

**การวิเคราะห์องค์ประกอบเชิงยืนยัน
ตอนที่ 3 (Confirmatory Factor
Analysis: Part 3)**

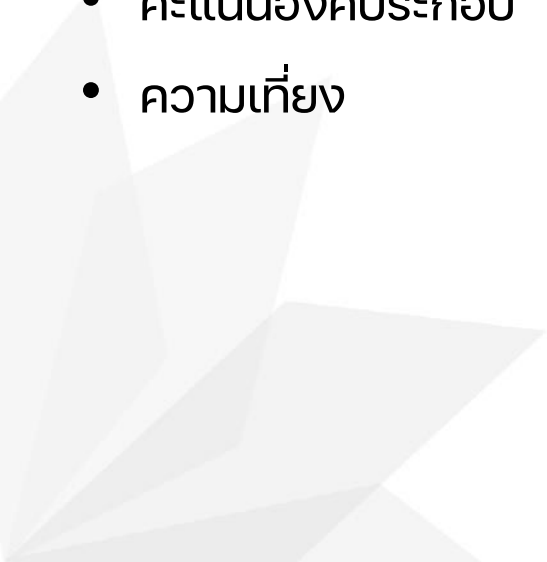


สันศักดิ์ พรประเสริฐมานิต



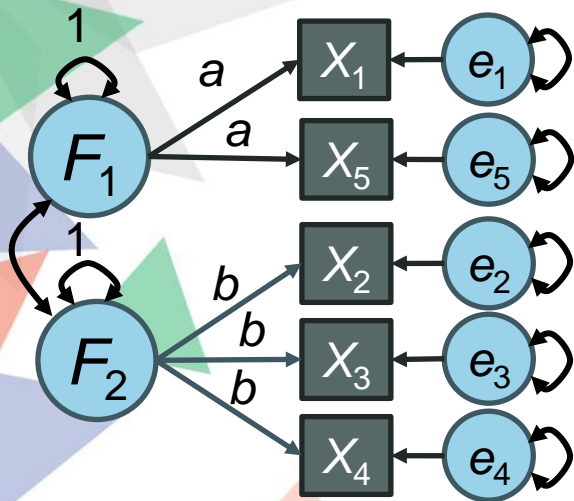
โครงร่างการนำเสนอ

- การกำหนดให้พารามิเตอร์เท่ากัน (Equality Constraints)
- การกำหนดช่วงพารามิเตอร์ที่เป็นไปได้
- พารามิเตอร์เพิ่มเติม
- คะแนนองค์ประกอบ
- ความเที่ยง



การกำหนดให้พารามิเตอร์เท่ากัน (Equality Constraints)

- การกำหนดให้พารามิเตอร์เท่ากัน (Equality Constraints) เป็นการกำหนดให้พารามิเตอร์ 2 ตัว หรือมากกว่า ให้มีผลการประมาณค่ามีค่าเท่ากัน Standard Errors (SE) มีค่าเท่ากัน เพิ่ม df ภายในโมเดลเมื่อเทียบกับการแยกประมาณค่า
- ใน lavaan วิธีการง่ายที่สุดในการกำหนดให้พารามิเตอร์เท่ากัน คือการสร้าง label แบบเดียวกันในพารามิเตอร์ที่อยากให้เท่ากัน
- เช่น มาตรการความไวแต่ละข้อคำถามเท่ากัน จะเรียกว่ามาตรแบบ Tau-equivalent ทำได้โดยการทำให้ค่าน้ำหนักองค์ประกอบเท่ากัน ถ้าโมเดลนี้ถูกต้อง จะทำให้ Coefficient alpha แสดงถึงค่าความเที่ยงแบบสอดคล้องภายใน
- แต่ถ้าไม่ใช่ Alpha จะไม่ได้สะท้อนถึงความเที่ยง
- โมเดลนี้จะบอกว่า ข้อคำถามแต่ละข้อมีความเป็นตัวแทนขององค์ประกอบใกล้เคียงกัน



ใส่ label เดียวกัน หน้าพารามิเตอร์

ที่ต้องการประมาณค่า

```
> mt1 <- '
+ f1 =~ a*x1 + a*x5
+ f2 =~ b*x2 + b*x3 + b*x4
+ '
> outmt1 <- cfa(mt1, data=dat, std.lv=TRUE)
> summary(outmt1)
lavaan 0.6-12 ended normally after 17 iterations
```

Estimator	ML
Optimization method	NLMINB
Number of model parameters	11
Number of equality constraints	3
Number of observations	200
Model Test User Model:	
Test statistic	26.021
Degrees of freedom	7
P-value (Chi-square)	0.000

Latent Variables:

		Estimate	Std.Err	z-value	P(> z)
f1 =~					
x1	(a)	0.309	0.053	5.849	0.000
x5	(a)	0.309	0.053	5.849	0.000
f2 =~					
x2	(b)	0.471	0.034	13.754	0.000
x3	(b)	0.471	0.034	13.754	0.000
x4	(b)	0.471	0.034	13.754	0.000

Covariances:

	Estimate	Std.Err	z-value	P(> z)
f1 ~~				
f2	0.727	0.141	5.142	0.000

Variances:

	Estimate	Std.Err	z-value	P(> z)
.x1	0.378	0.048	7.933	0.000
.x5	0.333	0.044	7.540	0.000
.x2				
.x3				
.x4				
f1				
f2				

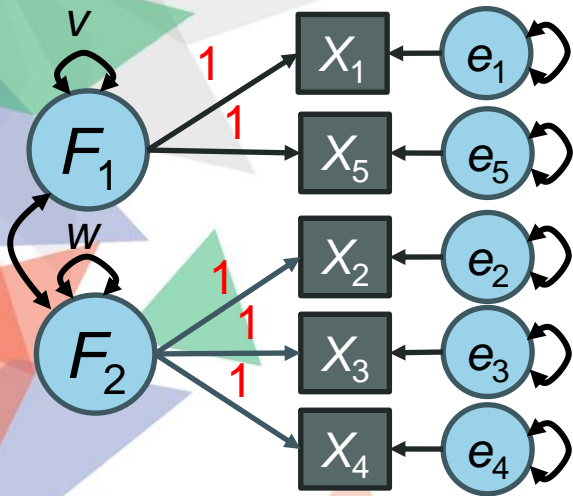
พารามิเตอร์ประมาณค่าน้อยลง 3 ตัว
การลดพารามิเตอร์ทำให้ค่าความเหมาะสม
ไม่เพิ่มขึ้นอย่างมีนัยสำคัญ

```
> anova(out2, outmt1)
```

Chi-Squared Difference Test

	Df	AIC	BIC	Chisq	Chisq diff	Df diff	Pr(>Chisq)
out2	4	1984.6	2020.9	19.550			
outmt1	7	1985.0	2011.4	26.021	6.4707	3	0.09083 .

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1



```

> mt2 <- '
+ f1 =~ 1*x1 + 1*x5
+ f2 =~ 1*x2 + 1*x3 + 1*x4
+ '
> outmt2 <- cfa(mt2, data=dat, std.lv=FALSE)
> summary(outmt2)
lavaan 0.6-12 ended normally after 18 iterations

```

Estimator	ML
Optimization method	NLMINB
Number of model parameters	8
Number of observations	200
Model Test User Model:	
Test statistic	26.021
Degrees of freedom	7
P-value (Chi-square)	0.000

Latent Variables:

	Estimate	Std.Err	z-value	P(> z)
f1 =~				
x1	1.000			
x5	1.000			
f2 =~				
x2	1.000			
x3	1.000			
x4	1.000			

Covariances:

	Estimate	Std.Err	z-value	P(> z)
f1 =~				
f2	0.106	0.022	4.795	0.000

Variances:

	Estimate	Std.Err	z-value	P(> z)
.x1	0.378	0.048	7.933	0.000
.x5	0.333	0.044	7.540	0.000
.x2	0.233	0.032	7.202	0.000
.x3	0.348	0.043	8.161	0.000
.x4	0.281	0.037	7.702	0.000
f1	0.095	0.033	2.925	0.003
f2	0.222	0.032	6.877	0.000

เปลี่ยนเป็น **Marker Variable Approach** ได้ ไม่จำเป็นต้องสร้าง **Equality Constraints** เพราะตัว **Marker Variable** กำหนดน้ำหนักองค์ประกอบเป็น **1** การกำหนดให้เท่ากันก็เป็นเพียงทำให้ค่าของน้ำหนักองค์ประกอบอื่นเท่ากับ **1**

การกำหนดให้พารามิเตอร์เท่ากัน (Equality Constraints)

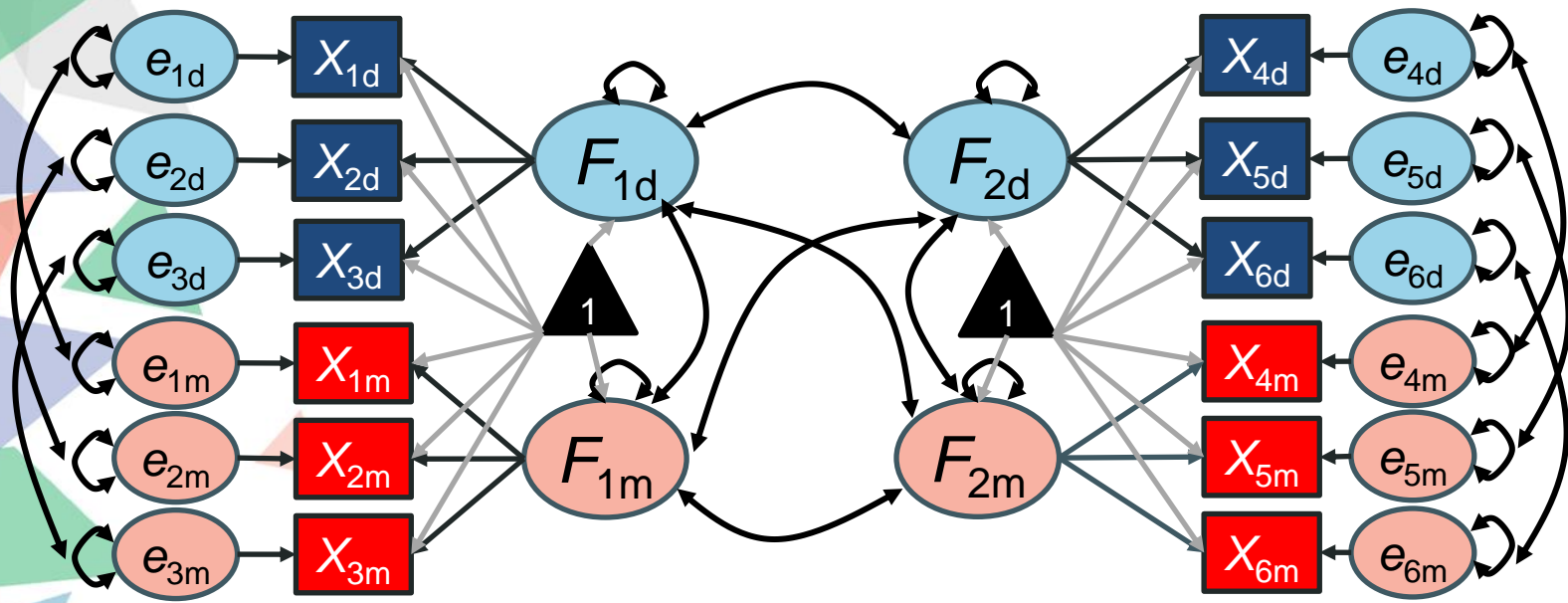
- การกำหนดให้พารามิเตอร์เท่ากันจะใช้บ่อยในการเปรียบเทียบคุณสมบัติของมาตรระหว่างกลุ่ม ระหว่างเวลา ระหว่างแหล่งผู้ประเมิน เพื่อยืนยันว่าคุณสมบัติของมาตรวัดไม่แตกต่างกัน (Measurement Invariance)
- สมมติมีมาตรวัด 6 ข้อที่แม่และพ่อประเมินลูกของตน มีข้อคำถาม 6 ข้อ 3 ข้อแรกประเมินอาการจดจ่อไม่ได้ (Inattentive) และ 3 ข้อหลังประเมินอาการอยู่ไม่สุข (Hyperactive)
- นักวิจัยอาจต้องการดูว่าคุณสมบัติของมาตรที่ประเมินลูกจากพ่อ และที่ประเมินลูกจากแม่ มีคุณสมบัติเหมือนกันหรือไม่ สามารถคะแนนคนใดคนหนึ่งเพื่อแทนค่าทั้งสองคนได้หรือไม่

การกำหนดให้พารามิเตอร์เท่ากัน (Equality Constraints)

