Review Session

Measurement & Evaluation of HCC Systems

Review Session

Today's goal:

Review the topics we have discussed in this course

Outline:

- path models
- pscyhometrics
- CFA and SEM
- EFA
- Advanced SEM

Path models

Summary

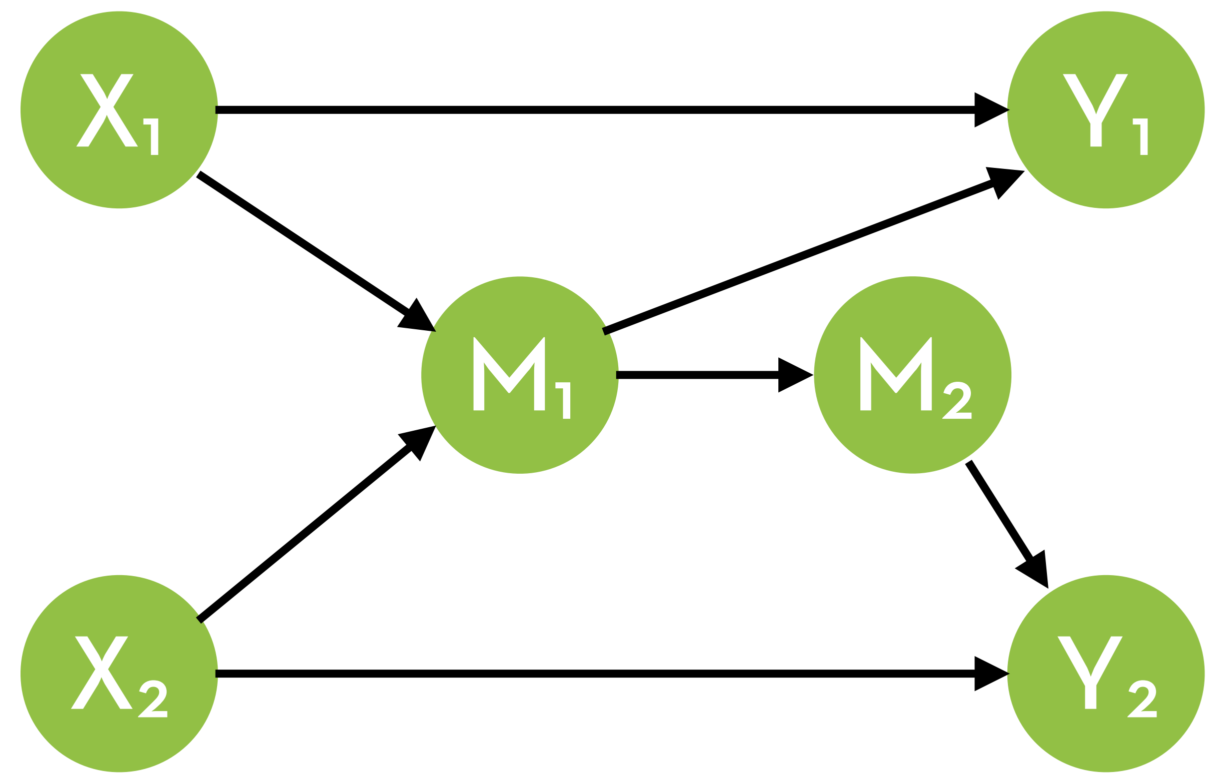
Path models

In regression, you have created models with one Y and several Xes

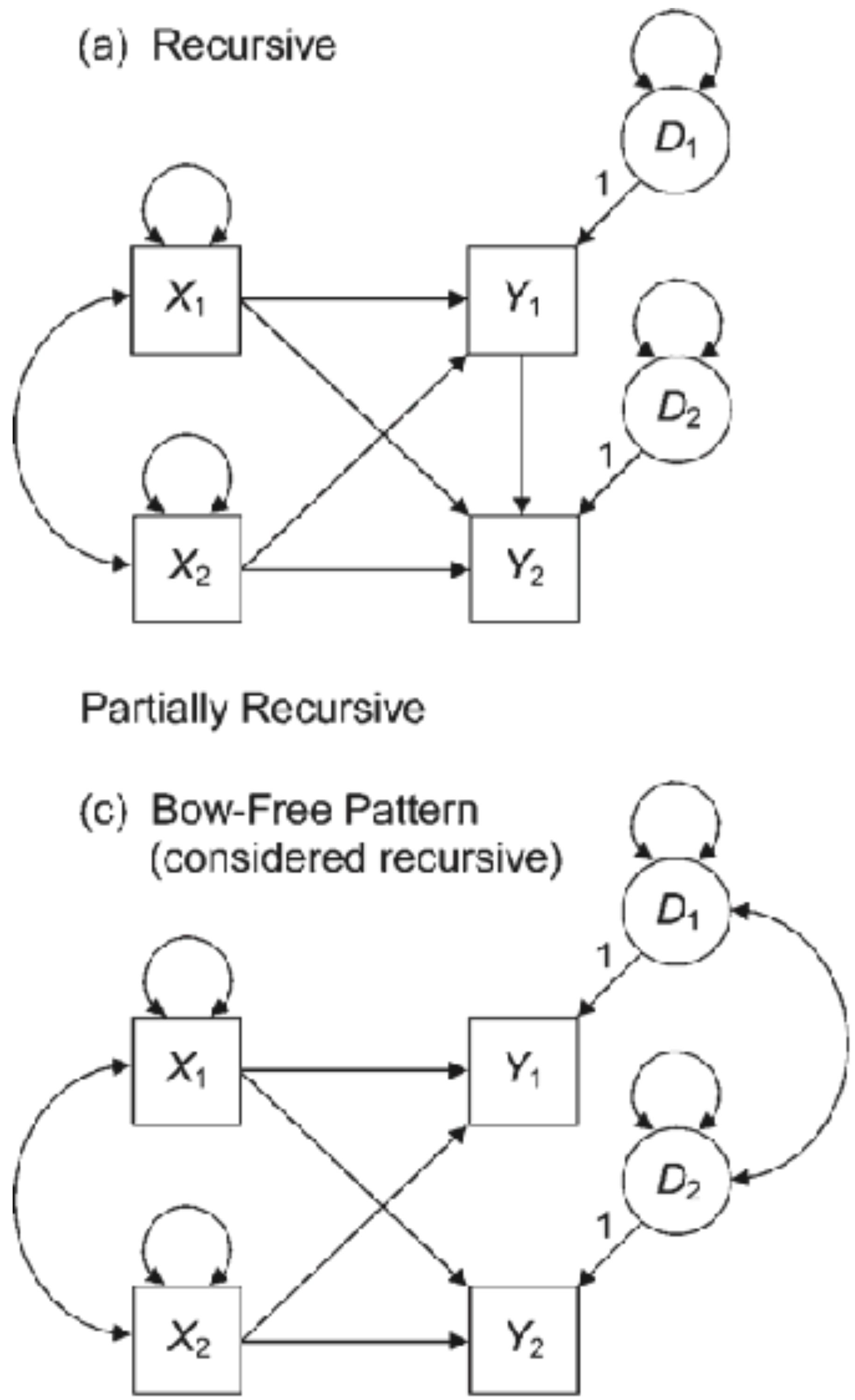
- In M&E I we talked about selecting suitable Xes

