WIRTSCHAFTS UNIVERSITÄT WIEN VIENNA UNIVERSITY OF ECONOMICS AND BUSINESS

EFMD EQUIS ACCREDITED

ePub^{WU} Institutional Repository

Bettina Grün and Achim Zeileis

Automatic generation of simple (statistical) exams

Working Paper

Original Citation:
Grün, Bettina and Zeileis, Achim (2008) Automatic generation of simple (statistical) exams. *Research Report Series / Department of Statistics and Mathematics*, 77. Department of Statistics and Mathematics, WU Vienna University of Economics and Business, Vienna.

This version is available at: http://epub.wu.ac.at/1366/
Available in ePub^{WU}: November 2008

ePub^{WU}, the institutional repository of the WU Vienna University of Economics and Business, is provided by the University Library and the IT-Services. The aim is to enable open access to the scholarly output of the WU.

http://epub.wu.ac.at/

Automatic Generation of Simple (Statistical) Exams

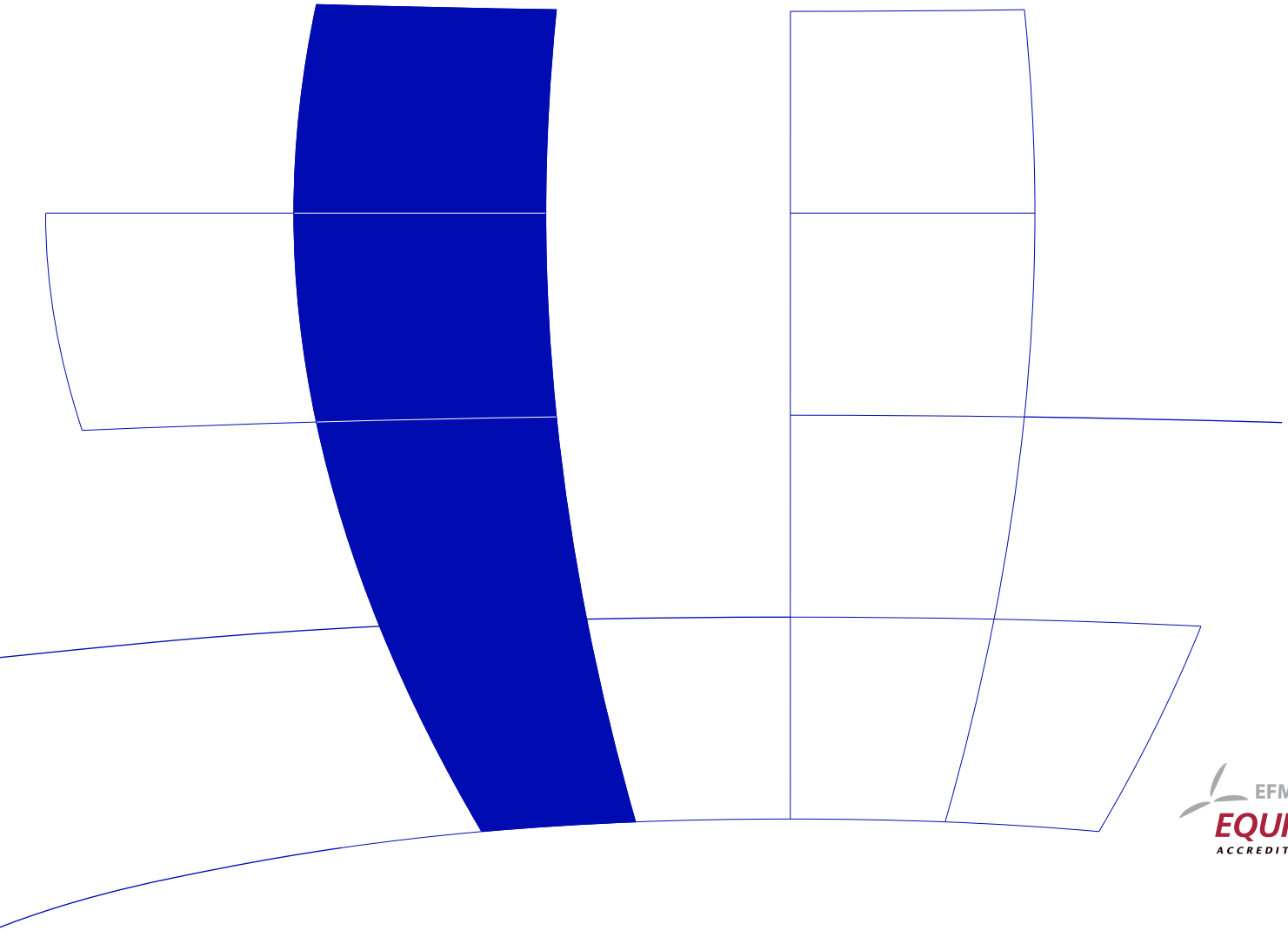
Bettina Grün, Achim Zeileis

Department of Statistics and Mathematics
Wirtschaftsuniversität Wien

Research Report Series

Report 77
November 2008

http://statmath.wu-wien.ac.at/



Automatic Generation of Simple (Statistical) Exams

Bettina Grün
Wirtschaftsuniversität Wien

Achim Zeileis
Wirtschaftsuniversität Wien

Abstract

Package **exams** provides a framework for automatic generation of simple (statistical) exams. To employ the tools, users just need to supply a pool of exercises and a master file controlling the layout of the final PDF document. The exercises are specified in separate Sweave files (containing R code for data generation and LaTeX code for problem and solution description) and the master file is a LaTeX document with some additional control commands. This paper gives an overview on the main design aims and principles as well as strategies for adaptation and extension. Hands-on illustrations—based on example exercises and control files provided in the package—are presented to get new users started easily.

Keywords: exams, multiple choice, arithmetic problems, Sweave, LaTeX, R.

1. Introduction

The introductory statistics lecture at the Wirtschaftsuniversität Wien (WU Wien, http://www.wu-wien.ac.at/), Vienna, Austria, is attended each semester by about 1,000–1,500 students (mostly first-year business students). Several lecturers from the Department of Statistics and Mathematics teach this course in parallel. The **exams** package was written as part of a collaborative department initiative to re-design the course format for this lecture, ensuring an efficient, consistent, and transparent organization. The re-design covered all aspects of teaching materials, i.e., new presentation slides, collections of exercises, exams, etc. The main challenges in this process—that the **exams** package tries to address—were:

