Homework 4 (SEM)

Leah Schultz

11/2/2017

Chapter 7: SEM

```
library(lme4)
library(ggplot2)
library(lavaan)
library(semPlot)
library(tidyverse)
oysup <- read.csv("~/Desktop/oysup_teacher_self.csv")
```

1) Fit a measurement model to your constructs at one time point. Try out the different types of scaling discussed in class. What changes? What stays the same?

```
# Marker variable
mod.1 <- 'neuro_t =~ TPER7_08 + TPER7_10 + TPER7_17R + TPER7_12R + TPER7_13R'
fit.1 <- cfa(mod.1, data=oysup)
summary(fit.1, fit.measures=TRUE)
```

```
## lavaan (0.6-1.1141) converged normally after  22 iterations
##
##                                                   Used       Total
##   Number of observations                           523        1074
##
##   Estimator                                         ML
##   Model Fit Test Statistic                      65.896
##   Degrees of freedom                                 5
##   P-value (Chi-square)                           0.000
##
## Model test baseline model:
##
##   Minimum Function Test Statistic              983.162
##   Degrees of freedom                                10
##   P-value                                        0.000
##
## User model versus baseline model:
##
##   Comparative Fit Index (CFI)                    0.937
##   Tucker-Lewis Index (TLI)                       0.875
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)              -3604.684
##   Loglikelihood unrestricted model (H1)      -3571.736
##
```

```
##   Number of free parameters                         10
##   Akaike (AIC)                                7229.368
##   Bayesian (BIC)                              7271.964
##   Sample-size adjusted Bayesian (BIC)         7240.222
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.153
##   90 Percent Confidence Interval          0.121  0.186
##   P-value RMSEA <= 0.05                          0.000
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.046
##
## Parameter Estimates:
##
##   Information                                 Expected
##   Standard Errors                             Standard
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   neuro_t =~
##     TPER7_08          1.000
##     TPER7_10          0.831    0.071   11.622    0.000
##     TPER7_17R         1.176    0.082   14.418    0.000
##     TPER7_12R         1.193    0.080   14.935    0.000
##     TPER7_13R         0.641    0.061   10.488    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .TPER7_08          1.031    0.071   14.564    0.000
##    .TPER7_10          0.824    0.056   14.793    0.000
##    .TPER7_17R         0.514    0.046   11.125    0.000
##    .TPER7_12R         0.273    0.037    7.365    0.000
##    .TPER7_13R         0.704    0.046   15.226    0.000
##     neuro_t           0.675    0.089    7.540    0.000
```

```r
# Fixed factor
mod.2 <- 'neuro_t =~ TPER7_08 + TPER7_10 + TPER7_17R + TPER7_12R + TPER7_13R'
fit.2 <- cfa(mod.2, std.lv = T, data=oysup)
summary(fit.2, fit.measures=TRUE)
```

```
## lavaan (0.6-1.1141) converged normally after  14 iterations
##
##                                                   Used       Total
##   Number of observations                           523        1074
##
##   Estimator                                         ML
##   Model Fit Test Statistic                      65.896
##   Degrees of freedom                                 5
##   P-value (Chi-square)                           0.000
##
## Model test baseline model:
##
```

```
##   Minimum Function Test Statistic              983.162
##   Degrees of freedom                                10
##   P-value                                        0.000
##
## User model versus baseline model:
##
##   Comparative Fit Index (CFI)                    0.937
##   Tucker-Lewis Index (TLI)                       0.875
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)              -3604.684
##   Loglikelihood unrestricted model (H1)      -3571.736
##
##   Number of free parameters                         10
##   Akaike (AIC)                                7229.368
##   Bayesian (BIC)                              7271.964
##   Sample-size adjusted Bayesian (BIC)         7240.222
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.153
##   90 Percent Confidence Interval          0.121  0.186
##   P-value RMSEA <= 0.05                          0.000
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.046
##
## Parameter Estimates:
##
##   Information                                 Expected
##   Standard Errors                             Standard
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   neuro_t =~
##     TPER7_08          0.821    0.054   15.080    0.000
##     TPER7_10          0.682    0.048   14.255    0.000
##     TPER7_17R         0.966    0.047   20.715    0.000
##     TPER7_12R         0.980    0.042   23.600    0.000
##     TPER7_13R         0.526    0.043   12.302    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .TPER7_08          1.031    0.071   14.564    0.000
##    .TPER7_10          0.824    0.056   14.793    0.000
##    .TPER7_17R         0.514    0.046   11.125    0.000
##    .TPER7_12R         0.273    0.037    7.365    0.000
##    .TPER7_13R         0.704    0.046   15.226    0.000
##     neuro_t           1.000
```

```r
# Effects coding
mod.3 <- 'neuro_t =~ NA*TPER7_08 + L1*TPER7_08 + L2*TPER7_10 + L3*TPER7_17R + L4*TPER7_12R + L5*TPER7_1
```

```r
L1 == 5 - L2 - L3 - L4 - L5'
fit.3 <- cfa(mod.3, data=oysup)
summary(fit.3, fit.measures=TRUE)
```

```
## lavaan (0.6-1.1141) converged normally after  19 iterations
##
##                                                   Used       Total
##   Number of observations                           523        1074
##
##   Estimator                                         ML
##   Model Fit Test Statistic                      65.896
##   Degrees of freedom                                 5
##   P-value (Chi-square)                           0.000
##
## Model test baseline model:
##
##   Minimum Function Test Statistic              983.162
##   Degrees of freedom                                10
##   P-value                                        0.000
##
## User model versus baseline model:
##
##   Comparative Fit Index (CFI)                    0.937
##   Tucker-Lewis Index (TLI)                       0.875
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)              -3604.684
##   Loglikelihood unrestricted model (H1)      -3571.736
##
##   Number of free parameters                         10
##   Akaike (AIC)                                7229.368
##   Bayesian (BIC)                              7271.964
##   Sample-size adjusted Bayesian (BIC)         7240.222
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.153
##   90 Percent Confidence Interval          0.121  0.186
##   P-value RMSEA <= 0.05                          0.000
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.046
##
## Parameter Estimates:
##
##   Information                                 Expected
##   Standard Errors                             Standard
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   neuro_t =~
##     TPER7_08  (L1)    1.033    0.052   19.929    0.000
```

```
##     TPER7_10  (L2)    0.858    0.048   17.957    0.000
##     TPER7_17R (L3)    1.215    0.043   28.410    0.000
##     TPER7_12R (L4)    1.232    0.040   30.723    0.000
##     TPER7_13R (L5)    0.662    0.045   14.666    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .TPER7_08          1.031    0.071   14.564    0.000
##    .TPER7_10          0.824    0.056   14.793    0.000
##    .TPER7_17R         0.514    0.046   11.125    0.000
##    .TPER7_12R         0.273    0.037    7.365    0.000
##    .TPER7_13R         0.704    0.046   15.226    0.000
##     neuro_t           0.632    0.047   13.332    0.000
##
## Constraints:
##                                                |Slack|
##     L1 - (5-L2-L3-L4-L5)                         0.000
```

Using the marker variable approach, the loading of the first factor onto the latent variable is fixed to 1, and the other loadings relative to this range from .64 to 1.19. The CFI is .94. Using the fixed factor approach, the item loadings change – they range from .53 to .98. The residual variances of the items stay the same, but of course the variance of the latent neuroticism variable changes to 1. Using effects coding, the item loadings onto the latent variable change once more, ranging from .66 to 1.23, and the variance of the latent variable changes back to being freely estimated (.63).

2) What do the fit statistics say about your latent variable? Good/bad? Is your latent variable just identified/saturated, under-identified, or over-identified?

```r
fitMeasures(fit.1)
```

```
##                npar                fmin               chisq
##              10.000               0.063              65.896
##                  df              pvalue      baseline.chisq
##               5.000               0.000             983.162
##         baseline.df     baseline.pvalue                 cfi
##              10.000               0.000               0.937
##                 tli                nnfi                 rfi
##               0.875               0.875               0.866
##                 nfi                pnfi                 ifi
##               0.933               0.466               0.938
##                 rni                logl   unrestricted.logl
##               0.937           -3604.684           -3571.736
##                 aic                 bic              ntotal
##            7229.368            7271.964             523.000
##                bic2               rmsea      rmsea.ci.lower
##            7240.222               0.153               0.121
##      rmsea.ci.upper        rmsea.pvalue                 rmr
##               0.186               0.000               0.062
##          rmr_nomean                srmr        srmr_bentler
##               0.062               0.046               0.046
## srmr_bentler_nomean         srmr_bollen  srmr_bollen_nomean
##               0.046               0.046               0.046
##          srmr_mplus   srmr_mplus_nomean               cn_05
```

```
##              0.046              0.046             88.863
##              cn_01                gfi               agfi
##            120.735              0.954              0.863
##               pgfi                mfi               ecvi
##              0.318              0.943              0.164
```

Across all approaches (regardless of scaling), the CFI is .94 (great!), while the RMSEA is .15 (poor). There are 5 degrees of freedom, meaning that the model is over-identified - yay.

