Язык R

Артем Артемов, Светлана Виноградова, Елена Ставровская

1 октября 2014

Зачем нужен R?

- Быстрая статистическая обработка данных
- Построение красивых графиков
- Бесплатный, удобный, быстрый для изучения язык
- Множество дополнительных пакетов, в особенности для биоинформатики

R – векторизованный язык

- Основной тип данных – вектор (упорядоченный набор чисел)
- Идея – работать с набором данных как с одним числом (параллельно обрабатывать все значения набора)
  - Это позволяет обходиться (в ряде случаев) без циклов

Вектор

```
> x<-1:5 ; y<-6:10
> x
[1] 1 2 3 4 5
> y
[1]  6  7  8  9 10
> x+y
[1]  7  9 11 13 15
```

```
> x*2
[1]  2  4  6  8 10
> x>4
[1] FALSE FALSE FALSE FALSE
TRUE
> y==7
[1] FALSE  TRUE FALSE FALSE
FALSE
> x*y
[1]  6 14 24 36 50
```

Как можно создать вектор?

Оператор c()

```
> c(1, 2, 3)
[1] 1 2 3
```

Последовательности

```
> 1:10
 [1]  1  2  3  4  5  6  7  8  9 10
> seq(from=1, to=8, by=2)
[1] 1 3 5 7
> seq(1, 10, 2)
[1] 1 3 5 7 9
```

Как можно создать вектор?

Объединение нескольких векторов

```
> x<-c(1, 2, 3)
> x<-c(x, 1:3); x
[1] 1 2 3 1 2 3
```

Повторы

```
> rep(0.5, 6)
[1] 0.5 0.5 0.5 0.5 0.5 0.5
```

Для целых чисел (работает быстрее)

```
> rep.int(1, 5)
[1] 1 1 1 1 1
```

Как можно создать вектор?

Распределение

- ✓ Нормальное распределение:
- ✓ dnorm(x) – плотность распределения
- ✓ pnorm(q) – функция распределения
- ✓ qnorm(p) – квантильная функция

Случайная генерация из распределения: > set.seed(100)

```
> rnorm(5)
[1]  1.1568405 -0.8248219  0.1428891 -0.4784408  0.7561443
```

Равномерное

runif(n, min=0, max=1)

```
> runif(5, 0, 1)
[1] 0.1972687 0.3090867 0.2865924 0.1409635 0.3441481
```

Биномиальное

rbinom(n, size, prob)

```
> rbinom(10, 100, 0.5)
[1] 54 47 55 50 47 45 52 45 58 52
```

Пуассона

rpois(n, lambda)

```
> rpois(10, 4)
 [1]  2  3  2  4 10  3  2  3  5  6
```

Срезы

```
> x<-c(1, 5, 7, 9, 15, 3)
> x[1]
[1] 1
> x[2:4]
[1] 5 7 9
> x[c(2, 5)]
[1]  5 15
```

```
> x[-1]
[1]  5  7  9 15  3
> x[-(1:3)]
[1]  9 15  3
> x[x>5]
[1]  7  9 15
> x[x>5 & x<10]
[1] 7 9
```

Задание - 1

- Сгенерировать выборку из чисел от 1 до 100
- Нарисовать график y=x^2

Что такое data frame

- Структура данных: таблица из нескольких векторов (по столбцам), в разных столбцах могут быть данные разных типов

	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
Datsun 710	22.8	4	108.0	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout	18.7	8	360.0	175	3.15	3.440	17.02	0	0	3	2
Valiant	18.1	6	225.0	105	2.76	3.460	20.22	1	0	3	1
Duster 360	14.3	8	360.0	245	3.21	3.570	15.84	0	0	3	4
Merc 240D	24.4	4	146.7	62	3.69	3.190	20.00	1	0	4	2
Merc 230	22.8	4	140.8	95	3.92	3.150	22.90	1	0	4	2
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
Merc 450SE	16.4	8	275.8	180	3.07	4.070	17.40	0	0	3	3

Как создать свой data frame?

```
> n <- c(2, 3, 5)
> s <- c("aa", "bb", "cc")
> b <- c(TRUE, FALSE, TRUE)
> df <- data.frame(n, s, b)
```

Или короче:

```
> df <- data.frame(n=c(2, 3, 5),
s=c("aa", "bb", "cc"),
b= c(TRUE, FALSE, TRUE))
```

Основные команды

```
> df <- data.frame(n=c(2, 3, 5),  s=c("aa", "bb", "cc"), b=
c(TRUE, FALSE, TRUE))
> df
  n  s     b
1 2 aa  TRUE
2 3 bb FALSE
3 5 cc  TRUE
> df$n
[1] 2 3 5
> colnames(df)
[1] "n" "s" "b"
> rownames(df)
[1] "1" "2" "3"
> dim(df)
[1] 3 3
```

Обращение к столбцу по имени, можно использовать tab!

Важно, что это имена строк, а не числа!