- *Scalable exams:* Automatic generation of a large number of different exams.
- *Associated self-study materials:* Collections of exercises and solutions from the same pool of examples.
- *Joint development:* Development and maintenance of a large pool of exercises in a multi-author and cross-platform setting.

Specifically, at WU Wien about 10–15 lecturers were working in small teams of 2–4 people on different chapters for the presentation slides. For each chapter, the corresponding team would also provide suitable exercise templates that could be used for self-study materials, exams, and solutions.

Prior experience had shown that the exercises needed to be flexible in order to allow for enough variation and avoid "learning by heart": Students should not be able to guess the correct

answer of a multiple choice question from a certain key word in the problem description. Ideally for each problem setting a set of exercises should exist where all combinations of possible answers are correct. In addition, the exercises should not only require students to do calculations but also cover interpretation of results, statistical graphics and statistical software output. However, fast and easy correction should still be possible based only on a short result as filled in by the students on the cover sheet of their exams.

These requirements on maintenance, variation, and correction of exercises led to the following design principles for package **exams**:

- *Maintenance:* Each exercise template is a single file (also just called "exercise").
- *Variation:* Exercises are dynamic documents, containing a problem/solution along with a data-generating process (DGP) so that random samples can be drawn easily.
- *Correction:* Solutions for exercises are either multiple-choice answers (logical vectors) or numeric values (e.g., a test statistic or a confidence interval).

Thus, the DGP of an exercise controls the distribution of possible solutions and can be utilized to make them (approximately) evenly distributed and difficult to "guess" or "learn by heart". In addition to the variability within an exercise, one can add further variation by providing several exercise templates for the same type of problem. Depending on the flexibility of the DGP, the pool of exercises can thus be rather small or needs to be somewhat larger.

