Statistical Analysis Using Structural Equation Models

EPsy 8266

Christopher David Desjardins

Research Methodology Consulting Center

4/9/19

Constant review

- The mean of an endogenous variable is a function of: 1) intercept, 2) the unstandardized path coefficient, and 3) the mean of the exogenous variable.
- The predicted mean for exo/endo variables is the total effect of the constant.
- For exogenous continuous variables in an SEM, the unstandardized path coefficient for the direct effect of the constant is a mean.
- For endogenous continuous variables in an SEM, the direct effect of the constant is the intercept but the total effect is a mean.

Latent Growth Models

- Longitudinal models
- SR models with a mean and covariance structure
- Measure the same construct across time
- Error terms may be auto correlated.
- Data need to be measured at the same time for each case (**time structured**).
  - Measure executive functioning at baseline, 2, 4, 12, and 24 months since baseline on each participant.
  - Intervals can be unequal
  - If measurements occur at different times for each participant, consider HLM (via **lme4** or **nlme** in R) or continuous time-series SEM (**ctsem**).

HLM or SEM

The advantages of HLM include

- Doesn't require time structured data
- Can readily handle more levels

The advantages of SEM include

- Multiple growth curves (can look at change in two or more constructs together)
- Handles more complex error structure
- A general, greatly extendable framework (e.g., can look at change in latent variables across time or allow intercept or slopes to be predictors)
- Can perform on means and covariance structure and sample size (i.e., don't need raw data)
- FIML (HLM uses ML)

Modelling approach

1. Determine the basic shape of change and get a good fit
   - If bivariate (or more), then determine shape of change separately.
   - Then combine models
2. Add predictors

Types of latent curves we'll cover

- Nonlinear curve fitting
- Linear/quadratic growth curves
- Bivariate growth curves (ex: Curran, Muthen & Harford, 2004)
- Latent curve model with structured residuals (ex: Curran et. al, 2014)

Sleep deprivation study

- "Patterns of performance degradation and restoration during sleep restriction and subsequent recovery: a sleep dose-response study." (Belenky et al, 2003)
- Sleep deprivation study.
- 18 participants were studied over a 10 day period
  - **A extremely small number for an SEM and we will use it just to illustrate visualizing longitudinal data.**
- On day 0 the subjects had their normal amount of sleep. Starting that night they were restricted to 3 hours of sleep per night. Each day the participants were given a series of reaction time tests and the average was computed.
- The response variable is the average reaction time in ms.

sleep.wide data set

```r
sleep.wide <- read.csv(file = "sleep_wide.csv")
apply(sleep.wide, 2, mean, na.rm = TRUE)

##      SubNum      female         gpa  Reaction.0  Reaction.1
## 109.5000000   0.5555556   2.5223711 256.6518056 264.4957556
##  Reaction.2  Reaction.3  Reaction.4  Reaction.5  Reaction.6
## 265.3619000 282.9920111 288.6494222 308.5184556 312.1782556
##  Reaction.7  Reaction.8  Reaction.9
## 315.9442062 337.5263250 348.2195062

apply(sleep.wide, 2, sd, na.rm = TRUE)

##     SubNum     female        gpa Reaction.0 Reaction.1 Reaction.2
##  5.3385391  0.5113100  0.5377201 32.1294511 33.4303337 29.4734233
## Reaction.3 Reaction.4 Reaction.5 Reaction.6 Reaction.7 Reaction.8
## 38.8577385 42.5378866 51.7696245 63.1737202 52.5745845 62.6855391
## Reaction.9
## 70.8443938

apply(sleep.wide, 2, function(x) sum(is.na(x)))

##     SubNum     female        gpa Reaction.0 Reaction.1 Reaction.2
##          0          0          0          0          0          0
## Reaction.3 Reaction.4 Reaction.5 Reaction.6 Reaction.7 Reaction.8
##          0          0          0          0          2          2
## Reaction.9
##          2
```

