R: Un lenguaje para análisis de datos y graficación

Ernesto Barrios Zamudio *

Departamento de Estadística
Instituto Tecnológico Autónomo de México

Noviembre 2010

Con el mismo título se publicó en 1996 el artículo donde Ross Ihaka y Robert Gentleman anunciaban R, su *software* estadístico. Basado en los lenguajes *S* y *Scheme* crearon un lenguaje para análisis de datos y graficación que hoy tiene un nivel de crecimiento y participación que se compara ya con el desarrollo de Linux.

1. El lenguaje y ambiente *S*

R toma mucho del lenguaje *S*, desarrollado por Rick Becker, John Chambers y colegas en Bell Labs en los años setentas y ochentas. Los creadores describen a *S* como un lenguaje y un ambiente de programación interactiva para el análisis de datos y graficación ([1, 3]). Aún mas, decían *"S le alienta a calcular, mirar los datos y programar de manera interactiva, con una respuesta rápida que le permite entender y aprender"*. Esta forma de interactuar con los datos, a diferencia de hacerlo por lotes, revolucionó la manera de hacer análisis estadísticos.

Más recientemente Chambers escribe: *"S es un lenguaje de programación y un ambiente para toda clase de cálculos que involucren datos. Tiene una meta simple: convertir ideas en* software*, rápidamente y de manera confiable"*([2]).

El objetivo principal del ambiente *S* es el alentar y permitir el buen análisis estadístico. Las particularidades de *S* van en esa dirección ([1]):

- *S* es sobre *datos*: provee de herramientas generales y fáciles de usar para la organización, almacenamiento y recuperación de varios tipos de datos.
- *S* es sobre *análisis*: es decir, cálculos necesarios para entender los datos. *S* provee de métodos numéricos y otras técnicas computacionales.
- *S* es sobre *programación*: usted puede programar funciones en el mismo lenguaje *S* aprovechando su poder y simplicidad. Si es necesario el lenguaje

* ebarrios@itam.mx

ofrece interfaces sencillas para comunicación con el sistema operativo o rutinas en *C* y *Fortran*.

- Especialmente, *S* es sobre *graficación*: ver a los datos de maneras interactivas, informativas y flexibles. Las capacidades de *S* están diseñadas para motivar la creación de nuevas herramientas e intentar nuevas ideas.

En 1998, la Association for Computing Machinery (ACM) reconoció a John Chambers por *"el sistema S, que ha alterado por siempre la manera en que la gente analiza, visualiza y manipula datos"*. Este mismo reconocimiento les fue otorgado años antes a los creadores del lenguaje *C*.

2. Antecedentes

A decir de Ihaka ([6]), *"R comenzó como un experimento para intentar usar métodos de Lisp para construir una pequeña base para probar ideas de como se debería construir un ambiente estadístico"*

Ross Ihaka y Robert Gentleman del departamento de Estadística de Auckland University, en Nueva Zelanda, estaban interesados en cómputo estadístico y ambos reconocieron la necesidad de un mejor ambiente de cálculo del que tenían. Ninguno de los productos comerciales les convencían por lo que decidieron desarrollar una herramienta propia.

Ihaka y Gentleman iniciaron su trabajo con un pequeño intérprete tipo-*Scheme* y para hacerlo útil debían incluir estructuras de datos que permitieran el trabajo estadístico y elegir la interface con el usuario [1]. Además querían la interface por comandos y puesto que ambos conocían el ambiente *S* les resultó natural utilizar una sintaxis parecida. Esta decisión más que nada decidió la dirección que tomaría el desarrollo de R. *Scheme* y *S* son similares en muchos aspectos y el adoptar la sintaxis de *S*, *"terminó produciendo algo muy parecido a* S*"*. En otras palabras, en el uso diario R y *S* son muy similares.

Sin embargo, R no es *S*. Las diferencias principales entre los lenguajes son resultado de la herencia de *Scheme*, fundamentalmente el manejo de la memoria y el acceso a las variables dependiendo de donde fueron definidas. Se distinguen también en el manejo del color, áreas de graficación, rotulación matemática, etc.

