Introduction à R - Régression logistique

Le but de cette séance est d'apprendre à utiliser efficacement le langage R et son environnement. Après avoir appris à effectuer les commandes de base, vous verrez les rudiments de la programmation en langage R, qui est très proche de nombreux autres langages de calculs comme Matlab, Octave, Scilab.

Vous pourrez vous aider de l'aide en ligne, ainsi que des documents suivants :

⋆⋆⋆ http://zoonek2.free.fr/UNIX/48_R/all.html
⋆⋆ http://cran.r-project.org/doc/contrib/Paradis-rdebuts_fr.pdf
⋆⋆ http://freakonometrics.hypotheses.org/category/r
⋆ http://www.burns-stat.com/pages/Tutor/R_inferno.pdf
⋆ http://cran.r-project.org/doc/manuals/R-intro.html

Remarque : Pour ceux familiers avec Matlab/Octave, une correspondance syntaxique est disponible
http://mathesaurus.sourceforge.net/octave-r.html

Parmi les ouvrages publiés sur R, les deux livres en français [1] et [2] peuvent être conseillés.

- Découverte de R -

Ligne de commande, aide en ligne et manuel

Basiquement, R a toutes les fonctionnalités d'une calculatrice moderne.

1. Essayer quelques commandes de bases, comme par exemple :

```
a = runif(1);M <- a*matrix(c(1:3,rep(4, 3)), ncol = 3, nrow = 2);ls()
```

2. A l'aide de l'aide en ligne `help('ls')` trouver comment effacer toutes les variables à la fois.
3. Trouver à quoi sert la fonction `mosaicplot`, et lancer les exemples fournis par l'aide (penser à utiliser `require(datasets)` si besoin).

Utilisation d'un éditeur

En matière d'ergonomie, la ligne de commande R montre vite ses limites. Il est indispensable de taper son code dans un autre éditeur, puis de les exécuter grâce à la commande `source` (ou bien en utilisant l'editeur de texte du logiciel, type Rstudio).

4. Récupérer sur le site ://josephsalmon.eu/index.php?page=teaching_12_13&lang=fr les fichiers loisAPI.R et introduction.R et les lancer dans R.
5. Trouver quelle est la définition de la médiane utilisée par R pour des vecteurs de taille impaire, en utilisant `getS3method` par exemple.

Fonctions

Les fonctions sont en R des objets comme les autres, qui sont affectées de la même façon.

6. Écrire et tester une fonction permettant de calculer la factorielle d'un entier positif. On pourra d'abord écrire une fonction récursive, puis utiliser une boucle.

Gestion des packages

Le chargement d'un package (=module complémentaire) se fait par la commande `require`.

7. Récupérer sur le site ://josephsalmon.eu/index.php?page=teaching_12_13&lang=fr le fichier `simpleRegAPI.R`, et regarder ce qu'il contient.

Graphiques

La commande de base pour les graphiques est `plot`. Par défaut, elle ne relie pas les points entre eux.

8. Regarder ce que fait la commande `lines`. Représenter sin et cos sur le même graphe.
9. Utiliser `persp` pour afficher la fonction sinus cardinale en 2D.
10. Générer et afficher 100 échantillons suivant des lois normales bi-dimensionelles, avec des moyennes $(0,0),(0,1)$ et $(1,0)$.
11. Illustrer graphiquement la loi forte des grands nombres pour les variables de Bernoulli $B(3/4)$.
12. Afficher la fonction de répartition théorique et empirique pour des variables gaussiennes centrées réduites. Idem pour la densité. Pour les versions empiriques on fera évoluer le nombre de variables générées.
13. Afficher une boite à moustache avec la fonction `boxplot` pour un échantillon de 100 variables gaussiennes.

Gestion des données

En plus des classiques tableaux, vecteurs et matrices, R possède deux structures de données utiles pour manipuler des données numériques : `list` et `data.frame`. Ces données peuvent être chargés grâce à la commande `read.table`. R contient aussi dans sa distribution quelques jeux de données que l'on utilisera pour illustrer les algorithmes vus en cours. Par ailleurs, les données auxquelles il est fait référence dans [3] sont disponible sur le site dans les références.

