

Липецкий государственный технический университет

Кафедра прикладной математики

КОМПЬЮТЕРНЫЕ ТЕХНОЛОГИИ МАТЕМАТИЧЕСКИХ ИССЛЕДОВАНИЙ

Лекция 6

5. Методы работы с пропущенными данными

Составитель - Сысоев А.С., к.т.н., доц.

Липецк – 2021

Outline

- 6.0. Исходные данные
- 6.1. Этапы работы с пропущенными данными
- 6.2. Обнаружение пропущенных значений
- 6.3. Исследование структуры пропущенных данных
- 6.4. Выявление источников пропущенных данных и эффекта от них
- 6.5. Анализ полных строк (построчное удаление)
- 6.6. Метод множественного восстановления пропущенных данных
- 6.7. Попарное удаление

6.0. Исходные данные

Пакеты VIM и mice (`install.packages(c("VIM", "mice"))`)

Набор данных sleep (VIM)

Работа по взаимосвязям между сном, экологией и морфологией 62 видов млекопитающих. Параметры сна служили зависимыми переменными, а экологические и морфологические характеристики были независимыми переменными.

Параметры сна: продолжительность сна со сновидениями (Dream) и без сновидений (NonD), а также их сумма (Sleep). Морфологические характеристики: вес тела в килограммах (BodyWgt), вес мозга в граммах (BrainWgt), продолжительность жизни в годах (Span) и продолжительность беременности в днях (Gest). Экологические характеристики – пресс хищников (Pred), степень уязвимости во время сна (Exp) и общая степень опасности, которой подвергается животное (Danger). Экологические характеристики оценивались по пятибалльной шкале, принимавшей значения от 1 (низкий) до 5 (высокий).

6.1. Этапы работы с пропущенными данными

Классическая литература - Little & Rubin (2002).

1. Обнаружить пропущенные данные.
2. Выявить причины их наличия.
3. Удалить наблюдения с пропущенными значениями или заменить пропущенные данные подходящими расчетными значениями.

