PHÙNG KHÁNH LÂM, ONG PHÚC THỊNH

PHÂN TÍCH SỐ LIỆU CƠ BẢN

Contents

Lời nói đầu

Test

0.1 Nội dung dự kiến

- Vì sao có cuốn sách này?
- Mục tiêu
- Đối tượng
- Phương pháp:
  - Lý thuyết kết hợp với ví dụ thực tế
  - Thực hành: R commander
- Đóng góp của nhóm tác giả

0.2 Dữ liệu sử dụng trong sách

Để minh hoạ cho các nội dung được trình bày, chúng tôi sử dụng các ví dụ thực tế, được phân tích bằng phần mềm R, trên bộ dữ liệu của một đoàn hệ tiền cứu trên trẻ nhập viện với chẩn đoán lâm sàng là sốt xuất huyết tại Bệnh viện Bệnh Nhiệt đới TP HCM từ năm 2001 đến năm 2009. Bộ dữ liệu này được công bố cùng với bài báo trình bày kết quả phân tích của nghiên cứu này (Lam et al., 2017). Truy cập website của bài báo `https://journals.plos.org/plosntds/article?id=10.1371/journal.pntd.0005498` để tải về bộ dữ liệu này.

1
Giới thiệu về phân tích dữ liệu

Phân tích dữ liệu có thể được chia làm hai giai đoạn chính:

- Giai đoạn ban đầu: bao gồm hai giai đoạn nhỏ:
  - Kiểm tra và làm sạch số liệu
  - Chuẩn bị dữ liệu cho phân tích
- Giai đoạn phân tích thực sự

You can label chapter and section titles using `{#label}` after them, e.g., we can reference Chapter 1. If you do not manually label them, there will be automatic labels anyway, e.g., Chapter **??**.

Figures and tables with captions will be placed in `figure` and `table` environments, respectively.

```
par(mar = c(4, 4, 0.1, 0.1))
plot(pressure, type = "b", pch = 19)
```

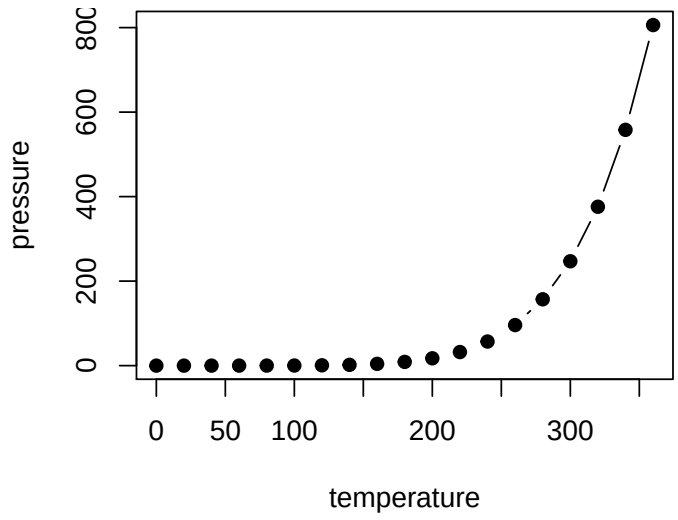


Figure 1.1: Here is a nice figure!

Table 1.1: Here is a nice table!

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5.0	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa
5.4	3.7	1.5	0.2	setosa
4.8	3.4	1.6	0.2	setosa
4.8	3.0	1.4	0.1	setosa
4.3	3.0	1.1	0.1	setosa
5.8	4.0	1.2	0.2	setosa
5.7	4.4	1.5	0.4	setosa
5.4	3.9	1.3	0.4	setosa
5.1	3.5	1.4	0.3	setosa
5.7	3.8	1.7	0.3	setosa
5.1	3.8	1.5	0.3	setosa

Reference a figure by its code chunk label with the `fig:` prefix, e.g., see Figure 1.1. Similarly, you can reference tables generated from `knitr::kable()`, e.g., see Table 1.1.

```
knitr::kable(
  head(iris, 20), caption = 'Here is a nice table!',
  booktabs = TRUE
)
```

You can write citations, too. For example, we are using the **bookdown** package (Xie, 2019) in this sample book, which was built on top of R Markdown and **knitr** (Xie, 2015).

