Psychology 454: Latent Variable Modeling
Why latent variables?

William Revelle

Department of Psychology
Northwestern University
Evanston, Illinois USA

October, 2016

Outline

Simple Regression
- Regressions from the raw data
- Regressions from correlation matrix
- Mediation analysis using regression

Compare to a latent variable approach

Why latent variables? What is wrong with regression?

1. Both regression and latent variable approaches try to predict outcomes from a set of predictors.
2. So why not just use regression?
3. What is the advantage of latent variables?

Consider the following correlation matrix

Create a correlation matrix using a congeneric reliability model.
The data are available as R$observed

```
set.seed(42)
R <- sim.congeneric(c(.9,.8,.7,.6,.5),N=100,short=FALSE)
R
```

```
Call: NULL

 $model (Population correlation matrix)
     V1   V2   V3   V4   V5
V1 1.00 0.72 0.63 0.54 0.45
V2 0.72 1.00 0.56 0.48 0.40
V3 0.63 0.56 1.00 0.42 0.35
V4 0.54 0.48 0.42 1.00 0.30
V5 0.45 0.40 0.35 0.30 1.00

$r  (Sample correlation matrix  for sample size =  100 )
     V1   V2   V3   V4   V5
V1 1.00 0.74 0.72 0.56 0.36
V2 0.74 1.00 0.59 0.54 0.41
V3 0.72 0.59 1.00 0.59 0.40
V4 0.56 0.54 0.59 1.00 0.41
V5 0.36 0.41 0.40 0.41 1.00
```

Create and analyze some data

R code

```
set.seed(42)
 R <- sim.congeneric(c(.9,.8,.7,.6,.5),N=100,short=FALSE)
 mod3 <- lm(V5 ~ V1 + V2+ V3,data=as.data.frame(R$observed))
 summary(mod3)
```

```
Call:
lm(formula = V5 ~ V1 + V2 + V3, data = as.data.frame(R$observed))

Residuals:
     Min       1Q   Median       3Q      Max
-1.92474 -0.56386  0.04134  0.60067  2.54943

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.0006735  0.0940828  -0.007   0.9943
V1          -0.0374155  0.1589632  -0.235   0.8144
V2           0.3122922  0.1464000   2.133   0.0355 *
V3           0.2693370  0.1378042   1.954   0.0536 .
---
Signif. codes:  0 ?***? 0.001 ?**? 0.01 ?*? 0.05 ?.? 0.1 ? ? 1
```

Now create another sample from the same population

R code

```
R <- sim.congeneric(c(.9,.8,.7,.6,.5),N=100,short=FALSE)
mod3 <- lm(V5 ~ V1 + V2+ V3,data=as.data.frame(R$observed))
summary(mod3)
```

```
Call:
lm(formula = V5 ~ V1 + V2 + V3, data = as.data.frame(R$observed))

Residuals:
    Min      1Q  Median      3Q     Max
-2.3082 -0.5224  0.0058  0.5213  2.5191

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.06041    0.08913   0.678   0.4995
V1           0.40547    0.13862   2.925   0.0043 **
V2           0.23464    0.12652   1.855   0.0667 .
V3          -0.09227    0.10219  -0.903   0.3688
---
Signif. codes:  0 ?***? 0.001 ?**? 0.01 ?*? 0.05 ?.? 0.1 ? ? 1
```

Can also do it from the correlation matrix.

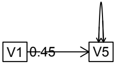
Display the results using the `setCor` function based upon the correlation matrix. For purposes of this display, find the population values.

R code

```
R <- sim.congeneric(c(.9,.8,.7,.6,.5))
op <- par(mfrow=c(2,2)) #set graphics window  to be 2 x 2
setCor(y=5, x=1,R,main="1 predictor")  #just a normal regression
setCor(y=5,x=1:2,R,main="2 predictors") #regressions from correlati
setCor(y=5,x=1:3,R,main="3 predictors")
setCor(5,1:4,R,main="4 predictors") #defaults to y, x order
```

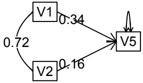
Several regressions using `setCor`

1 predictor



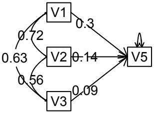
unweighted matrix correlation = 0.45

2 predictors



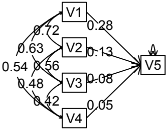
unweighted matrix correlation = 0.46

3 predictors



unweighted matrix correlation = 0.46

4 predictors



unweighted matrix correlation = 0.46

setCor will do regressions from the data or from a correlation matrix

```
Call: setCor(y = 5, x = 1:3, data = R)

Multiple Regression from matrix input

Beta weights
     V5
V1 0.30
V2 0.14
V3 0.09

Multiple R
     V5
V5 0.47
multiple R2
     V5
V5 0.22

 Unweighted multiple R
  V5
0.46
 Unweighted multiple R2
  V5
0.21
```

setCor will do regressions from the data or from a correlation matrix

If we specify the sample size, setCor will do the appropriate statistical tests.

