

Tutoriel FactoMineR pour l'analyse des correspondances multiples avec une petite annexe sur la classification automatique

par Claire Lemerrier, avec Pauline Milani et Séverine Sofio
merci à Sébastien Dubois pour ses remarques, et à Muriel Cohen, Laure Fourtage et Alix Heiniger
pour les tests et les questions stimulantes...

version du 20 avril 2010

(dernières modifications = comment obtenir les intitulés que vous souhaitez, p. 8 ; comment lire
les résultats de la classification automatique, *in fine*)

Commentaires bienvenus : Claire.Lemerrier@ens.fr

Nota bene : il existe maintenant un manuel papier dédié à FactoMineR, qui explique à partir d'exemples à la fois les principes de R et de l'analyse factorielle. À recommander, donc. Voir <http://www.pur-editions.fr/detail.php?idOuv=2166> (où on peut lire préface et table des matières).

1. Préparer les données

- Pour les bonnes pratiques de codage en matière d'ACM, voir notre Repères et la bibliographie correspondante. Mais voici quand même un résumé outrancièrement rapide et peu justifié : 1. éviter de constituer des classes regroupant trop peu d'individus ; s'il y a des raisons substantielles de le faire, traiter la variable correspondante comme supplémentaire ; 2. quand on ne sait pas trop comment constituer des classes (d'âge par exemple), répartir à peu près également les effectifs entre ces classes et ne pas faire des classes trop nombreuses (pas plus de 4 ou 5 le plus souvent) ; s'il y a des raisons substantielles de faire autrement, traiter la variable correspondante comme supplémentaire ; 3. éviter d'avoir deux variables différentes qui disent la même chose ou presque la même chose (caricature : l'âge et l'année de naissance ; cas plus fréquent : niveau d'études et niveau professionnel dans les cas où ceux-ci sont très corrélés). Dans ce cas, en conserver une seule (considérer l'autre, éventuellement, comme supplémentaire) ou bien créer une troisième variable qui résume les deux variables d'origine.
- Si les données sont dans un fichier Excel, il faut que la première ligne donne les intitulés des colonnes. Ces intitulés peuvent être un peu longs et comprendre des espaces (cela ne bloque pas le logiciel) mais ce n'est pas recommandé, notamment pour la lisibilité des graphiques.
- Les données peuvent être codées sous forme de chiffres ou d'étiquettes textuelles (« femme », « fem »...). Éviter par principe les codes contenant des caractères « compliqués » (espaces, accents, tirets...) ou trop longs. L'underscore (_) est en revanche OK. Si les mêmes codes (« 1 », « autre »...) se retrouvent dans plusieurs colonnes, le logiciel ajoutera automatiquement l'intitulé de la variable devant celui de la modalité (« sexe.1 », « profession.autre »...).
- Il est bon de coder les données manquantes « NA » pour que R les reconnaisse comme telles, ce qui peut servir pour certains traitements. Mais la plupart du temps vous pouvez utiliser un autre code au choix.

Attention : si la première ligne (les données sur le premier individu) contient des « NA », il est possible que les colonnes correspondantes ne se chargent pas. Vérifiez toujours le bon chargement de vos données, et en cas de problème, triez les lignes autrement pour ne pas avoir de « NA » chez le premier individu.

Attention : si certains individus concentrent les valeurs manquantes, cela va peser sur les premiers axes de l'analyse. Si ce n'est pas ce qui vous intéresse substantiellement, il faut peut-être considérer ces individus comme supplémentaires.

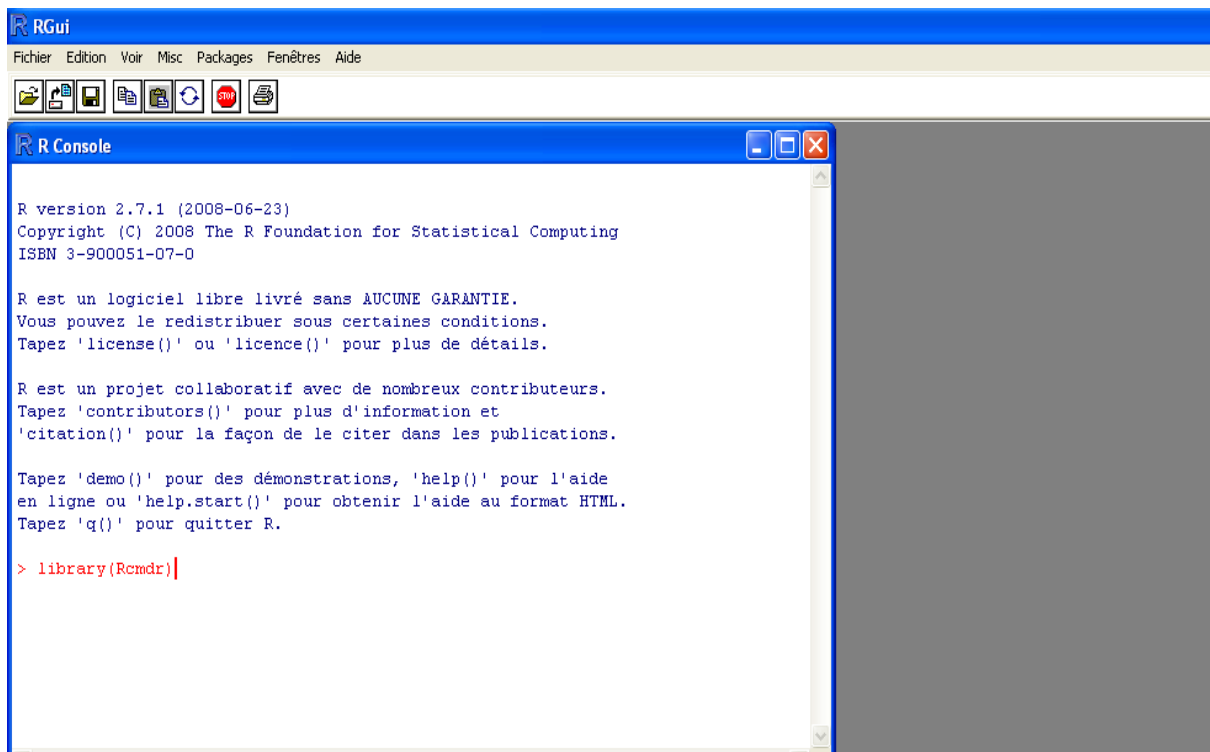
- Le logiciel utilisé ici va considérer les données chiffrées comme des données quantitatives (susceptibles de donner lieu à des calculs de moyennes par exemples) et les données textuelles comme des données qualitatives. Il est possible de corriger cela, si ce n'est pas adapté à vos données, dans RCommander. Cela dit, il est plus simple de coder en amont selon ces principes, donc de ne pas utiliser de codes purement chiffrés pour les données qualitatives. Exemple : mettre « femme » plutôt que « 2 », mettre « 1850_59 » plutôt que « 1850 » si cela représente la classe de dates « années 1850 »... Attention : si une même colonne comprend des chiffres et des lettres (par exemple en nombre d'enfants « 1 », « 2 » ou « plus »), cela va poser des problèmes. Mettez tout en chiffres ou (de préférence pour l'ACM) tout en lettres (« 1enf », « 2enf », « plusenf »).
- Il est possible d'utiliser les noms de vos individus, ou tout autre « label » un peu explicite, dans les graphiques d'analyse factorielle, résultats de classification, etc. (voir p. 8). Si vous pensez faire cela, mettez la colonne correspondante dans vos données... sans oublier de ne pas la considérer comme variable active dans l'analyse !

2. Installer les logiciels : voir <http://www.quantihmc.ens.fr/document.php?id=78> (*in fine*)

3. Lancer R, RCommander et FactoMineR

- Lancer R
- Taper : `library(Rcmdr)` et appuyer sur Entrée. À partir de là, on n'utilise plus la fenêtre R, mais seulement la fenêtre RCommander (avec menus déroulants). Cependant, *il faut* conserver ouverte la fenêtre R, où vont notamment s'afficher les graphiques.
- NB : quand RCommander travaille, il vous montre les instructions qu'il envoie à R (lignes de programmes). Cela peut être un moyen de commencer à apprendre R, ou d'affiner certaines instructions (*cf. infra*). Cela dit, vous n'êtes pas obligés de vous en préoccuper si c'est intimidant, surtout au début !

Instruction initiale tapée dans R :



```

RGui
Fichier  Edition  Voir  Misc  Packages  Fenêtres  Aide

R Console

R version 2.7.1 (2008-06-23)
Copyright (C) 2008 The R Foundation for Statistical Computing
ISBN 3-900051-07-0

R est un logiciel libre livré sans AUCUNE GARANTIE.
Vous pouvez le redistribuer sous certaines conditions.
Tapez 'license()' ou 'licence()' pour plus de détails.

R est un projet collaboratif avec de nombreux contributeurs.
Tapez 'contributors()' pour plus d'information et
'citation()' pour la façon de le citer dans les publications.

Tapez 'demo()' pour des démonstrations, 'help()' pour l'aide
en ligne ou 'help.start()' pour obtenir l'aide au format HTML.
Tapez 'q()' pour quitter R.

> library(Rcmdr)

```

4. Importer les données

- Menu Données → Importer des données depuis Excel → choix du nom (« Dataset » par défaut peut être conservé : ça n'est important que si vous prévoyez de travailler sur plusieurs bases de données différentes au cours d'une même session de travail avec FactomineR) → choix du document → choix de la feuille dans le classeur Excel (si certaines feuilles se présentent avec des noms cabalistiques commençant par \$, ne pas en tenir compte)
- Vérifier que l'importation s'est bien passée : cliquer sur Visualiser (bouton sous la ligne de menus) et jetez un coup d'oeil. À noter que le bouton « Editer » permet de modifier vos données directement sous RCommander, mais ça n'est pas forcément une bonne idée... mieux vaut souvent garder un fichier Excel « propre et à jour » à côté.

Importation des données (par le menu Données) : on voit les commandes s'afficher seules.

```
Dataset <- sqlQuery(channel = 1, select * from [LES MARIEES$])
names(Dataset) <- make.names(names(Dataset))
```

