Linear Regression - Intro Data Science Mini-Project

Carl Larson

2/1/2018

For this project we are applying Linear Regression analysis to the "states.rds" data set as follows.

```
states.data <- readRDS("/Users/EagleFace/Documents/!linear_regression/dataSets/states.rds")
states.info <- data.frame(attributes(states.data)[c("names", "var.labels")])
head(states.info, 12)
```

```
##      names                      var.labels
## 1    state                           State
## 2   region             Geographical region
## 3      pop                 1990 population
## 4     area         Land area, square miles
## 5  density          People per square mile
## 6    metro Metropolitan area population, %
## 7    waste    Per capita solid waste, tons
## 8   energy Per capita energy consumed, Btu
## 9    miles    Per capita miles/year, 1,000
## 10   toxic Per capita toxics released, lbs
## 11   green Per capita greenhouse gas, tons
## 12   house    House '91 environ. voting, %
```

```
tail(states.info, 12)
```

```
##      names                      var.labels
## 10   toxic Per capita toxics released, lbs
## 11   green Per capita greenhouse gas, tons
## 12   house    House '91 environ. voting, %
## 13  senate   Senate '91 environ. voting, %
## 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
```

```
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
```

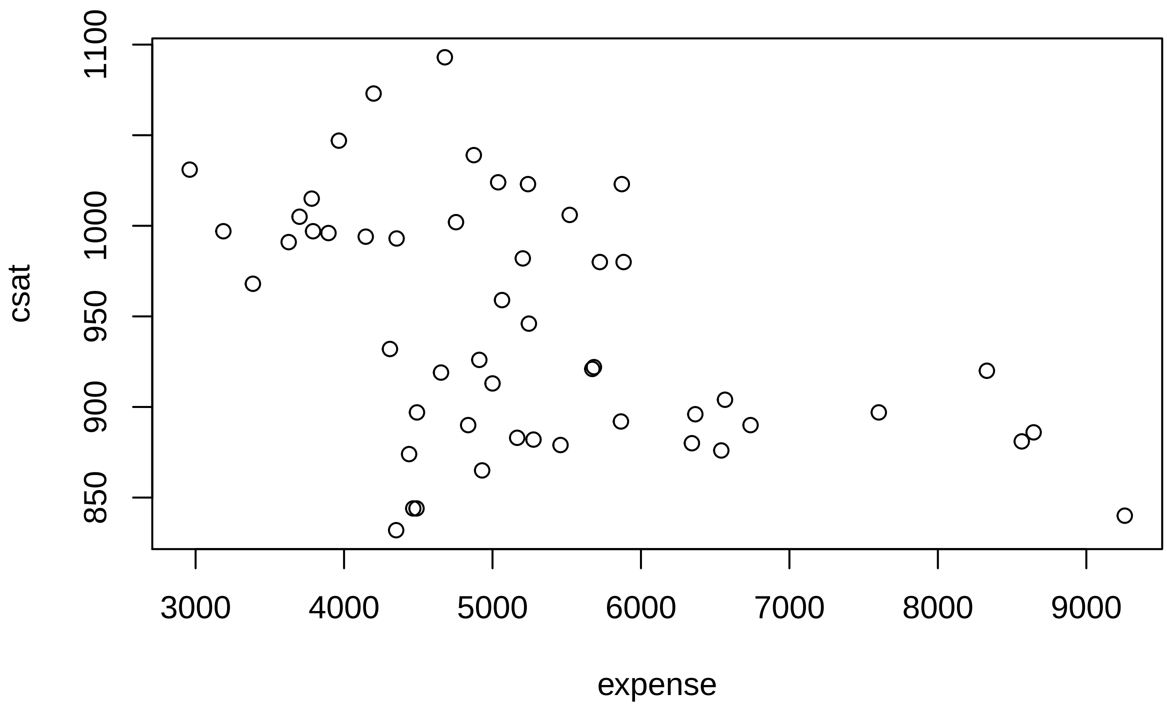
```
cor(sts.ex.sat)
```

```
##             expense       csat
```

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

This is registering some interesting data.

```
plot(sts.ex.sat)
```



This looks like a very loose negative correlation, possibly something roughly to the tune of y = (-0.4x^-0.4)+1400. This does strike me as a negative square root type shape of line-of-best-fit, but still it's hard to correlate anything to this dataset as it has a low R-squared value no matter how you draw a line through this set.

```
#Fitting the regression model

sat.mod <- lm(csat ~ expense, #apparent regression formula
              data=states.data) #data set of focus

#Opening up a view of 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
```

It seems that the more people spend on their SAT prep, actually the worse off they do. Could this be an indictment of the SAT prep industry?

```
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
```

```
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"
```

