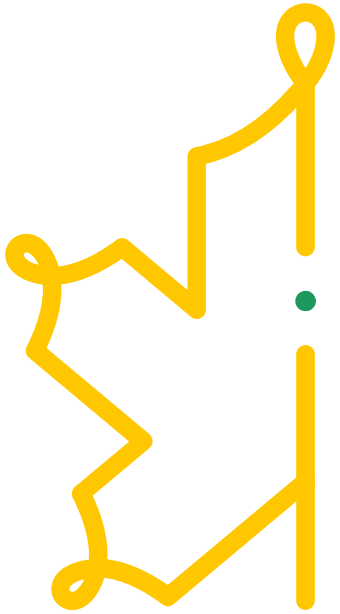
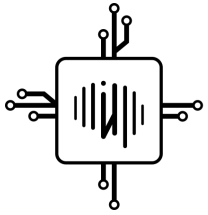


Razвед анализ данных

Введение в базовый R

Работа с данными табличного типа



Фонд интеллект

Анастасия Жарикова

Лекция 2 - 2022

Data Frame

Кадры данных или данные табличного вида

Данные табличного вида

Переменные

Наблюдения

Name	Age_month	Weight_g	Color	Sex	Group
R001	3	320	White	F	Case
R002	4	300	White	M	Control
R003	10	NA	Grey	M	Control
R004	2	280	Grey	F	Case
R005	NA	440	White	F	Control
R006	15	500	Grey	M	Case

Data frame

```
a <- c(5,4,8)
b <- c('a','b','c')
h <- c(T,F,T)
g <- c(1.5,2.5,3.7)
```

Data frame

```
a <- c(5,4,8)
b <- c('a','b','c')
h <- c(T,F,T)
g <- c(1.5,2.5,3.7)
df <- data.frame(a,b,h)
```

Data frame

```
a <- c(5,4,8)
b <- c('a','b','c')
h <- c(T,F,T)
g <- c(1.5,2.5,3.7)
df <- data.frame(a,b,h)
df
```

```
  a b     h
1 5 a  TRUE
2 4 b FALSE
3 8 c  TRUE
```

Data frame

```
a <- c(5,4,8)
b <- c('a','b','c')
h <- c(T,F,T)
g <- c(1.5,2.5,3.7)
df <- data.frame(a,b,h)
df
```

```
  a b     h
1 5 a  TRUE
2 4 b FALSE
3 8 c  TRUE
```

```
df <- data.frame(a = c(5,4,8),
                 b = c('a','b','c'),
                 h = c(T,F,T),
                 g = c(1.5,2.5,3.7))
df
```

```
  a b     h   g
1 5 a  TRUE 1.5
2 4 b FALSE 2.5
3 8 c  TRUE 3.7
```

Data frame

```
a <- c(5,4,8)
b <- c('a','b','c')
h <- c(T,F,T)
g <- c(1.5,2.5,3.7)
df <- data.frame(a,b,h)
df

df <- data.frame(a = c(5,4,8),
                 b = c('a','b','c'),
                 h = c(T,F,T),
                 g = c(1.5,2.5,3.7))
df
str(df)
```

```
  a b     h
1 5 a  TRUE
2 4 b FALSE
3 8 c  TRUE

  a b     h   g
1 5 a  TRUE 1.5
2 4 b FALSE 2.5
3 8 c  TRUE 3.7

'data.frame':    3 obs. of  4 variables:
 $ a: num  5 4 8
 $ b: Factor w/ 3 levels "a","b","c": 1 2 3
 $ h: logi  TRUE FALSE TRUE
 $ g: num  1.5 2.5 3.7
```

Data frame

```
a <- c(5,4,8)
b <- c('a','b','c')
h <- c(T,F,T)
g <- c(1.5,2.5,3.7)
df <- data.frame(a,b,h)
df

df <- data.frame(a = c(5,4,8),
                 b = c('a','b','c'),
                 h = c(T,F,T),
                 g = c(1.5,2.5,3.7))
df
str(df)
dim(df)
```

```
  a b     h
1 5 a  TRUE
2 4 b FALSE
3 8 c  TRUE

  a b     h   g
1 5 a  TRUE 1.5
2 4 b FALSE 2.5
3 8 c  TRUE 3.7

'data.frame':    3 obs. of  4 variables:
 $ a: num  5 4 8
 $ b: Factor w/ 3 levels "a","b","c": 1 2 3
 $ h: logi  TRUE FALSE TRUE
 $ g: num  1.5 2.5 3.7

[1] 3 4
```

