Format JSON : de la liste au dataframe

Arno Muller

2023-06-28

Résumé

Le format JSON est un format de plus en plus utilisé pour stocker des données, notamment pour les données issues du web. Cependant, sa structure particulière peut poser des problèmes pour obtenir un dataframe dans R.

**Fonctions utilisées dans la fiche**

| Packages | Fonctions |
| --- | --- |
| **jsonlite** | fromJSON |
| **rjson** | fromJSON |
| **data.table** | rbindlist |
| **Base R** | as.data.frame - knitr - class |

# Introduction

Pour importer des données au format JSON, nous proposons deux packages rjson et jsonlite. Le dernier est beaucoup plus facile d’utilisation et renvoie automatiquement une base de donnée faites de colonnes. Je propose rjson comme une alternative à jsonlite en cas de pépin.

# Données d’exemples

Le format JSON est surtout utilisé pour les données issues d’API web ou de source OpenData. Pour les exemples ci-dessous nous utiliserons donc des données issues du site de la SNCF qui mettent à disposition un grand nombre de jeux de données en libre accès, et notamment dans notre cas, une **base de données des menus des wagons-bars de TGV**[[1]](#footnote-21).

# Importer des données avec jsonlite

head(menu\_TGV,3)

date\_debut date\_fin produit prix\_au\_produit  
1 2023-06-22 2023-10-12 Café lungo, Illy 2.9  
2 2023-06-22 2023-10-12 Cappuccino grand, Illy 4.9  
3 2023-06-22 2023-10-12 Latte Macchiato grand, Illy 4.9  
 type categorie\_produit presence\_de\_porc presence\_d\_alcool  
1 Thé, café & chocolat Boissons NON NON  
2 Boissons gourmandes Boissons NON NON  
3 Boissons gourmandes Boissons NON NON  
 composants  
1 Capsule de 6,5 g de café moulu illy, mélange 100% Arabica  
2 2 Capsules de 15g de lait (lait entier, sirop de glucose, sucre), capsule de 14,5 g de café 100% Arabica, chocolat (chocolat en poudre 59.4% (sucre, cacao 32% minimum), lait écrémé en poudre 38.5%, lactose, antiagglomérant (E551), sel, arôme)  
3 Capsule de 6,5 g de café moulu illy, mélange 100% Arabica, 2 Capsules de 15g de lait (lait entier, sirop de glucose, sucre)  
 allergenes  
1 <NA>  
2 Lactose  
3 Lactose  
 allergenes\_susceptibles\_de\_provenir\_de\_contaminations\_croisees  
1 <NA>  
2 Arachides, amandes, noix, soja, oeufs, poissons, crustaces, sésame, céréales, sulfites, lupin, céleri, moutarde  
3 Arachides, amandes, noix, soja, oeufs, poissons, crustaces, sésame, céréales, sulfites, lupin, céleri, moutarde  
 kcal\_pour\_100\_ml\_ou\_100g recette\_vegetarienne recette\_vegane bio sans\_gluten  
1 4 NON NON NON NON  
2 NA NON NON NON NON  
3 NA NON NON NON NON  
 poids\_g\_cl proteines\_100g\_100ml glucides\_100g\_100ml lipides\_100g\_100ml  
1 NA NA NA NA  
2 NA NA NA NA  
3 NA NA NA NA

# Importer des données avec rjson

library(rjson)

Attachement du package : 'rjson'