Reshape data for visualization

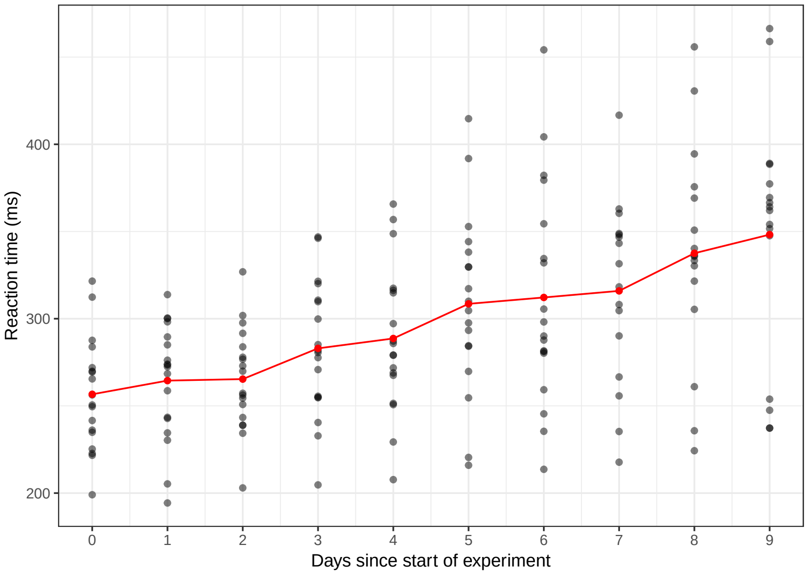
```
sleep.long <- reshape(sleep.wide,
                      direction = "long",
                      varying = paste0("Reaction.", 0:9),
                      timevar = "day",
                      idvar = "SubNum")

# Reorder by subject and then by day
sleep.long <- sleep.long[order(sleep.long$SubNum, sleep.long$day), ]
head(sleep.long)
```

```
##       SubNum female      gpa day Reaction
## 101.0    101      0 3.090871   0 249.5600
## 101.1    101      0 3.090871   1 258.7047
## 101.2    101      0 3.090871   2 250.8006
## 101.3    101      0 3.090871   3 321.4398
## 101.4    101      0 3.090871   4 356.8519
## 101.5    101      0 3.090871   5 414.6901
```

Connected means plot (code)

```
library(ggplot2)
ggplot(data = sleep.long, aes(x = day, y = Reaction)) +
  geom_point(alpha = .5) +
  stat_summary(fun.y = "mean", geom = "point", col ="red", size = 1.5) +
  stat_summary(fun.y = "mean", geom = "line", col = "red") +
  xlab("Days since start of experiment") +
  ylab("Reaction time (ms)") +
  scale_x_continuous(breaks = 0:9) +
  theme_bw()
```

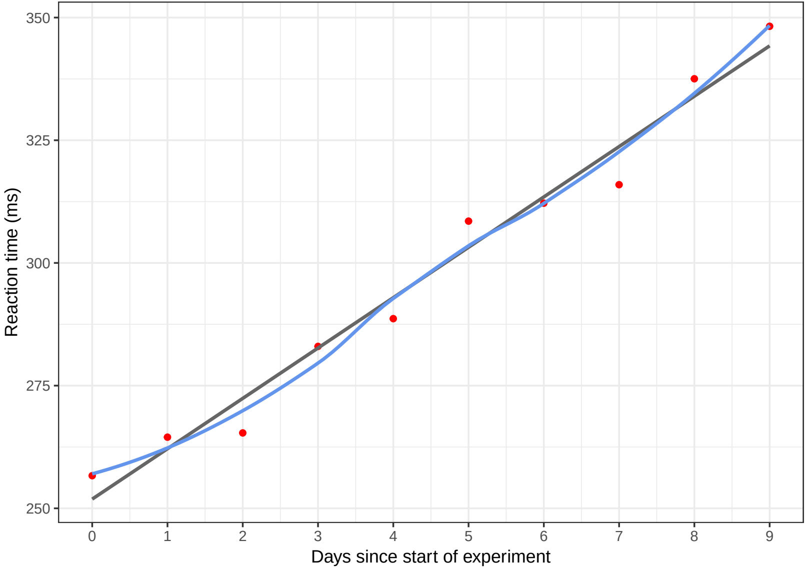
Connected means plot



Add two smoothers (code)

```
ggplot(data = sleep.long, aes(x = day, y = Reaction)) +
  stat_summary(fun.y = "mean", geom = "point", col ="red", size = 1.5) +
  geom_smooth(method = "lm", formula = y ~ x, se = FALSE, col = "gray40") +
  geom_smooth(method = "loess", se = FALSE, col = "cornflowerblue") +
  xlab("Days since start of experiment") +
  ylab("Reaction time (ms)") +
  scale_x_continuous(breaks = 0:9) +
  theme_bw()
```

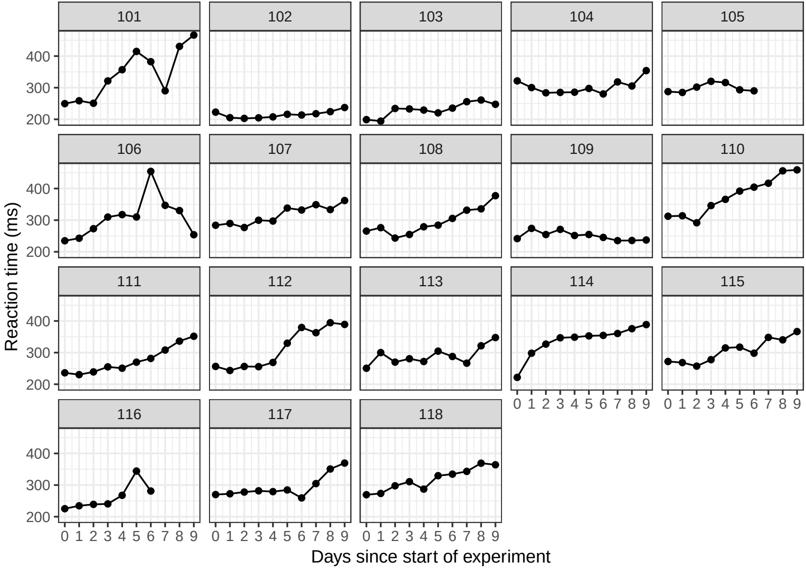
Add two smoothers



Individual (trellis) plots (code)

```
ggplot(data = sleep.long, aes(x = day, y = Reaction)) +
  geom_point() +
  geom_line() +
  xlab("Days since start of experiment") +
  ylab("Reaction time (ms)") +
  scale_x_continuous(breaks = 0:9) +
  theme_bw() +
  facet_wrap(~ SubNum)
```

