King County House Sales
Regression Analysis

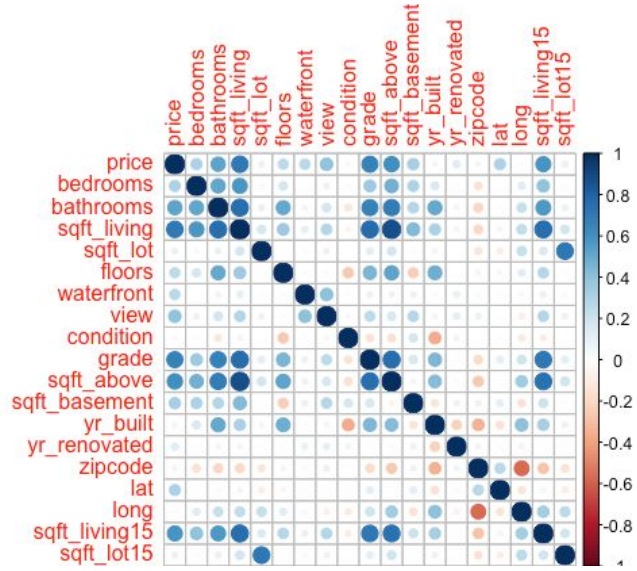
Brent Knight

Abstract

This study investigates how housing prices for King County is determined by explanatory variables. The eighteen explanatory variables are: the number of bedrooms, number of bathrooms, square feet of the living area, square feet of the lot, number of floors, whether the house is a waterfront property, the level of the view, condition, grade, the square feet above the ground, the square feet of the basement, the year built, year renovated, the zip code, latitude, longitude, the square feet of the living area in 2015, and the square feet of the lot in 2015. The data is made up of 21613 randomly selected samples of houses in King County. The assumptions made are that the housing prices, or response variables, are normally distributed and independent. From the normality assumption, a linear model is used in the form of a linear regression to fit the data. The final model has thirteen explanatory variables: the number of bedrooms, number of bathrooms, square feet of the living area, square feet of the lot, number of floors, whether the house is a waterfront property, the level of the view, condition, grade, year built, year renovated, the zip code, and the square feet of the living area in 2015.

Method

Before a model is fitted, I used a preliminary analysis to gain insight of possible types of relationships between individual explanatory variables and the response variable. The figure below shows these relationships with the darker color showing a stronger relationship between the corresponding variables. This graph indicates relatively strong relationships, so I begin the analysis with an initial model.



To begin the model selection process, the initial model:

$$Price_i=\beta_0+\beta_1 bedrooms_i+\beta_2 bathrooms_i+\beta_3 sqft\_living_i+\beta_4 sqft\_lot_i+\beta_5 floors_i+\beta_6 waterfront_i+\beta_7 view_i+\beta_8 condition_i+\beta_9 grade_i+\beta_{10} sqft\_above_i+\beta_{11} sqft\_basement_i+\beta_{12} yr\_built_i+\beta_{13} yr\_renovated_i+\beta_{14}\ zip+\beta_{15} sqft\_living15+\beta_{16} sqft\_lot15+\beta_{17} lat+\beta_{18} long+\varepsilon_i$$

where $i = 1, \ldots, 21613$

β_0 is the intercept of the model and $\beta_1,.. \beta_{18}$ are the effect coefficients for each explanatory variable's contribution to property price. ε_i is the additive random error term that is normally distributed with mean 0 and has constant variance. This initial model is the starting point of the model selection process. Then, diagnostics are used to show how well the model fits the data and if any other models are suggested.