3) Fit a longitudinal CFA model where you a) first correlate your latent factors across time and then b) a second model that predicts later times by a prevous time (i.e., auto-regressive; t1 -> t2 -> t3). What are your conclusions? How does one differ from the other?

```
# Residuals correlated over time
mod.4 <- 'neuro_t1 =~ TPER7_08 + TPER7_10 + TPER7_17R + TPER7_12R + TPER7_13R
neuro_t2 =~ TPER8_08 + TPER8_10 + TPER8_17R + TPER8_12R + TPER8_13R
neuro_t3 =~ TPER9_08 + TPER9_10 + TPER9_17R + TPER9_12R + TPER9_13R
neuro_t4 =~ TPER10_08 + TPER10_10 + TPER10_17R + TPER10_12R + TPER10_13R

TPER7_08 ~~ TPER8_08 + TPER9_08 + TPER10_08
TPER8_08 ~~ TPER9_08 + TPER10_08
TPER9_08 ~~ TPER10_08

TPER7_10 ~~ TPER8_10 + TPER9_10 + TPER10_10
TPER8_10 ~~ TPER9_10 + TPER10_10
TPER9_10 ~~ TPER10_10

TPER7_17R ~~ TPER8_17R + TPER9_17R + TPER10_17R
TPER8_17R ~~ TPER9_17R + TPER10_17R
TPER9_17R ~~ TPER10_17R

TPER7_12R ~~ TPER8_12R + TPER9_12R + TPER10_12R
TPER8_12R ~~ TPER9_12R + TPER10_12R
TPER9_12R ~~ TPER10_12R

TPER7_13R ~~ TPER8_13R + TPER9_13R + TPER10_13R
TPER8_13R ~~ TPER9_13R + TPER10_13R
TPER9_13R ~~ TPER10_13R'
fit.4 <- cfa(mod.4, data=oysup)
summary(fit.4, fit.measures=TRUE)
```

```
## lavaan (0.6-1.1141) converged normally after  49 iterations
##
##                                                   Used       Total
##   Number of observations                           134        1074
##
##   Estimator                                         ML
##   Model Fit Test Statistic                     245.340
##   Degrees of freedom                               134
##   P-value (Chi-square)                           0.000
##
## Model test baseline model:
```

```
##
##   Minimum Function Test Statistic             1338.250
##   Degrees of freedom                               190
##   P-value                                        0.000
##
## User model versus baseline model:
##
##   Comparative Fit Index (CFI)                    0.903
##   Tucker-Lewis Index (TLI)                       0.863
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)              -3496.104
##   Loglikelihood unrestricted model (H1)      -3373.434
##
##   Number of free parameters                         76
##   Akaike (AIC)                                7144.208
##   Bayesian (BIC)                              7364.443
##   Sample-size adjusted Bayesian (BIC)         7124.037
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.079
##   90 Percent Confidence Interval          0.063  0.094
##   P-value RMSEA <= 0.05                          0.002
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.071
##
## Parameter Estimates:
##
##   Information                                 Expected
##   Standard Errors                             Standard
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   neuro_t1 =~
##     TPER7_08          1.000
##     TPER7_10          0.653    0.106    6.130    0.000
##     TPER7_17R         1.105    0.114    9.697    0.000
##     TPER7_12R         1.086    0.102   10.671    0.000
##     TPER7_13R         0.602    0.089    6.738    0.000
##   neuro_t2 =~
##     TPER8_08          1.000
##     TPER8_10          0.949    0.126    7.509    0.000
##     TPER8_17R         1.027    0.124    8.266    0.000
##     TPER8_12R         0.893    0.106    8.431    0.000
##     TPER8_13R         0.654    0.111    5.874    0.000
##   neuro_t3 =~
##     TPER9_08          1.000
##     TPER9_10          0.896    0.121    7.430    0.000
##     TPER9_17R         0.844    0.120    7.030    0.000
##     TPER9_12R         0.916    0.117    7.817    0.000
```

```
##     TPER9_13R         0.438    0.097    4.538    0.000
##   neuro_t4 =~
##     TPER10_08         1.000
##     TPER10_10         0.602    0.124    4.849    0.000
##     TPER10_17R        0.963    0.133    7.256    0.000
##     TPER10_12R        1.004    0.134    7.489    0.000
##     TPER10_13R        0.615    0.116    5.307    0.000
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##  .TPER7_08 ~~
##    .TPER8_08         -0.006    0.069   -0.081    0.936
##    .TPER9_08          0.019    0.071    0.275    0.784
##    .TPER10_08         0.243    0.084    2.899    0.004
##  .TPER8_08 ~~
##    .TPER9_08          0.077    0.073    1.059    0.290
##    .TPER10_08         0.031    0.082    0.371    0.711
##  .TPER9_08 ~~
##    .TPER10_08         0.136    0.086    1.579    0.114
##  .TPER7_10 ~~
##    .TPER8_10          0.015    0.069    0.224    0.823
##    .TPER9_10         -0.041    0.077   -0.537    0.591
##    .TPER10_10         0.117    0.079    1.485    0.137
##  .TPER8_10 ~~
##    .TPER9_10         -0.006    0.066   -0.086    0.931
##    .TPER10_10         0.042    0.067    0.635    0.525
##  .TPER9_10 ~~
##    .TPER10_10         0.117    0.075    1.552    0.121
##  .TPER7_17R ~~
##    .TPER8_17R        -0.007    0.051   -0.134    0.893
##    .TPER9_17R         0.023    0.065    0.360    0.719
##    .TPER10_17R        0.021    0.050    0.418    0.676
##  .TPER8_17R ~~
##    .TPER9_17R         0.072    0.062    1.163    0.245
##    .TPER10_17R        0.083    0.046    1.798    0.072
##  .TPER9_17R ~~
##    .TPER10_17R        0.064    0.059    1.075    0.282
##  .TPER7_12R ~~
##    .TPER8_12R        -0.009    0.033   -0.287    0.774
##    .TPER9_12R         0.069    0.048    1.421    0.155
##    .TPER10_12R       -0.013    0.035   -0.377    0.706
##  .TPER8_12R ~~
##    .TPER9_12R         0.156    0.052    3.015    0.003
##    .TPER10_12R       -0.010    0.035   -0.293    0.769
##  .TPER9_12R ~~
##    .TPER10_12R        0.138    0.053    2.581    0.010
##  .TPER7_13R ~~
##    .TPER8_13R         0.068    0.063    1.087    0.277
##    .TPER9_13R         0.193    0.068    2.838    0.005
##    .TPER10_13R        0.160    0.062    2.598    0.009
##  .TPER8_13R ~~
##    .TPER9_13R         0.267    0.076    3.523    0.000
##    .TPER10_13R        0.137    0.067    2.052    0.040
##  .TPER9_13R ~~
```

```
##    .TPER10_13R        0.136    0.070    1.938    0.053
##   neuro_t1 ~~
##     neuro_t2          0.397    0.094    4.206    0.000
##     neuro_t3          0.454    0.106    4.276    0.000
##     neuro_t4          0.368    0.098    3.755    0.000
##   neuro_t2 ~~
##     neuro_t3          0.292    0.091    3.226    0.001
##     neuro_t4          0.244    0.079    3.082    0.002
##   neuro_t3 ~~
##     neuro_t4          0.288    0.092    3.131    0.002
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .TPER7_08          0.696    0.097    7.161    0.000
##    .TPER7_10          0.905    0.115    7.876    0.000
##    .TPER7_17R         0.507    0.080    6.339    0.000
##    .TPER7_12R         0.174    0.050    3.453    0.001
##    .TPER7_13R         0.635    0.081    7.812    0.000
##    .TPER8_08          0.706    0.101    6.976    0.000
##    .TPER8_10          0.580    0.085    6.844    0.000
##    .TPER8_17R         0.420    0.070    5.989    0.000
##    .TPER8_12R         0.285    0.050    5.651    0.000
##    .TPER8_13R         0.748    0.097    7.711    0.000
##    .TPER9_08          0.653    0.112    5.818    0.000
##    .TPER9_10          0.698    0.109    6.391    0.000
##    .TPER9_17R         0.785    0.116    6.783    0.000
##    .TPER9_12R         0.685    0.108    6.319    0.000
##    .TPER9_13R         0.849    0.108    7.853    0.000
##    .TPER10_08         1.016    0.139    7.287    0.000
##    .TPER10_10         0.842    0.108    7.805    0.000
##    .TPER10_17R        0.390    0.068    5.753    0.000
##    .TPER10_12R        0.248    0.060    4.134    0.000
##    .TPER10_13R        0.693    0.090    7.703    0.000
##     neuro_t1          0.851    0.172    4.940    0.000
##     neuro_t2          0.685    0.155    4.412    0.000
##     neuro_t3          0.865    0.183    4.719    0.000
##     neuro_t4          0.659    0.170    3.873    0.000
```