2
Kiểm tra và làm sạch dữ liệu

Đây là bước đầu tiên nên được thực hiện khi nhận được một bộ số liệu, với mục tiêu là xây dựng được bộ số liệu sạch cho phân tích.

Sai sót hoặc dữ liệu bất thường rất thường gặp. Những sai sót này có thể bắt nguồn từ những sai sót trong quá trình thu thập dữ liệu và/hoặc quá trình nhập liệu. Một số lỗi có thể tình cờ phát hiện được trong khi phân tích, nhưng nhiều trường hợp không thể dễ dàng phát hiện và sẽ ảnh hưởng đến kết quả phân tích, thậm chí dẫn đến các sai sót trầm trọng. Vì vậy trước khi phân tích cần phải kiểm tra tính chính xác của số liệu để đảm bảo kết quả phân tích là chính xác nhất có thể.

2.1 Các nhóm lỗi dữ liệu thường gặp

Rất nhiều lỗi khác nhau có thể gặp trong dữ liệu. Tuy nhiên, có thể tóm lại thành 4 nhóm lỗi thường gặp chính:

- Dữ liệu bị thiếu/mất (missing data): khi không có dữ liệu như mong đợi, có thể do trục trặc khi thu thập dữ liệu (đối tượng nghiên cứu bỏ nghiên cứu giữa chừng/không cung cấp thông tin, nghiên cứu viên quên thu thập thông tin) hoặc khi nhập liệu (nhập liệu sót, đặc biệt với phiếu thu thập thông tin dạng nhảy câu). Trong bộ dữ liệu, dữ liệu bị thiếu/mất thường được thể hiện bằng khoảng trắng (blank) hoặc giá trị NA (not available), ví dụ như ở biến số **ngaync** của đối tượng **001** trong Bảng 2.1.

- Dữ liệu bị lặp lại (duplicated data): khi dữ liệu từ một hay nhiều đối tượng bị lặp lại một hay nhiều lần, thường do sai sót trong quá trình nhập liệu. Trong bộ dữ liệu, dữ liệu bị lặp lại thể hiện bằng việc một hay nhiều hàng dữ liệu bị lặp lại. Ví dụ như trong Bảng 2.1, dữ liệu của đối tượng **003** bị lặp lại thêm một lần.

- Giá trị không hợp lý (out-of-range/inappropriate data): khi dữ liệu có giá trị nằm ngoài giới hạn thông thường (quá lớn hay quá bé,

id	ngaysinh	ngaync	gioitinh	hct0	cannang_kg
001	2013-05-08	NA	1	49	10
002	2018-12-03	2018-02-03	Nữ	250	50
003	2013-12-20	2018-12-13	Nam	50	20
003	2013-12-20	2018-12-13	Nam	50	20

Table 2.1: Dữ liệu chưa được làm sạch

gặp ở các biến số liên tục) hoặc ngoài các giá trị cho phép (gặp ở các biến số phân nhóm) hoặc không phù hợp với thuộc tính của biến số (lẽ ra là giá trị số nhưng dữ liệu được nhập lại là chữ hoặc ngược lại). Đây có thể là lỗi trong quá trình thu thập dữ liệu hoặc khi nhập liệu. Ví dụ như trong Bảng 2.1, dữ liệu về **gioitinh** của đối tượng **001** được ghi nhận là **1**, trong khi đây là biến số về giới tính, vốn chỉ có hai giá trị **Nữ** hoặc **Nam**.

