THE UNIVERSITY of EDINBURGH

Path Analysis

Data Analysis for Psychology in R 3

dapR3 Team

Department of Psychology
The University of Edinburgh

Weeks 7 to 11 Overview

- Section introduction (w7)
- Path analysis (w7)
- Path mediation (w8)
- Data Reduction:
  - Principal Components Analysis (w9)
  - Exploratory Factor Analysis (w10 & 11)
- Where next? (w11)

Learning Objectives

1. Understand the core principle of path modelling
2. Be able to use `lavaan` to estimate linear models and simple path models
3. Interpret the output from a `lavaan` model.

Part 1: Introduction and Motivation

Part 2: Introducing *lavaan*

Part 3: Model Specification & Estimation

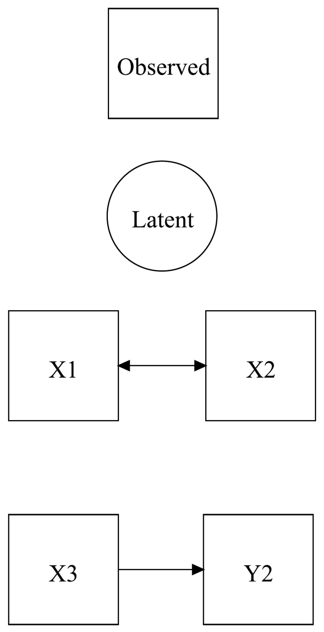
Part 4: Model Evaluation

What issue do these methods solve?

- Path models
    - Sometimes we have more than one variable that needs to be treated as an outcome/dependent variable
    - We cant do this in a linear model.
    - **A path model allows us to test several linear models together as a set**
    - Good way to learn basics of *structural equation modelling*
- Data reduction
    - Psychology uses many surveys and psychometric tools
    - Here we asked lots of questions we believe relate to some construct
    - We need a way to:
    - Check the relationships between each question
    - Produce plausible scores that represent this construct
- We will start with path models...

Diagrammatic Conventions

- Some conventions
  - Square = observed/measured
  - Circle/ellipse = latent/unobserved
  - Two-headed arrow = covariance
  - Single headed arrow = regression path



Terminology

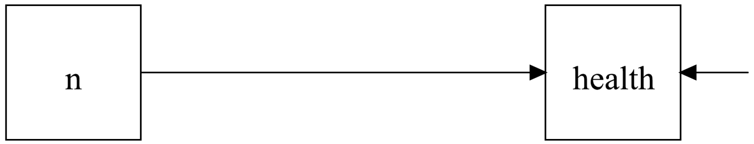
- A couple of distinctions are also useful here.
- Broadly, variables can be categorised as either exogenous or endogenous.
- **Exogenous:** are essentially independent variables.
  - Only have directed arrows going out.
- **Endogenous:** are dependent variables in at least one part of the model.
  - They have directed arrows going in.
  - In a linear model there is only one endogenous variable, but in a path model we can have multiple.
  - They also have an associated residual variance.
    - Just like in a `lm`
    - If something predicts (explains variance) a variable, there will be something left unexplained
- So how does this relate to practical research problem?

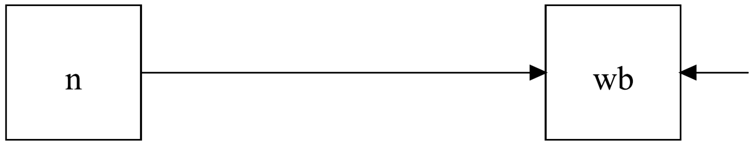
A hypothetical example

- Suppose we are interested in how Neuroticism predicts psychological well-being and physical health outcomes.
    - Neuroticism measured by a questionnaire with 5 items (5-point scale).
    - Well-being is measured by a questionnaire with 5 items (7-point scale).
    - Physical health is measured based on BMI, V02 max, and presence or absence of cancer (binary).
- How do we test our model?

