Psych 253
Advanced statistical modeling

Structural equation modeling (Part 1)

Daniel Yamins
Wu Tsai Neurosciences Institute
Departments of Psychology and Computer Science
Stanford University

Russ Poldrack
Department of Psychology
Stanford University

As scientists we care about understanding the underlying processes that give rise to our observations

"All observation is befuddled by error"

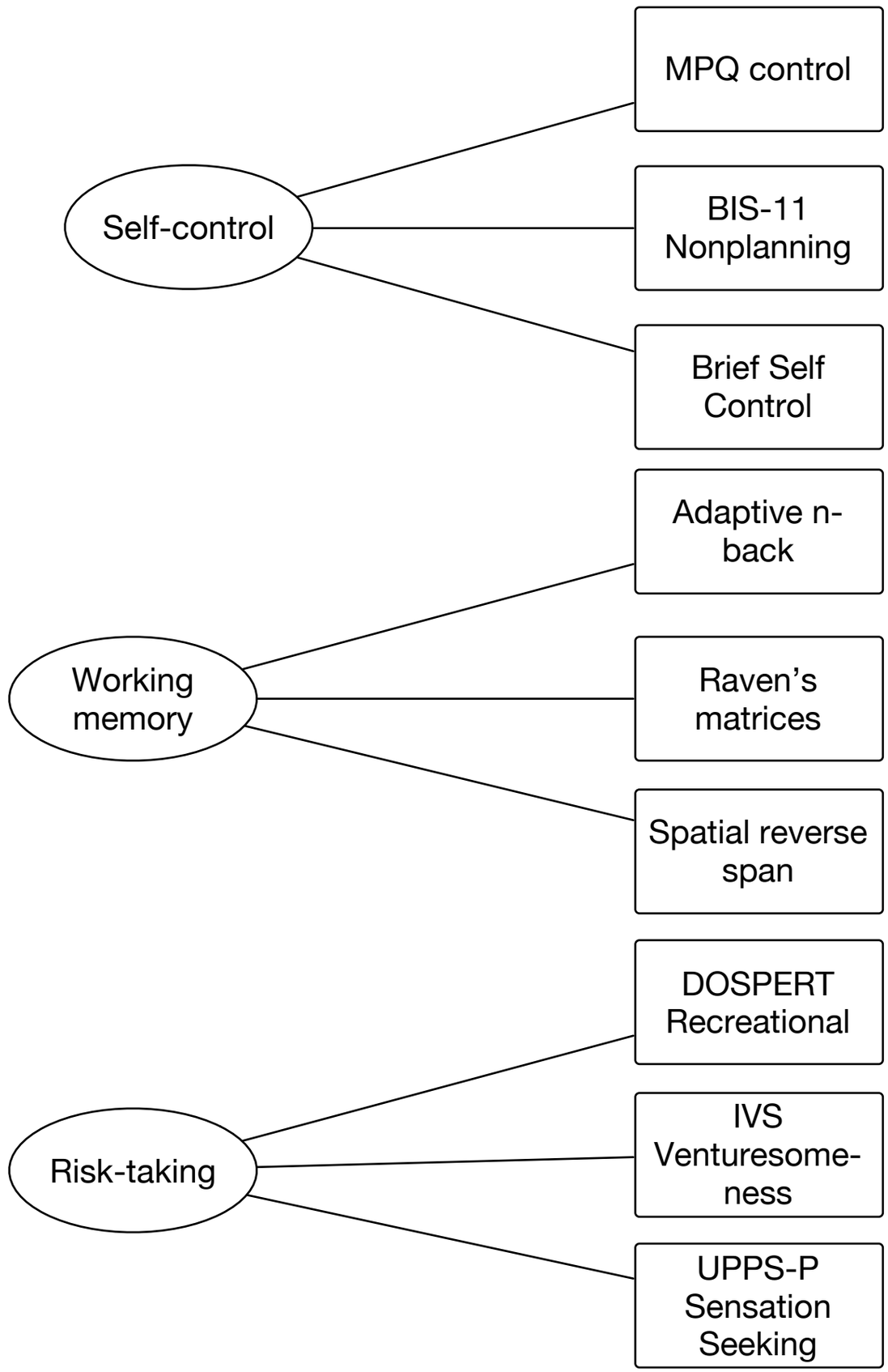
How can we understand the relations between observations and the constructs that underlie them?

Can we use data to test theories regarding the latent structure underlying our data?

Motivating example: Measurement and psychological constructs

How do different measures of self-control relate to one another?

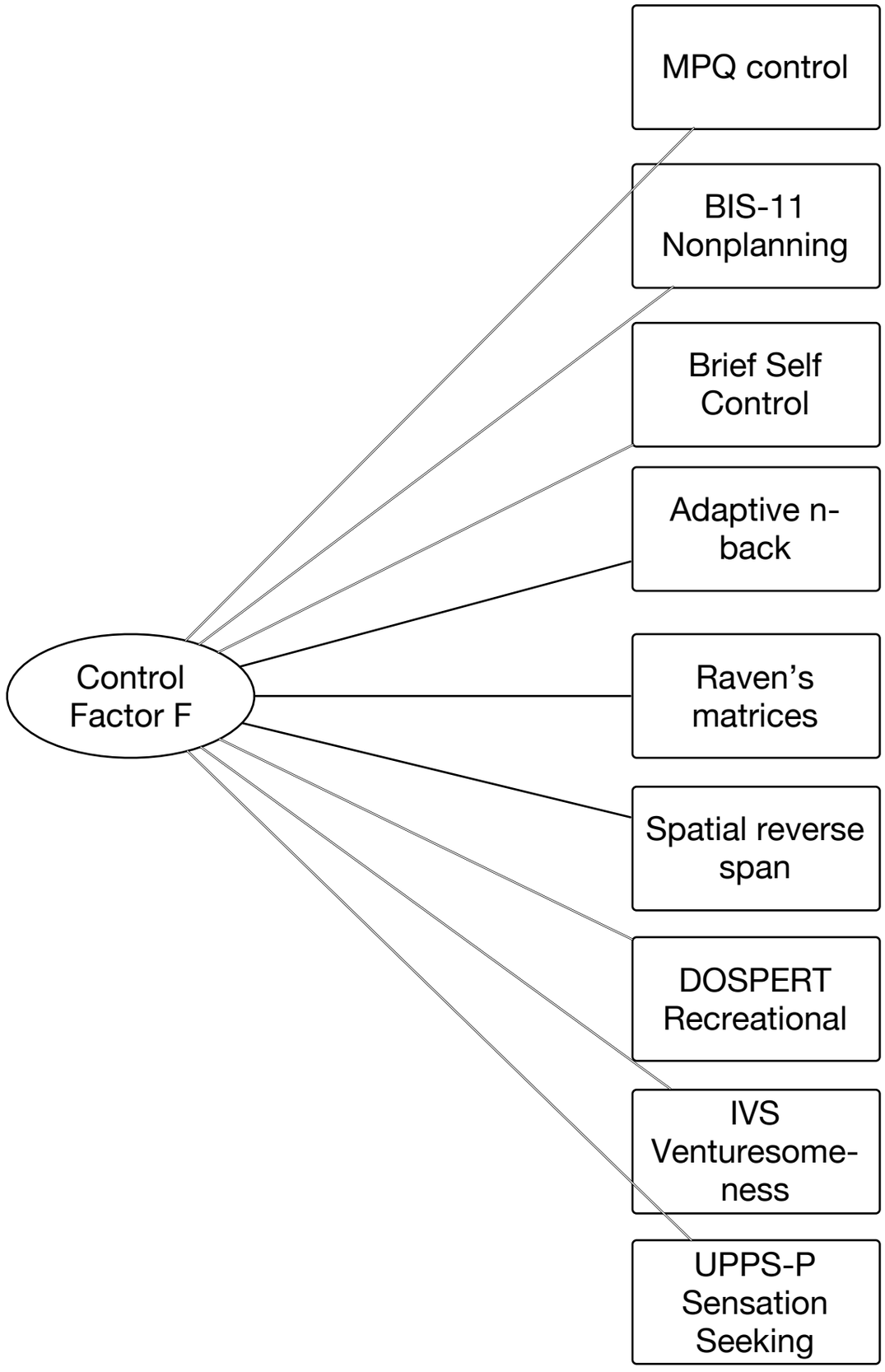
three-factor theory



Motivating example: Measurement and psychological constructs

How do different measures of self-control relate to one another?

