SEM & Lavaan

Bang Quan Zheng
STAT 242
Multivariate Analysis with Latent Variables

October 11, 2019

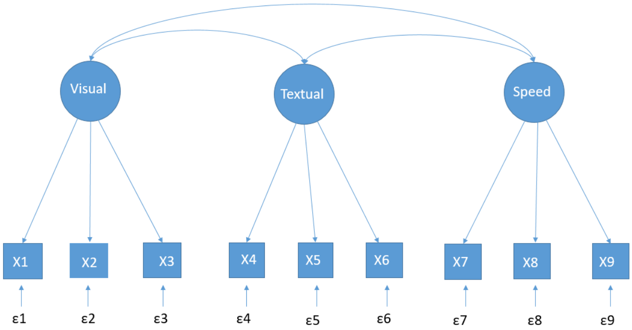
UCLA

Toy Data: Holzinger and Swineford (1939)

The classic Holzinger and Swineford (1939) dataset consists of mental ability test scores of 7th- and 8th-grade children. There are 9 variables, which are the scores of 9 tests. We use this widely used sample data to demonstrate the latent variable analysis.

UCLA

Example: Path Diagram (CFA)



UCLA

The Measurement Model

- a visual factor measured by 3 variables: x1, x2, and x3
- a textual factor measured by 3 variables: x4, x5, and x6
- a speed factor measured by 3 variables: x7, x8, and x9

UCLA

The Measurement Model

Latent variable = indicator1 + indicator2 + indicator3

visual = x1 + x2 + x3

textual = x4 + x5 + x6

speed = x7 + x8 + x9

Identification in SEM?

According to Bollen (1989: 88), "Investigation of identification begin with one or more equations relating known and unknown parameters. By "known" parameters, I do not mean that the exact values of the parameters are known. Rather, I mean parameters that are known to be identified." ..... "The ‘unknown’ parameters are the parameters whose identification status is not known."
According to EQS manual, "If the parameters were subject to any arbitrariness, it would be difficult to speak of them as true parameters that are to be estimated, since a wondering target would be involved." (p. 25)

UCLA

Identification in SEM?

- 3 latent factors
- 3 indicators per factor (3x3=9 indicators)
- Data point = Px(P+1)/2
- (9x10)/2= 45 data points
- 3 factor covariances, 9 factor loadings, 9 variances, the total is 21 free parameters
- Degrees of Freedom= (number of data point - number of parameter)
- (45-21)= 24 degrees of freedom

UCLA

Why 45 data points?

```
$`cov`
   x1    x2    x3    x4    x5    x6    x7    x8    x9
x1 1.358
x2 0.448 1.382
x3 0.590 0.327 1.275
x4 0.408 0.226 0.298 1.351
x5 0.454 0.252 0.331 1.090 1.660
x6 0.378 0.209 0.276 0.907 1.010 1.196
x7 0.262 0.145 0.191 0.173 0.193 0.161 1.183
x8 0.309 0.171 0.226 0.205 0.228 0.190 0.453 1.022
x9 0.284 0.157 0.207 0.188 0.209 0.174 0.415 0.490 1.015
```

$$\frac{Px(P+1)}{2} = \frac{9(9+1)}{2} = 45$$

UCLA

Why 45 data points?

$`cov`

	x1	x2	x3	x4	x5	x6	x7	x8	x9
x1	1.358								
x2	0.448	1.382							
x3	0.590	0.327	1.275						
x4	0.408	0.226	0.298	1.351					
x5	0.454	0.252	0.331	1.090	1.660				
x6	0.378	0.209	0.276	0.907	1.010	1.196			
x7	0.262	0.145	0.191	0.173	0.193	0.161	1.183		
x8	0.309	0.171	0.226	0.205	0.228	0.190	0.453	1.022	
x9	0.284	0.157	0.207	0.188	0.209	0.174	0.415	0.490	1.015

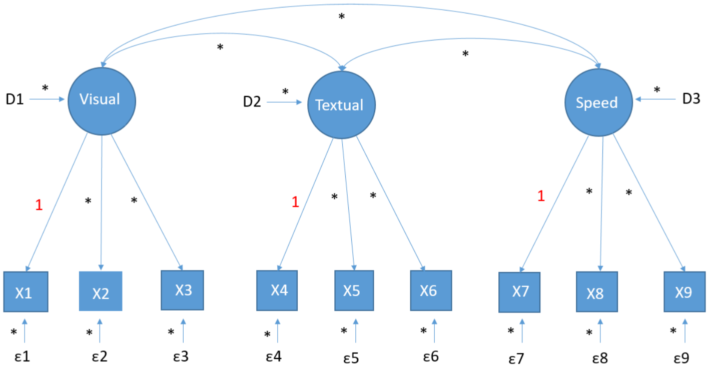
VARIANCES COVARIANCES

$$\frac{Px(P+1)}{2} = \frac{9(9+1)}{2} = 45$$

UCLA

Why 21 parameters?