Mixing problems/solutions and DGPs for exam generation poses challenges that are similar to those of making data analyses reproducible. Thus, **exams** employs many ideas from literate data analysis (Rossini 2001; Leisch and Rossini 2003), literate programming (Knuth 1992) and reproducible research (de Leeuw 2001). Specifically, it makes extensive use of Sweave (Leisch 2002) for mixing DGPs written in the R system for statistical computing (R Development Core Team 2008) and problem/solution descriptions written in the typesetting system LaTeX (Knuth 1984; Lamport 1994). Thus, the implementation in **exams** is based on (1) independent Sweave files for each exercise interweaving R and LaTeX code, (2) different master LaTeX files controlling the appearance and (3) an implementation of a minimal markup for communication with R plus R functions tying everything together.

In the development process at WU Wien, this addressed all requirements: All lecturers involved are familiar with R and LaTeX (which are both available on all standard platforms) and can thus contribute to the pool of exercises. To do so, they just need to know the structure of the exercise Sweave files while different master LaTeX files (for the department's exams, exercise collections, etc.) have been written by the authors of the **exams** package. To provide all lecturers involved with access to all resources, **Subversion** (SVN, Pilato, Collins-Sussman, and Fitzpatrick 2004) is used for version control. This approach proved to be rather successful in addressing the needs of multi-author and cross-platform development.

The package **exams** that emerged from the developments at WU Wien is now available from the Comprehensive R Archive Network at `http://CRAN.R-project.org/package=exams`. To utilize it for exam generation, users need to generate a pool of exercises and supply a master LaTeX file. The package provides several examples for both types of files along with documentation (like this paper) and hands-on examples to get new users started. Hence, the remainder of this paper is organized as follows: Section 2 explains the structure of the Sweave

files defining the exercises and demonstrates how the final output file is derived from the original R and LaTeX code. In Section 3, the structure of the master LaTeX file for constructing the final PDF file from the exercise files is presented. Finally Section 4 illustrates how to use the package in practice and how exams can be customized. Section 5 concludes the paper with a discussion.

2. Exercises

Each exercise is contained in a separate Sweave file and typically represents an exemplary application of a statistical procedure. It needs to contain (1) a question and a solution description which are encapsulated in LaTeX environments of the corresponding names and (2) some meta-information about the exercise such as type and the solution. To allow for variability, the data for the exercise can be generated within the Sweave file in a code chunk (typically suppressed in the final document). Meta-information for the examples needs to be provided, e.g., for computing with the solution within R or for generating lists with solutions.

In Table 1, the Sweave file for a simple exercise asking students to compute a one-sample t test statistic is shown for illustration. The four different blocks (R code with DGP, `question` environment, `solution` environment, meta-information) can easily be distinguished. The LaTeX file generated from a call to `Sweave()` is shown in Table 2, and the final compiled PDF output is provided in Table 3.

The R code for the DGP is used as in any other Sweave document: The code chunk is framed by `<<>>=` and `@` and options in the header control formatting (typically suppressing the display in the final document, as in this example). For further information about the application of Sweave and a detailed description of the different options see Leisch (2008a,b).

The core of an exercise are two blocks for the question and the solution in their accordingly named environments. The values generated by the DGP are typically included using `\Sexpr{}` statements. Furthermore, question/solution might contain graphics and/or R output created by Sweave. To include the LaTeX output (see Table 2) in some master LaTeX file, this needs to define the `question` and `solution` environment. This mechanism can be used to control the display, e.g., to suppress the solution in an exam but to display it in associated self-study material (see Section 3 for further details).