14. Regarder dans le manuel ce que sont les data frames.
15. Exécuter les commandes suivantes, et comprendre ce qui se passe :

```
data(); attach(cars); plot(speed,dist)
heart_disease=
 read.table("http://www-stat.stanford.edu/~tibs/ElemStatLearn/datasets/SAheart.data",
 sep=",",head=T,row.names=1)
plot(heart_disease[,-5], pch = 21,
 bg = c("red", "green","blue")[as.numeric(heart_disease$famhist)])
```

Quel(s) graphique(s) est-il pertinent de faire pour les représenter ? Lancer le même genre de commande sur les données "iris".

16. La régressions linéaire est gérée grâce à la commande `lm`. Executer alors les commandes suivantes

```
attach(cars);summary(cars); plot(speed,dist)
reglin<-lm(dist ~ speed); abline(reglin,col = "red" ); summary(reglin)
```

- RÉGRESSION LOGISTIQUE -

Considérons la classification binaire : $Y_i \in \{-1,1\}$ est expliqué par p régresseurs $X_i^1,\dots,X_i^p$.

Simulation

On prend $p=2$, et l'on suppose que (X_i^1,X_i^2) est uniformément répartis dans le carré $[0,1]\times[0,1]$, et que la loi conditionnelle de Y_i sachant (X_i^1,X_i^2) est donnée par :

$$\begin{cases} \mathbb{P}(Y_i=1)=\alpha & \text{si} \quad 2X_i^1+X_i^2<1.5, \\ \mathbb{P}(Y_i=1)=\beta & \text{si} \quad 2X_i^1+X_i^2>1.5, \end{cases}$$

où α et β sont deux paramètres sur lesquels on jouera un peu pour tester les méthodes ;

17. Chercher comment utiliser la fonction `glm` pour la régression logistique. Comprendre l'exemple suivant

```
X=c(1,2,3,4,5,6); Y=exp(X)+30*runif(length(X)); BASE = data.frame(X,Y)

plot(BASE,pch=19,cex=2)
reg = lm(Y~X,data=BASE)
abline(reg,col="red")

par(lwd=9,bg="lightcyan",mfcol=c(1,2))

reg1 = glm(Y~X,data=BASE,family=gaussian(link="identity"))
summary(reg1)
newx=seq(.1,6.4,by=0.1)
predY = predict(reg1,newdata=data.frame(X=newx))
plot(BASE,pch=19,cex=2)
lines(newx,predY,lwd=2,col="blue")
abline(reg,col="red")

reg2 = glm(floor(Y)~X,data=BASE,family=poisson(link="log"))
summary(reg2)
newx=seq(.1,6.4,by=0.1)
predY = predict(reg2,newdata=data.frame(X=newx),type = "response")
plot(BASE,pch=19,cex=2)
lines(newx,predY,lwd=2,col="blue")
abline(reg,col="red")
```

18. Afficher la fonction de régression logistique dans le cas unidimensionnel, pour des étiquettes (*i.e.,* les Y_i) générées selon la loi
$$\mathbb{P}(Y_i = 1|X = x) = \frac{\exp(a + bx)}{1 + \exp(a + bx)},$$
avec $a = -0.5$ et $b = 6$.
19. Construire une fonction R appelée `rexemple` prenant pour arguments n, α, et β, et renvoyant un n-échantillon $\left(X_i^1, X_i^2, Y_i\right)_{1\leq i\leq n}$ de l'exemple simulé sous la forme de liste de trois éléments : `list(x1,x2,y)`, où `x1,x2` et `y` sont des tableaux.
20. Vérifier sur cet exemple la convergence de l'estimateur de régression logistique (par exemple pour $\alpha = 0.9$ et $\beta = 0.1$). A partir de combien de points à peu près fournit-il un bon classifieur ?

Donneées réelles

L'exemple suivant considére des données réelles sur l'explication du risque d'attaques cardiaques par des facteurs comme l'âge, le tabagisme, etc., dont les données ont été récupérées à la questions 15.

21. Effectuer une régression logistique pour expliquer le risque d'attaque cardiaques par les facteurs donnés, en ignorant la variable `famhist`. Interpréter les résultats de la régression (*cf.* page 122).

Références

[1] P-A. Cornillon, A. Guyader, F. Husson, N. Jégou, J. Josse, Maela Kloareg, E. Matzner-Løber, and L. Rouvière. *Statistiques avec R*. Didact Statistiques. Presses Universitaires de Rennes, 2010. 1

[2] P-A. Cornillon and E. Matzner-Løber. *Régression avec R*. Springer, Collection Pratique R, 2011. 1

[3] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning*. Springer Series in Statistics. Springer, New York, second edition, 2009. http://www-stat.stanford.edu/~tibs/ElemStatLearn/. 2