TP 1 : premiers pas avec R et RStudio

L2 MIASHS - Université Lyon 2 - 2017/2018

Responsable : Julien Ah-Pine

1 R et Rstudio

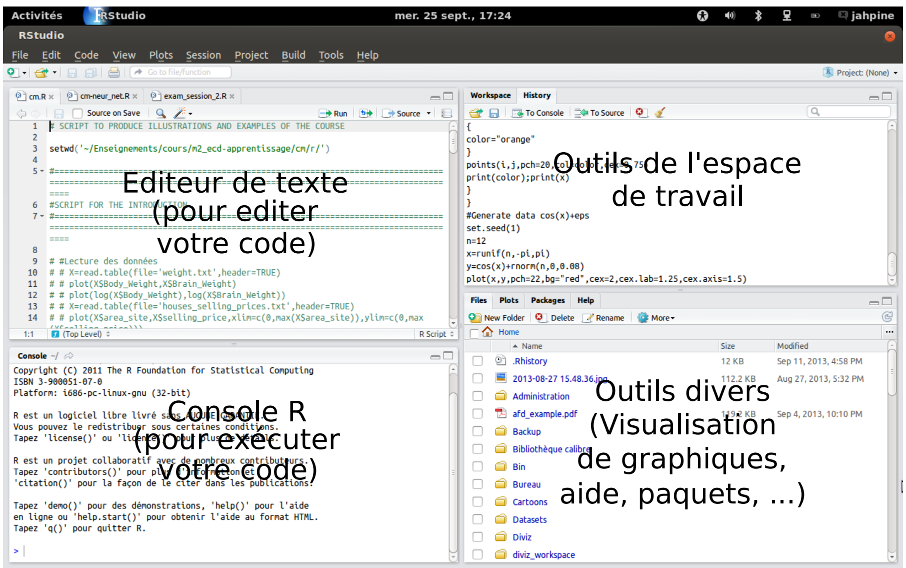
R (`http://www.r-project.org/`) est un langage de programmation et un environnement pour la statistique computationnelle et la visualisation graphique de données. Il est un dérivé du langage S développé initialement par John Chambers et dédié aux statisticiens. Il existe de nombreuses boîtes à outils ou librairies (“packages”) qui permettent à la communauté scientifique de mettre en oeuvre des méthodes statistiques classiques et nouvelles. Mais le succés de R en fait un outil de plus en plus utilisé non seulement par des universitaires mais également par des professionnels dans le cadre des domaines comme l’apprentissage automatique (“machine learning”), la fouille de données (“data-mining”), la science des données (“data science”), l’économétrie, la sociologie, la psychologie, ...

R peut être utilisée de façon rudimentaire via une console d’évaluation intéractive (console REPL “read-evaluate-print-loop”). Mais RStudio (`http://www.rstudio.com/`) est un environnement de développement (IDE) dédié à R qui est multiplateforme (Linux, Windows, Mac). Cet environnement permet de travailler avec R et ses graphiques de façon plus intéractive, d’écrire du code R avec un éditeur de texte proposant de nombreuses fonctionnalités, d’organiser votre code et de gérer plusieurs projets à la fois. . .

R et RStudio sont open-source. Vous pouvez donc les installer gratuitement sur votre machine. L’objectif de ce TP est de vous familiariser avec l’interface de RStudio et de pratiquer des commandes utiles dans le cadre de méthodes d’algèbre linéaire.

2 Interface de RStudio

1. Lancez RStudio (Menu `Démarrer` . . .)
2. L’interface de RStudio se compose par défaut de 4 fenêtres :



3 Création et manipulation de vecteurs

3. Dans la **console** tapez :

```
> v1=c(1,-2,-3)
```

Puis tapez :

```
> v1
```

4. Dans l'**éditeur de texte** tapez :

```
#Création de trois vecteurs
v1=c(1,-2,-3)
v2=c(2,3,-1)
v3=c(3,2,1)
```

Puis sélectionnez (avec la souris ou le clavier à l'aide de la touche `Shift`) les trois dernières lignes et faites `Ctrl + Entrée`. Vous pouvez donc entrez des commandes dans l'éditeur de texte et lancez celles-ci par bloc à la console qui les exécute.