A hypothetical example

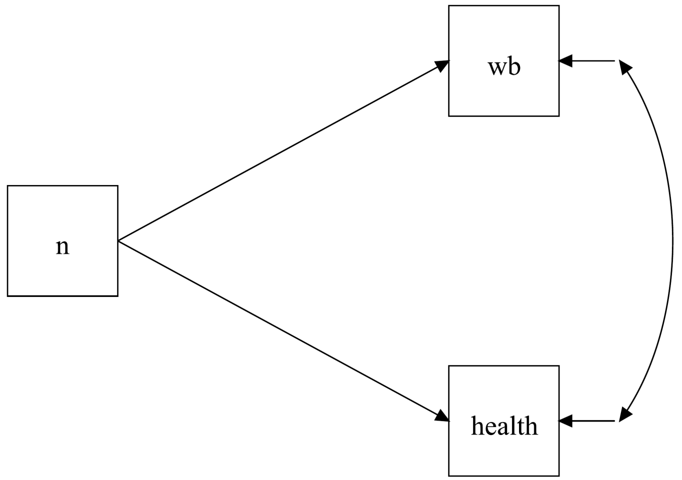
- Approach 1: Aggregate everything into composite scores and use 2 regression models.





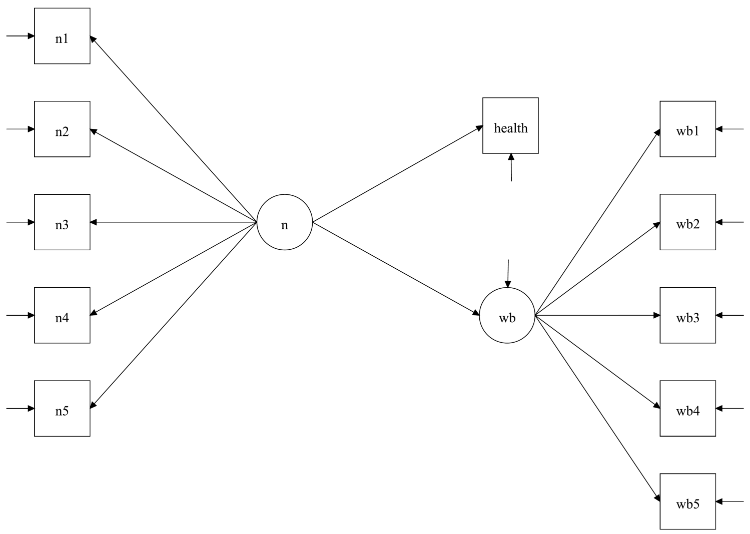
A hypothetical example

- Approach 2: Aggregate everything, and use a path model to simultaneously estimate model with 2 outcomes.



A hypothetical example

- Approach 3: Use a mix of latent and composite variables to simultaneously estimate model with 2 outcomes.



End of Part 1

Part 1: Introduction and Motivation

Part 2: Introducing *lavaan*

Part 3: Specification & Estimation

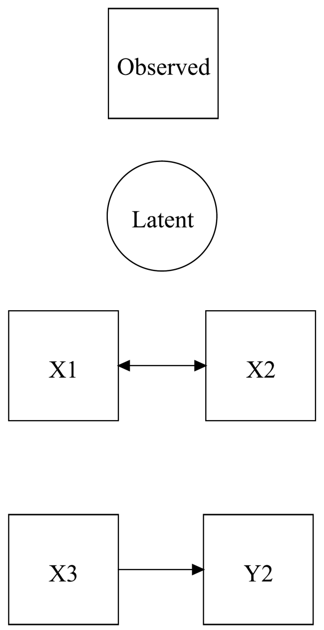
Part 4: Model Evaluation

`lavaan`

- The package we will use to fit our path models is called `lavaan`.
- Using `lavaan` requires us to write...
    - code to specify our model
    - code to run the model
- This is because we (a) generally use path models for slightly bigger models, and (b) have lots more options when running our model than in a `lm`

