Data Manipulation with `dplyr`

Nội dung

Giới thiệu `dplyr` package

Tidy data

- Các đặc trưng của tidy data
  - Mỗi biến/đặc trưng (variable/feature) tạo thành một cột
  - Mỗi đối tượng dữ liệu (observation/instance) tạo thành một dòng
  - Mỗi kiểu đối tượng dữ liệu (object/instance type) tạo thành một bảng

dplyr package

- dplyr package cung cấp các một **grammar** để thao tác trên dữ liệu. Mỗi hàm là một động từ
  - `select()`: chọn dòng
  - `filter()`: chọn dòng dùng điều kiện trên giá trị
  - `mutate()`: tạo biến mới là một hàm trên các biến đã có
  - `summarise()`: tổng hợp kết quả từ nhiều dòng
  - `group_by()`: nhóm các dòng theo giá trị
  - `arrange()`: sắp xếp dòng
  - ...
- Tham khảo: https://dplyr.tidyverse.org/reference/index.html

Nguyên tắc khi áp dụng các hàm của `dplyr`

- Đối số đầu tiên **luôn** là một `data frame`
- Các đối số tiếp theo nói đến điều ta muốn làm trên data frame đó
- Kết quả trả về **luôn** là một `data frame`

```
library(dplyr)
```

starwars dataset

- Thông tin về các nhân vật trong loạt phim Star Wars.

```
glimpse(starwars)
```

```
## Rows: 87
## Columns: 14
## $ name       <chr> "Luke Skywalker", "C-3PO", "R2-D2", "Darth Vader", "
## $ height     <int> 172, 167, 96, 202, 150, 178, 165, 97, 183, 182, 188,
## $ mass       <dbl> 77.0, 75.0, 32.0, 136.0, 49.0, 120.0, 75.0, 32.0, 84
## $ hair_color <chr> "blond", NA, NA, "none", "brown", "brown, grey", "bro
## $ skin_color <chr> "fair", "gold", "white, blue", "white", "light", "lig
## $ eye_color  <chr> "blue", "yellow", "red", "yellow", "brown", "blue",
## $ birth_year <dbl> 19.0, 112.0, 33.0, 41.9, 19.0, 52.0, 47.0, NA, 24.0,
## $ sex        <chr> "male", "none", "none", "male", "female", "male", "fe
## $ gender     <chr> "masculine", "masculine", "masculine", "masculine",
## $ homeworld  <chr> "Tatooine", "Tatooine", "Naboo", "Tatooine", "Alderaa
## $ species    <chr> "Human", "Droid", "Droid", "Human", "Human", "Human"
## $ films      <list> <"The Empire Strikes Back", "Revenge of the Sith"
```

starwars dataset

```
head(starwars)
```

```
## # A tibble: 6 x 14
##   name      height  mass hair_color skin_color eye_color birth_year sex
##   <chr>      <int> <dbl> <chr>      <chr>      <chr>           <dbl> <ch
## 1 Luke Sky~    172    77 blond      fair       blue             19   mal
## 2 C-3PO        167    75 <NA>       gold       yellow          112   non
## 3 R2-D2         96    32 <NA>       white, bl~ red              33   non
## 4 Darth Va~    202   136 none       white      yellow           41.9 mal
## 5 Leia Org~    150    49 brown      light      brown            19   fem
## 6 Owen Lars    178   120 brown, gr~ light      blue             52   mal
## # ... with 5 more variables: homeworld <chr>, species <chr>, films <lis
## #   vehicles <list>, starships <list>
```

Chọn cột

```
select(starwars, name, height, mass, gender)
```

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

Chọn dòng

```
filter(starwars, gender == 'male' & height > 180)
```

```
## # A tibble: 0 x 14
## # ... with 14 variables: name <chr>, height <int>, mass <dbl>,
## #   hair_color <chr>, skin_color <chr>, eye_color <chr>, birth_year <dbl
## #   sex <chr>, gender <chr>, homeworld <chr>, species <chr>, films <list
## #   vehicles <list>, starships <list>
```

Toán tử pipe (%>%)

- Trong lập trình, một pipe là một kỹ thuật truyền thông tin từ một process sang process khác.

```
starwars %>%
  select(name, height, mass, gender) %>%
  filter(gender == 'male' & height > 180)
```

```
## # A tibble: 0 x 4
## # ... with 4 variables: name <chr>, height <int>, mass <dbl>, gender <ch
```

Toán tử pipe (%>%)

- Thay vì dùng toán tử pipe (%>%) ta có thể áp dụng một chuỗi các hàm.

```
filter(select(starwars, name, height, mass, gender),
       gender == 'male' & height > 180)
```

```
starwars %>%
  select(name, height, mass, gender) %>%
  filter(gender == 'male' & height > 180)
```

- Tuy nhiên, toán tử pipe giúp mã dễ đọc và dễ hiểu hơn.

