

# Machine learning

Practicals

Κώστας Γιαννάκης

[kostasg@ionio.gr](mailto:kostasg@ionio.gr)

Τμ. Πληροφορικής, Ιόνιο Πανεπιστήμιο

# Εργαστήριο

- Θα χρησιμοποιήσουμε R και RStudio
- Όσοι και όσες χρησιμοποιούν ήδη Python, θα μπορούν να υλοποιούν και να παραδίδουν εργασίες σε Python
- Θα ακολουθούμε τη θεωρία και θα βλέπουμε στην πράξη μεθόδους και αλγόριθμους Μηχανικής Μάθησης

# Φόρμα σχολίων/ανάδρασης/προτάσεων

- ❖ Ανώνυμη φόρμα υποβολής σχολίων/ανάδρασης διδάσκοντα (Κώστα Γιαννάκη)
- ❖ Οποιαδήποτε στιγμή
- ❖ Γράφουμε με σεβασμό και με στόχο να βελτιώσουμε το μάθημα και τον διδάσκοντα
- ❖ <https://forms.office.com/e/908WhJKK5N>



Φόρμα ανώνυμης αξιολόγησης

Μπορείτε να γράψετε ελεύθερα την άποψή σας για το μάθημα, τον διδάσκοντα (Κώστα Γιαννάκη), την ύλη, τα εργαλεία κτλ. Και θετικά και αρνητικά σχόλια είναι καλοδεχούμενα.  
Ευπρόσδεκτα ιδιαιτέρως είναι σχόλια που οδηγούν σε άμεση ενέργεια, πχ "Η ροή του λόγου του διδάσκοντα ήταν πολύ γρήγορη, ίσως λίγο πιο αργά θα ήταν καλύτερα", "Θα προτιμούσα λιγότερα/περισσότερα διαλείμματα", "Θα ήθελα να εξηγήσεις τον αλγόριθμο Α όπως έκανες με τον αλγόριθμο Β", "άλλαξε ομάδα, η ΑΕΚ καταρρέει" κτλ.

Βοηθάτε έτσι εμένα και εν γένει τα μαθήματα να γίνονται καλύτερα :)

When you submit this form, it will not automatically collect your details like name and email address unless you provide it yourself.

Ευχαριστώ για το ενδιαφέρον!

1. Για ποιο μάθημα; (προαιρετικά)

Enter your answer

2. Αξιολόγηση - Σχόλιο - Πρόταση - Αίτημα

Enter your answer

3. Άλλο

Enter your answer

Submit

Never give out your password. [Report abuse](#)

# Απαραίτητες προηγούμενες γνώσεις

- Βασικές έννοιες προγραμματισμού
- Βασικές έννοιες Στατιστικής και Πιθανοτήτων
- Βασικές έννοιες Δομών Δεδομένων
- Προαπαιτούμενο είναι η όρεξη και χρόνος για δουλειά!

# Πλάνο εξαμήνου

- Εισαγωγή στην R και στο RStudio
  - Εγκατάσταση, περιβάλλον, βασικές εντολές, αποθήκευση/φόρτωση αρχείων
- Εισαγωγή στη MM
  - Γρήγορη ανασκόπηση θεωρίας
  - Παραδείγματα MM
- Στατιστικό και μαθηματικό υπόβαθρο
  - αποστάσεις, κατανομές, πιθανότητες
- Προεπεξεργασία δεδομένων
  - Visualization, missing values κ.ο.κ
- Μέθοδοι και αλγόριθμοι MM
  - Supervised
    - classification - knn, dtree, linear regression, etc
  - Unsupervised
    - § clustering kmeans, hierarchical etc
- Μέτρα αξιολόγησης
  - $P_{\chi}$ , RMSE, confusion matrix etc
- Neural networks

