Correlation and Regression: Example

405: Psychometric Theory

Department of Psychology
Northwestern University
Evanston, Illinois USA

April, 2012

Outline

Use R

Get the data

A nice feature of R is that you can read from remote data sets. The example dataset is on the personality-project.org server. Get it and describe it.

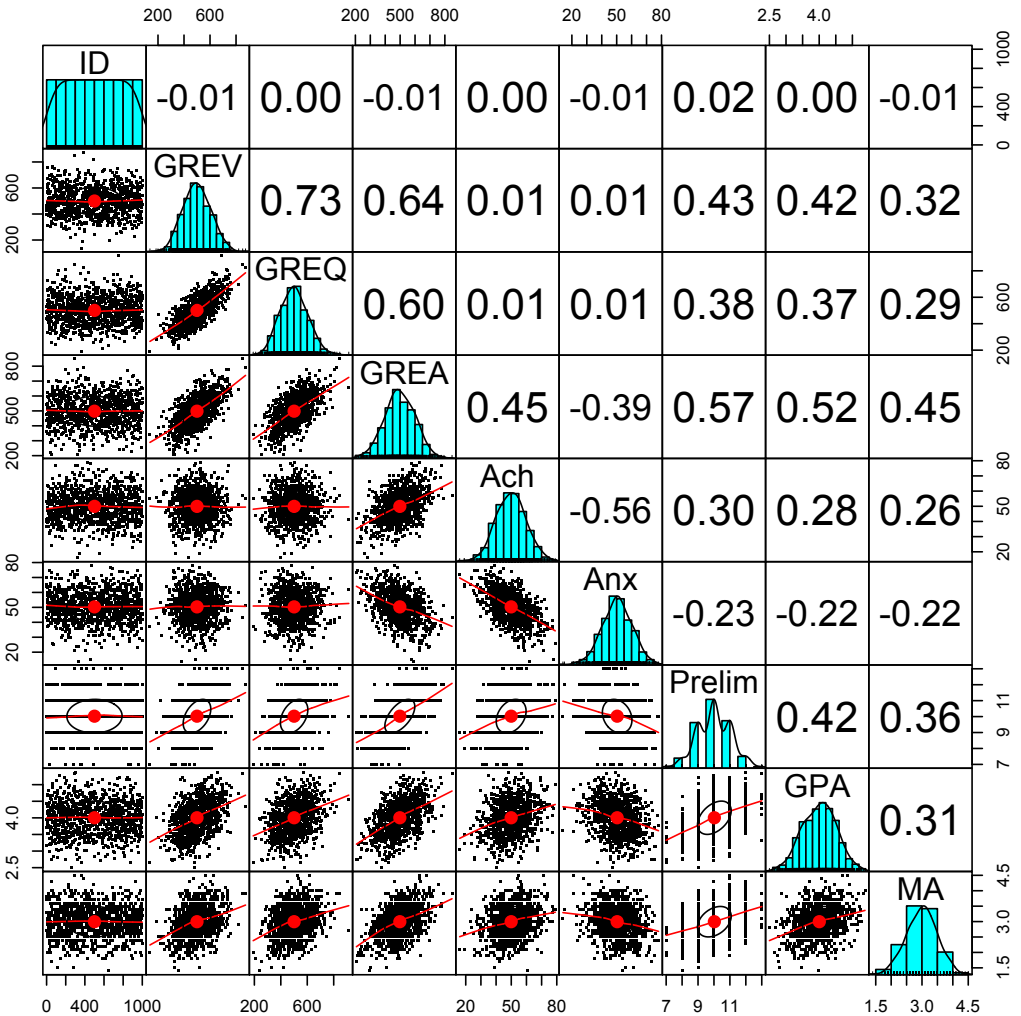
```
> datafilename="http://personality-project.org/R/datasets/psychometrics.prob2.txt"
> mydata =read.table(datafilename,header=TRUE)  #read the data file
> describe(mydata,skew=FALSE)
```