Использование data()

> mtcars

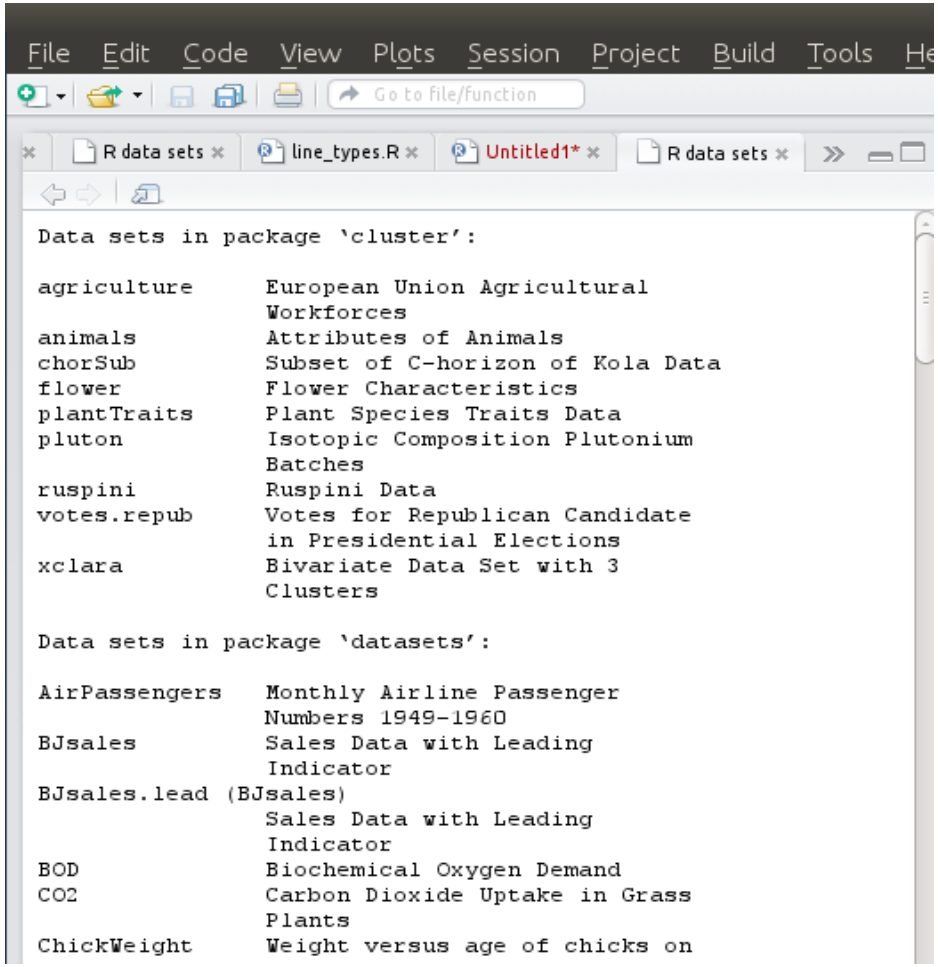
```
                   mpg cyl  disp  hp drat    wt  qsec vs am gear carb
Mazda RX4         21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag     21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4
Datsun 710        22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive    21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout 18.7   8 360.0 175 3.15 3.440 17.02  0  0    3    2
Valiant           18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
Duster 360        14.3   8 360.0 245 3.21 3.570 15.84  0  0    3    4
Merc 240D         24.4   4 146.7  62 3.69 3.190 20.00  1  0    4    2
Merc 230          22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2
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
Merc 450SE        16.4   8 275.8 180 3.07 4.070 17.40  0  0    3    3
```

Командой data() можно посмотреть, какие выборки загружены для использования !

> data()

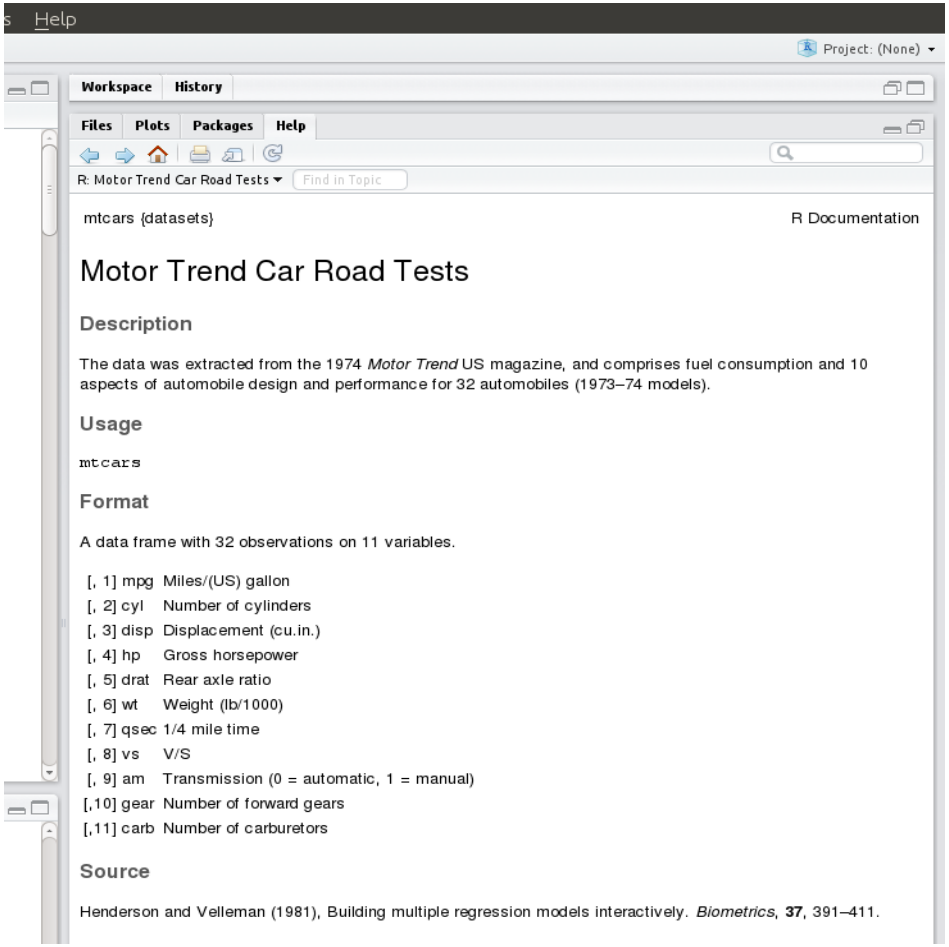
Использование data()