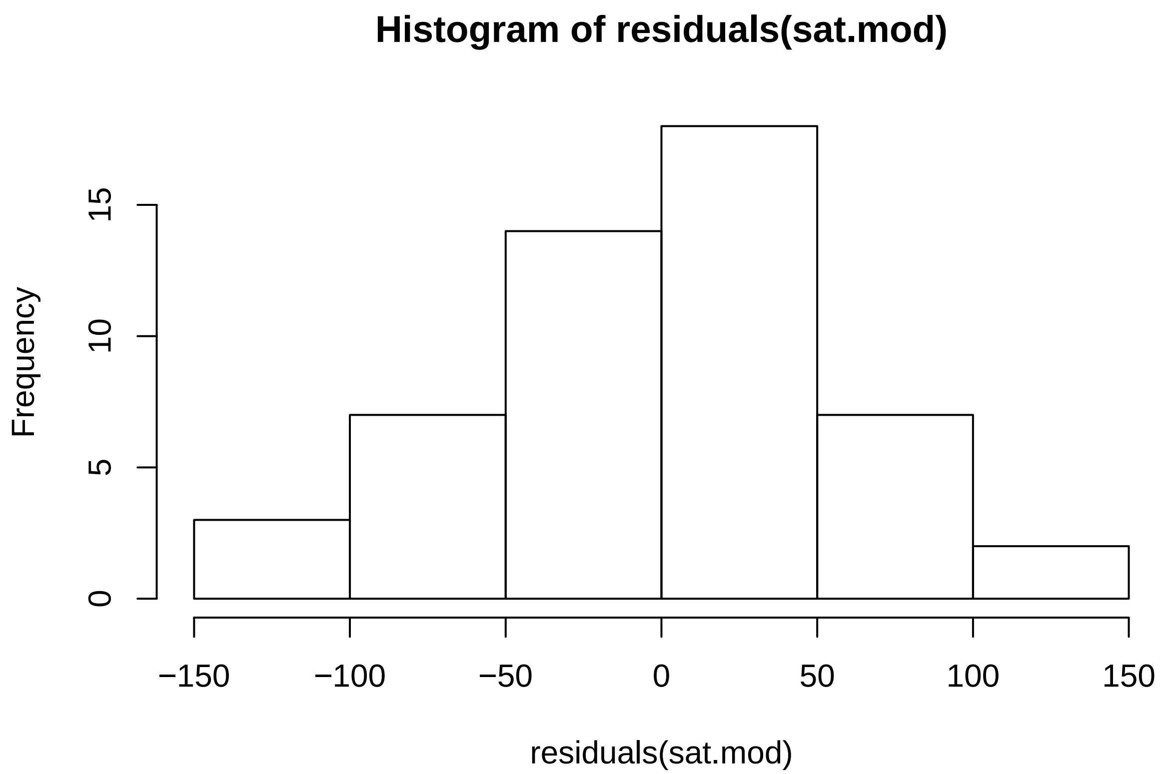
```
methods(class = class(sat.mod))[1:9]
```

```
## [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"
```

```
confint(sat.mod)
```

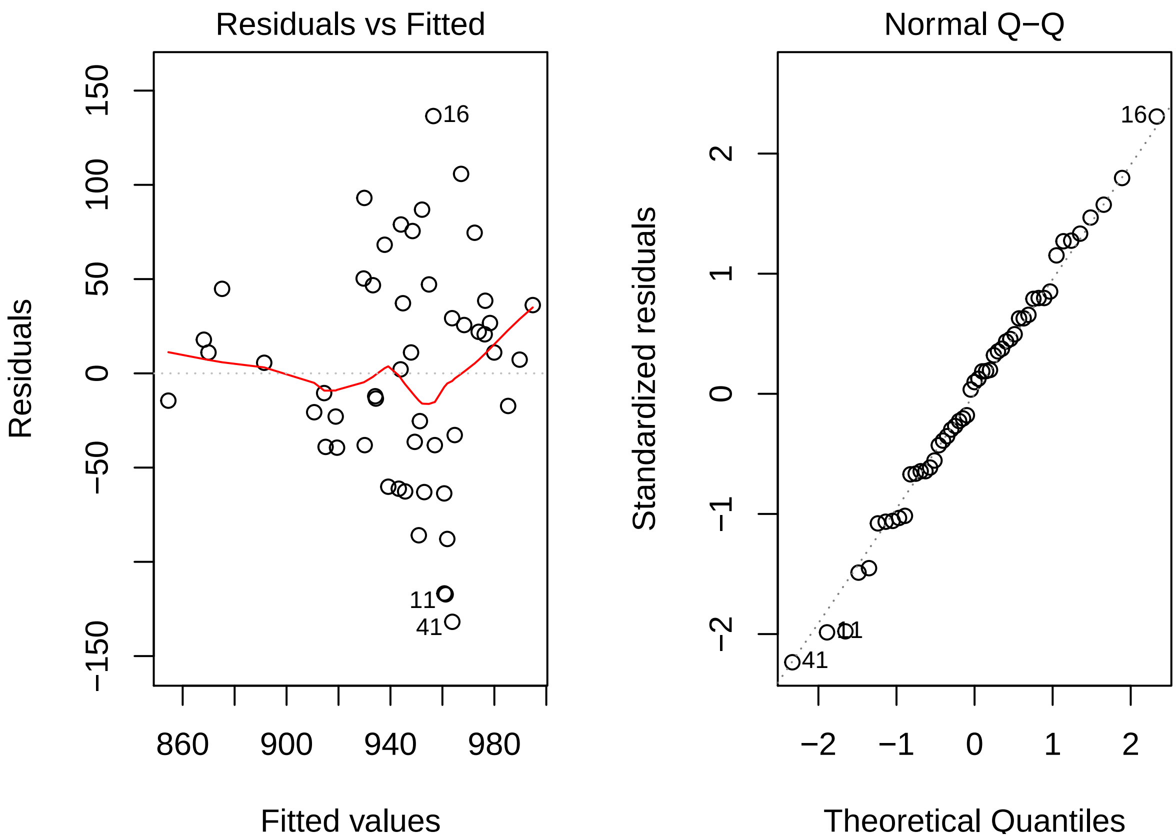
```
##                   2.5 %        97.5 %
## (Intercept) 995.01753164 1126.44735626
## expense      -0.03440768   -0.01014361
```

```r
hist(residuals(sat.mod))
```



