Relating Age, Brain and Cognition (ABC)

Rik Henson & Rogier Kievit

2023-08-11

A primer on relating Age, Brain and Cognition: As easy as “ABC”?

This is a primer on simple linear statistical models for relating Age, Brain and Cognition (or indeed any three variables A, B and C). The outcome variable we care about is C, whereas the A variable is generally assumed to be the main cause of variation in C, and one question is how this covariation relates to B. The focus is on different ways of modelling A, B and C, particularly how to treat A (age). This primer deals with cross-sectional data; future extensions will turn to longitudinal measures.

Cross-sectional measures of Age

Let’s start by assuming with have one value for A, B and C from each of “npt” participants (later we will consider latent factors for A, B and C derived from multiple measures of each). We’ll start with defining the Age variable (A) as being drawn from a Gaussian distribution:

```
set.seed(10)  # To render results reproducible
npt = 10000   # Number of participants, big enough to be precise
#A = runif(n = npt, min = 18, max = 88)  # Eg age-range in CamCAN!
Asd = 10      # SD of A
A = rnorm(n = npt, mean = 53, sd = Asd) # If Gaussian (around midpoint of CamCAN, which is also RH's age ;-)
```

We can define some possible “true” models that generate the data for Brain (B) and Cognition (C) from the Age (A) variable. We then fit different statistical models to those data and compare the inferences one might make. We start with the General Linear Model (GLM), which can estimate a single relationship between two sets of multiple variables (e.g. from multiple regression on a single variable, to multivariate methods like CCA/PLS). Later we move to more general path models like Structural Equation Modelling (SEM) that can estimate two or more relationships between variables.

Note that GLMs only assume that the error is Gaussian, while the default maximum likelihood (ML) estimator for SEMs only assumes that endogenous variables (those that “receive” a path from another variable) are Gaussian (though extensions of SEM exist to handle other distributions for variables).

1. General Linear Model for one relationship

1.1. Sequential Causes

We first generate data from a model where A causes B, and B causes C, with some random additional Gaussian variation in each case. This is shown in left panel of Figure 1 (M1).

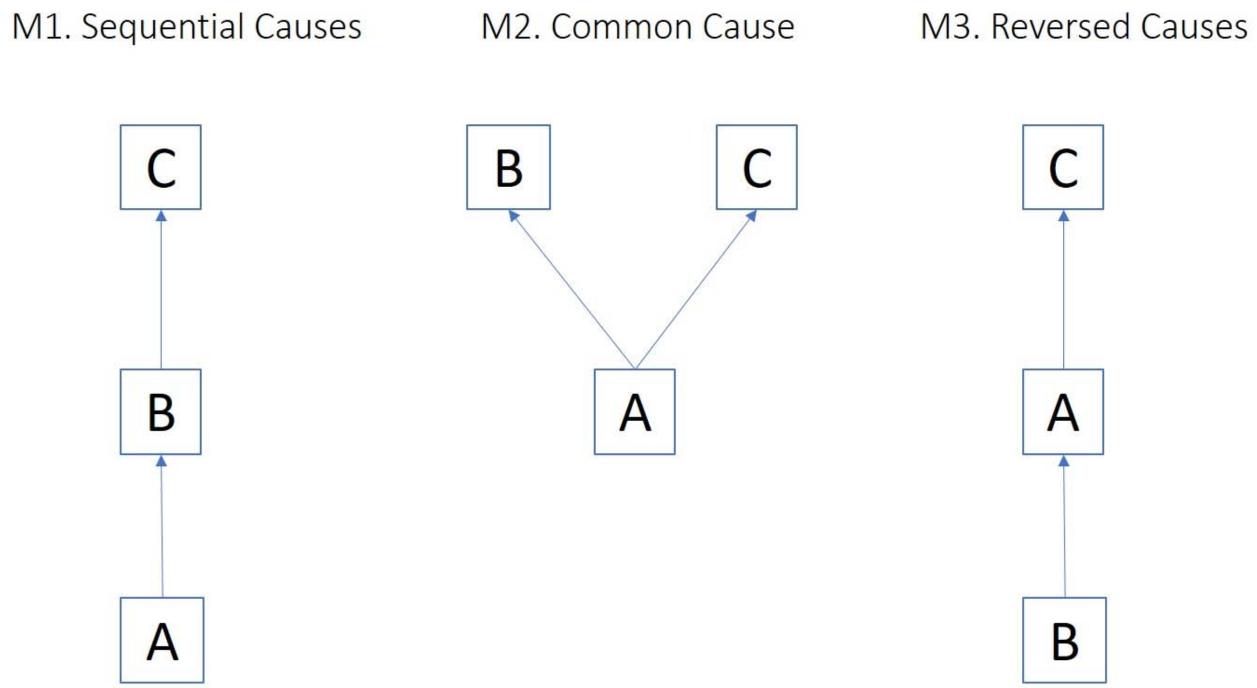


Figure 1. Initial models generating data.