3. Desarrollo

Entusiasmados por el *software* producido y listos para usarlo en el laboratorio de cómputo estadístico, Ihaka y Gentleman colocaron algunas copias binarias en *Statlib* ([10]) en agosto de 1993 y lo anunciaron en la lista de distribución *s-news*. En respuesta recibieron comentarios de varios interesados sobre su *ambiente*. El más persistente fue Martin Mächler de ETH Zurich quien los animó a

[1] *Scheme* es un dialecto de *Lisp*, uno de los principales lenguajes utilizados en inteligencia artificial, pero en la definición de las variables es parecido a *Algol*, uno de los primeros lenguajes de aplicación científica ([12]).

liberar el código fuente como *software* libre. En junio de 1995 los autores deciden distribuir R bajo licencia general de la fundación GNU de software libre [5]).

Poco a poco se van recibiendo más reportes y opiniones del *software* que les resulta difícil mantener la comunicación por correo electrónico. En marzo de 1996 se crean tres listas de distribución de mensajes para anuncios, desarrollo y ayuda sobre R. Desde entonces la página principal del proyecto R es *http://www.r-project.org/*. Ese mismo año se publica el artículo donde se anuncia a R formalmente ([7]).

A partir de la creación de las listas de distribución la aportación de mejoras, sugerencias y aplicaciones se hizo tan frecuente que Ihaka, Gentleman y Mächler no respondían con la rapidez necesaria. Así, a mediados de 1997 se creó un grupo de desarrollo más amplio, *R-core*, el único autorizado a modificar el código fuente. Actualmente el grupo cuenta con 18 miembros, incluyendo al mismo John Chambers, y estadísticos como Brian Ripley, de Oxford University ([9]).

Algunas fechas importantes en el desarrollo de R: en febrero de 2000 sale finalmente la versión 1.0 de R; en 2001 se publica el primer número de R-News, revista electrónica dedicada a la discusión y anuncios de nuevos procedimientos y paquetes de R; la revista es reemplazada por R-Journal en el 2009.

Finalmente, para garantizar que R sea siempre *software* libre de código abierto, se creó R-foundation ([8]), que entre sus objetivos están:

1. Avanzar el proyecto R para cálculo estadístico que provea de *software* libre y código abierto para el análisis de datos y gráficas.

2. Guardar y administrar los derechos de copia de R y su documentación.

La figura 1 muestra esquemáticamente el desarrollo de R, lenguajes antecedentes y otros lenguajes de cómputo estadístico contemporáneos como son *S-Plus*, la implementación comercial del lenguaje *S* en *Windows*, y *XLisp-Stat*, lenguaje estadístico basado en *Lisp* y desarrollado por Luke Tierney.

R es entonces un proyecto abierto y por colaboración. Además de los cambios y adiciones a los paquetes básicos por parte de *R-core*, el resto de la comunidad contribuye con paquetes de aplicación general o particular. Paquetes como *ggplot2* de graficación en general, *exactRankTests* que ofrece el cálculo de distribuciones exactas para pruebas de rangos y permutaciones, *RMySQL* para la comunicación de R con el manejador de bases de datos *MySQL*, etc. Hay paquetes que por su utilidad, aunque no son parte de R básico, se incluyen en la distribución original de R. Por ejemplo, los paquetes *lattice* y *foreign*, para graficación el primero y lectura-escritura de datos almacenadas por otras aplicaciones estadísticas como *SPSS*, *SAS*, el segundo. Así también se incluye el paquete *MASS* por su amplia colección de procedimientos. Este paquete fue desarrollado como apoyo al libro MASS ([11]), pionero de la difusión del empleo de *S*, *S-Plus* y después R.

