

Estimation de modèles COSAN sous l'environnement SAS: PROC CALIS

Eric LECOUTRE

Contents

1	Introduction	1
2	Comment entrer les données	3
2.1	Données brutes	3
2.2	Matrice de covariance	3
3	Syntaxe de la procédure CALIS	5
3.1	Syntaxe générale	5
3.2	Instruction LINEQS	5
3.3	Options de l'instruction PROC CALIS	6
3.4	Quelques instructions optionnelles dans la procédure	7
3.5	Contraintes sur les paramètres	7
4	Réécriture du modèle LISREL	7
5	Exemple d'application	8

1 Introduction

La procédure `CALIS` de SAS permet l'analyse de la covariance, l'ajustement de systèmes d'équations structurelles linéaires, ainsi que les analyses de *path diagram*. Tous ces termes sont plus ou moins interchangeable, bien qu'ils accentuent différents aspects de l'analyse. L'analyse de la covariance revient à formuler un modèle pour la structure d'une matrice de covariance d'un jeu de variables et à l'ajuster à une matrice observée. Dans un système d'équations structurelles linéaires, le modèle est formulé comme un ensemble d'équations entre différentes variables aléatoires, avec des hypothèses (restrictions) sur leurs variances et covariances. Dans les analyses de *path diagram*,

des flèches reliant des noeuds représentent les (co)variances et les coefficients de régression. Les modèles *path diagram* et les équations linéaires structurelles peuvent être réécrits comme modèles de la matrice de covariance et donc ajustés par les mêmes méthodes qu'en analyse de la covariance. Toutes ces méthodes ont en commun l'utilisation possible de variables *latentes* et d'erreurs de mesure dans les modèles.

Toutes ces approches peuvent être ramenées à un problème d'analyse de la covariance. La formulation du modèle induit des contraintes sur la structure de la matrice de covariance. On construit donc une matrice de covariance estimée sous la forme d'une fonctionnelle f dépendant des paramètres θ du modèle.

L'ajustement du modèle se fait donc par minimisation en θ d'une fonctionnelle mesurant une distance entre la matrice de covariance observée S et la matrice de covariance prédite $f(\theta)$. SAS propose 3 distances différentes :

– Moindres carrés non pondérés :

$$d = \frac{1}{2} \text{tr}(f(\theta) - S)^2$$

– Moindres carrés généralisée :

$$d = \frac{1}{2} \text{tr}(S^{-1}(f(\theta) - S))^2$$

– Maximum de vraisemblance :

$$d = \text{tr}(Sf(\theta)^{-1}) - n + \log(\det f(\theta)) - \log(\det S)$$

La matrice de covariance prédite $f(\theta)$ dépend des paramètres du modèle et fournit l'estimation de ces paramètres après minimisation. Le problème de minimisation sous-jacent est loin d'être trivial : bien souvent, le nombre de paramètres est important et le chercheur souhaite en plus imposer des contraintes (linéaires ou non) sur les paramètres. Même si SAS propose différents algorithmes d'optimisation non linéaires, on sait que cette optimisation peut impliquer différents problèmes numériques à prendre en compte (non convergence, minimum locaux, matrices singulières, etc).

La procédure CALIS fournit différents moyens de spécifier le modèle. Les équations structurelles sont retranscrites directement en utilisant une instruction LINEQS et en fournissant la liste des équations. Un *path diagram* sera adapté dans une instruction RAM en fournissant la liste des flèches le composant. Un modèle d'analyse factorielle de premier ordre sera spécifié au moyen des instructions FACTOR et MATRIX. Des modèles factoriels d'ordre plus grand et des modèles plus compliqués nécessiteront les instructions MATRIX et COSAN. Pour la majorité des applications, les instructions LINEQS et RAM sont suffisantes et s'avèrent les plus faciles.

2 Comment entrer les données

2.1 Données brutes

Généralement, on dispose d'un fichier comprenant les observations brutes. Il faut alors s'arranger pour créer un dataset dans SAS à partir de ces observations. On pourra les importer à l'aide de l'assistant SAS ou écrire le code pour l'exportation.