In path models, you can have many interconnected Xes and Ys

- Models can get very complicated



Types of models



Identification

Observations: number of covariances

- $v(v+1)/2$

Parameters: number of arrows

- Include variances, and implied correlations between Xs!

Degrees of freedom = observations – parameters

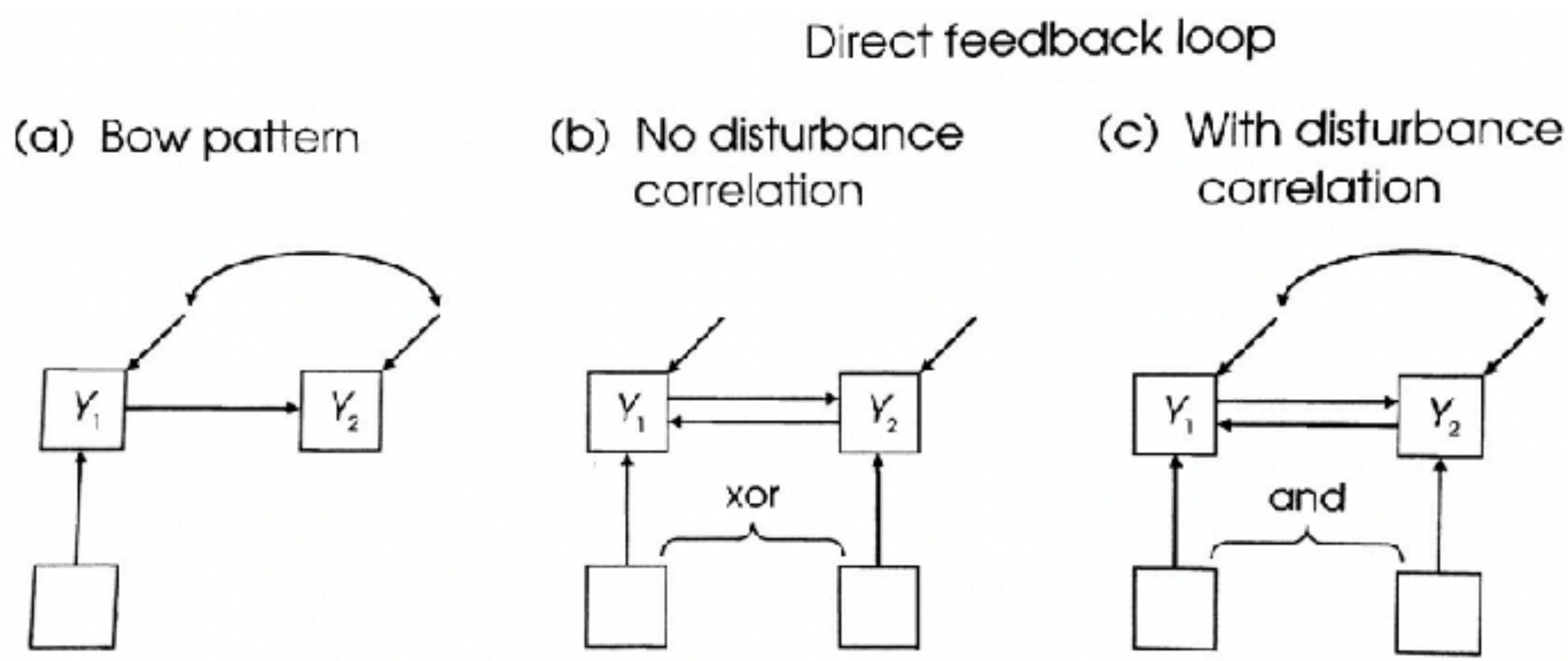
- > 0: over-identified, 0: just identified, < 0: under-identified
- Over-identified models are good

