Psychology 350: Advanced Statistics and Programming in R
Mediation and Moderation

William Revelle

Department of Psychology
Northwestern University
Evanston, Illinois USA

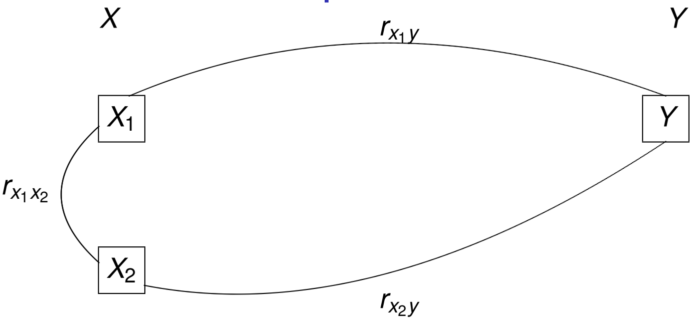
April, 2024

Outline

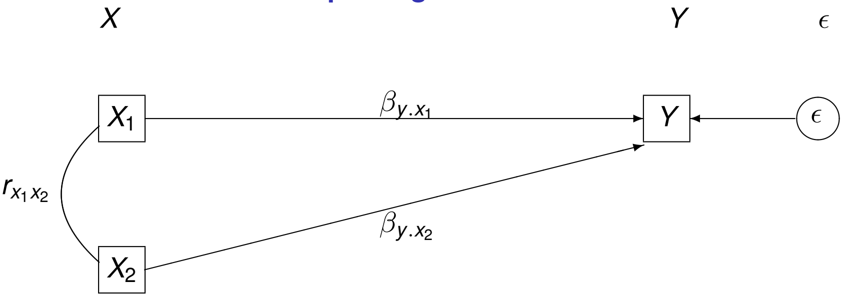
Multiple Regression and Mediation

1. A traditional approach in modeling data is the linear model
2. This is merely $\hat{y} = \beta_{y.x} x$ where
3. $\beta_{y.x} = \frac{\sigma_{xy}}{\sigma_x^2}$
4. In R we can use the `lm` or `lmCor` (in *psych*) to do this.
   - `lm` works on complete data
   - `lmCor` will work on incomplete data, or correlation matrices
5. Regression models the independent contribution of each predictor

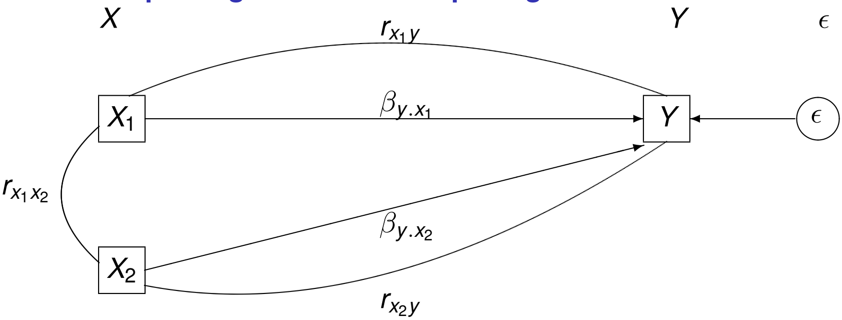
Multiple correlations



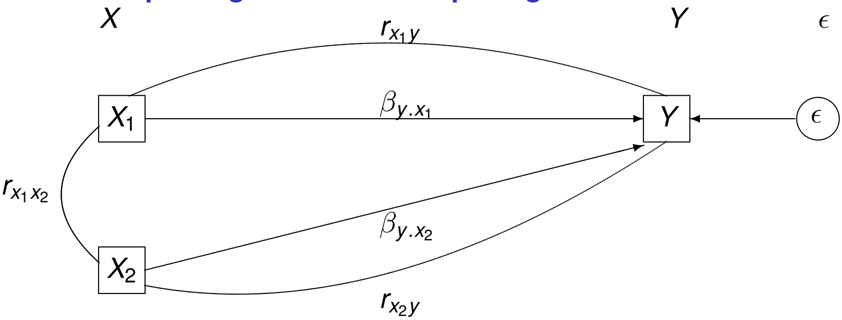
Multiple Regression



Multiple Regression: decomposing correlations



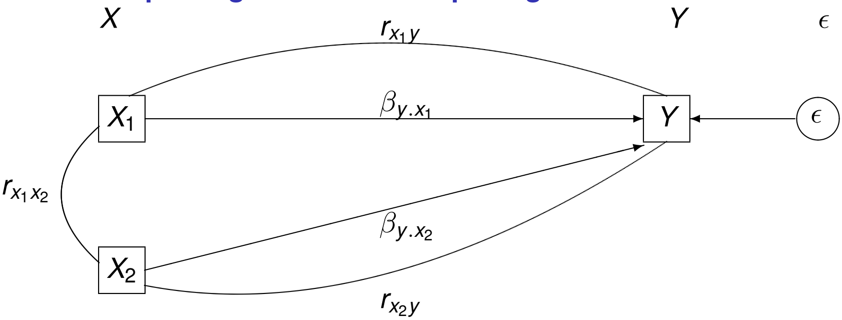
Multiple Regression: decomposing correlations



$$r_{x_1 y} = \overbrace{\beta_{y.x_1}}^{\text{direct}} + \overbrace{r_{x_1 x_2}\beta_{y.x_2}}^{\text{indirect}}$$

$$r_{x_2 y} = \underbrace{\beta_{y.x_2}}_{\text{direct}} + \underbrace{r_{x_1 x_2}\beta_{y.x_1}}_{\text{indirect}}$$

