

LMAFY1101 - Exercices - Série 1

Utilisation de R

Pour chaque fonction que vous utiliserez, n'hésitez pas à **consulter l'aide** afin de savoir comment bien l'utiliser.

N'écrivez pas votre code directement dans la **Console**, tapez-le dans un script . Sauvegardez ce dernier régulièrement. Pour cela, vous pouvez utiliser le raccourci clavier **CTRL+s**.

Exercice 1

1. Créez le vecteur $(-2, -1, 0, 1, 2, 3)$ en utilisant la fonction `c()` puis avec la fonction `seq()`. Assignez ce vecteur à `x`.
2. En appliquant une transformation adéquate au vecteur `x` précédent, créez le vecteur $(0, 1, 2, 3, 4, 5)$.
3. Créez les vecteurs `y`: 5, 5, 5, 5, 5 et `z`: -1, -1, -1, -1, -1, 2, 1, 1, 1, 1, 1, 1, 1, 1 à l'aide de la fonction `rep()`.
4. Calculez `y + c(0,z)` et `y + z`.
5. *Modifiez* `z` en la privant de son dernier élément (càd `z`: -1, -1, -1, -1, -1, 2, 1, 1, 1, 1, 1, 1). Vous ne devez pas recréer `z`.
6. Créez le vecteur $(-2, -1, 0, 1, 2, 3, 5, 5, 5, 5, 5)$ à partir de `x` et de `y`.
7. Modifiez les 3 premiers éléments de `x` en une seule instruction pour que `x` devienne le vecteur $(7, 6, 5, 1, 2, 3)$.
8. Ordonnez le vecteur `x` grâce à la fonction `sort()`. Calculez ensuite `order(x)`. Comment peut-on ordonner `x` en utilisant cette dernière fonction ?
9. Créez, à l'aide de deux méthodes différentes, un vecteur `w` qui contient uniquement les valeurs positives de `z`. *Indice* : utilisez la fonction `ifelse()`.

Exercice 2

1. Créez un vecteur `taille` qui contient les valeurs suivantes pour la taille d'une série de personnes, en centimètres : 160, 176, 161, 165, NA, 168, 161, 174, 161, 159, 164, 169, 163, 163, NA, 172, 165, 164, 170, 163, 169, 184.
2. Comment vous demandez à R la longueur de `taille` ?
3. Comment vous demandez à R s'il y a des valeurs manquantes, NA, dans `taille` ?
4. Comment vous demandez à R combien de valeurs manquantes il y a dans `taille` ?
5. Comment vous pouvez obtenir la moyenne de `taille` ?
6. Comment est-ce que vous pouvez avec une seule commande obtenir le minimum, maximum, la moyenne et le nombre de NA (entre autres) de `taille` ?

7. Créez un nouveau vecteur `taille_ok` qui contient les valeurs de taille qui ne sont pas NA.
8. Donnez les valeurs de `taille_ok` sauf les 10 premières.
9. Les valeurs dans `taille_ok` sont actuellement données en centimètres. Modifiez `taille_ok` pour avoir toutes les valeurs en mètres.
10. Combien de personnes ont une taille supérieure à 1,65 m ?
11. Combien de personnes ont une taille entre 1,60 et 1,70 m ?

Exercice 3

On a mesuré le diamètre de 34 cylindres construits par les étudiants de l'EPL de Louvain-la-Neuve selon un modèle bien particulier. Voici les résultats, en centimètres.

```
4.03 4.05 3.96 4.09 4.28 4.04 4.18 4.23 4.14 4.12 4.03 3.94 4.02 4.08 4.13
4.04 3.93 4.08 4.37 4.07 4.11 4.03 4.00 3.97 4.01 4.09 4.06 3.92 4.19 3.96
4.48 4.24 4.06 3.98
```

1. Calculez la moyenne, l'écart-type, et le coefficient de variation.
2. Calculez le résumé en cinq nombres (min, Q1, mediane, Q3, max).
3. Calculez les quantiles 30% et 70%.
4. Qu'arrive-t-il à la moyenne et à la médiane si les valeurs 4.18, 4.23, 4.14 et 4.12 sont remplacées par les valeurs 4.68, 4.73, 4.64 et 4.62 ?
5. Dessinez un histogramme.
6. Dessinez un boxplot.

Exercice 4

Sachant que

```
taille <- c(167, 192, 173, 174, 172, 167, 171, 185, 163, 175)
nom <- c("Benjamin", "Hugo", "Emma", "Alex", "Tom", "Axel", "Alice",
        "Martin", "Robin", "Enzo")
```

pouvez-vous dire à quoi correspondent exactement les sorties des commandes suivantes **sans les exécuter** ? Vérifiez votre réponse par la suite.

```
sum(taille >= 172)
taille[order(taille)]
taille[order(-taille)]
rev(sort(taille))
nom[order(taille)]
nom[which(taille == min(taille))]
nom[taille %in% c(163, 192)]
```

Exercice 5

1. Créez les 3 vecteurs suivants :
 - Noms : `c("Victor", "Sandrine", "Jonathan", "Marie")`
 - Ages : `c(4,7,6,4)`
 - Tailles : `c(110,122,125,118)`
2. A l'aide de la fonction `data.frame` créez une base de données prenant ces 3 vecteurs comme variables.
3. En utilisant la fonction `order`, affichez le jeu de données par ordre alphabétique des noms.
4. En utilisant la fonction `subset`, trouvez le noms du (ou des) enfant(s) le(s) plus jeune(s).
5. En utilisant la fonction `order`, affichez les noms des enfants par ordre décroissant de taille.

