

---

## PSP6075525 - Testing psicologico (matr. dispari)

Caso studio del 25-01-21

### Istruzioni iniziali

- Si avvii una nuova sessione di R (o RStudio).
- Si crei un nuovo script di R e lo si salvi come `cognome_nome.R`.
- Si effettui il download del file di dati dell'esame `dati_esame.Rdata` disponibile presso la pagina moodle del corso e lo si carichi nell'ambiente di lavoro di R.
- Si crei un nuovo documento di testo (mediante LibreOffice Writer, Microsoft Word o software analogo) e lo si salvi come `cognome_nome.doc`. Il file dovrà contenere le risposte ai quesiti d'esame accompagnati dai comandi di R, dai risultati ottenuti e dai grafici prodotti. Le risposte dovranno essere inserite in ordine, rispettando il numero del quesito a cui si riferiscono. Alla fine, il file dovrà essere convertito in formato non modificabile (PDF: `cognome_nome.pdf`) ed inviato al docente utilizzando la procedura "Consegna documento" disponibile presso la pagina Moodle del corso. Nel caso di utilizzo di **R-markdown** per la compilazione dinamica di documenti di testo, sarà necessario inviare il file sorgente `.Rnw` unitamente al file PDF generato.
- La valutazione della prova sarà effettuata utilizzando primariamente il file `cognome_nome.pdf`: si raccomanda pertanto la chiarezza nella scrittura delle risposte e la correttezza nel riportare i comandi e gli output di R. Il file `cognome_nome.R` dovrà essere allegato al file `cognome_nome.pdf` solo per un controllo aggiuntivo (pertanto non verrà primariamente valutato).

## Caso studio

Il dataset complessivo contiene 12 variabili riferite ad un test di personalità somministrato ad un campione di  $n = 1500$  partecipanti. Le variabili originarie, rilevate mediante scala di rating a cinque punti, sono state pre-trattate mediante un'adeguata procedura di quantificazione. Le variabili sono riferite ad item che indagano la struttura del sé (SELF), aspetti depressivi (DEPRESS) e di impulsività (IMPULS). L'obiettivo dell'analisi è quello di definire e adattare uno o più modelli CFA per lo studio della dimensionalità del test in oggetto. Considerata la numerosità delle unità statistiche a disposizione, il dataset originario è stato casualmente diviso in due sottoinsiemi rispettivamente di  $n = 900$  (Y0) e  $n = 600$  (Y1) unità. Questo consente di poter utilizzare il dataset Y0 per le analisi esplorative e di utilizzare invece Y1 per quelle confermative.

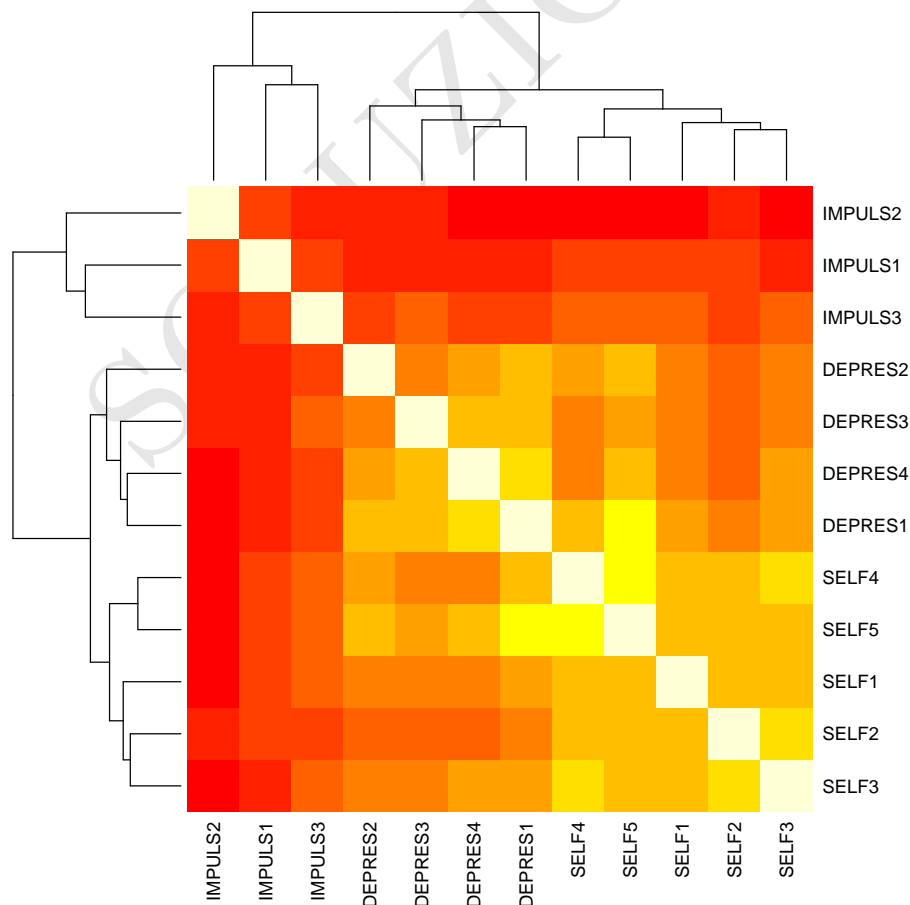
1. Si individuino il numero di unità statistiche e di variabili a disposizione, indicando per queste ultime il tipo di variabili coinvolte.