one-factor theory



History of SEM

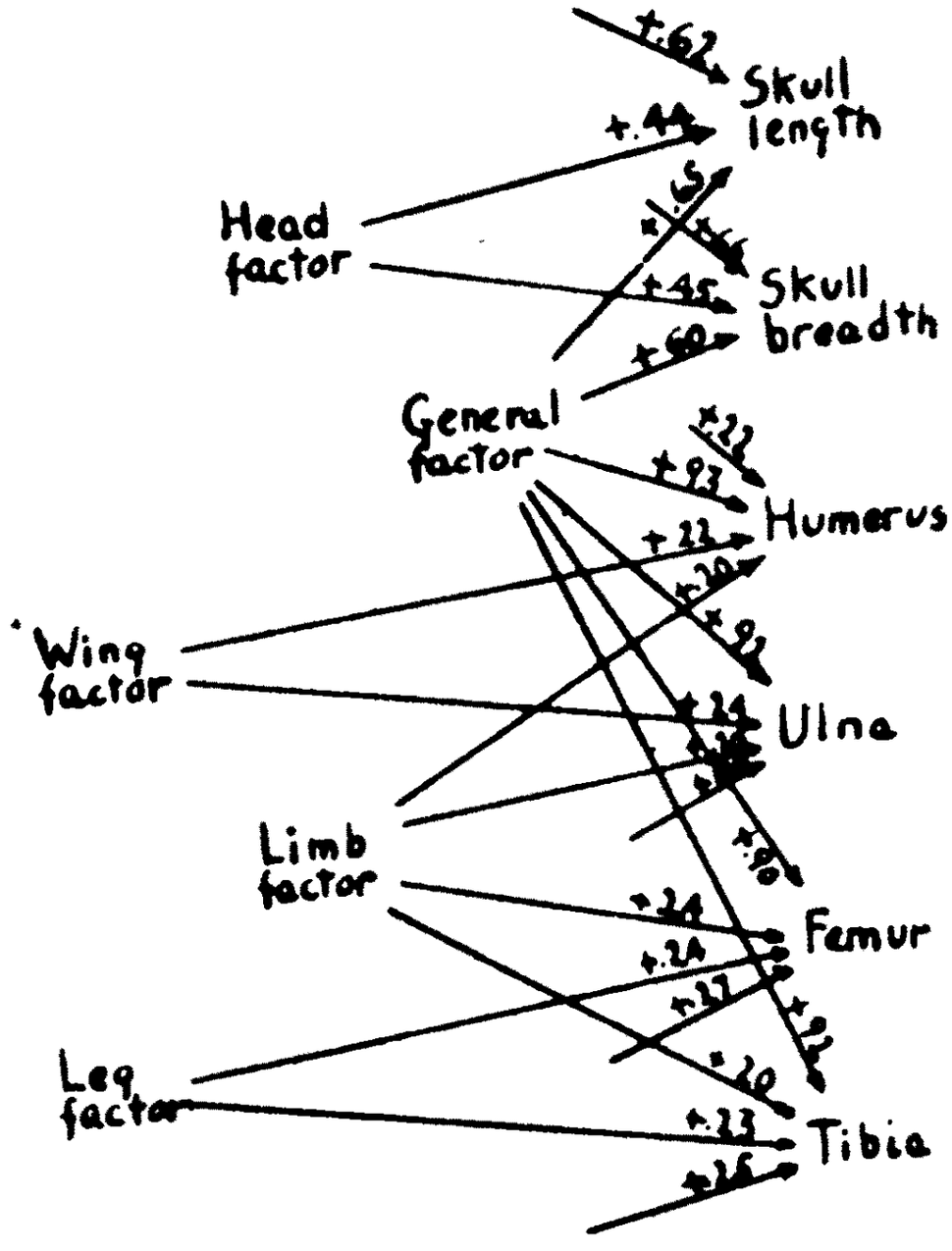


FIG. 21

Sewall Wright

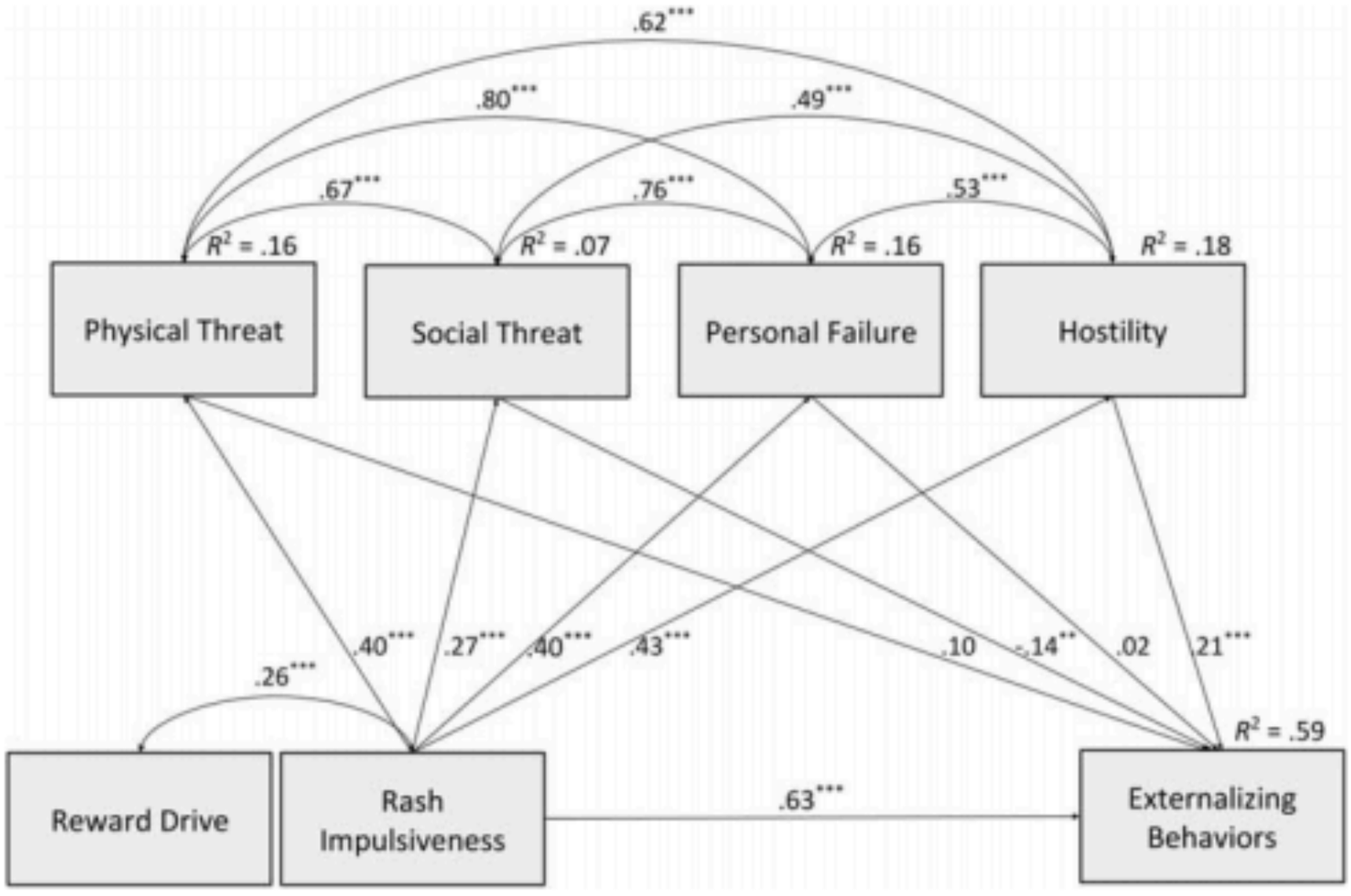
It has been emphasized that the method of path coefficients is not intended to accomplish the impossible task of deducing causal relations from the values of the correlation coefficients. It is intended to combine the quantitative information given by the correlations with such a qualitative information as may be at hand on causal relations to give a quantitative interpretation. The

Wright, 1934

more on causality next time…

Flavors of SEM: Path analysis

Fig. 2 Hypothesized model of negative automatic thoughts mediating the association between impulsivity and externalizing behaviors. *Note.* $*p < .05$; $**p < .01$; $***p < .001$.



Revill et al., 2020

Relates a set of observed variables

Flavors of SEM: "Confirmatory factor analysis"