Thao tác trên một data frame

Loại bỏ một số cột (biến) với select

```
starwars %>%
  select(-c(hair_color, skin_color, eye_color,
            films, vehicles, starships))
```

```
## # A tibble: 87 x 8
##    name               height  mass birth_year sex    gender    homeworl
##    <chr>               <int> <dbl>      <dbl> <chr>  <chr>     <chr>
##  1 Luke Skywalker        172    77       19   male   masculine Tatooine
##  2 C-3PO                 167    75      112   none   masculine Tatooine
##  3 R2-D2                  96    32       33   none   masculine Naboo
##  4 Darth Vader           202   136       41.9 male   masculine Tatooine
##  5 Leia Organa           150    49       19   female feminine  Alderaan
##  6 Owen Lars             178   120       52   male   masculine Tatooine
##  7 Beru Whitesun lars    165    75       47   female feminine  Tatooine
##  8 R5-D4                  97    32       NA   none   masculine Tatooine
##  9 Biggs Darklighter     183    84       24   male   masculine Tatooine
## 10 Obi-Wan Kenobi        182    77       57   male   masculine Stewjon
```

Chọn các cột (biến) liên tiếp với `select`

```
starwars %>%
  select(name:mass)
```

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

Sắp xếp các dòng theo thứ tự với arrange

```
starwars %>%
  select(name:mass) %>%
  arrange(mass)
```

```
## # A tibble: 87 x 3
##    name                  height  mass
##    <chr>                  <int> <dbl>
##  1 Ratts Tyerell             79    15
##  2 Yoda                      66    17
##  3 Wicket Systri Warrick     88    20
##  4 R2-D2                     96    32
##  5 R5-D4                     97    32
##  6 Sebulba                  112    40
##  7 Dud Bolt                  94    45
##  8 Padmé Amidala            165    45
##  9 Wat Tambor               193    48
## 10 Sly Moore                178    48
```

Sắp xếp các dòng theo thứ tự giảm dần với arrange và desc

```
starwars %>%
  select(name:mass) %>%
  arrange(desc(mass))
```

```
## # A tibble: 87 x 3
##    name                  height  mass
##    <chr>                  <int> <dbl>
##  1 Jabba Desilijic Tiure    175  1358
##  2 Grievous                 216   159
##  3 IG-88                    200   140
##  4 Darth Vader              202   136
##  5 Tarfful                  234   136
##  6 Owen Lars                178   120
##  7 Bossk                    190   113
##  8 Chewbacca                228   112
##  9 Jek Tono Porkins         180   110
## 10 Dexter Jettster          198   102
```

Chọn các dòng theo chỉ số với `slice`

```
starwars %>%
  select(name:mass) %>%
  slice(1:5)
```

```
## # A tibble: 5 x 3
##   name           height  mass
##   <chr>           <int> <dbl>
## 1 Luke Skywalker    172    77
## 2 C-3PO             167    75
## 3 R2-D2              96    32
## 4 Darth Vader       202   136
## 5 Leia Organa       150    49
```

Ghi chú mã với

```
starwars %>%
  #select(name:mass) %>%  # dòng này không thực thi
  slice(1:3)              # chọn dòng
```

```
## # A tibble: 3 x 14
##   name      height  mass hair_color skin_color eye_color birth_year sex
##   <chr>      <int> <dbl> <chr>      <chr>      <chr>           <dbl> <ch
## 1 Luke Sky~    172    77 blond      fair       blue             19   mal
## 2 C-3PO        167    75 <NA>       gold       yellow          112   non
## 3 R2-D2         96    32 <NA>       white, bl~ red              33   non
## # ... with 5 more variables: homeworld <chr>, species <chr>, films <lis
## #   vehicles <list>, starships <list>
```