- Không tương hợp giữa các biến số (inconsistent data): khi dữ liệu có giá trị không phù hợp trong mối tương quan với các biến số khác trong bộ dữ liệu. Đây có thể là lỗi trong quá trình thu thập dữ liệu hoặc khi nhập liệu. Lỗi này khó phát hiện hơn so với các lỗi còn lại. Ví dụ như trong Bảng 2.1, đối tượng **002** có **ngaync** (ngày vào nghiên cứu) là 2018-02-03 (ngày 03 tháng 02 năm 2018) trong khi **ngaysinh** (ngày sinh) lại là 2018-12-03 (ngày 03 tháng 12 năm 2018), nghĩa là đối tượng tham gia nghiên cứu trước khi sinh.

2.2 Phương pháp phát hiện lỗi

Để phát hiện các lỗi đã nêu trên, chúng ta có thể sử dụng các công cụ của thống kê mô tả, qua các chỉ số thống kê mô tả và/hoặc các biểu đồ.

2.2.1 Dữ liệu bị thiếu/mất

Được phát hiện bằng cách mô tả số giá trị có trong bộ dữ liệu và so sánh với số giá trị mong đợi. Hiện nay, các phần mềm thống kê đều cho phép mô tả số giá trị bị thiếu/mất, nếu các giá trị này được mã hoá ở dạng mà các phần mềm thống kê hiểu (khoảng trắng, hoặc NA đối với R). Nếu trong giai đoạn nhập liệu, giá trị bị thiếu/mất được mã hoá theo cách khác (điền vào một giá trị nào đó, ví dụ 9, 99, 999) thì trước khi dùng phần mềm thống kê để mô tả dữ liệu, cần định dạng dữ liệu về dạng mà phần mềm thống kê hiểu.

Ở ví dụ về bộ dữ liệu trong Bảng 2.1, bằng phần mềm R, chúng ta có thể dùng lệnh `summary()` để mô tả dữ liệu. Kết quả được trình bày dưới đây cho thấy có một giá trị bị thiếu/mất (NA) ở biến số **ngaync**.

```
summary(dat)
```

```
##    id          ngaysinh        ngaync
##  001:1   2013-05-08:1   2018-02-03:1
##  002:1   2013-12-20:2   2018-12-13:2
##  003:2   2018-12-03:1   NA's      :1
##
##
##
##  gioitinh      hct0          cannang_kg
##  1  :1    Min.   : 49.00   Min.   :10.0
##  Nam:2    1st Qu.: 49.75   1st Qu.:17.5
##  Nữ :1    Median : 50.00   Median :20.0
##           Mean   : 99.75   Mean   :25.0
##           3rd Qu.:100.00   3rd Qu.:27.5
##           Max.   :250.00   Max.   :50.0
```

2.2.2 Dữ liệu bị lặp lại

Được phát hiện bằng cách mô tả số giá trị có trong bộ dữ liệu và so sánh với số giá trị mong đợi. Lỗi này cũng có thể phát hiện bằng mắt thường khi đã xếp dữ liệu theo thứ tự của mã số nghiên cứu, hoặc mô tả biến số mã số nghiên cứu (xem số giá trị mã số nghiên cứu khác nhau). Ngoài ra, một số phần mềm thống kê có thể có câu lệnh để kiểm tra dữ liệu bị lặp lại.

Ở ví dụ về bộ dữ liệu trong Bảng 2.1, bằng phần mềm R, chúng ta có thể dùng các cách sau:

- Kiểm tra số hàng của bộ dữ liệu

```
nrow(dat)
```

```
## [1] 4
```

Có 3 đối tượng nhưng bộ dữ liệu có 4 hàng.

- Mô tả biến số mã số nghiên cứu (ví dụ dùng lệnh `describe()` trong package Hmisc)

```
Hmisc::describe(dat$id)
```

```
## dat$id
##        n  missing distinct
##        4        0        3
##
## Value         1    2    3
## Frequency     1    1    2
## Proportion 0.25 0.25 0.50
```

Biến số **id** (mã số nghiên cứu) có 4 giá trị, nhưng chỉ có 3 giá trị khác nhau (nghĩa là có một giá trị bị lặp lại).

