

1日目（10:00～14:30）Rを用いた多施設臨床データの統合・前処理と解析チュートリアル

国際脳MRI・臨床データ解析チュートリアル

# R Markdownによる臨床情報QC

—R Markdownの入門と実践

45 PAGES

上野雄己 | 東京大学

東京大学駒場キャンパス 令和2年2月6日

1

## 講義概要

---

Point 1 R Markdownを学ぶ

Point 2 R Markdownを実践する

- R Markdownの使用方法を学ぶ
- 臨床情報Quality Checkへの応用

---

4

2



3



4

## Webサイトのご紹介

### R Markdown入門 - kazutan on web

[ [https://kazutan.github.io/kazutanR/Rmd\\_intro.html](https://kazutan.github.io/kazutanR/Rmd_intro.html) ]



The screenshot shows a web browser window with a dark theme. The page title is "R Markdown入門". The left sidebar contains a navigation menu with items like "はじめに", "R Markdownとは", "R Markdownの書き方", "R Markdownのインストール", "R Markdownでスライド生成", "FAQ", and "お問い合わせ". The main content area is titled "R Markdown入門" and "はじめに". It includes a "今日の内容" section with a list of topics: "R Markdownとは", "Markdown記法", "R Markdownでドキュメント生成", "R Markdownでスライド生成", and "実際にコードを書いてみる". Below that is a "留意事項" section with a paragraph of text. Further down are sections for "R Markdownとは" and "Rでドキュメント生成".

7

5

## R Markdownで何ができるのか？

Rで分析・可視化を行い、その成果を報告書としてドキュメントにまとめることが可能

- Warning

① 分析結果やプロットを手作業で行う...
- Warning

② 共同研究者と共有するとき...
- Warning

③ 100個以上のグラフをまとめるとき...

8

6

## 特に重要な点

---

- Important  
**① 分析の間違いに気づける**
- Important  
**② 他の研究者と共有ができる**
- Important  
**③ 分析の再現可能性がある**

---

10

7

**R Markdown**を実践する

8

## 分析の前に

### RとRstudioをダウンロード

#### Rダウンロードサイト

[ <https://cran.r-project.org> ]

#### R studioダウンロードサイト

[ <https://www.rstudio.com/products/rstudio/download/> ]



#### ※ RとRstudioの使用方法：参考サイト

[ <https://kazutan.github.io/SappoRoR5/r-intro1.html> ]

### R Markdownの使用の際



#### RstudioのGlobal Optionsにて、UTF-8に文字コードを変更

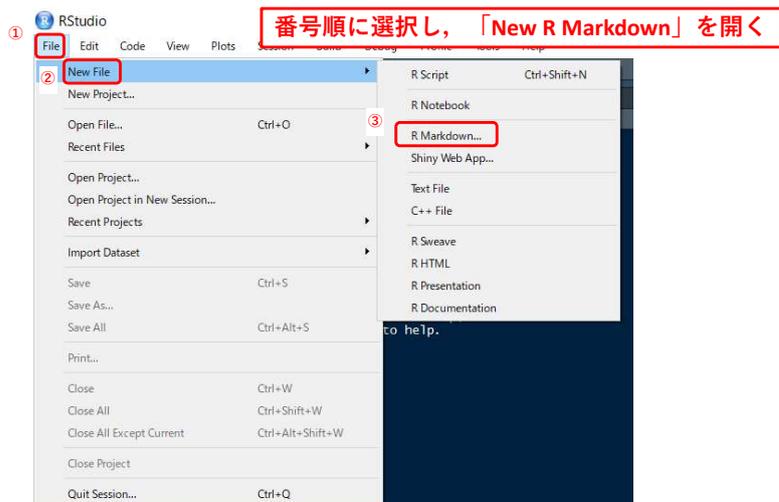
[ R Markdownのスクリプトが文字化けを起こす可能性が高いので必ず変更すること ]

11

9

## R Markdownの起動方法 1

チュートリアル実施



14

10

## R Markdownの起動方法 2

チュートリアル実施

Titleの入力

Title: Untitled

Author: 上野雄己

Authorの入力

Default Output Format:

HTML

HTMLの選択

Recommended format for authoring (you can switch to PDF or Word output anytime).