> data()



Можно узнать о доступной выборке более подробно

> ?mtcars



Выбор строк, столбцов, ячеек

```
> mtcars[12,2]    # строка 12, столбец 2
[1] 8

> 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,]    # строки 1 - 3, все столбцы
      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
```

Выбор строк, столбцов, ячеек

```
> mtcars[,2]          # все строки, столбец 2
 [1] 6 6 4 6 8 6 8 4 4 6 6 8 8 8 8 8 8 4 4 4 4 8 8 8 8 4 4 4 8 6 8 4
```

```
> mtcars[c(1,13),]   # строки 1 и 13, все столбцы
           mpg cyl  disp  hp drat   wt  qsec vs am gear carb
Mazda RX4   21.0   6 160.0 110 3.90 2.62 16.46  0  1    4    4
Merc 450SL  17.3   8 275.8 180 3.07 3.73 17.60  0  0    3    3
```

```
> mtcars[c(1,3,7,13),1]
# строки 1, 3, 7 и 13, столбец 1
[1] 21.0 22.8 14.3 17.3
```

Добавить столбец

```
> dim(mtnew)
[1] 33 11
> num<-1:33
> mtnew<-cbind(mtnew, num)          #добавляем столбец
> mtnew[30:33,]
               mpg cyl disp  hp drat   wt  qsec vs am gear carb num
Ferrari Dino  19.7   6  145 175 3.62 2.77 15.50  0  1    5    6  30
Maserati Bora 15.0   8  301 335 3.54 3.57 14.60  0  1    5    8  31
Volvo 142E    21.4   4  121 109 4.11 2.78 18.60  1  1    4    2  32
Lada          21.0   6  150 120 4.00 2.50 16.46  1  1    4    4  33
```

Добавить строку

> mtnew<-mtcars

> dim(mtnew)

```
[1] 32 11
```

> mtnew[1,]

```
          mpg cyl disp  hp drat   wt  qsec vs am gear carb
Mazda RX4  21   6  160 110  3.9 2.62 16.46  0  1    4    4
```

> newcar<-c(21, 6, 150, 120, 4.0, 2.5, 16.46, 1, 1, 4, 4)#работает только если все данные одного типа!!!!

> newcar<-data.frame(mpg=21, cyl=4, disp=100, hp=80, drat=1, wt=2, qsec=16, vs=1,am=0, gear=4, carb=1) # data.frame из 1 строки

> mtnew<-rbind(mtnew, newcar) #добавляем строку

> rownames(mtnew)[33]<-"Lada" #присваиваем ей имя

> mtnew[30:33,]

```
               mpg cyl disp  hp drat   wt  qsec vs am gear carb
Ferrari Dino  19.7   6  145 175 3.62 2.77 15.50  0  1    5    6
Maserati Bora 15.0   8  301 335 3.54 3.57 14.60  0  1    5    8
Volvo 142E    21.4   4  121 109 4.11 2.78 18.60  1  1    4    2
Lada          21.0   6  150 120 4.00 2.50 16.46  1  1    4    4
```

Задание - 2

- Выбрать из таблицы mtcars только те машины, у которых количество цилиндров от 4 до 8
- Отсортировать таблицу по мощности автомобиля

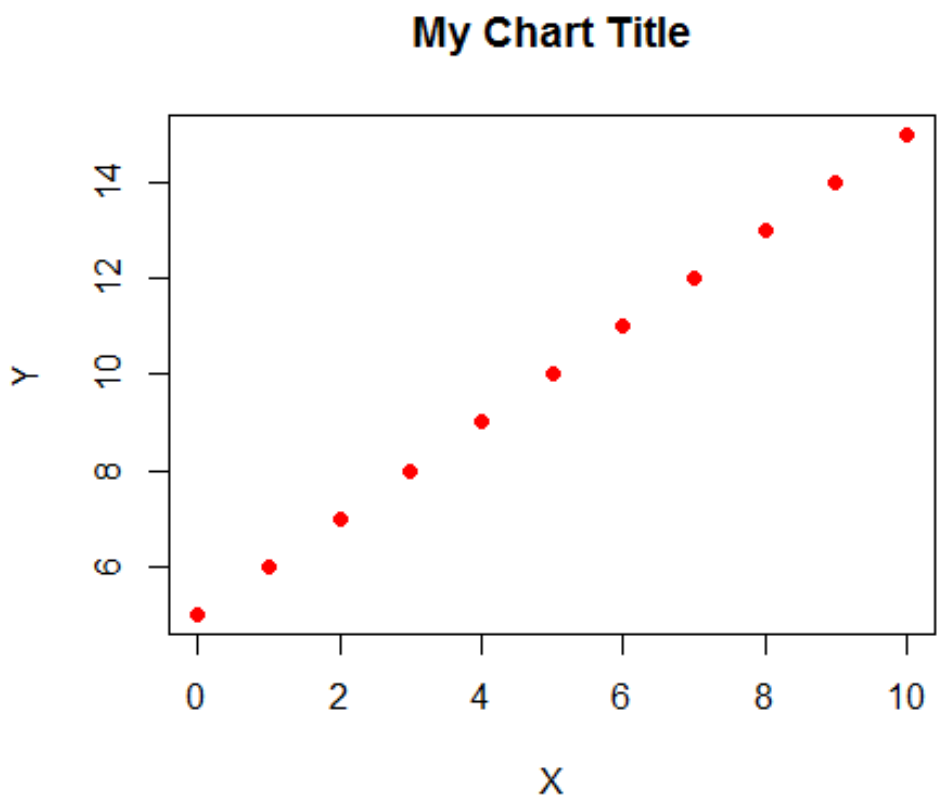
Логические условия и order

```
> mtcars1 <- mtcars[mtcars$cyl>4 & mtcars$cyl<8,]
> mtcars1
                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
```

```
> mtcars1[order(mtcars1$drat),]
               mpg cyl  disp  hp drat    wt  qsec vs am gear carb
Valiant        18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
Hornet 4 Drive 21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
Ferrari Dino   19.7   6 145.0 175 3.62 2.770 15.50  0  1    5    6
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
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
```