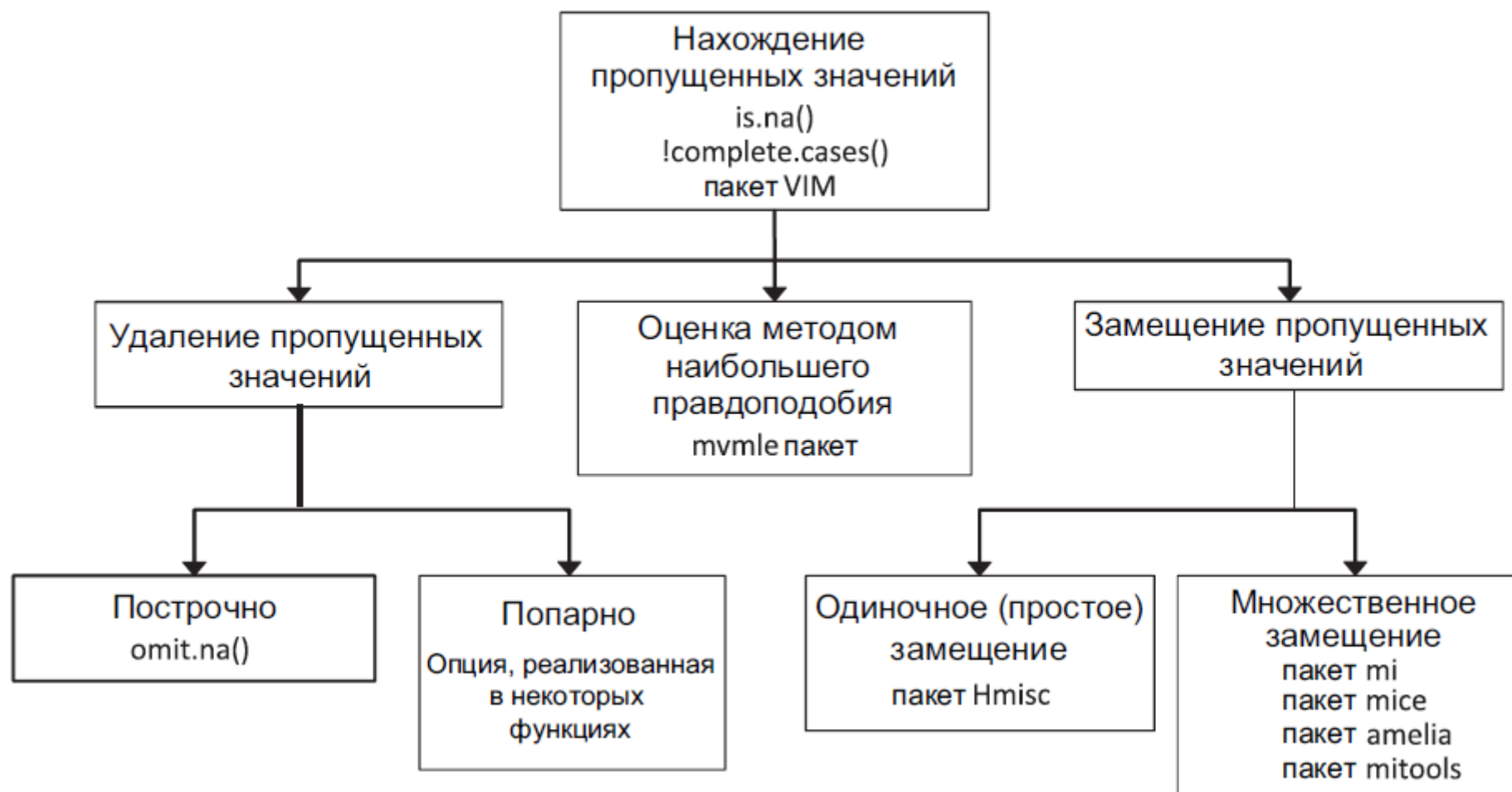
КЛАССИФИКАЦИЯ ТИПОВ ПРОПУЩЕННЫХ ДАННЫХ

Полностью случайный пропуск. Если наличие пропущенных значений в переменной не зависит от значений любой другой наблюдаемой или ненаблюдаемой переменной, тогда данные являются отсутствующими полностью случайно.

Случайный пропуск. Если наличие пропущенных значений в переменной зависит от других переменных, но не от самих неотмеченных значений, то данные являются отсутствующими случайно.

Неслучайный пропуск. Под эту категорию попадают пропущенные значения, которые не относятся к первым двум категориям.

6.2. Обнаружение пропущенных значений



6.2. Обнаружение пропущенных значений

x	is.na()	is.nan()	is.infinite()
x <- NA	TRUE	FALSE	FALSE
x <- 0 / 0	TRUE	TRUE	FALSE
x <- 1 / 0	FALSE	FALSE	TRUE

Функцию `complete.cases()` можно использовать для обнаружения строк в матрице или таблице данных, которые не содержат пропущенных значений. Эта функция возвращает логический вектор со значениями TRUE для всех полных строк и FALSE – для строк с одним и более пропущенными значениями.

```
sleep[complete.cases(sleep),]; sleep[!complete.cases(sleep),]  
> sum(is.na(sleep$Dream))  
[1] 12  
> mean(is.na(sleep$Dream))  
[1] 0.19  
> mean(!complete.cases(sleep))  
[1] 0.32
```

6.3. Исследование структуры пропущенных данных

ПРЕДСТАВЛЕНИЕ ПРОПУЩЕННЫХ ЗНАЧЕНИЙ В ВИДЕ ТАБЛИЦЫ

Функция `md.pattern()` из пакета `mice` представляет информацию о пропущенных значениях в табличной форме.

```
> library(mice)
> data(sleep, package="VIM")
> md.pattern(sleep)
```

