# Confirmatory Factor Analysis

Ivan Jacob Agaloos Pesigan

## Model

Let the confirmatory factor analysis model be given by

$$y = \nu + \Lambda \eta + \varepsilon. \tag{1}$$

The **random variables** are $y$, $\eta$, and $\varepsilon$ where

- $y$ which is a vector of observed random variables,
- $\eta$ which is a vector of latent random variables, and
- $\varepsilon$ is a vector of random error terms.

$$y = \begin{pmatrix} Y_1 \\ \vdots \\ Y_j \end{pmatrix} \quad y \sim \mathcal{N}\left(\mu, \Sigma\right) \tag{2}$$

$$\eta = \begin{pmatrix} \eta_1 \\ \vdots \\ \eta_k \end{pmatrix} \quad \eta \sim \mathcal{N}\left(\alpha, \Psi\right) \tag{3}$$

$$\varepsilon = \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_j \end{pmatrix} \quad \varepsilon \sim \mathcal{N}\left(0, \Theta\right) \tag{4}$$

The **fixed parameters** are $\nu$, $\Lambda$, $\alpha$, $\Psi$, and $\Theta$ where

- $\nu$ is the vector of intercepts,
- $\Lambda$ is the matrix of factor loadings,
- $\alpha$ is the mean vector of $\eta$,
- $\Psi$ is the covariance matrix of $\eta$, and
- $\Theta$ is the covariance matrix of $\varepsilon$.

The **model-implied mean vector** of $y$ is given by

$$\mu\left(\theta\right) = \nu + \Lambda\alpha. \tag{5}$$

The **model-implied covariance matrix** of $y$ is given by

$$\Sigma\left(\theta\right) = \Lambda\Psi\Lambda' + \Theta. \tag{6}$$

# Example

Let $j = 6$ and $k = 2$. Let the population vector of intercepts $\nu$ be given by

$$\nu = \begin{pmatrix} \nu_1 = 0 \\ \nu_2 = 0 \\ \nu_3 = 0 \\ \nu_4 = 0 \\ \nu_5 = 0 \\ \nu_6 = 0 \end{pmatrix}. \tag{7}$$

Let the population mean vector $\alpha$ be given by

$$\alpha = \begin{pmatrix} \alpha_1 = 0 \\ \alpha_2 = 0 \end{pmatrix}. \tag{8}$$

Let the population factor loading matrix $\Lambda$ be given by

$$\Lambda = \begin{pmatrix} \lambda_{11} = 1 & 0 \\ \lambda_{21} = 1 & 0 \\ \lambda_{31} = 1 & 0 \\ 0 & \lambda_{42} = 1 \\ 0 & \lambda_{52} = 1 \\ 0 & \lambda_{62} = 1 \end{pmatrix}. \tag{9}$$

Let the lower diagonal elements of the population covariance matrix of $\varepsilon$ be given by

$$\Theta = \begin{pmatrix} \theta_{11} = 0.25 & & & & & \text{Sym.} \\ 0 & \theta_{22} = 0.25 & & & & \\ 0 & 0 & \theta_{33} = 0.25 & & & \\ 0 & 0 & 0 & \theta_{44} = 0.25 & & \\ 0 & 0 & 0 & 0 & \theta_{55} = 0.25 & \\ 0 & 0 & 0 & 0 & 0 & \theta_{66} = 0.25 \end{pmatrix}. \tag{10}$$

Let the lower diagonal elements of the population covariance matrix of $\eta$ be given by

$$\Psi = \begin{pmatrix} \psi_{11} = 1 & \text{Sym.} \\ \psi_{21} = 0.5 & \psi_{22} = 1 \end{pmatrix}. \tag{11}$$

The model-implied covariance matrix of $y$ is given by

$$\begin{aligned}
\Sigma(\theta) &= \Lambda\Psi\Lambda' + \Theta \\
\Sigma(\theta) &= \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{pmatrix}' + \begin{pmatrix} 0.25 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.25 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.25 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.25 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.25 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.25 \end{pmatrix} \\
\Sigma(\theta) &= \begin{pmatrix} 1.25 & 1 & 1 & 0.5 & 0.5 & 0.5 \\ 1 & 1.25 & 1 & 0.5 & 0.5 & 0.5 \\ 1 & 1 & 1.25 & 0.5 & 0.5 & 0.5 \\ 0.5 & 0.5 & 0.5 & 1.25 & 1 & 1 \\ 0.5 & 0.5 & 0.5 & 1 & 1.25 & 1 \\ 0.5 & 0.5 & 0.5 & 1 & 1 & 1.25 \end{pmatrix}.
\end{aligned} \tag{12}$$

The model-implied mean vector is given by

$$
\begin{aligned}
\mu(\theta) &= \nu + \Lambda\alpha \\
\mu(\theta) &= \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\
\mu(\theta) &= \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}.
\end{aligned}
\tag{13}
$$

Using the model-implied mean vector and covariance matrix, we can simulate sample data using the `MASS::mvrnorm()` function.

```
Y <- MASS::mvrnorm(
  n = n,
  mu = mu,
  Sigma = Sigma
)
```