Exercice 6

1. Créez le dataframe `daf` avec les colonnes `col1` : 10.9, 12.4, 11.9, 13.2, 11.1; `col2`: 8, 4, 2, 6, 10; `col3`: "Anne", "Michel", "Dominique", "Camille", "Stéphane"; et `col4`: 1, 1, 2, 2, 1.
2. Avec quelle commande pouvez-vous voir le contenu de `daf` sous forme de tableur.
3. Comment vous obtenez le nombre de lignes et de colonnes (les dimensions) de `daf` avec une seule commande ? Et comment pour les lignes et colonnes séparément ?
4. Demandez à R de vous donner la structure de `daf`.
5. Comment pouvez-vous obtenir la valeur de la troisième ligne dans la deuxième colonne de `daf` ?
6. Comment obtenez-vous un sommaire (summary) de toutes les colonnes de `daf` avec une seule commande ?
7. Comment vous obtenez tous les éléments dans la première colonne de `daf` ?
8. Demandez à R de vous donner les noms des colonnes et les noms des lignes.
9. Changez le nom de la colonne "col3" en "Nom".
10. Quel est le type de la quatrième colonne ? Que faut-il faire pour changer le type de cette colonne en `factor` ?
11. Changez les noms des niveaux de `col4` : 1 devient "non" et 2 devient "oui". *Indice* : utilisez la fonction `levels()`.
12. Divisez les valeurs dans la deuxième colonne par 10 et remplacez les valeurs actuelles de cette colonne par les résultats du calcul.
13. Comment peut-on extraire tous les cas avec une valeur supérieure à 0.5 dans la deuxième colonne ?
14. Comment peut-on extraire le contenu de la colonne `Nom` pour tous les cas avec une valeur supérieure à 11.5 dans la première colonne ?
15. Combien de cas ont une valeur entre 11.0 et 12.0 dans la première colonne ?
16. Calculez la somme de la colonne 2 pour les "oui" de la colonne 4

Exercice 7

1. Créez manuellement un dossier “data” sur votre machine. Téléchargez le fichier [climate.csv](#) et mettez-le dans “data”.
2. Changez le répertoire de travail R et placez-le dans “data”.
3. Ouvrez `climate.csv` dans RStudio, utilisez pour cela le volet **Files** de RStudio. Examinez le contenu du fichier.
4. Créez un dataframe `climate` en important les données qui se trouvent dans le fichier `climate.csv`. Vérifiez que votre espace de travail R contient le dataframe `climate`. Vous pouvez pour cela utiliser la fonction `ls()`.
5. Sauvegardez le dataframe `climate` dans un fichier RData sous le nom `climate.rda` dans le dossier `data`. Efface le dataframe `climate` de votre espace de travail (Global environnement). Vérifiez que ce dernier ne contient plus `climate`.
6. Chargez le contenu de `climate.rda` dans votre espace de travail. Le dataframe `climate` doit de nouveau être disponible.
7. Appliquez la commande `rm(list = ls())` afin d’effacer toutes les objets de l’espace de travail. Ce dernier est désarmé vide !

Exercice 8

1. Chargez la librairie `ggplot2`. Cette librairie contient le jeu de données (de type `tibble`) `diamonds`. Cette dernière contient des caractéristiques d’un certain nombre de diamants:
 - `cut` : qualité de la coupe du diamant (Fair, Good, Very Good, Premium, Ideal)
 - `price` : prix du diamant
 - `color` : couleur du diamant, de J (pire) à D (meilleur)
 - `clarity`: clarté du diamant, de I1 (pire) à IF (meilleur)
 -
2. À l’aide de la fonction `data.frame`, transformez `diamonds` en data frame. Affichez les 6 premières lignes de `diamonds`.
3. Combien d’observations sont reportées dans cette base de données? Combien y a-t-il de variables?
4. Allez voir l’aide afin de voir la description du jeu de données (càd le nom et la signification de chaque variable.)
5. A l’aide de la fonction `str`, décrivez la structure du jeu de données (Nom, type des variables ...)
6. Ajoutez une colonne donnant le prix du diamant en euros. (Utilisez le taux de change suivant : 1 euros = 1.23 dollar)
7. Créez un jeu de données comportant uniquement les diamants dont la qualité de découpe est satisfaisante, càd `cut = Fair`.
8. Créez un jeu de données comportant uniquement les diamants dont la qualité de découpe est au moins bonne, càd `cut ≠ Fair`, et dont le prix de vente est supérieur à 10.000\$.