

Tutoriel RCommander pour la régression logistique

par Claire Lemerrier et Séverine Sofio

1^{re} version : 21 janvier 2009-mise à jour : 16 février 2009

Commentaires bienvenus : Claire.Lemerrier@ens.fr

1. Préparer les données

- Pour les bonnes pratiques de codage, voir notre Repères.
- Si les données sont dans un fichier Excel, il faut que la première ligne donne les intitulés des colonnes.
- 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.
- 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.

Mise à jour : le codage « NA » semble poser problème pour la régression logistique, dont le principe s'accommode mal de trop de données manquantes. Dans ce cas, remplacer « NA » par « inconnu » ou autre code (qui sera donc une modalité comme une autre pour la régression), et/ou éliminer de la base de données les individus pour lesquels il y a trop de données manquantes.

- Le logiciel 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 ça 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 »...

2. Installer les logiciels : voir <http://www.quanti.ihmc.ens.fr/document.php?id=78> (dernière partie)

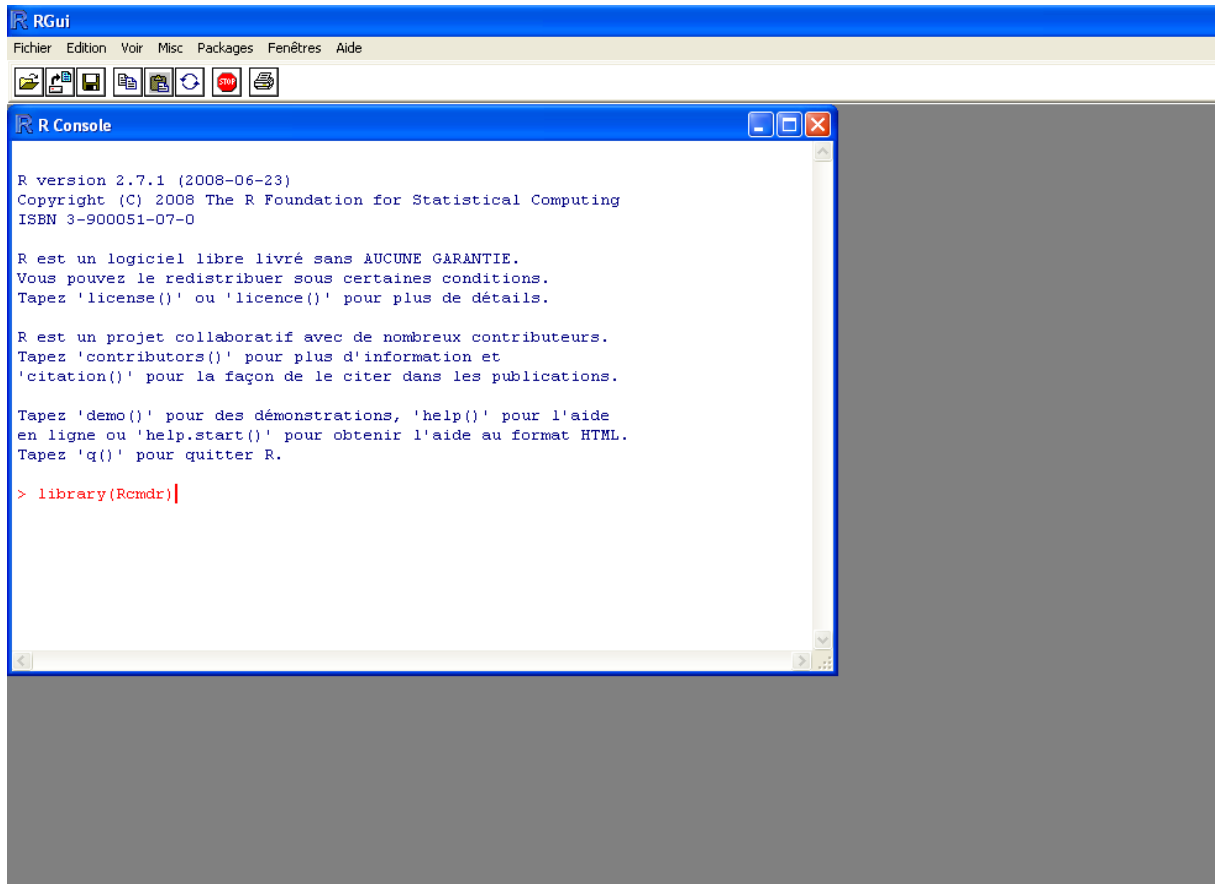
3. Lancer R, RCommander (et FactoMineR)

- Lancer R (il faut vous souvenir où il est dans votre ordinateur...)
- 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.
- 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, mais si tel n'est pas votre but, vous n'avez pas à vous en préoccuper.