Il numero di unità statistiche è pari a  $n = 1500$  mentre le variabili coinvolte sono  $p = 12$ , tutte di tipo numerico (variabili reali).

2. Utilizzando il dataset Y0, si rappresenti graficamente la struttura di correlazione delle variabili e la loro aggregazione gerarchica.

Una rappresentazione grafica che visualizzi al contempo l'associazione lineare tra le  $p$  variabili e la loro aggregazione gerarchica è ottenibile mediante un grafico tipo **heatmap**, come segue:

```
cor_Y0 = cor(Y0)
heatmap(x = cor_Y0, symm = TRUE)
```

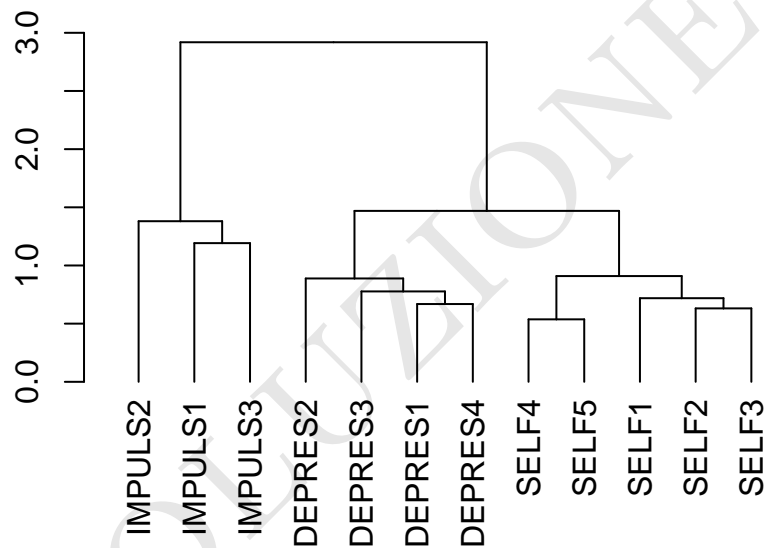


I parametri di default della funzione `heatmap()` consentono di convertire la matrice di correlazione  $\mathbf{R}_{p \times p}$  in matrice di distanze e di usare una tecnica di clustering gerarchico con metodo Ward per la definizione

dei cluster di variabili. Il risultato ci consente di osservare la presenza di due cluster separati tra loro, uno dei quali all'interno è suddiviso in due sottosistemi. Utilizzando la descrizione degli item a disposizione, notiamo come la soluzione di aggregazione a tre cluster segua la partizione degli item rispetto a SELF, DEPRESS e IMPULS.

3. Si ottenga lo stesso risultato del punto precedente mediante clustering gerarchico tipo Ward applicato sulla matrice di correlazione delle variabili osservate. Si scelgano ragionevolmente due diverse soluzioni di aggregazione sulla base dei risultati ottenuti.<sup>1</sup>

```
h0 = hclust(d = dist(cor_Y0),method = "ward.D2")
plot(as.dendrogram(h0))
```



La soluzione di clustering gerarchico con metodo Ward suggerisce la presenza di due cluster di aggregazione, il primo relativo alle variabili riferite a IMPULS e il secondo relativo alle variabili riferite a SELF e DEPRESS (soluzione 1). Una seconda soluzione suggerisce la presenza di tre cluster, ciascuno riferito alle variabili SELF, DEPRESS e IMPULS (soluzione 2).<sup>2</sup>

```
s1=cutree(h0,k = 2) #soluzione 1
s2=cutree(h0,k = 3) #soluzione 2
print(cbind(s1,s2))
```

	s1	s2
SELF1	1	1
SELF2	1	1
SELF3	1	1
SELF4	1	1
SELF5	1	1

<sup>1</sup> Si suggerisce di ricercare una prima soluzione a due cluster ed una seconda soluzione a tre cluster.