Finally, some meta-information needs to be provided. All meta-information commands are in LaTeX style but are actually commented out and hidden in the final output file. There are four commands:

- `\extype{}`: type of question. It may be either `mchoice` for multiple choice questions or `num` for questions with a numeric result.
- `\exsolution{}`: solution. It contains the numeric solution or a string of zeros and ones for multiple choice questions. For confidence intervals, it may also specify two solutions of type `\exsolution{`*lower*`}{`*upper*`}`.
- `\exname{}`: a name or description. This is some short informative text, only used for printing of solutions within R.
- `\extol{}`: tolerance limit (optional). For numeric solutions a tolerance limit can be specified; by default it is assumed to be 0. This can be useful for automatic evaluations

```
<<echo=FALSE, results=hide>>=
## DATA GENERATION
n <- sample(120:250, 1)
mu <- sample(c(125, 200, 250, 500, 1000), 1)
y <- rnorm(n, mean = mu * runif(1, min = 0.9, max = 1.1),
           sd = mu * runif(1, min = 0.02, max = 0.06))

## QUESTION/ANSWER GENERATION
Mean <- round(mean(y), digits = 1)
Var <- round(var(y), digits = 2)
tstat <- round((Mean - mu)/sqrt(Var/n), digits = 3)
@

\begin{question}
  A machine fills milk into $\Sexpr{mu}$ml packages. It is suspected that the
  machine is not working correctly and that the amount of milk filled differs
  from the setpoint $\mu_0 = \Sexpr{mu}$. A sample of $\Sexpr{n}$ packages
  filled by the machine are collected. The sample mean $\bar{y}$ is equal to
  $\Sexpr{Mean}$ and the sample variance $s^2_{n-1}$ is equal to
  $\Sexpr{Var}$.

  Test the hypothesis that the amount filled corresponds on average to the
  setpoint. What is the absolute value of the $t$~test statistic?
\end{question}

\begin{solution}
  The $t$~test statistic is calculated by:
  \begin{eqnarray*}
    t & = & \frac{\bar y - \mu_0}{\sqrt{\frac{s^2_{n-1}}{n}}}
    = \frac{\Sexpr{Mean} - \Sexpr{mu}}{\sqrt{\frac{\Sexpr{Var}}{\Sexpr{n}}}}
    = \Sexpr{tstat}.
  \end{eqnarray*}
  The absolute value of the $t$~test statistic is thus equal to
  $\Sexpr{format(abs(tstat), nsmall = 3)}$.
\end{solution}

%% META-INFORMATION
%% \extype{num}
%% \exsolution{\Sexpr{format(abs(tstat), nsmall = 3)}}
%% \exname{t statistic}
%% \extol{0.01}
```

Table 1: A simple Sweave exercise: `tstat.Rnw`.

```
\begin{question}
  A machine fills milk into $500$ml packages. It is suspected that the
  machine is not working correctly and that the amount of milk filled differs
  from the setpoint $\mu_0 = 500$. A sample of $226$ packages
  filled by the machine are collected. The sample mean $\bar{y}$ is equal to
  $517.2$ and the sample variance $s^2_{n-1}$ is equal to
  $262.56$.

  Test the hypothesis that the amount filled corresponds on average to the
  setpoint. What is the absolute value of the $t$~test statistic?
\end{question}

\begin{solution}
  The $t$~test statistic is calculated by:
  \begin{eqnarray*}
    t & = & \frac{\bar y - \mu_0}{\sqrt{\frac{s^2_{n-1}}{n}}}
    = \frac{517.2 - 500}{\sqrt{\frac{262.56}{226}}}
    = 15.958.
  \end{eqnarray*}
  The absolute value of the $t$~test statistic is thus equal to
  $15.958$.
\end{solution}

%% META-INFORMATION
%% \extype{num}
%% \exsolution{15.958}
%% \exname{t statistic}
%% \extol{0.01}
```

Table 2: LaTeX output of `Sweave("tstat.Rnw")`.

of numeric solutions if some deviation from the correct answer is allowed (e.g., due to rounding errors).

