TP 2 - Programmation de suites

L1 Biologie
2013-2014

Quelques compléments sur les vecteurs :

v=c(1,4,7,2,8)	# créé le vecteur $(1,4,7,2,8)$, noté v.
v[4]	# renvoie la valeur de la 4ème coordonnée de v.
v[4]<- 1	# affecte une nouvelle valeur de la 4ème coordonnée de v.
c(v,5,1)	# concaténation du vecteur v et du vecteur $(5,1)$.
sum(v)	# renvoie somme de toutes les coordonnées de v.
cumsum(v)	# renvoie le vecteur des sommes cumulées des coordonnées de v.
prod(v)	# renvoie le produit des coordonnées de v.
cumprod(v)	# renvoie le vecteur des produits cumulées des coordonnées de v.
mean(v)	# renvoie la moyenne du vecteur.
max(v)	# renvoie la valeur maximale.
min(v)	# renvoie la valeur minimale.
sort(v)	# range les valeurs dans l'ordre croissant.
length(v)	# renvoie la taille du vecteur
rep(0,5)	# créé le vecteur nul de taille 5

Exercice :

1. Calculer la somme des 20 premiers entiers pairs.
2. Trier les coordonées d'un vecteur dans l'ordre décroissant.

Programmation de suites avec la boucle for :

1. Un exemple : la suite arithmétique : $U_0 = 0$, $U_{n+1} = U_n + 3$.
   On cherche à renvoyer dans un vecteur les 21 valeurs $U_0, U_1, ..., U_{20}$.

   Méthode 1 :
   ```
    U=rep(0,21)                # Vecteur nul de taille 21.
   for (k in 1 :20) {
   U[k+1]<-U[k]+3
   }
   ```
   Attention ! !
   On est obligé d'inclure la valeur initiale U_0 dans le vecteur afin de pouvoir initialiser la boucle. En conséquence, cela décale les indices. En effet, pour avoir la valeur de U_{14}, il faudra alors taper la commande U[15].

   Méthode 2 :
   ```
    U=0                        # On initialise notre valeur de départ.
    V=U                        # C'est notre futur vecteur qui va contenir toutes les
                               valeurs de la suite.

   for (k in 1 :20){
   U=U+3
   V=c(V,U)
   }
   ```

Remarque :
Lorsque la boucle contient plusieurs instructions, il faut les mettre sur des lignes différentes.

2. **Exercice :** Donner les 20 premiers termes de la suite géométrique :
   - de premier terme $U_0 = 0$ et de raison 2.
   - de premier terme $U_0 = 1$ et de raison 2.
   - de premier terme $U_0 = 1$ et de raison 0.5.

Etude de la suite logistique :

Cette suite est définie par la relation de récurrence suivante :

$$U_0 \in (0,1), \quad U_{n+1} = \lambda U_n(1 - U_n)$$

On va étudier son comportement suivant différentes valeurs de U_0 et de λ :

1. Etudier les 40 premiers termes des suites :
   - $U_{n+1} = \frac{1}{2}U_n(1 - U_n)$, pour $U_0 = \frac{1}{3}$ et $U_0 = \frac{2}{3}$
   - $U_{n+1} = \frac{3}{2}U_n(1 - U_n)$, pour $U_0 = \frac{1}{3}$ et $U_0 = \frac{2}{3}$
   - $U_{n+1} = \frac{5}{2}U_n(1 - U_n)$, pour $U_0 = \frac{1}{3}$ et $U_0 = \frac{2}{3}$

   Rassembler les termes dans un vecteur $V = (U_0, ..., U_{40})$.
   Pour plus de lisibilité, on pourra tracer le graphique des points (n, U_n) à l'aide de la fonction :

   plot(0 :40,V)

2. Même exercice avec la suite :

   $U_{n+1} = \frac{7}{2}U_n(1 - U_n)$, pour $U_0 = 0.1, 0.2, \dots, 0.9$.

3. Même exercice avec la suite :

   $U_{n+1} = \frac{9}{2}U_n(1 - U_n)$, pour $U_0 = \frac{1}{2}$ et $U_0 = \frac{\pi}{6}$.

4. Rédiger un court commentaire sur les différents cas, et leur interprétation. On pourra par exemple pour chaque cas :
   - Noter ce que vous observer mathématiquement.
   - Le cas est-il pertinent biologiquement ?
   - Si oui, quelle interprétation biologique pouvez-vous en faire ?

5. Faire varier soit-même les paramètres, afin d'essayer de comprendre ce qui se passe...

Pour aller plus loin : les fonctions

Au lieu de faire varier U_0 et λ dans la boucle, on peut faire une fonction, que l'on nommera "suite", qui prend en paramètre d'entrée les variables U_0 et λ. Le coeur de la fonction contient les même instructions que la boucle précédente. On appelle ensuite la fonction en écrivant :

$$\text{suite}(U_0,\lambda)$$

Voici un exemple concret à partir de notre cas suivant :

```
suite<-function(U0,lambda) {
U=U0
V=U
for (k in 1 :40){
U=lambda*U*(1-U)
V=c(V,U)
}
plot(0 :40,V)
 title(c(U , lambda))      # pour afficher sur le graphique les valeurs de U0 et de λ.
 return(V)                 # pour afficher les coordonnées du vecteur V à l'écran une fois la
                           fonction exécutée. Attention : toutes les instructions qui arrivent
                           après ne seront pas éxécutées.
}
```

Pour exécuter la fonction précédente, il suffit alors de taper la commande :

suite(1/2,1/2) # effectue la procédure précédente pour les valeurs $U_0 = \frac{1}{2}$ et $\lambda = \frac{1}{2}$.

Pour un meilleur rendu : le découpage des fenêtres

On peut aussi découper la fenêtre graphique afin de superposer plusieurs graphiques, par exemple :

```
windows()            (ou X11() sous linux)
par(mfrow=c(3,2))    # divise la fenêtre graphique en 3 lignes et 2 colonnes.
```

Les 6 premiers graphiques vont s'afficher sur les 3 x 2 cases. Au 7ème, tout est ré-initialisé. Par exemple, pour le cas 1 :

```
windows()
par(mfrow=c(3,2))
suite(1/2,1/2)
suite(3/2,1/2)
suite(1/2,3/2)
suite(3/2,3/2)
suite(1/2,5/2)
suite(3/2,5/2)
```

De même pour le 2 et le 3 de l'étude :

```
windows()
par(mfrow=c(3,2))
suite(0.1,7/2)
suite(0.3,7/2)
suite(0.5,7/2)
suite(0.7,7/2)
suite(0.8,7/2)
suite(0.9,7/2)
# # # # # # # # # # # #
par(mfrow=c(2,1))
suite(1/2,9/2)
suite(pi/6,9/2)
```