

# Présentation de DPLYR

FONTAINE Grégoire

28/11/2020

## Contents

<b>I) Introduction</b>	<b>3</b>
<b>II) Installation</b>	<b>3</b>
<b>III) Usage</b>	<b>3</b>
<b>IV) Documentation</b>	<b>6</b>



## I) Introduction

Le paquet dplyr est une grammaire pour la data manipulation, fournissant un ensemble cohérent de verbes qui nous aident à résoudre les défis les plus courants de la manipulation des données. Lorsque qu'on travaille avec des données, nous devons :

- Trouver ce qu'il faut faire.
- Décrire ces tâches sous la forme d'un programme informatique.
- Exécuter le programme.

Le paquet dplyr rend ces étapes rapides et faciles :

- En limitant vos options, il vous aide à réfléchir à vos défis de manipulation de données.
- Il fournit des « verbes » simples, des fonctions qui correspondent aux tâches de manipulation de données les plus courantes, pour vous aider à traduire vos pensées en code.
- Il utilise des backends efficaces, de sorte que vous passez moins de temps à attendre l'ordinateur.

## II) Installation

- 1) Le moyen le plus rapide pour installer le packages

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

- 2) Alternative

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

## III) Usage

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

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

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

`summarise()` réduit plusieurs valeurs à un seul résumé.

`arrange()` change l'ordre des lignes.

Pour explorer les verbes de base de la manipulation des données de dplyr, nous utiliserons le jeu de données star wars. Cet ensemble de données contient 87 caractères et provient de l'API Star Wars.

```
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 femin~
## 6 BB8 NA NA none none black NA none masculi
## # ... with 5 more variables: homeworld <chr>, species <chr>, films <list>,
## # vehicles <list>, starships <list>
```

```
#> # 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 femin...
#> # ... with 1 more row, and 5 more variables: homeworld <chr>, species <chr>,
#> # films <list>, vehicles <list>, starships <list>
```

```
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
```

```
#> # 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
#> # ... with 82 more rows
```

```
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-3PO             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
```

```
#> # A tibble: 87 x 4
#>   name      height mass  bmi
#>   <chr>      <int> <dbl> <dbl>
#> 1 Luke Skywalker    172    77  26.0
#> 2 C-3PO             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
#> # ... with 82 more rows
```

```
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>
```

```
#> # 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...
```

```
#> # ... with 82 more rows, and 5 more variables: homeworld <chr>, species <chr>,
#> #   films <list>, vehicles <list>, starships <list>
```

```
starwars %>%
  group_by(species) %>%
  summarise(
    n = n(),
    mass = mean(mass, na.rm = TRUE)
  ) %>%
  filter(
    n > 1,
    mass > 50
  )
```

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

```
## # A tibble: 8 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
```

```
#> # A tibble: 8 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
#> # ... with 3 more rows
```

## IV) Documentation

Exemple d'entreprise française utilisant dplyr

Cheat Sheet pour DPLYR

Documentation R