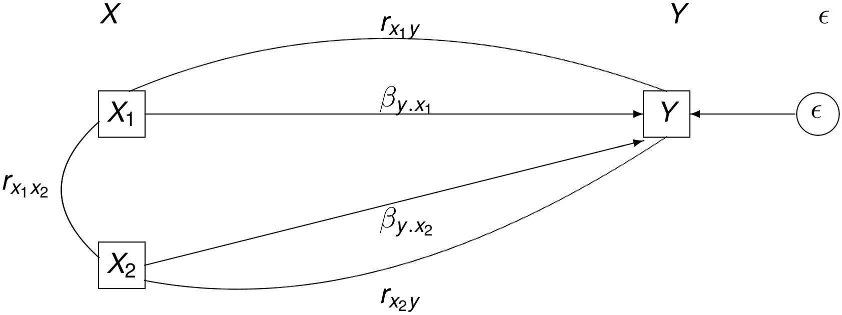
Multiple Regression: decomposing correlations



$$r_{x_1 y} = \overbrace{\beta_{y.x_1}}^{\text{direct}} + \overbrace{r_{x_1 x_2}\beta_{y.x_2}}^{\text{indirect}} \qquad \beta_{y.x_1} = \frac{r_{x_1 y} - r_{x_1 x_2} r_{x_2 y}}{1 - r^2_{x_1 x_2}}$$

$$r_{x_2 y} = \underbrace{\beta_{y.x_2}}_{\text{direct}} + \underbrace{r_{x_1 x_2}\beta_{y.x_1}}_{\text{indirect}} \qquad \beta_{y.x_2} = \frac{r_{x_2 y} - r_{x_1 x_2} r_{x_1 y}}{1 - r^2_{x_1 x_2}}$$

Multiple Regression: decomposing correlations

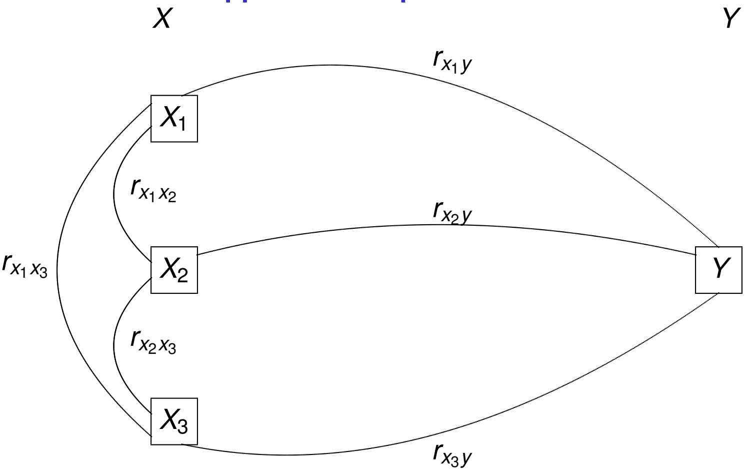


$$r_{x_1 y} = \overbrace{\beta_{y.x_1}}^{direct} + \overbrace{r_{x_1 x_2}\beta_{y.x_2}}^{indirect} \qquad \beta_{y.x_1} = \frac{r_{x_1 y} - r_{x_1 x_2} r_{x_2 y}}{1 - r^2_{x_1 x_2}}$$

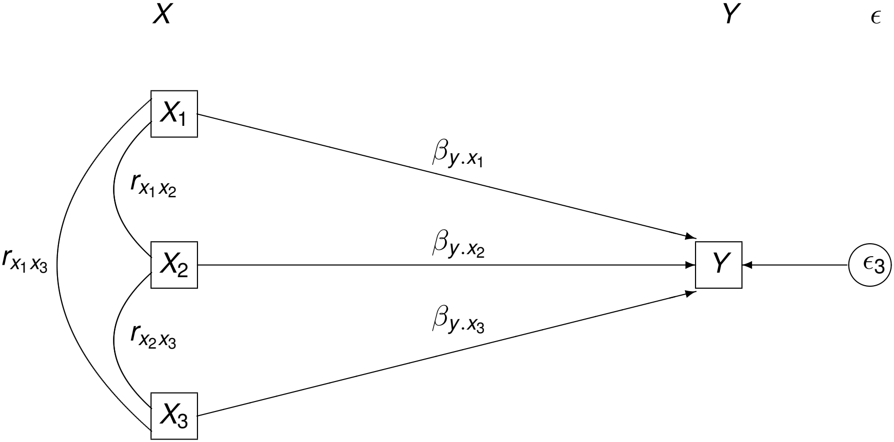
$$r_{x_2 y} = \underbrace{\beta_{y.x_2}}_{direct} + \underbrace{r_{x_1 x_2}\beta_{y.x_1}}_{indirect} \qquad \beta_{y.x_2} = \frac{r_{x_2 y} - r_{x_1 x_2} r_{x_1 y}}{1 - r^2_{x_1 x_2}}$$

$$R^2 = r_{x_1 y}\beta_{y.x_1} + r_{x_2 y}\beta_{y.x_2}$$

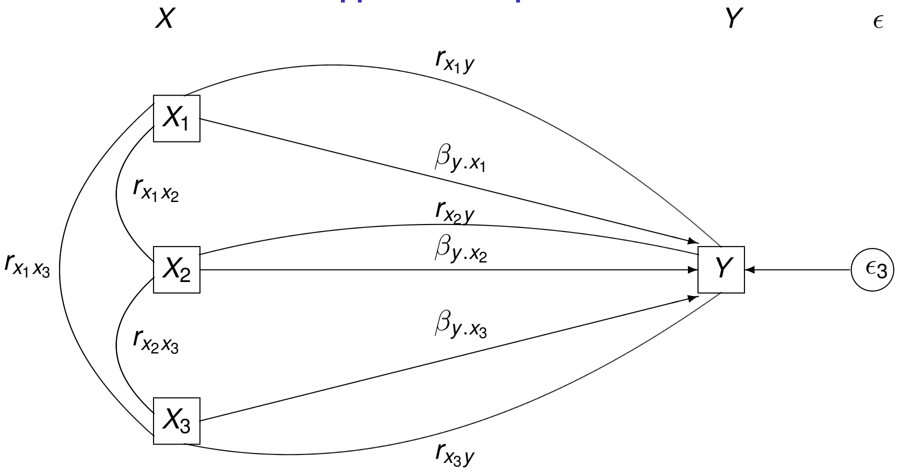
What happens with 3 predictors? The correlations



