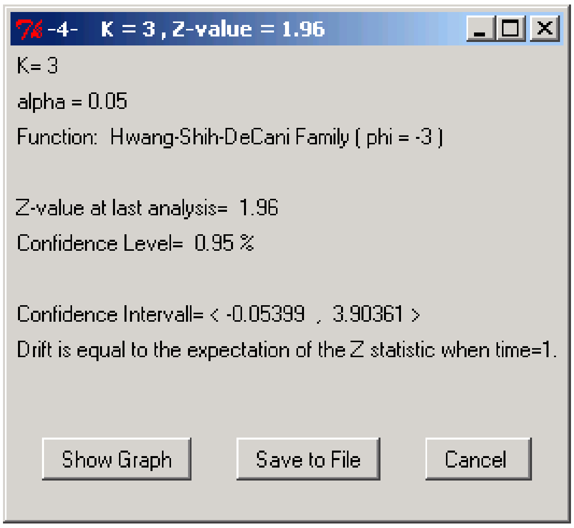

Figure 12: *Resulting confidence interval (Two-Sided-Bounds).*

The window displaying the calculated confidence interval differs somewhat from the other windows (see Figure 12), as no table is presented. The bounds still can be visualized using "Show Graph", however.

Future development

Future versions of **GroupSeq** will include additional software ergonomic improvements such as a status bar during calculation. Additionally, an "undo" function will be added, as well as a feature to let users specify their own default values, thereby achieving more efficiency in frequent use of **GroupSeq**. Furthermore, it is planned to implement *adaptive* group sequential designs.

Bibliography

I. K. Hwang, W. J. Shih, and J. S. DeCani. Group sequential designs using a family of type I error probability spending functions. *Statistics in Medicine*, 9:1439–1445, 1990.

C. Jennison and B. W. Turnbull. *Group Sequential Methods with Applications to Clinical Trials.* Chapman & Hall/CRC, 2000.

K. Kim and D. DeMets. Design and analysis of group sequential tests based on the type I error spending rate function. *Biometrika*, 74:25–36, 1987.

K. K. G. Lan and D. L. DeMets. Discrete sequential boundaries for clinical trials. *Biometrika*, 70:659–663, 1983.

J. Neyman and E. S. Pearson. On the use and interpretation of certain test criteria for purposes of statistical interference. *Biometrika*, 20A:263–295, 1928.

P. C. O'Brien and T. R. Fleming. A multiple testing procedure for clinical trials. *Biometrics*, 35:549–556, 1979.

S. J. Pocock. Group sequential methods in the design and analysis of clinical trials. *Biometrika*, 64: 191–199, 1977.

D. M. Reboussin, D. L. DeMets, K. Kim, and K. K. G. Lan. Programs for Computing Group Sequential Boundaries Using the Lan-DeMets Method, Version 2.1. 2003. URL `http://www.biostat.wisc.edu/landemets/`.

G. Wassmer. *Statistische Testverfahren für gruppensequentielle und adaptive Pläne in klassischen Studien.* Alexander Mönch Verlag, Köln, 2001.

A. Ziegler, G. Dahmen, I. R. König, G. Wassmer, P. Hildebrand, and M. Birth. Accelerating clinical trials by flexible designs. *Good Clinical Practice Journal*, 10:16–19, 2003.

Roman Pahl
`roman.pahl@gmx.de`
Andreas Ziegler
`ziegler@imbs.uni-luebeck.de`
Inke R. König
`Inke.Koenig@imbs.uni-luebeck.de`
Institute of Medical Biometry and Statistics
University at Lübeck, Germany

Using R/Sweave in everyday clinical practice

by Sven Garbade and Peter Burgard

The Sweave package by Friedrich Leisch is a powerful tool to combine R with LaTeX facilities for text formatting (R Development Core Team, 2005; Leisch, 2002a,b). Sweave allows the dynamic generation of statistical reports by using literate data analysis. A further application of Sweave is the package vignette (Leisch, 2003). An introduction to the abilities of Sweave can be found on Friedrich Leisch's Homepage (URL: http://www.ci.tuwien.ac.at/~leisch/) and in several publications, for example Leisch (2002b,c).