<sup>2</sup> Per maggiori informazioni sul clustering gerarchico, si veda:

<https://statsandr.com/blog/files/Hierarchical-clustering-cheatsheet.pdf>.

DEPRES1	1	2
DEPRES2	1	2
DEPRES3	1	2
DEPRES4	1	2
IMPULS1	2	3
IMPULS2	2	3
IMPULS3	2	3

4. Utilizzando il dataset **Y1**, si definisca un modello CFA ortogonale relativo alla soluzione 1 ottenuta al punto precedente, lo si adatti ai dati a disposizione e se ne valuti l'adattamento complessivo.

Il modello CFA ortogonale per la soluzione 1 (vedi punto 3) è definito dall'equazione lineare:

$$\Sigma_{y_{12 \times 12}} = \Lambda_{12 \times 2} \Phi_{2 \times 2} \Lambda_{12 \times 2}^T + \Theta_{\delta_{12 \times 12}}$$

dove la matrice delle correlazione tra le variabili latenti è diagonale:

$$\Phi_{2 \times 2} = \begin{bmatrix} \phi_{\eta_1} & 0 \\ 0 & \phi_{\eta_2} \end{bmatrix}$$

L'adattamento ai dati del modello ortogonale è effettuato mediante la libreria **lavaan** come segue:

```
model1 = "eta1=~SELF1+SELF2+SELF3+SELF4+SELF5+DEPRES1+DEPRES2+DEPRES3+DEPRES4
eta2=~IMPULS1+IMPULS2+IMPULS3
eta1~~0*eta2"
model1_fit = lavaan::cfa(model = model1, data = Y1)
```

Dopo aver adattato il modello ai dati, vale a dire dopo aver stimato i parametri del modello  $\Lambda$ ,  $\Phi$ ,  $\Theta_\delta$ , l'adattamento globale può essere valutato ad esempio mediante gli indici RMSEA o CFI:

```
fm1 = lavaan::fitmeasures(object = model1_fit, fit.measures = c("cfi", "rmsea", "df", "AIC"))
print(fm1)
```

cfi	rmsea	df	aic
0.876	0.102	54.000	18064.312

Il valore degli indici RMSEA e CFI indicano che il modello adattato non è globalmente soddisfacente e necessita di ulteriori miglioramenti. Rispetto alla parsimoniosità, il modello CFA relativo alla soluzione 1 presenta un rapporto tra numero di parametri totali e numero di parametri liberi pari a  $\frac{m}{(p(p+1)/2)} = 0.308$ , non particolarmente alto (un rapporto pari ad 1 indica un modello saturo).

5. Utilizzando il dataset **Y1**, si definisca un modello CFA relativo alla soluzione 2 ottenuta al punto precedente, lo si adatti ai dati a disposizione e se ne valuti l'adattamento complessivo. Si confronti inoltre l'adattamento ottenuto rispetto a quello del punto 4.

Il modello CFA ortogonale per la soluzione 2 (vedi punto 3) è definito dall'equazione lineare:

$$\Sigma_{y_{12 \times 12}} = \Lambda_{12 \times 3} \Phi_{3 \times 3} \Lambda_{12 \times 3}^T + \Theta_{\delta_{12 \times 12}}$$

dove, nel caso generale (non ortogonale), la matrice delle correlazioni tra le variabili latenti non è diagonale:

$$\Phi_{3 \times 3} = \begin{bmatrix} \phi_{\eta_{11}} & \phi_{\eta_{12}} & \phi_{\eta_{13}} \\ \phi_{\eta_{21}} & \phi_{\eta_{22}} & \phi_{\eta_{23}} \\ \phi_{\eta_{31}} & \phi_{\eta_{32}} & \phi_{\eta_{33}} \end{bmatrix}$$

L'adattamento del modello ai dati è effettuato ancora una volta mediante la libreria **lavaan** come segue:

```
model2 = "eta1=~SELF1+SELF2+SELF3+SELF4+SELF5
eta2=~IMPULS1+IMPULS2+IMPULS3
eta3=~DEPRES1+DEPRES2+DEPRES3+DEPRES4"
model2_fit = lavaan::cfa(model = model2,data = Y1)
```

Dopo aver adattato il modello ai dati, l'adattamento globale può essere valutato come segue:

```
fm2 = lavaan::fitmeasures(object = model2_fit,fit.measures = c("cfi","rmsea","df","AIC"))
fm = rbind(fm1,fm2); rownames(fm) = c("mod1","mod2")
print(fm)
```

	cfi	rmsea	df	aic
mod1	0.8763008	0.10209409	54	18064.31
mod2	0.9321612	0.07779794	51	17914.81

Il modello CFA relativo alla soluzione 2 presenta indici RMSEA e CFI migliori rispetto al modello relativo alla soluzione 1. Ciò è anche evidenziato dall'indice AIC. In termini di parsimoniosità, il modello appena adattato presenta un rapporto tra numero di parametri totali e numero di parametri liberi pari a  $\frac{m}{(p(p+1)/2)} = 0.346$ , leggermente superiore a quello del modello CFA precedente ma tuttavia non particolarmente alto. In generale, entrambi i modelli possono essere ritenuti parsimoniosi.