What happens with 3 predictors? β weights



What happens with 3 predictors?



$$r_{x_1 y} = \underbrace{\beta_{y.x_1}}_{\textit{direct}} + \underbrace{r_{x_1 x_2}\beta_{y.x_1} + r_{x_1 x_3}\beta_{y.x_3}}_{\textit{indirect}} \qquad r_{x_2 y} = \ldots \qquad r_{x_3 y} = \ldots$$

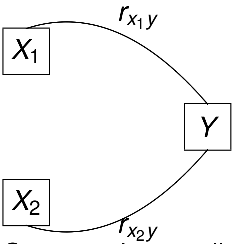
The math gets tedious

Multiple regression and linear algebra

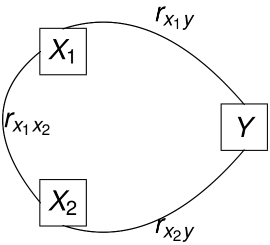
- Multiple regression requires solving multiple, simultaneous equations to estimate the direct and indirect effects.
  - Each equation is expressed as a $r_{x_i y}$ in terms of direct and indirect effects.
  - Direct effect is $\beta_{y.x_i}$
  - Indirect effect is $\sum_{j \neq i} beta_{y.x_j} r_{x_j y}$
- How to solve these equations?
- Tediously, or just use linear algebra.

3 special cases of regression

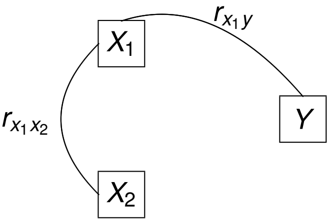
Orthogonal predictors



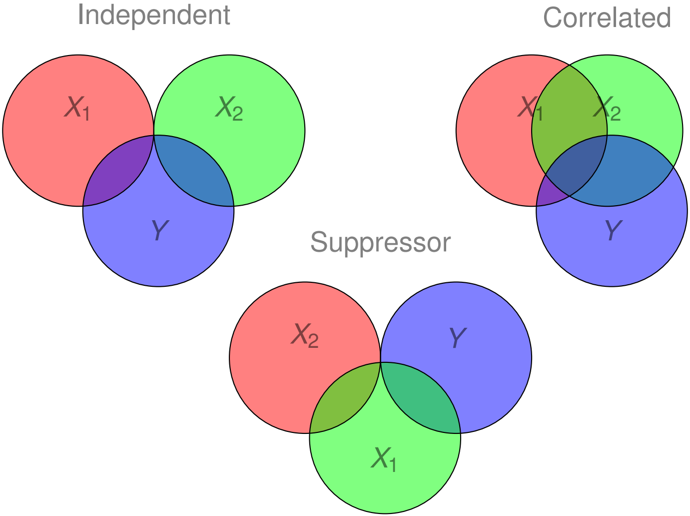
Correlated predictors



Suppressive predictors

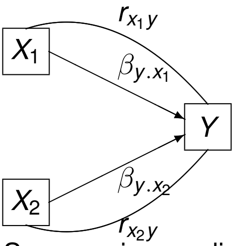


Three basic cases

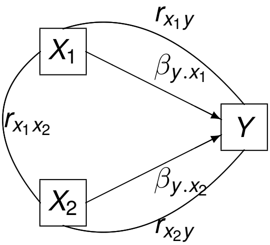


3 special cases of regression

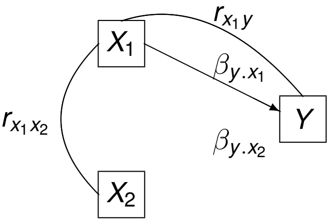
Orthogonal predictors



Correlated predictors



Suppressive predictors

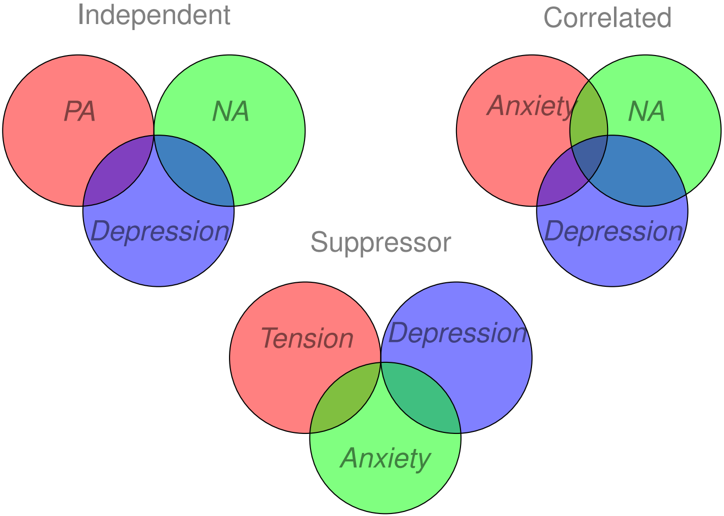


$$\beta_{y.x_1} = \frac{r_{x_1 y} - r_{x_1 x_2} r_{x_2 y}}{1 - r_{x_1 x_2}^2}$$

$$\beta_{y.x_2} = \frac{r_{x_2 y} - r_{x_1 x_2} r_{x_1 y}}{1 - r_{x_1 x_2}^2}$$

$$R^2 = r_{x_1 y}\beta_{y.x_1} + r_{x_2 y}\beta_{y.x_2}$$

Three basic cases: Theoretical examples



The epi.bfi data set (in *psychTools*)

1. A small data set of 5 scales from the Eysenck Personality Inventory, 5 from a Big 5 inventory, a Beck Depression Inventory, and State and Trait Anxiety measures. Used for demonstrations of correlations, regressions, graphic displays.
2. Self report personality scales tend to measure the “Giant 2" of Extraversion and Neuroticism or the “Big 5" of Extraversion, Neuroticism, Agreeableness, Conscientiousness, and Openness. Here is a small data set from Northwestern University undergraduates with scores on the Eysenck Personality Inventory (EPI) and a Big 5 inventory taken from the International Personality Item Pool.

