Імпорт даних : : шпаргалка

У R tidyverse побудована навколо охайних даних, що зберігаються в tibble (розвиток data frame).



На лицьовій сторінці описано. як зчитувати текстові файли в R за допомогою readr.



На зворотній сторінці описано, як створювати tibble з tibble i охайні дані з tidyr.

ІНШІ ТИПИ ДАНИХ

Використовуйте наступні пакети для імпорту файлів іншого типу

- haven файли SPSS. Stata та SAS
- readxl excel файли (.xls та .xlsx)
- **DBI** бази даних
- isonlite ison
- xml2 XML
- httr Web API
- rvest HTML (дані з Інтернету)

Збереження даних

Зберігайте \mathbf{x} , об'єкт \mathbf{R} , у \mathbf{path} , шлях до файлу, за допомогою:

Файл, розділений комами

write_csv(x, path, na = "NA", append = FALSE, col names = !append)

Файл із довільним роздільником

write delim(x, path, delim = " ", na = "NA", append = FALSE, col names = !append)

CSV для excel

write_excel_csv(x, path, na = "NA", append = FALSE, col names = !append)

Рядок у файл

write_file(x, path, append = FALSE)

Рядковий вектор у файл, один елемент на рядок

write lines(x, path, na = "NA", append = FALSE)

Об'єкт у файл RDS

write rds(x, path, compress = c("none", "gz", "bz2", "xz"), ...)

Файл, розділений табуляціями

write_tsv(x, path, na = "NA", append = FALSE, col_names = !append)

Зчитування табличних даних

Ці функції мають спільні аргументи:

read_*(file, col_names = TRUE, col_types = NULL, locale = default_locale(), na = c("", "NA"), quoted_na = TRUE, comment = "", trim_ws = TRUE, skip = 0, n_max = Inf, guess_max = min(1000, n_max), progress = interactive())



КОРИСНІ АРГУМЕНТИ



Зчитування нетабличних даних

Зчитування файлу в один рядок

read file(file, locale = default locale())

Зчитування рядків файлу в окремі рядки

read lines(file, skip = 0, n max = -1L, na = character(), locale = default_locale(), progress = interactive())

Зчитування логових файлів Apache

read_log(file, col_names = FALSE, col_types = NULL, skip = 0, n_max = -1, progress = interactive()) x\$A <- parse number(x\$A)

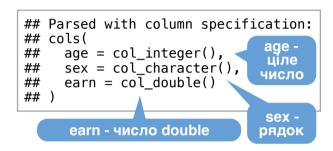
Зчитування файлу в raw вектор read file raw(file)

Зчитування рядків файлу в raw вектори

read_lines_raw(file, skip = 0, n_max = -1L, progress = interactive())

Типи даних





1. Використовуйте problems() для діагностики проблем

x <- read csv("file.csv"); problems(x)</pre>

- 2. Викорис. col_ функцію для розбору
 - col guess() за замовчуванням
 - col character()
 - col_double(), col_euro_double()
- col datetime(format = "") Також col date(format = "").
- col_time(format = "")
- col factor(levels, ordered = FALSE)
- · col_integer()
- · col_logical()
- col_number(), col_numeric()
- · col skip()
- x <- read csv("file.csv", col types = cols(A = col double(), $B = col_logical(),$ C = col factor()))
- 3. По-іншому: зчитуйте як рядкові вектори і розбирайте з parse_ функцією
- parse quess()
- parse character()
- parse_datetime() Також parse_date() та parse time()
- parse double()
- parse factor()
- parse integer()
- parse logical()
- parse number()

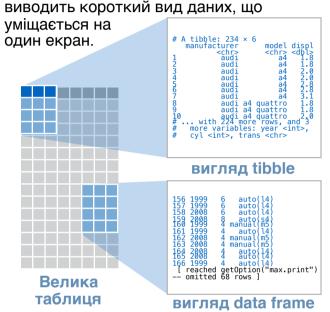
Tibble - розвиток data frame

Пакет tibble вводить новий S3 клас для зберігання табличних даних, tibble. Tibble успадковує клас data frame, але покращує три напрямки:



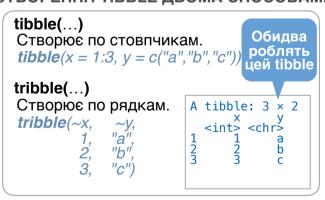
- Вибір елементів [завжди повертає новий tibble, [[та \$ - вектор.
- Немає часткового узгодження Ви повинні використовувати повні імена стовпчиків при виборі елементів.

• Відображення - При друку tibble, R виводить короткий вид даних, що



- Контроль виглядом за замовч. опціями: **options(**tibble.print max = n, tibble.print min = m, tibble.width = lnf)
- Перегляд повних даних: View() або glimpse()
- Повернення до data frame: as.data.frame()

СТВОРЕННЯ TIBBLE ДВОМА СПОСОБАМИ

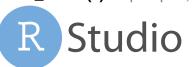


as_tibble(x, ...)

