linear regression Notebook

Contents

1 Introduction

- `Learning objectives:`
- `Learn the R formula interface`
- `Specify factor contrasts to test specific hypotheses`
- `Perform model comparisons`
- `Run and interpret variety of regression models in R`

2 Set working directory

It is often helpful to start your R session by setting your working directory so you don't have to type the full path names to your data and other files

set the working directory setwd("~/Desktop/Rstatistics") setwd("C:/Users/dataclass/Desktop/Rstatistics")

```
##   You might also start by listing the files in your working directory

setwd(getwd()) # where am I?
list.files("linear_regression/dataSets") # files in the dataSets folder
```

```
## [1] "Exam.rds"   "states.dta" "states.rds"
```

```
## Load the states data
##
```

```r
# read the states data
states.data <- readRDS("linear_regression/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
```

3 Linear regression

Examine the data before fitting models

```r
##   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
```

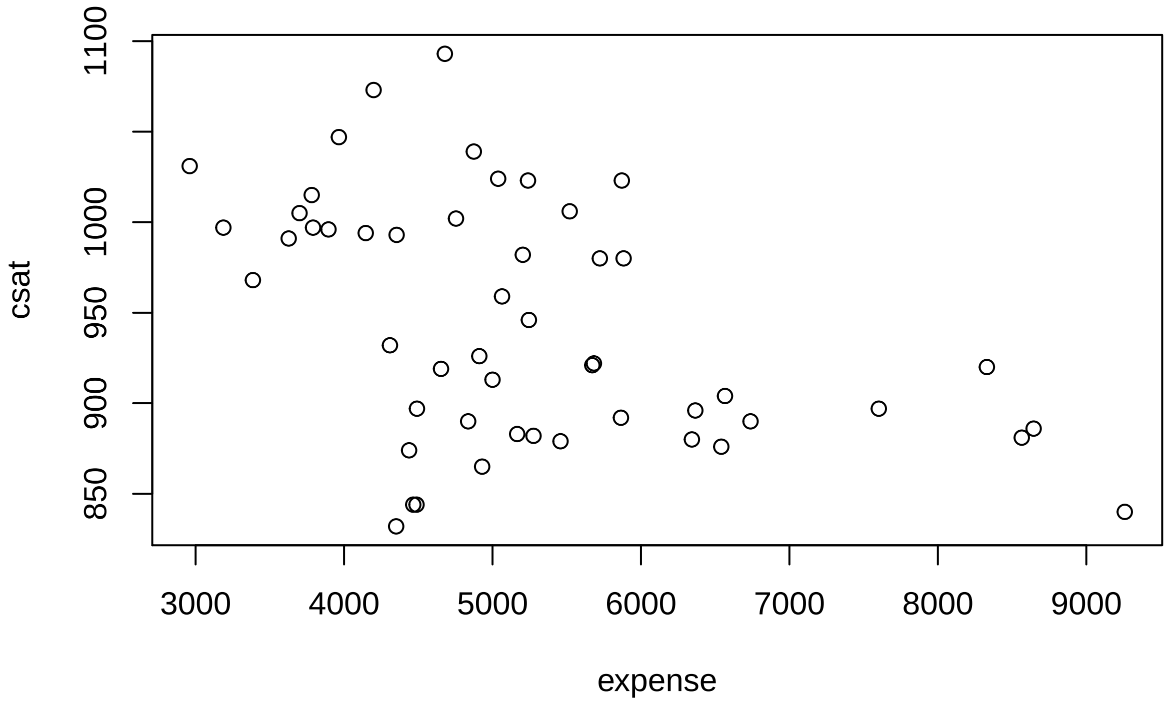
```r
# correlation between expense and csat
cor(sts.ex.sat)
```

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

3.1 Plot the data before fitting models

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

```r
# scatter plot of expense vs csat
plot(sts.ex.sat)
```



3.2 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-pupil expenditures:

```
# Fit our regression model
sat.mod <- lm(csat ~ expense, # regression formula
              data=states.data) # data set
# Summarize and print the results
summary(sat.mod) # show regression coefficients table
```

```
##
## 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
```

3.3 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
```

3.4 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]
```