# Disclaimer

- LLMs και εργαλεία GPT
  - Απαγορεύεται **ΑΥΣΤΗΡΑ** η χρήση τους για αντιγραφή εργασιών
  - Επιτρέπεται η χρήση τους ως βοηθοί
  - Rule of thumb:
    - Ρωτήστε σαν να απευθύνετε την ερώτηση σε μένα
    - Θα με ρωτούσατε ποτέ "Μπορείς να λύσεις το παρακάτω ερώτημα;"???
  - Καταγράφουμε οποιαδήποτε χρήση των εργαλείων
    - Πχ "Για τη χρήση της συνάρτησης `decision_tree()` από τη βιβλιοθήκη `Tidymodels`, συμβουλευτήκα το εργαλείο GPT-4o mini της OpenAI με χρήση του prompt
      - Τι παραμέτρους μου προσφέρει η συνάρτηση `decision_tree()` από τη βιβλιοθήκη `Tidymodels` στην R;
- Λόγοι να προσέχουμε όταν χρησιμοποιούμε τέτοια εργαλεία
  - Θέματα εκπαιδευτικής διαδικασίας
  - Ευθύνη
  - Πνευματική ιδιοκτησία
  - Hallucinations
  - Περιβαλλοντικό κόστος



# Μηχανική Μάθηση σε 5'

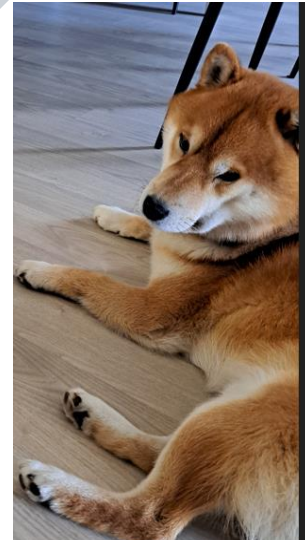
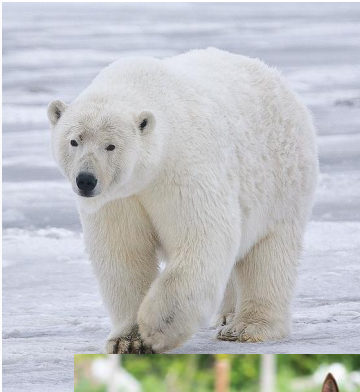
- Χαρακτηριστικά αρκούδας:
  - Μεγάλο μέγεθος
  - Μυώδης
  - ~~☐ Καφέ χρώμα~~ Καφέ ή λευκό χρώμα Καφέ, λευκό, ή ασπρόμαυρο
  - Μεγάλο τρίχωμα
  - Δυνατά σαγόνια
  - ~~☐ Ζουν σε δάση~~ Ζουν σε δάση ή σε πάγο
  - ~~☐ Παμφάγα~~ Παμφάγα ή σαρκοφάγα Παμφάγα ή σαρκοφάγα ή φυτοφάγα
- Υπόθεση: Αν δεν έχει τα παραπάνω χαρακτηριστικά, ΔΕΝ είναι αρκούδα

# Training

# Inference

# Testing

- Μεγάλο μέγεθος
- Μυώδης
- Καφέ, λευκό, ή ασπρόμαυρο
- Μεγάλο τρίχωμα
- Δυνατά σαγόνια
- Ζουν σε δάση ή σε πάγο
- Παμφάγα ή σαρκοφάγα ή φυτοφάγα