* denotes that the parameter is free to be estimated
1 denotes the parameter is fixed
Therefore, we have total 21 free parameters to be estimated



UCLA

The Model Syntax

formula type	operator	mnemonic
latent variable definition	=~	is measured by
regression	~	is regressed on
(residual) (co)variance	~~	is correlated with
intercept	~ 1	intercept

UCLA

Running the model in R

```
install.packages("lavaan", dependencies=TRUE)
library(lavaan)
data(HolzingerSwineford1939)
HS.model <- ?
visual= x1 + x2 + x3
textual = x4 + x4 + x5
speed = x7 + x8 + x90
fit<-cfa(HS.model, data=HolzingerSwineford193)
summary(fit)
```

Note that the functions of cfa() and sem() are the same in Lavaan

UCLA

Output-1

```
  Estimator                                         ML
  Optimization method                           NLMINB
  Number of free parameters                         21

  Number of observations                           301

Model Test User Model:

  Test statistic                                85.306
  Degrees of freedom                                24
  P-value (Chi-square)                           0.000

Parameter Estimates:

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

UCLA

Output-2

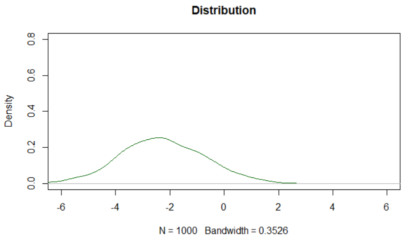
```
Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  visual =~
    x1                1.000
    x2                0.554    0.100    5.554    0.000
    x3                0.729    0.109    6.685    0.000
  textual =~
    x4                1.000
    x5                1.113    0.065   17.014    0.000
    x6                0.926    0.055   16.703    0.000
  speed =~
    x7                1.000
    x8                1.180    0.165    7.152    0.000
    x9                1.082    0.151    7.155    0.000

Covariances:
                   Estimate  Std.Err  z-value  P(>|z|)
  visual ~~
    textual           0.408    0.074    5.552    0.000
    speed             0.262    0.056    4.660    0.000
  textual ~~
    speed             0.173    0.049    3.518    0.000

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .x1                0.549    0.114    4.833    0.000
   .x2                1.134    0.102   11.146    0.000
   .x3                0.844    0.091    9.317    0.000
   .x4                0.371    0.048    7.779    0.000
   .x5                0.446    0.058    7.642    0.000
   .x6                0.356    0.043    8.277    0.000
   .x7                0.799    0.081    9.823    0.000
   .x8                0.488    0.074    6.573    0.000
   .x9                0.566    0.071    8.003    0.000
    visual            0.809    0.145    5.564    0.000
    textual           0.979    0.112    8.737    0.000
    speed             0.384    0.086    4.451    0.000
```

Goodness-of-Fit Index Summary

```
User Model versus Baseline Model:

  Comparative Fit Index (CFI)                    0.931
  Tucker-Lewis Index (TLI)                       0.896

Loglikelihood and Information Criteria:

  Loglikelihood user model (H0)              -3737.745
  Loglikelihood unrestricted model (H1)      -3695.092

  Akaike (AIC)                                7517.490
  Bayesian (BIC)                              7595.339
  Sample-size adjusted Bayesian (BIC)         7528.739

Root Mean Square Error of Approximation:

  RMSEA                                          0.092
  90 Percent confidence interval - lower         0.071
  90 Percent confidence interval - upper         0.114
  P-value RMSEA <= 0.05                          0.001

Standardized Root Mean Square Residual:

  SRMR                                           0.065
```

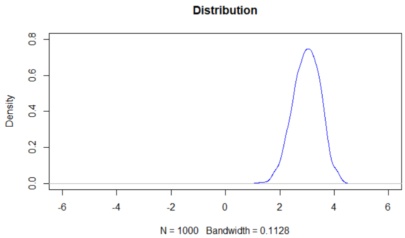
UCLA

Standardized Values



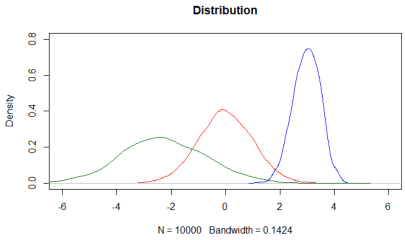
$$N \sim (0, 1)$$

Standardized Values



$$N \sim (0, 1)$$

Standardized Values



$N \sim (0, 1)$

Standardized parameter estimates

Latent Variables:

	Estimate	Std.Err	z-value	P(>|z|)	Std.lv	Std.all
visual =~						
x1	1.000				0.900	0.772
x2	0.554	0.100	5.554	0.000	0.498	0.424
x3	0.729	0.109	6.685	0.000	0.656	0.581
textual =~						
x4	1.000				0.990	0.852
x5	1.113	0.065	17.014	0.000	1.102	0.855
x6	0.926	0.055	16.703	0.000	0.917	0.838
speed =~						
x7	1.000				0.619	0.570
x8	1.180	0.165	7.152	0.000	0.731	0.723
x9	1.082	0.151	7.155	0.000	0.670	0.665

Covariances:

	Estimate	Std.Err	z-value	P(>|z|)	Std.lv	Std.all
visual ~~						
textual	0.408	0.074	5.552	0.000	0.459	0.459
speed	0.262	0.056	4.660	0.000	0.471	0.471
textual ~~						
speed	0.173	0.049	3.518	0.000	0.283	0.283

Variances:

	Estimate	Std.Err	z-value	P(>|z|)	Std.lv	Std.all
.x1	0.549	0.114	4.833	0.000	0.549	0.404
.x2	1.134	0.102	11.146	0.000	1.134	0.821
.x3	0.844	0.091	9.317	0.000	0.844	0.662
.x4	0.371	0.048	7.779	0.000	0.371	0.275
.x5	0.446	0.058	7.642	0.000	0.446	0.269
.x6	0.356	0.043	8.277	0.000	0.356	0.298
.x7	0.799	0.081	9.823	0.000	0.799	0.676
.x8	0.488	0.074	6.573	0.000	0.488	0.477
.x9	0.566	0.071	8.003	0.000	0.566	0.558
visual	0.809	0.145	5.564	0.000	1.000	1.000
textual	0.979	0.112	8.737	0.000	1.000	1.000
speed	0.384	0.086	4.451	0.000	1.000	1.000

UCLA

StandardizedSolution(fit)