Chọn các dòng theo chỉ số với `slice`

```
nrows = nrow(starwars)    # lấy số dòng của data frame
starwars %>%
  select(name:mass) %>%
  slice((nrows - 3):nrows)
```

```
## # A tibble: 4 x 3
##   name            height  mass
##   <chr>            <int> <dbl>
## 1 Poe Dameron         NA    NA
## 2 BB8                 NA    NA
## 3 Captain Phasma      NA    NA
## 4 Padmé Amidala      165    45
```

Chọn các dòng theo điều kiện với `filter`

```
starwars %>%
  filter(height > 220) %>%
  select(name, height, mass, gender, homeworld, species)
```

```
## # A tibble: 5 x 6
##   name          height  mass gender    homeworld species
##   <chr>          <int> <dbl> <chr>     <chr>     <chr>
## 1 Chewbacca        228   112 masculine Kashyyyk  Wookiee
## 2 Roos Tarpals     224    82 masculine Naboo     Gungan
## 3 Yarael Poof      264    NA masculine Quermia   Quermian
## 4 Lama Su          229    88 masculine Kamino    Kaminoan
## 5 Tarfful          234   136 masculine Kashyyyk  Wookiee
```

Chọn các dòng theo nhiều điều kiện với `filter`

```
starwars %>%
  filter(
    height > 200,
    mass > 100
  ) %>%
  select(name, height, mass, gender, homeworld, species)
```

```
## # A tibble: 4 x 6
##   name         height  mass gender    homeworld species
##   <chr>         <int> <dbl> <chr>     <chr>     <chr>
## 1 Darth Vader     202   136 masculine Tatooine  Human
## 2 Chewbacca       228   112 masculine Kashyyyk  Wookiee
## 3 Grievous        216   159 masculine Kalee     Kaleesh
## 4 Tarfful         234   136 masculine Kashyyyk  Wookiee
```

Các phép toán logic

Phép toán	Ý nghĩa	Phép toán	Ý nghĩa
$<$	bé hơn	`x & y`	`x` AND `y`
$<=$	bé hơn hoặc bằng	`x | y`	`x` OR `y`
$>$	lớn hơn	`!x`	NOT `x`
$>=$	lớn hơn hoặc bằng	`x %in% y`	kiểm tra `x` có thuộc `y`
$==$	bằng nhau (giá trị)	`is.na(x)`	kiểm tra `x` có là `NA`
$!=$	khác	`!is.na(x)`	kiểm tra `x` khác `NA`

Lọc để chọn các dòng duy nhất với distinct

```
starwars %>%
  distinct(species, homeworld) %>%
  arrange(species, homeworld)
```

```
## # A tibble: 58 x 2
##    homeworld   species
##    <chr>       <chr>
##  1 Aleen Minor Aleena
##  2 Ojom        Besalisk
##  3 Cerea       Cerean
##  4 Zolan       Clawdite
##  5 Champala    Chagrian
##  6 Naboo       Droid
##  7 Tatooine    Droid
##  8 <NA>        Droid
##  9 Malastare   Dug
## 10 Endor       Ewok
```

Tạo bảng tần số với count

```
starwars %>%
  count(species, gender) %>%
  arrange(species, gender)
```

```
## # A tibble: 42 x 3
##    species   gender        n
##    <chr>     <chr>     <int>
##  1 Aleena    masculine     1
##  2 Besalisk  masculine     1
##  3 Cerean    masculine     1
##  4 Clawdite  feminine      1
##  5 Chagrian  masculine     1
##  6 Droid     feminine      1
##  7 Droid     masculine     5
##  8 Dug       masculine     1
##  9 Ewok      masculine     1
## 10 Geonosian masculine     1
```

Tạo thêm biến mới với mutate

```
starwars %>%
  mutate(
    bmi = mass/(height/100)^2
  ) %>%
  select(name, bmi, gender) %>%
  arrange(name)
```

```
## # A tibble: 87 x 3
##    name                 bmi gender
##    <chr>              <dbl> <chr>
##  1 Ackbar              25.6 masculine
##  2 Adi Gallia          14.8 feminine
##  3 Anakin Skywalker    23.8 masculine
##  4 Arvel Crynyd        NA   masculine
##  5 Ayla Secura         17.4 feminine
##  6 Bail Prestor Organa NA   masculine
##  7 Barriss Offee       18.1 feminine
```

