Juillet 2003 **Numéro 32**

Cahiers de l'IMA

Comment mesurer la similarité entre deux structures factorielles latentes

Jean-Philippe Antonietti

Institut de Mathématiques Appliquées
Faculté des S.S.P.
Université de Lausanne
BFSH 2
1015 Lausanne

Comment mesurer la similarité entre deux structures factorielles latentes

Jean-Philippe Antonietti

1 Introduction

En psychologie, il est courant d'avoir à comparer différentes structures factorielles construites à partir des réponses fournies à un même questionnaire par des individus issus de populations distinctes.
L'un des indices statistiques le plus fréquemment utilisé pour comparer de telles structures factorielles est le *coefficient de congruence global*. Ce coefficient permet de mesurer la similarité entre deux structures factorielles obtenues à partir de deux échantillons indépendants [2].
La formule qui permet de calculer ce coefficient est la suivante :

$$r_c = \frac{\sum_{j=1}^{k}\sum_{i=1}^{p} F1_{ij} \cdot F2_{ij}}{\sqrt{\sum_{j=1}^{k}\sum_{i=1}^{p} F1_{ij}^2} \cdot \sqrt{\sum_{j=1}^{k}\sum_{i=1}^{p} F2_{ij}^2}}$$

où :

p	est le nombre de variables mesurées dans chacun des échantillons;
k	est le nombre de facteurs de chacune des structures comparées;
$F1_{ij}$	est la saturation entre la variable i et le facteur j dans la première structure factorielle $F1$;
$F2_{ij}$	est la saturation entre la variable i et le facteur j dans la seconde structure factorielle $F2$.

Plusieurs procédures ont déjà été proposées pour évaluer la significativité du coefficient de congruence [1, 3, 5, 7, 8, 10] mais aucune ne possède la simplicité, ni la transparence, de celles que nous allons décrire dans les pages qui suivent.
Nous décrirons deux tests. Le premier test devrait permettre de savoir si l'échantillon de structure factorielle empirique $F1$ pourrait avoir été tiré d'une population caractérisée par la structure factorielle $F0$ (§ 2). Le deuxième test, quant à lui, devrait permettre de savoir s'il est vraisemblable que les structures factorielles empiriques $F1$ et $F2$, qui semblent être identiques à $F0$, soient issues de la même population caractérisée par $F0$ (§ 3).

Ces deux tests sont généralement applicables pour autant que l'on puisse supposer que les variables se distribuent toutes normalement. Pour traiter les situations qui ne satisfont pas ces contraintes, nous proposerons un moyen alternatif de comparer différentes structures factorielles (§ 4). Ce moyen consiste à construire des intervalles de confiance par bootstrap à partir de la matrice des observations.
Les routines que nous allons décrire ont été implémentées dans le logiciel statistique R. Pour pouvoir les utiliser, il suffit de charger le fichier `procuste.R` que l'auteur met à disposition de tout lecteur intéressé [1] :

```
> source("procuste.R")
```

1. Le module `procuste.R` est téléchargeable à partir de la page <http://www-ssp.unil.ch/IMA/antonietti/Jean-PhilippeAntonietti.html>. Ce module utilise les librairies `mva` et `mvtnorm` disponibles sur le site officiel du logiciel R dont l'adresse est <http://www.cran.r-project.org/>.

2 La structure factorielle $F1$ peut-elle être obtenue à partir d'un échantillon tiré de la population caractérisée par $F0$?

Donnons un exemple. Trois cents sujets ont été observés selon 15 variables. Une analyse en composantes principales de la matrice des corrélations a été effectuée. Seules les composantes ayant une variance supérieure à un ont été retenues. On leur a alors fait subir une rotation varimax. Voici l'allure de la structure factorielle $F1$ obtenue par ce procédé :

```
> F1
        fac1   fac2   fac3
var01 -0.070  0.704  0.040
var02 -0.114 -0.019 -0.673
var03  0.086  0.057 -0.688
var04 -0.019 -0.033 -0.661
var05 -0.054 -0.003 -0.680
var06  0.033 -0.019 -0.788
var07 -0.729  0.025 -0.063
var08 -0.748  0.072  0.020
var09 -0.689 -0.031  0.088
var10 -0.665 -0.028 -0.053
var11 -0.732 -0.040 -0.067
var12  0.002  0.716  0.061
var13  0.036  0.701 -0.109
var14  0.031  0.751  0.056
var15  0.006  0.646 -0.025
```