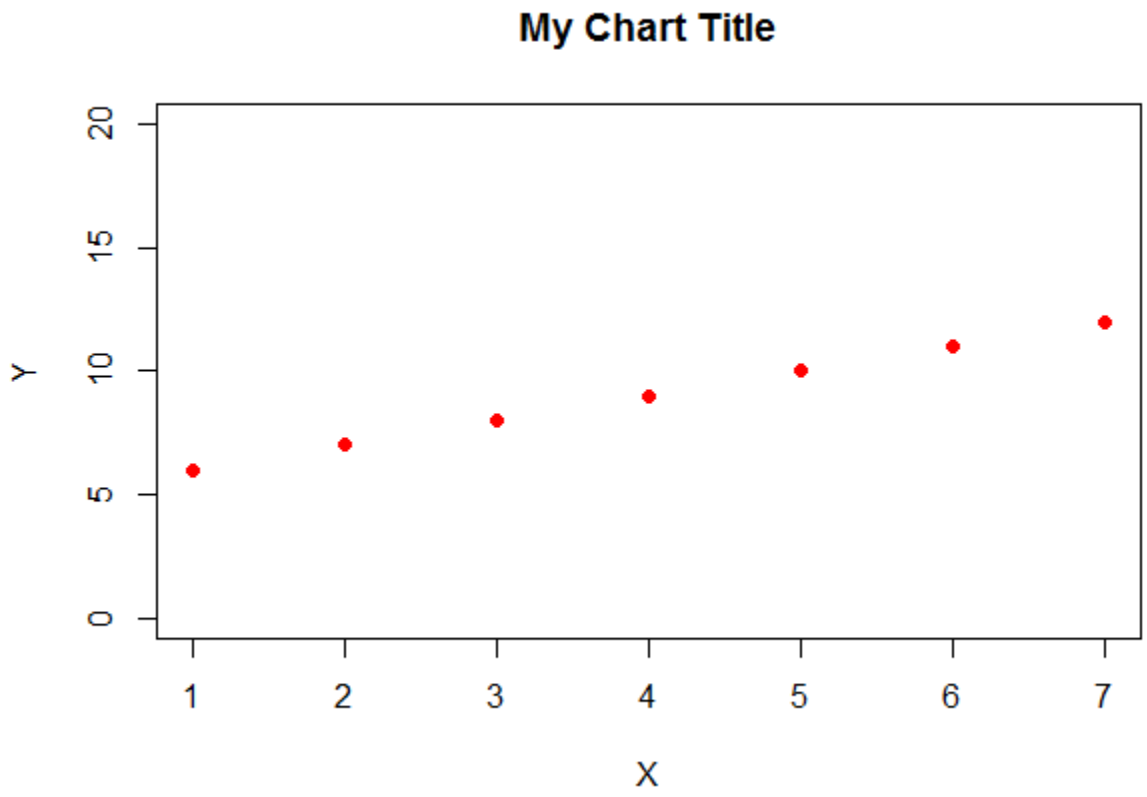
Самый простой график

```
>x_data <- c(0:10)
>y_data <- x_data +5
>plot(x_data, y_data,  main = "My Chart Title", xlab ="X", ylab = "Y", pch=16, col =
"red")
```



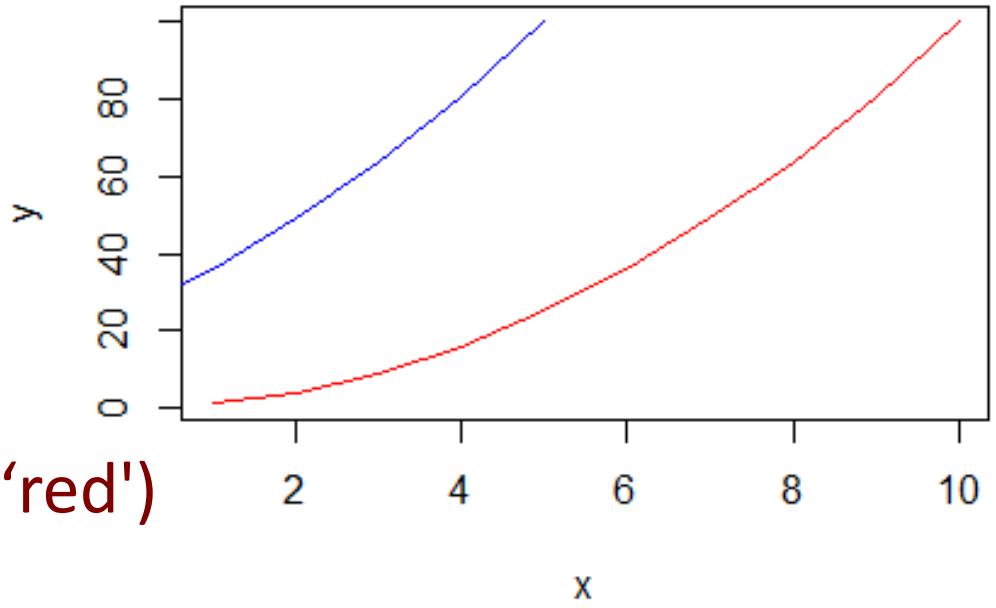
Параметры xlim, ylim

```
>plot(x_data, y_data, main = "My Chart Title", xlab ="X", ylab = "Y", pch=16, col = "red", xlim=c(1,7), ylim=c(0, 20))
```