- Kiểm tra số đối tượng nghiên cứu bằng cách kiểm tra số giá trị mã số nghiên cứu khác nhau

```
length(dat$id)
```

```
## [1] 4
```

```
length(unique(dat$id))
```

```
## [1] 3
```

Biến số **id** (mã số nghiên cứu) có 4 giá trị, nhưng chỉ có 3 giá trị khác nhau (nghĩa là có một giá trị bị lặp lại).

- Kiểm tra dữ liệu lặp lại bằng lệnh `anyDuplicated` hoặc `duplicated`

```
anyDuplicated(dat)
```

```
## [1] 4
```

```
duplicated(dat)
```

```
## [1] FALSE FALSE FALSE  TRUE
```

Dữ liệu ở hàng thứ 4 trong bộ dữ liệu **dat** là dữ liệu lặp lại.

2.2.3 Giá trị không hợp lý

Được phát hiện bằng cách mô tả tất cả các giá trị trong biến số bằng các chỉ số hoặc biểu đồ, từ đó tìm ra các giá trị khác biệt so với các giá trị còn lại của biến số.

Ở ví dụ về bộ dữ liệu trong Bảng 2.1, bằng phần mềm R, chúng ta có thể dùng các cách sau:

- Mô tả các giá trị khác nhau của biến số bằng lệnh `unique()` hoặc `table()` (chỉ thích hợp với biến số có ít giá trị khác nhau, ví dụ biến số phân nhóm)

```
unique(dat$gioitinh)
```

```
## [1] 1   Nữ  Nam
## Levels: 1 Nam Nữ
```

```
table(dat$gioitinh)
```

```
##
##   1 Nam  Nữ
##   1   2   1
```

Biến số **gioitinh** (giới tính) có 3 giá trị: 1, Nữ, Nam; trong đó giá trị **1** không phù hợp.

- Mô tả khoảng giá trị, giá trị nhỏ nhất, giá trị lớn nhất để tìm các giá trị quá lớn hoặc quá bé so với mong đợi

```
summary(dat$hct0)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.
##   49.00   49.75   50.00   99.75  100.00
##    Max.
##  250.00
```

```
range(dat$hct0)
```

```
## [1]  49 250
```

```
min(dat$hct0)
```

```
## [1] 49
```

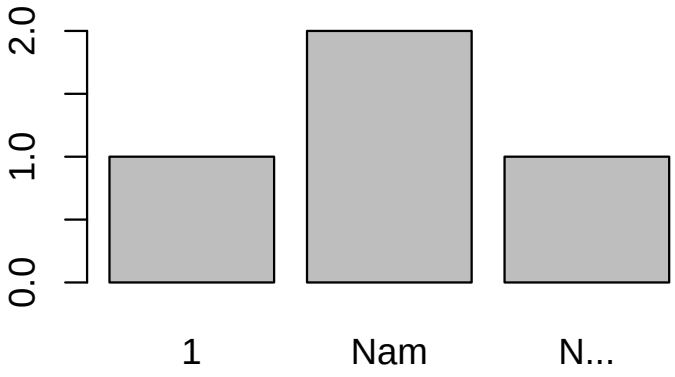
```
max(dat$hct0)
```

```
## [1] 250
```

Biến số **hct0** (dung tích hồng cầu ở thời điểm lúc mới vào nghiên cứu) có khoảng giá trị từ 49% (giá trị nhỏ nhất) đến 250% (giá trị lớn nhất), trong đó giá trị **250** là không phù hợp với giá trị mong đợi của dung tích hồng cầu.

- Mô tả giá trị bằng biểu đồ cột (với biến số phân nhóm)