Since ordinary least squares regression requires a number of assumptions we can apply to the following visualizations.

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




```
#Next we are comparing models, asking if congressional voting pattern could be
#a better predictor than expense, and expense wasn't very good so it's likely.

#Below fits a new 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))
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
```

These also look like pretty rough, low correlations.

We are next asked to plot our own model using the percentage of residents living in metropolitan areas to predict energy consumed per capita.

```
nrg.ex.dzt <- subset(states.data, select = c("density", "energy"))

summary(nrg.ex.dzt)
```

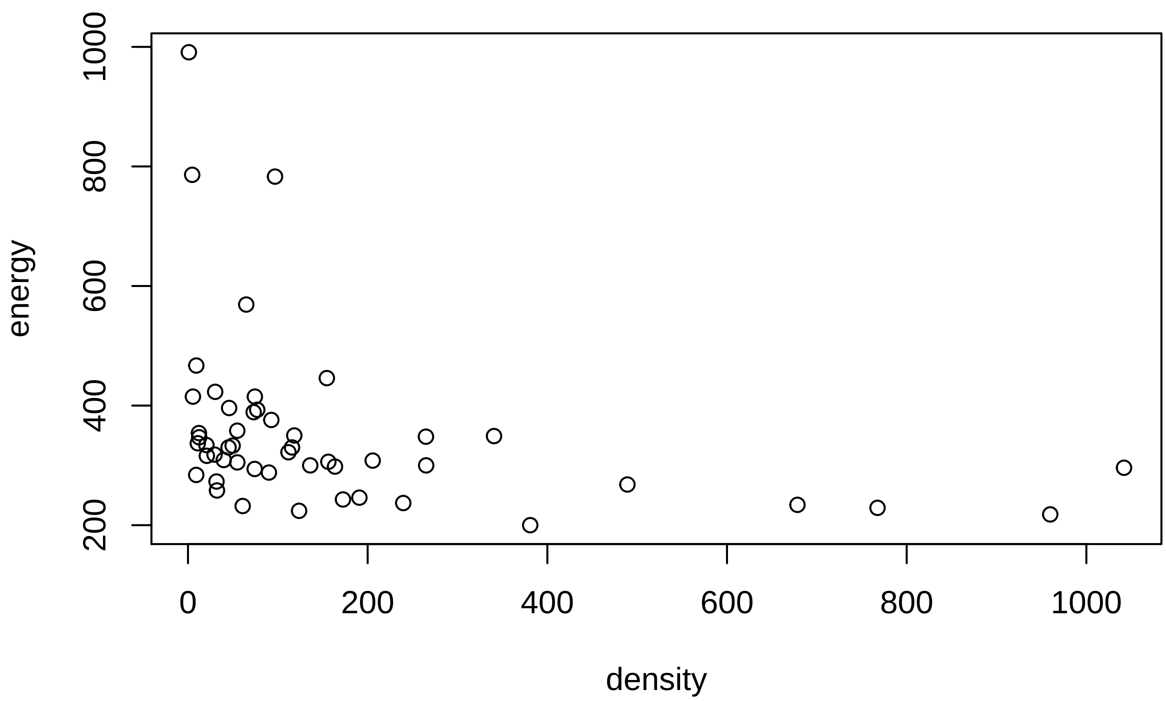
```
##     density            energy
##  Min.   :   0.96   Min.   :200.0
##  1st Qu.:  31.88   1st Qu.:285.0
##  Median :  75.76   Median :320.0
##  Mean   : 166.04   Mean   :354.5
##  3rd Qu.: 170.29   3rd Qu.:371.5
##  Max.   :1041.92   Max.   :991.0
##  NA's   :1         NA's   :1
```

```
cor(nrg.ex.dzt)
```

```
##         density energy
## density       1     NA
## energy       NA      1
```