La saisie peut également être effectuée directement dans SAS, comme dans l'exemple ci-dessous où le dataset appelé `mydata` contient les 5 variables `pop`, `school`, `employ`, `services`, `house` dont les valeurs sont ensuite listées pour quelques individus.

```
DATA work.mydata;
INPUT pop school employ services house;
cards;

5700 12.8 2500 270 25000
1000 10.9 600 10 10000
3400 8.8 1000 10 9000
3800 13.6 1700 140 25000
4000 12.8 1600 140 25000
8200 8.3 2600 60 12000
1200 11.4 400 10 16000
9100 11.5 3300 60 14000
9900 12.5 3400 180 18000
9600 13.7 3600 390 25000
9600 9.6 3300 80 12000
9400 11.4 4000 100 13000
;
RUN;
```

2.2 Matrice de covariance

On peut entrer dans une matrice de covariance, une matrice de corrélation, une matrice de covariance "non corrigée par la moyenne" (moments du second ordre non centrés), une matrice de "corrélation non corrigée par la moyenne". On doit alors préciser `TYPE=COV`, `TYPE=CORR`, `TYPE=UCOV` ou `TYPE=UCORR` après le nom du fichier de données.

Exemple : si l'on veut entrer une matrice de covariance, on tape :

```

DATA work.covmat(TYPE=COV);
TITLE ''Stability of Alienation, Example in EQS and LISREL Guide'';
_TYPE_ = 'COV'; INPUT _NAME_ $ V1-V6;
LABEL V1='Anomia (1967)' V2='Anomia (1971)' V3='Education'
V4='Powerlessness (1967)' V5='Powerlessness (1971)'
V6='Occupational Status Index';

CARDS;
V1 11.834 . . . . .
V2 6.947 9.364 . . . . .
V3 6.819 5.091 12.532 . . . . .
V4 4.783 5.028 7.495 9.986 . . . . .
V5 -3.839 -3.889 -3.841 -3.625 9.610 . . . . .
V6 -21.899 -18.831 -21.748 -18.775 35.522 450.288
;
RUN;

```

Si l'on souhaite fournir en plus d'autres statistiques telles que moyenne et écart-type, on devra utiliser la variable système `_TYPE_` et la renseigner pour chaque ligne du dataset.

Exemple pour une matrice de covariance avec des moyennes :

```

DATA work.covmat(TYPE=COV);
TITLE ''Stability of Alienation, Example in EQS and LISREL Guide'';
INPUT _type_ $ _NAME_ $ V1-V6;
LABEL V1='Anomia (1967)' V2='Anomia (1971)' V3='Education'
V4='Powerlessness (1967)' V5='Powerlessness (1971)'
V6='Occupational Status Index';
CARDS;
mean . 1. 0.5 1. .6 .7 .8
cov V1 11.834 . . . . .
cov V2 6.947 9.364 . . . . .
cov V3 6.819 5.091 12.532 . . . . .
cov V4 4.783 5.028 7.495 9.986 . . . . .
cov V5 -3.839 -3.889 -3.841 -3.625 9.610 . . . . .
cov V6 -21.899 -18.831 -21.748 -18.775 35.522 450.288
;

```

Puis on appelle la procédure PROC CALIS avec les bons arguments :

```
PROC CALIS DATA=covmat ;
```

3 Syntaxe de la procédure CALIS

3.1 Syntaxe générale

La syntaxe de la procédure est la suivante :

```
PROC CALIS <options> ;  
[1] FACTOR analyse factorielle ;  
[2] RAM liste flèches ;  
[3] COSAN déclaration matrices ;  
[4] LINEQS liste équations ;  
    <STD> variances ;  
    <COVAR> covariances ;  
  
    <instructions optionnelles> ;  
RUN ;
```

Les instructions **PROC CALIS** et **RUN** sont obligatoires, ainsi qu'une des 4 instructions [1], [2], [3] et [4] (suivant le modèle considéré). L'instruction **PROC CALIS** définit le lancement de la procédure. C'est donc dans cette instruction que se placeront les options générales de la procédure, telles que le choix de la distance à minimiser, de l'algorithme d'optimisation ou encore des paramètres d'affichage (sorties).