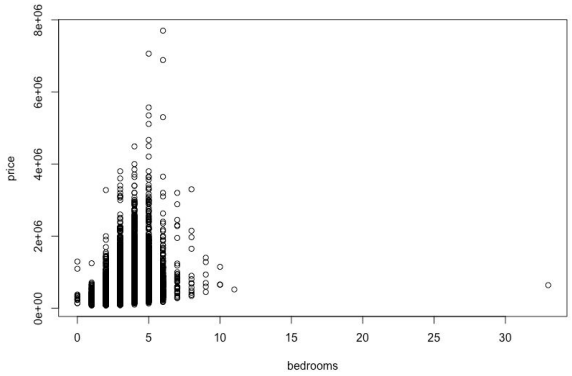
A residual is the difference between the observed value and fitted value from the proposed model. Various plots involving residuals can help detect outliers, violations of assumptions, and if any transformations are needed. For example, the residual versus fitted values plot is used to verify that the error is constant with mean 0. Cook's distance versus the leverage (ability to change fit of model) of the same observation can indicate if I should pay special attention to some influential points. One way to deal with an influential or questionable point is to exclude it from the model if it has considerable effect on the coefficients of the model, or equivalently, the fit of the model.

Model and variable selection are also important procedures when determining a proper model to fit the data. The goal is to build the simplest model that best explains property prices. I chose to use both the Akaike information criterion (AIC) and the Bayesian information Criterion (BIC) to select the simplest model. These tests are a measure of the relative quality of statistical models for a given set of data starting with candidate models, and then finding the models' corresponding AIC and BIC values. There will almost always be information lost due to using a candidate model to represent the "true" model (i.e. the process that generates the data). It is important to select, from among the candidate models, the model that minimizes the information loss.

Diagnostics should be performed on each proposed model to assure the assumptions remain valid under the different models and to keep track of influential points. In the next section, I will be discussing the final model after delineation of other fits.

Statistical Analysis

The statistical analysis was done with the statistical programming software, R. In the methods section, the initial model included all eighteen explanatory variables to predict the sale price. Before running variable selection methods in R, it is important to look for influential outliers and point with bad high leverage since they can affect the accuracy of these selection methods. As seen in price vs bedrooms graph below, there is an observation with 33 bedrooms. This outlier is obviously a typo considering it to be highly unlikely for a house to have 33 bedrooms with 1.75 bathrooms. This observation was removed from the data set.



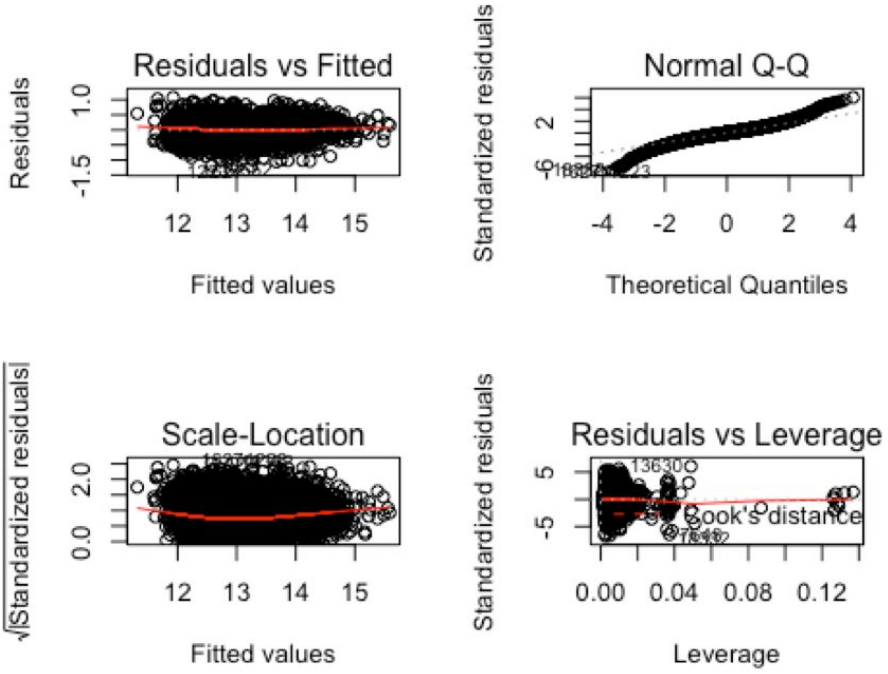
Multicollinearity, when two or more predictor variables in a multiple linear regression model are highly correlated, is another important topic to consider when selecting predictor variables for the final model. In response to small changes in the model, coefficient estimates of the multiple regression may change erratically. So, it is imperative that I consider removing variables with high correlation from the final model. Also, a simpler model with fewer variables that has the same predictive power as that of a large number of variables is preferred. After running the summary function for the initial model including all eighteen variables, the predictor variable "basement" had the term "NA" for all of its statistics. Through careful consideration, it is shown that sqft_basement is actual a linear combination of sqft_living – sqft_above. The information contained in "basement" was already found in this linear combination therefore making it redundant and unnecessary. Basement was excluded from the final model. In a similar vein, the variables longitude and latitude were excluded from the final model since zipcode contains the same information. The last variable excluded from the final model before running variable