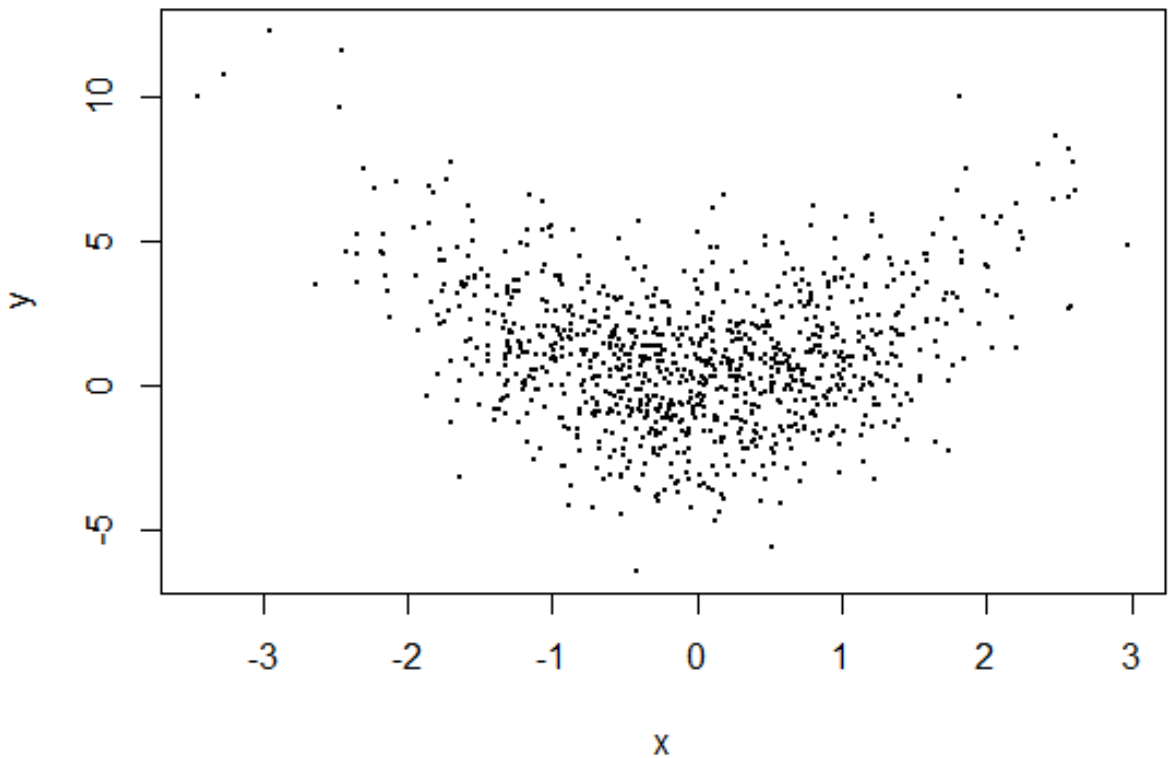
Линии

```
> x <- 1:10
> y <- x*x
> z <- x-5
```



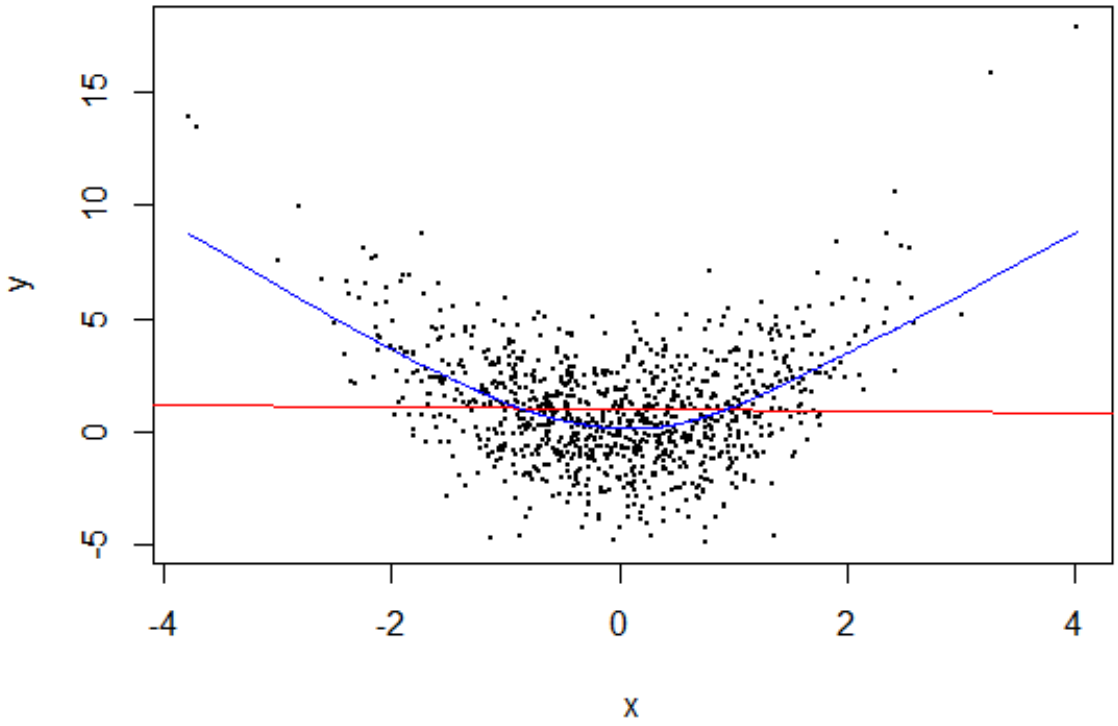

```
> plot(y ~ x, type="l", col = 'red')
> lines(y ~ z, col = 'blue')
```

