Unit 3: Multiple Regression 1

Contents

[STOR 455 Class 11 R Assessing Multiple Linear Regression Models 1](#_Toc89002558)

[STOR 455 Class 12 R Correlated Predictors & Model Selection Methods 1](#_Toc89002559)

[STOR 455 - Class 13 - R Model Section Methods 1](#_Toc89002560)

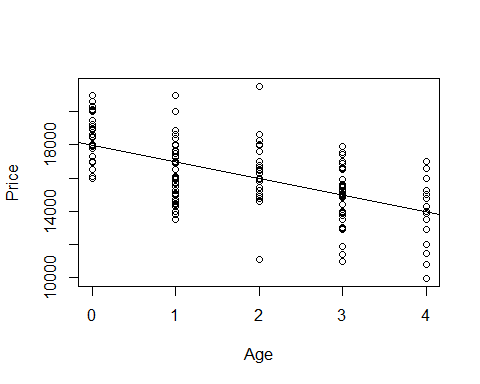
[STOR 455 Homework #2 1](#_Toc89002561)

## STOR 455 Class 11 R Assessing Multiple Linear Regression Models

library(readr)  
library(Stat2Data)  
  
data("Houses")  
  
source("https://raw.githubusercontent.com/JA-McLean/STOR455/master/scripts/anova455.R")  
  
# Or you can download the R script from Sakai  
# Save the script in the same folder as the notebook  
# comment out the above sourced script from github  
# Run code below  
#source("anova455.R)

# install if needed  
# dplyr package used for sample\_n() function  
  
library(dplyr)  
  
UsedCars <- read\_csv("UsedCars.csv")  
  
# selects same random sample each time  
set.seed(09132021)  
  
# random sample of 200 CamrySE  
CamrySE = sample\_n(subset(UsedCars, Model=="CamrySE"), 200)  
# Gets a random sample of a subset of the full dataset, where the model is camrySE, the second arguement is just how many it's taking

CamrySE$Age = 2017 - CamrySE$Year  
CamrySE\_Model = lm(Price~Age, data=CamrySE) # model  
plot(Price~Age, data=CamrySE) # Plot model   
abline(CamrySE\_Model)

 Look above; from here we have to think about the linear conditions - There is one situation were we need to look at a different way of the conditoins - When we have predictors wiht lots of values for the response for it ;we dont care about teh normalitiy to the residuals overall, we care about the normiality of the residuals for each response - In other words, for all the cars that are zero years old, are those residuals normally distirbuted for one, two, three and four are they all normally distributed? - Not a lot of years ar ehere, but we can try to see if we haev that normal distributio or are these some probailemns betwen the years

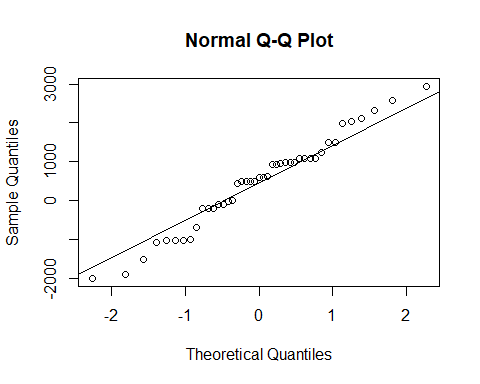
*How to lok at athat* - Can split it up by the ages of teh car

**Below:** - Looking at the camry’s by age

# Splits residuals by Age  
  
Resid\_by\_age = split(CamrySE\_Model$residuals, CamrySE$Age) # SUPER USEFUL FUNCTION  
Resid\_by\_age

## $`0`  
## 8 13 22 26 29 34   
## 1486.908341 -1022.091659 -204.091659 1095.908341 982.908341 -13.091659   
## 35 39 46 47 49 55   
## -1019.091659 1487.908341 2942.908341 591.908341 942.908341 2330.908341   
## 56 58 59 60 61 69   
## -1914.091659 937.908341 2043.908341 589.908341 -2008.091659 1983.908341   
## 71 75 80 85 88 89   
## -107.091659 987.908341 617.908341 2119.908341 1087.908341 -8.091659   
## 96 105 115 117 119 121   
## -204.091659 1095.908341 -1018.091659 430.908341 -1012.091659 495.908341   
## 123 128 129 137 143 149   
## 987.908341 495.908341 -1518.091659 -113.091659 1095.908341 924.908341   
## 150 159 165 171 175 193   
## -204.091659 2583.908341 -1077.091659 -704.091659 1237.908341 495.908341   
## 198   
## 495.908341   
##   
## $`1`  
## 2 5 16 18 24 25   
## -2036.580152 -1997.580152 -3506.580152 -1077.580152 -2722.580152 -1999.580152   
## 27 31 33 38 45 53   
## -891.580152 -26.580152 -1284.580152 1383.419848 257.419848 -942.580152   
## 57 62 63 65 67 68   
## -1100.580152 -3003.580152 -1514.580152 1868.419848 -2391.580152 -1342.580152   
## 77 78 79 81 83 84   
## -3175.580152 495.419848 -1493.580152 1384.419848 -992.580152 229.419848   
## 90 92 93 94 95 103   
## -3006.580152 593.419848 333.419848 -942.580152 -2609.580152 998.419848   
## 106 107 113 122 124 127   
## 1896.419848 3.419848 498.419848 293.419848 -1342.580152 673.419848   
## 130 135 138 139 140 141   
## -3210.580152 1003.419848 -2500.580152 -1996.580152 -96.580152 1007.419848   
## 144 148 152 155 157 161   
## -1091.580152 1612.419848 -492.580152 -2310.580152 1383.419848 -3006.580152   
## 164 167 170 172 174 180   
## -3506.580152 -2992.580152 508.419848 -1742.580152 -1891.580152 3003.419848   
## 185 190 191 196 199   
## -492.580152 2998.419848 599.419848 -1342.580152 4008.419848   
##   
## $`2`  
## 4 6 10 14 28 32   
## 2012.93135 2304.93135 2673.93135 530.93135 21.93135 -700.06865   
## 36 44 50 51 52 64   
## 997.93135 -976.06865 1643.93135 -976.06865 5548.93135 1003.93135   
## 70 74 76 97 99 101   
## 678.93135 477.93135 -67.06865 1014.93135 18.93135 1022.93135   
## 104 108 112 131 132 136   
## 323.93135 -189.06865 -549.06865 768.93135 11.93135 -978.06865   
## 142 145 146 156 160 162   
## -1094.06865 273.93135 2004.93135 1614.93135 1005.93135 -4898.06865   
## 166 169 188 189 192   
## -981.06865 -1377.06865 1018.93135 -1177.06865 2018.93135   
##   
## $`3`  
## 1 3 7 9 12 17   
## 289.442861 -1018.557139 -967.557139 -2060.557139 2448.442861 -960.557139   
## 19 20 21 23 37 43   
## -150.557139 -535.557139 30.442861 -521.557139 1939.442861 2019.442861   
## 48 54 66 72 73 86   
## 1534.442861 -1980.557139 260.442861 -1963.557139 37.442861 638.442861   
## 87 98 100 109 110 111   
## 1918.442861 539.442861 -1063.557139 499.442861 524.442861 1536.442861   
## 114 118 120 125 126 133   
## 651.442861 1530.442861 2939.442861 971.442861 -1970.557139 619.442861   
## 134 151 153 158 168 173   
## -3060.557139 -960.557139 -1461.557139 -3565.557139 -965.557139 4.442861   
## 176 177 178 179 181 182   
## 129.442861 1656.442861 -3975.557139 -1238.557139 -69.557139 2039.442861   
## 184 186 187 194 197 200   
## 2563.442861 2034.442861 1925.442861 36.442861 939.442861 1590.442861   
##   
## $`4`  
## 11 15 30 40 41 42   
## -450.04563 349.95437 50.95437 -3152.04563 1049.95437 2049.95437   
## 82 91 102 116 147 154   
## -1950.04563 -3988.04563 2649.95437 -2458.04563 3045.95437 849.95437   
## 163 183 195   
## -1057.04563 -70.04563 1304.95437

# Different lists, of the string 0; all residuals for 2017 models   
# Then 1, 2, and 3, etc old cars   
# Year year each year gets its own string   
  
# If want to look at normaility, then look at the qqnorm plot of it   
# Need to do for each thing individually   
  
# This is for the zero year old cars   
qqnorm(Resid\_by\_age$'0')  
qqline(Resid\_by\_age$'0')



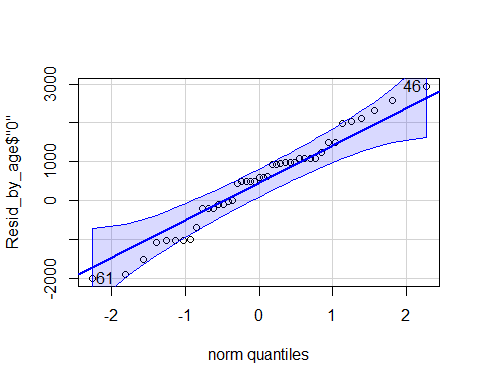
# install if needed  
# car package used for qqPlot()  
  
library(car)

## Loading required package: carData

##   
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':  
##   
## recode

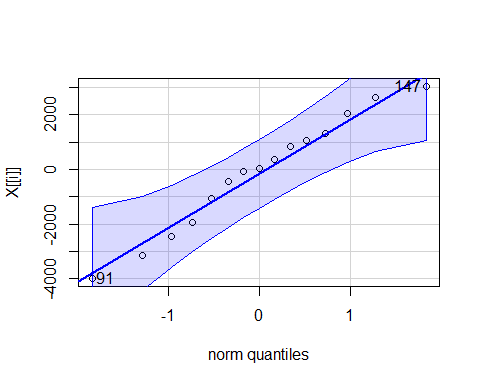
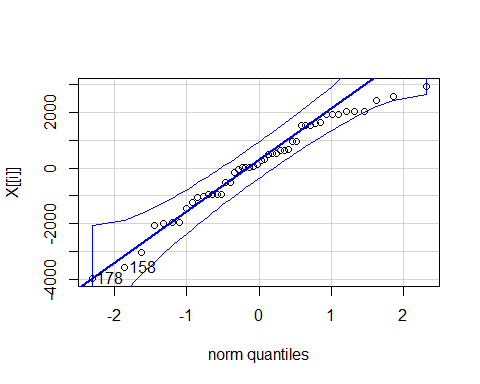
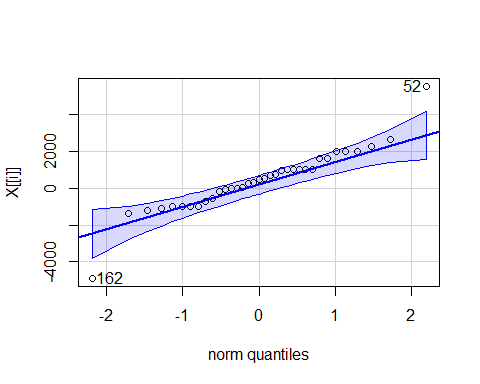
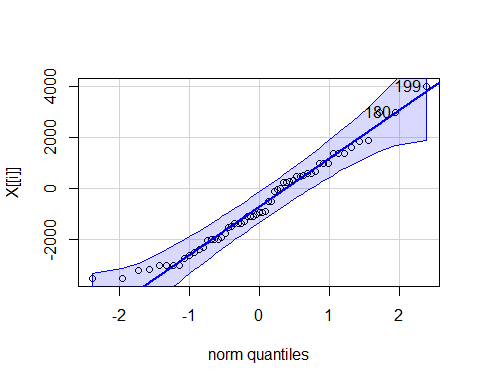
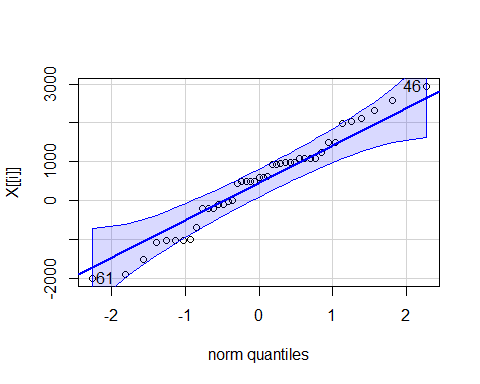
qqPlot(Resid\_by\_age$'0') # THIS IS A SUPER USEFUL FUNCTION



## 61 46   
## 17 9

# Shows teh QQNorm and QQLine plot if norma distributioned   
# Gives a band of what we would expect to see if this data of this size was normally distributed   
# The varibaility we would expect from a thing of this size   
# It looks pretty good in this interval   
# Want to use this for all the ages of the cars

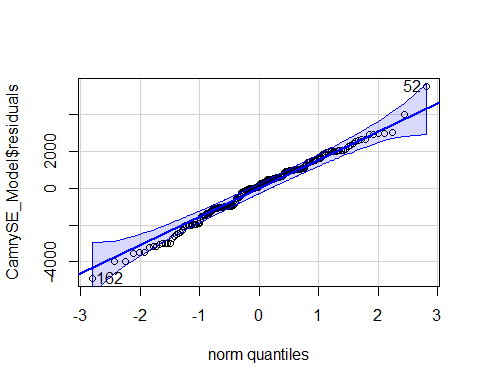
sapply(Resid\_by\_age, qqPlot)



## 0 1 2 3 4  
## 61 17 59 30 39 8  
## 46 9 54 11 34 11

# Applies the qqPlot function to all the residual\_by\_age   
# This shows us that it's normal enough for each age, so it's good to go

qqPlot(CamrySE\_Model$residuals)



## [1] 52 162

# Narrow band because bigger dataset   
# Age 4 is wider ebcause there is less data   
# Less data = wider band, more data = smaller band (Same idea as confidence intervals)

**Three different types of tests** **T-test for Slope** - Ho: B1 =0 - Ha: B2 != 0

* t = b1/SEofb1
* Compare to t(subof n-2)
* Do we have evidence to say the relationship bt the predict and resposne has a non horizontal relationship/non zero slope?

**ANOVA for regression** - Ho: B0 = 0 - Ha: B1 != 0

* F = MSModel/MSE
* Compare to F1, n-2
* (tn-2)^2 = F1, n-2
* For simple linear models, did the same as the t test; tried to look at ho wmuch variability is being explained by odel compared to a null model (so a horizontal line)

**T-test for correlation** - Ho: p = 0 - Ha: p != 0

* t = ((r\*sqrt(n-2))/sqrt(1-r^2))
* Compare to tn-2
* Do we have evidence to say there is some kind of correlation, +/- for the two variables?

**Simple Linear Regression Model** - Y = B0 + B1X + Error - Where Error follows N(0,stdof error) and independent (normal and independent) *What if we have more than one potential predictor?* - We got teh same values if we did the above three different tests on a simple linear regression model, so its not super helpful when you’re doing a simple linear regression model

**Multiple Regression Model** - Y = B0 +B1X1 + B2X2 +….+BkXk + Error - where error is assumed to follow a normal and independent - THis is in many dimentions with more predictors

Data?  
We need n data cases, each with values for Y and all of the predictors X1,…,Xk.

**R - Correlation Matrix**

head(Houses)

## Price Size Lot  
## 1 212000 4148 25264  
## 2 230000 2501 11891  
## 3 339000 4374 25351  
## 4 289000 2398 22215  
## 5 160000 2536 9234  
## 6 85000 2368 13329

# Just a small dataset of 20 houses   
  
cor(Houses)

## Price Size Lot  
## Price 1.0000000 0.6848219 0.7157072  
## Size 0.6848219 1.0000000 0.7668722  
## Lot 0.7157072 0.7668722 1.0000000

# We can do this because they are all quanatitive variables   
# all have fairly strong, postiive correlations of eachother   
# Are tehse big enough to take this claim tot he population?

*Look at a test between them*

**t-test for Correlation**

#cor.test(Houses) # This doesnt work because there's too many   
  
# Correlation looks at 2 vars once and looks at the relationship bt two vars   
  
cor.test(Houses$Size, Houses$Price)

##   
## Pearson's product-moment correlation  
##   
## data: Houses$Size and Houses$Price  
## t = 3.9871, df = 18, p-value = 0.0008643  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.3476582 0.8651583  
## sample estimates:  
## cor   
## 0.6848219

# Ho: No correlation between teh two in teh population   
# Ha: There is a correlation between teh two in the population   
# Want to see how likely it is that we would get the smapel line in the population   
# Output tells us the correlation bt the two, the t test of 3.9 (this tells us that it's pretty unlikely by chance); pvalue of 0.000008 - that is the probability that we would get a sample like this or one as extreme as this if there was no realtionship in teh popualtion; so this is pretty unlikely this would happen by chance   
  
# Can do the same thing above for the other realtionships as well   
cor.test(Houses$Lot, Houses$Price)

##   
## Pearson's product-moment correlation  
##   
## data: Houses$Lot and Houses$Price  
## t = 4.3478, df = 18, p-value = 0.0003878  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.3998134 0.8796343  
## sample estimates:  
## cor   
## 0.7157072

cor.test(Houses$Size,Houses$Lot)

##   
## Pearson's product-moment correlation  
##   
## data: Houses$Size and Houses$Lot  
## t = 5.0694, df = 18, p-value = 7.991e-05  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.4909631 0.9029654  
## sample estimates:  
## cor   
## 0.7668722

# BOth have teh same conclusion and hypothesis that   
# Ho: No relationshio in the popualtion   
# Ha: Some realtionship in population   
# Both have small pvalues   
# Both have strong evidence to say there is a realtionship between these things in teh population

**t-test for Correlation** - Ho: p = 0 - Ha: p != 0

* t = ((r\*sqrt(n-2))/sqrt(1-r^2))
* Find p-value with t n-2

Use this to: 1. Identify potential good predictors of Y. 2. Look for relationships among predictors.

**Prediction Equation** - where the coefficients are chosen to minimize: SSE = sum(y-yhat)^2

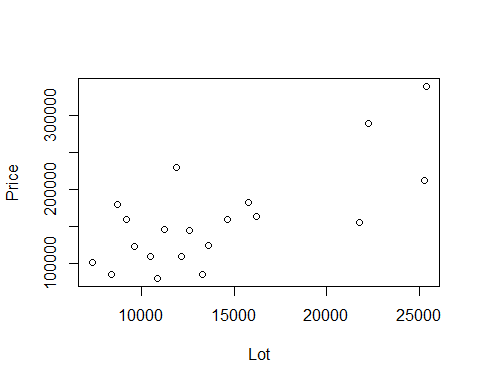
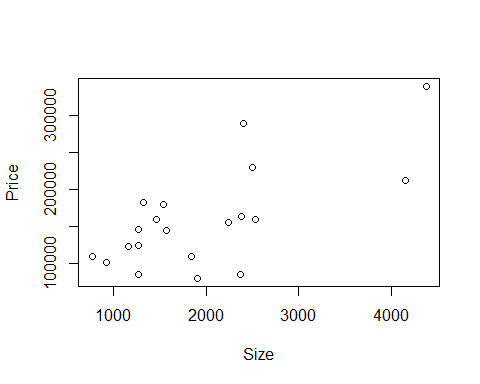
To fit a multiple regression in R: model=lm(y~pred1+pred2+pred3,data= ) - Other tests keep in mind the model that we are workign with, the correlation test always lookas only at 2, while the other tests look at more than 2 variables at once

**R Regression: Individual T-tests** - Look at the P value of the predictors (Size and lot in this example)

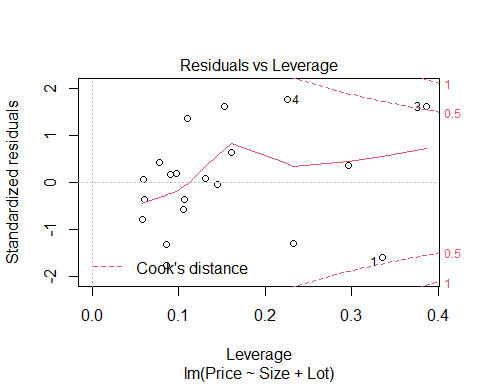
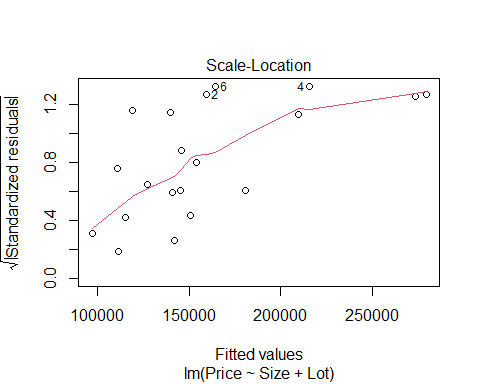
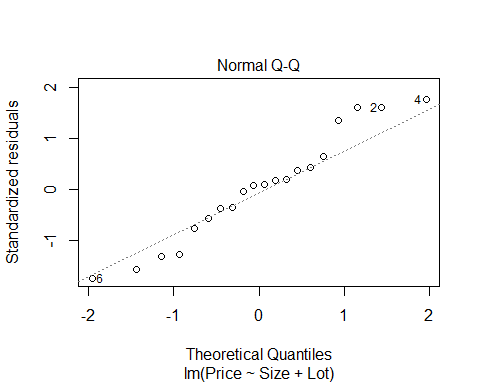
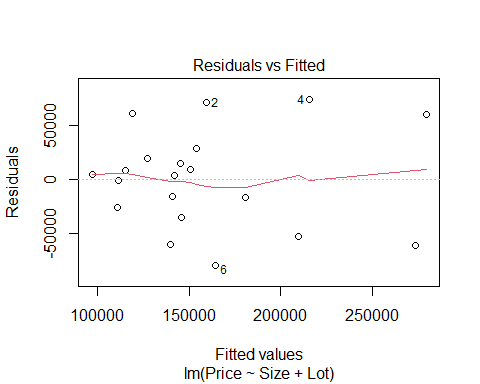
HouseModel=lm(Price~Size+Lot,data=Houses) # Multiple linear regression model   
summary(HouseModel)

##   
## Call:  
## lm(formula = Price ~ Size + Lot, data = Houses)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -79532 -28464 3713 21450 73507   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 34121.649 29716.458 1.148 0.2668   
## Size 23.232 17.700 1.313 0.2068   
## Lot 5.657 3.075 1.839 0.0834 .  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 47400 on 17 degrees of freedom  
## Multiple R-squared: 0.5571, Adjusted R-squared: 0.505   
## F-statistic: 10.69 on 2 and 17 DF, p-value: 0.000985

# The summary output:   
# The test we looked at before for individual slopes are still there, just an extra line is there for teh extra predictor   
# The resuls are not what we would expect; where invidually when we compare size with teh price of the house and the lot with t price of the house, we had small pvalues saying we would have a really low chance of getting this sample if there was no realtion   
# Same tests here that the coeff of size is vs teh alternatieve it's not, we get pvalues athat are higher and possibily not sign at this rate   
# There is something goeing on here, which is called multicolinearity (See next class's notes)  
# The bottome line gives the anova tests, does something different   
  
plot(Price~Size+Lot,data=Houses)



# It's not giving us one plot we can look at and think about lineariry and constant variance to the full modle   
# Its giving us two different plots, because this plot would be given in 3 dimentions   
# We have aplot by size by price and lot size by price   
# Hard to see the realtionship here with this visual format   
# We know teh realtionship with each redcitor in teh response could be useful when looking at t transformations in multiple regression   
  
plot(HouseModel)



# We see that it's not all conditions are really met   
# Same idea as simple lienar regression because for simple linear regression, just compare residuals to the fitted values or looking at teh normaility of the residuals   
# Here, no matter how many predictors we have, we still haev how far off are we and what are the residuals, so we can still look at the residuals and fitted plots   
# Does teh red line look roughly horizontal or some defined curve? IT's a small dataset, so we expect some kind of variability, that's looks pretty good. there's not some clear curve over the whole model   
# Normaility, can judge the same as simple lienar regression   
# the residuals by fitted, we can still look at this roughly the same, but leverage is a little different now  
# For simple linear models it was how far are we fro teh mean predictors, for mutkpel, we could be far from some predictors but close to others  
#Cook's distance, look at it the same as simple linear models outside bounds, teh have influence

We are saying that a price in house is about 34121.649 + 23.23size + 5.657lot **R - Multiple Regression** (𝑃𝑟𝑖𝑐𝑒) ̂=34121.6+23.232∙𝑆𝑖𝑧𝑒+5.657∙𝐿𝑜𝑡 - We are summing each test has a slope of zero - the t test stat = how many SE we are from zero - about the same, but different dfs

**Multicolinearity** is when size is a good predictor of lot already, so when we look at things in the future; if lots of predictors are correlated with eachother, then they may be explaineng a similar amount of variability

**T-test for Slope** -Note: We now have multiple “slopes” to test - Ho: B1 = 0 - Ha: Bi != 0

* t.s. = Bi/SEofBi
* All given in R with a p-value
* Compare to t n-k-1
* **lose 1 d.f. for each coefficient**
* Reject Ho if The ith predictor is useful in this model

**Coefficient of Multiple Determination** - 𝑅^2=𝑆𝑆𝑀𝑜𝑑𝑒𝑙/𝑆𝑆𝑇𝑜𝑡𝑎𝑙 - Now interpreted as the % of variability in the response variable (Y) that is “explained” by a linear combination of these predictors. *NOTes* - Variability in response being explained by the predictior - for Simple linear regression this was just correlation squared - now ti’s different because we haev 2 predictorsl its not just looing at each predictor and the response its lookig at the variability in teh resposne based on all teh predictors in the model - may explain overlaping responses in the mdoel

* Look at teh adjustesd R squared as 0.55, it says 55% of the variability in the price of the house is explained by the size of the house and the lot size of the house; that leaves an extray 44% that is not explained by these things by the data we have; so there may be other variables that explain that varibaility and we just don thaev those variablies

**t-test for Correlation vs. t-test for Slope** - **t-test for correlation:** Assesses the linear association between two variables by themselves. – in a vaccum; compare two things, ignore world - **t-test for slope:** Assesses the linear association after accounting for the other predictors in the model. – accounts for other predicotrs

**Partitioning Variability** - Y = B0+B1X1+…+BkXk + Error - SSTotal = SSModel + SSE - SSModel = Total explained by the regression - SSE = Error after regression - SSTotal = Total variability in Y

* About the same thing, but just in different dimentions
* we have amodel that predicts that data dn teh error around that model that we haev
* can look at teh SS the same as before and this case, we jsut condense teh SSModel not just from 1 predicotr, but from multiple predictors we are looking at
* ANOVA wise, the idea of accounting for that avariability is about the same

**ANOVA test for Overall Fit** - Ho: B1 = B2 = …= Bk = 0 (weak model) - Ha: Some Bi != 0 (Effective model)

Source, d.f, Sum of Squares (SS), MeanAquare, t.s., P-value Model, k, SSModel, SSModelk, MSModel/MSE, Fk, n-k-1 Residual, n-k-1, SSE, SSE/(n-k-1), t.s. is the same, p value is the same Total, n-1, SSTotal

* Only difference than before is that there can be multiple predictors
* Still haev the other things, just accounts for other htings in the model
* still trying to figure out how good is the mdoel doing
* WE ARE ASSUMING NOTHING IN THE MODEL IS USEFUL TO US
* We are assuming that the coeffe of the predictors are zero **ANOVA Hypo Test**
* Ho: There is no point of using the model; all coeffs are zero
* Ha: Assume at least one is non-zero
* Use to see if the predictors are useful
* will talk more about this when it comes to errors later
* if we haev 10 preictors in the mode, we dont wat to test each of teh predicotrs individuals because tehres agood chance we will amek an error an dhow a type 1 error
* if we do any overal test, it will give us an way to see if anything is useful to us
* good to use for nestsed tests
* comapres the mdoel with all predictors to a null modle where teh coeff are both 0
* the same process can be used to compare nested models together
* think: Model with price by size and lot, compared to a model of just price by size; does adding this extra variabile explain a sig amount more of the variability?