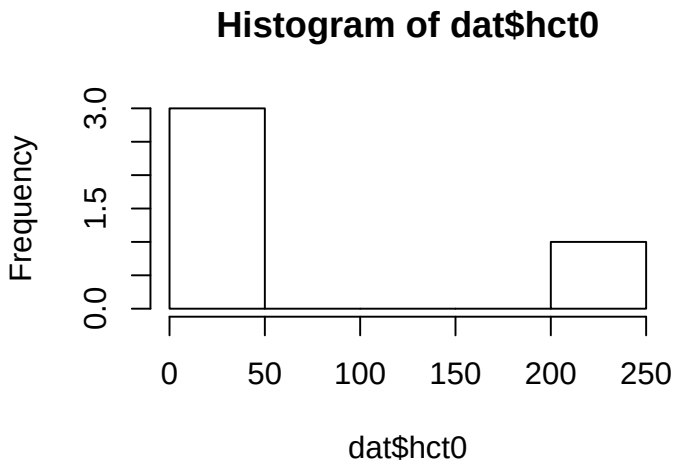
```
barplot(table(dat$gioitinh))
```



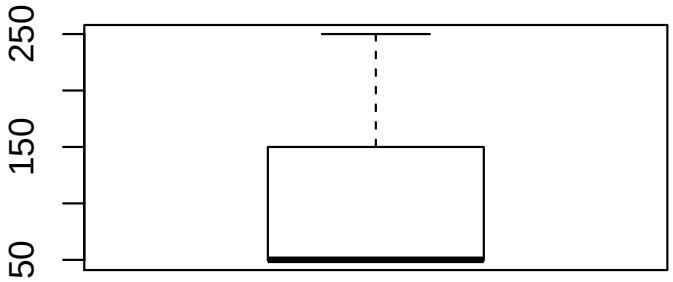
Biến số **gioitinh** có giá trị **1**, không phù hợp với các giá trị còn lại.

- Mô tả giá trị bằng histogram hoặc boxplot (với biến số liên tục)

```
hist(dat$hct0)
```




```
boxplot(dat$hct0)
```



Biến số **hct0** có giá trị **250**, rất khác biệt so với các giá trị còn lại và nằm ngoài khoảng giá trị mong đợi cho dung tích hồng cầu.

2.2.4 Không tương hợp giữa các biến số

Được phát hiện bằng cách mô tả biến số thứ cấp tạo ra từ các biến số liên quan, hoặc dùng biểu đồ mô tả mối liên hệ giữa hai biến số với nhau.

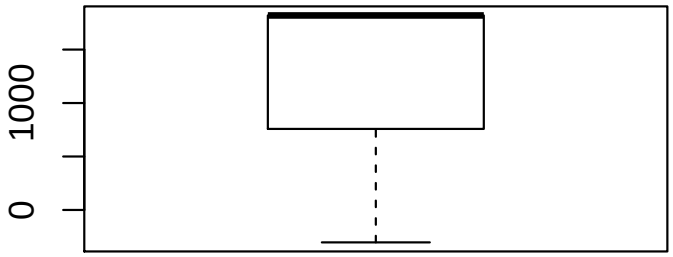
Ở ví dụ về bộ dữ liệu trong Bảng 2.1, bằng phần mềm R, chúng ta có thể dùng các cách sau:

- Mô tả biến số thứ cấp: ví dụ mô tả biến số **tuoi** (tuổi lúc vào nghiên cứu), được tính từ biến số **ngaysinh** và **ngaync**

```
dat$tuoi <- as.numeric(difftime(as.Date(dat$ngaync),
    as.Date(dat$ngaysinh), units = "days"))
summary(dat$tuoi)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.
##    -303     758    1819    1112    1819
##    Max.    NA's
##    1819       1
```

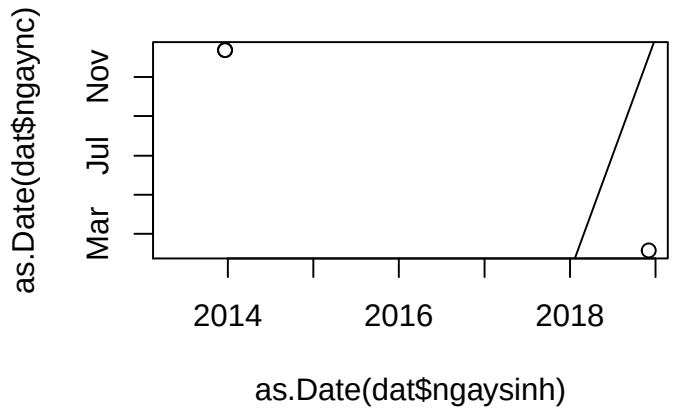
```
boxplot(dat$tuoi)
```



Có một trường hợp **tuoi** (tuổi lúc vào nghiên cứu) < 0.

