sexy-rgtk: a package for programming RGtk2 GUI in a user-friendly manner

Damien Leroux[a] and **Nathalie Villa-Vialaneix**[a,b]

[a] INRA, UR875, MIAT
F-31326 Castanet Tolosan - France
damien.leroux@toulouse.inra.fr

[b] SAMM, Université Paris 1
F-75634 Paris - France
nathalie.villa@univ-paris1.fr

Keywords: Gtk2, RGtk2, GUI

There are many different ways to program Graphical User Interfaces (GUI) in R. [1] provides an overview of the available methods, describing ways to program R GUI with **RGtk2**, **qtbase** and **tcltk**. More recently, the package **shiny**, for building interactive web applications, was also released (the first version has been published on December, 2012).
The package **RGtk2** [2] is probably one of the most complete packages to program complex and highly customizable GUI. It is based on GTK2 (the GIMP Toolkit, http://www.gtk.org/), which is a multi-platform toolkit for creating Graphical User Interfaces. GTK2 offers a complete set of widgets and can be used to develop complete application suites working on Linux, Windows and Mac OS X. Although very flexible, each **RGtk2** interface results in a long script that has a counterintuitive syntax for most R users. For instance, the simple window of Figure 1[1] is obtained with the command lines provided in Figure 2 (left).

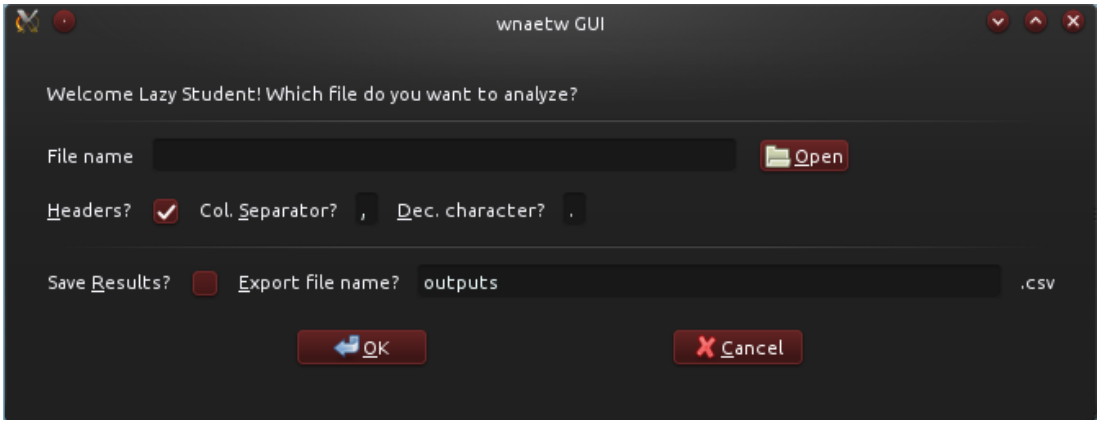


Figure 1: A simple GUI interface made with **RGtk2**.

One attempt to overcome the difficulty of the **RGtk2** syntax is the package **gWidgets** but, quoting its reference manual

> *"The excellent* **RGtk2** *package opens up the full power of the GTK2 toolkit, only a fraction of which is available though* **gWidgetsRGtk2**.*"*

[1] obtained on Kubuntu 12.04, Satanic Edition. The window's appearance differs depending on the OS and on the system's color configuration.