```
> standardizedSolution(fit)
       lhs op     rhs est.std    se      z pvalue ci.lower ci.upper
1   visual =~      x1   0.772 0.055 14.041      0    0.664    0.880
2   visual =~      x2   0.424 0.060  7.105      0    0.307    0.540
3   visual =~      x3   0.581 0.055 10.539      0    0.473    0.689
4  textual =~      x4   0.852 0.023 37.776      0    0.807    0.896
5  textual =~      x5   0.855 0.022 38.273      0    0.811    0.899
6  textual =~      x6   0.838 0.023 35.881      0    0.792    0.884
7    speed =~      x7   0.570 0.053 10.714      0    0.465    0.674
8    speed =~      x8   0.723 0.051 14.309      0    0.624    0.822
9    speed =~      x9   0.665 0.051 13.015      0    0.565    0.765
10      x1 ~~      x1   0.404 0.085  4.763      0    0.238    0.571
11      x2 ~~      x2   0.821 0.051 16.246      0    0.722    0.920
12      x3 ~~      x3   0.662 0.064 10.334      0    0.537    0.788
13      x4 ~~      x4   0.275 0.038  7.157      0    0.200    0.350
14      x5 ~~      x5   0.269 0.038  7.037      0    0.194    0.344
15      x6 ~~      x6   0.298 0.039  7.606      0    0.221    0.374
16      x7 ~~      x7   0.676 0.061 11.160      0    0.557    0.794
17      x8 ~~      x8   0.477 0.073  6.531      0    0.334    0.620
18      x9 ~~      x9   0.558 0.068  8.208      0    0.425    0.691
19  visual ~~  visual   1.000 0.000     NA     NA    1.000    1.000
20 textual ~~ textual   1.000 0.000     NA     NA    1.000    1.000
21   speed ~~   speed   1.000 0.000     NA     NA    1.000    1.000
22  visual ~~ textual   0.459 0.064  7.189      0    0.334    0.584
23  visual ~~   speed   0.471 0.073  6.461      0    0.328    0.613
24 textual ~~   speed   0.283 0.069  4.117      0    0.148    0.418
```

UCLA

A Basic Logic of Covariance Structure Estimation

Σ is a model implied covariance matrix
S is a sample covariance matrix
SEM test statistic tests the degree to the sample covariance matrix **S** is reproduced by the estimated model covariance matrix $\hat{\Sigma}$, by setting
Ho : $\Sigma = \Sigma(\hat{\theta})$

Estimators

- Maximumum Likelihood Estimator

$$F_{ML} = log|\Sigma(\theta)| - log|S_N| + tr(S_N\Sigma(\theta)^{-1}) - p$$

- Reweighted Least Squares (Browne, 1985)

$$RLS = tr[(\mathbf{S} - \Sigma(\theta))\hat{\Sigma}_{ML}^{-1}]^2$$

- Regularized GLS (Arruda and Bentler, 2017)

$$RGLS = tr[(\mathbf{S} - \Sigma(\theta))\hat{\Sigma}_{REG}^{-1}]^2$$

UCLA

ML Estimator–the default estimator in levaan

$$F_{ML} = log|\Sigma(\theta)| - log|S_N| + tr(S_N \Sigma(\theta)^{-1}) - p$$

$$\hat{\theta}_{ML} = argmin\, F_{ML}(\theta)$$

Therefore,

$$\Sigma(\hat{\theta}_{ML}) = \hat{\Lambda}\hat{\Phi}\hat{\Lambda}' + \hat{\Psi}$$

$$\hat{\Sigma}_{ML} = \Sigma(\hat{\theta}_{ML})$$

UCLA

Other Estimator Options

- `"GLS"`: generalized least squares. For complete data only.
- `"WLS"`: weighted least squares (sometimes called ADF estimation). For complete data only.
- `"DWLS"`: diagonally weighted least squares
- `"ULS"`: unweighted least squares

UCLA

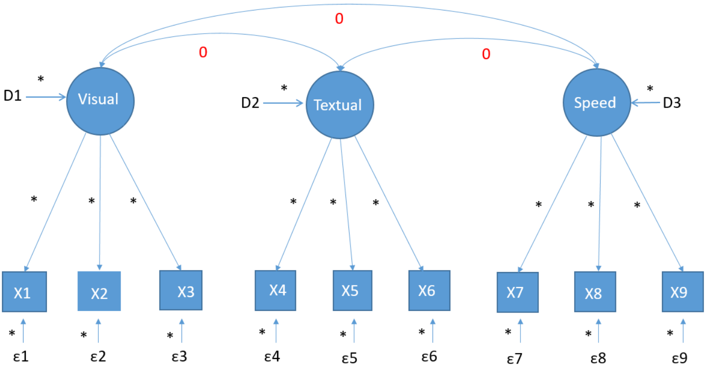
Other Estimators (R code)

```
library(lavaan)
data(HolzingerSwineford1939)
HS.model <- '
visual   =~ x1 + x2 + x3
textual =~ x4 + x5 + x6
speed    =~ x7 + x8 + x9'

fit_ML <- cfa(HS.model, data=HolzingerSwineford1939, estimator = "ML")
fit_GLS <- cfa(HS.model, data=HolzingerSwineford1939, estimator = "GLS")
fit_WLS <- cfa(HS.model, data=HolzingerSwineford1939, estimator = "WLS")
fit_ULS <- cfa(HS.model, data=HolzingerSwineford1939, estimator = "ULS")

summary(fit_ML)
summary(fit_GLS)
summary(fit_WLS)
summary(fit_ULS)
```

UCLA

Fixing covariances between latent factors (Diagram)

Fixing all covariances between latent variables



UCLA

Fixing covariances between latent factors (Output)

```
Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  visual =~
    x1                1.000
    x2                0.778    0.141    5.532    0.000
    x3                1.107    0.214    5.173    0.000
  textual =~
    x4                1.000
    x5                1.133    0.067   16.906    0.000
    x6                0.924    0.056   16.391    0.000
  speed =~
    x7                1.000
    x8                1.225    0.190    6.460    0.000
    x9                0.854    0.121    7.046    0.000

Covariances:
                   Estimate  Std.Err  z-value  P(>|z|)
  visual ~~
    textual           0.000
    speed             0.000
  textual ~~
    speed             0.000

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .x1                0.835    0.118    7.064    0.000
   .x2                1.065    0.105   10.177    0.000
   .x3                0.633    0.129    4.899    0.000
   .x4                0.382    0.049    7.805    0.000
   .x5                0.416    0.059    7.038    0.000
   .x6                0.369    0.044    8.367    0.000
   .x7                0.746    0.086    8.650    0.000
   .x8                0.366    0.097    3.794    0.000
   .x9                0.696    0.072    9.640    0.000
    visual            0.524    0.130    4.021    0.000
    textual           0.969    0.112    8.640    0.000
    speed             0.437    0.097    4.520    0.000
```

UCLA

Fixing covariances between latent factors (R code)