	var	n	mean	sd	median	trimmed	mad	min	max	range	se
ID	1	1000	500.50	288.82	500.50	500.50	370.65	1.0	1000.00	999.00	9.13
GREV	2	1000	499.77	106.11	497.50	498.75	106.01	138.0	873.00	735.00	3.36
GREQ	3	1000	500.53	103.85	498.00	498.51	105.26	191.0	914.00	723.00	3.28
GREA	4	1000	498.13	100.45	495.00	498.67	99.33	207.0	848.00	641.00	3.18
Ach	5	1000	49.93	9.84	50.00	49.88	10.38	16.0	79.00	63.00	0.31
Anx	6	1000	50.32	9.91	50.00	50.43	10.38	14.0	78.00	64.00	0.31
Prelim	7	1000	10.03	1.06	10.00	10.02	1.48	7.0	13.00	6.00	0.03
GPA	8	1000	4.00	0.50	4.02	4.01	0.53	2.5	5.38	2.88	0.02
MA	9	1000	3.00	0.49	3.00	3.00	0.44	1.4	4.50	3.10	0.02

Plot it

Use the pairs.panels function to show a splom plot (use gap=0 and pch='.').

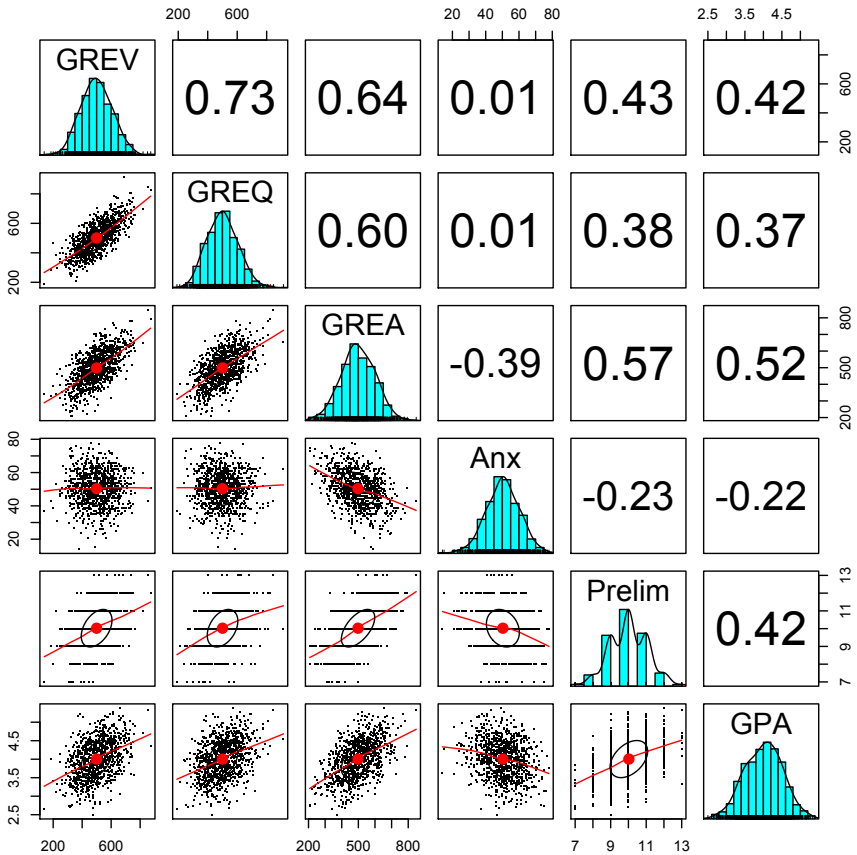
>pairs.panels(mydata,pch=".",gap=0) #pch='.' makes for a cleaner plot



