AIDE MÉMOIRE DU PACKAGE NLME DE R

D. CONCORDET

R est un logiciel de statistique professionnel qui permet d'effectuer un grand nombre d'analyses. Il est composé d'un cœur qui est capable d'interpréter et d'exécuter des commandes. Les commandes les plus simples sont présentes dans le cœur du logiciel. Les autres, plus ou moins spécifiques du type d'analyse à effectuer, sont décrites dans des bibliothèques nommées "package". La bibliothèque qui contient les commandes permettant l'analyse des modèle Non Linéaire à Effets Mixtes est nommée "nlme". L'objet de cette feuille est de proposer la description d'une sélection de commandes de R ainsi que des commandes du package nlme.

1. Quelques commandes générales de R

- `<-` (*ie* inférieur suivi de moins) est l'opérateur d'affectation.
- `Porc<-read.table("C:\\Documents and Settings\\Didier\\Bureau\\porc.txt",header=TRUE)` stocke les données contenues dans le fichier `C:\\Documents and Settings\\Didier\\Bureau\\porc.txt` dans le jeu de données reconnu par R nommé `Porc`. L'option `header=TRUE` permet de déclarer que la première ligne du fichier porc.txt contient le nom des variables contenues dans chacune des colonnes de ce fichier.
- `attach(Porc)` permet de charger en mémoire le jeu de données `Porc` rendant ainsi toutes les variables de ce jeu de données directement accessibles.
- `detach(Porc)` décharge de la mémoire de R le jeu de données `Porc`.
- `fix(Porc)` charge le jeu de données `Porc` dans l'éditeur de R et permet ainsi la modification de quelques données.
- `library(biblio)` charge en mémoire les fonctions de la bibliothèque nommée `biblio`.
- `fac<-as.factor(fac)` déclare la variable fac comme facteur.
- `plot(x,y)` permet de représenter la variable y en fonction de la variable x.

Date: 1/10/2007.

2. Quelques commandes de nlme

- `frame1<-groupedData(formula, data, outer, inner)`

  Cette commande permet de structurer les données et faciliter leur analyse. Son utilisation requiert une bonne compréhension des différentes sources de variation de la variable réponse étudiée.

  `formula` est une formule de la forme $Y\sim$ *covariable* | *group* où Y est la variable réponse, *covariable* est une variable explicative qui peut être soit un régresseur soit un facteur et *group* est le facteur de répétition (en général l'individu). Lorsqu'il n'y a pas de covariable, on remplace covariable par `1`. Lorsque le facteur fac_1 est hiérarchisé dans les niveaux du facteur *group*, il est possible de le déclarer de deux façons :

  1°) $Y\sim$ *covariable* | *group*/fac_1 ou

  2°) `inner`=$\sim fac_1$.

  Les hiérarchies multiples sont déclarées avec le symbole slash (/) par exemple, *group*/fac_1/fac_2 permet de déclarer que fac_2 est niché dans fac_1 lui même niché dans *group*. Les facteurs dans lesquels le facteur group est niché sont déclarés avec la commande `outer`. Par exemple, `outer`=$\sim fac_1 * fac_2$ permet de déclarer que le facteur group est niché dans le croisement des facteurs fac_1 et fac_2.

  `data` est le nom du jeu de données pour lequel cette organisation doit être appliquée.

  `frame1` est le nom de l'objet dans lequel cette organisation est stockée.

- `plot(frame1)` représente graphiquement les données organisées comme décrit dans frame1.

- `ana1<-lme(modèle fixe, data, random)` permet d'estimer les paramètres d'un modèle linéaire à effets mixtes. Les résultats de cette estimation sont stockées dans l'objet `ana1`.

  `modèle fixe` contient une description de la partie fixe du modèle. C'est donc une expression de la forme $Y \sim 1 + reg_1 + reg_2 + fac_1 + fac_2 + reg_1 * fac_1$ où reg_1 et reg_2 sont des régresseurs et fac_1 et fac_2 des facteurs.