Individual (trellis) plots



Linear and polynomial growth

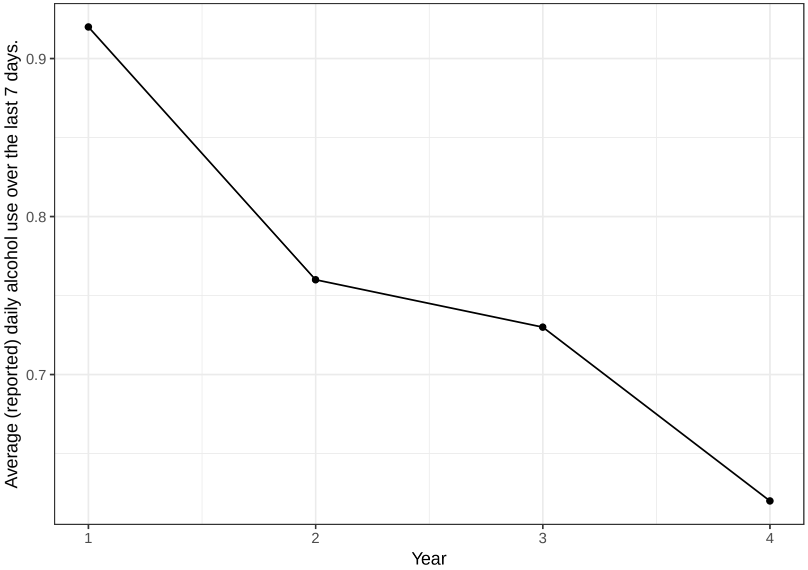
- Assumes that rate of change across time is linear, quadratic, cubic, ...
- Includes an intercept and at least one slope term.
- For linear need 3 time points, for quadratic need 4 time points, and so on.
- Manifest variables load onto all factors with fixed unstd. pattern coefficients.
  - Linear = 0, 1, 2, .., n.
  - Quadratic $= 0^2, 1^2, 2^2, ..., n^2$.
  - Cubic $= 0^3, 1^3, 2^3, ..., n^3$ and so on.
- Error terms for manifest variables across time could be correlated
- Variance of error terms could be homogeneous (restricted to be equal) or heterogeneous.
- Repeated measures are endogenous variables and adjusts for unexplained variation in our indicators when estimating the latent growth factors.

The Influence of Changes in Marital Status on Developmental Trajectories of Alcohol Use in Young Adults (Curran, Muthen, & Harford, 1998)

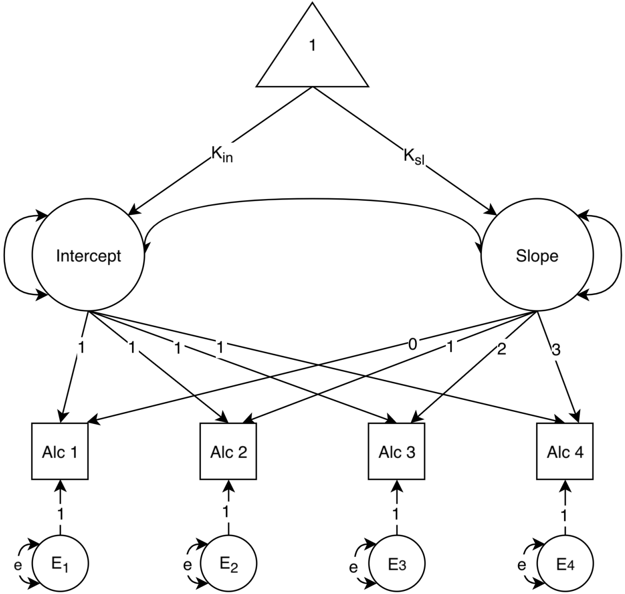
- Multiple group latent curve analysis was used to assess the impact of changes in marital status on alcohol use trajectories in young adults and test if these effects varied across ethnicity and gender.
- Four years of data from the National Longitudinal Survey of Adults (N = 4054)
- Alcohol use and marital status were assessed annually for four years and covariates included age, gender, education, and ethnicity

