182635
## zipcode98029   2.211e+02  2.534e+01   8.724  < 2e-16 ***
## zipcode98030   1.124e+01  1.527e+01   0.736 0.461858
## zipcode98031   1.229e+01  1.576e+01   0.780 0.435504
## zipcode98032  -1.227e+01  1.791e+01  -0.685 0.493303
## zipcode98033   3.011e+02  2.824e+01  10.662  < 2e-16 ***
## zipcode98034   1.269e+02  3.031e+01   4.187 2.84e-05 ***
## zipcode98038   6.687e+01  1.668e+01   4.008 6.14e-05 ***
## zipcode98039   1.160e+03  3.285e+01  35.307  < 2e-16 ***
## zipcode98040   4.581e+02  2.197e+01  20.851  < 2e-16 ***
## zipcode98042   1.890e+01  1.437e+01   1.316 0.188340
## zipcode98045   1.659e+02  3.079e+01   5.389 7.17e-08 ***
## zipcode98052   1.901e+02  2.885e+01   6.591 4.51e-11 ***
## zipcode98053   1.722e+02  3.082e+01   5.588 2.33e-08 ***
## zipcode98055   2.377e+01  1.747e+01   1.361 0.173681
## zipcode98056   6.117e+01  1.891e+01   3.234 0.001222 **
## zipcode98058   2.621e+01  1.645e+01   1.594 0.110998
## zipcode98059   6.322e+01  1.858e+01   3.403 0.000668 ***
## zipcode98065   1.262e+02  2.842e+01   4.440 9.04e-06 ***
## zipcode98070  -6.218e+01  2.169e+01  -2.867 0.004144 **
## zipcode98072   8.939e+01  3.371e+01   2.652 0.008008 **
## zipcode98074   1.600e+02  2.717e+01   5.890 3.93e-09 ***
## zipcode98075   1.580e+02  2.613e+01   6.048 1.49e-09 ***
## zipcode98077   5.815e+01  3.499e+01   1.662 0.096511 .
## zipcode98092  -7.295e+00  1.351e+01  -0.540 0.589329
## zipcode98102   4.201e+02  2.913e+01  14.421  < 2e-16 ***
## zipcode98103   2.550e+02  2.732e+01   9.334  < 2e-16 ***
## zipcode98105   4.023e+02  2.801e+01  14.362  < 2e-16 ***
## zipcode98106   6.441e+01  2.019e+01   3.190 0.001425 **
## zipcode98107   2.579e+02  2.819e+01   9.148  < 2e-16 ***
## zipcode98108   6.808e+01  2.237e+01   3.043 0.002346 **
## zipcode98109   4.492e+02  2.924e+01  15.365  < 2e-16 ***
## zipcode98112   5.635e+02  2.582e+01  21.826  < 2e-16 ***
## zipcode98115   2.559e+02  2.778e+01   9.212  < 2e-16 ***
## zipcode98116   2.110e+02  2.256e+01   9.353  < 2e-16 ***
## zipcode98117   2.259e+02  2.812e+01   8.034 1.01e-15 ***
## zipcode98118   1.137e+02  1.975e+01   5.756 8.77e-09 ***
## zipcode98119   4.180e+02  2.720e+01  15.368  < 2e-16 ***
## zipcode98122   2.873e+02  2.440e+01  11.772  < 2e-16 ***
## zipcode98125   1.131e+02  3.000e+01   3.769 0.000164 ***
## zipcode98126   1.270e+02  2.102e+01   6.041 1.57e-09 ***
## zipcode98133   6.233e+01  3.100e+01   2.010 0.044397 *
## zipcode98136   1.704e+02  2.140e+01   7.961 1.81e-15 ***
## zipcode98144   2.233e+02  2.276e+01   9.810  < 2e-16 ***
## zipcode98146   3.257e+01  1.903e+01   1.712 0.086979 .
## zipcode98148   3.880e+01  2.641e+01   1.469 0.141755
## zipcode98155   4.672e+01  3.223e+01   1.450 0.147124
## zipcode98166   1.738e+01  1.735e+01   1.002 0.316442
## zipcode98168   7.943e+00  1.829e+01   0.434 0.664016
## zipcode98177   1.131e+02  3.225e+01   3.506 0.000456 ***
## zipcode98178  -8.582e+00  1.903e+01  -0.451 0.652083
## zipcode98188  -5.594e+00  1.967e+01  -0.284 0.776073
## zipcode98198  -2.390e+01  1.464e+01  -1.632 0.102687
## zipcode98199   2.916e+02  2.666e+01  10.940  < 2e-16 ***
```