- Si utilizzi una procedura razionale per migliorare il modello finale scelto al punto 5 e si individui il modello che meglio si adatta ai dati rispetto a quest'ultimo.

Una procedura razionale per migliorare un modello adattato è quella che prevede l'utilizzo dei c.d. indici di modifica. Un indice di modifica è il risultato di un test fatto sulla struttura fattoriale corrente rispetto alle strutture fattoriali che si otterrebbero se venissero stimati dei parametri assenti nella forma attuale. La statistica test utilizzata segue in distribuzione la *t-Student*: valori della statistica  $|T| > 4$  possono essere utilizzati per aggiungere il parametro corrispondente al test effettuato. La procedura è implementata dalla funzione `modificationindices()` della libreria `lavaan`.

```
head(modificationindices(object = model2_fit,sort. = TRUE),10)
```

	lhs	op	rhs	mi	epc	sepc.lv	sepc.all	sepc.nox
51	eta3	=~	SELF5	57.840	0.799	0.650	0.651	0.651
97	SELF5	~~	DEPRES2	28.362	0.113	0.113	0.267	0.267
56	SELF1	~~	SELF3	25.788	0.123	0.123	0.241	0.241
66	SELF2	~~	SELF3	22.813	0.119	0.119	0.223	0.223
77	SELF3	~~	SELF5	18.108	-0.092	-0.092	-0.244	-0.244
119	DEPRES2	~~	DEPRES4	16.379	0.106	0.106	0.206	0.206
48	eta3	=~	SELF2	16.030	-0.469	-0.382	-0.382	-0.382
89	SELF4	~~	DEPRES1	14.362	0.071	0.071	0.213	0.213
37	eta1	=~	DEPRES4	14.002	-0.567	-0.380	-0.380	-0.380
68	SELF2	~~	SELF5	13.486	-0.083	-0.083	-0.194	-0.194

Notiamo dalla colonna `mi` che il parametro da aggiungere è  $\lambda_{5,3}$  relativo al legame tra la variabile latente  $\eta_3$  e la variabile manifesta  $Y_5$  (colonne: `lhs`, `op`, `rhs`). La procedura suggerisce anche altre modifiche da apportare. Procediamo, per il momento, aggiungendo un parametro alla volta.

```
model3 = "eta1=~SELF1+SELF2+SELF3+SELF4+SELF5
eta2=~IMPULS1+IMPULS2+IMPULS3
eta3=~DEPRES1+DEPRES2+DEPRES3+DEPRES4+SELF5"
model3_fit = lavaan::cfa(model = model3,data = Y1)
```

```
fm3 = lavaan::fitmeasures(object = model3_fit, fit.measures = c("cfi", "rmsea", "df", "AIC"))

fm = rbind(fm1, fm2, fm3); rownames(fm) = c("mod1", "mod2", "mod3")
print(fm)
```

	cfi	rmsea	df	aic
mod1	0.8763008	0.10209409	54	18064.31
mod2	0.9321612	0.07779794	51	17914.81
mod3	0.9510144	0.06676724	50	17864.34

Il nuovo modello presenta valori di adattamento globale decisamente migliori rispetto al modello adattato al punto 5. Procediamo valutando possibili ulteriori miglioramenti.

```
head(modificationindices(object = model3_fit, sort. = TRUE), 10)
```

