создание

манипуляции

VOTIMIACTRO CHAROTI

регистр букв

подстроки

Junicina

векторы стро

- - r - r -

**

....

-

подмножест

регуляркі

....

квантификация

Работа со строками в R

Г. Мороз

соэлание

чтени

манипуляц

соединение

регистр букв

подстроки

замена разбиени

пересечение

разност

подмножес

регуляркі

наборы символов квантификация

Как получить строку?

- o a <- "the quick brown fox jumps over the lazy dog"
- $\circ~$ a <- 'the quick brown fox jumps over the lazy dog'
- o a <- "the quick 'brown' fox jumps over the lazy dog"
- o a <- 'the quick "brown" fox jumps over the lazy dog'
- o b <- ""
- o c <- character(3)
 - [1] "" "" ""
- o d <- as.character(3)
 - [1] "3"
- d <- as.character(c(4:9))
- [1] "4" "5" "6" "7" "8" "9"
- o letters[3:8], LETTERS[3:8]
- data.frame(letters[3:8], letters[8:3], stringsAsFactors = F)

создает пустые строки

превращает в строку

вектор в строку

Чтение файлов?

o x <- read.csv2("c.csv")</pre>

o y <- read.csv("b.txt")</pre>

o w <- read.delim("d.txt")</pre>

o z <- read.table("a.csv", sep = "...")</pre>

v <- readLines("e.txt")

текст и разделители

разделитель ","

разделитель ";"

разделитель "\t"

чтение "голого" текста

создание

............

соединение

регистр букт

подстроки

замена

пересечение

разность сравнение

сравнение подмножест

регулярк

наборы символов квантификация

Соединение строк: paste()

```
paste("Школа", "Лингвистики")[1] "Школа Лингвистики"
```

- paste(9, 7, 8, sep = "-")# другой разделитель[1] "9-7-8"
- paste(LETTERS[1:5], letters[1:5], sep = "")# векторы[1] "Aa" "Вb" "Сс" "Dd" "Ее"
- paste("F", letters[1:5], sep = "")# векторы разной длины[1] "Fa" "Fb" "Fc" "Fd" "Fe"
- paste("F", letters[1:5], sep = "", collapse = " ")# в один вектор[1] "Fa Fb Fc Fd Fe"
- paste("F", letters[1:5], sep = "", collapse = "-") # другой разделитель[1] "Fa-Fb-Fc-Fd-Fe"
- paste(c("F", letters[1:5]), collapse = "-")# развекторизовать[1] "F-a-b-c-d-e"

создание

соединение

количество

регистр оукв подстроки

поиск замена

разбиение

объединение

пересеч

сравнение

регулярк

наборы символов

Соединение строк: library(stringr)

```
str_c("Школа", "Лингвистики")[1] "ШколаЛингвистики"
```

- \circ str_c("Школа", "Лингвистики", sep = " ") # разделитель [1] "Школа Лингвистики"
- str_c(LETTERS[1:5], letters[1:5], sep = "")# векторы[1] "Aa" "Вb" "Сс" "Dd" "Ее"
 - str_c("F", letters[1:5], sep = "")# векторы разной длины[1] "Fa" "Fb" "Fc" "Fd" "Fe"
- str_c("F", letters[1:5], sep = "", collapse = " ")# в один вектор[1] "Fa Fb Fc Fd Fe"
- str_c("F", letters[1:5], sep = "", collapse = "-") # другой разделитель[1] "Fa-Fb-Fc-Fd-Fe"
- str_c(c("F", letters[1:5]), collapse = "-")# развекторизовать[1] "F-a-b-c-d-e"

Cоединение строк: paste() vs. str_c()

```
создание
```

чтение

манипуляц

соединение

регистр був

подстроки

замена

векторы стро

- F - F

пересечени

разность

сравнение

регуляркі

наборы символ

наооры символо: квантификация

```
paste нативная функция 
paste paste("1", "3") 
[1] "1 3"
```

Количество символов

количество символов

```
a <- "the quick brown fox jumps over the lazy dog"
```

o nchar(a)

[1] 43

library(stringr); str_length(a)

[1] 43

b <- c("the", "quick", "brown", "fox")

o nchar(b)

[1] 3 5 5 3

str_length(b)

[1] 3 5 5 3

c <- c("the", "quick", NULL, "brown", "", "fox")

o nchar(c)