Note: Recursive models are always identified

Non-recursive

The following non-recursive models can still be identified if:

- Neither of the Ys causes the instrument X
- No other variable (indirectly) causes both the instrument X and the other Y
- The instrument X can be a correlation instead of a cause



Model fit

Chi-square test: whether there is significant misfit

- Does not mean the misfit is severe! Check chi-square / df

Baseline test: is the model better than the worst possible?

Relative fit: CFI (should be > 0.96), TLI (should be > 0.95)

Approximate fit: RMSEA (should be < 0.05)

- Should have a confidence interval below 0.10
- If higher than 0.05, check if it is significantly higher: p(RMSEA <= 0.05)

In practice

Establish causal order

Start from a saturated model

Stepwise trimming

- Start with least significant and non-hypothesized effects
- Remove interactions before main effects
- Remove all dummies of a manipulation at the same time

In practice

To test multiple experimental conditions, use lavTestWald

- Tests whether all conditions are equal

If significant (or not):

- The control manipulation has a (no) significant effect on understandability
- There's a (no) difference in understandability between users in the item control, friend control, and no control conditions

Reporting

Linear effects: Controlling for [other variables], a 1-point increase in [X] is associated with a x.xxx-point increase/decrease in [Y] (SE = x.xxx, z = x.xxx, p = .xxx).

Dummies: Controlling for [other variables], participants in the [X] condition are predicted to have a x.xxx higher/lower [Y] than participants in the [baseline] condition (SE = x.xxx, z = x.xxx, p = .xxx).

Reporting

Interaction effects: cgraph, citem, cfriend, cig, cfg:

cgraph is the difference between graph view and list view in the no control condition

citem is the diff. betw. item and no control in the list view condition

cfriend is the diff. betw. friend and no control in the list view condition

cig is the additional diff. betw. graph and list view in the item control condition (on top of cgraph) OR (and also) the additional diff. betw. item and no control in the graph view condition (on top of citem)

cfg is the additional diff. betw. graph and list view in the friend control condition (on top of cgraph) OR (and also) the additional diff. betw. friend and no control in the graph view condition (on top of cfriend)

Psychometrics

Summary

Reliability measures

Cronbach's Alpha uses the covariance matrix between items:

alpha = average(Cov) / average(Cov & Var)

Standardized alpha uses the average correlation r:

alpha = kr / (1+(k-1)r), where k is the number of variables

	A	B	C	D
A	Var_A	$Cov_{A,B}$	$Cov_{A,C}$	$Cov_{A,D}$
B	$Cov_{A,B}$	Var_A	$Cov_{B,C}$	$Cov_{B,D}$
C	$Cov_{A,C}$	$Cov_{B,C}$	Var_A	$Cov_{C,D}$
D	$Cov_{A,D}$	$Cov_{B,D}$	$Cov_{C,D}$	Var_A

Alpha in R

Use alpha in package "psych":

alpha(d[,c("v1","v2","v3", ...)], check.keys=T)

Output includes:

- Regular and standardized Alpha
- average correlation between items
- The values of these metrics **if each item is dropped**
- Several types of correlations of each item with the scale

Creating scales

Finding existing scales:

- In related work (especially if they tested them)
- inn.theorizeit.org

Create new scales:

- Adapt existing scales to your purpose
- Develop a brand new scale

Creating scales

1. Create a concept definition
2. Generate items (10-15)
3. Determine the response format (5-point? 7-point?)
4. Pre-Test the items (e.g. card sorting; end up with 7-10)
5. Include validation items
6. Administer the scale to a development sample
7. Evaluate the items (see CFA)
8. Optimize scale length (final scale 3-7 items)