```
# Auto-regressive model
mod.5 <- 'neuro_t1 =~ TPER7_08 + TPER7_10 + TPER7_17R + TPER7_12R + TPER7_13R
neuro_t2 =~ TPER8_08 + TPER8_10 + TPER8_17R + TPER8_12R + TPER8_13R
neuro_t3 =~ TPER9_08 + TPER9_10 + TPER9_17R + TPER9_12R + TPER9_13R
neuro_t4 =~ TPER10_08 + TPER10_10 + TPER10_17R + TPER10_12R + TPER10_13R

TPER7_08 ~~ TPER8_08 + TPER9_08 + TPER10_08
TPER8_08 ~~ TPER9_08 + TPER10_08
TPER9_08 ~~ TPER10_08

TPER7_10 ~~ TPER8_10 + TPER9_10 + TPER10_10
TPER8_10 ~~ TPER9_10 + TPER10_10
TPER9_10 ~~ TPER10_10

TPER7_17R ~~ TPER8_17R + TPER9_17R + TPER10_17R
TPER8_17R ~~ TPER9_17R + TPER10_17R
```

```
TPER9_17R ~~ TPER10_17R

TPER7_12R ~~ TPER8_12R + TPER9_12R + TPER10_12R
TPER8_12R ~~ TPER9_12R + TPER10_12R
TPER9_12R ~~ TPER10_12R

TPER7_13R ~~ TPER8_13R + TPER9_13R + TPER10_13R
TPER8_13R ~~ TPER9_13R + TPER10_13R
TPER9_13R ~~ TPER10_13R

neuro_t4 ~ neuro_t3
neuro_t3 ~ neuro_t2
neuro_t2 ~ neuro_t1'
fit.5 <- cfa(mod.5, data=oysup)
summary(fit.5, fit.measures=TRUE)
```

```
## lavaan (0.6-1.1141) converged normally after  44 iterations
##
##                                                   Used       Total
##   Number of observations                           134        1074
##
##   Estimator                                         ML
##   Model Fit Test Statistic                     277.039
##   Degrees of freedom                               137
##   P-value (Chi-square)                           0.000
##
## Model test baseline model:
##
##   Minimum Function Test Statistic             1338.250
##   Degrees of freedom                               190
##   P-value                                        0.000
##
## User model versus baseline model:
##
##   Comparative Fit Index (CFI)                    0.878
##   Tucker-Lewis Index (TLI)                       0.831
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)              -3511.953
##   Loglikelihood unrestricted model (H1)      -3373.434
##
##   Number of free parameters                         73
##   Akaike (AIC)                                7169.906
##   Bayesian (BIC)                              7381.449
##   Sample-size adjusted Bayesian (BIC)         7150.532
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.087
##   90 Percent Confidence Interval          0.072  0.102
##   P-value RMSEA <= 0.05                          0.000
##
## Standardized Root Mean Square Residual:
```

```
##
##   SRMR                                           0.117
##
## Parameter Estimates:
##
##   Information                                 Expected
##   Standard Errors                             Standard
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   neuro_t1 =~
##     TPER7_08          1.000
##     TPER7_10          0.659    0.109    6.033    0.000
##     TPER7_17R         1.127    0.118    9.573    0.000
##     TPER7_12R         1.112    0.106   10.495    0.000
##     TPER7_13R         0.600    0.090    6.627    0.000
##   neuro_t2 =~
##     TPER8_08          1.000
##     TPER8_10          0.943    0.127    7.421    0.000
##     TPER8_17R         1.028    0.125    8.226    0.000
##     TPER8_12R         0.903    0.108    8.387    0.000
##     TPER8_13R         0.655    0.112    5.853    0.000
##   neuro_t3 =~
##     TPER9_08          1.000
##     TPER9_10          0.905    0.125    7.215    0.000
##     TPER9_17R         0.871    0.125    6.958    0.000
##     TPER9_12R         0.958    0.123    7.788    0.000
##     TPER9_13R         0.436    0.099    4.407    0.000
##   neuro_t4 =~
##     TPER10_08         1.000
##     TPER10_10         0.604    0.131    4.622    0.000
##     TPER10_17R        1.010    0.143    7.061    0.000
##     TPER10_12R        1.085    0.149    7.281    0.000
##     TPER10_13R        0.621    0.122    5.107    0.000
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   neuro_t4 ~
##     neuro_t3          0.348    0.095    3.647    0.000
##   neuro_t3 ~
##     neuro_t2          0.478    0.118    4.037    0.000
##   neuro_t2 ~
##     neuro_t1          0.490    0.098    4.980    0.000
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##  .TPER7_08 ~~
##    .TPER8_08         -0.017    0.069   -0.251    0.801
##    .TPER9_08          0.031    0.072    0.430    0.667
##    .TPER10_08         0.239    0.085    2.812    0.005
##  .TPER8_08 ~~
##    .TPER9_08          0.075    0.074    1.019    0.308
##    .TPER10_08         0.035    0.083    0.415    0.678
##  .TPER9_08 ~~
```

```
##     .TPER10_08         0.151    0.089    1.710    0.087
##   .TPER7_10 ~~
##     .TPER8_10          0.008    0.069    0.122    0.903
##     .TPER9_10         -0.038    0.078   -0.489    0.625
##     .TPER10_10         0.114    0.080    1.424    0.155
##   .TPER8_10 ~~
##     .TPER9_10         -0.005    0.067   -0.077    0.939
##     .TPER10_10         0.049    0.068    0.728    0.466
##   .TPER9_10 ~~
##     .TPER10_10         0.117    0.077    1.531    0.126
##   .TPER7_17R ~~
##     .TPER8_17R        -0.003    0.050   -0.062    0.951
##     .TPER9_17R         0.031    0.065    0.471    0.637
##     .TPER10_17R        0.028    0.049    0.563    0.573
##   .TPER8_17R ~~
##     .TPER9_17R         0.054    0.061    0.875    0.381
##     .TPER10_17R        0.075    0.045    1.654    0.098
##   .TPER9_17R ~~
##     .TPER10_17R        0.053    0.059    0.898    0.369
##   .TPER7_12R ~~
##     .TPER8_12R        -0.006    0.033   -0.181    0.856
##     .TPER9_12R         0.089    0.048    1.852    0.064
##     .TPER10_12R        0.007    0.034    0.210    0.834
##   .TPER8_12R ~~
##     .TPER9_12R         0.146    0.052    2.840    0.005
##     .TPER10_12R       -0.012    0.035   -0.345    0.730
##   .TPER9_12R ~~
##     .TPER10_12R        0.115    0.052    2.211    0.027
##   .TPER7_13R ~~
##     .TPER8_13R         0.066    0.063    1.056    0.291
##     .TPER9_13R         0.196    0.069    2.860    0.004
##     .TPER10_13R        0.162    0.062    2.604    0.009
##   .TPER8_13R ~~
##     .TPER9_13R         0.265    0.076    3.492    0.000
##     .TPER10_13R        0.137    0.067    2.045    0.041
##   .TPER9_13R ~~
##     .TPER10_13R        0.137    0.071    1.928    0.054
##
## Variances:
##                     Estimate  Std.Err  z-value  P(>|z|)
##     .TPER7_08          0.709    0.098    7.208    0.000
##     .TPER7_10          0.912    0.116    7.883    0.000
##     .TPER7_17R         0.506    0.081    6.266    0.000
##     .TPER7_12R         0.168    0.052    3.230    0.001
##     .TPER7_13R         0.639    0.082    7.829    0.000
##     .TPER8_08          0.709    0.102    6.981    0.000
##     .TPER8_10          0.589    0.086    6.882    0.000
##     .TPER8_17R         0.416    0.070    5.970    0.000
##     .TPER8_12R         0.290    0.051    5.687    0.000
##     .TPER8_13R         0.744    0.096    7.709    0.000
##     .TPER9_08          0.689    0.115    5.969    0.000
##     .TPER9_10          0.712    0.111    6.412    0.000
##     .TPER9_17R         0.771    0.116    6.674    0.000
##     .TPER9_12R         0.646    0.107    6.044    0.000
```

```
##    .TPER9_13R         0.859    0.109    7.865    0.000
##    .TPER10_08         1.055    0.142    7.407    0.000
##    .TPER10_10         0.860    0.110    7.851    0.000
##    .TPER10_17R        0.382    0.068    5.592    0.000
##    .TPER10_12R        0.214    0.061    3.476    0.001
##    .TPER10_13R        0.705    0.091    7.754    0.000
##     neuro_t1          0.813    0.167    4.866    0.000
##    .neuro_t2          0.485    0.115    4.226    0.000
##    .neuro_t3          0.677    0.152    4.459    0.000
##    .neuro_t4          0.502    0.136    3.707    0.000
```