**R - Regression ANOVA** - **Important note:** R shows a “sequential” sum of squares in the ANOVA table, i.e. how much new variability is explained as each predictor is added. Add the components to find the SSModel.

anova(HouseModel)

## Analysis of Variance Table  
##   
## Response: Price  
## Df Sum Sq Mean Sq F value Pr(>F)   
## Size 1 4.0447e+10 4.0447e+10 18.0018 0.0005485 \*\*\*  
## Lot 1 7.6013e+09 7.6013e+09 3.3831 0.0833990 .   
## Residuals 17 3.8196e+10 2.2468e+09   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# ANOVA by default will looka at ta sequnetial SS  
# This iwll loo at the first line of the model that has price predicted by size and it says is teh slope of that mdoel nonzero? we have pvalue of 0.0005,   
# Next line says compares a model with asize and lot in it to a model of just size; if we add lot to our odel, does it explain more varibaility? In this case, no.   
# Use ANOVA455 if you DONT want it to look at it sequentially

**A “Local” ANOVA Function** - To find ANOVA for a multiple regression model that is NOT split sequentially for each predictor…

anova455(HouseModel)

## ANOVA Table  
## Model: Price ~ Size + Lot   
##   
## Df Sum Sq Mean Sq F value P(>F)   
## Model 2 4.8048e+10 2.4024e+10 10.693 0.000985 \*\*\*  
## Error 17 3.8196e+10 2.2468e+09   
## Total 19 8.6244e+10   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# It condesnes the model and looks at it instead of line by linea nd adding it one at a time, its looking at teh overal model   
# It condesnes all teh SS together rather than the SS of each individual predictor   
  
# We see that its going in twith teh assumption that the coeffes for lot and size are 0 and ttrying to see if we have evidence to say at least one of them is non zero   
# Have as amll pvalue that at least one is nonzero, but it seems that it contradicts what we have before   
# Where we lookeda t teh summary, then both the pvalues were big, so it's a bit contradictory because of teh multiocolinearity that we talk about next class

**Example: Houses** 1. Test #1: Compute and test the correlation between Size and Lot in Houses

cor.test(HousesLot) t = 5.0694, df = 18, p-value = 7.991e-05

1. Test #2: Compute and test the coefficient of Size in a multiple regression model (along with Lot) to predict Price. (Estimate Std Error t value Pr(>|t|) Intercept) 34121.649 29716.458 1.148 0.2668 Size 23.232 17.700 1.313 0.2068 Lot 5.657 3.075 1.839 0.0834

F-statistic: 10.69 on 2 and 17 DF, p-value: 0.000985

## STOR 455 Class 12 R [Correlated Predictors & Model Selection Methods](https://sakai.unc.edu/portal/site/ff98023c-6e12-47a7-acba-0c12abe4203b/tool/f03494dc-48e2-44b2-8904-0aa5ba69b16a#Class 12)

library(readr)  
library(Stat2Data)  
library(car)  
  
data("Houses")  
  
StateSAT <- read\_csv("https://raw.githubusercontent.com/JA-McLean/STOR455/master/data/StateSAT.csv")  
  
source("https://raw.githubusercontent.com/JA-McLean/STOR455/master/scripts/anova455.R")

head(Houses)

## Price Size Lot  
## 1 212000 4148 25264  
## 2 230000 2501 11891  
## 3 339000 4374 25351  
## 4 289000 2398 22215  
## 5 160000 2536 9234  
## 6 85000 2368 13329

cor(Houses)

## Price Size Lot  
## Price 1.0000000 0.6848219 0.7157072  
## Size 0.6848219 1.0000000 0.7668722  
## Lot 0.7157072 0.7668722 1.0000000

HouseModel=lm(Price~Size+Lot,data=Houses)  
# Linear model that predicts jprice by size and lot of the house.  
  
summary(HouseModel)

##   
## Call:  
## lm(formula = Price ~ Size + Lot, data = Houses)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -79532 -28464 3713 21450 73507   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 34121.649 29716.458 1.148 0.2668   
## Size 23.232 17.700 1.313 0.2068   
## Lot 5.657 3.075 1.839 0.0834 .  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 47400 on 17 degrees of freedom  
## Multiple R-squared: 0.5571, Adjusted R-squared: 0.505   
## F-statistic: 10.69 on 2 and 17 DF, p-value: 0.000985

# Tests teh coef size and lot are equal to zero; that they do not have a realtionship with price   
# Alternative: that at least one of those is nonzero   
# Very low pvalue, very unlikly that we would get this sample if the nuill was true; we have evide nce to say that at least one of these has a non zero slope   
  
# Whtat is the FTest stats and what it is useful?   
# The pvcaalue is really what we want  
# The f tests stat, if its big or small depends on the sample size and the number of predictors;   
# GFOr a small sample of small predictors, 10 = big number   
# Large sample wiht lots of predictors, 10 = small number   
# Mostly focus on the pvalue and how to interpret that   
  
# Looking at the coeff table   
# The Ho: Coeff of size = 0  
# Ha: Coef size != 0   
 # Same thing for lot   
# Each ahas an indivual test sthat ehre is probably not evidence of a relationshipo   
# The two appear to be contradictory there   
# When you're writng out the hypotehsis, you can say in words what he ahsa said or you can write it our mathmatically

#cor.test(Houses)  
  
cor.test(Houses$Size, Houses$Price)

##   
## Pearson's product-moment correlation  
##   
## data: Houses$Size and Houses$Price  
## t = 3.9871, df = 18, p-value = 0.0008643  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.3476582 0.8651583  
## sample estimates:  
## cor   
## 0.6848219

# jfThe relationship bt price and size are with a low pvalue, 0.0008; havbe evidence of a relationship here that is non zero correlation   
# the same test with the other; there is no evidence of ra realtionship   
cor.test(Houses$Lot, Houses$Price)

##   
## Pearson's product-moment correlation  
##   
## data: Houses$Lot and Houses$Price  
## t = 4.3478, df = 18, p-value = 0.0003878  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.3998134 0.8796343  
## sample estimates:  
## cor   
## 0.7157072

cor.test(Houses$Size,Houses$Lot)

##   
## Pearson's product-moment correlation  
##   
## data: Houses$Size and Houses$Lot  
## t = 5.0694, df = 18, p-value = 7.991e-05  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.4909631 0.9029654  
## sample estimates:  
## cor   
## 0.7668722

# We get contradcitory results because of multicollinearity   
# there is an issue wher ethe variance is being inflated when we do these tests   
# IF we loook at the relationship between teh predictors (Lot and size) we see that eh correaltion is really high 0.7686 ish (Not exaclty) it's a signifigant realtionship   
# This is driving the conficting results because too much is being explained by the two thing s  
# It's not inherantly bad, it's not telling ust htat there is no realtionshipship, it's just saying that we dont haev evidence to say ther eis s asignfigiant realtionship   
# If we have a lto of predictors that are highly correlationed you might not want to use them all on our model   
# If these predcitors are explaining the same thing, then why include both? IT sjust going to cause problems   
# THis can cause overfitting problems

*Simple models are idea, than overaly complicated ones*

**Multicollinearity**

* What is it? – When two or more predictors are strongly associated with each other.
* Why is it a problem? –Individual coefficients and t-tests can be deceptive and unreliable.

*NOPtes* - Makes the tests deceptive and we need to know that there is multicoloinarity going on - More its unrealiable tests if there are multicollinearity - It makes it harder to interpret but it means that our model acna be simpler than what we have - so it really means, jsut change you rmodel a little

**Effects of Multicollinearity** - If predictors are highly correlated among themselves: 1. The regression coefficients and tests can be extremely variable and difficult to interpret individually. 2. One variable alone might work as well as many.

anova(HouseModel)

## Analysis of Variance Table  
##   
## Response: Price  
## Df Sum Sq Mean Sq F value Pr(>F)   
## Size 1 4.0447e+10 4.0447e+10 18.0018 0.0005485 \*\*\*  
## Lot 1 7.6013e+09 7.6013e+09 3.3831 0.0833990 .   
## Residuals 17 3.8196e+10 2.2468e+09   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

anova455(HouseModel)

## ANOVA Table  
## Model: Price ~ Size + Lot   
##   
## Df Sum Sq Mean Sq F value P(>F)   
## Model 2 4.8048e+10 2.4024e+10 10.693 0.000985 \*\*\*  
## Error 17 3.8196e+10 2.2468e+09   
## Total 19 8.6244e+10   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

*NOtes* -0 WE can see where the correlation is bt lot and size - One way we can test is to see how closely correlated things are – THis is fine whwen you’re jsut looking at two things; if we have a lot of predictors then teh correlation between things cna be hard to use as a measure ebcause there are more things to look at - Solution: Build a new model for each predcitor where th remaining predcitors are the predictors of that model - In this case, we could build a model for the size of a house and use the rest of the predcitors as predictoyrs (Would just be lot in this case) and do teh same hting for lot and make a model where size is the predictor for that

*See below*

mod=lm(Size~Lot, data=Houses)  
summary(mod)

##   
## Call:  
## lm(formula = Size ~ Lot, data = Houses)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -872.42 -591.71 -47.96 397.03 1214.17   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 91.50286 395.12226 0.232 0.819   
## Lot 0.13324 0.02628 5.069 7.99e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 631.2 on 18 degrees of freedom  
## Multiple R-squared: 0.5881, Adjusted R-squared: 0.5652   
## F-statistic: 25.7 on 1 and 18 DF, p-value: 7.991e-05

# We want ot look at the multiple r sqaured; it says that almost 59% is beign predictoed by how big the lot is   
# Thats just the correlation squared,. the .77 squared; when we get the bigger models, we are going to have to do more to calcualte that   
# We see how much variability is explained there by the two predicotrs   
  
# can use this to se ehow much the variance is being inflated   
# The variance that is being calcualted that is for each paredictor is not done in isolation; its taking into account the other predictors; more multicollinearity will increase the variance   
summary(mod)$r.squared

## [1] 0.588093

# how to pull out the multiple r-squiared from the model

**How do we detect multicollinearity?** 1. Look at a correlation matrix of the predictors.

round(cor(Houses), 2)

## Price Size Lot  
## Price 1.00 0.68 0.72  
## Size 0.68 1.00 0.77  
## Lot 0.72 0.77 1.00

1. Compute the Variance Inflation Factor (VIF).

* (Beware if VIF > 5)
* where Ri2 is for predicting Xi with the other predictors.
* 𝑉𝐼𝐹 > 5 or 𝑅𝑖2 >80%

# How to account for the inflated variance in places with possible multicollinearity   
VIF = 1/(1-summary(mod)$r.squared)  
VIF

## [1] 2.427732

# If VIF is 5 or more, then ther emight be a lot of multicollinearity going on  
# This would mean the adjusted r sqaured would be above 80 or more   
# We are saying that the variance is being aadjusted by a factor of 2.42  
# We get the 2.42 by the VIF   
  
# If we look at the summary of the housemod   
# the variance of size and the stderror = 17.7, when we are doing a hypothesis test for the slope of size, then we are caclauting a t stest stat - the estimate for slope/Stderror;   
 # 23.2/17.9 = 1.313 which is the tvalue   
# that's where that tvalue is coming from   
# We could pull it our better with a summary funciton, but we're not going to   
# So this outcome is about 1.31 stdar devations away if we didnt have a relation between teh things if there was no realtion   
# The variance that we used in this calvcualtion, because of the multicollinearity is being inflated by this facotr   
# This is teh variance inflation factor and we are calculting the stadard error   
# Std = sqrt(variance)   
  
#Go down to sqrtt(VIF) code

**Finding VIF with R** 1. 1. Brute force. Fit a model to predict Xi using the other predictors and find 𝑅𝑖2. - Compute: 𝑉𝐼𝐹=1/(1−𝑅𝑖2) - Example: Find VIF for Size when using Lot to predict Size

summary(HouseModel)

##   
## Call:  
## lm(formula = Price ~ Size + Lot, data = Houses)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -79532 -28464 3713 21450 73507   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 34121.649 29716.458 1.148 0.2668   
## Size 23.232 17.700 1.313 0.2068   
## Lot 5.657 3.075 1.839 0.0834 .  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 47400 on 17 degrees of freedom  
## Multiple R-squared: 0.5571, Adjusted R-squared: 0.505   
## F-statistic: 10.69 on 2 and 17 DF, p-value: 0.000985

sqrt(VIF) # This is how much that variance is being inflated

## [1] 1.558118

# Look at the summary of the houses model   
summary(HouseModel)$coeff[2,2]/sqrt(VIF)

## [1] 11.36012

# sqrt(VIF) = how much its being inflated   
# If no correlation., then we would haev a variance of 11.1; so now if were to divide the slope by this value instead   
  
summary(HouseModel)$coeff[2,1]/summary(HouseModel)$coeff[2,2]/sqrt(VIF) # This is what we would get for the slope if we didn't have the multicollinearity

## [1] 0.8423846

# So multicollienarity has a really big impact on the model   
  
# We dont have to do the math above in practice, it's really just to learn how and why the infaltion is affecting teh table

**Finding VIF with R** 2. 2. Install car package - use vif( )function or use VIF.R script from Sakai *See below for installing car package and using the vif function*

vif(HouseModel)

## Size Lot   
## 2.427732 2.427732

# This is how to do what you did above, but really short form   
# This is what you would use to look at the inflation   
# This will be the same when you are lookign at t athing with 2 predictors   
# Ity will be different when you ahev mutliple predictors   
  
# Even though it changes our result in the data from a sig to a non sig realtionship; multilcolinearity wise, it snot a huge realtionshio  
# mostly because its samll dataset

**What to do if you’ve got Multicollinearity?** 1.Choose a better set of predictors 2.Eliminate some of the redundant predictors to leave a more independent set. 3.Combine predictors into a scale. 4.“Ignore” the individual coefficients and tests.

Note: Predictions aren’t necessarily worse if some predictors are related – it’s just conclusions about individual terms that might be confused.

**NOTES** - Looking at a bigger dataset when it’s not so straightforward to know if there is multicollinearity or not

**Example: State SAT Scores** Source: Statistical Sleuth, Case 12.1 pg. 339  
Response Variable:  
SAT =Average combined SAT Score Potential Predictors:  
Takers = % taking the exam Income = median family income ($100’s) Years = avg. years of study (SS, NS, HU) Public = % public school Expend = spend per student ($100’s) Rank = median class rank of takers

head(StateSAT)

## # A tibble: 6 x 8  
## State SAT Takers Income Years Public Expend Rank  
## <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Iowa 1088 3 326 16.8 87.8 25.6 89.7  
## 2 SouthDakota 1075 2 264 16.1 86.2 20.0 90.6  
## 3 NorthDakota 1068 3 317 16.6 88.3 20.6 89.8  
## 4 Kansas 1045 5 338 16.3 83.9 27.1 86.3  
## 5 Nebraska 1045 5 293 17.2 83.6 21.0 88.5  
## 6 Montana 1033 8 263 15.9 93.7 29.5 86.4

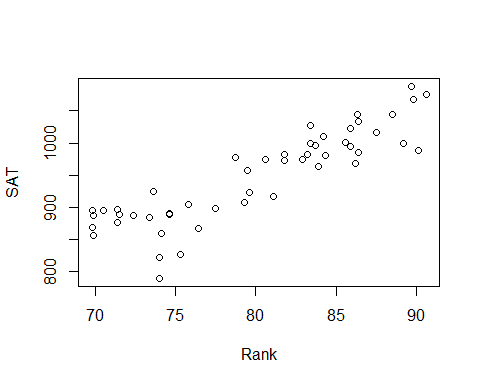
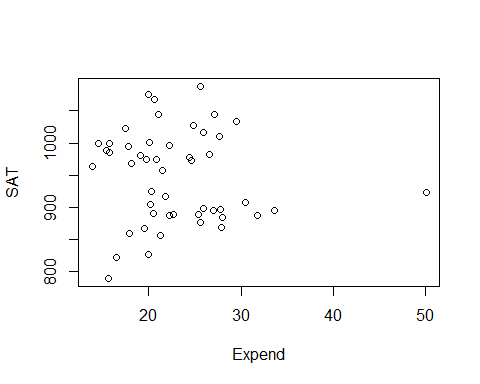
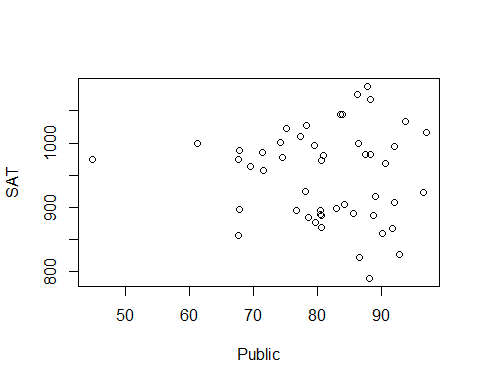
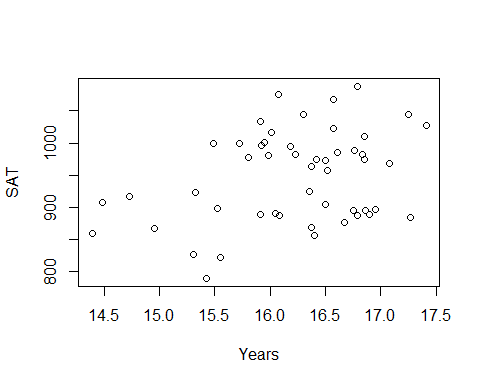
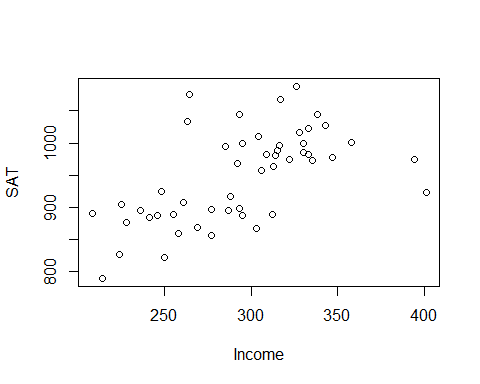
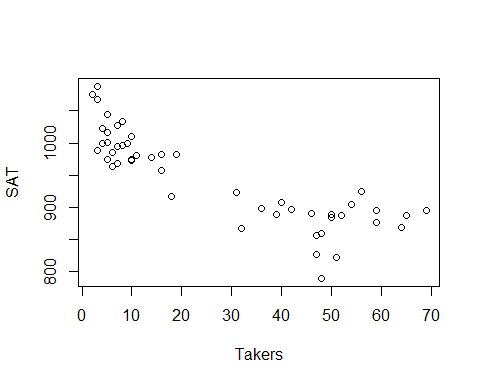
# Some states might have lower SAT takers ebcause the ACT mgiht be better

**Example: Predicting State SAT** Data: StateSAT  
Response: SAT Possible Predictors: Takers Income Years Public Expend Rank

Find the “best” model for GPA using some or all of these predictors.

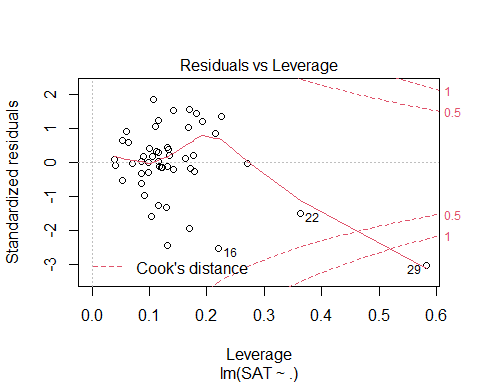
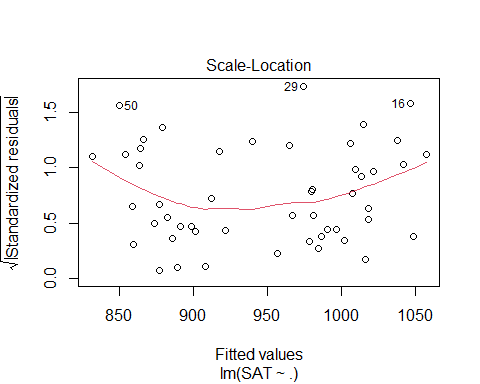
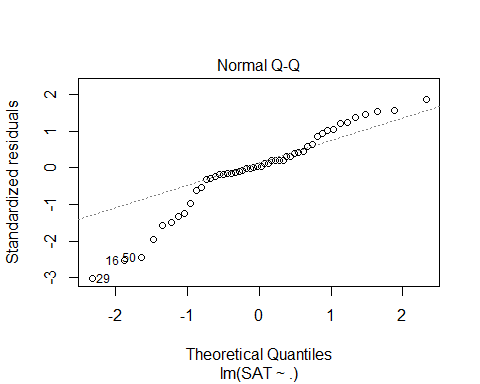
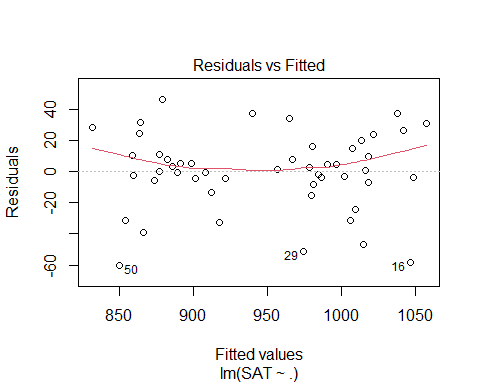
*What determines “best”?*

plot(SAT~., StateSAT[,2:8])



# Want ot predict the average SAT per state   
# The ~ will take all the remaining columns

SAT\_Model = lm(SAT~., data=StateSAT[,2:8])  
plot(SAT\_Model)



summary(SAT\_Model)

##   
## Call:  
## lm(formula = SAT ~ ., data = StateSAT[, 2:8])  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -60.046 -6.768 0.972 13.947 46.332   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -94.659109 211.509584 -0.448 0.656731   
## Takers -0.480080 0.693711 -0.692 0.492628   
## Income -0.008195 0.152358 -0.054 0.957353   
## Years 22.610082 6.314577 3.581 0.000866 \*\*\*  
## Public -0.464152 0.579104 -0.802 0.427249   
## Expend 2.212005 0.845972 2.615 0.012263 \*   
## Rank 8.476217 2.107807 4.021 0.000230 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 26.34 on 43 degrees of freedom  
## Multiple R-squared: 0.8787, Adjusted R-squared: 0.8618   
## F-statistic: 51.91 on 6 and 43 DF, p-value: < 2.2e-16

# We see some weird things are happening where there ar ea lot of NAs happenign here   
#It's because we have categorical variables and r is trying to get those results in number form of us   
# We want to use only the nnumeric things   
# That is what this model does above   
  
# The bottom line does a n anova test witht eh assumption that the slope for the columsn are all zero and there is no relation bt sat scores vs the alternative that a tleast one is non zero in thei model   
# We Think base don this model at least one is a good predictor   
# IF we look at the individual tests for slope, we see where 3 of them have low pvalues (YEars, Expend, and Rank) Where the p value is low; this model those seem to have a strong relationship with SAT score   
# These results might be decieving if there are multicollinearity   
# there is most likely multicollinearity here because its all interfomration fro one state   
# There is probably some lurking variables there that are in the background making these things highly correlated to each other

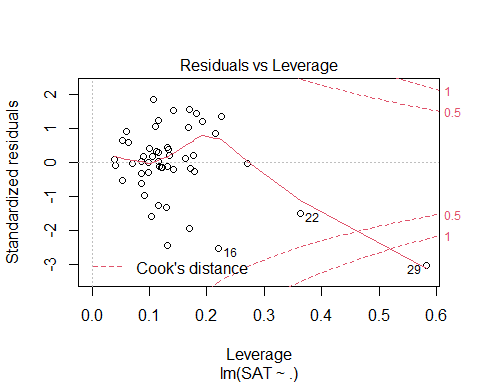
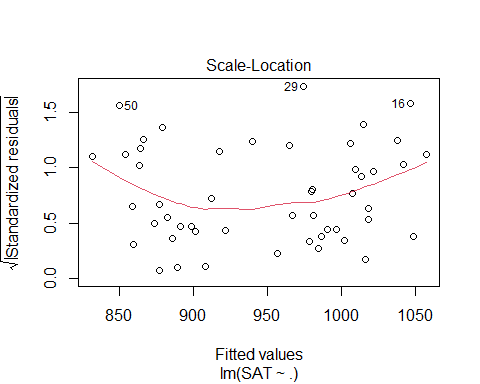
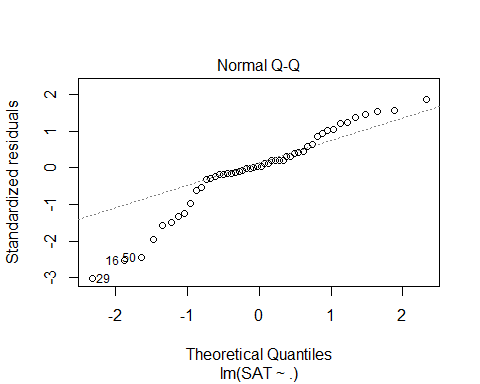
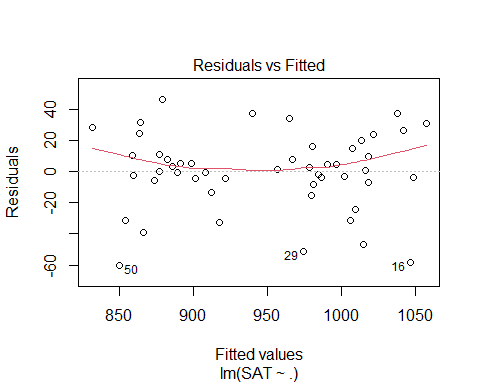
vif(SAT\_Model)

## Takers Income Years Public Expend Rank   
## 16.478636 3.128848 1.379408 2.288398 1.907995 13.347394

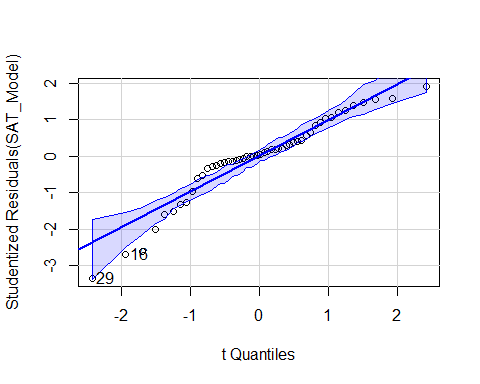
# looking at the VIF for this model, we see that Takers and Rank have really high inflation rates, which means they should probably be expcluded from the final regression model   
# It teslls us wif we make a mdoel with takers as the repsonse, we will get a really high r squared value; almost all the other varibaility in takers is being expained by the other varianceles   
# we probably dont need takers in teh model if everythign else is already doing that for us   
# We could probaly not need rank either as well for the same reason   
# Or maybe just rank or just takers is all we need to predcit.  
# It gives us some informatoin, but the biggest thing is we are skipping a good ifrst step   
# Does it meet the lienar model conditions?