latent factors

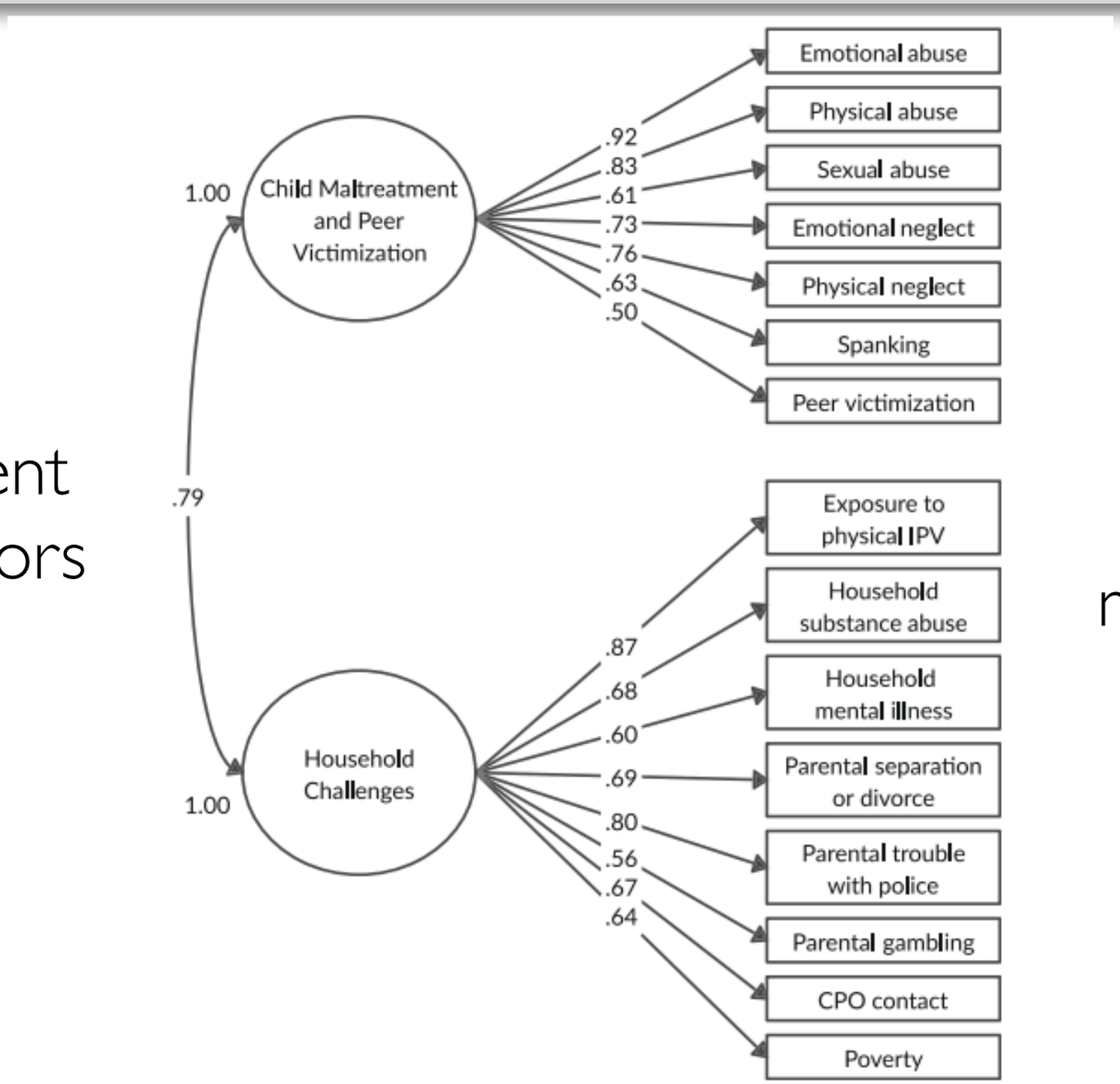


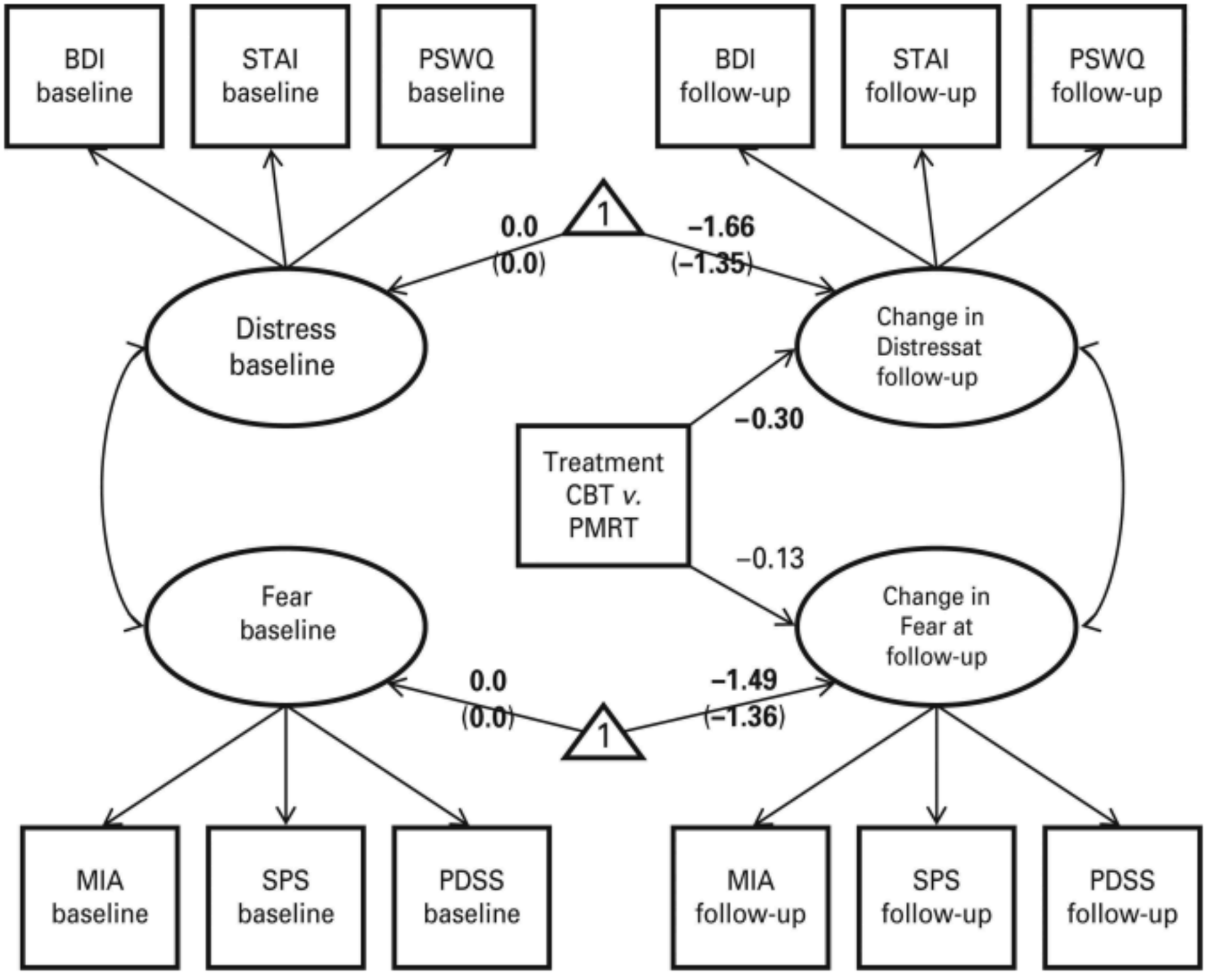
Fig. 1 Two-factor CFA model with standardized factor loadings for parent ACEs. *Model fit:* $X^2_{(89)} = 316.8$, $p < .001$; RMSEA = .051, 90% CI = .045–.057; CFI = .952; TLI = .943; SRMR = .071. *Abbreviations:* IPV = intimate partner violence; CPO = child protective organization

observed measurement

Afifi et al, 2020

relates observed measurements to latent factors

Flavors of SEM: full structural equation model

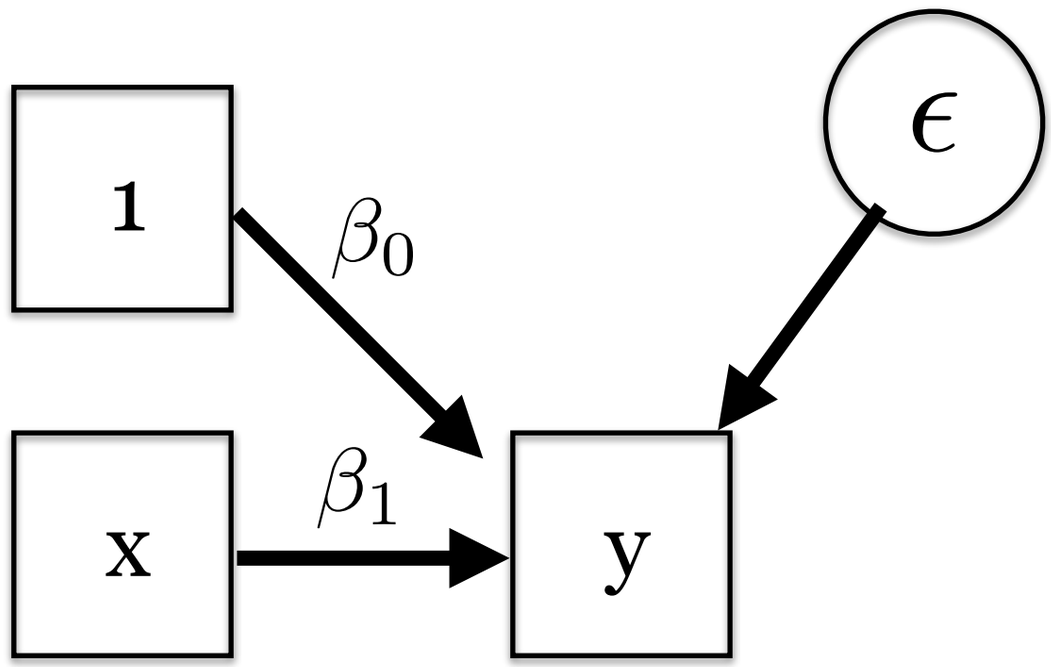


Kushner et al, 2013

Includes relations between both latent and observed ("manifest") variables

An example: Linear regression via SEM

$$y_i = \beta_0 + x_i * \beta_1 + \epsilon_i$$



Structural equation modeling in R using lavaan

```
%%R -i simdataDf -o results

library(lavaan)

myModel <- ' # regressions
             Y ~ x

             # variances and covariances
               Y ~~ Y

               # intercepts
               Y~ 1
           '

fit <- lavaan(myModel, data=simdataDf)
summary(fit, standardized=TRUE)
```

