
L'ANALYSE DES CORRESPONDANCES MULTIPLES AVEC STATA : COMMENT OBTENIR LES MÊMES RÉSULTATS QU'AVEC R ET SAS

Marie Plessz, le 26 juin 2014

N'hésitez pas à me tenir informée: Marie.plessz@ivry.inra.fr

Documentation pour le programme `mcacontrib.ado`

Cette note s'appuie sur un article de blog de Jan Friedrik Hovden

<http://janfredrikhovden.wordpress.com/2011/09/22/multiple-correspondence-analysis-with-stata/>

Et sur la note en bas de page 108 du livre :

Le Roux et Rouanet *Multiple Correspondance Analysis*. Collection Quantitative Applications in the Social Sciences, 163, SAGE, Thousand Oaks, 2010

- **Contexte**

Stata fait des Analyses des correspondances multiples, mais les résultats obtenus sont plus pauvres et non concordants avec ceux des ouvrages de référence en France (Le Roux et Rouanet par exemple) ou d'autres logiciels (R avec le package FactoMineR par exemple).

Je pense que si le but est de produire des résultats détaillés et des graphiques avancés, il vaut mieux investir dans un peu d'exploration d'un autre logiciel, (justement R avec FactoMineR). Toutefois pour l'usage et la maîtrise que j'ai de R (toute relative) et de Stata (plus avancée), je trouve utile de réaliser des analyses exploratoires sous Stata, pour y effectuer tous les recodages et sélections de variables. Ensuite on peut si besoin, importer uniquement les données nécessaires sous R pour faire de beaux graphiques et avoir des résultats plus détaillés.

Encore faut-il obtenir de Stata des résultats concordants avec ceux des autres logiciels. Ici ma référence est R car c'est le seul que j'ai sous la main. Il faudrait comparer avec Sas et Spad. Coline Ferrant a pu refaire la même ACM sous SPAD 8 et a obtenu les mêmes coordonnées, contributions et \cos^2 que moi dans FactoMineR.

Voici comment procéder :

- Réaliser l'ACM avec les bonnes options

```
. webuse issp93
. mca A B C D, method (indicator) normal(princ)
```

On se rappelle qu'on peut ajouter des variables supplémentaires avec `sup(varlist)` et préciser le nb de dimensions à extraire avec `dim(#)`.

`Method(indicator)` permet de travailler sur le tableau disjonctif complet.

normal(princ) permet d'obtenir les coordonnées des modalités t telles qu'on les obtient avec R (coordonnées principales et non coordonnées standardisées) .

Reste un problème, les contributions partielles (contributions des modalités aux axes ne sont toujours pas bonnes.

- Recalculer les contributions

Le Roux et Rouanet nous expliquent dans leur note susmentionnée que « Stata appelle contribution » quelque chose qui ne correspond pas à leur propre définition. Pour retrouver leurs résultats il faut diviser ces contributions par la racine carrée de l'Eigenvalue de la dimension considérée.

C'est un peu fastidieux à faire à la main, alors le programme `mcacontrib.ado` le fait tout seul.

- **Comment utiliser le programme mcacontrib**

1. Sous stata, récupérer le chemin vers le dossier par défaut des fichiers .ado personnels : pour cela taper `sysdir` .
2. Si besoin créer le dossier désigné comme « personal » qui doit s'appeler **personal** à l'emplacement désigné.
3. Sauver `mcacontrib.ado` dans ce dossier.
4. Redémarrer Stata (penser à sauver datas et dofiles si besoin).
5. Puis réaliser l'ACM avec les bonnes options et lancer le programme

- Exemple

```
. webuse issp93
. mca A B C D, method (indicator) normal(princ) sup(varlist) dim(#)
. mcacontrib
```

Mcacontrib sauve les nouvelles contributions dans une matrice `NewContrib`. On peut utiliser cette matrice pour l'exporter sous excel, et pour l'afficher joliment il faut taper

```
. mcalist NewContrib
```

- Plus de graphiques !

Pour ceux qui veulent faire des graphes avancés sous Stata plutôt que sous R, la page de blog citée plus haut donne de nombreux exemples pour compléter les sorties d'ACM de Stata.