PDF

PDF output requires a recent version of TeX (2013+ on OS X, TeX Live 2013+ on Linux).

Word

Previewing Word documents requires an installation of MS Word (or Libre/Open Office on Linux).

「OK」を選択すると、Rstudio上にスクリプトが表記

OK Cancel

15

11

## R Markdownの起動方法 3

チュートリアル実施

分析前に、「名前を付けて保存」

```
## Title and Author Information
title("臨床情報QC")
data("2020011515")
output: html_document

## R Markdown
This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see https://rmarkdown.rstudio.com.

When you click the "Run" button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```{r}
summary(Cars)
```

## Including Plots
You can also embed plots, for example:

```{r, pressure, echo=FALSE}
plot(pressure)
```

Note that the echo = FALSE parameter was added to the code chunk to prevent printing of the R code that generated the plot.

R version 3.5.3 (2019-03-15) -- "Great Truth"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: x86_64-mingw32/x64 (64-bit)
R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
type 'license()' or 'licence()' for distribution details.
R is a collaborative project with many contributors.
type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.
type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
type 'q()' to quit R.

>
```

Rstudio上にスクリプトが表記

16

12

## R Markdownの構造

チュートリアル実施

フロントマター部分：  
ドキュメント全体設定を記述

Markdownによる文章の記述

Rチャンク（Rコード入力）

Knitrパッケージで  
Rコードを評価

Pandocで  
ドキュメントへ変換

.Rmd → .md → html, docs, pdf...

17

13

## R Markdownの実施

チュートリアル実施

「Knit」を選択

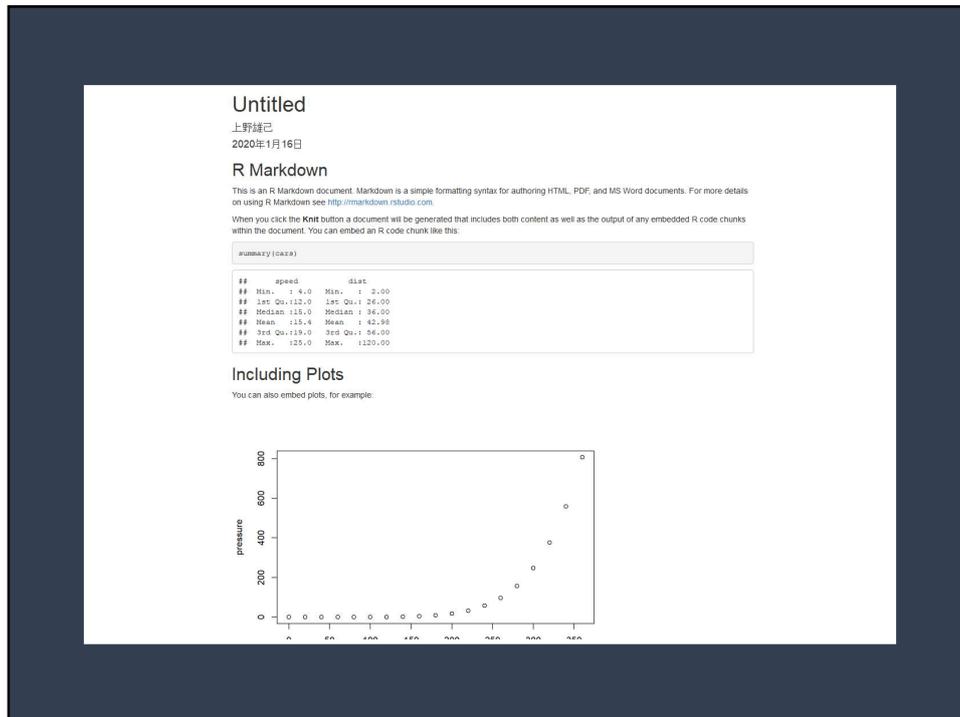
「Knit to HTML」を選択

②を選択すると、分析が開始

「100%」「Output created」  
が表記されればHTMLにて出力

18

14



15

## R Markdownの使用手順

- 1 R Markdownファイルを準備
- 2 文章（Markdown）やRチャンクを書く
- 3 Knitしてレンタリング
- 4 出力されたドキュメントをチェック

19

16

## ドキュメントの設定 1