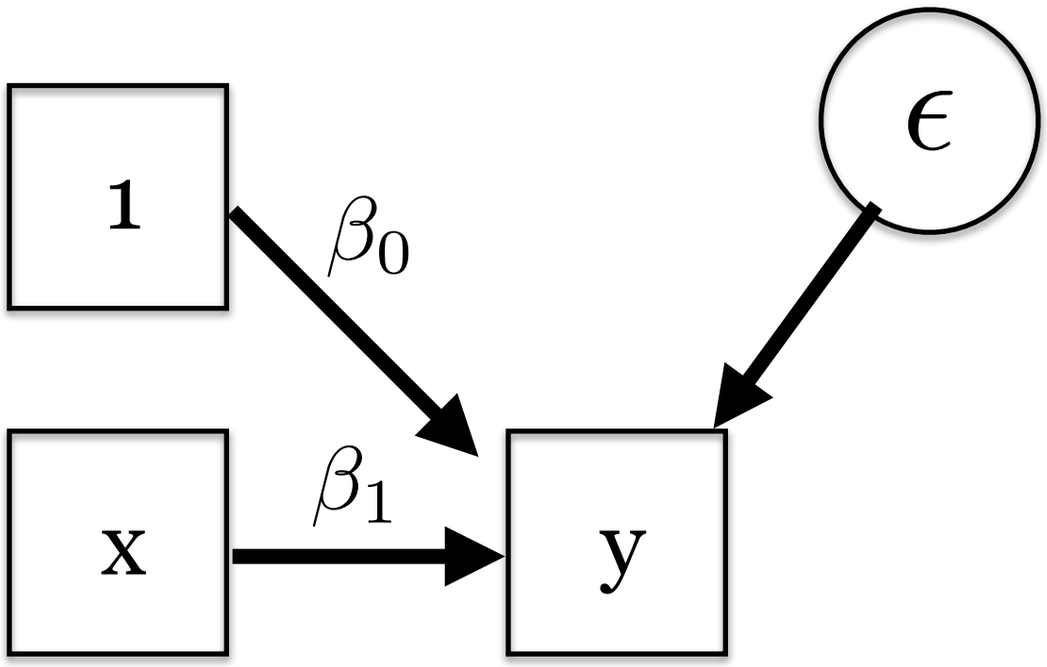
Using lavaan

```
lavaan 0.6-5 ended normally after 11 iterations

  Estimator                                         ML
  Optimization method                           NLMINB
  Number of free parameters                          3

  Number of observations                           500

Model Test User Model:

  Test statistic                                 0.000
  Degrees of freedom                                 0

Parameter Estimates:

  Information                                 Expected
  Information saturated (h1) model          Structured
  Standard errors                             Standard

Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  Y ~
    x                 0.475    0.046   10.366    0.000    0.475    0.421

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .Y                 0.026    0.045    0.579    0.562    0.026    0.024

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .Y                 1.026    0.065   15.811    0.000    1.026    0.823
```

cf. OLS results:

	coef	std err
const	0.0263	0.045
x	0.4750	0.046

Linear regression via SEM: Model-implied covariance



Observed covariance

$$\begin{bmatrix} \sigma_x^2 & \text{cov}(x,y) \\ \text{cov}(x,y) & \sigma_y^2 \end{bmatrix}$$

Implied covariance

$$\begin{bmatrix} \sigma_x^2 & cov(x, x\beta_1) \\ cov(x, x\beta_1) & \sigma_y^2 \end{bmatrix}$$

Basics of SEM: Loss function

Loss function: model-implied covariance vs sample covariance

How can we compare covariance matrices?

using the determinant (actually the log-determinant)

$$x = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \qquad vcov = \begin{bmatrix} \sigma_x^2 & cov(x,y) \\ cov(x,y) & \sigma_y^2 \end{bmatrix}$$

$$|x| = a * d - b * c \qquad |vcov| = \sigma_x^2 * \sigma_y^2 - cov(x,y)^2$$

The determinant of the covariance matrix as a loss

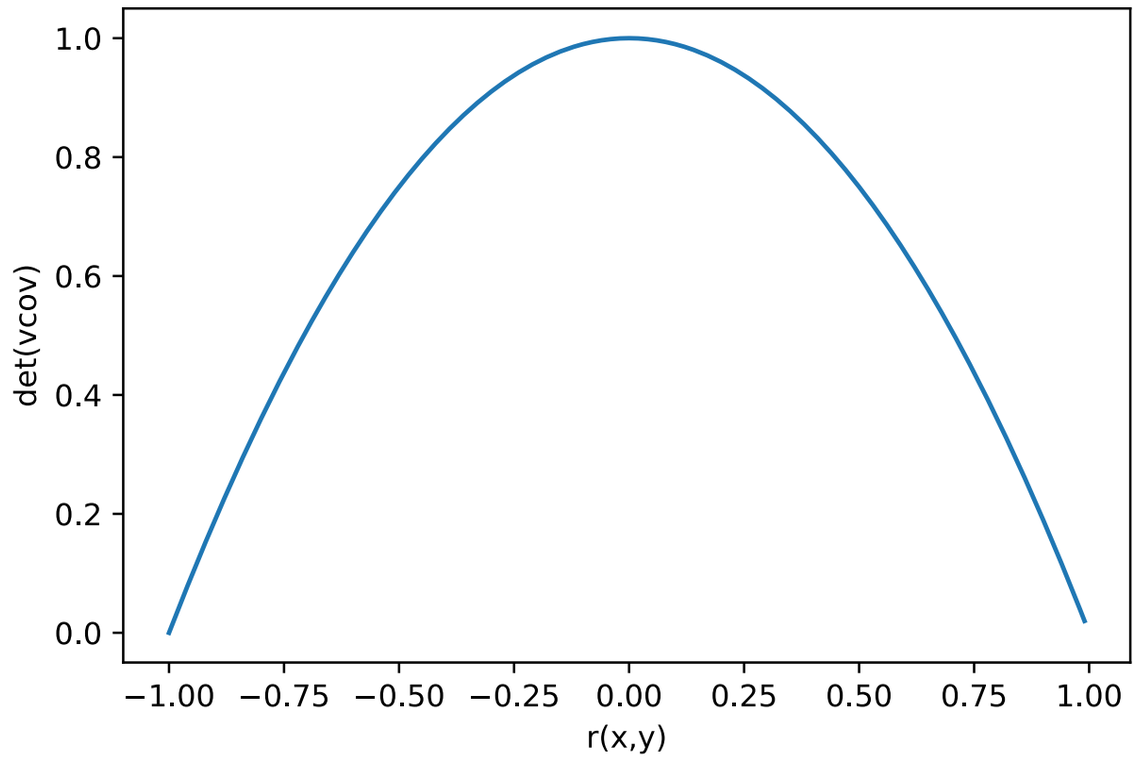
$$vcov = \begin{bmatrix} \sigma_x^2 & cov(x,y) \\ cov(x,y) & \sigma_y^2 \end{bmatrix}$$

$$r = \frac{covariance_{x,y}}{s_x * s_y}$$

$$|vcov| = \sigma_x^2 * \sigma_y^2 - cov(x,y)^2$$

if r(x, y) = 1:

$$|vcov| = 0$$



Basics of SEM: Loss function

Loss function: model-implied covariance vs sample covariance

Σ : model-implied covariance
S: observed sample covariance
p: number of dependent variables
q: number of independent variables

$$loss = log|\Sigma| - log|\mathbf{S}| + tr(\mathbf{S}\Sigma^{-1}) - (p+q)$$

log-determinant of model-implied covariance matrix

log-determinant of observed covariance matrix

divergence between observed and predicted variances

Basics of SEM: Loss function

Loss function: model-implied covariance vs sample covariance

Σ : model-implied covariance
S: observed sample covariance
p: number of dependent variables
q: number of independent variables

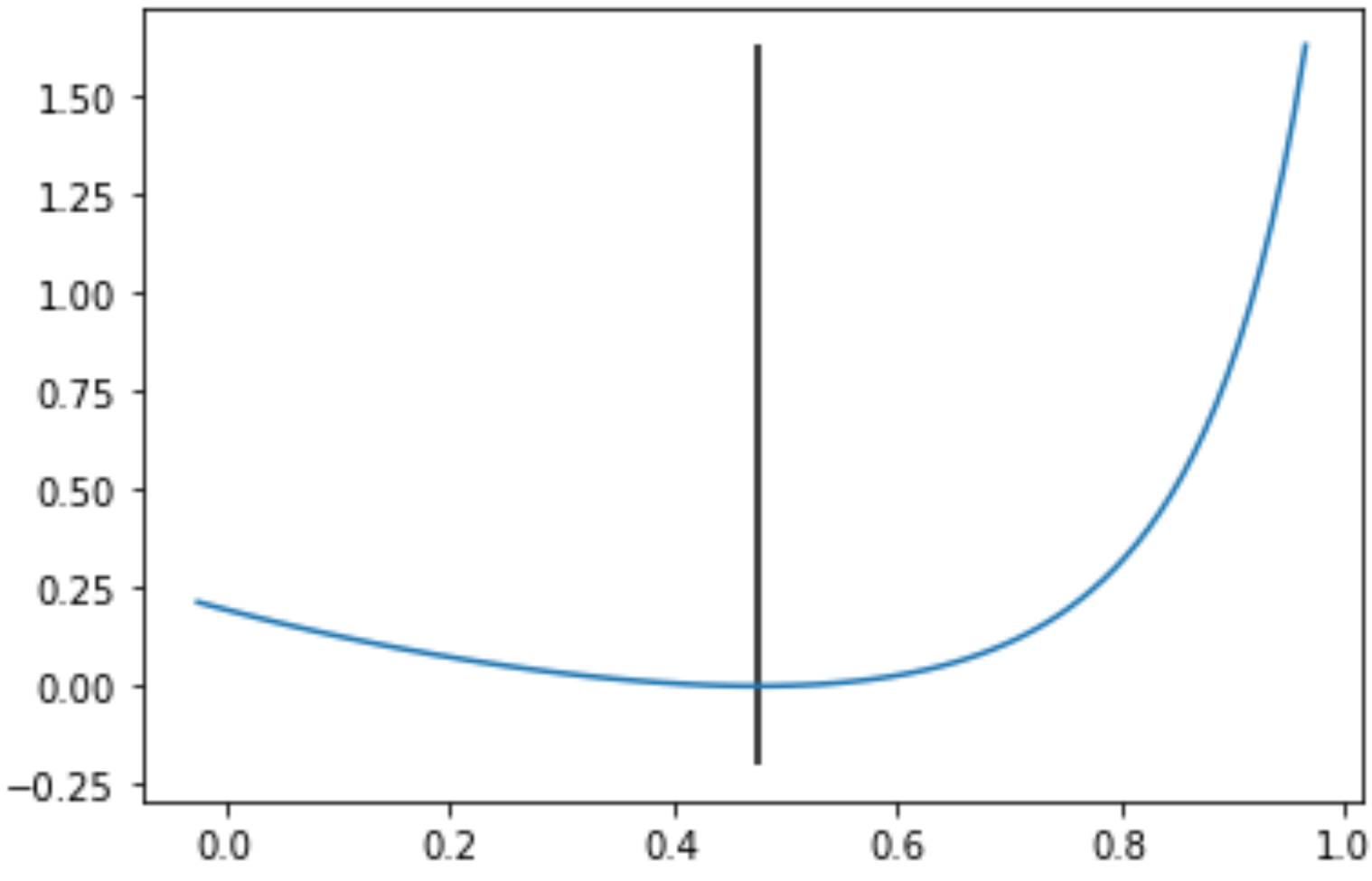
$$loss = log|\Sigma| - log|\mathbf{S}| + tr(\mathbf{S}\Sigma^{-1}) - (p + q)$$

