



# Plug-Ins para el paquete R Commander de R: una aplicación al cálculo de probabilidades

*Plug-ins for the R Commander package of R: an application to calculate probabilities*

Javier Osmar Artola García<sup>1</sup>

Pablo Gregori Huerta<sup>2</sup>

## Resumen

Res un prestigioso software para el análisis de datos estadísticos, y R Commander es una interfaz gráfica de R que proporciona menús y facilita su uso entre el público no especializado. En este trabajo se analizaron las opciones de menú que ofrece R Commander para el cálculo de probabilidades en modelos de variables aleatorias, y se construye un plug-in para mejorar dichas opciones de cara a estudiantes y usuarios que no estén familiarizados con las propiedades de las probabilidades. La programación de los plug-ins precisa del lenguaje de programación Tcl/Tk, que se puede utilizar a través de las librerías tcltk y tcltk2 de R.

Palabras clave: código fuente; lenguaje R; lenguaje tcl-tk; menú del paquete R Commander.

## Summary

R is a prestigious software for the analysis of statistical data, and R Commander is a graphical interface of R that provides menus and facilitates its use among the non-specialized public. In this work, it was analyzed the menu options offered by R Commander for the probabilities calculation in models of random variables, and built a plug-in to improve these options for students and users who are not familiar with the properties of the variables. The programming of the plug-ins requires the Tcl / Tk programming language, which can be used through the tcltk and tcltk2 of R libraries.

**Key Words:** source code; R language; tcl-tk language; R Commander Package menu.

## I. Introducción

R es una de las principales aplicaciones informáticas para el análisis estadístico de datos, por su gran calidad, por ser de licencia libre, y, por incorporar de manera casi inmediata los algoritmos de cálculo de las técnicas más recientes (en paquetes instalables que se ponen a disposición pública).

De manera más precisa, R es un conjunto integrado de programas para manipulación de datos, cálculo, gráficos y dispone de las características citadas a continuación.

<sup>1</sup> Máster en Matemática Computacional, Profesor de la Universidad de las Regiones Autónomas de la Costa Caribe Nicaragüense-Recinto Universitario Las Minas. Correo: [osmartola@gmail.com](mailto:osmartola@gmail.com)

<sup>2</sup> Doctor en Ciencias Matemáticas, Profesor Titular del Departamento de Matemáticas, Universitat Jaume I de Castellón, Instituto de Matemáticas y Aplicaciones de Castellón, España, correo: [gregori@uji.es](mailto:gregori@uji.es)

Recibido: 05/03/2017 - Aprobado: 12/06/2017

- Almacenamiento y manipulación efectiva de datos,
- Operadores para cálculo sobre variables indexadas (arrays), en particular en matrices,
- Una amplia, coherente e integrada colección de herramientas para análisis de datos,
- Posibilidades gráficas para análisis de datos, que funcionan directamente sobre pantalla o impresora.
- Un lenguaje de programación bien desarrollada, simple y efectiva, que incluye condicionales, ciclos, funciones recursivas y posibilidad de entradas y salidas (debe destacarse que muchas de las funciones suministradas con el sistema están escritas en el lenguaje R).

R no sólo se utiliza en el área de la Estadística, sino que, es bastante usado en diversas ramas del conocimiento científico. Para mencionar algunas, tenemos: la investigación biomédica, biología, sismología, psicología, programación lineal, física, ecuaciones diferenciales, matemáticas financieras, etc.

La ventaja principal descrita anteriormente, es que este software es de carácter libre bajo licencia GNU General Public Licence (Licencia Pública General) y lo podemos descargar en su sitio oficial <http://www.r-project.org> para cualquiera de los sistemas operativos Windows, (Mac) OS X, iOS, Linux y otros.

El uso más profesional y complejo de este software implica la programación en el lenguaje informático propio de R, pero la mayoría de usuarios necesita estadísticas (cálculos, tablas, figuras) que involucran la aplicación de una pequeña lista de funciones, con unos parámetros habituales. Por ese motivo, uno de los paquetes opcionales de R, el llamado R Commander, fabrica una ventana de menús para que el usuario no especializado pueda utilizar R de manera sencilla e intuitiva, sin necesidad de programar.

Los menús de R Commander son “ampliables” por usuarios avanzados, mediante la programación de unos paquetes llamados plug-ins, que aporten nuevas opciones de menú. El usuario no especializado sólo tendrá que instalar y cargar el plug-in, para disponer de esos nuevos menús.

