

**INFORME LABORATORIO 2:  
INTRODUCCIÓN A R**

**DIEGO POLANCO BERRIOS, JUAN ROA CARVAJAL, DANY RUBIANO JIMENEZ**

Profesores:

- Mónica Villanueva Ilufi
- Felipe Bello Robles



# TABLA DE CONTENIDOS

<b>ÍNDICE DE FIGURAS.....</b>	<b>iv</b>
<b>CAPÍTULO 1. INTRODUCCIÓN.....</b>	<b>5</b>
1.1    MOTIVACIÓN Y ANTECEDENTES . . . . .	5
1.2    OBJETIVOS . . . . .	5
1.3    ORGANIZACIÓN DEL DOCUMENTO . . . . .	5
<b>CAPÍTULO 2. SOFTWARE R.....</b>	<b>7</b>
2.1    ÁREAS DE USO . . . . .	7
2.2    COMANDOS . . . . .	8
2.3    SCRIPT MAGIC04.DATA . . . . .	14
<b>CAPÍTULO 3. CONCLUSIONES.....</b>	<b>17</b>
<b>CAPÍTULO 4. BIBLIOGRAFÍA.....</b>	<b>19</b>

# ÍNDICE DE FIGURAS

Figura 2-1: Gráfico N°1 . . . . .	8
Figura 2-2: Gráfico N°2 . . . . .	9
Figura 2-3: Gráfico N°3 . . . . .	10
Figura 2-4: Gráfico N°4 . . . . .	11
Figura 2-5: Gráfico N°5 . . . . .	12
Figura 2-6: Gráfico N°6 . . . . .	13

# **CAPÍTULO 1. INTRODUCCIÓN**

## **1.1 MOTIVACIÓN Y ANTECEDENTES**

R es un lenguaje y entorno de programación especialmente desarrollado para enfocarse en el análisis estadístico. R fue desarrollado inicialmente por Robert Gentleman y Ross Ihaka, que pertenecen al Departamento de Estadística de la Universidad de Auckland, ubicada en Nueva Zelanda. Es un proyecto de software libre, es considerado la versión libre de otro programa llamado S y S-Plus. R y S-Plus -versión comercial de S- son, probablemente, los dos lenguajes más utilizados en investigación por la comunidad estadística, siendo además muy populares en el campo de la investigación biomédica, la bioinformática y las matemáticas financieras. A esto contribuye la posibilidad de cargar diferentes bibliotecas o paquetes con finalidades específicas de cálculo o gráfico. Debido a su condición de software libre se encuentra bajo las condiciones de la licencia GNU-GPL. La dirección web desde la que se puede tener acceso tanto a los archivos necesarios para su instalación como al resto de recursos del proyecto R es <http://www.r-project.org>.

Como complemento a R existe RStudio que es un IDE (entorno de desarrollo integrado) para R que además cuenta con licencia GPLv3. Puede ser utilizado en los principales sistemas operativos como Windows, Mac y Linux o también puede ser utilizado en línea vía RStudio server. Al igual que R es un software libre que tiene como objetivo ser una herramienta potente que logre soportar los procedimientos y técnicas necesarias para realizar análisis de calidad. RStudio pretende ser una herramienta sencilla e intuitiva para ser un entorno de trabajo amigable para el usuario tanto experimentado como nuevo en R.

## **1.2 OBJETIVOS**

En el presente documento se tiene como finalidad dar a conocer al lector las utilidades que tiene el software libre R enfocado en distintas áreas de las ciencias. Además, como trabajo introductorio en R se expone al lector las funcionalidades de distintos comandos perteneciente al lenguaje R en conjunto con los resultados obtenidos respectivamente en donde los valores obtenidos al usar el conjunto de datos de la base de datos de interés deben tener coherencia.

## **1.3 ORGANIZACIÓN DEL DOCUMENTO**

Para cumplir con los objetivos el presente documento distribuye su información de la manera siguiente: primero se encuentra información de R, donde se presenta la información considerada relevante para informar al lector sobre qué es R y las áreas en las que se utiliza. A continuación se realiza una descripción de los distintos comandos de interés en R y los resultados obtenidos de cada uno. Con esta descripción de R, de las distintas áreas de uso y de los resultados obtenidos y presentados en el presente documento, se realiza el análisis de interés el cual es presentado en la conclusión del presente informe.