The values of the parameters ($\boldsymbol{\theta}$) that minimize this loss are those that make the implied covariance as similar as possible to the actual covariance, while also respecting the the constraints on the model (i.e. the graph)

Basics of SEM: Loss function

By computing loss for many values of the regression parameter, we see that the value identified with the minimum loss is the same as that determined using OLS

```
Parameter with minimum loss: 0.4750111814197142
OLS estimate of parameter: 0.4750111814197142
```



Psych 253
Advanced statistical modeling

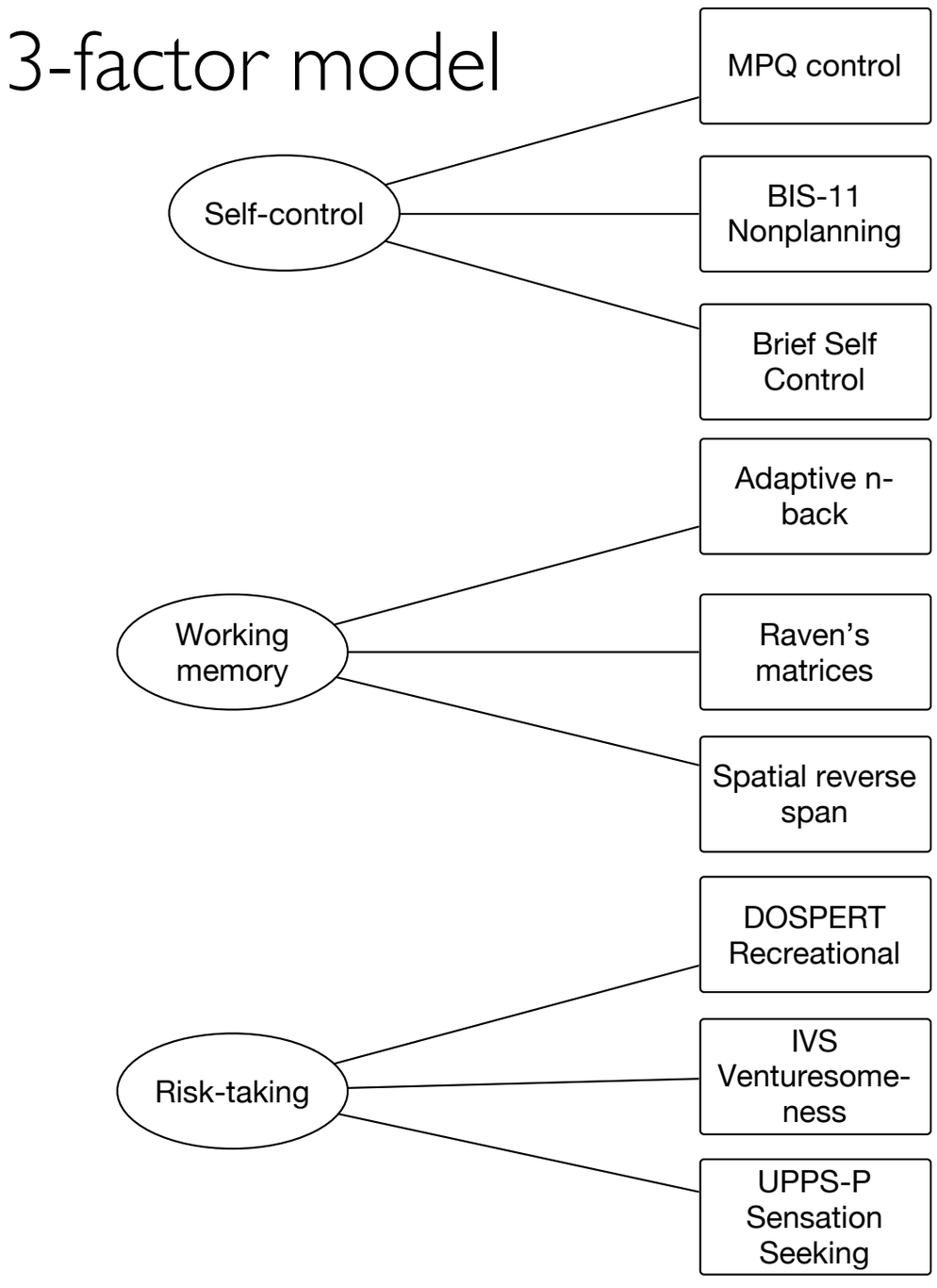
Structural equation modeling (Part 2)

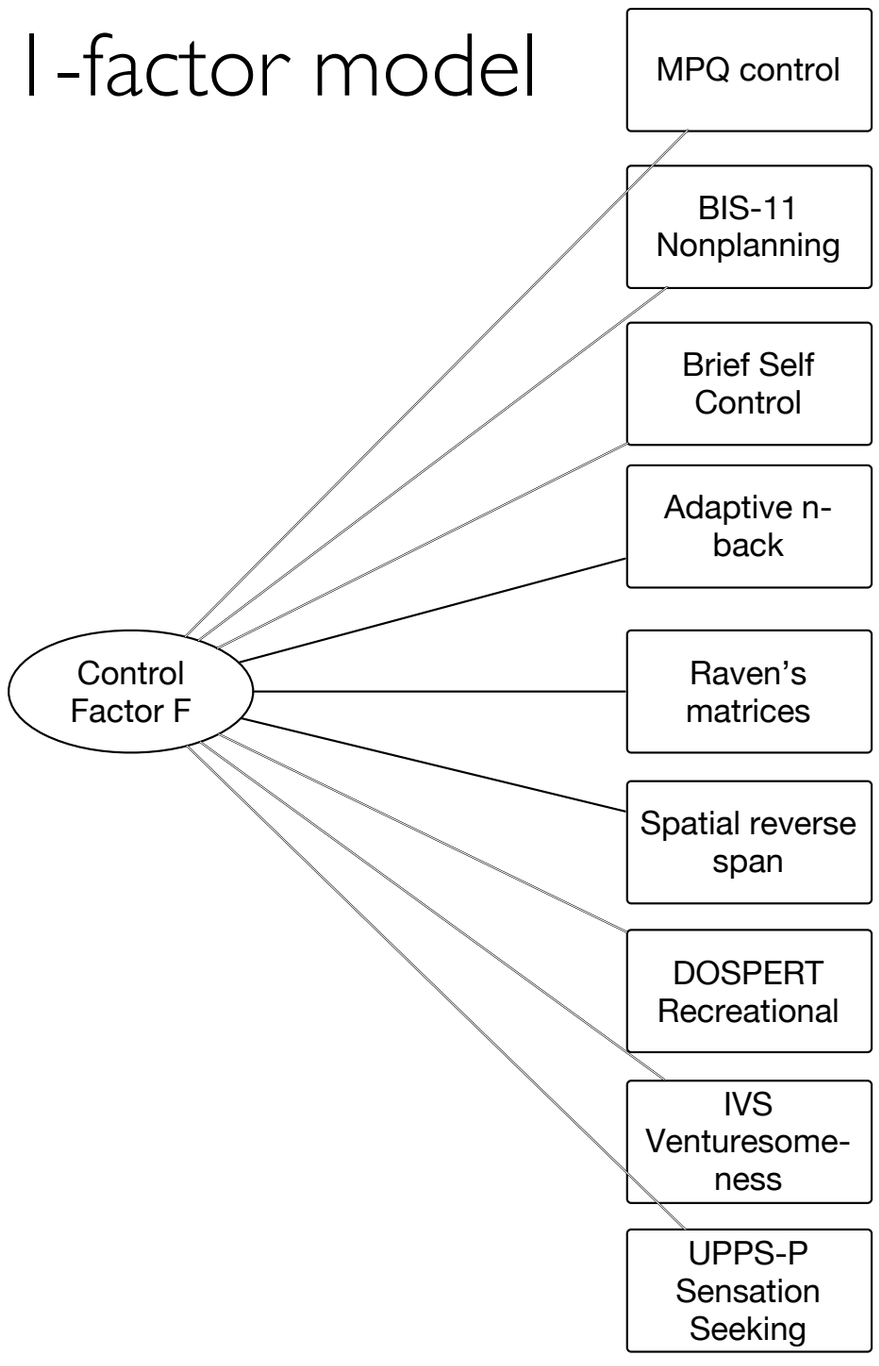
Daniel Yamins
Wu Tsai Neurosciences Institute
Departments of Psychology and Computer Science
Stanford University