- มาตรวัดมีข้อคำถามดังนี้ (นำมาจาก Vanderbilt ADHD Diagnostic Parent Rating Scale)
 1. ไม่ใส่ใจรายละเอียด หรือสร้างความผิดพลาดที่เกิดจากไม่ระวัง เช่น การทำการบ้าน
 2. มีสมาธิกับงานหรือกิจกรรมได้อย่างยากลำบากดูเหมือนไม่ได้ฟังเวลาคนพูดกับเขา
 3. ลืมกิจกรรมที่ต้องทำในชีวิตประจำวันได้ง่าย
 4. มือหรือเท้าอยู่ไม่สุข หรือโยกเก้าอี้
 5. ออกจากที่นั่งในระหว่างที่ควรจะนั่งที่
 6. ตอบคำตอบก่อนที่คำถามจะพูดออกมาครบ
- การตอบแบ่งเป็น 5 ระดับ 0 = ไม่เคย, 1 = บานๆ ที, 2 = เป็นบางครั้ง, 3 = บ่อย, 4 = บ่อยมาก

การกำหนดให้พารามิเตอร์เท่ากัน (Equality Constraints)

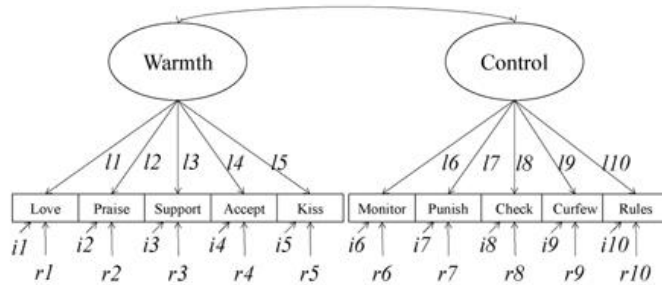
- การยืนยันว่าคุณสมบัติมาตรวัดไม่ต่างกัน (Measurement Invariance) สามารถแบ่งออกเป็น 5 ระดับ
 1. **โครงสร้างไม่ตรงกัน (Configural Noninvariance)** การจัดองค์ประกอบของตัวบ่งชี้ในสองกลุ่มไม่เหมือนกัน
 2. **โครงสร้างตรงกัน (Configural Invariance)** การจัดองค์ประกอบของตัวบ่งชี้ในสองกลุ่มเหมือนกัน แต่ไม่การันตีว่าค่าพารามิเตอร์ใดในสองกลุ่มเท่ากัน
 3. **หน่วยวัดเทียบเคียงกัน (Metric/Weak Invariance)** น้ำหนักองค์ประกอบของทั้งสองกลุ่มเท่ากัน
 4. **คะแนนดิบเท่าเทียมกัน (Scalar/Strong) Invariance)** จุดตัดของตัวบ่งชี้ของทั้งสองกลุ่มเท่ากัน
 5. **ค่าคงเหลือขนาดเท่าเทียมกัน (Residual/Strict Invariance)** ความแปรปรวนของค่าคงเหลือเท่ากัน



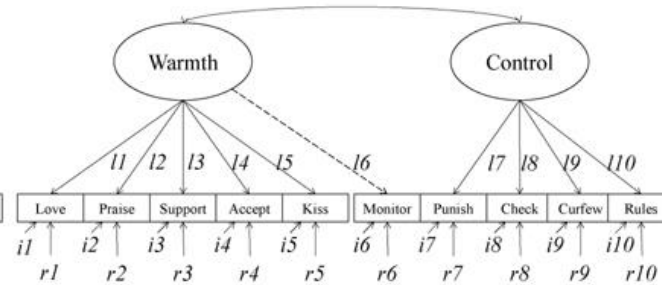
รูปแบบของข้อมูล แถวจะหมายถึงเด็กแต่ละคน จะมี 6 ตัวแปรแสดงการประเมินจากแม่ และ 6 ตัวแปรแสดงการประเมินจากพ่อ

โมเดลนี้จะแตกต่างจาก CFA แบบปกติ คือ จะให้มี residual correlation ระหว่างคะแนนข้อเดียวกันที่มาจากผู้ประเมินต่างกัน เนื่องจากอาจมีความแปรปรวนจำเพาะ (specific variance) ที่ไม่ได้ถูกอธิบายจากองค์ประกอบ แต่ไม่ใช้ความแปรปรวนแบบสุ่ม ปัจจัยนี้อาจเป็นปัจจัยที่เหมือนกันในเด็ก ที่ทำให้พ่อและแม่ประเมินคล้ายกัน เช่น เรื่องลืมนั่ง เด็กบางคนอาจมีพี่เลี้ยงคอยดูแล ทำให้ลืมนั่งต่างๆ ไม่เยอะ ขณะที่เด็กบางคนมีเรื่องที่ต้องรับผิดชอบเยอะ ทำให้เกิดเหตุการณ์ลืมนั่ง ไม่เกี่ยวกับอาการจดจ่อไม่ได้ของเด็ก

A. Configural Invariance



B. Configural Noninvariance



การทดสอบโครงสร้าง จะทดสอบว่าการจับองค์ประกอบของตัวบ่งชี้ทั้งสองกลุ่มตรงกันหรือไม่

ผ่าน โครงสร้างตรงกันระหว่างกลุ่ม
มีแนวโน้มที่ภาวะสันนิษฐานที่
ตัวบ่งชี้วัดจะเป็นตัวเดียวกัน

ไม่ผ่าน โครงสร้างไม่ตรงกัน
ความหมายของภาวะสันนิษฐานไม่
ตรงกันแน่นอน

ไม่ผ่านทำอย่างไร

1. ตัดข้อคำถามบางข้อที่มีปัญหา แล้วทดสอบใหม่ แต่หากทำโดยไม่มี
ทฤษฎีรองรับ จะทำให้ความน่าเชื่อถือของงานวิจัยน้อยลง
2. ยกเลิกการทดสอบ และสรุปว่าความหมายของภาวะสันนิษฐานของทั้ง
สองกลุ่มไม่ตรงกัน

```

> madhd1 <- '
+ f1m =~ x1m + x2m + x3m
+ f2m =~ x4m + x5m + x6m
+ f1d =~ x1d + x2d + x3d
+ f2d =~ x4d + x5d + x6d
+ x1m =~ x1d
+ x2m =~ x2d
+ x3m =~ x3d
+ x4m =~ x4d
+ x5m =~ x5d
+ x6m =~ x6d
+ '
> outadhd1 <- cfa(madhd1, data=datadhd, meanstructure=TRUE)

```

f1 = Inattentive, f2 = Hyperactivity
m = Mom, d = Dad

ข้อคำถามเดียวกัน จากผู้ประเมินที่ต่างกัน ให้ค่าคงเหลือสัมพันธ์กัน

```

> summary(outadhd1, fit=TRUE, std=TRUE)
lavaan 0.6.16 ended normally after 68 iterations

```

Estimator	ML
Optimization method	NLMINB
Number of model parameters	48
Number of observations	670

Model Test User Model:

Test statistic	42.194
Degrees of freedom	42
P-value (Chi-square)	0.463

User Model versus Baseline Model:

Comparative Fit Index (CFI)	1.000
Tucker-Lewis Index (TLI)	1.000

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-7737.303
Loglikelihood unrestricted model (H1)	-7716.206

Akaike (AIC)	15570.606
Bayesian (BIC)	15786.955
Sample-size adjusted Bayesian (SABIC)	15634.552

Root Mean Square Error of Approximation:

RMSEA	0.003
90 Percent confidence interval - lower	0.000
90 Percent confidence interval - upper	0.026
P-value H_0: RMSEA <= 0.050	1.000
P-value H_0: RMSEA >= 0.080	0.000

Standardized Root Mean Square Residual:

SRMR	0.024
------	-------

$\chi^2(42) = 42.19, p = .46, RMSEA = .003$
 $CFI = 1.00, TLI = 1.00, SRMR = .024$

ค่าความเหมาะสมดี ผ่าน **Configural Invariance**

Latent Variables:

	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
f1m ~						
x1m	1.000				0.382	0.587
x2m	0.886	0.117	7.569	0.000	0.339	0.544
x3m	0.826	0.114	7.267	0.000	0.316	0.476
f2m ~						
x4m	1.000				0.331	0.494
x5m	0.988	0.131	7.521	0.000	0.327	0.521
x6m	1.229	0.156	7.884	0.000	0.407	0.609
f1d ~						
x1d	1.000				0.465	0.657
x2d	0.783	0.102	7.687	0.000	0.364	0.532
x3d	0.661	0.093	7.103	0.000	0.308	0.433
f2d ~						
x4d	1.000				0.280	0.423
x5d	1.406	0.210	6.700	0.000	0.394	0.548
x6d	1.437	0.213	6.745	0.000	0.403	0.558

Covariances:

	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
.x1m ~						
.x1d	0.050	0.017	2.998	0.003	0.050	0.178
.x2m ~						
.x2d	0.062	0.015	4.101	0.000	0.062	0.204
.x3m ~						
.x3d	0.066	0.017	3.929	0.000	0.066	0.176
.x4m ~						
.x4d	0.037	0.016	2.329	0.020	0.037	0.107
.x5m ~						
.x5d	0.056	0.017	3.328	0.001	0.056	0.173
.x6m ~						
.x6d	0.089	0.019	4.722	0.000	0.089	0.279
f1m ~						
f2m	0.066	0.012	5.674	0.000	0.520	0.520
f1d	0.072	0.015	4.737	0.000	0.406	0.406
f2d	0.021	0.008	2.553	0.011	0.194	0.194
f2m ~						
f1d	0.006	0.010	0.576	0.564	0.039	0.039
f2d	0.060	0.011	5.353	0.000	0.642	0.642
f1d ~						
f2d	0.064	0.012	5.287	0.000	0.494	0.494

Intercepts:

	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
.x1m	2.924	0.025	116.200	0.000	2.924	4.489
.x2m	2.791	0.024	116.008	0.000	2.791	4.482
.x3m	2.694	0.026	104.997	0.000	2.694	4.056
.x4m	2.964	0.026	114.434	0.000	2.964	4.421
.x5m	3.170	0.024	130.621	0.000	3.170	5.046
.x6m	3.290	0.026	127.232	0.000	3.290	4.915
.x1d	2.700	0.027	98.631	0.000	2.700	3.810
.x2d	2.607	0.026	98.601	0.000	2.607	3.809
.x3d	2.534	0.027	92.239	0.000	2.534	3.564
.x4d	2.843	0.026	110.895	0.000	2.843	4.284
.x5d	3.001	0.028	107.925	0.000	3.001	4.169
.x6d	3.106	0.028	111.383	0.000	3.106	4.303
f1m	0.000				0.000	0.000
f2m	0.000				0.000	0.000
f1d	0.000				0.000	0.000
f2d	0.000				0.000	0.000

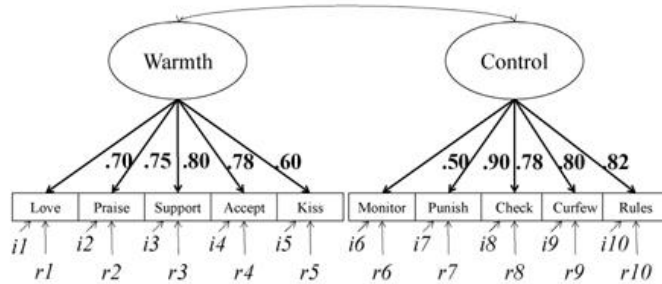
Variances:

	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
.x1m	0.278	0.024	11.401	0.000	0.278	0.655
.x2m	0.273	0.021	12.827	0.000	0.273	0.704
.x3m	0.341	0.023	14.566	0.000	0.341	0.774
.x4m	0.340	0.023	14.528	0.000	0.340	0.756
.x5m	0.287	0.021	13.830	0.000	0.287	0.728
.x6m	0.282	0.025	11.480	0.000	0.282	0.630
.x1d	0.286	0.031	9.214	0.000	0.286	0.569
.x2d	0.336	0.025	13.299	0.000	0.336	0.716
.x3d	0.411	0.026	15.541	0.000	0.411	0.813
.x4d	0.362	0.023	15.464	0.000	0.362	0.821
.x5d	0.363	0.029	12.643	0.000	0.363	0.700
.x6d	0.359	0.029	12.360	0.000	0.359	0.688
f1m	0.146	0.026	5.696	0.000	1.000	1.000
f2m	0.110	0.021	5.131	0.000	1.000	1.000
f1d	0.217	0.035	6.245	0.000	1.000	1.000
f2d	0.079	0.019	4.249	0.000	1.000	1.000

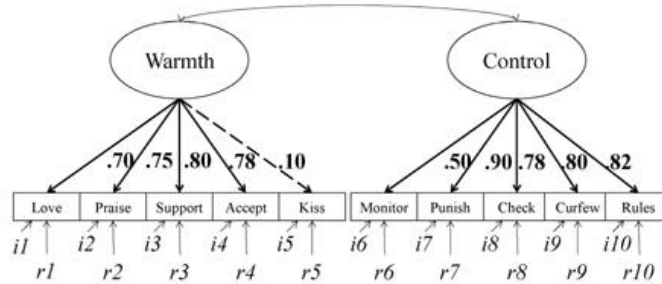
อย่าลืมว่าค่าเหล่านี้ไม่ใช่ค่าสหสัมพันธ์ของค่าคงเหลือ

ไม่ได้ระบุให้ค่าใดเท่ากันเลย

C. Metric Invariance



D. Metric Noninvariance



การทดสอบหน่วยวัด จะทดสอบว่าน้ำหนักองค์ประกอบของตัวบ่งชี้ทั้งสองกลุ่มตรงกันหรือไม่