En este trabajo se analizan las opciones de menú que ofrece R Commander para el cálculo de proba-

bilidades en modelos de variables aleatorias, se propone programar un plug-in para mejorar dichas opciones de cara a estudiantes y usuarios que no estén tan familiarizados con las propiedades de las probabilidades.

## II. Revisión de literatura

John Fox creó el paquete R Commander en Fox (2005), y representó la primera ocasión en que un programa reservado a expertos en Estadística y computación pudiera ser utilizado por usuarios de otras ramas científicas, con un manejo de ventanas al estilo de programas como SPSS (Statistical Package for Social Sciences de IBM), Statgraphics o SAS. La filosofía de R, de código abierto, permitió que diversos autores ampliaran los menús originales a partir del paquete original. Para mantener la modularidad, Fox redefinió la estructura del paquete original en Fox (2007), de modo que las extensiones se realizarían mediante paquetes independientes “normales” (de los que aportan nuevas funciones), sin tener que recompilar R Commander. Así, cada usuario tendrá los menús originales, más los que voluntariamente quiera instalar, separadamente, mediante plug-ins.

Para crear paquetes en R, la referencia fundamental es R Core Team (2015), la documentación oficial de R. De manera paralela, diversos autores han elaborado documentos simplificados para la creación de paquetes sencillos en Windows, entre los que podemos citar Leisch (2008), Rossi (2006) y Ripley (2008), y que permiten entender un proceso tan elaborado.

## III. Materiales y métodos

El presente estudio se realizó en la Universidad de las Regiones Autónomas de la Costa Caribe Nicaragüense, URACCAN - Recinto las Minas. Es una investigación tecnológica con carácter descriptivo, debido a que sus resultados son de mucha utilidad para investigadores, docentes y estudiantes de algún curso de modelos probabilísticos.

Se aplicaron técnicas, conocimientos y procesos para diseñar el menú de cálculo de probabilidades en modelos de variables aleatorias discretas y continuas. Además, se describe cada uno de los procesos realizados en el transcurso de la creación del menú.

Fue imprescindible el uso de computadora, software R versión 3.2.1, paquete R Commander de R, versión 2.1-7, y la versión 0.99.486 de RStudio.

La programación de las funciones se ha hecho en diferentes momentos, esto es, en el instante que es necesario integrar menú, y se programaron en lenguaje R, mediante ficheros con extensión R. Las ventanas emergentes de los menús y sus botones se programaron en lenguaje Tcl/Tk, con ayuda de las librerías tcltk y tcltk2.

## IV. Resultados y discusión

### Creación de paquetes Plug-In para R Commander

Antes de la versión 1.3.0 de R Commander, la única forma de extender los menús programados, era mediante la modificación del código fuente del propio R Commander (modificando el archivo Rcmdr-menus.txt, que es el código fuente de los menús, y añadiendo archivos con extensión .R con las nuevas funciones necesarias para activar los menús). Para ofrecer al público el R Commander con los nuevos menús, era necesario recompilar el código fuente, y crear un nuevo paquete, coincidente casi totalmente con el original R Commander.

Por ello, John Fox, su autor, redefinió la estructura de su paquete para facilitar la tarea de extensión de menús. Las extensiones serían realizables por medio de paquetes independientes que se pudieran instalar en R sin depender de R Commander. Una vez instalados, R Commander los detectaría, permitiría al usuario cargar el paquete, y con el reinicio de R Commander aparecerían los nuevos menús.

Para facilitar la búsqueda de extensiones de R Commander en CRAN, John Fox recomienda llamar a los paquetes plug-in con el nombre RcmdrPlugin.XXX, donde XXX representa la porción de nombre que su autor desee. De esta forma, al buscar en el listado, aparecerán todos juntos, tras el paquete Rcmdr (que es el correspondiente a R Commander).

### Creación de un paquete de R en Windows

Para empezar, se debe crear un paquete de R que contenga las funciones que van a ponerse en

marcha al activar cada uno de los nuevos botones de menú que se definan.

Como se comentaba anteriormente, la creación de paquetes en R es un proceso complejo (R Core Team, 2015). Dada la relativa simplicidad de los paquetes plug-in, podemos aprovechar las formas simplificadas de crear dichos paquetes, acudiendo a referencias como Leisch (2008), Rossi (2006) y Ripley (2008).

Por otro lado, el programa RStudio (RStudio Team, 2015), recrea una interfaz de usuario que aporta innumerables facilidades al programador de R. Entre ellas, simplifica y automatiza los pasos necesarios para la creación de nuevos paquetes de R, por lo que es ideal para la programación de estos plug-ins.

Así, para ilustrar el proceso, creamos el paquete que, por defecto, crea RStudio desde su interfaz.