```
## lat            2.112e+02  6.702e+01   3.151 0.001631 **
## long          -1.905e+02  4.706e+01  -4.047 5.21e-05 ***
## sqft_living15  1.349e-02  3.066e-03   4.399 1.09e-05 ***
## age            2.973e-01  6.996e-02   4.250 2.15e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 151.6 on 17188 degrees of freedom
## Multiple R-squared:  0.8367, Adjusted R-squared:  0.8357
## F-statistic: 871.9 on 101 and 17188 DF,  p-value: < 2.2e-16
```

(3 points) Try changing the reference group for grade variable to be high,(use the function ?relevel) run and save the 3rd model. Comment on the changes of coefficients, their significance and the overall model performance.

```
df$grade <- relevel(df$grade, ref = 12)
model3 <- lm(price~.-sqft_living-sqft_lot15-id, data = Train)
summary(model3)
```

```
##
## Call:
## lm(formula = price ~ . - sqft_living - sqft_lot15 - id, data = Train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1552.8   -61.1     2.5    57.0  3593.1
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -3.526e+04  6.379e+03  -5.527 3.30e-08 ***
## date           1.175e-01  1.025e-02  11.469  < 2e-16 ***
## bedrooms      -1.279e+01  1.632e+00  -7.833 5.02e-15 ***
## bathrooms      2.294e+01  2.785e+00   8.237  < 2e-16 ***
## sqft_lot       1.729e-04  3.160e-05   5.472 4.50e-08 ***
## floors        -2.704e+01  3.386e+00  -7.987 1.47e-15 ***
## waterfront1    5.849e+02  1.635e+01  35.778  < 2e-16 ***
## view1          8.076e+01  9.651e+00   8.368  < 2e-16 ***
## view2          6.738e+01  5.894e+00  11.431  < 2e-16 ***
## view3          1.557e+02  8.053e+00  19.334  < 2e-16 ***
## view4          2.992e+02  1.189e+01  25.177  < 2e-16 ***
## condition2     8.543e+01  3.377e+01   2.530 0.011414 *
## condition3     9.390e+01  3.148e+01   2.983 0.002857 **
## condition4     1.226e+02  3.149e+01   3.894 9.88e-05 ***
## condition5     1.714e+02  3.168e+01   5.411 6.37e-08 ***
## grade2         3.529e+00  1.791e+02   0.020 0.984284
## grade3        -4.885e+01  1.585e+02  -0.308 0.758010
## grade4        -1.030e+02  1.561e+02  -0.660 0.509348
## grade5        -1.038e+02  1.561e+02  -0.665 0.505982
## grade6        -1.002e+02  1.561e+02  -0.642 0.520860
## grade7        -7.600e+01  1.561e+02  -0.487 0.626465
## grade8        -2.604e+00  1.562e+02  -0.017 0.986700
## grade9         1.247e+02  1.563e+02   0.798 0.425125
## grade10        3.212e+02  1.566e+02   2.051 0.040269 *
## grade11        6.842e+02  1.576e+02   4.341 1.43e-05 ***
## grade12        1.815e+03  1.625e+02  11.164  < 2e-16 ***
```