ผ่าน น้ำหนักองค์ประกอบเท่าเทียมกัน ทำให้ระยะห่างระหว่างคะแนนภาวะสันนิษฐานสองจุด (เช่น ระหว่าง 0 และ 1 ในอาการอยู่ไม่สุขที่แม่ประเมิน, ระหว่าง 0.25 และ 1.25 ในอาการอยู่ไม่สุขที่พ่อประเมิน) เท่ากัน เมื่อระยะห่างเท่ากัน ทำให้สามารถเปรียบเทียบส่วนเบี่ยงเบนมาตรฐานระหว่างกลุ่มได้ เปรียบเทียบความแปรปรวนร่วมระหว่างกลุ่มได้ เปรียบเทียบสหสัมพันธ์ของตัวแปรระหว่างกลุ่มได้ และเปรียบเทียบสัมประสิทธิ์ถดถอยระหว่างกลุ่มได้

ไม่ผ่าน ระยะห่างระหว่างคะแนนมีความหมายไม่เหมือนกัน เปรียบเทียบ SD , Cov , Cor , b ระหว่างกลุ่มไม่ได้

- ไม่ผ่านทำอย่างไร
1. ทำให้ตัวบ่งชี้บางตัวมีน้ำหนักองค์ประกอบไม่เท่ากัน (Partial Invariance)
 2. ตัดตัวบ่งชี้ที่มีปัญหาออกไปเลย
 3. หยุดที่ Configural Invariance

กำหนดน้ำหนักองค์ประกอบเท่ากัน

```
> madhd2 <- '
+ f1m ~ x1m + c2*x2m + c3*x3m
+ f2m ~ x4m + c5*x5m + c6*x6m
+ f1d ~ x1d + c2*x2d + c3*x3d
+ f2d ~ x4d + c5*x5d + c6*x6d
+ x1m ~ x1d
+ x2m ~ x2d
+ x3m ~ x3d
+ x4m ~ x4d
+ x5m ~ x5d
+ x6m ~ x6d
+ '
> outadhd2 <- cfa(madhd2, data=datadhd, meanstructure=0)
```

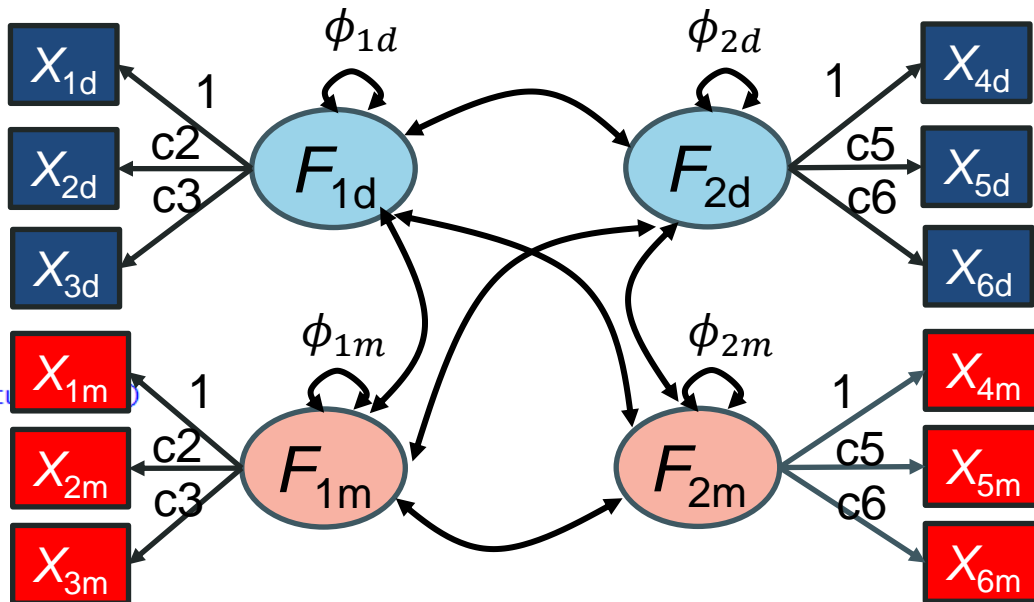
```
> summary(outadhd2, fit=TRUE, std=TRUE)
lavaan 0.6.16 ended normally after 55 iterations
```

Estimator	ML
Optimization method	NLMINB
Number of model parameters	48
Number of equality constraints	4
Number of observations	670

Model Test User Model:

Test statistic	47.287
Degrees of freedom	46
P-value (Chi-square)	0.420

ML
NLMINB



```
> anova(outadhd1, outadhd2)
```

Chi-Squared Difference Test

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff	Pr(>Chisq)
outadhd1	42	15571	15787	42.194				
outadhd2	46	15568	15766	47.287	5.093	0.020195	4	0.2779

Configural และ Metric Invariance Models
ไม่แตกต่างกันอย่างมีนัยสำคัญ เลือก Metric Invariance

Latent Variables:

	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
f1m =~						
x1m	1.000				0.401	0.613
x2m	(c2) 0.830	0.082	10.097	0.000	0.333	0.535
x3m	(c3) 0.737	0.076	9.636	0.000	0.296	0.448
f2m =~						
x4m	1.000				0.304	0.458
x5m	(c5) 1.152	0.123	9.398	0.000	0.350	0.552
x6m	(c6) 1.327	0.140	9.484	0.000	0.404	0.604
f1d =~						
x1d	1.000				0.445	0.632
x2d	(c2) 0.830	0.082	10.097	0.000	0.370	0.539
x3d	(c3) 0.737	0.076	9.636	0.000	0.328	0.459
f2d =~						
x4d	1.000				0.310	0.463
x5d	(c5) 1.152	0.123	9.398	0.000	0.357	0.502
x6d	(c6) 1.327	0.140	9.484	0.000	0.411	0.569

Covariances:

	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
.x1m ~						
.x1d	0.050	0.017	2.979	0.003	0.050	0.177
.x2m ~						
.x2d	0.062	0.015	4.103	0.000	0.062	0.204
.x3m ~						
.x3d	0.066	0.017	3.919	0.000	0.066	0.175
.x4m ~						
.x4d	0.037	0.016	2.315	0.021	0.037	0.106
.x5m ~						
.x5d	0.056	0.017	3.364	0.001	0.056	0.172
.x6m ~						
.x6d	0.088	0.019	4.660	0.000	0.088	0.277
f1m ~						
f2m	0.064	0.010	6.159	0.000	0.523	0.523
f1d	0.073	0.015	4.764	0.000	0.411	0.411
f2d	0.025	0.009	2.704	0.007	0.200	0.200
f2m ~						
f1d	0.005	0.009	0.565	0.572	0.038	0.038
f2d	0.060	0.011	5.394	0.000	0.638	0.638
f1d ~						
f2d	0.069	0.012	5.911	0.000	0.496	0.496

Intercepts:

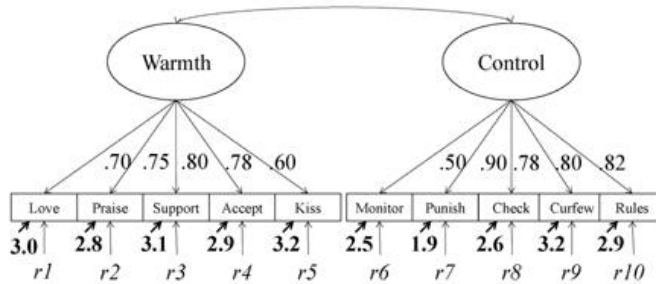
	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
.x1m	2.924	0.025	115.512	0.000	2.924	4.463
.x2m	2.791	0.024	116.095	0.000	2.791	4.485
.x3m	2.694	0.026	105.643	0.000	2.694	4.081
.x4m	2.964	0.026	115.528	0.000	2.964	4.463
.x5m	3.170	0.025	129.358	0.000	3.170	4.998
.x6m	3.290	0.026	127.430	0.000	3.290	4.923
.x1d	2.700	0.027	99.164	0.000	2.700	3.831
.x2d	2.607	0.026	98.508	0.000	2.607	3.806
.x3d	2.534	0.028	91.669	0.000	2.534	3.541
.x4d	2.843	0.026	109.821	0.000	2.843	4.243
.x5d	3.001	0.028	109.132	0.000	3.001	4.216
.x6d	3.106	0.028	111.210	0.000	3.106	4.296
f1m	0.000				0.000	0.000
f2m	0.000				0.000	0.000
f1d	0.000				0.000	0.000
f2d	0.000				0.000	0.000

Variances:

	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
.x1m	0.268	0.023	11.439	0.000	0.268	0.625
.x2m	0.276	0.020	13.806	0.000	0.276	0.714
.x3m	0.348	0.022	15.648	0.000	0.348	0.799
.x4m	0.349	0.022	15.574	0.000	0.349	0.790
.x5m	0.280	0.021	13.513	0.000	0.280	0.695
.x6m	0.284	0.024	11.997	0.000	0.284	0.635
.x1d	0.298	0.027	10.944	0.000	0.298	0.601
.x2d	0.333	0.024	13.813	0.000	0.333	0.709
.x3d	0.404	0.026	15.542	0.000	0.404	0.790
.x4d	0.353	0.023	15.246	0.000	0.353	0.786
.x5d	0.379	0.026	14.405	0.000	0.379	0.748
.x6d	0.353	0.028	12.605	0.000	0.353	0.676
f1m	0.161	0.023	6.906	0.000	1.000	1.000
f2m	0.092	0.016	5.753	0.000	1.000	1.000
f1d	0.198	0.028	7.065	0.000	1.000	1.000
f2d	0.096	0.017	5.750	0.000	1.000	1.000

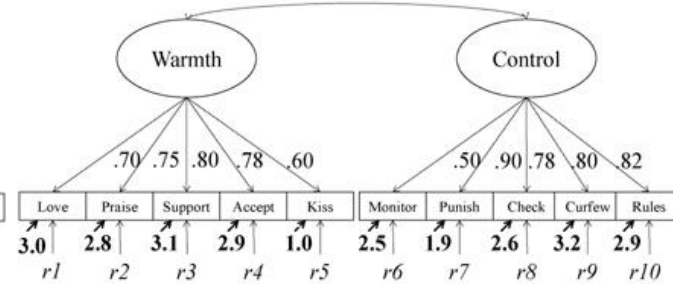
กำหนดให้น้ำหนักองค์ประกอบเท่ากัน และประมาณค่าความแปรปรวน
ทุกองค์ประกอบ จะเห็นว่าความแปรปรวนจากพ่อสูงกว่าแม่เล็กน้อย

E. Scalar Invariance



F. Scalar Noninvariance

รูปจาก Putnick & Bornstein, 2016



การทดสอบหน่วยวัด จะทดสอบว่าจุดตัดของตัวบ่งชี้ทั้งสองกลุ่มตรงกันหรือไม่

ผ่าน จุดตัดเท่าเทียมกัน หมายความว่า คะแนนของภาวะสันนิษฐานระดับเดียวกันทำให้เกิดคะแนนในแต่ละตัวบ่งชี้ในระดับเดียวกันด้วยในทั้งสองกลุ่มด้วย เช่น จากรูปข้างบน ถ้าคะแนน **Warmth = 0** ในทั้งสองกลุ่ม คะแนนของ **Love** จะเท่ากับ **3.0** โดยเฉลี่ยทั้งสองกลุ่ม ทำให้หากพบว่ากลุ่มหนึ่ง มีค่าภาวะสันนิษฐานสูงกว่าอีกกลุ่มหนึ่ง น่าจะเกิดจากความแตกต่างที่แท้จริง คะแนนภาวะสันนิษฐานสามารถเปรียบเทียบระหว่างกลุ่มได้ กล่าวคือ คะแนน α สามารถนำมาเปรียบเทียบระหว่างกลุ่ม เพื่อบอกว่ากลุ่มไหนมากกว่าได้

ไม่ผ่าน คะแนนของภาวะสันนิษฐานระดับเดียวกัน ไม่ได้สะท้อนค่าของภาวะสันนิษฐานในระดับเดียวกัน นำค่าเฉลี่ยของภาวะสันนิษฐาน (รวมถึง ค่าเฉลี่ยระหว่างข้อคำถามทั้งหมด) มาเปรียบเทียบว่าใครมากกว่าไม่ได้

ไม่ผ่านทำอะไร

1. ทำให้ตัวบ่งชี้บางตัวมีจุดตัดไม่เท่ากัน (Partial Invariance)
2. ตัดตัวบ่งชี้ที่มีปัญหาออกไปเลย
3. หยุดที่ Metric Invariance


```

> madhd3 <- '
+ f1m =~ x1m + c2*x2m + c3*x3m
+ f2m =~ x4m + c5*x5m + c6*x6m
+ f1d =~ x1d + c2*x2d + c3*x3d
+ f2d =~ x4d + c5*x5d + c6*x6d
+ x1m =~ x1d
+ x2m =~ x2d
+ x3m =~ x3d
+ x4m =~ x4d
+ x5m =~ x5d
+ x6m =~ x6d
+ f1m ~ NA*1
+ f2m ~ NA*1
+ f1d ~ NA*1
+ f2d ~ NA*1
+ x1m ~ 0*1
+ x2m ~ i2*1
+ x3m ~ i3*1
+ x4m ~ 0*1
+ x5m ~ i5*1
+ x6m ~ i6*1
+ x1d ~ 0*1
+ x2d ~ i2*1
+ x3d ~ i3*1
+ x4d ~ 0*1
+ x5d ~ i5*1
+ x6d ~ i6*1
+ '