[1] 3 5 5 0 3

str_length(c)

[1] 3 5 5 0 3

Изменение регистра

a <- "the QuiCk bRown fOx juMps ovEr tHe laZy dOg"

- tolower(a)
 - [1] "the quick brown fox jumps over the lazy dog"
- toupper(a)
 - [1] "THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG"

b <- "любЯ, сЪешЬ щипЦы, — взДохнёт мэр, — кайф Жгуч"

- o tolower(b)
 - [1] "любя, съешь щипцы, вздохнёт мэр, кайф жгуч"
- toupper(b)
 - [1] "ЛЮБЯ, СЪЕШЬ ЩИПЦЫ, ВЗДОХНЁТ МЭР, КАЙФ ЖГУЧ

чтение

соединение

регистр букв

понск замена

разбиение

векторы стр

объединение

разность

подмножест

регуляркі метасимволь

наборы символо квантификация

презентация доступна: https://goo.gl/QRCtyw

Выделение подстроки: substr()

создание

чтение

соединение

количество симво.

подстроки

поиск

разбиен

векторы стро

объединение

пересеч

сравнени

подмноже

регулярк

наборы символ

```
a <- "the quick brown fox"
b <- c("the", "quick", "brown", "fox")
```

substr(a, 2, 3)[1] "he"

второй и третий символ

substr(b, 2, 3)[1] "he" "ui" "ro" "ox"

второй и третий символ

substr(b, 2, 2) <- "."; b # заменяет второй символ точкой[1] "t.e" "q.ick" "b.own" "f.x"

substr(b, 2, 2) <- c("1", "2"); b # заменяет второй символ вектором[1] "t1e" "q2ick" "b1own" "f2x"

Выделение подстроки: substring()

```
создание
```

чтени

соединение

количество симво. регистр букв

подстроки

поиск

разбиение

векторы стр

объединение

разност

сравнени подмнож

регулярі

метасимволы наборы символов

```
a <- "the quick brown fox"
b <- c("the", "quick", "brown", "fox")
```

- substring(b, 2, 3)# второй и третий символ[1] "he" "ui" "ro" "ox"
- substring(b, 2, 2) <- "."; b # заменяет второй символ точкой[1] "t.e" "q.ick" "b.own" "f.x"
- substring(b, 2, 2) <- c("1", "2"); b # заменяет второй символ вектором
 [1] "t1e" "q2ick" "b1own" "f2x"
- - [1] "t1e" "q22ck" "b333n" "f1x"
- substring(a, 1:19, 1:19)# извлекает каждую букву[1] "t" "h" "e" " "q" "u" "i" "c" "k" " " "b" "r" "o" "w" "n" " "f" "o" "x"

Выделение подстроки: library(stringr)

```
создание
```

чтени

соединение

количество символо

подстроки

поиск

разбиени

векторы стро

объединение

попосонение

разность

сравнение

регулярк

наборы символо

a <- "the quick brown fox" b <- c("the", "quick", "brown", "fox")

- str_sub(b, 2, 3) # второй и третий символ[1] "he" "ui" "ro" "ox"
 - str_sub(b, 2, 2) <- "."; b # заменяет второй символ точкой[1] "t.e" "q.ick" "b.own" "f.x"
 - str_sub(b, 2, 2) <- ".;."; b # заменяет второй символ другими[1] "t.;.e" "q.;.ick" "b.;.own" "f.;.x"
 - \circ str_sub(b, 2) <- c("1", "2") ; b # заменяет второй символ вектором [1] "t1e" "q2ick" "b1own" "f2x"
 - str_sub(a, 1:19, 1:19)# извлекает каждую букву[1] "t" "h" "e" " "q" "u" "i" "c" "k" " "b" "r" "o" "w" "n" " "f" "o" "x"
- o str_sub(b, -1, -1) # последний символ

Выделение подстроки: library(stringr)

a <- c("the quick brown fox jumps over")
b <- c("jumps over the lazy dog")</pre>

- word(a, 2) # второе слово[1] "quick"
- word(c(b, a), 4)# четвертое слово каждой строки[1] "lazy" "fox"
 - \circ word(c(b, a), -3) # третье слово с конца каждой строки [1] "the" "fox"
 - word(c(b, a), 2, 4) # со второго по четвертое слово каждой строки
 [1] "over the lazy" "quick brown fox"
- word(c(b, a), -2, -1)
 # последние два слова каждой строки
 [1] "lazy dog" "jumps over"

