

Lab1 - R basics

Klimov, Titova

null

Part 1. RStudio as the IDE for R

- ▶ Free software
- ▶ Object-oriented
- ▶ Available for all major operating systems (Mac OS, Linux, Windows)
- ▶ Can easily be extended through the use of user-defined functions
- ▶ Widely used in political science, statistics, econometrics, actuarial sciences, sociology, finance, etc.

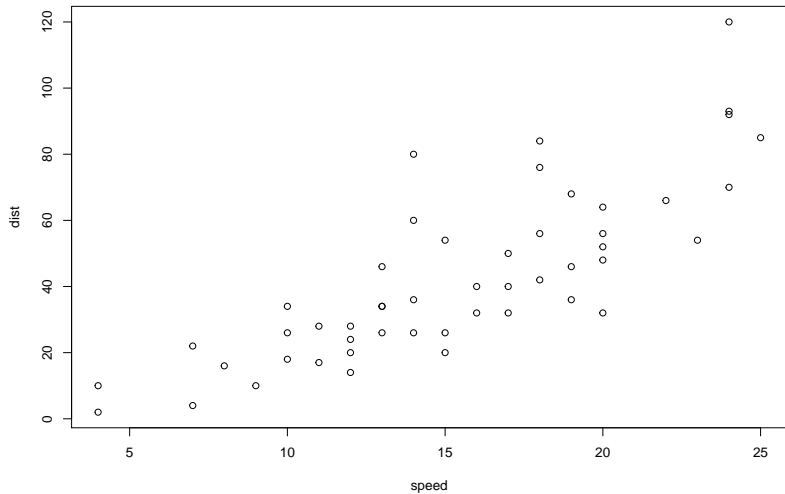
For more details on authoring R presentations please visit
<https://support.rstudio.com/hc/en-us/articles/200486468>.

Howto run code

```
summary(cars)
```

```
##           speed           dist
##  Min.      : 4.0      Min.      :  2.00
##  1st Qu.:12.0      1st Qu.: 26.00
##  Median :15.0      Median : 36.00
##  Mean   :15.4      Mean   : 42.98
##  3rd Qu.:19.0      3rd Qu.: 56.00
##  Max.    :25.0      Max.     :120.00
```

Slide With Plot



Part 2. Intro in R Language - python-alike console

```
# A comment  
5 # print a number
```

```
## [1] 5
```

```
2+2 # calculations
```

```
## [1] 4
```

```
x <- 10 # store  
x # print
```

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

```
(x <- 11) # store and print
```

```
## [1] 11
```

Vector - basic array

`c`, which is short for concatenate (paste together) [<https://cran.r-project.org/doc/contrib/Torfs+Brauer-Short-R-Intro.pdf>]

```
Ages <- c(32, 45, 15, 22, 29, 54, 35) # store a specified  
Ages
```

```
## [1] 32 45 15 22 29 54 35
```

Vector - basic array (2)

Generate vector with **seq** and set it to **Participant_id**

```
(Participant_id <- seq(1,7)) # define vector as a sequence
```

```
## [1] 1 2 3 4 5 6 7
```

```
(Participant_id <- (1:7)) # makes the same
```

```
## [1] 1 2 3 4 5 6 7
```

Seq could have additional parameters: seq(1,20, **by=2**)

```
## [1] 1 3 5 7 9 11 13 15 17 19
```

Operations with vectors

Builtin statistics (mean, sd, statistical tests ...)

```
mean(Ages)
```

```
## [1] 33.14286
```

```
length(Ages)
```

```
## [1] 7
```


Matrices

```
mymatrix <- cbind(Participant_id, Ages, TestingSkills)  
is.matrix(mymatrix); is.vector(TestingSkills) # to stack es
```

```
## [1] TRUE
```

```
## [1] TRUE
```

Let's check how much rows and columns we have

```
nrow(mymatrix)
```

```
## [1] 7
```

```
ncol(mymatrix)
```

```
## [1] 3
```