## CAPÍTULO 2. SOFTWARE R

### 2.1 ÁREAS DE USO

Como ya es sabido R corresponde a un lenguaje de programación destinado principalmente al análisis estadístico de datos. Este lenguaje nos permite utilizar herramientas de test estadísticos, algoritmos de clasificación y agrupamiento, gráficas, entre otras. También permite que los usuarios puedan publicar paquetes, extendiendo su configuración básica y así contribuir en los proyectos de otros usuarios. Por utilidades como estas es posible encontrar análisis estadísticos desarrollados en R dentro de distintas disciplinas.(Christian Salas, 2008). Dentro de estas disciplinas o áreas podemos identificar:

**Ciencias:** Esta es una de las áreas donde más podemos apreciar el uso de esta herramienta, pues la investigación forma una parte fundamental en el desarrollo de ésta. Es por esto que la encontramos en campos como: inmunología, bioinformática, química, biología, histología, genética, embriología, anatomía, fisiología, patología, ingeniería biomédica, zoología, microbiología, análisis químico, entre otros.

**Matemática financiera:** Esta área se preocupa de estudiar las tasas de interés, donde están incluidos los estudios de créditos, inversiones, capitalizaciones, entre otras operaciones financieras.

**Ecología:** Estudia el como afectan las interacciones entre los organismos y su ambiente a la distribución o la abundancia de materia prima. Es por eso que podemos identificar como sub-areas a forestales, agrícolas, ambientales, botánica,etc.

**Docencia:** El uso académico de esta herramienta reside en aplicar conceptos y desarrollar un mayor análisis de datos en los alumnos de distintas universidades, colegios, institutos, etc.

Se pudo ver que el uso de esta herramienta es mayoritaria en el área de investigación y que su uso puede albergar disciplinas muy variadas. Esto se debe a que permite la integración de paquetes por parte de los usuarios, es por esto que cada vez se van añadiendo mas disciplinas al uso de esta herramienta.

## 2.2 COMANDOS

Generando un gráfico: Primero se generan valores aleatorios para la distribución normal y son asignados a un vector "x" e "y" mediante el comando **rnorm**. Luego se grafican los valores contenidos en las variables "x" e "y" mediante el comando **plot** (el gráfico correspondiente es el que se ve a continuación). Seguido se listan las variables que han sido creadas por el usuario mediante el comando **ls**. Finalmente se eliminan las variables "x" e "y" mediante el comando **rm** (proveniente del ingles remove).

```

-$ > x <- rnorm(50)
-$ > y <- rnorm(50)
-$ > plot(x, y)
-$ > ls()
-$ > rm(x, y)

```

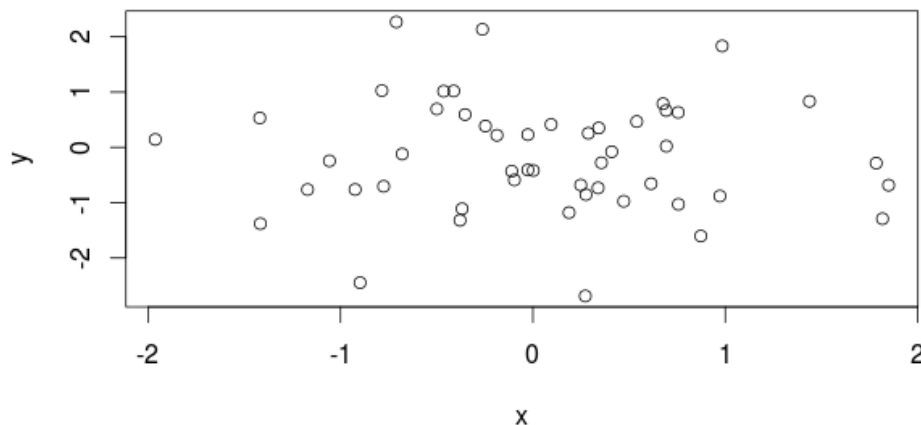


Figura 2-1: Gráfico N° 1

Aplicando funciones: Luego de crear los vectores "x" y "w" se crea la variable "hoja.de.datos" que proviene del comando **data.frame** que se utiliza para almacenar datos en tablas con propiedades similares a las matrices. Después de realiza un ajuste lineal mediante el comando **lm** a la variable "regr" y "regr.pon". Luego se utiliza el comando **summary** en la variable "regr" y "regr.pon" para obtener un resumen del ajuste lineal realizado. Lo siguiente que se realiza es aplicar la función **LOWESS** a la variable "regr.pon" en donde LOWESS es un método de regresión no parametrico que combina múltiples métodos basados en el vecino mas cercano. Finalmente se mediante **plot** se realiza un gráfico que es el que se muestra a continuación:

```

-$ > x <- -1 : 30vectorconvaloresdel1al30cada1
-$ > w <- -1 + sqrt(x)/2