Cette structure factorielle aurait-elle pu être obtenue à partir de l'observation d'un échantillon de taille $n = 300$ tiré d'une population caractérisée par la structure $F0$ suivante ?

```
> F0
        fac1   fac2   fac3
var01  0.600  0.000  0.000
var02  0.600  0.000  0.000
var03  0.600  0.000  0.000
var04  0.600  0.000  0.000
var05  0.600  0.000  0.000
var06  0.000  0.600  0.000
var07  0.000  0.600  0.000
var08  0.000  0.600  0.000
var09  0.000  0.600  0.000
var10  0.000  0.600  0.000
var11  0.000  0.000  0.600
var12  0.000  0.000  0.600
var13  0.000  0.000  0.600
var14  0.000  0.000  0.600
var15  0.000  0.000  0.600
```

Pour répondre à cette question, nous allons générer un millier d'échantillons à partir de la population théorique caractérisée par $F0$, puis calculer pour chacun de ces échantillons le coefficient de congruence entre sa structure factorielle, ajustée à $F0$ par rotation procustéenne, et $F0$.
Nous pourrons ainsi esquisser la distribution des coefficients de congruence entre la structure factorielle théorique $F0$ et la structure factorielle d'un échantillon de taille $n = 300$ tiré d'une population ayant comme structure $F0$ et voir où se positionne, dans cette distribution, le coefficient de congruence entre $F1$, ajustée à $F0$ par rotation procustéenne, et $F0$.
Décrivons cette procédure plus en détail ! Si la structure factorielle théorique est définie par $F0$, alors la matrice des corrélations Σ est la matrice $F0\,(F0)^t$ dans laquelle les éléments de la diagonale ont été remplacés par 1 :

```
> Sigma <- F0 %*% t(F0)
> diag(Sigma) <- rep(1, dim(Sigma)[1])
> Sigma
```

	var01	var02	var03	var04	var05	var06	var07	var08	var09	var10	var11	var12	var13	var14	var15
var01	1.00	0.36	0.36	0.36	0.36	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
var02	0.36	1.00	0.36	0.36	0.36	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
var03	0.36	0.36	1.00	0.36	0.36	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
var04	0.36	0.36	0.36	1.00	0.36	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
var05	0.36	0.36	0.36	0.36	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
var06	0.00	0.00	0.00	0.00	0.00	1.00	0.36	0.36	0.36	0.36	0.00	0.00	0.00	0.00	0.00
var07	0.00	0.00	0.00	0.00	0.00	0.36	1.00	0.36	0.36	0.36	0.00	0.00	0.00	0.00	0.00
var08	0.00	0.00	0.00	0.00	0.00	0.36	0.36	1.00	0.36	0.36	0.00	0.00	0.00	0.00	0.00
var09	0.00	0.00	0.00	0.00	0.00	0.36	0.36	0.36	1.00	0.36	0.00	0.00	0.00	0.00	0.00
var10	0.00	0.00	0.00	0.00	0.00	0.36	0.36	0.36	0.36	1.00	0.00	0.00	0.00	0.00	0.00
var11	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.36	0.36	0.36	0.36
var12	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.36	1.00	0.36	0.36	0.36
var13	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.36	0.36	1.00	0.36	0.36
var14	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.36	0.36	0.36	1.00	0.36
var15	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.36	0.36	0.36	0.36	1.00

C'est dans la population multinormale Ω_0 définie par la matrice Σ que nous extrairons nos 1000 échantillons. A titre d'exemple, tirons un premier échantillon de Ω_0 et calculons la matrice des corrélations :

```
> n <- 300
> X <- rmvnorm(n, Sigma)
> R <- cor(X)
> R
```

	var01	var02	var03	var04	var05	var06	var07	var08	var09	var10	var11	var12	var13	var14	var15
var01	1.00	0.42	0.41	0.41	0.47	-0.07	-0.05	0.04	0.04	-0.01	-0.01	0.12	-0.07	0.02	0.05
var02	0.42	1.00	0.42	0.46	0.46	-0.01	-0.04	-0.03	0.07	-0.11	-0.09	-0.03	-0.18	-0.10	-0.16
var03	0.41	0.42	1.00	0.41	0.37	-0.09	0.04	0.01	0.05	-0.02	-0.08	-0.02	-0.15	-0.07	-0.07
var04	0.41	0.46	0.41	1.00	0.40	-0.03	-0.03	0.08	0.09	-0.07	-0.08	-0.07	-0.15	-0.06	-0.05
var05	0.47	0.46	0.37	0.40	1.00	0.01	-0.04	0.00	0.05	-0.05	-0.10	0.01	-0.16	-0.02	-0.07
var06	-0.07	-0.01	-0.09	-0.03	0.01	1.00	0.30	0.29	0.30	0.27	0.01	0.00	-0.04	-0.04	0.00
var07	-0.05	-0.04	0.04	-0.03	-0.04	0.30	1.00	0.33	0.34	0.40	0.04	0.04	-0.12	-0.05	0.02
var08	0.04	-0.03	0.01	0.08	0.00	0.29	0.33	1.00	0.34	0.37	0.05	0.05	-0.08	-0.01	0.11
var09	0.04	0.07	0.05	0.09	0.05	0.30	0.34	0.34	1.00	0.30	-0.10	-0.06	-0.14	-0.10	-0.04
var10	-0.01	-0.11	-0.02	-0.07	-0.05	0.27	0.40	0.37	0.30	1.00	0.07	0.00	0.00	-0.06	0.12
var11	-0.01	-0.09	-0.08	-0.08	-0.10	0.01	0.04	0.05	-0.10	0.07	1.00	0.30	0.36	0.47	0.48
var12	0.12	-0.03	-0.02	-0.07	0.01	0.00	0.04	0.05	-0.06	0.00	0.30	1.00	0.33	0.33	0.35
var13	-0.07	-0.18	-0.15	-0.15	-0.16	-0.04	-0.12	-0.08	-0.14	0.00	0.36	0.33	1.00	0.45	0.37
var14	0.02	-0.10	-0.07	-0.06	-0.02	-0.04	-0.05	-0.01	-0.10	-0.06	0.47	0.33	0.45	1.00	0.46
var15	0.05	-0.16	-0.07	-0.05	-0.07	0.00	0.02	0.11	-0.04	0.12	0.48	0.35	0.37	0.46	1.00