3.2 Instruction LINEQS

L'instruction **LINEQS** permet de saisir le modèle sous la forme d'une liste d'équations linéaires. Les termes du côté gauche de l'équation sont des variables latentes ou manifestes. Une même variable ne peut pas apparaître des côtés gauche et droit d'une même équation.

La longueur du nom de chaque variable est limitée à 8 caractères. Les noms des variables manifestes sont déjà encodés dans le dataset utilisé. Les noms des variables latentes doivent commencer par la lettre F. Les noms des erreurs commencent par E pour les variables manifestes et D pour les variables latentes.

Chaque équation contient au plus une variable E ou D. Les équations sont séparées par une virgule et l'ordre est arbitraire.

Les coefficients à estimer sont indiqués dans les équations par un nom précédant le nom de la variable indépendante. Il peut être suivi par une valeur initiale fournie entre parenthèses. Si la variable indépendante est précédée par un nombre réel, alors il s'agit d'un terme constant.

Si le modèle contient beaucoup de paramètres, on peut les préciser tous ensemble à l'aide d'un préfixe : un nom court suivi de deux points. Un nom unique est alors automatiquement assigné à chaque paramètre en faisant suivre le préfixe par un suffixe entier.

Deux instructions importantes sont à utiliser conjointement à `LINEQS` :

- `STD` : pour renseigner les variances des variables libres ou imposées
- `COV` : pour renseigner les covariances des variables libres ou imposées

Chaque (co)variance du modèle non renseignée dans une de ces deux instructions est supposée égale à zéro.

Ces deux instructions sont aussi suivies d'une liste de déclarations séparées par des virgules. Les valeurs initiales des paramètres peuvent être indiqués entre parenthèses.

Exemple :

```
STD e1 = a1, e2 = a2(.5), e3 = 1 ;  
COV e1 e2 = co1, e2 e3 = co2(.5), e1 e3 = 1 ;
```

3.3 Options de l'instruction `PROC CALIS`

Sont données ici en vrac un tout petit nombre d'options de l'instruction `PROC CALIS`, qui en comporte plus d'une centaine. Pour plus de détails, se reporter au manuel de SAS/STAT (chapitre 19). A titre d'exemple, quelques options très utiles :

- `DATA=dataset` : nom du dataset comprenant les données à analyser (données brutes ou matrice de covariance).
- `COV` : permet l'analyse de la matrice de covariance à la place de la matrice de corrélation (par défaut). Ainsi, si on rentre en entrée une matrice de corrélation avec les variances des variables, la matrice de covariance sera calculée.
- `METHOD=` : Choix de la technique d'optimisation : GLS (moindres carrés), ML (maximum de vraisemblance - par défaut), ULS (moindres carrés non pondérés).
- `NODIAG` : Supprime la diagonale de l'analyse. Utile si on analyse une matrice de corrélation. Le nombre de degré de liberté est recalculé en conséquence.
- `NOBS=` : Nombre d'observations. Indispensables pour certains tests d'ajustement lorsque l'on fournit en entrée une matrice de corrélation ou de covariance.
- `RANDOM=` : Spécifie que les valeurs initiales seront aléatoires. Très utile pour vérifier la stabilité et l'unicité de la solution. La valeur fournie sert de graine (valeur de départ) au générateur pseudo-aléatoire.

3.4 Quelques instructions optionnelles dans la procédure

L'instruction `BY` permet d'obtenir autant d'analyses distinctes que de groupes définis par les variables discrètes `BY`.