Scatterplots




```
> x<-rnorm(1000)
> y<-x*x + rnorm(1000, sd=2)
> plot(x, y, pch=19, cex=0.3)
```

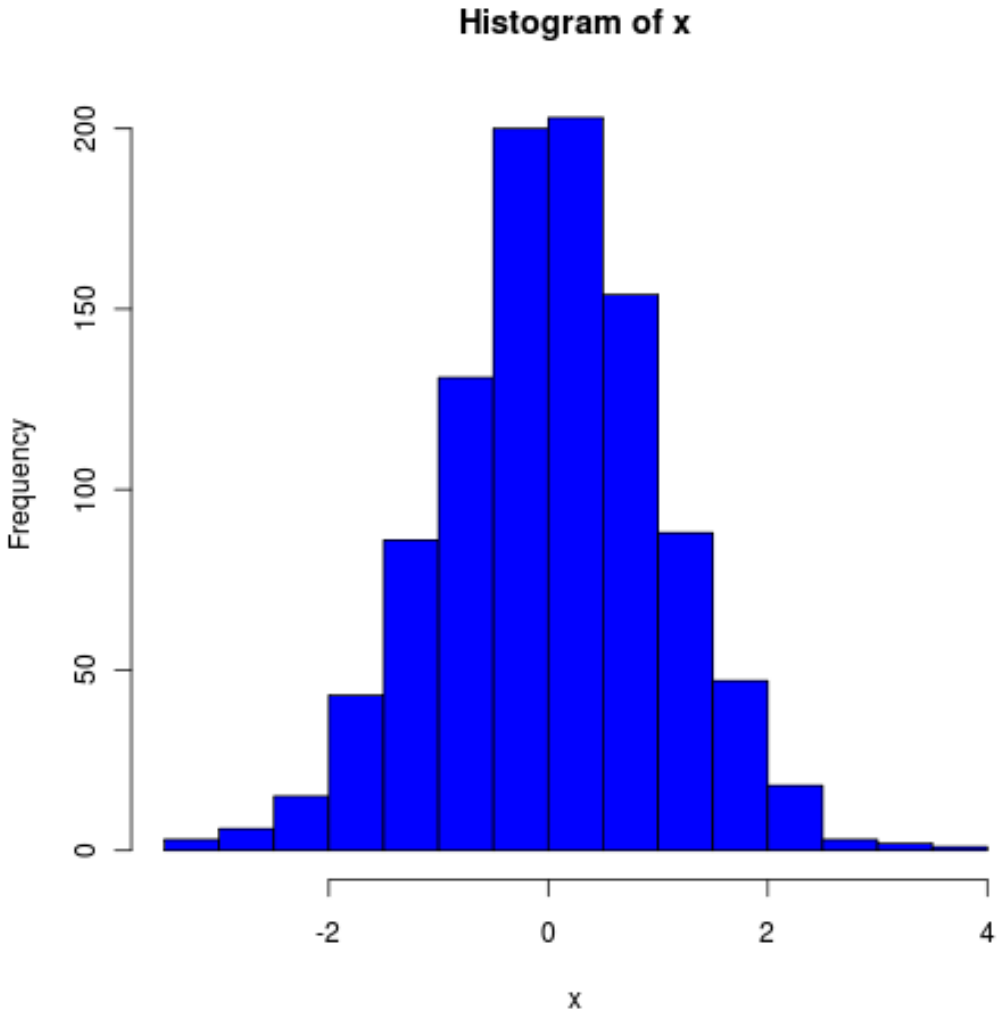
Scatterplots: добавим линии



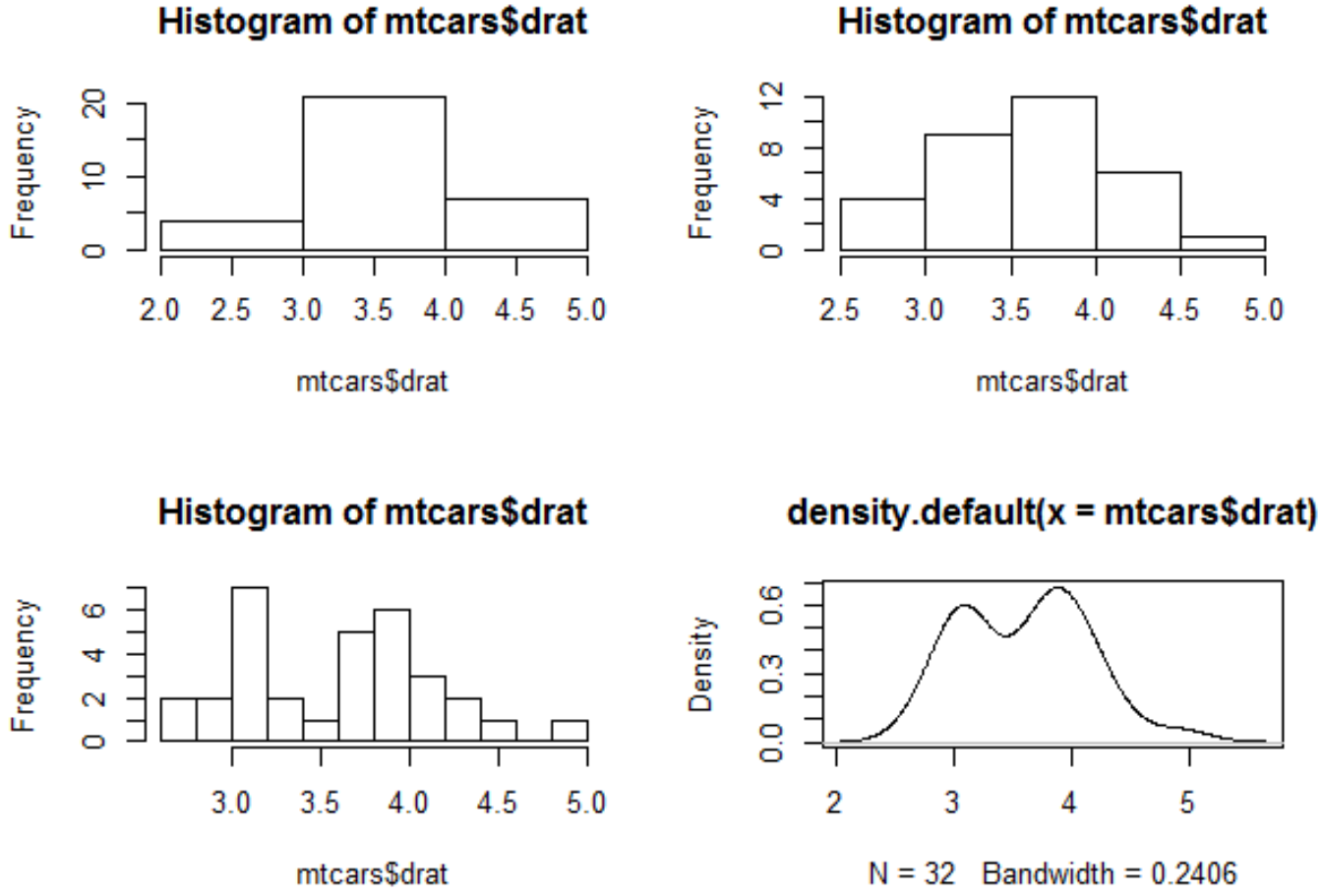

```
> abline(lm(y~x), col="red")
> lines(lowess(y~x), col="blue")
```