Model statements in `lavaan`

- When a variable is observed, we use the name it has in our data set.
    - Here `X1`, `X2`, `X3`, `Y2`
- When a variable is latent (more on this in the data reduction section) we give it a new name.
    - Here `Latent`
- To specify a covariance, we use `~~`
- To specify a regression path, we use `~`



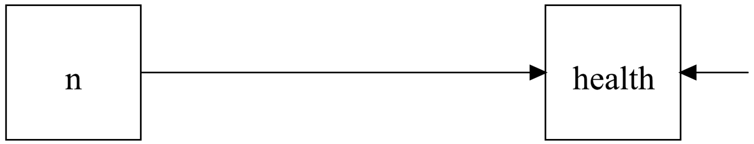
`lavaan` model code: Approach 1

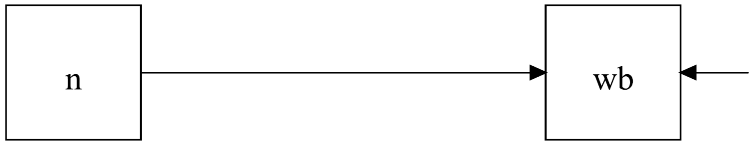
- Approach 1: Aggregate everything into composite scores and use 2 regression models.

```
a1a = '
wb ~ n
'
```

```
a1b = '
health ~ n
'
```

- But this gains us nothing

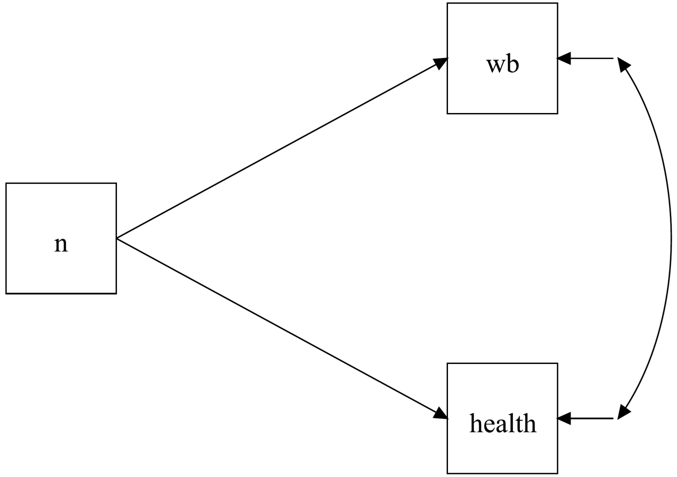




`lavaan` model code: Approach 2

- Approach 2: Aggregate everything, and use a path model to simultaneously estimate model with 2 outcomes.

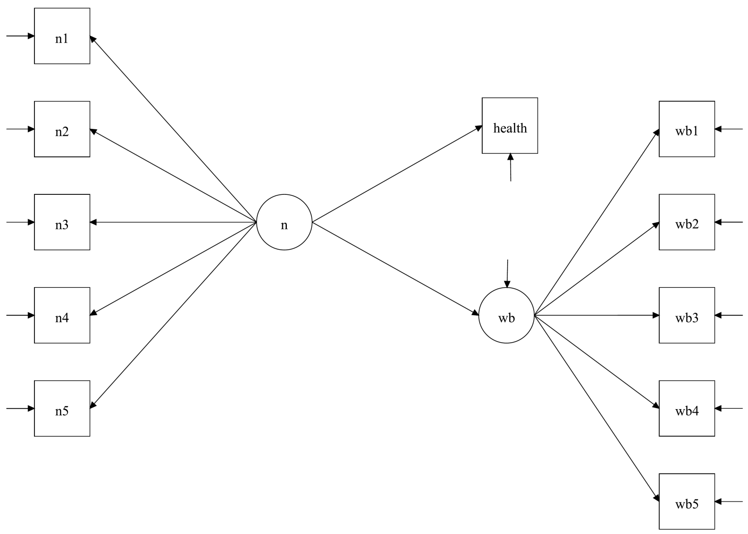
```
a2 = '
wb ~ n
health ~ n
wb ~~ health
'
```