After checking these results, we can try plotting this to see what it looks like on the same graph.

```
plot(nrg.ex.dzt)
```



This actually looks fairly well-correlated. The R-value for a "y=1/x" type algorithm here would fit fairly well and does make sense, as most people are in the middle, and the edges seem roughly normally distributed.

```
nrg.mod <- lm(energy ~ density,
              data=states.data)

summary(nrg.mod)
```

```
##
```

```
## Call:
## lm(formula = energy ~ density, data = states.data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -144.17  -70.73  -36.60   19.31  602.49
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 388.70969   24.45374  15.896   <2e-16 ***
## density      -0.20603    0.08553  -2.409   0.0199 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 140.8 on 48 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.1079, Adjusted R-squared:  0.08927
## F-statistic: 5.803 on 1 and 48 DF,  p-value: 0.01988
```

It seems as though the R-squared value is far too low for this to be a viable model for the correlation. This definitely isn't a linear relationship, but there is a correlation between these variables even though the above algorithm isn't seeing it.

The problem set asks us to add more variables into the equation to see if we can make this more accurate.

After looking back above, the best three other variables to grab would be

- miles (the number of per capita miles per year in thousands)
- green (per capita greenhouse emissions in tons)
- income

These should be great indicators for the output variable of energy used.

```
best.guess <- subset(states.data, select = c("energy", "density", "miles", "green", "income"))
summary(best.guess)
```

```
##      energy         density             miles             green
##  Min.   :200.0   Min.   :   0.96   Min.   : 5.900   Min.   : 11.76
##  1st Qu.:285.0   1st Qu.:  31.88   1st Qu.: 8.500   1st Qu.: 16.98
##  Median :320.0   Median :  75.76   Median : 9.100   Median : 21.38
##  Mean   :354.5   Mean   : 166.04   Mean   : 9.046   Mean   : 25.11
##  3rd Qu.:371.5   3rd Qu.: 170.29   3rd Qu.: 9.700   3rd Qu.: 26.34
##  Max.   :991.0   Max.   :1041.92   Max.   :12.800   Max.   :114.40
##  NA's   :1       NA's   :1         NA's   :1        NA's   :3
##      income
##  Min.   :23.46
##  1st Qu.:29.88
##  Median :33.45
##  Mean   :33.96
##  3rd Qu.:36.92
##  Max.   :48.62
##
```

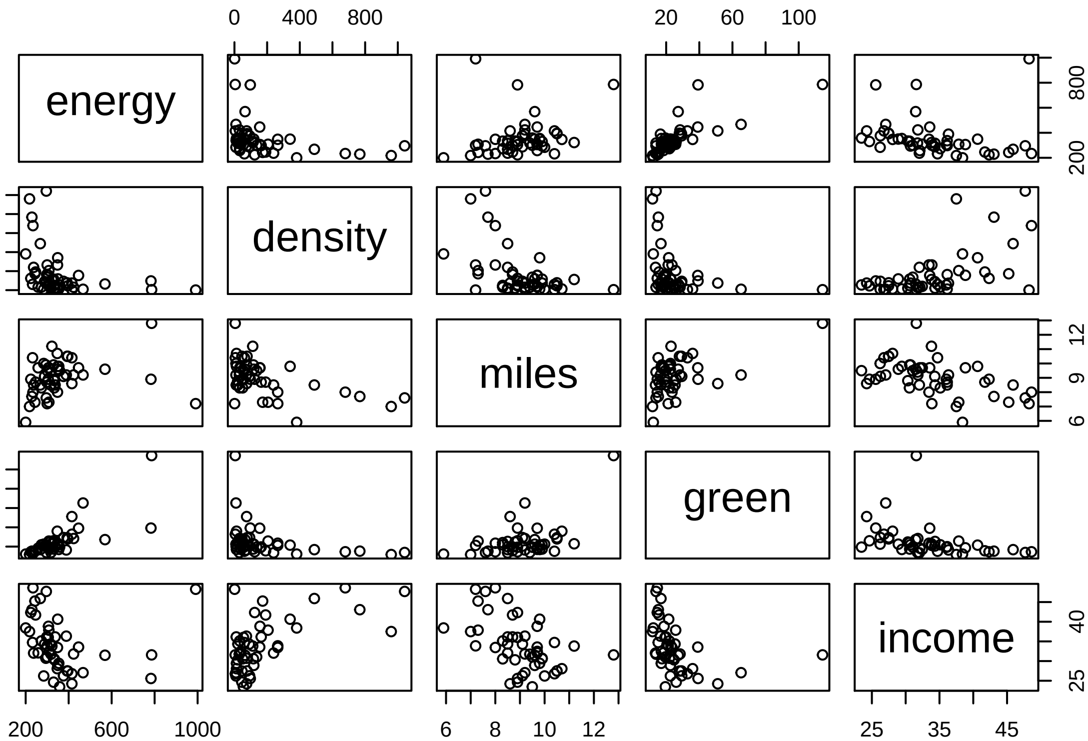
```
cor(best.guess)
```

```
##         energy density miles green income
## energy       1      NA    NA    NA     NA
## density     NA       1    NA    NA     NA
```

```
## miles        NA       NA      1     NA      NA
## green        NA       NA     NA      1      NA
## income       NA       NA     NA     NA       1
```