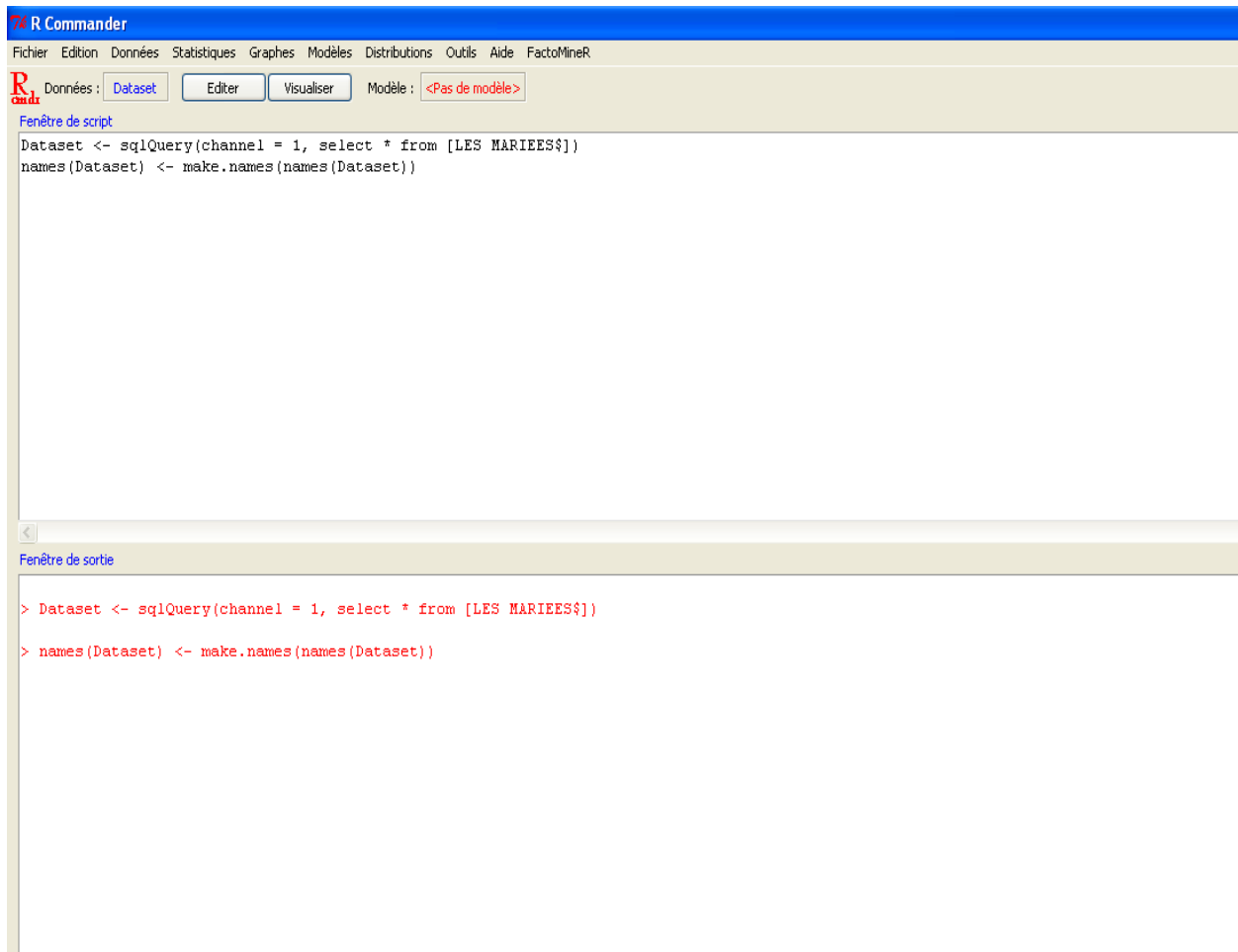
4. Importer les données

- 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 RCommander) → 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.