We can gather the results in a data-frame “df1”:

```
Bsd = 10     # SD of B
Csd = 10     # SD of C
B = A + rnorm(n = npt, mean = 0, sd = Bsd)
C = B + rnorm(n = npt, mean = 0, sd = Csd)
df1 = data.frame(A, B, C)
```

We can perform multiple regression (GLM1) to test dependence of C on A and/or B:

```
GLM1 <- lm(C ~ B + A, data = df1)
drop1(GLM1, test="F")
```

```
## Single term deletions
##
## Model:
## C ~ B + A
##        Df Sum of Sq     RSS   AIC   F value Pr(>F)
## <none>              1016303 46219
## B       1    996228 2012530 53050 9799.5277 <2e-16 ***
## A       1         8 1016311 46217    0.0763 0.7824
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

This shows a significant effect of B but not A. This is not incorrect, but note that one cannot really interpret this as Cognition being independent of Age, because a large amount of variation in Brain that causes Cognition is itself caused by Age (i.e, the Brain measure fully mediates the effect of Age on Cognition; see later).

As an aside, note that, by changing the relative sizes of variance in B and C, we can derive another situation where there is a significant effect of B on C when tested alone…

```
Bsd = 2
Csd = 100
B = A + rnorm(n = npt, mean = 0,  sd = Bsd)
C = B + rnorm(n = npt, mean = 0,  sd = Csd)

df = data.frame(A, B, C)
GLM <- lm(C ~ B, data = df)
drop1(GLM, test="F")
```

```
## Single term deletions
##
## Model:
## C ~ B
##        Df Sum of Sq       RSS   AIC F value    Pr(>F)
## <none>               101310678 92238
## B       1   1175678 102486356 92351  116.02 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

…but which no longer reaches significance once A is included in the model:

```
GLM <- lm(C ~ A + B, data = df)
drop1(GLM, test="F") # Incorrect conclusion that no effect of A
```

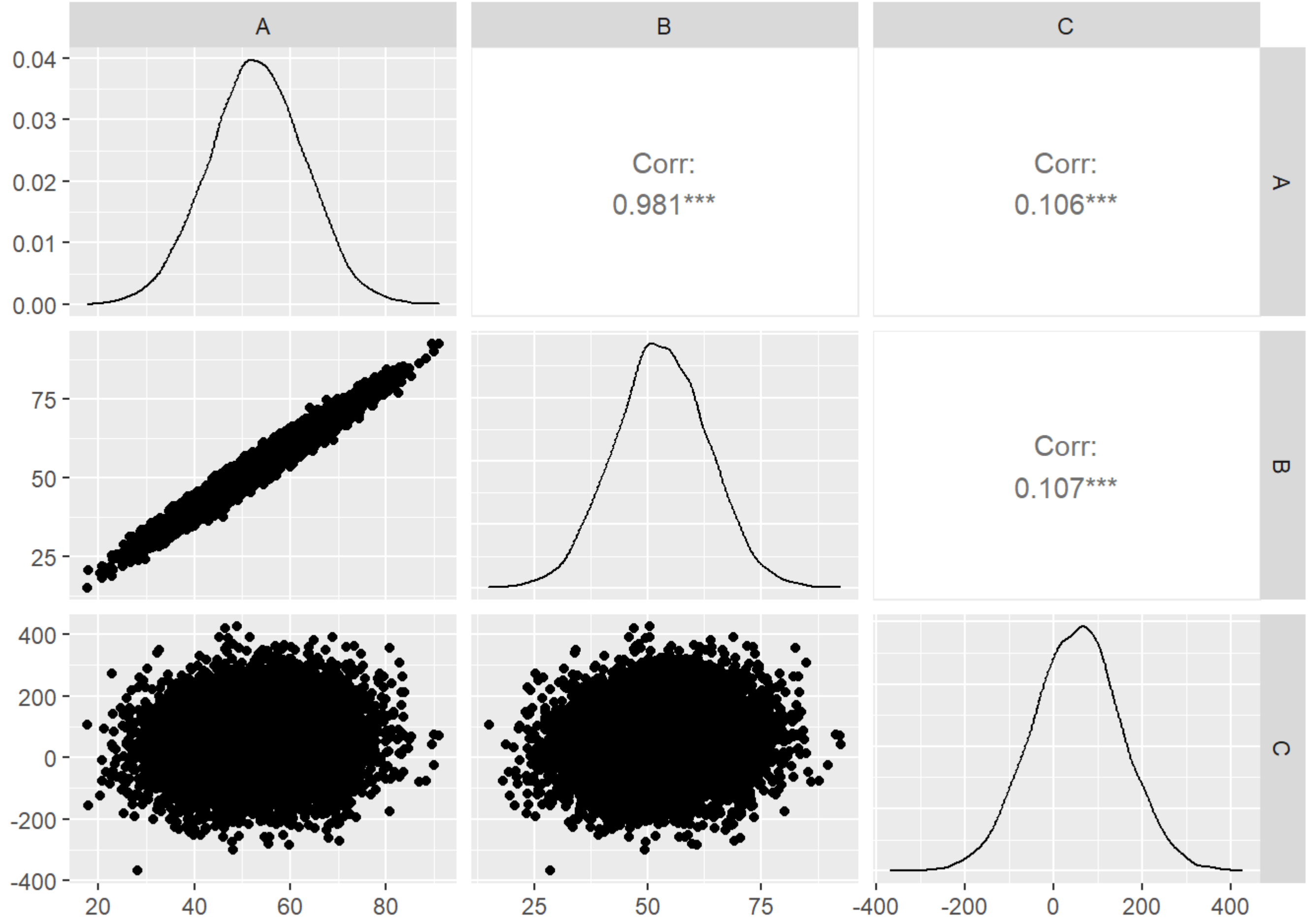
```
## Single term deletions
##
## Model:
## C ~ A + B
##        Df Sum of Sq       RSS   AIC F value  Pr(>F)
## <none>               101309364 92239
## A       1    1314.6 101310678 92238  0.1297 0.71873
## B       1   30010.0 101339374 92240  2.9613 0.08531 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

This “age as a covariate” is a common situation in our experience that tempts some people to argue that Cognition is not related to Brain afterall, since its effect disappears when adjusting for Age. But we know here (because we generated the data) that Brain does contribute to Cognition. However, because Brain also depends on Age, Brain and Age are highly correlated, so the linear model cannot attribute unique variance to either of them. This can be visualised:

```
## Warning: package 'ggplot2' was built under R version 4.2.3
```

```
## Warning: package 'GGally' was built under R version 4.2.3
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```



where the extremely high correlation between A and B means that it is difficult to detect their unique contributions to C. (Of course, this is just another case of “absence of evidence” not being “evidence of absence”, particularly pertinent when statistical power is low, here because of the massive correlation between regressors.) So it would be wrong to conclude Brain is irrelevant to Cognition (in contrast to the next example below).