selection methods in R was sqft_above. Sqft_above has a high correlation with sqft_living as seen in the first table, so I excluded one of the two. Multicollinearity will be visited again later on in the analysis in order to check the validity of the model.

Transformations were also used in order for the data to be analyzed and presented more clearly. First, the log transformation of price was considered without any transformations to the predictor variables. The scatterplot against the fitted values looked logarithmic instead of linear. Also, by looking at the several Standard Residual graphs, there needed to be a transformation since one third of the points were out of the interval on the graph. The final decision was to take log(price), log(sqft_living), and log(sqft_living15) in the final model. Also, by looking at the box plots of Price vs each dummy variable there wasn't linear trends in all of them showing to be problematic. In order to counteract this, categorical variables were used for the predictor variables of floors, waterfront, condition, and zipcode.

Finally, the variable selection methods AIC (Akaike information criterion) and BIC (Bayesian information criterion) are used to find out if any other variables need to be excluded. After running both forwards and backwards AIC/BIC on a newer model excluding the variables talked about in the previous paragraphs, the variable sqft_lot15 was excluded from the model.

Now that the predictor variables for the final model have been selected, a series of checks must be run in order to make sure the model is valid. Most importantly, the diagnostic plots must be checked. In the plots below, the upper left plot has equally spread residuals around a horizontal line centered at zero without any distinct patterns. The upper right plot shows a linear one to one relationship with little skew on the top tail and heavy skew on the bottom. A linear one to one relationship is looked for in valid linear models to show that the residuals are normally distributed. The lower left plot is used to check the assumption of constant variance. It is imperative that there is a horizontal line with equally spread points which is something this lower left plot shows. Finally, the lower left plot does not contain bad high leverage points past the Cook's distance showing that there aren't any individual observations heavily skewing my regression.



Other checks to include are the Adjusted R squared, ANOVA, and VIF. Using the Summary function in R, I found the Adjusted R squared to be 0.8794. Measured from 0 to 1, the higher Adjusted R squared the better. All the predictor variables are statistically significant looking at the P-values in both the R summary and the ANOVA function. Variance Inflation Factor or VIF quantifies the severity of multicollinearity. Any predictor variable with a VIF greater than 5 is said to be influenced by multicollinearity. After running the VIF function, all of my predictor variables, with the exception of some categorical variables, pass well under a VIF of 5.

The final model:

$$\log(\text{Price}_i)=\beta_0+\beta_1\text{bedrooms}_i+\beta_2\text{bathrooms}_i+\beta_3\log(\text{sqft\_living}_i)+\beta_4\text{sqft\_lot}_i+\beta_5\text{floors}_i+\beta_6\text{waterfront}_i+\beta_7\text{view}_i+\beta_8\text{condition}_i+\beta_9\text{grade}_i+\beta_{10}\text{yr\_built}_i+\beta_{11}\text{yr\_renovated}_i+\beta_{12}\,\text{zip}+\beta_{13}\log(\text{sqft\_living15})+\varepsilon_i$$