*Does it meet the linear model conditions?* - Look at residual analysis of the data - Does this data meet the criteria, and if it doesn’t, where are these problems occuring? - Becaufore we had 1 variable and wanted to do a transfomration, we could jsut ransforma the predictor/response - Now we have ore predictors and some might have lienar issues while others don’t

plot(SAT\_Model)



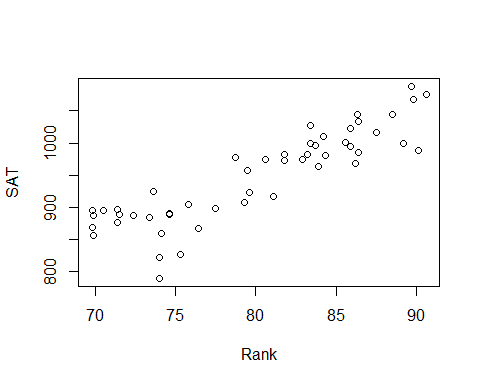
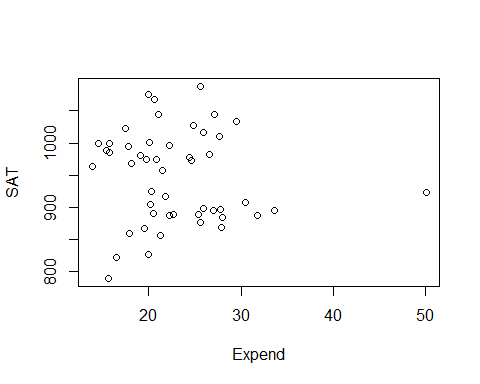
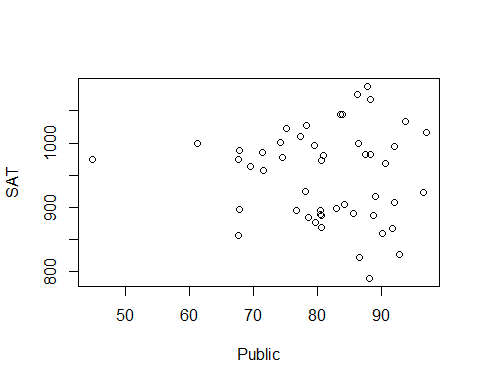
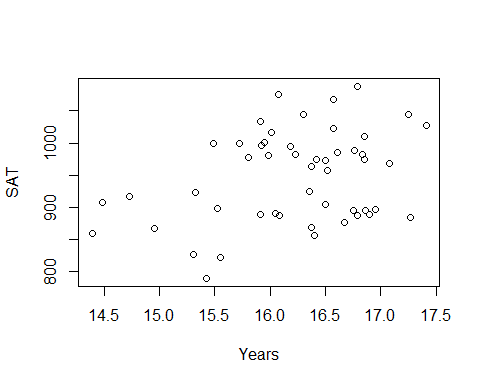
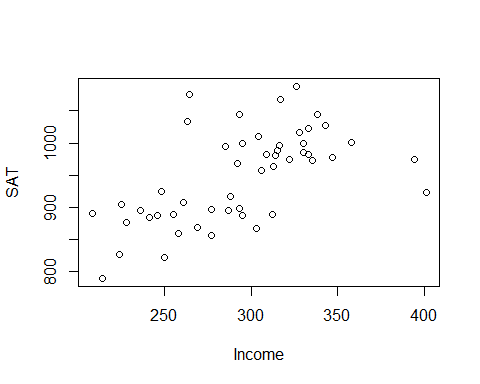
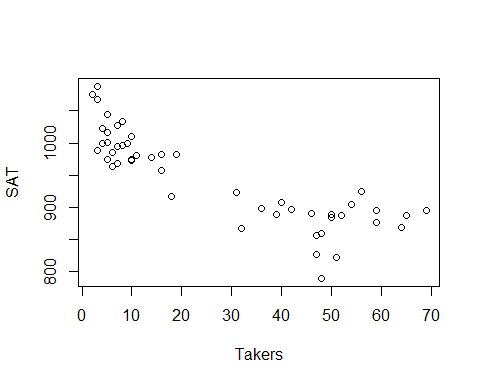
# We can look at these plots to see if the ocnditions are met   
# The model doesn't appear to fit a line well   
# Transfomration might be useful   
# Or, dont include all the variables in teh model   
# Maybe tranfomraiotn or may less predicotrs   
qqPlot(SAT\_Model)



## [1] 16 29

# normalitiy is a big issue   
  
# COnstancce variance: Not a big issue   
# Leverage/Cook's distance - super wonky looking, probably wouldn't trust it right now   
# One data point has high influence in model, should probably look at state 29 because of teh cook's plot

# WE could also look at which varibales are probkematic for us  
plot(SAT~., data = StateSAT[,2:8])



# Tehse plots will help us see wehre things might be a good fit for choosing a predictor   
# Will also help show where we might want to look towards what has outliers   
# first is SAT by takers - there is a curve we could use; does a transformation help this? not so much , but some others might   
# Takers seems a pretty good predicotrorl; would haev to worry about lower states   
  
# Income, doesn't look like a good predicotr, but it is appear to have a conneciton somewhere; but it's not the best; not super linear and not as clear as other s  
# YEars: nothing jumps out, but it's hard to see a pattern, there is something there   
# Public, its hard to say what is going on, that is one that is different than the rest, this is probably the state 29 that has high influence; this variable is probably messing up our data   
# Expend; same issue with one state is apearing to spend more omoeny than the rest of the state; one point has a lot of influence   
# Rank: This is pretty definded realtionship; not a line, but appears to be a good varibale here   
# Guessing: The high VIF bt Takers and RAnk; they appear to expain similar amount of varibility withteh SAT scores

* We have all tehse ariables; how do we make the best model?
* We could go on teh r-squared alone, then it’s pretty good; but you should check the condiotns and that makes it a sus model
* Different ransfomraitons could make the model better and make the model better
* need to see the realtionships to see if there are different combos that will give a better linear conditoins and realtionship between teh model predictors

**Predictor Selection Methods** Choosing an effective set of predictors - All subsets (All combinations of predicotrs in the model and compare all to each other; there is a certain point in which you cant really do this on a compauter because its really hard on a compauter) - Backward elimination - Forward selection - Stepwise regression

## STOR 455 - Class 13 - R Model Section Methods

library(readr)  
library(car)  
library(corrplot) #Install first if needed  
library(leaps) #Install first if needed  
  
StateSAT <- read\_csv("https://raw.githubusercontent.com/JA-McLean/STOR455/master/data/StateSAT.csv")  
  
source("https://raw.githubusercontent.com/JA-McLean/STOR455/master/scripts/ShowSubsets.R")

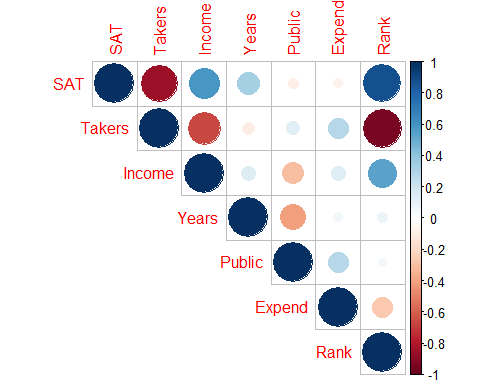
head(StateSAT)

## # A tibble: 6 x 8  
## State SAT Takers Income Years Public Expend Rank  
## <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Iowa 1088 3 326 16.8 87.8 25.6 89.7  
## 2 SouthDakota 1075 2 264 16.1 86.2 20.0 90.6  
## 3 NorthDakota 1068 3 317 16.6 88.3 20.6 89.8  
## 4 Kansas 1045 5 338 16.3 83.9 27.1 86.3  
## 5 Nebraska 1045 5 293 17.2 83.6 21.0 88.5  
## 6 Montana 1033 8 263 15.9 93.7 29.5 86.4

# want to keep in mind what teh corerlation between things are to see what may be useful for a good model   
  
cor(StateSAT[c(2:8)])

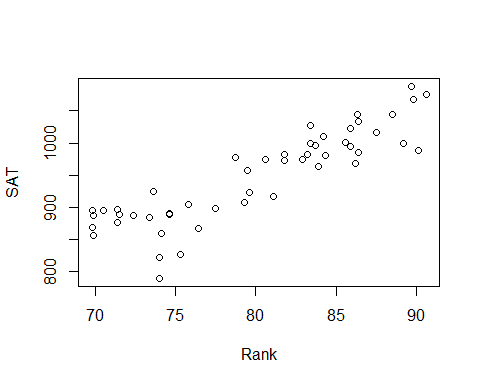
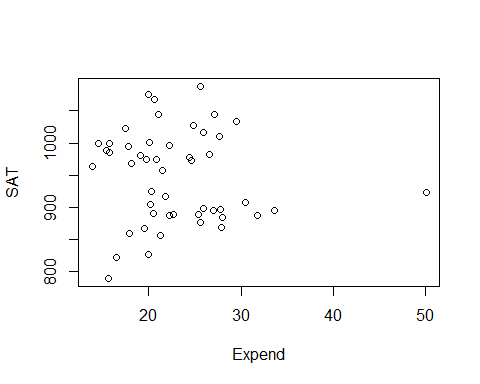
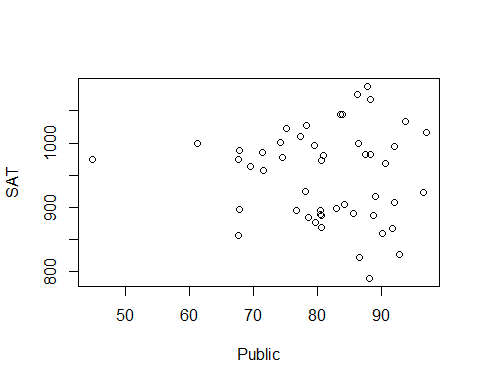
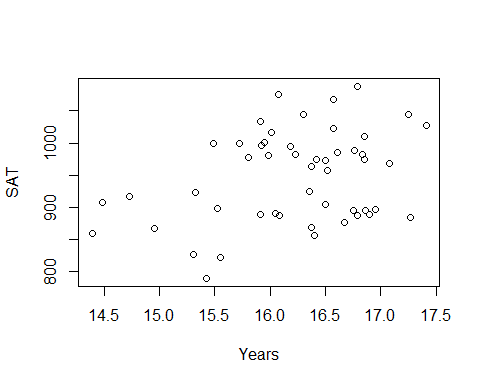
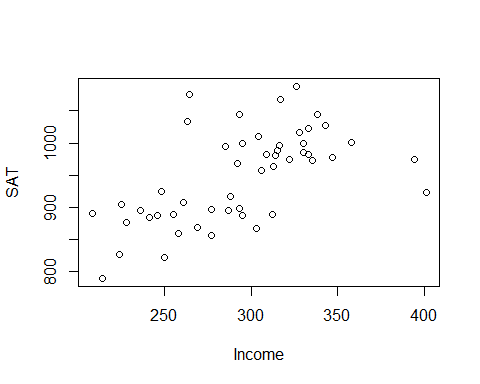
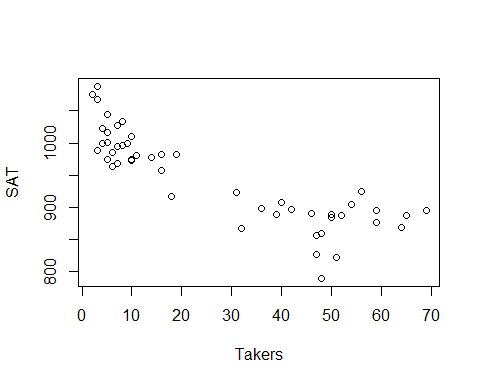
## SAT Takers Income Years Public Expend  
## SAT 1.00000000 -0.8578100 0.5844666 0.33096886 -0.08035688 -0.06287764  
## Takers -0.85780996 1.0000000 -0.6619351 -0.10154350 0.12355625 0.28363041  
## Income 0.58446657 -0.6619351 1.0000000 0.13476231 -0.30656703 0.13151942  
## Years 0.33096886 -0.1015435 0.1347623 1.00000000 -0.41711822 0.05982861  
## Public -0.08035688 0.1235563 -0.3065670 -0.41711822 1.00000000 0.28459116  
## Expend -0.06287764 0.2836304 0.1315194 0.05982861 0.28459116 1.00000000  
## Rank 0.87990910 -0.9428331 0.5326999 0.07022360 0.05062355 -0.26496897  
## Rank  
## SAT 0.87990910  
## Takers -0.94283311  
## Income 0.53269989  
## Years 0.07022360  
## Public 0.05062355  
## Expend -0.26496897  
## Rank 1.00000000

# This makes a correlation matrix that will tell us the correlation between everything in the dataste   
# only owrks for numeric data   
# Not super easy to read   
# Takers has a negative correlation   
# Rank has a strong postive correlation   
# Income and years and other corerlation   
# Doens't tel lme if there is a linera realtionship; its assuming linear relation   
  
corrplot(cor(StateSAT[c(2:8)]), type="upper")



# Helps to visualize the matrix better than other things   
# A nicer visual of the correlation matrix   
# Dark blue = strong correlation   
# Darker and bigger circle = stronger positive or negative correlation   
# Type = "upper" just gives us the upper part of it, it avoids duplicate infomraiotn   
# could also tell where we could have multicllinearity   
# Takers may have multicolinearity from income and rank   
# INcome andr ank will have the same prediction power as takers   
# We can see rank, income and takers have high correlation, so we proabbly dont need all three of those int eh same model because they might explain similar things

plot(SAT~., data=StateSAT[c(2:8)])

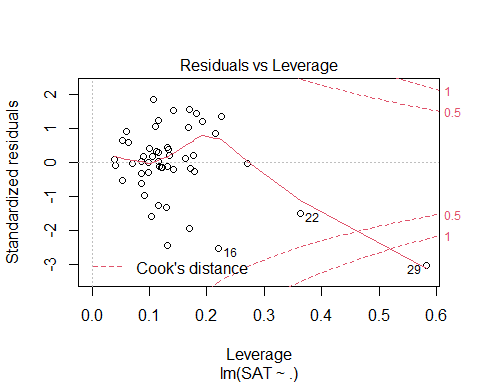
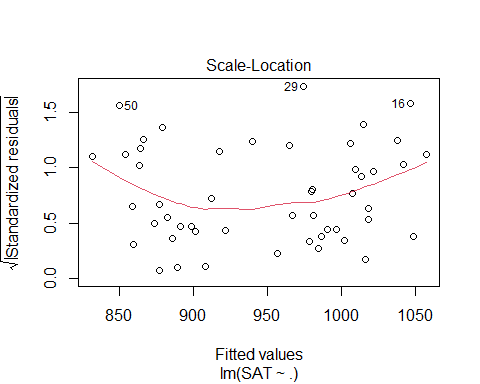
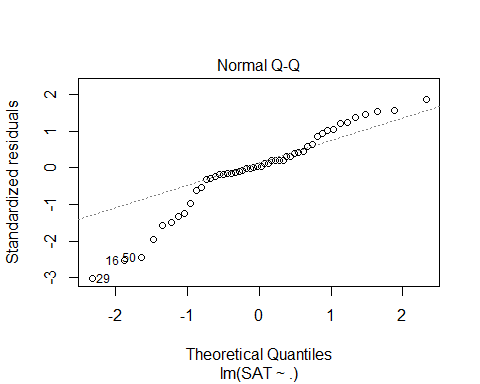
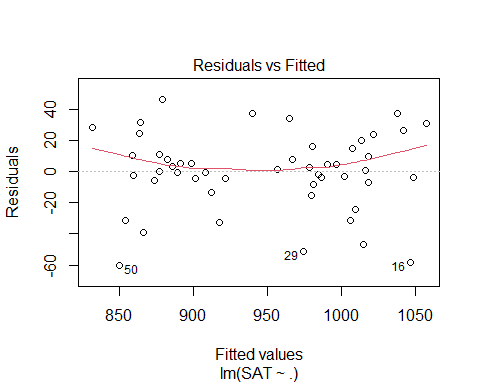


# Plot the data against each of the predictors int eh dataframe   
# Excludes state (Because we would have to factor state, and that would be a lot of information to process)  
  
# Rank adn takers have a recise pattern wtih SAT scores; its appears to have a curved realtionship there   
# Might not have a good linear realtion model conditions, but we can transforms them and work with them   
# Public and Expend = there is one state that is really different htan teh otehrs and thats causing some issues, so we might not want ot use that because it might impact the model in ways we dont wnat

modSAT1 = lm(SAT~., data=StateSAT[c(2:8)])  
# Make a linear model with all the variables   
summary(modSAT1)

##   
## Call:  
## lm(formula = SAT ~ ., data = StateSAT[c(2:8)])  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -60.046 -6.768 0.972 13.947 46.332   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -94.659109 211.509584 -0.448 0.656731   
## Takers -0.480080 0.693711 -0.692 0.492628   
## Income -0.008195 0.152358 -0.054 0.957353   
## Years 22.610082 6.314577 3.581 0.000866 \*\*\*  
## Public -0.464152 0.579104 -0.802 0.427249   
## Expend 2.212005 0.845972 2.615 0.012263 \*   
## Rank 8.476217 2.107807 4.021 0.000230 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 26.34 on 43 degrees of freedom  
## Multiple R-squared: 0.8787, Adjusted R-squared: 0.8618   
## F-statistic: 51.91 on 6 and 43 DF, p-value: < 2.2e-16

# Pvalue of kiw, so we can say that some of these we can sue   
# Rank, expend and years have low pvalues; but we could have icorrect infomraiton because of multicollinarity   
# Rank has a similar issue, but it's small pvalue, so it might be a better predictor model   
# Some have high pvalues even though the correlation looked okay  
  
# R squared is the precentage of sat scores that are explained by teh model; this is hgih, but teh conditoins are really met, so we cant use that as a relaibale model   
  
plot(modSAT1) # Too look at residuals



# Lineariry isnt super good   
# normial is really bad, the tail has an issue   
# Residual plot has one state that has really different values than other things  
vif(modSAT1) # To see if there is any inflation of variance

## Takers Income Years Public Expend Rank   
## 16.478636 3.128848 1.379408 2.288398 1.907995 13.347394

**Criteria to Compare Models?** 1. Look for large R2 - But R2 is always best for the model with all predictors - R squared will never go down because if you add something, you’re not explaining less variability you can only explain that much or more; - Just because it’s high rsquared, deosnt mean they are signifigiant

1. Look for large adjusted R2

* Helps factor in the number of predictors in the model
* Adj r squared formuals: -𝑅\_𝑎𝑑𝑗2=1−(𝜎̂2\_𝜀2)/(𝑆\_𝑌^2 )
* 𝑅^2=𝑆𝑆𝑀𝑜𝑑𝑒𝑙/𝑆𝑆𝑇𝑜𝑡𝑎𝑙 =1−𝑆𝑆𝐸/𝑆𝑆𝑇𝑜𝑡𝑎𝑙
* 𝑅\_𝑎𝑑𝑗^2=1−(𝑆𝑆𝐸⁄((𝑛−𝑘−1)))/(𝑆𝑆𝑇𝑜𝑡𝑎𝑙⁄((𝑛−1))) =1−(𝜎̂\_𝜀2)/(𝑠\_𝑌2 )
* (adjusts for the number of predictors in the model)
* THis penalizes teh r squared based ont eh predictors that we have
* it tells us that we know we will have an increased rsquared with extra predictors, so we need a certain amoutn explained to increase teh rsquared

1. Look at individual t-tests

* Might be susceptible to multicollinearity problems
* There could be decent variables, but we aren’t seeing the full story

**How to Choose Models to Compare?** 1. Method #1: **All Subsets!** - Consider all possible combinations of predictors. - How many are there? - Pool of k predictors then 2𝑘−1 subsets - *Advantage:* Find the best model for your criteria - *Disadvantage:* LOTS of computation

*NOtes* - All subsets: - Can look at all subsets or 1 predictors, 2, 3, 4, 5, etc. - We can make a lot of predictors.  
- Can get out of hand quickly if you have a lot of variables - Catgegorical variables make this message because when you factor it you get a variable for the category

all = regsubsets(SAT~., data = StateSAT[c(2:8)], nbest = 2, nvmax = 6)  
# nbest will tell you the two best models with 6, 5, 4, 3, 2, and 1 predictor   
# nvmax will say only look at models with up to 6 predicotrs here; so it is like an upper bound; its not applicable here, but if we had a bigger selection it would be needed   
summary(all)

## Subset selection object  
## Call: regsubsets.formula(SAT ~ ., data = StateSAT[c(2:8)], nbest = 2,   
## nvmax = 6)  
## 6 Variables (and intercept)  
## Forced in Forced out  
## Takers FALSE FALSE  
## Income FALSE FALSE  
## Years FALSE FALSE  
## Public FALSE FALSE  
## Expend FALSE FALSE  
## Rank FALSE FALSE  
## 2 subsets of each size up to 6  
## Selection Algorithm: exhaustive  
## Takers Income Years Public Expend Rank  
## 1 ( 1 ) " " " " " " " " " " "\*"   
## 1 ( 2 ) "\*" " " " " " " " " " "   
## 2 ( 1 ) " " " " "\*" " " " " "\*"   
## 2 ( 2 ) " " " " " " " " "\*" "\*"   
## 3 ( 1 ) " " " " "\*" " " "\*" "\*"   
## 3 ( 2 ) " " "\*" "\*" " " " " "\*"   
## 4 ( 1 ) " " " " "\*" "\*" "\*" "\*"   
## 4 ( 2 ) "\*" " " "\*" " " "\*" "\*"   
## 5 ( 1 ) "\*" " " "\*" "\*" "\*" "\*"   
## 5 ( 2 ) " " "\*" "\*" "\*" "\*" "\*"   
## 6 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*"

#ISsue: THis doesn't compare the models between eachother

# IMPORTANT  
ShowSubsets(all)

## Takers Income Years Public Expend Rank Rsq adjRsq Cp  
## 1 ( 1 ) \* 77.42 76.95 34.03  
## 1 ( 2 ) \* 73.58 73.03 47.64  
## 2 ( 1 ) \* \* 84.71 84.05 10.22  
## 2 ( 2 ) \* \* 80.54 79.71 24.97  
## 3 ( 1 ) \* \* \* 87.11 86.27 3.69  
## 3 ( 2 ) \* \* \* 85.84 84.91 8.21  
## 4 ( 1 ) \* \* \* \* 87.71 86.61 3.58  
## 4 ( 2 ) \* \* \* \* 87.67 86.57 3.72  
## 5 ( 1 ) \* \* \* \* \* 87.87 86.49 5.00  
## 5 ( 2 ) \* \* \* \* \* 87.73 86.34 5.48  
## 6 ( 1 ) \* \* \* \* \* \* 87.87 86.18 7.00

# this iwll give you more infomraiton   
# For each model, what's teh rsquared, the adj rsquared adn teh mallo cp  
  
# We want a small Mallo Cp  
# The first line with rank, it says 77% the stuff is explained, but its' not taking into accoun the otehr variables

**Mallow’s Cp** - Note: R2, Adjusted R2, SSE, all depend only on the predictors in the model being evaluated – NOT the other potential predictors in the pool. - Mallow’s Cp: When evaluating a subset of m predictors from a larger set of k predictors, - m = # predictors in the reduced model - 𝐶\_𝑝=(𝑆𝑆𝐸\_𝑚)/(𝑀𝑆𝐸\_𝑘 )+2(𝑚+1)−𝑛 *notes* - The amount of var explained with reduced model (What we are just using) compared with teh full model with all of the possible predictors in it (The entire model) - What fraction of the model is explained - It penalizes bigger models - If we look at the full model, it gives us the SSE fule/MSE + the left over (the 2(m-1) etc. - Mallow cp = number of predictors + 1 - If numbers are lower than that number, then thats a useful model - So 2 predictor model, look for a Cp of 3 or less

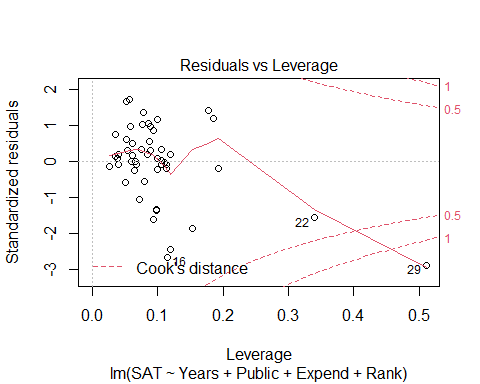
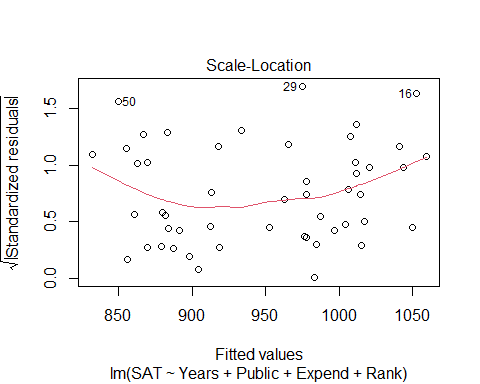
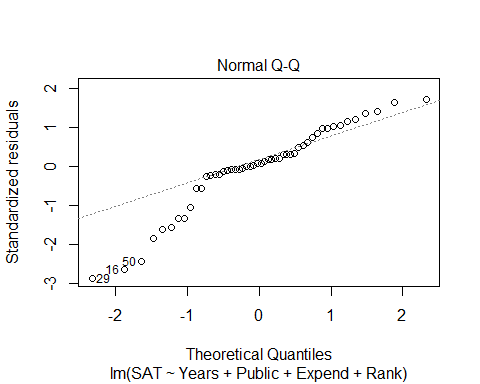
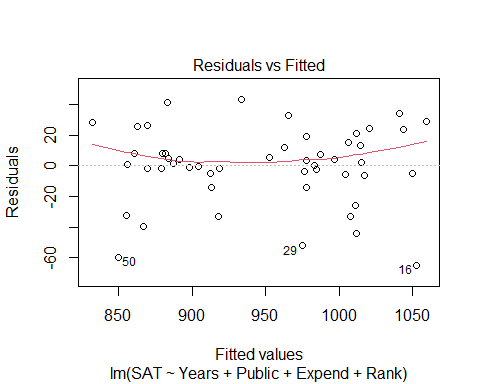
**Notes on Cp** - Cp depends on the larger pool of predictors as well as the set being considered. - For full model Cp = k+1 - For a “good” set of predictor, Cp should be small. - Like Adj R2, Cp weighs both the effectiveness of the model (SSEm) and the # of predictors (m).