1.2. Common Cause

We can also create second dataset (`df2`) where A causes both B and C, but B and C are not directly related (model M2 in Figure 1):

```
Bsd = 10
Csd = 10
B <- A + rnorm(n = npt, mean = 0, sd = Bsd)
C <- A + rnorm(n = npt, mean = 0, sd = Csd)
df2 <- data.frame(A, B, C)
```

In this case, while a simple regression might suggest a relationship between B and C…

```
GLM2   <- lm(C ~ B, data = df2)
drop1(GLM2, test="F")
```

```
## Single term deletions
## 
## Model:
## C ~ B
##        Df Sum of Sq     RSS   AIC F value    Pr(>F)    
## <none>              1549112 50433                      
## B       1    512447 2061560 53288  3307.3 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

…this is not the case when A is added as a covariate, which is correct:

```
GLM2   <- lm(C ~ B + A, data = df2)
drop1(GLM2, test="F")
```

```
## Single term deletions
## 
## Model:
## C ~ B + A
##        Df Sum of Sq     RSS   AIC   F value Pr(>F)    
## <none>              1021990 46275                     
## B       1        81 1022071 46274    0.7953 0.3725    
## A       1    527123 1549112 50433 5156.2637 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Thus to summarise, adjusting for A can lead to a correct conclusion (that no direct association between B and C), as in the last example above, but can also lead to the same conclusion even when we know that there is in fact a direct association between B and C, as in the previous example where A and B are highly correlated. This reflects a limitation with the GLM approach; so how can we know whether to adjust for a correlated variable like Age or not? SEM provides one solution…

2. Structural Equation Modelling for multiple relationships

2.1. Sequential Causes

The main advantage of path models like SEM is that they can handle more than one relationship (path), i.e., involve more than one “~” in linear modelling notation. We can then fit two models and test which is more likely to have generated the data (in terms of the full covariance matrix of A, B and C). Let’s start with data generated from the Sequential Model M1, where A->B->C, and then fit by the corresponding SEM (using the “lavaan” package in R):

```
## Warning: package 'lavaan' was built under R version 4.2.3
```

```
## This is lavaan 0.6-15
## lavaan is FREE software! Please report any bugs.
```

```
SEM1 <- 'C ~ B
         B ~ A'
SEM1_df1 <- sem(SEM1, data=df1) # SEM1 fit to Dataset1
#summary(SEM1_df1, fit.measures = TRUE)
anova(SEM1_df1)
```

```
## Chi-Squared Test Statistic (unscaled)
##
##           Df    AIC    BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## Saturated  0               0.0000
## Model      1 149086 149114 0.0763    0.07633       1     0.7823
```

The Chi-square test shows that the model cannot be rejected (i.e, fits reasonably well). Compare this to the fit of the second (incorrect) model (M2) in which B and C are correlated only through shared dependence on A, as below:

```
SEM2 <- 'B ~ A
         C ~ A'
SEM2_df1 <- sem(SEM2, data=df1)
#summary(SEM2_df1, fit.measures = TRUE)
anova(SEM2_df1)
```

```
## Chi-Squared Test Statistic (unscaled)
##
##           Df    AIC    BIC Chisq Chisq diff Df diff Pr(>Chisq)
## Saturated  0                   0
## Model      0 149087 149124     0           0       0
```

This model is more complex than the first one, because it also includes a parameter to model the residual covariance between B and C (to see this, uncomment the `summary` commands above). Indeed, the model is saturated (no df's left), so you cannot use the Chi-square test for model fit. However, you can use the BIC metric, which, as expected, is smaller (better) for the correct (first) model SEM1 (because it is simpler). (Note that this BIC result does depend on the data - for some random seeds, the BIC can be lower for the second, incorrect model, but if you simulate a number of datasets, you will see that BIC more often favours the correct model.) Alternatively, you could set the B-C covariance parameter of SEM2 to zero by adding to the SEM definition `C ~~ 0*B`, thereby equating the number of parameters in each model, and then model comparison will favour SEM1.