The user does not have to go from the Sweave file (Table 1) to the LaTeX file (Table 2) to the PDF file (Table 3) "by hand". The function `exams()` from package **exams** carries out all of these steps automatically and returns a suitable R object with the meta-information invisibly.

```
R> library("exams")
R> set.seed(1090)
R> tstat_sol <- exams("tstat.Rnw")
```

After loading the package, `exams()` is called with the Sweave file name. The final PDF file

---

1. **Problem**
   A machine fills milk into 500ml packages. It is suspected that the machine is not working correctly and that the amount of milk filled differs from the setpoint $\mu_0 = 500$. A sample of 226 packages filled by the machine are collected. The sample mean $\bar{y}$ is equal to 517.2 and the sample variance s^2_{n-1} is equal to 262.56.

   Test the hypothesis that the amount filled corresponds on average to the setpoint. What is the absolute value of the t test statistic?

   Solution
   The t test statistic is calculated by:

$$t = \frac{\bar{y} - \mu_0}{\sqrt{\frac{s^2_{n-1}}{n}}} = \frac{517.2 - 500}{\sqrt{\frac{262.56}{226}}} = 15.958.$$

   The absolute value of the t test statistic is thus equal to 15.958.

Table 3: Display of processed `tstat` exercise.

is displayed on screen by default (for further options see Section 4). The PDF viewer used depends on the operating system: On Windows, the application specified in the system's file associations is used. On other systems (e.g., Linux or Mac OS), the PDF viewer specified in the options (see `getOption("pdfviewer")`) is started. `exams()` returns an object of class "`exams_metainfo`" invisibly, which contains the meta-information provided in the exercise. Its `print()` method displays the information provided:

```
R> tstat_sol

plain1
    1. t statistic: 15.958 (15.948--15.968)
```

3. Combining exercises: The master LaTeX file

To produce the PDF for an exam, the exercises need to be weaved, tied together in the master LaTeX file and subsequently processed to PDF. More precisely, `exams()` takes the following approach:

1. Collect all Sweave files for the exercises, the master LaTeX file(s) and potentially additionally specified input files.
2. Copy all files to a (temporary, by default) directory.
3. Run `Sweave()` for each exercise.

```
\documentclass[a4paper]{article}

\usepackage{a4wide,Sweave}
\newenvironment{question}{\item \textbf{Problem}\newline}{}
\newenvironment{solution}{\textbf{Solution}\newline}{}

\begin{document}
\begin{enumerate}
%% \exinput{exercises}
\end{enumerate}
\end{document}
```

Table 4: A simple master LaTeX file: `plain.tex`.

4. Produce a copy of the master LaTeX file(s) in which certain control structures are substituted by dynamically generated LaTeX commands (e.g., for including the exercises).
5. Run `texi2dvi()` for each master LaTeX file.
6. Store the resulting PDF file(s) in an output directory or display it on the screen (for a single file only, by default).

Thus, the only thing needed to combine several independent exercises to an exam is a master LaTeX file. Its basic structure is quite straightforward but for more elaborate layouts it can become more complex. Package **exams** provides various examples of master LaTeX files in the `inst/tex/` directory of the source package. The simplest one is `plain.tex` (see Table 4) which is described here to provide a first overview.

The structure of `plain.tex` is rather straightforward: First, the document class is defined and then necessary add-on packages are loaded.[1] Then the environments for `question` and `solution` are defined (as part of an itemize or enumerate list). Finally, the document starts and consists only of an enumerate list. In Step 4 of the algorithm described above, the control command `%% \exinput{exercises}` is replaced by (a sequence of) `\input{`*filename*`}`, e.g., `\input{tstat.tex}` in our basic example from the previous section.

Thus, all appearance options can be controlled in the LaTeX code of the master file, e.g., by changing the definitions of the `question`/`solution` environments or by modifying the code around the `\exinput{}` command. As an example, one might want to show only the questions and hide the solutions in the PDF. This is easily obtained by changing the definition of the `solution` environment to `\newenvironment{solution}{\comment}{\endcomment}`.

The structure described so far is completely static, except for the `\exinput{}` command. In fact, `exercises` is not the only argument allowed; at the moment two further arguments can be used to modify aspects of the master LaTeX file dynamically:

[1] Note that `Sweave.sty` is included for displaying R output. `texi2dvi()` ensures that this style file will be found by including R's `texmf` directory in the `TEXINPUTS` path.

