

AstéRix_wenjun_zhao_dplyr

Teddy Leandre

23/12/2020

I. Présentation

Dans ce dossier, nous chercherons à évaluer 12 travaux comportant du code R et des concepts mathématiques. L'évaluation se fera sur la base de 5 critères de notation qui, additionnés, donnera une note globale sur 20 points.

1. Qualité du visuel du document sur 4 points
2. Pluralité des fonctionnalités sur 4 points
3. Fonctionnement du code sur 4 points
4. Lisibilité du code sur 4 points
5. Qualité des explications sur 4 points

II. Description

Ici nous verrons l'outil `dplyr`, c'est un paquet qui permet d'effectuer certaines manipulations de données et donc un meilleur traitement. Le travail de Wenjun Zhao se porte sur certaines des fonctionnalités principales du paquet.

[Le Github évalué](#)

a. Commentaire

`dplyr` est une grammaire de manipulation de données, fournissant un ensemble cohérent de verbes qui vous aident à résoudre les défis de manipulation de données les plus courants:

`mutate ()` ajoute de nouvelles variables qui sont des fonctions de variables existantes

`select ()` sélectionne les variables en fonction de leurs noms.

`filter ()` sélectionne les observations en fonction de leurs valeurs.

summary () réduit plusieurs valeurs à un seul résumé.

arrange () modifie l'ordre des lignes.

Tout cela se combine naturellement avec *group_by ()* qui vous permet d'effectuer n'importe quelle opération «par groupe». Vous pouvez en savoir plus sur eux dans *vignette ("dplyr")*. En plus de ces verbes à table unique, dplyr fournit également une variété de verbes à deux tables, que vous pouvez découvrir en *vignette ("two-table")*.

dplyr est conçu pour résumer la manière dont les données sont stockées. Cela signifie qu'en plus de travailler avec des trames de données locales, vous pouvez également travailler avec des tables de base de données distantes, en utilisant exactement le même code R. Installez le package dbplyr puis lisez *vignette ("databases", package = "dbplyr")*.

Installation

Le moyen le plus simple d'obtenir dplyr est d'installer tout le tidyverse:

```
install.packages ("tidyverse")
```

Alternativement, installez simplement dplyr:

```
install.packages ("dplyr")
```

Ou la version de développement de GitHub:

```
install.packages ("devtools")
```

```
devtools :: install_github ("tidyverse / dplyr")
```

Utilisation

```
library(dplyr)
```

```
##  
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':  
##  
## filter, lag
```

```
## The following objects are masked from 'package:base':
##
## intersect, setdiff, setequal, union
```

```
starwars %>%
  filter(species == "Droid")
```

```
## # A tibble: 6 x 14
##   name height mass hair_color skin_color eye_color birth_year sex gender
##   <chr>   <int> <dbl> <chr>      <chr>      <chr>      <dbl> <chr> <chr>
## 1 C-3P0    167    75 <NA>      gold        yellow        112 none masculi
## 2 R2-D2     96    32 <NA>      white, bl~ red          33 none masculi
## 3 R5-D4     97    32 <NA>      white, red red          NA none masculi
## 4 IG-88    200   140 none      metal        red          15 none masculi
## 5 R4-P~     96    NA none      silver, r~ red, blue    NA none femini
## 6 BB8      NA    NA none      none         black         NA none masculi
## # ... with 5 more variables: homeworld <chr>, species <chr>, films <list>,
## #   vehicles <list>, starships <list>
```

```
library(dplyr)
starwars %>%
  select(name, ends_with("color"))
```

```
## # A tibble: 87 x 4
##   name          hair_color skin_color eye_color
##   <chr>         <chr>      <chr>      <chr>
## 1 Luke Skywalker blond      fair      blue
## 2 C-3P0         <NA>      gold      yellow
## 3 R2-D2         <NA>      white, blue red
## 4 Darth Vader   none      white      yellow
## 5 Leia Organa   brown     light     brown
## 6 Owen Lars     brown, grey light     blue
## 7 Beru Whitesun lars brown     light     blue
## 8 R5-D4         <NA>      white, red red
```

```
## 9 Biggs Darklighter black light brown
## 10 Obi-Wan Kenobi auburn, white fair blue-gray
## # ... with 77 more rows
```

```
library(dplyr)
starwars %>%
  mutate(name, bmi = mass / ((height / 100) ^ 2)) %>%
  select(name:mass, bmi)
```

```
## # A tibble: 87 x 4
##   name          height mass  bmi
##   <chr>         <int> <dbl> <dbl>
## 1 Luke Skywalker    172    77  26.0
## 2 C-3P0             167    75  26.9
## 3 R2-D2              96    32  34.7
## 4 Darth Vader       202   136  33.3
## 5 Leia Organa       150    49  21.8
## 6 Owen Lars         178   120  37.9
## 7 Beru Whitesun lars 165    75  27.5
## 8 R5-D4              97    32  34.0
## 9 Biggs Darklighter 183    84  25.1
## 10 Obi-Wan Kenobi   182    77  23.2
## # ... with 77 more rows
```

```
library(dplyr)
starwars %>%
  arrange(desc(mass))
```

```
## # A tibble: 87 x 14
##   name height mass hair_color skin_color eye_color birth_year sex gender
##   <chr> <int> <dbl> <chr>      <chr>      <chr>      <dbl> <chr> <chr>
## 1 Jabb~   175  1358 <NA>      green-tan~ orange      600 herm~ mascu~
## 2 Grie~   216   159 none      brown, wh~ green, y~   NA male mascu~
## 3 IG-88   200   140 none      metal      red        15 none mascu~
```

```
## 4 Dart~    202   136 none      white      yellow      41.9 male mascu~
## 5 Tarf~    234   136 brown     brown      blue        NA  male mascu~
## 6 Owen~    178   120 brown, gr~ light      blue        52  male mascu~
## 7 Bossk    190   113 none      green      red         53  male mascu~
## 8 Chew~    228   112 brown     unknown    blue        200 male mascu~
## 9 Jek ~    180   110 brown     fair       blue        NA  male mascu~
## 10 Dext~   198   102 none      brown      yellow      NA  male mascu~
## # ... with 77 more rows, and 5 more variables: homeworld <chr>, species <chr>,
## #   films <list>, vehicles <list>, starships <list>
```

```
library(dplyr)
starwars %>%
  group_by(species) %>%
  summarise(
    n = n(),
    mass = mean(mass, na.rm = TRUE)
  ) %>%
  filter(n > 1)
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 9 x 3
##   species      n mass
##   <chr>    <int> <dbl>
## 1 Droid         6  69.8
## 2 Gungan        3   74
## 3 Human       35  82.8
## 4 Kaminoan      2   88
## 5 Mirialan       2  53.1
## 6 Twi'lek        2   55
## 7 Wookiee        2  124
## 8 Zabrak         2   80
## 9 <NA>           4   48
```

b. Notation

Critère 1 : 4/4 Visuel agréable

Critère 2 : 4/4 Fonction bien expliqué

Critère 3 : 4/4 Le code fonctionne.

Critère 4 : 2/4 Présentation abrupte du paquet.

Critère 5 : 3/4 Explication assez claire.

III. Conclusion

Note globale de 17/20. Bon travail qui explique bien le package dplyr. Les commentaires aurait pu être mieux positionné.