```



```
-$ > hoja.de.datos <- data.frame(x = x, y = x + rnorm(x) * w)
-$ > hoja.de.datos
-$ > regr <- lm(y ~ x, data = hoja.de.datos)
-$ > summary(regr)
-$ > regr.pon <- lm(y ~ x, data = hoja.de.datos, weight = 1/w^2)
-$ > summary(regr.pon)
-attach(hoja.de.datos)
-$ > regr.loc <- lowess(x, y)
-$ > plot(x, y)
```

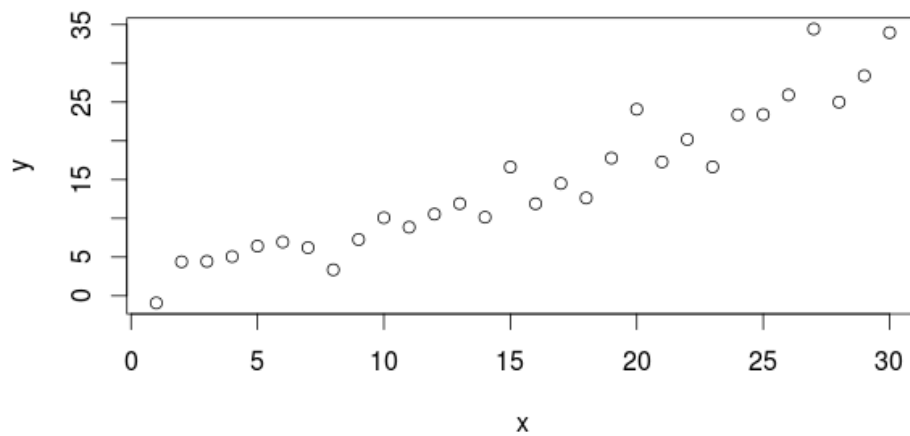


Figura 2-2: Gráfico N°2

Graficando: Primero mediante el comando **lines** se traza una linea que intenta unir todos los puntos que contenía el gráfico anterior. Luego tres veces se trazan rectas mediante el método **abline** que es utilizado para tener tres rectas de diferentes variables en donde en el caso de la variable "regr.pon" se le asigna un color (rojo en este caso). El gráfico resultante se muestra a continuación:

```
-$ > lines(x, regr.loc$y)
-$ > abline(0, 1, lty = 3)
-$ > abline(coef(regr))
-$ > abline(coef(regr.pon), col = "red")
```

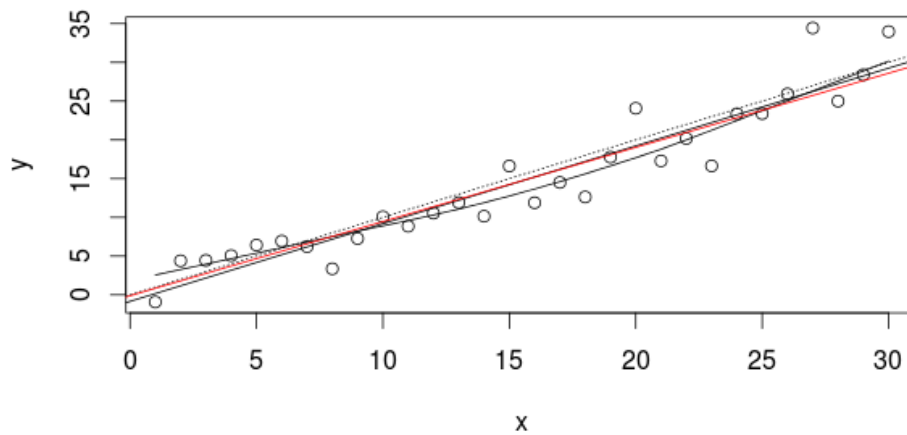


Figura 2-3: Gráfico N° 3

Después se realiza un nuevo gráfico en donde se observa la variable "regr" tanto en el eje vertical como en el horizontal pero en el eje horizontal se le aplico el comando **fitted** que es una función genérica que extrae valores ajustados de los objetos devueltos por las funciones de modelado. Y en el eje vertical se aplico el comando **resid** que es una función genérica que extrae residuos del modelo de objetos devueltos por las funciones de modelado. Finalmente se obtiene el gráfico expuesto a continuación:

```
-$ > plot(fitted(regr), resid(regr), xlab = " Predichos", ylab = " Residuos", main = " Residuos/Predichos")
```