	lhs	op	rhs	mi	epc	sepc.lv	sepc.all	sepc.nox
51	eta3	=~	SELF4	21.907	0.440	0.357	0.357	0.357
89	SELF4	~~	DEPRES1	21.623	0.087	0.087	0.263	0.263
57	SELF1	~~	SELF4	16.867	-0.100	-0.100	-0.244	-0.244
97	SELF5	~~	DEPRES2	16.534	0.087	0.087	0.210	0.210
85	SELF4	~~	SELF5	15.920	0.080	0.080	0.258	0.258
56	SELF1	~~	SELF3	14.169	0.093	0.093	0.195	0.195
53	eta3	=~	IMPULS2	13.532	-0.242	-0.196	-0.196	-0.196
118	DEPRES2	~~	DEPRES3	13.208	-0.098	-0.098	-0.173	-0.173
119	DEPRES2	~~	DEPRES4	12.771	0.092	0.092	0.180	0.180
101	IMPULS1	~~	IMPULS3	12.724	-0.529	-0.529	-0.832	-0.832

```
model4 = model3 = "eta1=~SELF1+SELF2+SELF3+SELF4+SELF5
eta2=~IMPULS1+IMPULS2+IMPULS3
eta3=~DEPRES1+DEPRES2+DEPRES3+DEPRES4+SELF5+SELF4"
model4_fit = lavaan::cfa(model = model4, data = Y1)
fm4 = lavaan::fitmeasures(object = model4_fit, fit.measures = c("cfi", "rmsea", "df", "AIC"))

fm = rbind(fm1, fm2, fm3, fm4); rownames(fm) = c("mod1", "mod2", "mod3", "mod4")
print(fm)
```

	cfi	rmsea	df	aic
mod1	0.8763008	0.10209409	54	18064.31
mod2	0.9321612	0.07779794	51	17914.81
mod3	0.9510144	0.06676724	50	17864.34
mod4	0.9599908	0.06095314	49	17840.83

Il nuovo modello presenta valori di adattamento globale abbastanza sovrapponibili a quelli ottenuti dal modello precedente.

```
head(modificationindices(object = model4_fit, sort. = TRUE), 10)
```

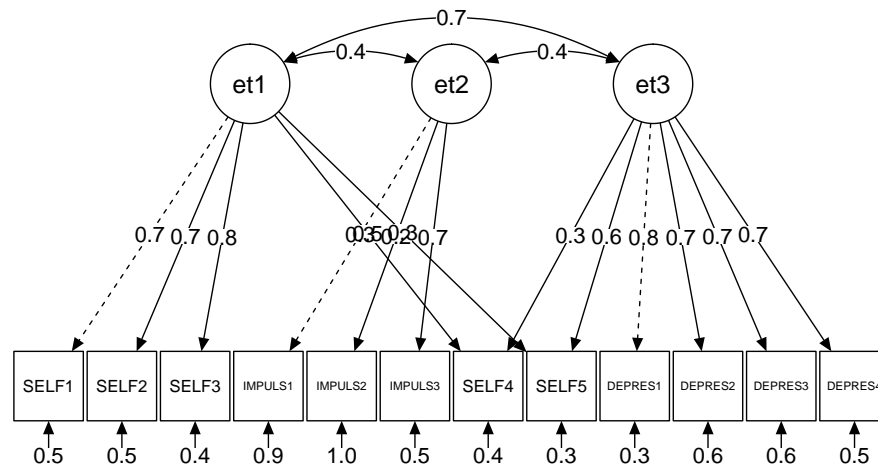
	lhs	op	rhs	mi	epc	sepc.lv	sepc.all	sepc.nox
92	SELF4	~~	DEPRES4	20.076	-0.093	-0.093	-0.228	-0.228
85	SELF4	~~	SELF5	16.022	0.072	0.072	0.221	0.221
53	eta3	=~	IMPULS2	14.866	-0.253	-0.206	-0.207	-0.207
119	DEPRES2	~~	DEPRES4	14.584	0.098	0.098	0.188	0.188
97	SELF5	~~	DEPRES2	13.102	0.078	0.078	0.188	0.188
101	IMPULS1	~~	IMPULS3	12.221	-0.514	-0.514	-0.800	-0.800

120	DEPRES3	~~	DEPRES4	11.708	0.087	0.087	0.170	0.170
115	DEPRES1	~~	DEPRES2	11.534	-0.084	-0.084	-0.191	-0.191
118	DEPRES2	~~	DEPRES3	10.766	-0.088	-0.088	-0.154	-0.154
69	SELF2	~~	IMPULS1	10.010	0.099	0.099	0.142	0.142

Un ulteriore utilizzo della procedura basata sugli indici di modifica suggerisce l'inclusione di parametri relativi alla correlazione tra gli errori delle variabili manifeste. Considerato che tali parametri, se inclusi, non migliorerebbero l'interpretazione del modello risultante e che l'ultimo modello (**model4**) presenta indici di adattamento accettabili, si decide di non procedere oltre con il miglioramento del modello.