Plot it

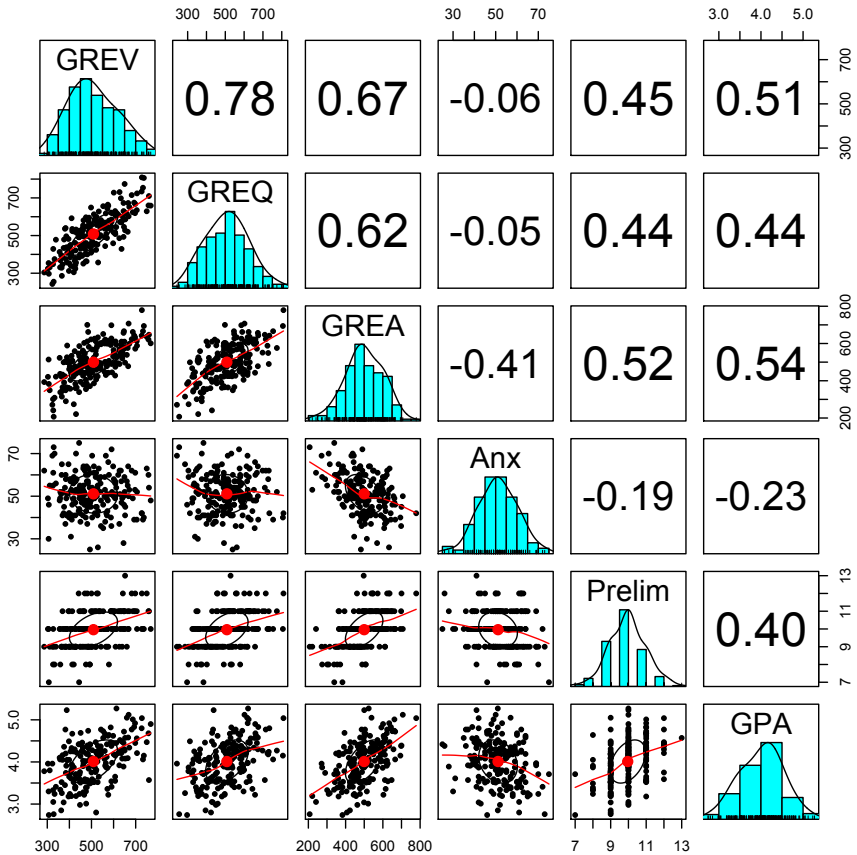
Use the pairs.panels function to show a splom plot. Select a subset of variables using the c() function.

>pairs.panels(mydata[c(2:4,6:8)],pch='.')



Do this for the first 200 subjects

```
> pairs.panels(mydata[mydata$ID < 200,c(2:4,6:8)])
```



0 center the data

In order to do interaction terms in regressions, it is necessary to 0 center the data. We need to turn the result into a data.frame in order to use it in the regression function.

```
> cent <- data.frame(scale(mydata, scale=FALSE))
> describe(cent, skew=FALSE)
```

	var	n	mean	sd	median	trimmed	mad	min	max	range	se
ID	1	1000	0	288.82	0.00	0.00	370.65	−499.50	499.50	999.00	9.13
GREV	2	1000	0	106.11	−2.27	−1.02	106.01	−361.77	373.23	735.00	3.36
GREQ	3	1000	0	103.85	−2.53	−2.02	105.26	−309.53	413.47	723.00	3.28
GREA	4	1000	0	100.45	−3.13	0.54	99.33	−291.13	349.87	641.00	3.18
Ach	5	1000	0	9.84	0.07	−0.05	10.38	−33.93	29.07	63.00	0.31
Anx	6	1000	0	9.91	−0.32	0.11	10.38	−36.32	27.68	64.00	0.31
Prelim	7	1000	0	1.06	−0.03	0.00	1.48	−3.03	2.97	6.00	0.03
GPA	8	1000	0	0.50	0.02	0.00	0.53	−1.50	1.38	2.88	0.02
MA	9	1000	0	0.49	0.00	0.00	0.44	−1.60	1.50	3.10	0.02

The standardized data

Alternatively, we could standardize it.

```
> z.data <- data.frame(scale(my.data))
>  describe(z.data)
```

	var	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
ID	1	1000	0	1	0.00	0.00	1.28	−1.73	1.73	3.46	0.00	−1.20	0.03
GREV	2	1000	0	1	−0.02	−0.01	1.00	−3.41	3.52	6.93	0.09	−0.07	0.03
GREQ	3	1000	0	1	−0.02	−0.02	1.01	−2.98	3.98	6.96	0.22	0.08	0.03
GREA	4	1000	0	1	−0.03	0.01	0.99	−2.90	3.48	6.38	−0.02	−0.06	0.03
Ach	5	1000	0	1	0.01	−0.01	1.05	−3.45	2.95	6.40	0.00	0.02	0.03
Anx	6	1000	0	1	−0.03	0.01	1.05	−3.67	2.79	6.46	−0.14	0.14	0.03
Prelim	7	1000	0	1	−0.02	0.00	1.40	−2.86	2.81	5.67	−0.02	−0.01	0.03
GPA	8	1000	0	1	0.03	0.01	1.06	−3.00	2.74	5.74	−0.07	−0.29	0.03
MA	9	1000	0	1	0.01	0.01	0.90	−3.23	3.04	6.27	−0.07	−0.09	0.03