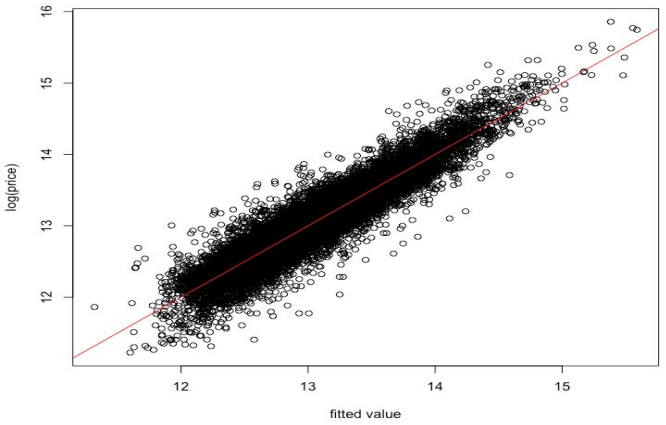
where $i = 1, \ldots, 21612$

Conclusion

The explanatory variables that best explain prices of listed properties are the number of bedrooms, number of bathrooms, square feet of the living area, square feet of the lot, number of

floors, whether the house is a waterfront property, the level of the view, condition, grade, year built, year renovated, the zip code, and the square feet of the living area in 2015. When stating the effect of a variable in determining the price of a property, the other explanatory variables are considered to be held constant. Since I used categorical variables for the predictor variables floor, zip code, waterfront, and condition, there are many coefficients for these predictor variables. These coefficients can be found in the final R-summary for the final model in the appendix. The other nine variables are numerical variables. The effect coefficient for bathrooms was .03115, indicating that for every increase of one bathroom, the price of the property will increase by .03155%. The effect coefficient for bedrooms was .0138, indicating that for every increase of one bedroom, the price of the property will increase by .0138%. The effect coefficient for the year built was .0007, indicating that for every increase of one year built, the price of the property will increase by .0007%. The effect coefficient for year renovated was .00003, indicating that for every one increase of the year renovated, the price of the property will increase by .00001%. The effect coefficient for sqft_lot was .0000007, indicating that for every increase of one square foot of the lot, the price of the property will increase by .0000007%. The effect coefficient for view was .0582, indicating that for every increase of one on the view rating, the price of the property will increase by .0582%. The effect coefficient for grade was .105, indicating that for every increase of 1 on the grade scale, the price of the property will increase by .105%. The effect coefficient for log(sqft_living) was .3889, indicating that for every 1% increase of square feet of the living area, the price of the property will increase by .3889%. The effect coefficient for log(sqft_living15) was .181, indicating that for every 1% increase of the square feet of the living area in 2015, the price of the property will increase by .181%. In a linear regression, the zip code variable was better off being a categorical variable, however there may be other non-linear models that make better use of the zip code information.

The final scatterplot of log(price) vs fitted values:



Appendix

```
setwd("~/desktop")
house = read.csv("kc_house_data.csv")
attach(house)

##clean the data
par(mfrow=c(3,3))
plot(bedrooms,price)
plot(bathrooms,price)
plot(sqft_living,price)
plot(sqft_lot,price)
plot(floors,price)
plot(waterfront,price)
plot(view,price)
plot(condition,price)
plot(grade,price)
plot(sqft_above,price)
plot(sqft_basement, price)
plot(yr_built,price)
plot(yr_renovated,price)
plot(zipcode, price)
plot(lat,price)
plot(long,price)
plot(sqft_living15,price)
plot(sqft_lot15,price)

corH=cor(house[, 3:21])
```

```
corH
library(corrplot)
par(mfrow=c(1,1))
corrplot(corH, type="full", method = "circle")

par(mfrow=c(2,2))
boxplot(price~waterfront, main ="waterfront" )
boxplot(price~floors, main ="floors")
boxplot(price~view, main ="view")
boxplot(price~condition, main ="condition")
boxplot(price~grade, main ="grade")
boxplot(price~yr_built, main ="yr_built")
boxplot(price~yr_renovated, main ="yr_renovated")
boxplot(price~zipcode, main ="zipcode")

## notice the bedroom plot
sub = subset(house, house$bedrooms > 10)
sub
##sqft_living is very low and have 1.75 bathrooms. delete
house = house[-15871, ]

## waterfront, floors, condition, zipcode as categorical
## grade and view still numeric, cause price increases with x
house$waterfront = as.factor(house$waterfront)
house$floors = as.factor(house$floors)
house$condition = as.factor(house$condition)
house$zipcode = as.factor(house$zipcode)
```

```
##model 1
attach(house)
fit.1 = lm(price~.-date-id, data = house)
summary(fit.1)
StanRes1 <- rstandard(fit.1)
par(mfrow=c(4,4))
plot(bedrooms,StanRes1, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(bathrooms,StanRes1, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(sqft_living,StanRes1, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(sqft_lot,StanRes1, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(floors,StanRes1, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(waterfront,StanRes1, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(view,StanRes1, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(condition,StanRes1, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(grade,StanRes1, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(sqft_above,StanRes1, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(sqft_basement,StanRes1, ylab="Standardized Residuals")
```

```
abline(h=c(-4,0,4),col="red", lty=2)
plot(yr_built,StanRes1, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(yr_renovated,StanRes1, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(zipcode,StanRes1, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(sqft_living15,StanRes1, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(sqft_lot15,StanRes1, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(lat,StanRes1, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(long,StanRes1, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(fit.1$fitted.values,StanRes1, ylab="Standardized
Residuals")
abline(h=c(-4,0,4),col="red", lty=2)

##model 2
par(mfrow=c(3,3))
plot(bedrooms,log(price))
plot(bathrooms,log(price))
plot(sqft_living,log(price))
plot(sqft_lot,log(price))
plot(floors,log(price))
plot(waterfront,log(price))
plot(view,log(price))
plot(condition,log(price))
```

```
plot(grade,log(price))
plot(sqft_above,log(price))
plot(yr_built,log(price))
plot(yr_renovated,log(price))
plot(zipcode, log(price))
plot(lat,log(price))
plot(long,log(price))
plot(sqft_living15,log(price))
plot(sqft_lot15,log(price))

fit.2 = lm(log(price)~.-date-id-lat-long-sqft_basement, data =
house)
summary(fit.2)

StanRes2 <- rstandard(fit.2)
par(mfrow=c(1,1))
