Alignment in Growth Modeling: Neuroticism Example Using MIDUS

January 2, 2021, Last compiled on 15 April, 2021

Contents

Load Required Packages

```
library(here) # for setting working directory
library(modelsummary) # for descriptive statistics
library(lavaan) # for fitting SEM model
library(sirt) # for alignment algorithm
```

Download and Import Data

First, download the MIDUS data from the following links (registration on ICPSUR is needed):

- Wave I: https://www.icpsr.umich.edu/web/NACDA/studies/2760
- Wave II: https://www.icpsr.umich.edu/web/NACDA/studies/4652
- Wave III: https://www.icpsr.umich.edu/web/NACDA/studies/36346

For each zip file, extract the .rda data file from the "DS0001" folder. Specifically, the data file names are

- Wave I: "02760-0001-Data.rda"
- Wave II: "04652-0001-Data.rda"
- Wave III: "36346-0001-Data.rda"

```r
# Import data (assuming rda data files are in the data directory)
midus1_name <- load(here::here("data", "02760-0001-Data.rda"))
midus2_name <- load(here::here("data", "04652-0001-Data.rda"))
midus3_name <- load(here::here("data", "36346-0001-Data.rda"))
# Select MIDUS 1 variables (ID, age, neuroticism items)
midus1_df <- get(midus1_name)[, c(
  "M2ID", "A1PRAGE_2019",
  "A1SF4C", "A1SF4H",
  "A1SF4M", "A1SF4S"
)]
# Select MIDUS 2 variables (ID, neuroticism items)
midus2_df <- get(midus2_name)[, c(
  "M2ID",
  "B1SE6C", "B1SE6H",
  "B1SE6M", "B1SE6S"
)]
# Select MIDUS 2 variables (ID, neuroticism items)
midus3_df <- get(midus3_name)[, c(
  "M2ID",
  "C1SE6C", "C1SE6H",
  "C1SE6M", "C1SE6S"
)]
# Merge data
midus12_df <- merge(midus1_df,
  y = midus2_df,
  by.x = "M2ID", by.y = "M2ID",
  all.x = TRUE, all.y = FALSE
)
midus123_df <- merge(midus12_df, midus3_df,
  by.x = "M2ID", by.y = "M2ID",
  all.x = TRUE, all.y = FALSE
)
```

Subset Data

```r
# Extract Neuroticism variables, and rename
midus_neurotic <-
  with(
    midus123_df,
    data.frame(
      m2id = M2ID,
      age = A1PRAGE_2019,
      moody1 = A1SF4C,
      worry1 = A1SF4H,
      nervous1 = A1SF4M,
      calm1 = A1SF4S,
      moody2 = B1SE6C,
      worry2 = B1SE6H,
      nervous2 = B1SE6M,
      calm2 = B1SE6S,
      moody3 = C1SE6C,
      worry3 = C1SE6H,
```

	Mean	SD	Median	Min	Max
moody1	2.40	0.86	2.00	1.00	4.00
worry1	2.62	0.95	3.00	1.00	4.00
nervous1	2.24	0.94	2.00	1.00	4.00
calm1	2.11	0.81	2.00	1.00	4.00
moody2	2.18	0.82	2.00	1.00	4.00
worry2	2.38	0.91	2.00	1.00	4.00
nervous2	1.98	0.86	2.00	1.00	4.00
calm2	2.17	0.79	2.00	1.00	4.00
moody3	2.09	0.83	2.00	1.00	4.00
worry3	2.41	0.93	2.00	1.00	4.00
nervous3	2.05	0.89	2.00	1.00	4.00
calm3	2.14	0.81	2.00	1.00	4.00

```
      nervous3 = C1SE6M,
      calm3 = C1SE6S
    )
  )
# Subset participants aged 55 or above in Wave 1
midus_neurotic <- subset(midus_neurotic, age <= 40)
# Drop missing data
midus_neurotic <- na.omit(midus_neurotic)
# Convert items from factor to numeric
midus_neurotic[, 3:14] <-
  lapply(midus_neurotic[, 3:14], as.numeric)
# Recode so that higher scores indicate higher neuroticism
# (4 to 1, 3 to 2, 2 to 3, 1 to 4)
midus_neurotic[c(
  "moody1", "worry1", "nervous1",
  "moody2", "worry2", "nervous2",
  "moody3", "worry3", "nervous3"
)] <-
  5 - midus_neurotic[c(
    "moody1", "worry1", "nervous1",
    "moody2", "worry2", "nervous2",
    "moody3", "worry3", "nervous3"
  )]
```