Find the regression of rated Prelim score on GREV

```
> mod1 <- lm(GPA~GREV, data=mydata)
> summary(mod1)

Call:
lm(formula = GPA ~ GREV, data = mydata)

Residuals:
     Min       1Q   Median       3Q      Max
-1.45807 -0.32322  0.00107  0.32811  1.44850

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 3.0117292  0.0694343   43.38   <2e-16 ***
GREV        0.0019839  0.0001359   14.60   <2e-16 ***
---
Signif. codes:  0 Ô***Õ 0.001 Ô**Õ 0.01 Ô*Õ 0.05 Ô.Õ 0.1 Ô Õ 1

Residual standard error: 0.4558 on 998 degrees of freedom
Multiple R-squared: 0.176, Adjusted R-squared: 0.1751
F-statistic: 213.1 on 1 and 998 DF,  p-value: < 2.2e-16
```

Regression on z transformed data

```
> mod2 <- lm(GPA~GREV, data=z.data)
> summary(mod2)

Call:
lm(formula = GPA ~ GREV, data = z.data)

Residuals:
     Min       1Q   Median       3Q      Max
-2.90526 -0.64404  0.00213  0.65377  2.88619

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 1.888e-17  2.872e-02    0.00        1
GREV        4.195e-01  2.873e-02   14.60   <2e-16 ***
---
Signif. codes:  0 Ô***Õ 0.001 Ô**Õ 0.01 Ô*Õ 0.05 Ô.Õ 0.1 Ô Õ 1

Residual standard error: 0.9082 on 998 degrees of freedom
Multiple R-squared: 0.176, Adjusted R-squared: 0.1751
F-statistic: 213.1 on 1 and 998 DF,  p-value: < 2.2e-16
```

Note that the slope is the same as the correlation.

```
> mod3 <- lm(GPA~GREV, data=cent)
> summary(mod3)

Call:
lm(formula = GPA ~ GREV, data = cent)

Residuals:
     Min       1Q   Median       3Q      Max
-1.45807 -0.32322  0.00107  0.32811  1.44850

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -3.332e-17  1.441e-02    0.00        1
GREV         1.984e-03  1.359e-04   14.60   <2e-16 ***
---
Signif. codes:  0 Ô***Õ 0.001 Ô**Õ 0.01 Ô*Õ 0.05 Ô.Õ 0.1 Ô Õ 1

Residual standard error: 0.4558 on 998 degrees of freedom
Multiple R-squared: 0.176, Adjusted R-squared: 0.1751
F-statistic: 213.1 on 1 and 998 DF,  p-value: < 2.2e-16
```

Note that the slope of the centered data is in the same units as the raw data, just the intercept has changed.

2 predictors

```
> summary(lm(GPA ~ GREV + GREQ , data= cent))

Call:
lm(formula = GPA ~ GREV + GREQ, data = cent)

Residuals:
     Min       1Q   Median       3Q      Max
-1.42442 -0.33228  0.00616  0.32465  1.43765

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -2.651e-17  1.435e-02   0.000  1.00000
GREV         1.534e-03  1.976e-04   7.760 2.10e-14 ***
GREQ         6.314e-04  2.019e-04   3.127  0.00182 **
---
Signif. codes:  0 Ô***Õ 0.001 Ô**Õ 0.01 Ô*Õ 0.05 Ô.Õ 0.1 Ô Õ 1

Residual standard error: 0.4538 on 997 degrees of freedom
Multiple R-squared: 0.184, Adjusted R-squared: 0.1823
F-statistic: 112.4 on 2 and 997 DF,  p-value: < 2.2e-16
```