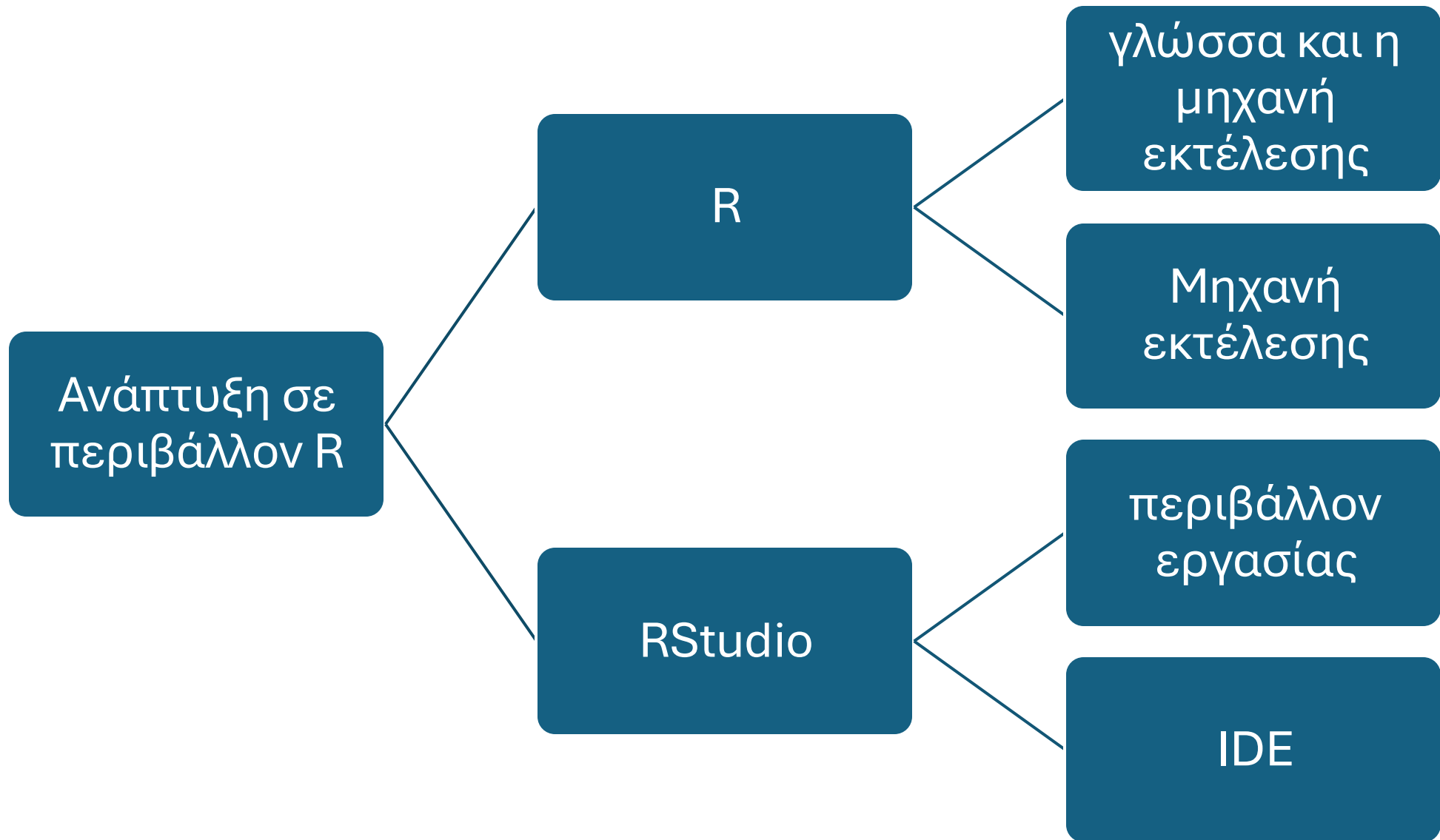


# R και R studio

- Η γλώσσα R είναι μια δημοφιλής, ανοικτού κώδικα γλώσσα προγρ. που χρησιμοποιείται κυρίως για στατιστική ανάλυση, επεξεργασία δεδομένων και γραφήματα (plotting).
- Δημιουργήθηκε τη δεκαετία του 1990
  - Εμπνευσμένη από τη γλώσσα προγρ. S
- Μεγάλη κοινότητα χρηστών και πλούσια υποστήριξη από βιβλιοθήκες και πακέτα
- Απαραίτητη η γνώση της αν θέλετε να ασχοληθείτε με Ανάλυση και Επιστήμη Δεδομένων
- Μεγάλη ευελιξία