```

1 title: "Untitled"
2 author: "上野雄己"
3 date: "2019年10月16日"
4 output: html_document
5
6 # A R Markdown document
7 # To learn more about R Markdown, see https://rmarkdown.rstudio.com
8 # For more on how to use R Markdown, see https://rmarkdown.rstudio.com/faq.html
9 # You can also embed plots, for example:
10 # plot(mtcars, aes(wt, mpg))
11 # Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R output in the browser.
12
13 ---
14 title: "Untitled"
15 author: "上野雄己"
16 date: "2019年10月16日"
17 output: html_document
18 ---
  
```

17

## ドキュメントの設定 2

```

1 ---
2 title: "臨床情報Quality Check | Rプログラム"
3 author: "上野雄己 | 東京大学"
4 date: "r format(Sys.time(), '%Y/%m/%d') "
5 output:
6   html_document:
7     toc: true
8     toc_float: true
9     df_print: paged
10    keep_me: true
11    theme: united
12 ---
  
```

1. ドキュメントのタイトル
2. ドキュメントの著者
3. ドキュメントの日付
4. 出力形式を指定
5. htmlに出力
6. 文書の先頭にコンテンツ一覧表を表示
7. 見出しレベル（ここでは3レベル）
8. 見出しリスト（サイドメニュー）
9. data.frame型のデータを表示する方法（paged：ページ区切りのついたインタラクティブな表形式）
10. Markdownファイルの保存
11. R Markdownのデザイン（おススメはflatly, united, journal）



細かい設定はこの書籍をご参照ください

18

## Markdownの基本 1

チュートリアル実施

### 文書を記述する軽量マークアップ言語のひとつ

#### 1 見出し

```
# レベル1  
## レベル2  
### レベル3  
#### レベル4
```

#### 2 箇条書き

```
- 箇条書き1  
  - 箇条書き1-1  
  - 箇条書き2  
  * 箇条書き2-1
```

1. -や\*と半角スペースの後に項目内容を記述
2. 行頭の空白4つで1レベル下がる

22

19

## Markdownの基本 2

チュートリアル実施

### 3 段落と強制改行

これは1つの段落として機能します。

この文は文末で改行していますが、実際には改行されません。そのためこの文は前の文とつながります。

この文は（見えませんが）文末に半角スペースが2つ挿入されています。そのためこの文は強制改行されます。

1. 上下に空行を挟むと、そこを段落として認識
2. 行末に半角スペースを2つ以上いれると強制改行

### 4 強調表示

```
*斜体*, もひとつ 斜体_  
**太字**, もひとつ __太字__
```

23

20

## Rチャンクの基本 1

Markdownのドキュメント内にRのコードを埋め込んで実行できること

24

21

## Rチャンクの基本 2