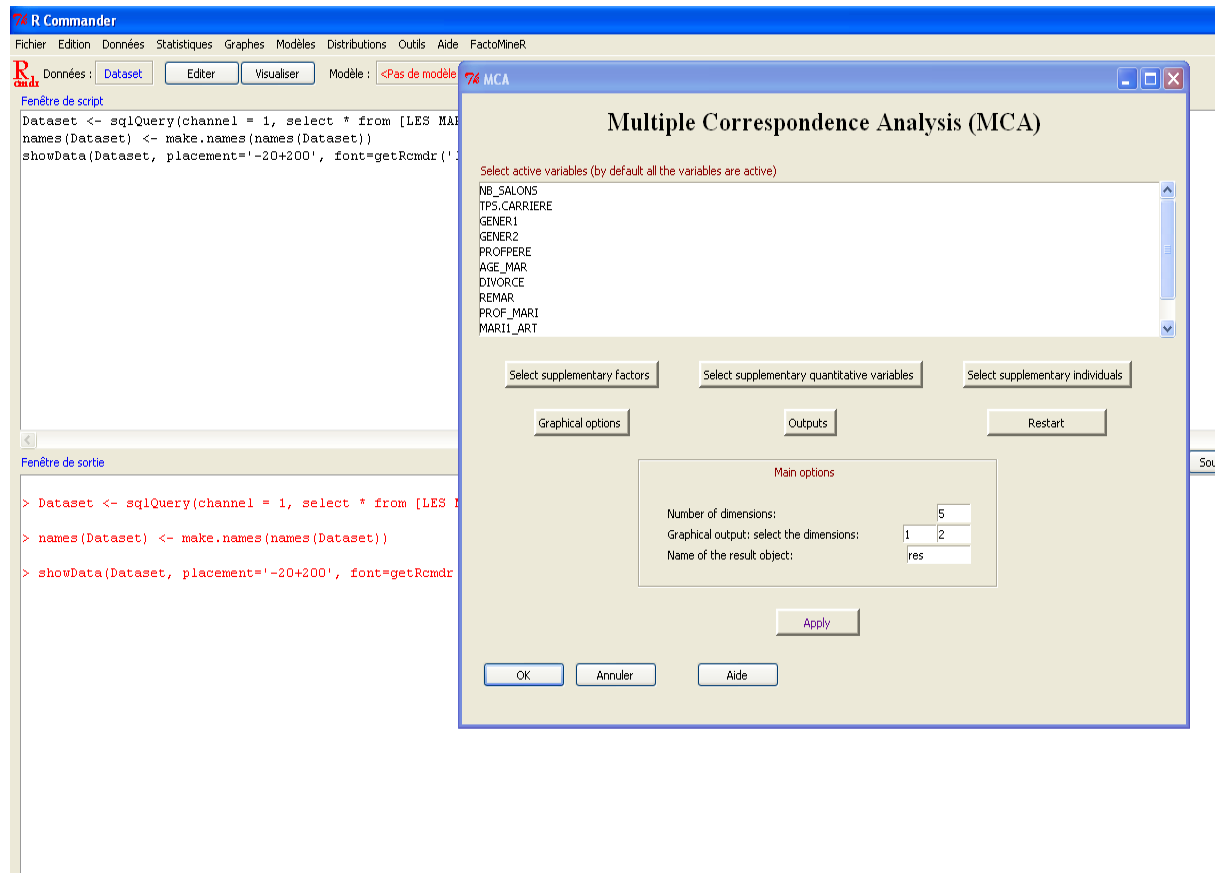
Vérification des données :

```
Dataset <- sqlQuery(channel = 1, select * from [LES MARIEES$])
names(Dataset) <- make.names(names(Dataset))
showData(Dataset, placement='-20+200', font=getRcmdr('logFont'), maxwidth=80, maxheight=30)
```

ID.des.582.artistes.mariées	NB_SALONS	TPS.CARRIERE	GENER1	GENER2	PROPRETE	AGE_M	
1	MOYEN	MOYEN	<NA>	D	<NA>	<N	
2	TRESPEU	TRESCOURT	<NA>	B	<NA>	<N	
3	TRESPEU	TRESCOURT	<NA>	B	<NA>	<N	
4	NBX	TRESLONG	D	D	AR1	<N	
5	TRESPEU	TRESCOURT	<NA>	B	<NA>	<N	
6	MOYEN	MOYENPLUS	B	B	HTF	<N	
7	NBX	TRESLONG	B	B	<NA>	<N	
8	TRESPEU	MOYENPLUS	B	B	<NA>	<N	
9	TRESPEU	TRESCOURT	<NA>	C	<NA>	<N	
10	TRESPEU	TRESCOURT	<NA>	C	<NA>	<N	
11	TRESPEU	MOYEN	<NA>	C	<NA>	<N	
12	TRESPEU	TRESCOURT	<NA>	D	<NA>	<N	
13	PEU	MOYEN	C	C	MLL	<N	
14	PEU	MOYENPLUS	<NA>	C	<NA>	<N	
15	TRESPEU	TRESCOURT	<NA>	D	<NA>	<N	
16	30	PEU	MOYEN	B	<NA>	<N	
17	31	PEU	MOYEN	C	<NA>	<N	
18	32	PEU	MOYEN	D	<NA>	<N	
19	36	TRESPEU	MOYENPLUS	D	<NA>	<N	
20	38	MOYEN	LONG	B	HTF	<N	
21	41	PEU	LONG	<NA>	C	<NA>	<N
22	44	TRESNBX	LONG	C	<NA>	<N	
23	45	PEU	MOYEN	C	AR1	<N	
24	46	TRESPEU	COURT	<NA>	C	<NA>	<N
25	47	TRESPEU	TRESCOURT	D	AR2	<N	
26	49	TRESPEU	TRESCOURT	D	<NA>	<N	
27	53	TRESPEU	COURT	C	<NA>	<N	
28	55	TRESPEU	TRESCOURT	B	<NA>	<N	
29	56	PEU	MOYENPLUS	B	<NA>	<N	
30	57	PEU	MOYENPLUS	D	<NA>	<N	

5. Faire une ACM

- Voir notre Repères pour les choix de variables actives et supplémentaires. Ici sont également proposés des « individus supplémentaires » (non inclus dans l'analyse mais dont on regarde le placement sur le plan obtenu) : cela peut être utile notamment pour ceux pour lesquels beaucoup de données sont manquantes.
- Menu FactomineR → choix Multiple Correspondence Analysis



- Dans la liste des variables qui s'affiche, sélectionner les variables actives (utiliser la sélection multiple : Ctrl+clic sur PC)
Que faire si mes variables ne s'affichent pas ici, ou pas toutes ?
C'est sans doute que RCommander prend certaines variables qualitatives pour des variables quantitatives (cf. supra).
Dans ce cas faire Annuler et transformer ces variables avant de revenir à l'ACM. Pour cela, dans le menu « Données »-> « Gérer les variables... », choisir « Convertir les données numériques en facteurs ». Sélectionner les variables à convertir en cochant à droite « utiliser les nombres » et dire « oui » à ce qui suit (« remplacer variable ? »). Puis revenir à l'ACM.
- Bouton « Select Supplementary Factors » pour sélectionner les variables supplémentaires (même procédure que pour les actives)
- Bouton « Select Supplementary Quantitative Variables » pour d'éventuelles variables quantitatives supplémentaires.
- Bouton « Select Supplementary Individuals » pour intégrer à l'analyse d'éventuels individus extérieurs au corpus de données.
- « Graphical options » : ne permet pas d'obtenir de très beaux graphiques, mais peut permettre de se faire une première idée. Décidez surtout si vous voulez afficher les variables actives, supplémentaires, et/ou les individus (selon le nombre de variables et

d'individus, pour ne pas trop surcharger le graphique). En général, vous pouvez décocher « Plot variable graph » dans la moitié droite de la boîte de dialogue (mais vous pouvez aussi le garder pour voir à quoi cela ressemble !).

- Bouton « Outputs » pour sélectionner le type de données que l'on veut transférer dans Excel (tout cocher par défaut) et l'endroit sur le disque où l'on veut les placer. Donner un nom au fichier de sortie ; faire attention à inclure dans ce nom l'extension .csv (par exemple « test.csv »). Ce fichier va avoir deux usages : permettre de compléter l'interprétation visuelle en regardant de près les coordonnées, contributions, etc. des variables et individus ; et de faire des graphiques plus jolis et plus facilement modifiables, surtout pour ceux qui ne sont pas des « pros » de R.

Attention : sur un de mes ordinateurs, si je ne précise rien, ce fichier est créé par défaut dans le répertoire « Mes documents » (et je le déplace ensuite où je veux). Sur un autre, une boîte de dialogue me demande où mettre le fichier...

Attention : ne pas cliquer sur le bouton « Restart » : cela annule tous les choix antérieurs...

- « Main Options » : garder les choix par défaut, ou, si l'on sait qu'on ne veut que les 2 ou 3 premiers axes, indiquer cela en face de « Number of dimensions ». Attention, si vous voulez faire ensuite une classification automatique (*cf. infra*), il peut être utile d'avoir plus d'axes, par exemple les 5 proposés par défaut.
- Cliquer sur « Apply » ou « OK »
- Attendre un peu si les données sont massives.
- Aller voir le graphique qui sort par défaut dans la fenêtre RGui (et pas R Commander) et constater qu'il n'est pas très joli ni très lisible... (*cf. page suivante*).

- À partir de là, trois solutions côté graphiques :

- **utiliser le sous-programme dynGraph** : cela impose de taper une ligne d'instructions, mais présente beaucoup d'avantages : des graphiques lisibles, et surtout une aide à l'interprétation des résultats qui peut épargner la lecture des sorties chiffrées. En réalité, les graphiques finaux obtenus ne sont pas forcément publiables à 100 %, mais ils sont très suffisants pour comprendre ses résultats, relancer une analyse un peu différente, etc. – quitte à passer éventuellement à l'option suivante pour faire un et un seul graphique définitif.

- les faire sous Excel, à partir du fichier test.csv (ou autre chose .csv) créé précédemment. Inconvénient : cela prend un peu de temps pour chaque graphique. Si on s'aperçoit d'une erreur (de codage par exemple), il faut reprendre tout le processus. Avantage : la simplicité plus grande de l'environnement Excel pour les utilisateurs débutants et les possibilités plus accessibles d'amélioration graphique fine.

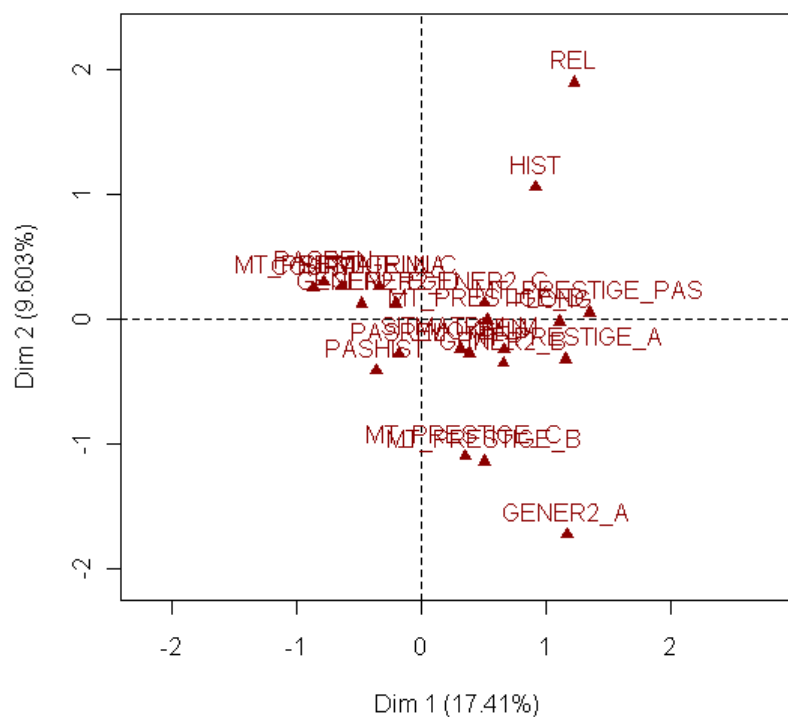
- améliorer les graphiques sous R. On peut faire très beau si on connaît les instructions, mais il est déjà facile de faire « plus lisible » que le graphique de base – le temps de vérifier qu'il n'y a pas d'erreur ou encore de tester différentes versions pour la répartition entre variables actives et supplémentaires. On peut ensuite choisir de ne faire que le graphique final sous Excel.

- Conclusion : si vous n'avez pas peur de taper une ligne de programme, l'option 5b et **surtout l'option 5c** (nouveau de 2009) ci-dessous peuvent vous faire gagner du temps. L'option 5c vous aidera aussi pour l'interprétation. Mais sinon, vous pouvez aussi passer au 6 !