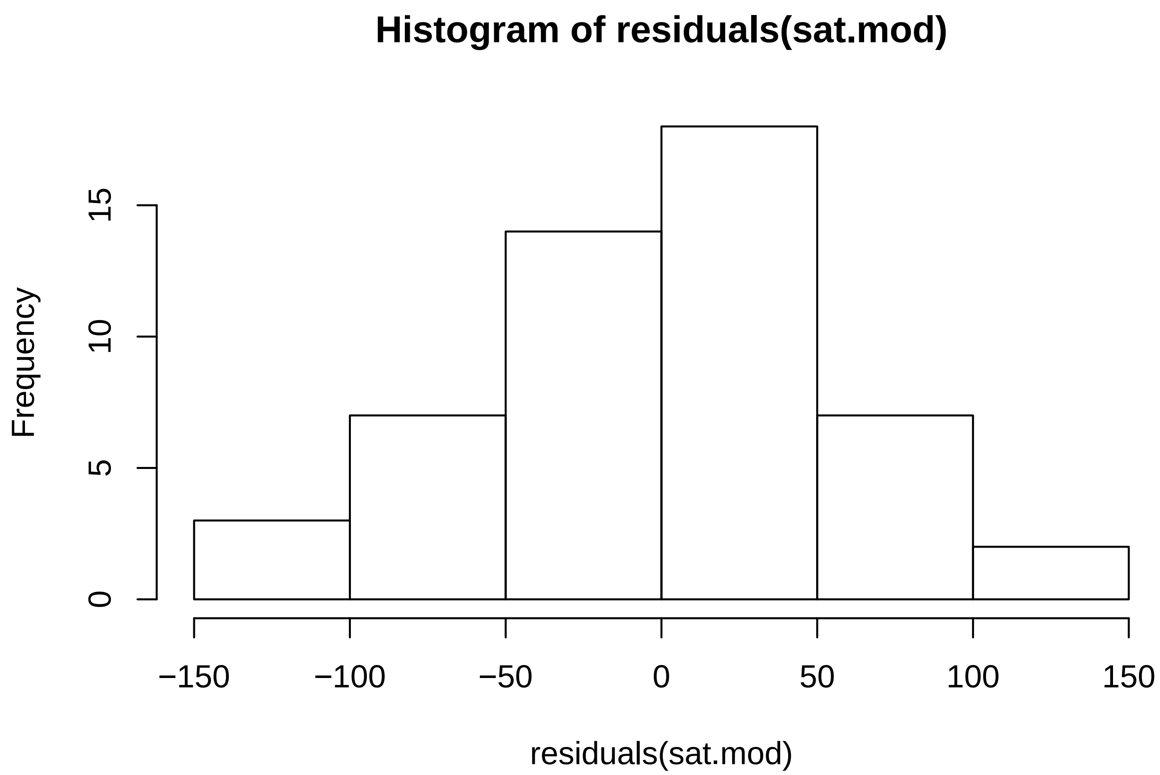
```
## [1] "add1.lm"                      "alias.lm"
## [3] "anova.lm"                     "case.names.lm"
## [5] "coerce,oldClass,S3-method" "confint.lm"
## [7] "cooks.distance.lm"         "deviance.lm"
## [9] "dfbeta.lm"
```

```
##   • 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
```