5. Enregistrez le fichier texte sur votre disque et donnez lui le nom de `tp1.R`. Ce fichier est un script dans lequel vous pourrez enregistrer toutes vos commandes et exécuter celles-ci ultérieurement. Ainsi, sauvegardez régulièrement ce fichier en tapant `Ctrl + s`. Par ailleurs, vous remarquerez que le caractère # permet d'entrer des commentaires. N'hésitez pas à vous en servir pour décrire les commandes ou faire référence à l'exercice que vous êtes en train de traiter.

6. Entrez, exécutez et commentez les commandes suivantes :

```
#Manipulations de vecteurs et affectations
v1+v2
v4=v1-v2
v5<-v2+v3
v1*v3->v6
```

7. Entrez, exécutez et commentez les commandes suivantes :

```
#Accès au contenu des cellules des vecteurs
v6[2]
v6[c(1,3)]
```

8. Dans la fenêtre relative à l'"**Help**", cherchez l'aide relative à la commande `seq`. A quoi sert cette commande ?

9. Entrez, exécutez et commentez les commandes suivantes :

```
#Création de séquences et utilisation
seq(1,15,2)
s=seq(1,3)
v6[s]
```

10. Entrez, exécutez et commentez les commandes suivantes les unes après les autres :

```
#Création de 2 vecteurs dans R2
u1=c(2,1)
u2=c(1,3)
#Création d'un plan et ajout d'une grille
plot(NA, xlim=c(0,5), ylim=c(0,5), xlab="x1", ylab="x2")
grid(col = "red", lty = "dotted", lwd = 1)
#Représentation des vecteurs dans le plan
arrows(0,0,u1[1],u1[2],col="black")
arrows(0,0,u2[1],u2[2],col="red")
arrows(0,0,u1[1]+u2[1],u1[2]+u2[2],col="green")
```

11. Calculez le vecteur $\mathbf{u_3} = 2\mathbf{u_1} - 0.5\mathbf{u_2}$. Stockez dans la variable a la première composante de $\mathbf{u_3}$ et dans la variable b la deuxième composante de $\mathbf{u_3}$. En utilisant ces deux variables, représentez le vecteur $\mathbf{u_3}$ dans le plan précédent par une flèche bleue.

4 Création et manipulation de matrices

12. Entrez, exécutez et commentez la commande suivante :

```
#Creation d'une matrice
V=matrix(c(1,-2,-3,2,3,-1,3,2,1),nrow=3)
```

Comment le vecteur `c(1,-2,-3,2,3,-1,3,2,1)` est-il transformé en matrice ? Essayez alors la commande suivante :

```
matrix(c(1,-2,-3,2,3,-1,3,2,1),nrow=3,byrow=TRUE)
```

13. Entrez, exécutez et commentez les commandes suivantes :

```
V=matrix(c(1,-2,-3,2,3,-1,3,2,1),nrow=2)
V
```

Que se passe t-il ?

14. Entrez, exécutez et commentez les commandes suivantes :

```
#Autres façons pour créer une matrice
V=matrix(c(v1,v2,v3),nrow=3)
W=rbind(v1,v2,v3)
cbind(v1,v2,v3)
```

Que représentent les colonnes de $\mathbf{V}$?

15. Entrez, exécutez et commentez la commande suivante :

```
#Transposée d'une matrice
t(W)
```

16. Entrez, exécutez les commandes suivantes les unes après les autres et dites à quoi correspondent ces opérations :

```
#Manipulations de matrices
V+W
V-W
V*W
V%*%W
```

17. Calculez la matrice $\mathbf{Z} = 2\mathbf{W}\mathbf{V}^\top$.

18. Entrez, exécutez les commandes suivantes les unes après les autres et dites à quoi correspondent ces opérations :

```
#Accés et modifications de cellules, lignes, colonnes d'une matrice
V
V[1,3]
V[1,3]=100
V[1,3]
V
V[,2]
V[3,]
V[,2]=c(1,2,3)
```

5 Applications à la résolution de systèmes d'équations

18. Résolvez le système d'équation linéaire suivant (Exercice du TD) :

$$\left\{\begin{array}{lcl} x+y+2z &=& 9 \\ 2x+4y-3z &=& 1 \\ 3x+6y-5z &=& 0 \end{array}\right.$$