En lo que respecta a los paquetes por contribución, a principios de 2000 había unos 100 paquetes. Para 2005 alrededor de 500. En octubre de 2010 se cuentan con poco más de 2700 paquetes que van desde aplicaciones financieras, en sicología, química, enseñanza, mapas, etc. Paquetes que incluyen aplicaciones de

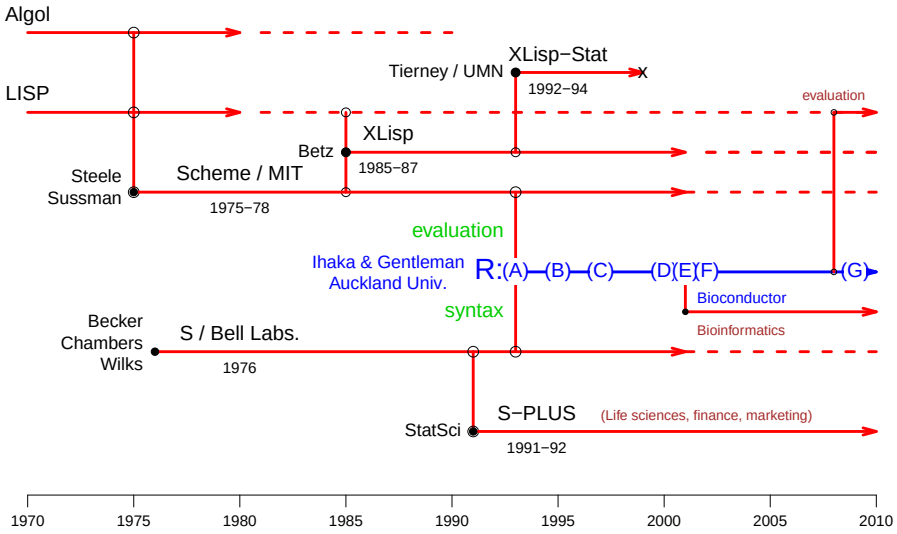

Figura 1: Genealogía de R: (A) 1993, Ross Ihaka y Robert Gentleman crean R en Auckland University; (B) 1995, R sea hace *código libre y abierto*; (C) 1997, se conforma el *R Development Core Team*; (D) 2000, la version 1.0 sale pública; (E) 2001, se publica el primer número de *R-News*; (F) 2002, se crea la *The R Foundation for Statistical Computing*; (G) 2009, *R-Journal* sustituye a *R-News*.

todas las áreas de la estadística: series de tiempo, muestreo, modelos lineales, generalizados aditivos, métodos bayesianos, métodos no paramétricos, aprendizaje estadístico, etc.

4. ¿Por qué usar R?

Si usted se inicia en la estadística y está por aprender un paquete de cómputo comience con R. En un principio requiere tiempo para familiarizarse con los comandos básicos pero eso sucede con cualquier paquete de cómputo. Piense por un momento el tiempo que ha invertido en dominar el procesador de palabras que utiliza.

Chambers opina que el tiempo que usted invierta en desarrollar y extender sus habilidades con el sistema R —lenguaje, paquetes y ambiente de programación, le redundará en la capacidad de cuestionar y contestar con confianza en el cálculo con datos. *"La exploración de los datos con las preguntas correctas y respuestas confiables son fundamentales para el análisis de datos"*([3]).

Por otro lado, ¿por qué cambiar si usted es ya un experto en aplicaciones estadísticas adecuadas? En cierta forma, no habría porque hacerlo, sobre todo si usted lleva a cabo regularmente cierto tipo de análisis estadístico y no piensa cambiarlo o extenderlo en el futuro. Pero aprender R también le ofrece la

posibilidad de cubrir con un mismo *software* la mayor extensión de áreas de la estadística, incluyendo aquellas en la frontera del desarrollo. Mucha de la investigación de punta se está presentando con apoyo de R vía la construcción de paquetes.

Además, utilizar R ofrece las siguientes ventajas ([4]):

- En el desarrollo de R están involucrados científicos de primer nivel tanto en el lado estadístico como de cómputo lo que garantiza un *software* de excelencia.
- R es sin duda el *software* estadística más empleado en investigación estadística, pero también en otras áreas como finanzas, medicina y sicología.
- El *R-core* ha creado una serie de procedimientos que ha hecho sencilla la participación de la gente aunque se tengan pocos elementos de cómputo. Basta saber un poco de R para poder colaborar con paquetes de su área de especialidad.
- El carácter de colaboración abierta por medio de paquetes se refleja en la posición que ocupa R en la frontera de la investigación.
- Por el mismo carácter de colaboración, la información de apoyo es muy extensa. La redes de comunicación incluye listas de discusión a varios niveles; documentos de distribución libre que explican a distintos niveles generalidades y detalles de R.
- R se distribuye bajo licencia GNU. El *software* es libre y de código abierto. Es decir, R es gratis y si lo desea, tiene disponible el código para modificarlo. Es el mismo caso para la mayoría de los paquetes disponibles.
- R está compilado y disponible para los sistemas operativos más populares: distintas versiones de *Linux*, *Mac OS X* y *Windows* 32 y 64 bits.
- R le ofrece un ambiente que permite llevar a cabo sus ideas sin limitarlas únicamente a los procedimientos incluidos en su aplicación estadística.