Tổng hợp với summarise

```
starwars %>%
  summarise(
    avg_height = mean(height, na.rm = T),
    avg_mass = mean(mass, na.rm = T)
  )
```

```
## # A tibble: 1 x 2
##   avg_height avg_mass
##        <dbl>    <dbl>
## 1       174.     97.3
```

Tổng hợp với summarise và group_by

```
starwars %>%
  group_by(gender) %>%
  summarise(
    avg_height = mean(height, na.rm = T),
    avg_mass = mean(mass, na.rm = T),
    num_chars = n()
  )
```

```
## # A tibble: 3 x 4
##   gender    avg_height avg_mass num_chars
##   <chr>          <dbl>    <dbl>     <int>
## 1 feminine        165.     54.7        17
## 2 masculine       177.    106.         66
## 3 <NA>            181.     48           4
```

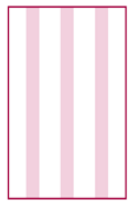
Thao tác trên nhiều data frame

Kết (join) các data frame

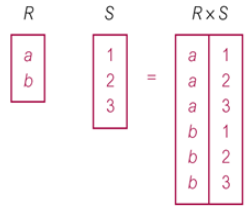
- Cú pháp: `xxx_join(x, y, by = "...")`
- `inner_join()`: tất cả các dòng của `x` có giá trị **match** với `y`
- `left_join()`: `inner_join()` hội với tất cả các dòng của `x`
- `right_join()`: `inner_join()` hội với tất cả các dòng của `y`
- `full_join()`: tất cả các dòng của `x` và `y`
- `semi_join()`: chiếu `inner_join()` trên các cột của `x`
- `anti_join()`: `x - semi_join()`

Các phép toán đại số quan hệ (1)

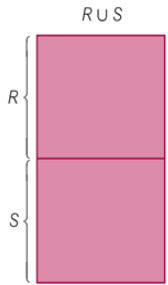
(a) Selection



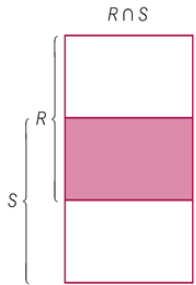
(b) Projection



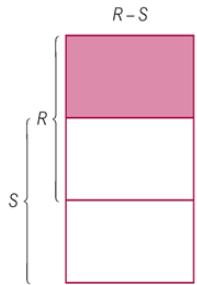
(c) Cartesian product



(d) Union

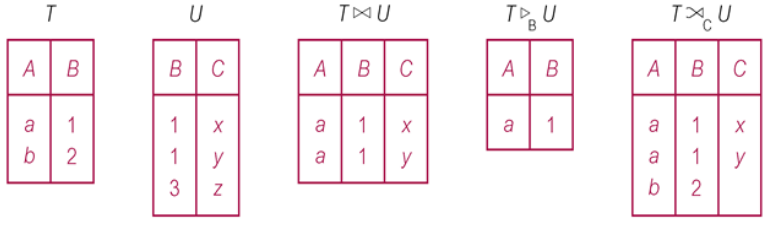


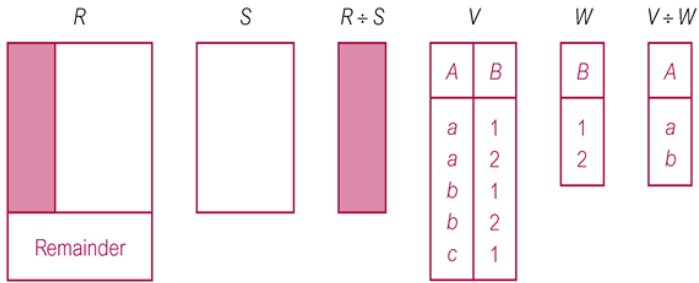
(e) Intersection



(f) Set difference

Các phép toán đại số quan hệ (2)





Inner join

```
inner_join(Employee, Dept, by = "DeptName")
```

- Cho trước $r(X),\ s(Y)$
- Cú pháp: $r \triangleright\!\triangleleft s$
- Ngữ nghĩa: $r \triangleright\!\triangleleft s = \{t[X \cup Y] \mid t[X] \in r, t[Y] \in s\}$