# Χαρακτηριστικά R

1. Δυναμική τυποποίηση
2. Ερμηνευόμενη γλώσσα
3. Δομές και τύποι δεδομένων
  - Υποστηρίζει σύνθετες δομές δεδομένων όπως vectors, matrices, data frames, lists, factors, και arrays.
  - Χρήση βιβλιοθηκών για πιο περίπλοκες δομές
4. Λειτουργίες για Στατιστική Ανάλυση
5. Γραφήματα & Οπτικοποίηση
  - Βιβλιοθήκες ggplot2, base graphics
6. Επεκτασιμότητα/Ευελιξία:
  - Πλήθος πακέτων και βιβλιοθηκών διαθέσιμα μέσω του CRAN (Comprehensive R Archive Network) ή από ανεξάρτητους προγραμματιστές
7. \*\*\*Διασύνδεση με άλλες γλώσσες\*\*\*



# Εγκατάσταση

## • Βήμα 1: Κατέβασμα και εγκατάσταση R

- Επισκεφθείτε την ιστοσελίδα <https://posit.co/download/rstudio-desktop/> ή απευθείας τη σελίδα του CRAN: <https://cran.r-project.org/>

- Επιλέξτε το λειτουργικό σύστημα:

- Για Windows:

- Κάντε κλικ στο link "Download R for Windows"
- Επιλέξτε "install R for the first time" ή "base"
- Λήψη αρχείου: R-x.x.x-win.exe (όπου x.x.x είναι η έκδοση)
- Ανοίξτε το αρχείο και ακολουθήστε τα βήματα του οδηγού εγκατάστασης

## • Βήμα 2: Κατέβασμα και εγκατάσταση του RStudio

- Επισκεφθείτε την ιστοσελίδα του RStudio: <https://posit.co/download/rstudio/>
- Επιλέξτε την έκδοση που ταιριάζει στο λειτουργικό σας σύστημα.
- Κατεβάστε το αρχείο εγκατάστασης.
- Ανοίξτε το αρχείο και ακολουθήστε τα βήματα του οδηγού εγκατάστασης.

- Το RStudio θα πρέπει να εντοπίσει αυτόματα την εγκατεστημένη έκδοση του R.

## 1: Install R

RStudio requires R 3.6.0+. Choose a version of R that matches your computer's operating system.

*R is not a Posit product. By clicking on the link below to download and install R, you are leaving the Posit website. Posit disclaims any obligations and all liability with respect to R and the R website.*

DOWNLOAD AND INSTALL R

## 2: Install RStudio

DOWNLOAD RSTUDIO DESKTOP FOR WINDOWS

Size: 287.98 MB | [SHA-256: BCA050AD](#) | Version: 2025.09.1+401 | Released: 2025-10-01

# Συνοπτική εισαγωγή στην R

- Στην πράξη

# Base R

## Cheat Sheet

### Getting Help

Accessing the help files

**?mean**

Get help of a particular function.

**help.search('weighted mean')**

Search the help files for a word or phrase.

**help(package = 'dplyr')**

Find help for a package.

More about an object

**str(iris)**

Get a summary of an object's structure.

**class(iris)**

Find the class an object belongs to.

### Using Libraries

**install.packages('dplyr')**

Download and install a package from CRAN.

**library(dplyr)**

Load the package into the session, making all its functions available to use.

**dplyr::select**

Use a particular function from a package.

**data(iris)**

Load a built-in dataset into the environment.

### Working Directory

**getwd()**

Find the current working directory (where inputs are found and outputs are sent).

**setwd('C://file/path')**

Change the current working directory.