  `data` est le nom du jeu de données ou du frame sur lequel on souhaite effectuer cette analyse.

  `random` est une expression de la forme `random`=$\sim 1+reg_1+reg_2+fac_1+fac_2+reg_1*fac_1|$`group` qui permet de déclarer la partie aléatoire du modèle. Les variables explicatives reg_1, reg_2, fac_1, fac_2 et `group` ont le même sens que précédemment.

On peut utiliser la notation des / vue dans la définition des frames pour décrire une organisation spécifique.

- `summary(ana1)` affiche les résultats de l'analyse stockée dans `ana1`
- `intervals(ana1)` calcul et affiche les intervalles de confiance et de prédictions de paramètres et des effets aléatoires du modèle décrit dans `ana1`
  `plot(intervals(ana1))` représente graphiquement ces intervals
- `plot(ana1)` représente graphiquement les résidus de l'analyse `ana1` en fonction des valeurs prédites.
- `qqnorm(ana1, ~resid(., type = "p") , abline = c(0, 1))` fait le QQplot des résidus de `ana1`
- `qqnorm(ana1, ~ranef(.))` fait le QQplot des effets aléatoires de `ana1`
- `anova(ana1,ana2)` fait le test de rapport de vraisemblance des modèles décrits dans `ana1` et `ana2`. Attention, ces modèles doivent être emboîtés.

3. Un exemple commenté

Afin de bien séparer les commandes, les commentaires et les sorties du logiciel nous utiliserons pour cet exemple des polices de caractères différentes. `Les commandes seront en police courrier et elles commenceront toutes par l'invite de commande de R représentée par le symbole >`. Les commentaires seront en texte normal et ils seront précédés du symbole #, `alors que les sorties du logiciel seront, comme les commandes, en courrier mais sans invite de commande`. Maintenant, l'histoire ...

Un même dose d'un médicament a été administrée par voie intraveineuse à trente patients. Afin d'étudier la dispersion des évolutions des concentrations au cours du temps de ce médicament, des prélèvements de sang ont été réalisés à des instants choisis. Après dosage, on dispose des concentrations de médicament pour chacun de ces instants de prélèvements. Par convention, on décide que l'instant 0 est l'instant auquel le médicament est administré. Le fichier "exemple.txt" contient trois colonnes nommées : Ind, Temps, concentration qui contiennent respectivement l'identification du patient, l'instant de prélèvement, la concentration.

```
>ex1<-read.table("C:\\Documents and Settings\\Didier CONCORDET\\Bureau\\M2R sept 07\\exemple.txt",header=TRUE) # charge les données dans le jeu de données nommés ex1
>ex1 #présente les données sous forme de liste
     Ind  Temps  concentration
1     1     1            44.52
2     1     2            47.96
3     1     5            27.38
4     1    10            16.61
5     1    20             7.66
...
 > attach(ex1) # charge le jeu de données ex1 en mémoire et le rend donc utilisable
> plot(Temps,log(concentration))# le graphique (cf 1 représente le log des concentrations en fonction du temps
 plot(Temps[Ind==1],log(concentration[Ind==1])) # le graphique (cf 2 représente le log des concentrations en fonction du temps mais simplement pour l'individu Ind = 1
 > library(nlme) # charge la bibliothèque nlme en mémoire et rend donc ses commandes utilisables
> Ind<-as.factor(Ind) # déclare Ind comme facteur.
```

Vous avez compris que la variable réponse Y est la variable log(concentration), que le

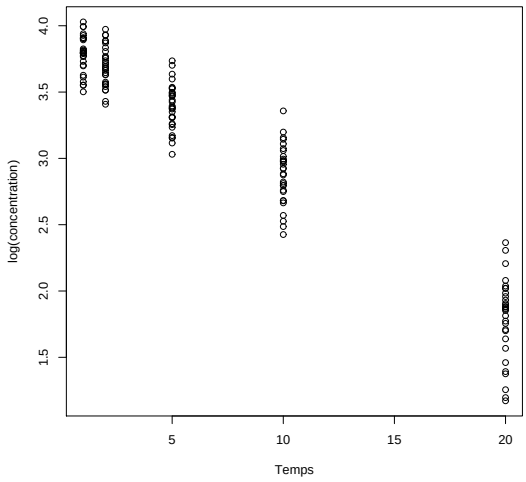

FIGURE 1. le log des concentrations à vaguement l'air d'évoluer linéairement avec le temps. Il y a une hétéroscédasticité marquée : la variance du log de la concentration semble augmenter avec le temps.