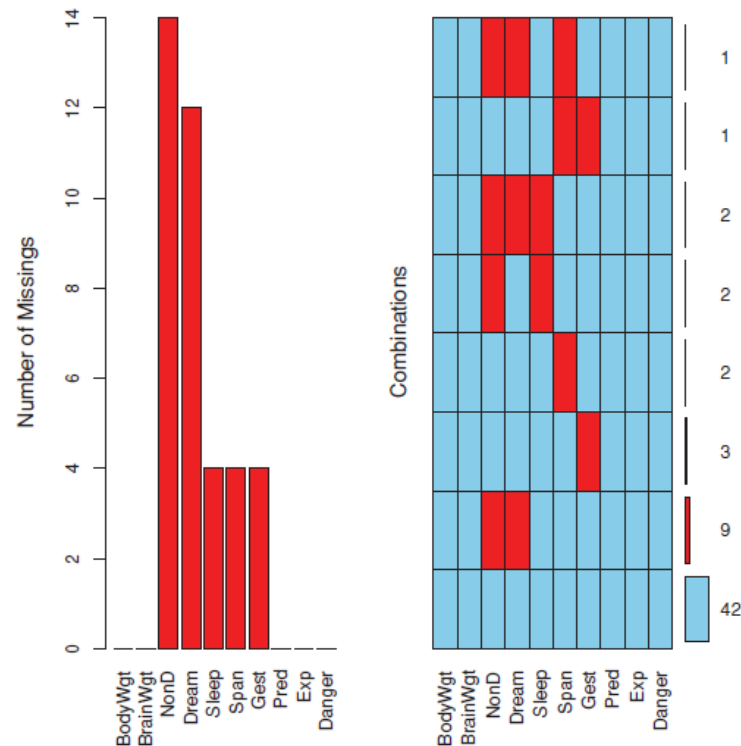
	BodyWgt	BrainWgt	Pred	Exp	Danger	Sleep	Span	Gest	Dream	NonD	
42	1	1	1	1	1	1	1	1	1	1	0
2	1	1	1	1	1	1	0	1	1	1	1
3	1	1	1	1	1	1	1	0	1	1	1
9	1	1	1	1	1	1	1	1	0	0	2
2	1	1	1	1	1	0	1	1	1	0	2
1	1	1	1	1	1	1	0	0	1	1	2
2	1	1	1	1	1	0	1	1	0	0	3
1	1	1	1	1	1	1	0	1	0	0	3
	0	0	0	0	0	4	4	4	12	14	38

6.3. Исследование структуры пропущенных данных

ВИЗУАЛЬНОЕ ИССЛЕДОВАНИЕ СТРУКТУРЫ ПРОПУЩЕННЫХ ДАННЫХ

Функция `aggr()` графически отображает число наблюдений для каждой отдельной переменной и для каждой комбинации переменных.

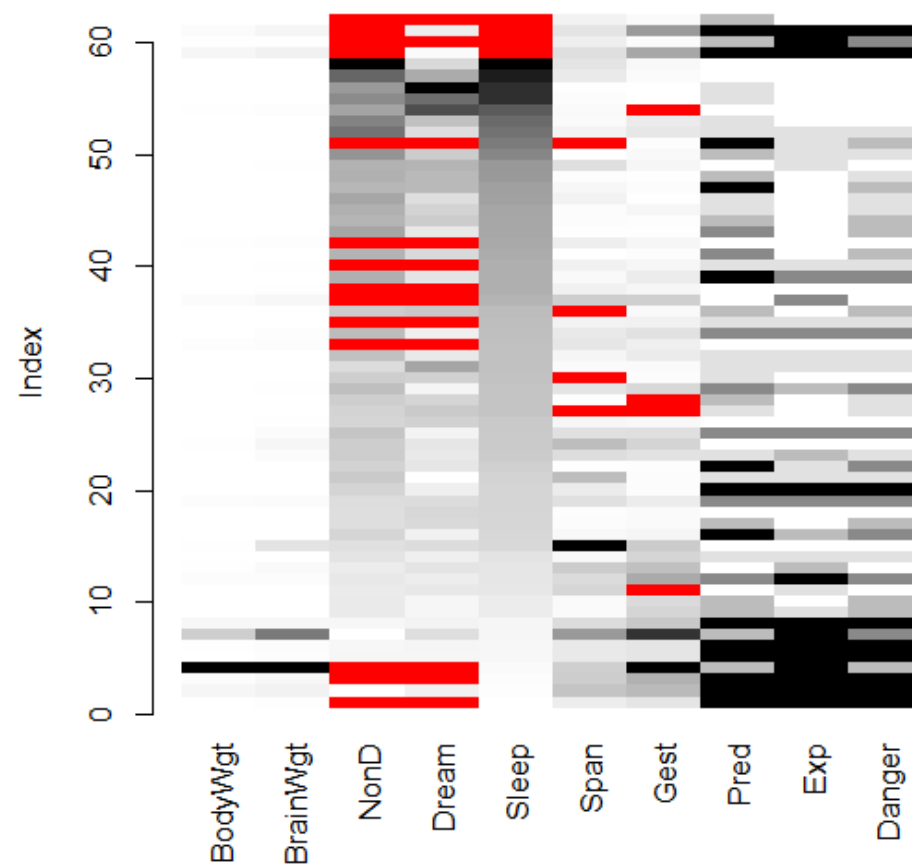
```
library("VIM")  
aggr(sleep, prop=FALSE, numbers=TRUE)
```