CFA

Summary

Factor Analysis

Factors are **latent constructs** that represent the trait or concept to be measured

- The latent construct cannot be measured directly

The latent construct "**causes**" users' answers to items

- Items are therefore also called **indicators**

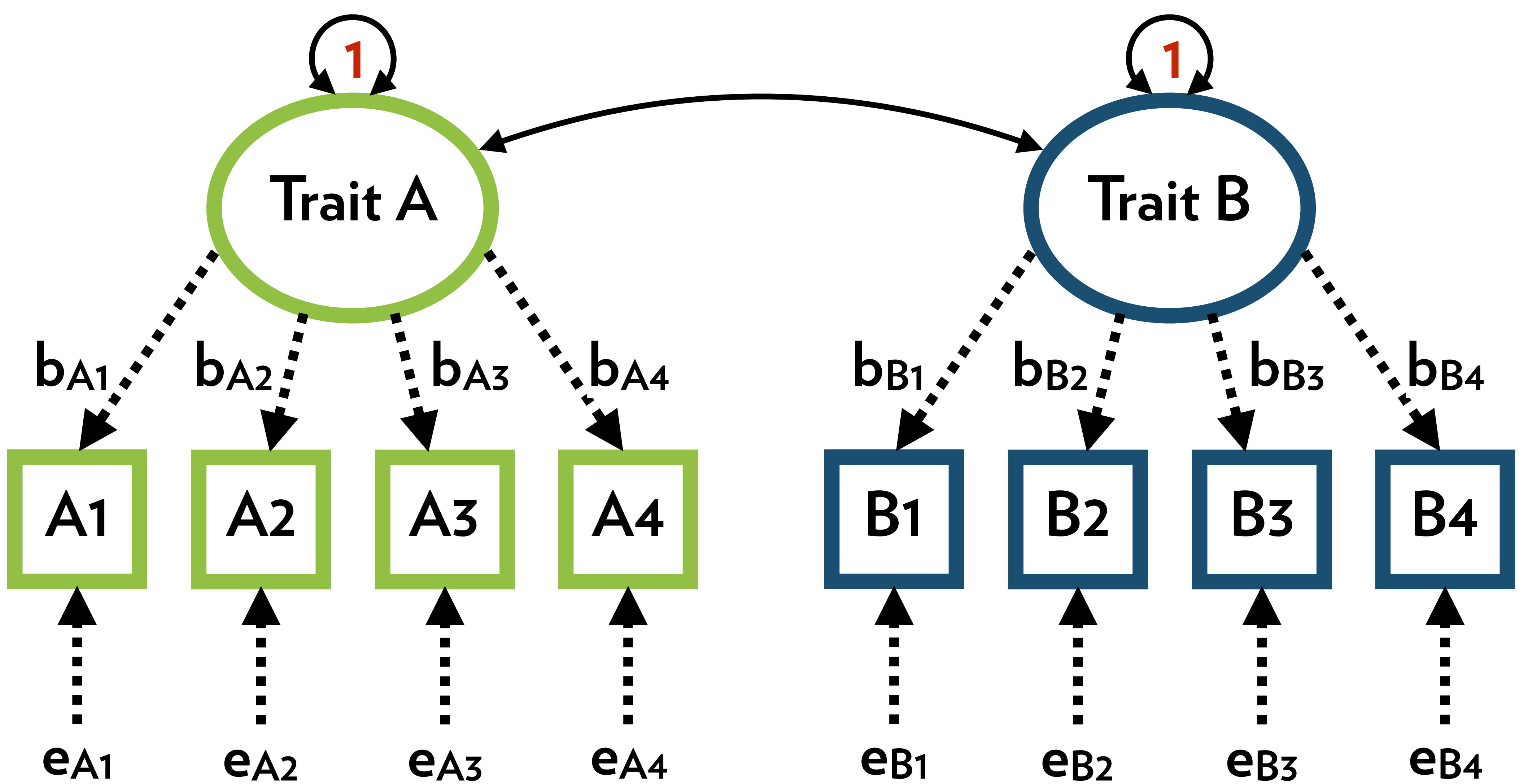
Like any measurement, indicators are not perfect measurements

- They depend on the true score (loading) as well as some measurement error (uniqueness)

Identification

Factors need to be **scaled** in order to be identified

Unit Variance Identification (UVI)



CFA in R

Write a model (e.g. F1 with items ABC, F2 with items DEF):

```
model <- '
F1 =~ A+B+C
F2 =~ D+E+F
'
```

Run the model:

```
fit <- cfa(model, data=d, ordered = c("A", "B", "C",
"D", "E", "F"), std.lv=T)
```

Get the results:

```
summary(fit, rsquare=T, fit.measures=T)
```

Improve

Item R^2 should be > 0.50 (although some argue 0.40 or even 0.30 is okay)

- If not: remove item

Modification indices should be low (relative)

- If not: respecify or remove item
- Anything < 8 is no problem, anything < 20 is often fine too

Remove until everything fits well

- But make sure at least 3 items per factor are left

Removing items...