Data frame

```r
a <- c(5,4,8)
b <- c('a','b','c')
h <- c(T,F,T)
g <- c(1.5,2.5,3.7)
df <- data.frame(a,b,h)
df

df <- data.frame(a = c(5,4,8),
                 b = c('a','b','c'),
                 h = c(T,F,T),
                 g = c(1.5,2.5,3.7))
df
str(df)
dim(df)
colnames(df)
```

```
  a b     h
1 5 a  TRUE
2 4 b FALSE
3 8 c  TRUE

  a b     h   g
1 5 a  TRUE 1.5
2 4 b FALSE 2.5
3 8 c  TRUE 3.7

'data.frame':    3 obs. of  4 variables:
 $ a: num  5 4 8
 $ b: Factor w/ 3 levels "a","b","c": 1 2 3
 $ h: logi  TRUE FALSE TRUE
 $ g: num  1.5 2.5 3.7

[1] 3 4

[1] "a" "b" "h" "g"
```

Data frame

```r
a <- c(5,4,8)
b <- c('a','b','c')
h <- c(T,F,T)
g <- c(1.5,2.5,3.7)
df <- data.frame(a,b,h)
df

df <- data.frame(a = c(5,4,8),
                 b = c('a','b','c'),
                 h = c(T,F,T),
                 g = c(1.5,2.5,3.7))
df
str(df)
dim(df)
colnames(df)
rownames(df)
```

```
  a b     h
1 5 a  TRUE
2 4 b FALSE
3 8 c  TRUE

  a b     h   g
1 5 a  TRUE 1.5
2 4 b FALSE 2.5
3 8 c  TRUE 3.7

'data.frame':    3 obs. of  4 variables:
 $ a: num  5 4 8
 $ b: Factor w/ 3 levels "a","b","c": 1 2 3
 $ h: logi  TRUE FALSE TRUE
 $ g: num  1.5 2.5 3.7

[1] 3 4

[1] "a" "b" "h" "g"

[1] "1" "2" "3"
```

Data frame

```r
a <- c(5,4,8)
b <- c('a','b','c')
h <- c(T,F,T)
g <- c(1.5,2.5,3.7)
df <- data.frame(a,b,h)
df

df <- data.frame(a = c(5,4,8),
                 b = c('a','b','c'),
                 h = c(T,F,T),
                 g = c(1.5,2.5,3.7))
df
str(df)
dim(df)
colnames(df)
rownames(df)
head(df)
```

```
  a b     h
1 5 a  TRUE
2 4 b FALSE
3 8 c  TRUE

  a b     h   g
1 5 a  TRUE 1.5
2 4 b FALSE 2.5
3 8 c  TRUE 3.7

'data.frame':    3 obs. of  4 variables:
 $ a: num  5 4 8
 $ b: Factor w/ 3 levels "a","b","c": 1 2 3
 $ h: logi  TRUE FALSE TRUE
 $ g: num  1.5 2.5 3.7

[1] 3 4

[1] "a" "b" "h" "g"

[1] "1" "2" "3"

  a b     h   g
1 5 a  TRUE 1.5
2 4 b FALSE 2.5
3 8 c  TRUE 3.7
```

Встроенные наборы данных

data()

?mtcars

mtcars

```
head(mtcars,4)
```

```
                mpg cyl disp  hp drat    wt  qsec vs am gear carb
Mazda RX4      21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag  21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
Datsun 710     22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive 21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
```

```
str(mtcars)
```

```
'data.frame':    32 obs. of  11 variables:
 $ mpg : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
 $ cyl : num  6 6 4 6 8 6 8 4 4 6 ...
 $ disp: num  160 160 108 258 360 ...
 $ hp  : num  110 110 93 110 175 105 245 62 95 123 ...
 $ drat: num  3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...
 $ wt  : num  2.62 2.88 2.32 3.21 3.44 ...
 $ qsec: num  16.5 17 18.6 19.4 17 ...
 $ vs  : num  0 0 1 1 0 1 0 1 1 1 ...
 $ am  : num  1 1 1 0 0 0 0 0 0 0 ...
 $ gear: num  4 4 4 3 3 3 3 4 4 4 ...
 $ carb: num  4 4 1 1 2 1 4 2 2 4 ...
```

mtcars

```
colnames(mtcars)
```

```
 [1] "mpg"  "cyl"  "disp" "hp"   "drat" "wt"   "qsec" "vs"   "am"   "gear"
[11] "carb"
```

```
table(mtcars$cyl)
```

```

 4  6  8
11  7 14
```