```
```{r チャンクラベル, チャンクオプション}  
(Rのコード)  
```
```

例：echo=TRUE, warning=FALSE, message=FALSE

### チャンクオプション一覧

[松村他（2018）の書籍参照]

| オプション名  | デフォルト値 | 内容   |
|---------|--------|--|
| eval    | TRUE   | チャンク内のコードを評価するかどうかを指定                      |
| echo    | TRUE   | チャンク内のコードをドキュメントに表示させるかどうかを指定              |
| include | TRUE   | 実行後にチャンクの内容をドキュメントに含めるかどうかを指定              |
| result  | markup | 実行結果をどう出力するかを指定                            |
| cache   | FALSE  | 実行した結果をキャッシュするかどうかを指定                      |
| error   | FALSE  | Rコードの実行時のエラーをどうするか指定                       |
| warning | TRUE   | Rコードを実行した際のwarning（警告）をドキュメントに表示させるかどうかを指定 |
| message | TRUE   | コードを実行した際のmessageをドキュメントに表示させるかどうかを指定      |

25

22

## R Markdownの使用上の注意点

---

- Warning  
**①** まずはRの使用方法を学ぶこと
- Warning  
**②** 警告・エラーが頻繁に起きること
- Warning  
**③** パッケージの競合による不具合

26

23

臨床情報Quality Check

24

## 臨床情報QCのRプログラム概要1-1

Point

### 1 CSVデータの取り込み

-DTパッケージ

Point

### 2 欠損値の確認方法

-mice, VIM, dplyrパッケージ

Point

### 3 外れ値の確認方法

-summarytools, ggplot2, ggrepelパッケージ



29

25

## 臨床情報QCのRプログラム概要1-2

```
1 #
2 title: "臨床情報Quality Check | Rプログラム"
3 author: "上野雄己 | 東京大学"
4 date: "rmarkdown::today(), Sys.Date()"
5 output:
6   html_document:
7     toc: true
8     toc_depth: 3
9     toc_float: true
10    keep_me: true
11    theme: united
12 ---
13 ## **1. データの確認**
14 ---
15 ## **1-1. csvデータを取り込む**
16 ""[r, echo=TRUE, warning=FALSE, message=FALSE]
17 rdata <- read.csv("rdata.csv", header=TRUE, sep=
18 ",")
19 ---
20 ## **1-2. csvデータの確認**
21 ""[r, echo=TRUE, warning=FALSE, message=FALSE]
22 library(dplyr) # データフレームを操作するためのパッケージをロードする
23 datatable(rdata)
24 ---
25 ## **2. 欠損値の確認**
26 ---
27 ## **2-1. miceパッケージ**
28 ""[r, echo=TRUE, warning=FALSE, message=FALSE]
29 library(mice) # miceパッケージをロードする
30 # 欠損値Quality Check | Rプログラム
```

臨床情報Quality Check | Rプログラム  
上野雄己 | 東京大学  
2020/01/17

### 1. データの確認

#### 1-1. CSVデータを取り込む

```
rdata <- read.csv("rdata.csv", header=TRUE, sep=",")
```

#### 1-2. CSVデータの確認

```
library(dplyr) # データフレームを操作するためのパッケージをロードする
datatable(rdata)
```

| ID | Sex | Age | Height | Weight | SBP | BS  |     |
|----|-----|-----|--------|--------|-----|-----|-----|
| 1  | 1   | 0   | 11     | 120    | 35  | 100 | 99  |
| 2  | 2   | 0   | 10     | 134    | 39  | 103 | 103 |
| 3  | 3   | 1   | 9      | 178    | 60  | 521 | 101 |
| 4  | 4   | 1   | 8      | 147    | 40  |     |     |
| 5  | 5   | 0   | 9      | 120    | 29  | 98  | 106 |
| 6  | 6   | 1   | 9      | 135    |     | 111 | 170 |
| 7  | 7   | 0   | 10     | 70     | 29  | 150 | 40  |
| 8  | 8   | 1   | 11     | 119    | 10  | 101 | 88  |
| 9  | 9   | 1   | 8      | 131    | 25  | 93  | 92  |
| 10 | 10  | 1   | 8      | 125    | 70  | 79  |     |

注) 本チュートリアルで使用するデータは以下のWebサイトを参照・加工し用いた。  
<http://jojoshin.hatenablog.com/entry/2017/02/03/220118>

30

26

## ダウンロードパッケージ（事前確認）

① Rコード入力