Effectuons une analyse en composantes principles de cette matrice. Voici les saturations des trois premières composantes principales :

```
> acp <- princomp(X, cor=TRUE)
> F <- acp$loadings %*% diag(acp$sdev)[, 1:3]
> F
```

```
        fac1   fac2   fac3
var01 -0.506 -0.443  0.356
var02 -0.657 -0.328  0.190
var03 -0.580 -0.310  0.247
var04 -0.608 -0.306  0.261
var05 -0.595 -0.347  0.258
var06  0.009  0.443  0.419
var07 -0.014  0.492  0.511
var08 -0.022  0.390  0.586
var09 -0.198  0.464  0.462
var10  0.076  0.464  0.532
var11  0.512 -0.352  0.394
var12  0.345 -0.373  0.376
var13  0.579 -0.380  0.173
var14  0.499 -0.479  0.328
var15  0.507 -0.346  0.459
```

Avant de calculer la valeur du coefficient de congruence, ajustons cette structure factorielle empirique à la structure factorielle théorique $F0$. Nous effectuerons cet ajustement par rotation procustéenne [4, 11]. Voici le résultat de l'ajustement [2] :

```
> F.ajust <- procuste(F, F0)[[2]]
> F.ajust
```

```
        fac1   fac2   fac3
var01  0.753 -0.001  0.113
var02  0.745 -0.045 -0.136
var03  0.699  0.007 -0.067
var04  0.724  0.021 -0.080
var05  0.734 -0.009 -0.050
var06 -0.064  0.606 -0.025
var07 -0.035  0.708 -0.019
var08  0.052  0.699  0.072
var09  0.101  0.660 -0.147
var10 -0.081  0.702  0.065
var11 -0.060  0.047  0.732
var12  0.072  0.026  0.627
var13 -0.185 -0.141  0.675
var14 -0.012 -0.086  0.761
var15 -0.034  0.100  0.759
```

Le coefficient de congruence global entre la matrice cible et la matrice ajustée vaut alors $r_c = 0.986$.

```
> congruence(F0, F.ajust)
[1] 0.986
```

2. Soit A la matrice à ajuster. Soit B la matrice cible. La matrice ajustée $B^\star$ est égale à AT. T est une matrice orthonormée construite de manière telle que la somme des écarts entre B et $B^\star$ au carré soit minimale. La fonction `procuste(A, B)` fournit une liste de trois éléments. Le premier élément est T, le deuxième est $B^\star = AT$ et le troisième est $E = B - B^\star$.

Réitérons 999 fois ces opérations. La distribution des coefficients de congruence ainsi obtenus est représentée dans la Figure 1.

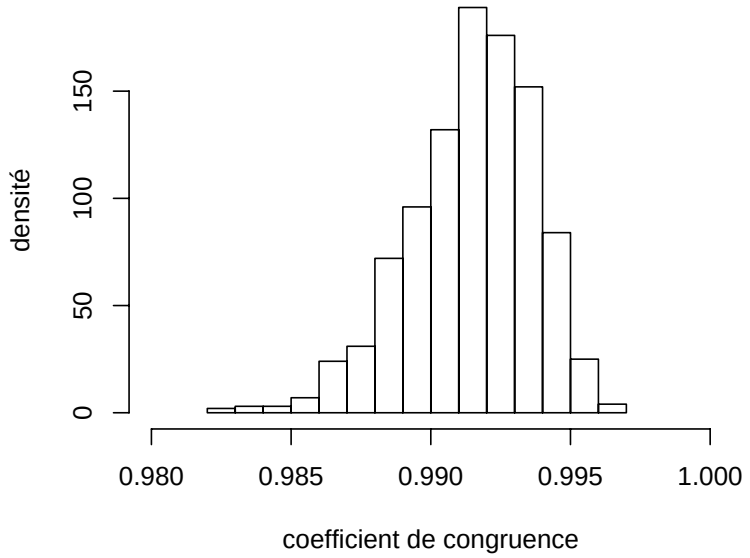

FIGURE 1 – *Distribution des coefficients de congruence calculés entre la structure factorielle théorique $F0$ et la structure factorielle d'échantillons issus d'une population définie par $F0$.*