```

กำหนดจุดตัดของตัวบ่งชี้เท่ากัน

```

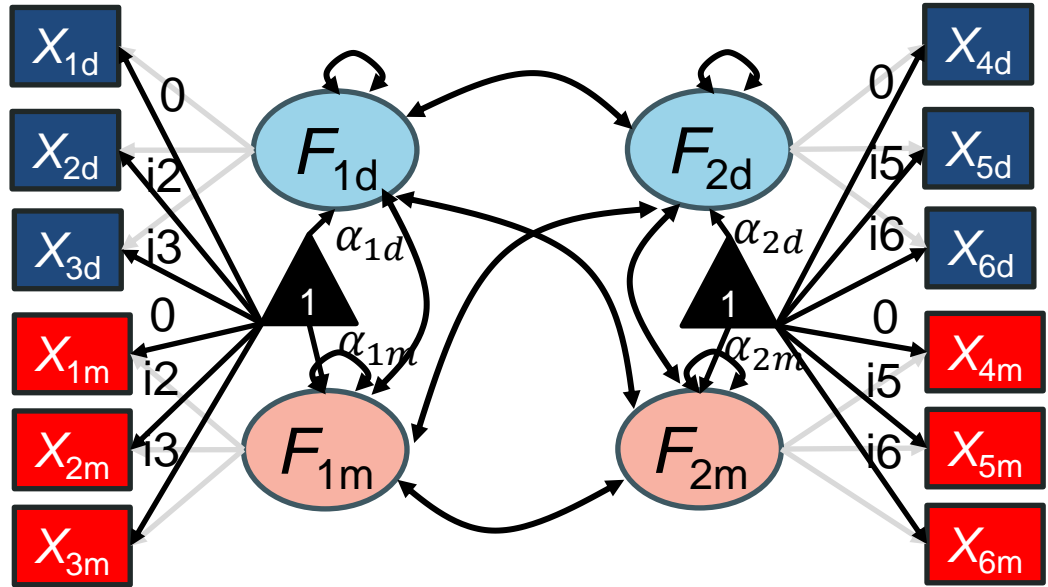
> outadhd3 <- cfa(madhd3, data=datadhd, meanstructure=TRUE)
> summary(outadhd3, std=TRUE)
lavaan 0.6.16 ended normally after 96 iterations

```

Estimator	ML
Optimization method	NLMINB
Number of model parameters	48
Number of equality constraints	8
Number of observations	670

Model Test User Model:

Test statistic	47.675
Degrees of freedom	50
P-value (Chi-square)	0.567



```
> anova(outadhd3, outadhd2)
```

Chi-Squared Difference Test

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff	Pr(>Chisq)
outadhd2	46	15568	15766	47.287				
outadhd3	50	15560	15740	47.675	0.3876	0	4	0.9835

Metric และ Scalar Invariance Models

ไม่แตกต่างกันอย่างมีนัยสำคัญ เลือก Scalar Invariance

Latent Variables:

	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
f1m ~						
x1m	1.000				0.402	0.614
x2m (c2)	0.827	0.073	11.408	0.000	0.333	0.535
x3m (c3)	0.732	0.069	10.659	0.000	0.295	0.446
f2m ~						
x4m	1.000				0.300	0.452
x5m (c5)	1.180	0.118	9.991	0.000	0.353	0.557
x6m (c6)	1.347	0.132	10.173	0.000	0.404	0.604
f1d ~						
x1d	1.000				0.446	0.633
x2d (c2)	0.827	0.073	11.408	0.000	0.369	0.539
x3d (c3)	0.732	0.069	10.659	0.000	0.327	0.457
f2d ~						
x4d	1.000				0.306	0.457
x5d (c5)	1.180	0.118	9.991	0.000	0.361	0.507
x6d (c6)	1.347	0.132	10.173	0.000	0.412	0.570

Covariances:

	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
.x1m ~ .x1d	0.050	0.017	2.997	0.003	0.050	0.176
.x2m ~ .x2d	0.062	0.015	4.138	0.000	0.062	0.204
.x3m ~ .x3d	0.066	0.017	3.941	0.000	0.066	0.176
.x4m ~ .x4d	0.038	0.016	2.379	0.017	0.038	0.108
.x5m ~ .x5d	0.055	0.017	3.328	0.001	0.055	0.170
.x6m ~ .x6d	0.087	0.019	4.717	0.000	0.087	0.276
f1m ~ f2m	0.063	0.010	6.275	0.000	0.523	0.523
f1m ~ f1d	0.074	0.015	4.886	0.000	0.410	0.410
f2m ~ f2d	0.025	0.009	2.715	0.007	0.200	0.200
f2m ~ f1d	0.005	0.009	0.564	0.573	0.038	0.038
f2d ~ f2d	0.059	0.011	5.529	0.000	0.639	0.639
f1d ~ f2d	0.068	0.011	6.008	0.000	0.496	0.496

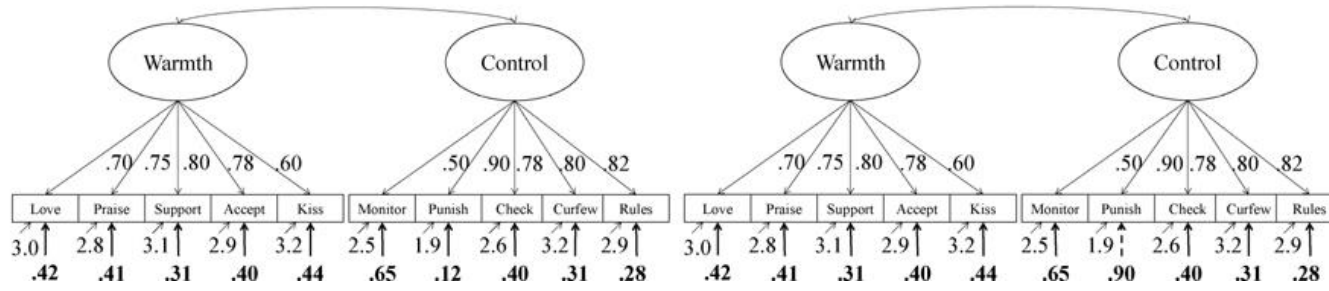
Intercepts:

	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
f1m	2.923	0.024	121.393	0.000	7.268	7.268
f2m	2.971	0.022	133.768	0.000	9.919	9.919
f1d	2.701	0.026	104.928	0.000	6.050	6.050
f2d	2.836	0.022	126.701	0.000	9.270	9.270
.x1m	0.000				0.000	0.000
.x2m (i2)	0.373	0.206	1.811	0.070	0.373	0.599
.x3m (i3)	0.554	0.195	2.843	0.004	0.554	0.840
.x4m	0.000				0.000	0.000
.x5m (i5)	-0.339	0.345	-0.985	0.325	-0.339	-0.534
.x6m (i6)	-0.714	0.386	-1.850	0.064	-0.714	-1.069
.x1d	0.000				0.000	0.000
.x2d (i2)	0.373	0.206	1.811	0.070	0.373	0.544
.x3d (i3)	0.554	0.195	2.843	0.004	0.554	0.775
.x4d	0.000				0.000	0.000
.x5d (i5)	-0.339	0.345	-0.985	0.325	-0.339	-0.476
.x6d (i6)	-0.714	0.386	-1.850	0.064	-0.714	-0.988

Variances:

	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
.x1m	0.268	0.023	11.717	0.000	0.268	0.623
.x2m	0.277	0.020	14.048	0.000	0.277	0.714
.x3m	0.349	0.022	15.782	0.000	0.349	0.801
.x4m	0.350	0.022	15.763	0.000	0.350	0.796
.x5m	0.278	0.020	13.582	0.000	0.278	0.690
.x6m	0.284	0.023	12.224	0.000	0.284	0.635
.x1d	0.298	0.026	11.238	0.000	0.298	0.599
.x2d	0.333	0.024	14.064	0.000	0.333	0.709
.x3d	0.405	0.026	15.686	0.000	0.405	0.791
.x4d	0.354	0.023	15.443	0.000	0.354	0.791
.x5d	0.377	0.026	14.437	0.000	0.377	0.743
.x6d	0.353	0.028	12.782	0.000	0.353	0.675
f1m	0.162	0.022	7.304	0.000	1.000	1.000
f2m	0.090	0.015	5.938	0.000	1.000	1.000
f1d	0.199	0.027	7.482	0.000	1.000	1.000
f2d	0.094	0.016	5.917	0.000	1.000	1.000

กำหนดให้จุดตัดตัวบ่งชี้เท่ากัน และประมาณค่าเฉลี่ยของ
ทุกองค์ประกอบ จะเห็นว่าค่าเฉลี่ยจากพ่อน้อยกว่าแม่เล็กน้อย



การทดสอบค่าคงเหลือ จะทดสอบว่าความแปรปรวนค่าคงเหลือของตัวบ่งชี้ทั้งสองกลุ่มตรงกันหรือไม่

ผ่าน ความแปรปรวนของค่าคงเหลือทั้งสองกลุ่มเท่ากัน
หมายความว่า ความแปรปรวนของค่าคงเหลือเท่ากัน
เท่านั้นแหละ สัดส่วนร่วม (Communality) ก็
อาจจะต่างกัน เพราะความแปรปรวนขององค์ประกอบ
ต่างกัน

ไม่ผ่าน ความแปรปรวนค่าคงเหลือไม่เท่ากัน แต่ยังคง
เปรียบเทียบค่า SD , b , cov , cor , m ของภาวะ
สันนิษฐานระหว่างกลุ่มได้

ไม่ผ่านทำอย่างไร

1. ทำให้ตัวบ่งชี้บางตัวมีความแปรปรวนค่าคงเหลือไม่เท่ากัน (Partial Invariance)
2. ตัดตัวบ่งชี้ที่มีปัญหาออกไปเลย
3. หยุดที่ Scalar Invariance

เนื่องจากภาวะสันนิษฐานมีคุณลักษณะเหมือนกัน ตั้งแต่ Scalar invariance แล้ว ดังนั้นนักวิจัยมักไม่สนใจทำ Strict invariance

```

> madhd4 <- '
+ f1m ~ x1m + c2*x2m + c3*x3m
+ f2m ~ x4m + c5*x5m + c6*x6m
+ f1d ~ x1d + c2*x2d + c3*x3d
+ f2d ~ x4d + c5*x5d + c6*x6d
+ x1m ~ x1d
+ x2m ~ x2d
+ x3m ~ x3d
+ x4m ~ x4d
+ x5m ~ x5d
+ x6m ~ x6d
+ f1m ~ NA*1
+ f2m ~ NA*1
+ f1d ~ NA*1
+ f2d ~ NA*1
+ x1m ~ 0*1
+ x2m ~ i2*1
+ x3m ~ i3*1
+ x4m ~ 0*1
+ x5m ~ i5*1
+ x6m ~ i6*1
+ x1d ~ 0*1
+ x2d ~ i2*1
+ x3d ~ i3*1
+ x4d ~ 0*1
+ x5d ~ i5*1
+ x6d ~ i6*1