plot(bedrooms,StanRes2, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(bathrooms,StanRes2, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(sqft_living,StanRes2, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(sqft_lot,StanRes2, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(floors,StanRes2, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(waterfront,StanRes2, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(view,StanRes2, ylab="Standardized Residuals")
```

```
abline(h=c(-4,0,4),col="red", lty=2)
plot(condition,StanRes2, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(grade,StanRes2, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(sqft_above,StanRes2, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(yr_built,StanRes2, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(yr_renovated,StanRes2, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(zipcode,StanRes2, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(sqft_living15,StanRes2, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(sqft_lot15,StanRes2, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(fit.1$fitted.values,StanRes2, ylab="Standardized
Residuals")
abline(h=c(-4,0,4),col="red", lty=2)

##model 3
par(mfrow=c(1,1))
plot(log(sqft_living),log(price))
plot(log(sqft_above),log(price))
plot(log(sqft_living15),log(price))

## I delete the sqft_living, sqft_above and sqft_living15,
because I want to do the log-transformation of those variables.
```

```
fit.3 =
lm(log(price)~.-date-id-lat-long-sqft_basement-sqft_living-sqft_
above-sqft_living15+log(sqft_living)+
             log(sqft_above)+log(sqft_living15), data = house)
summary(fit.3)

StanRes3 <- rstandard(fit.3)
par(mfrow=c(2,2))
plot(bedrooms,StanRes3, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(bathrooms,StanRes3, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(log(sqft_living),StanRes3, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(sqft_lot,StanRes3, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(floors,StanRes3, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(waterfront,StanRes3, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(view,StanRes3, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(condition,StanRes3, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(grade,StanRes3, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(log(sqft_above),StanRes3, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(yr_built,StanRes3, ylab="Standardized Residuals")
```

```
abline(h=c(-4,0,4),col="red", lty=2)
plot(yr_renovated,StanRes3, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(zipcode,StanRes3, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(log(sqft_living15),StanRes3, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(sqft_lot15,StanRes3, ylab="Standardized Residuals")
abline(h=c(-4,0,4),col="red", lty=2)
plot(fit.1$fitted.values,StanRes3, ylab="Standardized
Residuals")
abline(h=c(-4,0,4),col="red", lty=2)

n = length(house$price)
aic = step(lm(log(price)~1),
log(price)~bedrooms+bathrooms+log(sqft_living)+sqft_lot+floors+w
aterfront+

view+condition+grade+log(sqft_above)+yr_built+yr_renovated+zipco
de+log(sqft_living15)+sqft_lot15, direction = "both")
bic = step(lm(log(price)~1),
log(price)~bedrooms+bathrooms+log(sqft_living)+sqft_lot+floors+w
aterfront+

view+condition+grade+log(sqft_above)+yr_built+yr_renovated+zipco
de+log(sqft_living15)+sqft_lot15, direction = "both", k =
log(n))

##model 4, according to the output of bic in model 3
```

```
fit.4 = lm(log(price) ~ zipcode + log(sqft_living) + grade +
view + waterfront +
             log(sqft_living15) + condition + sqft_lot +
log(sqft_above) +
             floors + yr_renovated + bathrooms + bedrooms +
yr_built)
summary(fit.4)


##model 5, delete the sqft_above, for the high correlation.
fit.5 = lm(log(price) ~ zipcode + log(sqft_living) + grade +