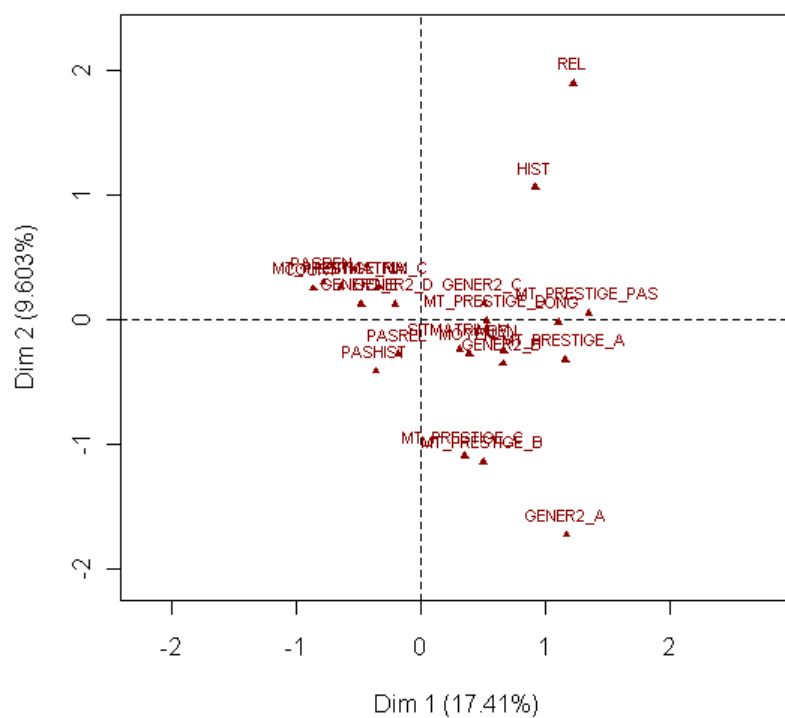
Dans tous les cas, allez jeter un coup d'oeil à la fin de ce tutoriel (p. 15-18) : ellipses et classifications offrent des façons complémentaires très intéressantes d'explorer vos données. Mais là aussi, il faudra copier-coller une ligne de programme !

5b. Améliorer les graphiques sous RCommander/FactoMineR

Le graphe de R avant...



Et après...



Pour obtenir ce résultat (pas encore hyper lisible, mais déjà beaucoup plus !), nous allons utiliser le fait que les instructions en langage R non seulement s'affichent dans la partie supérieure de l'écran de RCommander, mais peuvent y être modifiées. Retournons dans la fenêtre RCommander et regardons bravement ce qui apparaît en haut. Une des dernières instructions doit ressembler à ça :

```
plot.MCA(res, axes=c(1, 2), col.ind="black", col.ind.sup="blue",
col.var="darkred", col.quali.sup="darkgreen", label=c("ind.sup",
"quali.sup", "var", "quanti.sup"), invisible=c("ind"), title="")
```

« plot.MCA » est une instruction générique pour faire des graphiques à partir des résultats d'une ACM (ces résultats sont stockés dans un tableau que le logiciel a ici appelé « res » et qui a été créé par une des instructions précédentes). On peut y changer ou ajouter beaucoup d'options. Ici, nous vous proposons simplement de mettre des étiquettes un peu plus petites sur le graphique pour le rendre plus lisible.

Pour cela, ajouter dans la parenthèse de la première instruction ci-dessus, par exemple après la définition des axes : « cex=0.7 », ».

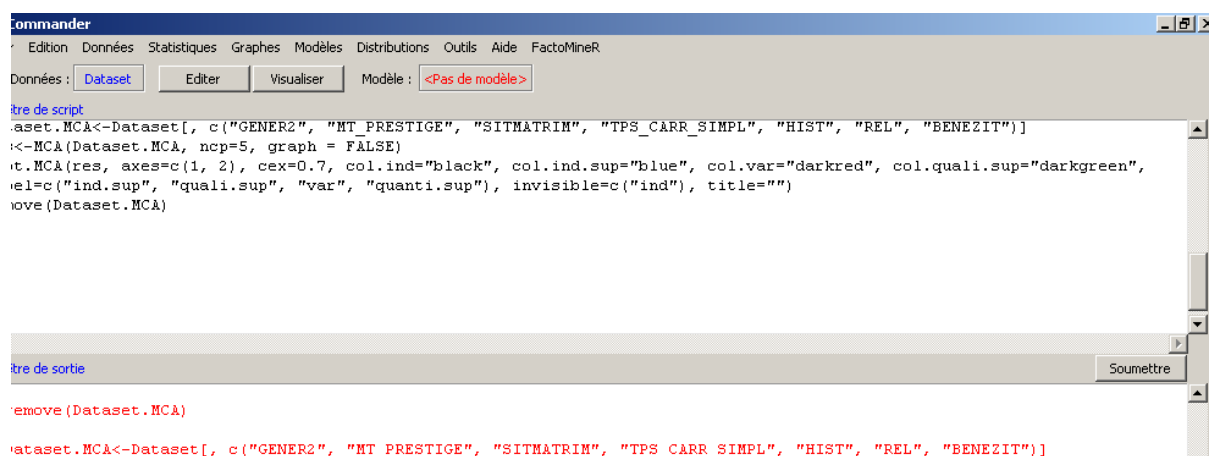
```
plot.MCA(res, axes=c(1, 2), cex=0.7, col.ind="black", col.ind.sup="blue",
col.var="darkred", col.quali.sup="darkgreen", label=c("ind.sup",
"quali.sup", "var", "quanti.sup"), invisible=c("ind"), title="")
```

Attention, il peut également s'avérer nécessaire, pour que la suite fonctionne, de s'assurer que toute l'instruction « se suit » sans passage à la ligne, en supprimant tout simplement ce dernier. Dans l'exemple, on avait en fait :

```
plot.MCA(res, axes=c(1, 2), cex=0.7, col.ind="black", col.ind.sup="blue",
col.var="darkred", col.quali.sup="darkgreen",
label=c("ind.sup", "quali.sup", "var", "quanti.sup"), invisible=c("ind"),
title="")
```

(passage à la ligne avant « label ») ; on a tout remis à la suite, en supprimant le saut de ligne. Une fois cela fait, positionner le curseur n'importe où dans cette longue instruction et cliquer sur le bouton « Soumettre », qui se situe à droite, vers le milieu de l'écran (voir image ci-dessous). Retourner dans la fenêtre RGui et admirer le graphique obtenu.

La réduction de taille des étiquettes doit déjà l'avoir rendu plus lisible. Vous pouvez réessayer avec cex=0.6, 0.9 ou autre pour faire varier cette taille. D'autres instructions vous permettent de changer d'autres choses, selon le même principe, pour une meilleure lisibilité (il y a une [première explication ici \(en anglais\)](#), mais elle n'épuise pas toutes les possibilités...). Cela dit, pour les non-geeks, on peut aussi se contenter de se faire une première idée sous R et faire ensuite, si tout va bien, un graphique plus joli sous Excel...



Le bouton « Soumettre » -----↑

Interlude :

Obtenir des identifiants explicites pour les individus

(dans les graphiques, classifications...)

Par défaut, R attribue à vos individus un numéro qui correspond à l'ordre dans lequel ils étaient rangés lorsque vous avez importé votre fichier (il ne reconnaît pas tout seul par magie que vous avez peut-être une colonne « ID » avec des identifiants numériques). Cela peut s'avérer peu pratique si vous réalisez un graphe des individus (sous dynGraph par exemple) ou une classification automatique (*infra*) et que vous voulez savoir qui est cette femme qui ne se comporte pas comme les autres ou qui est ce mystérieux paragon numéro 35 de la classe 4...

Pour indiquer à R où se trouve la colonne comprenant les identifiants individuels (numériques ou textuels, le nom par exemple) :

Juste après avoir importé vos données, ou à tout autre moment ensuite, taper ou copier-coller dans la fenêtre de script en haut (comme ci-dessus) l'instruction qui suit (ou une variante : cf. juste en dessous), et cliquer sur « soumettre » :

```
row.names(Dataset) <- Dataset$nom
```

Dataset, c'est le nom qui a été donné automatiquement à votre base lors de l'importation (si vous l'avez appelée autrement, changez l'instruction). Dataset\$nom se réfère à la colonne où se trouve le nom (en clair) ou l'identifiant que vous voulez utiliser pour les graphiques, etc. Chez vous, cette colonne ne s'appellera peut-être pas « nom » : ce sera peut-être par exemple Dataset\$ID ou Dataset\$identifiant.individuel. Il faut écrire, après \$, le nom de la colonne pertinente tel qu'il s'affiche quand on fait « visualiser ».

Nota bene : il faut que l'identifiant soit unique (si vous prenez les noms de famille et qu'il y a plusieurs fois le même pour différentes lignes, ça ne marchera pas).

5c. Utiliser dynGraph

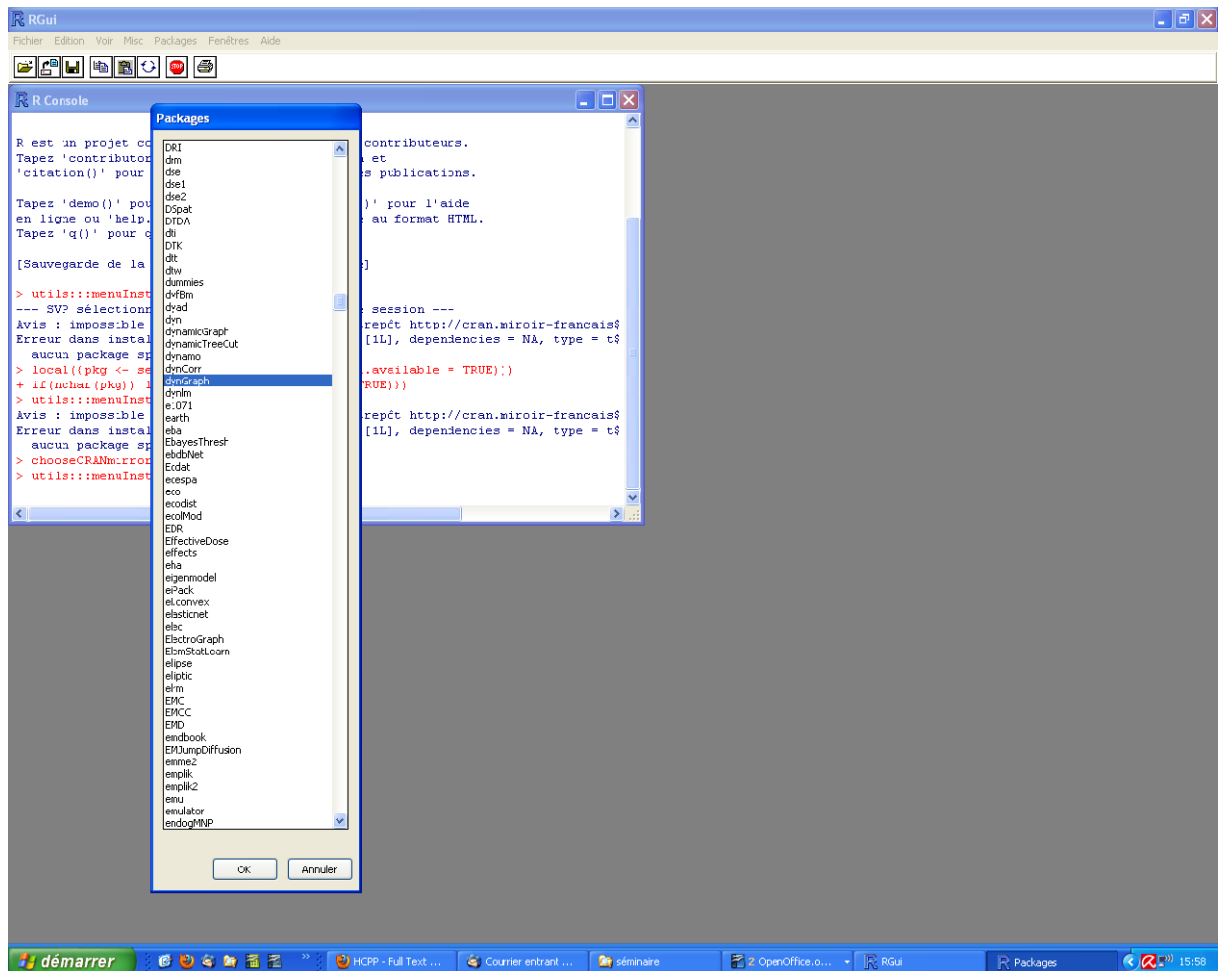
NB : en avril 2010, les conditions d'installation s'avèrent *fluctuantes* (le logiciel est vraiment bien, mais encore un peu expérimental...). Parallèlement, un site dynGraph est en construction sur <http://dyngraph.free.fr/index.html> : s'y référer en cas d'échec de la procédure décrite ici – ou pour d'autres précisions...