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

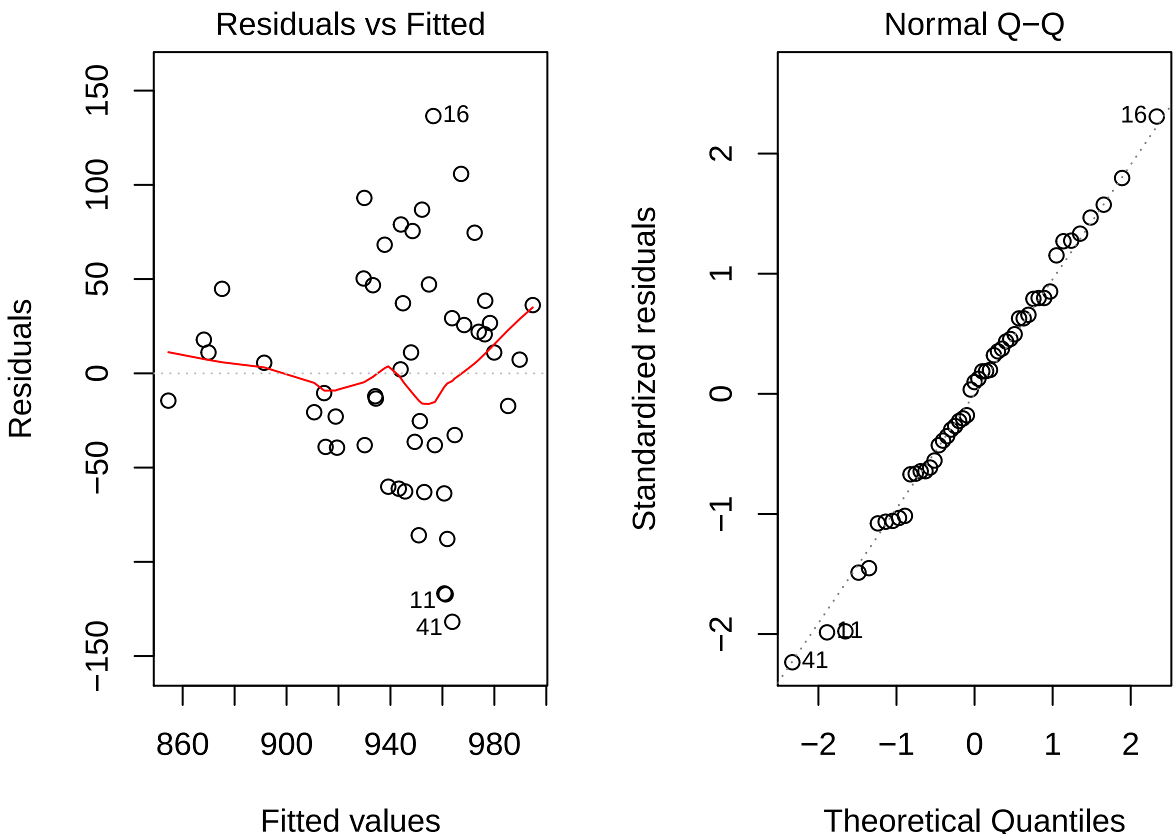


3.5 Linear Regression Assumptions

- 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)) #optional
plot(sat.mod, which = c(1, 2)) # "which" argument optional
```



3.6 Comparing models

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