Data frame - Срезы

```
dim(mtcars)
```

```
[1] 32 11
```

```
head(mtcars,4)
```

```
                mpg cyl disp  hp drat    wt  qsec vs am gear carb
Mazda RX4      21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag  21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
Datsun 710     22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive 21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
```

```
mtcars[3,6]
```

```
[1] 2.32
```

Data frame - Срезы

```
mtcars[8,]
```

```
          mpg cyl  disp hp drat   wt qsec vs am gear carb
Merc 240D 24.4   4 146.7 62 3.69 3.19   20  1  0    4    2
```

```
mtcars[1:3,]
```

```
               mpg cyl disp  hp drat    wt  qsec vs am gear carb
Mazda RX4     21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag 21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
Datsun 710    22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
```

```
mtcars['Mazda RX4',c('mpg','cyl')]
```

```
          mpg cyl
Mazda RX4  21   6
```

Data frame - Срезы

```
mtcars[8,]
```

```
          mpg cyl  disp hp drat   wt qsec vs am gear carb
Merc 240D 24.4   4 146.7 62 3.69 3.19   20  1  0    4    2
```

```
mtcars[,3]
```

```
 [1] 160.0 160.0 108.0 258.0 360.0 225.0 360.0 146.7 140.8 167.6 167.6 275.8
[13] 275.8 275.8 472.0 460.0 440.0  78.7  75.7  71.1 120.1 318.0 304.0 350.0
[25] 400.0  79.0 120.3  95.1 351.0 145.0 301.0 121.0
```

Data frame - Выбор данных по условию

```
mtcars[mtcars$cyl > 4 & mtcars$cyl < 8,]
```

```
                mpg cyl  disp  hp drat    wt  qsec vs am gear carb
Mazda RX4      21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag  21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4
Hornet 4 Drive 21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
Valiant        18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
Merc 280       19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4
Merc 280C      17.8   6 167.6 123 3.92 3.440 18.90  1  0    4    4
Ferrari Dino   19.7   6 145.0 175 3.62 2.770 15.50  0  1    5    6
```

Data frame - Выбор данных по условию

```
table(mtcars$cyl)
```

```
 4  6  8
11  7 14
```

Data frame - Выбор данных по условию

```
table(mtcars$cyl)
vec <- c(4,8)
vec
```

```
 4  6  8
11  7 14

[1] 4 8
```

Data frame - Выбор данных по условию

```
table(mtcars$cyl)
vec <- c(4,8)
vec
mt <- mtcars[mtcars$cyl %in% vec,]
table(mt$cyl)
```

```
 4  6  8
11  7 14

[1] 4 8


 4  8
11 14
```

Data frame - Выбор данных по условию

```
table(mtcars$cyl)
vec <- c(4,8)
vec
mt <- mtcars[mtcars$cyl %in% vec,]
table(mt$cyl)
mt <- mtcars[!(mtcars$cyl %in% vec),]
table(mt$cyl)
```

```
 4  6  8
11  7 14

[1] 4 8


 4  8
11 14


6
7
```

Чтение и запись

Сохранение данных

```
write.table(mtcars, file='mtnew.tab', quote=T, col.names = T, row.names=T, sep='\t')
write.csv(mtcars, file='mtnew.csv')
save(mtcars, file='mtnew.RData')
```

Чтение данных

```
mt <- read.table('mtnew.tab', sep='\t')
head(mt)
```

```
                   mpg cyl disp  hp drat    wt  qsec vs am gear carb
Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

Чтение данных

```
mt <- read.table('mtnew.tab', skip=2)
head(mt)
```

```
                 V1   V2 V3  V4  V5   V6    V7    V8 V9 V10 V11 V12
1     Mazda RX4 Wag 21.0  6 160 110 3.90 2.875 17.02  0   1   4   4
2        Datsun 710 22.8  4 108  93 3.85 2.320 18.61  1   1   4   1
3    Hornet 4 Drive 21.4  6 258 110 3.08 3.215 19.44  1   0   3   1
4 Hornet Sportabout 18.7  8 360 175 3.15 3.440 17.02  0   0   3   2
5           Valiant 18.1  6 225 105 2.76 3.460 20.22  1   0   3   1
6        Duster 360 14.3  8 360 245 3.21 3.570 15.84  0   0   3   4
```

Чтение данных