Le module complémentaire dynGraph, développé par la même équipe que FactoMineR, a plusieurs fonctions vraiment intéressantes ; en contrepartie, il faut l'installer spécialement et il n'a pas encore de manuel (et parfois il m'est arrivé qu'il ne se lance pas pour des raisons pas évidente : si vous avez un message d'erreur, écrivez-moi, qu'on comprenne ensemble !). Mais à mon sens, cela vaut la peine.

Pour que le module soit activé, deux étapes préalables sont nécessaires :

(a) à faire une fois pour toutes : l'installer. Pour cela, depuis la fenêtre RGui, dans le menu Packages, choisir Installer le(s) package(s). On vous demande alors de sélectionner un miroir, c'est-à-dire un endroit à partir duquel charger les fichiers. Prenez n'importe lequel, plutôt près de chez vous... et s'il se passe des choses étranges, essayez-en un autre (parfois, cela sature à Paris, alors que Lyon me semble plus efficace...). Quant tout va bien, une longue liste alphabétique s'ouvre : ce sont les dizaines de modules développés pour ajouter telle ou telle fonction à R. Il suffit de sélectionner dynGraph et taper sur OK pour qu'il s'installe (cela peut prendre un peu de temps). Ne pas confondre avec des modules aux noms voisins, comme dynamicGraph !

L'écran lors du choix du package à installer



(b) à faire à chaque fois que vous avez ouvert une nouvelle session RCommander, et de préférence dès le début de cette session (après l'ouverture de la fenêtre dédiée) : rendre possible le recours à dynGraph à partir de RCommander. Pour cela, dans le menu Outils de RCommander, choisir Charger des packages et sélectionner dynGraph.

Une fois que dynGraph est activé, réaliser l'ACM comme expliqué ci-dessus. Si vous n'avez rien changé aux instructions qui s'affichent automatiquement, les résultats de l'ACM (coordonnées, contributions, \cos^2 ...) sont automatiquement stockés dans un endroit (appelé « objet » en langage R) qui s'appelle `res` (trois lettres minuscules). Nous allons maintenant demander à dynGraph de réaliser un graphique modifiable à partir de ces résultats. Pour cela il suffit de taper, ou de copier-coller, dans la fenêtre d'instructions de RCommander (se reporter au 5b ci-dessus (p. 7) pour bien comprendre où cela se passe) ceci, en respectant majuscules et minuscules :

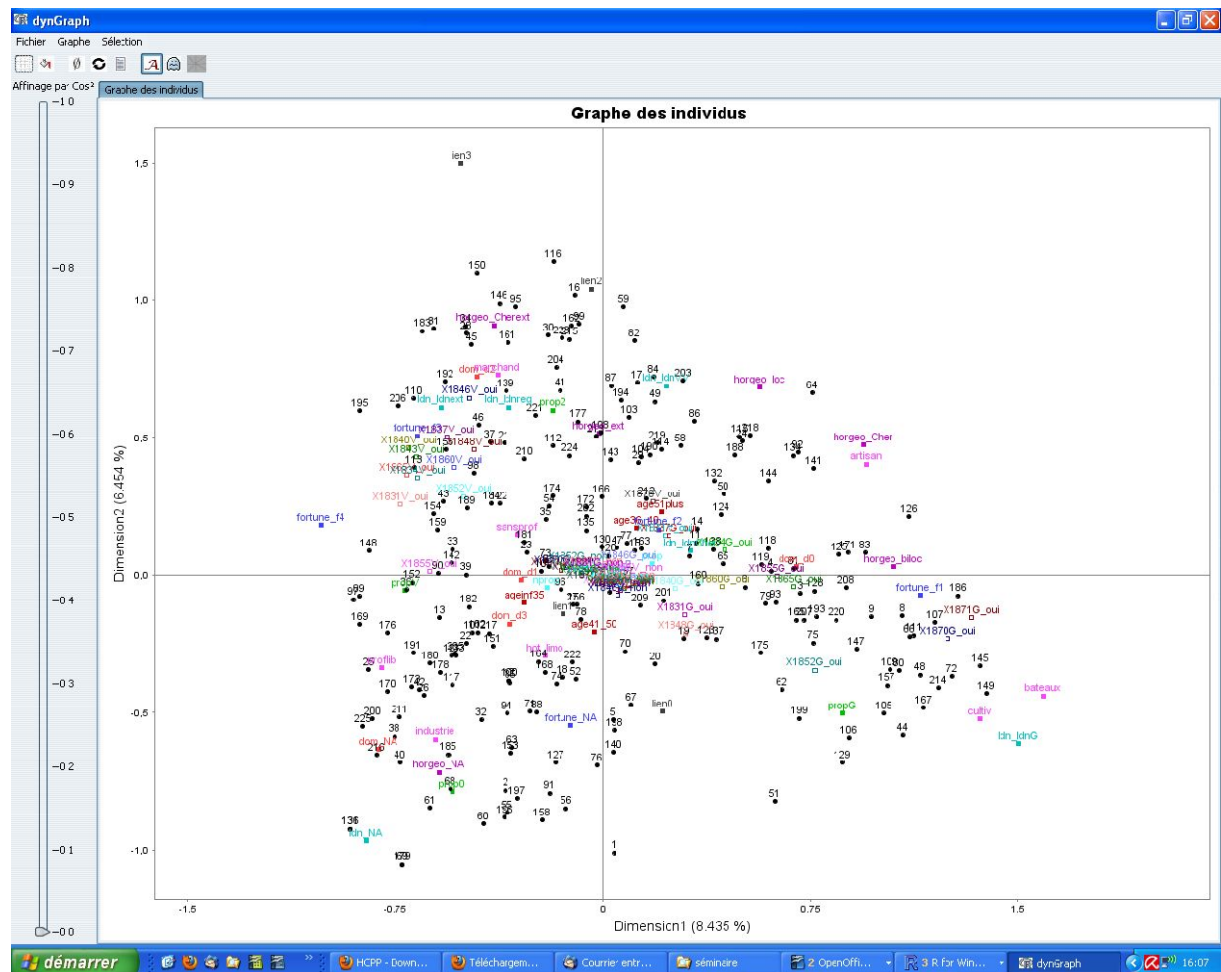
```
dynGraph(res)
```

Un certain temps s'écoule alors, puis une nouvelle fenêtre s'ouvre, où on va pouvoir explorer et améliorer le graphique, et le cas échéant l'exporter. Je ne vous liste pas ici tout ce que fait dynGraph (il est toujours utile d'essayer tous les menus, clics et boutons possibles...), mais voici quelques éléments intéressants.

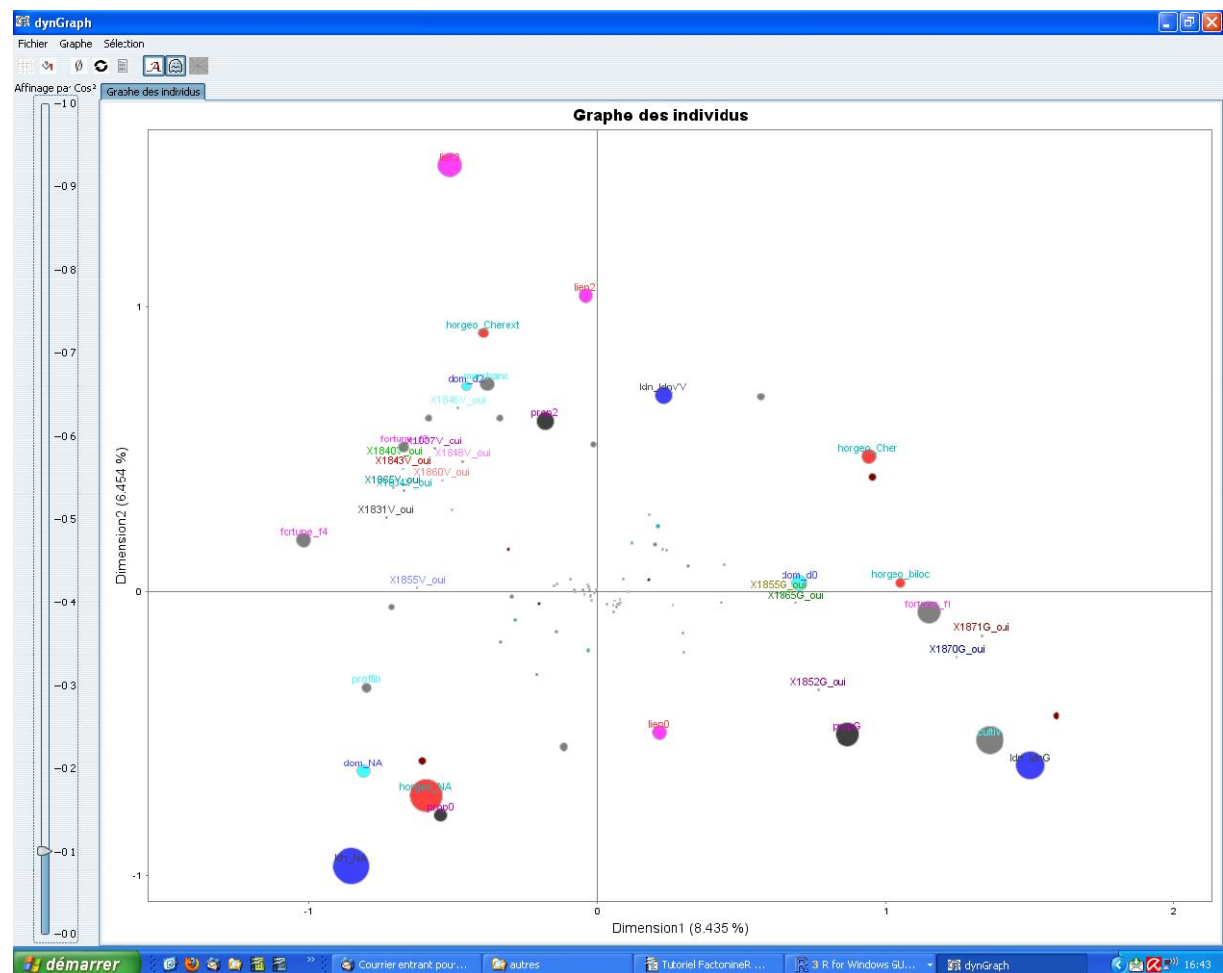
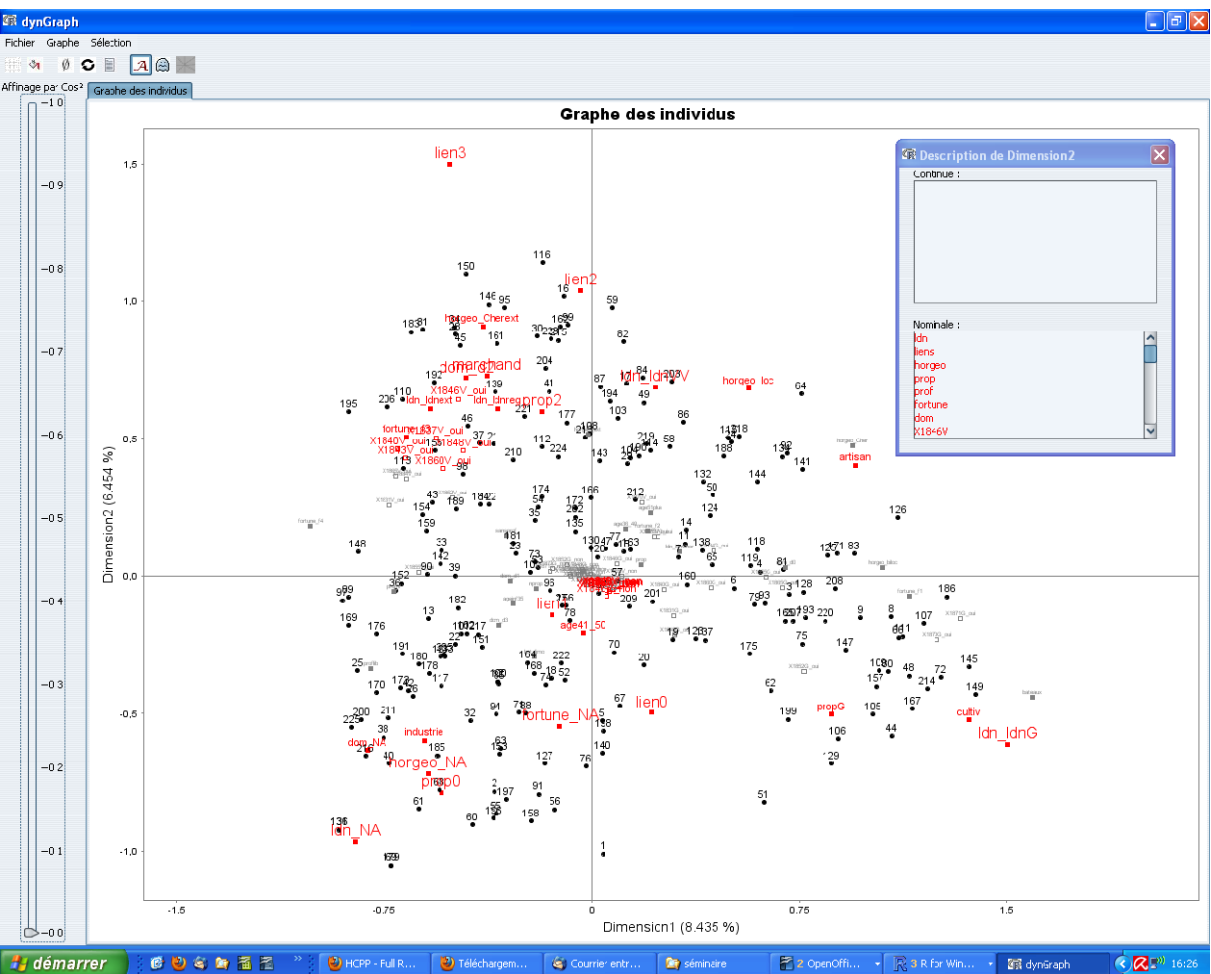
* au premier abord, on y voit déjà un peu plus clair, même si le fait que les individus soient systématiquement affichés avec les variables actives et supplémentaires peut obscurcir les choses. Mais on va voir comment y remédier. À noter que dans l'exemple, nos individus sont

