Linear Regression Mini-Project

Missy Lee

September 27, 2016

```
knitr::opts_chunk$set(echo = TRUE)
```

This is my work for the Linear Regression Mini-Project.

Instructions will be interwoven with code and results and the data used here can be found in the dataSets folder.

The only original code is in the Exercise section at the end.

Set working directory

set the working directory

```
getwd()

## [1] "C:/Users/mlee/Documents/GitHub/DataWranglingExercise1"
```

Load the states data

read the states data

```
states.data <- readRDS("dataSets/states.rds")
#get labels
states.info <- data.frame(attributes(states.data)[c("names", "var.labels")])
#look at last few labels
tail(states.info, 8)
```

```
##      names                        var.labels
## 14    csat        Mean composite SAT score
## 15    vsat           Mean verbal SAT score
## 16    msat             Mean math SAT score
## 17 percent       % HS graduates taking SAT
## 18 expense Per pupil expenditures prim&sec
## 19  income Median household income, $1,000
## 20    high             % adults HS diploma
## 21 college          % adults college degree
```

Linear regression

Examine the data before fitting models

Start by examining the data to check for problems.

summary of expense and csat columns, all rows

```
sts.ex.sat <- subset(states.data, select = c("expense", "csat"))
summary(sts.ex.sat)
```

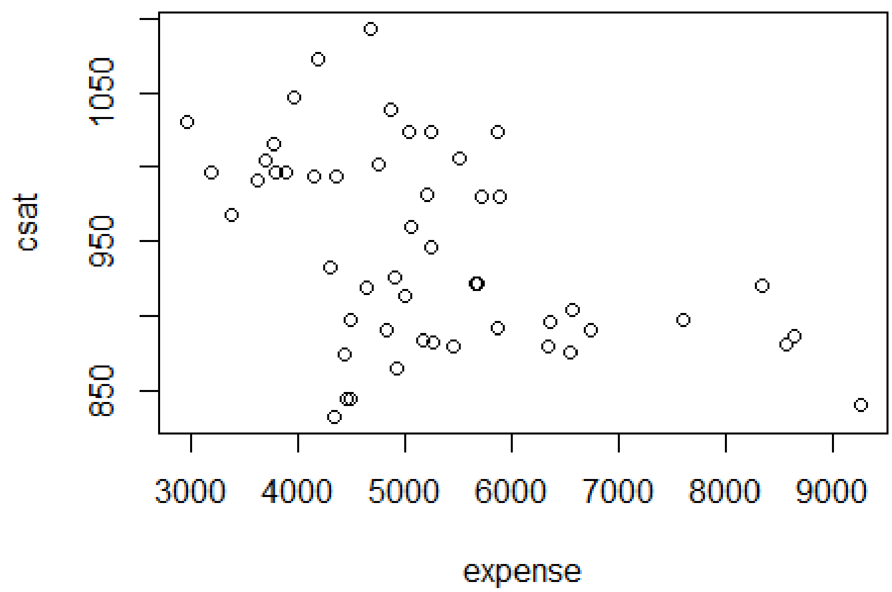
```
##     expense          csat       
##  Min.   :2960   Min.   : 832.0  
##  1st Qu.:4352   1st Qu.: 888.0  
##  Median :5000   Median : 926.0  
##  Mean   :5236   Mean   : 944.1  
##  3rd Qu.:5794   3rd Qu.: 997.0  
##  Max.   :9259   Max.   :1093.0
```

```
# correlation between expense and csat
cor(sts.ex.sat)
```

```
##            expense       csat
## expense  1.0000000 -0.4662978
## csat    -0.4662978  1.0000000
```

Plot the data before fitting models

Plot the data to look for multivariate outliers, non-linear relationships etc.



Linear regression example

• Linear regression models can be fit with the `lm()' function #   • For example, we can use`lm' to predict SAT scores based on per-pupal expenditures:

Fit our regression model

```
sat.mod <- lm(csat ~ expense,data=states.data)
```

Summarize and print the results

```
summary(sat.mod)

##
## Call:
## lm(formula = csat ~ expense, data = states.data)
```

```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -131.811  -38.085    5.607   37.852  136.495
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.061e+03  3.270e+01   32.44  < 2e-16 ***
## expense     -2.228e-02  6.037e-03   -3.69 0.000563 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 59.81 on 49 degrees of freedom
## Multiple R-squared:  0.2174, Adjusted R-squared:  0.2015
## F-statistic: 13.61 on 1 and 49 DF,  p-value: 0.0005631
```

Why is the association between expense and SAT scores /negative/?

Many people find it surprising that the per-capita expenditure on students is negatively related to SAT scores. The beauty of multiple regression is that we can try to pull these apart. What would the association between expense and SAT scores be if there were no difference among the states in the percentage of students taking the SAT?

```
summary(lm(csat ~ expense + percent, data = states.data))

##
## Call:
## lm(formula = csat ~ expense + percent, data = states.data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -62.921 -24.318   1.741  15.502  75.623
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 989.807403  18.395770  53.806  < 2e-16 ***
## expense       0.008604   0.004204   2.046   0.0462 *
## percent      -2.537700   0.224912 -11.283 4.21e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 31.62 on 48 degrees of freedom
```

```
## Multiple R-squared:  0.7857, Adjusted R-squared:  0.7768
## F-statistic: 88.01 on 2 and 48 DF,  p-value: < 2.2e-16
```

The lm class and methods

OK, we fit our model. Now what?

- **Examine the model object:**

```
class(sat.mod)

## [1] "lm"

names(sat.mod)

##  [1] "coefficients"  "residuals"     "effects"       "rank"
##  [5] "fitted.values" "assign"        "qr"            "df.residual"
##  [9] "xlevels"       "call"          "terms"         "model"

methods(class = class(sat.mod))[1:9]

## [1] "add1.lm"                      "alias.lm"
## [3] "anova.lm"                     "case.names.lm"
## [5] "coerce,oldClass,S3-method" "confint.lm"
## [7] "cooks.distance.lm"            "deviance.lm"
## [9] "dfbeta.lm"
```