Russ Poldrack
Department of Psychology
Stanford University

The confirmatory factor analysis (CFA) model

CFA relates observed variables to a set of latent variables
Let's say I wanted to test between two different models of my data:





The confirmatory factor analysis (CFA) model

$$y = \Lambda\eta + \epsilon$$

$$y = N(0, S)$$

$$\eta = N(0, \Psi)$$

$$\epsilon = N(0, \Theta)$$

S : observed covariance

Ψ : factor var/cov

Θ : error variances

where:

- y is a p x 1 vector of responses
- Λ is a p x m matrix of factor loadings
- η is an m x 1 vector of latent variables
- ϵ is a p x 1 vector of errors

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \end{bmatrix} = \begin{bmatrix} \lambda_{11} & 0 & 0 \\ \lambda_{21} & 0 & 0 \\ \lambda_{31} & 0 & 0 \\ 0 & \lambda_{42} & 0 \\ 0 & \lambda_{52} & 0 \\ 0 & \lambda_{62} & 0 \\ 0 & 0 & \lambda_{73} \\ 0 & 0 & \lambda_{83} \\ 0 & 0 & \lambda_{93} \end{bmatrix} \begin{bmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \epsilon_4 \\ \epsilon_5 \\ \epsilon_6 \\ \epsilon_7 \\ \epsilon_8 \\ \epsilon_9 \end{bmatrix}$$

The confirmatory factor analysis (CFA) model

$$y = \Lambda\eta + \epsilon$$

$$y = N(0, S)$$

$$\eta = N(0, \Psi)$$

$$\epsilon = N(0, \Theta)$$

where:

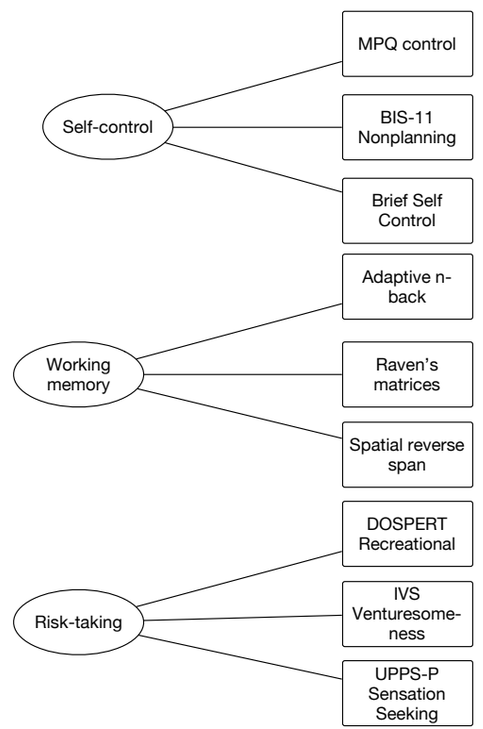
- y is a p x 1 vector of responses
- Λ is a p x m matrix of factor loadings
- η is an m x 1 vector of latent variables
- ϵ is a p x 1 vector of errors

Goal:
Identify model parameters that minimize the loss

core goal is to make the implied covariance as close to the observed covariance given the model constraints
(i.e. the graph)

CFA model in lavaan

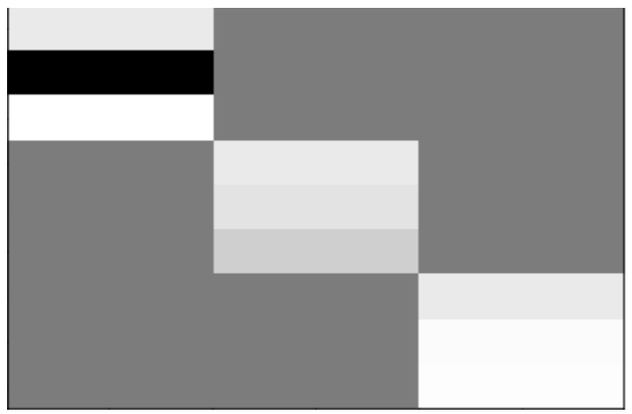
```
%%R -i cfadata

cfa.model <- 'SC =~ mpq_control_survey.control +
                    bis11_survey.Nonplanning +
                    brief_self_control_survey.self_control
              WM =~ adaptive_n_back.mean_load +
                     ravens.score +
                     spatial_span.reverse_span
             RT =~  dospert_rt_survey.recreational +
                    impulsive_venture_survey.venturesomeness +
                    upps_impulsivity_survey.sensation_seeking
'

fit <- cfa(cfa.model, data=cfadata)
```



CFA: Implied covariance

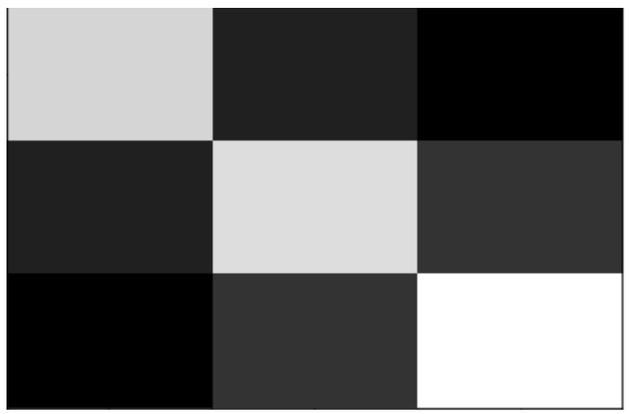
$$\Sigma = \Lambda \Psi \Lambda' + \Theta$$