Based on CFI and RSMEA values, the auto-regressive model fits the data a bit poorer than the other.

4) Fit a longitudinal growth model in SEM and in HLM. Compare and contrast the differences.

```
# SEM
model.6 <- 'neuro_i =~ 1*neuro_7t + 1*neuro_8t + 1*neuro_9t + 1*neuro_10t
            neuro_s =~ 0*neuro_7t + 1*neuro_8t + 2*neuro_9t + 3*neuro_10t'
fit.6 <- growth(model.6, data = oysup)
```

```
## Warning in lav_object_post_check(object): lavaan WARNING: some estimated lv
## variances are negative
```

```
summary(fit.6)
```

```
## lavaan (0.6-1.1141) converged normally after  30 iterations
##
##                                                   Used       Total
##   Number of observations                           135        1074
##
##   Estimator                                         ML
##   Model Fit Test Statistic                       3.768
##   Degrees of freedom                                 5
##   P-value (Chi-square)                           0.583
##
## Parameter Estimates:
##
##   Information                                 Expected
##   Standard Errors                             Standard
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   neuro_i =~
##     neuro_7t          1.000
##     neuro_8t          1.000
##     neuro_9t          1.000
##     neuro_10t         1.000
##   neuro_s =~
##     neuro_7t          0.000
##     neuro_8t          1.000
##     neuro_9t          2.000
##     neuro_10t         3.000
##
```

```
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   neuro_i ~~
##     neuro_s           0.054    0.048    1.112    0.266
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .neuro_7t          0.000
##    .neuro_8t          0.000
##    .neuro_9t          0.000
##    .neuro_10t         0.000
##     neuro_i           2.287    0.077   29.776    0.000
##     neuro_s          -0.066    0.030   -2.165    0.030
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .neuro_7t          0.806    0.147    5.476    0.000
##    .neuro_8t          0.795    0.116    6.834    0.000
##    .neuro_9t          0.897    0.127    7.086    0.000
##    .neuro_10t         0.946    0.161    5.890    0.000
##     neuro_i           0.228    0.125    1.824    0.068
##     neuro_s          -0.050    0.027   -1.826    0.068
```

```r
# restructuring for HLM
oysup_long <- tbl_df(oysup) %>%
  gather(c(neuro_7t:neuro_10t), key = "grade", value = "value") %>%
  separate(grade, into = c("variable", "grade"), sep = "_", convert = T) %>%
  separate(grade, into = c("grade", "delete"), sep = "t") %>%
  mutate(grade = as.numeric(grade)) %>%
  dplyr::select(-delete) %>%
  spread(variable, value)

# HLM
model.7 <- lmer(neuro ~ 1 + grade + (1 + grade | FAMID), data=oysup_long)
summary(model.7)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: neuro ~ 1 + grade + (1 + grade | FAMID)
##    Data: oysup_long
##
## REML criterion at convergence: 6640.1
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.99345 -0.74109 -0.09588  0.68525  2.35649
##
## Random effects:
##  Groups   Name        Variance Std.Dev. Corr
##  FAMID    (Intercept) 0.758431 0.87088
##           grade       0.001348 0.03671  -1.00
##  Residual             0.755907 0.86943
## Number of obs: 2337, groups:  FAMID, 939
##
## Fixed effects:
##             Estimate Std. Error t value
```

```
## (Intercept)  2.82385    0.16432  17.185
## grade       -0.07221    0.01902  -3.796
##
## Correlation of Fixed Effects:
##       (Intr)
## grade -0.987
```

Both models estimate the average slope at about -.07. The intercepts and random effects are estimated differently.

5) Constrain the residual variances to be equal. Does this change the fit of your model?

```
model.8 <- 'neuro_i =~ 1*neuro_7t + 1*neuro_8t + 1*neuro_9t + 1*neuro_10t
            neuro_s =~ 0*neuro_7t + 1*neuro_8t + 2*neuro_9t + 3*neuro_10t

neuro_7t ~~ u*neuro_7t
neuro_8t ~~ u*neuro_8t
neuro_9t ~~ u*neuro_9t
neuro_10t ~~ u*neuro_10t'
fit.8 <- growth(model.8, data = oysup)
```

```
## Warning in lav_object_post_check(object): lavaan WARNING: some estimated lv
## variances are negative
```

```
summary(fit.8)
```

```
## lavaan (0.6-1.1141) converged normally after  24 iterations
##
##                                                   Used       Total
##   Number of observations                           135        1074
##
##   Estimator                                         ML
##   Model Fit Test Statistic                       4.583
##   Degrees of freedom                                 8
##   P-value (Chi-square)                           0.801
##
## Parameter Estimates:
##
##   Information                                 Expected
##   Standard Errors                             Standard
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   neuro_i =~
##     neuro_7t          1.000
##     neuro_8t          1.000
##     neuro_9t          1.000
##     neuro_10t         1.000
##   neuro_s =~
##     neuro_7t          0.000
##     neuro_8t          1.000
##     neuro_9t          2.000
##     neuro_10t         3.000
```

```
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   neuro_i ~~
##     neuro_s           0.061    0.039    1.561    0.119
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .neuro_7t          0.000
##    .neuro_8t          0.000
##    .neuro_9t          0.000
##    .neuro_10t         0.000
##     neuro_i           2.287    0.077   29.745    0.000
##     neuro_s          -0.064    0.030   -2.117    0.034
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .neuro_7t   (u)    0.853    0.073   11.619    0.000
##    .neuro_8t   (u)    0.853    0.073   11.619    0.000
##    .neuro_9t   (u)    0.853    0.073   11.619    0.000
##    .neuro_10t  (u)    0.853    0.073   11.619    0.000
##     neuro_i           0.201    0.110    1.830    0.067
##     neuro_s          -0.045    0.021   -2.150    0.032
```

```
fitMeasures(fit.8)
```

```
##                npar                fmin               chisq
##               6.000               0.017               4.583
##                  df              pvalue      baseline.chisq
##               8.000               0.801              56.057
##         baseline.df     baseline.pvalue                 cfi
##               6.000               0.000               1.000
##                 tli                nnfi                 rfi
##               1.051               1.051               0.939
##                 nfi                pnfi                 ifi
##               0.918               1.224               1.071
##                 rni                logl   unrestricted.logl
##               1.068            -759.039            -756.747
##                 aic                 bic              ntotal
##            1530.078            1547.510             135.000
##                bic2               rmsea      rmsea.ci.lower
##            1528.530               0.000               0.000
##      rmsea.ci.upper        rmsea.pvalue                 rmr
##               0.065               0.911               0.045
##          rmr_nomean                srmr        srmr_bentler
##               0.045               0.051               0.051
## srmr_bentler_nomean         srmr_bollen  srmr_bollen_nomean
##               0.043               0.028               0.023
##          srmr_mplus   srmr_mplus_nomean               cn_05
##               0.040               0.041             457.752
##               cn_01                 gfi                agfi
##             592.738               0.997               0.995
##                pgfi                 mfi                ecvi
##               0.570               1.013                  NA
```

Yep. These constraints are overly restrictive for the nature of this data across four years, and so the model becomes fully saturated and the cfi defaults to 1.