Le quantile d'ordre $\alpha = 0.05$ vaut 0.988. Ce quantile représente la valeur critique du coefficient de congruence au-dessous de laquelle nous considérerons que l'échantillon de structure factorielle empirique $F1$ est issu d'une population ayant une structure différente de $F0$.
Calculons maintenant la valeur du coefficient de congruence entre la structure $F1$, ajustée à la structure $F0$ par rotation procustéenne, et la structure $F0$:

```
> F1.ajust <- procuste(F1, F0)[[2]]
> F1.ajust
```

```
        fac1   fac2   fac3
var01  0.051 -0.024  0.706
var02  0.645  0.207 -0.085
var03  0.695  0.003 -0.040
var04  0.643  0.114 -0.111
var05  0.661  0.147 -0.079
var06  0.776  0.079 -0.120
var07 -0.022  0.722  0.119
var08 -0.100  0.723  0.178
var09 -0.174  0.669  0.077
var10 -0.032  0.664  0.060
var11 -0.028  0.734  0.055
var12  0.040 -0.099  0.711
var13  0.209 -0.107  0.670
var14  0.054 -0.131  0.740
var15  0.116 -0.082  0.631
```

```
> congruence(F1.ajust, F0)
[1] 0.785
```

Comme le coefficient de congruence empirique $r_c^{emp} = 0.785$ est plus petit que le coefficient de congruence critique $r_c^{crit} = 0.988$, nous conclurons, au seuil de 5%, que la structure de la population dont est issu l'échantillon de structure factorielle $F1$ est différente de $F0$.
Pour effectuer dans R le test que nous venons de décrire, il suffit d'invoquer la fonction `congruence.test1`. Cette fonction possède cinq arguments : `F0`, `F1`, `n`, `alpha` et `nb.iters`.

`F0` représente la structure factorielle théorique.

`F1` représente la structure factorielle empirique.

`n` est la taille de l'échantillon ayant permis d'établir la structure factorielle $F1$.

`alpha` est le seuil de signification du test. Par défaut `alpha` est fixé à 0.05.

`nb.iters` est le nombre d'échantillons choisi pour estimer la distribution des coefficients de congruence sous H_0. Rappelons que selon H_0 la structure de la population dont est issu l'échantillon de structure factorielle $F1$ est $F0$. Par défaut `nb.iters` est fixé à 1000.

La fonction `congruence.test1(F0, F1, n, alpha, nb.iters)` fournit en sortie une liste de trois éléments. Le premier élément est la valeur critique du coefficient de congruence. Le deuxième est la valeur empirique et le troisième est la liste de tous les coefficients de congruence calculés pour esquisser la

distribution sous H_0.

```
> test <- congruence.test1(F0, F1, 300, 0.05, 1000)
> test$Valeur.critique
   5%
0.988
> test$Valeur.empirique
[1] 0.785
```

3 Les structures factorielles $F1$ et $F2$ peuvent-elles être obtenues à partir d'échantillons tirés de la même population caractérisée par $F0$?

Supposons que nous ayons interrogé les individus de deux échantillons indépendants, de tailles n_1 et n_2 respectivement, et que le résultat des analyses en composantes principales ait fourni les structures $F1$ et $F2$. Supposons également que nous ayons appliqué deux fois le test présenté au § 2 et que, dans les deux cas, nous n'ayons pas pu rejeter H_0. Il semblerait donc que les échantillons 1 et 2 soient tous deux issus de la même population caractérisée par la structure factorielle $F0$. Afin de mettre cette hypothèse plus sévèrement à l'épreuve, nous allons effectuer un nouveau test. Ce test consiste à construire la distribution des coefficients de congruence calculés entre les structures factorielles de deux échantillons indépendants de tailles n_1 et n_2 respectivement extraits de la population multinormale caractérisée par $F0$, puis à examiner si le coefficient de congruence global calculé entre $F1$ et $F2$ est compatible avec cette distribution. Comme précédemment le coefficient de congruence est calculé après ajustement des structures par rotation procustéenne.
Montrons, à l'aide d'un exemple, comment effectuer ces calculs dans R.
Prenons comme structure factorielle théorique la matrice suivante :

```
> F0
        fac1   fac2   fac3
var01  0.600  0.000  0.000
var02  0.600  0.000  0.000
var03  0.600  0.000  0.000
var04  0.600  0.000  0.000
var05  0.600  0.000  0.000
var06  0.000  0.600  0.000
var07  0.000  0.600  0.000
var08  0.000  0.600  0.000
var09  0.000  0.600  0.000
var10  0.000  0.600  0.000
var11  0.000  0.000  0.600
var12  0.000  0.000  0.600
var13  0.000  0.000  0.600
var14  0.000  0.000  0.600
var15  0.000  0.000  0.600
```

