

Capítulo 1

Introducción

1.1. ¿Que es R?

Si este es tu primer acercamiento a R, es muy probable que te cuestiones sobre la ventaja y la utilidad de R sobre otras paqueterías de estadística, como veremos adelante R es más que eso. La intención de este primer capítulo, es responder algunas dudas y animarte a que explores este software poderoso, que puede ser aplicado ampliamente en el procesamiento de datos en ciencias.

Empezaremos diciendo que R es un lenguaje intérprete (de programación) de distribución libre (Licencia GNU) y un ambiente para el cómputo estadístico y grafico. Este software corre en distintas plataformas Linux, Windows, MacOS, incluso en PlayStation 3. El término ambiente pretende caracterizarlo como un sistema totalmente planificado y coherente, en lugar de una acumulación gradual de herramientas muy específicas y poco flexibles, como suele ser con otro software de análisis de datos. El hecho que R sea un lenguaje y un sistema, es porque forma parte de la filosofía de creación (desde la codificación de S como veremos en la sección siguiente), como lo explica John Chambers, cito “Buscamos que los usuarios puedan iniciar en un entorno interactivo, en el que no se vean, concientemente, a ellos mismos como programadores. Conforme sus necesidades sean más claras y su complejidad se incrementa, deberían gradualmente poder profundizar en la programación, es cuando los aspectos del lenguaje y el sistema se vuelven más importantes”. Por esta razón, en lugar de pensar de R como un sistema estadístico, es preferible verlo como un ambiente en el que se aplican técnicas estadísticas. Por ejemplo, en este libro nos inclinaremos hacia el lado de la programación (lenguaje) más que tocar los aspectos estadísticos. Esto con la finalidad de ampliar la gamma de aplicaciones en el tratamiento de datos.

1.2. Historia de R y S

R fue creado en 1992 en Nueva Zelanda por Ross Ihaka y Robert Gentleman. La intención inicial con R, era hacer un lenguaje didáctico, para ser utilizado

en el curso de Introducción a la Estadística de la Universidad de Nueva Zelanda. Para ello decidieron adoptar la sintaxis del lenguaje S desarrollado por Bell Laboratories. Como consecuencia, la sintaxis es similar al lenguaje S, pero la semántica aparentemente es parecida a S, pero en realidad esta última es bastante diferente. A modo de broma Ross y Robert, comienzan a llamar “R” al lenguaje que implementaron, por las iniciales de sus nombres. Debido a que R es una implementación de S, a continuación daremos una breve reseña histórica de este lenguaje, para entender los fundamentos y alcances de R.

S es un lenguaje que fue desarrollado por John Chambers y colaboradores en Laboratorios Bell (AT&T), actualmente Lucent Technologies, en 1976. Este lenguaje, originalmente fue codificado e implementado como unas bibliotecas de fortran. Por razones de eficiencia, en 1988 S fue reescrito en lenguaje C, dando origen al sistema estadístico S, Versión 3. Con la finalidad de impulsar comercialmente a S, Bell Laboratories dio a StatSci (ahora Insightful Corporation) en 1993, una licencia exclusiva para desarrollar y vender el lenguaje S. En 1998 S ganó el premio de la Association for Computing Machinery’s Software System, y se liberó la versión 4, la cual es prácticamente la versión actual.

El éxito de S fue tal que, en 2004 Insightful decide comprar el lenguaje a Lucent (Bell Laboratories) por la suma de 2 millones de dólares, convirtiéndose hasta la fecha en el dueño. Desde entonces, Insightful vende su implementación del lenguaje S bajo el nombre de S-PLUS, donde le añade un ambiente gráfico amigable. En el año 2008, TIBCO compra Insightful por 25 millones de dólares y se continúa vendiendo S-PLUS, sin modificaciones. Sin embargo, lo fundamental de R no ha sufrido un cambio dramático desde 1998.

Regresando a R, luego de la creación de R (en 1992) dan un primer anuncio al público del software R en 1993. En el año de 1995 Martin Mächler, de la Escuela Politécnica Federal de Zúrich, convence a Ross y Robert a usar la Licencia GNU para hacer a R un software libre. A partir de 1997, R forma parte del proyecto GNU.