```
install.packages(c("DT","mice","VIM","dplyr",  
"summarytools","ggplot2","ggrepel"))
```

パッケージ

- ① 「DT」
- ② 「mice」
- ③ 「VIM」
- ④ 「dplyr」
- ⑤ 「summarytools」
- ⑥ 「ggplot2」
- ⑦ 「ggrepel」

注) 7つ全てダウンロード

② 「Packages」を選択

③上記のPackageの名前を1つずつ入力

④Installを選択

①Packagesを選択

②Installを選択

31

27

## Csvデータの取り込み<sub>1</sub>

チュートリアル実施

```
データ名 <- read.csv("データ名.csv",header=TRUE,sep=",")
```

- CSVデータの場合はread.csv関数で取り込む。データ名称はCSVファイルの名称を入力する。
- 1行目に変数名が書いてある場合には[header=TRUE], ない場合には[header=FALSE]とする。
- sepはデータ間の区切り(separator)を意味し, [sep=","]を指定することで区切ることができる。

32

28

## Csvデータの取り込み<sub>2</sub>

チュートリアル実施

```
install.packages("DT") #DTパッケージをダウンロード  
library(DT) #DTパッケージを読み込む  
datatable(データ名)
```

- DTパッケージのdatatable関数を使用することで、データと全ての変数名を確認することができる。
- 空欄は欠損値を意味する。欠損値を指定したい場合には、[データ名[データ名==9999] <- NA]で行える。
- DTパッケージを使用することで、様々な表を作成することができる。以下のサイトをご参照ください。  
[ [https://kazutan.github.io/SappoRoR5/DT\\_demo.html](https://kazutan.github.io/SappoRoR5/DT_demo.html) ]

33

29

Show  entries Search:

|    | ID | Sex | Age | Height | Weight                        | SBP                           | BS                            |
|----|----|-----|-----|--------|-------------------------------|-------------------------------|-------------------------------|
| 1  | 1  | 0   | 11  | 120    | 35                            | 100                           | 99                            |
| 2  | 2  | 0   | 10  | 134    | 39                            | <input type="text" value=""/> | 103                           |
| 3  | 3  | 1   | 9   | 178    | 60                            | 121                           | 101                           |
| 4  | 4  | 1   | 8   | 147    | 40                            | <input type="text" value=""/> | <input type="text" value=""/> |
| 5  | 5  | 0   | 9   | 120    | 29                            | 98                            | 106                           |
| 6  | 6  | 1   | 9   | 135    | <input type="text" value=""/> | 111                           | 170                           |
| 7  | 7  | 0   | 10  | 70     | 29                            | 150                           | 40                            |
| 8  | 8  | 1   | 11  | 119    | 10                            | 101                           | 88                            |
| 9  | 9  | 1   | 8   | 131    | 25                            | 93                            | 92                            |
| 10 | 10 | 1   | 8   | 125    | <input type="text" value=""/> | 70                            | 79                            |

Showing 1 to 10 of 10 entries Previous  Next

注) ID = 被験者番号, Sex = 性別 (男性=0, 女性=1), Age = 年齢, Height = 身長, Weight = 体重, SBP = 最高血圧, BS = 血糖値

30

## 欠損値の確認方法 1

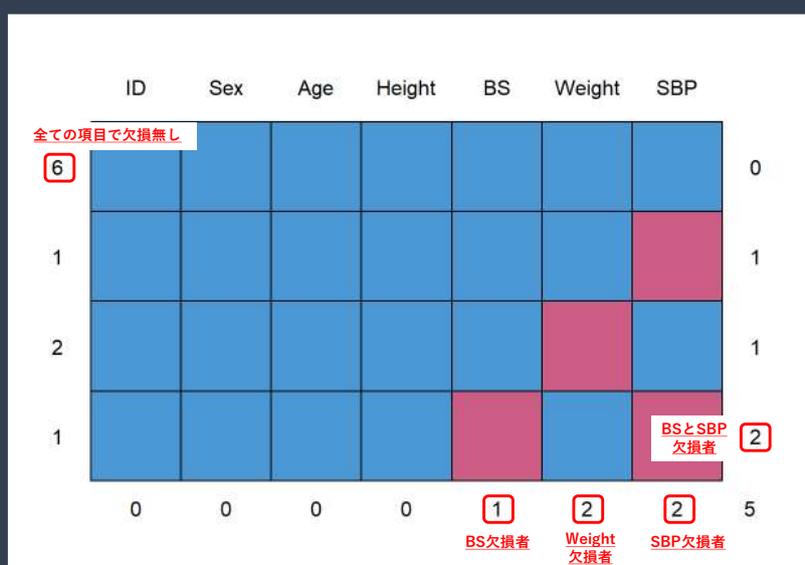
チュートリアル実施