Some issues ago Damian Betebenner wrote an article on how to use R/Sweave in batch mode for automatic generation of prototype reports (Betebenner, 2005). His article pointed out very helpful and instructive notes on how to combine R and Sweave in batch processing and how to use control structures with Sweave.

Our article focuses on a combination of R/Sweave/TclTk in batch mode to produce statistical reports of a neuropsychological examination. The emphasis of this article is on the background, functional range and handling of our tool and its use in our hospital.

Background and Motivation

We routinely use the computer based neuropsychological test-battery **CORTEX** (**CO**ntextual **R**eaction **T**ime **EX**ercises) in our metabolic polyclinic for clinical diagnostics, monitoring and evaluation. A *neuropsychological test-battery* is a set of several psychological tests, each testing a specific function of the brain. The emphasis of our tests is on cognitive skills, namely visual short term memory, detection and recognition abilities. The results from the tests give an insight into a patient's cognitive state, and are useful for further diagnostics and treatment recommendations.

Currently six different tests are bundled in the battery. An investigation with all six tests lasts about one hour. Here we describe briefly the six tests; for a more detailed description of some of these tests and their theoretical background see Burgard et al. (1997).

The first test is a simple visuomotor reaction task requiring the testee to press the button of a response panel as fast as possible after a square appears on the monitor screen (Test *Reaction*). The test *CPT (Continuous Performance Task)* measures sustained attention by a binary choice task. The testee is required to press the response button of the dominant hand as quickly as possible when four dots appear, and the response button of the non-dominant hand when three or five dots appear. The *LDT (Letter Detection Task)* requires the testee to scan four letters displayed in a rectangular layout around the center of the computer screen for one, two or three target letters. The number of target letters determines task difficulty. In the *VWM (Visual Working Memory)* task the testee sees a 7×7 matrix filled with black and white squares for one second. After a retention interval of four seconds, the previous matrix and three other matrixes are shown, and the testee has to click on the previously shown matrix. Task difficulty is determined by similarity between the target matrix and the three comparative matrixes. The *Dual Task* contains three different tasks and the testee has to deal with visual and auditory stimuli. In the first task, the testee has to press a response button as fast as possible when a particular tone among other tones is played; in the second task, large or small rectangles are displayed in random order and the testee has to press a button when the small rectangle appears. The third task is a combination of both; i. e., the testee has to press a response key when the small rectangle or the particular tone is played. The *Tracking* task is a test of the ability of eye-hand coordination. The testee has to pursue manually with the mouse pointer a randomly moving target on the computer screen. The target is presented among 10 other moving objects. To complicate this task, the movements of the mouse pointer are delayed. The results of each test are saved as integer variables in a tab-separated text file.

The neuropsychological tests and the analysis tool for generating statistical reports are implemented on a notebook computer running a Debian GNU/Linux system. The response panel is connected to the computer with a USB data acquisition board (USB-DUX). Using a notebook guarantees mobility and the neuropsychological examinations can be arranged in almost every room in our clinic.

The motivation for the present project was the need for a tool which could be easily operated by psychologists, nurses, physicians, etc., to generate immediately after testing a statistical report based on the results of the neuropsychological tests.

Handling and functional range

A screen-shot of the main window is shown in Figure 1. A click on the Help button starts a PDF viewer showing an instruction manual, and the Quit button closes the application.

The process to generate a statistical report with our tool can be summarized in four steps:

1. The user selects with radio buttons the neuropsychological test for which a report should be generated.
2. The user browses the file system and chooses one or more raw data files associated with the selected neuropsychological test. Choosing data files from several testees as well as repeated measurements of the same testee is possible.
3. The user can add information about the testee and investigator not provided by the data files from the neuropsychological test-battery, for example, names, diagnosis, date of birth, etc.
4. When all raw data files are selected, the user generates the statistical report with a single mouse click.

