Data Types and Structures

Different objects in R

Download the section 3 .Rmd handout to `STAT240/lecture/sect03-data-types`.

Material in this section is covered by Chapters 2, 3 and 4 on the notes website.

Variables

- Used to refer to an object
- Created with <- or =
- Capitalization matters!

Best practice is to use the underscore _ in long variable names.

R has lots of variable types. Basic types:

- Numeric
- Character
- Logical

Each has different uses and characterisitics.

Numeric objects are numbers.

- Can be integers or decimals
- Used with operators `+`, `-`, `*`, etc.
- R has some special numbers built-in

View an object's type with `class`.

Character objects, also called "strings", are interpreted as letters.

- Indicated by quotation marks `"like this"`
- The contents of a string do not have a special meaning

Logical objects

- Can be `TRUE` or `FALSE`
- Also called "booleans"
- There's a secret third one, `NA`

Usually a result of a logical operation

We've seen basic math operators like + and −.

There are also **logical operators**.

- Use `<`, `>`, `<=`, `>=` to compare numbers
- Use `==` and `!=` to check equality

How is = different from ==?

The operators "and" and "or" are `&` and `|`.

- `&` returns `TRUE` if **both** sides are `TRUE`
- `|` returns `TRUE` if **either or both** sides are `TRUE`

`NA` refers to an empty or “missing” space.

- Different from 0 and different from `NaN`.
- Doesn’t work with arithmetic or logical operators

Check missing values with `is.na`.

Functions, or commands, are a set of instructions for R.

- You provide one or more inputs (arguments)
- R follows the instructions and gives an output

Have the form

```
function_name(arg1, arg2, ...)
```

We've seen a few functions so far, like `sqrt` and `class`. We can also design our own functions.

When using a function, the arguments must be of an appropriate type.

Also, you can save the output of a function.

A **vector** is an ordered sequence of objects

- Can be numeric, character, or logical
- All items must be the same type
- Vectors are the building blocks for **dataframes**.

Initialize a vector with the `c()` (combine) function. There are shortcuts to making numeric sequences.

You can get the item at a particular index with `[]`.

R does **elementwise** operations with vectors.

- An operation is performed separately on every element
- Be careful with vectors of different sizes!

Try stuff out!

- Find the sum of all the odd numbers between 1 and 99.
- Evaluate $[(\frac{1}{2}x - 8)^2 - 20]$ for all of the integers between 0 and 50. For what values of x is the function negative?

A **dataframe** is an object that stores multiple vectors as columns.

- Represents a set of data
- Row = item in sample, column = variable measured on item

There are some dataframes built in to R.

We can initialize a dataframe with `tibble`.

- Give any number of columns as input
- Columns should have the same length

Let's make a datafame of the alphabet.

We use the syntax `df[r, c]` to get the value at row r and column c.

- `df[r, ]` to get a specific row
- `df[ , c]` to get a specific column

Get a row by its name with `$`.

Now we have the tools to work with real data!

- Run `read_csv` from `tidyverse` to get a dataset on recent volcano eruptions.
- The data should be saved in a dataframe object called `eruptions_recent`.

We can use `View(eruptions_recent)` to get a look at the data.

Test out the following functions:

- `View`
- `head`
- `glimpse`
- `colnames`
- `dim`
- `nrow`
- `ncol`

What does each command do? Which ones do you prefer to use to explore the dataframe?