<http://janfredrikhovden.wordpress.com/2011/09/22/multiple-correspondence-analysis-with-stata/>

- **Annexe1 : analyse des contributions**

Pour interpréter une analyse factorielle il est dangereux de s'appuyer uniquement sur les coordonnées des points (par exemple par l'inspection visuelle du nuage de points des modalités). En effet une modalité rare peut avoir une coordonnée importante mais contribuer peu à définir l'axe (parce qu'elle est rare).

Les contributions peuvent s'interpréter de la façon suivante :

1. Les *modalités* dont la contribution sur une dimension dépassent la **contribution moyenne** contribuent significativement à l'interprétation de cet axe ;

2. Si pour une *variable*, aucune modalité n'a une contribution supérieure à la **moitié de la contribution moyenne**, sur les premières dimensions, alors on peut la considérer comme un « bruit blanc », qui n'apporte rien à l'analyse. On peut la retirer de l'ACM. Attention, il est utile de regarder les contributions sur un nb d'axes supérieur à celui qu'on gardera au final (ex, si on veut garder 2 axes, regarder les contributions sur 3 ou 4 dimensions). Il ne faudrait pas en effet déformer les dimensions non interprétées du nuage en retirant des variables qui y jouent un rôle important.

Calcul de la **contribution moyenne** :

C'est la moyenne des contributions sur un axe :

- Sous R et SPAD : $C_m = 100 / \text{nb de modalités actives}$
- Sous stata avec les nouvelles contributions calculées par mcacontrib :
 $C_m = 1 / \text{nb de modalités actives}$
- Sous stata si on n'a pas utilisé les nouvelles contributions :
 $C_m = \text{racine carrée}(\text{inertie principale}) / \text{nb de modalités actives}$

On peut retrouver facilement ce résultat en faisant la somme des éléments de la colonne « contribution » d'un axe et en divisant par le nb de lignes.

• Annexe 2 : le programme mcacontrib à sauver sous le nom mcacontrib.ado

```
*****
*      Programme mcacontrib
*      Marie Plessz le 24 juin 2014
/*      après une MCA, calcule la contribution des variables aux axes selon les
        formules de Le Roux et Rouanet (Multiple Correspondance analysis)
        et comme dans R, en divisant le résultat de stata par la racine carrée
        de la eigenvalue de la dimension correspondante.
*/
*      à utiliser après : mca [varlist], method (indicator) normal(princ)

*****
program mcacontrib
version 12

preserve
*      Récupérer les valeurs propres et calculer les racines carrées
        *les valeurs propres sont dans e(Ev).

matrix A=e(Ev)
matrix C=J(1,`e(f)',0)
forval j = 1/`=e(f)' {
    local colnames "`colnames' rcEv`j' "
}
// a créé de jolis noms pour les colonnes
matrix colnames C = "`colnames' "
forval j = 1/`=e(f)' {
    mat C[1,`j'] = sqrt(A[1,`j'])
}
svmat2 C, name(col)
// a sauvé les racines carrées des Ev dans des variables

*      récupérer les contributions stata
        //sauver les bonnes colonnes dans de nouvelles variables
forval j = 1/`=e(f)' {
    local k = (3*`j') +3
    matselrc e(cGS) Contr`j', col(`k')
```

```

        svmat2 Contr`j', rname(varname`j') names(oldc`j')
    }
        // dispensable : sauve aussi le nom des lignes dans une seule var pour
pouvoir vérifier
    local j=2
    while `j'<= `e(f)' {
        drop varname`j'
        local j=`j'+1
    }

* calculer les nouvelles contributions
    //calcul effectué sur les variables
quietly {
    forval j = 1/`= e(f)' {
        replace rcEv`j'=rcEv`j'[1]
        gen newct`j'=oldc`j' / rcEv`j'
        label var newct`j' "Contrib `j' nouveau calcul"
    }
}
* sauver le résultat dans une matrice NewContrib
mkmat newct* in 1/`=rowsof(e(cGS))', matrix(NewContrib) //rownames(varname1)

//donner les noms de lignes d'origine à la nouvelle matrice
local subs : rownames e(cGS)
local eqs : roweq e(cGS)
matrix rownames NewContrib = `subs'
matrix roweq NewContrib = `eqs'

//afficher le résultat
di " - Contributions calculées comme dans R - "
di "Rappel : utiliser les options method(indicator) normal(principal)"
matlist NewContrib
di _n "Résultats sauvés dans la matrice NewContrib"
* restaurer les données
restore
end
*****

```