**Predictor Selection Methods** - Think, consult, graph… but if that fails, then: 1. All subsets 2. Backward elimination 3. Forward selection 4. Stepwise regression

modSAT3 = lm(SAT~Years+Public+Expend+Rank, data=StateSAT) # this is lowest mallow Cp from best subsets above   
summary(modSAT3)

##   
## Call:  
## lm(formula = SAT ~ Years + Public + Expend + Rank, data = StateSAT)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -64.931 -5.471 1.932 14.980 43.280   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -204.5982 117.6871 -1.738 0.088963 .   
## Years 21.8905 6.0372 3.626 0.000731 \*\*\*  
## Public -0.6638 0.4500 -1.475 0.147154   
## Expend 2.2416 0.6782 3.305 0.001868 \*\*   
## Rank 10.0032 0.6033 16.581 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 25.93 on 45 degrees of freedom  
## Multiple R-squared: 0.8771, Adjusted R-squared: 0.8661   
## F-statistic: 80.25 on 4 and 45 DF, p-value: < 2.2e-16

plot(modSAT3)



vif(modSAT3)

## Years Public Expend Rank   
## 1.301929 1.426831 1.266145 1.129034

# Look at sum; it's sig because we know allsubsets   
# Public has a higher pvalue, but thats because of multicollinearity; they were all highly correlated; public is being inflated a bit   
# We can see that ints not inflated too much because teh VIF is amll; maybe Public just isnt that good   
# Problem: The residual anaysis, we still have nonlinearitiy; if we too things taht din't haev lienar relation with teh response, then we are going to have problems   
# We need to try and make tehse lienar realtions work first, then put it in the model selction process.

**Backward Elimination** 1. Start with the full model (all predictors) 2. Calculate if the model would be “better” by removing each of the predictor individually 3. Find the “least significant” predictor 4. Does removing the predictor create a “better” model? - No, then Keep the predictor & stop - Yes, then Delete the predictor and go back to step 2 with the reduced model.

* *Advantages:* Removes “worst” predictors early Relatively few models to consider Leaves only “important” predictors
* *Disadvantages:* Most complicated models first Individual t-tests may be unstable Susceptible to multicollinearity

summary(modSAT1)

##   
## Call:  
## lm(formula = SAT ~ ., data = StateSAT[c(2:8)])  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -60.046 -6.768 0.972 13.947 46.332   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -94.659109 211.509584 -0.448 0.656731   
## Takers -0.480080 0.693711 -0.692 0.492628   
## Income -0.008195 0.152358 -0.054 0.957353   
## Years 22.610082 6.314577 3.581 0.000866 \*\*\*  
## Public -0.464152 0.579104 -0.802 0.427249   
## Expend 2.212005 0.845972 2.615 0.012263 \*   
## Rank 8.476217 2.107807 4.021 0.000230 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 26.34 on 43 degrees of freedom  
## Multiple R-squared: 0.8787, Adjusted R-squared: 0.8618   
## F-statistic: 51.91 on 6 and 43 DF, p-value: < 2.2e-16

# See that we would want t amodel with no income in it because it's the worse predictor   
  
#This is what backwards elimiation is doing, but step by step  
  
modSAT2.1 = lm(SAT~Takers+Years+Public+Expend+Rank, data=StateSAT)  
summary(modSAT2.1)

##   
## Call:  
## lm(formula = SAT ~ Takers + Years + Public + Expend + Rank, data = StateSAT)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -59.890 -6.637 0.975 13.872 46.261   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -100.4737 179.7256 -0.559 0.578969   
## Takers -0.4621 0.6007 -0.769 0.445883   
## Years 22.6688 6.1486 3.687 0.000620 \*\*\*  
## Public -0.4523 0.5291 -0.855 0.397344   
## Expend 2.1859 0.6851 3.190 0.002620 \*\*   
## Rank 8.4964 2.0505 4.144 0.000153 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 26.04 on 44 degrees of freedom  
## Multiple R-squared: 0.8787, Adjusted R-squared: 0.8649   
## F-statistic: 63.74 on 5 and 44 DF, p-value: < 2.2e-16

# We look at the summary of the new model and then choose the next worse predictor that we want to get rid of   
  
modSAT2.2 = lm(SAT~Years+Public+Expend+Rank, data=StateSAT)  
# This is the new model without takers, because takers probably wasn't signfigant   
summary(modSAT2.2)

##   
## Call:  
## lm(formula = SAT ~ Years + Public + Expend + Rank, data = StateSAT)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -64.931 -5.471 1.932 14.980 43.280   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -204.5982 117.6871 -1.738 0.088963 .   
## Years 21.8905 6.0372 3.626 0.000731 \*\*\*  
## Public -0.6638 0.4500 -1.475 0.147154   
## Expend 2.2416 0.6782 3.305 0.001868 \*\*   
## Rank 10.0032 0.6033 16.581 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 25.93 on 45 degrees of freedom  
## Multiple R-squared: 0.8771, Adjusted R-squared: 0.8661   
## F-statistic: 80.25 on 4 and 45 DF, p-value: < 2.2e-16

# We look at the summary of the new model ad tehn choose the next worse predictor that we want to get rid of   
  
modSAT2.3 = lm(SAT~Years+Expend+Rank, data=StateSAT)  
# This is the new model without takers and public, because public probably wasnt signifigant either   
summary(modSAT2.3)

##   
## Call:  
## lm(formula = SAT ~ Years + Expend + Rank, data = StateSAT)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -64.802 -6.798 2.169 17.525 49.706   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -303.7243 97.8415 -3.104 0.00326 \*\*   
## Years 26.0952 5.3894 4.842 1.49e-05 \*\*\*  
## Expend 1.8609 0.6351 2.930 0.00526 \*\*   
## Rank 9.8258 0.5987 16.412 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 26.25 on 46 degrees of freedom  
## Multiple R-squared: 0.8711, Adjusted R-squared: 0.8627   
## F-statistic: 103.6 on 3 and 46 DF, p-value: < 2.2e-16

**How to do backwards elimination in R**

# Fit the full model  
Full=lm(SAT~Takers+Income+Years+Public+Expend+Rank, data=StateSAT)  
# Find the MSE for the full model  
  
MSE=(summary(Full)$sigma)^2  
# Backward: use the step( ) command starting with the full model  
#MSE = variance of the residuals   
  
step(Full,scale=MSE) # this is the step back so it can step by the mallow cp, so it will get teh model with the smallest mallo cp

## Start: AIC=7  
## SAT ~ Takers + Income + Years + Public + Expend + Rank  
##   
## Df Sum of Sq RSS Cp  
## - Income 1 2.0 29844 5.0029  
## - Takers 1 332.4 30175 5.4789  
## - Public 1 445.8 30288 5.6424  
## <none> 29842 7.0000  
## - Expend 1 4744.9 34587 11.8369  
## - Years 1 8897.8 38740 17.8208  
## - Rank 1 11223.0 41065 21.1712  
##   
## Step: AIC=5  
## SAT ~ Takers + Years + Public + Expend + Rank  
##   
## Df Sum of Sq RSS Cp  
## - Takers 1 401.3 30246 3.5812  
## - Public 1 495.5 30340 3.7169  
## <none> 29844 5.0029  
## - Expend 1 6904.4 36749 12.9515  
## - Years 1 9219.7 39064 16.2876  
## - Rank 1 11645.9 41490 19.7836  
##   
## Step: AIC=3.58  
## SAT ~ Years + Public + Expend + Rank  
##   
## Df Sum of Sq RSS Cp  
## <none> 30246 3.5812  
## - Public 1 1462 31708 3.6884  
## - Expend 1 7343 37589 12.1618  
## - Years 1 8837 39083 14.3141  
## - Rank 1 184786 215032 267.8394

##   
## Call:  
## lm(formula = SAT ~ Years + Public + Expend + Rank, data = StateSAT)  
##   
## Coefficients:  
## (Intercept) Years Public Expend Rank   
## -204.5982 21.8905 -0.6638 2.2416 10.0032

#R uses Cp (AIC) to pick next model  
# Builds model with all predictors; if we removed any predictors, tehn what would the model be if weremove: none = 7; if we remove income, takers, or public then it would get better, but the expend, years, and rank would be bad to get rid of   
# It will take teh worse predictor and get rid of it   
# the best model will be at the bottom   
# This can take a lot of screen, so you can add "trace = FALSE" to the end, which will just give you the last output

**Forward Selection** 1. Start with the best single predictor 2. Is that predictor significant? Yes, then Include predictor in the model No, then Don’t include predictor & stop 3. Find the “most significant” new predictor from among those NOT in the model. Return to step 2.

* *Advantages:* Uses smaller models early (parsimony) Less susceptible to multicollinearity Shows “most important” predictors
* *Disadvantages:* Need to consider more models Predictor entered early may become redundant later
* Continue until adding something is no longer useful
* Want to start with no predictors in the model

# Start with a model with NO predictors  
none=lm(SAT~1,data=StateSAT)  
  
 #Specify the direction  
step(none,scope=list(upper=Full),scale=MSE, direction= "forward")# Full is the full model, you have to tell R what the end point is, it wouldn't have an end point if you didn't include that

## Start: AIC=306.48  
## SAT ~ 1  
##   
## Df Sum of Sq RSS Cp  
## + Rank 1 190471 55539 34.027  
## + Takers 1 181024 64987 47.639  
## + Income 1 84038 161973 187.388  
## + Years 1 26948 219063 269.648  
## + Public 1 1589 244422 306.189  
## <none> 246011 306.478  
## + Expend 1 973 245038 307.076  
##   
## Step: AIC=34.03  
## SAT ~ Rank  
##   
## Df Sum of Sq RSS Cp  
## + Years 1 17913.6 37626 10.215  
## + Expend 1 7671.0 47868 24.974  
## + Income 1 4601.1 50938 29.397  
## + Public 1 3847.7 51692 30.483  
## + Takers 1 1761.8 53778 33.488  
## <none> 55539 34.027  
##   
## Step: AIC=10.22  
## SAT ~ Rank + Years  
##   
## Df Sum of Sq RSS Cp  
## + Expend 1 5917.6 31708 3.6884  
## + Income 1 2782.4 34843 8.2059  
## <none> 37626 10.2152  
## + Takers 1 778.7 36847 11.0931  
## + Public 1 37.0 37589 12.1618  
##   
## Step: AIC=3.69  
## SAT ~ Rank + Years + Expend  
##   
## Df Sum of Sq RSS Cp  
## + Public 1 1462.46 30246 3.5812  
## <none> 31708 3.6884  
## + Takers 1 1368.28 30340 3.7169  
## + Income 1 848.47 30860 4.4659  
##   
## Step: AIC=3.58  
## SAT ~ Rank + Years + Expend + Public  
##   
## Df Sum of Sq RSS Cp  
## <none> 30246 3.5812  
## + Takers 1 401.32 29844 5.0029  
## + Income 1 70.95 30175 5.4789

##   
## Call:  
## lm(formula = SAT ~ Rank + Years + Expend + Public, data = StateSAT)  
##   
## Coefficients:  
## (Intercept) Rank Years Expend Public   
## -204.5982 10.0032 21.8905 2.2416 -0.6638

# Shows you what will happen to the mallow cp if you add a certian predictor to it   
# Computationally, it is a little heavy because it has a lot to look at   
# Sometimes though, the first predictor isnt good once you reach the end

step(none, scope=list(upper=Full), scale=MSE, direction="forward", trace=FALSE) # This is how you get the forward selection, but just the end solution

##   
## Call:  
## lm(formula = SAT ~ Rank + Years + Expend + Public, data = StateSAT)  
##   
## Coefficients:  
## (Intercept) Rank Years Expend Public   
## -204.5982 10.0032 21.8905 2.2416 -0.6638

**Stepwise Regression** - Basic idea: Alternate forward selection and backward elimination 1. Use forward selection to choose a new predictor and check its significance. 2. Use backward elimination to see if predictors already in the model can be dropped.

* What would happen if you add or substract certain things and how would that impact eth mallow cp

# Start with a model with NO predictors  
none=lm(SAT~1,data=StateSAT)  
  
 # Don’t specify a direction  
step(none,scope=list(upper=Full),scale=MSE)

## Start: AIC=306.48  
## SAT ~ 1  
##   
## Df Sum of Sq RSS Cp  
## + Rank 1 190471 55539 34.027  
## + Takers 1 181024 64987 47.639  
## + Income 1 84038 161973 187.388  
## + Years 1 26948 219063 269.648  
## + Public 1 1589 244422 306.189  
## <none> 246011 306.478  
## + Expend 1 973 245038 307.076  
##   
## Step: AIC=34.03  
## SAT ~ Rank  
##   
## Df Sum of Sq RSS Cp  
## + Years 1 17914 37626 10.215  
## + Expend 1 7671 47868 24.974  
## + Income 1 4601 50938 29.397  
## + Public 1 3848 51692 30.483  
## + Takers 1 1762 53778 33.488  
## <none> 55539 34.027  
## - Rank 1 190471 246011 306.478  
##   
## Step: AIC=10.22  
## SAT ~ Rank + Years  
##   
## Df Sum of Sq RSS Cp  
## + Expend 1 5918 31708 3.6884  
## + Income 1 2782 34843 8.2059  
## <none> 37626 10.2152  
## + Takers 1 779 36847 11.0931  
## + Public 1 37 37589 12.1618  
## - Years 1 17914 55539 34.0268  
## - Rank 1 181437 219063 269.6479  
##   
## Step: AIC=3.69  
## SAT ~ Rank + Years + Expend  
##   
## Df Sum of Sq RSS Cp  
## + Public 1 1462 30246 3.5812  
## <none> 31708 3.6884  
## + Takers 1 1368 30340 3.7169  
## + Income 1 848 30860 4.4659  
## - Expend 1 5918 37626 10.2152  
## - Years 1 16160 47868 24.9737  
## - Rank 1 185667 217375 269.2161  
##   
## Step: AIC=3.58  
## SAT ~ Rank + Years + Expend + Public  
##   
## Df Sum of Sq RSS Cp  
## <none> 30246 3.5812  
## - Public 1 1462 31708 3.6884  
## + Takers 1 401 29844 5.0029  
## + Income 1 71 30175 5.4789  
## - Expend 1 7343 37589 12.1618  
## - Years 1 8837 39083 14.3141  
## - Rank 1 184786 215032 267.8394

##   
## Call:  
## lm(formula = SAT ~ Rank + Years + Expend + Public, data = StateSAT)  
##   
## Coefficients:  
## (Intercept) Rank Years Expend Public   
## -204.5982 10.0032 21.8905 2.2416 -0.6638

# In this case we end up with the same case, but this isn't always the case   
# you might end up with different things

**Missing Values** - Warning! If data are missing for any of the predictors in the pool, R’s “Stepwise” and “Best Subsets” procedures will eliminate the data case from all\* models. - Thus, running the model for the selected subset of predictors alone may produce different results than within the stepwise or best subsets procedures. - \*R’s step( ) sometimes gives an error.

## STOR 455 Homework #2

40 points - Due Wednesday 9/15 at 5:00pm

**Situation:** Suppose that you are interested in purchasing a used car. How much should you expect to pay? Obviously the price will depend on the type of car you get (the model) and how much it’s been used. For this assignment you will investigate how the price might depend on the age and mileage.

**Data Source:** To get a sample of cars, begin with the UsedCars CSV file. The data was acquired by scraping TrueCar.com for used car listings on 9/24/2017 and contains more than 1.2 million used cars. For this assignment you will choose a car *Model* for which there are at least 100 of that model listed for sale in a state of your choice (that is not North Carolina). After constructing a subset of the UsedCars data under these conditions, check to make sure that there is a reasonable amount of variability in the years and mileages for your car (ie, all of your cars are not from the same year). The model that you choose should have cars ranging over at least 5 years. You should add a variable called *Age* which is 2017-year (since the data was scraped in 2017).

**Directions:** The code below should walk you through the process of selecting data from a particular model and state of your choice. Each of the following two R chunks begin with {r, eval=FALSE}. eval=FALSE makes these chunks not run when I knit the file. Before you knit these chunks, you should revert them to {r}.

library(readr)

# This line will only run if the UsedCars.csv is stored in the same directory as this notebook!

UsedCars <- read\_csv("UsedCars.csv")

# Delete the \*\* below and enter the two letter abbreviation for the state of your choice.

StateOfMyChoice = "NY"

# Delete the \*\* below and enter the model that you chose from the Enough\_Cars data.

ModelOfMyChoice = "Civic"

# Takes a subset of your model car from your state

MyCars = subset(UsedCars, Model==ModelOfMyChoice & State==StateOfMyChoice)

# Add a new variable for the age of the cars.

MyCars$Age = 2017 - MyCars$Year

**MODEL #1: Use Age as a predictor for Price**

1. Calculate the least squares regression line that best fits your data. Interpret (in context) what the slope estimate tells you about prices and ages of your used car model. Explain why the sign (positive/negative) makes sense.

**0.5 point** code for model  
**0.5 point** explanation that the slope describes the change in price of the car as it ages 1 year. For my data, for each one year a Civic ages, I predict that the Price of the car will decrease by $1312.35

modq1 = lm(Price~Age, data=MyCars)

summary(modq1)$coef[2,1]

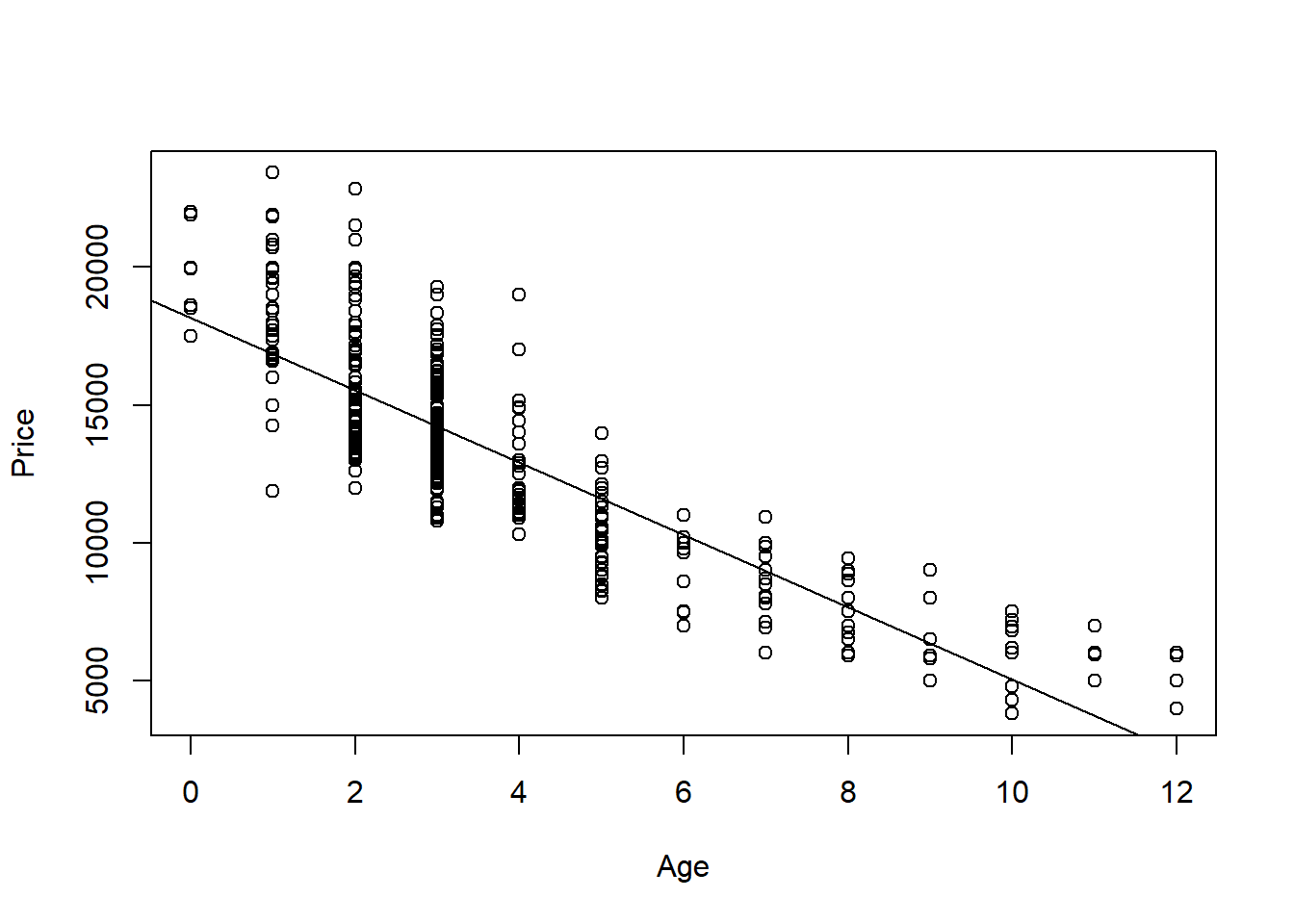
## [1] -1312.345

1. Produce a scatterplot of the relationship with the regression line on it.

**0.5 point** code for plot  
**0.5 point** abline

plot(Price~Age, data=MyCars)

abline(modq1)



1. Produce appropriate residual plots and comment on how well your data appear to fit the conditions for a linear model. Don’t worry about doing transformations at this point if there are problems with the conditions.