- **Use function methods to get more information about the fit**

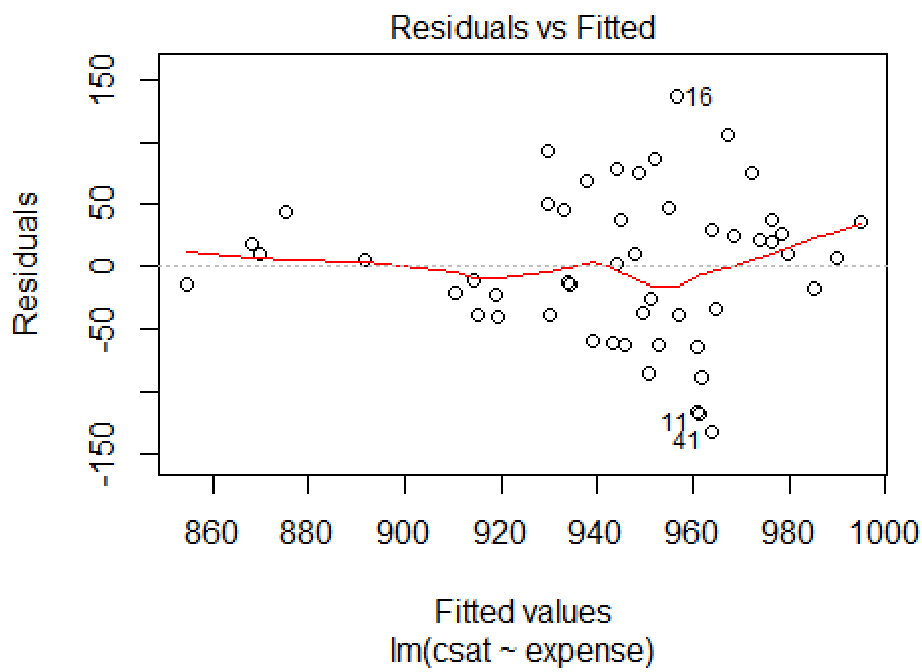
```
confint(sat.mod)

##                    2.5 %        97.5 %
## (Intercept) 995.01753164 1126.44735626
## expense      -0.03440768   -0.01014361
```

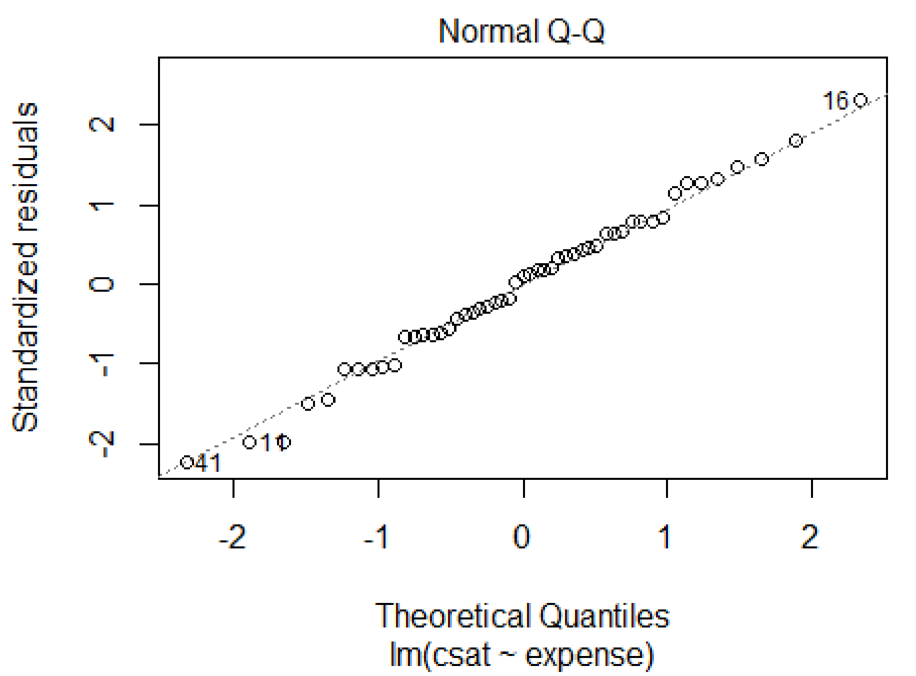
Linear Regression Assumptions

- **Ordinary least squares regression relies on several assumptions, including that the residuals are normally distributed and homoscedastic, the errors are independent and the relationships are linear.**

- **Investigate these assumptions visually by plotting your model:**

```
par(mar = c(4, 4, 2, 2), mfrow = c(1, 2))
```