```
mt <- read.table('mtnew.tab', skip=2, header=T)
head(mt)
```

```
      Mazda.RX4.Wag  X21 X6  X160 X110 X3.9 X2.875 X17.02 X0 X1 X4 X4.1
1         Datsun 710 22.8  4 108.0   93 3.85  2.320  18.61  1  1  4    1
2     Hornet 4 Drive 21.4  6 258.0  110 3.08  3.215  19.44  1  0  3    1
3  Hornet Sportabout 18.7  8 360.0  175 3.15  3.440  17.02  0  0  3    2
4            Valiant 18.1  6 225.0  105 2.76  3.460  20.22  1  0  3    1
5         Duster 360 14.3  8 360.0  245 3.21  3.570  15.84  0  0  3    4
6          Merc 240D 24.4  4 146.7   62 3.69  3.190  20.00  1  0  4    2
```

Еще полезные параметры:

dec = "."

comment.char = "#"

stringsAsFactors

Чтение данных

Имена колонок:

"mpg" "1" "disp new" "hp" "hp" "wt" "qsec" "vs" "am" "gear" "carb"

```
mt <- read.table('mtnew_ver2.tab', sep='\t')
head(mt)
```

```
                   mpg X1 disp.new  hp hp.1    wt  qsec vs am gear carb
Mazda RX4         21.0  6      160 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag     21.0  6      160 110 3.90 2.875 17.02  0  1    4    4
Datsun 710        22.8  4      108  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive    21.4  6      258 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout 18.7  8      360 175 3.15 3.440 17.02  0  0    3    2
Valiant           18.1  6      225 105 2.76 3.460 20.22  1  0    3    1
```

```
colnames(mt)
```

```
 [1] "mpg"      "X1"       "disp.new" "hp"       "hp.1"     "wt"
 [7] "qsec"     "vs"       "am"       "gear"     "carb"
```

RData

```
rm(list=ls())
ls()
```

```
character(0)
```

```
load('mtnew.RData')
ls()
```

```
[1] "mtcars"
```

Полезные функции

Сортировка

```
m1 <- mtcars[order(mtcars$qsec),]
m2 <- mtcars[order(mtcars$qsec, decreasing = T),]
head(m1, 3)
```

```
               mpg cyl disp  hp drat   wt  qsec vs am gear carb
Ford Pantera L 15.8   8  351 264 4.22 3.17 14.50  0  1    5    4
Maserati Bora  15.0   8  301 335 3.54 3.57 14.60  0  1    5    8
Camaro Z28     13.3   8  350 245 3.73 3.84 15.41  0  0    3    4
```

```
head(m2, 3)
```

```
              mpg cyl  disp  hp drat    wt  qsec vs am gear carb
Merc 230      22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2
Valiant       18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
Toyota Corona 21.5   4 120.1  97 3.70 2.465 20.01  1  0    3    1
```

Округление

```
x <- c(1,3,5,9,2,3)
mean(x)
```

```
[1] 3.833333
```

```
round(mean(x),3)
```

```
[1] 3.833
```

```
round(mean(x),1)
```

```
[1] 3.8
```

Не нужно указывать много знаков после запятой, если это не несет глубокого смысла

Округление

```
round(3.5, 0)
```

```
[1] 4
```

```
round(3.4, 0)
```

```
[1] 3
```

```
round(3.6, 0)
```

```
[1] 4
```

which

```
mtcars[1:3,1:3]
```

```
                mpg cyl disp
Mazda RX4      21.0   6  160
Mazda RX4 Wag  21.0   6  160
Datsun 710     22.8   4  108
```

which

```
mtcars[1:3,1:3]
which(mtcars$mpg == 21)
```

```
                mpg cyl disp
Mazda RX4      21.0   6  160
Mazda RX4 Wag  21.0   6  160
Datsun 710     22.8   4  108

[1] 1 2
```

which

```
mtcars[1:3,1:3]
which(mtcars$mpg == 21)
mtcars$mpg == 21
```

```
               mpg cyl disp
Mazda RX4     21.0   6  160
Mazda RX4 Wag 21.0   6  160
Datsun 710    22.8   4  108

[1] 1 2

 [1]  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[13] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[25] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
```

which

```r
mtcars[1:3,1:3]
which(mtcars$mpg == 21)
mtcars$mpg == 21
which(colnames(mtcars) == 'disp')
```

```
               mpg cyl disp
Mazda RX4     21.0   6  160
Mazda RX4 Wag 21.0   6  160
Datsun 710    22.8   4  108

[1] 1 2

 [1]  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[13] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[25] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE

[1] 3
```

which

```r
mtcars[1:3,1:3]
which(mtcars$mpg == 21)
mtcars$mpg == 21
which(colnames(mtcars) == 'disp')
which.min(mtcars$qsec)
```