Multiple R with z transformed data

Do the same regression, but on the z transformed data. The units are now in correlation units.

```
> z.data <- data.frame(scale(my.data))
>  summary(lm(GPA ~ GREV + GREQ , data= z.data))

Call:
lm(formula = GPA ~ GREV + GREQ, data = z.data)

Residuals:
     Min       1Q   Median       3Q      Max
-2.83821 -0.66208  0.01228  0.64688  2.86457

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 3.205e-17  2.860e-02   0.000  1.00000
GREV        3.242e-01  4.179e-02   7.760 2.10e-14 ***
GREQ        1.306e-01  4.179e-02   3.127  0.00182 **
---
Signif. codes:  0 Ô***Õ 0.001 Ô**Õ 0.01 Ô*Õ 0.05 Ô.Õ 0.1 Ô Õ 1

Residual standard error: 0.9043 on 997 degrees of freedom
Multiple R-squared: 0.184, Adjusted R-squared: 0.1823
F-statistic: 112.4 on 2 and 997 DF,  p-value: < 2.2e-16
```

The 3 correlations produce the beta weights

```
> R.small <- cor(my.data[c(2,3,8)])
> round(R.small,2)
     GREV GREQ  GPA
GREV 1.00 0.73 0.42
GREQ 0.73 1.00 0.37
GPA  0.42 0.37 1.00
> solve(R.small[1:2,1:2])
           GREV       GREQ
GREV   2.133188 -1.554768
GREQ -1.554768  2.133188
> beta <- solve(R.small[1:2,1:2],
    R.small[3,1:2])
> beta
     GREV       GREQ
0.3242492 0.1306439
> beta.1 <- (.42 - .73*.37)/(1-.73^2)
> beta.1
[1] 0.3209163
> beta.2 <- (.37 - .73 * .42)/(1-.73^2)
> beta.2
[1] 0.1357311
```

- Find the correlation matrix
- Display it to two decimals
- Find the inverse of GREV and GREQ correlations
- Show them
- Find the beta weights by solving the matrix equation
- show them
- Find the beta weights by using the formula
- Show them

3 predictors, no interactions

Use three predictors, but print it with only 2 decimals

```
> print(summary(lm(GPA ~ GREV + GREQ + GREA , data= cent)),digits=

Call:
lm(formula = GPA ~ GREV + GREQ + GREA, data = cent)

Residuals:
     Min       1Q   Median       3Q      Max
-1.2668  -0.3038   0.0073   0.3051   1.3022

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -6.89e-17   1.35e-02    0.00  1.00000
GREV         6.66e-04   2.00e-04    3.32  0.00092 ***
GREQ         7.75e-05   1.96e-04    0.40  0.69233
GREA         2.08e-03   1.81e-04   11.52  < 2e-16 ***
---
Signif. codes:  0 Ô***Õ 0.001 Ô**Õ 0.01 Ô*Õ 0.05 Ô.Õ 0.1 Ô Õ 1

Residual standard error: 0.427 on 996 degrees of freedom
Multiple R-squared: 0.28, Adjusted R-squared: 0.278
F-statistic:  129 on 3 and 996 DF,  p-value: <2e-16
```

