About this book

R has a number of packages that provide a link between the R user and a graphical toolkit, such as `tcltk`, `RGtk2` and `rJava` (for Swing). In addition, an R user can interface with python or other external languages to provide access to graphical toolkits within those languages. This book is about writing graphical user interfaces (GUI) within R that do not rely on knowing an external language.

Currently, there is a range of graphical interfaces for R that are programmed within R. For example, several package authors have provided GUIs for their functions. Examples include `limmaGUI`, `caGUI`, `clustTool`, `Metabonomic`, . There are a few tools to automatically generate such GUIs, such as the `fgui` package and the `guiDlgFunction` function from the `svDialogs` package. Other authors have provided graphical interfaces to explore data sets, such as `playwith`, `latticist` and `aplpack`. While others have provided packages with GUIs aimed at allowing students to perform some simulation, e.g., `teachingDemos`. The `rattle` package provides an interface for several data mining operations. The `Rcmdr` package provides a menu- and dialog-driven interface to a wide range of R's functionality. There are several user-contributed plugins that extend the `Rcmdr`. Additionally, as R finds wider usage outside of academia, it is not uncommon for people who work in a team setting to desire an interface to their R code that allows non-R users access.

Such examples are all within the scope of this book. We set out to show that for many purposes adding a graphical interface to one's work is not terribly sophisticated nor time-consuming (atleast each piece isn't). This book does not attempt to cover GUIs for R that require knowledge of another programming language, although several such projects provide some of the most sophisticated interfaces. Examples are the `JGR` GUI or `iPlots` GUI written in Java through `rJava`, the `rkWard` GUI written within KDE, the `biocep` GUI written using Java and the `RServe` package, or even the Windows GUI that comes with R's Windows package.

The bulk of this text covers three different approaches to writing GUIs. The `gWidgets` package is covered first. This provides a common programming interface to some of the different graphical toolkits provided with R. This interface is much simpler (and less powerful) than the native toolkits, so is useful for a programmer who does not wish to invest too much time into perfecting their GUI. There are a few other packages that provide an R interface to a toolkit such as `rpanel` or `svDialogs`, but we focus on this one.

Next, we discuss the `RGtk2` package which provides a link between R and the cross-platform GTK+ libraries. These libraries are feature rich and used by several widely used projects. For the R user, they provide the most complete and modern-looking interfaces.

Finally, we discuss the `tcltk` package. This package provides the R user

access to the Tk libraries. Although not as modern as GTK+, these libraries come pre-installed with the Windows binary and so there are no installation issues for the average end-user. The bindings to Tk were the first ones to appear for R and the several of the GUI projects above, notably `Rcmdr`, use this toolkit.

In addition to chapters on each of these three packages, the book has an introductory chapter on GUIs, , and a chapter on web GUIs.

The text is written with the belief that much can be learned by studying examples, and so several examples are given. Some of these are meant as sketches of what can be done, while a few illustrate how to code actual useful interfaces. This text can't expect to cover all of the features of a graphical toolkit. For the `tcltk` and `RGtk2` packages, both underlying toolkits have well documented APIs.

This text comes with an accompanying package `ProgGUIinR`. This package includes the complete code for all the examples. In order to save space, some examples in the text have code that is not shown. The package provides the functions `browsegWidgetsFiles`, `browseTclTkFiles` and `browseRGtk2Files` for browsing the examples from the respective chapters. Additionally, this package will contain vignettes describing aspects that did not make it into the text.

This text was written with the `Sweave` package. To suppress superflous output an assignment to a variable named `QT` is made at times.