Consider the `epi.bfi` data set

R code

```
describe(epi.bfi); lowerCor(epi.bfi)  #two commands separated by ;
```

```
         vars   n   mean    sd median trimmed   mad min max range  skew kurtosis   se
epiE        1 231  13.33  4.14     14   13.49  4.45   1  22    21 -0.33    -0.06 0.27
epiS        2 231   7.58  2.69      8    7.77  2.97   0  13    13 -0.57    -0.02 0.18
epiImp      3 231   4.37  1.88      4    4.36  1.48   0   9     9  0.06    -0.62 0.12
epilie      4 231   2.38  1.50      2    2.27  1.48   0   7     7  0.66     0.24 0.10
epiNeur     5 231  10.41  4.90     10   10.39  4.45   0  23    23  0.06    -0.50 0.32
bfagree     6 231 125.00 18.14    126  125.26 17.79  74 167    93 -0.21    -0.27 1.19
bfcon       7 231 113.25 21.88    114  113.42 22.24  53 178   125 -0.02     0.23 1.44
bfext       8 231 102.18 26.45    104  102.99 22.24   8 168   160 -0.41     0.51 1.74
bfneur      9 231  87.97 23.34     90   87.70 23.72  34 152   118  0.07    -0.55 1.54
bfopen     10 231 123.43 20.51    125  123.78 20.76  73 173   100 -0.16    -0.16 1.35
bdi        11 231   6.78  5.78      6    5.97  4.45   0  27    27  1.29     1.50 0.38
traitanx   12 231  39.01  9.52     38   38.36  8.90  22  71    49  0.67     0.47 0.63
stateanx   13 231  39.85 11.48     38   38.92 10.38  21  79    58  0.72    -0.01 0.76
          epiE  epiS  epImp epili epiNr bfagr bfcon bfext bfner bfopn bdi   trtnx sttnx
epiE       1.00
epiS       0.85  1.00
epiImp     0.80  0.43  1.00
epilie    -0.22 -0.05 -0.24  1.00
epiNeur   -0.18 -0.22 -0.07 -0.25  1.00
bfagree    0.18  0.20  0.08  0.17 -0.08  1.00
bfcon     -0.11  0.05 -0.24  0.23 -0.13  0.45  1.00
bfext      0.54  0.58  0.35 -0.04 -0.17  0.48  0.27  1.00
bfneur    -0.09 -0.07 -0.09 -0.22  0.63 -0.04  0.04  0.04  1.00
bfopen     0.14  0.15  0.07 -0.03  0.09  0.39  0.31  0.46  0.29  1.00
bdi       -0.16 -0.13 -0.11 -0.20  0.58 -0.14 -0.18 -0.14  0.47 -0.08  1.00
traitanx  -0.23 -0.26 -0.12 -0.23  0.73 -0.31 -0.29 -0.39  0.59 -0.11  0.65  1.00
stateanx  -0.13 -0.12 -0.09 -0.15  0.49 -0.19 -0.14 -0.15  0.49 -0.04  0.61  0.57  1.00
```

Neuroticism and risk for depression

Two models

R code

```
mod1 <- lmCor(bdi ~  epiNeur, data=epi.bfi)  #by default standardizes
mod1a <- lmCor(bdi ~  epiNeur, data=epi.bfi,std=FALSE)  # don't standa
```

```
mod1
Call: lmCor(y = bdi ~ epiNeur, data = epi.bfi)

Multiple Regression from raw data
 DV =  bdi
            slope   se     t       p lower.ci upper.ci VIF
(Intercept)  0.00 0.05  0.00 1.0e+00    -0.11     0.11   1
epiNeur      0.58 0.05 10.74 4.8e-22     0.47     0.68   1
Residual Standard Error =  0.82  with  229  degrees of freedom
 Multiple Regression
      R   R2  Ruw R2uw Shrunken R2 SE of R2 overall F df1 df2        p
bdi 0.58 0.33 0.58 0.33        0.33     0.05    115.25   1 229 4.84e-22

mod1a
Call: lmCor(y = bdi ~ epiNeur, data = epi.bfi, std = FALSE)

Multiple Regression from raw data
 DV =  bdi
            slope   se     t       p lower.ci upper.ci  VIF
(Intercept) -0.32 0.73 -0.44 6.6e-01    -1.76     1.12 5.53
epiNeur      0.68 0.06 10.74 4.8e-22     0.56     0.81 5.53
Residual Standard Error =  4.72  with  229  degrees of freedom
 Multiple Regression
      R   R2  Ruw R2uw Shrunken R2 SE of R2 overall F df1 df2        p
bdi 0.58 0.33 0.41 0.17        0.33     0.05    115.25   1 229 4.84e-22
```