6.3. Исследование структуры пропущенных данных

ВИЗУАЛЬНОЕ ИССЛЕДОВАНИЕ СТРУКТУРЫ ПРОПУЩЕННЫХ ДАННЫХ

Функция `matrixplot()` графически отображает данные по каждой строке.

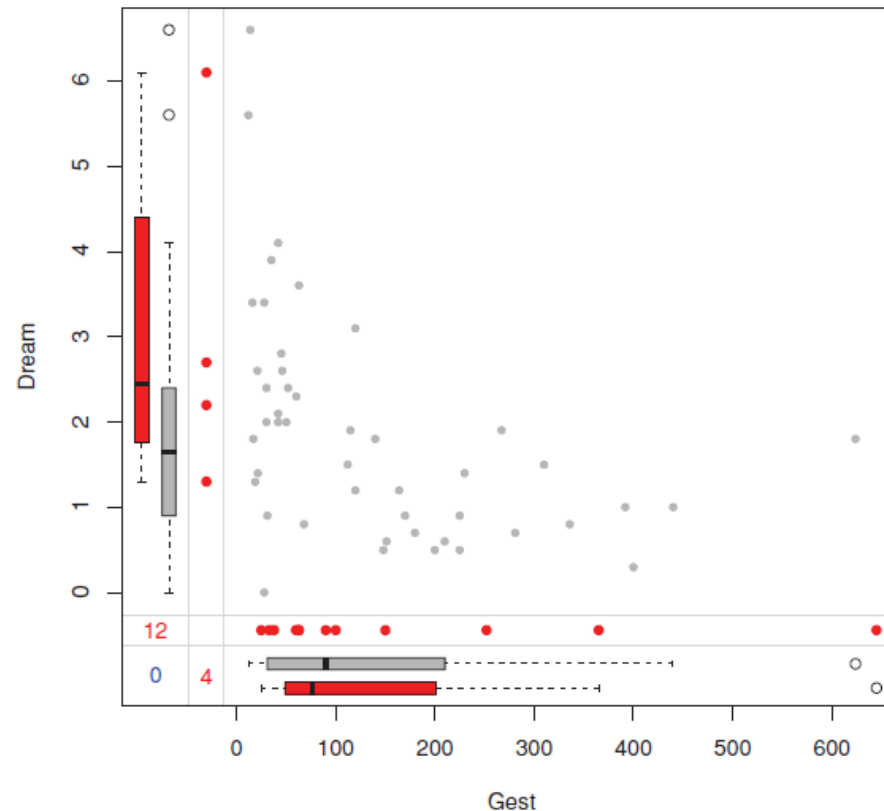


6.3. Исследование структуры пропущенных данных

ВИЗУАЛЬНОЕ ИССЛЕДОВАНИЕ СТРУКТУРЫ ПРОПУЩЕННЫХ ДАННЫХ

Функция `marginplot()` позволяет получить диаграмму рассеяния для двух переменных, где информация о пропущенных значениях представлена на полях.

```
marginplot(sleep[c("Gest", "Dream")], pch=c(20),  
           col=c("darkgray", "red", "blue"))
```