```
install.packages("mice") #miceパッケージをダウンロード
library(mice) #miceパッケージを読み込む
md.pattern(データ名)
```

- miceパッケージのmd.pattern関数を使用することで、欠損値の有無（0:欠損有り，1:欠損無し）の組み合わせを表示できる。
- 参照データを用いた場合，全ての項目の欠損無しが6名，weightだけの欠損有りが2名，SBPだけの欠損有りが2名，BSだけの欠損有りが1名，BSとSBPの欠損有りが2名であった。
- なお，変数が多い場合，図が見つらくなるため，10個程度に留め，変数名は短くすることをお勧めする。

34

31



32

## 欠損値の確認方法 2

チュートリアル実施

`md.pairs(データ名)` *#miceパッケージを使用*

- miceパッケージの`md.pairs`関数で2項目の欠損有無の組み合わせごとに件数を表示できる。
- `r`は非欠損値, `m`は欠損値であり, 例えば`rm`は行(横)が非欠損値, 列(縦)が欠損値を示している。
- AgeとWeightを例にすると, ともに観察されたのは8名(`rr`), Ageのみ欠損者は0名(`rm`), Weightのみ欠損者は2名(`mr`), 両方とも欠損者は0名(`mm`)であった。

35

33

### 例えば, AgeとWeightの組み合わせ

注) `r`は非欠損値, `m`は欠損値, 先頭が行, 後頭が列

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|
| <pre>## \$rr ## ID Sex Age Height Weight SBP BS ## ID 10 10 10 10 8 8 9 ## Sex 10 10 10 10 8 8 9 ## Age 10 10 10 10 8 8 9 ## Height 10 10 10 10 8 8 9 ## Weight 8 8 8 8 8 6 7 ## SBP 8 8 8 8 6 8 8 ## BS 9 9 9 9 7 8 9</pre> |  |  |  |  |  |  |  | <pre>## \$mr ## ID Sex Age Height Weight SBP BS ## ID 0 0 0 0 0 0 0 ## Sex 0 0 0 0 0 0 0 ## Age 0 0 0 0 0 0 0 ## Height 0 0 0 0 0 0 0 ## Weight 2 2 2 2 0 2 2 ## SBP 2 2 2 2 2 0 1 ## BS 1 1 1 1 1 0 0</pre> |  |  |  |  |  |  |  |
| <pre>## \$rm ## ID Sex Age Height Weight SBP BS ## ID 0 0 0 0 2 2 1 ## Sex 0 0 0 0 2 2 1 ## Age 0 0 0 0 2 2 1 ## Height 0 0 0 0 2 2 1 ## Weight 0 0 0 0 0 2 1 ## SBP 0 0 0 0 2 0 0 ## BS 0 0 0 0 2 1 0</pre>                 |  |  |  |  |  |  |  | <pre>## \$mm ## ID Sex Age Height Weight SBP BS ## ID 0 0 0 0 0 0 0 ## Sex 0 0 0 0 0 0 0 ## Age 0 0 0 0 0 0 0 ## Height 0 0 0 0 0 0 0 ## Weight 0 0 0 0 2 0 0 ## SBP 0 0 0 0 0 2 1 ## BS 0 0 0 0 0 1 1</pre> |  |  |  |  |  |  |  |

34



## 欠損値の確認方法 4

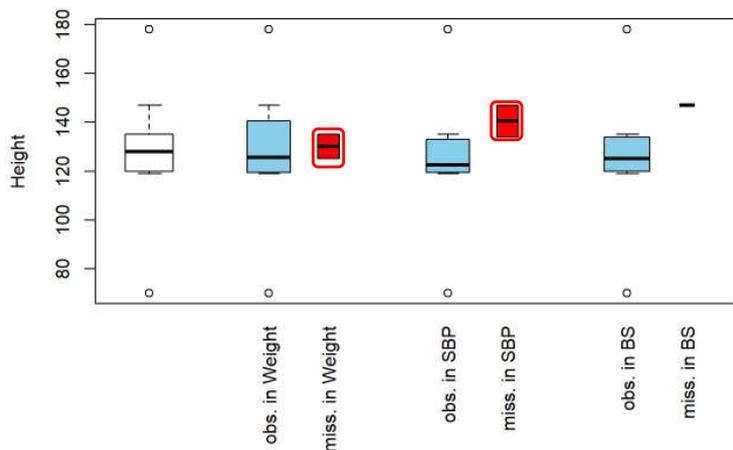
チュートリアル実施