Descriptive Statistics

```
modelsummary::datasummary(
  (moody1 + worry1 + nervous1 + calm1 +
    moody2 + worry2 + nervous2 + calm2 +
    moody3 + worry3 + nervous3 + calm3) ~
  (Mean + SD + Median + Min + Max),
  data = midus_neurotic
)
```

Longitudinal Configural Invariance Model

lavaan script

```
config_mod <- "
    eta1 =~ (lam11) * moody1 + (lam21) * worry1 + (lam31) * nervous1 +
            (lam41) * calm1
        eta2 =~ (lam12) * moody2 + (lam22) * worry2 + (lam32) * nervous2 +
                (lam42) * calm2
        eta3 =~ (lam13) * moody3 + (lam23) * worry3 + (lam33) * nervous3 +
                (lam43) * calm3
        # Measurement intercepts
    moody1 ~ (nu11) * 1
    worry1 ~ (nu21) * 1
    nervous1 ~ (nu31) * 1
    calm1 ~ (nu41) * 1
    moody2 ~ (nu12) * 1
    worry2 ~ (nu22) * 1
    nervous2 ~ (nu32) * 1
    calm2 ~ (nu42) * 1
    moody3 ~ (nu13) * 1
    worry3 ~ (nu23) * 1
    nervous3 ~ (nu33) * 1
    calm3 ~ (nu43) * 1
    # Unique factor covariances
    moody1 ~~ moody2 + moody3
    moody2 ~~ moody3
    worry1 ~~ worry2 + worry3
    worry2 ~~ worry3
    nervous1 ~~ nervous2 + nervous3
    nervous2 ~~ nervous3
    calm1 ~~ calm2 + calm3
    calm2 ~~ calm3
"
```

Fit model in lavaan

```
# Use lavaan::cfa() to fit a longitudinal configural model
config_fit <- cfa(
  config_mod,
  data = midus_neurotic,
  std.lv = TRUE # identify model by standardizing latent factors
)
# Uncomment to show model summary
# summary(config_fit, fit.measures = TRUE, standardized = TRUE)
```

Longitudinal Alignment Optimization

```r
# Extract loadings and intercepts for alignment
lam_mat <- lavInspect(config_fit, what = "est")$lambda
nu_vec <- lavInspect(config_fit, what = "est")$nu
# Put them into T x p matrices
num_items <- 4
num_waves <- 3
lam_config <- crossprod(lam_mat, rep(1, num_waves) %x% diag(num_items))
nu_config <- matrix(nu_vec, nrow = num_waves, ncol = num_items, byrow = TRUE)
# Add indicator names
colnames(lam_config) <- colnames(nu_config) <-
  c("moody", "worry", "nervous", "calm")
# Alignment optimization
aligned_pars <- sirt::invariance.alignment(
  lambda = lam_config,
  nu = nu_config,
  fixed = TRUE
)
```