```

```

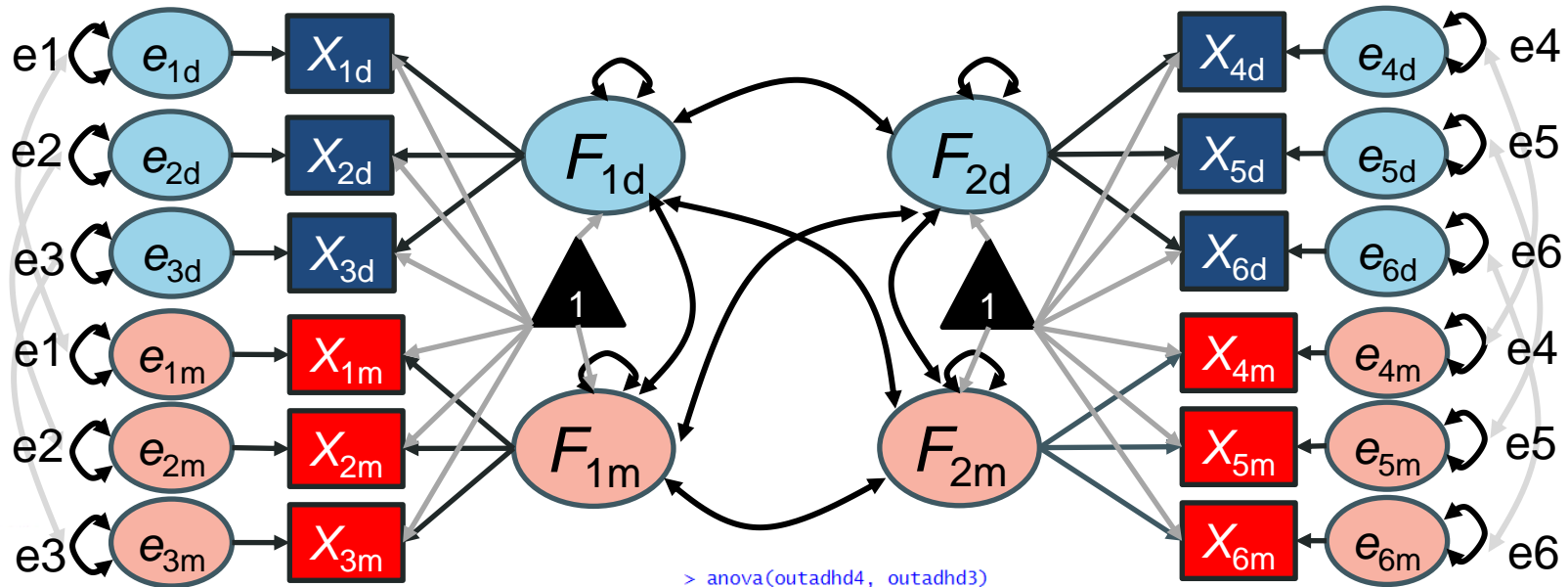
+ x1m ~ e1*x1m
+ x2m ~ e2*x2m
+ x3m ~ e3*x3m
+ x4m ~ e4*x4m
+ x5m ~ e5*x5m
+ x6m ~ e6*x6m
+ x1d ~ e1*x1d
+ x2d ~ e2*x2d
+ x3d ~ e3*x3d
+ x4d ~ e4*x4d
+ x5d ~ e5*x5d
+ x6d ~ e6*x6d
+ '

```

```

> outadhd4 <- cfa(madhd4, data=datadhd, meanstructure=TRUE)

```



กำหนดความแปรปรวนของ
ค่าคงเหลือเท่ากัน

```

> anova(outadhd4, outadhd3)

```

Chi-Squared Difference Test

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff	Pr(>Chisq)
outadhd3	50	15560	15740	47.675				
outadhd4	56	15572	15725	71.332	23.657	0.066275	6	0.0006037 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Scalar และ Strict Invariance Models
แตกต่างกันอย่างมีนัยสำคัญ เลือก **Scalar Invariance**

Variances :

	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
.x1m	(e1) 0.281	0.019	14.468	0.000	0.281	0.647
.x2m	(e2) 0.305	0.017	18.417	0.000	0.305	0.745
.x3m	(e3) 0.379	0.018	21.314	0.000	0.379	0.826
.x4m	(e4) 0.355	0.017	21.300	0.000	0.355	0.818
.x5m	(e5) 0.325	0.018	18.102	0.000	0.325	0.735
.x6m	(e6) 0.316	0.020	15.649	0.000	0.316	0.678
.x1d	(e1) 0.281	0.019	14.468	0.000	0.281	0.571
.x2d	(e2) 0.305	0.017	18.417	0.000	0.305	0.680
.x3d	(e3) 0.379	0.018	21.314	0.000	0.379	0.775
.x4d	(e4) 0.355	0.017	21.300	0.000	0.355	0.783
.x5d	(e5) 0.325	0.018	18.102	0.000	0.325	0.690
.x6d	(e6) 0.316	0.020	15.649	0.000	0.316	0.628
f1m	0.153	0.021	7.156	0.000	1.000	1.000
f2m	0.079	0.014	5.828	0.000	1.000	1.000
f1d	0.211	0.027	7.922	0.000	1.000	1.000
f2d	0.098	0.016	6.030	0.000	1.000	1.000

แสดงให้เห็นว่าความแปรปรวนของค่าคงเหลือถูกจำกัดให้เท่ากันระหว่างการประเมินจากแม่และพ่อ อย่างไรก็ตาม โมเดลนี้ไม่ได้ใช้อยู่แล้ว เพราะปฏิเสธ **Strict Invariance Model**

การกำหนดให้พารามิเตอร์เท่ากัน (Equality Constraints)

- Strict Invariance เป็นการบอกว่า ปริมาณค่าคงเหลือที่มามีผลกระทบต่อมาตรเท่ากันระหว่างผู้ประเมิน
- แต่ไม่ได้การันตีว่าสัดส่วนร่วมของแต่ละตัวบ่งชี้ระหว่างผู้ประเมินเท่ากัน เพราะความแปรปรวนขององค์ประกอบไม่เท่ากันระหว่างผู้ประเมิน
- กล่าวคือ ความเที่ยงของมาตร ก็จะไม่เท่ากันระหว่างผู้ประเมินด้วย
- แต่อย่างไรก็ตาม Strict Invariance ย่อมแสดงถึงความเท่าเทียมของมาตรระหว่างผู้ประเมินได้ดีกว่า Scalar Invariance
- แนะนำให้ทดสอบไว้ เพื่อแสดงถึงระดับความเท่าเทียมกันของการวัดระหว่างสองกลุ่ม

การกำหนดให้พารามิเตอร์เท่ากัน (Equality Constraints)

- ในที่นี้ ใช้ Marker Variable Approach ในการระบุโมเดล สามารถใช้ Fixed Factor Approach ในการระบุโมเดลได้เช่นกัน แต่รายการละเอียดการสร้างโมเดลจะซับซ้อนเล็กน้อย
- การให้ตัวบ่งชี้บางตัวมีค่าเท่ากัน แต่บางตัวมีค่าไม่เท่ากัน จะเรียกว่าการระบุให้เท่ากันบางส่วน (Partial Invariance) มี 2 วิธี
 - เริ่มจากให้ต่างกันทุกตัว แล้วระบุให้เท่ากันทีละตัว
 - เริ่มจากทำให้เท่ากันทุกตัว แล้วระบุให้ต่างกันทีละตัว
- รายละเอียดทั้งหมดจะไปกล่าวในเรื่อง Measurement Invariance across Groups อีกครั้งหนึ่ง

การกำหนดให้พารามิเตอร์เท่ากัน (Equality Constraints)

- การกำหนดให้พารามิเตอร์เท่ากันอีกรูปแบบหนึ่ง คือ การกำหนดด้วยสมการ แทนที่จะกำหนดด้วยพารามิเตอร์ เช่น

```
> mt1a <- '  
+ f1 =~ a1*x1 + a2*x5  
+ f2 =~ b1*x2 + b2*x3 + b3*x4  
+ a1 == a2  
+ b1 == b2  
+ b1 == b3  
'  
> outmt1a <- cfa(mt1a, data=dat, std.lv=TRUE)  
> summary(outmt1a)  
lavaan 0.6.16 ended normally after 17 iterations
```

Estimator	ML
Optimization method	NLMINB
Number of model parameters	11
Number of equality constraints	3
Number of observations	200
Model Test User Model:	
Test statistic	26.021
Degrees of freedom	7
P-value (Chi-square)	0.000

Latent Variables:					
		Estimate	Std.Err	z-value	P(> z)
f1 =~					
x1	(a1)	0.309	0.053	5.849	0.000
x5	(a2)	0.309	0.053	5.849	0.000
f2 =~					
x2	(b1)	0.471	0.034	13.754	0.000
x3	(b2)	0.471	0.034	13.754	0.000
x4	(b3)	0.471	0.034	13.754	0.000
Covariances:					
		Estimate	Std.Err	z-value	P(> z)
f1 ~					
f2		0.727	0.141	5.142	0.000
Variances:					
		Estimate	Std.Err	z-value	P(> z)
.x1		0.378	0.048	7.933	0.000
.x5		0.333	0.044	7.540	0.000
.x2		0.233	0.032	7.202	0.000
.x3		0.348	0.043	8.161	0.000
.x4		0.281	0.037	7.702	0.000
f1		1.000			
f2		1.000			
Constraints:					
					Slack
a1	- (a2)				0.000
b1	- (b2)				0.000
b1	- (b3)				0.000

การกำหนดให้พารามิเตอร์เท่ากัน (Equality Constraints)

- ตัวอย่างการทำ Equality Constraints รูปแบบนี้ คือ การระบุโมเดลแบบ Effect Coding เป็นวิธีที่พัฒนาจาก Marker Variable Method แต่ไม่ได้อิงกับตัวบ่งชี้หนึ่ง แต่ใช้ตัวบ่งชี้ทุกตัวกำหนดมาตร ทำให้ค่าเฉลี่ยองค์ประกอบ (α) เป็นค่าเฉลี่ยแบบถ่วงน้ำหนักของตัวบ่งชี้ (\bar{m}) สามารถกำหนดได้โดย
 - ทำให้น้ำหนักองค์ประกอบเฉลี่ยเท่ากับ 1
 - ทำให้จุดตัดของตัวบ่งชี้เฉลี่ยเท่ากับ 0
- การระบุโมเดลแบบ Effect Coding แม้มีประโยชน์ในการสร้างสเกลในองค์ประกอบ แต่นำไปใช้ในโมเดลที่ซับซ้อนต่อไปได้ลำบาก จึงไม่ได้เป็นวิธีที่ได้รับความนิยม

```

> m1eff <- 'ใส่ทั้ง NA และ label ในตัวบ่งชี้เดียวกัน
+ f1 =~ a1*x1 + NA*x1 + a2*x5
+ f2 =~ b1*x2 + NA*x2 + b2*x3 + b3*x4
+ f1 =~ v1*f1
+ f2 =~ v2*f2
+ x1 ~ i1*1
+ x5 ~ i2*1
+ x2 ~ j1*1
+ x3 ~ j2*1
+ x4 ~ j3*1
+ f1 ~ m1*1 + NA*1
+ f2 ~ m2*1 + NA*1
+ # (a1 + a2)/2 == 1
+ # a1 + a2 == 2
+ a1 == 2 - a2
+ # (b1 + b2 + b3)/3 == 1
+ # b1 + b2 + b3 == 3
+ b1 == 3 - b2 - b3
+ # (i1 + i2)/2 == 0
+ i1 == 0 - i2
+ # (j1 + j2 + j3)/3 == 0
+ j1 == 0 - j2 - j3
+ '
> outleff <- cfa(m1eff, data=dat)
> summary(outleff)
lavaan 0.6-12 ended normally after 48 iterations

```

ใส่ทั้ง NA และ label ในตัวบ่งชี้เดียวกัน

เครื่องหมาย # ไว้สำหรับคอมเมนต์

แสดงการไล่สมการ เพื่อให้หน้าหนัก

องค์ประกอบเฉลี่ยเป็น 1

แสดงการไล่สมการ เพื่อให้จุดตัด

ตัวบ่งชี้เฉลี่ยเป็น 1

Estimator	ML
Optimization method	NLMINB
Number of model parameters	20
Number of equality constraints	4
Number of observations	200
Model Test User Model:	
Test statistic	19.550
Degrees of freedom	4
P-value (Chi-square)	0.001

ค่าไคสแควร์เท่ากับการระบุโมเดลแบบอื่น

Latent Variables:

		Estimate	Std.Err	z-value	P(> z)
f1 =~					
x1	(a1)	1.097	0.157	6.980	0.000
x5	(a2)	0.903	0.157	5.751	0.000
f2 =~					
x2	(b1)	1.047	0.092	11.331	0.000
x3	(b2)	1.179	0.097	12.160	0.000
x4	(b3)	0.774	0.086	9.027	0.000

Covariances:

		Estimate	Std.Err	z-value	P(> z)
f1 =~					
f2		0.106	0.022	4.819	0.000

Intercepts:

		Estimate	Std.Err	z-value	P(> z)
.x1	(i1)	-0.152	0.342	-0.444	0.657
.x5	(i2)	0.152	0.342	0.444	0.657
.x2	(j1)	0.116	0.210	0.555	0.579
.x3	(j2)	-0.601	0.220	-2.729	0.006
.x4	(j3)	0.484	0.196	2.478	0.013
f1	(m1)	2.168	0.037	58.646	0.000
f2	(m2)	2.248	0.040	56.309	0.000

Variances:

		Estimate	Std.Err	z-value	P(> z)
f1	(v1)	0.096	0.033	2.930	0.003
f2	(v2)	0.226	0.032	6.968	0.000
.x1		0.359	0.055	6.496	0.000
.x5		0.349	0.045	7.749	0.000
.x2		0.221	0.040	5.546	0.000
.x3		0.293	0.051	5.706	0.000
.x4		0.319	0.038	8.430	0.000