Data

```
library(lavaan)
alcuse.xbar <- c(0.92, 0.76, 0.73, 0.62, 0.08, 0.08, 0.08, 0.07, 22.69, 0.54, 0.30,
          0.14, 12.72)
alcuse.sigma <- c(1.68, 1.35, 1.26, 1.08, 0.27, 0.27, 0.27, 0.26, 1.38, 0.49, 0.46,
           0.35, 2.02)
lower.cor <- '
1.00
.494 1.00
.440 .519 1.00
.382 .471 .510 1.00
-.074 -.068 -.062 -.035 1.00
-.023 -.048 -.057 -.055 -.086 1.00
.009 -.003 -.036 -.039 -.085 -.083  1.00
.043 .025 .011 -.022 -.081 -.080 -.079 1.00
.032 .020 -.005 .010 .050 .028 .007 .028 1.00
.231, .238 .252 .264 -.048 -.005 -.028 -.003 .011 1.00
-.142 -.146 -.120 -.118 -.085 -.080 -.067 -.071 -.014 -.013 1.00
-.025 -.014 -.028 -.001 .012 .019 -.021 -.013 -.023 .040 -.265 1.00
.012 -.005 -.021 -.018 .006 .006 .009 .049 .222 -.110 -.149 -.125 1.00'
alcuse.cov <- getCov(lower.cor,
                     names = c(paste0("alc", 1:4), paste0("mar", 1:4),
                               "age", "male", "black", "hispanic", "education"),
                     sds = alcuse.sigma)
```


Average (reported) daily alcohol use over the last 7 days.
0.9
0.8
0.7
1
2
3
4
Year

Linear model with fixed manifest variable variances

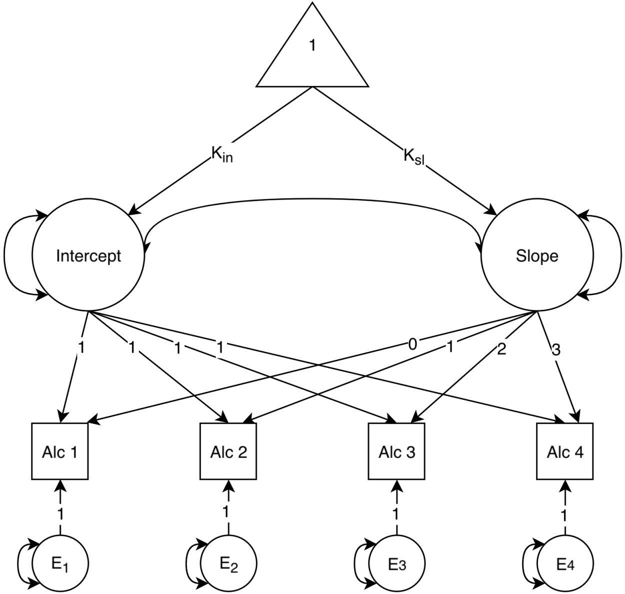


Characteristics of this model

- Unstd. pattern coefficients for the intercept factor fixed to 1.
- Unstd. pattern coefficients for the slope factor are constrained to 0 through 3 for linearity.
  - The 0 sets the origin to the initial performance at Year 1.
  - The 1 sets the scale of the slope factors for UL identification.
  - The 2 and 3 make the increase linear.
- Covariance between the intercept and slope factor indicates the degree to which performance at Year 1 covaries with change in alcohol use across the years.
- The κ (kappa) are the means of the latent growth factors (free).
  - κ_{in} is the mean alcohol use at Year 1, controlling for error variance at Year 1 and the **variance** represents variation around initial alcohol use (**random intercept**).
  - κ_{sl} is the average linear slope over the 4 years (i.e., how much alcohol use is expected to change each year), controlling for error variance and the **variance** represents variation in change in alcohol use over time (**random slope**).
- Predicted alcohol use for each can be calculated and error terms of adjacent times (could be) allowed to covary.

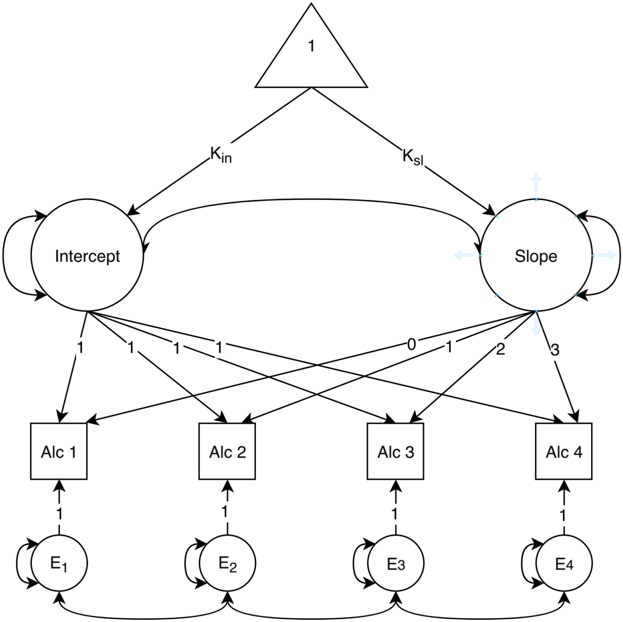
```r
line.mod <- "
# define factors
int =~ 1*alc1 + 1*alc2 + 1*alc3 + 1*alc4
slope =~ 0*alc1 + 1*alc2 + 2*alc3 + 3*alc4

# estimate factor means
int ~ 1
slope ~ 1

# estimate factor variance/covariance
int ~~ int + slope
slope ~~ slope

# assume variances are equivalent across time
alc1 ~~ e*alc1
alc2 ~~ e*alc2
alc3 ~~ e*alc3
alc4 ~~ e*alc4
"
line.fit <- lavaan(model = line.mod, sample.cov = alcuse.cov,
                   sample.mean = alcuse.xbar, sample.nobs = 4052)
fitMeasures(line.fit, c("chisq", "df", "p", "rmsea", "cfi", "tli"))

##   chisq      df   rmsea     cfi     tli
## 533.696   8.000   0.127   0.877   0.908
```