view + waterfront +
             log(sqft_living15) + condition + sqft_lot +
             floors + yr_renovated + bathrooms + bedrooms +
yr_built+sqft_lot15)
summary(fit.5)
aic = step(lm(log(price)~1),
log(price)~bedrooms+bathrooms+log(sqft_living)+sqft_lot+floors+w
aterfront+

view+condition+grade+yr_built+yr_renovated+zipcode+log(sqft_livi
ng15)+sqft_lot15, direction = "both")
bic = step(lm(log(price)~1),
log(price)~bedrooms+bathrooms+log(sqft_living)+sqft_lot+floors+w
aterfront+

view+condition+grade+yr_built+yr_renovated+zipcode+log(sqft_livi
ng15)+sqft_lot15, direction = "both", k = log(n))
```

```
##model 6, according to the output of bic in model 5
fit.6 = lm(log(price) ~ zipcode + log(sqft_living) + grade +
view + waterfront +
             log(sqft_living15) + condition + sqft_lot +
             floors + yr_renovated + bathrooms + bedrooms +
yr_built)
summary(fit.6)
par(mfrow=c(2,2))
plot(fit.6)
par(mfrow=c(1,1))
plot(log(price) ~ fit.6$fitted.values, xlab = "fitted value")
abline(lm(log(price)~fit.6$fitted.value), col = "red")
anova(fit.6)

##This is the R summary for the fit.6.
## Call:
## lm(formula = log(price) ~ zipcode + log(sqft_living) + grade
+
##   view + waterfront + log(sqft_living15) + condition +
sqft_lot +
##   floors + yr_renovated + bathrooms + bedrooms + yr_built)
##
## Residuals:
##   Min       1Q   Median    3Q   Max
## -1.23083 -0.09893  0.00289  0.10253  1.07957
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         8.578e+00  1.671e-01  51.330  < 2e-16 ***
```

```
## zipcode98002       -4.331e-03  1.620e-02  -0.267 0.789254
## zipcode98003        7.274e-03  1.458e-02   0.499 0.617835
## zipcode98004        1.115e+00  1.431e-02  77.910  < 2e-16 ***
## zipcode98005        7.191e-01  1.727e-02  41.630  < 2e-16 ***
## zipcode98006        6.253e-01  1.289e-02  48.510  < 2e-16 ***
## zipcode98007        6.331e-01  1.824e-02  34.700  < 2e-16 ***
## zipcode98008        6.360e-01  1.461e-02  43.538  < 2e-16 ***
## zipcode98010        2.588e-01  2.073e-02  12.483  < 2e-16 ***
## zipcode98011        4.292e-01  1.629e-02  26.342  < 2e-16 ***
## zipcode98014        3.130e-01  1.924e-02  16.264  < 2e-16 ***
## zipcode98019        3.175e-01  1.646e-02  19.292  < 2e-16 ***
## zipcode98022        4.306e-02  1.553e-02   2.774 0.005550 **
## zipcode98023       -3.701e-02  1.265e-02  -2.925 0.003454 **
## zipcode98024        4.502e-01  2.268e-02  19.851  < 2e-16 ***
## zipcode98027        4.927e-01  1.326e-02  37.144  < 2e-16 ***
## zipcode98028        3.995e-01  1.454e-02  27.470  < 2e-16 ***
## zipcode98029        5.713e-01  1.415e-02  40.367  < 2e-16 ***
## zipcode98030        4.595e-02  1.495e-02   3.074 0.002113 **
## zipcode98031        6.299e-02  1.466e-02   4.296 1.75e-05 ***
## zipcode98032       -2.872e-02  1.901e-02  -1.510 0.130998
## zipcode98033        7.734e-01  1.310e-02  59.028  < 2e-16 ***
## zipcode98034        5.294e-01  1.243e-02  42.589  < 2e-16 ***
## zipcode98038        1.610e-01  1.228e-02  13.108  < 2e-16 ***
## zipcode98039        1.307e+00  2.788e-02  46.895  < 2e-16 ***
## zipcode98040        8.603e-01  1.485e-02  57.919  < 2e-16 ***
## zipcode98042        6.559e-02  1.241e-02   5.285 1.27e-07 ***
## zipcode98045        3.361e-01  1.567e-02  21.445  < 2e-16 ***
## zipcode98052        6.193e-01  1.237e-02  50.086  < 2e-16 ***
## zipcode98053        5.941e-01  1.337e-02  44.439  < 2e-16 ***
```

```