représentés par des numéros ; mais si vous avez suivi la manipulation indiquée en p. 8 et que vous avez leurs noms, ce sont eux que vous verrez....

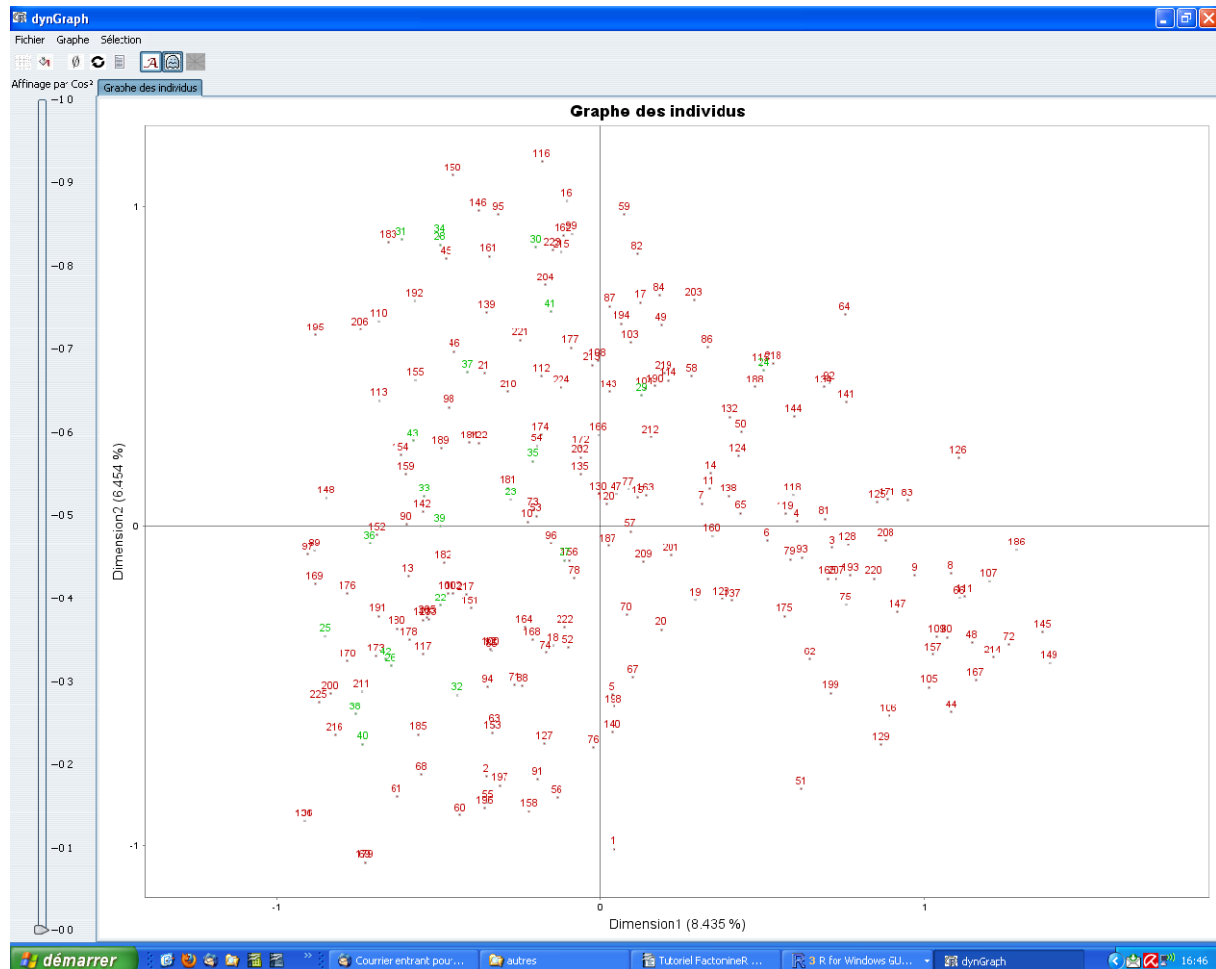
À noter aussi que le menu Fichier permet de changer la langue de l'interface (français ou anglais), ce qui est très appréciable.



* le curseur en bas à gauche de l'écran (\cos^2 : voir les bons manuels pour sa définition) permet à tout moment de réduire l'affichage aux points les mieux représentés sur les axes (ie, en gros, les plus loin du centre), ce qui est déjà un moyen à la fois d'y voir plus clair et d'éviter la surinterprétation.

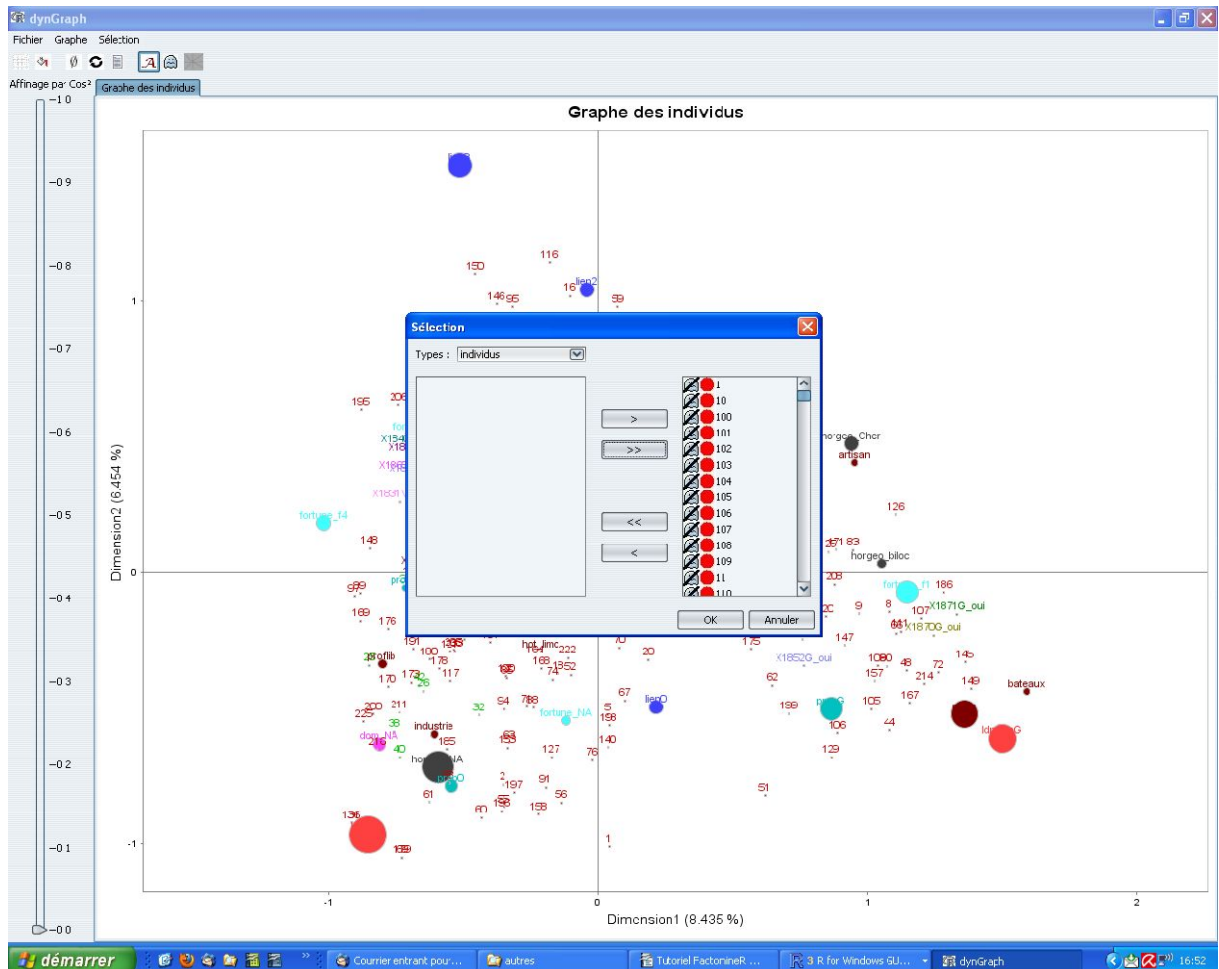


Résultat d'un « habillage » des individus selon une variable qualitative, qui est ici binaire (oui en vert, non en rouge)



Enfin, il est possible, comme ci-dessus, de n'afficher que les points individus, que les points variables, ou bien une partie seulement de ces points, etc. La manœuvre n'est pas très intuitive et je vous conseille de procéder par essais et erreurs, mais voici une base de travail, par exemple pour constituer un graphique ne montrant que les variables.

- choisir « sélectionner par liste » (clic droit, bouton ou menu). S'affiche une boîte de dialogue listant tous les individus et toutes les variables.
- faire passer tous les individus à droite en cliquant sur le bouton >>



- une fois que vous avez cliqué sur OK, tous les individus s'affichent en vert. C'est le signe du fait que vous allez pouvoir les faire disparaître d'un seul clic.
- et en effet, en cliquant sur « fantôme » (bouton, clic droit ou menu), il ne reste que les variables (et on peut retravailler leur affichage).