Employee

Name	EmpId	DeptName
Harry	3415	Finance
Sally	2241	Sales
George	3401	Finance
Harriet	2202	Sales

Dept

DeptName	Manager
Finance	George
Sales	Harriet
Production	Charles

Employee $\bowtie$ ***Dept***

Name	EmpId	DeptName	Manager
Harry	3415	Finance	George
Sally	2241	Sales	Harriet
George	3401	Finance	George
Harriet	2202	Sales	Harriet

Left join

```
left_join(Employee, Dept, by = "DeptName")
```

– Cho trước $r(X),\ s(Y)$

– Cú pháp: $r \ltimes s$

– Ngữ nghĩa: $r \triangleright\!\triangleleft s \cup \big((r \triangleright s) \times \{(null, ..., null)\}\big)$

$$r \triangleright\!\triangleleft s \cup \big\{t[X \cup Y] \mid t[X] \in r \triangleright s, t[Y] = (null, ..., null)\big\}$$

Employee

Name	EmpId	DeptName
Harry	3415	Finance
Sally	2241	Sales
George	3401	Finance
Harriet	2202	Sales
Tim	1123	Executive

Dept

DeptName	Manager
Sales	Harriet
Production	Charles

Employee $\ltimes$ *Dept*

Name	EmpId	DeptName	Manager
Harry	3415	Finance	ω
Sally	2241	Sales	Harriet
George	3401	Finance	ω
Harriet	2202	Sales	Harriet
Tim	1123	Executive	ω

Right join

```
right_join(Employee, Dept, by = "DeptName")
```

– Cho trước $r(X),\ s(Y)$

– Cú pháp: $r \rtimes s$

– Ngữ nghĩa: $r \triangleright\triangleleft s \cup \left(\{(null,...,null)\} \times (s \triangleright r)\right)$

$$r \triangleright\triangleleft s \cup \left\{t\left[X \cup Y\right] \mid t\left[Y\right] \in s \triangleright r, t\left[X\right] = (null,...,null)\right\}$$

Employee

Name	EmpId	DeptName
Harry	3415	Finance
Sally	2241	Sales
George	3401	Finance
Harriet	2202	Sales
Tim	1123	Executive

Dept

DeptName	Manager
Sales	Harriet
Production	Charles

Employee $\rtimes$ *Dept*

Name	EmpId	DeptName	Manager
Sally	2241	Sales	Harriet
Harriet	2202	Sales	Harriet
ω	ω	Production	Charles

Full join

```
full_join(Employee, Dept, by = "DeptName")
```

- Cho trước $r(X), s(Y)$
- Cú pháp: $r \rtimes s$
- Ngữ nghĩa: $(r \bowtie s) \cup (r \rtimes s)$

Employee

Name	EmpId	DeptName
Harry	3415	Finance
Sally	2241	Sales
George	3401	Finance
Harriet	2202	Sales
Tim	1123	Executive

Dept

DeptName	Manager
Sales	Harriet
Production	Charles

Employee $\rtimes$ ***Dept***

Name	EmpId	DeptName	Manager
Harry	3415	Finance	ω
Sally	2241	Sales	Harriet
George	3401	Finance	ω
Harriet	2202	Sales	Harriet
Tim	1123	Executive	ω
ω	ω	Production	Charles

Semi join

```
semi_join(Employee, Dept, by = "DeptName")
```

– Cho trước $r(X),\ s(Y)$

– Cú pháp: $r \ltimes s$

– Ngữ nghĩa: $\pi_X\left(r \triangleright\!\triangleleft s\right)$

Employee

Name	EmpId	DeptName
Harry	3415	Finance
Sally	2241	Sales
George	3401	Finance
Harriet	2202	Production

Dept

DeptName	Manager
Sales	Bob
Sales	Thomas
Production	Katie
Production	Mark

Employee $\ltimes$ *Dept*

Name	EmpId	DeptName
Sally	2241	Sales
Harriet	2202	Production

Anti join

```
anti_join(Employee, Dept, by = "DeptName")
```

- Cho trước $r(X),\ s(Y)$
- Cú pháp: $r \triangleright s$
- Ngữ nghĩa: $r - (r \ltimes s)$