Supposons que les structures factorielles empiriques soient $F1$ et $F2$:

```
> F1
        fac1   fac2   fac3
var01 -0.026 -0.086 -0.592
var02 -0.057  0.035 -0.601
var03  0.028  0.004 -0.591
var04  0.017  0.032 -0.675
var05  0.063 -0.027 -0.659
var06 -0.095 -0.695 -0.042
var07  0.101 -0.752  0.068
var08 -0.019 -0.711 -0.060
var09  0.180 -0.698  0.058
var10  0.070 -0.678 -0.055
var11 -0.705  0.079 -0.036
var12 -0.749  0.017  0.009
var13 -0.722  0.123  0.019
var14 -0.702 -0.050  0.078
var15 -0.683  0.048 -0.044
> F2
        fac1   fac2   fac3
var01 -0.713  0.076 -0.050
var02 -0.652 -0.008  0.129
var03 -0.720 -0.051  0.005
var04 -0.771  0.005  0.004
var05 -0.678 -0.110 -0.018
var06  0.019  0.066  0.558
var07 -0.010 -0.023  0.513
var08 -0.063 -0.071  0.731
var09 -0.142 -0.114  0.568
var10  0.117  0.078  0.604
var11 -0.026 -0.714  0.016
var12 -0.051 -0.616  0.030
var13  0.018 -0.700 -0.033
var14 -0.066 -0.705 -0.005
var15  0.037 -0.687  0.036
```

La structure factorielle $F1$ a été construite à partir d'un échantillon de taille $n_1 = 300$. La structure $F2$ a été construite quant à elle à partir d'un échantillon de taille $n_2 = 250$.
Il est tout à fait possible que l'échantillon 1 soit issu de la population de structure $F0$ (seuil de signification $\alpha = 0.01$). En effet $(r_c^{emp} = 0.988) > (r_c^{crit} = 0.986)$:

```
> test01 <- congruence.test1(F0, F1, 300, 0.01, 1000)
> test01$Valeur.critique
   1%
0.986
> test01$Valeur.empirique
[1] 0.988
```

Il est vraisemblable que l'échantillon 2 soit, lui aussi, issu de la population de structure $F0$, car $(r_c^{emp} = 0.985) > (r_c^{crit} = 0.983)$:

```
> test02 <- congruence.test1(F0, F2, 250, 0.01, 1000)
> test02$Valeur.critique
   1%
0.983
> test02$Valeur.empirique
[1] 0.985
```

Afin d'éprouver plus drastiquement l'hypothèse selon laquelle les échantillons 1 et 2 sont issus de la même population, effectuons le test du coefficient de congruence qui porte sur deux échantillons indépendants. Pour ce faire, il suffit d'invoquer la fonction `congruence.test2`. Cette fonction possède sept arguments qui sont `F0`, `F1`, `F2`, `n1`, `n2`, `alpha` et `nb.iters`.

`F0` représente la structure factorielle théorique.

`F1` représente la première structure factorielle empirique.

`F2` représente la seconde structure factorielle empirique.

`n1` est la taille du premier échantillon.

`n2` est la taille du second échantillon.

`alpha` est le seuil de signification du test. Par défaut `alpha` est fixé à 0.05.

`nb.iters` est le nombre d'échantillons choisi pour estimer la distribution des coefficients de congruence calculé entre les structures factorielles de deux échantillons, de tailles n_1 et n_2 respectivement, tirés de manière indépendante de la population de structure factorielle $F0$. Par défaut `nb.iters` est fixé à 1000.

La fonction `congruence.test2(F0, F1, F2, n1, n2, alpha, nb.iters)` fournit une liste de trois éléments. Le premier élément est la valeur critique du test (rappelons que, sous H_0, la proportion des coefficients de congruence prenant une valeur inférieure à cette valeur critique vaut α). Le deuxième élément est la valeur empirique de la variable de décision qui, en l'occurrence, est le coefficient de congruence entre $F1$ et $F2$ calculé après leur ajustement par rotation procustéenne. Le troisième et dernier élément est la liste des `nb.iters` coefficients de congruence calculés pour estimer la distribution sous H_0.
Si la valeur empirique de la variable de décision est inférieure à la valeur critique, alors nous rejetons H_0, hypothèse selon laquelle les échantillons de structures $F1$ et $F2$ sont issus de la même population caractérisée par $F0$. Dans notre exemple $(r_c^{emp} = 0.962) < (r_c^{crit} = 0.970)$, nous rejetons donc H_0 et concluons, au seuil de 0.01, que les échantillons ne sont pas issus de la

même population caractérisée par $F0$.

```
> test12 <- congruence.test2(F0, F1, F2, 300, 250, 0.01)
> test12$Valeur.critique
   1%
0.970
> test12$Valeur.empirique
[1] 0.962
```

Les deux tests que nous venons de décrire supposent que les variables suivent dans la population une distribution multinormale. Afin d'outrepasser cette contrainte, nous allons proposer dans le paragraphe suivant une autre procédure applicable dans n'importe quelle situation.

