Stat 291 - Recitation 1

Orçun Oltulu

22 / 10 / 2021

Question of the day: Why R?

- R is a language and environment for statistical computing and graphics.
- R provides a wide variety of statistical (linear and nonlinear modeling, classical statistical tests, time-series analysis, classification, clustering, …) and graphical techniques, and is highly extensible.
- R is **free** and open source !
- R runs on a variety of platforms including Windows, Unix and MacOS.
- R can be used for many purposes: from simplest arithmetical calculations to much more advanced statistical modeling.
- R has a dynamic community; it is super easy to get help from its documentations or directly from experts via some web pages like stackoverflow.

Exercises:

Exercise 1: Arithmetic Operators

Using basic arithmetic operators, calculate following:

- 12 + 6 = ?
- 22 - 11 = ?
- 3 x 9 = ?
- 34 / 2 = ?
- 4^3 = ?
- $20 \equiv ?\ (\text{mod}3)$

```
12 + 6
```

```
## [1] 18
```

```
22 - 11
```

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

```
3 * 9
```

```
## [1] 27
```

```
34 / 2
```

```
## [1] 17
```

```
4^3
```

```
## [1] 64
```

```
20 %% 3
```

```
## [1] 2
```

Exercise 2: Using Help Menu

Use **help** menu in order to reach out description of those following built-in functions.

sum, length, unique, mean, var, typeof, seq, rep, max, min, which, ls, rm

To use **help** menu, you can either run **"?*functionName*"** or if you are using RStudio, there is a sub-menu on the right-hand side of your window. Click 'Help' and simply search for **"*functionName*"**.

```
?mean
```

Exercise 3: Setting Variables

The operators <- and = can be used, *almost* interchangeably, to assign to variable in the same environment.

Create those variables based on your information;

- Age
- Height
- CGPA
- EyeColor

```
Age <- 25
Height = 172
CGPA <- 3.8
EyeColor = "Brown"
```

Ways to print a variable;

- simply write the name of the variable and press 'Enter'

- use print() command
- use cat() command

```r
Height
```

```
## [1] 172
```

```r
# remember print function prints only one object at a time
print(CGPA)
```

```
## [1] 3.8
```

```r
# when you want to print out multiple items, you can combine them with cat.
cat(" I am", Age, "years old.", "\n", "Also, I have", EyeColor, "eyes.")
```

```
##  I am 25 years old.
##  Also, I have Brown eyes.
```

Now, check which variables you have in your environment. (Hint: Use ls() command)

```r
ls()
```

```
## [1] "Age"      "CGPA"     "EyeColor" "Height"
```

Now, remove ‘CGPA’ variable from your environment. (Hint: Use rm() command)

```r
rm(CGPA)
```

Exercise 4: Data Types

Data Type	Description	Example
Character	Character or a string	“superman”, “milk”
Numeric (double)	Real numbers or decimals	9.65, pi, sqrt(2)
Integer	Whole numbers	-3L, 12L, 255L
Complex	Complex numbers	2+3i, 4+3i
Logical	A boolean, True or False	TRUE, FALSE

R has five basic or “atomic” classes of objects: The most basic object is a vector. Empty vectors can be created with the vector() function. The c() function can be used to create vectors of objects.

Structures:

- Vector
- Factor
- List
- Matrix
- Array

- Data Frame
- Tibble

Now, let's create some vectors containing 4 elements.