6) Contrain your slope to be fixed, not random. How does this change your model?

```
model.9 <- 'neuro_i =~ 1*neuro_7t + 1*neuro_8t + 1*neuro_9t + 1*neuro_10t
            neuro_s =~ 0*neuro_7t + 1*neuro_8t + 2*neuro_9t + 3*neuro_10t
            neuro_s ~~ 0*neuro_s'
fit.9 <- growth(model.9, data = oysup)
```

```
## Warning in lav_object_post_check(object): lavaan WARNING: covariance matrix of latent variables
##                   is not positive definite;
##                   use inspect(fit,"cov.lv") to investigate.
```

```
summary(fit.9)
```

```
## lavaan (0.6-1.1141) converged normally after  24 iterations
##
##                                                   Used       Total
##   Number of observations                           135        1074
##
##   Estimator                                         ML
##   Model Fit Test Statistic                       7.073
##   Degrees of freedom                                 6
##   P-value (Chi-square)                           0.314
##
## Parameter Estimates:
##
##   Information                                 Expected
##   Standard Errors                             Standard
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   neuro_i =~
##     neuro_7t          1.000
##     neuro_8t          1.000
##     neuro_9t          1.000
##     neuro_10t         1.000
##   neuro_s =~
##     neuro_7t          0.000
##     neuro_8t          1.000
##     neuro_9t          2.000
##     neuro_10t         3.000
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   neuro_i ~~
##     neuro_s          -0.022    0.028   -0.782    0.434
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .neuro_7t          0.000
```

```
##    .neuro_8t          0.000
##    .neuro_9t          0.000
##    .neuro_10t         0.000
##     neuro_i           2.296    0.079   28.890    0.000
##     neuro_s          -0.068    0.033   -2.082    0.037
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##     neuro_s           0.000
##    .neuro_7t          0.655    0.111    5.919    0.000
##    .neuro_8t          0.796    0.116    6.858    0.000
##    .neuro_9t          0.873    0.124    7.072    0.000
##    .neuro_10t         0.779    0.121    6.456    0.000
##     neuro_i           0.368    0.105    3.512    0.000
```

```r
anova(fit.6, fit.9)
```

```
## Chi Square Difference Test
##
##       Df    AIC    BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## fit.6  5 1535.3 1561.4 3.7675
## fit.9  6 1536.6 1559.8 7.0729     3.3053       1    0.06906 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Honestly, not much changed, but there wasn't much variation in the slope to begin with. Now, the slope's variance is constrained to zero, whereas before, it was estimated at -.05 (basically zero). The intercept increased by a teeny tiny bit, and the magnitude of the (negative) fixed slope became a teeny tiny bit larger. However, the intercept variance did increase from .23 to .37, likely because that extra variance has to go somewhere!

7) Change the time metric in your SEM growth model. How does that change your estimates? Does it change your fit statistics?

```r
model.10 <- 'neuro_i =~ 1*neuro_7t + 1*neuro_8t + 1*neuro_9t + 1*neuro_10t
            neuro_s =~ 1*neuro_7t + 2*neuro_8t + 3*neuro_9t + 4*neuro_10t'
fit.10 <- growth(model.10, data = oysup)
```

```
## Warning in lav_object_post_check(object): lavaan WARNING: some estimated lv
## variances are negative
```

```r
summary(fit.10)
```

```
## lavaan (0.6-1.1141) converged normally after  36 iterations
##
##                                                   Used       Total
##   Number of observations                           135        1074
##
##   Estimator                                         ML
##   Model Fit Test Statistic                       3.768
##   Degrees of freedom                                 5
##   P-value (Chi-square)                           0.583
##
## Parameter Estimates:
##
```

```
##   Information                                 Expected
##   Standard Errors                             Standard
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   neuro_i =~
##     neuro_7t          1.000
##     neuro_8t          1.000
##     neuro_9t          1.000
##     neuro_10t         1.000
##   neuro_s =~
##     neuro_7t          1.000
##     neuro_8t          2.000
##     neuro_9t          3.000
##     neuro_10t         4.000
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   neuro_i ~~
##     neuro_s           0.103    0.073    1.422    0.155
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .neuro_7t          0.000
##    .neuro_8t          0.000
##    .neuro_9t          0.000
##    .neuro_10t         0.000
##     neuro_i           2.353    0.099   23.879    0.000
##     neuro_s          -0.066    0.030   -2.165    0.030
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .neuro_7t          0.806    0.147    5.476    0.000
##    .neuro_8t          0.795    0.116    6.834    0.000
##    .neuro_9t          0.897    0.127    7.086    0.000
##    .neuro_10t         0.946    0.161    5.890    0.000
##     neuro_i           0.071    0.234    0.305    0.760
##     neuro_s          -0.050    0.027   -1.826    0.068
```

```
fitMeasures(fit.10)
```

```
##                npar                fmin               chisq
##               9.000               0.014               3.768
##                  df              pvalue      baseline.chisq
##               5.000               0.583              56.057
##         baseline.df     baseline.pvalue                 cfi
##               6.000               0.000               1.000
##                 tli                nnfi                 rfi
##               1.030               1.030               0.919
##                 nfi                pnfi                 ifi
##               0.933               0.777               1.024
##                 rni                logl   unrestricted.logl
##               1.025            -758.631            -756.747
##                 aic                 bic              ntotal
##            1535.262            1561.410             135.000
```

```
##                bic2               rmsea      rmsea.ci.lower
##            1532.940               0.000               0.000
##      rmsea.ci.upper        rmsea.pvalue                 rmr
##               0.103               0.738               0.030
##          rmr_nomean                srmr        srmr_bentler
##               0.030               0.043               0.043
## srmr_bentler_nomean         srmr_bollen  srmr_bollen_nomean
##               0.028               0.028               0.019
##          srmr_mplus   srmr_mplus_nomean               cn_05
##               0.031               0.026             397.685
##               cn_01                 gfi                agfi
##             541.581               0.998               0.993
##                pgfi                 mfi                ecvi
##               0.356               1.005                  NA
```

The estimate of the slope did not change; however, the estimate of the intercept and its variance did.

8) Try a different type of estimation (see lavaan tutorial for details). How does that change your model?

```
model.11 <- 'neuro_i =~ 1*neuro_7t + 1*neuro_8t + 1*neuro_9t + 1*neuro_10t
            neuro_s =~ 0*neuro_7t + 1*neuro_8t + 2*neuro_9t + 3*neuro_10t'
fit.11 <- growth(model.11, data = oysup, estimator = "DWLS")
```

```
## Warning in lav_object_post_check(object): lavaan WARNING: some estimated lv
## variances are negative
```

```
summary(fit.11)
```

```
## lavaan (0.6-1.1141) converged normally after  21 iterations
##
##                                                   Used       Total
##   Number of observations                           135        1074
##
##   Estimator                                       DWLS
##   Model Fit Test Statistic                       2.962
##   Degrees of freedom                                 5
##   P-value (Chi-square)                           0.706
##
## Parameter Estimates:
##
##   Information                                 Expected
##   Standard Errors                             Standard
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   neuro_i =~
##     neuro_7t          1.000
##     neuro_8t          1.000
##     neuro_9t          1.000
##     neuro_10t         1.000
##   neuro_s =~
##     neuro_7t          0.000
##     neuro_8t          1.000
```

```
##     neuro_9t          2.000
##     neuro_10t         3.000
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   neuro_i ~~
##     neuro_s           0.055    0.063    0.868    0.385
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .neuro_7t          0.000
##    .neuro_8t          0.000
##    .neuro_9t          0.000
##    .neuro_10t         0.000
##     neuro_i           2.286    0.074   30.903    0.000
##     neuro_s          -0.064    0.040   -1.625    0.104
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .neuro_7t          0.828    0.169    4.912    0.000
##    .neuro_8t          0.750    0.112    6.705    0.000
##    .neuro_9t          0.927    0.108    8.574    0.000
##    .neuro_10t         0.948    0.183    5.192    0.000
##     neuro_i           0.231    0.136    1.695    0.090
##     neuro_s          -0.051    0.039   -1.315    0.189
```