**Use projects in RStudio to set the working directory to the folder you are working in.**

### Vectors

#### Creating Vectors

|                                |                          |                             |
|--------------------------------|--------------------------|-----------------------------|
| <code>c(2, 4, 6)</code>        | <code>2 4 6</code>       | Join elements into a vector |
| <code>2:6</code>               | <code>2 3 4 5 6</code>   | An integer sequence         |
| <code>seq(2, 3, by=0.5)</code> | <code>2.0 2.5 3.0</code> | A complex sequence          |
| <code>rep(1:2, times=3)</code> | <code>1 2 1 2 1 2</code> | Repeat a vector             |
| <code>rep(1:2, each=3)</code>  | <code>1 1 1 2 2 2</code> | Repeat elements of a vector |

#### Vector Functions

|                       |                    |
|-----------------------|--------------------|
| <b>sort(x)</b>        | <b>rev(x)</b>      |
| Return x sorted.      | Return x reversed. |
| <b>table(x)</b>       | <b>unique(x)</b>   |
| See counts of values. | See unique values. |

#### Selecting Vector Elements

##### By Position

|                         |                                  |
|-------------------------|----------------------------------|
| <code>x[4]</code>       | The fourth element.              |
| <code>x[-4]</code>      | All but the fourth.              |
| <code>x[2:4]</code>     | Elements two to four.            |
| <code>x[-(2:4)]</code>  | All elements except two to four. |
| <code>x[c(1, 5)]</code> | Elements one and five.           |

##### By Value

|                                   |                                 |
|-----------------------------------|---------------------------------|
| <code>x[x == 10]</code>           | Elements which are equal to 10. |
| <code>x[x &lt; 0]</code>          | All elements less than zero.    |
| <code>x[x %in% c(1, 2, 5)]</code> | Elements in the set 1, 2, 5.    |

##### Named Vectors

|                         |                            |
|-------------------------|----------------------------|
| <code>x['apple']</code> | Element with name 'apple'. |
|-------------------------|----------------------------|

### Programming

#### For Loop

```
for (variable in sequence){  
  Do something  
}
```

##### Example

```
for (i in 1:4){  
  j <- i + 10  
  print(j)  
}
```

#### While Loop

```
while (condition){  
  Do something  
}
```

##### Example

```
while (i < 5){  
  print(i)  
  i <- i + 1  
}
```

#### If Statements

```
if (condition){  
  Do something  
} else {  
  Do something different  
}
```

##### Example

```
if (i > 3){  
  print('Yes')  
} else {  
  print('No')  
}
```

#### Functions

```
function_name <- function(var){  
  Do something  
  return(new_variable)  
}
```

##### Example

```
square <- function(x){  
  squared <- x*x  
  return(squared)  
}
```

### Reading and Writing Data

| Input                                        | Output                                     | Description                                                                                    |
|----------------------------------------------|--------------------------------------------|------------------------------------------------------------------------------------------------|
| <code>df &lt;- read.table('file.txt')</code> | <code>write.table(df, 'file.txt')</code>   | Read and write a delimited text file.                                                          |
| <code>df &lt;- read.csv('file.csv')</code>   | <code>write.csv(df, 'file.csv')</code>     | Read and write a comma separated value file. This is a special case of read.table/write.table. |
| <code>load('file.RData')</code>              | <code>save(df, file = 'file.Rdata')</code> | Read and write an R data file, a file type special for R.                                      |

#### Conditions

|                     |           |                       |              |                        |                          |                         |            |
|---------------------|-----------|-----------------------|--------------|------------------------|--------------------------|-------------------------|------------|
| <code>a == b</code> | Are equal | <code>a &gt; b</code> | Greater than | <code>a &gt;= b</code> | Greater than or equal to | <code>is.na(a)</code>   | Is missing |
| <code>a != b</code> | Not equal | <code>a &lt; b</code> | Less than    | <code>a &lt;= b</code> | Less than or equal to    | <code>is.null(a)</code> | Is null    |

Τσεκάρετε το μάθημα στο opencourses!