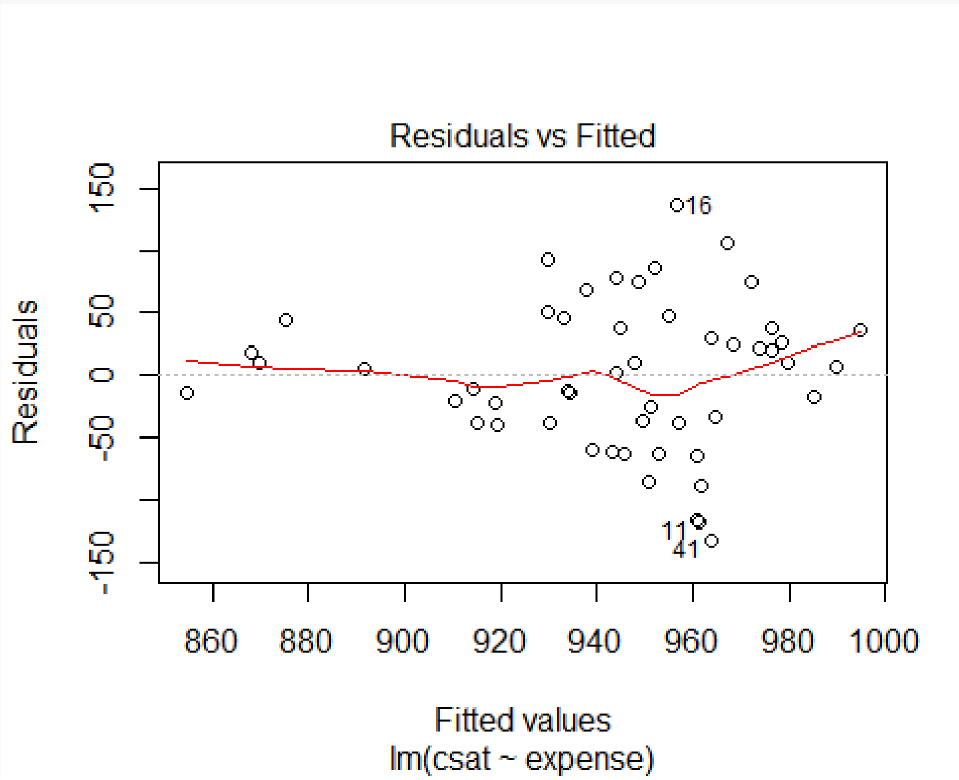


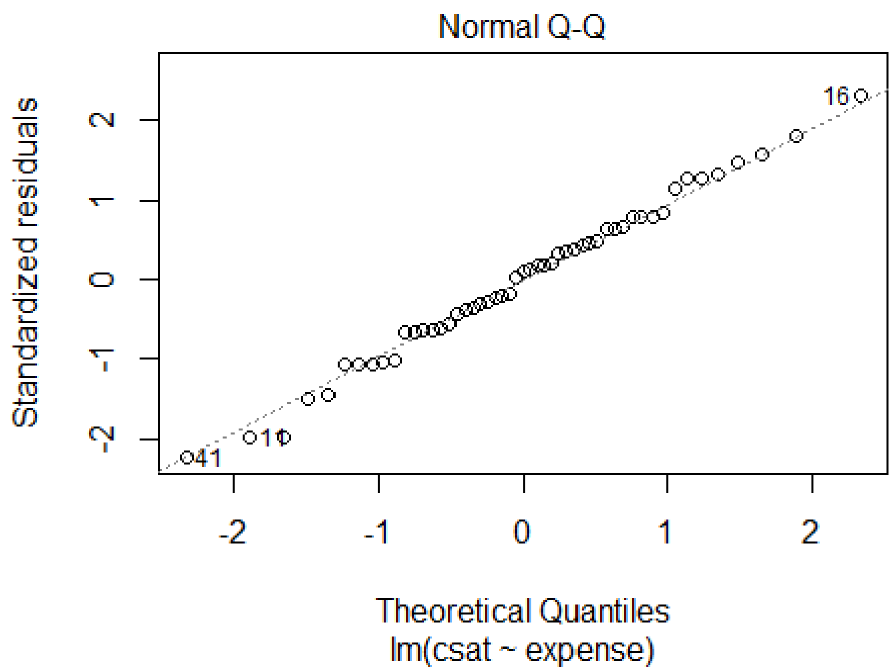


Linear Regression Assumptions

- **Ordinary least squares regression relies on several assumptions, including that the residuals are normally distributed and homoscedastic, the errors are independent and the relationships are linear.**
- **Investigate these assumptions visually by plotting your model:**

```
par(mar = c(4, 4, 2, 2), mfrow = c(1, 2), plot(sat.mod, which = c(1, 2)))
```





Comparing Models

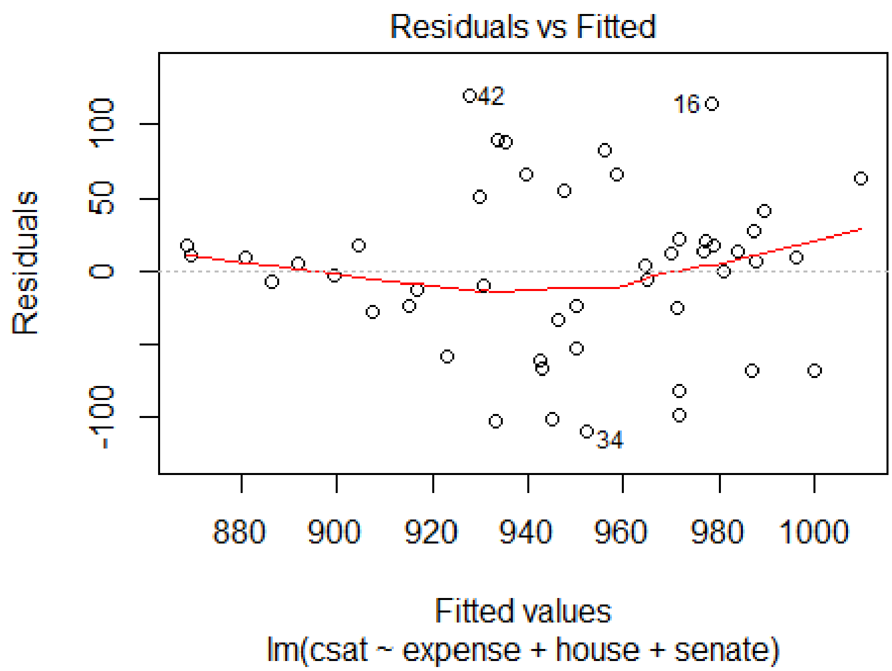
Do congressional voting patterns predict SAT scores over and above expense? Fit two models and compare them:

Fit first model, adding house and senate as predictors

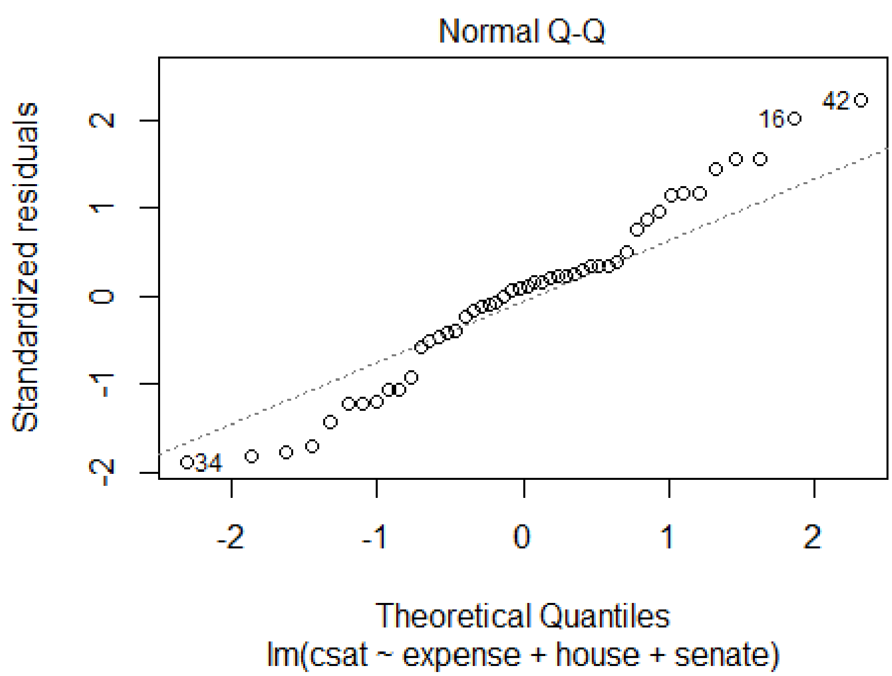
```
sat.voting.mod <-  lm(csat ~ expense + house + senate, data = na.omit(states.
data))
```

