The basics of working in R

The objective of the lecture:

1. Basic R tools needed to work in R.
2. Access R packages
3. Learn the methods and rules for loading data into R

Recommended literature:

1. Robert I. Kabakov. R in action. Analysis and visualization of data in the language R. DMK Press, 2014. - 588 p.
2. An Introduction to R. internet source: https://cran.r-project.org/doc/manuals/r-release/R-intro.html Packages in R.
3. Fundamentals of programming in R. Video (10 min)

https://www.youtube.com/watch?v=DXzHCVEkFz8&list=PLu5flfwrnSD7wxKXFgsiuxrMKLfFHm6CD&index=10

1. Package Overview

A package is a collection of functions created to perform a specific class of tasks, or a collection of tables with data

Getting package information

1. not installed - the package was not installed using the install.packages function. You can get a list of such packages with the following command:

```
>setdiff(row.names(available.packages()), .packages(all.available = TRUE))
```

2. installed but not connected - the package was installed using the install.packages function, but not connected using the library function. You can get a list of such packages with the following command:

```
>setdiff(.packages(all.available = TRUE), (.packages()))
```

3. installed and connected - the package was installed using the install.packages function and connected using the library function. You can get a list of such packages with the following command

```
>(.packages())
```

2. Installing packages in R

Installing a new package (Internet connection required):

```
> install.packages("package_name")
```

3. Using Packages

Download an already installed package:

```
>library(package)
```

or

```
>require(installed_package_name)
```

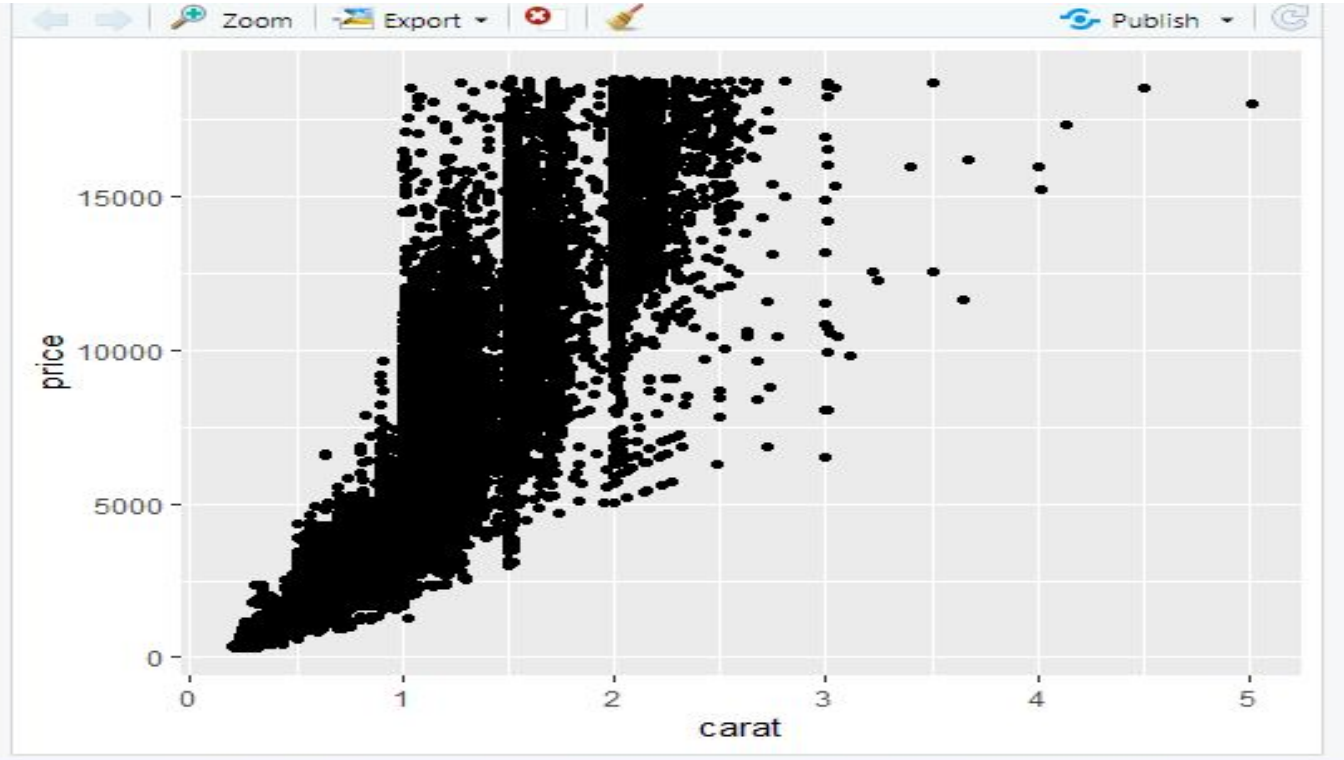
When downloaded, the package may report various diagnostic information. You can suppress the output of these messages with the suppressPackageStartupMessages () function.

```
>suppressPackageStartupMessages(library(rvest))
```