Конвертує data frame у tibble.

enframe(x, name = "name", value = "value") Конвертує іменований вектор в tibble

is tibble(x) Перевіряє, чи є x tibble.



Охайні дані з tidyr

Охайні дані - це спосіб організації табличних даних. Це встановлює структуру даних. узгоджену між пакетами.

Таблиця охайна, якщо:

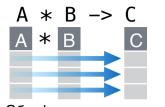




спостереження рядок

В охайних даних:





Зберіг. спостереження при векторизованих операціях

Форматування даних - зміна розташування елементів в таблиці

Використовуйте gather() та spread() для реорганізації елементів таблиці.

convert = FALSE, factor key = FALSE)

Переміщує імена стовпчиків в стовпчик кеу, збираючи значення стовпчиків в спільний стовпчик value.

table4a A 0.7K 2K 37K 80K C 212K 213K 1999 212K 2K 2000 80K key value

gather(table4a, `1999`, `2000`, kev = "vear", value = "cases")

gather(data, key, value, ..., na.rm = FALSE, spread(data, key, value, fill = NA, convert = FALSE, drop = TRUE, sep = NULL)

> Переміщує унікальні значення стовпчика кеу в імена стовпчиків, розповсюджуючи значення стовпчика **value** по нових стовпчиках.

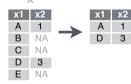
> > table2

country	year	type	count
Α	1999	cases	0.7K
Α	1999	pop	19M
Α	2000	cases	2K
Α	2000	рор	20M
В	1999	cases	37K
В	1999	pop	172M
В	2000	cases	80K
В	2000	pop	174M
С	1999	cases	212K
С	1999	рор	1T
С	2000	cases	213K
С	2000	рор	1T
		kov	value

spread(table2, type, count)

Робота с відсутніми значеннями

drop_na(data, ...) Прибиряє рядки з NA у стовпчиках з ...



 $drop_na(x, x2)$

C 1 E 3

Замінює NA у стовпчиках з ...

крайніми не-NA значеннями.

fill(x, x2)

replace na(data, **fill(**data, ..., .direction = c("down", "up"))

> replace = list(), ...) Замінює NA по стовпчиках.

2000

1999

2000

C 2000 213K

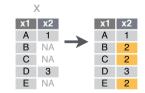
1999 212K

37K

80K

174M

1T



 $replace_na(x, list(x2 = 2))$

Розширення таблиць - швидке створення таблиць з комб-ми значеннь

complete(data, ..., fill = list())

Додає до даних відсутні комбінації значень змінних з ... complete(mtcars, cyl, gear, carb)

expand(data, ...)

Створює новий tibble з усіма комбінаціями значень змінних з ...

expand(mtcars, cyl, gear, carb)

Поділ клітинок

Функції для поділу або комбінування клітинок в окремі значення.



separate(data, col, into, sep = "[^[:alnum:]] +", remove = TRUE, convert = FALSE, extra = "warn". fill = "warn". ...)

Поділяє клітинки стовпчика по окремих стовпчиках.

table3

country	year	rate		country	year	cases	рор
Α	1999	0.7K / 19M		Α	1999	0.7K	19M
Α	2000	2K/20M	\rightarrow	Α	2000	2K	20M
В	1999	37K/172M		В	1999	37K	172
В	2000	80K/174M		В	2000	80K	174
С	1999	212K / 1T		С	1999	212K	1T
С	2000	213K/1T		С	2000	213K	1T

separate(table3, rate, into = c("cases", "pop"))

separate rows(data, ...,

sep = "[^[:alnum:].]+", convert = FALSE)

Поділяє клітинки стовпчика по окремих рядках. Також separate_rows_().

table3

country	year	rate
Α	1999	0.7K / 19M
Α	2000	2K/20M
В	1999	37K/172M
В	2000	80K/174M
С	1999	212K / 1T
С	2000	213K / 1T

Α	1999	0.7K
Α	1999	19M
Α	2000	2K
Α	2000	20M
В	1999	37K
В	1999	172M
В	2000	80K
В	2000	174M
С	1999	212K
С	1999	1T
С	2000	213K
С	2000	1T

country year rate

separate_rows(table3, rate)

.....

unite(data, col, ..., sep = " ", remove = TRUE)

Об'єднує клітинки кількох стовпчиків в один стовпчик.

table5

country	century	year		country	year
Afghan	19	99		Afghan	1999
Afghan	20	0	—	Afghan	2000
Brazil	19	99		Brazil	1999
Brazil	20	0		Brazil	2000
China	19	99		China	1999
China	20	0		China	2000

unite(table5, century, year, col = "year", sep = ""