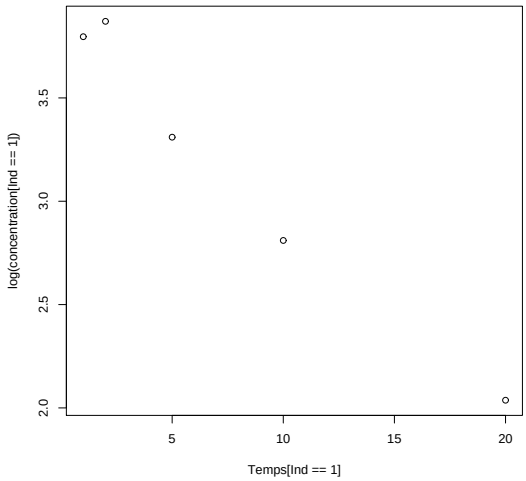

FIGURE 2. Pour cet individu le log des concentrations à l'air d'évoluer linéairement avec le temps mais sans hétéroscédasticité. On pourrait suspecter que l'hétéroscédasticité vue sur le graphique 1 soit due à des évolutions différentes entre individus. Peut-être qu'un modèle linéaire à effets mixtes pourrait décrire cela ?

temps est un régresseur, et qu'on dispose d'une courbe par Individu. La commande suivante permet de déclarer cette organisation et de la stocker dans le frame nommé ex2.

```
> ex2<-groupedData(log(concentration)~Temps|Ind, ex1)
```

`>ex2` #présente les données sous forme de liste. On peut remarquer sur cet exemple que les options de commandes `inner` et `outer` peuvent être omises lorsque la description de la structure ne les requiert pas.

```
Grouped Data:  log(concentration) ~ Temps | Ind
```

```
     Ind  Temps  concentration
1     1      1          44.52
2     1      2          47.96
3     1      5          27.38
4     1     10          16.61
5     1     20           7.66
...
> plot(ex2) # donne le graphique 3. En regardant ces petit graphiques, on arrive à se
```

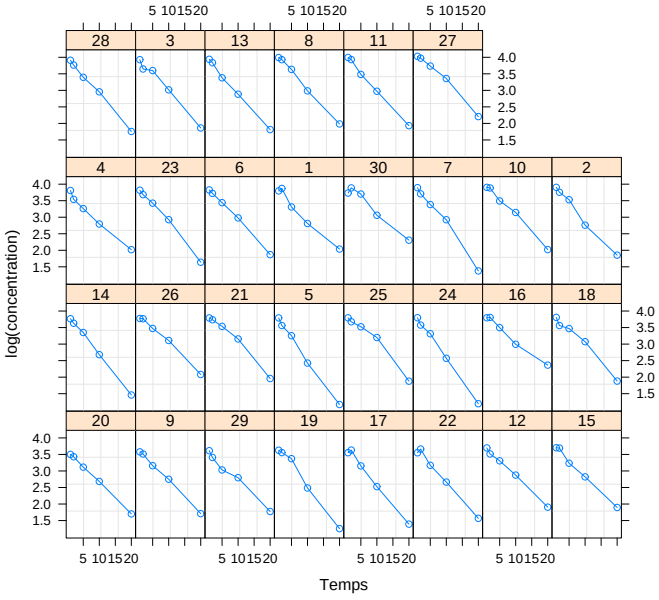

FIGURE 3. Représente pour chaque individu l'évolution au cours du temps du logarithme des concentrations. Le numéro de l'individu est indiqué juste au dessus de chaque graphique.