The sample moments are given by

```
(mu_Y <- colMeans(Y))

#>          Y1          Y2          Y3          Y4          Y5          Y6
#> 0.021471929 0.044589169 0.049140629 0.028624219 0.015476623 0.001100356
```

```r
(Sigma_Y <- var(Y))
```

```
#>           Y1        Y2        Y3        Y4        Y5        Y6
#> Y1 1.1792897 0.9696425 0.9632292 0.4238055 0.4074245 0.4391180
#> Y2 0.9696425 1.2551607 0.9792525 0.4709927 0.4393695 0.4485927
#> Y3 0.9632292 0.9792525 1.2029355 0.4551631 0.4169896 0.4318705
#> Y4 0.4238055 0.4709927 0.4551631 1.2395617 0.9717966 0.9988268
#> Y5 0.4074245 0.4393695 0.4169896 0.9717966 1.2418737 0.9655866
#> Y6 0.4391180 0.4485927 0.4318705 0.9988268 0.9655866 1.2540873
```

Constrain the first factor loading for each latent variable to 1.

```r
model_cfa1 <- "
  eta1 =~ Y1 + Y2 + Y3
  eta2 =~ Y4 + Y5 + Y6
"
fit_lav1 <- lavaan::cfa(
  model = model_cfa1,
  data = as.data.frame(Y),
  meanstructure = TRUE
)
lavaan::summary(fit_lav1)
```

```
#> lavaan 0.6.15 ended normally after 28 iterations
#>
#>   Estimator                                         ML
#>   Optimization method                           NLMINB
#>   Number of model parameters                        19
#>
#>   Number of observations                           500
```

```
#>
#> Model Test User Model:
#>
#>   Test statistic                                 5.082
#>   Degrees of freedom                                 8
#>   P-value (Chi-square)                           0.749
#>
#> Parameter Estimates:
#>
#>   Standard errors                             Standard
#>   Information                                 Expected
#>   Information saturated (h1) model          Structured
#>
#> Latent Variables:
#>                    Estimate  Std.Err  z-value  P(>|z|)
#>   eta1 =~
#>     Y1                1.000
#>     Y2                1.019    0.036   28.319    0.000
#>     Y3                1.011    0.035   28.929    0.000
#>   eta2 =~
#>     Y4                1.000
#>     Y5                0.965    0.036   26.861    0.000
#>     Y6                0.992    0.036   27.857    0.000
#>
#> Covariances:
#>                    Estimate  Std.Err  z-value  P(>|z|)
#>   eta1 ~~
#>     eta2              0.438    0.053    8.309    0.000
```

```
#>
#> Intercepts:
#>                    Estimate  Std.Err  z-value  P(>|z|)
#>    .Y1                0.021    0.049    0.443    0.658
#>    .Y2                0.045    0.050    0.891    0.373
#>    .Y3                0.049    0.049    1.003    0.316
#>    .Y4                0.029    0.050    0.575    0.565
#>    .Y5                0.015    0.050    0.311    0.756
#>    .Y6                0.001    0.050    0.022    0.982
#>     eta1              0.000
#>     eta2              0.000
#>
#> Variances:
#>                    Estimate  Std.Err  z-value  P(>|z|)
#>    .Y1                0.227    0.023    9.929    0.000
#>    .Y2                0.266    0.025   10.616    0.000
#>    .Y3                0.230    0.023    9.870    0.000
#>    .Y4                0.232    0.025    9.202    0.000
#>    .Y5                0.303    0.027   11.162    0.000
#>    .Y6                0.262    0.026   10.004    0.000
#>     eta1              0.950    0.075   12.632    0.000
#>     eta2              1.005    0.080   12.637    0.000
```

Constrain the variances of latent variables to 1.

```
model_cfa2 <- "
  eta1 =~ NA * Y1 + Y2 + Y3
  eta2 =~ NA * Y4 + Y5 + Y6
  eta1 ~~ 1 * eta1
```

```
  eta2 ~~ 1 * eta2
"
fit_lav2 <- lavaan::cfa(
  model = model_cfa2,
  data = as.data.frame(Y),
  meanstructure = TRUE
)
lavaan::summary(fit_lav2)
```

```
#>
#> Latent Variables:
#>                    Estimate  Std.Err  z-value  P(>|z|)
#>   eta1 =~
#>     Y1                0.974    0.039   25.263    0.000
#>     Y2                0.993    0.040   24.786    0.000
#>     Y3                0.985    0.039   25.303    0.000
#>   eta2 =~
#>     Y4                1.003    0.040   25.274    0.000
#>     Y5                0.967    0.041   23.875    0.000
#>     Y6                0.995    0.040   24.747    0.000
#>
#> Covariances:
#>                    Estimate  Std.Err  z-value  P(>|z|)
#>   eta1 ~~
#>     eta2              0.448    0.040   11.317    0.000
#>
#> Intercepts:
#>                    Estimate  Std.Err  z-value  P(>|z|)
#>    .Y1                0.021    0.049    0.443    0.658
#>    .Y2                0.045    0.050    0.891    0.373
#>    .Y3                0.049    0.049    1.003    0.316
#>    .Y4                0.029    0.050    0.575    0.565
#>    .Y5                0.015    0.050    0.311    0.756
#>    .Y6                0.001    0.050    0.022    0.982
#>     eta1              0.000
#>     eta2              0.000
#>
```

```
#> Variances:
#>                    Estimate  Std.Err  z-value  P(>|z|)
#>     eta1              1.000
#>     eta2              1.000
#>    .Y1                0.227    0.023    9.929    0.000
#>    .Y2                0.266    0.025   10.616    0.000
#>    .Y3                0.230    0.023    9.870    0.000
#>    .Y4                0.232    0.025    9.202    0.000
#>    .Y5                0.303    0.027   11.162    0.000
#>    .Y6                0.262    0.026   10.004    0.000
```

## References

R Core Team. (2023). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. Vienna, Austria. https://www.R-project.org/