En 1996 se crea una lista pública de correos, sin embargo debido al gran éxito de R, los creadores fueron rebasados por la continua llegada de correos. Por lo que tuvieron que crear, en 1997, 2 listas de correos R-help y R-devel, que son las que actualmente funcionan. Además se consolida el grupo núcleo de R, donde se involucran personas asociadas con S-PLUS, con la finalidad de administrar el código fuente de R.

Fue hasta febrero de 29 del 2000, que se considera al software completo y lo suficientemente estable, para liberar la versión 1.0.

1.3. Algunas características importantes de R

El sistema R está dividido en dos partes conceptuales: 1) El sistema base de R, que es el que puedes bajar de CRAN; 2) y en todo lo demás. La funcionalidad de R consta de paquetes modulares. Donde el sistema base de R contiene el paquete básico que es requerido para correrlo y la mayoría de las funciones fundamentales. Los otros paquetes contenidos en la “base” del sistema incluye

a *utils*, *stats*, *datasets*, *graphics*, *grDevices*, *grid*, *tools*, *parallel*, *compiler*, *splines*, *tcltk*, *stats4*.

La capacidad de gráficos de R es muy sofisticada y mejor que la mayoría de los paquetes estadísticos. R cuenta con varios paquetes gráficos especializados, por ejemplo, hay paquetería para graficar, crear y manejar los shapefiles, para hacer contornos sobre mapas en distintas proyecciones, graficado de vectores, contornos, etc. También existen paqueterías que te permite manipular y crear datos en distintos formatos como netcdf, matlab, excel entre otros. Cabe señalar que además del paquete base de R, existen más de 4000 paquetes en CRAN (<http://cran.r-project.org>), que han sido desarrollados por usuarios y programadores alrededor del mundo, esto sin contar los paquetes disponibles en redes personales.

R es muy útil para el trabajo interactivo, pero también es un poderoso lenguaje de programación para el desarrollo de nuevas herramientas, por ejemplo rclimindex, cliMTA-R, etc. Otra ventaja muy importante es que tiene una comunidad muy activa, por lo que, haciendo las preguntas correctas rápidamente encontrarás la solución. Estas características han promovido que el número de usuarios en el área de las ciencias se incremente enormemente.

R al ser software libre lo hace un lenguaje atractivo, debido a que no hay que preocuparse por licencias y cuenta con la libertad que garantiza GNU. Es decir con R se tiene la libertad de: 1) correrlo para cualquier propósito, 2) de estudiar como trabaja el programa y adaptarlo a sus necesidades (acceso al código fuente), 3) de redistribuir copias, y 4) mejorar el programa y liberar sus mejoras al público en general.

R tiene una limitante que es importante mencionarla, debido a su estructura R consume mucho recurso de memoria, por lo tanto si se utilizan datos de tamaño enorme, el programa se alentaría o en el peor de los casos no podría procesarlos. Cuando se tenga problemas de lentitud al correr el código, en la mayoría de los casos tiene solución, teniendo cuidado de vectorizar el código; ya que esto nos permitiría particionarlo y aprovechar los equipos con multi núcleos.

1.4. Ayuda en R

R cuenta con una muy buena ayuda en el uso de funciones de manera muy similar al *man* de UNIX. para obtener información de cualquier función en específico, por ejemplo *lm*, el comando es:

```
help(lm)

# Una forma abreviada sería

`?`(lm)
```

Cuando se desea información sobre caracteres especiales de R, el argumento se debe encerrar entre comillas sencillas o dobles, con la finalidad que lo identifique como una cadena de caracteres. Esto también es necesario hacer para unas

cuantas palabras con significado sintáctico incluyendo al *if*, *for*, y *function*. Por ejemplo:

```
help("[[")

help("if")

# Si no utilizas las comillas R pensará que le faltan
# argumentos

# help(if)

# Una forma abreviada sería

`?`(lm)

v <- c(9, 8, 31)
sqrt(v)

## [1] 3.000 2.828 5.568

# El sin de 30, 45 y 60 grados: Primero se hace la
# conversion a radianes:
angulos <- c(30, 45, 60) * (pi/180)
angulos # En radianes

## [1] 0.5236 0.7854 1.0472

senos <- sin(angulos)
senos

## [1] 0.5000 0.7071 0.8660
```

Además de esto existe un foro muy activo donde seguramente podrás encontrar las respuestas apropiadas (Mailing Lists) en www.r-project.org.

R en las ciencias