Include Trait Anxiety in the model

R code

```
mod1b  <- lmCor(bdi ~  traitanx, data=epi.bfi)  #by default standardi
mod2  <- lmCor(bdi ~ epiNeur + traitanx, data = epi.bfi)
```

```
Call: lmCor(y = bdi ~ traitanx, data = epi.bfi)

Multiple Regression from raw data

 DV =  bdi
            slope   se     t       p lower.ci upper.ci VIF
(Intercept)  0.00 0.05  0.00 1.0e+00    -0.10     0.10   1
traitanx     0.65 0.05 13.11 1.2e-29     0.56     0.75   1
Residual Standard Error =  0.76  with  229  degrees of freedom
 Multiple Regression
       R   R2  Ruw R2uw Shrunken R2 SE of R2 overall F df1 df2        p
bdi 0.65 0.43 0.65 0.43        0.43     0.05    171.85   1 229 1.15e-29
> mod2
Call: lmCor(y = bdi ~ epiNeur + traitanx, data = epi.bfi)

Multiple Regression from raw data
 DV =  bdi
            slope   se    t       p lower.ci upper.ci  VIF
(Intercept)  0.00 0.05 0.00 1.0e+00    -0.10     0.10 1.00
epiNeur      0.22 0.07 3.02 2.8e-03     0.08     0.36 2.13
traitanx     0.50 0.07 6.94 4.1e-11     0.36     0.64 2.13

Residual Standard Error =  0.74  with  228  degrees of freedom
 Multiple Regression
       R   R2  Ruw R2uw Shrunken R2 SE of R2 overall F df1 df2        p
bdi 0.67 0.45 0.66 0.44        0.45     0.05     93.53   2 228 2.19e-30
```

Show how we do this in matrix terms

1. The `lm` and `lmCor` functions use linear algebra do to the regressions.
2. The crucial operators are matrix multipliation (shown using the %*% operator and
3. solving for the inverse of a matrix (pfunsolve.
4. The inverse of a matrix $\boldsymbol{R}$ is shown as $\boldsymbol{R}^{-1}$ and is that matrix, which when muliplied by the original matrix will yield an Identity matrix *vecI*.
5. $R \,\%*\%\, R^{-1} = I$
6. We can find the beta weights in a regression by solving the equation:

$$\hat{y} = \beta X \implies r_{xy} = \beta R \implies \beta = r_{xy} R^{-1} \quad (1)$$

Show our work

R code

```
R <- lowerCor(epi.bfi[cs(bdi,epiNeur,traitanx)])
X <-  R[2:3,2:3]
Y <- R[2:3,1]
X.inv <-solve(X)
X: Y; X.inv
Y %*% X.inv  #these are the beta weights
```

```
> R
               bdi   epiNeur  traitanx
bdi      1.0000000 0.5786086 0.6547648
epiNeur  0.5786086 1.0000000 0.7286885
traitanx 0.6547648 0.7286885 1.0000000
> X
           epiNeur  traitanx
epiNeur  1.0000000 0.7286885
traitanx 0.7286885 1.0000000
> Y
  epiNeur  traitanx
0.5786086 0.6547648
> X.inv
           epiNeur  traitanx
epiNeur   2.132137 -1.553664
traitanx -1.553664  2.132137
> Y %*% X.inv  #these are the beta weights
       epiNeur traitanx
[1,] 0.2163885 0.497085
solve(X,Y)
  epiNeur  traitanx
0.2163885 0.4970850
```

Do it on the covariances

R code

```
C <-cov(epi.bfi[cs(bdi,epiNeur,traitanx)])
X <-  C[2:3,2:3]
Y <- C[2:3,1,drop=FALSE]
X.inv <-solve(X)
X;Y; X.inv
X.inv %*% Y   #the beta weights
```

```
piNeur  24.00839 33.99642
traitanx 33.99642 90.66079
              bdi
epiNeur  16.37380
traitanx 36.00627
             epiNeur    traitanx
epiNeur   0.08880797 -0.03330164
traitanx -0.03330164  0.02351774
> X.inv %*% Y   #the beta weights
               bdi
epiNeur  0.2550561
traitanx 0.3015115
```

Compare this to the lm function

R code

```
summary(lm(bdi ~ epiNeur + traitanx,data=epi.bfi))
```

```
Call:
lm(formula = bdi ~ epiNeur + traitanx, data = epi.bfi)

Residuals:
     Min       1Q   Median       3Q      Max
-12.0288  -2.6848  -0.5121   1.9823  13.3081

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -7.63779    1.24729  -6.124 3.97e-09 ***
epiNeur      0.25506    0.08448   3.019  0.00282 **
traitanx     0.30151    0.04347   6.935 4.13e-11 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.299 on 228 degrees of freedom
Multiple R-squared:  0.4507,        Adjusted R-squared:  0.4459
F-statistic: 93.53 on 2 and 228 DF,  p-value: < 2.2e-16
```

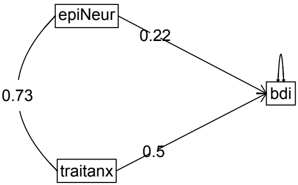
Compare to lmCor without standardization