4 Si l'expérience réalisée était réitérée, dans quel intervalle de confiance se trouverait le coefficient de congruence calculé entre les structures factorielles F et F' associées respectivement à l'expérience et à sa réplication ?

Nous adopterons ici un point de vue légèrement différent. Au lieu de prendre comme référence une structure théorique hypothétique $F0$, nous nous focaliserons plutôt sur la structure de la population dont est issu l'échantillon. A titre d'exemple, donnons-nous un échantillon de 300 sujets caractérisés selon 12 variables discrètes ayant comme modalités 1, 2, 3, 4, 5, 6, 7, 8 et 9. Nous supposerons que l'échantillon observé fournit une image relativement fidèle de la population dont il est issu [6]. Voici les caractéristiques des 10 premiers sujets :

```
> X[1:10, ]
     V01 V02 V03 V04 V05 V06 V07 V08 V09 V10 V11 V12
S001   5   3   2   4   4   3   6   3   4   6   4   5
S002   4   3   5   2   2   3   3   7   5   4   2   3
S003   6   7   5   9   9   9   7   6   5   5   7   6
S004   6   7   5   6   5   4   4   5   4   4   2   5
S005   6   4   6   8   8   8   6   6   4   2   6   3
S006   6   7   5   7   7   8   6   5   6   6   7   6
S007   6   6   4   4   6   4   4   7   8   4   4   4
S008   8   6   5   6   6   7   7   5   4   9   9   8
S009   3   5   6   4   5   4   7   6   7   5   3   4
S010   1   3   3   3   4   4   4   4   5   6   5   6
```

La structure factorielle de cet échantillon est la suivante :

```
> acp <- princomp(X, cor=TRUE)
> F <- acp$loadings %*% diag(acp$sdev) [, acp$sdev>1]
> F <- varimax(F)[[1]]
> F
      fac1   fac2   fac3   fac4
V01 -0.065 -0.037  0.841 -0.029
V02 -0.041  0.012  0.862  0.072
V03  0.028  0.069  0.829  0.048
V04 -0.876  0.013  0.015  0.066
V05 -0.889  0.068  0.041  0.075
V06 -0.878  0.065  0.026  0.028
V07 -0.040 -0.027  0.105  0.838
V08 -0.042 -0.102  0.052  0.861
V09 -0.085 -0.047 -0.067  0.876
V10  0.004  0.867  0.003 -0.079
V11 -0.059  0.857  0.003 -0.055
V12 -0.087  0.850  0.040 -0.037
```

Tirons au hasard de cette pseudo-population deux échantillons de même taille que l'échantillon d'origine. Le tirage s'effectue avec remise.

```
> X1 <- X[sample(1:300, size=300, replace=TRUE), ]
> X2 <- X[sample(1:300, size=300, replace=TRUE), ]
```

Déterminons la structure factorielle de chacun de ces échantillons :

```
> acp1 <- princomp(X1, cor=TRUE)
> F1 <- acp1$loadings %*% diag(acp1$sdev)
> F1 <- F1[, 1:4]
> acp2 <- princomp(X2, cor=TRUE)
> F2 <- acp2$loadings %*% diag(acp2$sdev)
> F2 <- F2[, 1:4]
```

Calculons enfin le coefficient de congruence entre ces deux structures :

```
> F1.ajust <- procuste(F1, F2)[[2]]
> r.c <- congruence(F1.ajust, F2)
> r.c
[1] 0.990
```

Réitérons mille fois ces opérations. Cela nous permet d'esquisser la distribution théorique de ces coefficients de congruence (voir Figure 2). Calculons le quantile d'ordre $\alpha = 0.01$ de cette distribution. Ce point, que nous nommerons r_c^{inf}, peut être utilisé pour apprécier la similarité entre la structure factorielle de notre échantillon F et une autre structure F'. Pour ce faire, il suffit de calculer le coefficient de congruence entre les structures F et F', ajustées par rotation procustéenne, et de voir si ce coefficient appartient à l'intervalle de confiance défini par r_c^{inf} et 1.