- Menú: New project > New directory > R Package.
- Elegimos Tipo: Package, Nombre: Artola (el escogido para nuestro caso).
- Elegimos una carpeta donde crearlo.
- Create project.
- Se abre RStudio con un archivo ya creado y completo, llamado hello.R, en la ventana de edición, que contiene la definición de una función sencilla llamada hello ().
- Observamos que se ha creado la carpeta Artola en el lugar indicado del disco, con la estructura:
  - Carpeta man (con el archivo hello.Rd, de documentación de ayuda). Para cada archivo de extensión .Rd que incorpore la definición de una o varias funciones, debe documentarse un archivo de extensión .Rd que explique los detalles del uso de la función.
  - Carpeta R (con el archivo hello.R, script de la función definida). En esta carpeta deben incluirse todos los archivos de extensión .R que definen nuevas funciones.
  - Archivo DESCRIPTION (informativo del uso del paquete, su autor, etc.). Se debe editar con un editor de texto para informar de la utilidad, el autor, etc.

- Archivo `NAMESPACE`. Define el espacio de nombres para que las funciones no entren en conflicto con otras funciones de otros paquetes.
- Otros archivos propios de RStudio (el del proyecto y otro adicional).
- Se construye el nuevo paquete y se carga en la sesión (por medio de `Ctrl + Shift+B`).
- La función `hello ()` ya está disponible para que la invoque el usuario.
- Se chequea el paquete (`Ctrl + Shift + E`). Comprueba que no hay errores en los metadatos del archivo `DESCRIPTION` y `NAMESPACE`, que no hay errores de sintaxis o campos vacíos en la documentación de ayuda, que funcionan los ejemplos si se han proporcionado, etc. El resultado es `R CMD check succeeded`.
- Se hace un test del paquete (`Ctrl + Shift + T`), por las dependencias que puede tener de otros paquetes. El resultado sale `ERROR`, porque no encuentra una carpeta `testthat` para hacer las pruebas (pero no se trata de un error real del paquete, sino de que no se ha seguido el protocolo de testear).
- Menú `Build > Build binary package`: crea el archivo `Artola_o.1.zip`, que R puede instalar desde sus opciones de menú.
- Menú `Build > Build source package`: crea el archivo `Artola_o.1.tar.gz` que contiene el código fuente del paquete, es decir, la estructura completa de carpetas y archivos, que otro programador puede usar para desarrollar sus iniciativas a partir del paquete original.

El proceso consiste, entonces, en definir la función adecuada que tome los argumentos que el usuario de R Commander va a introducir en una nueva caja de opciones que proporcionará un nuevo botón de menú. Pasamos entonces a la creación del nuevo botón de menú.

### Creación de nuevas opciones de menú por plug-ins de R Commander

Tal como se describe en (Fox, 2007, p. 51), se debe crear y editar un archivo llamado `menus.txt` en la carpeta `\inst\etc` de la estructura de carpetas del nuevo paquete. Por el momento, dicha carpeta no existe y se crea.

Creamos un fichero de texto llamado `menus.txt` en la carpeta creada anteriormente (carpeta `\inst\`

etc), y lo editamos de tal modo que contenga los parámetros deseados en nuestro menú.

Las funciones a usar se copian en sendos archivos con extensión `.R` en la carpeta `\R` de la estructura de carpetas.

Finalmente se ejecuta el montaje de la librería con ayuda de RStudio y se instala el nuevo paquete.

En la siguiente sección se detalla la creación e instalación de un nuevo botón de menú para facilitar el cálculo de probabilidades a usuarios menos expertos.

### Menú para cálculo de probabilidades

Las opciones de menú que ofrece la distribución actual de R Commander, para el cálculo de probabilidades de variables aleatorias que siguen un modelo de distribución conocido, son una gran ayuda comparada con la opción de escribir en la consola de R el comando que realiza dicho cálculo. Sin embargo, pensamos que aún exigen, del usuario, un conocimiento más que intuitivo de las propiedades de la probabilidad. Entre todas las acciones sobre variables aleatorias, las que corresponden al cálculo de probabilidades, están referidas como `Probabilidades...` y `Probabilidades acumuladas`.

En las situaciones habituales, el usuario reconoce una situación aleatoria concreta, de la que quiere calcular cierto riesgo. El riesgo corresponde a la probabilidad de un suceso, que puede ser de tipo intervalo de valores, o un valor concreto. Por otro lado, la cantidad aleatoria se consigue asimilar a una de las variables aleatorias conocidas e implementadas en R, que corresponde a un modelo y a unos valores concretos de los parámetros de dicho modelo.