```
               mpg cyl disp
Mazda RX4     21.0   6  160
Mazda RX4 Wag 21.0   6  160
Datsun 710    22.8   4  108

[1] 1 2

 [1]  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[13] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[25] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE

[1] 3

[1] 29
```

which

```r
mtcars[1:3,1:3]
which(mtcars$mpg == 21)
mtcars$mpg == 21
which(colnames(mtcars) == 'disp')
which.min(mtcars$qsec)
which.max(mtcars$qsec)
```

```
               mpg cyl disp
Mazda RX4     21.0   6  160
Mazda RX4 Wag 21.0   6  160
Datsun 710    22.8   4  108

[1] 1 2

 [1]  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[13] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[25] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE

[1] 3

[1] 29

[1] 9
```

which

```
mtcars[1:3,1:3]
which(mtcars$mpg == 21)
mtcars$mpg == 21
which(colnames(mtcars) == 'disp')
which.min(mtcars$qsec)
which.max(mtcars$qsec)
```

```
               mpg cyl disp
Mazda RX4     21.0   6  160
Mazda RX4 Wag 21.0   6  160
Datsun 710    22.8   4  108

[1] 1 2

 [1]  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[13] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[25] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE

[1] 3

[1] 29

[1] 9
```

Работа с пропущенными данными

Данные о качестве воздуха

```
head(airquality)
```

```
  Ozone Solar.R Wind Temp Month Day
1    41     190  7.4   67     5   1
2    36     118  8.0   72     5   2
3    12     149 12.6   74     5   3
4    18     313 11.5   62     5   4
5    NA      NA 14.3   56     5   5
6    28      NA 14.9   66     5   6
```

Данные о качестве воздуха

```
head(airquality)
str(airquality)
```

```
  Ozone Solar.R Wind Temp Month Day
1    41     190  7.4   67     5   1
2    36     118  8.0   72     5   2
3    12     149 12.6   74     5   3
4    18     313 11.5   62     5   4
5    NA      NA 14.3   56     5   5
6    28      NA 14.9   66     5   6

'data.frame':    153 obs. of  6 variables:
 $ Ozone  : int  41 36 12 18 NA 28 23 19 8 NA ...
 $ Solar.R: int  190 118 149 313 NA NA 299 99 19 194 ...
 $ Wind   : num  7.4 8 12.6 11.5 14.3 14.9 8.6 13.8 20.1 8.6 ...
 $ Temp   : int  67 72 74 62 56 66 65 59 61 69 ...
 $ Month  : int  5 5 5 5 5 5 5 5 5 5 ...
 $ Day    : int  1 2 3 4 5 6 7 8 9 10 ...
```

Данные о качестве воздуха

```
head(airquality)
str(airquality)
mean(airquality$Ozone)
```

```
  Ozone Solar.R Wind Temp Month Day
1    41     190  7.4   67     5   1
2    36     118  8.0   72     5   2
3    12     149 12.6   74     5   3
4    18     313 11.5   62     5   4
5    NA      NA 14.3   56     5   5
6    28      NA 14.9   66     5   6

'data.frame':    153 obs. of  6 variables:
 $ Ozone  : int  41 36 12 18 NA 28 23 19 8 NA ...
 $ Solar.R: int  190 118 149 313 NA NA 299 99 19 194 ...
 $ Wind   : num  7.4 8 12.6 11.5 14.3 14.9 8.6 13.8 20.1 8.6 ...
 $ Temp   : int  67 72 74 62 56 66 65 59 61 69 ...
 $ Month  : int  5 5 5 5 5 5 5 5 5 5 ...
 $ Day    : int  1 2 3 4 5 6 7 8 9 10 ...

[1] NA
```

Работа с отсутствующими данными

- NA - пропущенное значение: is.na()
- NaN - результат недопустимой арифметической операции: is.nan()
- NULL - отсутствие субъекта: is.null()

Работа с отсутствующими данными

Не учитывать отсутствующие данные

```
mean(airquality$Ozone, na.rm = T)
```

```
[1] 42.12931
```

Работа с отсутствующими данными

Удалить строки с отсутствующими данными

Удаляется ВСЯ строка!

```
dim(airquality)
air <- na.omit(airquality)
```

```
[1] 153   6
```

Работа с отсутствующими данными

Удалить строки с отсутствующими данными

Удаляется ВСЯ строка!

```
dim(airquality)
air <- na.omit(airquality)
dim(air)
```

```
[1] 153   6
```

```
[1] 111   6
```

Работа с отсутствующими данными

Удалить строки с отсутствующими данными

Удаляется ВСЯ строка!