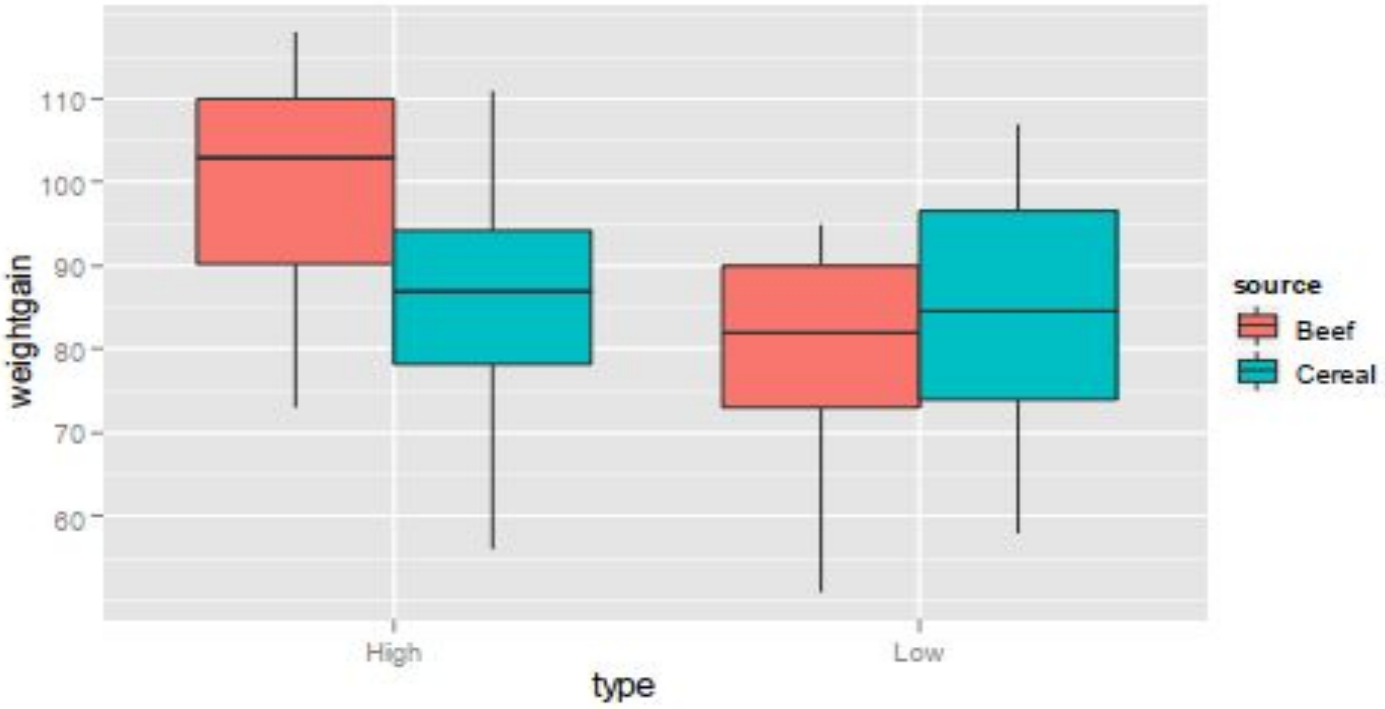
The exercise

Connect the ggplot2 package:

```
>library(ggplot2)
>qplot(carat, price, data=diamonds)
```



```
library(HSAUR2)
data(weightgain)
library(ggplot2)
ggplot(data = weightgain, aes(x = type, y = weightgain)) +
geom_boxplot(aes(fill = source))
```




```
>help(package = “package_name")
```

Package removal

```
>remove.packages(“package_name")
```

For example:

```
>remove.packages(“ggplot2")
```

Packages

Other functions for working with packages:

.libPaths() # returns the directory where the packages are installed

library() # listing installed packages
search() # listing downloaded packages

1. Preparing data for R

Data can be entered from the keyboard, imported from text files, from Microsoft Excel and Access.

1. Preparing data for R

Microsoft Excel is one of the most common programs for preparing data for R.

Before uploading to R, the Excel file is usually saved as a text file .txt or .csv