`lavaan` model code: Approach 3

- **Approach 3**: Use a mix of latent and composite variables to simultaneously estimate model with 2 outcomes.
    - This one is just for fun (we wont get this far on this course)
    - I will include some examples of this in the final week for anyone interested

```
a3 <- '
n =~ n1 + n2 + n3 + n4 + n5
wb =~ wb1 + wb2 + wb3 + wb4 + wb5
health ~ n
wb ~ n
'
```



Running a `lavaan` model

- Once we have our model statement, we then need to run our model.
  - There are a number of functions to do this, we will only use `sem()`

```
library(lavaan)
m1 <- sem(model, # your model statement
          orderd = c(), # if variables are ordered categories list them
          estimator ="ml" , # name of the estimation method you wish to use
          missing = "" , # name of the missing data method you wish to use
          data = tbl) # your data set
```

- `lavaan` has sensible defaults, meaning most of the time you will only need to state you model and data.

- There is **lots** of information on using `lavaan` with lots of examples on-line

Viewing the results

- Lastly, we need to use a `summary` function (like in `lm` and `glm`) to see results.

```
summary(m1, # name given to our results object
        fit.measures = T, # model fit information
        standardized = T # provides standardized coefficients
        )
```

End of Part 2

Part 1: Introduction and Motivation

Part 2: Introducing *lavaan*

Part 3: Model Specification & Estimation

Part 4: Model Evaluation

Stages in path model

- Specification:
    - This is what we have just seen in our motivating examples.
    - Specification concerns which variables relate to which others, and in what ways.

- We have seen the types of path we can include, but there are some other standard "rules"

1. All exogenous variables correlate
2. For endogenous variables, we correlate the residuals, not the variables.
3. Endogenous variable residuals **do not correlate** with exogenous variables.
4. All paths are recursive (i.e. we cant not have loops like A->B, B->A).

Model identification

- Identification concerns the number of **knowns** versus **unknowns**
- There must be more knowns than unknowns in order to test our model.
- The knowns are variances and covariances of the observed variables.
- The unknowns are the parameters we want to estimate.
- **Degrees of freedom** are the difference between knowns and unknowns

Levels of identification

- There are three levels of identification:
  - **Under-identified** models: have < 0 degrees of freedom
  - **Just Identified** models: have 0 degrees of freedom (all standard linear models are just identified)
  - **Over-Identified** models: have > 0 degrees of freedom

Model identification illustration

- Chou & Bentler (1995) provide an illustration based on simultaneous linear equations:
  - Eq.1: $x + y = 5$
  - Eq.2: $2x + y = 8$
  - Eq.3: $x + 2y = 9$
- Eq.1 is on its own is *under-identified*
- Eq.1 & 2 are together *just identified*
- Eq.1, 2 & 3 are together *over identified*

Model estimation

- After we have specified our model (& checked it is identified) we proceed to **estimation**
- Model estimation refers to finding the 'best' values for the unknown parameters

Maximum likelihood estimation

- Maximum likelihood estimation is most commonly used
- Finds the parameters that maximise the likelihood of the data
- Begins with a set of starting values
- Iterative process of improving these values
  - i.e. to minimise the difference between the sample covariance matrix and the covariance matrix implied by the parameter values
- Terminates when the values are no longer substantially improved across iterations
  - At this point **convergence** is said to have been reached

Maximum likelihood estimation assumptions

- Large sample size
- Multivariate normality
- Variables are on a continuous scale
- If we believe these are not met, there are alternatives:
    - Robust maximum likelihood estimation
    - For non-normal data
    - Weighted least squares, unweighted least squares or diagonally weighted least squares
    - For ordinal data
- Estimation is quite a complex topic, for now, working with ML will suffice.