Λ



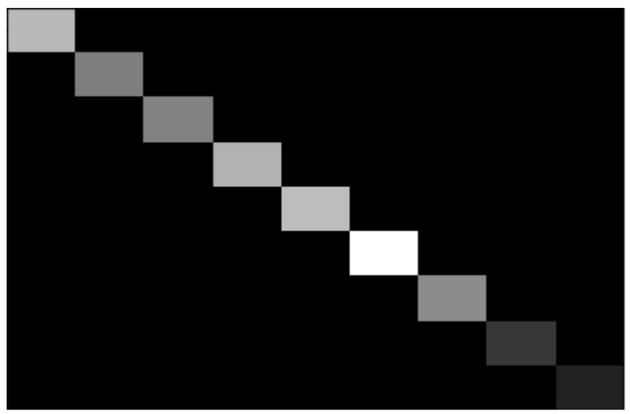
mapping of measures to factors

Ψ



variance/ covariance of factors

Θ

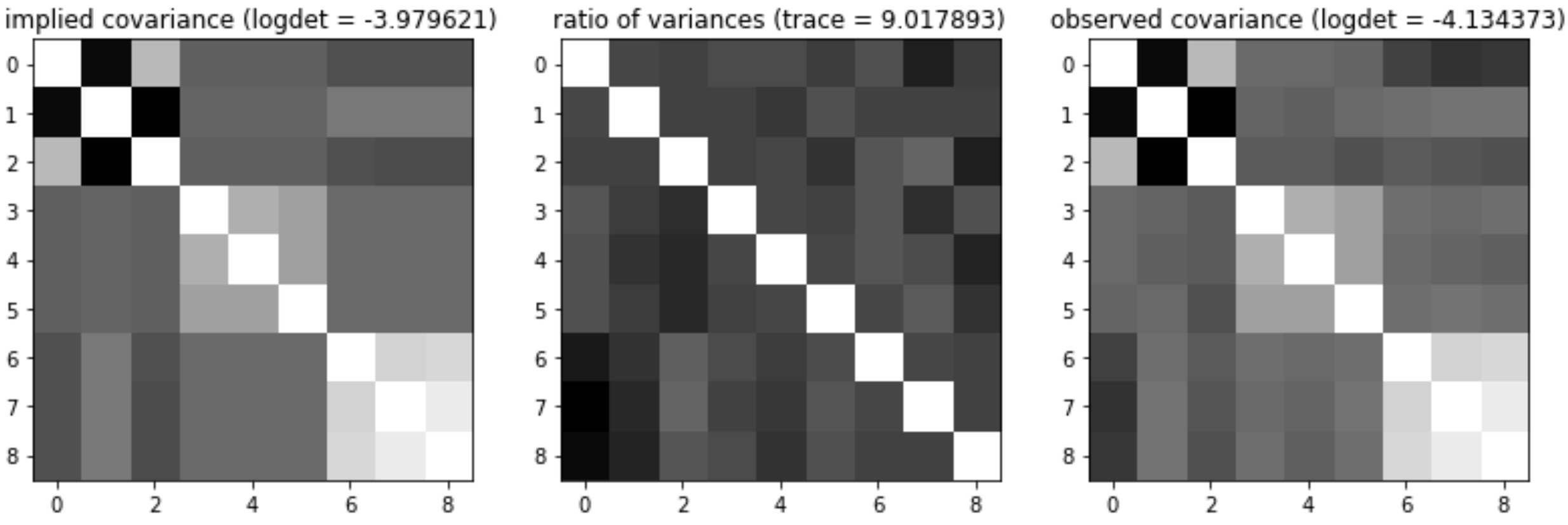


error variances for observed measures

Goal: find parameters that minimize loss…

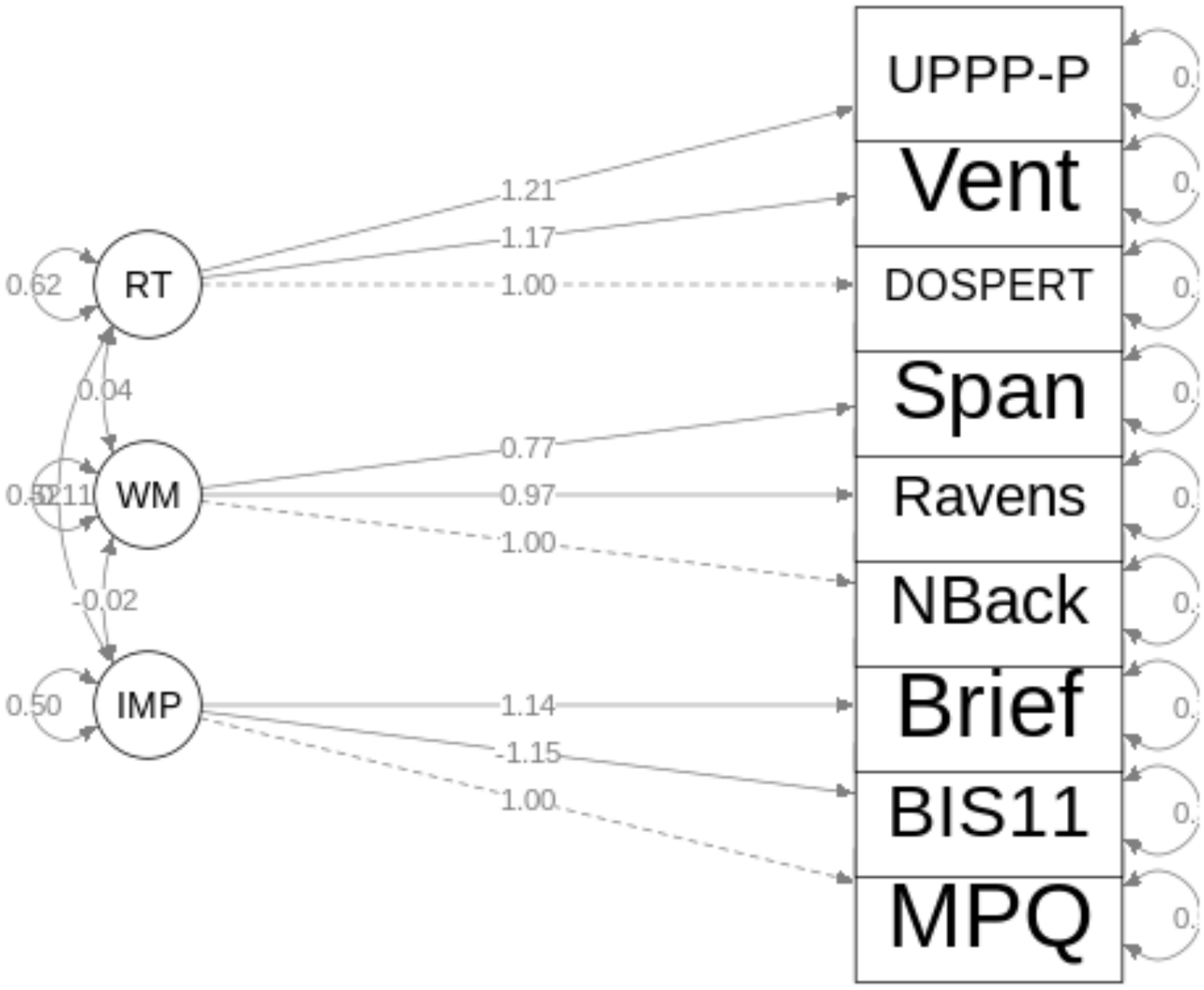
CFA: Computing the loss

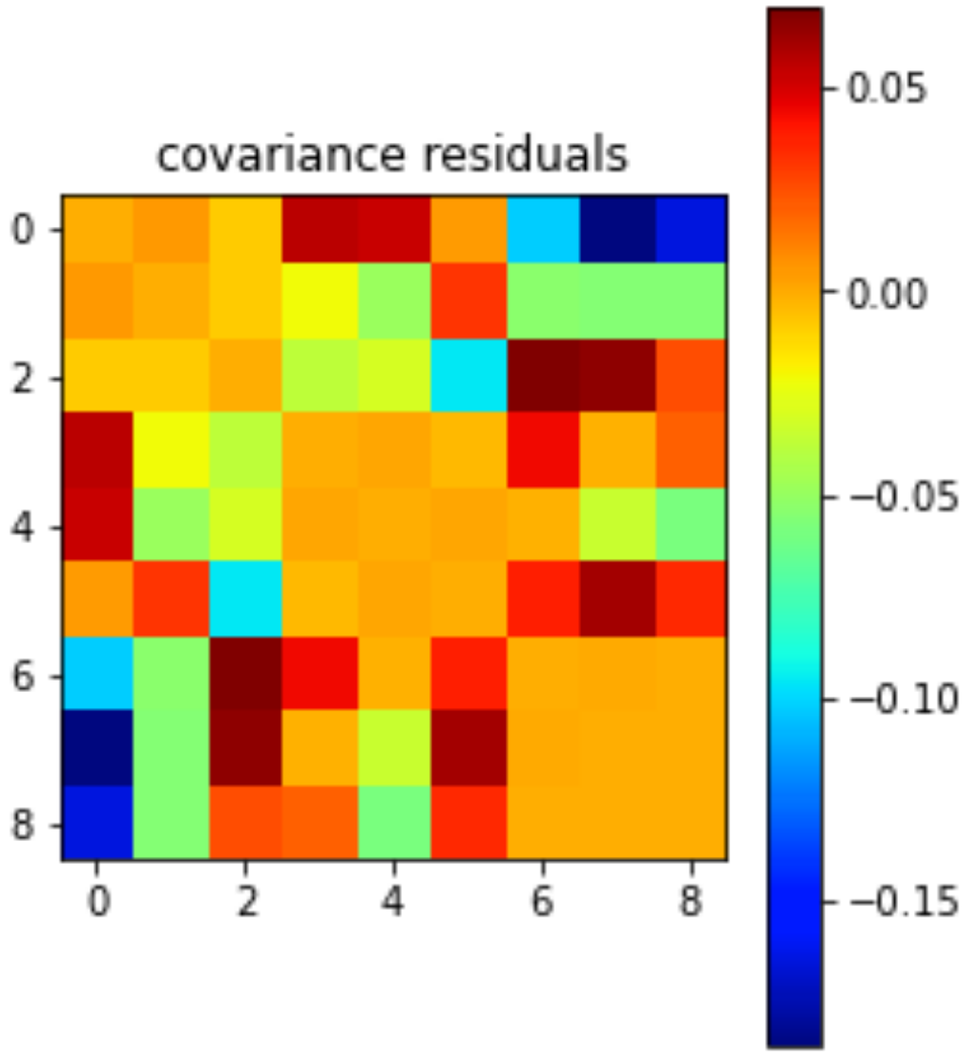
$$\text{loss} = log|\Sigma| + tr(\mathbf{S}\Sigma^{-1}) - log|\mathbf{S}| - (p+q)$$