```
fitMeasures(fit.11, c("cfi", "rmsea"))
```

```
##   cfi rmsea
##     1     0
```

```
fitMeasures(fit.6, c("cfi", "rmsea"))
```

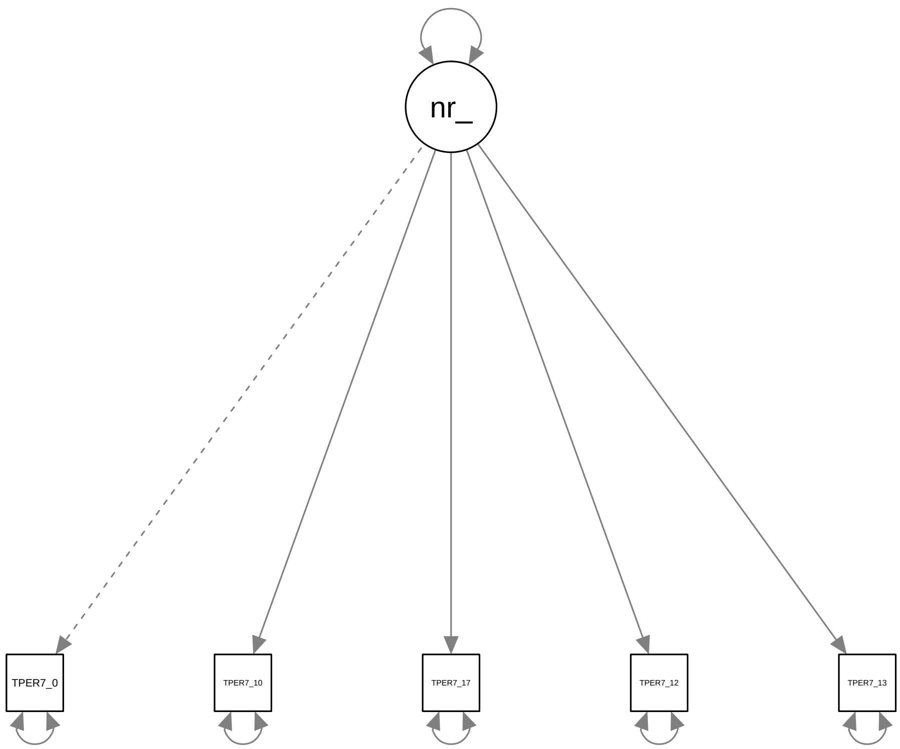
```
##   cfi rmsea
##     1     0
```

Not much of a difference between using diagonal weighted least squares estimation and the default (maximum likelihoods).

9) Provide semplots for each of the models.

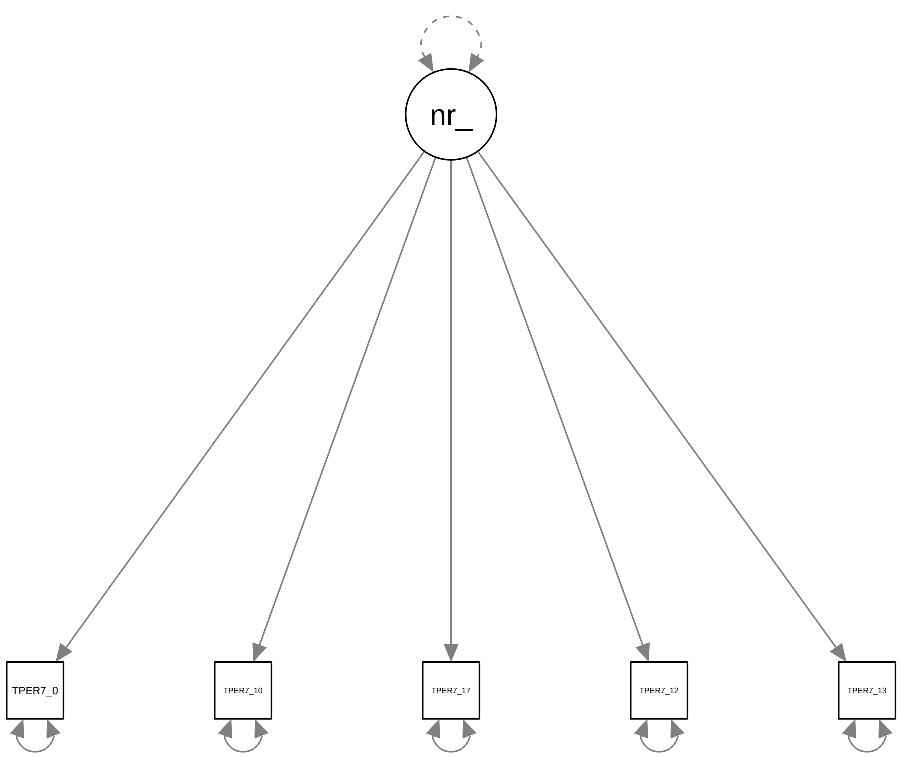
```
semPaths(fit.1)
```

```
## Warning in qgraph(Edgelist, labels = nLab, bidirectional = Bidir, directed
## = Directed, : The following arguments are not documented and likely not
## arguments of qgraph and thus ignored: loopRotation; residuals; residScale;
## residEdge; CircleEdgeEnd
```



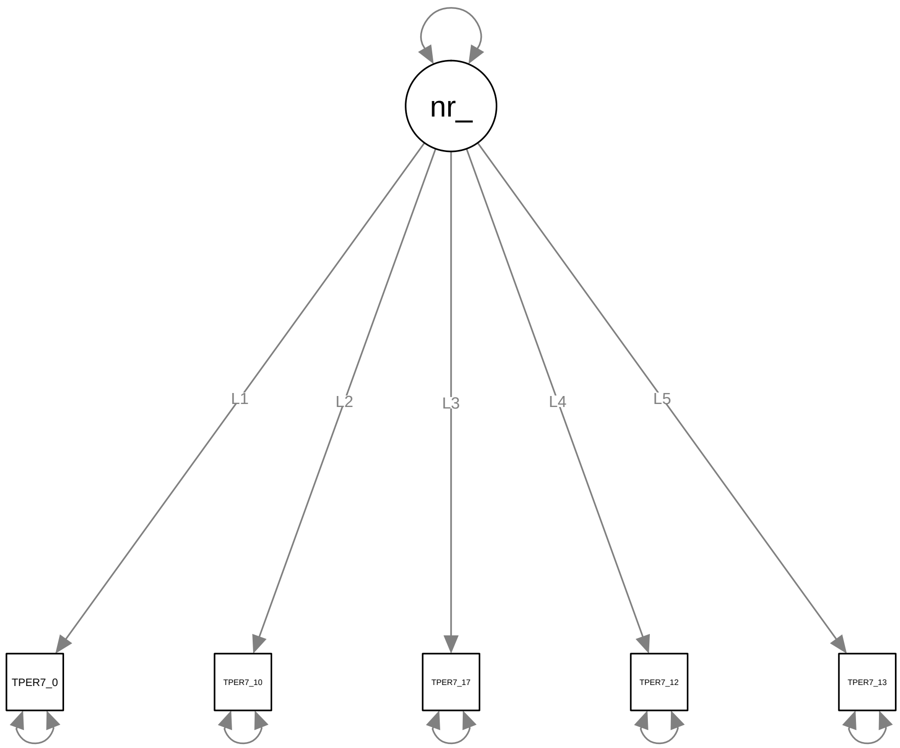

