

Correction 1

Entrainement 1

Question 1

Concernant les vecteurs, quelles affirmations parmi les suivantes sont vraies ?

Attention, plusieurs réponses sont possibles.

1. Les vecteurs peuvent contenir plusieurs types
2. La longueur d'un vecteur peut être obtenue via la fonction `length`
3. `seq` permet de créer un vecteur en répétant plusieurs fois un ou plusieurs éléments
4. `c` permet de créer un vecteur en spécifiant les différents éléments de ce dernier

Explication [Les vecteurs sont monotypes, ils ne peuvent donc contenir qu'un seul type donné.]

`seq` permet de générer une séquence de nombre également répartis dans un intervalle donné. La fonction pour répéter un ou plusieurs éléments un certain nombre de fois est `rep`.

`c` permet de créer un vecteur en spécifiant les éléments à l'intérieur. Si par exemple je veux créer le vecteur 1, 2, 3 : `a = c(1, 2, 3)`

Question 2

Quel sera le résultat affiché dans la console suite à l'exécution des lignes suivantes :

```
x <- c(rep(10, times=3), seq(1, 5, by=1), 1, 2, 3)
```

x

1. 10, 10, 10, 1, 5, 1, 2, 3

2. 10, 3, 1, 2, 3, 4, 5, 1, 2, 3

3. 10, 10, 10, 1, 2, 3, 4, 5, 1, 2, 3

4. 10, 10, 10, 1, 2, 3, 4, 5, 1, 1, 2, 3

Explication [Décomposons l'instruction :

`c(rep(10, times=3)` va répéter l'entier 10 trois fois, le début sera donc 10, 10, 10

`seq(1, 5, by=1)` va créer une séquence de 1 à 5, par 1. La suite sera donc 1, 2, 3, 4, 5

1, 2, 3 correspondra simplement à 1, 2, 3 à la suite.

Mis bout à bout, on obtient :10, 10, 10, 1, 2, 3, 4, 5, 1, 2, 3]

Question 3

Concernant les facteurs, quelles affirmations parmi les suivantes sont vraies ?

Attention, plusieurs réponses sont possibles.

1. les facteurs sont des vecteurs particuliers pour des données qualitatives

2. les facteurs ne peuvent contenir qu'une seule modalité

3. les différentes valeurs possibles d'un facteur peuvent être obtenues via la fonction `levels`

4. il n'est pas possible d'ordonner les niveaux d'un facteurs

Explication [Les facteurs sont des vecteurs permettant la manipulation de données qualitatives. Par conséquent, ils prennent en compte plusieurs modalités d'une variable qualitative. Les différentes modalités peuvent d'ailleurs être obtenus via la fonctions `levels`.

Il est tout à fait possible d'ordonner un facteur, via la fonction `ordered`.]

Question 4

La fonction pour transformer un vecteur en facteur est :

1. `as.numeric()`

2. `as.factor()`

3. `table()`

4. `to.factor()`

Explication [La fonction pour transformer un vecteur existant en facteur est `as.factor()`

`as.numeric()` : permet de transformer un vecteur en numérique

`table()` : permet de créer des tableaux croisés

`to.factor()` n'existe simplement pas !]

Question 5

Concernant les matrices, quelles affirmations parmi les suivantes sont vraies ?

Attention, plusieurs réponses sont possibles.

1. Une matrice ne peut contenir qu'un seul type

2. On peut accéder à un élément d'une matrice via son indice de ligne et de colonne

3. La fonction `length` permet d'avoir le nombre de lignes et de colonnes d'une matrice

4. Il est possible de créer une matrice de caractères

Explication [Une matrice peut être vue comme une sorte de tableau, on peut donc accéder à un élément via son numéro de ligne/colonne. Elle ne peut contenir qu'un seul type, qui peut être numérique, caractère, booléen, etc...]

Pour avoir les dimensions de la matrice (nombres de lignes et de colonnes), il faut utiliser la fonction `dim`

Question 6

Quel sera le résultat affiché dans la console suite à l'exécution des lignes suivantes :

```
1. m <- matrix(1,nrow=3,ncol=3)
```

```
vec <- c(1, 2, 3)
```

```
m[1, 3] = 3
```

```
m <- m + diag(vec)
```

```
t(m)
```

```
2.  [1] [2] [3]
```

```
[1,] 2  1  3
```

```
[2,] 1  3  1
```

```
[3,] 1  1  4
```

```
3.  [1] [2] [3]
```

```
[1,] 1  1  3
```

```
[2,] 1  2  1
```

```
[3,] 1  1  3
```

LearnerWorld

4. [1] [2] [3]

[1,] 1 1 1

[2,] 1 2 1

[3,] 3 1 3

5V. [1] [2] [3]

[1,] 2 1 1

[2,] 1 3 1

[3,] 3 1 4

Explication [Évaluons ligne par ligne :

on crée une matrice 3x3, remplie par des 1

on crée un vecteur `vec` contenant les valeurs 1, 2 et 3

on remplace la valeur située à la première ligne et à la troisième colonne de `m` par 3

on ajoute à `m` une matrice dont la diagonale a les valeurs de `vec`

A ce stade, la matrice stockée dans `m` est :

[1] [2] [3]

[1,] 2 1 3

[2,] 1 3 1

[3,] 1 1 4

Dans la dernière ligne, on prend la transposée de cette matrice. Ainsi, l'affichage final est bien :

[1] [2] [3]

[1,] 2 1 1

[2,] 1 3 1

[3,] 3 1 4]

Question 7

Concernant les listes, quelles affirmations parmi les suivantes sont vraies ?

Attention, plusieurs réponses sont possibles.

1. Une liste ne peut contenir qu'un seul type

2. Si j'utilise la fonction `length` sur la liste suivante :

`list(c("A","B","C","A"),matrix(1:4,2,2))` Celle-ci renverra bien la valeur 8.

3. Lorsqu'ils sont nommés, on peut accéder aux différents objets contenus dans une liste via l'opérateur `$`

4. On peut accéder aux différents objets d'une liste via leurs indices

Explication [Une liste peut contenir plusieurs types.]

Lorsqu'ils sont nommés, on peut accéder aux différents objets contenus dans une liste via l'opérateur `$`

On peut accéder aux différents objets d'une liste via leurs indices

Dans l'exemple donné, la fonction `length` renverra bien 2, car il y a uniquement deux objets : un vecteur et une matrice.]

Question 8

Concernant les dataframes, quelles affirmations parmi les suivantes sont vraies ?

Attention, plusieurs réponses sont possibles.

1. Les dataframes sont des listes particulières dont les composantes sont de même taille

2. Les modes des différentes composantes d'un dataframe doivent être tous identiques
3. On peut créer un dataframe à partir d'un tableau de données externe (.txt, .csv, etc...)
4. Il n'est pas possible de transformer une matrice en dataframe

Explication [Les dataframes sont des listes particulières dont les composantes sont de même taille.

Les modes des différentes composantes d'un dataframe peuvent être différents. Il faut voir un dataframe comme un tableau de données. Ainsi, on peut avoir une colonne représentant des valeurs quantitatives, une autre représentant une variable qualitative, etc...

On peut créer un dataframe à partir d'un tableau de données externe (.txt, .csv, etc...) via la fonction `read.table` que nous verrons prochainement.

Il est possible de transformer une matrice en dataframe via la fonction `as.data.frame`]



LearnerWorld