The aligned loadings and intercepts are shown below:

	moody	worry	nervous	calm
Loadings				
Time1	0.443	0.786	0.767	0.321
Time2	0.446	0.798	0.737	0.356
Time3	0.462	0.774	0.753	0.359
Intercepts				
Time1	2.400	2.618	2.236	2.110
Time2	2.317	2.630	2.206	2.279
Time3	2.208	2.617	2.251	2.236

```r
# lam_resid <- lam_config - tcrossprod(aligned_pars$pars[ , 2],
#                                      aligned_pars$itempars.aligned$M.lambda)
# 1 - apply(lam_resid, 2, var) / apply(lam_config, 2, var)
# Function for dMACS
dmacs <- function(loadings, intercepts, pooled_item_sd,
                  latent_mean = 0, latent_var = 1) {
  dloading <- diff(loadings)
  dintercept <- diff(intercepts)
  integral <- dintercept^2 + 2 * dintercept * dloading * latent_mean +
    dloading^2 * (latent_var + latent_mean^2)
  sqrt(integral) / pooled_item_sd
}
# Use item SDs at first time point
item_sds_wave1 <-
  apply(
    midus_neurotic[c("moody1", "worry1", "nervous1", "calm1")],
    2, sd
  )
dmacs_pairwise <- function(loading_mat, intercept_mat, pooled_item_sd,
                           latent_mean = 0, latent_var = 1) {
  ngroups <- nrow(loading_mat)
```

```r
  pairs <- combn(ngroups, 2)
  out <- matrix(NA, nrow = ncol(pairs), ncol = ncol(loading_mat))
  for (i in seq_len(ncol(pairs))) {
    out[i, ] <- dmacs(loading_mat[pairs[, i], ],
      intercepts = intercept_mat[pairs[, i], ],
      pooled_item_sd,
      latent_mean,
      latent_var
    )
  }
  rownames(out) <- apply(pairs, 2, paste, collapse = " vs ")
  colnames(out) <- colnames(loading_mat)
  out
}
# All pairwise dMACS
dmacs_pairwise(aligned_pars$lambda.aligned,
  intercept_mat = aligned_pars$nu.aligned,
  pooled_item_sd = item_sds_wave1,
  latent_mean = 0,
  latent_var = 1
)
```

```
#>             moody      worry    nervous       calm
#> 1 vs 2 0.09636307 0.01727181 0.04560825 0.21354288
#> 1 vs 3 0.22540743 0.01255804 0.02208555 0.16245097
#> 2 vs 3 0.12932246 0.02829659 0.05069884 0.05368448
```

An equivalent configural model with aligned loadings:

```r
config_mod2 <- "
    # (First loadings fixed to alignment solution)
    eta1 =~ 0.4429136 * moody1 + (lam21) * worry1 + (lam31) * nervous1 +
            (lam41) * calm1
        eta2 =~ 0.4461272 * moody2 + (lam22) * worry2 + (lam32) * nervous2 +
                (lam42) * calm2
        eta3 =~ 0.4615046 * moody3 + (lam23) * worry3 + (lam33) * nervous3 +
                (lam43) * calm3
        # (First intercepts fixed to alignment solution)
    moody1 ~ 2.600240 * 1
    worry1 ~ (nu21) * 1
    nervous1 ~ (nu31) * 1
    calm1 ~ (nu41) * 1
    moody2 ~ 2.682681 * 1
    worry2 ~ (nu22) * 1
    nervous2 ~ (nu32) * 1
    calm2 ~ (nu42) * 1
    moody3 ~ 2.792330 * 1
    worry3 ~ (nu23) * 1
    nervous3 ~ (nu33) * 1
    calm3 ~ (nu43) * 1
    # Unique factor covariances
```

```
    moody1 ~~ moody2 + moody3
    moody2 ~~ moody3
    worry1 ~~ worry2 + worry3
    worry2 ~~ worry3
    nervous1 ~~ nervous2 + nervous3
    nervous2 ~~ nervous3
    calm1 ~~ calm2 + calm3
    calm2 ~~ calm3
    # Free latent means
    eta1 + eta2 + eta3 ~ NA*1
"
```

```
# Use lavaan::cfa() to fit a longitudinal configural model
config_fit2 <- cfa(
  config_mod2,
  data = midus_neurotic
)
# Uncomment to show model summary
# summary(config_fit2, fit.measures = TRUE, standardized = TRUE)
```