What about epc (expected parameter change)?

This depends on the "decision"

- If for a high mi the decision is **(m)** then you can remove it regardless
- If for a high mi the decision is **epc** then you may have an overpowered modification index
- In that case, check epc to see if the parameter value is substantial

Removing items...

Always check the wording of the item before removing it

Try to come up with a reason why the item doesn't fit

Metrics

Factor fit:

- Convergent validity: AVE (average R^2) should be > 0.50
- Discriminant validity: √AVE should be higher than the highest correlation with any other factor

Model fit:

- See path models
- As CFA models are often (very) over-identified, perfect fit is often not attained —> Check the approx. fit metrics!

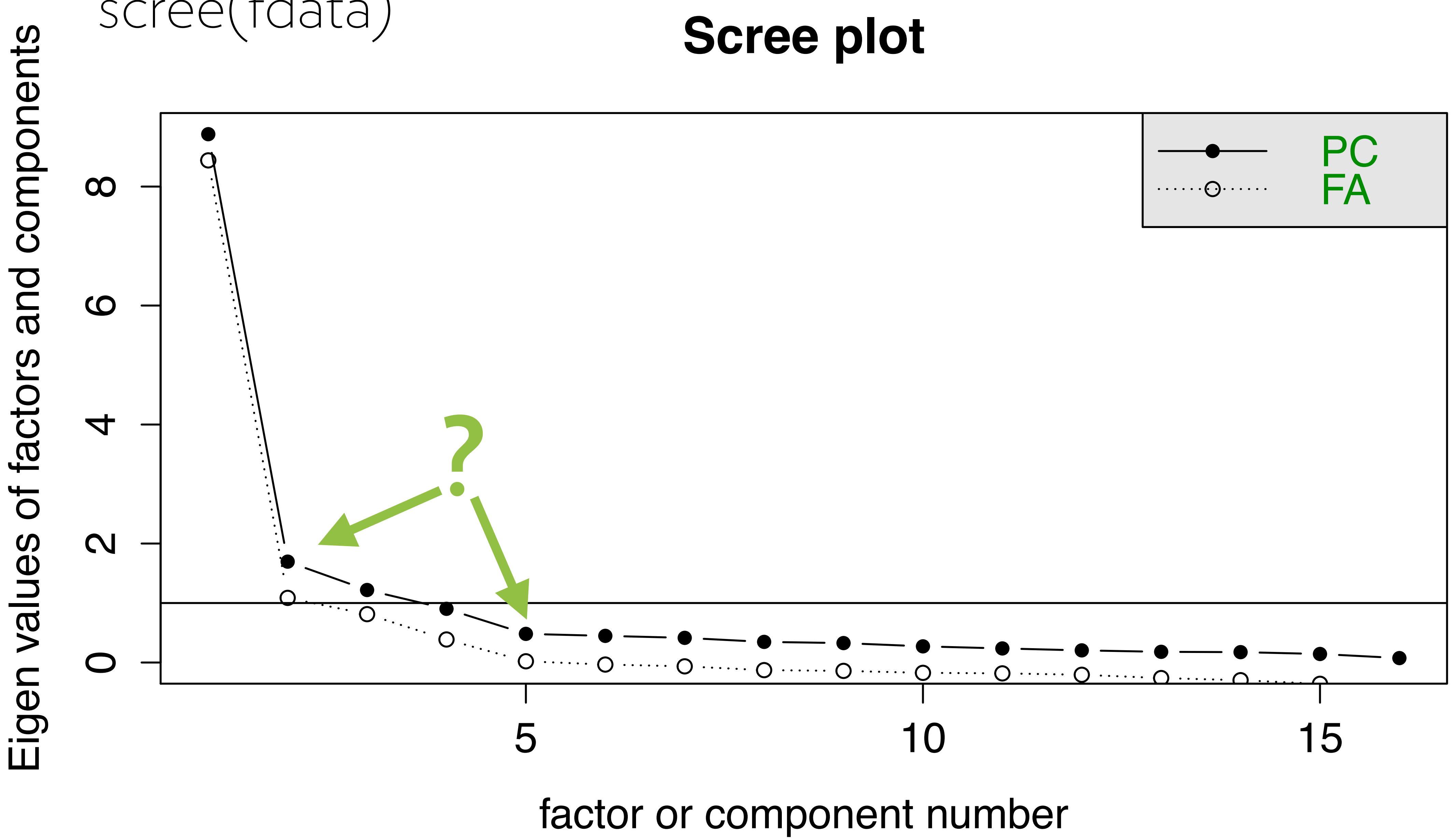
EFA

Summary

Number of factors

Get the number of factors: use "scree" in the psych package:

scree(fdata)



Find # of factors

Check cross-loadings (factor complexity)

Check uniquenesses

Check Chi-square fit, TLI, and RMSEA

- If Chi-square is not significant: “perfect” fit!

