Introduction to R*

Lecture 4: Heterogeneous vectors (Lists & Dataframes) and IO

Wim R.M. Cardoen

Last updated: 10/17/2022 @ 13:10:23

Contents

In the **first** part of this section, heterogeneous vectors will be discussed i.e. the following two kinds:[1]:

- lists
- data frames & tibbles

Input-output (IO) in `R` forms the subject of the latter part.

1 R Lists

A `list` is a heterogeneous vector that **may** contain one or more **components**.
The components can be **heterogeneous** objects (atomic types, functions, lists[2], ...).

Under the hood, he list is implemented as a vector of pointers to its top-level components.
Therefore, the list's length equals the number of top-level components.

1.1 Creation of a list

1.1.1 Examples

1.2 Accessing elements

1.2.1 Examples

1.3 Modifying lists

- modifying elements
- inserting elements
- deleting elements
- concatenating lists

1.3.1 Examples

1.4 Notes

1.4.1 Adding a return a list

1.4.2 More on [[]], vs. []

1.5 Exercises

[1] `R` also has the pairlist. This topic will not be discussed in this section. People interested in this subject, should have a look at R-internals.

[2] Due to this feature they also called `recursive` vectors.

2 R Dataframes

A `$date frame` is a `list` with three `attributes`:

- **names** : component names
- **row.names** : row names
- **class**: **data.frame**

From the above we can infer that the number of rows is the **same** for each component. The components of a data frame are either vectors, factors,numerical matrices, lists or other data frames.

2.1 Examples

2.2 Creating a data frame

- read.table
- data.frame

2.3 attach and detach

3 Input-Output (IO)