- `\exinput{exercises}`: Inclusion of exercises.
  Replaced by: `\input{`*filename*`}` (one for each exercise).
  Example: `\input{tstat}`.
- `\exinput{questionnaire}`: Inclusion of questionnaires, e.g., for cover sheets.
  Replaced by: `\exnum{...}` or `\exmchoice{...}`, respectively (one for each exercise).
  Example: `\exnum{}{}{}{}{1}{5}{9}{5}{8}`.
- `\exinput{header}`: Further commands and definitions.
  Replaced by: `\`*command*`{`*value*`}` (one for each header command).
  Example: `\Date{2008-11-10}`.

For the latter two, the master LaTeX file has to define the commands that are used in the replacement step, e.g., `\exnum` has to be a command that takes nine arguments (corresponding to six and three digits before and after the decimal point for numeric solutions) and `\exmchoice` has to take five arguments (corresponding to the logical multiple choice answers).[2] For the header, arbitrary commands can be set up, see Section 4 for details.

Two master LaTeX files that illustrate all of the commands above are provided in **exams**: `exam.tex` and `solution.tex`. `exam.tex` can be used to generate exams including a cover sheet for students to fill in their names as well as their answers to the problems (hiding solutions for the exercises, obviously). `solution.tex` produces PDF files containing a cover sheet similar to `exam.tex` but with the correct answers already filled in. Furthermore, the `solution` environments are displayed for each exercise.

4. Application and customization: Function `exams()`

In the simple case where only a single Sweave exercise is processed running `exams()` essentially corresponds to first calling `Sweave()` and then `texi2dvi()` on the file after including it in a master LaTeX file. This is quite convenient, especially for non-experts (in R and/or LaTeX), but beyond that not much simplification is gained by `exams()`. The main advantages of the function, however, are its flexibility and customizability: It controls the (dynamic) combination of the (static) Sweave and LaTeX files and allows for

- construction of exams with stratified sampling of exercises,
- automatic generation of multiple copies (potentially of multiple layouts) with suitable names and storage,
- inclusion of a suitable cover page with answer fields,
- collection of meta-information for problems and solutions in an R object.

In the following, we illustrate how `exams()` can be employed to exploit all of these features. First, the interface of `exams()` is briefly described before using it to generate a small set of exams with corresponding solutions.

Function `exams()` has the following arguments:

[2] Note that this implementation has the limitation that at most five answer options are possible for multiple choice questions as well as only a precision up to three digits right from the decimal mark and up to six digits left from the decimal mark for numeric results is possible.

```
exams(file, n = 1, dir = NULL, template = "plain", inputs = NULL,
      header = list(Date = Sys.Date()), name = NULL, quiet = TRUE,
      edir = NULL, tdir = NULL, control = NULL)
```

where `file` specifies a list/vector of exercise Sweave files (see Section 2), `template` is the name of the master LaTeX file (see Section 3), and `n` is the number of random replications. The remaining arguments control details of the processing, e.g., the directories for input/output files. A technical manual is available on the help page `?exams`, a brief hands-on introduction to selected arguments is given below:

`file`: This is either a character vector containing the file names of Sweave exercises or a list of such vectors, e.g.,

```
R> myexam <- list("boxplots",
+                 c("confint", "ttest", "tstat"),
+                 c("anova", "regression"),
+                 "scatterplot",
+                 "relfreq")
```

Exams generated from `myexam` always have five exercises: `"boxplots"`, `"scatterplot"`, and `"relfreq"` are always included but the second exercise is randomly drawn from `"confint"`, `"ttest"`, `"tstat"`. Similarly, the third exercise is randomly chosen from `"anova"` and `"regression"`. This stratified sampling strategy is useful if there are several exercises related to the same topic, or several exercises for the same statistical techniques with different "stories". If only a single vector (rather than a list) of file names is specified, each exercise is always included in the final output file. The extension `.Rnw` can be omitted from the file names and the corresponding files should either be in the local directory, the `edir` directory or in the `exercises` directory of the installed package.