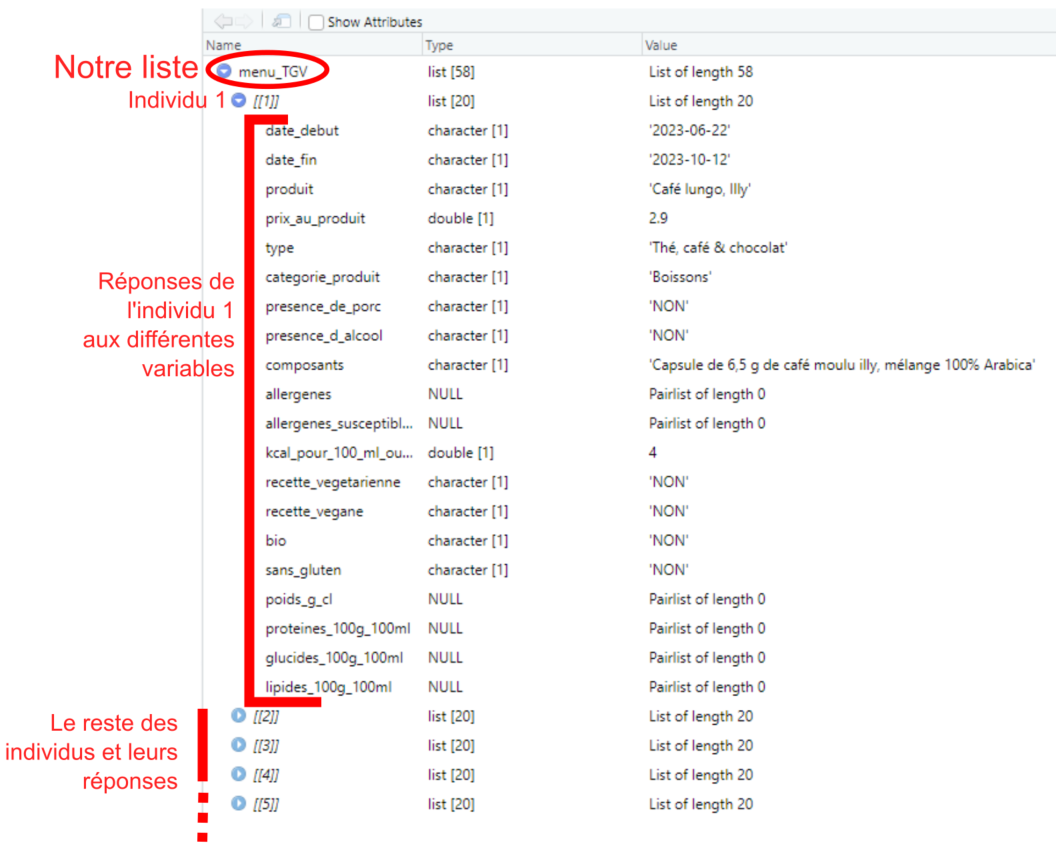
Les objets suivants sont masqués depuis 'package:jsonlite':  
  
 fromJSON, toJSON

menu\_TGV <- rjson::fromJSON(file = "DATA/menus-des-bars-tgv.json")  
class(menu\_TGV)

[1] "list"

On obtient un objet de type list [[2]](#footnote-25), que l’on peut transformer en base de donnée avec l’exemple suivant.

Une liste peut-être un objet assez complexe mais en fouillant dedans, il est possible de retrouver (ou de mettre en forme) une base de donnée. Dans notre cas plutôt simple, on remarque que la liste a autant d’élément que d’individus, et lorsqu’on regarde à l’intérieur de chaque élément, on trouve la réponse pour chaque individu à chacune des variables.



On va donc extraire ces informations et les compiler dans une base de donnée ordonnées.