Fit another model, adding house and senate as predictors

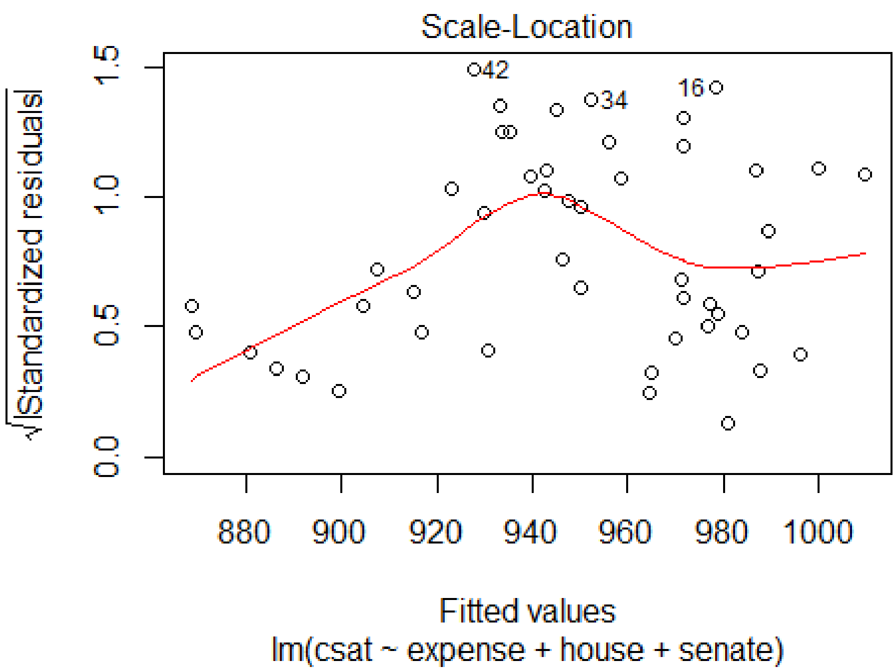
```
sat.voting.mod <-  lm(csat ~ expense + house + senate, data = na.omit(states.
data))
sat.mod <- update(sat.mod, data=na.omit(states.data))
```

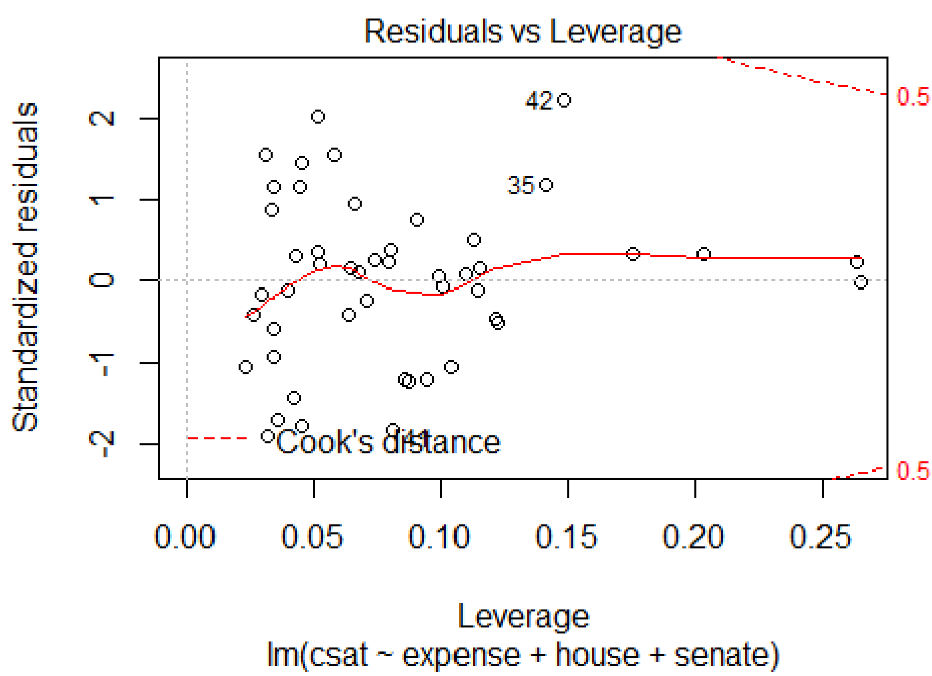


Residuals vs Fitted
Residuals
Fitted values
lm(csat ~ expense + house + senate)
42
16
34



Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(csat ~ expense + house + senate)
42
16
34


Scale-Location
√|Standardized residuals|
Fitted values
lm(csat ~ expense + house + senate)
42
34
16


Residuals vs Leverage
Standardized residuals
Leverage
lm(csat ~ expense + house + senate)
42
35
Cook's distance
0.5
0.5

compare using the anova() function

```
anova(sat.mod, sat.voting.mod)

## Analysis of Variance Table
##
## Model 1: csat ~ expense
## Model 2: csat ~ expense + house + senate
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1     46 169050
## 2     44 149284  2     19766 2.9128 0.06486 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

coef(summary(sat.voting.mod))

##                  Estimate   Std. Error    t value     Pr(>|t|)
## (Intercept) 1082.93438041 38.633812740 28.0307405 1.067795e-29
## expense       -0.01870832  0.009691494 -1.9303852 6.001998e-02
## house         -1.44243754  0.600478382 -2.4021473 2.058666e-02
## senate         0.49817861  0.513561356  0.9700469 3.373256e-01
```

Exercise: least squares regression

Use the /states.rds/ data set. Fit a model predicting energy consumed per capita (energy) from the percentage of residents living in metropolitan areas (metro). Be sure to

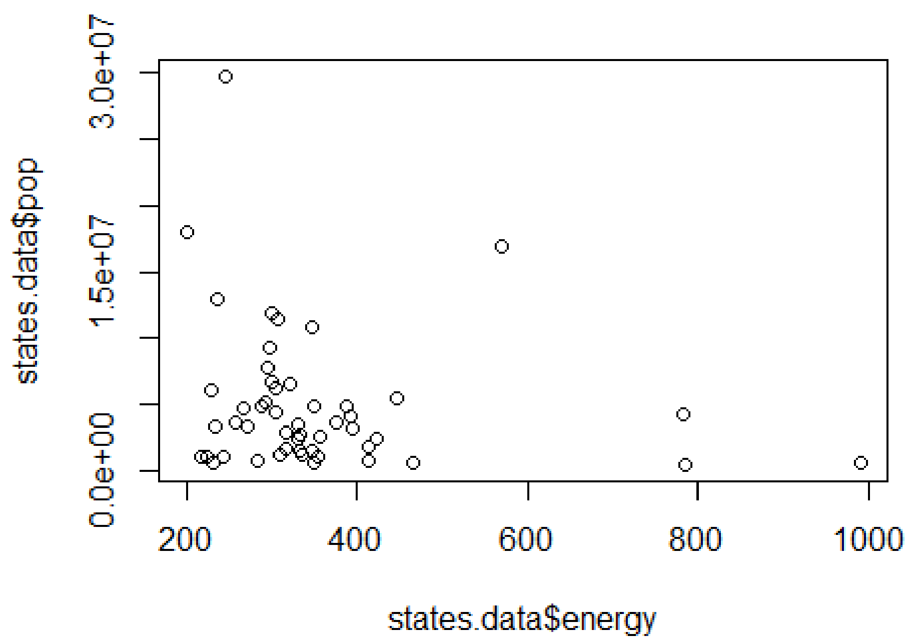
1. Examine/plot the data before fitting the model