2.2. Common Cause

We can also do the converse, ie compare SEM1 and SEM2 in their ability to fit data generated by model M2 with common cause (where SEM2 is now the correct model).

```
SEM1_df2 <- sem(SEM1, data=df2)
SEM2_df2 <- sem(SEM2, data=df2)
#anova(SEM1_df2)
anova(SEM2_df2, SEM1_df2)
```

```
##
## Chi-Squared Difference Test
##
##          Df    AIC    BIC  Chisq Chisq diff   RMSEA Df diff Pr(>Chisq)
## SEM2_df2  0 149038 149074    0.0
## SEM1_df2  1 153195 153224 4159.3     4159.3 0.64485       1  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

This time, the Chi-square, AIC and BIC all favour SEM2 instead, as expected (indeed, if you uncomment the preceding "anova" call, you will see that SEM1 is correctly rejected by the data).

2.3. Reversing “causality”

A third SEM we can test is one in which the direction of the sequential model is altered, ie A causes C, and C causes B (M3 in Figure 1). When we fit SEM3 to data generated by M1:

```
SEM3 <- 'C ~ A
         A ~ B'
SEM3_df1 <- sem(SEM3, data=df1)
anova(SEM3_df1)
```

```
## Chi-Squared Test Statistic (unscaled)
##
##           Df    AIC    BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## Saturated  0                  0.0
## Model      1 148995 149024 6832.2     6832.2       1  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

we can see that SEM3 is correctly rejected by the data (unlike the fit of SEM1, which is not rejected by data, as shown in Section 2.1 above). This is because SEM1 predicts greater covariance between B and C than between A and C (as found in data, given their sequential generation), whereas SEM3 predicts opposite of greater covariance between A and C than between B and C. In other words, SEM can provide some insight on the possible causal direction of relationships between variables (the “watershed” approach, e.g., Kievit et al. 2016 (https://doi.org/10.1016/j.neuropsychologia.2016.08.008)).

However, you will note that the AIC and BIC are actually lower for SEM3 (above) than SEM1 (Section 2.1). This relates to the scaling of the data, which affects AIC/BIC (but not Chi-square): The B variable has greater spread than the A variable (since B is generated by adding independent variance to A), so the residuals of the model fit will be numerically greater when B is an endogeneous variable (in SEM1) than when it is an exogeneous variable (at base of SEM3), leading to worse AIC/BIC, even though the proportion of total variance explained is less. This can be addressed by re-scaling (Z-scoring) all the variables first, as illustrated with the commented code below.

```
#df1$Az<-scale(df1$A)
#df1$Bz<-scale(df1$B)
#df1$Cz<-scale(df1$C)
#SEM1z <- 'Cz ~ Bz
#          Bz ~ Az'
#SEM1z_df1 <- sem(SEM1z, data=df1)
#SEM3z <- 'Cz ~ Az
#          Az ~ Bz'
#SEM3z_df1 <- sem(SEM3z, data=df1)
#anova(SEM1z_df1)
#anova(SEM3z_df1)
```

Note also that we could reverse M1 fully, i.e. have B cause A and C cause B. If you try this (with commented code below), you will see that this fully reversed SEM3 fits the data as well as SEM1 according to Chi-square. This is because both predict less covariance between A and C than between B and C, and than between A and B. In fact, the fully-reversed SEM3 actually fits better according to AIC/BIC. The latter again relates to the scaling of the data: The greater spread of the C variable (since generated by adding A and B) leads to higher absolute residuals (and hence worse AIC/BIC) when it is the outcome variable than when A is the outcome variable. After Z-scoring however, the two models can no longer be distinguished (have equivalent AIC/BIC), as also illustrated with the commented code below. This is because their predicted covariance matrices (after scaling) are identical, demonstrating some limitations of SEM for inferring directionality.

```
#SEM3 <- 'A ~ B
#         B ~ C'
#SEM3_df1 <- sem(SEM3, data=df1)
#anova(SEM1_df1)
#anova(SEM3_df1)
#
#SEM3z <- 'Cz ~ Bz
#          Bz ~ Az'
#SEM3z_df1 <- sem(SEM3z, data=df1)
#anova(SEM1z_df1)
#anova(SEM3z_df1)
```

More generally, note that there are some restrictions on using SEM for such model comparison. For example, you can only compare SEMs that include the same variables, i.e, that fit the same data covariance matrix. Thus you cannot use Chi-square, AIC or BIC to compare the SEM `C~A` with the SEM `C~A+B` (this is unlike the GLM, where you can use such metrics, because the model fit is always restricted to the outcome variable C). Thus if you wanted to ask whether a SEM in which C depends only on A is better than one where it depends on A and B, then you would compare `C~A+B` with `C~A+0*B`. Note furthermore that while Chi-square could be used in this situation, it can only be used to compare nested models, e.g., cannot be used to compare `C~A+B` with `C~A, A~B`, because one parameter (C~B path) has been removed in second case while another (A~B path) has been added. In the latter situation, you need to use AIC/BIC (or some other metric).

2.4. Rejecting over-complete models

A fourth SEM we can test is one in which the sequential model is augmented with an additional, direct connection from A to C (rendering it fully saturated), as in M4 shown in leftmost panel of Figure 2:

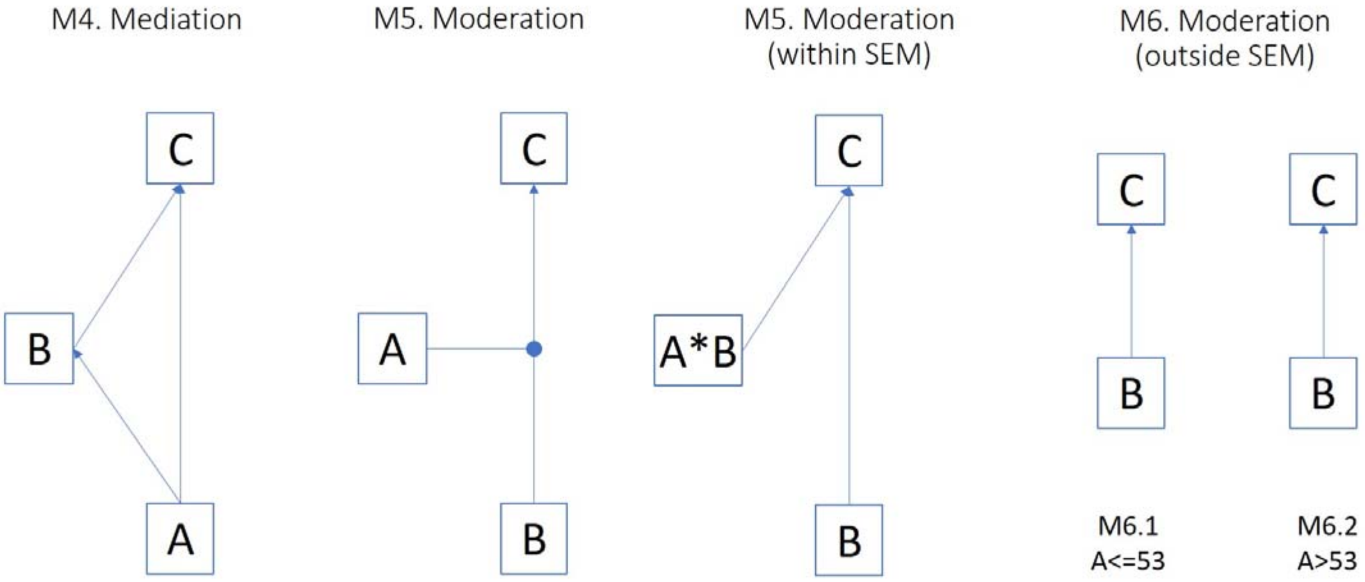


Figure 2. Mediation and Moderation models.

This is implemented by SEM4:

```
SEM4 <- 'C ~ B
         B ~ A
         C ~ A'
SEM4_df1 <- sem(SEM4, data=df1)
#anova(SEM4_df1)
anova(SEM1_df1, SEM4_df1)
```

```
##
## Chi-Squared Difference Test
##
##          Df    AIC    BIC  Chisq Chisq diff RMSEA Df diff Pr(>Chisq)
## SEM4_df1  0 149087 149124 0.0000
## SEM1_df1  1 149086 149114 0.0763    0.07633     0       1     0.7823
```

The Chi-square difference is too small to be significant (given the difference in 1 df), but the BIC is lower for SEM1 than SEM4, suggesting that the extra parameter is not needed, i.e, the more parsimonious SEM1 model is more likely to be the true model.

2.5. Simple (cross-sectional) mediation

While SEM4 above was worse when fitting the data generated by SEM1, we can generate data from SEM4 to illustrate the concept of Brain as a mediator of the dependence of Cognition on Age:

```
B <- A + rnorm(n = npt, mean = 0, sd = Bsd)
C <- B + A + rnorm(n = npt, mean = 0, sd = Csd) # Now direct contribution from A too
df4 <- data.frame(A, B, C)
SEM4 <- '
    C ~ b1 * B
    B ~ a1 * A
    C ~ c1 * A
    Total := (abs(a1*b1)) + (abs(c1))
    Mediation := abs(a1*b1)
    Proportion_mediated := Mediation / Total'
SEM4_df4 <- sem(SEM4, data=df4)
summary(SEM4_df4)
```

```
## lavaan 0.6.15 ended normally after 1 iteration
## 
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                         5
## 
##   Number of observations                         10000
## 
## Model Test User Model:
##                                                       
##   Test statistic                                 0.000
##   Degrees of freedom                                 0
## 
## Parameter Estimates:
## 
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
## 
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   C ~                                                 
##     B         (b1)    0.979    0.010   98.055    0.000
##   B ~                                                 
##     A         (a1)    0.990    0.010   99.009    0.000
##   C ~                                                 
##     A         (c1)    1.032    0.014   73.466    0.000
## 
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .C               100.811    1.426   70.711    0.000
##    .B               101.082    1.430   70.711    0.000
## 
## Defined Parameters:
##                    Estimate  Std.Err  z-value  P(>|z|)
##     Total             2.002    0.014  143.136    0.000
##     Mediation         0.969    0.014   69.670    0.000
##     Proportin_mdtd    0.484    0.006   80.059    0.000
```