R code

```
mod2a <- lmCor(bdi ~ epiNeur + traitanx,data=epi.bfi,std=FALSE)
mod2a
```

```
Call: lmCor(y = bdi ~ epiNeur + traitanx, data = epi.bfi, std = FALSE)

Multiple Regression from raw data

 DV =  bdi
            slope   se     t        p lower.ci upper.ci   VIF
(Intercept) -7.64 1.25 -6.12 4.0e-09   -10.10    -5.18 19.44
epiNeur      0.26 0.08  3.02 2.8e-03     0.09     0.42 11.80
traitanx     0.30 0.04  6.94 4.1e-11     0.22     0.39 38.07

Residual Standard Error =  4.3  with  228  degrees of freedom

 Multiple Regression
       R   R2  Ruw R2uw Shrunken R2 SE of R2 overall F df1 df2        p
bdi 0.67 0.45 0.58 0.34        0.45     0.05     93.53   2 228 2.19e-30

anova(mod1a,mod2a)
Model 1 = lmCor(y = bdi ~ epiNeur, data = epi.bfi, std = FALSE)
Model 2 = lmCor(y = bdi ~ epiNeur + traitanx, data = epi.bfi, std = FALSE)
ANOVA Test for Difference Between Models

  Res Df Res SS Diff df Diff SS      F  Pr(F > )
1    229 5103.3
2    228 4214.3       1  889.08 48.101 4.133e-11 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Regression Models



Mediation as a way of organizing a regression

1. Regression models the effect of multiple Xs on Y
2. This partials out all the Xs from each other
3. Mediation models the effect of one X on another X (the Mediator) which then affects Y
4. Typically, we refer to four different paths:
   - a The effect of the X on the mediator
   - b The effect of the Mediator on Y
   - c The effect of X on Y without the mediator
   - c' the effect of X on Y with the mediator c' = c - ab
5. The confidence intervals of ab are found by bootstrapping.
6. The math can not tell if such a model actually makes sense!

The `mediate` function

We use `lm` notation with the addition of () to show mediation. To be consistent with others, by default we do not standardized, but it is easier to understand if we standardize.

R code

```
mod.m <- mediate(bdi ~ epiNeur + (traitanx), data=epi.bfi, std=FALSE)
summary(mod.m)
```

```
Call: mediate(y = bdi ~ epiNeur + (traitanx), data = epi.bfi, std = FALSE)
Direct effect estimates (traditional regression)    (c')
            bdi   se     t  df     Prob
Intercept -7.64 1.25 -6.12 228 3.97e-09
epiNeur    0.26 0.08  3.02 228 2.82e-03
traitanx   0.30 0.04  6.94 228 4.13e-11
R = 0.67 R2 = 0.45   F = 93.53 on 2 and 228 DF   p-value:  2.19e-30

 Total effect estimates (c)
        bdi   se     t  df     Prob
epiNeur 0.68 0.06 10.76 230 3.94e-22
 'a'  effect estimates
         traitanx   se     t  df     Prob
Intercept    24.27 1.01 23.99 229 2.05e-64
epiNeur       1.42 0.09 16.10 229 1.61e-39
 'b'  effect estimates
        bdi   se    t  df     Prob
traitanx 0.3 0.04 6.95 229 3.75e-11
 'ab'  effect estimates (through mediators)
        bdi boot   sd lower upper
epiNeur 0.43 0.42 0.08  0.26  0.58
```

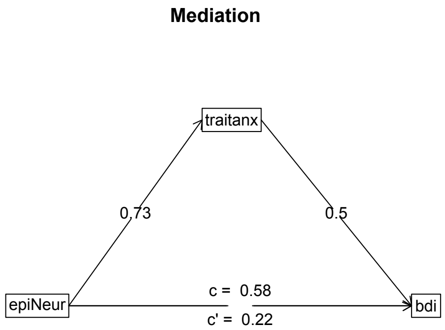
Standardized mediation

R code

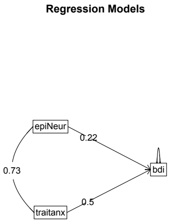
```
mod.m <- mediate(bdi ~ epiNeur + (traitanx), data=epi.bfi, std=TRUE)
summary(mod.m)
```

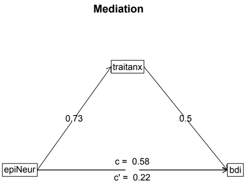
```
Call: mediate(y = bdi ~ epiNeur + (traitanx), data = epi.bfi, std = TRUE)

Direct effect estimates (traditional regression)    (c')
           bdi   se    t  df     Prob
Intercept 0.00 0.05 0.00 228 1.00e+00
epiNeur   0.22 0.07 3.02 228 2.82e-03
traitanx  0.50 0.07 6.94 228 4.13e-11

R = 0.67 R2 = 0.45   F = 93.53 on 2 and 228 DF   p-value:  2.19e-30

 Total effect estimates (c)
         bdi   se     t  df     Prob
epiNeur 0.58 0.05 10.76 230 3.94e-22

 'a'  effect estimates
          traitanx   se    t  df     Prob
Intercept     0.00 0.05  0.0 229 1.00e+00
epiNeur       0.73 0.05 16.1 229 1.61e-39

 'b'  effect estimates
         bdi   se    t  df     Prob
traitanx 0.5 0.07 6.95 229 3.75e-11

 'ab'  effect estimates (through mediators)
         bdi boot   sd lower upper
epiNeur 0.36 0.36 0.06  0.24  0.48
```

One predictor, one criterion



Compare regression to mediation





Add in a more reasonable mediator

1. Trait Neuroticism and Trait Anxiety are stable (although correlated)
2. It is unlikely that Trait anxiety mediates the effect
3. But state anxiety is more like the state of depression
4. Thus, we can examine what happens if we use state anxiety as the mediator
5. First just do the regular regression
6. Then compare it to the mediation model

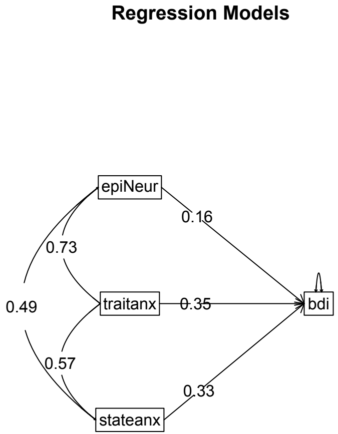
Simple multiple regression

R code

```
mod4 <- lmCor(bdi ~ epiNeur + traitanx + stateanx, data = epi.bfi)
mod4
```

```
Call: lmCor(y = bdi ~ epiNeur + traitanx + stateanx, data = epi.bfi)