La même manoeuvre (faire passer dans la colonne de droite ce qu'on veut enlever, laisser ou refaire passer à gauche ce qu'on veut garder affiché) permet de ne garder que les variables, que les variables actives, que les modalités d'une seule variable, etc., au choix : il suffit d'utiliser le bouton > pour faire bouger une seule modalité ou un seul individu à la fois, ou >> pour tous les individus ou toutes les modalités de toutes les variables à la fois. Tout cela est bien sûr réversible (par des clics successifs sur « fantôme » ou retours à la liste de sélection).

6. Faire des meilleurs graphiques sous Excel

Nota bene 1 : Maintenant (2010), **la macro d'Olivier Godechot existe pour Excel 2007 !** Merci à Olivier, ainsi qu'à Jérémy Clairat qui me l'a signalé.

Nota bene 2 : À noter que, si l'on utilise plutôt [Openoffice.org version 3.0](http://Openoffice.org), ce qui est de toute façon une bonne idée (logiciel libre avec un environnement presque identique à Word, Excel, etc., et meilleur pour certaines fonctions). Dans ce cas, la simple réalisation d'un diagramme (Insertion>Diagramme) en « XY » (type de graphique XY(Dispersion)) à partir des coordonnées donne un graphique factoriel tout à fait joli, avec les bonnes étiquettes (ce que ne faisait pas Excel, d'où la nécessité de la macro). Avec un *caveat* : cela ne permet pas bien d'avoir plusieurs figurés différents sur le même graphique (un pour les variables actives et un pour les supplémentaires, par exemple).

Attention : faire des graphiques ne dispense pas de lire le reste de la sortie chiffrée (valeurs des axes, contributions...). Cf. pour cela, par exemple, le « Que sais-je ? » de Philippe Cibois.

- Ouvrir Excel, puis, depuis Excel, ouvrir le fichier .csv créé précédemment. Normalement, les données se mettent bien en place dans les colonnes.
- *Remplacer tous les points du fichier par des virgules* (sinon, les nombres décimaux ne sont pas reconnus), grâce à la fonction « Remplacer ».

Nota bene pour Openoffice : cette histoire de points et de virgules peut créer des problèmes particuliers (reconnaissance de certains nombres comme des dates). Il peut être plus prudent d'ouvrir d'abord le fichier .csv dans le bloc-notes ou un traitement de textes, d'y remplacer les points par des virgules, et seulement ensuite d'ouvrir le fichier obtenu sous Openoffice Calc (ou d'y copier-coller son contenu en collage spécial « texte non formaté »).

- Regarder où se trouvent les parties à représenter : coordonnées (« Coord ») pour les variables actives, supplémentaires, les individus... Par convention, colonne « Dim 1 » = axe horizontal ; colonne « Dim 2 » = axe vertical. En général, on va commencer par regarder les coordonnées sur ces deux axes...
- Copier sur une autre feuille de calcul (onglet) ces parties des données qui seront utilisées pour la représentation.
- Les mettre en forme dans cette nouvelle feuille de calcul, suivant le modèle donné dans le fichier macroACM97, ou macroACM2007 (pour le télécharger, voir ici : <http://www.quanti.ihmc.ens.fr/document.php?id=123>, deuxième partie). Attention, au moment où vous ouvrez ce fichier, il faut autoriser les macros, sinon rien ne se passera.
- Quand vous avez d'une part votre feuille avec vos coordonnées, d'autre part le fichier macroACM97, ou macroACM2007, également ouvert, placez-vous dans votre feuille et allez chercher dans le menu Outils>Macro...>Exécuter la macro ACM97 (ou 2007). Puis laissez-vous guider par les boîtes de dialogue.
- Une fois le graphique réalisé, il est possible de l'améliorer visuellement, comme tout graphique Excel (changements de couleurs, polices, etc.).

7. Faire d'autres choses avec FactoMineR...

- Ne pas hésiter à explorer les autres onglets, qui offrent entre autres une solution rapide et pratique pour les tableaux croisés assortis de tests de χ^2 (voir Statistiques>Tables de contingence).
- Certains graphiques sont également très intéressants (« Boîte de dispersion » pour des données quantitatives par exemple).
- Voir notre tutoriel pour la régression logistique, en annexe de la page <http://www.quanti.ihmc.ens.fr/document.php?id=108>

8. Produire des graphiques en ellipse et des classifications automatiques à partir des résultats de l'ACM

Attention : *ceci n'est pas un cours sur la classification automatique*. Pour comprendre de quoi il retourne, on peut recourir :

- sur les ellipses, et en attendant un exemple très éclairant à paraître sous la plume de François Denord (merci à lui pour des explications sur cette technique), à Jean Chiche, Brigitte Le Roux, Pascal Perrineau et Henry Rouanet, « [L'espace politique des électeurs français à la fin des années 1980](#) », *Revue française de science politique*, vol. 50, n°3, juin 2000, p. 463-468.
- sur les méthodes de classification, en introduction à quelques pages très claires dans le petit manuel d'Olivier Martin, *L'analyse de données quantitatives*, Paris, Armand Colin (« 128 »), 2005 et en approfondissement au gros et bon manuel de Ludovic Lebart, Alain Morineau et Marie Piron, *Statistique exploratoire multidimensionnelle*, Paris, Cours, Dunod, 2004 (même en sautant les pages purement mathématiques, il est très éclairant).

Il est possible de *réaliser des classifications* (pas des ellipses) *à partir des menus de RCommander* ; toutefois, cela demande une petite gymnastique. Ici sera donc privilégiée l'approche par copier-coller (et petite modification) d'instructions dans la moitié supérieure de la fenêtre, qui a été esquissée plus haut à propos des graphiques. Pour ceux qui voudraient tout de même s'y essayer, la technique pour réaliser une classification à partir d'une ACM est la suivante :

- fabriquer un fichier Excel contenant les coordonnées sur les dimensions de l'ACM que l'on veut prendre pour base, pour les individus étudiés (cela peut être le fichier d'origine augmenté des coordonnées, ou bien un fichier avec seulement les coordonnées, extrait du .csv créé par FactoMineR). Importer ce fichier dans RCommander
- dans Statistiques>Analyse multivariée>Classification, choisir K-means ou Classification ascendante hiérarchique. Dans le premier cas, choisir le nombre de classes et cocher toutes les options. Dans le second cas, choisir la méthode, cocher « dessiner le dendrogramme », l'observer pour choisir un nombre de classes, puis
- dans Statistiques>Analyse multivariée>Classification, choisir « résumer une classification hiérarchique », indiquer un nombre de classes et laisser les cases cochées, puis choisir « Ajouter les groupes de la classification au jeu de données ».

Principal défaut de ces méthodes : à part leur placement sur le plan factoriel, elles ne fournissent pas directement d'informations pour caractériser les classes. Comme l'appartenance aux classes est ajoutée au jeu de données, il faut ensuite faire d'autres traitements (tableaux croisés...) pour voir ce que recouvrent ces classes.

Pour plus de détails sur ces fonctions, voir [la notice du package FactoMineR](#) (en anglais).

Ellipses

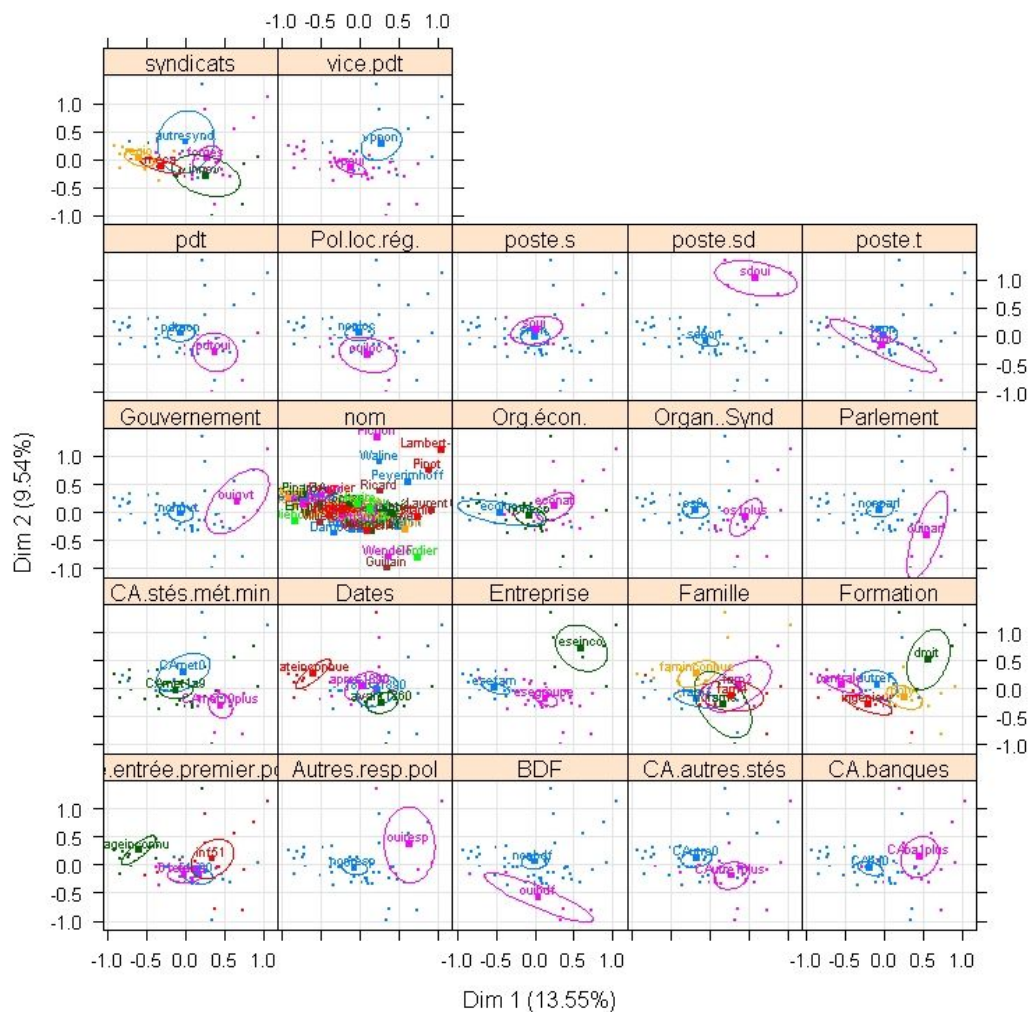
Après avoir réalisé une ACM comme ci-dessus, copier-coller dans la fenêtre d'instructions celle qui suit (sur une seule ligne !) et appuyer sur « Soumettre » (quand le curseur est dans l'instruction) :

```
plotellipses(res, keepvar = "all", axis = c(1, 2), means=TRUE, level =
0.95, magnify = 2, cex = 0.3, pch = 20, pch.means=15, type = c("g", "p"),
keepnames = TRUE, namescat = NULL)
```

