DATA HANDLING AND DATA MINING

UNIVERSITÄT
LEIPZIG

Erik Kusch

erik.kusch@uni-leipzig.de

Behavioural Ecology Research Group
University of Leipzig

30/04/2019

Why Care?

Biostatisticians often use 70% of their time to handle data and just 30% to actually analyse it.

Why care?

- Proper data collection and data handling ensure accurate results
- Proper data collection cuts down on data handling time
- Proper data handling will make reproducing an analysis much easier

What to consider?

- Which data format to use
- What kind of data to record
- How data values are recorded/stored
- What kind of data values are feasible

Recording Data

Guidelines for data recording:

- When collecting categorical data, know what values the variables are allowed to take
- When collecting continuous data, know which range the variable values can fall into
- Make sure everyone involved in data collection is on the same page
- Make regular back-ups of your data set

Recording Data Collection - The README File

Documenting data recording is just as important as proper data collection!

To do so, one usually uses a **README** file containing the following:

- Project Name and Summary
- Primary contact information
- Your name and title (if you aren't the primary contact)
- Other people working on the project
- Location of data and supporting info
- Organization and naming conventions used for the data
- Any previous work on the project and where its located
- Funding information

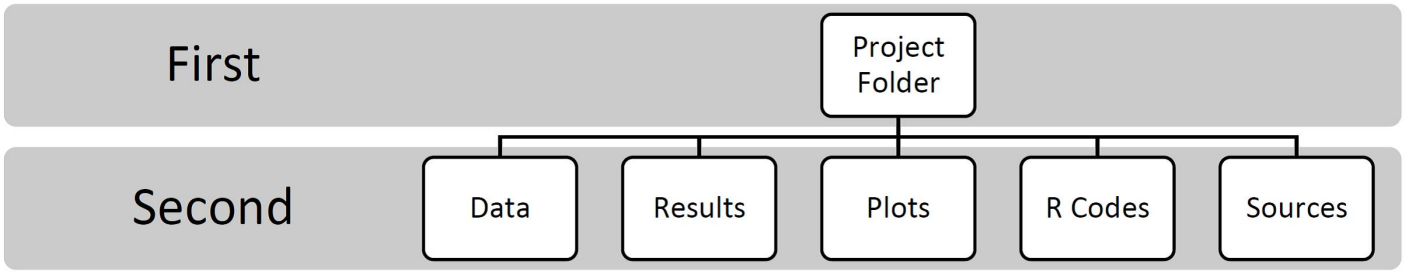
This file is always **saved in conjunction with the actual data set**!

Data Structure

I recommend a structure like the one shown below with at least two hierarchy levels.

The only files allowed in your first hierarchy level are:

- R master file
- Manuscript master file



Additionally, make sure to **back-up your project folder frequently** and use **version control** on it.

Which Format To Use

When storing your data, you have a plethora of file formats to choose from.

R works very well with:

- excel files (.xls, .xlsx, .csv)
- text files (.txt)

Whilst both of these are accessible to everyone of your co-workers, excel is easier to operate outside of R.

→ Make sure to provide co-workers with a master file before data collection to avoid cell formatting issues on different computers

Common Issues

The Decimals

Always use a *dot* to indicate decimals.

$\rightarrow$ It is the standard in science.

To `NA` Or Not To `NA`?

Never enter `NA` *values manually* into your data.

$\rightarrow$ They cause problems in `R`.

Redundancy Or Sparsity?

Don't clutter data with unnecessary data records.

$\rightarrow$ Reduces storage space and chances for errors.

Workflow I

No data set is ever perfect (except fabricated ones)!

The etiquette of fixing data:

- **Never overwrite/alter** your original data file
- **Never** apply fixes by hand (you completely break the process of reproducibility by doing so)

`R` is a beyond powerful tool for fixing your data!

Workflow II

Fixing an data set is usually a **two-step** process:

Column/Variable Class

- Variable record classes are paramount to get right for specific analyses in R.
- Before data recording is done, we should already have a desired variable record class for each variable's records.

Column/Variable Content/Values

- Typos and the like can often lead to false data/variable records and need to be fixed or removed for dependable results to be obtained.

Since records are usually stored in one column for each variable, we may wish to asses column classes as "variable record classes".

Useful Functions in R

- `dim(object)` returns the dimensions of the object
- `summary(object)` gives you a summary of values contained within the object (see seminar 4)
- `View(object)` opens almost any R object in a new tab within R for visual inspection
- `which(object == value)` returns a vector of `TRUE` and `FALSE` values according to the statement in brackets
- `sum(object)` returns the sum an objects values (also works for `TRUE`/`FALSE` values)
- `vector[position]` subsets elements of a `vector`
- `data.frame[Row,Column]` subsets elements of a `data.frame`

The README File Revisited

Using the **README file**, one can identify what information is contained within the data set and thus decide:

- What type/class a data record should be of
- Which variables may be redundant
- Which data records exceed their variable-specific feasible thresholds
- Where to get comparative data sets from

Data Mining should then focus on:

- *Identifying problems* within the data records
- *Explorative* data analyses

Numbers or Visualisations?

For data mining, one may wish to enlist the use of methods contained in seminar 4 & 5 (Descriptive Statistics & Data Visualisation):

Descriptive Statistics:

As far as descriptive statistics go, the `summary()` command in R is probably the most useful tool for data mining.

Data Visualisations:

- Histograms
- Scatter plots

Holistic data mining is best **achieved using a combination** of data visualisations tools and parameters of descriptive statistics!

What To Do I

Data Loading

- The data is located at https://github.com/ErikKusch/An-Introduction-to-Biostatistics-Using-R
- The file is called SparrowData.xls (also available as SparrowData.rar)

Expectations

- Browse the README file to get a feeling for the variables contained within our data set.
- Write down your expectations of data range and variable mode within R.

Inspection

- Use common functions to get a feeling for the data set.
- Can you spot some data errors right away?
- Do **not** do this in Excel.

What To Do II

Column/Variable Class

- Find columns whose record classes do not match up with your educated expectation
- Transform these column classes

Column/Variable Content/Values

- Identify columns which, after fixing their record class, still exhibit faulty/unreasonable values.
- Decide what to do about them.

Redundancy

- Remove columns from the data set which are not needed because their information is contained in another column or are unnecessarily bloating the data set.

$\rightarrow$ Do this **variable by variable**.