`n`: Number of randomly generated exams.

`dir`: Output directory for storing the resulting PDF files (and meta-information). If only a single PDF file is generated, this can be omitted (and the result is displayed directly on the screen), otherwise it has to be specified. Below we use a temporary output directory

```
R> odir <- tempfile()
```

but the reader can easily change `odir` to some local directory.

`template`: A (vector of) master LaTeX file(s). If more than one `template` is specified, one PDF output file is created for each in each of the `n` runs. The extension `.tex` can be omitted in the file name and the corresponding files should either be in the local directory (or provided with the full path) or in the `tex` directory of the installed package.

`header`: Additional LaTeX commands for replacement of `\exinput{header}` in the master LaTeX file. It has to be a `list()` of *command* `=` *value* pairs, where *value* can either be a static string or a function computing a string from the index i of the ith exam. Below we use `ID = getID` with

```
R> getID <- function(i)
+    paste("myexam", gsub(" ", "0", format(i, width = 2)), sep = "")
R> getID(1)

[1] "myexam01"
```

Thus, a different `\ID{myexam`*ii*`}` command will be substituted in the header of each exam. (The LaTeX command `\ID{}` has to be defined in the master LaTeX files.)

Further arguments: `inputs` can specify a list of files needed during the LaTeX compilation, e.g., private `.sty` files etc. `name` is the (vector of) prefix(es) for the final PDF files. `quiet = TRUE` suppresses output when calling `Sweave()` and `texi2dvi()`. `edir` is the path to the exercise directory (defaulting to the current working directory). `tdir` is the temporary (by default) directory into which all files are copied and where `Sweave()` and `texi2dvi()` are called. `control` allows a few further control options (see also below).

Using these arguments, a set of exams can now be easily produced:

```
R> set.seed(1090)
R> sol <- exams(myexam, n = 5, dir = odir, template = c("exam", "solution"),
+    header = list(ID = getID, Date = Sys.Date()))
```

This takes the exam `myexam` (for which the corresponding Sweave files are all provided in the `exercises` directory of the package) and produces five exams (from the `exam.tex` LaTeX file) with associated solutions (from the `solution.tex` LaTeX file). The templates are both provided in the `tex` directory of the package and allow for specification of `\ID{}` and `\Date{}`. All output files are stored in `odir`

```
R> list.files(odir)

 [1] "exam1.pdf"     "exam2.pdf"     "exam3.pdf"     "exam4.pdf"
 [5] "exam5.pdf"     "metainfo.rda"  "solution1.pdf" "solution2.pdf"
 [9] "solution3.pdf" "solution4.pdf" "solution5.pdf"
```

and can now be easily inspected by the reader. In addition to the $2 \cdot 5$ PDF files, there is a file `metainfo.rda` containing the `save()`d "`exams_metainfo`" object which is also stored in object `sol` above. When evaluating an exam, this can be conveniently `load()`ed and individual results can be printed:

```
R> print(sol, 1)

exam1
    1. Multiple choice: abde
    2. t statistic: 0.188 (0.178--0.198)
    3. Prediction: 236.678 (236.668--236.688)
    4. Multiple choice: abde
    5. Multiple choice: d
```

```
R> print(sol, "exam5")

exam5
    1. Multiple choice: b
    2. Confidence interval: [128.746, 132.654] ([128.736--128.756,
        132.644--132.664])
    3. Multiple choice: be
    4. Multiple choice: abde
    5. Multiple choice:
```

For numeric results, the solution is displayed, possibly including tolerance limits (if non-zero). For multiple choice answers, the true statements (coded with letters a–e) are listed and the false ones are omitted. Consequently, if a multiple choice question has no true statements, no letters are displayed.