Гистограммы

```
> x=rnorm(1000)
> hist(x, col='blue')
```

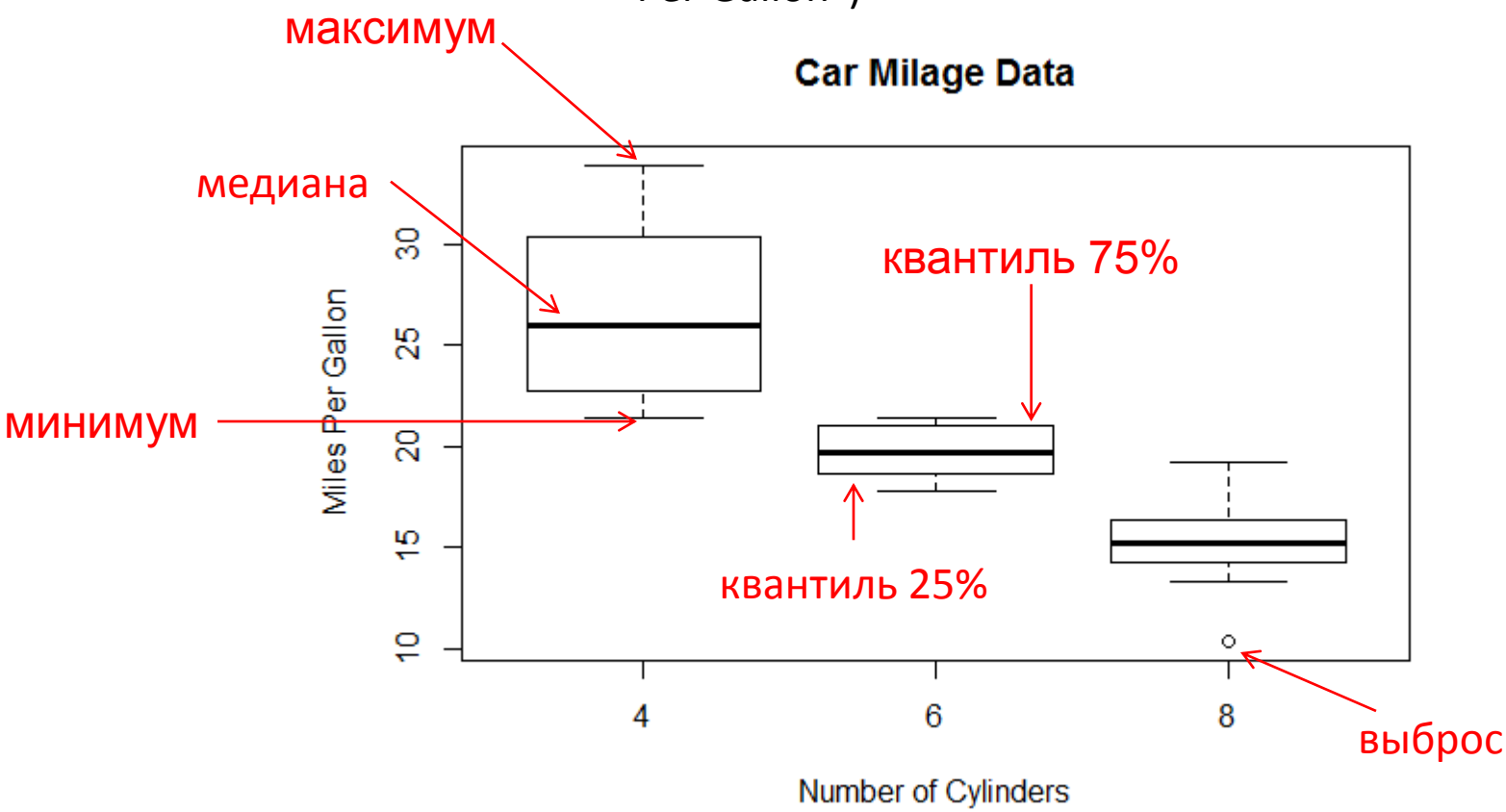


Гистограммы



Boxplots

```
> boxplot(mpg~cyl,data=mtcars, main="Car Milage Data",    xlab="Number of Cylinders", ylab="Miles Per Gallon")
```