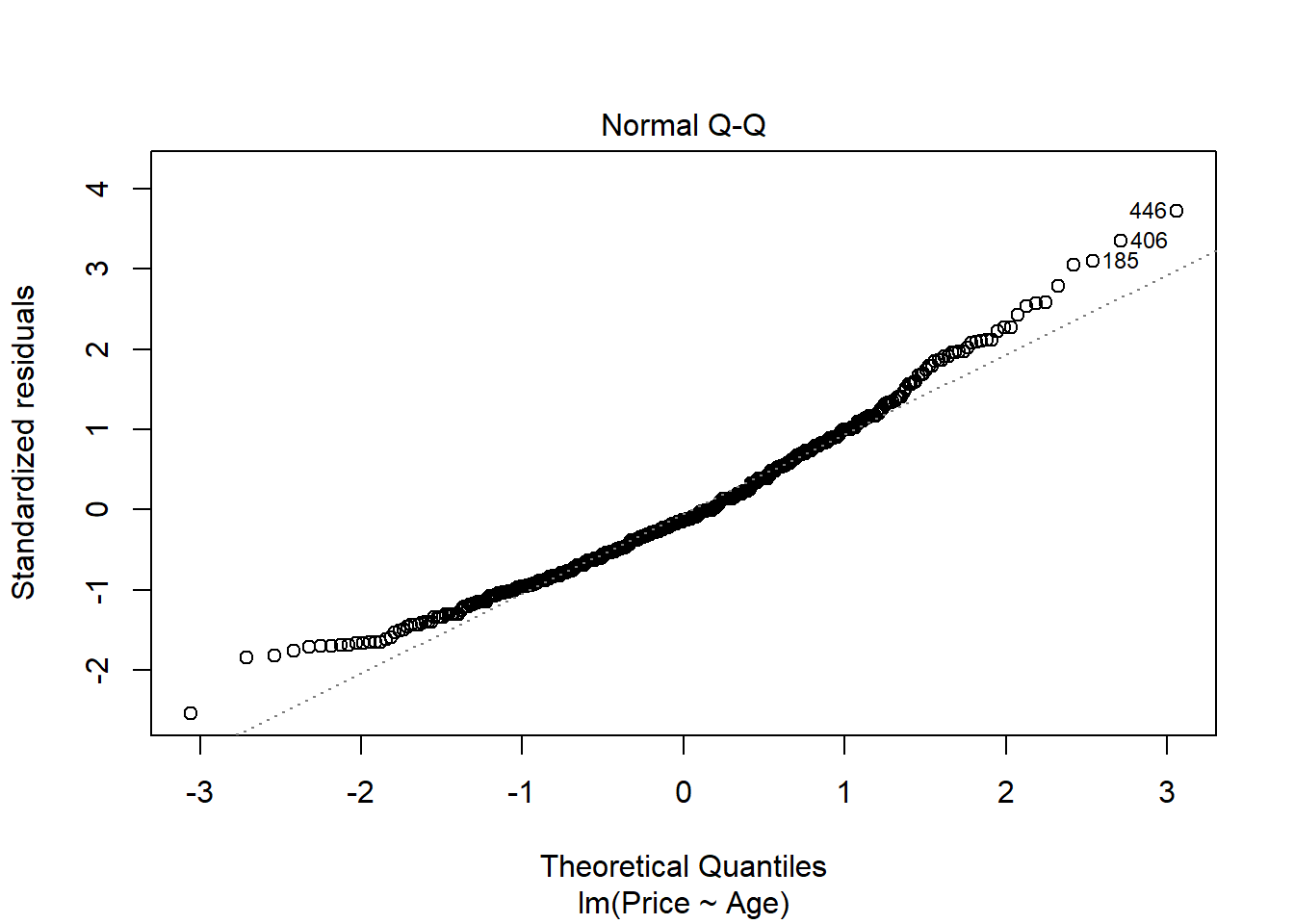
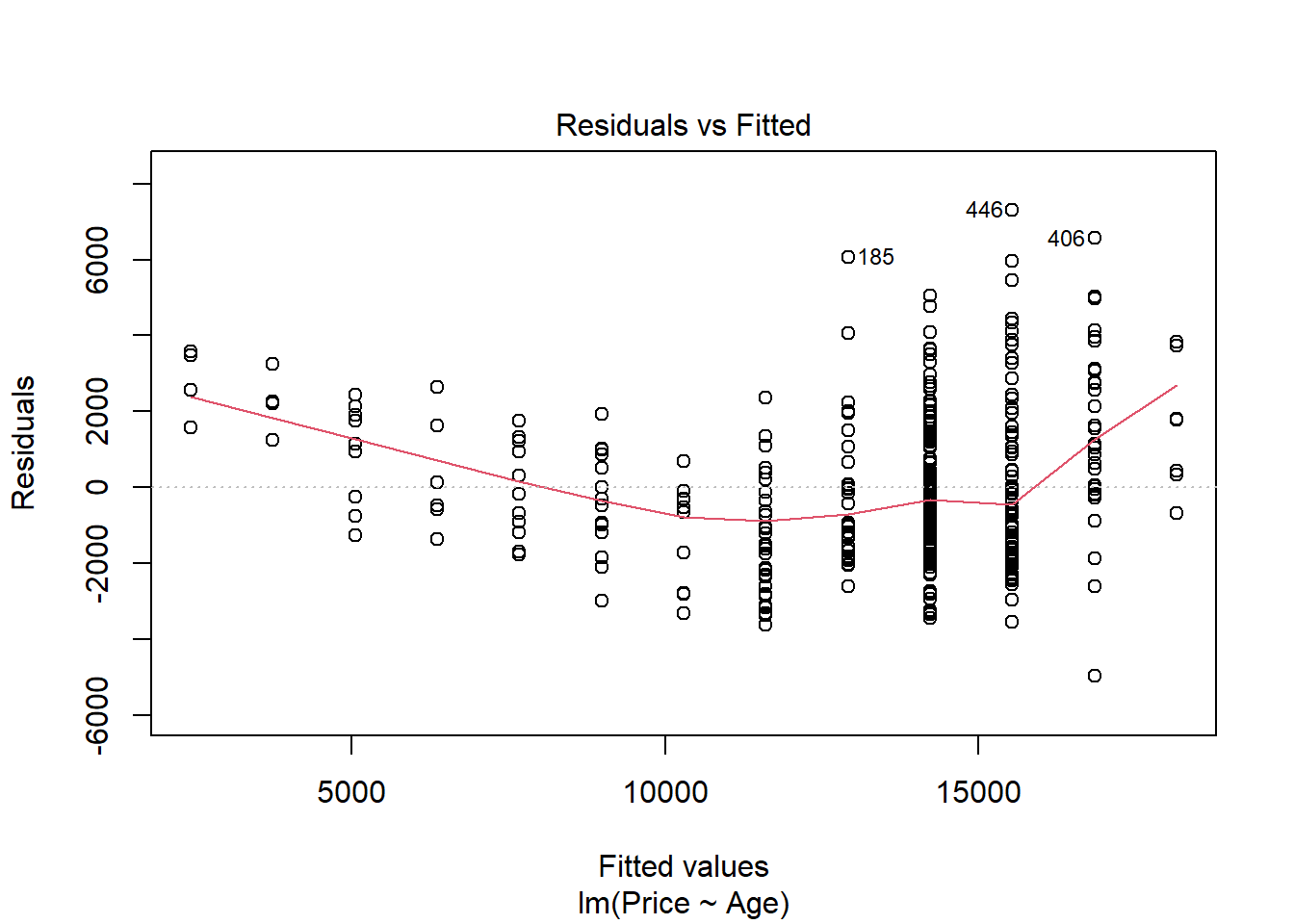
**0.5 points** residuals vs fitted plot  
**0.5 points** qqnorm (or histogram) for Normality of residuals  
**1.5 points** discussion of conditions (linearity, constant variance, and normality of residuals) You can give 0.5 pts each. They can describe the conditions without explicitly using these terms.

Note 1: For linearity, they should have some discussion if the line seems to describe the data, using either the scatter plot or residual vs fitted plot. For constant variance they should discuss if the variability (vertical distances) from the line seems to follow any pattern as value of the predictor changes. For normality of the residuals, they should note the adherence of the residuals (or not) to the qqline, or bell curve shape or skew in a histogram. As each student will have a different plot, any reasonable assertions of the conditions being met (or not) supported by an argument is fine for full credit.

Note 2: plot and discussion of the additional condition, normality of residuals for each value of the predictor, is not required for this assignment (although realistically likely needed for this data). They should split the residuals by Age (or using some other method) and comment on how for each predictor value, the residuals do, or do not seem to be distributed approximately normal.

Note 3: they may use the plot(model) to produce all of the plots at once, or separately produce each of the plots with different lines of code.

plot(modq1, 1:2)

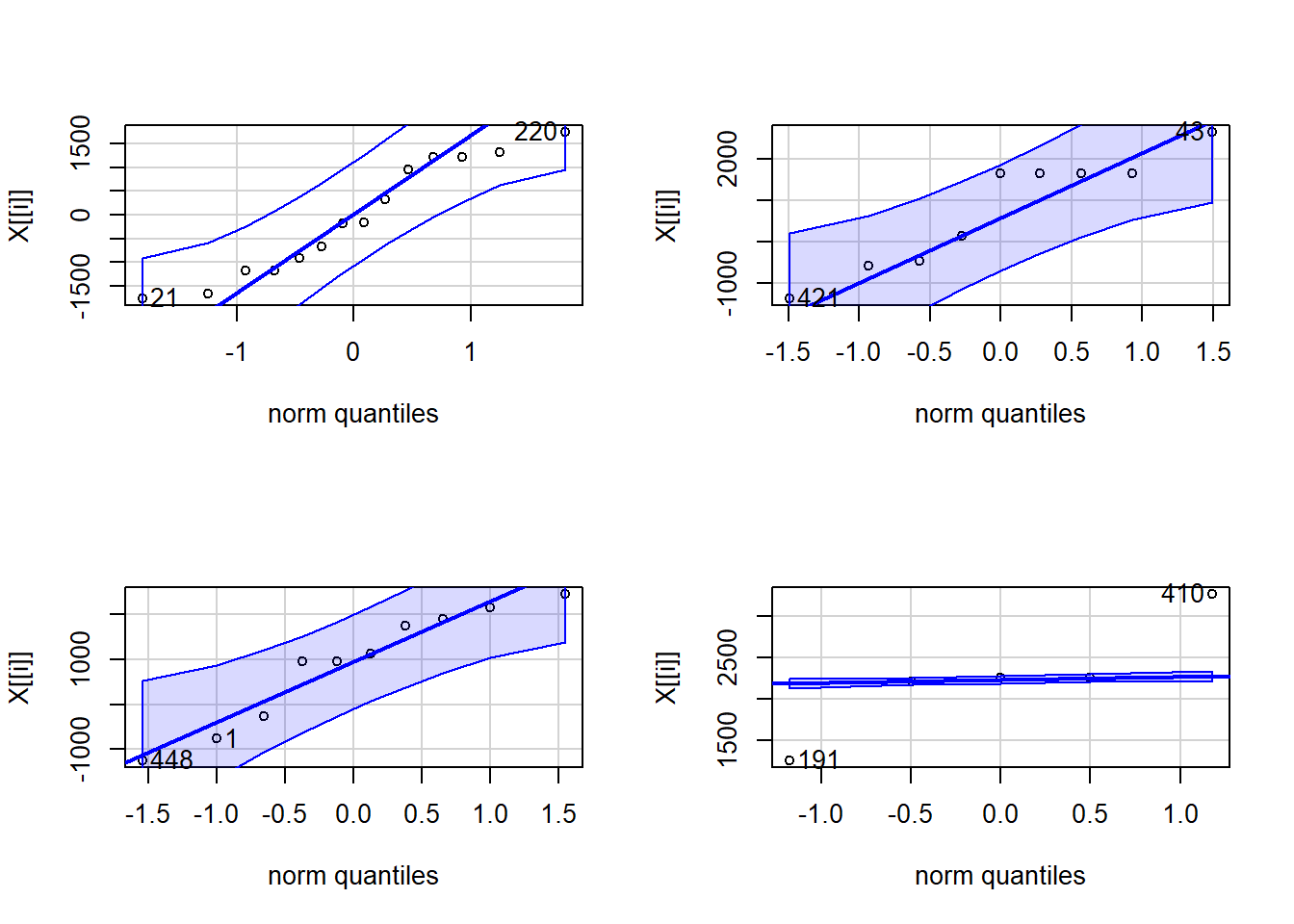
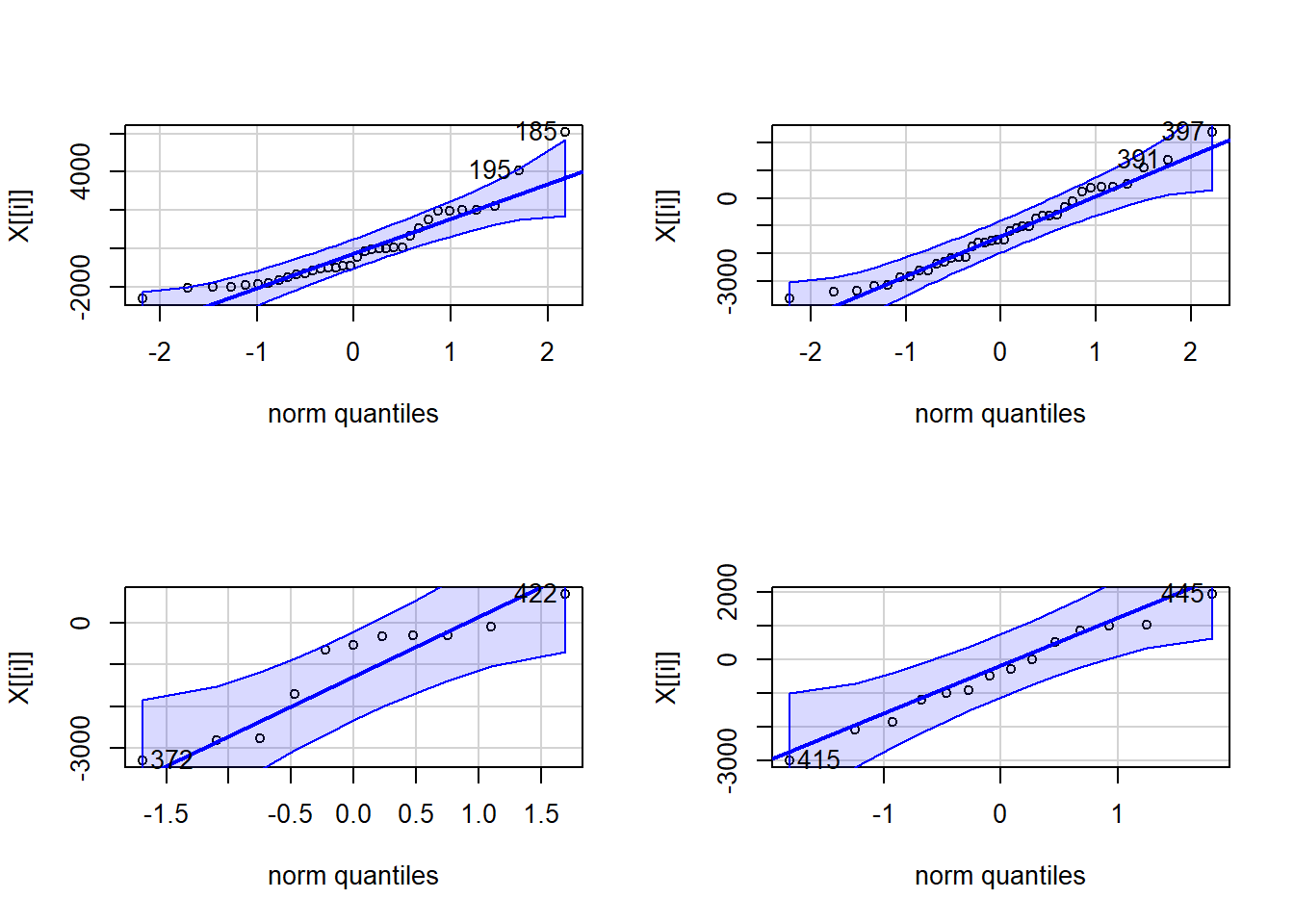
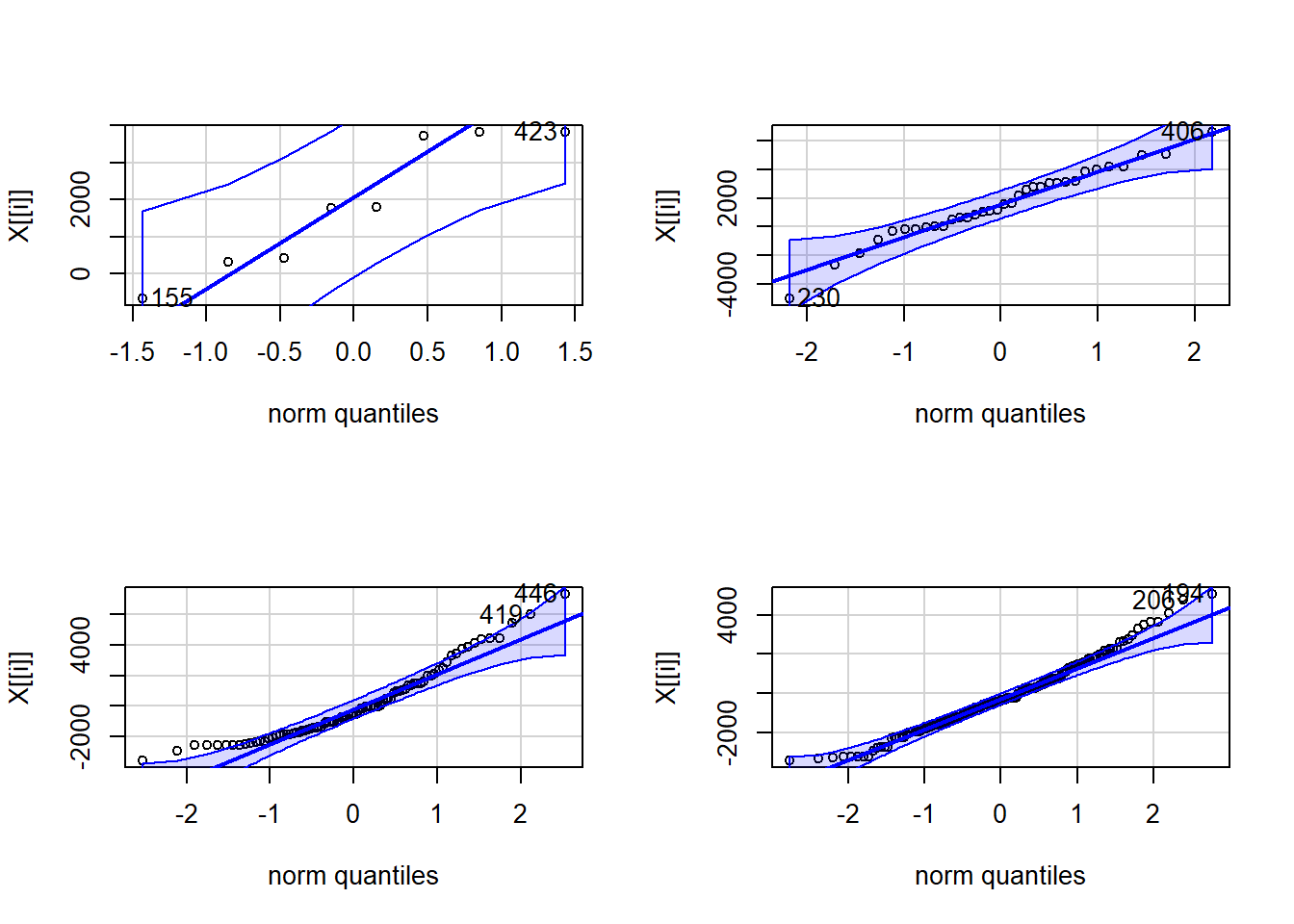


# Not required for this assignment

library(car)

par(mfrow=c(2,2)) #combined the plots for better output

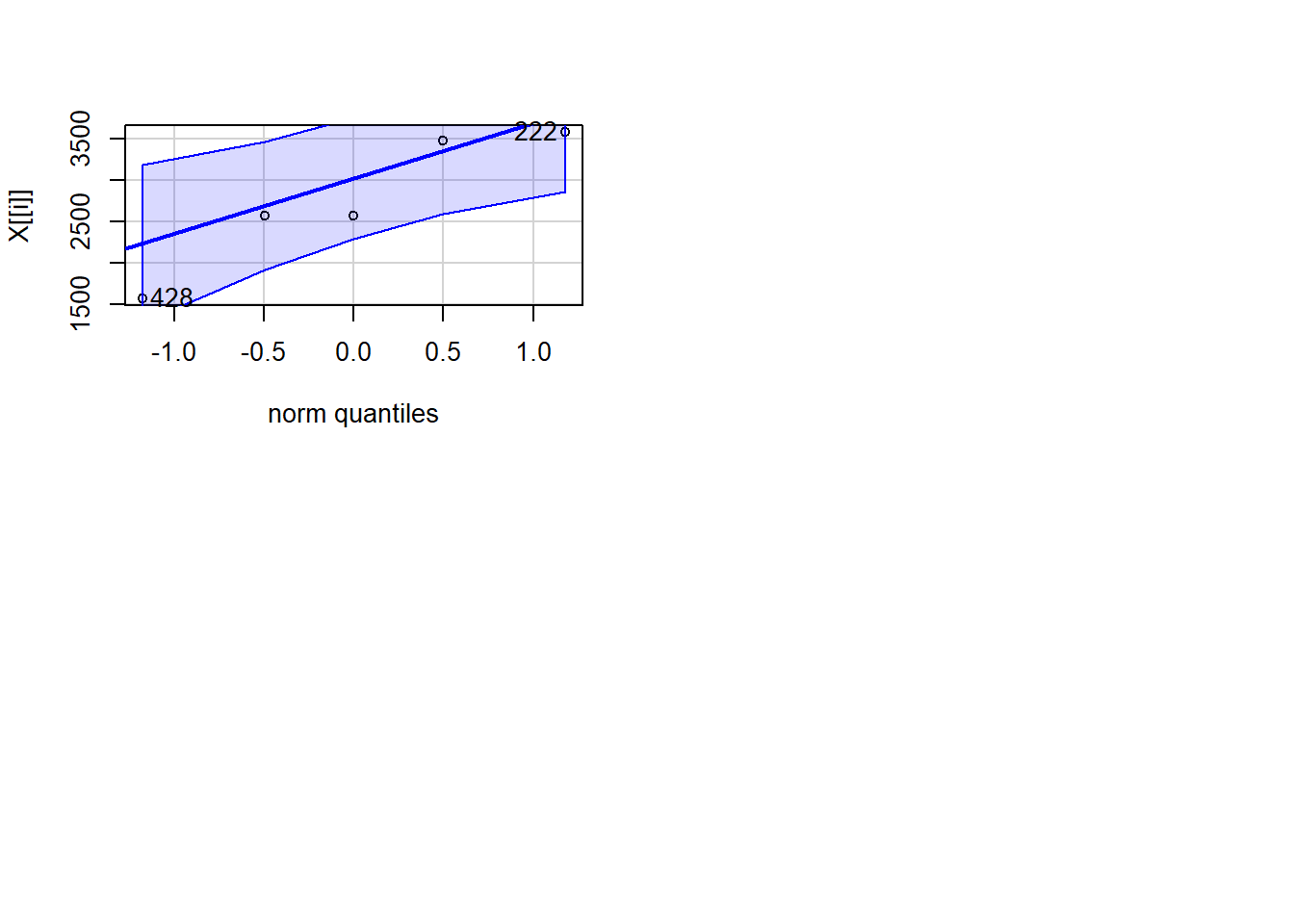
sapply(split(modq1$residuals, MyCars$Age), qqPlot)



## 0 1 2 3 4 5 6 7 8 9 10 11 12

## 155 2 18 87 86 21 33 9 13 9 9 10 4 5

## 423 7 31 80 88 22 32 11 14 3 2 1 2 3



1. Find the car in your sample with the largest residual (in magnitude - positive or negative). For that car, find its standardized and studentized residual. Based on these residuals, could this value be considered influential?

**0.5 points** largest absolute residual  
**0.5 points** standarized residual  
**0.5 points** studentized residual  
**0.5 points** discussion of influence. This will likely be based on the similarity or difference in the standardized and studentized residual. If they are similar, that point likely has little influence. They should not earn points for saying the point is an outlier, with a standardized/studentized residual over 2 or 3, and is hence influential. This is not always true. For my values, the standardized and studentized values are similar for each index, so it does not seem they that have much influence on the model. They may also find the cooks distance and compare the value to 0.5 or 1 to determine if it has a large influence.

which.max(abs(modq1$resid))

## 446

## 446

rstandard(modq1)[446]

## 446

## 3.729341

rstudent(modq1)[446]

## 446

## 3.784635

cooks.distance(modq1)[446]

## 446

## 0.02335437

1. Determine the leverages for the cars with the ten largest absolute residuals. What do these leverage values say about the potential for each of these ten cars to be influential on your model?

**1.0 point** Determine leverages  
**1.0 point** Discuss potential influence from leverage. Should compare to 2(2/sample size) and 3(2/sample). If the leverage of their value is more than double or triple the average leverage, then the point has higher potential to influence the model.

Note: Leverage measures potential for influence. This alone does not determine if the points actually have influence. They do not need to use the manner that I did to find the indices and check the leverages.

# I assume that there are better ways to do this...

# Extracts the indices for the 10 largest absolute residuals

top\_resid\_indices = sort(

abs(

modq1$resid

),

decreasing=TRUE,

index.return=TRUE

)$ix[1:10]

# Find the leverages for the points with the 10 largest absolute residuals

hatvalues(modq1)[top\_resid\_indices]

## 446 406 185 419 208 194

## 0.003347170 0.005085658 0.002276505 0.003347170 0.003347170 0.002410786

## 186 230 386 206

## 0.005085658 0.005085658 0.005085658 0.002410786

# Compares leverages to 2(2/n)

# Since no leverages are above this 2(2/n), the statement is false for all 10 points

# This make the line below sum to 0.

# This implies that these 10 points have low potential to influence the model

sum(hatvalues(modq1)[top\_resid\_indices] > (2 \* 2/dim(MyCars)[1]))

## [1] 0

1. Determine the Cook’s distances for the cars with the ten largest absolute residuals. What do these Cook’s distance values say about the influence of each of these ten cars on your model?

**0.5 point** Determine Cook’s distances  
**0.5 point** Discuss influence based on comparing Cook’s Distance values to 0.5 or 1.0

cooks.distance(modq1)[top\_resid\_indices]

## 446 406 185 419 208 194

## 0.023354365 0.028754590 0.010959237 0.015632738 0.013006155 0.008060807

## 186 230 386 206

## 0.016869283 0.016503209 0.016442434 0.007136331

# Since no Cook's distances are greater than 0.5, this sums to 0 (all FALSE values)

sum(cooks.distance(modq1)[top\_resid\_indices] > 0.5)

## [1] 0

1. Compute and interpret in context a 90% confidence interval for the slope of your regression line.

**0.5 point** code for interval  
**1.0 point** explanation - With 90% confidence the prices of this model car decrease between $1377.09 and $1247.60 as the age increases by one year

confint(modq1, level=.90)

## 5 % 95 %

## (Intercept) 17889.589 18454.264

## Age -1377.091 -1247.599

1. Test the strength of the linear relationship between your variables using each of the three methods (test for correlation, test for slope, ANOVA for regression). Include hypotheses for each test and your conclusions in the context of the problem.

**0.5 points** cor.test() code  
**0.5 points** cor.test() hypotheses and conclusion; Null: *ρ*

= 0, Alternative *ρ* ≠ 0. They can write this in notation or in words. Conclusion should note the p-value from the code, either explicitly stating its value or noting that it is very small, and then reject the null or fail to support the alternative either with those words explicitly or inferred.  
**0.5 points** slope test code (likely just summary())  
**0.5 points** slope test hypotheses and conclusion; Same as cor.test, but with *β* instead of *ρ*

**0.5 points** anova() code (or anova455() or from summary())  
**0.5 points** anova hypotheses and conclusion; same as slope test

**Note:** They should not mention significance or p-values in their hypotheses. The hypotheses have nothing to do with these things. The p-values will help support or fail to support the hypotheses. Give 0 points for any of the three hypotheses that do this. The hypotheses state the the parameter is equal to 0 vs it is not equal to 0.

cor.test(MyCars$Age, MyCars$Price)

##

## Pearson's product-moment correlation

##

## data: MyCars$Age and MyCars$Price

## t = -33.409, df = 446, p-value < 2.2e-16

## alternative hypothesis: true correlation is not equal to 0

## 95 percent confidence interval:

## -0.8698098 -0.8165831

## sample estimates:

## cor

## -0.8452806

summary(modq1)

##

## Call:

## lm(formula = Price ~ Age, data = MyCars)

##

## Residuals:

## Min 1Q Median 3Q Max

## -4971.6 -1407.0 -250.2 1212.6 7302.8

##

## Coefficients:

## Estimate Std. Error t value Pr(>|t|)

## (Intercept) 18171.93 171.29 106.09 <2e-16 \*\*\*

## Age -1312.35 39.28 -33.41 <2e-16 \*\*\*

## ---

## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

##

## Residual standard error: 1961 on 446 degrees of freedom

## Multiple R-squared: 0.7145, Adjusted R-squared: 0.7139

## F-statistic: 1116 on 1 and 446 DF, p-value: < 2.2e-16

anova(modq1)

|  |
| --- |
|  |

|  | **Df**  **<int>** | **Sum Sq**  **<dbl>** | **Mean Sq**  **<dbl>** | **F value**  **<dbl>** | **Pr(>F)**  **<dbl>** |
| --- | --- | --- | --- | --- | --- |
| Age | 1 | 4294333596 | 4294333596 | 1116.167 | 1.778551e-123 |
| Residuals | 446 | 1715936839 | 3847392 | NA | NA |

2 rows

# May also use anova455

source("https://raw.githubusercontent.com/JA-McLean/STOR455/master/scripts/anova455.R")

anova455(modq1)

|  |
| --- |
|  |

|  | **Df**  **<dbl>** | **Sum Sq**  **<dbl>** | **Mean Sq**  **<dbl>** | **F value**  **<dbl>** | **P(>F)**  **<dbl>** |
| --- | --- | --- | --- | --- | --- |
| Model | 1 | 4294333596 | 4294333596 | 1116.167 | 0 |
| Error | 446 | 1715936839 | 3847392 | NA | NA |
| Total | 447 | 6010270436 | NA | NA | NA |

3 rows

1. Suppose that you are interested in purchasing a car of this model that is four years old (in 2017). Determine each of the following: 90% confidence interval for the mean price at this age and 90% prediction interval for the price of an individual car at this age. Write sentences that carefully interpret each of the intervals (in terms of car prices).