convaincre de la linéarité de la variation du log des concentrations avec le temps. Aussi, voilà un modèle qui pourrait être adopté au moins en première intention :

$$Y_{ij} = A_i + B_i t_{ij} + \varepsilon_{ij},\ j = 1 \dots 5,\ i = 1 \dots 30,$$

où Y_{ij} est le log de la $j^{\text{ème}}$ concentration mesurée sur le $i^{\text{ème}}$ individu à un instant t_{ij} après l'administration de la dose de médicament,
les vecteurs (A_i, B_i) sont indépendants, gaussiens, de moyenne (a, b) et de variance-covariance

$$\Sigma = \begin{pmatrix} \sigma_A^2 & C_{AB} \\ C_{AB} & \sigma_B^2 \end{pmatrix},$$

les ε_{ij} sont des variables aléatoires mutuellement indépendantes de même loi $\mathcal{N}(0, \sigma^2)$. Elles sont par ailleurs indépendantes des $(A_i, B_i)_i$. Ce modèle peut se ré-écrire de la façon

suivante :

$$Y_{ij} = \underbrace{a + bt_{ij}}_{\text{partie fixe du modèle}} + \underbrace{\eta_i^A + \eta_i^B t_{ij}}_{\text{partie aléatoire du modèle}} + \underbrace{\varepsilon_{ij}}_{\text{résidu}}, \ j = 1 \dots 5, \ i = 1 \dots 30,$$

où

$$\left(\eta_i^A, \eta_i^B\right)_i \sim_{\text{iid}} \mathcal{N}(0, \Sigma).$$

Ce dernier modèle s'écrit de la façon suivante dans nlme :

```
> ana1<-lme(log(concentration)~ 1+Temps,ex2, random =~ 1+Temps | Ind)
```

log(concentration)~ 1+Temps est la partie fixe du modèle et random =~ 1+Temps | Ind traduit la partie aléatoire du modèle. Les résidus sont implicitement supposés gaussiens homoscédastiques. Les résultats de l'analyse sont stockés dans ana1.

```
> ana1 # renvoie les résultats de l'analyse
Linear mixed-effects model fit by REML
Data: ex2
Log-restricted-likelihood: 75.1305 # log vraisemblance restreinte
Fixed: log(concentration) ~ 1 + Temps
(Intercept) Temps
3.9076723 -0.1050142 # â et b̂


Random effects:
Formula: ~ 1 + Temps | Ind Structure: General positive-definite, Log-Cholesky
parametrization

StdDev Corr
(Intercept) 0.12739881 (Intr) # √σ̂²_A
Temps 0.01359827 -0.045 # √σ̂²_B et corrélation entre A et B = C_AB/√(σ̂²_A σ̂²_B)
Residual 0.09112162 # écart-type résiduel √σ̂²

Number of Observations: 150 # nombre total d'observations
Number of Groups: 30 # nombre total d'individus
```

La commande summary donne en plus les écarts-type des estimations :

```
> summary(ana1)
Linear mixed-effects model fit by REML
Data: ex2
AIC BIC logLik
```

```
-138.261 -120.2777 75.1305

Random effects:
Formula:  ~ 1 + Temps | Ind Structure:  General positive-definite, Log-Cholesky
parametrization
StdDev Corr
(Intercept) 0.12739881 (Intr)
Temps 0.01359827 -0.045
Residual 0.09112162

Fixed:  log(concentration) ~ 1 + Temps
Value Std.Error DF t-value p-value
(Intercept) 3.907672 0.025741947 119 151.80174 0
Temps -0.105014 0.002703931 119 -38.83759 0
Correlation:
(Intr)
Temps -0.162

Standardized Within-Group Residuals:
Min Q1 Med Q3 Max
-1.83800816 -0.57519541 -0.07883175 0.59921714 2.49604903# extrêmes et quar-
```

tiles des résidus $\hat{\varepsilon}_{ij}$

```
Number of Observations:  150
Number of Groups:  30
```

Si vous désirez en plus les intervalles de confiance de tous les paramètres du modèle, tapez :

`> intervals(ana1)`# les sorties obtenues avec cette commande ne sont pas reproduites ici.