Si une variable représente la fréquence d'apparition des autres valeurs dans les observations, on spécifie le nom de la variable dans une instruction `FREQ`. La procédure traitera alors les données comme si chaque observation apparaissait n fois où n est la valeur de la variable `FREQ`.

Si on souhaite faire une analyse sur une sous-matrice d'une matrice de covariance ou de corrélation, on utilise `PARTIAL` pour nommer les variables qui nous intéressent.

Enfin, on peut introduire une pondération en utilisant une variable dans l'instruction `WEIGHT`.

3.5 Contraintes sur les paramètres

Les instructions `BOUNDS`, `LINCON` et `NLINCON` permettent d'introduire des contraintes sur les paramètres de types respectifs : bornes, linéaires, non-linéaires. La syntaxe est toujours la même : l'instruction suivie d'une liste de contraintes séparées par des virgules.

Exemple :

```
BOUNDS
e1 < 1,
0. <= a1 <= 1.,
b1 y > 0;

LINCON x1 + 3*x2 <=1;
NLINCON x1*x1 + u1=10;
```

Attention : les contraintes nécessitent plus de temps de calcul et peuvent avoir des impacts difficiles à maîtriser sur les résultats. Il est déconseillé d'ajouter des contraintes.

4 Réécriture du modèle LISREL

Le modèle LISREL tel que défini par Jöreskog n'est pas tel quel implémenté dans la procédure CALIS. Le modèle LINEQS, toutefois, est très proche, et tout modèle LISREL peut être réécrit sous la forme d'un modèle LINEQS. Pour rappel, le modèle LISREL se compose de trois équations :

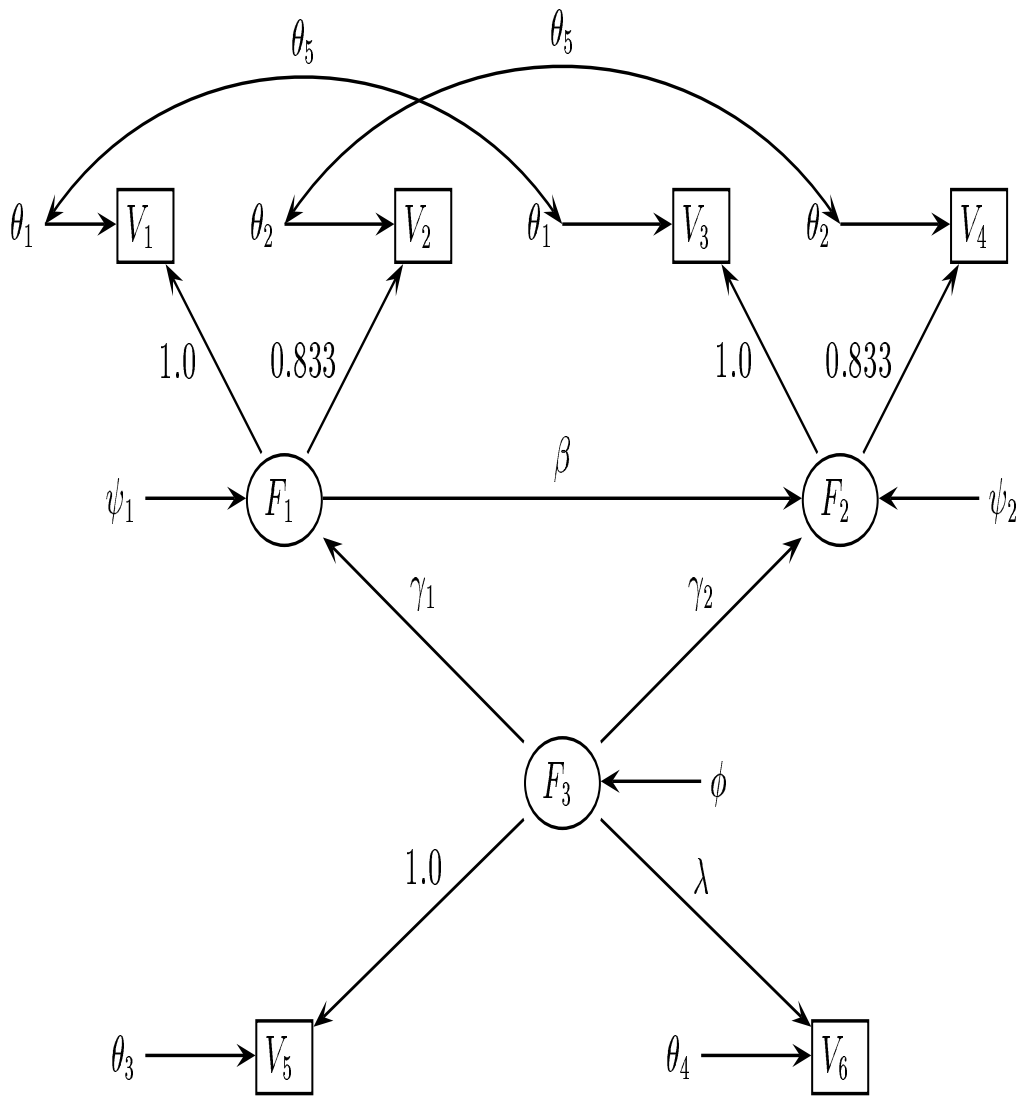
$$\begin{aligned}\eta &= B\eta + \Gamma\zeta + u \\ y &= \Lambda_y\eta + \epsilon_y \\ z &= \Lambda_z\zeta + \epsilon_z\end{aligned}$$