Given this we can try a chart.

```
plot(best.guess)
```




```
best.mod <- lm(energy ~ density + miles + green + income,
              data=states.data)

summary(best.mod)
```

```
##
## Call:
## lm(formula = energy ~ density + miles + green + income, data = states.data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -89.41 -34.13  -7.99   9.75 345.34
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 277.79231  136.99841   2.028   0.0488 *
## density       0.02118    0.06860   0.309   0.7590
## miles         7.76418   12.20424   0.636   0.5280
## green         4.76871    0.77615   6.144 2.26e-07 ***
## income       -3.84541    2.54599  -1.510   0.1383
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 74.43 on 43 degrees of freedom
```

```
##   (3 observations deleted due to missingness)
## Multiple R-squared:  0.6294, Adjusted R-squared:  0.595
## F-statistic: 18.26 on 4 and 43 DF,  p-value: 7.816e-09
```

This time the R-squared is up at about 0.6, which is a lot better than the 0.08 last time. I would say this does represent a significant improvement, while showing it's still far from perfect, we are getting some signal out of the noise here.

Modeling Interactions and Factors

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

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
```

Next we are asked to try to predict SAT scores from region.

```
#Saving this as a string and factor to be safe
str(states.data$region)
```

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

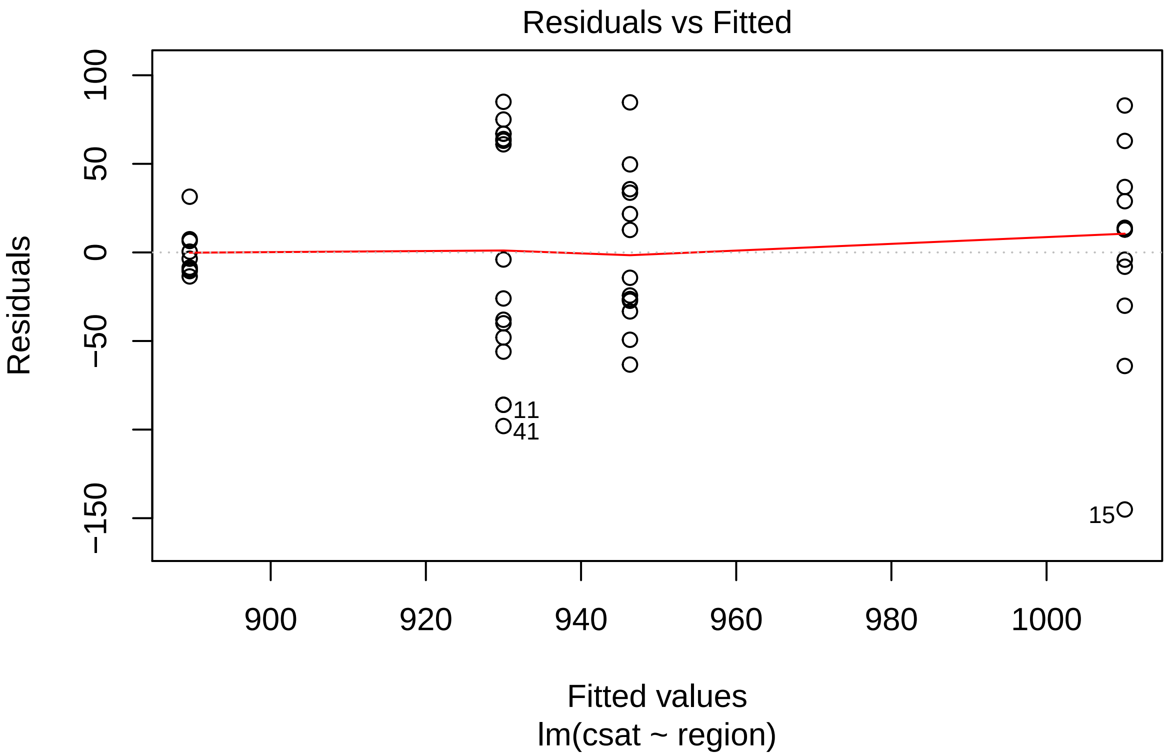
```
states.data$region <- factor(states.data$region)