Multiple Regression from raw data

 DV =  bdi
            slope   se    t       p lower.ci upper.ci  VIF
(Intercept)  0.00 0.05 0.00 1.0e+00    -0.09     0.09 1.00
epiNeur      0.16 0.07 2.44 1.6e-02     0.03     0.30 2.17
traitanx     0.35 0.07 4.83 2.6e-06     0.20     0.49 2.45
stateanx     0.33 0.06 5.87 1.6e-08     0.22     0.44 1.51

Residual Standard Error =  0.7  with  227  degrees of freedom

 Multiple Regression
      R   R2  Ruw R2uw Shrunken R2 SE of R2 overall F df1 df2        p
bdi 0.72 0.52 0.72 0.52        0.52     0.04     82.96   3 227 2.84e-36
```

Three predictors



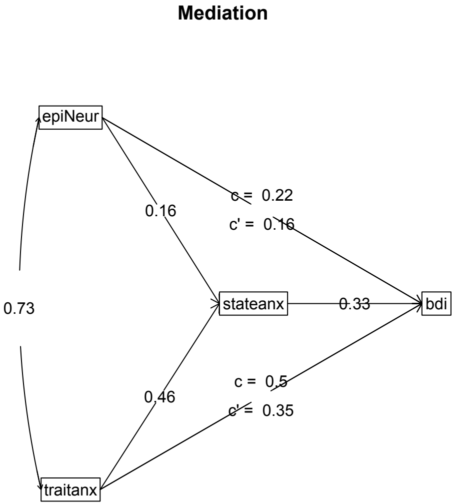
Now express this as a mediation model

R code

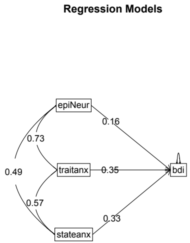
```
mod4m <- mediate(bdi ~ epiNeur + traitanx +( stateanx), data =
      epi.bfi, std=TRUE)
```

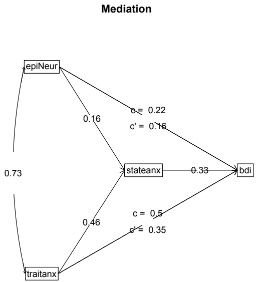
```
summary(mod4m)
Call: mediate(y = bdi ~ epiNeur + traitanx + (stateanx), data = epi.bfi,
    std = TRUE)
Direct effect estimates (traditional regression)    (c')
           bdi   se    t  df     Prob
Intercept 0.00 0.05 0.00 227 1.00e+00
epiNeur   0.16 0.07 2.44 227 1.56e-02
traitanx  0.35 0.07 4.83 227 2.55e-06
stateanx  0.33 0.06 5.87 227 1.56e-08
R = 0.72 R2 = 0.52   F = 82.96 on 3 and 227 DF   p-value:  2.84e-36

 Total effect estimates (c)
          bdi   se    t  df     Prob
epiNeur  0.22 0.07 3.03 229 2.76e-03
traitanx 0.50 0.07 6.95 229 3.75e-11
 'a'  effect estimates
          stateanx   se    t  df     Prob
Intercept     0.00 0.05 0.00 228 1.00e+00
epiNeur       0.16 0.08 1.99 228 4.75e-02
traitanx      0.46 0.08 5.81 228 2.14e-08
 'b'  effect estimates
          bdi   se    t  df     Prob
stateanx 0.33 0.06 5.88 228 1.45e-08
 'ab'  effect estimates (through mediators)
          bdi boot   sd lower upper
epiNeur  0.05 0.05 0.03 -0.01  0.12
traitanx 0.15 0.15 0.04  0.07  0.24
```

Two predictors,one mediator



Compare the two models





Personality, arousal and performance

1. Humphreys and Revelle (1984) presented a theory of cognitive performance that integrates individual differences in personality, situational stressors, and performance on simple and complex.
2. Many of their studies used caffeine as a stressor.
3. Lets look at some simulated data to represent these effects
4. We store a correlation matrix using the `dput` function, which allows us to save the data compactly.

Caffeine, arousal and performance

R code

```
toy <- structure(c(1, 0.6, 0.36, 0.6, 1, 0.6, 0.36, 0.6, 1),
        .Dim = c(3L, 3L), .Dimnames = list(c("caffeine", "arousal", "performance"),
    c("caffeine", "arousal", "performance")))
lowerMat(toy)
dput(toy) #to see how to do it
```

```
 lowerMat(toy)
            caffn arosl prfrm
caffeine    1.00
arousal     0.60  1.00
performance 0.36  0.60  1.00

dput(toy)
structure(c(1, 0.6, 0.36, 0.6, 1, 0.6, 0.36, 0.6, 1), dim = c(3L,
3L), dimnames = list(c("caffeine", "arousal", "performance"),
    c("caffeine", "arousal", "performance")))
```

It seems as if caffeine affects performance,

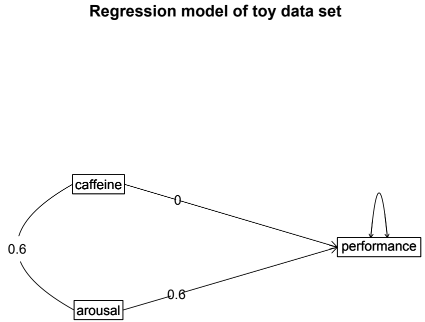
But what if we do a multiple regression

R code

```
mod1 <- lmCor(performance ~ caffeine, data = toy)
mod2 <- lmCor(performance ~ caffeine + arousal, data=toy)
mod1
mod2
```

```
Call: lmCor(y = performance ~ caffeine, data = toy)

Multiple Regression from matrix input

 DV =  performance
         slope VIF Vy.x
caffeine  0.36   1 0.13

 Multiple Regression
              R   R2  Ruw R2uw
performance 0.36 0.13 0.36 0.13
> mod2
Call: lmCor(y = performance ~ caffeine + arousal, data = toy)