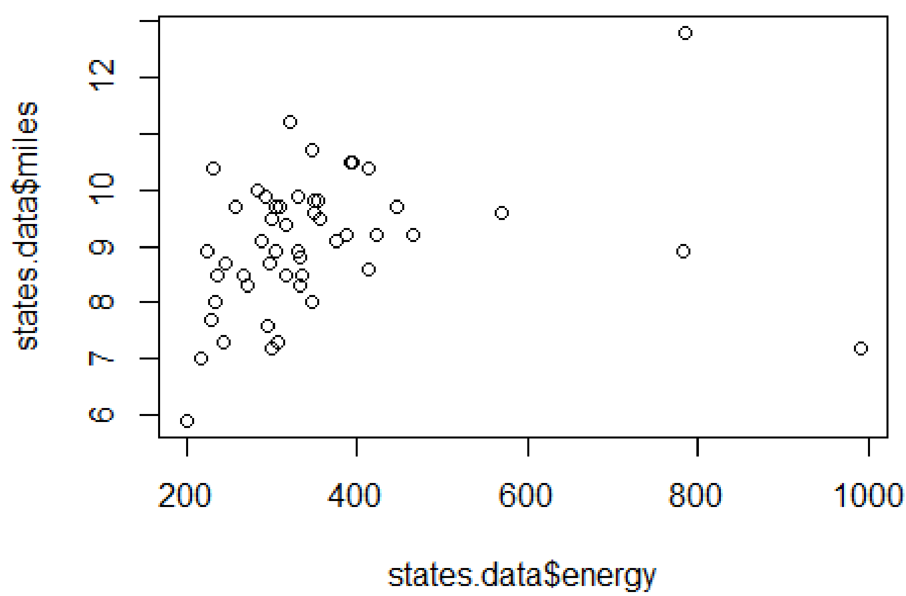
```
str(states.data)

## 'data.frame':    51 obs. of  21 variables:
##  $ state  : chr  "Alabama" "Alaska" "Arizona" "Arkansas" ...
##  $ region : Factor w/ 4 levels "West","N. East",..: 3 1 1 3 1 1 2 3 NA 3 .
..
##  $ pop    : num  4041000 550000 3665000 2351000 29760000 ...
##  $ area   : num  52423 570374 113642 52075 155973 ...
##  $ density: num  77.08 0.96 32.25 45.15 190.8 ...
##  $ metro  : num  67.4 41.1 79 40.1 95.7 ...
##  $ waste  : num  1.11 0.91 0.79 0.85 1.51 ...
##  $ energy : int  393 991 258 330 246 273 234 349 NA 237 ...
##  $ miles  : num  10.5 7.2 9.7 8.9 8.7 ...
##  $ toxic  : num  27.86 37.41 19.65 24.6 3.26 ...
##  $ green  : num  29.2 NA 18.4 26 15.6 ...
##  $ house  : int  30 0 13 25 50 36 64 69 NA 45 ...
```

```
##  $ senate : int  10 20 33 37 47 58 87 83 NA 47 ...
##  $ csat   : int  991 920 932 1005 897 959 897 892 840 882 ...
##  $ vsat   : int  476 439 442 482 415 453 429 428 405 416 ...
##  $ msat   : int  515 481 490 523 482 506 468 464 435 466 ...
##  $ percent: int  8 41 26 6 47 29 81 61 71 48 ...
##  $ expense: int  3627 8330 4309 3700 4491 5064 7602 5865 9259 5276 ...
##  $ income : num  27.5 48.3 32.1 24.6 41.7 ...
##  $ high   : num  66.9 86.6 78.7 66.3 76.2 ...
##  $ college: num  15.7 23 20.3 13.3 23.4 ...
##  - attr(*, "datalabel")= chr "U.S. states data 1990-91"
##  - attr(*, "time.stamp")= chr " 6 Apr 2012 08:40"
##  - attr(*, "formats")= chr  "%20s" "%9.0g" "%9.0g" "%9.0g" ...
##  - attr(*, "types")= int  20 251 254 254 254 254 254 252 254 254 ...
##  - attr(*, "val.labels")= chr  "" "region" "" "" ...
##  - attr(*, "var.labels")= chr  "State" "Geographical region" "1990 populat
ion" "Land area, square miles" ...
##  - attr(*, "expansion.fields")=List of 4
##   ..$ : chr  "_dta" "_lang_c" "default"
##   ..$ : chr  "_dta" "_lang_list" "default"
##   ..$ : chr  "_dta" "__xi__Vars__To__Drop__" "_Iregion_2 _Iregion_3 _Iregi
on_4 _IregXperce_2 _IregXperce_3 _IregXperce_4"
##   ..$ : chr  "_dta" "__xi__Vars__Prefix__" "_I _I _I _I _I _I"
##  - attr(*, "version")= int 12
##  - attr(*, "label.table")=List of 1
##   ..$ region: Named int  1 2 3 4
##   .. ..- attr(*, "names")= chr  "West" "N. East" "South" "Midwest"
```