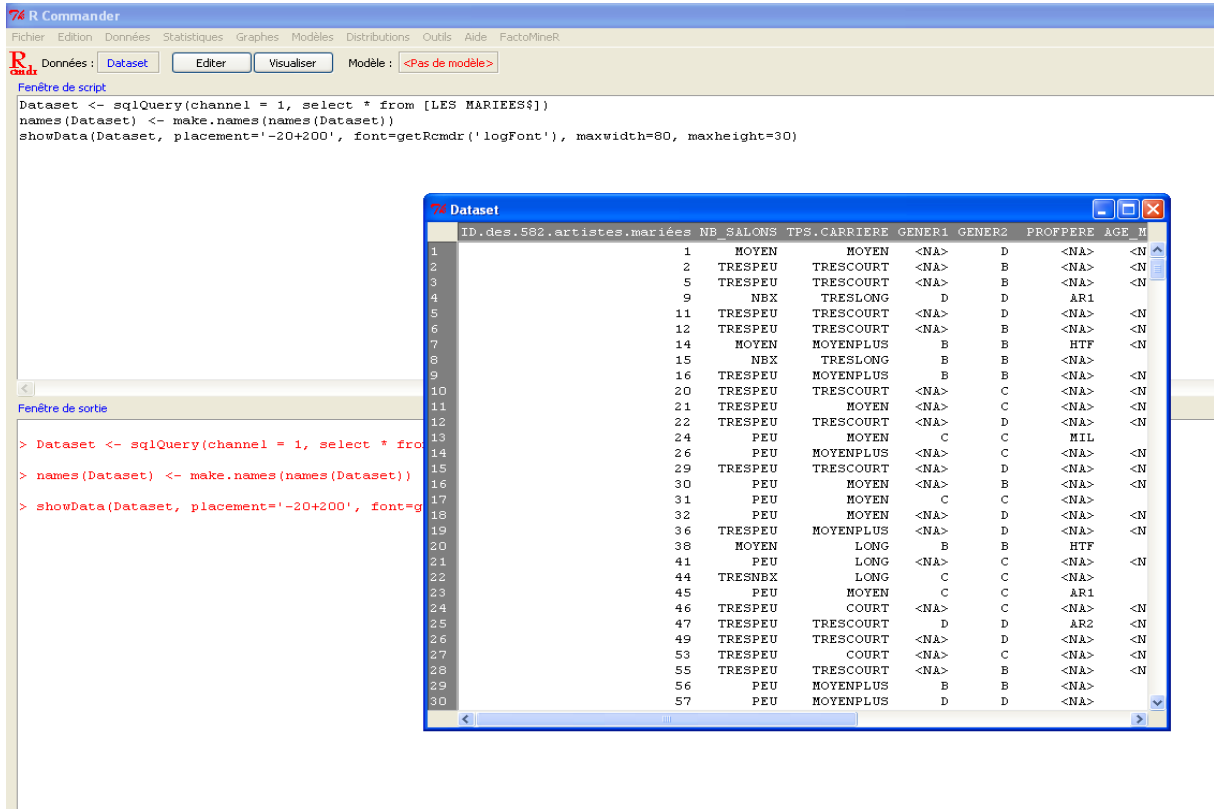
Instruction initiale tapée dans R :



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



Vérification des données :



The screenshot shows the R Commander interface. The main window displays the following R code:

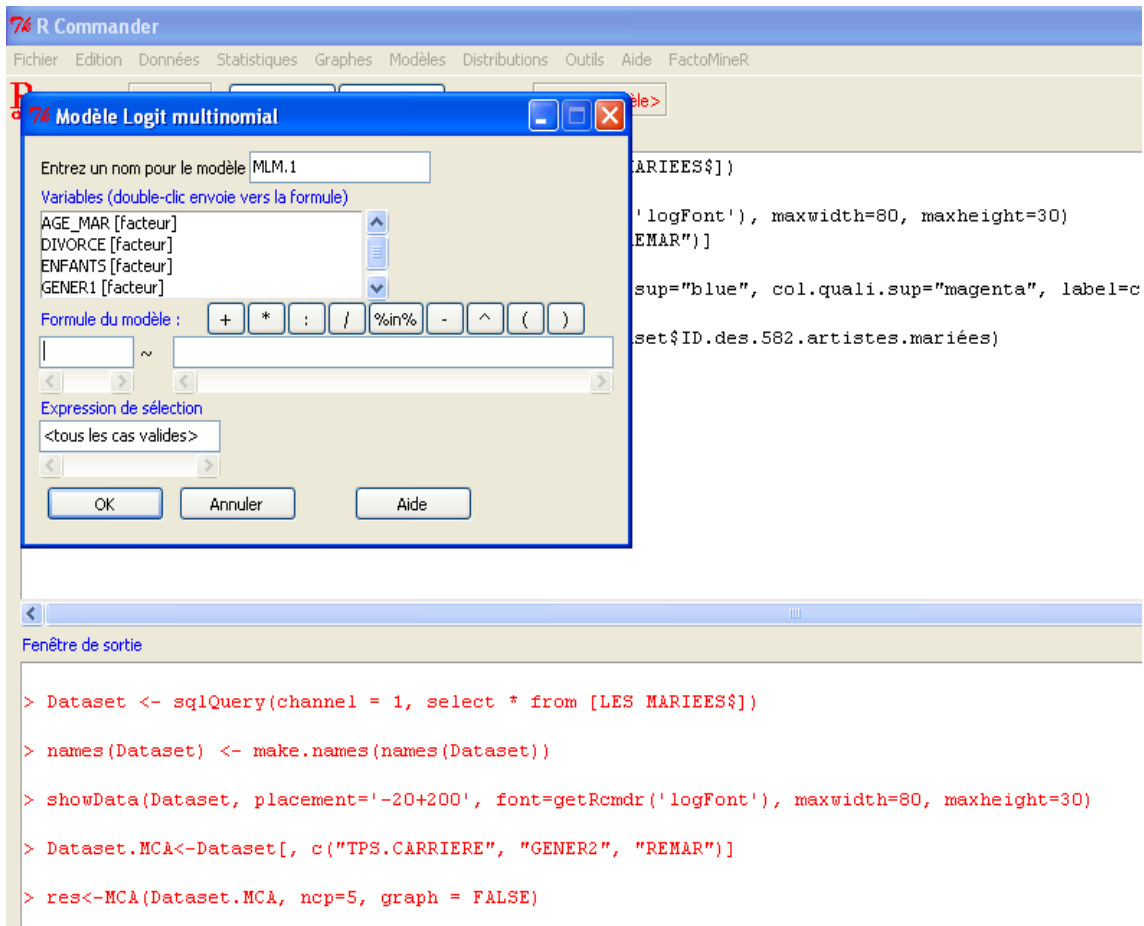
```
Dataset <- sqlQuery(channel = 1, select * from [LES MARIEES$])
names(Dataset) <- make.names(names(Dataset))
showData(Dataset, placement='-20+200', font=getRcmdr('logFont'), maxwidth=80, maxheight=30)
```

A secondary window titled "Dataset" displays a table with the following columns: ID.des.582.artistes.mariées, NB_SALONS, TPS.CARRIERE, GENER1, GENER2, PROPPERE, AGE, M. The table contains 30 rows of data.

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

5. Faire une régression logistique

- Voir notre Repères (ou des manuels plus pointus !) pour les grands principes de construction d'un modèle.
- Menu Statistiques > Ajustement de modèles > Modèle logit multinomial



The screenshot shows the R Commander interface with the "Modèle Logit multinomial" dialog box open. The dialog box contains the following information:

- Entrez un nom pour le modèle: MLM.1
- Variables (double-clic envoi vers la formule): AGE_MAR [facteur], DIVORCE [facteur], ENFANTS [facteur], GENER1 [facteur]
- Formule du modèle: [] ~ []
- Expression de sélection: <tous les cas valides>

The console window shows the following R code:

```
> 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)
> Dataset.MCA<-Dataset[, c("TPS.CARRIERE", "GENER2", "REMAR")]
> res<-MCA(Dataset.MCA, ncp=5, graph = FALSE)
```

- Dans la liste des variables qui s'affiche, sélectionner la variable à expliquer en double-cliquant sur son nom.
*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 la régression. 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 la régression.*
- Dans la liste des variables qui s'affiche, sélectionner l'une après l'autre les variables explicatives à tester en double-cliquant sur leur nom. Noter qu'elles s'affichent avec un « + » entre elles : c'est un modèle simple, qui « pose les variables les unes à côté des autres ». On peut aussi créer des modèles plus complexes en utilisant les signes qui sont en face de « Formule du modèle ». Ainsi, le signe « * » permet de créer automatiquement une variable croisée à partir de deux variables de la base de données (sur les variables croisées, voir notre « Repères »).
- Appuyer sur OK.
- Les résultats s'affichent dans RCommander. Pour les conserver, les copier sur une feuille Word. Pour une mise en page plus lisible, on peut la présenter en format paysage, et surtout en police Courier New pour un bon alignement des colonnes.