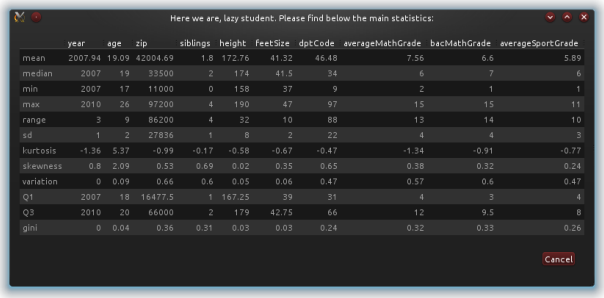
By automatically indexing all objects and methods available in **RGtk2**, we[2] developed a method for creating GTK2-based GUI, in a friendlier and more compact manner. Widgets are accessible with simple functions and options, as is more natural for a R language programmer. The window of Figure 1 is thus generated by the script provided in Figure 2 (right).

```
window <- gtkWindow()
window["title"] <- "wnaetw GUI"
vbox <- gtkVBoxNew(FALSE, 8)
vbox$setBorderWidth(24)
window$add(vbox)
hbox <- gtkHBoxNew(FALSE, 8)
vbox$packStart(hbox, FALSE, FALSE, 0)
label <- gtkLabelNew("Welcome Lazy Student!
  Which file do you want to analyze?")
hbox$packStart(label,FALSE,FALSE,0)
vbox$packStart(gtkHSeparatorNew(), FALSE, FALSE, 0)
hbox <- gtkHBoxNew(FALSE,8)
vbox$packStart(hbox, FALSE, FALSE, 0)
label <- gtkLabelNewWithMnemonic("_File name")
hbox$packStart(label,FALSE,FALSE,0)
filename <- gtkEntryNew()
filename$setWidthChars(50)
label$setMnemonicWidget(filename)
hbox$packStart(filename,FALSE,FALSE,0)
buttonOpen <- gtkButtonNewFromStock("gtk-open")
gSignalConnect(buttonOpen, "clicked", openFile)
hbox$packStart(buttonOpen,FALSE,FALSE,0)
# ... (script is cut at 1/3 of its length)
```

```
main <- Window(title="wnaetw GUI",
  contents=Rows(
    Label("Welcome Lazy Student! Which file do you want
        to analyze?"),
    br,HSeparator(), br,
    LabeledWidget("_File location ",
      Entry(use.name='filename',width.chars=50),
      mnemonic=T),
    Button(from.stock="gtk-open", on('clicked',run=
      chooseFile()), fill=F),br,
  # ... (script is cut at 1/3 of its length))
```

Figure 2: **RGtk2** (left) vs **sexy-rgtk2** (right).

This method has been used for recoding in a very short and simple manner the basic GUI of the (toy) package **wnaetw** [3]. Also a function has been developed to ease the use of the function `rGtkDataFrame`. A `data.frame` object `res` can thus be displayed in a window with the single command `DataFrame(res)` instead of having to define individually each column renderer. This feature is illustrated in Figure 3. The method should be released as a package next summer but the first scripts, without documentation, as well as the demo code, are available upon request.




```
performWmtw(main$filename$text, main$headers$active,
  main$sep$text,main$dec$text, main$quote$active,
  main$saveres$active,main$savename$text)
#...
performWmtw <- function(fn, headers, sep, dec, quote,
      save, sn) {
  # ... reading file
  res <- applyWmtw(my.data)
  ## GUI part
  resGUI <- Dialog(title="Here we are, lazy student.
    Please find below the main statistics:")
  DialogRows(resGUI,DataFrame(res, row.names=TRUE),br,
    pack(Button(from.stock='Cancel', on('clicked',run=
    resGUI$destroy())),expand=F, fill=T, padding=20,
    whence="end"))
```

Figure 3: Use case example for the function `DataFrame`.

References

[1] M. Lawrence and J. Verzani. *Programming Graphical User Interfaces in R*. CRC The R Series. Chapman & Hall, June 2012.

[2] L. Michael and T.L. Duncan. RGtk2: A graphical user interface toolkit for R. *Journal of Statistical Software*, 37(8):1–52, 2010.

[2]The authors did not contribute equally to the work: Damien developped the method to extract and interface **RGtk2** objects and methods, whereas Nathalie was the friendly useR and beta testeR.

[3]"What Nicolas's Teacher Wants", described at http://tuxette.nathalievilla.org/?p=885&lang=en.