Compare against next model

- If not significant, then this model is sufficient
- If it is significant, this model could still be better (simpler)

Model comparison

f4$STATISTIC

chi-square (misfit) of the simpler (4-factor) model

f5$STATISTIC

chi-square (misfit) of the more complex (5-factor) model

f4$STATISTIC-f5$STATISTIC

difference in fit (how much worse is f4?)

f4$dof-f5$dof

additional degrees of freedom in f4

Model comparison

1-pchisq(f4$STATISTIC-f5$STATISTIC,f4$dof-f5$dof)

- χ^2(f4$dof-f5$dof) = f4$STATISTIC-f5$STATISTIC
- p = the outcome of 1-pchisq

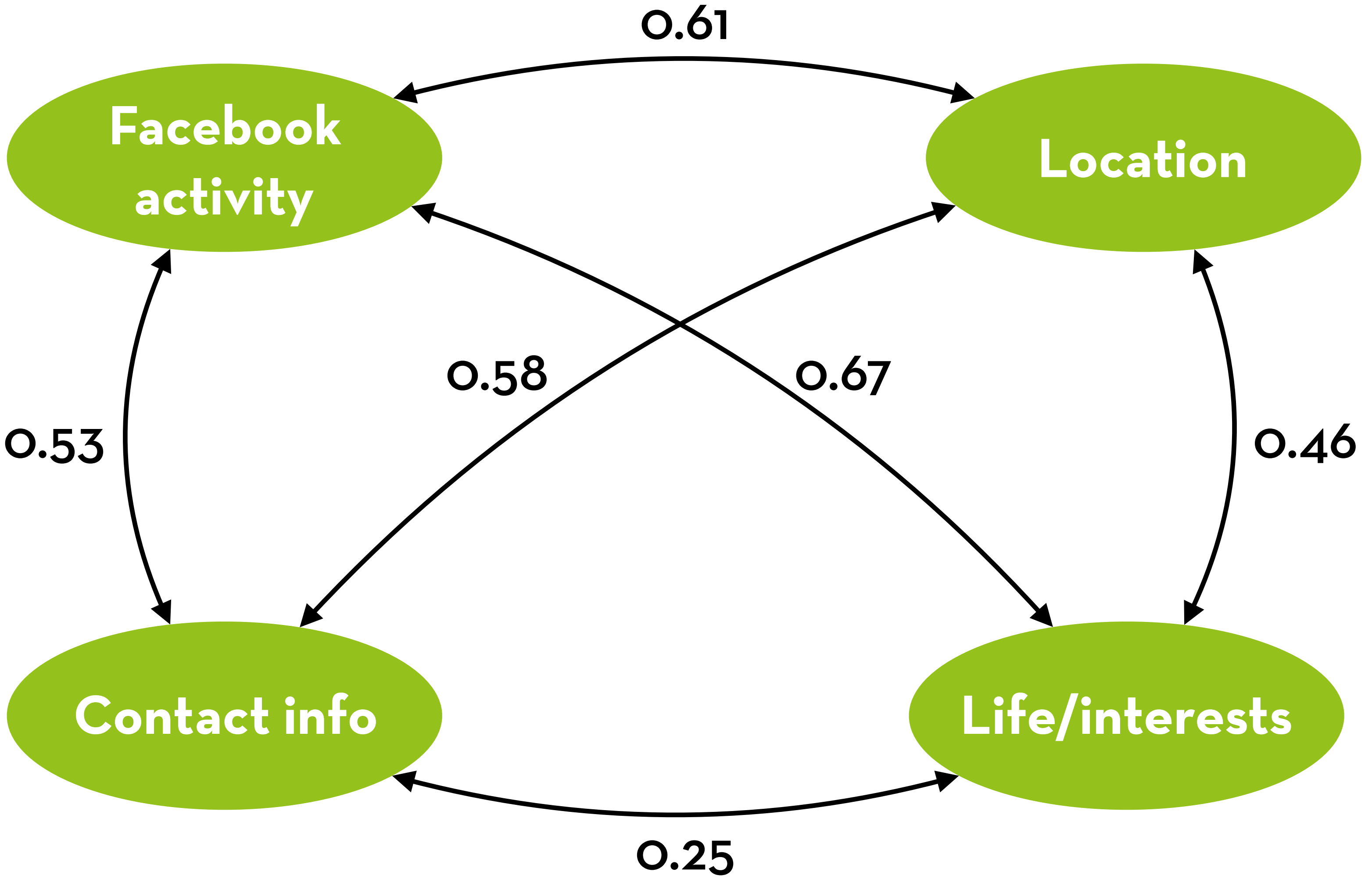
Is f4 significantly worse than f5?