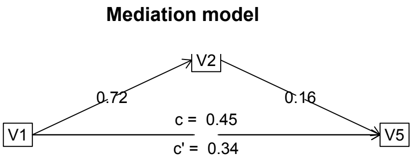
```
setCor(y=5,x=1:3,data=R,n.obs=250)
Call: setCor(y = 5, x = 1:3, data = R, n.obs = 250)

Multiple Regression from matrix input

Beta weights
     V5
V1 0.30
V2 0.14
V3 0.09

Multiple R
     V5
V5 0.47
multiple R2
     V5
V5 0.22
 Unweighted multiple R
  V5
0.46
 Unweighted multiple R2
  V5
0.21
 SE of Beta weights
     V5
V1 0.09
V2 0.08
```

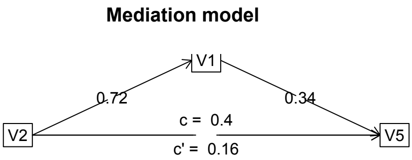
```
 t of Beta Weights
    V5
V1 3.35
V2 1.68
V3 1.16
Probability of t <
       V5
V1 0.00094
V2 0.09500
V3 0.25000

 Shrunken R2
    V5
V5 0.21
Standard Error of R2
      V5
V5 0.045
F
      V5
V5 22.96
Probability of F <
       V5
V5 3.87e-13

 degrees of freedom of regression
```

Mediation models can be even more confusing

1. An alternative regression model is to allow one variable (M) to mediate the relationship of two others (X,Y).
2. This is only appropriate if M has a theoretical mediating role.
3. The analysis compares the $X- > Y$ path (c) with and without the $X- > M$ (a) and $M- > Y$(b) paths.
4. The comparison is then between c and $c' = c - ab$ paths.
5. The significance of the mediation (ab) path is estimated by bootstrapping (Preacher & Hayes, 2004; Preacher, 2015)

Two examples of mediation analysis

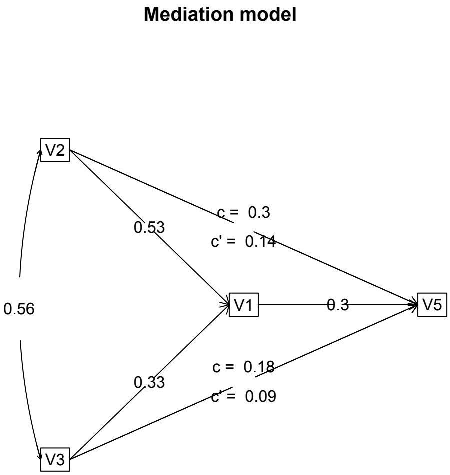




Mediation analysis: output from the bootstrap

```
Mediation analysis
Call: mediate(y = 5, x = 1, m = 2, data = R)

The DV (Y) was  V5 . The IV (X) was  V1 . The mediating variable(s) =  V2 .

Total Direct effect(c) of  V1  on  V5  =  0.45   S.E. =  0.03
                      t direct =  15.92   with probability =  0
Direct effect (c') of  V1  on  V5  removing  V2  =  0.34   S.E. =  0.04
                        t direct =  8.31   with probability =  4.4e-16
Indirect effect (ab) of  V1  on  V5  through  V2   =  0.11
Mean bootstrapped indirect effect =  0.08  with standard error =  0.03
                          Lower CI =  0.02    Upper CI =  0.14
Summary of a, b, and ab estimates and ab confidence intervals
      a   V5   ab mean.ab ci.ablower ci.abupper
V2 0.72 0.16 0.11    0.08       0.02       0.14

ratio of indirect to total effect=   0.25
ratio of indirect to direct effect=  0.34
```