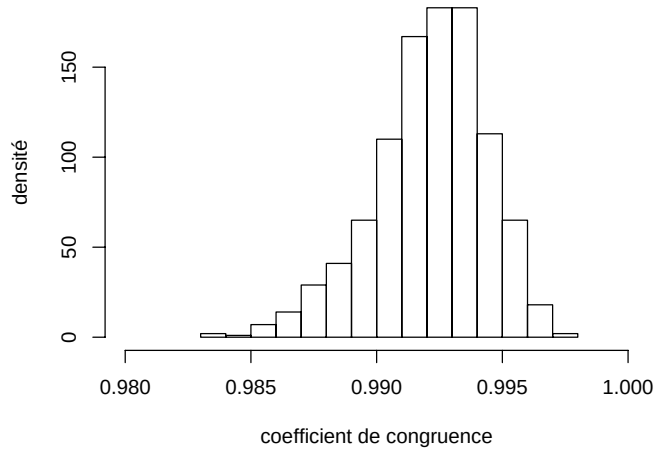

FIGURE 2 – *Distribution des coefficients de congruence calculés par bootstrap.*

Au cas où le coefficient de congruence entre F et F' appartient à l'intervalle de confiance, nous concluons que les structures F et F' sont similaires.
Ces calculs peuvent être aisément réalisés à l'aide de la fonction `congruence.IC`. Cette fonction possède quatre arguments : `X`, `nf`, `alpha` et `B`.

`X`	représente la matrice des observations. X est de dimension $n \times p$, où n est le nombre de sujets observés et p le nombre de variables mesurées.
`nf`	est le nombre de composantes principales retenues dans l'analyse.
`alpha`	est le complément à 1 du niveau de confiance des intervalles que l'on désire construire.
`B`	est le nombre d'échantillons bootstrap.

La fonction `congruence.IC(X, nf, alpha, B)` fournit la borne inférieure de l'intervalle de confiance pour le coefficient de congruence global; la borne supérieure vaut toujours 1. Mais cela n'est pas tout, cette fonction fournit également les bornes inférieures des intervalles de confiance pour les coefficients de congruence des facteurs et des variables [9].

```
> congruence.IC(X, 4, alpha=0.01, B=1000)
$IC.glob
   1%
0.986

$IC.var
 [1] 0.960 0.968 0.960 0.970 0.978 0.975 0.966 0.970
 [9] 0.976 0.973 0.970 0.968

$IC.fac
[1] 0.986 0.984 0.982 0.974
```

5 Conclusion

Dans la pratique, l'on utilise encore souvent une règle de cuisine pour comparer deux structures factorielles. Cette règle consiste à affirmer que les structures sont similaires si leur coefficient de congruence est supérieur à 0.9. Les trois procédures que nous venons de décrire devraient permettre de comparer deux structures factorielles de manière plus fondée !