Finalmente y citando nuevamente a John Chambers ([3]):

> ¿Por qué concentrarse en R? Claramente y no por coincidencia R refleja la misma filosofía que evolucionó con el lenguaje *S* y la actitud hacia el análisis de datos en Bell Labs. Es relevante que *S* haya comenzado como un medio para que los investigadores estadísticos expresaran sus propios cálculos como apoyo en la investigación del análisis de datos y sus aplicaciones. Existe una conexión directa entre entre aquellos inicios y la gran comunidad que ahora usa R para implementar nuevas ideas en estadística resultando en el gran recurso que son los paquetes de R.
>
> Aunque hay mucho espacio para mejorar y para nuevas ideas, creo que por el momento R representa el mejor medio como *software* de calidad como apoyo para el análisis de datos.

5. ¿Cómo conseguir R?

The R Project for Statistical Computing tiene su página principal en *http://www.r-project.org/*. Ahí encontrará ligas o vínculos con todo lo relacionado a R. En particular, en el marco de la izquierda elija *CRAN* (The Comprehensive R Archive Network) y ahí tendrá que seleccionar el sitio-espejo (*mirror*) de donde descargar R y paquetes. Aquí en México, ITAM ofrece uno de estos servidores. A saber, *http://cran.itam.mx*. Si desea hacer este servidor el que utilice R por defecto, llame el comando

```
> options(repos='http://cran.itam.mx')
```

antes que instale o actualize algún paquete. O bien, incluya el comando en el archivo `.Rprofile`, que lo ejecutará automáticamente al inicio de cada sesión.

El primer cuadro en el cuerpo de la página principal de *CRAN* le ofrece versiones compiladas de R para distintos sistemas operativos. Por ejemplo, si su sistema operativo es *Windows* siga la liga y seleccione *base*. En la parte superior de la página se ofrece la liga para descargar R (actualmente versión 2.12.0) además de instrucciones de instalación.

Note que en la parte inferior del marco de la izquierda está la sección *Documentation*. Siga la liga de *Contributed*. Ahí encontrará documentos aportados por colaboradores externos al *R-core*. Hay documentos generales y específicos, a distintos niveles y en varios idiomas. Por ejemplo, encontrará *Introducción al uso y programación del sistema estadístico R* por Ramón Díaz-Uriarte, o bien, *R para Principiantes*, de Emmanuel Paradis, traducido por Jorge A. Ahumada.

Si no recuerda el *url* de *CRAN* simplemente haga la consulta "R" en internet y muy posiblemente la primer liga que arroje su buscador sea a la página de R.

6. Ejemplos

Una vez instalado exitosamente el *software* y para darse una idea del potencial del R básico, ejecute el comando `demo()` que le ofrecerá un menú de posibilidades. De ellas, seleccione por ejemplo,

```
> demo(graphics)
```

que desplegará distintas gráficas construidas en el momento y le mostrará el código correspondiente con el que se construyeron.

En esta sección se presentan además tres ejemplos sencillos: el primero donde se utiliza R como lenguaje de programación para determinar mediante simulación la probabilidad de cierto evento; el segundo, la simulación de muestras de dos distribuciones distintas y su comparación gráfica con la distribución normal; finalmente se presenta el ejemplo de una gráfica condicional incluido en R.

6.1. Lanzamiento de dados.

Suponga que se lanzan repetidamente dos dados y se cuenta la suma de las caras hacia arriba en cada lanzamiento. Determine la probabilidad de que la suma 4 sale antes que la suma 7. La respuesta es $p = 1/3$.