```
## sqft_above     1.677e-01  3.304e-03  50.763  < 2e-16 ***
## sqft_basement  1.181e-01  3.734e-03  31.625  < 2e-16 ***
## yr_renovated   3.175e-02  3.102e-03  10.238  < 2e-16 ***
## zipcode98002   1.275e+01  1.498e+01   0.851 0.394758
## zipcode98003  -1.501e+01  1.375e+01  -1.091 0.275080
## zipcode98004   7.174e+02  2.487e+01  28.852  < 2e-16 ***
## zipcode98005   2.490e+02  2.652e+01   9.387  < 2e-16 ***
## zipcode98006   2.115e+02  2.170e+01   9.747  < 2e-16 ***
## zipcode98007   2.228e+02  2.771e+01   8.041 9.52e-16 ***
## zipcode98008   2.321e+02  2.600e+01   8.928  < 2e-16 ***
## zipcode98010   9.090e+01  2.339e+01   3.887 0.000102 ***
## zipcode98011   5.597e+01  3.391e+01   1.650 0.098897 .
## zipcode98014   9.909e+01  3.706e+01   2.674 0.007500 **
## zipcode98019   6.401e+01  3.665e+01   1.747 0.080682 .
## zipcode98022   5.917e+01  2.003e+01   2.953 0.003149 **
## zipcode98023  -4.967e+01  1.241e+01  -4.002 6.30e-05 ***
## zipcode98024   1.669e+02  3.234e+01   5.160 2.50e-07 ***
## zipcode98027   1.583e+02  2.210e+01   7.162 8.25e-13 ***
## zipcode98028   4.385e+01  3.290e+01   1.333 0.182635
## zipcode98029   2.211e+02  2.534e+01   8.724  < 2e-16 ***
## zipcode98030   1.124e+01  1.527e+01   0.736 0.461858
## zipcode98031   1.229e+01  1.576e+01   0.780 0.435504
## zipcode98032  -1.227e+01  1.791e+01  -0.685 0.493303
## zipcode98033   3.011e+02  2.824e+01  10.662  < 2e-16 ***
## zipcode98034   1.269e+02  3.031e+01   4.187 2.84e-05 ***
## zipcode98038   6.687e+01  1.668e+01   4.008 6.14e-05 ***
## zipcode98039   1.160e+03  3.285e+01  35.307  < 2e-16 ***
## zipcode98040   4.581e+02  2.197e+01  20.851  < 2e-16 ***
## zipcode98042   1.890e+01  1.437e+01   1.316 0.188340
## zipcode98045   1.659e+02  3.079e+01   5.389 7.17e-08 ***
## zipcode98052   1.901e+02  2.885e+01   6.591 4.51e-11 ***
## zipcode98053   1.722e+02  3.082e+01   5.588 2.33e-08 ***
## zipcode98055   2.377e+01  1.747e+01   1.361 0.173681
## zipcode98056   6.117e+01  1.891e+01   3.234 0.001222 **
## zipcode98058   2.621e+01  1.645e+01   1.594 0.110998
## zipcode98059   6.322e+01  1.858e+01   3.403 0.000668 ***
## zipcode98065   1.262e+02  2.842e+01   4.440 9.04e-06 ***
## zipcode98070  -6.218e+01  2.169e+01  -2.867 0.004144 **
## zipcode98072   8.939e+01  3.371e+01   2.652 0.008008 **
## zipcode98074   1.600e+02  2.717e+01   5.890 3.93e-09 ***
## zipcode98075   1.580e+02  2.613e+01   6.048 1.49e-09 ***
## zipcode98077   5.815e+01  3.499e+01   1.662 0.096511 .
## zipcode98092  -7.295e+00  1.351e+01  -0.540 0.589329
## zipcode98102   4.201e+02  2.913e+01  14.421  < 2e-16 ***
## zipcode98103   2.550e+02  2.732e+01   9.334  < 2e-16 ***
## zipcode98105   4.023e+02  2.801e+01  14.362  < 2e-16 ***
## zipcode98106   6.441e+01  2.019e+01   3.190 0.001425 **
## zipcode98107   2.579e+02  2.819e+01   9.148  < 2e-16 ***
## zipcode98108   6.808e+01  2.237e+01   3.043 0.002346 **
## zipcode98109   4.492e+02  2.924e+01  15.365  < 2e-16 ***
## zipcode98112   5.635e+02  2.582e+01  21.826  < 2e-16 ***
## zipcode98115   2.559e+02  2.778e+01   9.212  < 2e-16 ***
## zipcode98116   2.110e+02  2.256e+01   9.353  < 2e-16 ***
## zipcode98117   2.259e+02  2.812e+01   8.034 1.01e-15 ***
```