```
semPaths(fit.2)
```

```
## Warning in qgraph(Edgelist, labels = nLab, bidirectional = Bidir, directed
## = Directed, : The following arguments are not documented and likely not
## arguments of qgraph and thus ignored: loopRotation; residuals; residScale;
## residEdge; CircleEdgeEnd
```



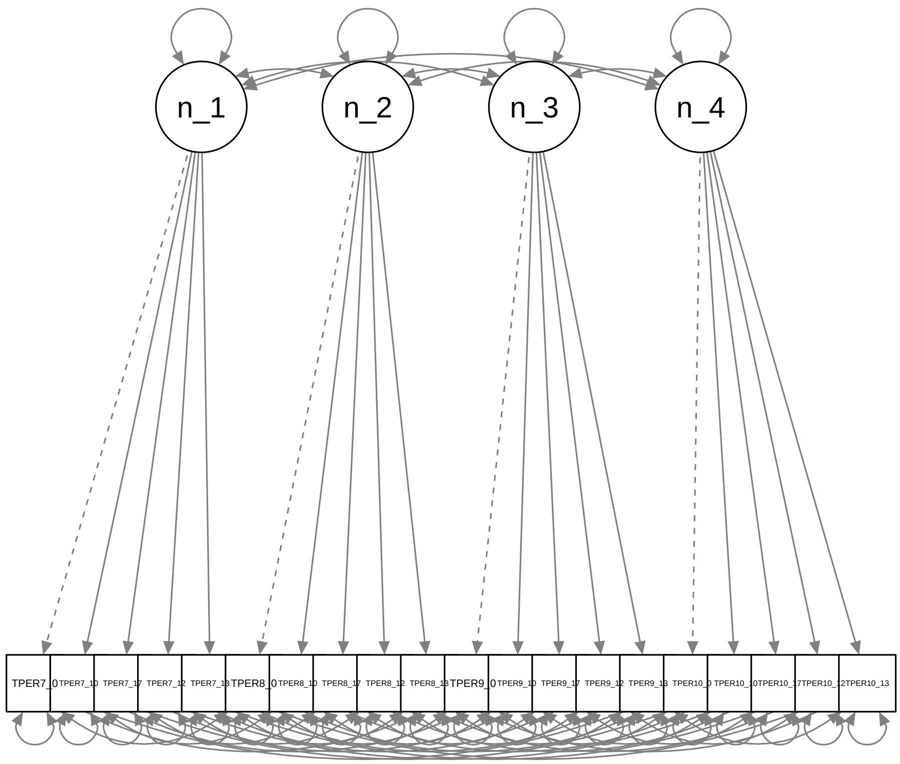
```r
semPaths(fit.3)
```

```
## Warning in qgraph(Edgelist, labels = nLab, bidirectional = Bidir, directed
## = Directed, : The following arguments are not documented and likely not
## arguments of qgraph and thus ignored: loopRotation; residuals; residScale;
## residEdge; CircleEdgeEnd
```



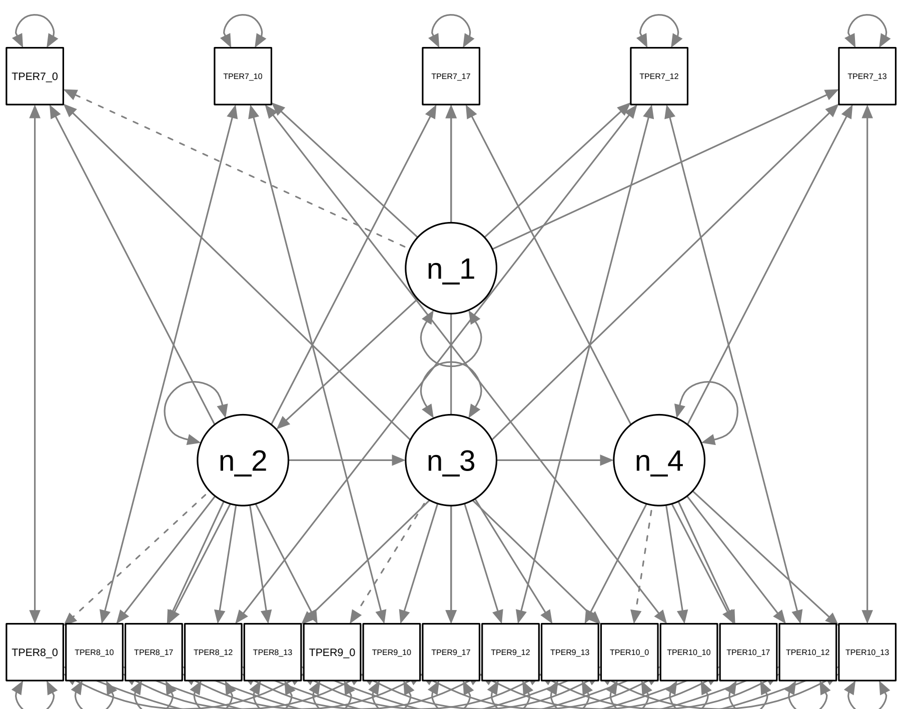

```r
semPaths(fit.4)
```

```
## Warning in qgraph(Edgelist, labels = nLab, bidirectional = Bidir, directed
## = Directed, : The following arguments are not documented and likely not
## arguments of qgraph and thus ignored: loopRotation; residuals; residScale;
## residEdge; CircleEdgeEnd
```



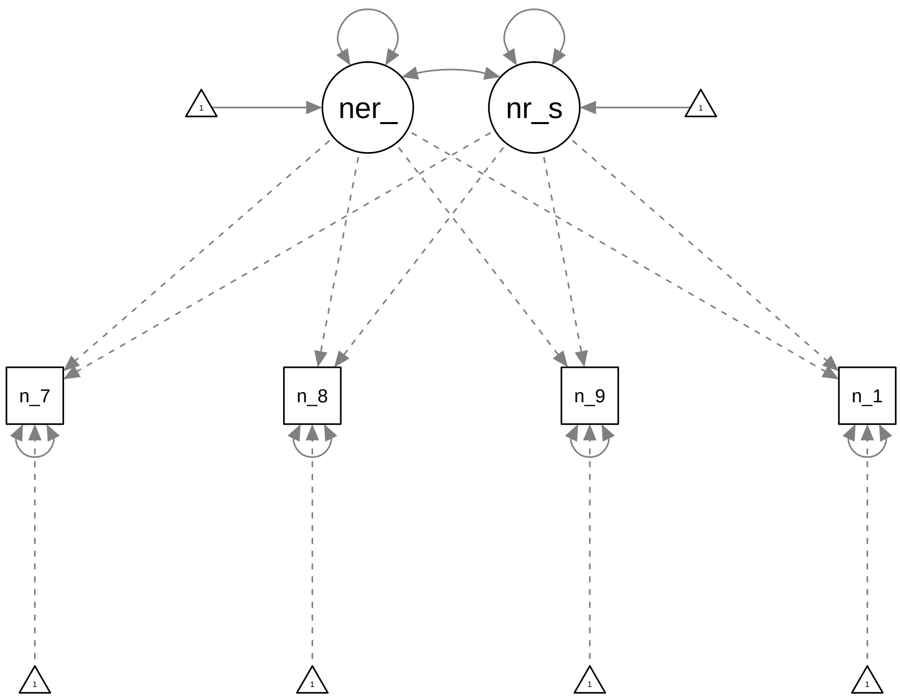

```r
semPaths(fit.5)
```

```
## Warning in qgraph(Edgelist, labels = nLab, bidirectional = Bidir, directed
## = Directed, : The following arguments are not documented and likely not
## arguments of qgraph and thus ignored: loopRotation; residuals; residScale;
## residEdge; CircleEdgeEnd
```



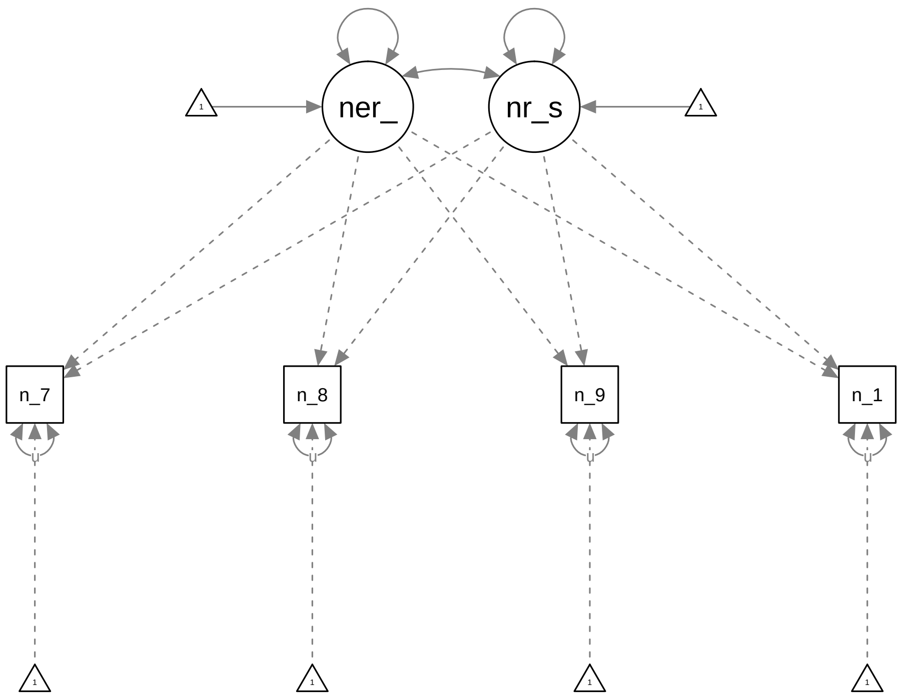
```r
semPaths(fit.6)
```