Compare model fit (they're equivalent)

```
anova(config_fit, config_fit2)
```

```
#> Chi-Squared Difference Test
#>
#>             Df   AIC   BIC  Chisq  Chisq diff Df diff Pr(>Chisq)
#> config_fit  39 21394 21635 74.492
#> config_fit2 39 21394 21635 74.492 -2.5836e-09       0
```

Alignment-Within-CFA (AwC) Approach for Growth Modeling

```
awc_growth_mod <- "
        # (First loadings fixed to alignment solution)
    eta1 =~ 0.4429136 * moody1 + (lam21) * worry1 + (lam31) * nervous1 +
            (lam41) * calm1
        eta2 =~ 0.4461272 * moody2 + (lam22) * worry2 + (lam32) * nervous2 +
                (lam42) * calm2
        eta3 =~ 0.4615046 * moody3 + (lam23) * worry3 + (lam33) * nervous3 +
                (lam43) * calm3
        # (First intercepts fixed to alignment solution)
    moody1 ~ 2.399760 * 1
    worry1 ~ (nu21) * 1
    nervous1 ~ (nu31) * 1
    calm1 ~ (nu41) * 1
    moody2 ~ 2.317319 * 1
    worry2 ~ (nu22) * 1
    nervous2 ~ (nu32) * 1
    calm2 ~ (nu42) * 1
    moody3 ~ 2.207670 * 1
```

```
    worry3 ~ (nu23) * 1
    nervous3 ~ (nu33) * 1
    calm3 ~ (nu43) * 1
    # Unique factor covariances
    moody1 ~~ moody2 + moody3
    moody2 ~~ moody3
    worry1 ~~ worry2 + worry3
    worry2 ~~ worry3
    nervous1 ~~ nervous2 + nervous3
    nervous2 ~~ nervous3
    calm1 ~~ calm2 + calm3
    calm2 ~~ calm3
    # Linear Growth Model
    i =~ 1 * eta1 + 1 * eta2 + 1 * eta3
    s =~ 0 * eta1 + 1 * eta2 + 2 * eta3
    # Variance-covariances of intercepts and slopes
    i ~~ i
    s ~~ s
    i ~~ s
    # Means of level and slope
    i ~ 1
    s ~ 1
    # Fixed disturbances of latent outcomes to zero
    eta1 ~ 0 * 1
    eta2 ~ 0 * 1
    eta3 ~ 0 * 1
"
```

Fit model in lavaan

```
# Use lavaan::sem() to fit a second-order growth model
awc_growth_fit <- sem(
  awc_growth_mod,
  data = midus_neurotic
)
# Uncomment to show model summary
# summary(awc_growth_fit, fit.measures = TRUE, standardized = TRUE)
```

Parameter Estimates

```
lavaan::parameterEstimates(awc_growth_fit) %>%
  subset(
    lhs %in% c("i", "s") & substr(rhs, 1, 3) != "eta",
    -label
  ) %>%
  knitr::kable(format = "simple", digits = 3L)
```

	lhs	op	rhs	est	se	z	pvalue	ci.lower	ci.upper
43	i	~~	i	0.667	0.107	6.224	0.000	0.457	0.877
44	s	~~	s	0.045	0.045	1.006	0.314	-0.043	0.134
45	i	~~	s	-0.034	0.051	-0.675	0.499	-0.134	0.066
46	i	~1		-0.070	0.063	-1.118	0.264	-0.194	0.053
47	s	~1		-0.121	0.032	-3.791	0.000	-0.184	-0.059