Por tanto, se busca una probabilidad del tipo  $P(X=0)$  o  $P(X \geq 1)$ , o  $P(10 < X \leq 20)$ , entre muchas otras posibilidades sencillas. Formas más complejas implican la combinación (adición) de estos tipos.

Para cierto tipo de usuario (aprendices de una titulación menos técnica, o profesionales), la elección del modelo y el planteamiento del suceso cuya probabilidad interesa medir, ya es todo un logro, pues consiguen modelizar matemáticamente su pregunta. Lo normal es que no dominen matizaciones más allá de dicha modelización, y por eso pensamos que las referencias que hace R Commander,

en sus menús, a las colas izquierda y derecha, superan la comprensión de parte de ese público. Y más todavía la diferencia entre la cola izquierda, que incluye el valor que la limita, y la cola derecha, que no lo incluye: una diferencia nada despreciable cuando se trabaja con una variable discreta.

Por ello, para ayudar al colectivo referido, pensamos en crear una interfaz más amigable para el cálculo de probabilidades de modo que, el menú, muestre una pestaña de “probabilidad”, donde se ofrece la opción

$$P([ ] < X \leq [ ])$$

y el usuario debe ingresar un valor numérico en el espacio en blanco dentro del corchete correspondientes a la probabilidad que le interesa calcular.

Una vez determinado el suceso cuya probabilidad se pretende calcular, en otra pestaña se ofrece el modelo de distribución de la variable en cuestión:

- Binomial de [ ] intentos con probabilidad de [ ]
- Poisson de media [ ] Exponencial de media [ ] (o bien ligada a un proceso de Poisson de intensidad [ ])
- Uniforme en el intervalo [ ]
- Normal de media [ ] y desviación típica [ ] (o varianza [ ])

De modo que el usuario debe ingresar valores para los parámetros correspondientes al modelo que sigue su variable de interés.

### El plug-in RcmdrPlugin.Artola

Para crear las opciones de menú deseadas, creamos un nuevo paquete de R, con la ayuda del software RStudio, que contiene una única función, capaz de calcular todo.

El lugar más indicado para esta nueva opción de menú es el botón de Distribuciones. Por ello, el archivo menus.txt que debe hacer en la carpeta \inst\etc tiene el código:

```
# Rcmdr menus para el paquete Artola
# last modified 2015-09-22 by J. Artola
# type menu/item operation/parent label command/menú activation install?
# Distributions
```

item distributionsMenu command “Calcula probabilidades (Artola)...” probArtola “” “”

El paquete contiene un único archivo probArtola.R, que incorpora la función que se invoca desde el menú, y que representa la caja con las opciones y calcula el valor solicitado.

Tal y como recomienda Fox (2007), se puede partir de una función del paquete Rcmdr o cualquiera de sus plug-ins, para adaptar las condiciones de la caja de opciones, ya que el lenguaje Tcl/Tk es muy complejo. Es lo que se ha hecho en este caso, y el resultado se halla en un archivo llamado prob.Artola.R, en el que contiene alrededor de 200 líneas de código. Por tal motivo no se ha mostrado aquí, pero puede consultarlo en el apéndice A - anexo - del trabajo disponible en [http://repositori.uji.es/xmlui/bitstream/handle/10234/158933/TFM\\_2015\\_ArtolaGarciaJO.pdf?sequence=1&isAllowed=y](http://repositori.uji.es/xmlui/bitstream/handle/10234/158933/TFM_2015_ArtolaGarciaJO.pdf?sequence=1&isAllowed=y).

Una vez que disponemos del paquete plug-in en un archivo de tipo .zip, iniciamos sesión en R, vamos a paquetes y nos trasladamos a instalar paquetes a partir de archivos zip locales, lo cargamos desde el directorio en el cual lo tenemos guardado y listo.

Iniciamos sesión en R Commander, vamos a Herramientas, Cargar plugins de Rcmdr y elegimos RcmdrPlugin.Artola. Una vez que aceptamos, nos pedirá reiniciar R Commander, así que también aceptamos y listo.

Ya aparece nuestro ítem de menú con el nombre Calcula probabilidades (Artola)... en el menú Distribuciones de Rcmdr, vea la Figura 1.



Figura 1: Menú Distribuciones de R Commander

Pinchando en el ítem Calcula probabilidades (Artola)... aparece la ventana Probabilidades en modelos, en la que directamente introducimos los datos para calcular probabilidades del tipo  $P(a < X \leq b)$  (vea la parte (a) de la Figura 2).