loss = -3.979 + 9.017 - -4.134 - 9 = 0.172

CFA: Fitted model and residuals



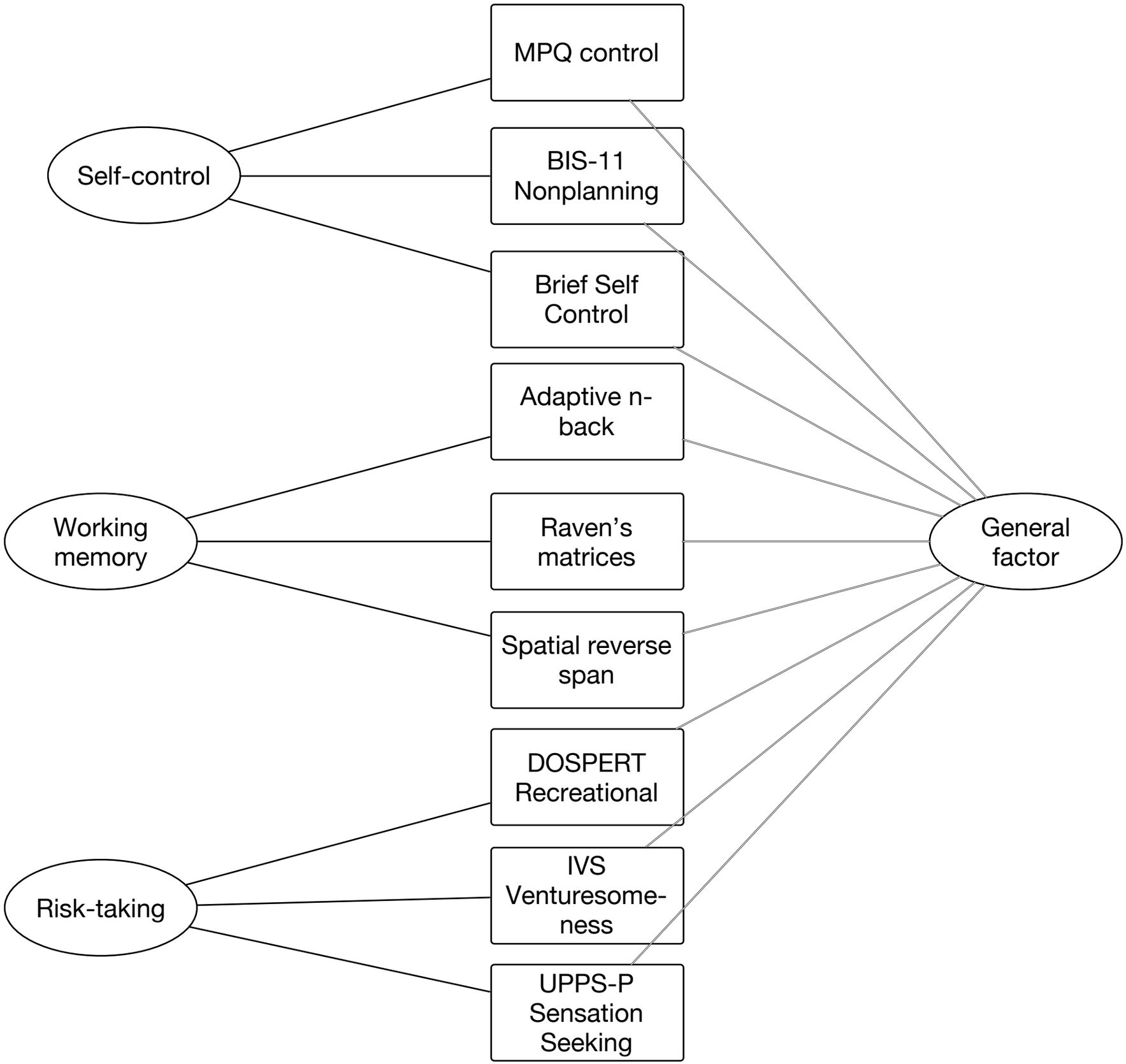


Model fit assessment

```
%%R

summary(fit,header=FALSE, estimates=FALSE, fit.measures=TRUE)
```

```
Model Test Baseline Model:

  Test statistic                              2083.724
  Degrees of freedom                            36
  P-value                                     0.000

User Model versus Baseline Model:

  Comparative Fit Index (CFI)                   0.969
  Tucker-Lewis Index (TLI)                      0.954

Loglikelihood and Information Criteria:

  Loglikelihood user model (H0)              -5433.441
  Loglikelihood unrestricted model (H1)      -5389.939

  Akaike (AIC)                               10908.881
  Bayesian (BIC)                             10997.555
  Sample-size adjusted Bayesian (BIC)        10930.899

Root Mean Square Error of Approximation:

  RMSEA                                          0.072
  90 Percent confidence interval - lower         0.056
  90 Percent confidence interval - upper         0.089
  P-value RMSEA <= 0.05                          0.012

Standardized Root Mean Square Residual:

  SRMR                                           0.053
```

Bifactor (hierarchical) model



Model comparison

```
%%R -i cfadata

bifactor.model <- 'SC =~ mpq_control_survey.control +
                       bis11_survey.Nonplanning +
                       brief_self_control_survey.self_control
                  WM =~ adaptive_n_back.mean_load +
                       ravens.score +
                       spatial_span.reverse_span
                  RT =~  dospert_rt_survey.recreational +
                       impulsive_venture_survey.venturesomeness +
                       upps_impulsivity_survey.sensation_seeking

                  F =~ mpq_control_survey.control +
                       bis11_survey.Nonplanning +
                       brief_self_control_survey.self_control +
                       adaptive_n_back.mean_load +
                       ravens.score +
                       spatial_span.reverse_span +
                       dospert_rt_survey.recreational +
                       impulsive_venture_survey.venturesomeness +
                       upps_impulsivity_survey.sensation_seeking

                  # fix correlation bw general and specific factors to zero
                  F         ~~ 0*WM
                  F         ~~ 0*IMP
                  F         ~~ 0*RT
                     '
bifactor_fit <- cfa(bifactor.model, data=cfadata)
```

Model comparison

We want to compare the fit of two models.

Why can't we just choose the model with the best fit to the data?

"SRMR": standardized root mean square residual

	srmr
bifactor_fit	.022†
fit_3factor	.053

Model selection criteria

We want to compare the fit of two models.

Why can't we just choose the model with the best fit to the data?

Model comparison: Standard approaches

```
compareFit(fit_3factor, bifactor_fit)

##################### Nested Model Comparison ##############################
Chi-Squared Difference Test

              Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
bifactor_fit 15 10860 10986 19.994
fit_3factor  24 10909 10998 87.004     67.009       9  5.857e-11 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

######################### Model Fit Indices ###############################
                chisq df pvalue    cfi   tli         aic         bic rmsea  srmr
bifactor_fit 19.994† 15   .172 0.998† .994† 10859.872† 10986.549† .026† .022†
fit_3factor  87.004  24   .000  .969  .954  10908.881  10997.555  .072  .053

#################### Differences in Fit Indices ############################
                             df    cfi   tli    aic    bic rmsea  srmr
fit_3factor - bifactor_fit  9 -0.028 -0.04 49.009 11.006 0.046 0.031
```

Model comparison: Information criteria

We need to balance model fit with model complexity

Information criteria:

Akaike information criterion (AIC):

$$AIC = -2\ log\ \mathcal{L} + 2p$$

Bayesian information criterion (BIC):

$$BIC = -2\ log\ \mathcal{L} + p\ log(n)$$

BIC penalizes more than AIC whenever n > e^2 (~7)

Model comparison: Crossvalidation

- Split data
- Estimate model on training data
- Compute loss on held-out data

	Train	Test
3-factor	18.174445	18.665081
1-factor	19.869873	20.537528
bifactor	18.055857	18.553369

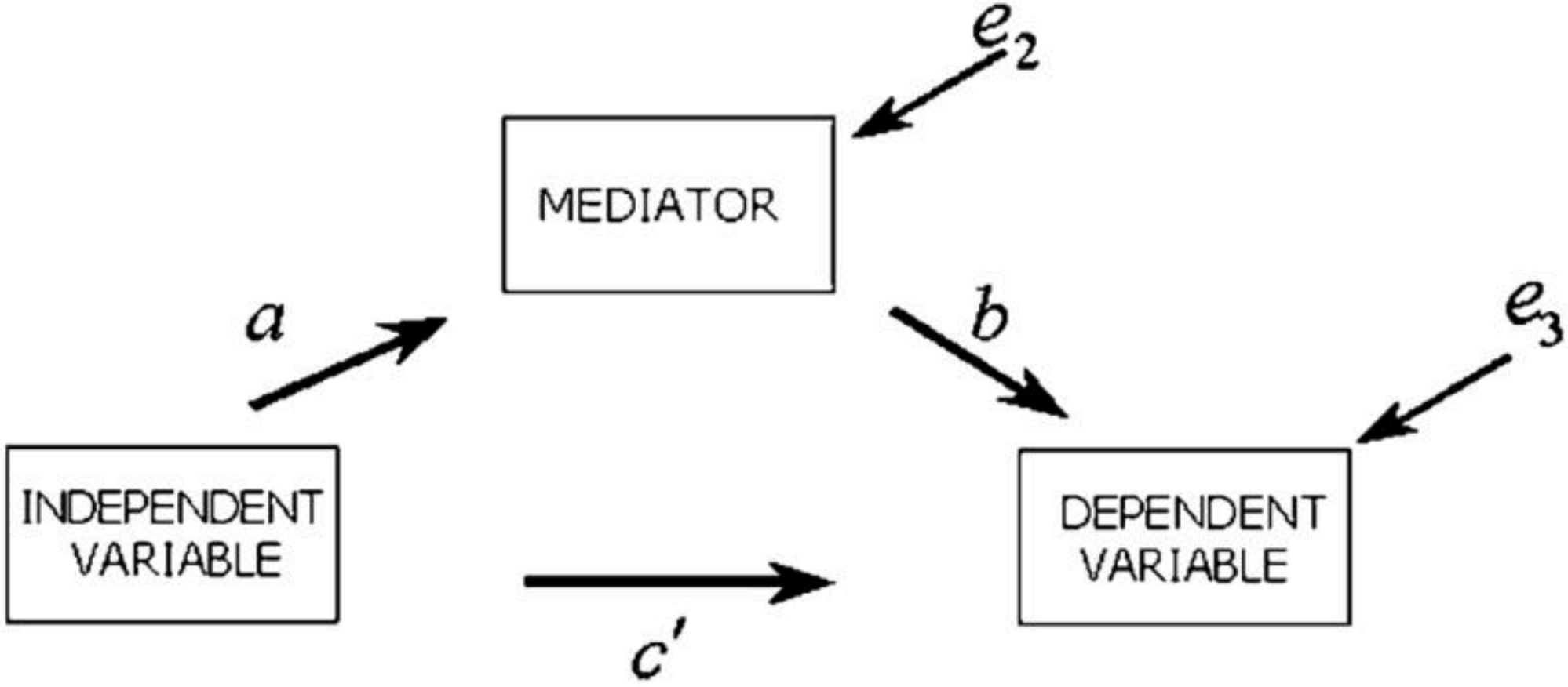
Asymptotically, AIC is equivalent to cross-validation

- assuming we properly specify the number of parameters

Crossvalidation requires fewer assumptions
(we don't have to explicitly measure model complexity)

Estimating indirect effects: "Causal mediation analysis"

It is common in psychology to use "mediation analysis" to assess whether the effect of one variable on another variable occurs through an indirect path via a third variable.



Testing of indirect effects is usually done via a set of individual regression analyses, but using SEM is preferred solution

Estimating indirect effects: "Causal mediation analysis"

```
%%R -i mediationDf

mediation_formula <- ' # regressions
             M ~ a * X
             Y ~ c * X + b * M

              # direct effect
              direct := c

              # indirect effect
              indirect := a*b

              # total effect
               total := c + (a*b)

             # variances and covariances
             X ~~ X
             Y ~~ Y
             M ~~ M

             # intercepts
             X ~ 1
             Y ~ 1
             M ~ 1
           '

# use bootstrap to obtain standard errors
model_sem = sem(mediation_formula, data=mediationDf, se='boot', bootstrap=500)
summary(model_sem, rsq=T)
```