# Convertir la liste en data frame par lignes  
menu\_df <- as.data.frame(do.call(rbind, menu\_TGV))  
head(menu\_df,3)

date\_debut date\_fin produit prix\_au\_produit  
1 2023-06-22 2023-10-12 Café lungo, Illy 2.9  
2 2023-06-22 2023-10-12 Cappuccino grand, Illy 4.9  
3 2023-06-22 2023-10-12 Latte Macchiato grand, Illy 4.9  
 type categorie\_produit presence\_de\_porc presence\_d\_alcool  
1 Thé, café & chocolat Boissons NON NON  
2 Boissons gourmandes Boissons NON NON  
3 Boissons gourmandes Boissons NON NON  
 composants  
1 Capsule de 6,5 g de café moulu illy, mélange 100% Arabica  
2 2 Capsules de 15g de lait (lait entier, sirop de glucose, sucre), capsule de 14,5 g de café 100% Arabica, chocolat (chocolat en poudre 59.4% (sucre, cacao 32% minimum), lait écrémé en poudre 38.5%, lactose, antiagglomérant (E551), sel, arôme)  
3 Capsule de 6,5 g de café moulu illy, mélange 100% Arabica, 2 Capsules de 15g de lait (lait entier, sirop de glucose, sucre)  
 allergenes  
1 NULL  
2 Lactose  
3 Lactose  
 allergenes\_susceptibles\_de\_provenir\_de\_contaminations\_croisees  
1 NULL  
2 Arachides, amandes, noix, soja, oeufs, poissons, crustaces, sésame, céréales, sulfites, lupin, céleri, moutarde  
3 Arachides, amandes, noix, soja, oeufs, poissons, crustaces, sésame, céréales, sulfites, lupin, céleri, moutarde  
 kcal\_pour\_100\_ml\_ou\_100g recette\_vegetarienne recette\_vegane bio sans\_gluten  
1 4 NON NON NON NON  
2 NULL NON NON NON NON  
3 NULL NON NON NON NON  
 poids\_g\_cl proteines\_100g\_100ml glucides\_100g\_100ml lipides\_100g\_100ml  
1 NULL NULL NULL NULL  
2 NULL NULL NULL NULL  
3 NULL NULL NULL NULL

# Même chose avec le package data.table  
library(data.table)  
menu\_df2 <- rbindlist(menu\_TGV)  
head(menu\_df2,3)

date\_debut date\_fin produit prix\_au\_produit  
1: 2023-06-22 2023-10-12 Café lungo, Illy 2.9  
2: 2023-06-22 2023-10-12 Cappuccino grand, Illy 4.9  
3: 2023-06-22 2023-10-12 Latte Macchiato grand, Illy 4.9  
 type categorie\_produit presence\_de\_porc presence\_d\_alcool  
1: Thé, café & chocolat Boissons NON NON  
2: Boissons gourmandes Boissons NON NON  
3: Boissons gourmandes Boissons NON NON  
 composants  
1: Capsule de 6,5 g de café moulu illy, mélange 100% Arabica  
2: 2 Capsules de 15g de lait (lait entier, sirop de glucose, sucre), capsule de 14,5 g de café 100% Arabica, chocolat (chocolat en poudre 59.4% (sucre, cacao 32% minimum), lait écrémé en poudre 38.5%, lactose, antiagglomérant (E551), sel, arôme)  
3: Capsule de 6,5 g de café moulu illy, mélange 100% Arabica, 2 Capsules de 15g de lait (lait entier, sirop de glucose, sucre)  
 allergenes  
1: <NA>  
2: Lactose  
3: Lactose  
 allergenes\_susceptibles\_de\_provenir\_de\_contaminations\_croisees  
1: <NA>  
2: Arachides, amandes, noix, soja, oeufs, poissons, crustaces, sésame, céréales, sulfites, lupin, céleri, moutarde  
3: Arachides, amandes, noix, soja, oeufs, poissons, crustaces, sésame, céréales, sulfites, lupin, céleri, moutarde  
 kcal\_pour\_100\_ml\_ou\_100g recette\_vegetarienne recette\_vegane bio sans\_gluten  
1: 4 NON NON NON NON  
2: NA NON NON NON NON  
3: NA NON NON NON NON  
 poids\_g\_cl proteines\_100g\_100ml glucides\_100g\_100ml lipides\_100g\_100ml  
1: NA NA NA NA  
2: NA NA NA NA  
3: NA NA NA NA

1. <https://ressources.data.sncf.com/explore/dataset/menus-des-bars-tgv/table/> [↑](#footnote-ref-21)
2. Pour découvrir comment utiliser des listes : <https://juba.github.io/tidyverse/16-structures-donnees.html#sec-listes> [↑](#footnote-ref-25)