The appearance of multiple choice solutions when printing "`exams_metainfo`" objects in R or in the questionnaire in LaTeX can be modified using the `control` argument. This is specified by a named list with components `mchoice.print` and `mchoice.symbol`. Component `mchoice.print` is a list with elements `True` and `False` which each give the symbols used for printing true and false answers respectively. Component `mchoice.symbol` is a vector of length 2 with elements `True` and `False` which specify the symbols used in the answer fields of the questionnaire. An example for changing the printing is given by:

```
R> mycontrol <- list(mchoice.print = list(True = LETTERS[1:5], False = "_"))
R> (exams(myexam, n = 1, template = "exam", control = mycontrol))

exam1
    1. Multiple choice: AB_D_
    2. Confidence interval: [167.856, 171.944] ([167.846--167.866,
        171.934--171.954])
    3. Multiple choice: A__D_
    4. Multiple choice: ___D_
    5. Multiple choice: ___D_
```

5. Discussion

Package **exams** provides a framework for automatic generation of simple (statistical) exams and associated self-study materials. It is based on independent exercises in Sweave format which can be compiled in exams (or other collections of exercises) by providing one (or more) master LaTeX template(s). Because contributing to the pool of exercises just requires knowledge of Sweave and minimal markup for meta-information, **exams** facilitates joint development of lecture materials.

During 2007, **exams** was employed by about 10–15 lecturers at the Department of Statistics and Mathematics of WU Wien for jointly developing materials for the basic statistics lecture. Since Spring 2008, **exams** is used for generating collections of exercises and trial exams (both available prior to the actual exam) as well as the exams and associated solutions (which are

e-mailed to the students individually after correction of the test). Currently, exams are still corrected "by hand" but an automated procedure using optical character recognition (OCR) software is in preparation.

Acknowledgments

We are indebted to our colleagues at the Department of Statistics and Mathematics at WU Wien—in particular Regina Tüchler and Josef Leydold—for testing and challenging the code and making suggestions for improvement. This research was partially supported by the Austrian Science Foundation (FWF) under Hertha-Firnberg grant T351.

References

de Leeuw J (2001). "Reproducible Research: The Bottom Line." *Technical Report 2001031101*, Department of Statistics Papers, University of California, Los Angeles. URL http://repositories.cdlib.org/uclastat/papers/2001031101/.

Knuth DE (1984). *The TEXbook*, volume A of *Computers and Typesetting*. Addison-Wesley, Reading, Massachusetts.

Knuth DE (1992). *Literate Programming*, volume 27 of *CSLI Lecture Notes*. Center for the Study of Language and Information, Stanford, California.

Lamport L (1994). *LATEX: A Document Preparation System*. 2nd edition. Addison-Wesley, Reading, Massachusetts.

Leisch F (2002). "Dynamic Generation of Statistical Reports Using Literate Data Analysis." In W Härdle, B Rönz (eds.), "COMPSTAT 2002 – Proceedings in Computational Statistics," pp. 575–580. Physica Verlag, Heidelberg.

Leisch F (2008a). "Sweave FAQ." URL http://www.stat.uni-muenchen.de/~leisch/Sweave/.

Leisch F (2008b). *Sweave User Manual*. URL http://www.stat.uni-muenchen.de/~leisch/Sweave/.

Leisch F, Rossini AJ (2003). "Reproducible Statistical Research." *Chance*, **16**(2), 46–50.

Pilato CM, Collins-Sussman B, Fitzpatrick BW (2004). *Version Control with **Subversion***. O'Reilly. Full book available online at http://svnbook.red-bean.com/.

R Development Core Team (2008). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/.

Rossini AJ (2001). "Literate Statistical Practice." In K Hornik, F Leisch (eds.), "Proceedings of the 2nd International Workshop on Distributed Statistical Computing, Vienna, Austria," ISSN 1609-395X, URL http://www.ci.tuwien.ac.at/Conferences/DSC-2001/Proceedings/.

Affiliation:

Bettina Grün, Achim Zeileis
Department of Statistics and Mathematics
Wirtschaftsuniversität Wien
Augasse 2–6
A-1090 Wien, Austria
E-mail: Bettina.Gruen@wu-wien.ac.at, Achim.Zeileis@R-project.org
URL: http://statmath.wu-wien.ac.at/~gruen/,
http://statmath.wu-wien.ac.at/~zeileis/