```
dim(airquality)
air <- na.omit(airquality)
dim(air)
mean(air$Ozone)
```

```
[1] 153   6

[1] 111   6

[1] 42.0991
```

Работа с отсутствующими данными

Удалить строки с отсутствующими данными

Удаляется ВСЯ строка!

```
dim(airquality)
air <- na.omit(airquality)
dim(air)
mean(air$Ozone)
mean(airquality$Ozone, na.rm = T)
```

```
[1] 153   6
```

```
[1] 111   6
```

```
[1] 42.0991
```

```
[1] 42.12931
```

Работа с отсутствующими данными

```
air <- head(airquality,10)
is.na(air$Ozone)
```

```
[1] FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE
```

Работа с отсутствующими данными

```
air <- head(airquality,10)
is.na(air$Ozone)
sum(is.na(air$Ozone))
```

```
 [1] FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE
```

```
[1] 2
```

Работа с отсутствующими данными

```
air <- head(airquality,10)
is.na(air$Ozone)
sum(is.na(air$Ozone))
which(is.na(air$Ozone))
```

```
 [1] FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE
```

```
[1] 2
```

```
[1]  5 10
```

Работа с отсутствующими данными

```
air <- head(airquality,10)
is.na(air$Ozone)
sum(is.na(air$Ozone))
which(is.na(air$Ozone))
anyNA(air$Ozone)
```

```
 [1] FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE
```

```
[1] 2
```

```
[1]  5 10
```

```
[1] TRUE
```

NaN

```
0/0
```

```
[1] NaN
```

Удаление столбцов

```
head(air,3)
```

```
  Ozone Solar.R Wind Temp Month Day
1    41     190  7.4   67     5   1
2    36     118  8.0   72     5   2
3    12     149 12.6   74     5   3
```

```
air$Ozone = NULL
head(air,3)
```

```
  Solar.R Wind Temp Month Day
1     190  7.4   67     5   1
2     118  8.0   72     5   2
3     149 12.6   74     5   3
```

Списки

Списки

Можно хранить данные разного типа

```
L <- list("A",c(1,2),30)
L
```

```
[[1]]
[1] "A"

[[2]]
[1] 1 2

[[3]]
[1] 30
```

Списки

Можно хранить данные разного типа

```
L <- list("A",c(1,2),30)
L
print('L1')
L1 <- list(L,40)
L1
```

```
[[1]]
[1] "A"

[[2]]
[1] 1 2

[[3]]
[1] 30


[1] "L1"


[[1]]
[[1]][[1]]
[1] "A"

[[1]][[2]]
[1] 1 2

[[1]][[3]]
[1] 30


[[2]]
[1] 40
```

Списки

Добавление нового элемента

```
L[[4]] <- 'new_element'
L
```

```
[[1]]
[1] "A"

[[2]]
[1] 1 2

[[3]]
[1] 30

[[4]]
[1] "new_element"
```

Списки

Срезы

```
L
```

```
[[1]]
[1] "A"

[[2]]
[1] 1 2

[[3]]
[1] 30

[[4]]
[1] "new_element"
```

Списки

Срезы

```
L
L[3]
```

```
[[1]]
[1] "A"

[[2]]
[1] 1 2

[[3]]
[1] 30

[[4]]
[1] "new_element"


[[1]]
[1] 30
```

Списки

Срезы

```
L
L[3]
L[[3]]
```

```
[[1]]
[1] "A"

[[2]]
[1] 1 2

[[3]]
[1] 30

[[4]]
[1] "new_element"


[[1]]
[1] 30


[1] 30
```

Списки

```
L1
```

```
[[1]]
[[1]][[1]]
[1] "A"

[[1]][[2]]
[1] 1 2

[[1]][[3]]
[1] 30


[[2]]
[1] 40
```

Списки

```
L1
L1 [[1]]
```

```
[[1]]
[[1]][[1]]
[1] "A"

[[1]][[2]]
[1] 1 2

[[1]][[3]]
[1] 30


[[2]]
[1] 40


[[1]]
[1] "A"

[[2]]
[1] 1 2

[[3]]
[1] 30
```

Списки

```
L1
L1 [[1]]
L1 [[1]] [[2]]
```

```
[[1]]
[[1]][[1]]
[1] "A"

[[1]][[2]]
[1] 1 2

[[1]][[3]]
[1] 30


[[2]]
[1] 40


[[1]]
[1] "A"

