

Manipulation des facteurs de Claire MAZZUCATO

Tarik HAKAM

22/12/2020

1. Critères d'évaluation

1. Comportement du Rmd à l'exécution
2. Qualité de la rédaction du dossier
3. Accessibilité, didactisme et pertinence du dossier
4. Qualité et lisibilité du Rmarkdown
5. Qualité des applications permettant d'illustrer le package

2. Lien vers le document commenté

En cliquant **ici**, vous trouverez le lien menant au GitHub de Claire MAZZUCATO hébergeant le fruit de sa réalisation.

3. Auteurs du document commenté

Le document évalué dans le cadre de ce rendu a été produit par Claire MAZZUCATO, étudiante en Master of Science Data Management à Paris School of Business.

4. Synthèse du document

L'objectif de ce document est d'explicitier et de présenter ce que sont les facteurs.

Il s'agit donc de vecteurs quelque peu particuliers, facilitant la manipulation de données qualitatives, qu'elles soient numériques ou caractères.

En effet, en plus de stocker les différents éléments comme des vecteurs classiques, les facteurs stockent également l'ensemble des différentes modalités possibles dans un attribut accessible via la commande `levels`.

Ils forment une classe d'objets et bénéficient de traitements particuliers lors de leur manipulation, ainsi que lors de l'utilisation de certaines fonctions. Les facteurs peuvent être non ordonnés (exemples : homme, femme) ou ordonnés (exemple : différents niveaux de ski).

Les facteurs sont utilisés dans R pour encoder les réponses à une question fermée.

En statistique, un facteur est typiquement utilisé pour stocker les valeurs observées d'une variable qualitative ou catégorique.

Pour illustrer le fonctionnement des facteurs, l'auteure a créé un facteur avec des attributs par défaut, puis des niveaux personnalisés, et enfin des niveaux et des étiquettes personnalisés.

5. Extrait commenté des parties de code

L'auteure a mis en avant dans son document plusieurs chunks intéressants à commenter. Tout d'abord, il existe 3 fonctions pour créer les facteurs :

1. La fonction `factor`
2. La fonction `as.factor`
3. La fonction `ordered`

1. La fonction `factor`

La fonction `factor` permet de créer un facteur en définissant directement les différents éléments du facteur.

```
sexe <- factor(c("H", "H", "F", "H", "H", "F", "F", "F"))
```

Ici, les éléments du facteur nouvellement créé seront donc : H, H, F, H, H, F, F et F.

2. La fonction `as.factor`

A l'aide de cette deuxième fonction, on définit tout d'abord le vecteur `salto`.

```
salto <- c(1:5,5:1)
```

Celui-ci aura pour éléments les suites de chiffres suivantes :

- de 1 à 5 : 1, 2, 3, 4, 5
- de 5 à 1 : 5, 4, 3, 2, 1

La sortie de ce chunk donnera donc : 1, 2, 3, 4, 5, 5, 4, 3, 2, 1.

Puis on définit le facteur `salto.f` à partir du vecteur `salto` précédemment défini, tel que :

```
salto.f <- as.factor(salto)
salto.f
```

3. La fonction `ordered`

La 3^{ème} fonction `ordered` va quant à elle, permettre de créer des facteurs ordonnés.

```
niveau <- ordered(c("débutant","débutant","champion",
                    "champion","moyen","moyen","moyen",
                    "champion"),
levels=c("débutant","moyen","champion"))
niveau
```

On définit tout d'abord un facteur `niveau` avec les éléments suivants : débutant, débutant, champion, champion, moyen, moyen, moyen, champion.

Puis, dans un second temps on applique un **ordre** aux éléments du facteur, tel que : débutant < moyen < champion.

Toujours à partir de la fonction `levels`, il est possible de renommer un facteur.

```
levels(sexe) <- c("Femme", "Homme")
```

Ici, nous renommons, les éléments “F” et “H” respectivement en “Femme” et “Homme”.

Il est également possible de modifier la valeur d’un facteur facilement par indexation comme pour un vecteur avec la fonction `levels`.

Par exemple, précédemment dans le document, un facteur `de` a été défini par l’auteure, ayant pour éléments les chiffres suivants : 3 2 2 1 3 1 3 1.

```
de[c(2, 3)] <- c(6, 5)
```

Cette manipulation permet de modifier les éléments en position 2 et 3 respectivement par les chiffres 6 et 5. La sortie sera donc : 3 6 5 1 3 1 3 1.

Par défaut, les niveaux d’un facteur nouvellement créés sont classés par ordre alphanumérique croissant ou selon l’ordre qui figure dans l’option `levels`. Cet ordre est utilisé chaque fois que le facteur est employé.

```
sexe <- factor(c("H", "H", "F", "H", "H", "F", "F", "F"))
```

La sortie affichera :

→ Levels : F H

Le F arrivant avant le H dans l’alphabet latin.

Il est possible de modifier l’ordre des niveaux de ce facteur en utilisant la manipulation suivante à l’aide de la fonction `levels`:

```
sexe <- factor(sexe, levels = c("H", "F"))
```

Le résultat de ce chunk et l’affichage des niveaux ne sera plus “F H” mais “H F”.

6. Evaluation du travail suivant les 5 critères précités

1. Comportement du Rmd à l’exécution

L’exécution du code Rmd se réalise parfaitement, à condition de supprimer la ligne de code permettant d’insérer le logo PSB.

Autrement, il aurait été judicieux de communiquer via GitHub l’image .jpg.

2. Qualité de la rédaction du dossier

La rédaction de ce document est correcte.

A mon sens, afin de construire davantage sa pensée, l’auteure aurait pu certainement commenter davantage les différentes parties de son document dans son code.

3. Accessibilité, didactisme et pertinence du dossier

La lecture de ce dossier est relativement aisée et accessible.

Les descriptions auraient pu être davantage explicitées, cependant cela reste un travail bien illustré.

L’auteure arrive tout de même à faire adhérer le lecteur à sa production. Le document a vocation à transmettre une connaissance et le but, pour ma part, est atteint.

Le sujet est pertinent et a vocation à servir de mémo pour de futures réalisations.

4. Qualité et lisibilité du RMarkdown

Le RMarkdown est bien écrit, lisible et aéré.

J’aurai éventuellement qu’une seule suggestion :

- J’aurai, pour ma part, ajouter quelques sauts de lignes dans le code afin d’aérer davantage encore le document PDF.

Autrement, il s’agit d’un code bien réalisé.

5. Qualité des applications permettant d’illustrer le package

Les applications sont de bonne qualité, maîtrisées. Cependant, elles auraient pu être davantage expliquées.

7. Conclusion

Selon moi, il s’agit globalement d’un bon travail.

L’auteure fournit un dossier recherché, documenté et facile à lire.

On y apprend des choses nouvelles. Personnellement, j’aurai aimé davantage de guidance dans la lecture des chunks pour orienter également ce travail à destination d’un public encore plus novice.

Vous retrouvez ce document sur mon **GitHub**.