(a) Datos

(b) Opciones

Figura 2: Ventanas de Datos y Opciones

Para especificar el tipo de modelo y sus características, pulsamos sobre la pestaña Opciones, mostrándose la ventana que se ve en la parte (b) de la Figura 2.

Completamos las opciones del modelo para el cual queremos realizar el cálculo de probabilidades, y la salida la encontramos en la ventana de salida de R Commander.

En la ventana de instrucciones (script Windows) se visualiza el comando de R que realiza internamente los cálculos lanzados por la elección del menú.

A manera de ejemplo, suponga que deseamos calcular la probabilidad de que la variable aleatoria sea mayor que tres, pero menor o igual a cinco, en un modelo de probabilidad Binomial con siete intentos y probabilidad de éxito de 0.40. En la pestaña de Datos, colocamos un 3 a la izquierda y un 5 a la derecha

(parte (a) de la Figura 3), en opciones, nos ubicamos en Binomial: Intentos 7 y Prob. de éxito 0.40 (parte (b) de la Figura 3).



(a) Datos

(b) Opciones

Figura 3: Datos y opciones del ejemplo

Aceptamos y obtenemos en la ventana de salida de R Commander. En la ventana de instrucciones nos devuelve la instrucción del lenguaje R:

```
pbinom(q=5, size=7, prob=0.40) - pbinom(q=3, size=7, prob=0.40)
```

Así comprobamos que nuestro menú funciona correctamente. El lector podrá realizar otros cálculos y comprobarlos de manera manual.

## V. Conclusiones

La creación de nuevas opciones de menú para el paquete R Commander puede acercar el uso del R a un mayor público. Ya existen muchas aportaciones de menús especializados, pero en nuestro caso hemos cubierto una opción muy básica, que es el cálculo de probabilidades en modelos conocidos. Han quedado por implementar otras variantes de sucesos, que se irán incorporando en versiones posteriores.

En la misma línea, hay otras opciones, como son el cálculo de intervalos de confianza, que están cubiertos por menús del R Commander correspondientes al plug-in HH, donde las opciones pueden no ser muy claras para estudiantes o profesionales no científicos, y se podría aportar en ese sentido.

Agregar nuevos menús a R Commander es un proceso de dos partes, en las que una de ellas es sencilla. Para que aparezcan nuevas opciones en la barra de menús, basta con editar un pequeño archivo de texto, donde se indique en qué submenú se sitúa, a qué función llama, si se trata de una opción que lleve a ejecutar un comando, etc.

La parte más laboriosa es aquella en la que se define la ventana de opciones, en la que el usuario elige variables, o introduce valores de parámetros que pasarán como argumentos de la función a ejecutar. A pesar de disponer de un gran número de ejemplos en los que ya existen (gracias a que R es de código abierto), si se quiere hacer algo original, es preciso acudir a funciones de las librerías tcltk y tcltk2, donde están las funciones que diseñan las ventanas al detalle.

Se ha pretendido contribuir al software R con la creación de un paquete, aunque se trata de uno muy simple. De otra parte, se ha pretendido que la contribución pudiese ser útil a la comunidad, de modo que una vez completadas las opciones, se dará a conocer en el ámbito universitario, para que profesores y estudiantes puedan utilizarlo con la esperanza de mejorar rendimiento en el cálculo de probabilidades con R Commander.

## VI. Lista de referencias

- Fox, J. (2005). The R Commander: A Basic-Statistics Graphical User Interface to R. *Statistical Software*, 14 (9), 1-41.
- Fox, J. (2007). Extending the R Commander by “Plug-In” Packages. *R News*, 7 (3), 1-7.
- Leisch, F. (2009). *Creating R Packages: A Tutorial*. Recuperado de <https://cran.r-project.org/doc/contrib/LeischCreatingPackages.pdf>
- R Core Team (2015). *Writing R Extensions*. Recuperado de <https://cran.r-project.org/doc/manuals/r-release/R-exts.pdf>
- Ripley, R. M. (2012). *Making an R package*. University of Oxford. Recuperado de <http://portal.stats.ox.ac.uk/userdata/ruth/APTS2012/Rcourse104up.pdf>
- Rossi, P. (2006). *Making R Packages Under Windows: A Tutorial*. Recuperado de [http://www.math.ncu.edu.tw/~chenwc/R\\_note/reference/package/packages.pdf](http://www.math.ncu.edu.tw/~chenwc/R_note/reference/package/packages.pdf)
- RStudio Team (2015). *RStudio: Integrated Development Environment for R*. Boston, MA: RStudio, Inc. Recuperado de <http://www.rstudio.com>