เท่ากับค่าเฉลี่ยของตัวบ่งชี้

การกำหนดช่วงพารามิเตอร์ที่เป็นไปได้



- นักวิเคราะห์อาจต้องการให้ค่าพารามิเตอร์บางค่าอยู่ในช่วงที่ต้องการ เช่น
 - ภาวะสันนิษฐานหนึ่งมีทั้งตัวบ่งชี้ทางบวกและลบ ต้องการให้ตัวบ่งชี้ทางบวกมีน้ำหนักองค์ประกอบทางบวก
 - หรืออาจเจอปัญหาที่ค่าความแปรปรวนของค่าคงเหลือติดลบ ต้องการให้ ML หาค่าของพารามิเตอร์ความแปรปรวนเฉพาะค่ามากกว่า 0 เท่านั้น
 - ค่า residual variance < 0 จะเรียกว่า Heywood case
- Lavaan ลองรับค่า $>$ หรือ $<$ ในการกำหนดช่วงที่เป็นไปได้ของพารามิเตอร์
- ไม่มีผลกระทบต่อ df เนื่องจากจำนวนค่าพารามิเตอร์ที่ต้องการประมาณค่ายังเท่าเดิม เพียงแค่กำหนดช่วงที่เป็นไปได้เท่านั้น
- แต่ดัชนีความเหมาะสมอาจจะน้อยลง เนื่องจากช่วงพารามิเตอร์ที่อนุญาตไม่ถึงตำแหน่งที่ทำให้ความเป็นไปได้สูงสุด

```

> mlinequal <- '
+ f1 =~ a1*x1 + a2*x5
+ f2 =~ b1*x2 + b2*x3 + b3*x4
+ x1 =~ e1*x1
+ a1 > 0.01
+ b1 > 0.01
+ e1 > 0.01
+ b3 > b2
+ '
> outmlinequal<- cfa(mlinequal, data=dat, std.lv=TRUE)
> summary(outmlinequal)
lavaan 0.6.16 ended normally after 162 iterations

```

จำกัดช่วงที่เป็นไปได้ของพารามิเตอร์

Estimator	ML
Optimization method	NLMINB
Number of model parameters	11
Number of inequality constraints	4
Number of observations	200
Model Test User Model:	
Test statistic	25.313
Degrees of freedom	4
P-value (Chi-square)	0.000

Latent Variables:						
		Estimate	Std.Err	z-value	P(> z)	
f1 =~						
	x1	(a1)	0.315	0.074	4.240	0.000
	x5	(a2)	0.303	0.071	4.263	0.000
f2 =~						
	x2	(b1)	0.505	0.054	9.382	0.000
	x3	(b2)	0.452	0.042	10.683	0.000
	x4	(b3)	0.452	0.042	10.683	0.000
Covariances:						
		Estimate	Std.Err	z-value	P(> z)	
f1 ~						
	f2	0.708	0.141	5.039	0.000	
Variances:						
		Estimate	Std.Err	z-value	P(> z)	
.x1	(e1)	0.375	0.052	7.159	0.000	
.x5		0.336	0.048	7.046	0.000	
.x2		0.214	0.041	5.214	0.000	
.x3		0.355	0.044	8.055	0.000	
.x4		0.289	0.039	7.483	0.000	
f1		1.000				
f2		1.000				
Constraints:						
						Slack
a1	- (0.01)					0.305
b1	- (0.01)					0.495
e1	- (0.01)					0.365
b3	- (b2)					0.000

b2 = b3 เลย เพราะในหากไม่จำกัดช่วงแล้ว **b3 < b2** แต่เราจำกัดช่วงให้ **b3 > b2** ดังนั้นค่าที่เป็นไปได้สูงสุดในโจทย์นี้ คือตำแหน่งที่ **b2 = b3**

พารามิเตอร์เพิ่มเติม

- ใน lavaan รวมถึง SEM packages อื่น สามารถสร้างพารามิเตอร์เพิ่มเติม ได้จาก ฟังก์ชันจากพารามิเตอร์เดิมได้ เช่น ความแตกต่างระหว่างค่าเฉลี่ยสององค์ประกอบ
- ใช้ := ในการบอก lavaan ว่า label ทางซ้ายเท่ากับฟังก์ชันของค่าพารามิเตอร์เพิ่มเติม
- ไม่ส่งผลต่อดัชนีความเหมาะสมของโมเดล
- ค่าพารามิเตอร์บางค่า การกระจายจากการสุ่ม (Sampling Distribution) ไม่ได้เป็นโค้งปกติ (Normal Distribution) การหาระดับนัยสำคัญ อาจใช้วิธี Bootstrap ได้ โดยเลือก `se="boot"` ในตอนใช้คำสั่ง `cfa`

```

> m2eff <- '
+ f1 =~ a1*x1 + NA*x1 + a2*x5
+ f2 =~ b1*x2 + NA*x2 + b2*x3 + b3*x4
+ f1 =~ v1*f1
+ f2 =~ v2*f2
+ x1 ~ i1*1
+ x5 ~ i2*1
+ x2 ~ j1*1
+ x3 ~ j2*1
+ x4 ~ j3*1
+ f1 ~ m1*1 + NA*1
+ f2 ~ m2*1 + NA*1
+ # (a1 + a2)/2 == 1
+ # a1 + a2 == 2
+ a1 == 2 - a2
+ # (b1 + b2 + b3)/3 == 1
+ # b1 + b2 + b3 == 3
+ b1 == 3 - b2 - b3
+ # (i1 + i2)/2 == 0
+ i1 == 0 - i2
+ # (j1 + j2 + j3)/3 == 0
+ j1 == 0 - j2 - j3
+ mdiff := m1 - m2
+ vratio := v1/v2 - 1
> out2eff <- cfa(m2eff, data=dat)
> summary(out2eff)

```

เพิ่มพารามิเตอร์ใหม่ หาความแตกต่างระหว่าง
ค่าเฉลี่ย และความแปรปรวนระหว่างองค์ประกอบ

$$H_0: \sigma_1^2 = \sigma_2^2$$

$$H_0: \frac{\sigma_1^2}{\sigma_2^2} = 1$$

$$H_0: \frac{\sigma_1^2}{\sigma_2^2} - 1 = 0$$

Model Test User Model:

Test statistic	19.550
Degrees of freedom	4
P-value (Chi-square)	0.001

ดัชนีความเหมาะสมไม่แตกต่างจากเดิม

Defined Parameters:

	Estimate	Std.Err	z-value	P(> z)
mdiff	-0.081	0.044	-1.856	0.064
vratio	0.425	0.151	2.809	0.005

ความแตกต่างระหว่างค่าเฉลี่ยไม่ถึงระดับนัยสำคัญ และสัดส่วนความแปรปรวน
ลบ 1 แตกต่างจาก 0 อย่างมีนัยสำคัญ

แสดงว่าความแปรปรวนขององค์ประกอบที่ 2 สูงกว่าองค์ประกอบที่ 1
อย่างมีนัยสำคัญ

นักสถิติจะเรียกการทดสอบนี้ว่า Wald test หรือ Delta Method

การทดสอบนัยสำคัญด้วยวิธี bootstrap จำนวนการสุ่ม bootstrap ยิ่งมากยิ่งดี

```
> out2eff2 <- cfa(m2eff, data=dat, se="boot", bootstrap=500)
```

Warning message:

```
In lav_model_nvcov_bootstrap(lavmodel = lavmodel, lavsamplestats = lavsamplestats, :  
lavaan WARNING: 28 bootstrap runs resulted in nonadmissible solutions.
```

ถ้าใช้คำสั่ง **summary** การทดสอบจะเป็น **Wald test** เหมือนเดิม ต้องใช้ **parameterEstimates**

```
> parameterEstimates(out2eff2, boot.ci.type = "bca")
```

	lhs	op	rhs	label	est	se	z	pvalue	ci.lower	ci.upper
1	f1	≈	x1	a1	1.097	0.173	6.339	0.000	0.725	1.432
2	f1	≈	x5	a2	0.903	0.173	5.223	0.000	0.568	1.275
3	f2	≈	x2	b1	1.047	0.086	12.132	0.000	0.883	1.229
4	f2	≈	x3	b2	1.179	0.130	9.059	0.000	0.993	1.528
5	f2	≈	x4	b3	0.774	0.094	8.213	0.000	0.583	0.941
6	f1	≈	f1	v1	0.096	0.031	3.125	0.002	0.037	0.153
7	f2	≈	f2	v2	0.226	0.030	7.561	0.000	0.177	0.295
8	x1	≈	i1		-0.152	0.377	-0.403	0.687	-0.878	0.661
9	x5	≈	i2		0.152	0.377	0.403	0.687	-0.661	0.878
10	x2	≈	j1		0.116	0.200	0.582	0.560	-0.285	0.513
11	x3	≈	j2		-0.601	0.288	-2.089	0.037	-1.388	-0.181
12	x4	≈	j3		0.484	0.209	2.321	0.020	0.115	0.934
13	f1	≈	m1		2.168	0.036	60.494	0.000	2.098	2.238
14	f2	≈	m2		2.248	0.040	56.495	0.000	2.177	2.336
15	x1	≈	x1		0.359	0.065	5.489	0.000	0.237	0.476
16	x5	≈	x5		0.349	0.045	7.695	0.000	0.260	0.444
17	x2	≈	x2		0.221	0.045	4.879	0.000	0.140	0.311
18	x3	≈	x3		0.293	0.056	5.201	0.000	0.143	0.379
19	x4	≈	x4		0.319	0.036	8.777	0.000	0.250	0.400
20	f1	≈	f2		0.106	0.022	4.844	0.000	0.068	0.154
25	mdiff	:=	m1-m2	mdiff	-0.081	0.046	-1.743	<u>0.081</u>	-0.176	0.004
26	vratio	:=	v1/v2-1	vratio	-0.575	0.153	-3.756	<u>0.000</u>	-0.830	-0.258

วิธีการหาช่วงเชื่อมั่น อาจใช้ "perc" หรือ "bca" ที่ไม่ได้หมายความว่า **sampling distribution** เป็นโค้งปกติ

ความแตกต่างระหว่างค่าเฉลี่ยไม่แตกต่างจาก 0 อย่างมีนัยสำคัญ

สัดส่วนความแปรปรวนแตกต่างจาก 1 อย่างมีนัยสำคัญ

คะแนนองค์ประกอบ

- การหาคะแนนองค์ประกอบ (Factor Scores) คือ การหาค่าขององค์ประกอบของแต่ละบุคคล เพื่อใช้ในการประเมินรายบุคคล
- นอกจากนี้ โมเดลที่นักวิจัยสร้างขึ้นมาอาจซับซ้อนมาก จนต้องวิเคราะห์แยกเป็นส่วนๆ แล้วนำคะแนนองค์ประกอบที่ได้จากโมเดลที่แยกเป็นส่วนๆ นั้นไปวิเคราะห์ต่อ เช่น วิธีการวิเคราะห์เชิงโครงสร้างหลังการวัด (Structural after Measurement) หรือการทำกลุ่มข้อคำถาม (Parceling)
- การหาคะแนนองค์ประกอบ สามารถแบ่งวิธีออกได้อย่างง่าย 2 วิธี คือ
 - คะแนนองค์ประกอบอย่างละเอียด (Refined Factor Scores)
 - คะแนนองค์ประกอบอย่างหยาบ (Coarsen Factor Scores)

คะแนนองค์ประกอบ

- ให้โมเดลมีองค์ประกอบ K ตัว มีจำนวนข้อคำถาม J ข้อ คะแนนข้อคำถามข้อที่ j ของคนที่ i จะเกิดจากคะแนนองค์ประกอบดังนี้

$$X_{ij} = \mu_j + \lambda_{j1}F_{1i} + \lambda_{j2}F_{2i} + \cdots + \lambda_{jk}F_{ki} + e_{ij}$$

- จาก CFA เราจะทราบค่า μ และ λ แล้ว และทางซ้ายเราทราบค่า X_{ij} แล้ว จากข้อมูลนี้ เราต้องการหาค่า F_{ki} และ e_{ij}

คะแนนองค์ประกอบ

$$\begin{array}{l} J \text{ คำ} \\ X_{i1} = \mu_1 + \lambda_{11} F_{1i} + \lambda_{12} F_{2i} + \cdots + \lambda_{1k} F_{Ki} + e_{i1} \\ X_{i2} = \mu_2 + \lambda_{21} F_{1i} + \lambda_{22} F_{2i} + \cdots + \lambda_{2k} F_{Ki} + e_{i2} \\ X_{i3} = \mu_3 + \lambda_{31} F_{1i} + \lambda_{32} F_{2i} + \cdots + \lambda_{3k} F_{Ki} + e_{i3} \\ \vdots \\ X_{ij} = \mu_j + \lambda_{j1} F_{1i} + \lambda_{j2} F_{2i} + \cdots + \lambda_{jk} F_{Ki} + e_{ij} \end{array}$$

K คำ

J คำ

- หากมีข้อคำถาม J ข้อ และมีองค์ประกอบ K ข้อ จากสมการนี้ ข้อมูลของ 1 คน
 - คะแนนที่ทราบค่า (Knowns) คือ X มีทั้งหมด J คำ
 - คะแนนที่ไม่ทราบค่า (Unknowns) คือ F มีทั้งหมด K คำ และ e มี J คำ

คะแนนองค์ประกอบ

- จะเห็นว่า คะแนนที่ทราบค่า (J) น้อยกว่าคะแนนที่ไม่ทราบค่า ($J + K$) ทำให้ไม่สามารถหาค่า F และ e ที่แท้จริงได้
 - กล่าวคือ มีค่า F และ e ที่เป็นไปได้มากมาย ที่ทำให้คำนวณได้ X ตามข้อมูลที่เก็บได้
 - เรียกปรากฏการณ์นี้ว่า ความไม่แน่นอนของคะแนนองค์ประกอบ (Factor score indeterminacy)
- นักวิจัยไม่มีวันรู้ว่า คะแนน F และ e ในประชากรเป็นอย่างไร และไม่รู้เลยว่าชุดคะแนนไหนที่ใกล้เคียงกับคะแนนประชากร นักวิจัยจึงทำได้แค่เลือกชุดของคะแนนที่มีคุณสมบัติสอดคล้องกับโมเดล และน่าจะใกล้เคียงกับประชากร