Employee

Name	EmpId	DeptName
Harry	3415	Finance
Sally	2241	Sales
George	3401	Finance
Harriet	2202	Production

Dept

DeptName	Manager
Sales	Sally
Production	Harriet

Employee ▷ Dept

Name	EmpId	DeptName
Harry	3415	Finance
George	3401	Finance

Summary

Các thao tác cơ bản

- `select()`: chọn dòng
- `filter()`: chọn dòng dùng điều kiện trên giá trị
- `mutate()`: tạo biến mới là một hàm trên các biến đã có
- `summarise()`: tổng hợp kết quả từ nhiều dòng
- `group_by()`: nhóm các dòng theo giá trị
- `arrange()`: sắp xếp dòng

Chọn dòng với `filter()`

- Cú pháp: `filter(dataframe, ...criteria...)`
  - `name == value`: chọn các dòng có giá trị là `value` ở biến `name` (chú ý dùng `==` thay vì `=`)
  - Sử dụng `name %in% (value1, value2, ...)` để ràng buộc giá trị biến `name` nhận giá trị từ danh sách cho trước
  - `name > value`: chọn các dòng có giá trị ở biến `name` lớn hơn `value` (tương tự cho `>=`, `!=`, `<`, và `<=`).
  - Sử dụng `near(expression, value)` để so sánh số thực thay vì `==`
  - Các phép toán logic: `&` (and), `|` (or), và `!` (not)

Giá trị NA

- NA (not available) thể hiện một giá trị chưa/không biết
- Hầu hết các thao tác trên NA trả về NA (không biết input thì cũng không rõ output)
- Sử dụng is.na(value) để kiểm tra giá trị NA
- Hàm filter() sẽ bỏ qua các dòng có criteria nhận giá trị FALSE và NA.

Một số ghi chú (1)

- `desc(name)`: sắp xếp giảm dần
- `select(dataframe, name1, name2, ...)`: chọn biến
- `name1:namek`: chọn các biến từ `name1` đến `namek`
- `-(name1:namek)`: bỏ qua các biến từ `name1` đến `namek`
- `rename(dataframe, new_name = old_name)`: chọn và đổi tên biến
- `mutate(dataframe, name1=expression1, name2=expression2, ...)`: thêm biến mới
- `transmute(frame, name1=expression1, name2=expression2, ...)`: tạo dataframe mới

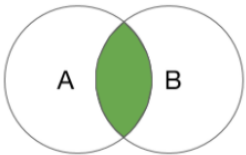
Một số ghi chú (2)

- `group_by(dataframe, name1, name2, ...)`: phân hoạch các giá trị của dataframe theo nhóm
- `summarize(dataframe, name = function(...))`: tổng hợp các giá trị trên cả dataframe hoặc theo nhóm dùng hàm
  - Mặc định, `summarize` trả về `NA` nếu input là `NA`
  - Sử dụng `na.rm = TRUE` để bỏ qua `NA` trước khi `summarize`
- `dataframe %>% operation1(...) %>% operation2(...) %>% ...`: tạo ra dataframe mới mỗi khi áp dụng một `operation`
- `n()`: đếm (count), `sum(!is.na(name))`: đếm giá trị khác `NA`

Một số hàm tổng hợp phổ biến

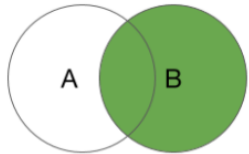
- `mean`, `median`
- `sd`: standard deviation
- `min`, `quantile`, `max`
- `first`, `nth`, `last`
- `n_distinct`: đếm số giá trị khác nhau
- `count`

Các phép kết

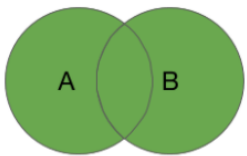


INNER JOIN

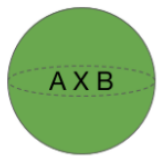
LEFT OUTER JOIN



RIGHT OUTER JOIN



FULL OUTER JOIN



CARTESIAN (CROSS) JOIN

Tham khảo

1. https://r4ds.had.co.nz/transform.html
2. https://github.com/rstudio/cheatsheets/raw/master/data-transformation.pdf
3. https://en.wikipedia.org/wiki/Relational_algebra