#Below we try the next model

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

#This model's results are below
coef(summary(sat.region))
```

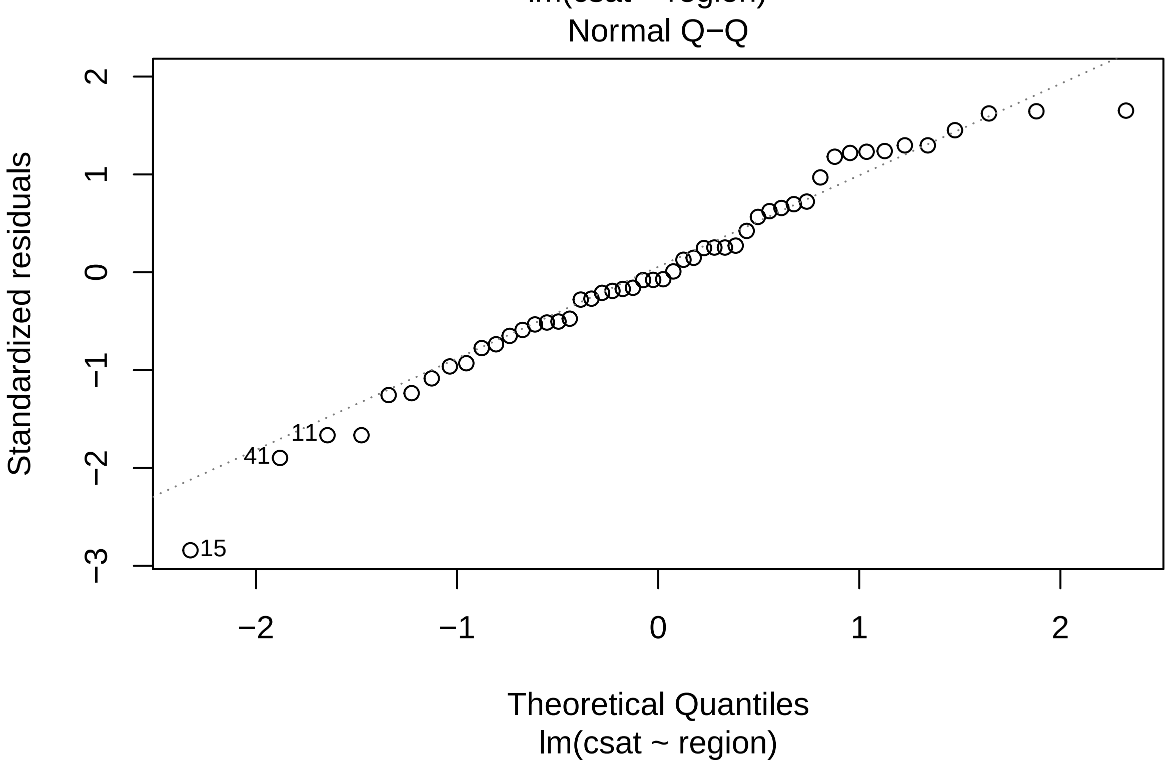
```
##                 Estimate Std. Error    t value     Pr(>|t|)
## (Intercept)    946.30769   14.79582 63.9577807 1.352577e-46
## regionN. East -56.75214   23.13285 -2.4533141 1.800383e-02
## regionSouth   -16.30769   19.91948 -0.8186806 4.171898e-01
## regionMidwest  63.77564   21.35592  2.9863209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

