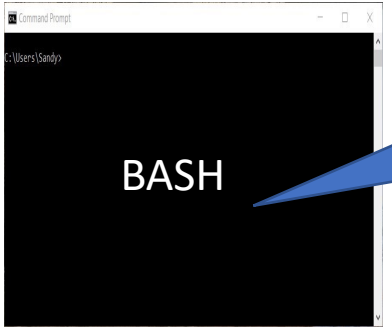
Module 10: Introduction to R

Learning Outcomes

- Identify the different components of RStudio
- Understand differences between variable/object types and data structures in R
- Perform basic operations using variables and objects
- Be able to recognize and use functions in R
- Load data in R and inspect that it is loaded properly

The languages of computers



BASH

Communicate with the operating system

DATA SCIENCE!

HTML

To build websites

Software development

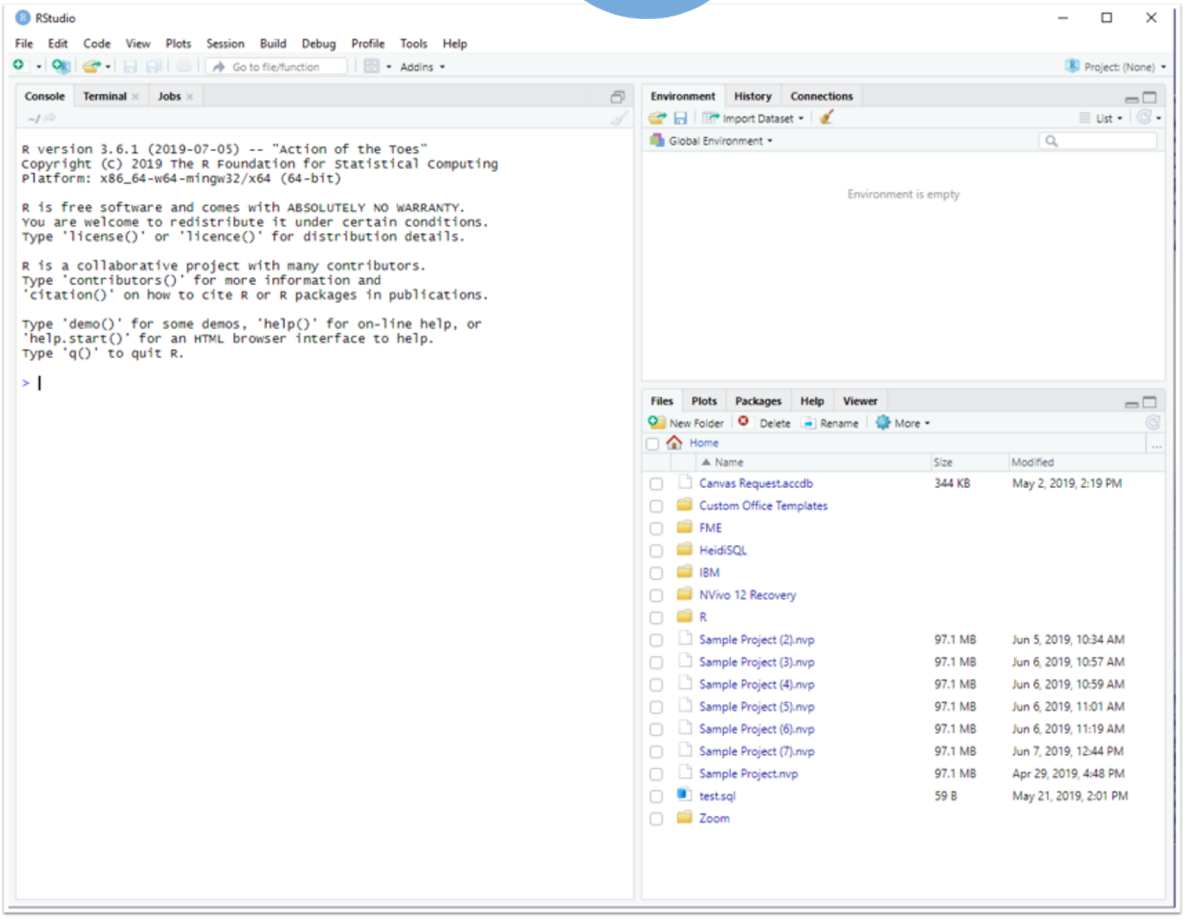
R programming language

- Common bioinformatic and statistical software for biologists
- Strengths are data mining/wrangling abilities and data visualization
- Open-source; lots of community support