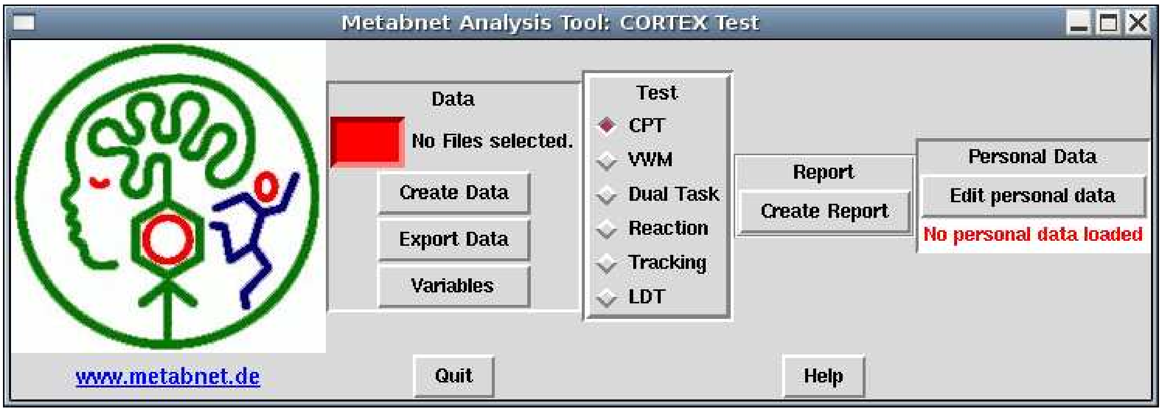

Figure 1: Screen shot of the main window. The label on top of the **Data** frame indicates that no data files are selected.

The main window has four frames with several buttons. The following paragraphs describe briefly the functions and use of the program.

Data. The first frame, **Data**, integrates buttons for loading and exporting data files. On top of this group, information about how many data files were loaded by the user and for which test the report will be generated is displayed.

The data frame can be exported to a tab-separated text file with a click on the Export data button. The variables in the data frame could be inspected by a click on Variables.

A click on the Create Data button opens the dialog shown in Figure 2. The dialog provides buttons for browsing the filesystem, selecting, deleting and inspecting raw data files saved by the neuropsychological test-battery. Once a file is selected, its name and path are shown in the list.

The callback function for the Add button opens a file selection dialog to browse the file system. With Remove the user can delete a marked data file from the list. The File Info button calls a function to display the content of a marked data file.

The Load button calls a function which does several computations. First, duplicate entries are identified. Then the selected files shown in the list widget are merged into a data frame. Clicking on the Cancel button closes the dialog without applying any changes.

Test. The **Test** frame is a set of several radio buttons. The value of the activated radio button defines the Sweave file associated with a statistical report. The value is also used to create the window title and label in Figure 2.

Report. The Create Report button calls the function responsible for constituting the correct command generating the statistical report. When the report is created successfully, a PDF viewer will be started showing the generated report.

Personal Data. Clicking the button in this frame opens a new window where the user can add information not provided by the raw data from the test-battery (see Figure 3). These data concern the name of the testee, diagnosis and date of birth, the values of medical parameters, the investigator's name, and

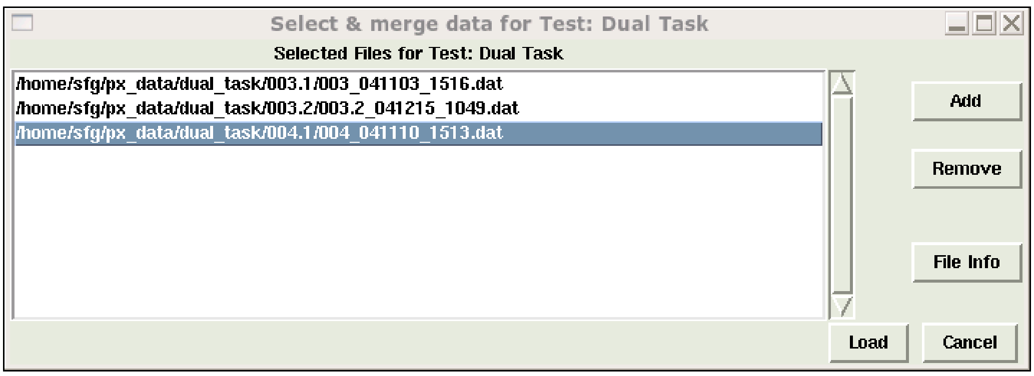

Figure 2: The dialog for creating the data frame with some raw data files selected. The last file in the list is marked by a mouse click.