7. Si rappresenti graficamente il modello finale scelto al punto 6 e lo si interpreti.

```
semPlot::semPaths(object = model4_fit, whatLabels = "std.all", edge.label.cex = 0.95,
  edge.color = "black", sizeMan = 7, sizeLat = 8, style = "lisrel", nDigits = 1)
```



```
summary(model4_fit, standardized=TRUE)
```

lavaan 0.6-7 ended normally after 46 iterations

Estimator	ML
Optimization method	NLMINB
Number of free parameters	29

Number of observations	600
------------------------	-----

Model Test User Model:

Test statistic	158.229
Degrees of freedom	49
P-value (Chi-square)	0.000

Parameter Estimates:

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

Latent Variables:						
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
eta1 =~						
SELF1	1.000				0.714	0.714
SELF2	0.954	0.064	15.018	0.000	0.681	0.681
SELF3	1.106	0.065	17.055	0.000	0.789	0.790
SELF4	0.749	0.082	9.144	0.000	0.535	0.535
SELF5	0.451	0.072	6.281	0.000	0.322	0.322
eta2 =~						
IMPULS1	1.000				0.313	0.313
IMPULS2	0.552	0.190	2.897	0.004	0.173	0.173
IMPULS3	2.348	0.681	3.448	0.001	0.735	0.735
eta3 =~						
DEPRES1	1.000				0.814	0.815
DEPRES2	0.798	0.049	16.260	0.000	0.650	0.651
DEPRES3	0.808	0.049	16.496	0.000	0.658	0.658
DEPRES4	0.893	0.048	18.593	0.000	0.727	0.728
SELF5	0.707	0.064	10.963	0.000	0.575	0.576
SELF4	0.402	0.066	6.090	0.000	0.327	0.328
Covariances:						
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
eta1 ~~						
eta2	0.089	0.027	3.253	0.001	0.397	0.397
eta3	0.414	0.039	10.632	0.000	0.712	0.712
eta2 ~~						
eta3	0.093	0.029	3.217	0.001	0.366	0.366
Variances:						
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.SELF1	0.489	0.035	14.063	0.000	0.489	0.490
.SELF2	0.535	0.037	14.632	0.000	0.535	0.536
.SELF3	0.375	0.031	11.995	0.000	0.375	0.376
.SELF4	0.356	0.025	13.977	0.000	0.356	0.357
.SELF5	0.300	0.022	13.485	0.000	0.300	0.300
.IMPULS1	0.900	0.059	15.201	0.000	0.900	0.902
.IMPULS2	0.969	0.057	16.896	0.000	0.969	0.970
.IMPULS3	0.458	0.151	3.034	0.002	0.458	0.459
.DEPRES1	0.335	0.027	12.189	0.000	0.335	0.336
.DEPRES2	0.576	0.037	15.480	0.000	0.576	0.577
.DEPRES3	0.565	0.037	15.398	0.000	0.565	0.566
.DEPRES4	0.469	0.032	14.457	0.000	0.469	0.470
eta1	0.509	0.054	9.395	0.000	1.000	1.000
eta2	0.098	0.038	2.593	0.010	1.000	1.000
eta3	0.663	0.058	11.496	0.000	1.000	1.000

Il modello finale è composto da  $q = 3$  variabili latenti e  $p = 12$  variabili manifeste. L'adattamento globale, sia in termini di RMSEA (0.061) sia in termini di CFI (0.96) è accettabile (i valori ottenuti sono prossimi a quelli soglia usati solitamente). Il modello presenta un rapporto tra numero di parametri totali e numero di parametri liberi pari a  $\frac{m}{(p(p+1)/2)}=0.372$  non alto, il che evidenzia una certa parsimonia nella definizione della dimensionalità del test. Rispetto alla matrice stimata dei coefficienti fattoriali  $\hat{\mathbf{A}}$ , notiamo una certa adeguatezza nella struttura fattoriale: le scale relative a  $\eta_1$  o  $\eta_3$  complessivamente riportano coefficienti fattoriali alti o medio-alti; non così invece per quanto riguarda la scala  $\eta_2$  che, ad eccezione



del coefficiente  $\lambda_{82}$ , presenta coefficienti fattoriali medio-bassi o bassi. Una successiva valutazione del modello, difatti, potrebbe includere una rivalutazione - se non proprio eliminazione - dell'item **IMPULS2** che presenta basso coefficiente fattoriale e alta varianza residua. Rispetto alla matrice stimata  $\hat{\Phi}$ , notiamo che le variabili latenti  $\eta_1$  e  $\eta_2$  sono poco correlate. Allo stesso modo lo sono le variabili  $\eta_2$  e  $\eta_3$ . Al contrario,  $\eta_1$  e  $\eta_3$  presentano una rilevante correlazione (si veda ad esempio il valore della statistica  $Z$  associata), fatto questo che conferma quanto ottenuto dall'analisi esplorativa al punto 3. Questo risultato evidenzia la natura "non unidimensionale" del test oggetto di analisi. Infine, la presenza di due cross-loading sottolinea la natura "non semplice" della sua struttura fattoriale.