[[2]]
[1] 1 2

[[3]]
[1] 30


[1] 1 2
```

Списки

```
L1
L1 [[1]]
L1 [[1]] [[2]]
```

```
[[1]]
[[1]][[1]]
[1] "A"

[[1]][[2]]
[1] 1 2

[[1]][[3]]
[1] 30


[[2]]
[1] 40


[[1]]
[1] "A"

[[2]]
[1] 1 2

[[3]]
[1] 30


[1] 1 2
```

Списки

```
L <- list(10,20)
```

Списки

```
L <- list(10,20)
L
```

```
[[1]]
[1] 10

[[2]]
[1] 20
```

Списки

```
L <- list(10,20)
L
L$abc <- 123
```

```
[[1]]
[1] 10

[[2]]
[1] 20
```

Списки

```
L <- list(10,20)
L
L$abc <- 123
L
```

```
[[1]]
[1] 10

[[2]]
[1] 20


[[1]]
[1] 10

[[2]]
[1] 20

$abc
[1] 123
```

Списки

```
L <- list(10,20)
L
L$abc <- 123
L
names(L)
```

```
[[1]]
[1] 10

[[2]]
[1] 20


[[1]]
[1] 10

[[2]]
[1] 20

$abc
[1] 123


[1] ""    ""    "abc"
```

Списки

```
L <- list(10,20)
L
L$abc <- 123
L
names(L)
L[[3]]
```

```
[[1]]
[1] 10

[[2]]
[1] 20


[[1]]
[1] 10

[[2]]
[1] 20

$abc
[1] 123


[1] ""    ""    "abc"


[1] 123
```

Списки

```
L <- list(10,20)
L
L$abc <- 123
L
names(L)
L[[3]]
L$abc
```

```
[[1]]
[1] 10

[[2]]
[1] 20


[[1]]
[1] 10

[[2]]
[1] 20

$abc
[1] 123


[1] ""    ""    "abc"


[1] 123


[1] 123
```

Списки

```
L <- list(10,20)
L
L$abc <- 123
L
names(L)
L[[3]]
L$abc
L['abc']]
```

```
[[1]]
[1] 10

[[2]]
[1] 20


[[1]]
[1] 10

[[2]]
[1] 20

$abc
[1] 123


[1] ""    ""    "abc"


[1] 123


[1] 123


[1] 123
```

Циклы

Циклы

Циклы в R медленные!

Можно придумать, как обойтись без цикла

```
vec <- c(1:3)
vec + 1
```

```
[1] 2 3 4
```

for

```
for(year in c(2010:2015)){
  print(paste('The year is', year))
}
```

```
[1] "The year is 2010"
[1] "The year is 2011"
[1] "The year is 2012"
[1] "The year is 2013"
[1] "The year is 2014"
[1] "The year is 2015"
```

for

```
mt <- mtcars
head(mt,2)
```

```
              mpg cyl disp  hp drat    wt  qsec vs am gear carb
Mazda RX4      21   6  160 110  3.9 2.620 16.46  0  1    4    4
Mazda RX4 Wag  21   6  160 110  3.9 2.875 17.02  0  1    4    4
```

```
for(i in 1:nrow(mt)){
  mt$new[i] <- i^2
}
head(mt,4)
```

```
                mpg cyl disp  hp drat    wt  qsec vs am gear carb new
Mazda RX4      21.0   6  160 110 3.90 2.620 16.46  0  1    4    4   1
Mazda RX4 Wag  21.0   6  160 110 3.90 2.875 17.02  0  1    4    4   4
Datsun 710     22.8   4  108  93 3.85 2.320 18.61  1  1    4    1   9
Hornet 4 Drive 21.4   6  258 110 3.08 3.215 19.44  1  0    3    1  16
```

Деление

```
5/2
```

```
[1] 2.5
```

```
5%%2
```

```
[1] 1
```

for + if

```
mt <- mtcars
for(i in 1:nrow(mt)){
  if((i%%2) == 0){
    mt$new[i] = i^2
    mt$type[i] = 'even'
  }
  else{
    mt$new[i] = i^3
    mt$type[i] = 'odd'
  }
}
mt[1:4,(ncol(mt)-5):ncol(mt)]
```

```
                vs am gear carb new type
Mazda RX4        0  1    4    4   1  odd
Mazda RX4 Wag    0  1    4    4   4 even
Datsun 710       1  1    4    1  27  odd
Hornet 4 Drive   1  0    3    1  16 even
```

ifelse

Description ifelse returns a value with the same shape as test which is filled with elements selected from either yes or no depending on whether the element of test is TRUE or FALSE.

Usage ifelse(test, yes, no)