- Dùng biểu đồ mô tả mối liên hệ giữa hai biến số **ngaysinh** và **ngaync**

```
## phân tán đồ (scatterplot) của ngaysinh và
## ngaync
plot(x = as.Date(dat$ngaysinh), y = as.Date(dat$ngaync))
## đường thẳng qua các điểm ngaysinh bằng
## ngaync
abline(a = 0, b = 1)
```



Có một trường hợp **ngaync** nhỏ hơn **ngaysinh** (nằm phía dưới đường thẳng đi qua các giá trị **ngaync** bằng **ngaysinh**) trong khi mong đợi **ngaync** phải lớn hơn **ngaysinh** (đối tượng phải được sinh ra trước khi vào nghiên cứu).

2.3 Quy trình kiểm tra và làm sạch dữ liệu

Để phát hiện và loại bỏ tối đa các sai sót có thể có nhằm có được bộ dữ liệu sạch và đáng tin cậy cho phân tích, chúng ta cần kiểm tra và làm sạch dữ liệu một cách có hệ thống theo 3 bước như sau:

2.3.1 Tìm lỗi

Dựa vào các phương pháp thống kê mô tả (sử dụng chỉ số và/hoặc biểu đồ, như đã trình bày ở mục 2.2) để tìm các lỗi thường gặp (như đã trình bày ở mục 2.1).

Các lỗi tìm thấy nên được tổng hợp lại trong một file dữ liệu trước khi tiến hành bước tiếp theo. File dữ liệu này nên bao gồm các thông tin sau:

- Lỗi tìm thấy là gì? (loại lỗi)
- Lỗi ở đâu? (biến số liên quan, mã số nghiên cứu liên quan)

2.3.2 Chẩn đoán lỗi

Trong bước này, chúng ta sẽ dựa vào kiến thức chuyên môn và hiểu biết về nghiên cứu và quá trình thu thập dữ liệu - nhập liệu trong nghiên cứu để đánh giá xem các lỗi tìm thấy trong dữ liệu có thực sự là lỗi hay không. Khi đánh giá, cần đối chiếu với dữ liệu gốc (dữ liệu thu thập trên phiếu thu thập bằng giấy) nếu có. Sau khi đánh giá, các lỗi này có thể được phân loại thành:

- Lỗi thực sự
- Không phải lỗi (chỉ là giá trị hiếm gặp)
- Không chắc: không thể xác định có phải là lỗi không dựa vào các thông tin hiện có

2.3.3 Sửa lỗi (làm sạch dữ liệu)

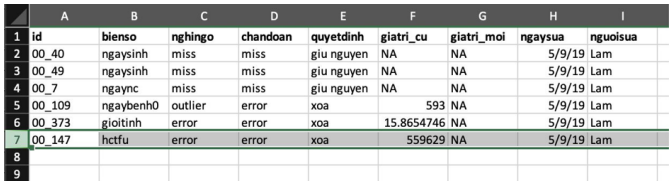
Tùy theo phân loại ở bước chẩn đoán lỗi, chúng ta sẽ có cách sửa lỗi phù hợp:

- Lỗi thực sự: sửa lại thành giá trị đúng (nếu có cơ sở cho giá trị đúng, ví dụ như giá trị gốc được lưu trên giấy) hoặc xoá hẳn giá trị sai (nếu không thể xác định được giá trị đúng).
- Không phải lỗi: giữ nguyên, không thay đổi giá trị trong dữ liệu.
- Không chắc: cân nhắc giữa giữ nguyên và xoá hẳn giá trị này.