**1.0 points** - Construct dataframe with single car of age 4. Give 0 points if they instead take a subset from their sample of all cars that are of age 3.  
**1.0 points** - code for confidence interval.  
**1.0 points** - code for predcition interval.  
**0.5 points** - sentence interpreting the confidence interval, such as with 90% confidence I predict that the mean price of all 4 year old Civics sold in NY is between $12768.29 and $13076.8  
**0.5 points** - sentence interpreting the prediction interval, such as with 90% confidence I predict that the price of a 4 year old Civic sold in NY is between $9685.81 and $16159.28

Note: They don’t need sentences with this exact wording, but it should be clear that the confidence interval is a prediction for the mean price of all cars like this, while the prediction interval is predicting the price of one specific car. It should also be clear that there is 90% confidence in the process.

single\_car = data.frame(Age = 4)

predict.lm( modq1, single\_car, interval = "confidence", level = 0.90)

## fit lwr upr

## 1 12922.55 12768.29 13076.8

predict.lm( modq1, single\_car, interval = "prediction", level = 0.90)

## fit lwr upr

## 1 12922.55 9685.81 16159.28

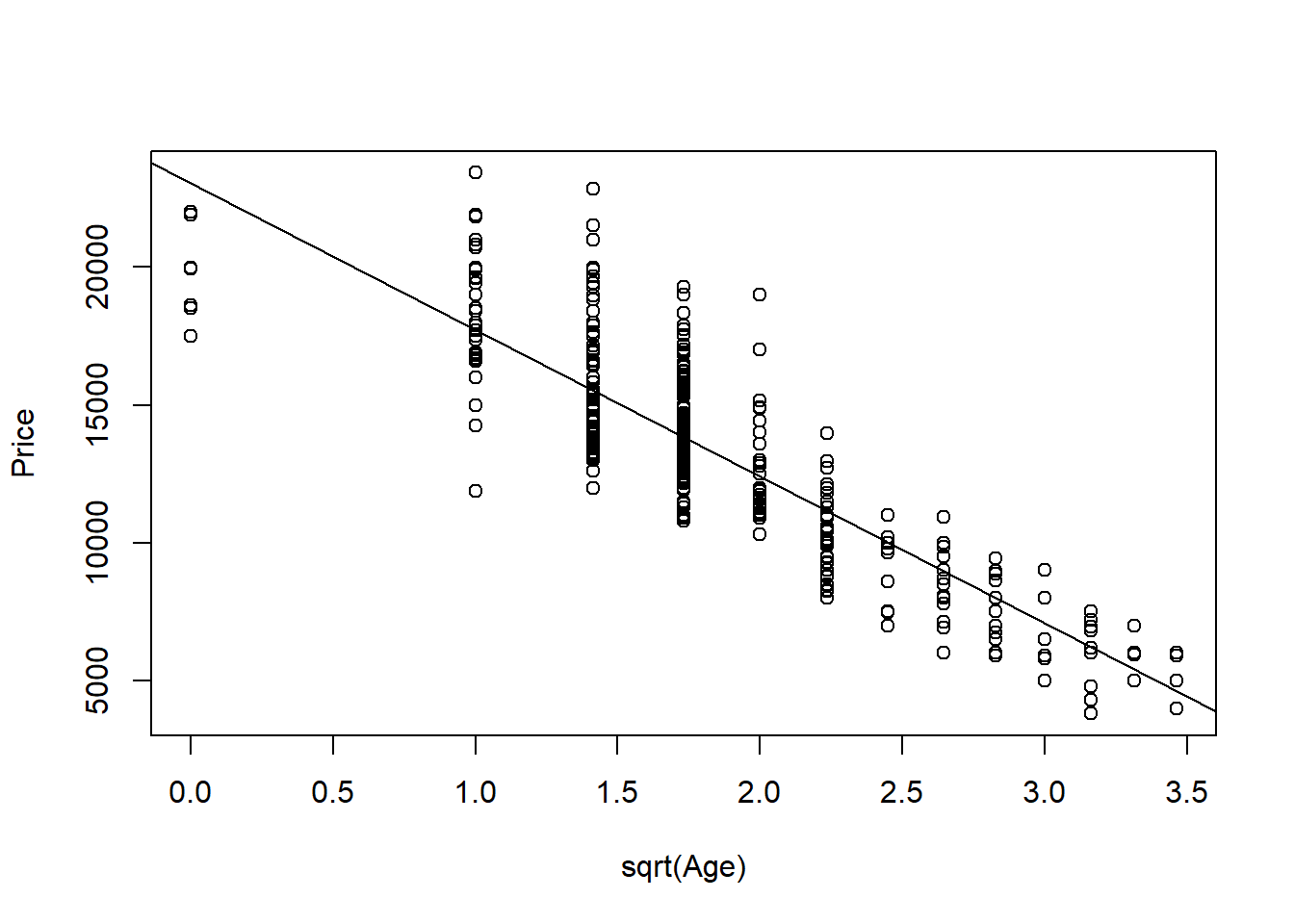
1. Experiment with some transformations to attempt to find one that seems to do a better job of satisfying the linear model conditions. Include the summary output for fitting that model and a scatterplot of the original data with this new model (which is likely a curve on the original data). Explain why you think that this transformation does or does not improve satisfying the linear model conditions.

**2 points** - transformation of some kind that tries to improve the model. It’s possible that for some students, no transformation is needed, but they should still show the attempt to improve the model.  
**1 points** - Discussion of how the transformed model improves at least one of the conditions for a linear model. Or, if no better transformed model is found, a discussion of how the transformation did not improve linear model conditions.  
**1 points** - Plot of the best model on the orginal axes.

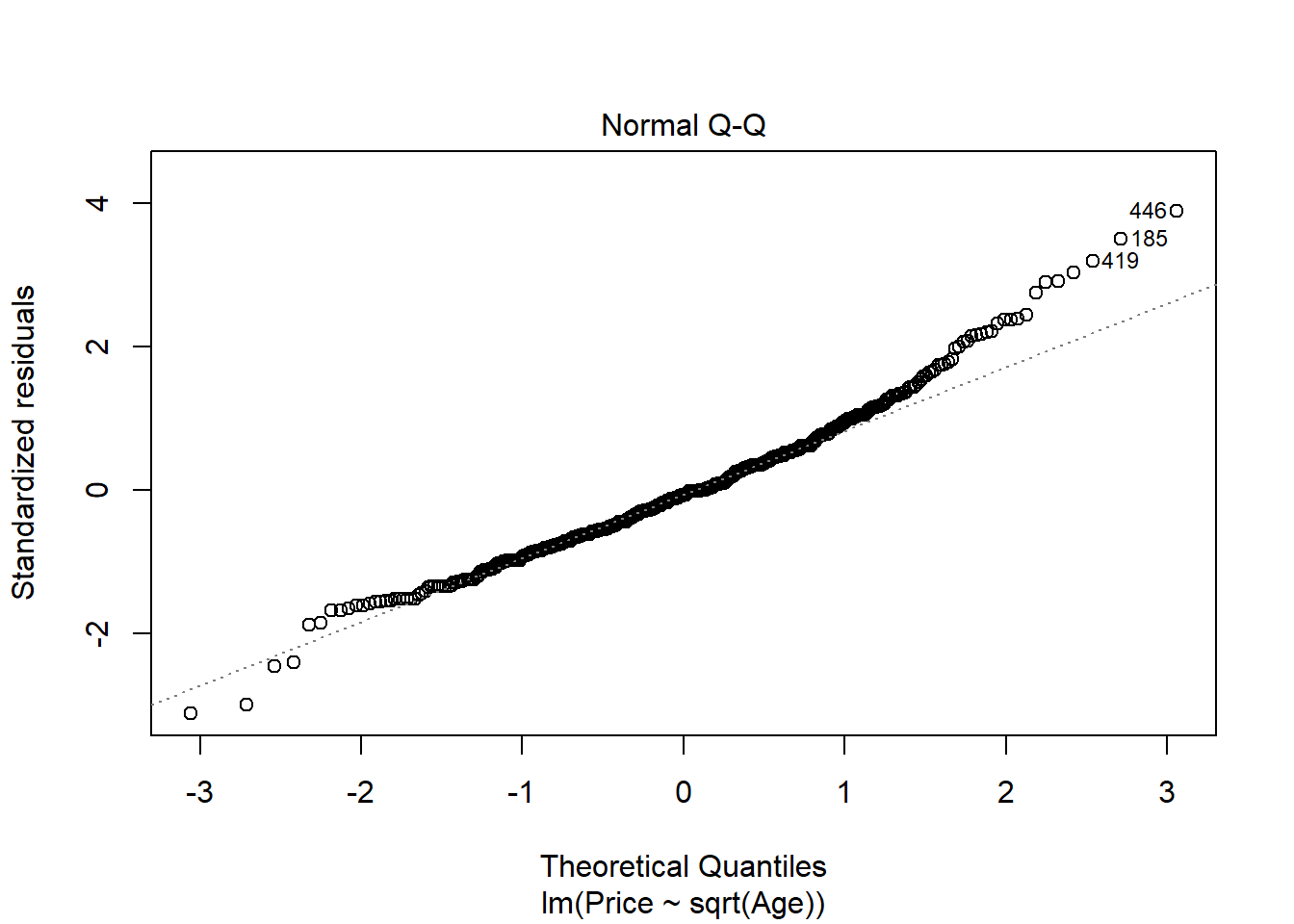
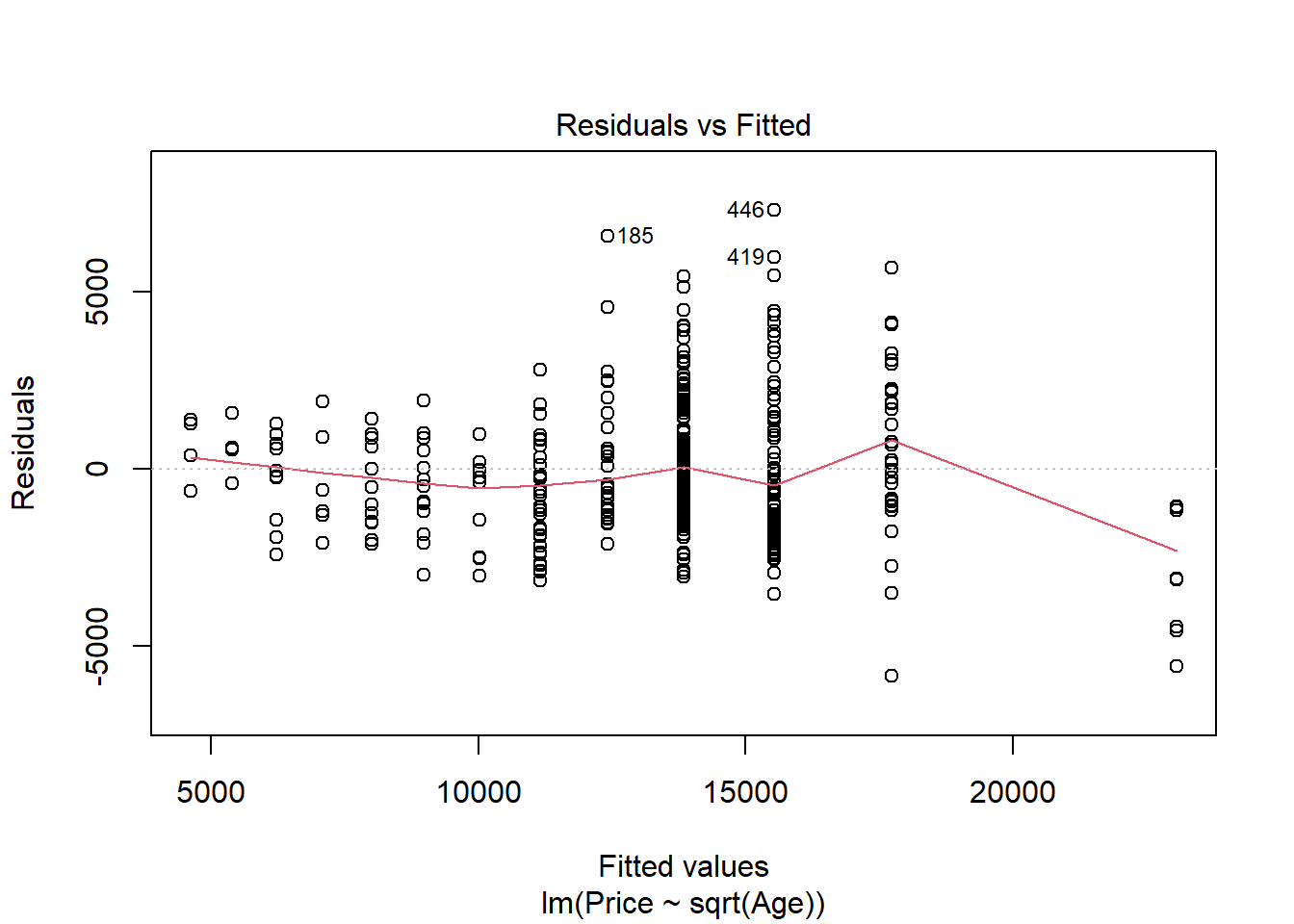
modq9.1 = lm(Price~sqrt(Age), data=MyCars)

plot(Price~sqrt(Age), data=MyCars)

abline(modq9.1)



plot(modq9.1, 1:2)



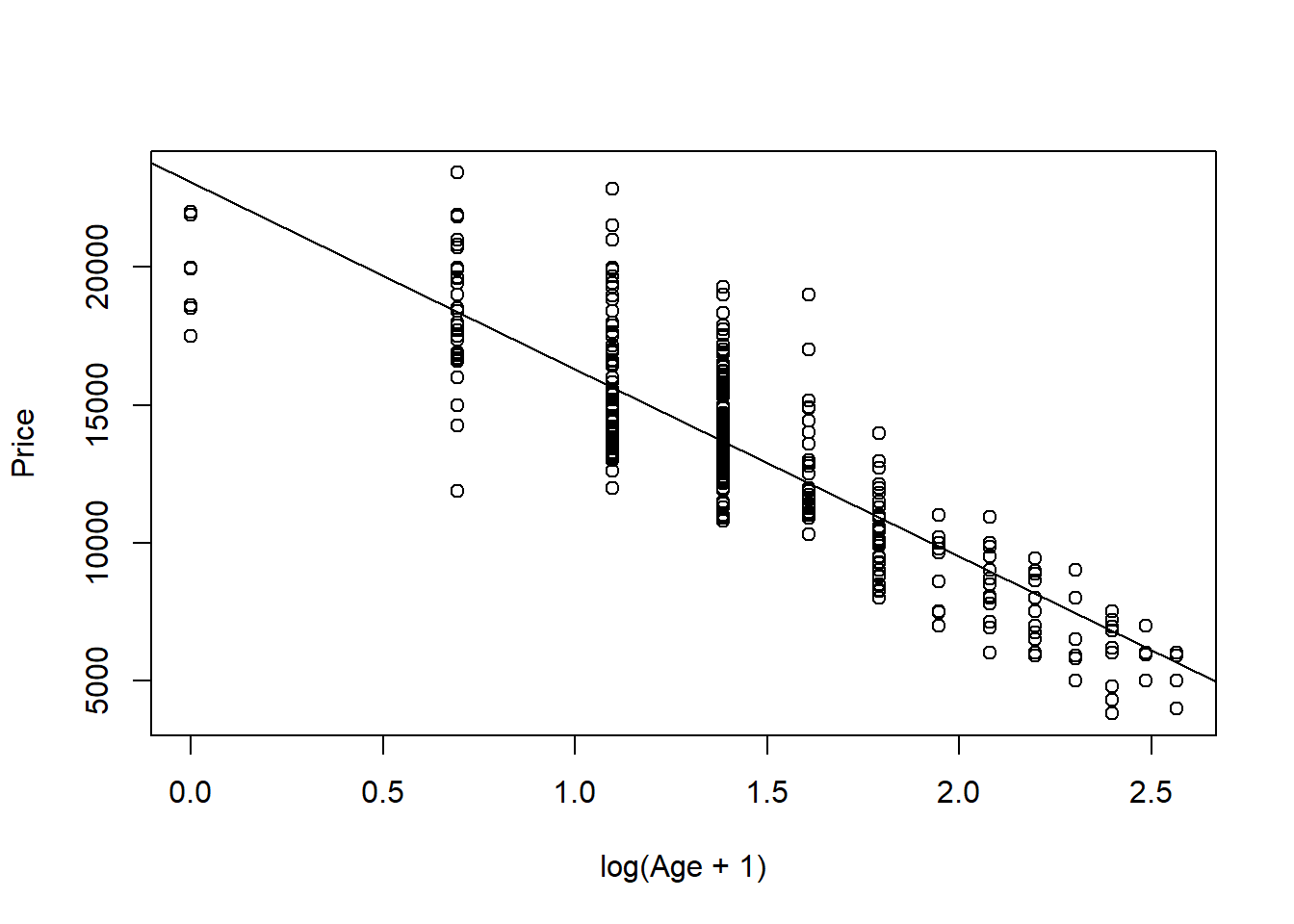
# Since there are 0s in the data, I added 1 to Age so I could use log

# I didn't do this in class, but some students may do this.

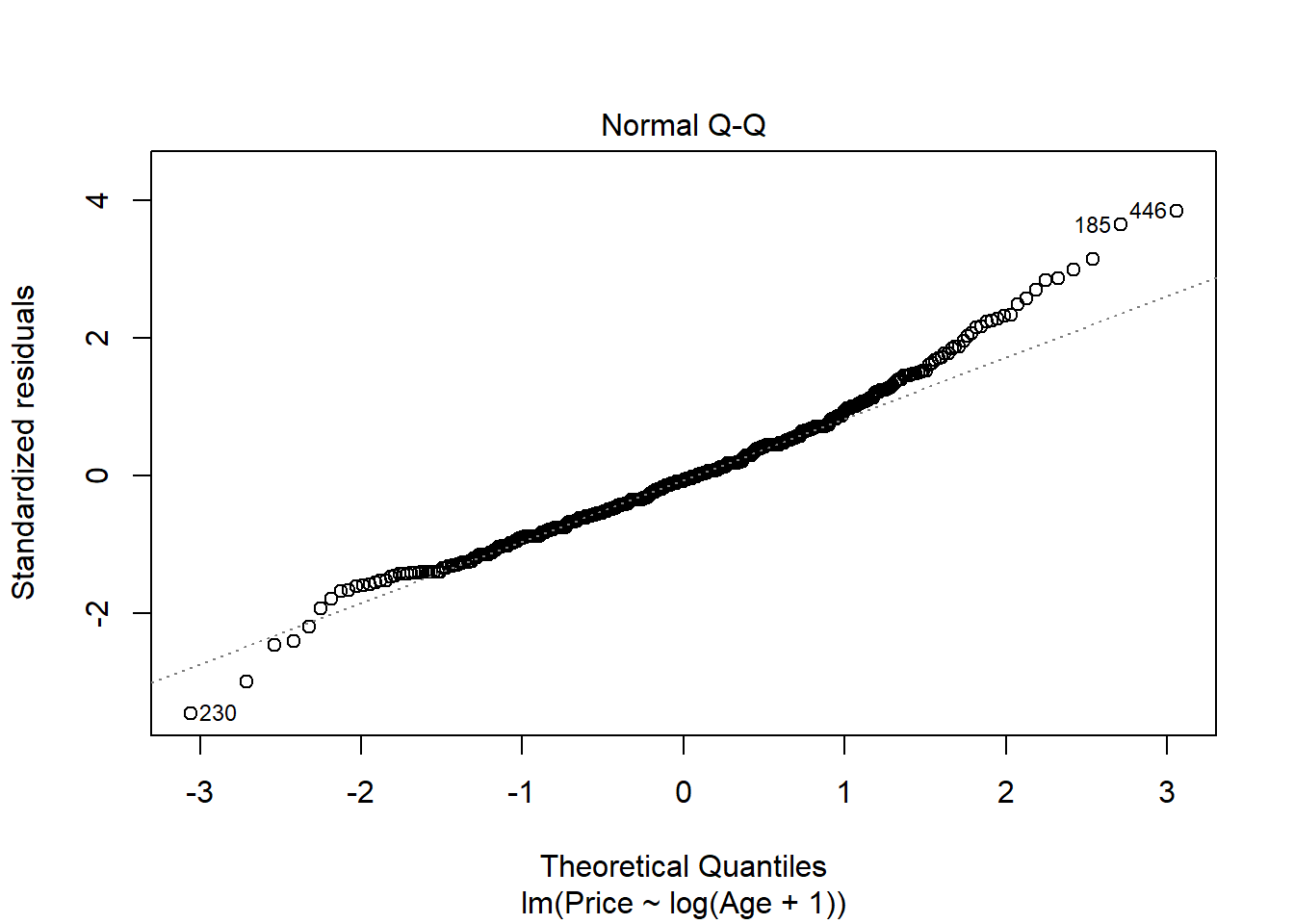
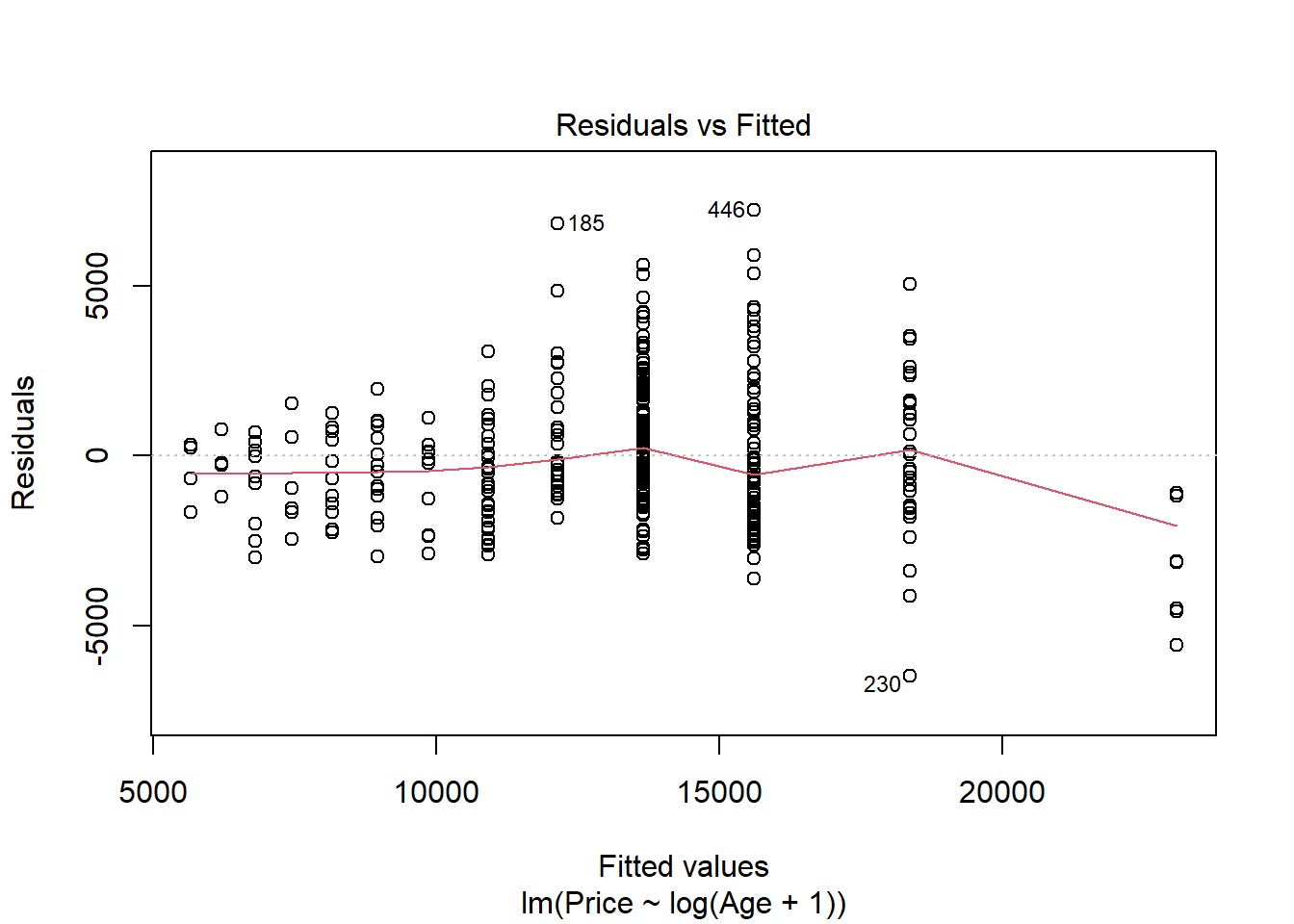
modq9.2 = lm(Price~log(Age+1), data=MyCars)

plot(Price~log(Age+1), data=MyCars)

abline(modq9.2)



plot(modq9.2, 1:2)