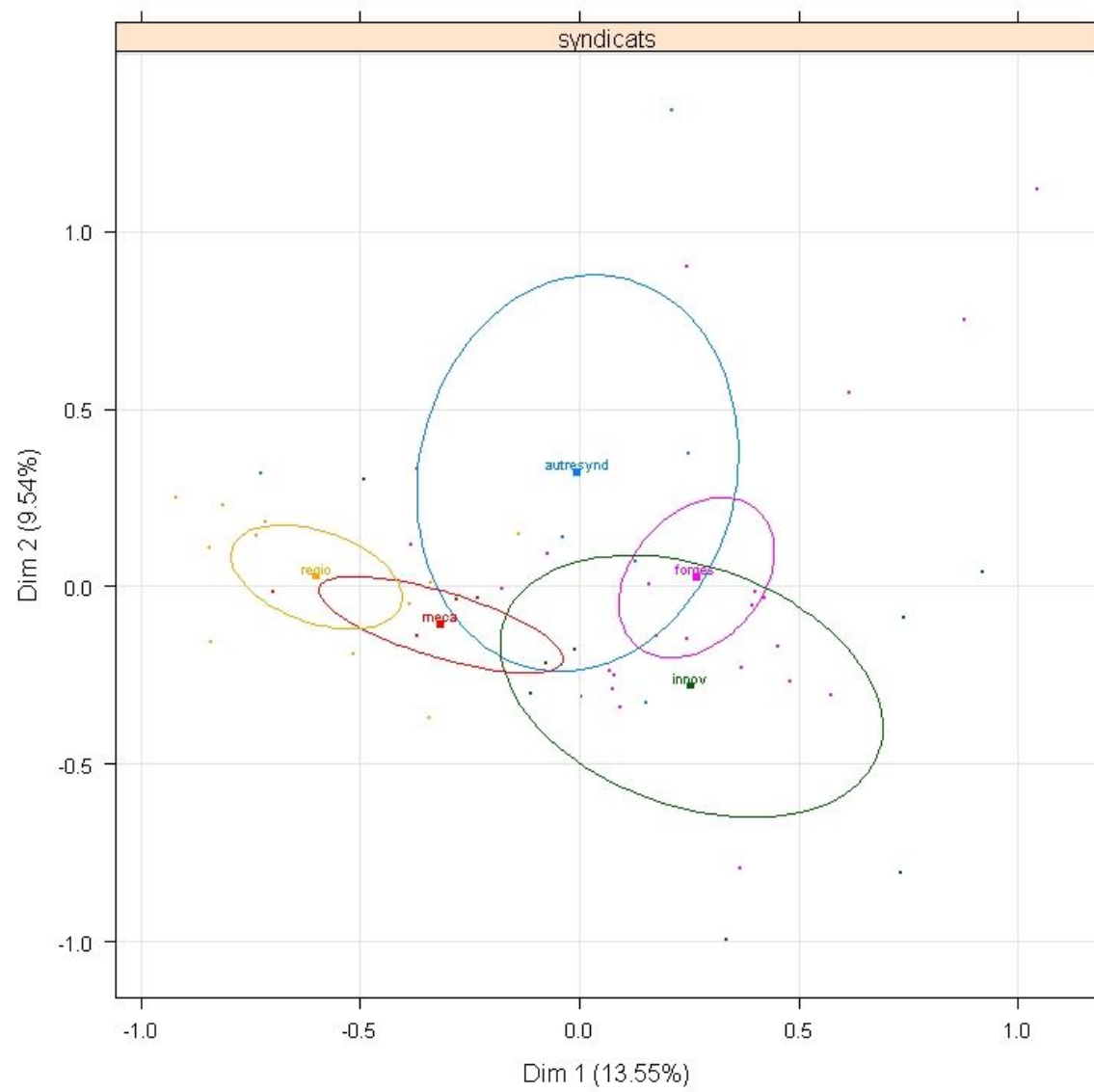
Aller voir le graphique dans la fenêtre RGui (voir un exemple ci-dessous). Cette version de l'instruction représente toutes les variables (actives et supplémentaires), ce qui peut faire trop. Pour obtenir une représentation variable par variable, remplacer « all » dans l'instruction ci-dessus par un nom de variable (tel que lu par RCommander : attention aux points, etc. dans les intitulés – visualisez votre base de données s'il y a un doute). Par exemple :

```
plotellipses(res, keepvar = "syndicats", axis = c(1, 2), means=TRUE, level =
0.95, magnify = 2, cex = 0.5, pch = 5, pch.means=25, type = c("g", "p"),
keepnames = TRUE, namescat = NULL)
```

Graphique général (données inspirées d'un travail réalisé avec Danièle Fraboulet)

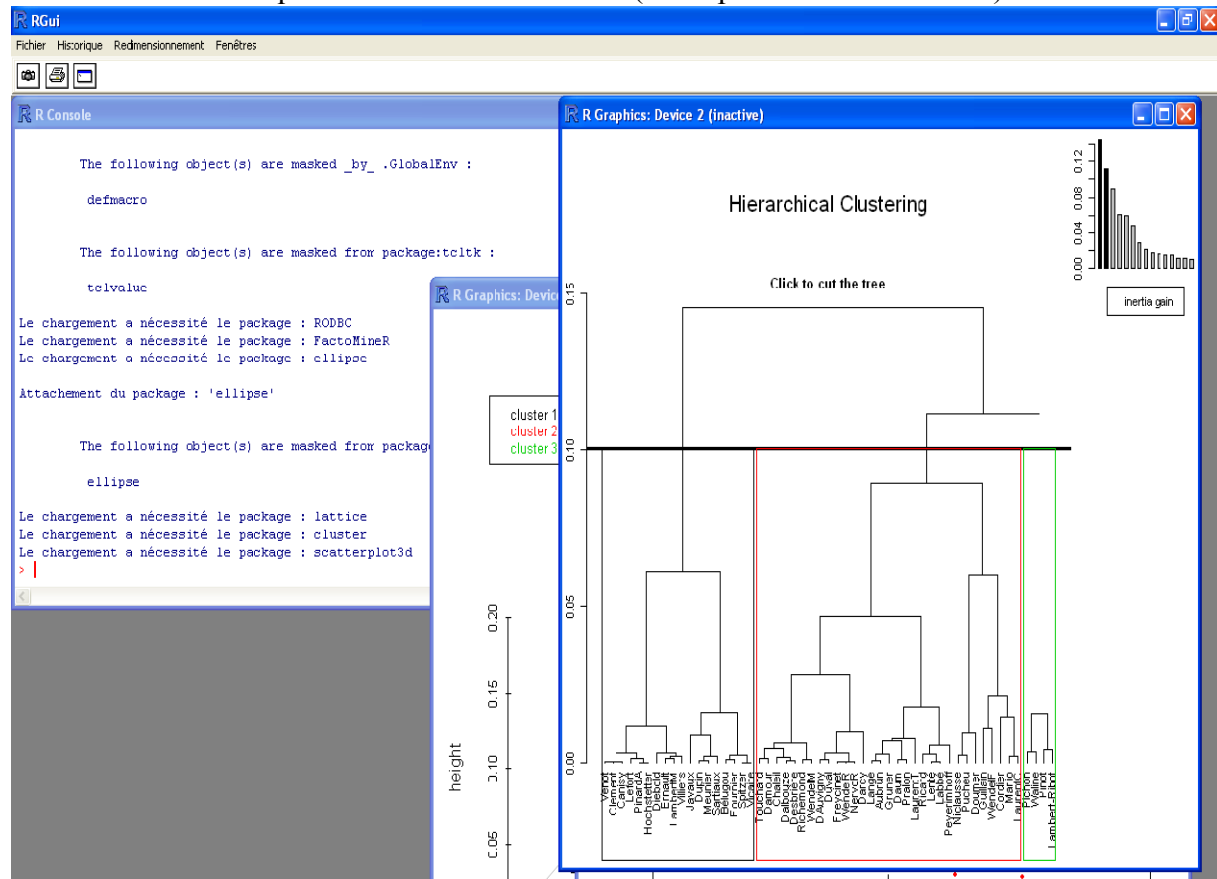


Graphique sur la variable “syndicats”

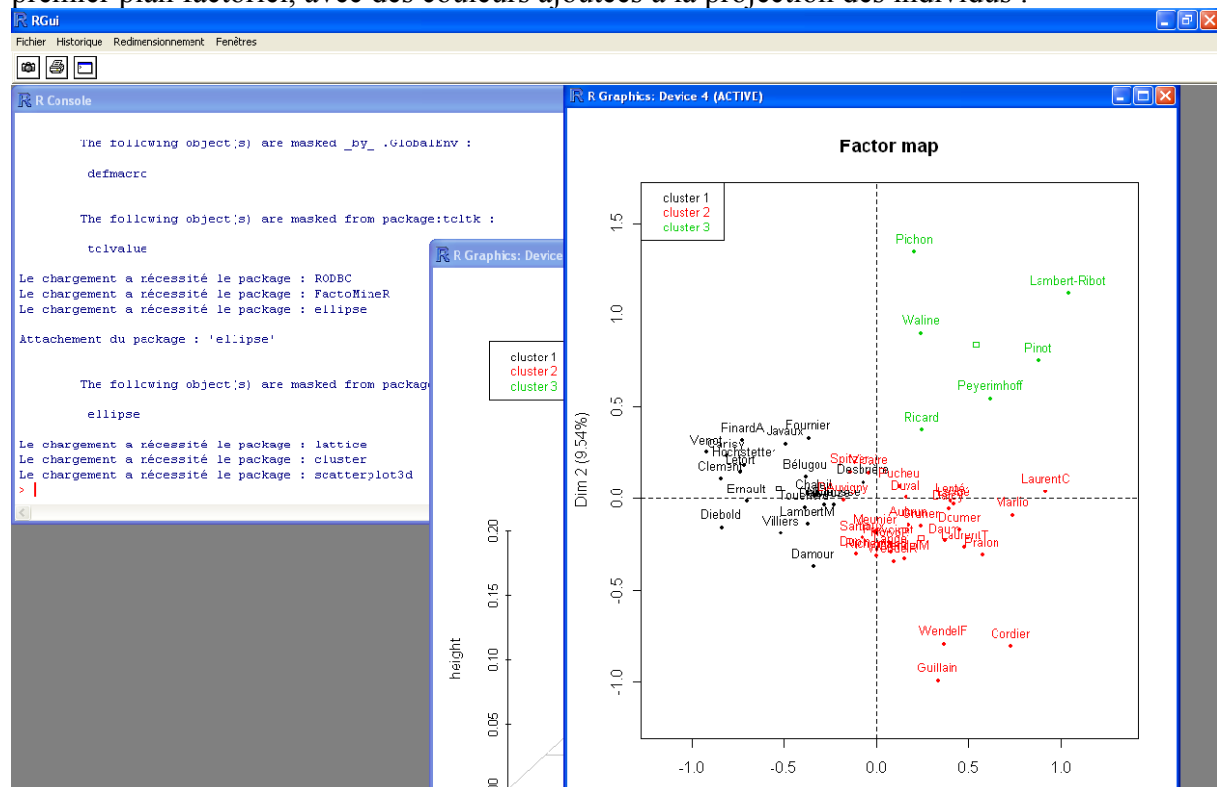


après avoir « coupé » :

on voit les couleurs qui seront celles des classes (et les proximités entre elles)



2. dans la même fenêtre (tout ça se superpose), outre un graphique 3D qui n'apporte pas beaucoup de plus (au lecteur moyen du moins), un graphique situant les classes dans le premier plan factoriel, avec des couleurs ajoutées à la projection des individus :