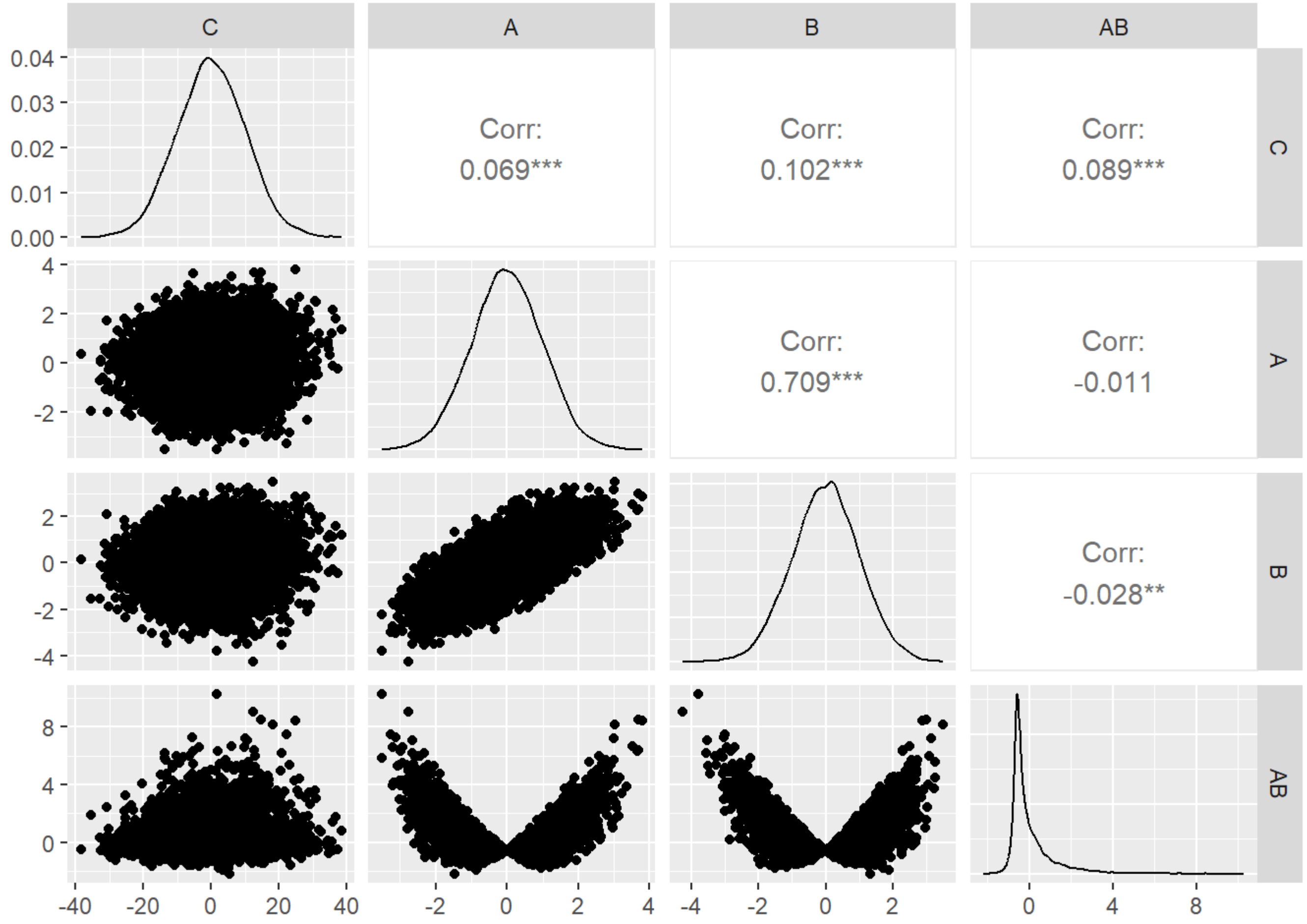
Here we just have named the paths in SEM4 (`a1, b1...` etc) in order to calculate the proportion of variance explained by the mediator (B), which is close to 50% (and highly significant). Thus we can say that Brain (partially) mediates the effect of Age on Cognition. However, note the dangers in inferring causal mediation of Age effects using cross-sectional data (Raz and Lindenberger, 2011 (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3160731/)); see future primer for an example on longitudinal data.

Note that mediation is sometimes tested with a series of GLMs, i.e., showing that `C~A` is significant, `B~A` is significant, and that B still has a significant effect in the (partial regression) model `C~B+A` , but it is more elegant to test within a single SEM.

2.6.1 Moderation within SEM

Age can also be a moderator, such that the effect of Brain on Cognition depends on Age (e.g, perhaps a stronger relationship in older people). This is sometimes depicted as one variable impacting on the path between two others, as in second panel of Figure 2. One way to test this within a single SEM is to add the A*B interaction term as a new variable that causes variance in C (as in third panel of Figure 2). First we will create some data in which an interaction is present. Note it is important to mean-correct A and B before multiplying them so that the interaction is less correlated with the main effects. It is often also advisable to scale them so that they have similar SD (i.e, Z-score), to ensure comparable ranges (see my Rmd on interactions (https://github.com/MRC-CBU/miscellaneous/tree/master/power-for-interactions)).

```
A   <- scale(df1$A)
#A  <- scale(as.numeric(df1$A>53)) # If want binary groups for age
B   <- scale(df1$B)
AB  <- scale(A*B)
C   <- B + AB + rnorm(n = npt, mean = 0, sd = Csd) # Now interaction term A*B
df5 <- data.frame(C, A, B, AB)
ggpairs(df5)
```



Note the interaction term `AB` is not Gaussian, since it is the (nonlinear) product of two distributions - but again this does not matter for SEM provided it is an exogeneous variable only. We can then test the SEM:

```
SEM5 <- 'C ~ B + AB'
SEM5_df5 <- sem(SEM5, data=df5)
summary(SEM5_df5, fit.measures = FALSE)
```

```
## lavaan 0.6.15 ended normally after 1 iteration
## 
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                         3
## 
##   Number of observations                         10000
## 
## Model Test User Model:
##                                                       
##   Test statistic                                 0.000
##   Degrees of freedom                                 0
## 
## Parameter Estimates:
## 
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
## 
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   C ~                                                 
##     B                 1.057    0.101   10.509    0.000
##     AB                0.937    0.101    9.309    0.000
## 
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .C               101.155    1.431   70.711    0.000
```

which confirms that the `AB` path on `C` is significant, as expected. Note that while we have talked about this indicating that Age moderates effect of Brain on Cognition, the interaction is symmetrical, so one could also say that Brain moderates the effect of Age on Cognition.

We can compare this interaction approach to an alternative way of modelling moderations, in which we fit separate SEMs for two or more levels of the moderator, using “multi-group” SEMs, as M6 shows graphically in Figure 2.