```
## zipcode98118   1.137e+02  1.975e+01   5.756 8.77e-09 ***
## zipcode98119   4.180e+02  2.720e+01  15.368  < 2e-16 ***
## zipcode98122   2.873e+02  2.440e+01  11.772  < 2e-16 ***
## zipcode98125   1.131e+02  3.000e+01   3.769 0.000164 ***
## zipcode98126   1.270e+02  2.102e+01   6.041 1.57e-09 ***
## zipcode98133   6.233e+01  3.100e+01   2.010 0.044397 *
## zipcode98136   1.704e+02  2.140e+01   7.961 1.81e-15 ***
## zipcode98144   2.233e+02  2.276e+01   9.810  < 2e-16 ***
## zipcode98146   3.257e+01  1.903e+01   1.712 0.086979 .
## zipcode98148   3.880e+01  2.641e+01   1.469 0.141755
## zipcode98155   4.672e+01  3.223e+01   1.450 0.147124
## zipcode98166   1.738e+01  1.735e+01   1.002 0.316442
## zipcode98168   7.943e+00  1.829e+01   0.434 0.664016
## zipcode98177   1.131e+02  3.225e+01   3.506 0.000456 ***
## zipcode98178  -8.582e+00  1.903e+01  -0.451 0.652083
## zipcode98188  -5.594e+00  1.967e+01  -0.284 0.776073
## zipcode98198  -2.390e+01  1.464e+01  -1.632 0.102687
## zipcode98199   2.916e+02  2.666e+01  10.940  < 2e-16 ***
## lat            2.112e+02  6.702e+01   3.151 0.001631 **
## long          -1.905e+02  4.706e+01  -4.047 5.21e-05 ***
## sqft_living15  1.349e-02  3.066e-03   4.399 1.09e-05 ***
## age            2.973e-01  6.996e-02   4.250 2.15e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 151.6 on 17188 degrees of freedom
## Multiple R-squared:  0.8367, Adjusted R-squared:  0.8357
## F-statistic: 871.9 on 101 and 17188 DF,  p-value: < 2.2e-16
```

```
#As the reference group for the grade variable is now the highest,
#the coefficients all become negative.
#Now all the groups are significant.
#The overall performance (R squared and R squared adjusted) does not change.

df$grade <- relevel(df$grade, ref = 1)
```

(3 points) Make predictions on the testing data set using all 3 models, calculate RMSE and comment what model is doing better

```
predictModel<-predict(model1, newdata = Test)
RMSE1<-sqrt(mean((predictModel-Test$price)^2))
RMSE1
```

```
## [1] 136.0831
```

```
model2 <- lm(price~.-sqft_lot-sqft_lot15, data = Train)
predictModel<-predict(model2, newdata = Test)
RMSE2<-sqrt(mean((predictModel-Test$price)^2))
RMSE2
```

```
## [1] 136.4634
```

```
predictModel<-predict(model3, newdata = Test)
RMSE3<-sqrt(mean((predictModel-Test$price)^2))
RMSE3
```

```
## [1] 136.1381
```

```
#According to the RMSE-s, the first model is doing the best.
#However, there is multicollinearity issues among its variables and
#some coefficients do not make sense.
#However, the overall performance difference is not big.
```