```
fit.HS.ortho <- cfa(HS.model,data =
HolzingerSwineford1939,orthogonal = TRUE)
```

UCLA

Fix all variances of latent variables

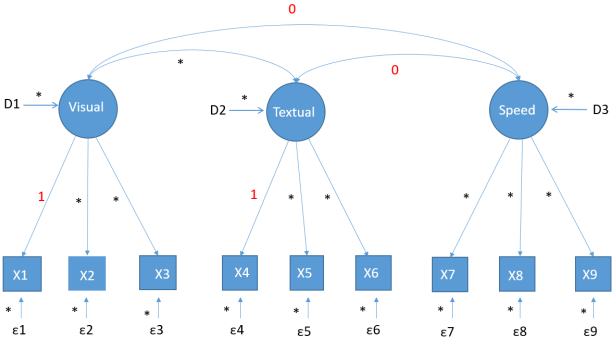
fit.HS.ortho <- cfa(HS.model,data = HolzingerSwineford1939, std.lv = TRUE)

Fix variances of latent variables

```
Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  visual =~
    x1                0.900    0.081   11.128    0.000
    x2                0.498    0.077    6.429    0.000
    x3                0.656    0.074    8.817    0.000
  textual =~
    x4                0.990    0.057   17.474    0.000
    x5                1.102    0.063   17.576    0.000
    x6                0.917    0.054   17.082    0.000
  speed =~
    x7                0.619    0.070    8.903    0.000
    x8                0.731    0.066   11.090    0.000
    x9                0.670    0.065   10.305    0.000

Covariances:
                   Estimate  Std.Err  z-value  P(>|z|)
  visual ~~
    textual           0.459    0.064    7.189    0.000
    speed             0.471    0.073    6.461    0.000
  textual ~~
    speed             0.283    0.069    4.117    0.000

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .x1                0.549    0.114    4.833    0.000
   .x2                1.134    0.102   11.146    0.000
   .x3                0.844    0.091    9.317    0.000
   .x4                0.371    0.048    7.779    0.000
   .x5                0.446    0.058    7.642    0.000
   .x6                0.356    0.043    8.277    0.000
   .x7                0.799    0.081    9.823    0.000
   .x8                0.488    0.074    6.573    0.000
   .x9                0.566    0.071    8.003    0.000
    visual            1.000
    textual           1.000
    speed             1.000
```

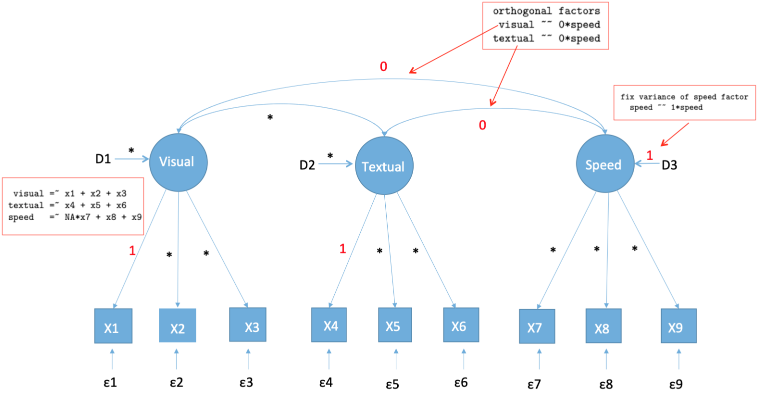
UCLA

Fixing selected parameters



UCLA

Fixing selected parameters



UCLA

Fixing selected parameters (R code)

```
model2<- '
visual =~ x1 + x2 + x3
textual =~ x4 + x5 + x6
speed   =~ NA*x7 + x8 + x9