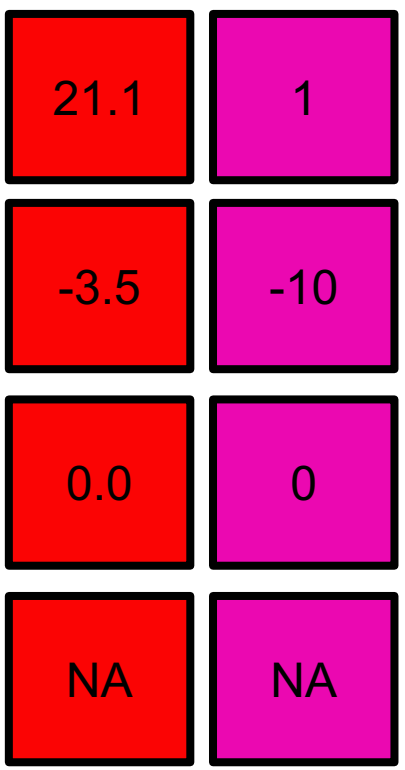
https://favtutor.com/blogs/data-wrangling

RStudio

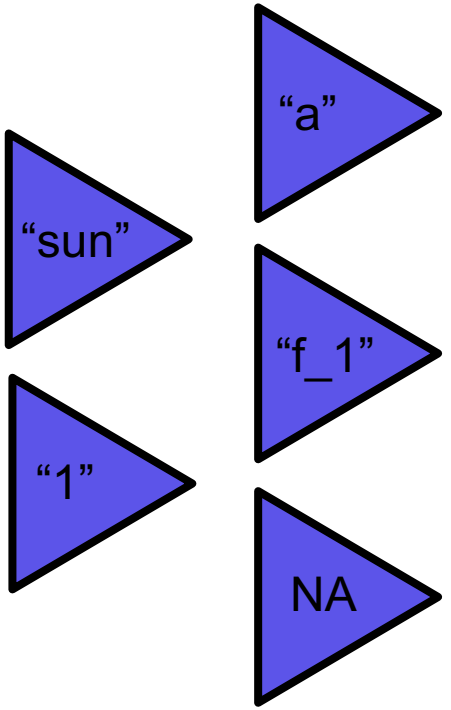
- GUI (graphical user interface) software for R programming language
- Integrates writing code, running code, and producing figures



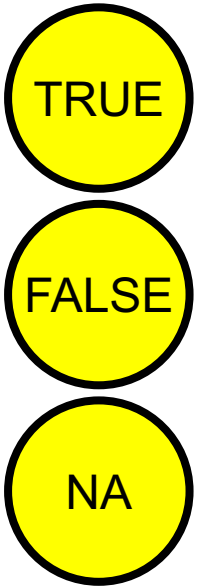
R object types (“atomic classes”)



Numeric/double (integers)

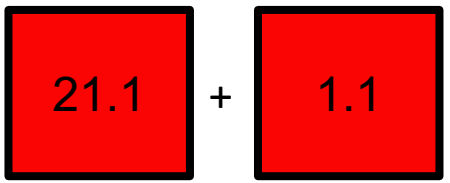


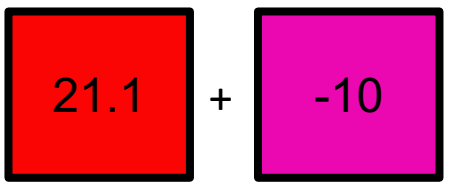
Character

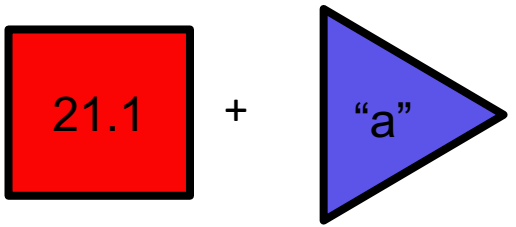


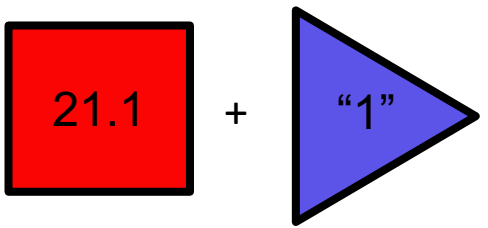
Logical (True/False)