No convergence?

- Sometimes estimation fails
- Common reasons are:
  - The model is not identified
  - The model is very mis-specified
  - The model is very complex so more iterations are needed than the program default

From path models to model evaluation

- Our path models are based on covariances or correlations between our measured variables.
  - Typically what we would call our observed correlation/covariance.
- When we specify a model, we can work out the correlations from paths in our model.
  - This is referred to as a model implied correlation/covariance.
  - This process is called path tracing (see lab)
- If our model contains less paths than we have correlations, then we have produced a model that is a simplified version of our data.
  - Knowing if this simplified model well reproduces our data is at the core of model evaluation.

End of Part 3

Part 1: Introduction and Motivation

Part 2: Introducing *lavaan*

Part 3: Model Specification & Estimation

Part 4: Model Evaluation

Model Evaluation (Fit)

- In path models and it's extensions, we tend not to focus on the variance explained in the outcome (though we can calculate this)

- Instead, we tend to think about:

  1. Does our model fit the data?
  2. If it fit's the data, what are the parameter estimates?

- "Fitting the data" refers to the comparison between the observed and the model implied covariance matrices.

  - If our model reproduces the observed covariances well, then it is deemed to fit.
  - If our model reproduces the observed covariances poorly, then it is deemed to not fit (and we wouldn't interpret the model)

Model fit

- Just-identified models will always fit perfectly.
    - They exactly reproduce the observed covariances.
- When we have positive degrees of freedom, we can calculate a variety of model fit indices.
    - We have seen some of these before (AIC and BIC)
    - But there are a huge number of model fit indices.
- For ease, we will note a small number, and focus on the suggested values that indicate good vs bad fit.
    - This will give an impression of certainty in the fit vs non-fit decision.
    - But be aware this is not a binary choice.
    - Model fit is a continuum and the use of fit indices much debated.

Global fit

- χ^2
  - When we use maximum likelihood estimation we obtain a χ^2 value for the model
  - This can be compared to a χ^2 distribution with degrees of freedom equal to our model degrees of freedom
  - Statistically significant χ^2 suggests the model does not do a good job of reproducing the observed variance-covariance matrix
- However, χ^2 does not work well in practice
  - Leads to the rejection of models that are only trivially mis-specified

Alternatives to χ^2

- Absolute fit
  - Standardised root mean square residual (**SRMR**)
  - measures the discrepancy between the observed correlation matrix and model-implied correlation matrix
  - ranges from 0 to 1 with 0=perfect fit
  - values <.05 considered good
- Parsimony-corrected
  - Corrects for the complexity of the model
  - Adds a penalty for having more degrees of freedom
  - Root mean square square error of approximation (**RMSEA**)
  - 0=perfect fit
  - values <.05 considered good

Incremental fit indices

- Compares the model to a more restricted baseline model
  - Usually an 'independence' model where all observed variable covariances fixed to 0
- Comparative fit index (**CFI**)
  - ranges between 0 and 1 with 1=perfect fit
  - values > .95 considered good
- Tucker-Lewis index (**TLI**)
  - includes a parsimony penalty
  - values >.95 considered good

Local Fit

- It is also possible to examine local areas of mis-fit
- Modification indices estimate the improvement in χ^2 that could be expected from including an additional parameter
- Expected parameter changes estimate the value of the parameter were it to be included

Making model modifications

- Modification indices and expected parameter changes can be helpful for identifying how to improve the model.
    - These can be extracted using the `summary(model, mod.indices=T)`
    - They indicate the amount the model fit would improve if you added a path to your model
- However:
    - Modifications should be made iteratively
    - May just be capitalising on chance
    - Make sure the modifications can be justified on substantive grounds
    - Be aware that this becomes an exploratory modelling practice
    - Ideally replicate the new model in an independent sample
- As a general rule for dapR3 course, we want you to specify and test a specific model, and not seek to use exploratory modifications.

End