Free the manifest variable error variance



```r
line.mod.f <- "
# define factors
int =~ 1*alc1 + 1*alc2 + 1*alc3 + 1*alc4
slope =~ 0*alc1 + 1*alc2 + 2*alc3 + 3*alc4

# estimate factor means
int ~ 1
slope ~ 1

# estimate factor variance/covariance
int ~~ int + slope
slope ~~ slope

# assume variances are NOT equivalent across time
alc1 ~~ alc1
alc2 ~~ alc2
alc3 ~~ alc3
alc4 ~~ alc4
"
line.fit.f <- lavaan(model = line.mod.f, sample.cov = alcuse.cov,
                     sample.mean = alcuse.xbar, sample.nobs = 4052)
anova(line.fit, line.fit.f)

## Chi Square Difference Test
##
##             Df   AIC   BIC   Chisq Chisq diff Df diff Pr(>Chisq)
## line.fit.f  5 50890 50947  17.364
## line.fit    8 51400 51438 533.696     516.33       3  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

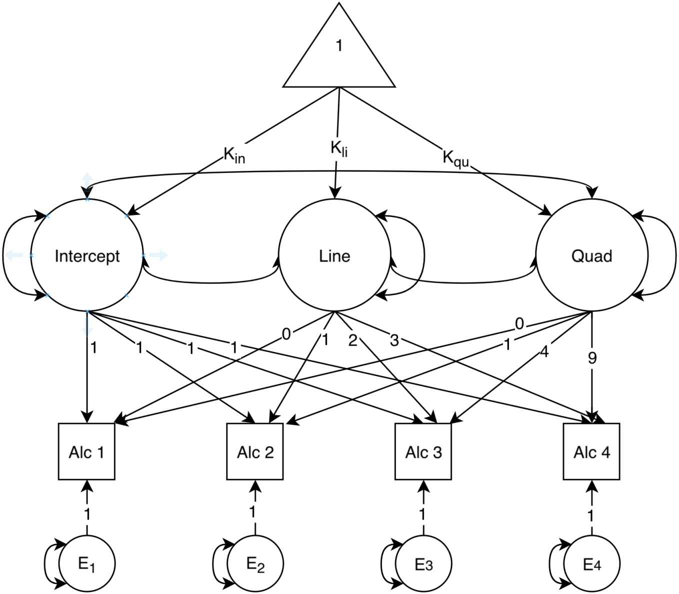
Add auto correlated errors (lag of 1)



Do we need auto correlated errors?

```
line.mod.auto <- "
# define factors
int =~ 1*alc1 + 1*alc2 + 1*alc3 + 1*alc4
slope =~ 0*alc1 + 1*alc2 + 2*alc3 + 3*alc4

# estimate factor means
int ~ 1
slope ~ 1

# estimate factor variance/covariance
int ~~ int + slope
slope ~~ slope

# assume variances are NOT equivalent and auto correlated
alc1 ~~ alc1 + alc2
alc2 ~~ alc2 + alc3
alc3 ~~ alc3 + alc4
alc4 ~~ alc4
"
line.fit.auto <- lavaan(model = line.mod.auto, sample.cov = alcuse.cov,
                  sample.mean = alcuse.xbar, sample.nobs = 4052)
anova(line.fit.f, line.fit.auto)