Object types are treated differently

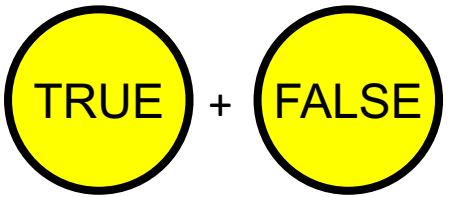


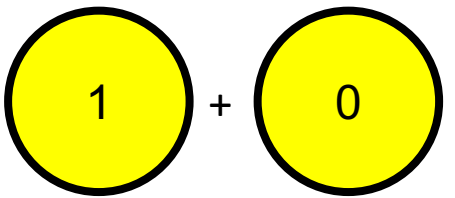




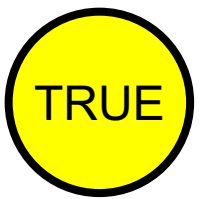


Can’t work





if (



}

Mathematical operators:
+ - * /

R functions

- Functions are packages of code that perform a specific task

output <- function(input)

e.g. mean(), sum(), min(), max()

?function for help about the function

Manipulating type/class

as.character()

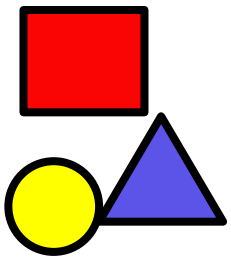
as.numeric()

as.integer()

as.logical()

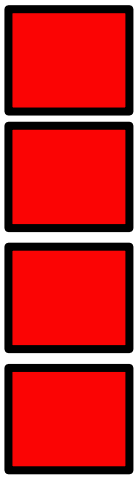
- What type do I have?
  - typeof()
  - class()

R data structures



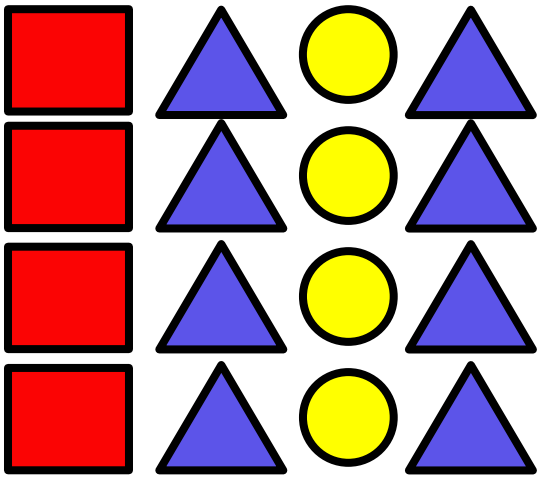
variables

vector()



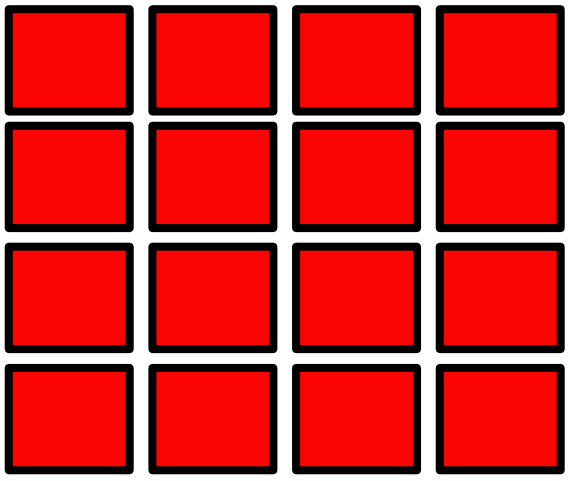
Collection of one variable type

data.frame()