# orthogonal factors

visual ~~ 0*speed
textual ~~ 0*speed

# fix variance of speed factor

speed ~~ 1*speed'

fit2 <- cfa(model2, data=HolzingerSwineford1939)
summary(fit2)
```

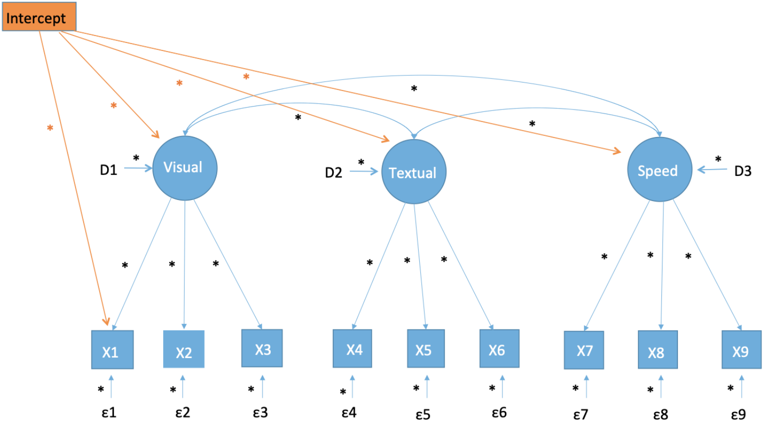
UCLA

Fixing selected parameters (R code)

```
Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  visual =~
    x1                1.000
    x2                0.559    0.105    5.300    0.000
    x3                0.708    0.118    6.004    0.000
  textual =~
    x4                1.000
    x5                1.111    0.065   16.996    0.000
    x6                0.925    0.055   16.703    0.000
  speed =~
    x7                0.661    0.073    9.040    0.000
    x8                0.810    0.074   10.899    0.000
    x9                0.565    0.066    8.509    0.000

Covariances:
                   Estimate  Std.Err  z-value  P(>|z|)
  visual ~~
    speed             0.000
  textual ~~
    speed             0.000
  visual ~~
    textual           0.414    0.074    5.562    0.000

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
    speed             1.000
   .x1                0.536    0.129    4.155    0.000
   .x2                1.125    0.103   10.965    0.000
   .x3                0.863    0.095    9.085    0.000
   .x4                0.369    0.048    7.735    0.000
   .x5                0.449    0.059    7.662    0.000
   .x6                0.356    0.043    8.263    0.000
   .x7                0.746    0.086    8.650    0.000
   .x8                0.366    0.097    3.794    0.000
   .x9                0.696    0.072    9.640    0.000
    visual            0.822    0.158    5.188    0.000
    textual           0.981    0.112    8.745    0.000
```

UCLA

Means Structure Model (path diagram)



UCLA

Means Structure Model (R code)

```
means_model<-'visual =~ x1 + x2 + x3
textual =~ x4 + x5 + x6
speed   =~ x7 + x8 + x9

x1 ~ 1
x2 ~ 1
x3 ~ 1
x4 ~ 1
x5 ~ 1
x6 ~ 1
x7 ~ 1
x8 ~ 1
x9 ~ 1
'
fit_means <- cfa(means_model,data = HolzingerSwineford1939)
summary(fit_means)
```

UCLA

Means Structure Model (output)

Note that we cannot estimate both the intercepts of LV and indicators at the same time

```
Covariances:
                   Estimate  Std.Err  z-value  P(>|z|)
  visual ~~
    textual           0.408    0.074    5.552    0.000
    speed             0.262    0.056    4.660    0.000
  textual ~~
    speed             0.173    0.049    3.518    0.000

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)
   .x1                4.936    0.067   73.473    0.000
   .x2                6.088    0.068   89.855    0.000
   .x3                2.250    0.065   34.579    0.000
   .x4                3.061    0.067   45.694    0.000
   .x5                4.341    0.074   58.452    0.000
   .x6                2.186    0.063   34.667    0.000
   .x7                4.186    0.063   66.766    0.000
   .x8                5.527    0.058   94.854    0.000
   .x9                5.374    0.058   92.546    0.000
    visual            0.000
    textual           0.000
    speed             0.000

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .x1                0.549    0.114    4.833    0.000
   .x2                1.134    0.102   11.146    0.000
   .x3                0.844    0.091    9.317    0.000
   .x4                0.371    0.048    7.779    0.000
   .x5                0.446    0.058    7.642    0.000
   .x6                0.356    0.043    8.277    0.000
   .x7                0.799    0.081    9.823    0.000
   .x8                0.488    0.074    6.573    0.000
   .x9                0.566    0.071    8.003    0.000
    visual            0.809    0.145    5.564    0.000
    textual           0.979    0.112    8.737    0.000
    speed             0.384    0.086    4.451    0.000
```

By default, Levaan sets latent variable intercepts to be zero

UCLA

Extracting sample covariance matrix