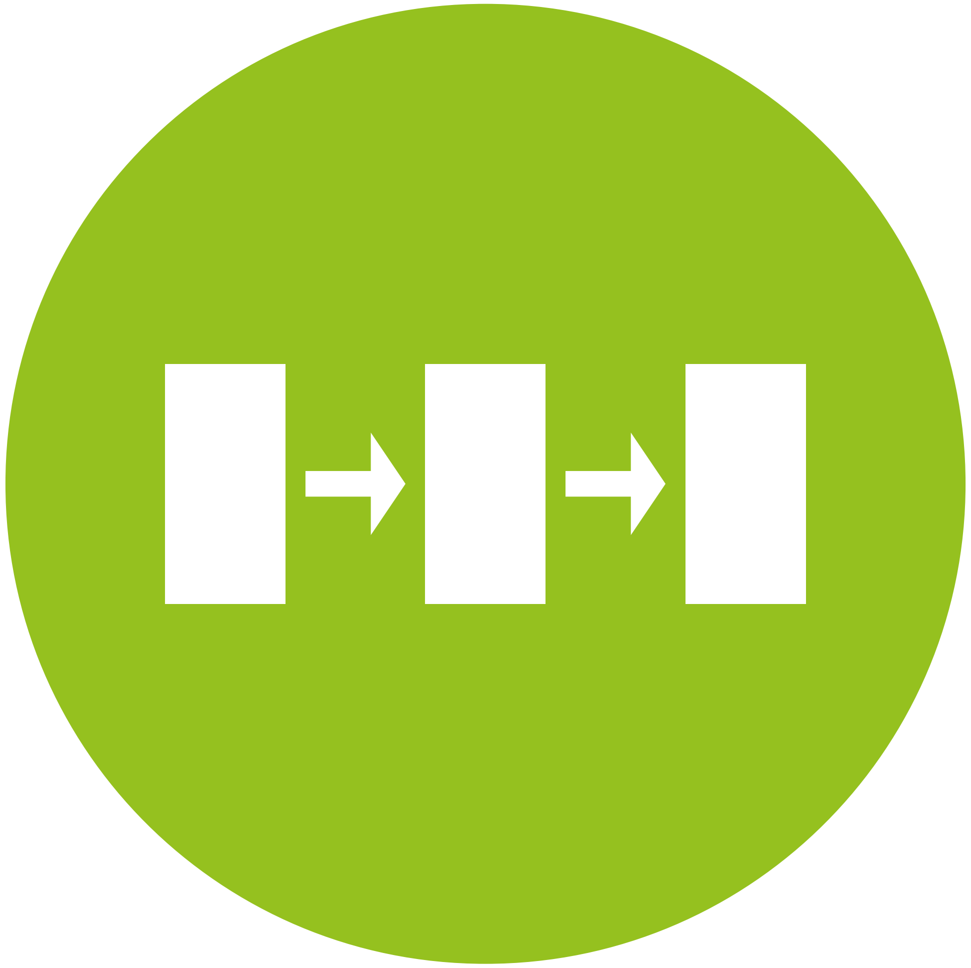
- Is f5 significantly better?

Does not mean that f5 is substantially better

- f4 might "look better"

Factor correlation





SEM

Summary

SEM

Create regression path models on top of a CFA

Two advantages:

1. Path models allow for **simple mediation analysis**
2. Factor models allow for **more precise tests**

SEM

In a SEM you can get the following estimates (all at once):

- CFA stuff: Item loadings and communality
- Regression stuff: R^2, B, s.e., p-values
- Model fit statistics
- Modification indices
- ANOVA stuff: omnibus tests for multiple conditions
- Total (mediated and non-mediated) effects

MIMIC in R

Take two manipulations x1 (levels: u, v, w) and x2 (levels: t, s)

Write a model (run it once for F1, once for F2):

```
model <- ‘
F1 =~ A+B+C
F2 =~ D+E+F
F1 ~ p1*v+p2*w+s+p3*vs+p4*ws’
```

Run the model:

```
fit <- sem(model, data=d, ordered = c(“A”, “B”, “C”,
“D”, “E”, “F”), std.lv=T);
summary(fit, rsquare=T, fit.measures=T)
```

SEM in R

Add a (saturated) path model to CFA

Let's say we take the causal order: conditions -> F1 -> F2:

In R:

```
model <- ‘
F1 =~ A+B+C
F2 =~ D+E+F
F2 ~ F1+v+w+s+vs+ws
F1 ~ v+w+s+vs+ws’;

fit <- sem(model, data=d, ordered = c(“A”, “B”, “C”, “D”, “E”, “F”),
std.lv=T);

summary(fit);
```

Trim & Evaluate

Trim like a path model

Assess the model fit

Report the regression R^2s

Conduct omnibus tests

Get total effects

- Label all the regression coefficients
- Define a variable that follows all the paths from one variable to another