```
pbox(データ名, pos=2, int=FALSE, cex=1.2) #VIMパッケージを使用
```

- VIMパッケージのpbox関数を使用することで、pbox関数で特定の項目とその他の項目間の関係について、欠損値の有無ごとに箱ひげ図をプロットしている。
- posでは確認したいデータが出力されるので、[pos=確認したいデータの番号]を入力する。ここではHeight(4)とする。
- [int=FALSE, cex=1.2]に関してはデフォルトのままにする（intは切り捨てて整数にする実数、cexは点の大きさを示し、デフォルトの何倍にするか指定）。

37

37



38

## 欠損値の確認方法 5

チュートリアル実施

```
is.na(Rdata1)
table(is.na(Rdata1))
データ名称[!complete.cases(データ名称),]
```

- パッケージのダウンロードは特になし。
- `is.na`関数で欠損値 (TRUE) と欠損値でない (FALSE) を確認することができる。
- `table`関数で欠損値 (TRUE) と欠損値でない (FALSE) , それぞれの数を算出できる。
- データ名称[!complete.cases(データ名称),]を指定することで欠損値が存在するIDとデータを表記できる。

38

39

```
is.na(Rdata)

##      ID Sex Age Height Weight SBP BS
## [1,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [2,] FALSE FALSE FALSE FALSE FALSE TRUE FALSE
## [3,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [4,] FALSE FALSE FALSE FALSE FALSE TRUE TRUE
## [5,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [6,] FALSE FALSE FALSE FALSE TRUE FALSE FALSE
## [7,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [8,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [9,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [10,] FALSE FALSE FALSE FALSE TRUE FALSE FALSE

table(is.na(Rdata))

## FALSE TRUE
##  65      5

Rdata[!complete.cases(Rdata),]

##      ID Sex Age Height Weight SBP BS
## 2  2  0  10  134  39 NA 103
## 4  4  1  8   147  40 NA  NA
## 6  6  1  9   135  NA 111 170
## 10 10 1  8   125  NA  70  79
```

40

## 欠損値の確認方法 6

チュートリアル実施

```
install.packages("dplyr") #dplyrパッケージをダウンロード  
library(dplyr) #dplyrパッケージを読み込む  
変数名 <- filter(データ名, is.na(変数名))$ID  
変数名
```

- dplyrパッケージのfilter関数を用いることで、欠損値のIDのみを示すことができる。
- is.na関数は先ほど説明した通り、欠損値を探す時に利用され、\$IDと結合することで、ある特定の変数の欠損しているIDを抽出することができる。

39

41