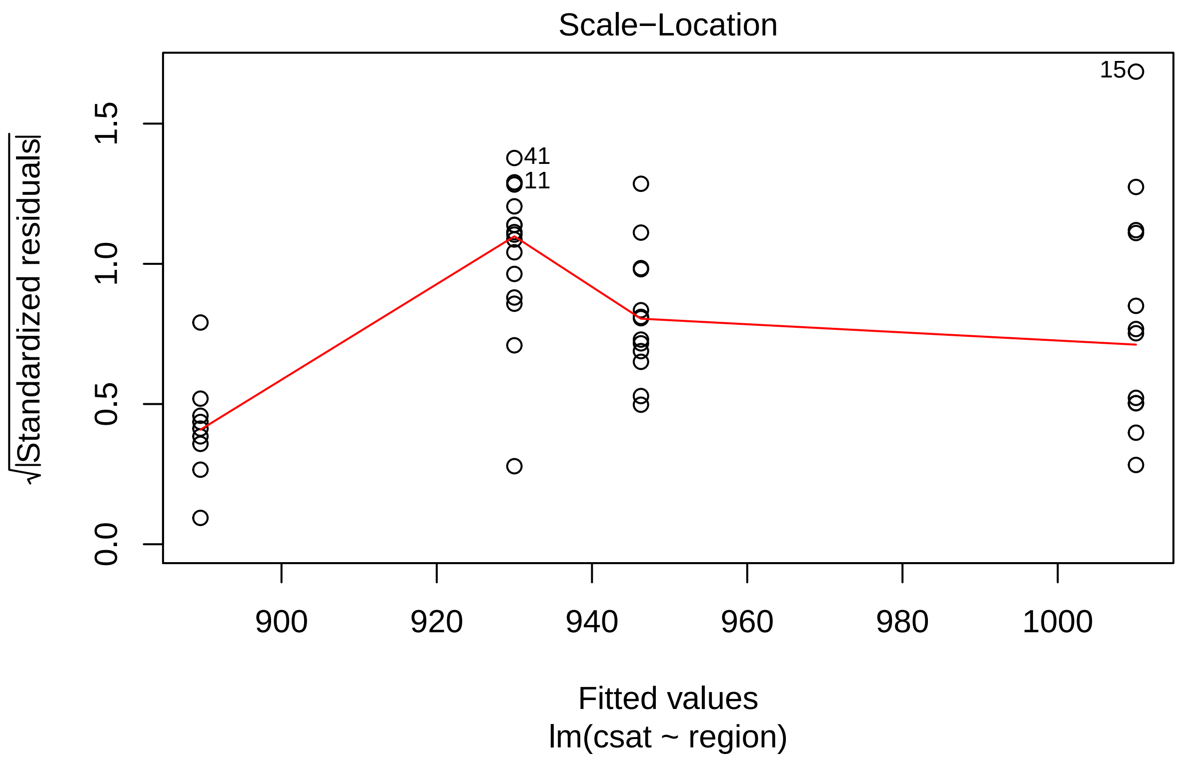
```
anova(sat.region)
```

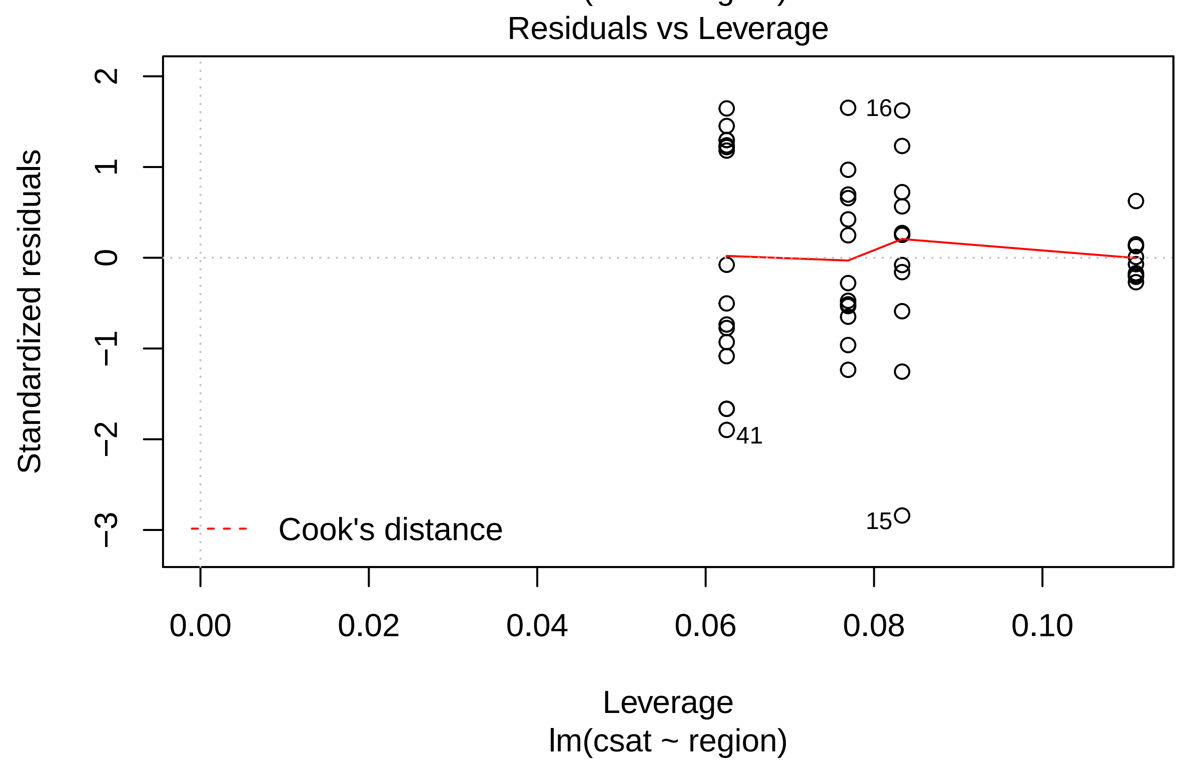
```
## Analysis of Variance Table
##
## Response: csat
##           Df Sum Sq Mean Sq F value    Pr(>F)
## region     3  82049 27349.8  9.6102 4.859e-05 ***
## Residuals 46 130912  2845.9
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
plot(sat.region)
```









It doesn't look like we are getting significant results at all by region.

```
#Prints default contrasts
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
```

```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
```

```r
#Changes 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
```