6.3. Исследование структуры пропущенных данных

ИСПОЛЬЗОВАНИЕ КОРРЕЛЯЦИИ ДЛЯ ИССЛЕДОВАНИЯ ПРОПУЩЕННЫХ ЗНАЧЕНИЙ

Можно заменить данные условными значениями: 1 – обозначает пропущенное значение, 0 – имеющееся. Полученную таблицу называют **матрицей теней (shadow matrix)**. Вычисление корреляций между этими преобразованными переменными и между ними и исходными переменными поможет узнать, значения каких переменных имеют тенденцию отсутствовать согласованно, а также выявить связи между отсутствием значений в одной переменной и значениями других переменных.

```
x <- as.data.frame(abs(is.na(sleep)))
> head(sleep, n=5)
```

	BodyWgt	BrainWgt	NonD	Dream	Sleep	Span	Gest	Pred	Exp	Danger
1	6654.000	5712.0	NA	NA	3.3	38.6	645	3	5	3
2	1.000	6.6	6.3	2.0	8.3	4.5	42	3	1	3
3	3.385	44.5	NA	NA	12.5	14.0	60	1	1	1
4	0.920	5.7	NA	NA	16.5	NA	25	5	2	3
5	2547.000	4603.0	2.1	1.8	3.9	69.0	624	3	5	4

```
> head(x, n=5)
```

	BodyWgt	BrainWgt	NonD	Dream	Sleep	Span	Gest	Pred	Exp	Danger
1	0	0	1	1	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	0	1	1	0	0	0	0	0	0
4	0	0	1	1	0	1	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0

```
y <- x[which(sd(x) > 0)]
cor(y)
```

	NonD	Dream	Sleep	Span	Gest
NonD	1.000	0.907	0.486	0.015	-0.142
Dream	0.907	1.000	0.204	0.038	-0.129
Sleep	0.486	0.204	1.000	-0.069	-0.069
Span	0.015	0.038	-0.069	1.000	0.198
Gest	-0.142	-0.129	-0.069	0.198	1.000

6.3. Исследование структуры пропущенных данных

ИСПОЛЬЗОВАНИЕ КОРРЕЛЯЦИИ ДЛЯ ИССЛЕДОВАНИЯ ПРОПУЩЕННЫХ ЗНАЧЕНИЙ

```
> cor(sleep, y, use="pairwise.complete.obs")
      NonD   Dream   Sleep   Span   Gest
BodyWgt  0.227  0.223  0.0017 -0.058 -0.054
BrainWgt  0.179  0.163  0.0079 -0.079 -0.073
NonD      NA     NA     NA    -0.043 -0.046
Dream    -0.189    NA  -0.1890  0.117  0.228
Sleep    -0.080 -0.080     NA   0.096  0.040
Span      0.083  0.060  0.0052    NA -0.065
Gest      0.202  0.051  0.1597 -0.175    NA
Pred      0.048 -0.068  0.2025  0.023 -0.201
Exp       0.245  0.127  0.2608 -0.193 -0.193
Danger    0.065 -0.067  0.2089 -0.067 -0.204
Warning message:
In cor(sleep, y, use = "pairwise.complete.obs")
  the standard deviation is zero
```

6.4. Выявление источников пропущенных данных и эффекта от них

- Какая доля данных пропущена?
- Сосредоточены ли пропущенные данные в нескольких переменных или они широко распределены по всему набору данных?
- Можно ли их считать случайными?
- Позволяет ли ковариация пропущенных данных друг с другом или с наблюдаемыми данными обнаружить возможный механизм, лежащий в основе пропущенных значений?

Примеры: 1) руководители
2) пример с полом.