คะแนนองค์ประกอบ

- เพื่อความสะดวก นักสถิติจึงหาวิธีคำนวณคะแนนองค์ประกอบแต่ละตัว จากคะแนนดิบแทน

$$F_{ik} = w_{1k}d_{i1} + w_{2k}d_{i2} + \cdots + w_{jk}d_{ij}$$

$$d_{ij} = X_{ij} - \mu_j$$

- โดย d เป็นข้อมูลที่ย้ายศูนย์กลางไปที่ค่าเฉลี่ย หรือ ค่าเบี่ยงเบน (deviation score) การหาคะแนนองค์ประกอบรายบุคคลสามารถเขียนในรูปเมทริกซ์ได้ดังนี้

$$\mathbf{f}_{K \times 1} = \mathbf{W}_{K \times J} \mathbf{d}_{J \times 1}$$

- ถ้าเป็น N คน เขียนได้ดังนี้

$$\mathbf{F}_{N \times K} = \mathbf{D}_{N \times J} \mathbf{W}_{J \times K}$$

คะแนนองค์ประกอบ

- คะแนนแบบละเอียด คือ การเลือกน้ำหนักที่ทำให้ \mathbf{F} ทุกตัว สามารถนำมาสร้างคะแนนดิบได้ กล่าวคือ เลือกน้ำหนักที่ทำให้สมการด้านล่างเป็นจริง

$$X_{ij} = \mu_j + \lambda_{j1}F_{1i} + \lambda_{j2}F_{2i} + \dots + \lambda_{jk}F_{Ki} + e_{ij}$$

- นักสถิติจึงมุ่งที่จะหาค่า $\mathbf{W}_{J \times K}$ ที่ทำให้สมการข้างบนเป็นจริง และบรรลุวัตถุประสงค์ที่ต้องการ
- Thurstone (1935) เสนอวิธีการวิเคราะห์ถดถอย (Regression Method) โดยใช้วิธีการหาสัมประสิทธิ์ \mathbf{W} คล้ายกับการหา $\mathbf{b} = \frac{\text{Cov}(\mathbf{x}, \mathbf{y})}{\text{Var}(\mathbf{x})}$ ในการวิเคราะห์ถดถอยได้สูตรดังนี้

$$\mathbf{W} = \mathbf{\Sigma}^{-1} \mathbf{\Lambda} \mathbf{\Phi}$$

คะแนนองค์ประกอบ

- วิธีการของ Thurstone จะทำให้ความสัมพันธ์ระหว่างคะแนนองค์ประกอบที่คำนวณได้ และคะแนนองค์ประกอบในโมเดล (ในเชิงทฤษฎี) มีค่าสูงสุด
- อย่างไรก็ตาม วิธีการของ Thurstone มีปัญหาที่ค่าสหสัมพันธ์ระหว่างคะแนนองค์ประกอบที่ได้มา ไม่ตรงกับความสัมพันธ์ระหว่างองค์ประกอบตามโมเดล

คะแนนองค์ประกอบ

- วิธีการของ Barlett เป็นอีกวิธีการหนึ่งที่มีใน lavaan เขาหา \mathbf{W} โดยพยายามลดความแตกต่างระหว่างคะแนนองค์ประกอบที่แท้จริงในประชากรและคะแนนองค์ประกอบที่ทำนายได้

$$\mathbf{W} = \mathbf{\Sigma}^{-1} \mathbf{\Lambda} (\mathbf{\Lambda}' \mathbf{\Sigma}^{-1} \mathbf{\Lambda})^{-1}$$

- คะแนนของ Barlett นอกจากจะพยายามเพิ่มความสัมพันธ์ระหว่างคะแนนองค์ประกอบที่คำนวณได้ กับคะแนนองค์ประกอบที่แท้จริงในประชากรแล้ว ยังไปลดความสัมพันธ์กับคะแนนองค์ประกอบอื่นที่ตั้งว่าไม่สัมพันธ์ด้วย (สำคัญตอนทำ EFA แบบหมุนแกนมุมฉาก)
- แต่วิธีการของ Barlett ก็ยังมีปัญหาเหมือนกับ Thurstone ที่ความสัมพันธ์ระหว่างคะแนนองค์ประกอบที่ได้มา ไม่ตรงกับความสัมพันธ์ระหว่างองค์ประกอบในโมเดล

คะแนนองค์ประกอบ

- Anderson & Rubin (1959) และ ten Berge et al. (1999) สามารถสร้างคะแนนองค์ประกอบที่มีความสัมพันธ์กัน ใกล้เคียงกับความสัมพันธ์ระหว่างองค์ประกอบในโมเดล แต่ไม่มีใน lavaan
 - จะมีใน psych package กรณีทำ EFA ซึ่งจะพูดถึงตอนเรียน EFA อีกครั้งหนึ่ง
- ค่าปกติของ lavaan คือ วิธี regression ของ Thurstone


```

> datneed <- read.table("lecture8manifestneed.csv", sep="," , header=TRUE)
> mneed <- '
+ naff ~ x2 + x6 + x10 + x13
+ nach ~ x1 + x5 + x9
+ ndom ~ x4 + x8 + x12 + x15
+ nauto ~ x3 + x7 + x11 + x14
+ '
> outneed <- cfa(mneed, data=datneed, std.lv=TRUE)
> fscores <- lavPredict(outneed)
> head(fscores)

```

ใช้คำสั่ง **predict** หรือ **lavPredict** ในการหาคะแนนองค์ประกอบ

```

      naff      nach      ndom      nauto
[1,]  0.2036527  1.2912463  0.56577283  0.4190600
[2,]  1.4227386  1.5613940  0.73154407 -0.9222163
[3,]  0.2323873  0.6957316 -0.07018300 -0.2944772
[4,]  1.4294810  1.1706076 -0.31319461 -1.5142991
[5,] -0.1093437  0.8583938 -0.07940991 -0.4176451
[6,] -2.4289915 -1.3070585  0.47954739  1.0201066

```

ค่าปกติ คือ **regression method** สำหรับตัวบ่งชี้ที่เป็นแบบต่อเนื่อง

```

> fscores2 <- lavPredict(outneed, method="Bartlett")
> head(fscores2)

```

เปลี่ยนวิธีการประมาณค่าคะแนนองค์ประกอบเป็นวิธีของ **Bartlett**

```

      naff      nach      ndom      nauto
[1,]  0.08648834  2.021546  0.6437096  0.5060543
[2,]  1.62352031  2.021546  1.0248302 -1.0311580
[3,]  0.13733687  1.070601 -0.1030289 -0.3300581
[4,]  1.62352031  1.452098 -0.3187202 -1.6915272
[5,] -0.44825791  1.452098 -0.1030289 -0.5142757
[6,] -3.22368623 -1.364726  0.6437096  0.9713116

```

```
> lavInspect(outneed, "cov.lv")
      naff  nach  ndom  nauto
naff  1.000
nach  0.523  1.000
ndom  -0.041  0.100  1.000
nauto -0.461 -0.225  0.305  1.000
```

หาเมทริกซ์ความแปรปรวนร่วมขององค์ประกอบในโมเดล

```
> cov(fscores)
```

```
      naff      nach      ndom      nauto
naff  0.78298424  0.4788903 -0.04355769 -0.4440930
nach  0.47889032  0.7110094  0.08719210 -0.2211318
ndom  -0.04355769  0.0871921  0.75401803  0.2922806
nauto -0.44409304 -0.2211318  0.29228055  0.8670278
```

เมทริกซ์ความแปรปรวนร่วมของคะแนนองค์ประกอบวิธี **Regression**

```
> cor(fscores)
```

```
      naff      nach      ndom      nauto
naff  1.00000000  0.6418327 -0.05668882 -0.5389903
nach  0.64183274  1.0000000  0.11908267 -0.2816417
ndom  -0.05668882  0.1190827  1.00000000  0.3614869
nauto -0.53899028 -0.2816417  0.36148688  1.0000000
```

เมทริกซ์สหสัมพันธ์ของคะแนนองค์ประกอบวิธี **Regression**

```
> cov(fscores2)
```

```
      naff      nach      ndom      nauto
naff  1.32172899  0.5235611 -0.04078519 -0.4619663
nach  0.52356107  1.4639016  0.10019654 -0.2255011
ndom  -0.04078519  0.1001965  1.34181234  0.3054054
nauto -0.46196627 -0.2255011  0.30540537  1.1636887
```

เมทริกซ์ความแปรปรวนร่วมของคะแนนองค์ประกอบวิธี **Bartlett**

```
> cor(fscores2)
```

```
      naff      nach      ndom      nauto
naff  1.00000000  0.37639183 -0.03062566 -0.3724952
nach  0.37639183  1.00000000  0.07149092 -0.1727724
ndom  -0.03062566  0.07149092  1.00000000  0.2444063
nauto -0.37249518 -0.17277239  0.24440629  1.0000000
```

เมทริกซ์สหสัมพันธ์ของคะแนนองค์ประกอบวิธี **Bartlett**

ความแปรปรวน และสหสัมพันธ์ของคะแนนองค์ประกอบของ
ทั้งสองวิธี ไม่ตรงกับในโมเดล

วิธีการคำนวณคะแนนองค์ประกอบจากข้อมูลใหม่ ให้ใส่ข้อมูลใหม่ในรูป **data.frame**
ใน **newdata**

```
> newdat <- datneed
> newdat <- newdat[1,]
> newdat[,paste0("x", 1:15)] <- 1
> newdat
  ID x1 x2 x3 x4 x5 x6 x7 x8 x9 x10 x11 x12 x13 x14 x15
1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
> lavPredict(outneed, newdata=newdat)
      naff  nach  ndom  nauto
[1,] -4.945 -4.69 -2.477 -1.083
>
> newdat[,paste0("x", 1:15)] <- 5
> newdat
  ID x1 x2 x3 x4 x5 x6 x7 x8 x9 x10 x11 x12 x13 x14 x15
1  1  5  5  5  5  5  5  5  5  5  5  5  5  5  5
> lavPredict(outneed, newdata=newdat)
      naff  nach  ndom  nauto
[1,]  0.996  1.541  2.735  2.512
```

คะแนนองค์ประกอบ

- วิธีการ Regression เป็นวิธีที่ได้รับความนิยมสูงสุดในการหาคะแนนองค์ประกอบใน SEM
- วิธีของ Bartlett ไม่ได้ให้คุณสมบัติที่ดีกว่าวิธี Regression
- ดังนั้น ถ้าไม่ติดอะไร แนะนำให้ใช้วิธี Regression ในการหาคะแนนองค์ประกอบแบบละเอียดไป เพราะถ้าใช้วิธี Bartlett อาจต้องอธิบาย Reviewers เพิ่มเติม
- วิธีของ ten Berge เป็นวิธีการที่ดี เพราะจะทำให้เมทริกซ์ความแปรปรวนร่วมของโมเดล และจากคะแนนองค์ประกอบใกล้เคียงกัน เสียหายที่ไม่มีวิธีการนี้ใน lavaan

คะแนนองค์ประกอบ

- คะแนนองค์ประกอบแบบหยาบ (Coarsen Factor Score) เป็นวิธีการหาคะแนนองค์ประกอบ โดยไม่ได้อิงว่าสมการนี้ต้องเป็นจริง

$$X_{ij} = \mu_j + \lambda_{j1}F_{1i} + \lambda_{j2}F_{2i} + \cdots + \lambda_{jk}F_{ki} + e_{ij}$$

- กล่าวคือ นักวิเคราะห์หาน้ำหนักอะไรก็ได้ มาเป็นน้ำหนักของคะแนนดิบ (ไม่ใช่ค่าเบี่ยงเบน) แล้วทำผลรวมเชิงเส้น (Linear Combination) เพื่อประมาณค่าคะแนนองค์ประกอบ

$$F_{ik} = w_{1k}X_{i1} + w_{2k}X_{i2} + \cdots + w_{jk}X_{ij}$$

- วิธีการที่ง่ายที่สุด คือ ให้ $w = 1$ สำหรับข้อที่น้ำหนักองค์ประกอบเป็นบวก และ $w = -1$ สำหรับข้อที่น้ำหนักองค์ประกอบเป็นลบ คะแนนที่ได้ก็คือผลรวมของคะแนนดิบแต่ละข้อ

```
> parameterEstimates(outneed)
```

	lhs	op	rhs	est	se	z	pvalue	ci.lower	ci.upper
1	naff	≈	x2	0.399	0.025	15.995	0.000	0.351	0.448
2	naff	≈	x6	0.516	0.026	19.898	0.000	0.465	0.567
3	naff	≈	x10	0.452	0.025	18.090	0.000	0.403	0.501
4	naff	≈	x13	0.539	0.028	19.408	0.000	0.484	0.593
5	nach	≈	x1	0.522	0.034	15.398	0.000	0.456	0.589
6	nach	≈	x5	0.543	0.031	17.703	0.000	0.483	0.603
7	nach	≈	x9	0.380	0.031	12.133	0.000	0.319	0.441
8	ndom	≈	x4	0.725	0.037	19.529	0.000	0.652	0.798
9	ndom	≈	x8	0.707	0.037	19.166	0.000	0.635	0.779
10	ndom	≈	x12	0.499	0.036	13.938	0.000	0.429	0.569
11	ndom	≈	x15	0.349	0.036	9.669	0.000	0.278	0.419
12	nauto	≈	x3	0.877	0.035	25.308	0.000	0.809	0.944
13	nauto	≈	x7	0.631	0.034	18.538	0.000	0.564	0.697
14	nauto	≈	x11	0.958	0.034	28.247	0.000	0.892	1.025
15	nauto	≈	x14	0.705	0.035	19.986	0.000	0.636	0.774