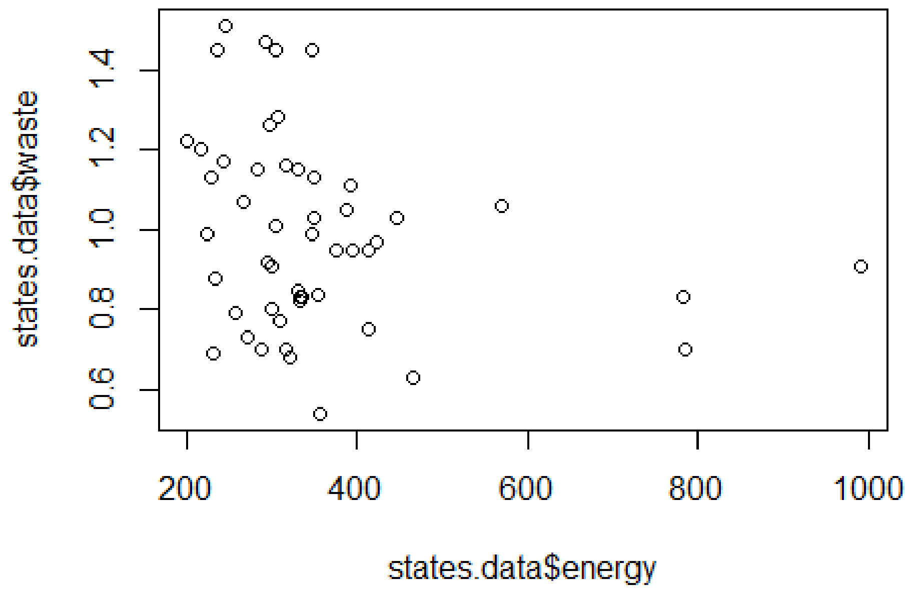
Plot the data: Energy and popuation

Energy and miles

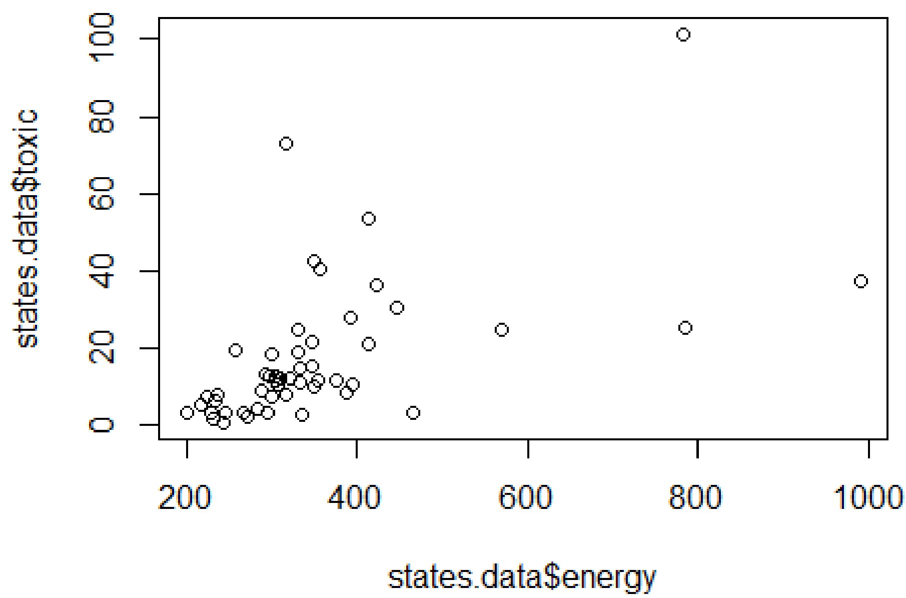




Energy and waste



Energy and toxic waste



Fit the model

```
energy.mod <- lm(states.data$energy ~ states.data$metro + states.data$density
+ states.data$miles)
summary(energy.mod)

## Call:
## lm(formula = states.data$energy ~ states.data$metro + states.data$density
+      states.data$miles)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -197.33  -69.60  -33.74   15.00  588.64
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)          442.9017   228.9018   1.935   0.0592 .
## states.data$metro     -1.4462     1.2271  -1.179   0.2446
## states.data$density   -0.1183     0.1113  -1.062   0.2937
## states.data$miles      2.6409    20.6078   0.128   0.8986
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 141.2 on 46 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:   0.14,  Adjusted R-squared:  0.08388
## F-statistic: 2.495 on 3 and 46 DF,  p-value: 0.07158
```

A revised model

```
lm(formula = states.data$energy ~ states.data$metro + states.data$density +
    states.data$miles)

##
## Call:
## lm(formula = states.data$energy ~ states.data$metro + states.data$density
+
##     states.data$miles)
##
## Coefficients:
##         (Intercept)    states.data$metro  states.data$density
##            442.9017              -1.4462              -0.1183
##   states.data$miles
##              2.6409
```

Comment:

Enery.mod with extra predictors has a lower adjusted R-squared than energy.mod2 with only one predictor.

Interactions and factors

Modeling interactions

Interactions allow us assess the extent to which the association between one predictor and the outcome depends on a second predictor. For example: Does the association between expense and SAT scores depend on the median income in the state?

Add the interaction to the model

```
sat.expense.by.percent <- lm(csat ~ expense*income, data=states.data)
```

Show the results

```
coef(summary(sat.expense.by.percent))

##                      Estimate   Std. Error   t value     Pr(>|t|)
## (Intercept)     1.380364e+03 1.720863e+02  8.021351 2.367069e-10
## expense        -6.384067e-02 3.270087e-02 -1.952262 5.687837e-02
## income         -1.049785e+01 4.991463e+00 -2.103161 4.083253e-02
## expense:income  1.384647e-03 8.635529e-04  1.603431 1.155395e-01
```

Regression with categorical predictors

Let's try to predict SAT scores from region, a categorical variable. Note that you must make sure R does not think your categorical variable is numeric.

Make sure R knows region is categorical

```
str(states.data$region)

##  Factor w/ 4 levels "West","N. East",..: 3 1 1 3 1 1 2 3 NA 3 ...
```

Add region to the model

```
sat.region <- lm(csat ~ region, data=states.data)
```

Show the results

```r
coef(summary(sat.region))
```

```
##                  Estimate Std. Error    t value     Pr(>|t|)
## (Intercept)   946.30769   14.79582 63.9577807 1.352577e-46
## regionN. East -56.75214   23.13285 -2.4533141 1.800383e-02
## regionSouth   -16.30769   19.91948 -0.8186806 4.171898e-01
## regionMidwest  63.77564   21.35592  2.9863209 4.514152e-03
```

```r
anova(sat.region)
```

```
## Analysis of Variance Table
##
## Response: csat
##           Df Sum Sq Mean Sq F value    Pr(>F)
## region     3  82049 27349.8  9.6102 4.859e-05 ***
## Residuals 46 130912  2845.9
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Setting factor reference groups and contrasts

In the previous example we use the default contrasts for region. The default in R is treatment contrasts, with the first level as the reference. We can change the reference group or use another coding scheme using the `C' function.

Print default contrasts

```r
contrasts(states.data$region)
```

```
##         N. East South Midwest
## West          0     0       0
## N. East       1     0       0
## South         0     1       0
## Midwest       0     0       1
```

Change the reference group