Pour cela vous entrerez au préalable dans une matrice `A` les coefficients du système et dans un vecteur `b` les membres de droite des equations. On a alors le problème $\mathbf{Ax} = \mathbf{b}$ et on le résoud en utilisant la commande `solve` :

```
#Résolution d'un système d'équations linéaires Ax=b
solve(A,b)->x
```

19. La matrice inverse d'une matrice carrée $\mathbf{A}$ est une matrice $\mathbf{X}$ telle que $\mathbf{AX} = \mathbf{I}$. Ainsi défini, on voit que chaque colonne de $\mathbf{X}$ peut être vue comme la solution de trois systèmes d'équation ayant les même coefficients $\mathbf{A}$ mais correspondant à 3 membres de droite distincts à savoir $(1, 0, 0)$, $(0, 1, 0)$ et $(0, 0, 1)$. Pour vous en rendre compte entrez et exécutez les commandes suivantes :

```
solve(A,c(1,0,0))
solve(A,c(0,1,0))
solve(A,c(0,0,1))
#calcul de l'inverse de A
Ainv=solve(A)
A%*%Ainv
```

20. Précédemment on a résolu le problème $\mathbf{Ax} = \mathbf{b}$ en utilisant la commande `solve(A,b)`. On peut retrouver le résultat en appliquant la formule $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$. Quelle est la commande permettant de faire ce calcul ?

21. On étudie l'endomorphisme $f : \mathbb{R}^2 \to \mathbb{R}^2$ correspondant à la projection sur la 1ère bissectrice parallèlement à la 2ème bissectrice (cf CM). On suppose la base canonique $\{\mathbf{e_1}, \mathbf{e_2}\}$ de $\mathbb{R}^2$. Entrez la matrice $\mathbf{F}$ suivante qui représente l'endomorphisme f dans la base canonique :

```
#Matrice représentant la projection sur la 1ère bissectrice
#parallèlement à la 2ème bissectrice
F=matrix(0.5,nrow=2,ncol=2)
```

22. Soit $\mathbf{u_1} = (2, 1)$ et $\mathbf{u_2} = (1, 3)$. Déterminez les images $\mathbf{s_1} = f(\mathbf{u_1})$ et $\mathbf{s_2} = f(\mathbf{u_2})$ par calcul matriciel. Représentez par des flèches ces vecteurs dans le plan.

23. Soit le changement de base suivant $\epsilon_\mathbf{1} = \mathbf{e_1} + \mathbf{e_2}$ et $\epsilon_\mathbf{2} = \mathbf{e_1} - \mathbf{e_2}$. Déterminez la matrice de passage $\mathbf{P}_{\mathbf{e_i} \to \epsilon_\mathbf{i}}$ et à partir de cette dernière la matrice $\mathbf{P}_{\epsilon_\mathbf{i} \to \mathbf{e_i}}$. Déterminez ensuite l'expression de $\mathbf{F}$, de $\mathbf{u_1}$ et de $\mathbf{s_1}$ dans cette nouvelle base (cf CM).

6 Applications à la réduction d'un endomorphisme

24. Sotckez dans une variable `A` la matrice suivante :

$$\mathbf{A} = \begin{pmatrix} 2 & 0 & 4 \\ 3 & -4 & 12 \\ 1 & -2 & 5 \end{pmatrix}$$

25. Entrez, exécutez et commentez les commandes suivantes :

```
#Décomposition spectrale - Réduction des endomorphismes
eig_A=eigen(A)
eig_A$values
eig_A$vectors
```

Faites notamment le lien avec l'exemple du cours slide 143 à 149.

26. Entrez, exécutez et commentez les commandes suivantes :

```
#Normalisation de vecteurs
u0=c(-4,3,2)
u1=c(-4,0,1)
u2=c(2,1,0)
n0=sqrt(t(u0)%*%u0)
```

```
n1=sqrt(t(u1)%*%u1)
n2=sqrt(t(u2)%*%u2)
u0/n0
u1/n1
u2/n2
```

Comparez avec `eig_A$vectors`.