Current status and propsects of R-packages for the design of experiments

by Emi Tanaka

Abstract Re-running an experiment is generally costly and in some cases impossible due to limited resources, so the design of experiment plays a critical role in increasing the quality of experimental data. In this article I describe the current state of the R-packages for the design of experiments through textual analysis and download trends. I discuss also the software design of widely utilised R packages in the field of experimental design and conclude with discussion of some future prospects for the field.
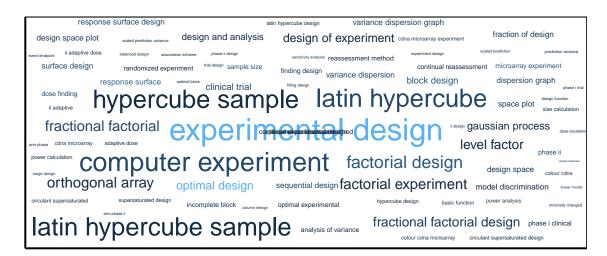
Introduction

The critical role of data collection is well captured in the expression "garbage in, garbage out" – in other words, if the collected data is rubbish then no analysis, however complex it may be, can make something out of it. Methods for data collection can be dichotomised by the type of data collected – namely, experimental or observational – or alternatively, categorised as experimental design (including quasi-experimental design) or survey design. This dichotomisation, to a great extent, is seen in CRAN task views where R-packages in experimental design are in *ExperimentalDesign* and R-packages in survey designs are in *OfficialStatistics*. Survey designs often, although not always, aim to collect observational data whilst experimental designs exclusively center on experimental data. This paper is concerned with the latter.
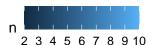
In the CRAN task view of *ExperimentalDesign*, there are 110 R packages for experimental design and analysis of data from experiments, henceforth referred to as "DoE packages" in this paper. The sheer quantity and variation in the output experimental design in the R-packages are arguably unmatched with any other programming languages, e.g. in Python (van Rossum, 1995), only a handful of libraries that generate design of experiment exist (namely pyDOE, pyDOE2, dexpy, experimenter and GPdoemd) with limited outputs. Thus, the study of DoE packages is also revealing into the current status of the field of experimental design.

The paper is organisd as follows. Section @ref(eda)

Explorative analysis

- Not much difference between title and description.
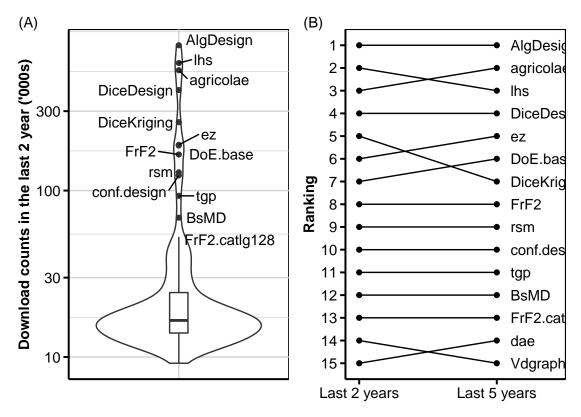- Just go with description and combine bi & tri.

Figure 1: The above graph shows the total download counts from 2019-10-04 to 2021-10-02 of DoE packages.

What are the common types of experimental designs?

How do packages interplay with each other?

Which packages are widely utilised?

Figure 1 suggests that the **AlgDesign** followed by **agricolae** are far more downloaded than any other DoE packages. It is worth noting that **agricolae** imports **AlgDesign** thus any download of **agricolae** likely results also in a download of **AlgDesign**.

Software design

Discussion

An experiment involves running a number of steps:

1. Defining the hypothesis or question
2. Formulating methods to test hypothesis or answer question
3. Planning the experimental protocol:

- Determining experimental resources,
- specifying the data collection procedure,
- constructing the experimental design layout, and
- proposing a model for analysis.

4. Collect data
5. Analyse data, interpret result and conclusion

word	n
optimal design	9
experimental design	8
clinical trial	5
dose finding	5
block design	4
microarray experiment	4
response surface	4
sequential design	4

word	n
experimental design	10
optimal design	10
package provide	7
graphical user	6
response surface	6
user interface	6
block design	5
contour plot	5
design based	5
effect model	5
factorial design	5
microarray experiment	5
mixed effect	5
provide function	5
sample size	5

Table 1: My tables

Bibliography

G. van Rossum. Python tutorial. Technical Report CS-R9526, Centrum voor Wiskunde en Informatica (CWI), Amsterdam, May 1995. [p1]

Emi Tanaka
Monash University
Monash University
Clayton campus, VIC 3800, Australia
http://emitanaka.org/
ORCiD: 0000-0002-1455-259X
emi.tanaka@monash.edu