Sensitivity Check: Identification Using Second Indicator

```
awc_growth_mod2 <- "
    # (First loadings fixed to alignment solution)
    eta1 =~ NA * moody1 + (lam11) * moody1 + 0.7855913 * worry1 +
            (lam31) * nervous1 + (lam41) * calm1
        eta2 =~ NA * moody2 + (lam12) * moody2 + 0.7975706 * worry2 +
                (lam32) * nervous2 + (lam42) * calm2
        eta3 =~ NA * moody3 + (lam13) * moody3 + 0.7736944 * worry3 +
                (lam33) * nervous3 + (lam43) * calm3
        # (First intercepts fixed to alignment solution)
    moody1 ~ (nu11) * 1
    worry1 ~ 2.618247 * 1
    nervous1 ~ (nu31) * 1
    calm1 ~ (nu41) * 1
    moody2 ~ (nu12) * 1
    worry2 ~ 2.629519 * 1
    nervous2 ~ (nu32) * 1
    calm2 ~ (nu42) * 1
    moody3 ~ (nu13) * 1
    worry3 ~ 2.617024 * 1
    nervous3 ~ (nu33) * 1
    calm3 ~ (nu43) * 1
    # Unique factor covariances
    moody1 ~~ moody2 + moody3
    moody2 ~~ moody3
    worry1 ~~ worry2 + worry3
    worry2 ~~ worry3
    nervous1 ~~ nervous2 + nervous3
    nervous2 ~~ nervous3
    calm1 ~~ calm2 + calm3
    calm2 ~~ calm3
    # Linear Growth Model
    i =~ 1 * eta1 + 1 * eta2 + 1 * eta3
    s =~ 0 * eta1 + 1 * eta2 + 2 * eta3
    # Variance-covariances of intercepts and slopes
    i ~~ i
    s ~~ s
    i ~~ s
    # Means of level and slope
    i ~ 1
    s ~ 1
    # Fixed disturbances of latent outcomes to zero
    eta1 ~ 0 * 1
```

```
    eta2 ~ 0 * 1
    eta3 ~ 0 * 1
"
```

Fit model in lavaan

```
# Use lavaan::sem() to fit a second-order growth model
awc_growth_fit2 <- sem(
  awc_growth_mod2,
  data = midus_neurotic
)
# Uncomment to show model summary
# summary(awc_growth_fit, fit.measures = TRUE, standardized = TRUE)
```

Parameter Estimates

```
lavaan::parameterEstimates(awc_growth_fit2) %>%
  subset(
    lhs %in% c("i", "s") & substr(rhs, 1, 3) != "eta",
    -label
  ) %>%
  knitr::kable(format = "simple", digits = 3L)
```

	lhs	op	rhs	est	se	z	pvalue	ci.lower	ci.upper
43	i	~~	i	0.660	0.077	8.601	0.000	0.510	0.811
44	s	~~	s	0.042	0.035	1.225	0.221	-0.025	0.110
45	i	~~	s	-0.031	0.040	-0.782	0.434	-0.109	0.047
46	i	~1		-0.074	0.040	-1.877	0.060	-0.152	0.003
47	s	~1		-0.124	0.021	-5.772	0.000	-0.166	-0.082

Note the smaller standard errors when using item 2 as the reference indicator, which has larger loadings.