Exemple dans le cas d'un modèle simple (« avoir un niveau excellent ou non » en fonction de « être bachelier [BA_0] ou non » et « être normalien de Paris [EN_P] ou non » - données de Jérôme Krop sur des carrières d'instituteurs)

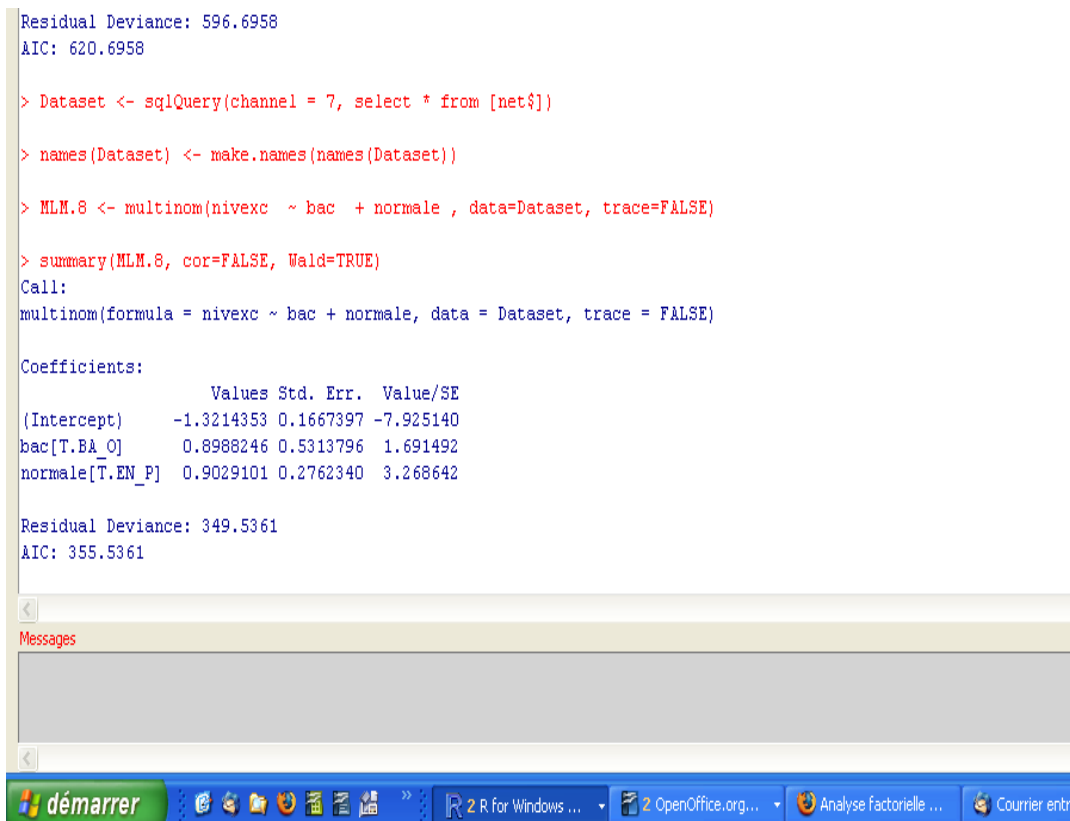
```
Residual Deviance: 596.6958
AIC: 620.6958

> Dataset <- sqlQuery(channel = 7, select * from [net$])
> names(Dataset) <- make.names(names(Dataset))
> MLM.8 <- multinom(nivexc ~ bac + normale, data=Dataset, trace=FALSE)
> summary(MLM.8, cor=FALSE, Wald=TRUE)
Call:
multinom(formula = nivexc ~ bac + normale, data = Dataset, trace = FALSE)

Coefficients:
              Values Std. Err. Value/SE
(Intercept)  -1.3214353 0.1667397 -7.925140
bac[T.BA_0]   0.8988246 0.5313796  1.691492
normale[T.EN_P] 0.9029101 0.2762340  3.268642

Residual Deviance: 349.5361
AIC: 355.5361
```