## Chi Square Difference Test
##
##                Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## line.fit.auto  2 50891 50966 12.389
## line.fit.f     5 50890 50947 17.364     4.9749       3     0.1736
```

Assessing quadratic growth - do we deviate from linearity?



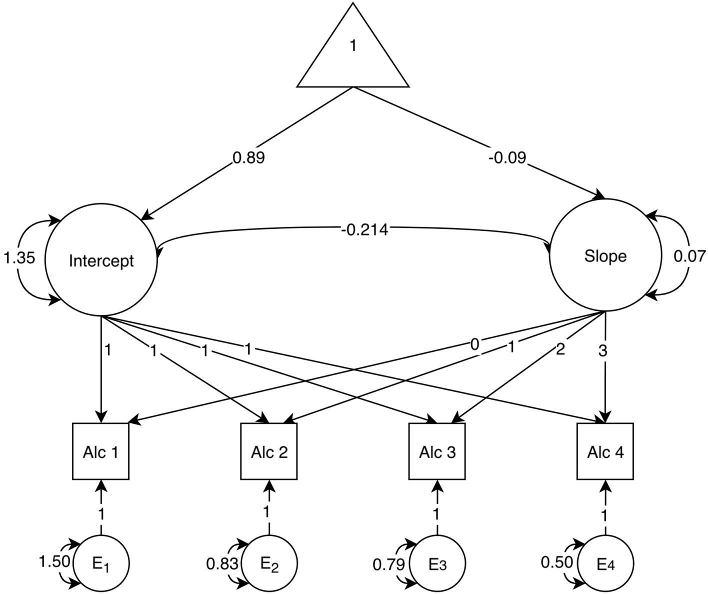
```
quad.mod <- "
# define factors
int =~ 1*alc1 + 1*alc2 + 1*alc3 + 1*alc4
line =~ 0*alc1 + 1*alc2 + 2*alc3 + 3*alc4
quad =~ 0*alc1 + 1*alc2 + 4*alc3 + 9*alc4

# estimate factor means
int ~ 1
line ~ 1
quad ~ 1

# estimate factor variance/covariance
int ~~ int + line + quad
line ~~ line + quad
quad ~~ quad

# assume variances are NOT equivalent
alc1 ~~ alc1
alc2 ~~ alc2
alc3 ~~ alc3
alc4 ~~ alc4
"
quad.fit <- lavaan(model = quad.mod, sample.cov = alcuse.cov,
                  sample.mean = alcuse.xbar, sample.nobs = 4052)
anova(line.fit.f, quad.fit)

## Chi Square Difference Test
##
##            Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## quad.fit    1 50891 50973 11.001
## line.fit.f  5 50890 50947 17.364     6.3628       4     0.1736
```

```
summary(line.fit.f, fit.measures = TRUE)

##   Number of free parameters                          9
##   Number of observations                          4052
##
##   Estimator                                         ML
##   Model Fit Test Statistic                      17.364
##   Degrees of freedom                                 5
##   P-value (Chi-square)                           0.004
##
## User model versus baseline model:
##
##   Comparative Fit Index (CFI)                    0.997
##   Tucker-Lewis Index (TLI)                       0.997
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.025
##   90 Percent Confidence Interval          0.013  0.038
##   P-value RMSEA <= 0.05                          1.000
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.014
```

```
summary(line.fit.f, fit.measures = TRUE)

## Latent Variables:
##                     Estimate  Std.Err  z-value  P(>|z|)
##   int =~
##     alc1               1.000
##     alc2               1.000
##     alc3               1.000
##     alc4               1.000
##   slope =~
##     alc1               0.000
##     alc2               1.000
##     alc3               2.000
##     alc4               3.000
##
## Covariances:
##                     Estimate  Std.Err  z-value  P(>|z|)
##   int ~~
##     slope             -0.214    0.016  -13.356    0.000
##
## Intercepts:
##                     Estimate  Std.Err  z-value  P(>|z|)
##     int                0.887    0.023   38.366    0.000
##     slope             -0.089    0.008  -11.527    0.000
##    .alc1               0.000
##    .alc2               0.000
##    .alc3               0.000
##    .alc4               0.000
```

```r
summary(line.fit.f, fit.measures = TRUE)

## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##     int               1.354    0.051   26.509    0.000
##     slope             0.068    0.007    9.467    0.000
##    .alc1              1.499    0.049   30.534    0.000
##    .alc2              0.827    0.025   32.687    0.000
##    .alc3              0.790    0.022   35.871    0.000
##    .alc4              0.500    0.025   19.858    0.000
```

Predicted linear model



Interpretations & calculations

- The expected alcohol use in Year 1 is 0.89. This is the average number of drinks per day at Year 1.
- Alcohol use was expected to decrease .09 drinks per day each year (the mean of the slope factor).
- Participants varied in their initial alcohol use and rate of change (based on variance of factors).
- Participants who reported higher alcohol use in Year 1 tended to decrease in alcohol use more quickly compared to participants who reported less alcohol use.
- Predicted means for each Year
  - 1 $\rightarrow$ Intercept $\rightarrow$ Year + 1 $\rightarrow$ Slope $\rightarrow$ Year
  - (0.89 * unstd pattern coef for Intercept) + (-.09 * unstd pattern coef for Slope)

95% confidence (compatible) intervals

How do we interpret 95% confidence intervals (CIs)? If you took 100 samples from your population and computed a 95% CI for each sample, **then we would expect that 95 of the 100 confidence intervals** would contain our **true parameter** (e.g., a population mean).

Alternatively, one can interpret a 95% CI as a **compatibility interval**. For example, a 95% CI **contains values for the true parameter that are reasonably compatible with your data, given your assumptions**.

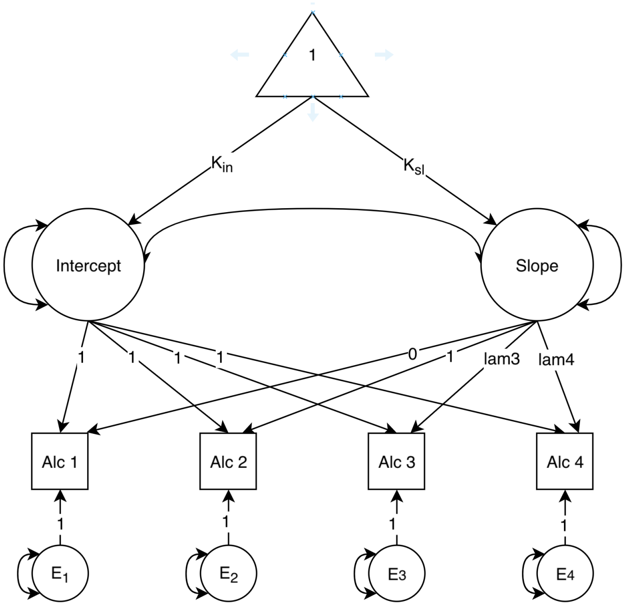
```
param <- parameterEstimates(line.fit.f, ci = TRUE)
subset(param, z != "NA", ) # drop parameters that were fixed
```