Some data preparation rules

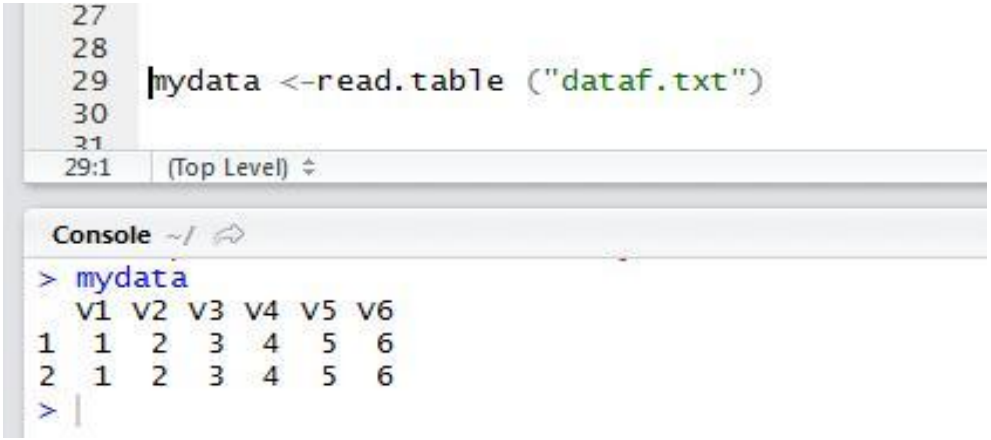
- ✔ No empty cells – missing values are denoted as NA
- ✔ Assign a name to each variable:
- ✔ No spaces in names
- ✔ Names must not start with dots or numbers
- ✔ The file should be placed in the current working folder

	A	B	C	D
1	Treatment	Barrel	Length	Weight
2	Control	Control3	29.28	0.992
3	Control	Control3	29.83	0.772
4	Control	Control3	31.93	0.894
5	Control	Control3	26.63	0.822

Preparing Data for R

Consider reading data from a text document: R can read data stored in a text (ASCII) file. Three functions are used for this: read.table () (which has two options: read.csv (), scan ().

For example, if we have a file data.txt, then in order to read it you can type:
mydata <-read.table ("dataf.txt")



read.table() function

Key arguments:

- **File** = "*имя*.txt": file name (or URL link)
- **Header** = TRUE : are there column headers in the file
- **Sep** = = "\t" or sep = "," : file delimiter

An example of LOADING DATA

Iris Dataset
(archive.ics.uci.edu/ml/datasets/Iris)

download.file() – downloading file

read.csv() – reading data in csv

Upload the file to R

```
>fileUrl <- "http://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data"

>download.file(fileUrl, destfile="./iris.csv")
```

```
>iris.data <- read.csv("./iris.csv")  #   iris.data became data frame
```

Primary analysis in R

```
>head(iris.data, 1)
```

```
  X5.1 X3.5 X1.4 X0.2 Iris.setosa
1  4.9  3.0  1.4  0.2  Iris-setosa
```

```
colnames(iris.data) <- c("Sepal.Length", "Sepal.Width",
"Petal.Length", "Petal.Width", "Species")
```

Saving a workspace

```
> save.image(file =
  "pH_experiment.rda")
```

Downloading a file from the Internet

Birth data for boys and girls from 1940 to 2002 in the United States

```
>source("http://www.openintro.org/stat/data/present.R")
>str(present)
>head(present)
>summary(present)
```

4. The treatment of missing values

Consider the following example: suppose we have the result of a survey of a seven employees. They were asked: how many hours they sleep on average, while one of the respondents refused to answer, another said "I do not know", and the third at the time of the survey was simply not in the office. So there was a missing data:

```
>h <- c(8, 10, NA, NA, 8, NA, 8)
```

```
 h
```

```
 [1] 8 10 NA NA 8 NA 8
```

From the example you can see that NA should be entered without quotes

4. The treatment of missing values

If we try to calculate the average value (the mean () function), we get:

```
>mean(h)
[1] NA
```

To calculate the average value without including NA, you can use one of two ways:

```
>mean(h, na.rm=TRUE)
>[1] 8.5

>mean(na.omit(h))
>[1] 8.5
```

4. The treatment of missing values

Often there is another problem: how to make a substitution of the missing data, say, replace all NA with the average value.

```
>h[is.na(h)] <- mean(h, na.rm=TRUE)

>h

>[1] 8.0 10.0 8.5 8.5 8.0 8.5 8.0
```

In the left part of the first expression, indexing is performed, that is, the selection of the desired values, such as those that are missing (is.na ()). After the expression is executed, the "old" values disappear.

Examples

American Community Survey provides downloadable data from a variety of community surveys in the United States. Use the download.file () command to download data from an Idaho Housing Survey in 2006 from:

https://d396qusza40orc.cloudfront.net/getdata%2Fdata%2Fss06hid.csv
Download this data in R. An encoding book that describes variable names can be found at:

https://d396qusza40orc.cloudfront.net/getdata%2Fdata%2FPUMSDataDict06.pdf
How many categories are worth $ 1 million or more?

```
fileUrl <- ”https://d396qusza40orc.cloudfront.net/getdata%2Fdata%2Fss06hid.csv”
download.file(fileUrl, destfile="./a1.csv")
data1 <- read.csv("./a1.csv")
res<-sum(data1$VAL==24, na.rm=TRUE)
res
```

Self Test Questions

What data sources for R are you aware of?
How to read text files in R?
How to read files from MS Excel in R?
How to read Internet files in R?

Conclusions of the lecture

WE LEARNED :

- ✔ What data sources can be used in RWhat data is considered suitable for analysis in R
- ✔ How to download data from files *.txt, Excel, Internet and databasesHow to work with missing valuesHow to name columns and rows