MRC Clinical Trials Unit

Smarter Studies
Global Impact
Better Health

UCL

Of NA, NaN, NULL & friends

Matteo Quartagno

MRC Clinical Trials Unit at UCL

31st January 2019

Missing data and R

- R has three different “missing data” values, unlike most other programming languages.
  1. NA: Not Available, general missing data.
     - Data missing for technical issues, privacy, lack of attention… Example: missing salary in a survey, because the interviewee did not respond.
  2. NaN: Not a Number, result of impossible operation.
     - The result of an impossible operation. Ex: 0/0.
  3. NULL: A non-existing object.
     - Something that does not exist even “behind the curtains”. E.g. missing salary because we are using same questionnaire as before on a sample of unemployed people.

NA: Not available

- NA: Not Available. It is the most standard way of defining missing values in R.
  - Similar to ‘.’ in Stata.
  - It is a logical constant, not a specific number like in SAS or Stata.
  - Function is.na to check whether a value is missing.
  - There are 4 different types of NA, one for each of integer, numeric, character and complex.
  - Generic NA automatically coerces to right data type.

NA: things to remember

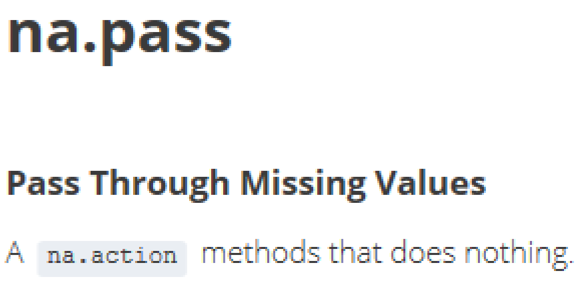
- NA is different from “NA”, which is a string with characters NA.
- anyNA similar to any(is.na());
- Is.na(vec) <- i introduces missing data in the i-th element of vector vec;
- NA are slighlty different coded in factors as <NA>;
- addNA makes NA an additional level.

NA: applying functions

- Standard functions, like sum and mean, do not work if there are any NAs;
- Need to add option na.rm=T;

NA.action: regression models

- In regression models 4 different types of na.action, i.e. 4 different ways of dealing with missing data:
  1. NA.omit: discards incomplete units, default in lm;
  2. NA.fail: model not fitted, error message, default in gls;
  3. NA.exclude: similar to NA.omit, but incomplete records are still predicted as NA.
  4. NA.pass:



na.pass

Pass Through Missing Values

A `na.action` methods that does nothing.

NaN: Not a Number

- NaN: the result of an impossible operation:
  - 0/0, sin(Inf), Inf-Inf, etc etc
- Similar to NA: it is not a number (well, of course!) so need to use is.nan and can't compare it with a number.
- Similar to NA also in the way it is used in functions/regression models.
- However, is.na(NaN)=T, is.nan(NA)=F (!!!)

NULL: null object

- NULL: an undefined value. Usually returned as output (or passed as input) of a function;
    - Very different from NA: it has its own class;
    - Excluded from vectors, but not from lists;
    - Can use is.null similarly to is.na;
    - Different behavior when applying sum or mean to NULL;
    - Different from integer(0), numeric(0), etc etc, for example cannot set an attribute to a NULL object.

Differences between NA and NULL

- Vectors:
  - if we select a non-existing element, we get NA;
  - NA is regarded as an element, NULL is ignored;
  - Not possible to set one particular element to NULL.
- Lists:
  - If we select a non-existing element, we get NULL;
  - Both NA and NULL are regarded as list elements;
  - If we chenge one element of the list to NULL, we delete it.
- Data frames:
  - Very different ways of treating a NULL data frame or one with NA

Tabulate missing data

- A very useful function is complete.cases:
  - If used on a matrix or dataframe gives a logical vector stating whether an observation has any NA;
  - If used on a vector, equivalent to is.na;
  - If used on a list, be careful (see R script).
- Several packages with utility functions. I find particularly useful mice function md.pattern, which gives frequencies of each missing data pattern.

Reserved words

- NA, NaN and NULL are all reserved words in R;
- Cannot be overwritten;
- However, R is case-sensitive.
- In conclusion:

```
> na<-"NA"
> is.na(na) [1] FALSE
```