El problema se puede resolver numéricamente simulando el juego muchas veces. El siguiente código ensaya el juego $N = 20000$ veces. El primer comando define la *semilla* que permite se reproduzca el ejemplo aquí presentado. Elimine el comando o cambie la semilla y obtendrá distintos resultados pero todos cercanos a $p = 1/3$. En este ejemplo, se obtuvo 0.332.

Para calcular la probabilidad de que la suma 3 sale antes que la suma 7, modifique el código con `K <- 3`. El resultado teórico es $p = 1/4$.

Note que el código es parecido a otros lenguajes, excepto quizá, en la construcción de la función `extract` que a su vez hace uso de la función `sample` que permite simular distribuciones finitas con una probabilidad dada. En este ejemplo, las probabilidades de las distintas sumas.

Código:

```
================================================================================
set.seed(54321)
N <- 20000
K <- 4
output <- rep(NA,N)
n <- rep(1,N)
extract <- function() sample(seq(2,12),1,prob=c(1,2,3,4,5,6,5,4,3,2,1)/36)
tab <- rep(0,12); tab[7] <- -1; tab[K] <- +1
for(i in seq(N)) {
    k <- 1
    fin <- FALSE
    while(!fin) {
        k <- k+1
        out <- tab[extract()]
        fin <- ifelse(out==0,FALSE,TRUE)
    }
        output[i] <- out
        n[i] <- k
}
cat("\nNúmero de juegos simulados =",N,"\n")
cat("Prob{ Sale primero el",K,"que el 7 } =",sum(output>0)/N,"\n")
cat("Número promedio de lanzamientos =",mean(n),"\n")
================================================================================
```

Salida:

```
Número de juegos simulados = 20000
Prob{ Sale primero el 4 que el 7 } = 0.332
Número promedio de lanzamientos = 4.9813
```

6.2. Distribuciones.

Este ejemplo ilustra la generación de una muestra de tamaño $N = 2000$ de una población normal de media -5 y desviación estándar 3. Se reportan la mediana (-4.954) y la desviación estándar (3.012) de la muestra y se construye el histograma correspondiente que se muestra en el panel superior izquierdo de la figura 2. En el panel de la derecha se muestra la gráfica *cuantil-cuantil* de la distribución normal, también llamada *gráfica de probabilidad normal*. Si la muestra proviene de una población normal su correspondiente gráfica debe verse aproximadamente como una línea recta como se observa en la figura.

El panel de abajo a la izquierda muestra la curva de densidad de una muestra también de tamaño 2000 de una población ji-cuadrada con 3 grados de libertad (χ^2_3). La media y varianza muestrales se reportan como 2.985 y 6.080, cuando teóricamente son 3 y 6 respectivamente. El panel de la derecha muestra la correspondiente gráfica de probabilidad normal. Es claro lo alejado de los puntos a la recta, lo que se interpretaría que es poco probable que la muestra viniese de una población normal.