```
##      lhs op   rhs    est    se       z pvalue ci.lower ci.upper
## 9    int ~1          0.887 0.023  38.366      0    0.842    0.932
## 10 slope ~1         -0.089 0.008 -11.527      0   -0.104   -0.074
## 11   int ~~   int  1.354 0.051  26.509      0    1.254    1.454
## 12   int ~~ slope -0.214 0.016 -13.356      0   -0.246   -0.183
## 13 slope ~~ slope  0.068 0.007   9.467      0    0.054    0.082
## 14  alc1 ~~  alc1  1.499 0.049  30.534      0    1.403    1.595
## 15  alc2 ~~  alc2  0.827 0.025  32.687      0    0.778    0.877
## 16  alc3 ~~  alc3  0.790 0.022  35.871      0    0.747    0.833
## 17  alc4 ~~  alc4  0.500 0.025  19.858      0    0.451    0.550
```

Nonlinear curve fitting

- Requires at least 3 time points.
- An intercept (initial) and a slope (shape) factor.
- Indicators of these factors are the repeated measures.
- Repeated measures are endogenous variables, which allows us to adjust for the unexplained variation in our indicators when estimating the latent growth factors.

Nonlinear curve fitting - path diagram



Characteristics of this model

- Unstd. pattern coefficients for the intercept factor fixed to 1.
- Two unstd. pattern coefficients for the slope factor are constrained to 0 and 1 and the rest are freely estimated (allow for nonlinearity).
  - The 0 sets the origin to the initial performance at Trial 1.
  - The 1 sets the scale of the slope factors for UL identification.
- Covariance between the intercept and slope factor indicates the degree to which alcohol use at Year 1 covaries with change in alcohol use across time.
- The κ (kappa) are the means of the latent growth factors as free parameters.
  - κ_{in} is the mean alcohol use at Year 1, controlling for error variance at Year 1 and the **variance** represents variation around in initial alcohol use (**random intercept**).
  - κ_{sh} is the mean change in alcohol use from Year 1 to Year 2, controlling for error variance and the **variance** represents variation in change in alcohol use over time (**random slope**).

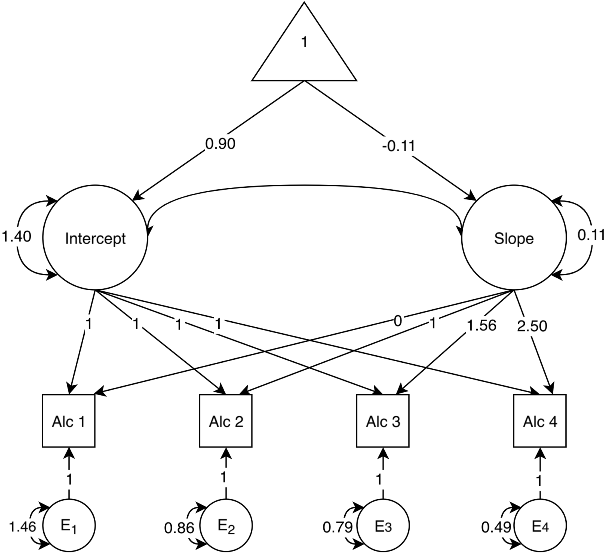
Curran, Muthen, & Hartford's model of alcohol use

```
nlc.mod <- "
# define factors
int =~ 1*alc1 + 1*alc2 + 1*alc3 + 1*alc4
slope =~ 0*alc1 + 1*alc2 + lam3*alc3 + lam4*alc4