```
# 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))
# 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
```

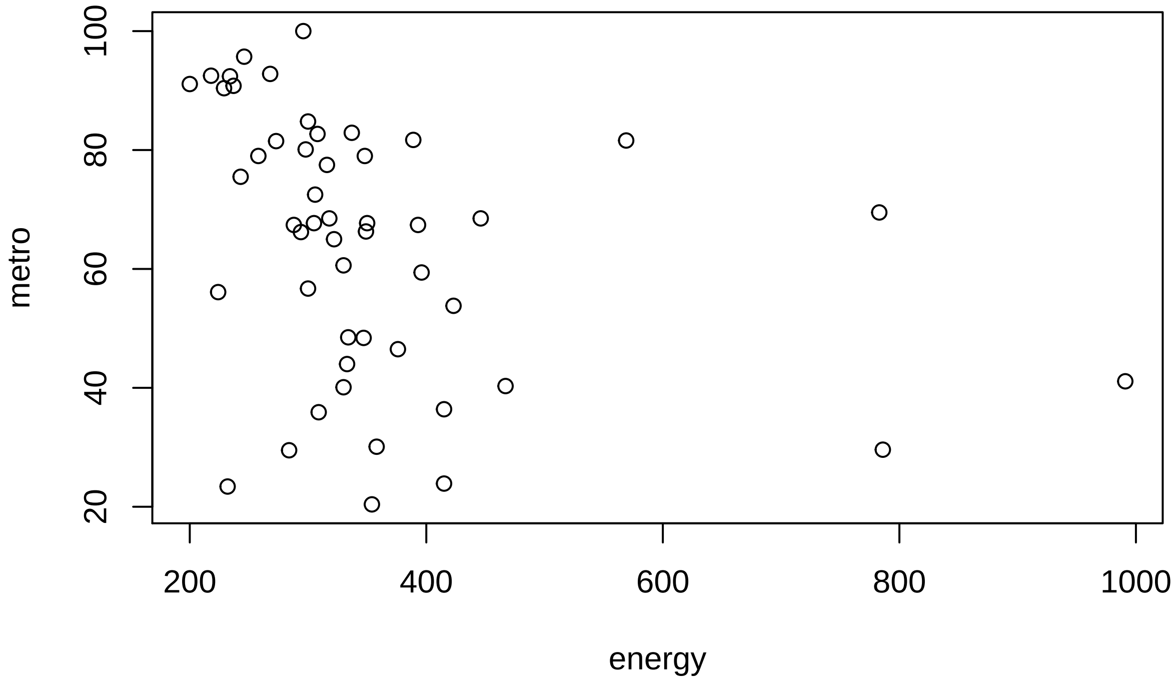
4 Exercise: least squares regression

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

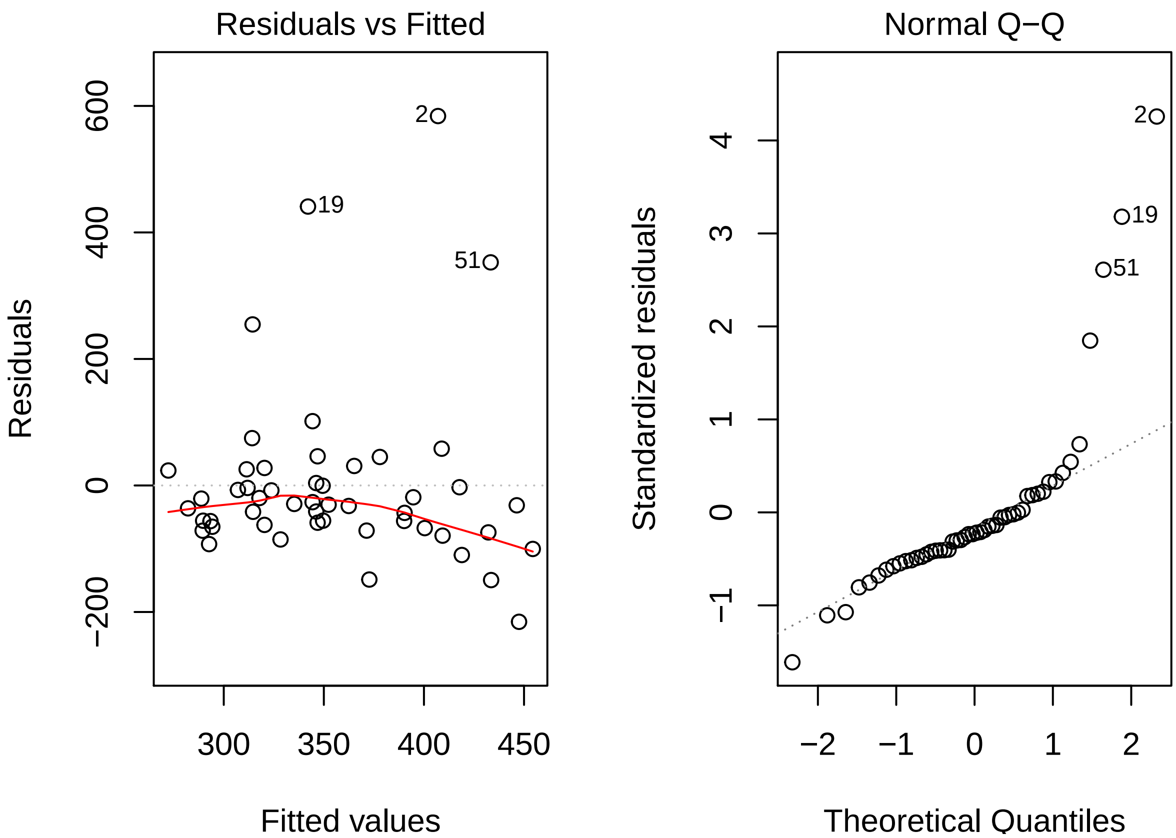
```
## 'data.frame':    51 obs. of  21 variables:
##  $ state  : chr  "Alabama" "Alaska" "Arizona" "Arkansas" ...
##  $ region : Factor w/ 4 levels "West","N. East",..: 3 1 1 3 1 1 2 3 NA 3 ...
##  $ pop    : num  4041000 550000 3665000 2351000 29760000 ...
##  $ area   : num  52423 570374 113642 52075 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 population" "Land area, square mil
##  - 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 _Iregion_4 _IregXperce_2 _IregXpe
##   ..$ : 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"
```

```
## 'data.frame':    51 obs. of  2 variables:
##  $ energy: int  393 991 258 330 246 273 234 349 NA 237 ...
##  $ metro : num  67.4 41.1 79 40.1 95.7 ...
```