เนื่องจากน้ำหนักองค์ประกอบเป็นบวกทั้งหมด
ดังนั้น จึงให้ W เป็น 1 ทั้งหมด

```
> datneed[,"naff"] <- with(datneed, x2 + x6 + x10 + x13)
> datneed[,"nach"] <- with(datneed, x1 + x5 + x9)
> datneed[,"ndom"] <- with(datneed, x4 + x8 + x12 + x15)
> datneed[,"nauto"] <- with(datneed, x3 + x7 + x11 + x14)
> head(datneed)
```

ID	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	naff	nach	ndom	nauto	
1	1	5	4	2	3	5	4	4	3	5	4	3	4	5	3	4	17	15	14	12
2	2	5	5	2	3	5	5	3	4	5	5	1	4	5	1	3	20	15	14	7
3	3	4	4	2	2	5	5	2	3	4	5	2	4	3	3	3	17	13	12	9
4	4	4	5	1	2	5	5	1	2	5	5	1	4	5	1	5	20	14	13	4
5	5	4	4	2	2	5	4	2	3	5	4	2	4	4	2	3	16	14	12	8
6	6	4	4	4	3	3	2	4	3	3	2	3	4	3	2	4	11	10	14	13

คะแนนองค์ประกอบเป็นผลรวมของข้อคำถาม
ที่อยู่ภายใต้ข้อประกอบ (จะใช้เป็นค่าเฉลี่ยระหว่าง
ตัวบ่งชี้ก็ได้)

คะแนนองค์ประกอบ

- นักวิจัยเชื่อว่าคะแนนองค์ประกอบแบบหยาบจะไม่แกว่งไปแกว่งมาระหว่างกลุ่มตัวอย่าง
- คะแนนองค์ประกอบแบบละเอียด ทำให้เมื่อคำนวณจากกลุ่มตัวอย่างหนึ่ง จะได้น้ำหนักแบบหนึ่ง แต่คำนวณอีกตัวอย่างหนึ่ง จะได้น้ำหนักอีกแบบหนึ่ง ไม่มีความแน่นอน
- นอกจากนี้ คะแนนองค์ประกอบแบบหยาบสามารถคำนวณได้ง่าย จึงได้รับความนิยมในการประมาณค่าคะแนนองค์ประกอบ ในมาตรฐานใหญ่
- มาตรฐานที่มีตารางการเทียบคะแนน หรือมีโปรแกรมคำนวณคะแนน และมีกลุ่มตัวอย่างอ้างอิงขนาดใหญ่ อาจใช้คะแนนองค์ประกอบแบบละเอียด เช่น มาตรฐานวัดเชาว์ปัญญาของ Wechsler

คะแนนองค์ประกอบ

- ต้องระมัดระวังเรื่องการรวมคะแนน ที่ตัวบ่งชี้แต่ละตัวต้องอยู่ในสเกลเดียวกัน (เช่น เกิดจากการประเมินคะแนนช่วง 1-5 เหมือนกัน) ถ้าไม่ได้อยู่ในมาตรเดียวกัน ควรทำให้เป็นคะแนนมาตรฐานก่อนรวมคะแนน
- บางคน เสนอว่าให้นำน้ำหนักองค์ประกอบ มาใช้เป็นน้ำหนักในการสร้างผลรวมเชิงเส้น วิธีนี้ไม่เหมาะสม เพราะน้ำหนักองค์ประกอบเหวี่ยงไปมาระหว่างกลุ่มตัวอย่าง และมีวิธีการคำนวณคะแนนองค์ประกอบอย่างละเอียด ที่มีความถูกต้องมากกว่าอยู่แล้ว

ความเที่ยง

- ความเที่ยงเป็นสัดส่วนที่บอกว่า คะแนนองค์ประกอบที่ประมาณการได้นั้น เกิดจากคะแนนองค์ประกอบที่แท้จริงมากน้อยเพียงใด
 - ดังนั้น วิธีการหาคะแนนองค์ประกอบที่แตกต่างกัน ก็ส่งผลให้เกิดความเที่ยงที่แตกต่างกัน
 - เริ่มต้น อธิบายความเที่ยงที่เกิดจากคะแนนองค์ประกอบจากวิธีแบบหยาบ ที่น้ำหนักองค์ประกอบเท่ากับ 1, 0, และ -1
- สมมติว่า คะแนนองค์ประกอบหนึ่ง เกิดจากผลรวม 3 ตัวบ่งชี้ที่โมเดลเป็นโครงสร้างแบบง่าย (Simple Structure) ที่ตัวบ่งชี้หนึ่งอธิบายเพียงแค่องค์ประกอบเดียว

$$\hat{F}_{1i} = X_{i1} + X_{i2} + X_{i3} \quad X_{ij} = \mu_j + \lambda_{j1}F_{1i} + e_{ij}$$

$$\hat{F}_{1i} = \mu_1 + \lambda_{11}F_{1i} + e_{i1} + \mu_2 + \lambda_{21}F_{1i} + e_{i2} + \mu_3 + \lambda_{31}F_{1i} + e_{i3}$$

$$\hat{F}_{1i} = \sum_{j=1}^3 \mu_j + F_{1i} \sum_{j=1}^3 \lambda_{j1} + \sum_{j=1}^3 e_{ij}$$

ความเที่ยง

- หาความแปรปรวนผลรวมของคะแนนตัวบ่งชี้

$$\text{Var}(\hat{F}_{1i}) = \text{Var}\left(\sum_{j=1}^3 \mu_j + F_{1i} \sum_{j=1}^3 \lambda_{j1} + \sum_{j=1}^3 e_{ij}\right)$$

$$\text{Var}(\hat{F}_{1i}) = \text{Var}\left(\sum_{j=1}^3 \mu_j\right) + \text{Var}\left(F_{1i} \sum_{j=1}^3 \lambda_{j1}\right) + \text{Var}\left(\sum_{j=1}^3 e_{ij}\right)$$

$$\text{Var}(\hat{F}_{1i}) = 0 + \text{Var}(F_{1i}) \left(\sum_{j=1}^3 \lambda_{j1}\right)^2 + \sum_{j=1}^3 \text{Var}(e_{ij})$$

ส่วนที่เกิดจาก F จริง

$$\text{Var}(\hat{F}_{1i}) = \phi_{11} \left(\sum_{j=1}^3 \lambda_{j1}\right)^2 + \sum_{j=1}^3 \theta_{jj}$$

ส่วนที่เกิดจากค่าคงเหลือ

ความเที่ยง

- ความเที่ยงจะชี้ให้เห็นว่า ในการเปลี่ยนแปลงที่เกิดขึ้นจากผลรวมนั้น เกิดจากองค์ประกอบคิดเป็นกี่เปอร์เซ็นต์ เรียกค่านี้ว่าสัมประสิทธิ์โอเมก้า (Coefficient Omega)

$$\omega_k = \frac{\phi_{kk} \left(\sum_{j=1}^J \lambda_{jk} \right)^2}{\phi_{kk} \left(\sum_{j=1}^J \lambda_{jk} \right)^2 + \sum_{j=1}^J \theta_{jj}}$$

- ตัวเศษ คือ ความแปรปรวนที่เกิดขึ้นจากองค์ประกอบ และตัวส่วน คือ ความแปรปรวนของผลรวมคะแนน ที่เกิดจากทั้งองค์ประกอบและค่าคงเหลือ
- ถ้า λ_j มีค่าเท่ากันทุกค่า (หรือที่เรียกว่าโมเดล tau-equivalent) ค่าโอเมก้าจะมีค่าเท่ากับสัมประสิทธิ์อัลฟา (Coefficient Alpha) ที่เรียนในวิชาการวัดและการประเมิน

ความเที่ยง

- นอกจากนี้ ตัวส่วนของสูตรโอเมก้าควรเป็นความแปรปรวนของผลรวมคะแนน ซึ่งสามารถคำนวณได้สองแบบ คือ คำนวณจากโมเดลเป้าหมายในสูตรที่เพิ่งแสดงไป หรือคำนวณจากความแปรปรวนของผลรวมจริงๆ ดังนี้

$$\omega_k = \frac{\phi_{kk} \left(\sum_{j=1}^J \lambda_{jk} \right)^2}{\text{Var} \left(\sum_{j=1}^J X_j \right)}$$

- การคำนวณสัมประสิทธิ์โอเมก้ารูปแบบนี้จะเหมาะสมมากกว่า เพราะโมเดลเป้าหมายอาจไม่สอดคล้องกับข้อมูลจริงๆ อาจมีความผิดพลาดของโมเดลบ้าง (Minor misspecification) การคำนวณผ่านผลรวมโดยตรง จะรวมความผิดพลาดโมเดลไปแล้ว และสูตรนี้ยังเหมาะสมในกรณีที่โมเดลมีความสัมพันธ์ระหว่างค่าคงเหลือด้วย

```
> library(semTools)
```

```
> compRelSEM(outneed, obs.var=TRUE) ความเที่ยงแบบคำนวณตัวส่วน ด้วยค่าจากใน โมเดล
```

```
naff nach ndom nauto  
0.754 0.659 0.692 0.831
```

```
> compRelSEM(outneed)
```

ความเที่ยงแบบคำนวณตัวส่วน ด้วยความแปรปรวนของคะแนนผลรวม แนะนำใช้ค่านี้

```
naff nach ndom nauto  
0.754 0.659 0.692 0.831
```

```
> compRelSEM(outneed, tau.eq=TRUE)
```

```
naff nach ndom nauto  
0.751 0.629 0.693 0.830
```

ความเที่ยงแบบจำกัดให้น้ำหนักองค์ประกอบเท่ากัน ให้สะท้อนค่า **alpha** ซึ่งไม่แนะนำ

ฟังก์ชันนี้จะเหมารวม ให้องค์ประกอบเท่ากับผลรวมของตัวบ่งชี้
ที่มีน้ำหนักองค์ประกอบไม่เท่ากับ 0 โดยให้น้ำหนัก 0 หรือ 1
เท่านั้น ไม่คำนึงถึงการลบข้อที่มีการกลับทิศ

หากจะใช้สูตรนี้ นักวิเคราะห์ต้องกลับคะแนนก่อนนำตัวบ่งชี้วิเคราะห์ด้วยโมเดล

เดิม **semTools** มีฟังก์ชันการวิเคราะห์ชื่อว่า **reliability** แต่ผลที่ออกมาค่อนข้าง
อ่านยาก จึงได้อัพเดทฟังก์ชันใหม่ชื่อว่า **compRelSEM** ที่สามารถคำนวณความเที่ยงได้

ความเที่ยง

- สำหรับความเที่ยงของคะแนนองค์ประกอบแบบละเอียด หรือแบบหยาบที่ไม่ใช่ผลรวม

$$\omega_k = \frac{\phi_{kk} \left(\sum_{j=1}^J w_{jk} \lambda_{jk} \right)^2}{\text{Var} \left(\sum_{j=1}^J w_{jk} X_j \right)}$$

- กรณีการหาคะแนนองค์ประกอบแบบละเอียดผ่าน lavaan สามารถใช้ script คำนวณดังนี้

```
> fscores <- lavPredict(outneed, fsm=TRUE)
> wfs <- attr(fscores, "fsm")[[1]]
> lambda <- inspect(outneed, "est")[[ "lambda" ]]
> psi <- inspect(outneed, "est")[[ "psi" ]]
>
> n1 <- wfs %*% lambda %*% psi %*% t(lambda) %*% t(wfs)
> d1 <- cov(fscores)
> diag(n1) / diag(d1)
      naff      nach      ndom      nauto
0.8010853 0.7333659 0.7590879 0.8692057
```

ความเที่ยง

- แนวทางในการตีความหมายความเที่ยง (Kline, 2005)
 - .7 คือ พอใช้
 - .8 คือ ดี
 - .9 คือ ดีมาก
- ขึ้นอยู่กับวัตถุประสงค์การใช้ด้วย หากคุณต้องการใช้แบบวัดในการตัดสินใจ (เช่น เซาวัน ปัญญา) ความเที่ยงควรจะสูงมาก แต่หากนำมาใช้ในการวิจัยเท่านั้น ความเที่ยงระดับดี หรือพอใช้ ยังถือว่าใช้ได้