Сохранение графика в файл

```
> png(file="Pictures/boxplot.png”, width=400, height=350, res=72)
> boxplot(x,y)
> dev.off()
```

Другие форматы:

pdf("mygraph.pdf")	pdf file: для печати
win.metafile("mygraph.wmf")	windows metafile
png("mygraph.png")	png file: для веба
jpeg("mygraph.jpg")	jpeg file: не рекомендуем
bmp("mygraph.bmp")	bmp file
postscript("mygraph.ps")	postscript file

Задания - 3

- Построить scatter plot зависимости мощности от экономичности автомобиля
- Построить распределение мощностей автомобилей

Работа с файлами: основные функции

Чтение	Запись	Применение
read.table	***write.table***	Чтение/запись табулированных текстовых файлов
read.csv	***write.csv***	Чтение/запись файлов в формате CSV
readLines	***writeLines***	Чтение/запись текстовых файлов по строкам
load	***save***	Загрузка/сохранение объектов R из/в бинарные файлы (.RData)

Работа с файлами: рабочая директория

Узнать рабочую директорию:

```
> getwd()
[1] "C:/Users/anna/FBB/R"
```

Поменять рабочую директорию:

```
> setwd("Week3")  # путь указан относительно рабочей директории!
> getwd()
[1] "C:/Users/anna/FBB/R/Week3"
```

Узнать список файлов в рабочей директории

```
> dir()
```

Узнать список файлов в указанной директории

```
> dir("C:/Users/anna/FBB/R/")
```

<u>В RStudio:</u>

закладка Files (справа внизу) -> выбрать нужную директорию -> More -> Set As Working Directory

Работа с файлами: *read.table*

- Читает файл с разделителями
- Возвращает ***data.frame***

```
> students <- read.table("FBBRStudents.tab",sep="\t",
header=T)

> students[101:102,]
             Name        Faculty         Level Year
101 Широкий В. Р.    химический  специалитет    4
102 Базылев С. С. биологический  бакалавриат    1
```

Работа с файлами: *read.table*

Основные аргументы:

- `file` – имя файла или соединение (connection)
- `header` – есть ли в файле заголовок? (по умолчанию, FALSE)
- `sep` – разделитель полей (колонок) (по умолчанию, пробел)
- `colClasses` – вектор с названиями классов колонок
- `nrows` – количество строчек, которые нужно прочитать
- `skip` – количество строчек, которые нужно пропустить
- `comment.char` – знак комментариев
- `stringsAsFactors` – преобразовывать строковые поля в фактор? (по умолчанию, TRUE)

Работа с файлами: *read.table*

```
> students<-read.table("FBBRStudents.tab",sep="\t",header=T,
+ colClasses = c("character","factor","factor","integer"))

> str(students)
'data.frame': 141 obs. of 4 variables:
 $ Name : chr "Антонов С. В." "Дмитриев Д. И." "Золотов И.
А." "Иванова Т. В." ...
 $ Faculty: Factor w/ 10 levels "биологический",..: 3 3 3 3
3 3 3 3 3 3 ...
 $ Level : Factor w/ 3 levels "бакалавриат",..: 3 3 3 3 3 3
3 3 3 3 ...
 $ Year : int 3 3 3 3 4 4 4 4 4 4 ...
```

Работа с файлами: *read.csv, write.csv, readLines*

- ***read.csv*** – то же, что read.table, но с другими дефолтными значениями параметров (`header=TRUE, sep=“,”`)
  - См. также **read.csv2** (для русской локали: десятичные разделители “,”, разделители элементов списка “;”)

- ***write.csv:***

```
> write.csv(students, "FBBRStudents.csv")
```

- ***readLines***:

```
> lines <- readLines("FBBRStudents.txt“,3)
> lines
[1] "Name\tFaculty\tLevel\tYear"
[2] "Антонов С. В.\tмеханико-математический\tспециалитет\t3"
[3] "Дмитриев Д. И.\tмеханико-математический\tспециалитет\t3"
```

Работа с файлами: *save, load*

Сохраняем объекты *students* и *lines* в файл:

```
> save(students, lines, file="Students.RData")
```

Удаляем все объекты из рабочего пространства:

```
> rm(list=ls())
> ls()
character(0)
```

Загружаем объекты из файла:

```
> load("Students.RData")
> ls()
[1] "lines" "students“  # объекты появляются в
                        # рабочем пространстве
```

Соединения

- **file** – открывает соединение с файлом
- **gzfile, bzfile** – открывает соединение с архивированным файлом
- **url** – открывает соединение с веб-страницей

```
> con <- file("FBBRStudents.txt", "r")
> readLines(con, 1)
[1] "Name\tFaculty\tLevel\tYear"
> readLines(con, 1)
[1] "Антонов С. В.\tмеханико-математический\tспециалитет\t3"
> close(con)

> con <- gzfile("FBBRStudents.gz")
> read.csv(con, nrow=2)
X Name Faculty Level Year
1 1 Антонов С. В. механико-математический специалитет 3
2 2 Дмитриев Д. И. механико-математический специалитет 3
> close(con)
```

Задание - 3

- Загрузить файл stats.txt

http://kodomo.fbb.msu.ru/FBB/year_10/term8/R_course/stats.txt

- Вычислите длины контигов и добавьте их как отдельный столбец
- Постройте распределение покрытия контигов (short1_cov)
- (*) Вычислите N50
- Создайте любую таблицу в Excel, сохраните в csv и загрузите в R