Annexe : Les fonctions du module `procuste.R`

```
library(mva)
library(mvtnorm)
# La librairie mvtnorm n'est utilisable qu'à partir de
# la version 1.7.0 de R


procuste <- function(A, B){
  # Recherche des matrices T et E telles que A T = B + E
  # A représente souvent les résultats d'une analyse
  # antérieure
  S <- t(A) %*% B
  svd1 <- svd(S)
  u <- svd1$u
  v <- svd1$v
  B.star <- A %*% u %*% t(v)
  E <- B - B.star
  return(list(T=u %*% t(v), AT=B.star, E=E))
  }


congruence <- function(F1, F2){
  # F1 est la première structure factorielle
  # F2 est la seconde structure factorielle
  phi <- sum(F1 * F2)/ sqrt(sum(F1^2) * sum(F2^2))
  return(phi)
  }


congruence.coef <- function(F1, F2){
  congruence.glob <- sum(F1*F2)/
    sqrt(sum(F1^2) * sum(F2^2))
  congruence.var <- apply(F1*F2, 1, sum)/
    sqrt(apply(F1^2, 1, sum) * apply(F2^2, 1, sum))
  congruence.fac <- apply(F1*F2, 2, sum)/
    sqrt(apply(F1^2, 2, sum) * apply(F2^2, 2, sum))
  return(list(congruence.glob, congruence.var,
    congruence.fac))
  }
```

```
congruence.test1 <- function(F0, F1, n,
  alpha=0.05, nb.iters=1000){
  Sigma <- F0 %*% t(F0)
  diag(Sigma) <- rep(1, dim(Sigma)[1])
  p <- dim(F0)[2]
  # p représente le nombre de composantes principales
  distribution <- NULL
  for(i in 1:nb.iters){
    X <- rmvnorm(n, sigma=Sigma)
    acp <- princomp(X, cor=TRUE)
    F <- acp$loading %*% diag(acp$sdev)[, 1:p]
    F.ajust <- procuste(F, F0)[[2]]
    r.c <- congruence(F.ajust, F0)
    distribution <- c(distribution, r.c)
    }
  Valeur.critique <- quantile(distribution, alpha)
  F1.ajust <- procuste(F1, F0)[[2]]
  Valeur.empirique <- congruence(F1.ajust, F0)
  return(list(Valeur.critique=Valeur.critique,
    Valeur.empirique=Valeur.empirique, X=distribution))
  }


congruence.test2 <- function(F0, F1, F2, n1, n2,
  alpha=0.05, nb.iters=1000){
  Sigma <- F0 %*% t(F0)
  diag(Sigma) <- rep(1, dim(Sigma)[1])
  p <- dim(F0)[2]
  # p représente le nombre de composantes principales
  distribution <- NULL
  for(i in 1:nb.iters){
    x1 <- rmvnorm(n1, sigma=Sigma)
    acp1 <- princomp(x1, cor=TRUE)
    f1 <- acp1$loading %*% diag(acp1$sdev)[, 1:p]
    x2 <- rmvnorm(n2, sigma=Sigma)
    acp2 <- princomp(x2, cor=TRUE)
    f2 <- acp2$loading %*% diag(acp2$sdev)[, 1:p]
    f1.ajust <- procuste(f1, f2)[[2]]
    r.c <- congruence(f1.ajust, f2)
    distribution <- c(distribution, r.c)
    }
  Valeur.critique <- quantile(distribution, alpha)
  F1.ajust <- procuste(F1, F2)[[2]]
  Valeur.empirique <- congruence(F1.ajust, F2)
  return(list(Valeur.critique=Valeur.critique,
    Valeur.empirique=Valeur.empirique, X=distribution))
  }
```

```
congruence.IC <- function(X, nf, alpha=0.01, B=1000){
  n <- dim(X)[1]
  liste.g <- NULL
  liste.v <- NULL
  liste.f <- NULL
  for(i in 1:B){
    X1 <- X[sample(1:n, size=n, replace=TRUE),]
    X2 <- X[sample(1:n, size=n, replace=TRUE),]
    acp1 <- princomp(X1, cor=TRUE)
    F1 <- acp1$loadings %*% diag(acp1$sdev)[, 1:nf]
    acp2 <- princomp(X2, cor=TRUE)
    F2 <- acp2$loadings %*% diag(acp2$sdev)[, 1:nf]
    F1.ajust <- procuste(F1, F2)[[2]]
    r.c <- congruence.coef(F1.ajust, F2)
    liste.g <- c(liste.g, r.c[[1]])
    liste.v <- rbind(liste.v, r.c[[2]])
    liste.f <- rbind(liste.f, r.c[[3]])
    }
  IC.g <- quantile(liste.g, alpha)
  IC.v <- apply(liste.v, 2, quantile, alpha)
  IC.f <- apply(liste.f, 2, quantile, alpha)
  return(list(IC.glob=IC.g, IC.var=IC.v, IC.fac=IC.f))
  }
```

Bibliographie

[1] W. J. Broadbooks et P. B. Elmore, *A Monte Carlo study of the sampling distribution of the congruence coefficient*, Educational and Psychological Measurement, 47 (1987), pp. 1–11.

[2] C. Burt, *The factorial study of temperamental traits*, British Journal of Psychology, 1 (1948), pp. 178–203.

[3] W. Chan, R. M. Ho, K. Leung, D. K.-S. Chan, et Y.-F. Yung, *An alternative method for evaluating congruence coefficients with procrustes rotation : A bootstrap procedure*, Psychological Methods, 4 (1999), pp. 378–402.

[4] N. Cliff, *Orthogonal rotation to congruence*, Psychometrika, 31 (1966), pp. 33–42.

[5] E. C. Davenport, *Significance testing of congruence coefficients : A good idea ?*, Educational and Psychological Measurement, 50 (1990), pp. 289–296.

[6] B. Efron et R. Tibshirani, *An introduction to the bootstrap*, Chapman & Hall, London, 1993.

[7] B. Korth, *A significance test for congruence coefficients for Cattell's factors matched by scanning*, Multivariate Behavioral Research, 13 (1978), pp. 419–430.

[8] B. Korth et L. R. Tucker, *The distribution of chance congruence coefficients from simulated data*, Psychometrika, 40 (1975), pp. 361–372.

[9] R. R. McCrae, A. B. Zonderman, P. T. Costa, M. H. Bond, et S. V. Paunonen, *Evaluating replicability of factors in the revised NEO personality inventory : Confirmatory factor analysis versus procrustes rotation*, Journal of Personality and Social Psychology, 70 (1996), pp. 552–566.

[10] S. V. Paunonen, *On chance and factor congruence following orthogonal procrustes rotation*, Educational and Psychological Measurement, 57 (1997), pp. 33–59.

[11] P. H. Schönemann, *A generalized solution of the orthogonal procrustes problem*, Psychometrika, 31 (1966), pp. 1–10.

Table des matières