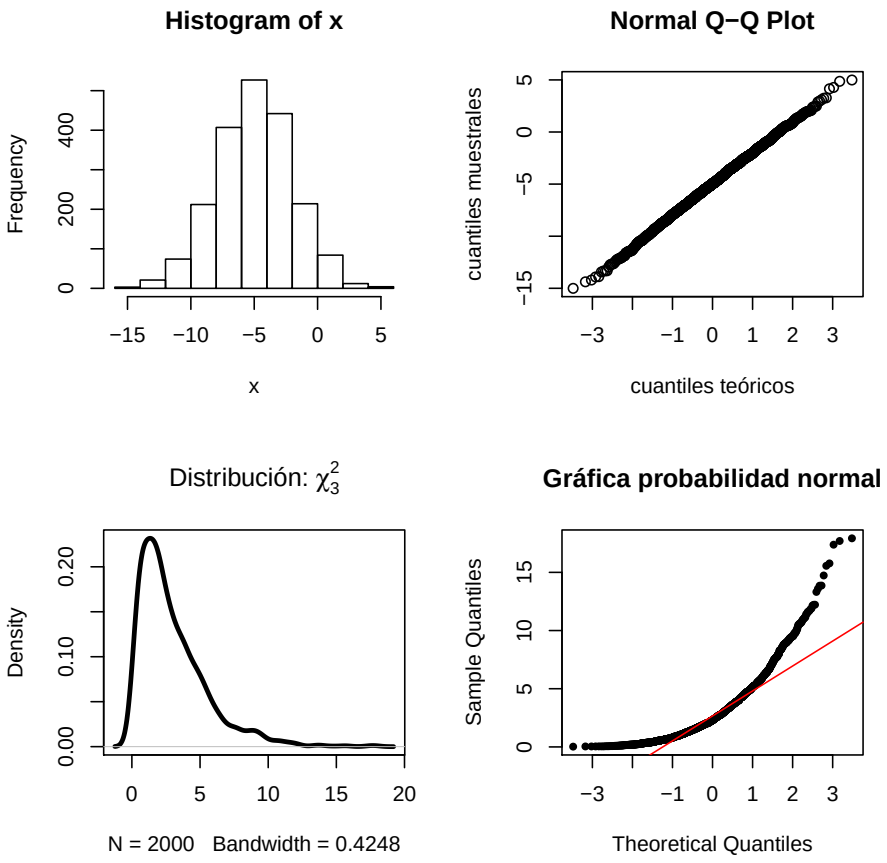

Figura 2: Histograma, densidad y gráficas de probabilidad normal para muestras de tamaño 2000: a) distribución normal estándar (parte superior); b) distribución χ^2_3 (parte inferior).

Observe el código con el que se generaron las cuatro gráficas. El comando `par(mfrow=c(2,2))` le indica a R que habrá 4 páneles en la misma ventana. Además se han utilizado algunas opciones que modifican el estándar. Fíjese por ejemplo en la notación matemática de la densidad de la distribución χ^2_3.

Código:

```
================================================================================
par(mfrow=c(2,2))
set.seed(12345)
N <- 2000
x <- rnorm(N,-5,3)
cat("Muestra normal:\n"); print(c(median=median(x),sd=sd(x)))
hist(x) qqnorm(x,xlab="cuantiles teóricos",ylab="cuantiles muestrales")
y <- rchisq(N,3)
cat("Muestra ji-cuadrada con 3 gl:\n");print(c(mean=mean(y),var=var(y)))
plot(density(y),lwd=3,main=substitute("Distribución:"*x,list(x=quote(chi[3]^2))))
qqnorm(y,main="Gráfica probabilidad normal",pch=20); qqline(y,col="red")
================================================================================
```

Salida:

```
Muestra normal:
   median        sd
-4.954043  3.012125
Muestra ji-cuadrada con 3 gl:
    mean      var
2.985078 6.080158
```

6.3. Gráficas condicionales.

Finalmente, con este ejemplo se ilustra la capacidad de graficación de R. Un simple comando construye la *gráfica condicional* que se muestra en la figura 3. Cada uno de los páneles inferiores muestra la asociación de la longitud (*long*) con la latitud (*lat*), *dados* distintos niveles de la variable profundidad (*depth*). La función `coplot` y los datos *quakes* son parte de R básico, disponibles con la instalación inicial de R.

Código:

```
================================================================================
coplot(lat ~ long | depth, data = quakes, pch = 21, bg = "green3")
================================================================================
```

7. Conclusiones

R es un lenguaje de alto nivel y un ambiente para el análisis de datos y graficación. Creado por Ross Ihaka y Robert Gentleman en 1993 su diseño sigue la sintaxis de *S* pero el manejo de memoria y la manera de evaluar lo hace más eficientemente como *Scheme*.

En el desarrollo actual de R colaboran investigadores de primer nivel estadístico y en computación.

Entre otras razones de porque aprender R se enuncian: a) es de excelente calidad; b) es libre y de código abierto; c) es un proyecto por colaboración por lo que hay mucho material de apoyo y ayuda; d) por lo mismo, hay una

