02. Toolkit — Part 2

Data Science for Economists — Summer 2025

Julian Hinz

Bielefeld University

VSCODE WORKING?

Session Roadmap

- R Basics
- tidyverse + data.table and actual data

R BASICS

Basics

- a great calculator
- logic, negation, evaluation (`==`) matching (`%in%`)
  - → careful: floating-point numbers
  - → better: `all.equal()`
- assignment with with `=` or `<-`
- Questions? `help(plot)` or `?plot`
- Commenting with #

Objects

- vectors
- matrices
- data frames (and derivatives like `data.table` and `tibble`
- lists
- functions
- etc.

Conversion between objects

```
> # Create a small data frame called "d"
> d = data.frame(x = 1:2, y = 3:4)
> d
  x y
1 1 3
2 2 4

> # Convert it to (i.e. create) a matrix call "m"
> m = as.matrix(d)
> m
     x y
[1,] 1 3
[2,] 2 4
```

Class, type and structure

```
> # Evaluate its class
> class(d)
[1] "data.frame"

> # Evaluate its type
> typeof(d)
[1] "list"

> # Show its structure
> str(d)
'data.frame':	2 obs. of  2 variables:
 $ x: int  1 2
 $ y: int  3 4
```

Global environment

```
> # View d
> View(d)
> d
  x y
1 1 3
2 2 4

> # Evaluate its type
> typeof(d)
[1] "list"

> # Use d to run command
> lm(y ~ x)
Error in eval(predvars, data, env) : object 'y' not found

> lm(y ~ x, data = d)
```

Reserved words

- Fundamental commands, operators and relations cannot be reassigned

```
→
if
else
while
function
for
TRUE
FALSE
NULL
Inf
NaN
NA
```

Semi-reserved words

```
> my_vector = c(1, 2, 5)
> my_vector
[1] 1 2 5

> c = 4
> c(1, 2 ,5)
[1] 1 2 5
> c
[1] 4

> pi
[1] 3.141593

> pi = 2
> pi
[1] 2
```

Indexing

- Option 1: `[]`

```
> a = 1:10
> a[4]
[1] 4
> a[c(4, 6)]
[1] 4 6

> m[1,1]
x
1

> my_list = list(a = "hello", b = c(1,2,3), c = data.frame(x = 1:5, y = 6:10)
> my_list[[1]]
[1] "hello"
> my_list[[2]][3]
[1] 3
```

Indexing

- Option 2: `$`

```
> my_list
$a
[1] "hello"

$b
[1] 1 2 3

$c
  x  y
1 1  6
2 2  7
3 3  8
4 4  9
5 5 10
```

Indexing

- Option 2: `$`

```
> my_list$a
[1] "hello"

> my_list$b[3]
[1] 3

> my_list$c$x
[1] 1 2 3 4 5
```

Indexing

- Option 2: `$`

```
> # Remember the earlier problem?
> lm(d$y ~ d$x)

Call:
lm(formula = d$y ~ d$x)

Coefficients:
(Intercept)          d$x
          2            1
```

functions

- A lot of functionality in “base R”
  → in-built functions, like `lm( )`
- User-built functions easy to implement

```
> example_function = function (a, b) {
+ output = a + b
+ return (output)
+ }
> example_function(1,2)
[1] 3
```

libraries

- Community-built (set of) functions: libraries or packages

```
> library(data.table)
data.table 1.14.2 using 2 threads (see ?getDTthreads).
Latest news: r-datatable.com

Attaching package: ‘data.table’

The following objects are masked from ‘package:dplyr’:

    between, first, last

The following object is masked from ‘package:purrr’:

    transpose
```

WRAP UP

- So far: First R experience
- Next session: Web scraping and APIs

02. Toolkit — Part 2

Data Science for Economists — Summer 2025

Julian Hinz

Bielefeld University