## zipcode98055       1.343e-01  1.477e-02   9.098  < 2e-16 ***
## zipcode98056       3.205e-01  1.325e-02  24.182  < 2e-16 ***
## zipcode98058       1.555e-01  1.290e-02  12.048  < 2e-16 ***
## zipcode98059       3.417e-01  1.287e-02  26.560  < 2e-16 ***
## zipcode98065       3.971e-01  1.428e-02  27.803  < 2e-16 ***
## zipcode98070       3.063e-01  1.985e-02  15.430  < 2e-16 ***
## zipcode98072       4.783e-01  1.474e-02  32.454  < 2e-16 ***
## zipcode98074       5.439e-01  1.314e-02  41.408  < 2e-16 ***
## zipcode98075       5.559e-01  1.387e-02  40.069  < 2e-16 ***
## zipcode98077       4.575e-01  1.638e-02  27.924  < 2e-16 ***
## zipcode98092       1.674e-02  1.374e-02   1.218 0.223305
## zipcode98102       9.108e-01  2.067e-02  44.061  < 2e-16 ***
## zipcode98103       8.001e-01  1.273e-02  62.846  < 2e-16 ***
## zipcode98105       8.996e-01  1.581e-02  56.893  < 2e-16 ***
## zipcode98106       3.406e-01  1.400e-02  24.325  < 2e-16 ***
## zipcode98107       8.182e-01  1.517e-02  53.953  < 2e-16 ***
## zipcode98108       3.348e-01  1.660e-02  20.163  < 2e-16 ***
## zipcode98109       9.368e-01  2.034e-02  46.066  < 2e-16 ***
## zipcode98112       9.929e-01  1.524e-02  65.164  < 2e-16 ***
## zipcode98115       7.850e-01  1.249e-02  62.865  < 2e-16 ***
## zipcode98116       7.239e-01  1.417e-02  51.076  < 2e-16 ***
## zipcode98117       7.869e-01  1.268e-02  62.066  < 2e-16 ***
## zipcode98118       4.438e-01  1.276e-02  34.771  < 2e-16 ***
## zipcode98119       9.214e-01  1.696e-02  54.309  < 2e-16 ***
## zipcode98122       7.588e-01  1.479e-02  51.307  < 2e-16 ***
## zipcode98125       5.556e-01  1.333e-02  41.678  < 2e-16 ***
## zipcode98126       5.393e-01  1.390e-02  38.805  < 2e-16 ***
## zipcode98133       4.535e-01  1.277e-02  35.518  < 2e-16 ***
## zipcode98136       6.558e-01  1.500e-02  43.711  < 2e-16 ***
```

```
## zipcode98144        6.299e-01  1.400e-02  44.989  < 2e-16 ***
## zipcode98146        2.824e-01  1.456e-02  19.400  < 2e-16 ***
## zipcode98148        1.671e-01  2.614e-02   6.394 1.65e-10 ***
## zipcode98155        4.246e-01  1.301e-02  32.638  < 2e-16 ***
## zipcode98166        3.025e-01  1.507e-02  20.068  < 2e-16 ***
## zipcode98168        8.572e-02  1.485e-02   5.774 7.85e-09 ***
## zipcode98177        5.720e-01  1.512e-02  37.842  < 2e-16 ***
## zipcode98178        1.371e-01  1.496e-02   9.165  < 2e-16 ***
## zipcode98188        9.859e-02  1.844e-02   5.346 9.10e-08 ***
## zipcode98198        5.984e-02  1.461e-02   4.094 4.25e-05 ***
## zipcode98199        8.126e-01  1.429e-02  56.854  < 2e-16 ***
## log(sqft_living)    3.889e-01  6.722e-03  57.855  < 2e-16 ***
## grade               1.052e-01  1.957e-03  53.754  < 2e-16 ***
## view                5.821e-02  1.972e-03  29.517  < 2e-16 ***
## waterfront1         4.848e-01  1.608e-02  30.149  < 2e-16 ***
## log(sqft_living15)  1.806e-01  6.660e-03  27.122  < 2e-16 ***
## condition2          1.088e-01  3.634e-02   2.994 0.002757 **
## condition3          2.111e-01  3.370e-02   6.263 3.84e-10 ***
## condition4          2.491e-01  3.370e-02   7.392 1.50e-13 ***
## condition5          3.079e-01  3.392e-02   9.076  < 2e-16 ***
## sqft_lot            7.356e-07  3.309e-08  22.226  < 2e-16 ***
## floors1.5           1.905e-02  4.921e-03   3.872 0.000108 ***
## floors2             6.943e-03  3.684e-03   1.885 0.059472 .
## floors2.5           3.504e-02  1.497e-02   2.340 0.019280 *
## floors3            -8.231e-02  9.105e-03  -9.041  < 2e-16 ***
## floors3.5          -1.420e-02  6.498e-02  -0.219 0.826955
## yr_renovated        3.146e-05  3.372e-06   9.331  < 2e-16 ***
## bathrooms           3.115e-02  2.947e-03  10.570  < 2e-16 ***
## bedrooms           -1.378e-02  1.900e-03  -7.252 4.25e-13 ***
```

```
## yr_built           -6.976e-04  7.819e-05  -8.922  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' '
1
##
## Residual standard error: 0.1829 on 21523 degrees of freedom
## Multiple R-squared:  0.8799, Adjusted R-squared:  0.8794
## F-statistic:  1791 on 88 and 21523 DF,  p-value: < 2.2e-16
```