```
##        energy metro
## energy      1    NA
## metro      NA     1
```



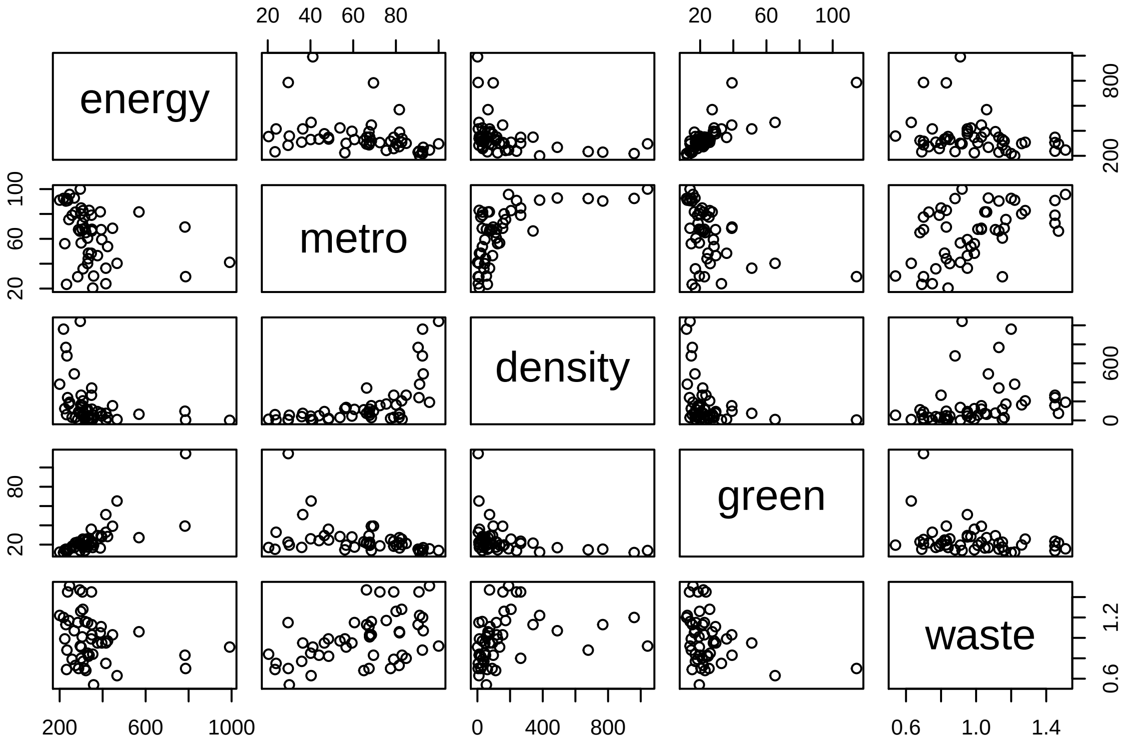

```
##
## Call:
## lm(formula = energy ~ metro, data = states.info)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -215.51  -64.54  -30.87   18.71  583.97
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 501.0292    61.8136   8.105 1.53e-10 ***
## metro        -2.2871     0.9139  -2.503   0.0158 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 140.2 on 48 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.1154, Adjusted R-squared:  0.097
## F-statistic: 6.263 on 1 and 48 DF,  p-value: 0.01578
```



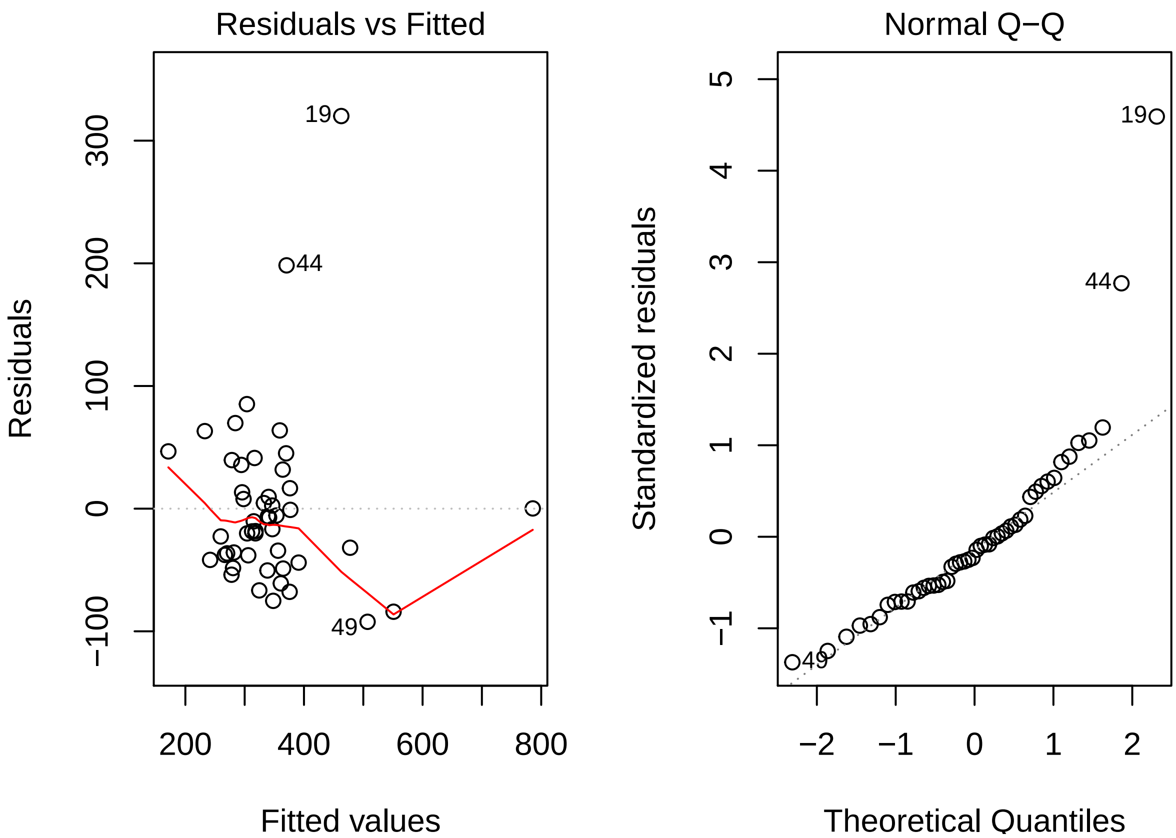
Select one or more additional predictors to add to your model and repeat steps 1-3.