```
## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in qgraph(Edgelist, labels = nLab, bidirectional = Bidir, directed
## = Directed, : The following arguments are not documented and likely not
## arguments of qgraph and thus ignored: loopRotation; residuals; residScale;
## residEdge; CircleEdgeEnd
```



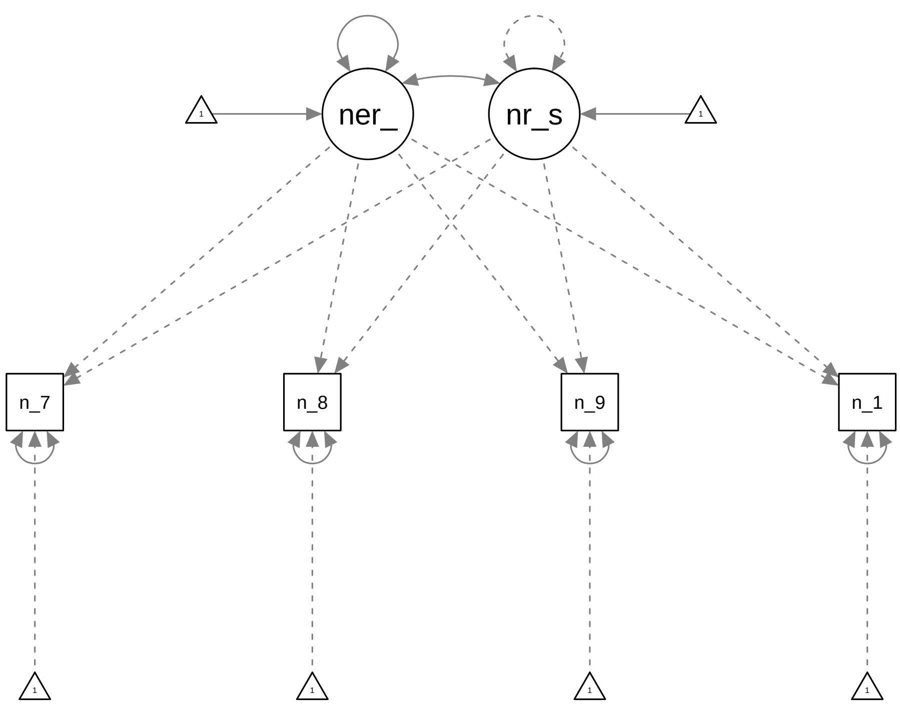

```r
semPaths(fit.8)
```

```
## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in qgraph(Edgelist, labels = nLab, bidirectional = Bidir, directed
## = Directed, : The following arguments are not documented and likely not
## arguments of qgraph and thus ignored: loopRotation; residuals; residScale;
## residEdge; CircleEdgeEnd
```




```r
semPaths(fit.9)
```

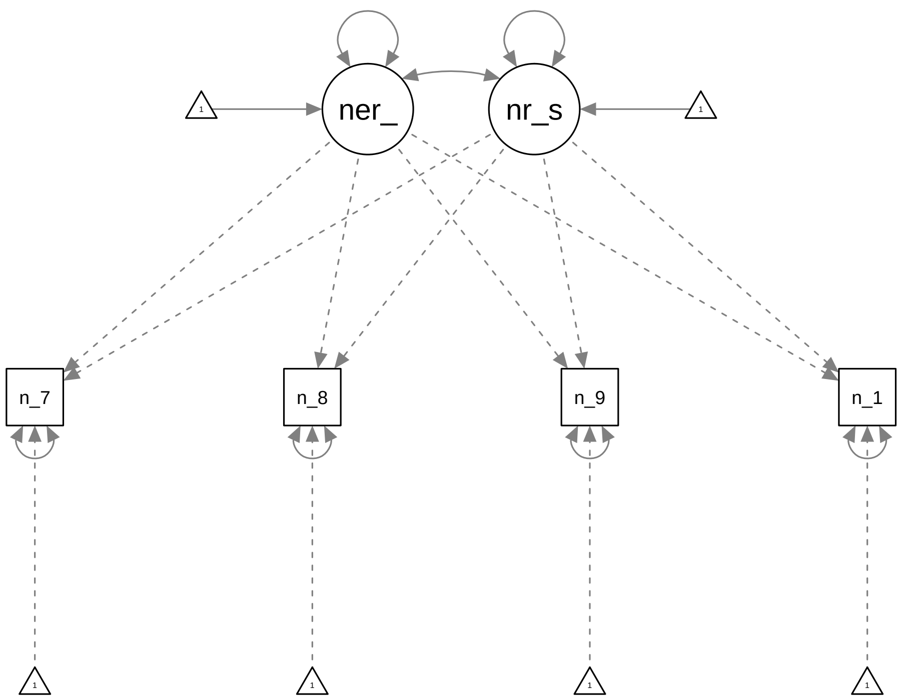
```
## Warning in qgraph(Edgelist, labels = nLab, bidirectional = Bidir, directed
## = Directed, : The following arguments are not documented and likely not
## arguments of qgraph and thus ignored: loopRotation; residuals; residScale;
## residEdge; CircleEdgeEnd
```




```r
semPaths(fit.10)
```

```
## Warning in sqrt(ETA2): NaNs produced
```

```
## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in qgraph(Edgelist, labels = nLab, bidirectional = Bidir, directed
## = Directed, : The following arguments are not documented and likely not
## arguments of qgraph and thus ignored: loopRotation; residuals; residScale;
## residEdge; CircleEdgeEnd
```



10) Test measurement invariance across time for your construct. Can you run growth models? If there is evidence of non-invariance, what seems to be the problem?

11) Fit a second order growth model. Compare and contrast the estimates with the normal latent growth model.

12) Fit a series of multiple group models. Constrain some parameters and compare the fit.

mod.8 <- 'neuro_t1 =~ TPER7_08 + TPER7_10 + TPER7_17R + TPER7_12R + TPER7_13R neuro_t2 =~ TPER8_08 + TPER8_10 + TPER8_17R + TPER8_12R + TPER8_13R neuro_t3 =~ TPER9_08 + TPER9_10 + TPER9_17R + TPER9_12R + TPER9_13R neuro_t4 =~ TPER10_08 + TPER10_10 + TPER10_17R + TPER10_12R + TPER10_13R

TPER7_08 ~~ TPER8_08 + TPER9_08 + TPER10_08 TPER8_08 ~~ TPER9_08 + TPER10_08 TPER9_08 ~~ TPER10_08

TPER7_10 ~~ TPER8_10 + TPER9_10 + TPER10_10 TPER8_10 ~~ TPER9_10 + TPER10_10 TPER9_10 ~~ TPER10_10

TPER7_17R ~~ TPER8_17R + TPER9_17R + TPER10_17R TPER8_17R ~~ TPER9_17R + TPER10_17R TPER9_17R ~~ TPER10_17R

TPER7_12R ~~ TPER8_12R + TPER9_12R + TPER10_12R TPER8_12R ~~ TPER9_12R + TPER10_12R TPER9_12R ~~ TPER10_12R

TPER7_13R ~~ TPER8_13R + TPER9_13R + TPER10_13R TPER8_13R ~~ TPER9_13R + TPER10_13R TPER9_13R ~~ TPER10_13R

TPER7_08 ~~ u*TPER7_08 TPER7_10* ~~ *u*TPER7_10 TPER7_17R ~~ u*TPER7_17R TPER7_12R* ~~ *u*TPER7_12R TPER7_13R ~~ u*TPER7_13R TPER8_08* ~~ *u*TPER8_08 TPER8_10 ~~ u*TPER8_10 TPER8_17R* ~~ *u*TPER8_17R TPER8_12R ~~ u*TPER8_12R TPER8_13R* ~~ *u*TPER8_13R TPER9_08 ~~ u*TPER9_08 TPER9_10* ~~ *u*TPER9_10 TPER9_17R ~~ u*TPER9_17R TPER9_12R* ~~ *u*TPER9_12R TPER9_13R ~~ u*TPER9_13R TPER10_08* ~~ *u*TPER10_08 TPER10_10 ~~ u*TPER10_10 TPER10_17R* ~~ *u*TPER10_17R TPER10_12R ~~ u*TPER10_12R TPER10_13R* ~~ *u*TPER10_13R'