3 predictors, no interactions

Use three predictors, but just the middle 200 subjects

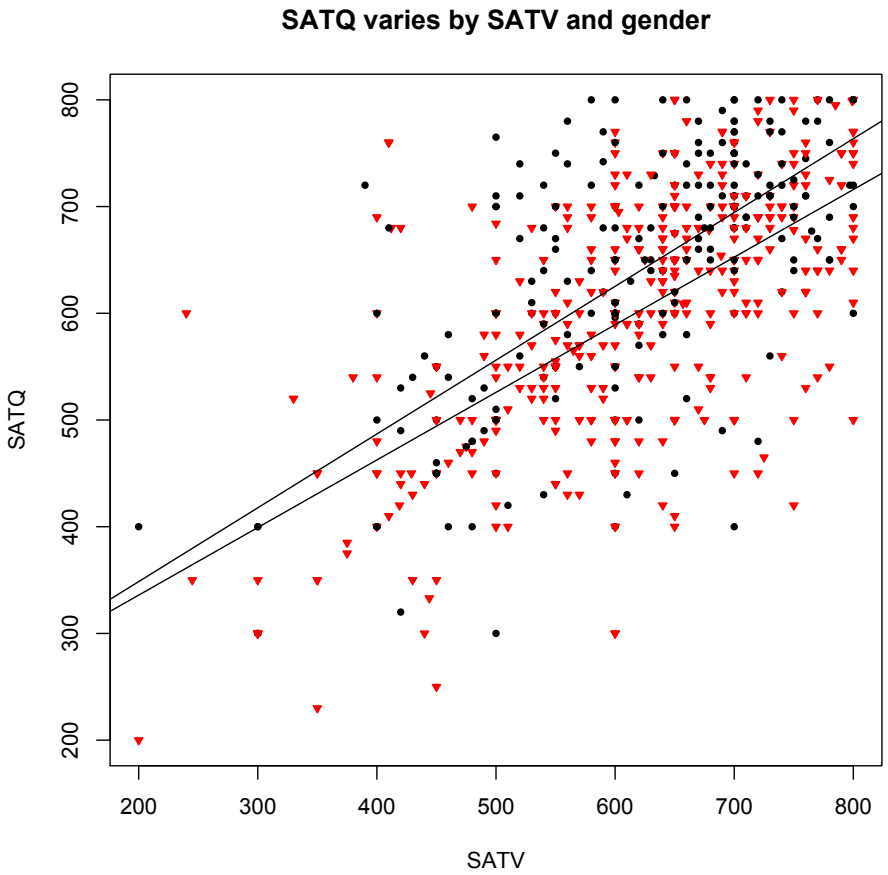
```
> mod4 <- lm(GPA ~ GREV + GREQ + GREA , data= cent[400:600,])
> summary(mod4)

Call:
lm(formula = GPA ~ GREV + GREQ + GREA, data = cent[400:600, ])

Residuals:
     Min       1Q   Median       3Q      Max
-1.03553 -0.30799 -0.00889  0.29320  1.20228

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.0397399  0.0310412   1.280    0.202
GREV        0.0004706  0.0004530   1.039    0.300
GREQ        0.0005236  0.0004515   1.160    0.248
GREA        0.0017904  0.0004360   4.107 5.88e-05 ***
---
Signif. codes:  0 Ô***Õ 0.001 Ô**Õ 0.01 Ô*Õ 0.05 Ô.Õ 0.1 Ô Õ 1

Residual standard error: 0.4394 on 197 degrees of freedom
Multiple R-squared: 0.2259, Adjusted R-squared: 0.2141
F-statistic: 19.16 on 3 and 197 DF,  p-value: 6.051e-11
```

Interaction terms are just products in regression

- To interpret all effects, the data need to be 0 centered.
  - This makes the main effects orthogonal to the interaction term.
  - Otherwise, need to compare model with and without interactions
- Graph the results in non-standardized form
- Consider a real data set of SAT V, SAT Q and Gender

```
> data(sat.act)
>  colors=c("black","red")          #choose some nice colors
>  symb=c(19,25)
>  colors=c("black","red")          #choose some nice colors
> with(sat.act,plot(SATQ~SATV,pch=symb[gender], col=colors[gender]
  bg=colors[gender],cex=.6,main="SATQ varies by SATV and gender")
>  by(sat.act,sat.act$gender,function(x)
                abline(lm(SATQ~SATV,data=x)))
```

An example of an interaction plot




```
>   data(sat.act)
>
c.sat <- data.frame(scale(sat.act,scale=FALSE
>
summary(lm(SATQ~SATV * gender,data=c.sat))

Call:
lm(formula = SATQ ~ SATV * gender, data = c.s

Residuals:
     Min       1Q   Median
3Q      Max
-294.423  -49.876    5.577   53.210
291.100

Coefficients:
              Estimate Std. Error t value Pr(>
(Intercept)  -0.26696    3.31211
-0.081    0.936
SATV          0.65398    0.02926
22.350   < 2e-16 ***
gender      -36.71820    6.91495
-5.310 1.48e-07 ***
SATV:gender  -0.05835    0.06086
-0.959    0.338
---
Signif. codes:
0 Ô***Õ 0.001 Ô**Õ 0.01 Ô*Õ 0.05 Ô.Õ 0.1 Ô Õ

Residual standard error: 86.79 on 683 degrees
  (13 observations deleted due to missingness
```