# RStudio IDE :: CHEAT SHEET



## Documents and Apps

Check spelling, Render output, Choose output format, Configure render options, Insert code chunk, Publish to server

Jump to previous chunk, Jump to next chunk, Run this and all previous code chunks, Show file outline, Visual Editor (reverse side), Run this code chunk, Set knitr chunk options

Access markdown guide at **Help > Markdown Quick Reference**  
See reverse side for more on **Visual Editor**

RStudio recognizes that files named **app.R**, **server.R**, **ui.R**, and **global.R** belong to a shiny app

Run app, Choose location to shinyapps.io, Publish app, Manage shinyapps.io account

## Source Editor

Navigate backwards/forwards, Open in new window, Save, Find and replace, Compile as notebook, Run selected code

Re-run previous code, Source with or w/out echo or as a Local Job, Show file outline as a Local Job

Multiple cursors/column selection with **Alt + mouse drag**, Code diagnostics that appear in the margin. Hover over diagnostic symbols for details, Syntax highlighting based on your file's extension, Tab completion to finish function names, file paths, arguments, and more.

Multi-language code snippets to quickly use common blocks of code, Jump to function in file, Change file type

Working Directory, Run scripts in separate sessions, Maximize, minimize panes, Ctrl/Cmd + arrow up to see history, Markdown Build Log, Drag pane boundaries

Path to displayed directory, Click on file or directory name to open

## Tab Panes

Import data with wizard, History of past commands to run/copy, Manage external databases, View memory usage, R tutorials

Load workspace, Save workspace, Clear R workspace, Search inside environment, Choose environment to display from list of parent environments, Display objects as list or grid

Displays saved objects by type with short description, View in data viewer, View function source code

Create folder, Delete file, Rename file, Change directory

## Version Control

Turn on at **Tools > Project Options > Git/SVN**

Added, Deleted, Modified, Renamed, Untracked

Stage files, Commit staged files, Push/Pull, View History, Current branch, Open shell to type commands

Show file diff to view file differences

## Debug Mode

Use **debug()**, **browser()**, or a breakpoint and execute your code to open the debugger mode.

Launch debugger mode from origin, Open traceback to examine the functions that R called before the error occurred

Console, Terminal, Jobs

Error, Show Traceback, Return with Debug

Τσεκάρετε το μάθημα στο opencourses!

## Package Development

Create a new package with **File > New Project > New Directory > R Package**  
Enable roxygen documentation with **Tools > Project Options > Build Tools**

Roxygen guide at **Help > Roxygen Quick Reference**  
See package information in the **Build Tab**

Install package and restart R, Run devtools::load\_all() and reload changes, Run R CMD check, Customize package build options, Clear output and rebuild, Run package tests

RStudio opens plots in a dedicated **Plots** pane

Navigate recent plots, Open in window, Export plot, Delete plot, Delete all plots

GUI Package manager lists every installed package

Install, Update, Browse package site, Click to load package with library(). Unlick to detach package with detach(), Package version installed, Delete from library

RStudio opens documentation in a dedicated **Help** pane

Home page of helpful links, Search within help file, Search for help file

Viewer pane displays HTML content, such as Shiny apps, RMarkdown reports, and interactive visualizations

Stop Shiny app, Publish to shinyapps.io, rpubs, RStudio Connect, Refresh

View(<data>) opens spreadsheet like view of data set

Filter rows by value or value range, Sort by values, Search for value

Click next to line number to add/remove a breakpoint. Highlighted line shows where execution has paused

Run commands in environment where execution has paused, Examine variables in executing environment, Select function in traceback to debug

Console, Terminal, Jobs

Next, Continue, Stop

Step through code one line at a time, Step into and out of functions to run, Resume execution, Quit debug mode

# Σήμερα - R crash course

- Working directory, περιβάλλον RStudio
  - File -> New Script
- Μεταβλητές και βασικές πράξεις
- Εκτύπωση μεταβλητών στο terminal
- For loops, if statements
- Δημιουργία vector και βασικές πράξεις
- Δημιουργία matrix
- Δημιουργία dataframe
- Βασικές πράξεις και μετασχηματισμοί dataframe
  - colnames, rownames, rbind, cbind, data types, table
- Βοήθεια στο R
- Τυχαίο δείγμα από Κανονική Κατανομή (συμβολίζεται ως  $N(\mu, \sigma^2)$ , όπου  $\mu$  η μέση τιμή και  $\sigma^2$  η διακύμανση)
- Summary and descriptive statistics
- Οπτικοποίηση και γραφήματα (plotting)
- Διάβασμα αρχείου
- Εξαγωγή αρχείου
- Αποθήκευση (όχι εξαγωγή!) μεταβλητών
- Seed (ή αλλιώς πώς 'ελέγχουμε' την τυχαιότητα)