```
v = c(14,7,6,9,2)
ifelse(v %% 2 == 1,"odd","even")
```

```
[1] "even" "odd"  "even" "odd"  "even"
```

for + if + break

```
x <- c(1, 4, 6, 3, 7, 2, 3, 9)
for(val in x){
if (val == 3){
break
}
print(val)
}
```

```
[1] 1
[1] 4
[1] 6
```

for + if + break

```
x <- c(1, 4, 6, 3, 7, 2, 3, 9)
for(val in x){
if (val == 3){
break
}
print(val)
}
```

```
[1] 1
[1] 4
[1] 6
```

```
x[1:(which(x == 3)[1] - 1)]
```

```
[1] 1 4 6
```

Для вектора длиной 10000

Time difference of 0.4777727 secs - с циклом

Time difference of 0.03518772 secs - без цикла

for + if + next

```
x <- c(1, 4, 6, 3, 7, 2, 3, 9)
for(val in x){
if (val == 3){
next
}
print(val)
}
```

```
[1] 1
[1] 4
[1] 6
[1] 7
[1] 2
[1] 9
```

for + if + next

```
x <- c(1, 6, 3, 3, 9)
for(val in x){
if (val == 3){
next
}
print(val)
}
```

```
[1] 1
[1] 6
[1] 9
```

```
x[x != 3]
```

```
[1] 1 6 9
```

Для вектора длиной 10000

Time difference of 1.005887 secs - с циклом

Time difference of 0.03425145 secs - без цикла

while

```
i <- 1
while (i < 6){
  print(i)
  i = i+1
}
```

```
[1] 1
[1] 2
[1] 3
[1] 4
[1] 5
```

Случайная матрица

```
set.seed(123)
mt <- matrix(sample(1:5,10000,replace=T),ncol=10)
dim(mt)
```

```
[1] 1000   10
```

```
mt[1:5,1:5]
```

```
     [,1] [,2] [,3] [,4] [,5]
[1,]    3    4    5    4    1
[2,]    3    4    4    1    2
[3,]    2    2    3    2    3
[4,]    2    4    1    4    5
[5,]    3    5    2    4    2
```

Случайная матрица

```
colnames(mt) <- paste("D",1:ncol(mt),sep='_')
rownames(mt) <- paste("Stud",1:nrow(mt),sep='.')
head(mt)
```

```
       D_1 D_2 D_3 D_4 D_5 D_6 D_7 D_8 D_9 D_10
Stud.1   3   4   5   4   1   1   5   1   5    1
Stud.2   3   4   4   1   2   3   5   1   3    3
Stud.3   2   2   3   2   3   3   5   5   1    3
Stud.4   2   4   1   4   5   2   3   5   5    2
Stud.5   3   5   2   4   2   3   5   1   3    1
Stud.6   5   4   5   5   1   1   5   3   3    4
```

Создать вектор из средних по столбцам mt

Давайте сначала создадим вектор нужной длины, заполним NA

```
x <- rep(NA,ncol(mt))
length(x)
```

```
[1] 10
```

```
x
```

```
 [1] NA NA NA NA NA NA NA NA NA NA
```

Создать вектор из средних по столбцам mt

Теперь заполним созданный вектор средними

```
for(i in 1:ncol(mt)){
  x[i] <- mean(mt[,i])
}
x
```

```
[1] 2.985 2.987 2.966 2.954 2.974 3.003 2.960 2.938 2.989 2.962
```

Функции семейства apply

apply

```
x.2 <- apply(mt,2,mean)
x
```

```
[1] 2.985 2.987 2.966 2.954 2.974 3.003 2.960 2.938 2.989 2.962
```

```
x.2
```

```
  D_1   D_2   D_3   D_4   D_5   D_6   D_7   D_8   D_9  D_10
2.985 2.987 2.966 2.954 2.974 3.003 2.960 2.938 2.989 2.962
```

```
a <- apply(mt,1,sum)
a[1:6]
```

```
Stud.1 Stud.2 Stud.3 Stud.4 Stud.5 Stud.6
    30     29     29     33     29     36
```

lapply

Возвращает список

```
x <- list(a=1,b=1:3,d=10:100)
lapply(x, FUN = length)
```

```
$a
[1] 1

$b
[1] 3

$d
[1] 91
```

sapply

Возвращает вектор (если не получается - список)

```
x <- list(a=1,b=1:3,d=10:100)
sapply(x, FUN = length)
```

```
a  b  d
1  3 91
```