создание

манипуляции соединение

регистр букв подстроки

поиск

разбиение

объединение

разность

сравнение подмножес

регулярки метасимволы

наборы символо квантификация

```
созлание
```

чтение

манипуляции соединение

количество символо регистр букв

подстроки

замена разбиени

векторы стро

объединение

разность

сравнение подмножес

регуляркі

метасимволы
наборы символо
квантификация

Поиск вектора по фрагменту: grep()

```
\label{eq:alpha} $a <- c("the quick", "brown", "fox", "jumps", "over")$$ $b <- c("jumps", "over", "the", "lazy", "dog")$$
```

- \circ grep("the", c(a, b)) номер элемента, содержащего подстроку [1] 1 8
- grep("the", c(a, b), value = T)элемент, содержащий подстроку[1] "the quick" "the"
- grep("the", c(a, b), invert = T) номер элемента, не содержащего подстроку
 [1] 2 3 4 5 6 7 9 10
- grep("the", c(a, b), invert = T, value = T) элемент, не содержащий подстроку
 [1] "brown" "fox" "jumps" "over" "jumps" "over" "lazy" "dog"
- o grepl("the", c(a, b)) элемент, содержащий подстроку [1] TRUE FALSE FALSE FALSE FALSE FALSE TRUE FALSE FALSE

Поиск вектора по фрагменту: library("stringr")

```
a <- c("the quick", "brown", "fox", "jumps", "over")
b <- c("jumps", "over", "the", "lazy", "dog")</pre>
```

- str_detect("the", c(a, b))
 элемент, содержащий подстроку
 [1] TRUE FALSE FALSE FALSE FALSE FALSE FALSE TRUE FALSE FALSE
- \circ c(a,b)[str_detect("the", c(a, b))] элемент, содержащий подстроку [1] "the" "the"

создание

чтение

манипуляции соединение

количество симво регистр букв

подстроки

пазбиение

векторы стро

векторы стро

объединение

разность

сравнение подмножество

регулярк

наборы симво

квантифика

Замена фрагмента

a <- c("the quick brown fox jumps over")
b <- c("jumps over the lazy dog")

- $\circ \ \, \mathrm{sub}("o","_",c(a,b))$ # одно вхождение [1] "the quick br_wn fox jumps over" "jumps _ver the lazy dog"
- $\circ \ \ gsub("o", "_", c(a,b))$ # все вхождения [1] "the quick br_wn f_x jumps _ver" "jumps _ver the lazy d_g"
- \circ library("stringr"); str_replace("o", "_", c(a, b)) # одно вхождение [1] "the quick br_wn fox jumps over" "jumps _ver the lazy dog"
- library("stringr"); str_replace_all("o", "_", c(a, b)) # все вхождения
 [1] "the quick br_wn f_x jumps _ver" "jumps _ver the lazy d_g"

создание

соединение

количество символог регистр букв подстроки

замена разбиение

векторы строн

разность

регулярки

метасимволы наборы символов квантификация

Разбиение

разбиение

a <- c("the quick brown fox jumps over") b <- c("jumps over the lazy dog")

- strsplit(c(a,b), " ")
- library("stringr"); str_split(c(a,b), " ")
- # возвращает список слов
- # возвращает список слов

Вектор уникальных строк

создание

манипуляции

соединение

регистр букв

подстроки

замена

венторы стро

объединение

пересеч

сравнение

подмножество

регулярк

метасимволь

наооры симво

```
a <- c("the", "quick", "brown", "fox", "jumps", "over")
b <- c("jumps", "over", "the", "lazy", "dog")
o union(a, b)
[1] "the" "quick" "brown" "fox" "jumps" "over" "lazy" "dog"</pre>
```

Пересечение векторов строк

пересечение

```
a <- c("the", "quick", "brown", "fox", "jumps", "over")
b <- c("jumps", "over", "the", "lazy", "dog")
  o intersect(a, b)
     [1] "the" "jumps" "over"
```

Разность векторов строк

```
создание
```

чтение

манипуляции

соединение

регистр бук

подстроки

поиск

разбиени

векторы стро

объединение

пересечение разность

сравнение

подмножество

регулярк

ore recentions.