```
library(dplyr) #dplyrパッケージを読み込む  
Weight_missing <- filter(Rdata, is.na(Weight))$ID  
Weight_missing
```

```
## [1] 6 10
```

```
SBP_missing <- filter(Rdata, is.na(SBP))$ID  
SBP_missing
```

```
## [1] 2 4
```

```
BS_missing <- filter(Rdata, is.na(BS))$ID  
BS_missing
```

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

42



## 外れ値の確認方法 2

チュートリアル実施

```
install.packages("ggplot2") #ggplot2パッケージをダウンロード
install.packages("ggrepel") #ggrepelパッケージをダウンロード
library(ggplot2) #ggplot2パッケージを読み込む
library(ggrepel) #ggrepelパッケージを読み込む
表の名称 <- ggplot(Rdata1, aes(x = "", y = y軸のデータ)) +
  stat_boxplot(geom = "errorbar", width = 0.3) +
  geom_boxplot(outlier.colour = "red", outlier.size = 2) +
  labs(x = "ラベルの名称") + geom_text(aes(label = ID),
  hjust = -0.3) + ggrepel::geom_label_repel
(aes(label=ID), box.padding=0.2, point.padding=0.1,
  segment.color='grey50')
```

表の名称

- ggplot2パッケージを使用することで、Boxplotを作成できる。
- geom\_text 関数 及び , ggrepel パッケージの ggrepel::geom\_label\_repel関数を使用することで外れ値のIDを図中に表記できる。

41

45

## Ggplotに関する書籍のご紹介



42

46



47

## 参考文献・サイト

- 1 浅野正彦・中村公亮（2018）. はじめてのRstudio：エラーメッセージなんかこわくないオーム社.
- 2 ggplot2 パッケージによる可視化の際のラベルの重なりを防ぐ Retrieved from <https://id.fnshr.info/2017/03/19/ggplot/> (2020年1月15日)
- 3 松村優哉・湯谷啓明・紀ノ定保礼・前田和寛（2018）. RユーザーのためのRstudio [実践] 入門：tidyverseによるモダンな分析フローの世界 技術評論社.
- 4 小杉考司（2019）. Rでらくらく心理統計：Rstudio徹底活用 講談社.
- 5 R Markdown 入門 Retrieved from [https://kazutan.github.io/kazutanR/Rmd\\_intro.html](https://kazutan.github.io/kazutanR/Rmd_intro.html) (2020年1月10日)
- 6 R 欠損値の対応（missing value treatment） Retrieved from <http://jojoshin.hatenablog.com/entry/2017/02/03/220118> (2020年1月10日)

43

48

## 使用したパッケージ

- 1 dplyr package. Retrieved from <https://cran.r-project.org/web/packages/dplyr/index.html> (2020年1月16日)
- 2 DT package. Retrieved from <https://cran.r-project.org/web/packages/DT/index.html> (2020年1月16日)
- 3 ggrepel package. Retrieved from <https://cran.r-project.org/web/packages/ggrepel/index.html> (2020年1月16日)
- 4 ggplot2 package. Retrieved from <https://cran.r-project.org/web/packages/ggplot2/index.html> (2020年1月16日)
- 5 mice package. Retrieved from <https://cran.r-project.org/web/packages/mice/index.html> (2020年1月16日)
- 6 summarytools package. Retrieved from <https://cran.r-project.org/web/packages/summarytools/index.html> (2020年1月16日)
- 7 VIM package. Retrieved from <https://cran.r-project.org/web/packages/VIM/index.html> (2020年1月16日)

44

49

## 本日のまとめ

Point

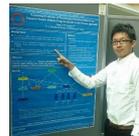
1 R Markdownの使用方法を学んだ

Point

2 臨床情報QCのプログラムを学んだ

自身の研究で実践しプログラムのアレンジを！

本発表に関して質問などございましたら、上野雄己 (yukitf0111@gmail.com) までご連絡ください。



45

50