Interaction of Anxiety with Verbal

```
> mod5 <- lm(GPA ~ GREV * Anx, data=cent)
> summary(mod5)

Call:
lm(formula = GPA ~ GREV * Anx, data = cent)

Residuals:
     Min       1Q   Median       3Q      Max
-1.49677 -0.31527 -0.00054  0.31223  1.32156

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -2.375e-04  1.395e-02  -0.017    0.986
GREV         1.996e-03  1.316e-04  15.167   < 2e-16 ***
Anx         -1.131e-02  1.414e-03  -7.997 3.51e-15 ***
GREV:Anx     2.219e-05  1.377e-05   1.612    0.107
---
Signif. codes:  0 Ô***Õ 0.001 Ô**Õ 0.01 Ô*Õ 0.05 Ô.Õ 0.1 Ô Õ 1

Residual standard error: 0.4412 on 996 degrees of freedom
Multiple R-squared: 0.2294, Adjusted R-squared: 0.227
F-statistic: 98.81 on 3 and 996 DF,  p-value: < 2.2e-16
```

mat.regress and set.cor

- set.cor (formerly mat.regress) in the `psych` package does multiple regressions (without interactions) from the correlation matrix.
- Data can be either a correlation matrix or
- Raw data
- Interface is a bit cruder than `lm` model

Using our data set, first find the correlations. Then show the correlations to two decimals using the `lower.mat` function.

```
> my.R <- cor(mydata)
> lower.mat(my.R,2)
      ID    GREV  GREQ  GREA  Ach   Anx   Prelm GPA   MA
ID     1.00
GREV  -0.01  1.00
GREQ   0.00  0.73  1.00
GREA  -0.01  0.64  0.60  1.00
Ach    0.00  0.01  0.01  0.45  1.00
Anx   -0.01  0.01  0.01 -0.39 -0.56  1.00
Prelim 0.02  0.43  0.38  0.57  0.30 -0.23  1.00
GPA    0.00  0.42  0.37  0.52  0.28 -0.22  0.42  1.00
MA    -0.01  0.32  0.29  0.45  0.26 -0.22  0.36  0.31  1.00
```

Now, find the multiple regression of the first five (not counting ID) variables and the last three. This is in some sense snooping the data.

mat.regress

First, find the correlations, then do the regression

```
> my.R <- cor(mydata)
> set.cor(y=c(7:9),x=2:6,data=my.R)
Call: set.cor(y = c(7:9), x = 2:6, data = my.R)

Multiple Regression from matrix input

Beta weights
      Prelim   GPA    MA
GREV    0.14  0.20  0.10
GREQ    0.04  0.05  0.03
GREA    0.40  0.29  0.31
Ach     0.11  0.12  0.10
Anx    -0.01 -0.05 -0.05

Multiple R
Prelim   GPA    MA
  0.59  0.54  0.47

Multiple R2
Prelim   GPA    MA
  0.34  0.29  0.22
```

mat.regress

Specifying the number of observations gives significance tests.

```
> set.cor(data=my.R,x=c(2:6),y=c(7:9),n.obs=1000)
Call: set.cor(y = c(7:9), x = c(2:6), data = my.R, n.obs = 1000)
Multiple Regression from matrix input
Beta weights
      Prelim   GPA    MA
GREV    0.14  0.20  0.10
GREQ    0.04  0.05  0.03
...
Multiple R
Prelim     GPA      MA
  0.59    0.54    0.47
Multiple R2
Prelim     GPA      MA
  0.34    0.29    0.22
 SE of Beta weights
      Prelim  GPA   MA
GREV    0.04 0.04 0.05
...
 t of Beta Weights
      Prelim   GPA    MA
GREV    3.28  4.50  2.24
...
Probability of t <
       Prelim      GPA       MA
...
 Shrunken R2
Prelim     GPA      MA
  0.34    0.29    0.21
Standard Error of R2
Prelim     GPA      MA
```