```r
coef(summary(lm(csat ~ C(region, base=4), data=states.data)))
```

```
##                         Estimate Std. Error   t value     Pr(>|t|)
## (Intercept)           1010.08333   15.39998 65.589930 4.296307e-47
## C(region, base = 4)1   -63.77564   21.35592 -2.986321 4.514152e-03
## C(region, base = 4)2 -120.52778   23.52385 -5.123641 5.798399e-06
## C(region, base = 4)3   -80.08333   20.37225 -3.931000 2.826007e-04
```

Change the coding scheme

```
coef(summary(lm(csat ~ C(region, contr.helmert), data=states.data)))

##                               Estimate Std. Error     t value     Pr(>|t|)
## (Intercept)                 943.986645   7.706155 122.4977451 1.689670e-59
## C(region, contr.helmert)1 -28.376068  11.566423  -2.4533141 1.800383e-02
## C(region, contr.helmert)2   4.022792   5.884552   0.6836191 4.976450e-01
## C(region, contr.helmert)3  22.032229   4.446777   4.9546509 1.023364e-05
```

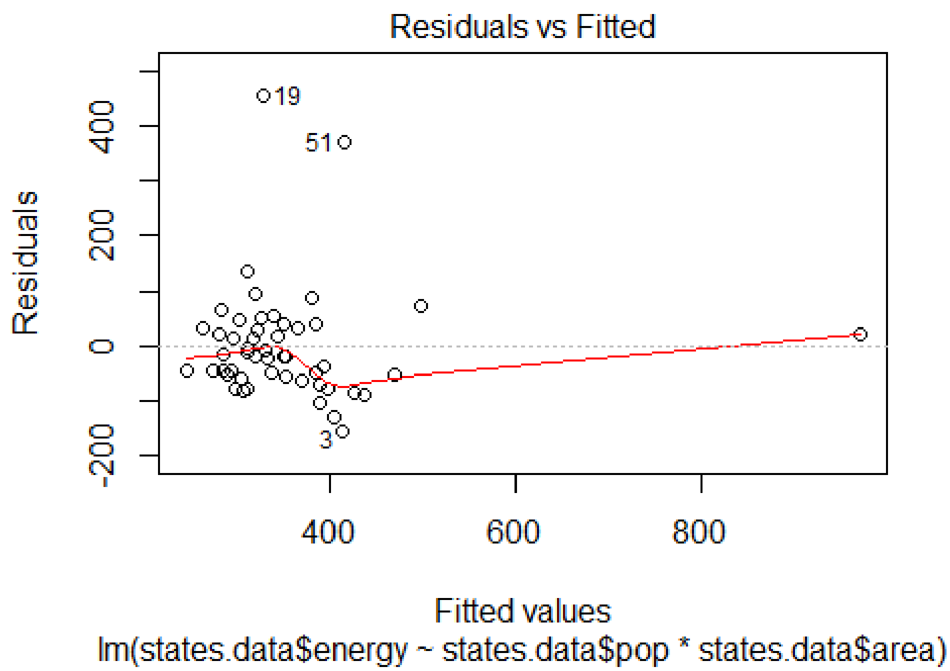
Original coding begins here--

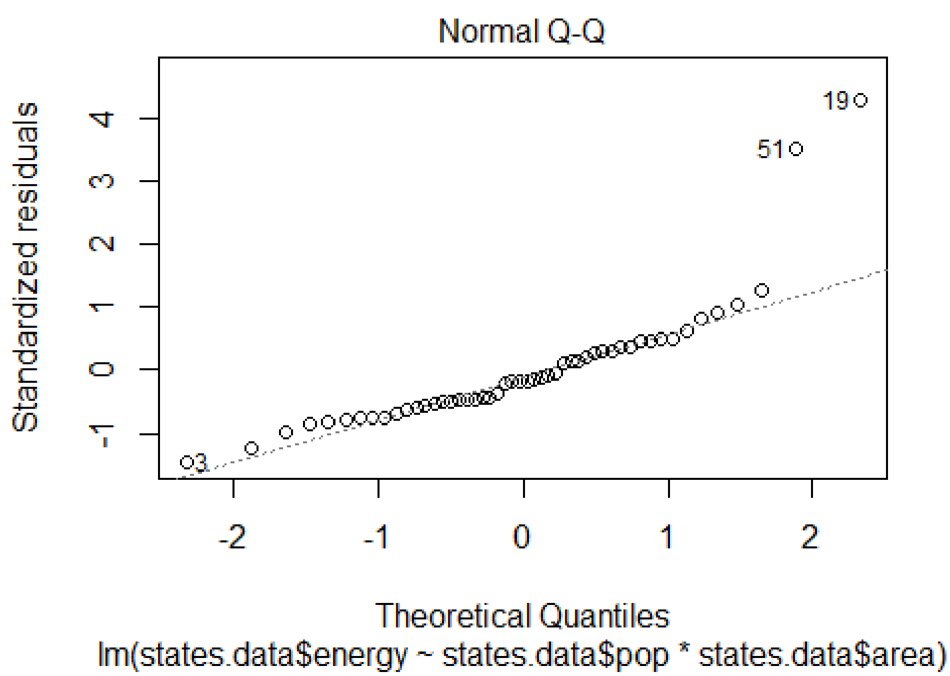
Exercise: interactions and factors

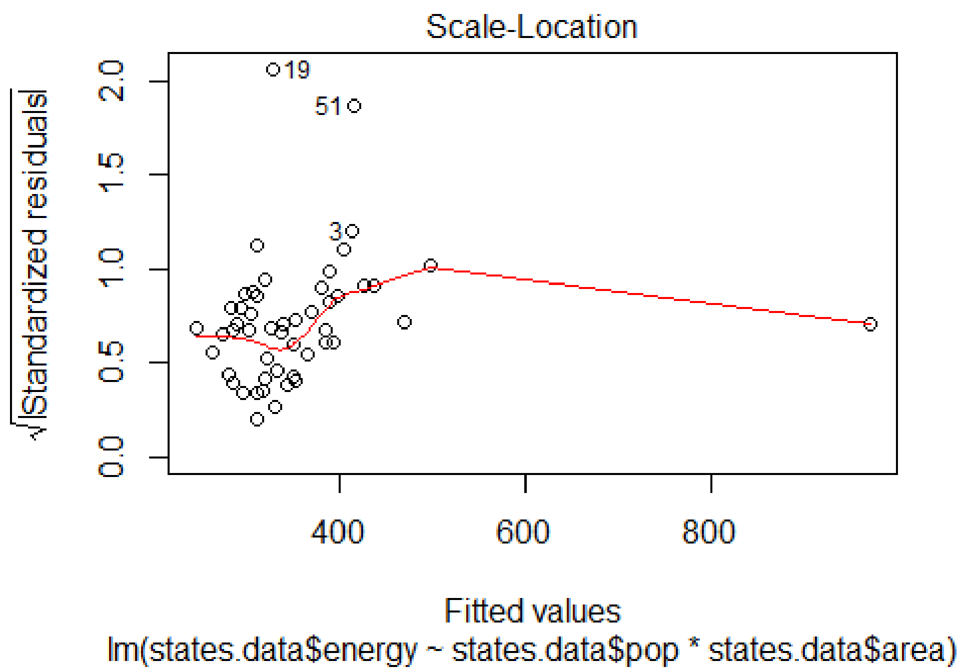
Use the states data set.

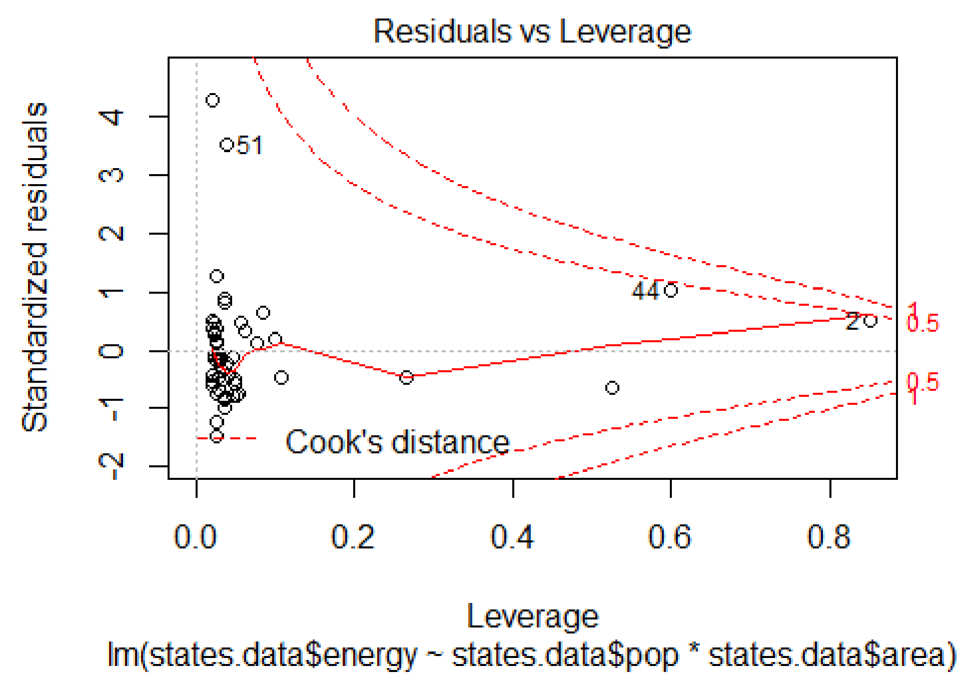
1. Add on to the regression equation that you created in exercise 1 by generating an interaction term and testing the interaction.