Matrices is good but dataframe is better

Didn't work in matrices:

```
mymatrix$Participant_id
```

```
## Error in mymatrix$Participant_id: $ operator is invalid
```

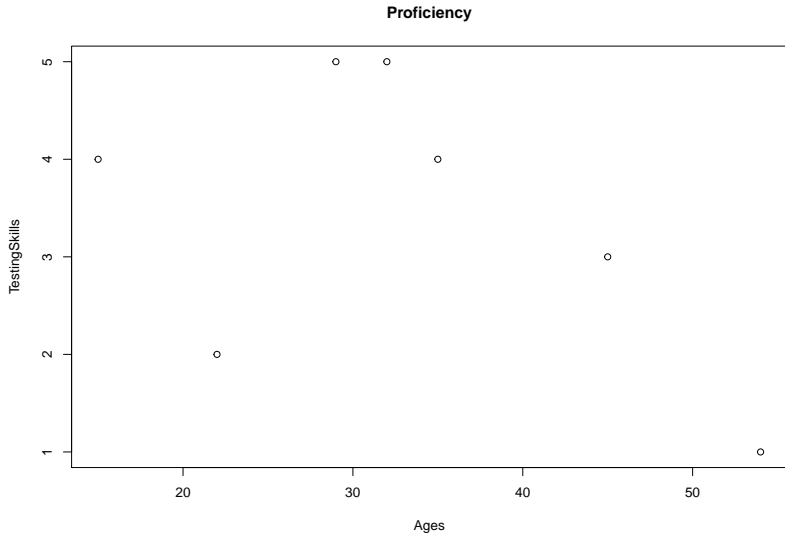
But works in Dataframes:

```
as.data.frame(mymatrix)$Participant_id
```

```
## [1] 1 2 3 4 5 6 7
```

```
mydataframe <- as.data.frame(mymatrix)
```

Building a plot



```
plot(mydataframe$Ages, mydataframe$TestingSkills  
     , xlab="Ages", ylab="TestingSkills"  
     , main="Proficiency"  
     )
```

Lists, dataframes

Can be viewed with:


- ▶ `data`
- ▶ `str(data)`
- ▶ `View(data)`
- ▶ `DT(data)`
- ▶ `kable(data)`

Motor Trend Car Road dataset

Description The data was extracted from the 1974 Motor Trend US magazine, and comprises fuel consumption and 10 aspects of automobile design and performance for 32 automobiles (1973–74 models).

Show entries

Search:

	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
Mazda RX4 Wag	21	6	160	110	3.9	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108	93	3.85	2.32	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	3.44	17.02	0	0	3	2

Showing 1 to 5 of 32 entries

Previous 2 3 4 5 6 7 Next

Task 1

Load *mtcars_data* in a variable and display its - summary - last 5 rows

Task 2

- ▶ Display the number of variables
- ▶ Display the number of records
- ▶ Add a new column “#” and number all the rows

Task 3

- ▶ Describe (Display all the info for) car #7
- ▶ Display the summary for column “qsec”
- ▶ Display first 6 values of “qsec”

Task 4

For first column display: - values from 10 to 20 rows - values for rows #4, #7, #9 in one line

Task 5

- ▶ Copy *kilometers per liter* to a new column “kml2”
- ▶ Declare new “kml2” variable from values of “miles per gallon”, knowing that ***1 mile per liter = 0,425144 kilometer per liter***

Task 6

- Display *True/False table* for all the rows of column *cyl*, checking if they are the same as 6 (+ Find the *Median* of cylinder numbers)

Task 7

- ▶ Display top 5 rows, where *disp* is lesser than 150

Task 8

- Display *car names*, for which *disp* is greater than 200

Task 9

- ▶ Count the *Mean Value* of *wt*, rounded to the nearest 10th.

Task 10

- ▶ Count those car models, that have 8 cylinders and horsepower greater than 150.
- ▶ From the subset above name a car with minimal $qsec$.