Messages



Exemple dans le cas d'un modèle où toutes les variables ne sont pas binaires : idem, en ajoutant le niveau d'enseignement (plusieurs modalités).

```
AIC: 355.5361
> MLM.9 <- multinom(nivexc ~ bac + normale + nivens , data=Dataset, trace=FALSE)
> summary(MLM.9, cor=FALSE, Wald=TRUE)
Call:
multinom(formula = nivexc ~ bac + normale + nivens, data = Dataset,
         trace = FALSE)

Coefficients:
              Values Std. Err.  Value/SE
(Intercept)  -1.9943059  0.5169639  -3.8577278
bac[T.BA_O]   1.0769562  0.5667737   1.9001520
normale[T.EN_P] 0.7998199  0.2992843   2.6724414
nivens[T.ET_INS] 0.9303913  0.8540237   1.0894209
nivens[T.ET_NON] 0.8056251  0.5393206   1.4937778
nivens[T.ET_PRI] 0.6136439  0.5947547   1.0317597
nivens[T.ET_SEC] 0.4925371  0.9913315   0.4968439

Residual Deviance: 346.8107
AIC: 360.8107
```

Messages

6. Lire les résultats :

- La « valeur/SE » (valeur coefficient/standard error) doit être supérieure à 1,64 pour que le résultat soit significatif au seuil de 10 % [*], à 2 pour 5 % [**] et à 2,58 pour 1 % [***]. Cela s'entend en valeur absolue (une valeur inférieure à -2 est significative au seuil de 5 %).
- Le signe du nombre dans la colonne « Value » indique le sens de l'effet, positif ou négatif.
- L'effet doit être interprété par rapport à la modalité de référence (*cf.* notre « Repères »). Le logiciel choisit par défaut la modalité de référence selon *l'ordre alphabétique des codes* (ici, dans le deuxième cas, ET_INC est la variable de référence, car son code vient avant ET_INS, ET_NON, etc., par ordre alphabétique). Pour changer de référence, il faut donc recoder habilement... (ce qui peut être fait dans RCommander : voir le menu Données). Il en va de même pour la variable à expliquer : ici, on explique le niveau excellent (EXC_O) par rapport au niveau non excellent (EXC_N) parce que le O vient après le N dans l'alphabet... Attention donc à ne pas lire les résultats à l'envers.
- Note sur l'affichage des nombres très grands ou très petits :
 - « 5,314e+07 » désigne le nombre $5,314 \times 10^7$, ou encore 53 140 000. « e+quelque chose » désigne en général des très grands nombres.
 - « 5,314e-07 » désigne le nombre $5,314 \times 10^{-7}$, ou encore 0,0000005314 « e-quelque chose » désigne en général des nombres minuscules.

7. Calculer les *odds ratios*

- Ceux-ci (cf. notre « Repères ») aident à mieux se représenter l'importance comparée des effets. Malheureusement, RCommander ne les donne pas directement.
- Pour les coefficients positifs, le *odds ratio*, c'est l'exponentielle de la valeur du coefficient (qui, elle, nous est fournie). Cela peut se calculer par exemple sous Excel, en tapant dans une case la formule :
=EXP(*écrire ici le coefficient*)
puis en appuyant sur Entrée.
Par exemple =EXP(3,037029) donne un *odds ratio* d'environ 21 soit « 21 fois plus de chances ».
- Pour les coefficients négatifs, il faut utiliser une formule différente pour avoir un résultat directement lisible :
=1/EXP(*écrire ici le coefficient, y compris le signe -*)
puis en appuyant sur Entrée.
Par exemple =1/EXP(-3,037029) donne un *odds ratio* d'environ 21 soit « 21 fois moins de chances ».

8. Faire d'autres choses avec RCommander...

- 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 chi² (voir Statistiques>Tables de contingence).
- Certains graphiques sont également très intéressants (« Boîte de dispersion » pour des données quantitatives par exemple).
- Voir notre tutoriel pour l'analyse des correspondances multiples, en annexe de la page <http://www.quanti.ihmc.ens.fr/document.php?id=123>