```
energy.mod <- lm(states.data$energy ~ states.data$pop * states.data$area, dat
a=states.data)


coef(summary(energy.mod))

##                                       Estimate   Std. Error     t value
## (Intercept)                        3.029774e+02 2.838904e+01 10.67233658
## states.data$pop                   -6.288445e-06 5.187475e-06 -1.21223613
## states.data$area                   1.176763e-03 2.079198e-04  5.65969696
## states.data$pop:states.data$area -1.392399e-12 3.438859e-11 -0.04049015
##                                      Pr(>|t|)
## (Intercept)                       4.934838e-14
## states.data$pop                   2.316117e-01
## states.data$area                  9.343431e-07
## states.data$pop:states.data$area 9.678776e-01
```


Residuals vs Fitted
Residuals
Fitted values
lm(states.data$energy ~ states.data$pop * states.data$area)
-200
0
200
400
400
600
800
19
51
3


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(states.data$energy ~ states.data$pop * states.data$area)
-1
0
1
2
3
4
-2
-1
0
1
2
19
51
3


Scale-Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
19
51
3
400
600
800
Fitted values
lm(states.data$energy ~ states.data$pop * states.data$area)


Residuals vs Leverage
Standardized residuals
-2
-1
0
1
2
3
4
51
44
2
1
0.5
0.5
1
Cook's distance
0.0
0.2
0.4
0.6
0.8
Leverage
lm(states.data$energy ~ states.data$pop * states.data$area)

```r
coef(summary(energy.mod))
```

```
##                                        Estimate   Std. Error     t value
## (Intercept)                        3.029774e+02 2.838904e+01 10.67233658
## states.data$pop                   -6.288445e-06 5.187475e-06 -1.21223613
## states.data$area                   1.176763e-03 2.079198e-04  5.65969696
## states.data$pop:states.data$area -1.392399e-12 3.438859e-11 -0.04049015
##                                       Pr(>|t|)
## (Intercept)                       4.934838e-14
## states.data$pop                   2.316117e-01
## states.data$area                  9.343431e-07
## states.data$pop:states.data$area 9.678776e-01
```

2. Try adding region to the model. Are there significant differences across the four regions?

```r
energy.mod3 <- lm(states.data$energy ~ states.data$metro + states.data$densit
y + states.data$miles + states.data$region)


coef(summary(energy.mod3))
```

```
##                                Estimate  Std. Error    t value   Pr(>|t|)
## (Intercept)                 636.68378162 245.4176158  2.5942872 0.01290973
## states.data$metro            -2.36303503   1.2915234 -1.8296494 0.07424112
## states.data$density           0.03849528   0.1430191  0.2691618 0.78909254
## states.data$miles            -9.31951252  21.5856905 -0.4317449 0.66808398
## states.data$regionN. East -156.04967990  82.1351593 -1.8999133 0.06416398
## states.data$regionSouth    -25.09299582  54.0558457 -0.4642050 0.64484256
## states.data$regionMidwest  -69.82374799  56.7612911 -1.2301297 0.22533620
```

Interpretation of exercise results: I would interpret this coefficient table to mean there are no significant reationships in this model. The small t values are not high enough to indicate the null hypothesis can be rejected. The standard deviations for most variables are too high. Pr(>|t|) are high enough in most cases are high enough to say the observed results are due to chance. None of the p-values are indicated by asterisks to be significant.