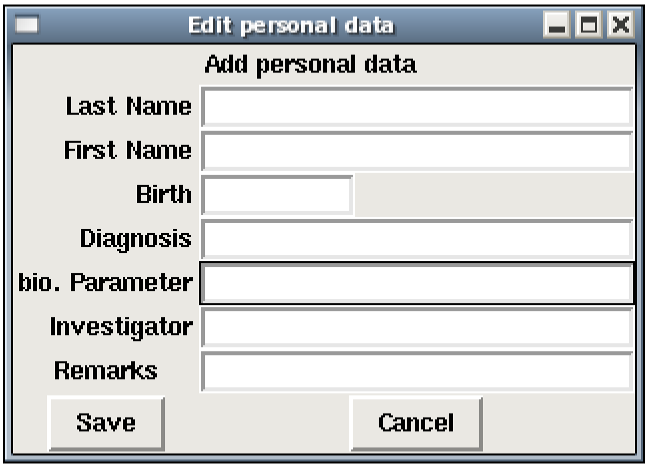

Figure 3: The dialog for adding additional information about testee and investigator.

possible remarks. Only information that is present is included in the statistical report.

The red label in the **Personal Data** frame in the main window (see Figure 1) changes to a green colored label "Personal data loaded" when any text is supplied. When all characters from the text entries are deleted, the label color changes back to red and the label text to "No personal data loaded."

Variables and statistics

The results of each neuropsychological test are saved as integer variables in a tab-separated text file, one row per trial and one column per variable. The data files are saved in separate directories, one directory for each neuropsychological test. The amount of saved data depends on the neuropsychological test. Examples of saved variables are the reaction time a testee needed to respond to the appearance of a stimulus, the number of stimuli on the screen the testee had to attend to, the mode of the response, e. g., wrong, correct or false alarm (that is the testee reported that there was a target to respond to, but indeed there was none), or the distance in pixels between the mouse pointer and the target to pursue.

Because all variables are saved as integers, the program recodes integer variables representing a grouping variable into a factor variable, e. g., 0/1 to wrong/correct response.

The raw data files from the test-battery provide no information about the ID (*Identification Number*) of the testee or date of psychological examination. But the file name does: The ID, date and number of the examination are extracted from the filename. The names of the data files have the pattern `id_date_time.dat`, e. g. `004_041110_1513.dat` for testee 004 tested on 10 November 2004 at 15:13 o'clock (see marked file in Figure 2). A function splits the filename at the underscores and builds three factor variables coding the ID, date and time of examination and the session, that is the number of examination. The first part `code` of the filename is only evaluated until the first dot ".", e. g. `001` and `001.MSUD` result in the same ID `001`. This gives the investigator the possibility to include some more information into the file name without disturbing the computation of the ID, date and session factors.

The statistical reports are compiled using the basic LaTeX `article` class and have about three to five pages, depending on the neuropsychological test. Each report consists of a fixed part, that is a header and a basic analysis, and a second part covering a detailed statistical description of the variables acquired by the neuropsychological test.

The header contains a logo, the address of our hospital, a title, and the issue date of the report. Then variables from the text entries of the personal data dialog (see Figure 3) are included, when available. The subsequent basic analysis lists the number of testees, trials and dates of examination, and names of the data files the statistics are based on.

The second part describes the results of the neuropsychological test by tables and graphics, mostly scatter plots and trellis graphics. The statistical reports were designed to give a first glance into the data, and therefore they have a primarily descriptive character. In most cases, for each variable of interest (e. g. the reaction time from the *Reaction* test) the mean, median and standard deviation across testees and number of examinations are computed and displayed graphically. So a report allows the comparison of the results from one or more testees at several dates of examination.

Some remarks about the implementation

In principle, our tool has two parts: An R/TclTk function containing all routines for the graphical user interface (GUI) and data manipulation, and several

Sweave style files with R/LaTeX routines for dynamic generation of statistical reports. A simple flowchart of the tool is shown in Figure 4.

The GUI was built with the **tcltk** package that comes with R. For a short but comprehensive introduction to the **tcltk** package see Dalgaard (2001), the Homepage of Wettenhall and of course the R help pages.

The R/TclTk function holds the rank of a "master" function. It provides dialogs for the user and defines and calls R functions to generate a data frame suitable to load within an Sweave file. Further it constitutes the correct command to generate the desired statistical report.