8. Si calcoli mediante un opportuno indice l'attendibilità delle scale derivanti dal modello finale scelto al punto 6 e se ne interpreti il risultato.

```
semTools::reliability(model4_fit)[c("alpha", "omega"),]

              eta1      eta2      eta3
alpha 0.8606278 0.3489655 0.8725501
omega 0.8180724 0.3902346 0.8440871
```

Un indice idoneo per valutare l'attendibilità delle scale  $\eta_1$ - $\eta_3$  secondo il principio della coerenza interna è l'indice  $\omega \in [0, 1]$ . In questo caso, la funzione `reliability()` della libreria `semTools`<sup>3</sup> restituisce in output diversi indici di attendibilità, tra cui diverse versioni (corrette) dell'indice  $\omega$  (**omega**). I risultati suggeriscono che le scale  $\eta_1$  e  $\eta_3$  presentino buoni indici di attendibilità/precisione. Come atteso, la precisione della scala  $\eta_2$  è medio-bassa e ciò suggerisce come gli item utilizzati non siano completamente adeguati per quantificare il costrutto **IMPULS**.

9. Si consideri la sola scala **SELF** del dataset **Y1** e si valuti, mediante opportuni modelli CFA, se essa sia un modello di misura parallelo,  $\tau$ -equivalente o congenerico.

Per valutare una particolare ipotesi di misura mediante la definizione e l'adattamento di un modello CFA ai dati si può procedere utilizzando sottomodelli CFA di tipo annidato, ponendo di volta in volta maggiori vincoli sui parametri del modello. I sottomodelli di misura sono i seguenti:

$$\begin{aligned}\Sigma_{y_{p \times p}} &= \lambda \mathbf{1}_p \mathbf{1}_p^T \phi + \mathbf{I} \theta_\delta & (\text{modello parallelo}) \\ \Sigma_{y_{p \times p}} &= \lambda \mathbf{1}_p \mathbf{1}_p^T \phi + \Theta_{\delta_{p \times p}} & (\text{modello } \tau\text{-equivalente}) \\ \Sigma_{y_{p \times p}} &= \lambda_p \lambda_p^T \phi + \Theta_{\delta_{p \times p}} & (\text{modello congenerico})\end{aligned}$$

Si noti come il modello congenerico presenti meno vincoli rispetto a quello parallelo. La stima dei parametri di tali modelli può essere fatta come per qualsiasi modello CFA mediante la funzione `cfa(...std.lv=TRUE)` della libreria `lavaan` come di seguito riportato.

```
#modello parallelo
model5 = "eta~1*SELF1+1*SELF2+1*SELF3+1*SELF4+1*SELF5
SELF1~~t*SELF1
SELF2~~t*SELF2
SELF3~~t*SELF3
SELF4~~t*SELF4
SELF5~~t*SELF5"
model5_fit = lavaan::cfa(model = model5, data = Y1, std.lv=TRUE)

#modello tau-equivalente
```

<sup>3</sup> La stessa funzione è disponibile senza installare la libreria `semtools` nella cartella "Utilities" nella pagina Moodle del corso.

```

model6 = "eta=~1*SELF1+1*SELF2+1*SELF3+1*SELF4+1*SELF5
          SELF1~~t1t1*SELF1
          SELF2~~t2t2*SELF2
          SELF3~~t3t3*SELF3
          SELF4~~t4t4*SELF4
          SELF5~~t5t5*SELF5"
model6_fit = lavaan::cfa(model = model6,data = Y1,std.lv=TRUE)

#modello congenerico
model7 = "eta=~11*SELF1+12*SELF2+13*SELF3+14*SELF4+15*SELF5
          SELF1~~t1t1*SELF1
          SELF2~~t2t2*SELF2
          SELF3~~t3t3*SELF3
          SELF4~~t4t4*SELF4
          SELF5~~t5t5*SELF5"
model7_fit = lavaan::cfa(model = model7,data = Y1,std.lv=TRUE)

cfa.fits = matrix(data = NA,nrow = 3,6)
colnames(cfa.fits)=c("Chisq","df","pvalue","RMSEA","CFI","AIC")
rownames(cfa.fits)=c("parallelo","tau-equiv","congenerico")

cfa.fits[1,] = fitMeasures(model5_fit,
                           fit.measures = c("Chisq","df","pvalue","RMSEA","CFI","AIC"))
cfa.fits[2,] = fitMeasures(model6_fit,
                           fit.measures = c("Chisq","df","pvalue","RMSEA","CFI","AIC"))
cfa.fits[3,] = fitMeasures(model7_fit,
                           fit.measures = c("Chisq","df","pvalue","RMSEA","CFI","AIC"))

cfa.fits = round(cfa.fits,3)

print(cfa.fits)