6.5. Анализ полных строк (построчное удаление)

Функцию `complete.cases()` можно использовать для извлечения полных строк матрицы или таблицы данных.

```
newdata <- mydata[complete.cases(mydata),]
```

Этого же результата можно добиться при помощи функции `na.omit`

```
newdata <- na.omit(mydata)
```

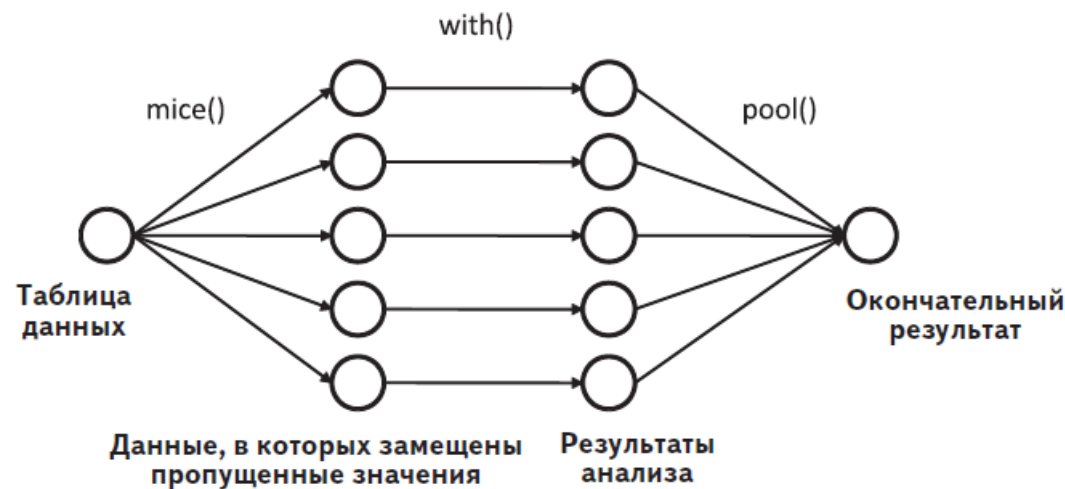
```
> options(digits=1)
```

```
> cor(na.omit(sleep))
```

	BodyWgt	BrainWgt	NonD	Dream	Sleep	Span	Gest	Pred	Exp	Danger
BodyWgt	1.00	0.96	-0.4	-0.07	-0.3	0.47	0.71	0.10	0.4	0.26
BrainWgt	0.96	1.00	-0.4	-0.07	-0.3	0.63	0.73	-0.02	0.3	0.15
NonD	-0.39	-0.39	1.0	0.52	1.0	-0.37	-0.61	-0.35	-0.6	-0.53
Dream	-0.07	-0.07	0.5	1.00	0.7	-0.27	-0.41	-0.40	-0.5	-0.57
Sleep	-0.34	-0.34	1.0	0.72	1.0	-0.38	-0.61	-0.40	-0.6	-0.60
Span	0.47	0.63	-0.4	-0.27	-0.4	1.00	0.65	-0.17	0.3	0.01
Gest	0.71	0.73	-0.6	-0.41	-0.6	0.65	1.00	0.09	0.6	0.31
Pred	0.10	-0.02	-0.4	-0.40	-0.4	-0.17	0.09	1.00	0.6	0.93
Exp	0.41	0.32	-0.6	-0.50	-0.6	0.32	0.57	0.63	1.0	0.79
Danger	0.26	0.15	-0.5	-0.57	-0.6	0.01	0.31	0.93	0.8	1.00

6.6. Метод множественного восстановления пропущенных данных

Метод множественного восстановления пропущенных данных (multiple imputation, MI) – это способ заполнения пропусков при помощи повторного моделирования.



Функция `mice()` использует исходную таблицу данных с пропущенными значениями, а возвращает объект, содержащий несколько полных наборов данных (пять по умолчанию). Каждый такой полный набор данных получается при восстановлении пропущенных данных исходной таблицы. В алгоритме восстановления данных есть случайная составляющая, поэтому все производные полные наборы данных немного отличаются друг от друга. Затем при помощи функции `with()` применяется статистическая модель (например, линейная или обобщенная линейная). Функция `pool()` объединяет результаты, полученные для отдельных производных наборов данных.