- vec1 = {-3L,2L,5L,7L}
- vec2 = {apple, samsung, xiaomi, huawei}
- vec3 = {1.2, pi, sqrt(2), -5.5}
- vec4 = {1+3i, 2i, 2+$\sqrt{3}$, 4-3i}
- vec5 = {TRUE, TRUE, FALSE, TRUE}

```r
vec1 <- c(-3L,2L,5L,7L)
print(vec1)
```

```
## [1] -3  2  5  7
```

```r
typeof(vec1)
```

```
## [1] "integer"
```

```r
vec2 <- c("apple", "samsung", "xiaomi", "huawei")
print(vec2)
```

```
## [1] "apple"   "samsung" "xiaomi"  "huawei"
```

```r
typeof(vec2)
```

```
## [1] "character"
```

```r
vec3 <- c(1.2, pi, sqrt(2), -5.5)
print(vec3)
```

```
## [1]  1.200000  3.141593  1.414214 -5.500000
```

```r
typeof(vec3)
```

```
## [1] "double"
```

```r
vec4 <- c(1+3i, 2i, 2+sqrt(3), 4-3i)
print(vec4)
```

```
## [1] 1.000000+3i 0.000000+2i 3.732051+0i 4.000000-3i
```

```r
typeof(vec4)
```

```
## [1] "complex"
```

```r
vec5 <- c(TRUE, TRUE, FALSE, TRUE)
print(vec5)
```

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

```
typeof(vec5)
```

```
## [1] "logical"
```

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

Exercise 5: Operators

R contains a number of operators. They are listed in the table below.

Operator	Description
-	Minus, can be unary or binary
+	Plus, can be unary or binary
!	Unary not
~	Tilde, used for model formulae, can be either unary or binary
?	Help
:	Sequence, binary
	Multiplication, binary
/	Division, binary
ˆ	Exponentiation, binary
%%	Modulus, binary
%/%	Integer divide, binary
%*%	Matrix product, binary
%in%	Matching operator, binary
<	Less than, binary
>	Greater than, binary
==	Equal to, binary
>=	Greater than or equal to, binary
<=	Less than or equal to, binary
&	And, binary, vectorized
&&	And, binary, not vectorized
|	Or, binary, vectorized
||	Or, binary, not vectorized
<-	Left assignment, binary
->	Right assignment, binary
$	List subset, binary

Exercise 5.1: Relational Operators

Let $x = 5,\ y = -3,\ z = 10$, first try to guess what the following statements will return and then check if you are correct;

- $x \leq 5$

- $y > z$
- $z = 2x$
- $yz \geq 0$
- $y \neq z$

```
x <- 5; y <- -3; z <- 10
x <= 5
```

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

```
y > z
```

```
## [1] FALSE
```

```
z == x * 2
```

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

```
y * z > 0
```

```
## [1] FALSE
```

```
y != z
```

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

Exercise 5.2:

Consider a vector x = {-1,1,4,8,-2,pi} and try to guess what the following statements will return and then check if you are correct;

- $x \leq 3$
- $x == -1$
- $any(x) == 8$
- $all(x) \geq 0$
- $x! = x^2$

```
x <- c(-1,1,4,8,-2,pi)
x
```

```
## [1] -1.000000  1.000000  4.000000  8.000000 -2.000000  3.141593
```

```
x <= 3
```

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

```
x == -1
```

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

```
any(x == pi)
```

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

```r
all(x >= 0)
```

```
## [1] FALSE
```

```r
x != x^2
```

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

```r
# Additionally;
TRUE != FALSE
```

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

```r
TRUE < FALSE
```

```
## [1] FALSE
```

```r
TRUE > 0
```

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

```r
FALSE == 0
```

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

Exercise 6: Computing Basic Statistics

Let's assume x is a vector of prime numbers up to 30 and y = x^{-1}. Find the

- Length
- Mean
- Median
- Standard deviation
- Variance of x and y.

Also find correlation and covariance between x and y.

```r
x <- c(2, 3, 5, 7, 11, 13, 17, 19, 23, 29)
y <- x^-1
```

```r
length(x);length(y)
```

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

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

```r
mean(x); mean(y)
```

```
## [1] 12.9
```

```
## [1] 0.1533439
```

```
median(x); median(y)
```

```
## [1] 12
```

```
## [1] 0.08391608
```

```
sd(x); sd(y)
```

```
## [1] 9.024042
```

```
## [1] 0.1527042
```

```
var(x); var(y)
```

```
## [1] 81.43333
```

```
## [1] 0.02331857
```

```
cor(x,y)
```

```
## [1] -0.7886869
```

```
cov(x,y)
```

```
## [1] -1.086818
```