Columns of different vector types

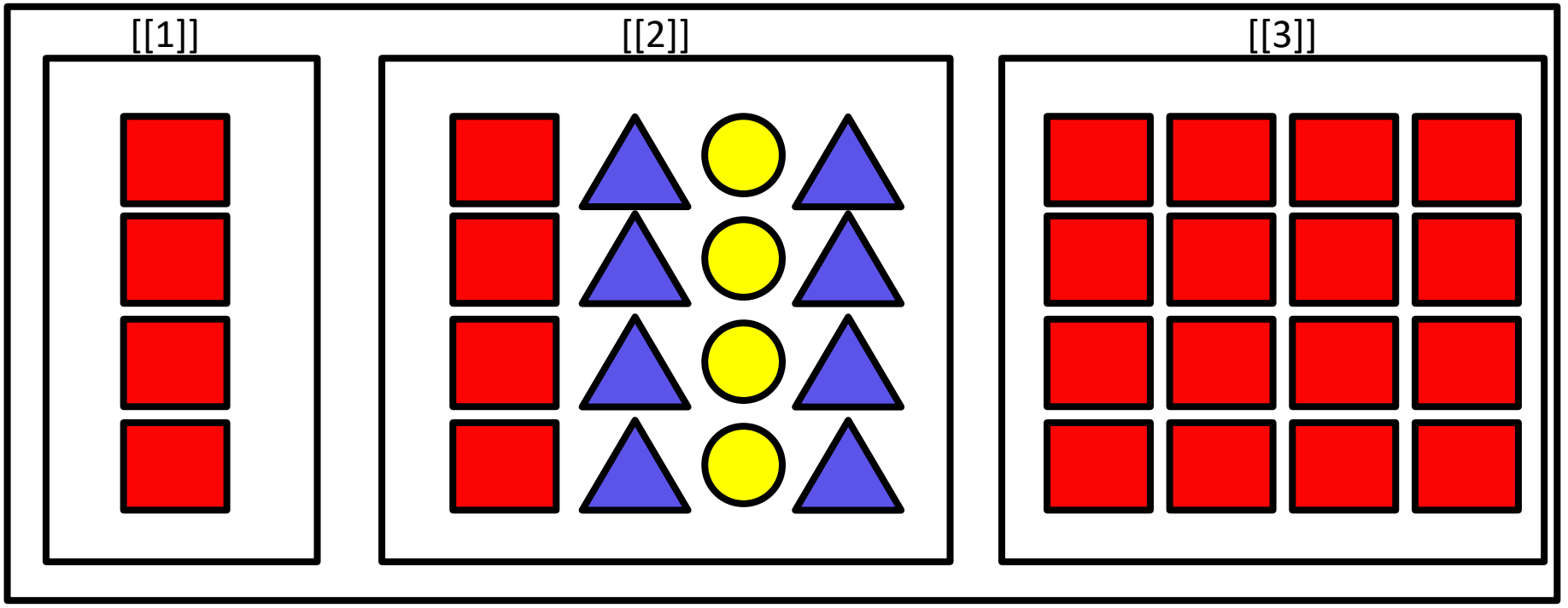
matrix()



Columns of the same vector types

R data structures

list()



Collection of multiple data structure types

R data structures

Data structures are ways to group individual variables:

- Vector `c()` -> collection of one variable type
- Factor `factor()` -> used for organizing vector objects
- List `list()` -> bins different vector types
- Matrix `matrix()` -> array of one vector type
- Data frame `data.frame()` -> multiple vector types

For loops, if/else, while loops

What are humans good at?

- Pattern recognition
- Making decisions based on complex inputs

- We are bad at:
  - Doing repetitive/intense tasks accurately

What are computers good at?

- Doing repetitive/intense tasks accurately

For loops

- Iterate through a vector and performs the same calculations each time

for ( in c()) { #DO THIS }

For loops

- Iterate through a vector and performs the same calculations each time

for ( in c()) {

#DO THIS

}

if/else statements

if (#CONDITION) { #DO THIS }

if/else statements

if (#CONDITION) { #DO THIS }

if (#CONDITION) { #DOTHIS } else { #DOTHISINSTEAD }

if/else statements

```
if (  #CONDITION   ) {
        #DO THIS
} else if (  #CONDITION   ) {
        #DO THIS INSTEAD
} else {
        #DO THIS FOR ALL OTHER THINGS
}
```

While loops

```
# SET CONDITION
while (  #CONDITION   ) {
        # DO THIS
        # UPDATE CONDITION
}
```

Loops, if/else, while

- Loops iterate through a vector
  - for (x in vector) { }
- if/else statements create decisions based on conditions
  - if (TRUE) {} else {}
  - If (TRUE) {} else if (TRUE) {} else {}
- While loops continue looping until a condition is met
  - while (TRUE) {}

Loading and viewing data

- read.csv() or read.delim()

- To see object:
  - Call it directly
  - Str()
  - Click on item in environment box