Pour estimer les paramètres au moyen du modèle LINEQS, on utilisera les vecteurs suivants :

$$\begin{aligned}(E) &= (\epsilon_y \epsilon_z) \\ (F_{\text{endogènes}}) &= (\eta) \\ (F_{\text{exogènes}}) &= (\zeta) \\ (D) &= (u)\end{aligned}$$

5 Exemple d'application

Soit le *path diagram* suivant, dont les notations sont déjà changées par rapport à celles d'un modèle LISREL en vue de l'implémentation dans une instruction LINEQS.



Ce diagramme montre donc les relations directes et indirectes entre les variables du modèle en utilisant des flèches pour indiquer la direction de la causalité souhaitée. Les coefficients de régression entre les variables sont indiqués par des flèches à une tête. Les variances et covariances sont indiquées par des flèches à deux têtes. Les flèches à deux têtes qui pointent sur des variables endogènes représentent les termes d'erreurs (par exemple : $Y_1 = 1.0F_1 + E_1, \theta_1 = \text{Var}E_1$).

Le programme SAS à utiliser pour évaluer le modèle est le suivant (avec l'instruction LINEQS) :

```
DATA CMAT(TYPE=COV);
    _type_ = 'cov'; input _name_ $ v1-v6;
    datalines;
v1  11.834      .      .      .      .      .
v2   6.947     9.364      .      .      .      .
v3   6.819     5.091    12.532      .      .      .
v4   4.783     5.028     7.495     9.986      .      .
v5  -3.839    -3.889    -3.841    -3.625     9.610      .
v6 -21.899   -18.831   -21.748   -18.775    35.522   450.288
;

/* On travaille sur la matrice de covariance, il y a 1000 observations */

PROC CALIS COV data=CMAT tech=nr nobs=1000 pall;
    LINEQS
        V1 = F1 + E1,
        V2 = .833 F1+ E2,
        V3 = F2 + E3,
        V4 = .833 F2+ E4,
        V5 = F3+ E5,
        V6 = Lamb (.5) F3+ E6,
        F1 = Gam1(-.5) F3+ D1,
        F2 = Beta (.5) F1 + Gam2(-.5) F3 + D2;
    STD
        E1-E6 = The1-The2 The1-The4 (6 * 3.),
        D1-D2 = Psi1-Psi2 (2 * 4.),
        F3 = Phi (6.) ;
    COV
        E1 E3 = The5 (.2),
        E4 E2 = The5 (.2);
RUN;
```