```

	Chisq	df	pvalue	RMSEA	CFI	AIC
parallelo	78.807	13	0	0.092	0.949	7282.259
tau-equiv	60.591	9	0	0.098	0.960	7272.043
congenerico	44.467	5	0	0.115	0.970	7263.919

Per i tre sottomodelli di misura, data la numerosità campionaria elevata, il valore della statistica del  $\chi^2$  sotto l'ipotesi nulla  $\mathcal{H}_0 : \Sigma_y - \hat{\Sigma}_y = \mathbf{0}$  non può essere utilizzato (il test tende ad essere quasi sempre rigettato). Per tale motivo, confrontiamo i tre modelli sulla base degli indici RMSEA e CFI. Rispetto all'indice CFI, i modelli  $\tau$ -equivalente e congenerico sembrano essere migliori. Al contrario, rispetto all'indice RMSEA, i modelli parallelo e  $\tau$ -equivalente sarebbero da preferire. L'indice AIC suggerisce la selezione del modello congenerico. In generale, tutti e tre i modelli presentano indici RMSEA superiori al valore soglia di riferimento (pari a 0.05). Ciò è anche in relazione al fatto che sia il modello parallelo sia quello  $\tau$ -equivalente presentano una struttura fattoriale poco flessibile all'adattamento ai dati rispetto al modello congenerico. Tuttavia quest'ultimo presenta un indice di RMSEA elevato rispetto agli altri due. Sulla base dei dati a disposizione, non possiamo affermare che la scala SELF sia un modello di misura parallelo o  $\tau$ -equivalente. La scelta ricade sul modello congenerico che rappresenta peraltro il modello base della CFA.

10. Per ogni sottomodello calcolato al punto 9, si calcoli la statistica  $s = \|\Sigma_y - \hat{\Sigma}_y\|^2$  che rappresenta una distanza tra la matrice delle covarianze osservata  $\Sigma_y$  e quella riprodotta dal modello  $\hat{\Sigma}_y$ .<sup>4</sup> Si interpreti

<sup>4</sup> In generale,  $\|\mathbf{X}\|$  con  $\mathbf{X}_{p \times p}$  matrice quadrata può essere calcolata in R mediante la funzione `norm()`.

il risultato anche alla luce di quanto ottenuto al punto 9.

```
covY_self = cov(Y1)[1:5,1:5]
s1=norm(covY_self - fitted(model5_fit)$cov)^2
s2=norm(covY_self - fitted(model6_fit)$cov)^2
s3=norm(covY_self - fitted(model7_fit)$cov)^2
print(c(s1,s2,s3))

[1] 0.06519885 0.09704275 0.02394905
```

Il risultato evidenzia che il modello congenerico, rispetto ai primi due, riproduce una matrice di covarianza  $\hat{\Sigma}_y$  che approssima meglio quella osservata  $\Sigma_y$ . Ciò è atteso, poiché quest'ultimo modello presenta meno vincoli nello spazio parametrico e presenta una maggiore flessibilità nell'adattarsi ai dati osservati.

```
covY_self

      SELF1      SELF2      SELF3      SELF4      SELF5
SELF1 1.0000000 0.4901806 0.5775525 0.5209583 0.5278123
SELF2 0.4901806 1.0000000 0.5463488 0.5338077 0.4744345
SELF3 0.5775525 0.5463488 1.0000000 0.5987247 0.5605786
SELF4 0.5209583 0.5338077 0.5987247 1.0000000 0.6953422
SELF5 0.5278123 0.4744345 0.5605786 0.6953422 1.0000000

fitted(model5_fit)$cov

      SELF1 SELF2 SELF3 SELF4 SELF5
SELF1 0.998
SELF2 0.552 0.998
SELF3 0.552 0.552 0.998
SELF4 0.552 0.552 0.552 0.998
SELF5 0.552 0.552 0.552 0.552 0.998

fitted(model6_fit)$cov

      SELF1 SELF2 SELF3 SELF4 SELF5
SELF1 1.063
SELF2 0.562 1.099
SELF3 0.562 0.562 0.977
SELF4 0.562 0.562 0.562 0.928
SELF5 0.562 0.562 0.562 0.562 0.969

fitted(model7_fit)$cov

      SELF1 SELF2 SELF3 SELF4 SELF5
SELF1 0.998
SELF2 0.457 0.998
SELF3 0.522 0.502 0.998
SELF4 0.564 0.543 0.620 0.998
SELF5 0.540 0.520 0.593 0.641 0.998
```