2.6.2 Moderation across SEMs

Let’s define two groups of participants - the young and the old - depending on whether their age is above 53 (so RH still counts as “young” ;-). This is coded by the new factor `Ag` below, which is passed to SEM via the additional “group” argument. This effectively fits separate SEMs for each group:

```
df5$Ag <- factor(df1$A>53)
levels(df5$Ag) = c("Young","Old")
SEM6.free <- 'C ~ B'
SEM6.free_df5 <- sem(SEM6.free, data=df5, group="Ag")
summary(SEM6.free_df5, fit.measures = FALSE)
```

```
## lavaan 0.6.15 ended normally after 44 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                         6
##
##   Number of observations per group:
##     Old                                           4981
##     Young                                         5019
##
## Model Test User Model:
##
##   Test statistic                                 0.000
##   Degrees of freedom                                 0
##   Test statistic for each group:
##     Old                                          0.000
##     Young                                        0.000
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
##
## Group 1 [Old]:
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   C ~
##     B                 1.930    0.177   10.881    0.000
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .C                -0.450    0.176   -2.558    0.011
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .C               102.437    2.053   49.905    0.000
##
##
## Group 2 [Young]:
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   C ~
##     B                 0.195    0.170    1.149    0.251
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .C                -0.404    0.171   -2.359    0.018
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .C               100.619    2.009   50.095    0.000
```

It can be seen that the path from B to C is significant in the Old group, but not the Young group, consistent with how the data were generated in terms of the effect of B on C being greater with higher values of A (together with an overall main effect of B). To formally test this interaction, we can compare the above SEM, in which the B-C path is freely estimated within each age group, to another SEM in which the parameter for that path is equated across age groups:

```
SEM6.eqtd <- 'C ~ c(b1,b1) * B'
SEM6.eqtd_df5 <- sem(SEM6.eqtd, data=df5, group="Ag")
#summary(SEM6.free_df5, fit.measures = FALSE)
anova(SEM6.free_df5, SEM6.eqtd_df5)
```

```
##
## Chi-Squared Difference Test
##
##               Df   AIC   BIC  Chisq Chisq diff    RMSEA Df diff Pr(>Chisq)
## SEM6.free_df5  0 74593 74637  0.000
## SEM6.eqtd_df5  1 74641 74677 49.735     49.735 0.098727       1   1.76e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

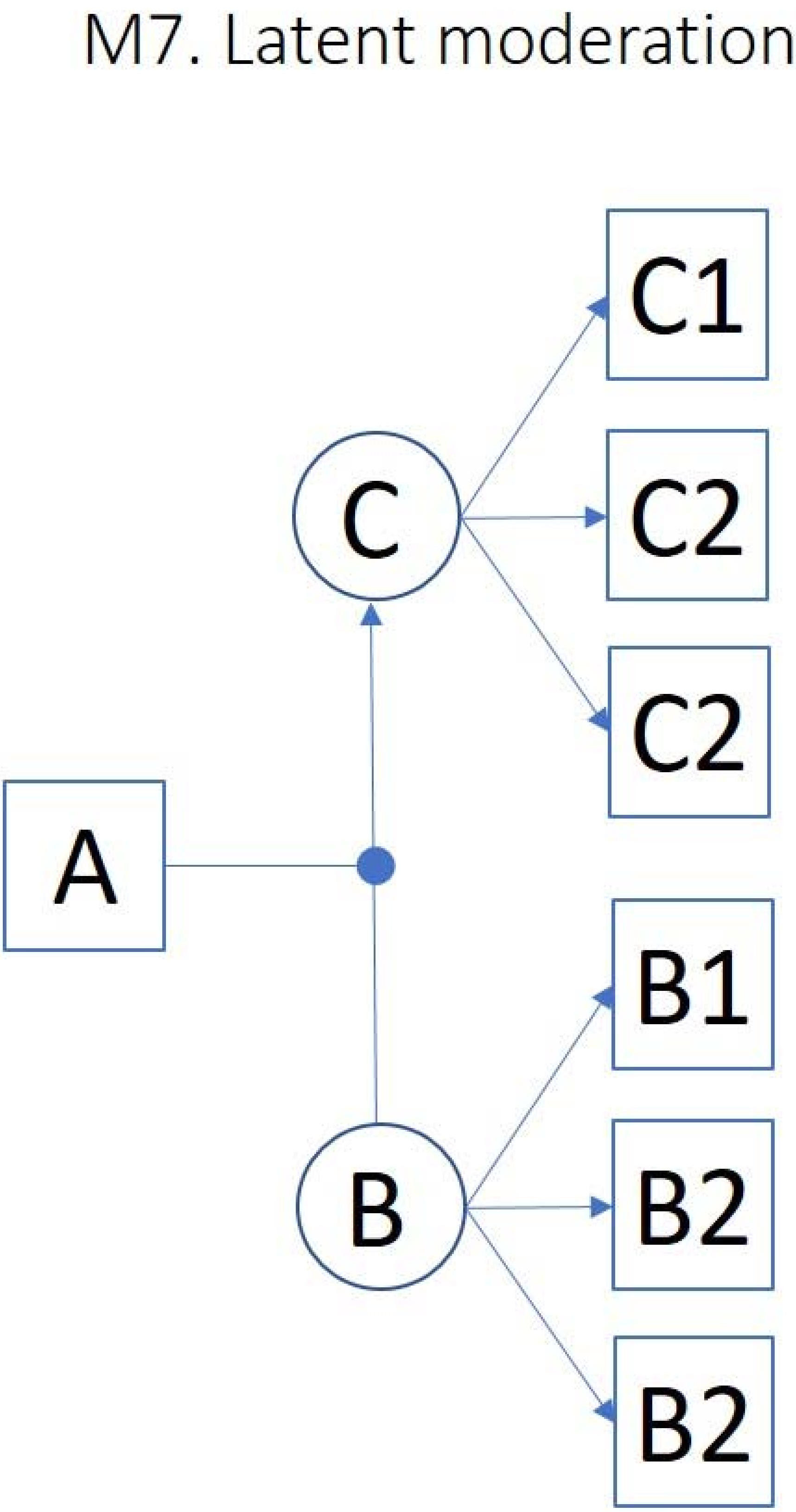
As expected, the model in which the B-C relationship was equated across age groups (`SEM6.eqtd`) is significantly worse, supporting the presence of a moderation of this relationship by A.

Note that the data were generated with a continuous linear interaction, whereas here we are modelling it in terms of binary age split, so the latter will not be as sensitive as the within-SEM model above, consistent with the equivalent Z-value for the interaction path when comparing `SEM6.free` with `SEM6.eqtd` (`sqrt(49.735)=7.052`), being less than the Z-value for the AB-C path in `SEM5` (`9.309`). Of course this would not be case if we generated the data with a binary moderation instead, in which case the two models would have equivalent sensitivity (which can be checked by uncommenting the line that generates the `A` variable for `df5` above, to make it binary; after re-running all subsequent code, you should find equivalent Z-scores for the moderation in both `SEM5` and comparison of `SEM6.free` with `SEM6.eqtd`). Alternatively, one could use a moving window of ages to trace out the B-C relationship as a smooth function of the (median) age within each window (see Robitzsch, 2023 (https://www.researchgate.net/publication/373623425_Estimating_Local_Structural_Equation_Models) and di Mooij et al. (https://doi.org/10.1523/JNEUROSCI.1627-17.2018) for an example).

2.7 Latent Variables

So far, we have treated A, B and C as single measured variables. Often we might have multiple measures of Brain, Cognition, etc, and wish to summarise them in terms of a latent factor (cf. factor analysis). This brings the real power of SEM in combining definition of latent variables (indicated in lavaan by `~=`) and regressions between variables (with `~` , as in examples above).

Let’s assume we have three measurements of B and C, called B1, B2, B3, and C1, C2, C3 respectively, as shown in Figure 3.



We can generate some data via the moderation model above and also generate the measured variables:

```
A   <- scale(as.numeric(df1$A>53)) # Now binary groups for age
B   <- scale(df1$B)
AB  <- scale(A*B)
C   <- B + AB + rnorm(n = npt, mean = 0, sd = Csd)

Msd <- 1 # measurement noise

C1 <- C + rnorm(n = npt, mean = 0, sd = Msd)
C2 <- C + rnorm(n = npt, mean = 0, sd = Msd)
C3 <- C + rnorm(n = npt, mean = 0, sd = Msd)

B1 <- B + rnorm(n = npt, mean = 0, sd = Msd)
B2 <- B + rnorm(n = npt, mean = 0, sd = Msd)
B3 <- B + rnorm(n = npt, mean = 0, sd = Msd)

Ag  <- factor(A); levels(Ag) = c("Young","Old")

df6 <- data.frame(Ag, C1, C2, C3, B1, B2, B3)
#ggpairs(df6)
```

We can now define a new SEM consisting of two parts: the “measurement model”, in which new latent factors `Cf` and `Bf` are defined from their respective measurements, and the “structural model”, in which these factors are related via regression (as in examples above):

```
SEM7.free <-  '
# measurement model
    Cf =~ C1 + C2 + C3
    Bf =~ B1 + B2 + B3
# structural model
    Cf  ~ Bf'
SEM7.free_df6 <- sem(SEM7.free, data=df6, group="Ag")
#summary(SEM7.free_df6, fit.measures = FALSE)
```

Using multi-group SEM, we could ask whether the relationship between Brain and Cognition varies with age (as in examples above), but we could also ask whether the factors themselves vary with age, by comparing the above model with one in which the loadings of each factor on their measurements are equated across groups:

```
SEM7.eqtd <-  '
# measurement model
    Cf =~ c(c1,c1) * C1 + c(c2,c2) * C2 + c(c3,c3) * C3
    Bf =~ c(b1,b1) * B1 + c(b2,b2) * B2 + c(b3,b3) * B3
# structural model
    Cf  ~ Bf'
SEM7.eqtd_df6 <- sem(SEM7.eqtd, data=df6, group="Ag")
#summary(SEM7.eqtd_df6, fit.measures = FALSE)
anova(SEM7.eqtd_df6,SEM7.free_df6)
```

```
## 
## Chi-Squared Difference Test
## 
##               Df    AIC    BIC  Chisq Chisq diff RMSEA Df diff Pr(>Chisq)
## SEM7.free_df6 16 238080 238354 12.207                                    
## SEM7.eqtd_df6 20 238075 238320 15.410     3.2039     0       4     0.5243
```

The decrease in Chi-square for the more complex model (SEM7.free) is not significant, given its extra 4 dfs (parameters), and the AIC and BIC are lower for the equated model in which the factor loadings are fixed across age groups. This “measurement invariance” is normally good news, because it suggests that we can use the same latent factors across age groups.

Finally, note that we could equally define Age as a latent factor, itself estimated from multiple measures, such as year of birth, telomere length, number of wrinkles, etc. Such estimates of “biological age” might be more accurate than simply the number of revolutions around the sun.

Longitudinal measures and Age

A future primer that hope to share soon…!