1.) Add an interaction to the "energy" regression above

```r
energy.by.green.income <- lm(energy ~ income*green,
  data=states.data)

coef(summary(energy.by.green.income))
```

```
##                 Estimate  Std. Error    t value   Pr(>|t|)
## (Intercept)  408.641754 154.3345691  2.6477655 0.01120448
## income        -5.785596   4.6924582 -1.2329563 0.22413903
## green          1.772152   6.6853777  0.2650788 0.79218667
## income:green   0.104115   0.2167739  0.4802929 0.63339848
```

```r
summary(energy.by.green.income)
```

```
##
## Call:
## lm(formula = energy ~ income * green, data = states.data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -84.40 -34.20 -12.80  13.52 348.35
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  408.6418   154.3346   2.648   0.0112 *
## income        -5.7856     4.6925  -1.233   0.2241
## green          1.7722     6.6854   0.265   0.7922
## income:green   0.1041     0.2168   0.480   0.6334
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 73.74 on 44 degrees of freedom
##   (3 observations deleted due to missingness)
## Multiple R-squared:  0.6279, Adjusted R-squared:  0.6025
## F-statistic: 24.75 on 3 and 44 DF,  p-value: 1.552e-09
```

This helped, and produced a better R-squared of 6.025.

2.) Add region to the model

Now we are asked to add region to the model here and see if there are any significant differences in the results between regions in energy usage.

```
energy.by.region <- lm(energy ~ income * green * region,
                       data=states.data)

#Here we can see if throwing in region made our results
#clearer or more confusing

coef(summary(energy.by.region))
```

```
##                                Estimate   Std. Error     t value
## (Intercept)                 446.0903181  623.8857851  0.71501921
## income                       -6.8237011   19.1263584 -0.35676949
## green                        -1.8628994   27.1083792 -0.06872043
## regionN. East             -2458.1693837 2855.6827582 -0.86079918
## regionSouth                 532.7779223  716.5383312  0.74354420
## regionMidwest               151.3307700  811.9143926  0.18638759
## income:green                  0.2094844    0.8560181  0.24471961
## income:regionN. East         65.7544204   80.2997617  0.81886196
## income:regionSouth          -25.1632877   23.3097136 -1.07951938
## income:regionMidwest         -6.6949526   25.8767996 -0.25872413
## green:regionN. East         150.4924682  186.4523237  0.80713646
## green:regionSouth           -31.5158569   32.3235242 -0.97501302
## green:regionMidwest         -11.3751231   32.7549584 -0.34727942
## income:green:regionN. East   -4.1232722    5.2928730 -0.77902345
## income:green:regionSouth      1.4361931    1.1003677  1.30519378
## income:green:regionMidwest    0.4164093    1.0734908  0.38790211
##                             Pr(>|t|)
## (Intercept)                0.4797817
## income                     0.7236063
## green                      0.9456398
## regionN. East              0.3957527
## regionSouth                0.4625770
## regionMidwest              0.8533174
## income:green               0.8082360
## income:regionN. East       0.4189219
## income:regionSouth         0.2884260
## income:regionMidwest       0.7975051
## green:regionN. East        0.4255467
## green:regionSouth          0.3368673
## green:regionMidwest        0.7306552
## income:green:regionN. East 0.4416891
## income:green:regionSouth   0.2011343
## income:green:regionMidwest 0.7006569
```

```r
summary(energy.by.region)
```

```
##
## Call:
## lm(formula = energy ~ income * green * region, data = states.data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -121.186  -30.849   -1.966   22.551  280.430
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  446.0903   623.8858   0.715    0.480
## income                        -6.8237    19.1264  -0.357    0.724
## green                         -1.8629    27.1084  -0.069    0.946
## regionN. East              -2458.1694  2855.6828  -0.861    0.396
## regionSouth                  532.7779   716.5383   0.744    0.463
## regionMidwest                151.3308   811.9144   0.186    0.853
## income:green                   0.2095     0.8560   0.245    0.808
## income:regionN. East          65.7544    80.2998   0.819    0.419
## income:regionSouth           -25.1633    23.3097  -1.080    0.288
## income:regionMidwest          -6.6950    25.8768  -0.259    0.798
## green:regionN. East          150.4925   186.4523   0.807    0.426
## green:regionSouth            -31.5159    32.3235  -0.975    0.337
## green:regionMidwest          -11.3751    32.7550  -0.347    0.731
## income:green:regionN. East    -4.1233     5.2929  -0.779    0.442
## income:green:regionSouth       1.4362     1.1004   1.305    0.201
## income:green:regionMidwest     0.4164     1.0735   0.388    0.701
##
## Residual standard error: 71.44 on 32 degrees of freedom
##   (3 observations deleted due to missingness)
## Multiple R-squared:  0.7459, Adjusted R-squared:  0.6268
## F-statistic: 6.263 on 15 and 32 DF,  p-value: 7.031e-06
```

Surprisingly enough, the R-squared bumped slightly up to 0.62. It seems this didn't hurt the analysis to include region.

There do seem to be significant differences across the regions, but that also could change if the regions were drawn differently.