# Καλές πρακτικές κώδικα

- Καλός και σαφής σχολιασμός
- Ευανάγνωστος κώδικας
  - `px vect = rnorm(100, 5, 1)` είναι προτιμότερο του `vect=rnorm(100,5,1)`
  - Εσοχές και καλή στοίχιση όπου πρέπει να εμφωλεύεται τμήμα του κώδικα
    - Αντίθετα από την Python, αυτό δεν επιφέρει συντακτικό λάθος. Είναι απλά καλή πρακτική

# Ξεκινήστε με νέο script

(δείτε το συνοδευτικό pdf στο [opencourses](#))

```
# working directory

getwd()

setwd("C:/Users/kgi062/ML_2025")

print("Hello, world!")

2 + 2

x <- 10      # Numeric

name <- "Alice" # Character

flag <- TRUE  # Logical

numbers <- c(1, 2, 3, 4, 5)

mean(numbers)

sum(numbers)

?CO2

View(CO2)

head(CO2)

tail(CO2)

colnames(CO2)

dim(CO2)

str(CO2)
```

```
summary(CO2)

summary(CO2$uptake)

median(CO2$uptake)

class(CO2$uptake)

unique(CO2$Treatment)

subset(CO2, conc == min(CO2$conc))

hist(CO2$uptake)

plot(Orange$age, Orange$circumference)

plot(Orange$age, Orange$circumference,

      xlab="Age", ylab="Circumference",

      main="Circumference vs. Age",

      col="blue", pch=16)


# data come from an unknown normal distribution

data <- read.csv("external_file.csv")

d = rnorm(100, mean = 5, sd=1)

plot(data)

plot(d)
```

# Άσκηση

1. Δημιουργήστε ένα νέο script και αποθηκεύστε το ως `askisi1.R`
  - Σημαντικό: Αν χρησιμοποιείτε Η/Υ του εργαστηρίου, θυμηθείτε να κρατήσετε αντίγραφο του αρχείου (email στον εαυτό σας, usb, dropbox, onedrive, κοκ)
2. Να κατασκευάσετε ένα vector με όνομα `vector10` που να περιέχει 10 τυχαία δείγματα (τ.δ.) από κατανομή  $N(10, 1)$ .
  - Αντίστοιχα, ένα vector με όνομα `vector50` με 50 τ.δ. από  $N(10, 1)$
  - Και ένα ακόμα vector με όνομα `vector100` με 100 τ.δ. από  $N(10, 1)$
3. Υπολογίστε και τυπώστε στην οθόνη βασικά περιγραφικά στατιστικά του κάθε vector
4. Για κάθε vector δημιουργήστε ένα scatterplot και ένα histogram. Αποθηκεύστε τα με ενδεικτικά ονόματα
5. Κοιτάζοντας τα περιγραφικά στατιστικά και ειδικότερα τα παραπάνω plots, τι παρατηρείτε?
  - Γράψτε σε μία μικρή παράγραφο με 1-2 προτάσεις τα ευρήματά σας. Για πληρότητα, εάν ήταν παραδοταία εργασία θα έπρεπε να προσθέσετε και τα παραγώμενα plots
6. Κατεβάστε από το Opencourses το αρχείο `check.csv` και διαβάστε το στο πρόγραμμά σας σε μια μεταβλητή με όνομα `external_file`.
7. Ακολουθήστε τα βήματα που κάνατε για τα `vector10`, `vector50`, και `vector100`, αυτή τη φορά για το `external_file`.
8. Τι παρατηρείτε (διαφορές, ομοιότητες,...);
  - Συμπληρώστε τη μικρή αναφορά σας από το βήμα 5 με τα νέα ευρήματα