```
> fitted(fit_means)
$cov
   x1    x2    x3    x4    x5    x6    x7    x8    x9
x1 1.358
x2 0.448 1.382
x3 0.590 0.327 1.275
x4 0.408 0.226 0.298 1.351
x5 0.454 0.252 0.331 1.090 1.660
x6 0.378 0.209 0.276 0.907 1.010 1.196
x7 0.262 0.145 0.191 0.173 0.193 0.161 1.183
x8 0.309 0.171 0.226 0.205 0.228 0.190 0.453 1.022
x9 0.284 0.157 0.207 0.188 0.209 0.174 0.415 0.490 1.015

$mean
   x1    x2    x3    x4    x5    x6    x7    x8    x9
4.936 6.088 2.250 3.061 4.341 2.186 4.186 5.527 5.374
```

UCLA

Means structure with fixed intercept values (R code)

EX. We want the means of x1, x2, x3, x4= 0.5

```
means_model<-'visual =~ x1 + x2 + x3
textual =~ x4 + x5 + x6
speed   =~ x7 + x8 + x9

# intecept with fixed lues
x1 + x2 + x3 + x4 ~ 0.5*1'

fit_meansfixed <- cfa(means_model,data = HolzingerSwineford1939)
summary(fit_meansfixed)
```

Fixing values at 0.5

UCLA

Means structure with fixed intercept values

Intercepts:

	Estimate	Std.Err	z-value	P(>|z|)
.x1	0.500			
.x2	0.500			
.x3	0.500			
.x4	0.500			
.x5	1.625	0.050	32.530	0.000
.x6	-0.083	0.043	-1.932	0.053
.x7	3.083	0.061	50.440	0.000
.x8	4.222	0.056	75.567	0.000
.x9	4.216	0.056	75.038	0.000
visual	0.000			
textual	0.000			
speed	0.000			

Variances:

	Estimate	Std.Err	z-value	P(>|z|)
.x1	0.442	0.105	4.214	0.000
.x2	1.757	0.208	8.439	0.000
.x3	0.964	0.083	11.677	0.000
.x4	0.355	0.045	7.915	0.000
.x5	0.463	0.055	8.479	0.000
.x6	0.361	0.041	8.891	0.000
.x7	0.791	0.076	10.380	0.000
.x8	0.473	0.061	7.730	0.000
.x9	0.582	0.062	9.389	0.000
visual	20.593	1.717	11.993	0.000
textual	7.554	0.645	11.713	0.000
speed	1.610	0.189	8.510	0.000

UCLA

Latent variables intercepts

```
means_model<-'visual =~ x1 + x2 + x3
textual =~ x4 + x5 + x6
speed   =~ x7 + x8 + x9

# intecept with fixed lues
x1 + x2 + x3 + x4 + x5 +x6 +x7 +x8 +x9 ~ 0*1
visual+textual+speed~1
'
fit_meanslv <- cfa(means_model,data = HolzingerSwineford1939)
summary(fit_meanslv)
```

UCLA

Latent variables intercepts

Intercepts:

	Estimate	Std.Err	z-value	P(>|z|)
.x1	0.000			
.x2	0.000			
.x3	0.000			
.x4	0.000			
.x5	0.000			
.x6	0.000			
.x7	0.000			
.x8	0.000			
.x9	0.000			
visual	4.945	0.065	76.241	0.000
textual	3.075	0.064	47.778	0.000
speed	4.191	0.061	68.343	0.000

Variances:

	Estimate	Std.Err	z-value	P(>|z|)
.x1	0.830	0.087	9.496	0.000
.x2	0.949	0.113	8.422	0.000
.x3	1.044	0.088	11.845	0.000
.x4	0.465	0.050	9.364	0.000
.x5	0.263	0.063	4.144	0.000
.x6	0.516	0.047	11.065	0.000
.x7	0.837	0.076	10.967	0.000
.x8	0.503	0.060	8.328	0.000
.x9	0.539	0.061	8.818	0.000
visual	0.439	0.068	6.427	0.000
textual	0.800	0.076	10.523	0.000
speed	0.302	0.037	8.192	0.000

UCLA

Latent variables intercepts

Intercepts:

	Estimate	Std.Err	z-value	P(>|z|)
.x1	0.000			
.x2	0.000			
.x3	0.000			
.x4	0.000			
.x5	0.000			
.x6	0.000			
.x7	0.000			
.x8	0.000			
.x9	0.000			
visual	4.945	0.065	76.241	0.000
textual	3.075	0.064	47.778	0.000
speed	4.191	0.061	68.343	0.000

We have to hold these intercepts to zero to estimate LV intercepts

Variances:

	Estimate	Std.Err	z-value	P(>|z|)
.x1	0.830	0.087	9.496	0.000
.x2	0.949	0.113	8.422	0.000
.x3	1.044	0.088	11.845	0.000
.x4	0.465	0.050	9.364	0.000
.x5	0.263	0.063	4.144	0.000
.x6	0.516	0.047	11.065	0.000
.x7	0.837	0.076	10.967	0.000
.x8	0.503	0.060	8.328	0.000
.x9	0.539	0.061	8.818	0.000
visual	0.439	0.068	6.427	0.000
textual	0.800	0.076	10.523	0.000
speed	0.302	0.037	8.192	0.000

UCLA

ML Robust Standard Errors Scaled Test statistics

- "MLM": maximum likelihood estimation with robust standard errors and a Satorra-Bentler scaled test statistic. For complete data only.
- "MLMVS": maximum likelihood estimation with robust standard errors and a mean- and variance adjusted test statistic (aka the Satterthwaite approach). For complete data only.
- "MLMV": maximum likelihood estimation with robust standard errors and a mean- and variance adjusted test statistic (using a scale-shifted approach). For complete data only.
- "MLF": for maximum likelihood estimation with standard errors based on the first-order derivatives, and a conventional test statistic. For both complete and incomplete data.
- "MLR": maximum likelihood estimation with robust (Huber-White) standard errors and a scaled test statistic that is (asymptotically) equal to the Yuan-Bentler test statistic. For both complete and incomplete data.

UCLA

ML Robust Standard Errors Scaled Test statistics (R code)