The code chunks in Sweave files have no access to variables within the running "master" function. To supply the Sweave files with all necessary data, the "master" function saves an R image which includes all variables and a data frame created from the selected raw data files. The variables are, for example, names, investigator, and diagnosis that come from the dialog in Figure 3, as well as some other variables that might be used to create a statistical report, e. g., the value of the activated radio button in the **Test** frame in the main window, and the path and names of the selected raw data files.

To generate the statistical report, two Sweave processes are started sequentially. Both Sweave processes load the previously saved R image. An Sweave process first evaluates the test-specific Sweave file; the second Sweave command processes the template Sweave file and includes the previously generated test-specific LaTeX file. The resulting LaTeX file is then processed by pdfLaTeX using the `texi2dvi()` function in package **tools**. When the report is generated successfully, a PDF reader showing the statistical report is started.

Because the program has to deal with different Sweave and R files, each report is built in a separate directory which includes an Sweave file for the test-specific routines and sometimes an additional file with R functions. The template Sweave file and the R image are stored in a shared directory. In this way, a more maintainable structure can be achieved.

Summary

Our purpose was to introduce a tool we developed with a combination of R, Sweave and the **tcltk** package and which we now use routinely in our everyday clinical practise. The sophisticated statistical analysis and graphic capabilities of R can be bundled with Sweave and TclTk to build an application allowing the user to generate a comprehensive report of a neuropsychological examination immediately after test-