Traditional Invariance Testing

Weak and Strong Invariance

```
# Weak invariance model
weak_mod <- "
    eta1 =~ NA * moody1 + (lam1) * moody1 + (lam2) * worry1 + (lam3) * nervous1 +
            (lam4) * calm1
        eta2 =~ NA * moody2 + (lam1) * moody2 + (lam2) * worry2 + (lam3) * nervous2 +
                (lam4) * calm2
        eta3 =~ NA * moody3 + (lam1) * moody3 + (lam2) * worry3 + (lam3) * nervous3 +
                (lam4) * calm3
        # Measurement intercepts
```

```
    moody1 ~ (nu11) * 1
    worry1 ~ (nu21) * 1
    nervous1 ~ (nu31) * 1
    calm1 ~ (nu41) * 1
    moody2 ~ (nu12) * 1
    worry2 ~ (nu22) * 1
    nervous2 ~ (nu32) * 1
    calm2 ~ (nu42) * 1
    moody3 ~ (nu13) * 1
    worry3 ~ (nu23) * 1
    nervous3 ~ (nu33) * 1
    calm3 ~ (nu43) * 1
    # Unique factor covariances
    moody1 ~~ moody2 + moody3
    moody2 ~~ moody3
    worry1 ~~ worry2 + worry3
    worry2 ~~ worry3
    nervous1 ~~ nervous2 + nervous3
    nervous2 ~~ nervous3
    calm1 ~~ calm2 + calm3
    calm2 ~~ calm3
    # First factor variance to 1
    eta1 ~~ 1 * eta1
"
```

```
# Use lavaan::cfa() to fit a longitudinal configural model
weak_fit <- cfa(
  weak_mod,
  data = midus_neurotic
)
# Uncomment to show model summary
# summary(weak_fit, fit.measures = TRUE, standardized = TRUE)
```

```
# Strong invariance model
strong_mod <- "
    eta1 =~ NA * moody1 + (lam1) * moody1 + (lam2) * worry1 + (lam3) * nervous1 +
            (lam4) * calm1
        eta2 =~ NA * moody2 + (lam1) * moody2 + (lam2) * worry2 + (lam3) * nervous2 +
                (lam4) * calm2
        eta3 =~ NA * moody3 + (lam1) * moody3 + (lam2) * worry3 + (lam3) * nervous3 +
                (lam4) * calm3
        # Measurement intercepts
    moody1 ~ (nu1) * 1
    worry1 ~ (nu2) * 1
    nervous1 ~ (nu3) * 1
    calm1 ~ (nu4) * 1
    moody2 ~ (nu1) * 1
    worry2 ~ (nu2) * 1
    nervous2 ~ (nu3) * 1
    calm2 ~ (nu4) * 1
    moody3 ~ (nu1) * 1
    worry3 ~ (nu2) * 1
    nervous3 ~ (nu3) * 1
```

```
    calm3 ~ (nu4) * 1
    # Unique factor covariances
    moody1 ~~ moody2 + moody3
    moody2 ~~ moody3
    worry1 ~~ worry2 + worry3
    worry2 ~~ worry3
    nervous1 ~~ nervous2 + nervous3
    nervous2 ~~ nervous3
    calm1 ~~ calm2 + calm3
    calm2 ~~ calm3
    # First factor variance and mean to 1 and 0
    eta1 ~~ 1 * eta1
    eta1 ~ 0
    # Other factor means to free
    eta2 ~ NA * 1
    eta3 ~ NA * 1
"
```

```
# Use lavaan::cfa() to fit a longitudinal configural model
strong_fit <- cfa(
  strong_mod,
  data = midus_neurotic
)
# Uncomment to show model summary
# summary(strong_fit, fit.measures = TRUE, standardized = TRUE)
```

```
fit_tab <- lapply(
  list("Configural" = config_fit,
       "Weak" = weak_fit,
       "Strong" = strong_fit),
  fitmeasures,
  fit.measures = c(
    "chisq", "df", "rmsea",
    "rmsea.ci.lower", "rmsea.ci.upper",
    "cfi", "tli", "srmr"
  )
)
fit_tab <- do.call(rbind, fit_tab)
knitr::kable(fit_tab, format = "simple", digits = 3)
```

	chisq	df	rmsea	rmsea.ci.lower	rmsea.ci.upper	cfi	tli	srmr
Configural	74.492	39	0.033	0.021	0.044	0.991	0.986	0.038
Weak	77.939	45	0.030	0.018	0.040	0.992	0.988	0.040
Strong	170.307	51	0.053	0.044	0.062	0.971	0.963	0.051