Total effects

```
model <- 'F1 =~ A+B+C
          F2 =~ D+E+F
          F2 ~ p12*F1
          F1 ~ pv1*v+pw1*w+ps1*s

          v2F2 := pv1*p12
          w2F2 := pw1*p12
          s2F2 := ps1*p12'
```

Note: this is much easier in Mplus

- Simply do: F2 IND v
- F2 IND w
- F2 IND s

Advanced SEM

Summary

Method bias

To test common method bias:

```
MODEL:
 F1 by A–C*;
 F2 by D–F*;
 F3 by G–I*;

 method by A–I* (a);
 F1–method@1;
 method with F1–F3@0;
```

Interpretation

We force UVI in the main model

- Using stars to free up the first loading and F1-method@1 to set the factor variances to 1

We create a "method" factor with all items

- They all load equally on the factor (a)
- We make the factor orthogonal to the other factors (method with F1-F3@0)

Check whether bias is significant, and whether it affects the loadings of the model

Why multi-level?

Repeated measurements

e.g. participants make 30 decisions

(Partially) within-subjects design

e.g. participants are randomly assigned to 1 of 3 games, and tested once with sound on and once with sound off

Grouped data

e.g. participants perform tasks in groups of 5

A combination of the above

Multi-level in R

Specify the survey design (random intercept)

Specify the model

Run using sem(model) —> output: fit

Rerun using lavaan.survey(fit, design)

Get the summary (as usual)

Multi-level in Mplus

Under variable, add the random intercept:

Cluster = userId;

Under analysis, specify multi-level:

type = complex;

That is all!

Why invariance?

To see if comparisons are warranted

e.g. can we compare measures of satisfaction between cultures?

To detect conceptual differences between groups

e.g. do privacy practices have similar meanings in different cultures?

To detect conceptual drift over time

e.g. does “information overload” mean the same thing now as it did 20 years ago?

Types of invariance

Configural (same general structure)

If not, stop here

Metric invariance (equal factor loadings)

- Test against configural

If significant, try partial, freeing up loadings one by one

- E.g. MODEL Y: F1 BY C*;
- MODEL Z: F1 BY C*;
- Continue until no longer worse than the configural model

Types of invariance

Scalar invariance (equal intercepts/thresholds)

- Added on top of (partial) metric model
- If partial metric, free up those intercepts, e.g.
- MODEL Y: [C];
- MODEL Z: [C];
- Test against the (partial) metric model

If signficant, try partial, freeing up (additional) intercepts

- Continue until no longer significant

Interactions

Factor * condition:

- Is the effect of F1 on F2 different between conditions u, v, and w?

Multiple group modeling approach:

- Run the same model in each condition, but free up the parameter of F2 ON F1
- Make sure to also constrain factor intercepts where no effect of the conditions is expected!

Multiple groups

Under VARIABLE:

- Specify the moderating manipulation as a "grouping" variable: grouping = cctrl(0=u 1=v 2=w)

Add a MODEL section for all groups except the baseline

- Model v:
- Model w:

Add corresponding labels to each MODEL to restrict the moderation

Factor * condition

multi.inp

```
MODEL:
 F1 BY A* B C;
 F2 BY D* E F;
 F3 BY G* H I;

 F3 ON F2 F1 (1-2)
 F2 ON F1 (p1);

MODEL v:
 F3 ON F2 F1 (1-2)
 F2 ON F1 (p2);

 [F1] (p4);
 [F3] (p5);
```

```
MODEL w:
 F3 ON F2 F1 (1-2)
 F2 ON F1 (p3);

 [F1] (p4);
 [F3] (p5);


MODEL CONSTRAINT:
 p4=0;
 p5=0;

MODEL TEST:
p1=p2;
p1=p3;
```

Clustering in Mplus

LCA: cluster people on the value of the items

Does not assume a latent factor structure

FMA: cluster people on the value of the factors

Assumes a latent factor structure

Sometimes they show essentially the same result

But not always!

LCA

Under VARIABLE:

Specify the number of classes: classes = c(2)

Under ANALYSIS:

Specify mixture model: type = mixture

Optionally, specify iterations etc

Under OUTPUT:

Ask for tech11; (for model comparison)

Run, increase iterations if needed, then repeat for c(3), etc.

FMA

Same, but under MODEL:

Add %overall% and then the factor model

Prepare to wait :-)

(this approach often requires a lot of iterations and starts)

How many classes?

Balance the following criteria

- Minimum of BIC
- Maximum entropy
- Loglikelihood levels off
- p-value of successor > .05 (use Lo-Mendell-Rubin adjusted LRT test, available in output: tech11)
- **Solution makes sense**

That's all, folks!

I hope you enjoyed the course!

You now know more about stats than 99% of the people in this field :-)

Please fill out the course evaluation!

"It is the mark of a truly intelligent person to be moved by statistics."

T H A N K S !

George Bernard Shaw