```
## 'data.frame':    51 obs. of  5 variables:
##  $ energy : int  393 991 258 330 246 273 234 349 NA 237 ...
##  $ metro  : num  67.4 41.1 79 40.1 95.7 ...
##  $ density: num  77.08 0.96 32.25 45.15 190.8 ...
##  $ green  : num  29.2 NA 18.4 26 15.6 ...
##  $ waste  : num  1.11 0.91 0.79 0.85 1.51 ...
```

```
##         energy metro density green waste
## energy       1    NA      NA    NA    NA
## metro       NA     1      NA    NA    NA
## density     NA    NA       1    NA    NA
## green       NA    NA      NA     1    NA
## waste       NA    NA      NA    NA     1
```




```
##
## Call:
## lm(formula = energy ~ metro + density + green + waste + density *
##     green, data = states.info2)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -92.25 -38.97 -13.52  20.44 320.07
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    225.34689   61.21256   3.681 0.000656 ***
## metro            0.52713    0.69681   0.756 0.453577
## density         -0.35077    0.16238  -2.160 0.036511 *
## green            4.94475    0.76131   6.495  7.7e-08 ***
## waste          -42.76206   52.12242  -0.820 0.416611
## density:green    0.02012    0.01154   1.743 0.088734 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 74.06 on 42 degrees of freedom
##   (3 observations deleted due to missingness)
## Multiple R-squared:  0.6416, Adjusted R-squared:  0.5989
## F-statistic: 15.04 on 5 and 42 DF,  p-value: 1.845e-08
```



Is this model significantly better than the model with /metro/ as the only predictor?

- Multiple R-squared: 0.1154, Adjusted R-squared: 0.097 # energy ~ metro
- Multiple R-squared: 0.1397, Adjusted R-squared: 0.1031 # energy ~ metro + density
- Multiple R-squared: 0.5962, Adjusted R-squared: 0.5687 # energy ~ metro + green + waste
- Multiple R-squared: 0.5939, Adjusted R-squared: 0.5758 # energy ~ metro + green
- Multiple R-squared: 0.6157, Adjusted R-squared: 0.58 # energy ~ metro + density + green + waste

4.1 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)) # show regression coefficients table
```

```
##                     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
```

4.2 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 ...
```

```
states.data$region <- factor(states.data$region)
#Add region to the model
sat.region <- lm(csat ~ region,
                 data=states.data)
#Show the results
coef(summary(sat.region)) # show regression coefficients table
```

```
##                 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) # show ANOVA table
```

```
## 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
```

Again, *make sure to tell R which variables are categorical by converting them to factors!*

4.3 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
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
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
# 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
```

See also `?contrasts',?contr.treatment', and '?relevel'.

```r
?contrasts
?contr.treatment
?relevel
```

5 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.
```

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