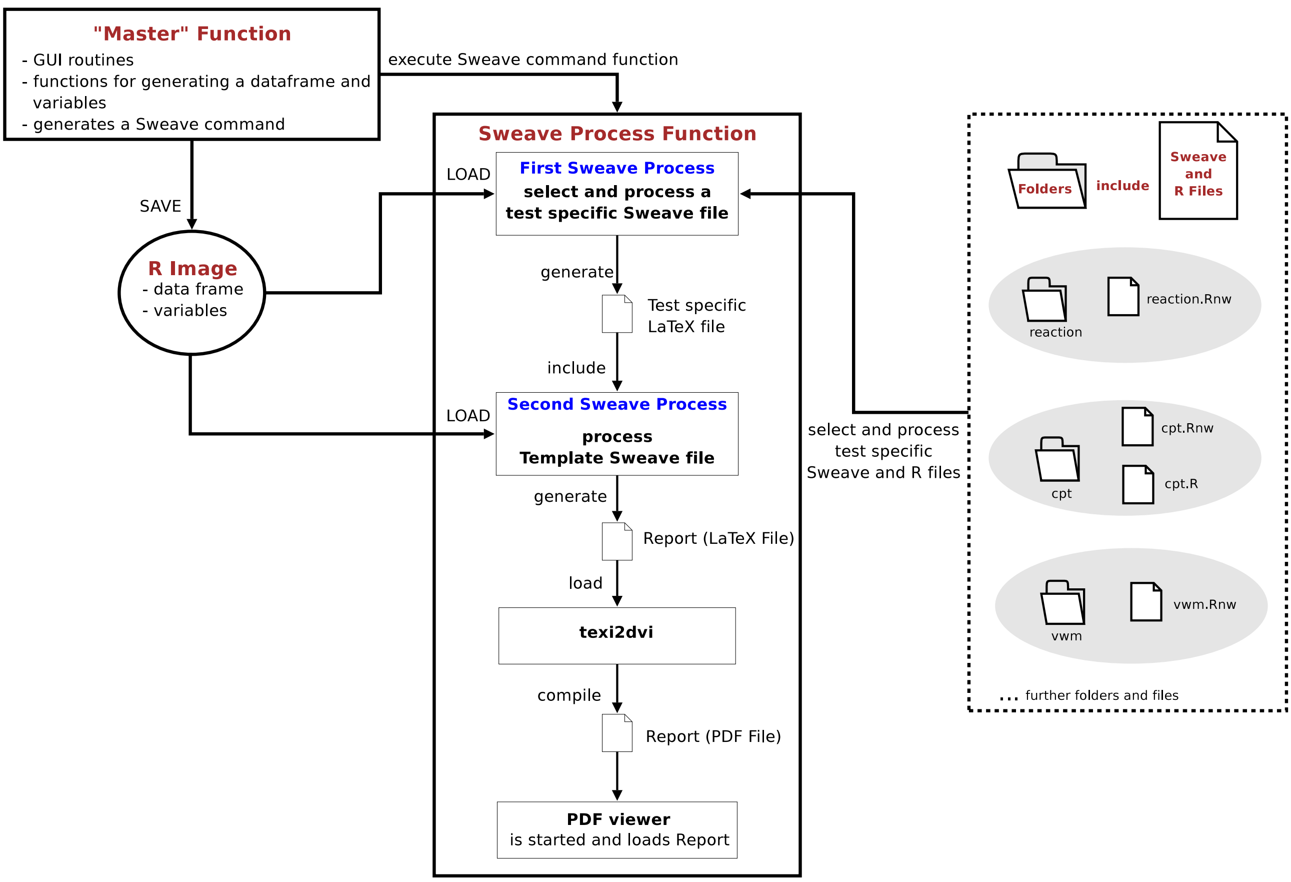

Figure 4: Simple flowchart of the application.

ing, even without statistical expertise.

This software is currently in a testing phase and under active development. We still are adapting some of the statistical reports to the needs of the users and maybe we will expand the number of tests of our neuropsychological test-battery and therewith the range of statistical reports our tool supports. But so far, we can point out that using R/Sweave/TclTk is much more powerful compared to data analysis of test results in the past. Some of the neuropsychological tests we briefly reported about ran formerly in an old version of MS-DOS, and subsequent data analysis was a bit hampered because exporting the raw data from an examination to a statistical analysis system was necessary. After reprogramming the neuropsychological tests on a modern computer system with the facilities of R, it was a relatively simple matter to combine the neuropsychological test-battery with a convenient tool for exploring the data.

One further improvement we are considering is an interface for connecting our tool to a data base system. This offers the opportunity to save and reload easily the data from an examination.

Acknowledgment

We would like to thank an anonymous reviewer who gave us helpful remarks on a previous version of this manuscript, as well as suggestions about how to improve our code.

Bibliography

D. Betebenner. Using control structures with Sweave. *R News*, 5(1):40–44, May 2005. URL http://CRAN.R-project.org/doc/Rnews/.

P. Burgard, F. Rey, A. Rupp, V. Abadie, and J. Rey. Neuropsychologic functions of early treated patients with phenylketonuria, on and off diet: results of a cross-national and cross-sectional study. *Pediatric Research*, 41:368–374, 1997.

P. Dalgaard. A primer on the R-Tcl/Tk package. *R News*, 1(3):27–31, September 2001. URL http://CRAN.R-project.org/doc/Rnews/.

F. Leisch. Sweave: Dynamic generation of statistical reports using literate data analysis. Presented at "Österreichische Statistiktage 2002", Vienna, Austria, October 23–25 2002a.

F. Leisch. Sweave, part I: Mixing R and LaTeX. *R News*, 2(3):28–31, December 2002b. URL http://CRAN.R-project.org/doc/Rnews/.

F. Leisch. Sweave user manual. URL http://www.ci.tuwien.ac.at/~leisch/Sweave. Institut für Statistik und Wahrscheinlichkeitstheorie, Technische Universität Wien, Vienna, Austria, 2002c.

F. Leisch. Sweave, part II: Package vignettes. *R News*, 3(2):21–24, October 2003. URL http://CRAN.R-project.org/doc/Rnews/.

R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2005. URL http://www.R-project.org. ISBN 3-900051-07-0.

USB-DUX. URL http://www.linux-usb-daq.co.uk/.

J. Wettenhall. URL http://bioinf.wehi.edu.au/~wettenhall/RTclTkExamples/.

Sven Garbade and Peter Burgard
University Hospital for Pediatric and Adolescent Medicine
Department of General Pediatrics
METABNET - Epidemiological Register
D-69120 Heidelberg, Germany
www.metabnet.de
Sven.Garbade@med.uni-heidelberg.de

changeLOS: An R-package for change in length of hospital stay based on the Aalen-Johansen estimator

by M. Wangler, J. Beyersmann, and M. Schumacher

Introduction

Length of hospital stay (LOS) is used to assess the utilization of hospital resources, the costs and the general impact of a disease. Change in LOS due to a complication is frequently used to assess the impact and the costs of a complication. Prominent examples include nosocomial infections and adverse drug events. In a data analysis, it is important to regard the timing of events: A complica-