```r
library(lavaan)
data(HolzingerSwineford1939)
HS.model <- 'visual  =~ x1 + x2 + x3
          textual =~ x4 + x5 + x6
          speed   =~ x7 + x8 + x9'

fit_MLM <- cfa(HS.model, data=HolzingerSwineford1939, estimator = "MLM")
fit_MLMVS <- cfa(HS.model, data=HolzingerSwineford1939, estimator = "MLMVS")
fit_MLMVS <- cfa(HS.model, data=HolzingerSwineford1939, estimator = "MLMVS")
fit_MLMV <- cfa(HS.model, data=HolzingerSwineford1939, estimator = "MLMV")
fit_MLF <- cfa(HS.model, data=HolzingerSwineford1939, estimator = "MLF")
fit_MLR <- cfa(HS.model, data=HolzingerSwineford1939, estimator = "MLR")
```

UCLA

Missing values and standard errors

When we have missing values in data, we can use missing="ML" command to fix them.
In this case, expected information will be used to calculate standard errors. However, we can choose to calculate standard errors based on observed information (Hessian information)

```
fit1 <- cfa(HS.model, data=HolzingerSwineford1939,
information="observed", estimator = "ML", se="robust.sem")
fit2 <- cfa(HS.model, data=HolzingerSwineford1939,
information="expected", estimator = "ML", se="robust.sem")
```

UCLA

Test Statistic Options

- "standard", a conventional chi-square test is computed
- "Satorra.Bentler", a Satorra-Bentler scaled test statistic is computed
- "Yuan.Bentler", a Yuan-Bentler scaled test statis-tic is computed.
- "Yuan.Bentler.Mplus", a test statistic is computed that is asymptotically equal to the Yuan-Bentler scaled test statistic

UCLA

Test Statistic Options (R code)

```
fit_1 <- cfa(HS.model, data=HolzingerSwineford1939, test="standard")
fit_2 <- cfa(HS.model, data=HolzingerSwineford1939, test="Satorra.Bentler")
fit_3 <- cfa(HS.model, data=HolzingerSwineford1939, test="Yuan.Bentler")
fit_4 <- cfa(HS.model, data=HolzingerSwineford1939, test="Yuan.Bentler.Mplus")
```

UCLA

Test Statistic Options–an example

```
  Optimization method                           NLMINB
  Number of free parameters                         21

  Number of observations                           301

  Estimator                                         ML      Robust
  Model Fit Test Statistic                      85.306      92.281
  Degrees of freedom                                24          24
  P-value (Chi-square)                           0.000       0.000
  Scaling correction factor                                  0.924
    for the Satorra-Bentler correction

Parameter Estimates:

  Information                                 Observed
  Observed information based on                Hessian
  Standard Errors                             Standard
```

UCLA

References

All these information in this presentation come from:
http://lavaan.ugent.be/tutorial/tutorial.pdf
https://cran.r-project.org/web/packages/lavaan/lavaan.pdf
Structural Equational with Latent Variables (1989) by Kenneth Bollen
EQS Manual by Peter Bentler

UCLA

Thank You

UCLA