6.6. Метод множественного восстановления пропущенных данных

```
library(mice)
imp <- mice(mydata, m)
fit <- with(imp, analysis)
pooled <- pool(fit)
summary(pooled),
```

- **mydata** – это матрица или таблица данных с пропущенными значениями;
- **imp** – список, содержащий **m** наборов данных с восстановленными пропущенными значениями вместе с информацией о том, как это восстановление было проведено. По умолчанию **m** = 5;
- **analysis** – формула, определяющая тип статистического метода, который должен быть применен к каждому из **m** восстановленных наборов данных. К таким методам относятся **lm()** – линейная регрессия, **glm()** – обобщенная линейная регрессия, **gam()** – обобщенные аддитивные модели и **nbrm()** – отрицательные биномиальные модели. В формулах внутри скобок зависимая переменная указывается слева от знака ~, а независимые (разделенные знаком +) – справа;
- **fit** – список, содержащий результаты **m** отдельных статистических анализов;
- **pooled** – список, содержащий усредненные результаты этих **m** отдельных статистических анализов.

6.6. Метод множественного восстановления пропущенных данных

```
> library(mice)
> data(sleep, package="VIM")
> imp <- mice(sleep, seed=1234)
[...выводимая информация удалена для экономии места...]
> fit <- with(imp, lm(Dream ~ Span + Gest))
> pooled <- pool(fit)
> summary(pooled)
```

	est	se	t	df	Pr(> t)	lo 95
(Intercept)	2.58858	0.27552	9.395	52.1	8.34e-13	2.03576
Span	-0.00276	0.01295	-0.213	52.9	8.32e-01	-0.02874
Gest	-0.00421	0.00157	-2.671	55.6	9.91e-03	-0.00736

	hi 95	nmis	fmi
(Intercept)	3.14141	NA	0.0870
Span	0.02322	4	0.0806
Gest	-0.00105	4	0.0537

6.7. Попарное удаление

Попарное удаление (pairwise deletion) при работе с неполными наборами данных обычно рассматривается как альтернатива построчному удалению. При попарном удалении наблюдения удаляются только в том случае, если это пропущенные значения в переменных, которые используются в конкретном анализе данных.

```
> cor(sleep, use="pairwise.complete.obs")
```

	BodyWgt	BrainWgt	NonD	Dream	Sleep	Span	Gest	Pred	Exp	Danger
BodyWgt	1.00	0.93	-0.4	-0.1	-0.3	0.30	0.7	0.06	0.3	0.13
BrainWgt	0.93	1.00	-0.4	-0.1	-0.4	0.51	0.7	0.03	0.4	0.15
NonD	-0.38	-0.37	1.0	0.5	1.0	-0.38	-0.6	-0.32	-0.5	-0.48
Dream	-0.11	-0.11	0.5	1.0	0.7	-0.30	-0.5	-0.45	-0.5	-0.58
Sleep	-0.31	-0.36	1.0	0.7	1.0	-0.41	-0.6	-0.40	-0.6	-0.59
Span	0.30	0.51	-0.4	-0.3	-0.4	1.00	0.6	-0.10	0.4	0.06
Gest	0.65	0.75	-0.6	-0.5	-0.6	0.61	1.0	0.20	0.6	0.38
Pred	0.06	0.03	-0.3	-0.4	-0.4	-0.10	0.2	1.00	0.6	0.92
Exp	0.34	0.37	-0.5	-0.5	-0.6	0.36	0.6	0.62	1.0	0.79
Danger	0.13	0.15	-0.5	-0.6	-0.6	0.06	0.4	0.92	0.8	1.00

Список литературы

Кабаков Р. К. (2014) R в действии. Анализ и визуализация данных на языке R Издательство: ДМК Пресс, 580 с.

Мастичкий С. Э., Шитиков В. К. (2014) Статистический анализ и визуализация данных с помощью R. - Электронная книга, 400 с