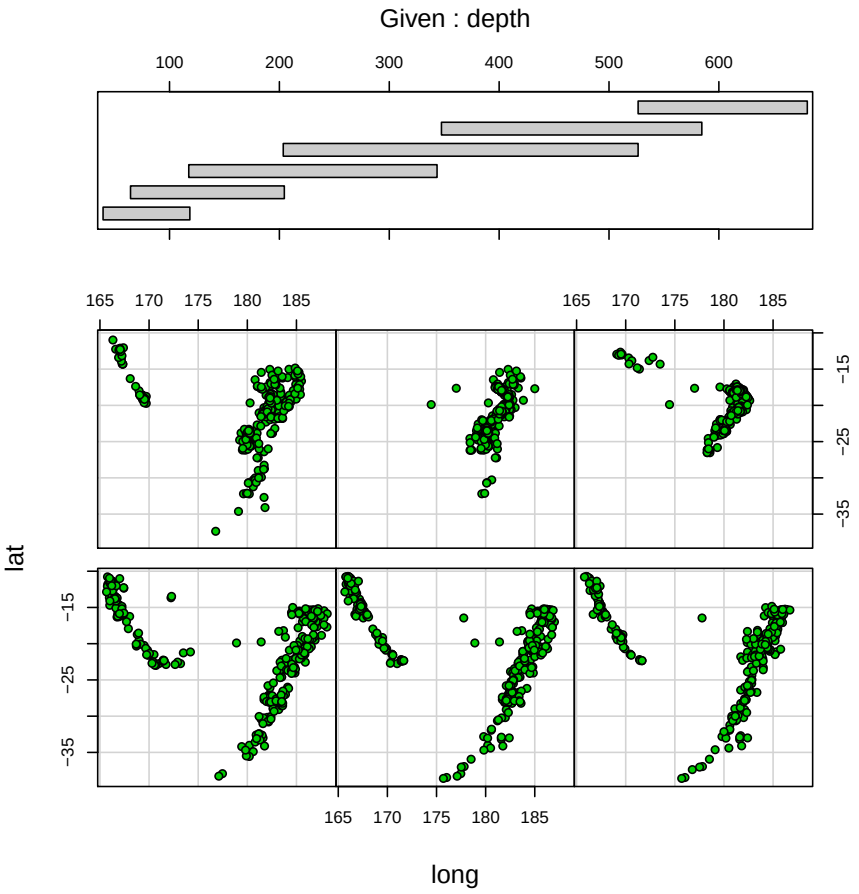

Figura 3: Gráfica condicional: asociación de latitud con longitud *dada* la variable profundidad en la base de datos *quakes* incluido en R.

gran variedad de paquetes por lo que posiblemente haya disponible lo que usted necesite; e) y si no lo hubiera, finalmente, el sistema le ofrece la facilidad para que usted construya su procedimiento y si así lo considera, contribuya al acervo de R.

Por último, R se enriquece con la colaboración de personas literalmente de todo el mundo. Lo mismo que le sucede al sistema operativo Linux y al procesador tipográfico LaTeX, utilizado para la escritura de esta nota. A todas ellas ¡Muchas Gracias!.

Referencias

[1] R. A. Becker, J. M. Chambers, and A. R. Wilks. *The New S Language*. Wadsworth&Brooks/Cole, Pacific Grove, CA., 1988.

[2] J. M. Chambers. *Programming with Data. A Guide to the S Language*. Spinger, New York, N. Y., 1998.

[3] J. M. Chambers. *Software for Data Analysis. Programming with R*. Spinger, New York, N. Y., 2008.

[4] M. J. Crawley. *The R book*. Wiley, Hoboken, N. J., 2007.

[5] GNU. The GNU General Public License, 2010. url:http://www.gnu.org/licenses (10/29/2010).

[6] R. Ihaka. R: Past and Future History, 1998. A Draft of a Paper for Interface 1998.

[7] R. Ihaka and R. Gentleman. R: A Language for Data Analysis and Graphics. *Journal of Computational and Graphical Statistics*, 3:299–314, 1996.

[8] R. Statutes of "The R Foundation for Statistical Computing", 2002. url: www.r-project.org/foundation/main.html. (28/Oct/2010).

[9] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2010. ISBN 3-900051-07-0.

[10] Statlib. Data, Software and News from the Statistics Community, 2010. url:http://lib.stat.cmu.edu (10/29/2010).

[11] W. N. Venables and B. D. Ripley. *Modern Applied Statistics with S*. Springer, New York, N. Y., 4th edition, 2002.

[12] N. Ziring. *Dictionary of Programming Languages*, 2010. url:http://cgibin.erols.com/ziring/dopl.html (10/29/2010).