Two predictors, one mediator



Mediation analysis: output from the bootstrap

```
Mediation analysis
Call: mediate(y = 5, x = 2:3, m = 1, data = R)

The DV (Y) was  V5 . The IV (X) was  V2 V3 . The mediating variable(s) =  V1 .

Total Direct effect(c) of  V2  on  V5  =  0.3   S.E. =  0.03
                          t direct =  10   with probability =  0
Direct effect (c') of  V2  on  V5  removing  V1  =  0.14   S.E. =  0.04
                                t direct =  3.37   with probability =  0.00078
Indirect effect (ab) of  V2  on  V5  through  V1   =  0.16
Mean bootstrapped indirect effect =  0.17  with standard error =  0.02
                                 Lower CI =  0.12    Upper CI =  0.22

Total Direct effect(c) of  V3  on  V5  =  0.18   S.E. =  0.03
                                t direct =  5.99   with probability =  2.8e-09
Direct effect (c') of  V3  on  V5  removing  V1  =  0.09   S.E. =  0.04
                                        t direct =  2.33   with probability =  0.02
Indirect effect (ab) of  V3  on  V5  through  V1   =  0.1
Mean bootstrapped indirect effect =  0.11  with standard error =  0.02
                                         Lower CI =  0.08    Upper CI =  0.15
Summary of a, b, and ab estimates and ab confidence intervals
      a  V5   ab mean.ab ci.ablower ci.abupper
V2 0.53 0.3 0.16    0.17       0.12       0.22
V3 0.33 0.3 0.10    0.11       0.08       0.15

ratio of indirect to total effect=   0.53 0.53
ratio of indirect to direct effect=  1.14 0.7
```

Now consider the latent variable approach

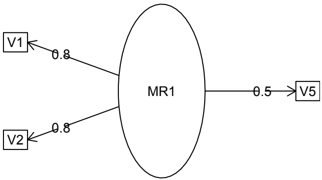
Note that a 2 variable problem is improper and the loadings are set to be equal. Also note that if just one variable is being used as a Y variable, we need to specify that drop=FALSE.

R code

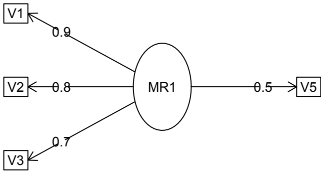
```
op <- par(mfrow=c(2,2))  #do a 2 by 2 output grid
fo <- fa(R[1:2,1:2])  # factor analysis is not properly defined
Fe <- fa.extension(R[1:2,5,drop=FALSE],fo)
fa.diagram(fo,fe=Fe,main = "1 latent, 2 observed")
fo <- fa(R[1:3,1:3]) #with 3 variables, factors are defined
Fe <- fa.extension(R[1:3,5,drop=FALSE],fo)
fa.diagram(fo,fe=Fe, main = "1 latent, 3 observed")
fo <- fa(R[1:4,1:4]) #with 4 variables, model is overdetermined
Fe <- fa.extension(R[1:4,5,drop=FALSE],fo)
fa.diagram(fo,fe=Fe, main = "1 latent, 4 observed")
```

Latent variable model does not change paths as variables are added

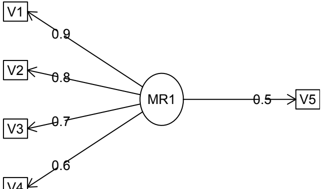
1 latent, 2 observed



1 latent, 3 observed



1 latent, 4 observed



This analysis was done using Factor Extension (Dwyer, 1937)

1. Originally developed (Dwyer, 1937) to *extend* a factor analysis to new variables after doing a laborious factor analysis
2. Now may be done to not change the factor structure when adding "extension variables".
3. Done using the `fa.extension` or `fa.extend` functions.
4. Supply original factor analysis and then the extension correlation matrix (`fa.extension`
5. Or, supply the complete correlation matrix and specify the original and then the extension variables.

Dwyer, P. S. (1937). The determination of the factor loadings of a given test from the known factor loadings of other tests. *Psychometrika*, *2*(3), 173–178.

Preacher, K. J. (2015). Advances in mediation analysis: A survey and synthesis of new developments. *Annual Review of Psychology*, *66*, 825–852.

Preacher, K. J. & Hayes, A. F. (2004). SPSS and SAS procedures for estimating indirect effects in simple mediation models. *Behavior Research Methods, Instruments, & Computers*, *36*(4), 717–731.