Using modification indices, two items were identified to have noninvariant intercepts: `moody` in Wave 3 and `calm` in Wave 1.

```
modindices(strong_fit, sort. = TRUE, free.remove = FALSE, min = 10)
```

```
#>        lhs op      rhs     mi    epc sepc.lv sepc.all sepc.nox
```

```
#> 21  moody3 ~1          35.496 -0.129  -0.129   -0.154   -0.154
#> 16   calm1 ~1          29.011 -0.124  -0.124   -0.155   -0.155
#> 13  moody1 ~1          25.260  0.113   0.113    0.131    0.131
#> 20   calm2 ~1          15.491  0.086   0.086    0.111    0.111
#> 149 worry3 ~~ nervous3 13.894  0.086   0.086    0.277    0.277
#> 86    eta2 =~   moody3 13.344  0.097   0.088    0.105    0.105
```

Partial Strong Invariance Model

```
pstrong_growth_mod <- "
    eta1 =~ NA * moody1 + (lam1) * moody1 + (lam2) * worry1 + (lam3) * nervous1 +
            (lam4) * calm1
        eta2 =~ NA * moody2 + (lam1) * moody2 + (lam2) * worry2 + (lam3) * nervous2 +
                (lam4) * calm2
        eta3 =~ NA * moody3 + (lam1) * moody3 + (lam2) * worry3 + (lam3) * nervous3 +
                (lam4) * calm3
        # Measurement intercepts
    moody1 ~ (nu1) * 1
    worry1 ~ (nu2) * 1
    nervous1 ~ (nu3) * 1
    calm1 ~ (nu41) * 1
    moody2 ~ (nu1) * 1
    worry2 ~ (nu2) * 1
    nervous2 ~ (nu3) * 1
    calm2 ~ (nu4) * 1
    moody3 ~ (nu13) * 1
    worry3 ~ (nu2) * 1
    nervous3 ~ (nu3) * 1
    calm3 ~ (nu4) * 1
    # Unique factor covariances
    moody1 ~~ moody2 + moody3
    moody2 ~~ moody3
    worry1 ~~ worry2 + worry3
    worry2 ~~ worry3
    nervous1 ~~ nervous2 + nervous3
    nervous2 ~~ nervous3
    calm1 ~~ calm2 + calm3
    calm2 ~~ calm3
    # Linear Growth Model
    i =~ 1 * eta1 + 1 * eta2 + 1 * eta3
    s =~ 0 * eta1 + 1 * eta2 + 2 * eta3
    # Variance-covariances of intercepts and slopes
    i ~~ phi1 * i
    s ~~ s
    i ~~ s
    # Means of level (fixed to zero for identification) and slope
    i ~ 0 * 1
    s ~ 1
    # Fixed disturbances of latent outcomes to zero
    eta1 ~ 0 * 1
    eta2 ~ 0 * 1
    eta3 ~ 0 * 1
```

```
    # Constrain total variance at Time 0 to 1
    eta1 ~~ psi1 * eta1
    psi1 + phi1 == 1
"
# Use lavaan::sem() to fit a second-order growth model
pstrong_growth_fit <- sem(
  pstrong_growth_mod,
  data = midus_neurotic
)
```

Table comparing AwC growth model and partial strong invariance 2nd order growth model

The growth parameter estimates of the two models are very similar.

```
msummary(
  list(
    "AwC growth" = awc_growth_fit2,
    "Partial strong invariance" = pstrong_growth_fit
  ),
  output = "markdown",
  statistic = "conf.int",
  coef_map = c(
    "i ~1 " = "Mean(Level)",
    "s ~1 " = "Mean(Slope)",
    "i ~~ i" = "Var(Level)",
    "s ~~ s" = "Var(Slope)",
    "i ~~ s" = "Cov(Level, Slope)"
  )
)
```