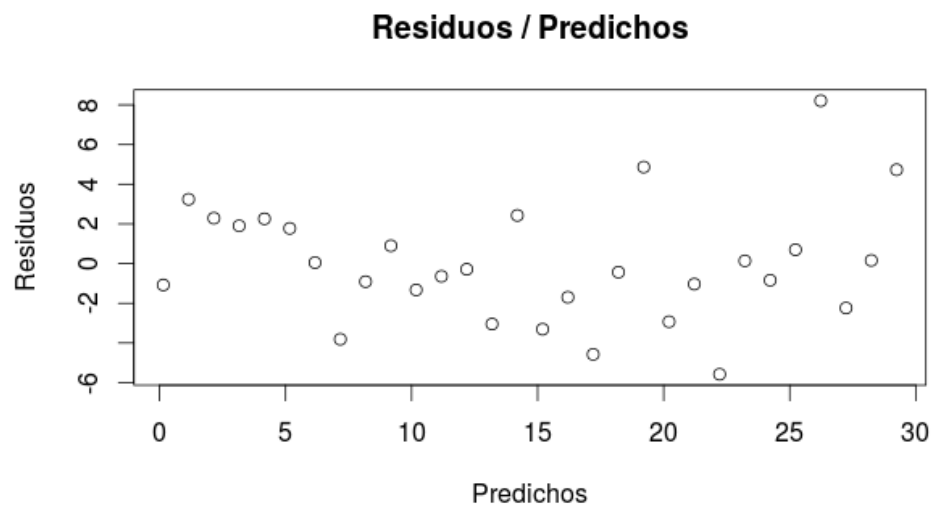


Figura 2-4: Gráfico N°4

Mas gráficos: Primero se realiza un gráfico ayudado por el comando **par** que se se utiliza para configurar o consultar los parámetros gráficos. Los parámetros se pueden ajustar mediante la especificación de argumentos en forma de etiqueta = valor , o pasando como una lista de valores etiquetados . De lo anterior se obtiene el siguiente grafico:

```
-$ > th <- seq(-pi, pi, len = 100); z <- -exp(1i * th)
-$ > par(pty = "s"); plot(z, type = "l")
```

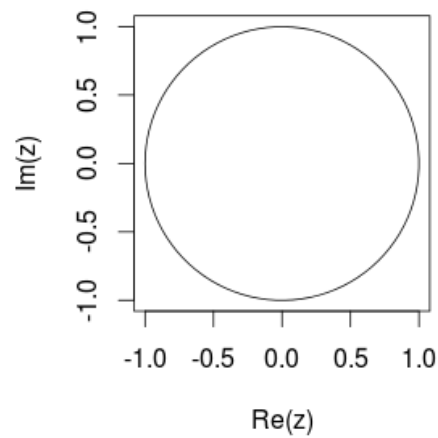


Figura 2-5: Gráfico N°5

En los siguientes comandos su objetivo es graficar nuevos valores obtenido de la variable "w". El comando que se puede destacar es **ifelse** que es un método que devuelve verdadero o falso según se cumpla o no una condicional que acompaña al mencionado comando. Finalmente se obtiene el gráfico expuesto a continuación:

```
-$ > w < -rnorm(100) + rnorm(100) * 1i
```

```
-$ > w < -ifelse(Mod(w) > 1, 1/w, w)
```

```
-$ > plot(w, xlim = c(-1,1), ylim = c(-1,1), pch = " +", xlab = " x", ylab = " y")
```

```
-$ > lines(z)
```