Multiple Regression from matrix input

 DV =  performance
         slope  VIF Vy.x
caffeine   0.0 1.56 0.00
arousal    0.6 1.56 0.36

 Multiple Regression
             R   R2  Ruw R2uw
performance 0.6 0.36 0.54 0.29
```

Showing the regression plot



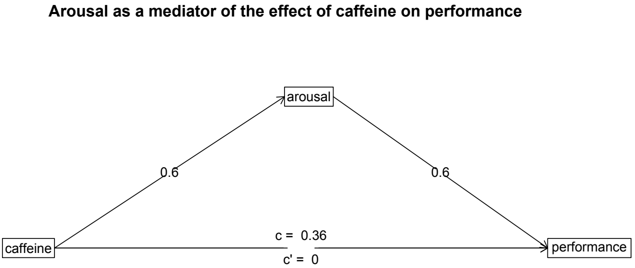
Does arousal mediate the effect?

R code

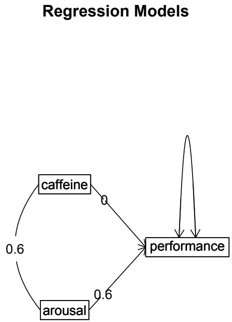
```
mod3 <-  mediate(performance ~ caffeine +(arousal), data=toy)
summary(mod3)
diagram(mod3,main="Arousal as a mediator of the effect of caffeine on
```

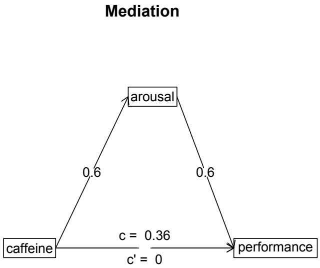
```
Mediation/Moderation Analysis
Call: mediate(y = performance ~ caffeine + (arousal), data = toy)

Direct effect estimates (traditional regression)    (c') X + M on Y
          performance   se     t  df     Prob
Intercept         0.0 0.03  0.00 997 1.00e+00
caffeine          0.0 0.03  0.00 997 1.00e+00
arousal           0.6 0.03 18.95 997 1.32e-68
R = 0.6 R2 = 0.36   F = 280.41 on 2 and 997 DF   p-value:  2.4e-97

 Total effect estimates (c) (X on Y)
          performance   se     t  df     Prob
Intercept        0.00 0.03  0.00 998 1.00e+00
caffeine         0.36 0.03 12.19 998 5.79e-32

 'a'  effect estimates (X on M)
          arousal   se     t  df     Prob
Intercept     0.0 0.03  0.00 998 1.00e+00
caffeine      0.6 0.03 23.69 998 8.08e-99

 'b'  effect estimates (M on Y controlling for X)
        performance   se     t  df     Prob
arousal         0.6 0.03 18.95 997 1.32e-68

 'ab'  effect estimates (through all  mediators)
         performance boot   sd lower upper
caffeine        0.36 0.34 0.02   0.3  0.39
```

Showing the mediation plot



Compare the two plots





Examples of mediation and moderation

1. Garcia et al. (2010) report data for 129 subjects on the effects of perceived sexism on anger and liking of women's reactions to ingroup members who protest discrimination. This data set is also used as the 'protest' data set by Hayes (2013). It is a useful example of mediation and moderation in regression. It may also be used as an example of plotting interactions.
2. Run the examples from the `Garcia` data set in *psychTools*.

R code

```
mod1 <- mediate(respappr ~ prot2 * sexism +(sexism),data=Garcia,
   n.iter=50  ,main="Moderated mediation (mean centered)")
```

```
all: mediate(y = respappr ~ prot2 * sexism + (sexism), data = Garcia,
    n.iter = 50, main = "Moderated mediation (mean centered)")

Direct effect estimates (traditional regression)    (c') X + M on Y
             respappr   se     t  df      Prob
Intercept       -0.01 0.10 -0.12 125 9.07e-01
prot2            1.46 0.22  6.73 125 5.52e-10
prot2*sexism     0.81 0.28  2.87 125 4.78e-03
sexism           0.02 0.13  0.18 125 8.56e-01

R = 0.54 R2 = 0.3   F = 17.53 on 3 and 125 DF   p-value:  1.46e-09

 Total effect estimates (c)  (X on Y)
             respappr   se     t  df      Prob
Intercept       -0.01 0.10 -0.12 126 9.06e-01
prot2            1.46 0.22  6.77 126 4.43e-10
prot2*sexism     0.81 0.28  2.89 126 4.49e-03

 'a'  effect estimates (X on M)
             sexism   se     t  df  Prob
Intercept      0.00 0.07 -0.02 126 0.986
prot2          0.07 0.15  0.47 126 0.642
prot2*sexism   0.09 0.19  0.44 126 0.661

 'b'  effect estimates (M on Y controlling for X)
       respappr   se    t  df  Prob
sexism     0.02 0.13 0.18 125 0.856

 'ab'  effect estimates (through all  mediators)
             respappr boot    sd lower upper
prot2               0    0 0.02 -0.06  0.04
prot2*sexism        0    0 0.05 -0.06  0.04
```

Garcia, D. M., Schmitt, M. T., Branscombe, N. R., and Ellemers, N. (2010). Women's reactions to ingroup members who protest discriminatory treatment: The importance of beliefs about inequality and response appropriateness. *European Journal of Social Psychology*, 40(5):733–745.

Hayes, A. F. (2013). *Introduction to mediation, moderation, and conditional process analysis: A regression-based approach*. Guilford Press, New York.

Humphreys, M. S. and Revelle, W. (1984). Personality, motivation, and performance: A theory of the relationship between individual differences and information processing. *Psychological Review*, 91(2):153–184.