создание

чтепис

манипуляции

NO TIMEOTRO CITATRO TO

регистр букв

подстроки

\_\_\_\_

20MATE

векторы стро

объединение

пепесечение

разность

сравнени

подмножест

регуляркі

....

квантификация

#### Работа со строками в R

Г. Мороз

#### созлание

#### чтени

манипуляци

соединение

регистр букв

подстроки

замена

разбиени

векторы стр

объединение

разност

сравнение подмноже

регуляркі

метасимволы наборы символ

наборы символог квантификация

### Как получить строку?

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

презентация доступна: http://idrv.ms/iPMoiZj

# создает пустые строки

# превращает в строку

# вектор в строку

## Чтение файлов?

создание

TICINC

манипуляции

соединение

регистр букв

подстроки

поиск

разбиени

векторы стро

объединение

разност

сравнение

регулярн

изборы симп

наборы симво

o z <- read.table("a.csv sep = "...")

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

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

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

o v <- readLines("e.txt")</pre>

# текст и разделители

# разделитель ","

# разделитель ";" # разделитель "\t"

# разделитель "\t"

# чтение "голого" текста

создание

соединение

количество

регистр оукв подстроки

поиск

разбиени

векторы стро

объединение

разность

сравнение подмножест

регуляркі

наборы символог 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" "Ee"
- 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

подстроки поиск

замена разбиение

of annual

пересече

сравнение

егуляркі

наборы символог

## 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 бук

подстроки

разбиени

#### векторы стро

#### . . .

пересечени

разность

разность

сравнение

nominani

#### регуляркі

наборы симв

квантификация

```
paste нативная функция
paste paste("1", "3")
        [1] "1 3"
paste paste("f", NULL, "", "g", sep = "-")
        [1] "f---g"
```

```
str_c library(stringr)
str_c str_c("1", "3")
        [1] "13"
str_c str_c("f", NULL, "", "g", sep = "-")
        [1] "f-q"
```

#### 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

манипуляции

соединение количество символ

регистр букв подстроки

замена разбиение

векторы стро

объединение

пересечение разность

сравнение

регулярк

наборы символов

квантификация

# Выделение подстроки: substr()

создание

чтение

соединение

количество символ

подстроки

поиск

замена

векторы стр

1

ооъединение

разност

сравнение

регулярі

метасимволь

наборы символо квантификация

```
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</li>[1] "t.e" "q.ick" "b.own" "f.x"

substr(b, 2, 2) <- c("1", "2"); b # заменяет второй символ вектором</li>[1] "t1e" "g2ick" "b1own" "f2x"

# Выделение подстроки: substring()

```
создание
```

чтени

соединение

регистр букв

подстроки

поиск

разбиени

векторы стр

объединение

разност

сравнени

регулярь

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</li>[1] "t.e" "q.ick" "b.own" "f.x"
- substring(b, 2, 2) <- c("1", "2"); b # заменяет второй символ вектором</li>
   [1] "t1e" "q2ick" "b1own" "f2x"
- substring(b, 2) <- c("1", "22", "333"); b # заменяет второй символ вектором</li>
   [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

регистр букв

подстроки

замена

разбиение

векторы стро

объединение

пересечен 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</li>[1] "t.e" "q.ick" "b.own" "f.x"
- str\_sub(b, 2, 2) <- ".;."; b # заменяет второй символ другими</li>[1] "t.;.e" "q.;.ick" "b.;.own" "f.;.x"
- str\_sub(b, 2) <- c("1", "2"); b # заменяет второй символ вектором</li>[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"
- str\_sub(b, -1, -1)# последний символ[1] "e" "k" "n" "x"

# Выделение подстроки: library(stringr)

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

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

создание

манипуляции

соединение количество символов

подстроки

замена разбиение

векторы стро

объединение пересечение

сравнение

подмножес

метасимволы наборы символо

наборы символо квантификация

## Поиск вектора по фрагменту: grep()

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

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

создание

чтение

манипуляции соединение

количество символо

подстроки

замена разбиение

векторы стро

объединение

разность сравнение

регулярки

наборы символо квантификация

презентация доступна: http://idrv.ms/iPMoiZj

# Поиск вектора по фрагменту: library("stringr")

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

- b <- c("jumps", "over", "the", "lazy", "dog")
  - $\circ$  str\_detect("the", c(a, b)) элемент, содержащий подстроку [1] TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
  - $\circ$  c(a,b)[str\_detect("the c(a, b))] элемент, содержащий подстроку [1] "the" "the"

```
создание
```

чтение

манипуляции соединение

регистр букв

подстроки

nashuauua

векторы стро

-

объединение

разность сравнение

сравнение подмножество

регулярк

метасимволы

наооры симво

## Замена фрагмента

$$\label{eq:alpha} \begin{split} &a <- c(\text{"the quick brown fox jumps over"}) \\ &b <- c(\text{"jumps over the lazy dog"}) \end{split}$$

- $\circ \ \, \mathrm{sub}("o","\_",c(a,b))$  # 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

чтение

соединение

количество символов регистр букв подстроки поиск

**замена** разбиение

объединение

разность сравнение

регулярки

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

объединение

```
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"
```

### Пересечение векторов строк

создание

манипуляции

соединение

регистр букв

подстроки

замена

POUTONLI CTN

объединение

пересечение

сравнение

подмножество

регулярк

violony v ovneno

квантификаци

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

#### Разность векторов строк

разность

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

## 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?

#### подмножество

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

• is.element(b, a) [1] TRUE

• is.element(c, a) [1] FALSE

 $\circ$  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" "шоссе" "ка.а" "ма.а"
               \circ gsub("c\{1,\}", "\\.", a)
                                                                   # один и более раз
квантификация
                 [1] "тара" "ко.а" "шоссе" "ка.а" "ма.а"
               \circ gsub("c{1,2}", "\\.", a)
                                                              # от одного до двух раз
                 [1] "тара" "ко.а" "шоссе" "ка.а" "ма.а"
            Пробел: c\{1,2\} — правильно, c\{1,2\} — неправильно.
```

презентация доступна: http://idrv.ms/iPMoiZj

создание

манинулиции

количество символо

регистр оук

подстроки

TOWCY

Junciu

Dentrophi cipo.

ооъединение

пересечени

разпость

сравнени

метасимволы

наборы симво

вантификация

## Спасибо за внимание!

Пишите письма agricolamz@gmail.com