Các bước trên và mọi thay đổi trong bộ dữ liệu nên được ghi nhận lại với các thông tin như:

- Lỗi tìm thấy là gì? (loại lỗi)
- Lỗi ở đâu? (biến số liên quan, mã số nghiên cứu liên quan)
- Chẩn đoán lỗi là gì?
- Người chẩn đoán lỗi?
- Quyết định đưa ra với lỗi?
- Người đưa ra quyết định?
- Nếu có sửa lỗi thì giá trị cũ là gì, giá trị mới là gì? người sửa lỗi? ngày sửa lỗi?

Vấn đề này đặc biệt quan trọng khi làm việc trong một nhóm và nhằm đảm bảo tính minh bạch và rõ ràng trong quản lý dữ liệu. Một ví dụ về file dữ liệu ghi nhận các bước liên quan đến việc kiểm tra và làm sạch dữ liệu như sau:



	A	B	C	D	E	F	G	H	I
1	**id**	**bienso**	**nghingo**	**chandoan**	**quyetdinh**	**giatri_cu**	**giatri_moi**	**ngaysua**	**nguoisua**
2	00_40	ngaysinh	miss	miss	giu nguyen	NA	NA	5/9/19	Lam
3	00_49	ngaysinh	miss	miss	giu nguyen	NA	NA	5/9/19	Lam
4	00_7	ngaync	miss	miss	giu nguyen	NA	NA	5/9/19	Lam
5	00_109	ngaybenh0	outlier	error	xoa	593	NA	5/9/19	Lam
6	00_373	gioitinh	error	error	xoa	15.8654746	NA	5/9/19	Lam
7	00_147	hctfu	error	error	xoa	559629	NA	5/9/19	Lam
8									
9									

Figure 2.1: File ghi nhận việc kiểm tra và làm sạch dữ liệu

3
Chuẩn bị dữ liệu cho phân tích

Dữ liệu sau khi làm sạch có thể chưa phân tích được ngay, mà cần phải biến đổi để phù hợp với mục tiêu và kỹ thuật phân tích.

3.1 Tạo thêm biến số mới

Tạo thêm biến số mới dựa trên các biến số sẵn có: Ví dụ thu thập ngày tháng năm sinh, cần tạo thêm biến số là tuổi; hoặc thu thập cân nặng và chiều cao, cần tạo thêm biến BMI để phân tích.

3.2 Định dạng biến số

(phần này em không chắc là trình bày nội dung gì)

4
Chuyển dạng dữ liệu cho phù hợp với phân tích

4.1 Kết hợp các bảng dữ liệu với nhau

Dữ liệu có thể nằm ở nhiều tập tin khác nhau hoặc do nhiều người nhập liệu, cần phải ghép lại thành một bảng dữ liệu hoàn chỉnh để có đầy đủ thông tin phục vụ cho việc phân tích.

4.2 Dữ liệu dạng dài hay dạng rộng

Ví dụ ta có bảng dữ liệu như sau gồm 4 cột:

nosmote_gbm	smote_gbm	nosmote_normalized_gbm	smote_normalized_gbm
0.5194016	0.0000000	0.5194016	0.0000000
0.0000000	0.2575760	0.0000000	0.2638353
0.5476280	0.2928422	0.5476280	0.0000000
0.0000000	0.5490189	0.0000000	0.3181968
0.9242180	0.3156049	0.9242180	0.3567285
2.6414027	10.1025078	2.6414027	9.0878182
1.0600750	1.0623258	1.0600750	1.0092043
7.7281674	1.0942214	7.7281674	3.1325067
1.5369658	4.0954080	1.5369658	2.8702463
0.0000000	3.0846788	0.0000000	5.5248105
0.5743480	2.0454574	0.5743480	3.1771070
1.6205714	0.0000000	1.6205714	0.0000000

Figure 4.1: Ví dụ về dữ liệu dạng rộng

Ta có thể biến đổi dữ liệu từ có 4 cột này thành chỉ có 2 cột như sau:

Các giá trị trong cột Method là nosmote_gbm, smote_gbm, nosmote_normalized_gbm, smote_normalized_gbm chính là tên của 4 cột ban đầu. Giá trị trong cột Value là những con số tương ứng với mỗi phương pháp trong Method. Việc biến đổi này không làm mất đi thông tin trong bộ dữ liệu mà chỉ làm thay đổi hình dạng trình bày

Method	Value
nosmote_gbm	0.5194016
nosmote_gbm	0.0000000
nosmote_gbm	0.5476280
nosmote_gbm	0.0000000
nosmote_gbm	0.9242180
nosmote_gbm	2.6414027
nosmote_gbm	1.0600750
nosmote_gbm	7.7281674
nosmote_gbm	1.5369658
nosmote_gbm	0.0000000
nosmote_gbm	0.5743480
nosmote_gbm	1.6205714

Figure 4.2: Ví dụ về dữ liệu dạng dài

thông tin. Dữ liệu to bè theo chiều ngang (gồm nhiều cột) gọi là dữ liệu dạng rộng (wide format). Dữ liệu sau khi biến đổi chỉ có 2 cột nhưng sẽ kéo dài xuống thành rất nhiều dòng gọi là dữ liệu dạng dài (long format).

5
Phân tích mô tả

6
Kiểm định giả thuyết thống kê

6.1 Nội dung dự kiến

- Nguyên tắc kiểm định thống kê
- Tổng quan về các kiểm định thống kê thường gặp
- Trị số p: ý nghĩa và giá trị
- Mối quan hệ giữa trị số p và khoảng tin cậy

7
Mô hình thống kê

7.1 Nội dung dự kiến

- Ước lượng, khoảng ước lượng, khoảng tin cậy
- Mô hình thống kê là gì? So sánh với kiểm định thống kê
- Nguyên tắc xây dựng mô hình thống kê
- Tổng quan các mô hình thống kê thường gặp
- Cách diễn giải ý nghĩa ước lượng và khoảng ước lượng
- Cách lập kế hoạch xây dựng mô hình thống kê
- Cách diễn giải kết quả từ mô hình thống kê
- Cách lựa chọn mô hình thống kê phù hợp

8
Viết báo cáo phân tích phân tích

8.1 Nội dung dự kiến

- Đọc kết quả phân tích như thế nào?
- Diễn giải kết quả phân tích như thế nào?
- Những sai lầm thường gặp khi đọc và diễn giải kết quả phân tích?

9

Đọc và diễn giải kết quả phân tích, Lập kế hoạch phân tích

9.1 Nội dung dự kiến

- Kế hoạch phân tích là gì?
- Vì sao phải lập kế hoạch phân tích?
- Lập kế hoạch phân tích như thế nào? WHO-WHAT-WHEN-WHERE-HOW
- Kế hoạch phân tích mẫu

10
Thu thập dữ liệu

10.1 Nội dung dự kiến

- Những điểm càn lưu ý khi thu thập dữ liệu
- Thu thập dữ liệu như thế nào? WHO-WHAT-WHEN-WHERE-HOW
- Epidata

11
Tóm tắt

11.1 Nội dung dự kiến

- Tóm tắt những điểm chính trong quá trình phân tích dữ liệu

12
Giới thiệu về R và R Commander

Ước tính cỡ mẫu

This book aims to provide tips in doing basic data analysis

13
Bibliography

Lam, P. K., Ngoc, T. V., Thuy, T. T. T., Van, N. T. H., Thuy, T. T. N., Tam, D. T. H., Dung, N. M., Tien, N. T. H., Kieu, N. T. T., Simmons, C., Wills, B., and Wolbers, M. (2017). The value of daily platelet counts for predicting dengue shock syndrome: Results from a prospective observational study of 2301 vietnamese children with dengue. *PLOS Neglected Tropical Diseases*, 11(4):e0005498.

Xie, Y. (2015). *Dynamic Documents with R and knitr*. Chapman and Hall/CRC, Boca Raton, Florida, 2nd edition. ISBN 978-1498716963.

Xie, Y. (2019). *bookdown: Authoring Books and Technical Documents with R Markdown*. R package version 0.11.