# estimate factor means
int ~ 1
slope ~ 1

# estimate factor variance/covariance
int ~~ int + slope
slope ~~ slope

# assume variances are NOT equivalent across time
alc1 ~~ alc1
alc2 ~~ alc2
alc3 ~~ alc3
alc4 ~~ alc4
"
nlc.fit <- lavaan(model = nlc.mod, sample.cov = alcuse.cov,
                  sample.mean = alcuse.xbar, sample.nobs = 4052, start = "simple")
anova(line.fit.f, nlc.fit)

## Chi Square Difference Test
##
##            Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## nlc.fit     3 50887 50957 10.822
## line.fit.f  5 50890 50947 17.364     6.5421       2    0.03797 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(nlc.fit, fit.measures = TRUE)

##   Number of free parameters                         11
##   Number of observations                          4052
##
##   Estimator                                         ML
##   Model Fit Test Statistic                      10.822
##   Degrees of freedom                                 3
##   P-value (Chi-square)                           0.013
##
## User model versus baseline model:
##
##   Comparative Fit Index (CFI)                    0.998
##   Tucker-Lewis Index (TLI)                       0.996
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.025
##   90 Percent Confidence Interval          0.010  0.042
##   P-value RMSEA <= 0.05                          0.993
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.010
```

```r
summary(nlc.fit, fit.measures = TRUE)

## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   int =~
##     alc1              1.000
##     alc2              1.000
##     alc3              1.000
##     alc4              1.000
##   slope =~
##     alc1              0.000
##     alc2              1.000
##     alc3    (lam3)    1.558    0.167    9.354    0.000
##     alc4    (lam4)    2.502    0.319    7.832    0.000
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   int ~~
##     slope            -0.278    0.061   -4.583    0.000
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)
##     int               0.898    0.025   35.445    0.000
##     slope            -0.113    0.020   -5.757    0.000
##    .alc1              0.000
##    .alc2              0.000
##    .alc3              0.000
##    .alc4              0.000
```

```
summary(nlc.fit, fit.measures = TRUE)

## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##     int               1.395    0.079   17.626    0.000
##     slope             0.109    0.038    2.902    0.004
##    .alc1              1.457    0.070   20.818    0.000
##    .alc2              0.857    0.030   28.913    0.000
##    .alc3              0.786    0.022   35.265    0.000
##    .alc4              0.486    0.028   17.152    0.000
```

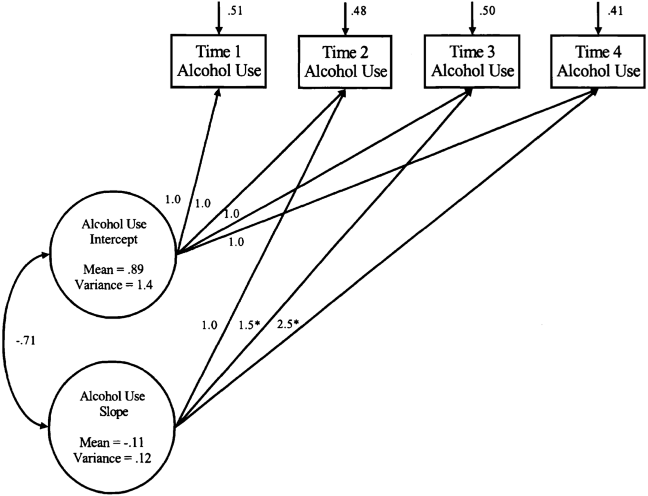


1
0.90
-0.11
1.40
Intercept
Slope
0.11
1
1
1
1
0
1
1.56
2.50
Alc 1
Alc 2
Alc 3
Alc 4
1
1
1
1
1.46
E1
0.86
E2
0.79
E3
0.49
E4

Interpretations & calculations

- The expected alcohol use in Year 1 is 0.90. This is the average number of drinks per day at Year 1.
- Alcohol use was expected to decrease .11 drinks from Year 1 to Year 2.
- Participants vary in their initial alcohol use and rate of change in alcohol use (based on variance of factors).
- Participants who reported higher alcohol use in Year 1 tended to decrease in alcohol use more quickly compared to participants who reported less alcohol use.
- Predicted means for each Year
  - 1 $\rightarrow$ Intercept $\rightarrow$ Year + 1 $\rightarrow$ Slope $\rightarrow$ Year
  - (0.90 * unstd pattern coef for Intercept) + (-.11 * unstd pattern coef for Slope)

Curran, Muthen, & Harford model



Predicted models (code)

```
dat.pred <- data.frame(obs.alc = c(0.92, 0.76, 0.73, 0.62),
                       year = 1:4,
                       line.alc = c(0.89 + 0:3 * -.09),
                       nlc.alc = c(0.90 + c(0, 1, 1.56, 2.50) * -.11))
ggplot(dat.pred, aes(x = year, y = obs.alc)) +
  geom_point() +
  geom_line(aes(y = line.alc, col = "Linear model")) +
  geom_line(aes(y = nlc.alc, col = "Nonlinear curve model")) +
  xlab("Year") +
  ylab("Average (reported) daily alcohol use over the last 7 days.") +
  theme_bw() +
  scale_color_discrete("")
```

Predicted models