`> plot(ana1)` # représente les résidus en fonction des valeurs prédites par le modèle (cf graphique 4).

`>qqnorm(ana1, ~resid(., type = "p") , abline = c(0, 1))`# renvoie le QQplot des résidus (cf graphique 5).

`>qqnorm(ana1, ~ranef(.))` # renvoie le QQplot des $\hat{A}_i$ et $\hat{B}_i$ (cf graphique 6).

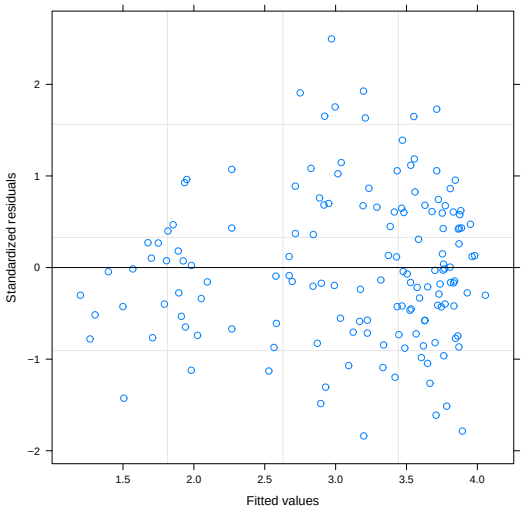

FIGURE 4. Les résidus en fonction des valeurs prédites par le modèle. Ce graphique est utile pour détecter une hétéroscédasticité. Dans cet exemple, on peut hésiter et conclure à l'homoscédasticité ou voir un cône et conclure à l'hétéroscédasticité. Une solution, au moins pour cet exemple, consiste à écrire un modèle non linéaire. Nous verrons en cours comment traiter cela.

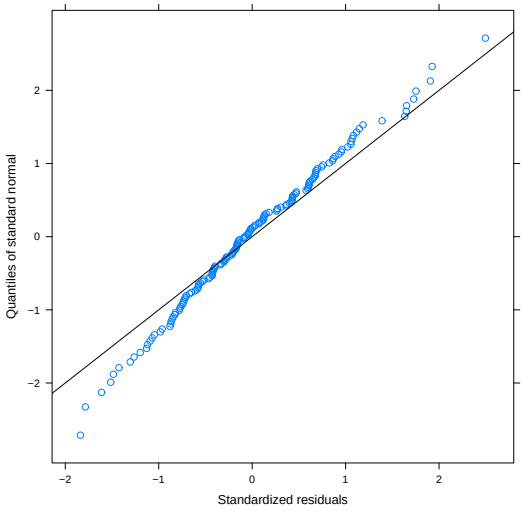

FIGURE 5. On observe des points qui ondulent autour d'une droite : l'hypothèse de normalité des résidus semble ici raisonnable.

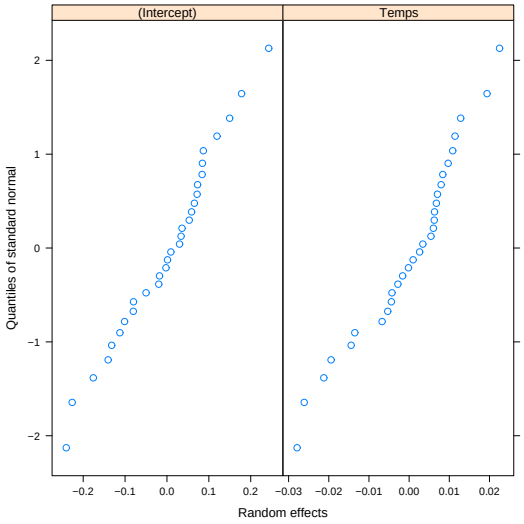

FIGURE 6. L'hypothèse de de normalité des effets aléatoires semble raisonnable.