inioopia casno.

```
\begin{split} &a <- c("the", "quick", "brown", "fox", "jumps", "over")\\ &b <- c("jumps", "over", "the", "lazy", "dog")\\ &\circ setdiff(a,b)\\ &[1] \; "quick" \; "brown" \; "fox" \end{split}
```

Сравнение векторов строк

сравнение

```
a <- c("the", "quick", "brown", "fox", "jumps", "over")
b <- c("jumps", "over", "the", "lazy", "dog")
  o setequal(a, b)
     [1] FALSE
a <- c("the", "quick", "brown", "fox", "jumps", "over")
c <- c("brown", "fox", "jumps", "over", "the", "quick")
  setequal(c, a)
     [1] TRUE
a <- c("the", "quick", "brown", "fox", "jumps", "over")
c <- c("brown", "fox", "jumps", "over", "the", "quick")
  o identical(c, a)
```

[1] FALSE

Является ли вектор подмножеством другого?

```
создание
```

чтение

соединение

регистр букв

подстроки

замена

векторы стро

.

объединение

разность

сравнение

подмножество

регуляркі

наборы символ

квантификация

```
a <- c("the", "quick", "brown", "fox")
b <- "the"
c <- "lazy"
d <- "dog"
```

is.element(b, a)[1] TRUE

is.element(c, a)[1] FALSE

is.element(c(b, c, d), a)[1] TRUE FALSE FALSE

Экранирование метасимволов

В R для экранирования метасимволов используется двойной слэш (а для экранирования слэша аж три слэша):

```
\\.
\\$
\\*
\\+
\\?
///
1111
\\[
\\]
\\{
\\}
\\(
```

\\)

метасимволы

Классы знаков

метасимволы

```
\\d
\\D
\\s
\\S
\\w
١١W
```

цифры # не цифры # пробел # не пробел # слово # не слово

- o gsub("\\d", "\\.", "2016 год") [1] ".... год"
- o gsub("\\D", "\\.", "2016 год") [1] "2016...."
- o gsub("\\s", "\\.", "2016 год") [1] "2016.год"

- o gsub("\\S", "\\.", "2016 год") [1] ".... ..."
- o gsub("\\w", "\\.", "2016 год") [1] ".... ..."
- o gsub("\\W", "\\.", "2016 год") [1] "2016.год"

Наборы символов

наборы символов

o gsub("[o-9]", "\\.", "2016 год") [1] ".... год"

- o gsub("[a-z]", "\\.", "2016 year") [1] "2016"
- gsub("[а-я]", "\\.", "2016 год") [1] "2016 ..."

- o gsub("[A-Z]", "\\.", "2016 yEar") [1] "2016 y.ar"
- o gsub("[А-Я]", "\\.", "2016 гОД") [1] "2016 г.."
- gsub("[a-zA-Z]", "\\.", "yEar") [1] "...."
- o gsub("[^a-я]", "\\.", "2016 год") [1] "....год"

Квантификаторы

а <- с("тара", "коса", "шоссе", "касса", "масса") gsub("cc?", "\\.", a) # ноль или один раз [1] "тара" "ко.а" "шо.е" "ка.а" "ма.а" gsub("cc*", "\\.", a) # ноль и более раз [1] "тара" "ко.а" "шо.е" "ка.а" "ма.а" gsub("cc+", "\\.", a) # один и более раз [1] "тара" "коса" "шо.е" "ка.а" "ма.а" o gsub("a.a", "\\.", a) # любой символ [1] "т." "коса" "шоссе" "касса" "масса" gsub("a.*a", "\\.", a) # любой символ ноль и более раз [1] "т." "коса" "шоссе" "к." "м." o gsub("c{2}", "\\.", a) # два раза [1] "тара" "коса" "шо.е" "ка.а" "ма.а" \circ gsub("c $\{1,\}$ ", "\\.", a) # один и более раз квантификация [1] "тара" "ко.а" "шо.е" "ка.а" "ма.а" \circ gsub("c{1,2}", "\\.", a) # от одного до двух раз [1] "тара" "ко.а" "шо.е" "ка.а" "ма.а" Пробел: $c\{1,2\}$ — правильно, $c\{1,2\}$ — неправильно.

создание

манинулиции

количество символог

регистроука

подстроки

поиск

ทองกับอยน

векторы стро

1 1

оонцинение

-

TO THE HOUSE

егулярки

наборы симво.

вантификация

Спасибо за внимание!

Пишите письма agricolamz@gmail.com