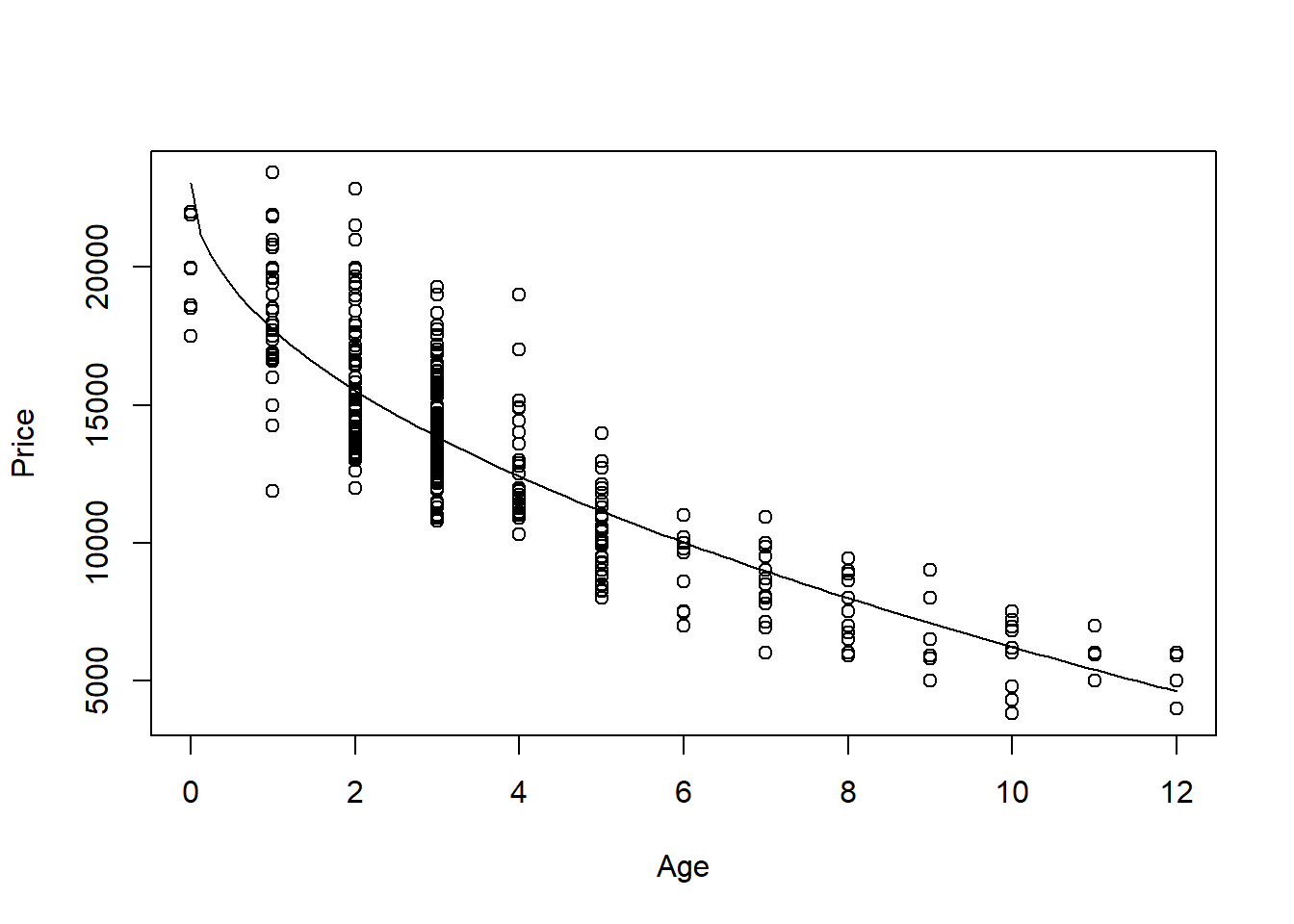


plot(Price~Age, data=MyCars)

B0 = summary(modq9.1)$coef[1,1]

B1 = summary(modq9.1)$coef[2,1]

curve(B0 + B1\*sqrt(x), add=TRUE)



1. According to your transformed model, is there an age at which the car should be free? If so, find this age and comment on what the “free car” phenomenon says about the appropriateness of your model.

**1.0 points** - Find (or approximate) at what age the car has a price of zero.  
**1.0 points** - Brief description of the meaning here. This could relate to the price that the car is worth $0 being well outside of the range of ages in the sample, which leads to extrapolation. The model may fit the given data, but not fit data for a wider range of cars. Or they may say somethng along the lines that you could sell the car up until this age, after that you’ll need to pay someone to take it away. Anything reasonable is fine.

Note: They will likely approximate this by manually solving for Price=0 in their linear model. Other methods such as approximating from the plot (shown below) or use of specific functions are fine as well. It’s possible that their models will never have a car worth $0, which is fine for full credit as well.

# For modq9.1 Car price is $0 at almost 19 years old

((-1\*B0)/B1)^2

## [1] 18.74863

1. Again suppose that you are interested in purchasing a car of this model that is four years old (in 2017). Determine each of the following using your model constructed in question 9: 90% confidence interval for the mean price at this age and 90% prediction interval for the price of an individual car at this age. Write sentences that carefully interpret each of the intervals (in terms of car prices).

**1.0 points** - code for confidence interval.  
**1.0 points** - code for predcition interval.  
**0.5 points** - sentence interpreting the confidence interval, such as with 90% confidence I predict that the mean price of all 4 year old Civic sold in NY is between $12256.62 and 12562.88  
**0.5 points** - sentence interpreting the prediction interval, such as with 90% confidence I predict that the price of a 4 year old Civic sold in NY is between $9303.123 and 15516.37

Note: They don’t need sentences with this exact wording, but it should be clear that the confidence interval is a prediction for the mean price of all cars like this, while the prediction interval is predicting the price of one specific car. It should also be clear that there is 90% confidence in the process. For my transformed model, I only transformed the predictor. This left my intervals below in dollars. If students used a transformation that transformed the response, then the intervals below will be in terms of transformed dollars. They must “untransform” the intervals to be back in dollars to earn full credit.

predict.lm( modq9.1, single\_car, interval = "confidence", level = 0.90)

## fit lwr upr

## 1 12409.75 12256.62 12562.88

predict.lm( modq9.1, single\_car, interval = "prediction", level = 0.90)

## fit lwr upr

## 1 12409.75 9303.123 15516.37

**MODEL #2: Use Mileage as a predictor for Price**

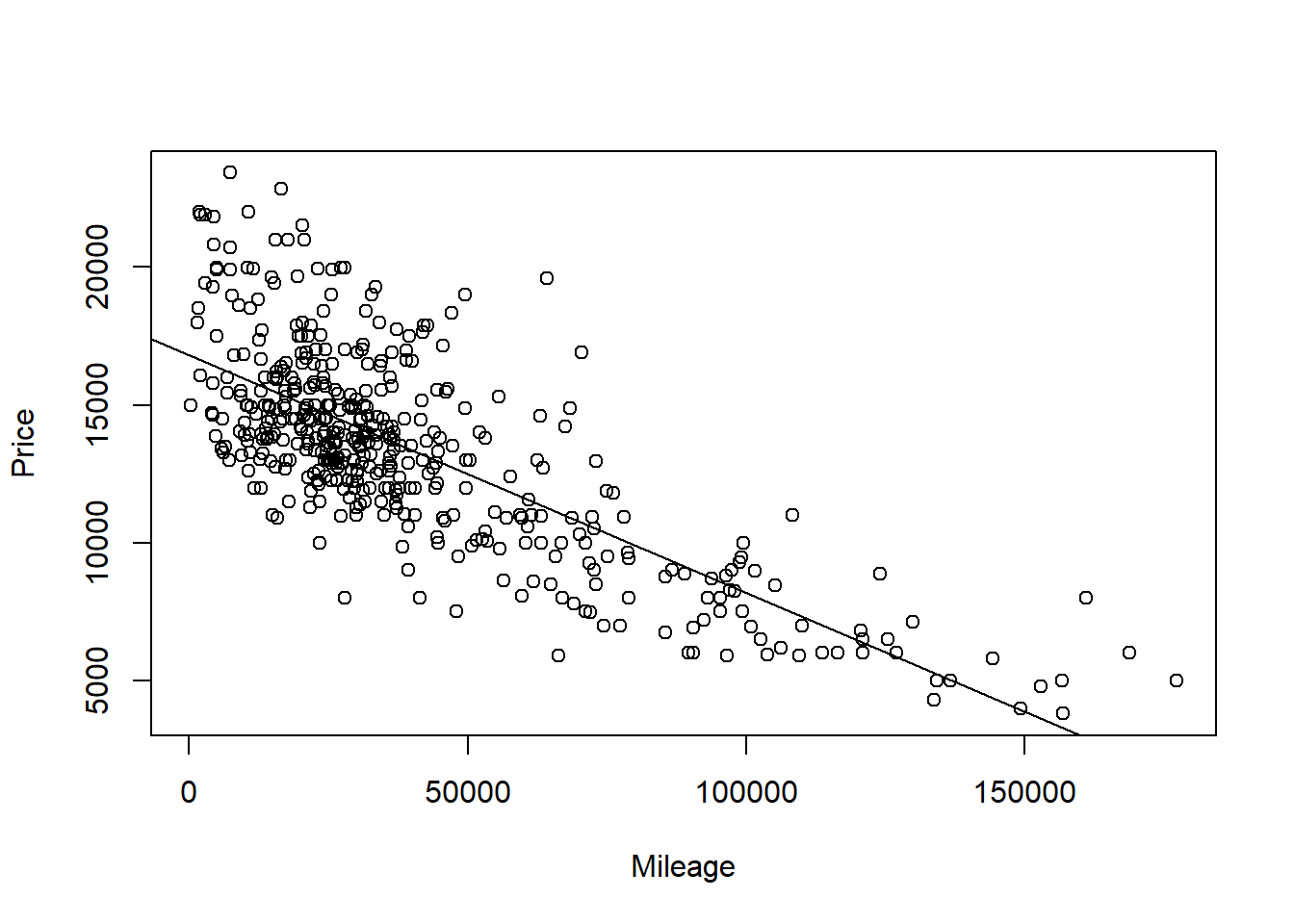
1. Calculate the least squares regression line that best fits your data (with *Mileage* now as the predictor) and produce a scatterplot of the relationship with the regression line on it.

**0.5 point** code for model  
**0.5 point** code for plot  
**0.5 point** abline

modq12 = lm(Price~Mileage, data=MyCars)

plot(Price~Mileage, data=MyCars)

abline(modq12)



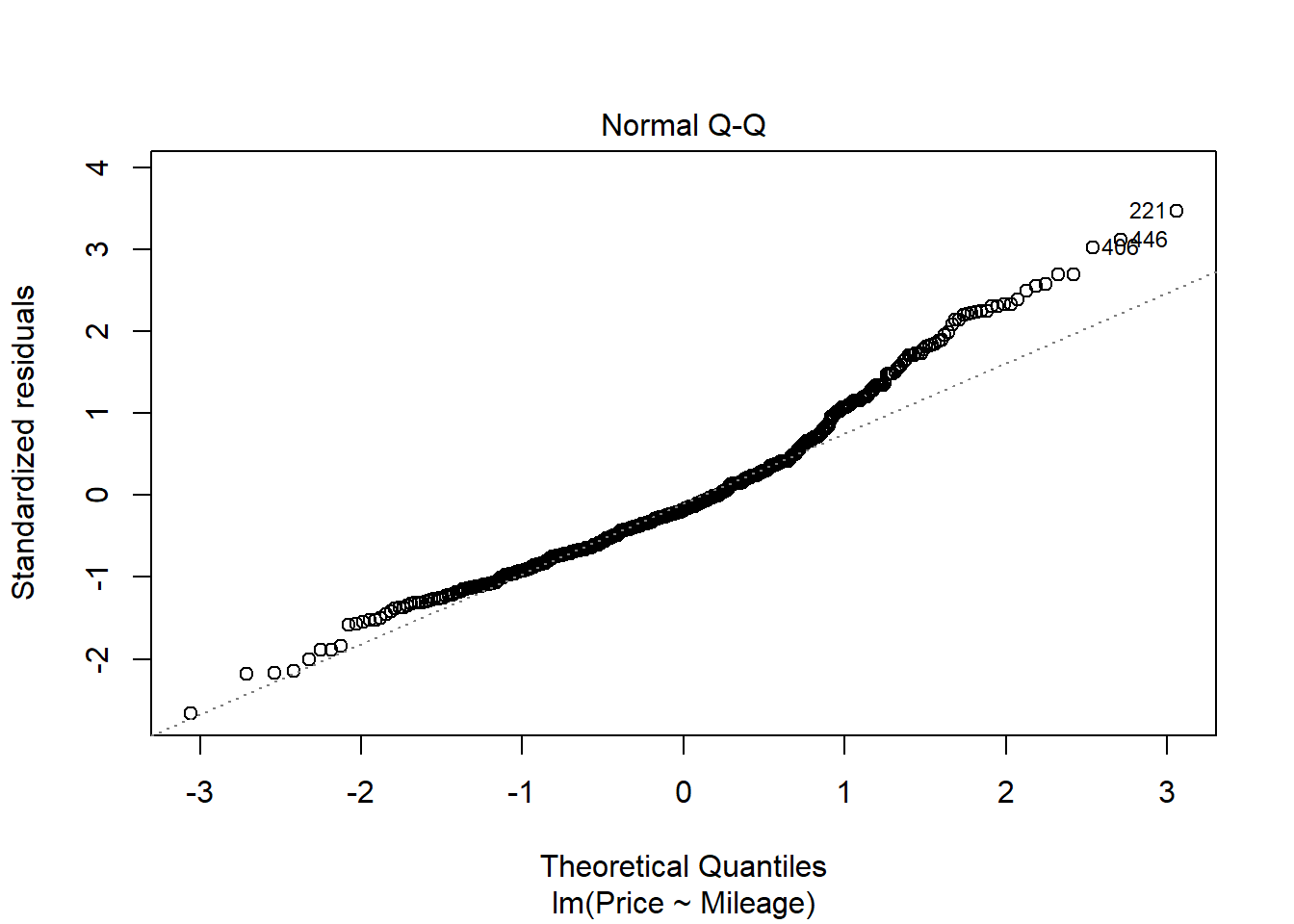
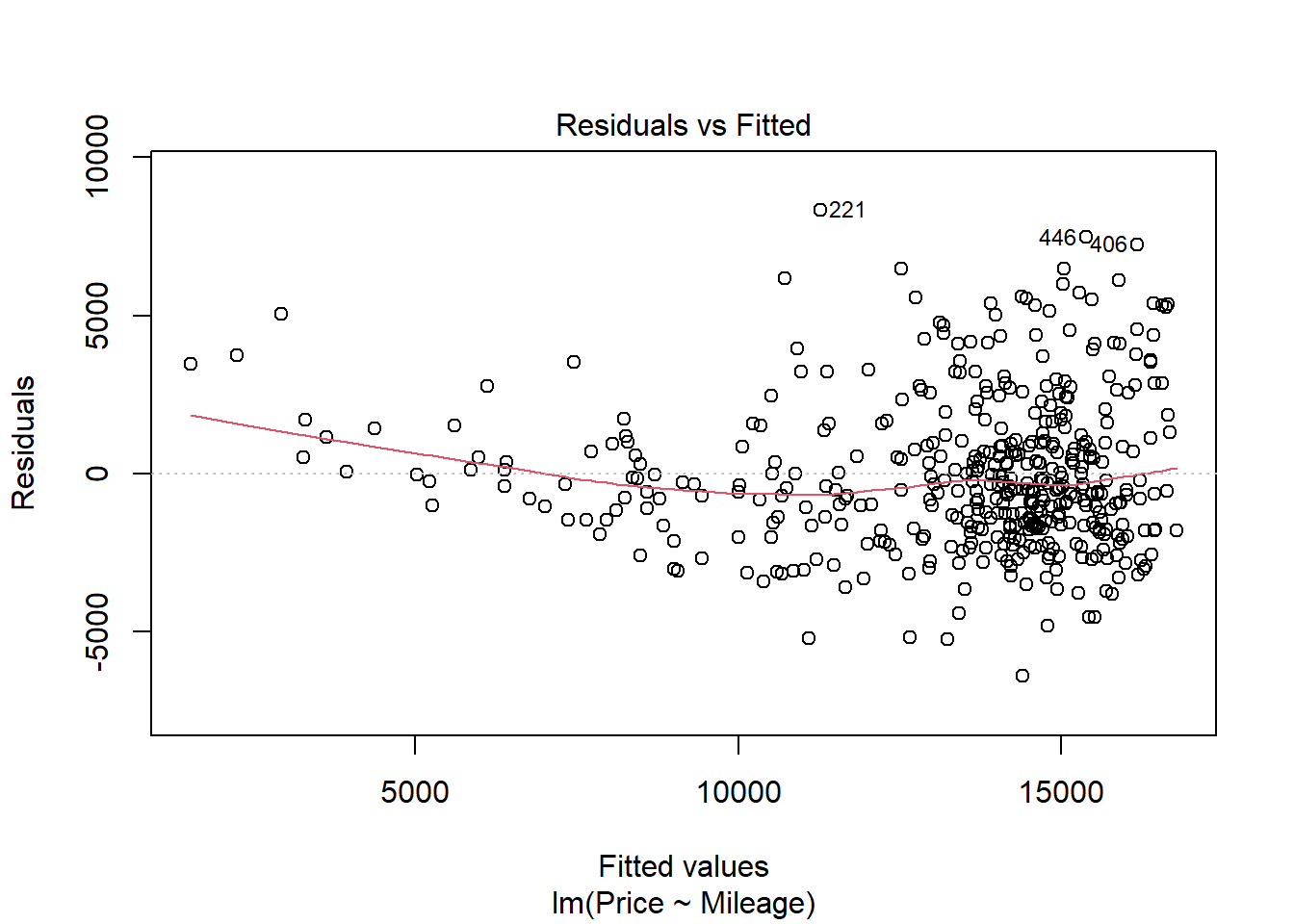
1. Produce appropriate residual plots and comment on how well your data appear to fit the conditions for a simple linear model. Don’t worry about doing transformations at this point if there are problems with the conditions.

**0.5 points** residuals vs fitted plot  
**0.5 points** qqnorm (or histogram) for Normality of residuals  
**1.5 points** discussion of conditions (linearity, constant variance, and normality of residuals) You can give 0.5 pts each. They can describe the conditions without explicitly using these terms.

Note 1: For linearity, they should have some discussion if the line seems to describe the data, using either the scatter plot or residual vs fitted plot. For constant variance they should discuss if the variability (vertical distances) from the line seems to follow any pattern as value of the predictor changes. For normality of the residuals, they should note the adherence of the residuals (or not) to the qqline, or bell curve shape or skew in a histogram. As each student will have a different plot, any reasonable assertions of the conditions being met (or not) supported by an argument is fine for full credit.

Note 2: they may use the plot(model) to produce all of the plots at once, or separately produce each of the plots with different lines of code.

plot(modq12, 1:2)



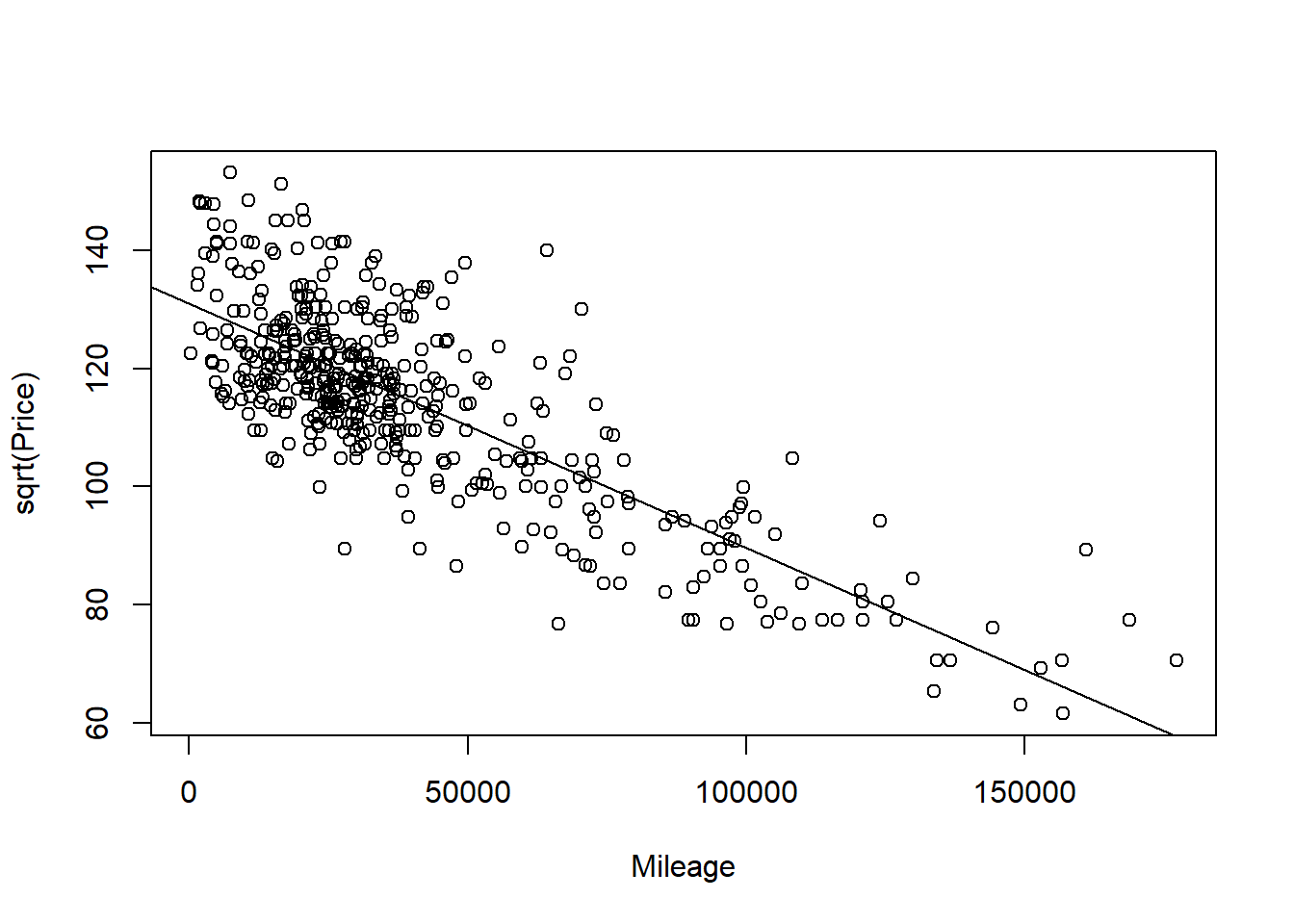
1. Experiment with some transformations to attempt to find one that seems to do a better job of satisfying the linear model conditions. Include the summary output for fitting that model and a scatterplot of the original data with this new model (which is likely a curve on the original data). Explain why you think that this transformation does or does not improve satisfying the linear model conditions.

**2 points** - transformation of some kind that tries to improve the model. It’s possible that for some students, no transformation is needed, but they should still show the attempt to improve the model.  
**1 points** - Discussion of how the transformed model improves at least one of the conditions for a linear model. Or, if no better transformed model is found, a discussion of how the transformation did not improve linear model conditions.  
**1 points** - Plot of the best model on the orginal axes.