	AwC growth	Partial strong invariance
Mean(Level)	-0.074	0.000
	[-0.152, 0.003]	[0.000, 0.000]
Mean(Slope)	-0.124	-0.121
	[-0.166, -0.082]	[-0.157, -0.086]
Var(Level)	0.660	0.632
	[0.510, 0.811]	[0.520, 0.744]
Var(Slope)	0.042	0.030
	[-0.025, 0.110]	[-0.029, 0.089]
Cov(Level, Slope)	-0.031	-0.018
	[-0.109, 0.047]	[-0.084, 0.048]
Num.Obs.	833	833
AIC	21419.4	21445.8
BIC	21655.7	21634.8
agfi	0.989	0.987
cfi	0.985	0.976
chisq	101.704	148.053
converged	TRUE	TRUE
estimator	ML	ML
missing_method	listwise	listwise
nexcluded	0.000	0.000
ngroups	1.000	1.000

	AwC growth	Partial strong invariance
norig	833.000	833.000
npar	50.000	40.000
rmsea	0.043	0.049
rmsea.conf.high	0.053	0.058
srmr	0.041	0.047
tli	0.976	0.969

Version Information

```
sessionInfo()
```

```
#> R version 4.0.5 (2021-03-31)
#> Platform: x86_64-pc-linux-gnu (64-bit)
#> Running under: Ubuntu 20.04.2 LTS
#>
#> Matrix products: default
#> BLAS/LAPACK: /opt/OpenBLAS/lib/libopenblas-r0.3.13.so
#>
#> locale:
#>  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
#>  [3] LC_TIME=en_US.UTF-8        LC_COLLATE=en_US.UTF-8
#>  [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
#>  [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
#>  [9] LC_ADDRESS=C               LC_TELEPHONE=C
#> [11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
#>
#> attached base packages:
#> [1] stats     graphics  grDevices utils     datasets  methods   base
#>
#> other attached packages:
#> [1] kableExtra_1.3.4   magrittr_2.0.1     sirt_3.9-4         lavaan_0.6-8
#> [5] modelsummary_0.6.6 here_1.0.1
#>
#> loaded via a namespace (and not attached):
#>  [1] tinytex_0.31        tidyselect_1.1.0    xfun_0.22
#>  [4] tinylabels_0.2.1    purrr_0.3.4         colorspace_2.0-0
#>  [7] generics_0.1.0      vctrs_0.3.7         htmltools_0.5.1.1
#> [10] stats4_4.0.5        viridisLite_0.4.0   yaml_2.2.1
#> [13] utf8_1.2.1          rlang_0.4.10        pillar_1.6.0
#> [16] DBI_1.1.1           glue_1.4.2          lifecycle_1.0.0
#> [19] stringr_1.4.0       munsell_0.5.0       CDM_7.5-15
#> [22] rvest_1.0.0         mvtnorm_1.1-1       evaluate_0.14
#> [25] papaja_0.1.0.9997-1 knitr_1.32          fansi_0.4.2
#> [28] highr_0.8           broom_0.7.6         Rcpp_1.0.6
#> [31] scales_1.1.1        backports_1.2.1     checkmate_2.0.0
#> [34] webshot_0.5.2       tmvnsim_1.0-2       systemfonts_1.0.1
#> [37] polycor_0.7-10      mnormt_2.0.2        digest_0.6.27
#> [40] stringi_1.5.3       dplyr_1.0.5         rprojroot_2.0.2
#> [43] tools_4.0.5         tibble_3.1.0        tidyr_1.1.3
#> [46] crayon_1.4.1        pbivnorm_0.6.0      TAM_3.5-19
#> [49] pkgconfig_2.0.3     MASS_7.3-53.1       ellipsis_0.3.1
#> [52] xml2_1.3.2          assertthat_0.2.1    rmarkdown_2.7
#> [55] svglite_2.0.0       httr_1.4.2          rstudioapi_0.13
#> [58] R6_2.5.0            tables_0.9.6        compiler_4.0.5
```