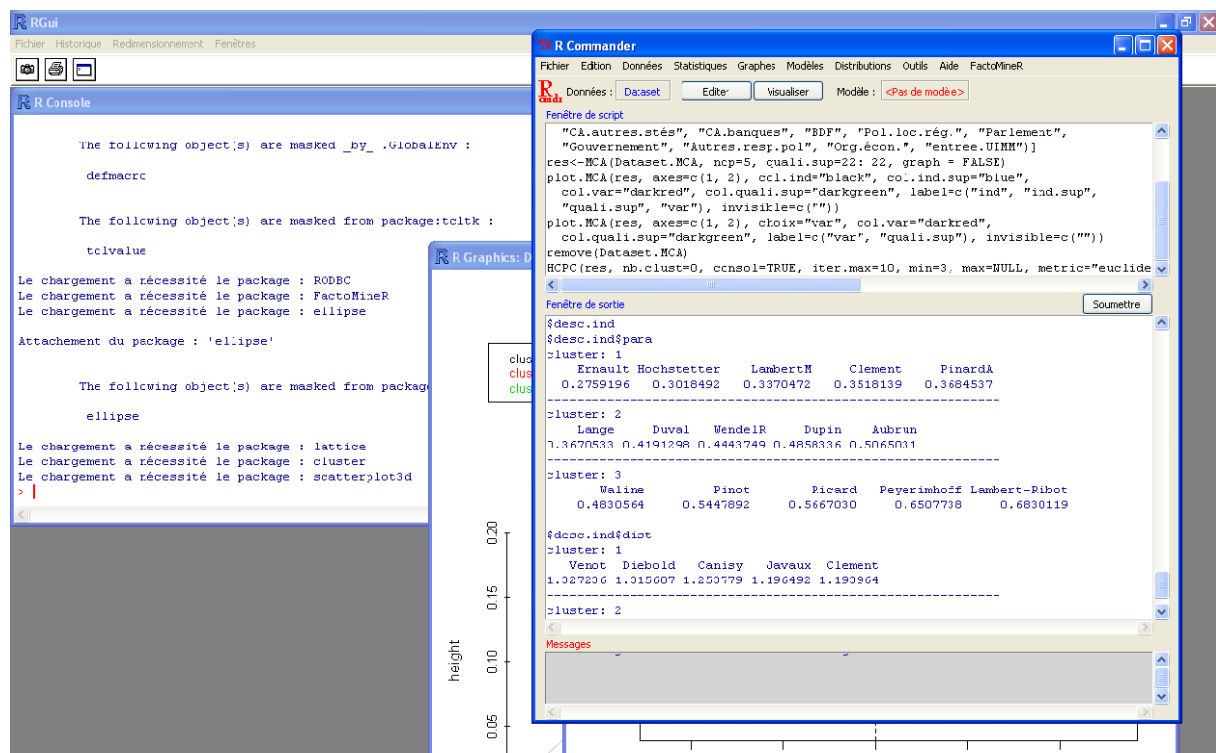
Sur cette base, on peut déjà repérer un peu à quel groupe appartient chaque individu, mais il faut encore comprendre comment sont construits ces groupes, ce qui est possible grâce aux indications données dans la fenêtre RCommander.

3. Cette fois dans la fenêtre RCommander, un très grand nombre de renseignements chiffrés, touffus et redondants, mais essentiels pour l'interprétation. Reprenons ce fichier à partir de la fin (quand vous revenez à la fenêtre RCommander, c'est la fin que vous voyez), sans commenter (au moins pour l'instant !) les parties « moins utiles » pour une première interprétation. Trois morceaux vont nous intéresser.

a. Deux listes d'individus :

- les « parangons », individus « moyens » de chaque classe, situés au centre du nuage de points de leur couleur : `ind$para` ;

- les individus les plus « extrêmes » de la classe, les plus distants des autres classes, les plus éloignés des autres couleurs sur le graphique : ils représentent en quelque sorte l'« idéal-type » au sens d'accentuation des caractéristiques distinctives, et non pas l'individu « typique » au sens de « moyen » : `ind$dist`.



Ces individus sont désignés par leur numéro, sauf si vous avez pris la précaution (cf. *supra* p. 8) de faire reconnaître les noms par le logiciel. Par exemple, ci-dessus, dans le cluster 1, le paragon est Ernault (et à un moindre degré Hochstetter, etc.) et l'extrême Venot (et à un moindre degré Diebold, etc.), ce que confirme l'illustration plus haut : Ernault est au centre du nuage de points noirs, le Venot à l'extrême (gauche, en l'occurrence, les autres groupes, rouge et vert, étant à droite). Cela permet de façon intéressante de revenir au qualitatif et même au narratif...

b. Une description de ce que « représente » chaque classe, par un simple croisement entre les classes et les variables d'origine. En fait, le logiciel réalise tout simplement un tableau croisé pour chaque variable (croisant l'appartenance aux classes et cette variable), assorti d'un test de chi-2. Évidemment, quand on a beaucoup de variables, c'est bien qu'il fasse tout cela automatiquement... et il présente les résultats de façon synthétique, ce qui permet de comprendre « ce qui définit » chaque classe.

Ces résultats sont introduits par

```
$desc.var
$test.chi2
```

qui indique simplement de façon très générale quelles sont les variables les plus corrélées au découpage en classes. Par exemple dans le cas ci-dessous (cas de l'UIMM pris en exemple dans le Repères), ce découpage est très corrélé avec le type d'entreprise et le fait de tenir ou non le poste de secrétaire délégué, un peu moins corrélé mais corrélé quand même avec le fait de tenir le poste de président, et pas corrélé significativement avec les variables qui ne sont pas dans cette liste (comme la date d'entrée à la direction de l'UIMM : cela veut dire que nos trois types sont présents dans toutes les générations).

```
$desc.var
$test.chi2
```

	p.value	df
Entreprise	3.511342e-14	4
poste.sd	4.371484e-08	2
Age.entrée.premier.poste	5.355945e-06	6
Formation	2.761383e-05	8
Dates	5.891589e-05	6
syndicats	1.254588e-04	8
CA.banques	2.587376e-03	2
Organ..Synd	2.718414e-03	2
vice.pdt	3.254626e-03	2
CA.autres.stés	5.183036e-03	2
Org.écon.	1.000547e-02	4
CA.stés.mét.min	1.348569e-02	4
pdt	1.490067e-02	2

Mais les détails classe par classe sont plus parlants et plus intéressants. Ils sont introduits par

```
$category
```

Voyons leur lecture sur un exemple (toujours l'UIMM). À noter que suivant la largeur de la fenêtre RCommander, tout ce qui suit peut ne pas s'afficher sur une seule ligne. C'est un peu moins commode, mais tous les chiffres sont bien là...

```
$category
$category$`1`
```

	Cla/Mod	Mod/Cla	Global	p.value	v.test
Entreprise=esefam	87.50000	73.68421	30.18868	8.967667e-07	4.913036
Age.entrée.premier.poste=ageinconnu					
	91.66667	57.89474	22.64151	1.964274e-05	4.268914
Dates=dateinconnue	91.66667	57.89474	22.64151	1.964274e-05	4.268914
syndicats=regio	90.90909	52.63158	20.75472	8.439453e-05	3.931565
Formation=centrale	90.00000	47.36842	18.86792	3.316286e-04	3.589252
Organ..Synd=os0	50.00000	100.00000	71.69811	5.938766e-04	3.434396
CA.autres.stés=CAutre0	51.51515	89.47368	62.26415	4.087201e-03	2.871352
vice.pdt=vpoui	47.22222	89.47368	67.92453	2.238273e-02	2.283810
Org.écon.=ecoloc	83.33333	26.31579	11.32075	3.680578e-02	2.087912
pdt=pdtnon	42.22222	100.00000	84.90566	4.096974e-02	2.043836
pdt=pdtoui	0.00000	0.00000	15.09434	4.096974e-02	-2.043836
Formation=poly	12.50000	10.52632	30.18868	3.711729e-02	-2.084472
Age.entrée.premier.poste=51a60					
	11.76471	10.52632	32.07547	2.238273e-02	-2.283810
vice.pdt=vpnon	11.76471	10.52632	32.07547	2.238273e-02	-2.283810
CA.stés.mét.min=CAmet10plus					
	0.00000	0.00000	18.86792	1.344966e-02	-2.471635
syndicats=forgeries	13.04348	15.78947	43.39623	4.870722e-03	-2.815462
Dates=avant1860	0.00000	0.00000	22.64151	4.110860e-03	-2.869527
CA.autres.stés=CAautre1plus					
	10.00000	10.52632	37.73585	4.087201e-03	-2.871352
Entreprise=esegroupe	16.66667	26.31579	56.60377	2.199936e-03	-3.061823
Organ..Synd=os1plus	0.00000	0.00000	28.30189	5.938766e-04	-3.434396

La dernière colonne (v.test : une mesure d'association entre variables) nous permet d'abord de distinguer les modalités positivement corrélées avec la classe (surreprésentées : ici de « entreprise familiale » à « non président ») à celles qui sont négativement corrélées avec la classe, *i. e.* sous-représentées en son sein (les autres, qui ont une v.test négative). Toutes les modalités énumérées ici sont significativement corrélées avec la classe ; toutes celles qui ne sont pas énumérées ici ne le sont pas. Les plus corrélées positivement sont au début, les plus corrélées négativement sont à la fin (ici : « entreprise familiale », notamment, *vs.* « entreprise groupe », notamment).

Les deux premières colonnes méritent d'être lues avec attention, car elles sont concrètes et grappantes.

- cla/mod indique quelle part (pourcentage) de tous les individus présentant cette modalité se retrouve dans cette classe (ce cluster, cette catégorie).

- mod/cla indique quelle part (pourcentage) de tous les individus de la classe présentent cette modalité.

Ce sont deux façons de parler de sur-représentation, qui, selon les effectifs totaux de la modalité et de la classe, peuvent paraître assez différentes.

Dans notre exemple, près de 90 % des chefs d'entreprise familiale, et 90 % des dirigeants de syndicats régionaux, se trouvent classés dans la classe 1. En revanche, seulement 74 % des individus classés dans cette classe 1 ont une entreprise familiale, et 53 % dirigent un syndicat régional. Ou encore, on ne trouve aucun président de l'UIMM dans la classe 1, mais on y trouve seulement 42 % des « non présidents »... Tout cela représente souvent un utile rappel de la diversité à l'intérieur de chaque profil... et donc de ce que c'est que de faire une typologie (même qualitativement !).

c. Enfin, beaucoup plus haut dans la sortie RCommander, on retrouve l'énumération des caractéristiques de chaque individu (comme dans le fichier de données importé), avec à la fin une colonne ajoutée, qui indique dans quel groupe R a classé l'individu ; Tout cela peut prendre beaucoup de place : chaque individu peut apparaître plusieurs fois jusqu'à ce que toutes ses variables (colonnes) soient mentionnées. Et ce qui est nouveau est dans la dernière...

Tout cela se trouve au tout début de la sortie, après `$data.clust`

On retrouve par exemple, après d'autres variables importées, ceci :

```
Org.écon. entree.UIMM clust
Venot          ecoloc  ent1918a39      1
Clement       noneco  entap1939      1
Diebold        ecoloc  entap1939      1
Canisy         ecoloc  entap1939      1
Lefort         noneco  entap1939      1
PinardA        noneco  entav1918      1
Hochstetter    noneco  ent1918a39     1
Ernault        noneco  ent1918a39     1
(... autant de lignes en plus que d'individus en plus)
```

La colonne « clust » correspond à l'appartenance de classe : ici, on a des membres de la classe 1.

Inconvénient : la colonne indiquant l'affectation de chaque individu à une classe n'est pas ajoutée automatiquement à votre base de données et aucune option à l'intérieur même de l'instruction n'est prévue pour vous permettre de la récupérer aisément sous forme de tableau. Cependant, la base avec cette colonne ajoutée nous est montrée : on peut copier-coller la partie pertinente, ou bien demander l'exportation de l'ensemble dans un fichier .txt (ensuite lisible sous Excel : l'ouvrir en indiquant l'espace comme séparateur). Pour l'instant, je n'ai pas trouvé plus élégant que ce qui suit pour créer ce fichier (si vous avez mieux, écrivez-moi !):

1. reproduire l'instruction ci-dessus en changeant juste le début, et faire à nouveau « Soumettre » : les résultats chiffrés ne sont plus montrés, mais « stockés » dans l'objet « dc ».

```
dc <- HCPC(res, nb.clust=0, consol=TRUE, iter.max=10, min=3, max=NULL,  
metric="euclidean", method="ward", order=TRUE, graph.scale="inertia",  
nb.par=5, graph=TRUE)
```

2. exporter la partie de l'objet « dc » qui contient les données et la colonne supplémentaire :

```
write.table(dc$data.clust, "dc.txt")
```

Le nom du fichier à créer peut évidemment être adapté, et il est possible d'indiquer le « chemin » pour le faire créer à un endroit précis (par exemple "C:/Documents and Settings/ciaire/Mes documents/méthodo/dc.txt"). Par défaut, « chez moi », il arrive dans « Mes documents ».