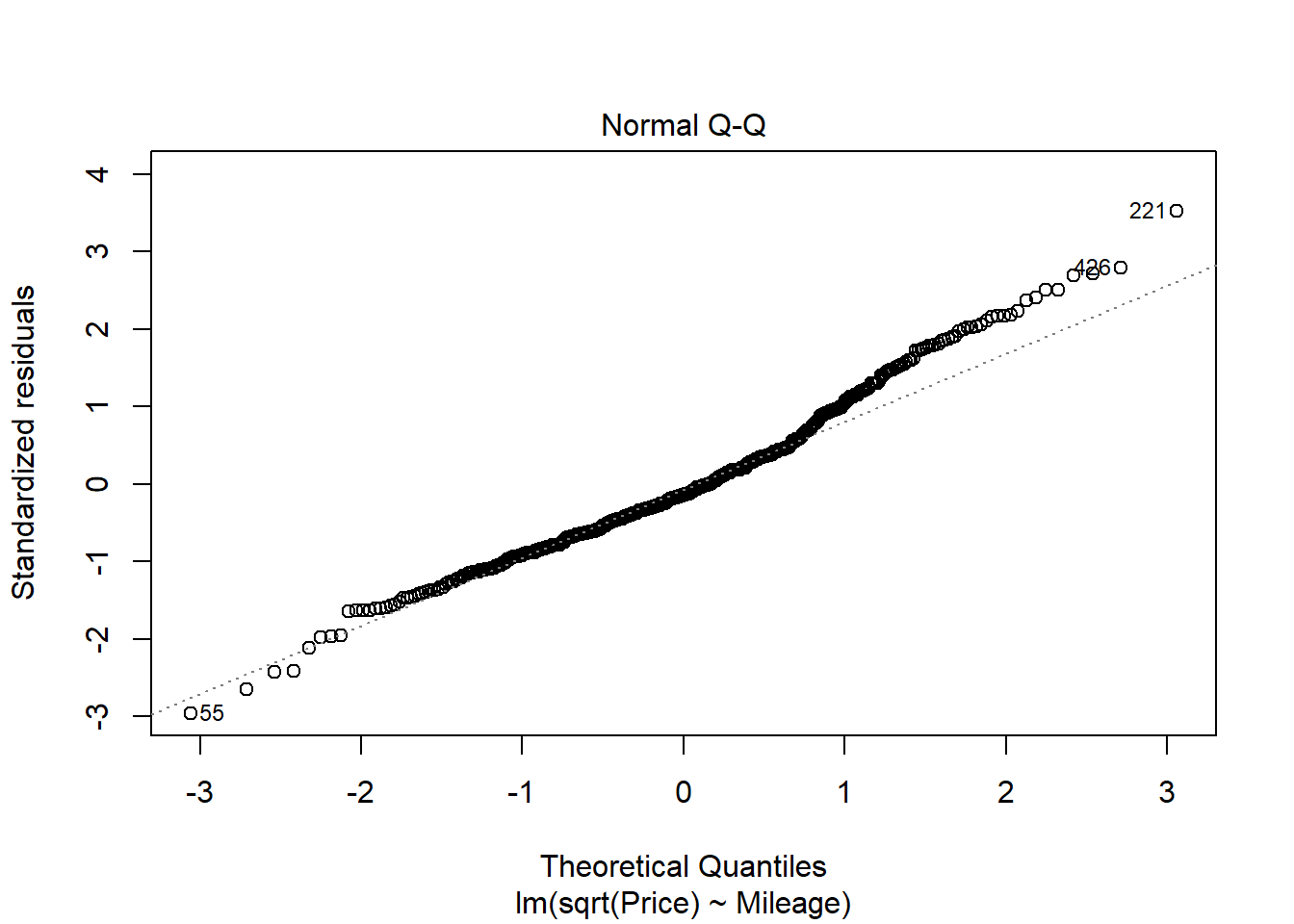
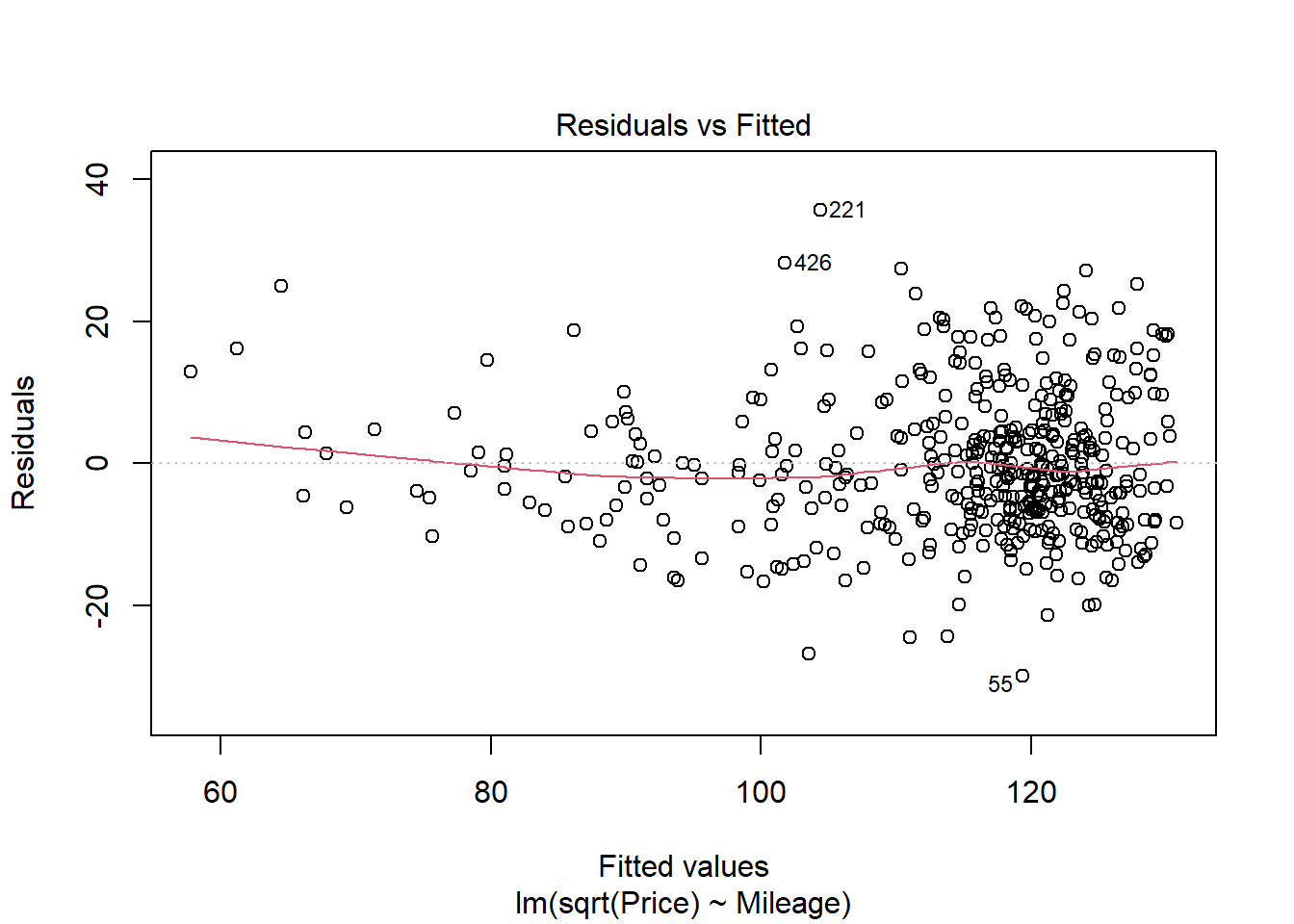
modq14.1 = lm(sqrt(Price)~Mileage, data=MyCars)

plot(sqrt(Price)~Mileage, data=MyCars)

abline(modq14.1)



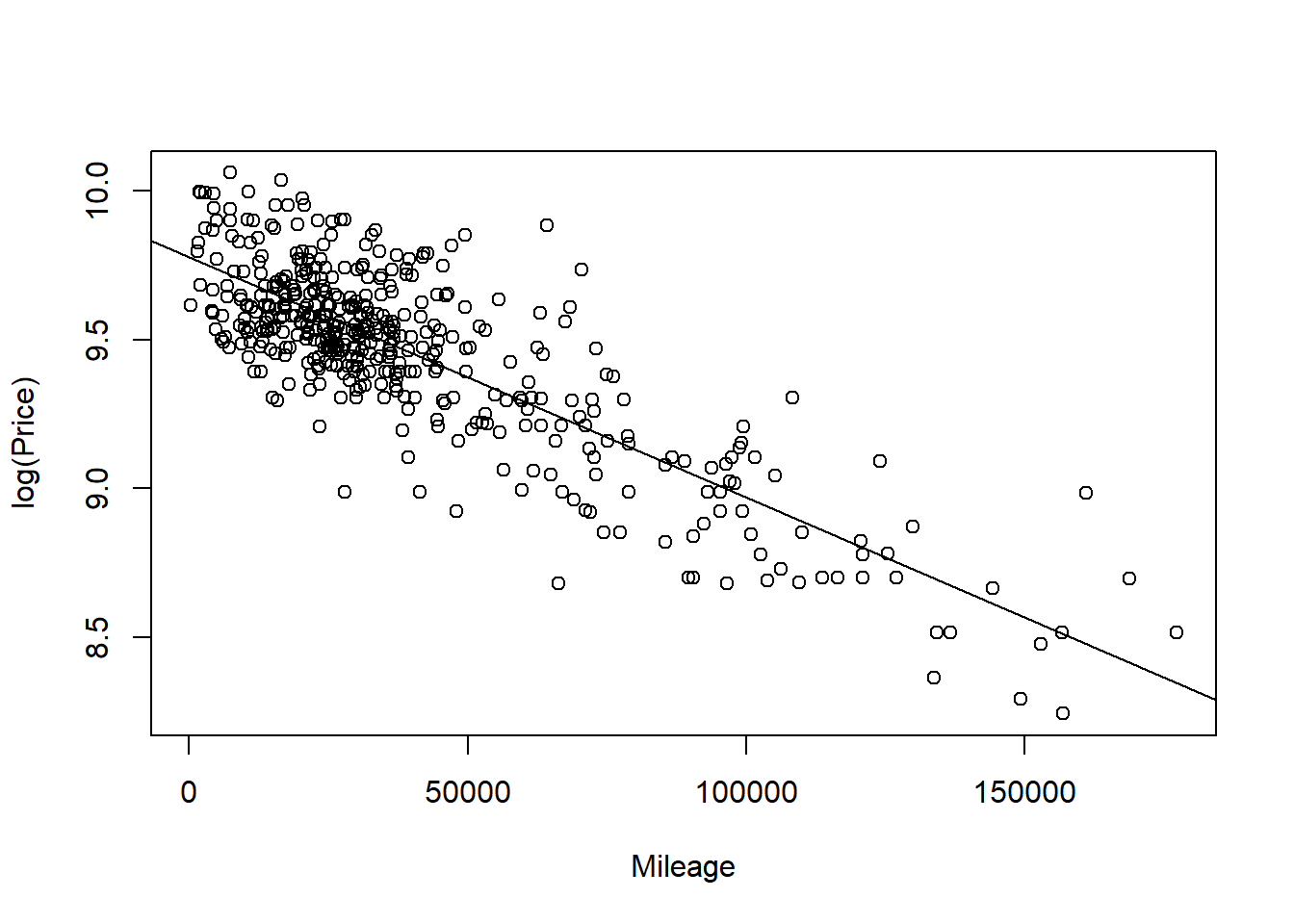
plot(modq14.1, 1:2)



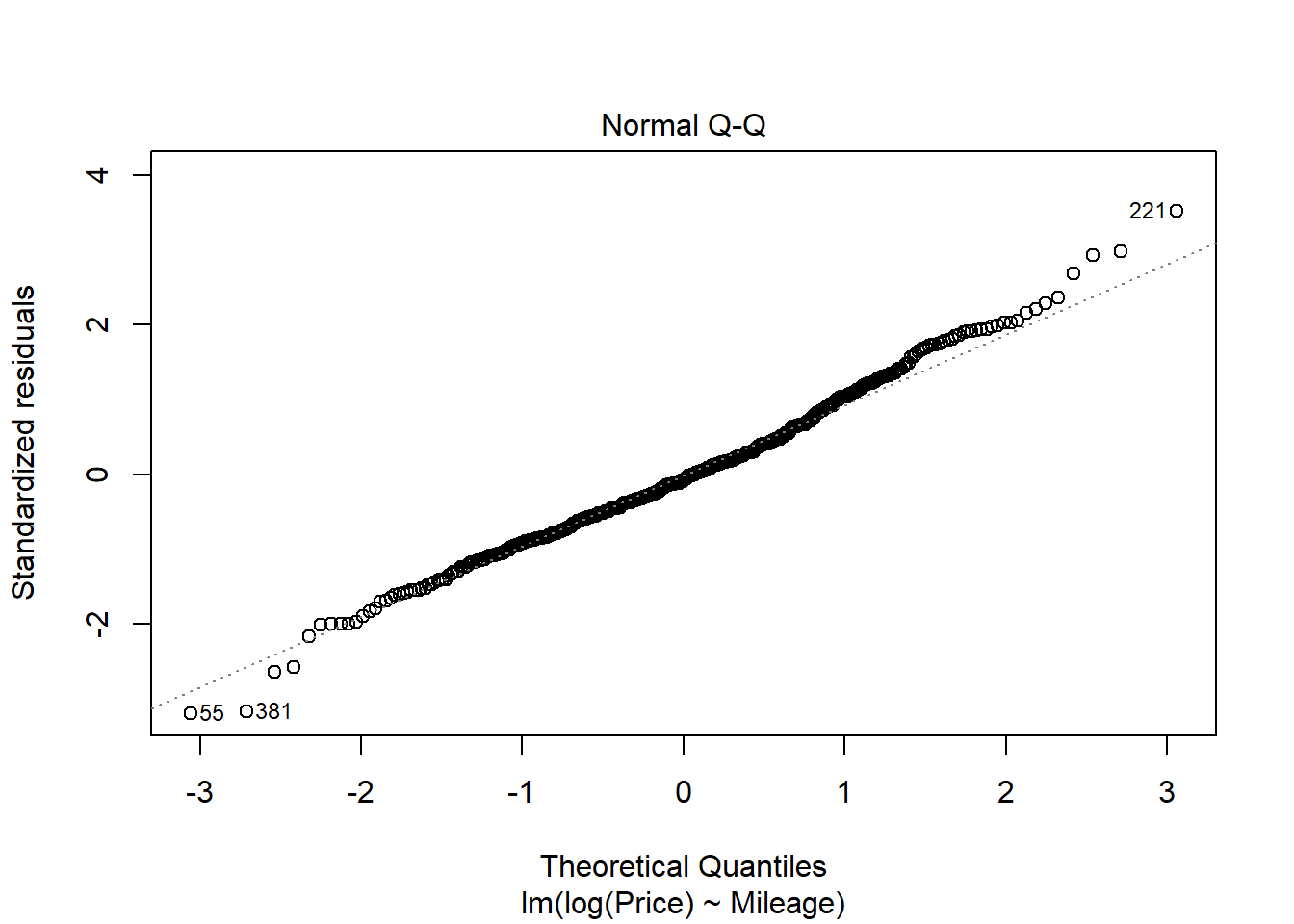
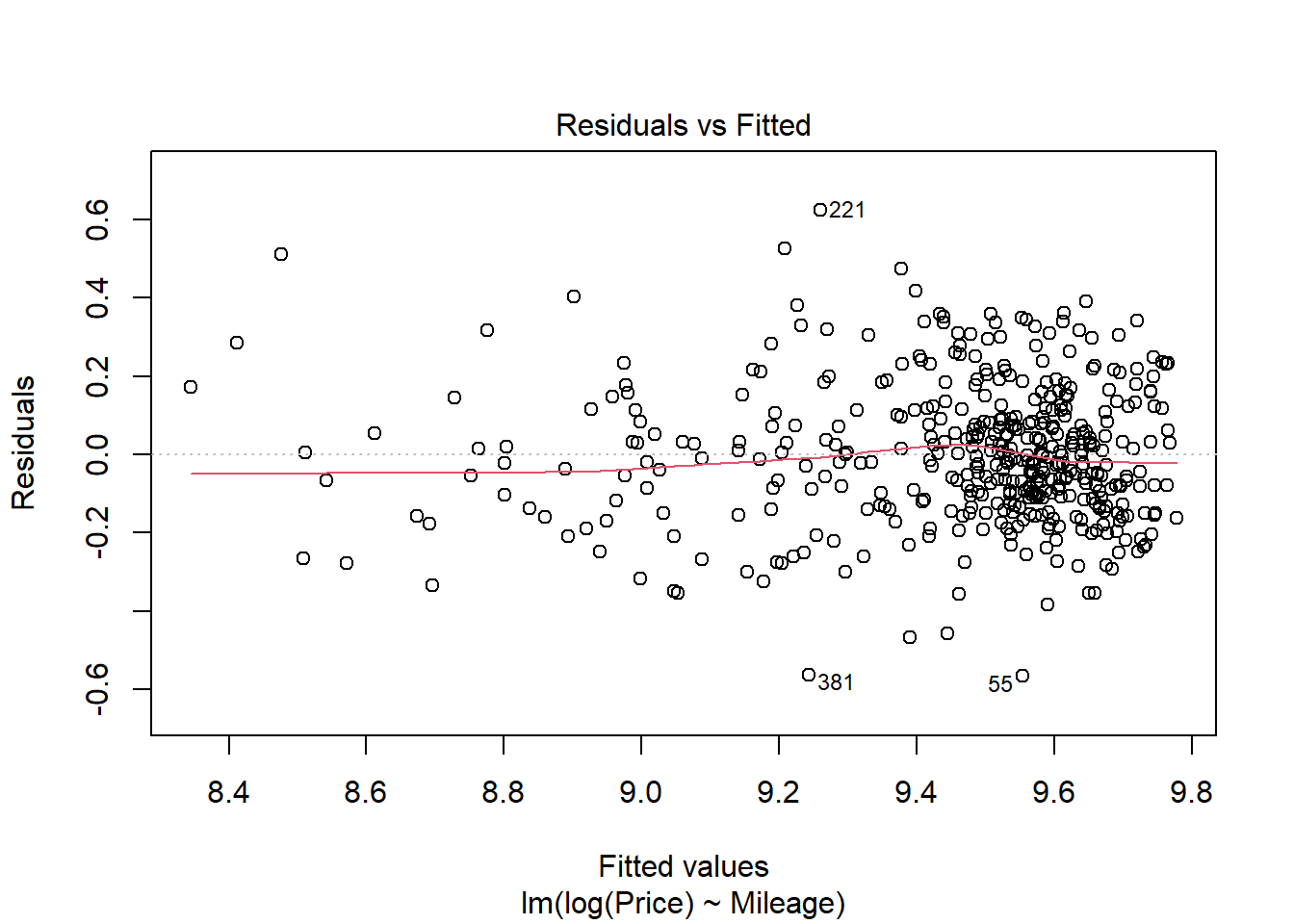
modq14.2 = lm(log(Price)~Mileage, data=MyCars)

plot(log(Price)~Mileage, data=MyCars)

abline(modq14.2)



plot(modq14.2, 1:2)

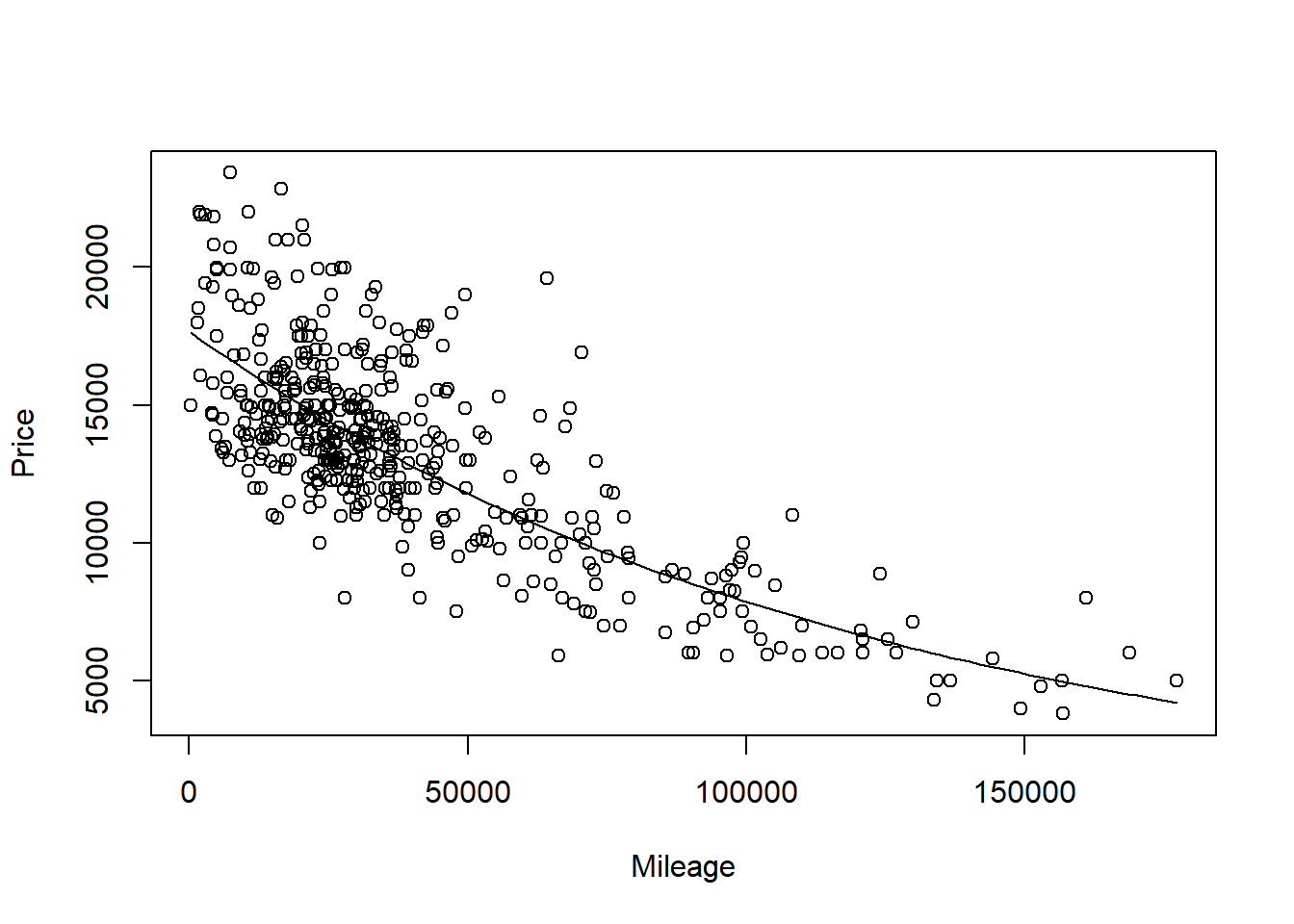


plot(Price~Mileage, data=MyCars)

B0\_q15 = summary(modq14.2)$coef[1,1]

B1\_q15 = summary(modq14.2)$coef[2,1]

curve(exp(B0\_q15 + B1\_q15\*x), add=TRUE)



1. How do the models, using either *Age* or *Mileage* as the predictor compare? Does one of the models seem “better” or do they seem similar in their ability to predict *Price*? Explain.

**1 points** - Any reasonable comparison is fine for full credit. Students may comment on things such as the p-value of signifcance tests, Rsq, etc…

**MODEL #3: Again use Age as a predictor for Price, but now for new data**

1. Select a new sample from the UsedCar dataset using the same *Model* car that was used in the previous sections, but now from cars for sale in North Carolina. You can mimic the code used above to select this new sample.

**1 points** - THey can set this up differently, but should have the same model car, a different state, and a new variable *Age*.

StateOfMyChoice2 = "NC"

ModelOfMyChoice = "Civic"

# Takes a subset of your model car from your state

MyCars2 = subset(UsedCars, Model==ModelOfMyChoice & State==StateOfMyChoice2)

# Add a new variable for the age of the cars.

MyCars2$Age = 2017 - MyCars2$Year

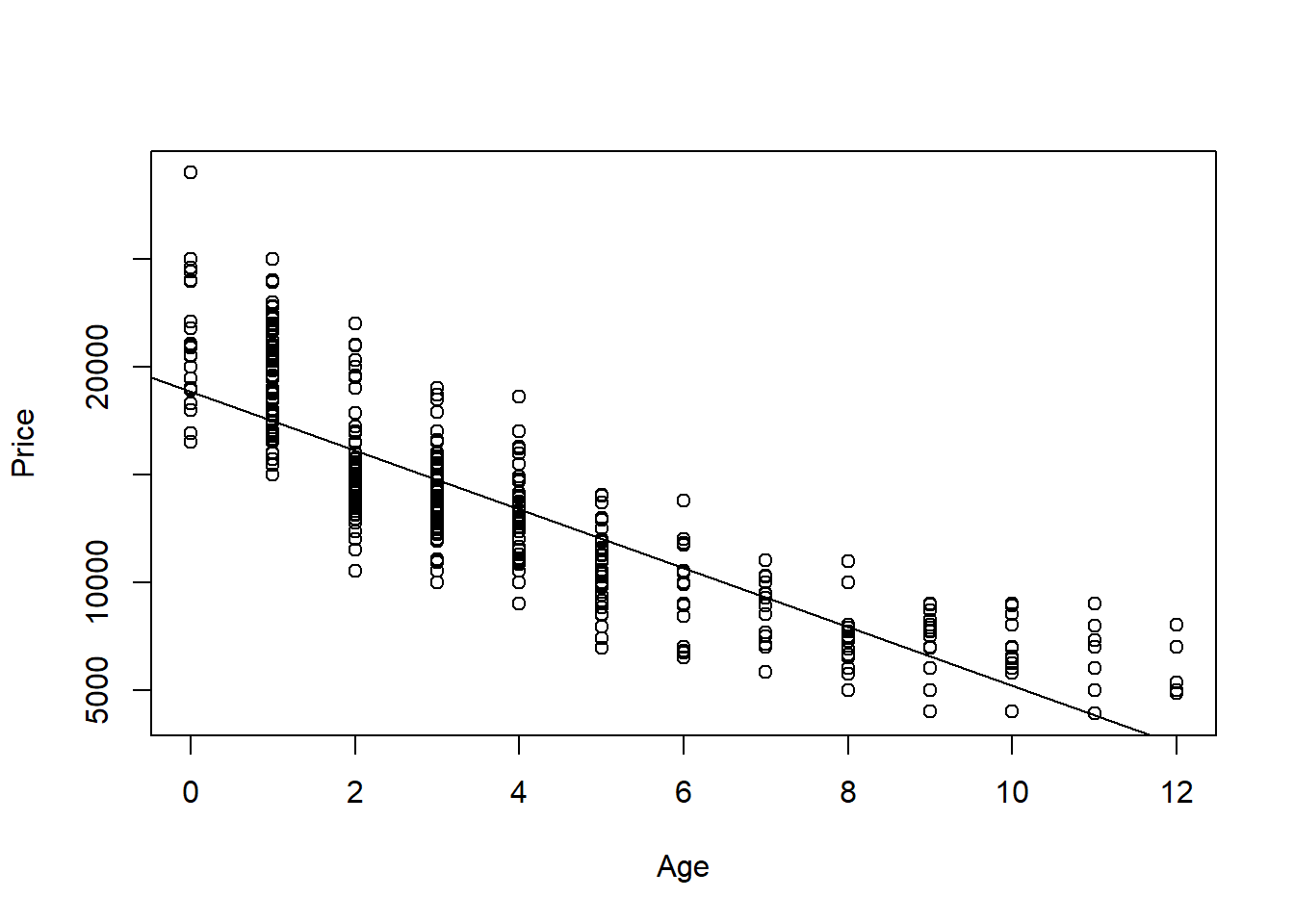
1. Calculate the least squares regression line that best fits your new data and produce a scatterplot of the relationship with the regression line on it.

**0.5 point** code for model  
**0.5 point** code for plot  
**0.5 point** abline

modq17 = lm(Price~Age, data=MyCars2)

plot(Price~Age, data=MyCars2)

abline(modq17)



1. How does the relationship between *Price* and *Age* for this new data compare to the regression model constructed in the first section? Does it appear that the relationship between *Age* and *Price* for your *Model* of car is similar or different for the data from your two states? Explain.

**1 points** - Any reasonable comparison is fine for full credit. They may compare scatterplots, summary intercepts and slope, etc. My data for NY may not show quite the same curvature, as for NC cars, but the overall trend of a curve better fitting the data than a line holds the same.

1. Again suppose that you are interested in purchasing a car of this model that is four years old (in 2017) from North Carolina. How useful do you think that your model will be? What are some possible cons of using this model?

**0.5 points** - Any reasonable discussion is fine for full credit.

In an earlier question I predicted with 90% confidence I predict that the price of a 4 year old Civic sold in NY is between $9303.123 and 15516.37. This gives me some idea of how much a car will cost, but is still a wide interval. For students that chose models with fewer of that car in the data, their intervals will be much wider. It would be more useful to have a narrower interval, that would require more data, or possibly a multiple regression model to explain more of the variability in price.