



cleanR: Your Personal Maid for Cleaning Datasets in R

Claus Thorn Ekstrøm

Biostatistics

Department of Public Health

University of Copenhagen

Anne Helby Petersen

Biostatistics

Department of Public Health

University of Copenhagen

Abstract

Data cleaning and validation is the first step in any data analysis since the validity of the conclusions from the analysis hinged on the quality of the input data. Ideally, a human investigator should go through each variable in the dataset and look for potential errors — both in input values and coding.

We describe an R package which implements an extensive and customizable suite of checks to be applied to the variables in a dataset in order to identify potential problems in the corresponding variables. The typical output is a stand-alone document that summarizes the variables and lists potential errors. The results are typically presented in a stand-alone document that could be perused by an investigator with an understanding of the variables in the data and the experimental design.

The **cleanR** package is designed to be easily extended with custom user-created checks that are relevant in particular situations.

Keywords: data cleaning, quality control, R.

1. Introduction

Statisticians and data analysts spend a large portion of their time on data cleaning and on data wrangling. Packages such as **data.table**, **plyr**, and **tidyverse** make data wrangling a lot easier in R, but there are only a few options available for automated data cleaning.

Data cleaning is a time consuming endeavour and it inherently requires human interaction since every dataset is different and the variables in the dataset can only be understood in the proper context of the experiment. While each dataset is different and require ... (“simple questions”

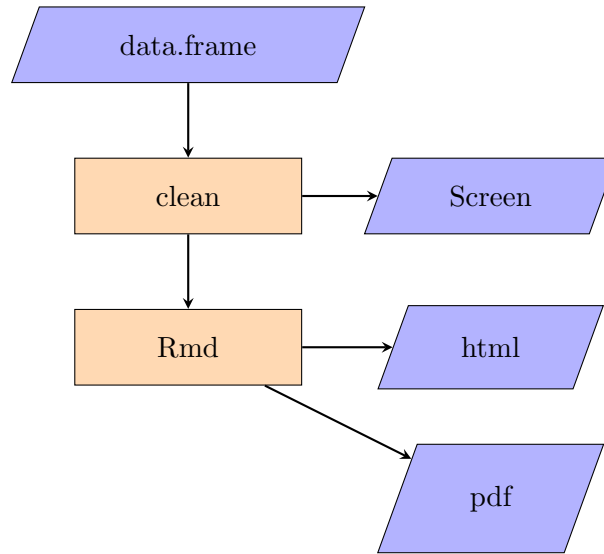


Figure 1: The process of cleaning data frames using the **cleanR** package. A series of check is applied to each variable in a `data.frame` and a summary of the result is either printed to the screen, or an R markdown file is produced which is subsequently rendered.

... there are a number of common occurrences that This is particularly Automated
 ... partly be done by ... However ... enough ... similar tasks that pop
 However it is not uncommon for ... to run a set of

In this paper, we will briefly present some properties of the simplex distribution, which provides a foundation for the inference and computation in the package `simplexreg` (Zhang, Qiu, and Shi 2016). The package is implemented in the R system and available from the Comprehensive R Archive Network (CRAN) at <https://CRAN.R-project.org/package=simplexreg>. There are three major lines of functions: 1. Calculation of the density, cumulative distribution, quantile and random number generating functions for the simplex distribution. 2. Statistical inference in the simplex generalized linear model (SGLM, Zhang and Qiu 2014) via maximum likelihood (ML) for cross-sectional proportional data set. 3. Analysis of longitudinal proportional data set via an extended version of generalized estimating equations (GEE) using the simplex distribution.

The manuscript is organized as follows. Section 2 presents a worked example and tutorial on how to use the **cleanR** package to create a summary of potential errors. Section 3 illustrates the methodology of modeling proportional data using simplex generalized regression. The generalized estimating equations for longitudinal proportional outcomes are given in Section 4. Then we address model diagnostics in Section 5. Section 6 presents the details of the `simplexreg` package. Section 7 further conducts analyses based on the simplex distribution in R with real data sets. Finally, plans for extending the package are described in Section 8.

2. Checking a dataset for errors

In **cleanR** the `clean` function

```

> library(cleanR)
> data(testData)
> head(testData)

  charVar factorVar numVar intVar boolVar keyVar emptyVar _joeVar jack__var
1      a         a      1      1   TRUE      1         1         1         1
2      b         b      2      2  FALSE      2         1         2         2
3      c         c      3      3   TRUE      3         1         3         3
4      a         a      4      4   TRUE      4         1         4         4
5      b         b      5      5   TRUE      5         1         5         5
6      d         d      6      6  FALSE      6         1         6         6

  numOutlierVar smartNumVar      cprVar      cprKeyVar miscodedMissingVar
1              1           0 010101-1111 010101-1111                      .
2              2           0 020102-2929 020102-2929                      .
3              3           0 121201-1902 121201-1902                      nan
4              4           0 030729-2222 030729-2222                      NaN
5              5           0 080909-1212 080909-1212                      NAN
6              6           0 010101-1111 020202-0101                      na

>
# clean(testData)
> 2+3

```

```
[1] 5
```

Arguments

3. The structure of cleanR

asd asd

4. Extending cleanR by adding custom error checks

Lav en situation svarende til eksemplet

```

> characterFoo <- function(v) {
+   if (substr(substitute(v), 1, 1) == "_") {
+     out <- list(problem=TRUE, message="Note that the variable name begins with \\_")
+   } else out <- list(problem=FALSE, message="")
+   out
+ }
> class(characterFoo) <- "checkFunction"
> attr(characterFoo, "description") <- "I really hate underscores"
> #clean(testData, characterChecks=c(defaultCharacterChecks(), "characterFoo"))
>

```

Lav også et eksempel med rangecheck.

5. Using the online web-app

asd asd

Affiliation:

Claus Thorn Ekstrøm
Biostatistics, Department of Public Health
University of Copenhagen
Denmark
E-mail: ekstrom@sund.ku.dk
URL: <http://eeecon.uibk.ac.at/~zeileis/>