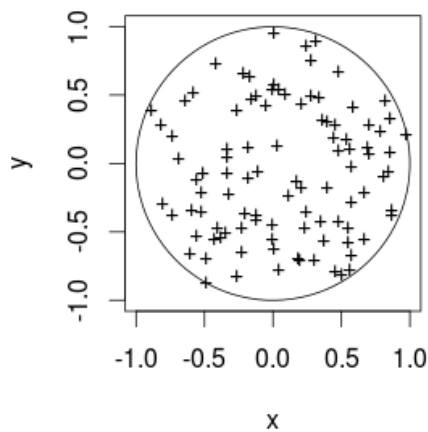


Figura 2-6: Gráfico N°6

## 2.3 SCRIPT MAGIC04.DATA

A continuación se pasará a explicar paso a paso como se importó la base de datos trabajada en el laboratorio 1 `magic04.data`, junto con la obtención de máximos y mínimos. Es importante aclarar que para la realización de éste, fue necesario buscar los comandos en el manual de "R". (R Project, 2015)

- Primero se cambió el directorio de trabajo, ubicándolo en el escritorio.

```
setwd(/home/usuario/Escritorio/)
```

- Luego se leyeron los datos del archivo `magic04.data` dándole un valor `FALSE` al "header", ya que el archivo no poseía los nombres de las columnas. A continuación con "col.names" se prosiguió a asignarle los nombres de las columnas ordenadamente según de como se presentan los valores de las variables en el archivo `magic04.data`.

```
data<-read.csv(header=FALSE,col.names=c("fLength","fWidth","fSize","fConc","fConc1","fAsym",  
"fM3Long", "fM3Trans","fAlpha","fDist","class"),"magic04.data")
```

- Posteriormente como es pedido en el laboratorio 2, se crea una tabla con los primeros diez datos de cada variable creada.

```
primerosdiez<-head(data,n=10)
```

- Después se asignan variables de igual nombre para cada columna creada para poder encontrar el mínimo y el máximo de cada variable, es por esto que se omite la variable `class`, la cual solo varia entre "g" y "h".

```
fLength<-data$fLength  
fWidth<-data$fWidth  
fSize<-data$fSize  
fConc<-data$fConc  
fConc1<-data$fConc1  
fAsym<-data$fAsym  
fM3Long<-data$fM3Long  
fM3Trans<-data$fM3Trans  
fAlpha<-data$fAlpha  
fDist<-data$fDist
```

- Se crea la columna atributo la cual contiene el nombre de las variables a estudiar a excepción de la variable `class`.

```
atributo<-c("fLength","fWidth","fSize","fConc","fConc1","fAsym","fM3Long", "fM3Trans","fAlpha","fDist")
```

- Se crea la columna mínimo, la cual contiene el mínimo valor que alcanzan las variables. Fue importante seguir con el mismo orden de las variables ingresadas.

```
minimo<-c(min(fLength),min(fWidth),min(fSize),min(fConc),min(fConc1),min(fAsym),min(fM3Long),  
min(fM3Trans), min(fAlpha),min(fDist))
```

- Se crea la columna máximo, la cual contiene el máximo valor que alcanzan las variables. Fue importante seguir con el mismo orden de las variables ingresadas.

```
maximo<-c(max(fLength),max(fWidth),max(fSize),max(fConc),max(fConc1),max(fAsym),max(fM3Long),  
max(fM3Trans),max(fAlpha),max(fDist))
```

- Se crea una tabla añadiendo las columnas establecidas anteriormente.

```
table<-data.frame(atributo,minimo,maximo)
```

- Finalmente se remueven del espacio de trabajo las variables utilizadas.

```
rm(fLength,fWidth,fSize,fConc,fConc1,fAsym,fM3Long,fM3Trans,fAlpha,fDist,atributo,minimo,maximo)
```





## **CAPÍTULO 3. CONCLUSIONES**

Luego de realizado este informe se puede decir que se cumplió con el objetivo de averiguar acerca de las utilidades del software R y disciplinas en donde se desarrolla. Se utilizaron también los comandos establecidos en la presentación del laboratorio 2, para el mayor entendimiento de R y RStudio. Donde se realizaron distintos tipos de gráficos y tablas, las cuales no presentaron un mayor problema.

Posteriormente se prosiguió a realizar una tabla que mostrara los valores mínimos y máximos de nuestra base de datos (proveniente de MAGIC GAMMA TELESCOPE data set) y otra tabla que nos arrojara los 10 primeros datos de cada variable. En esta etapa surgieron problemas al momento de importar la base de datos, pero fueron prontamente solucionados, ya que se recurrió al manual que se encuentra en la página web de R. Finalmente podemos decir que adquirimos los conocimientos básicos de manejo de R y RStudio, por lo que nos encontramos mejor preparados para continuar el análisis del respectivo conjunto de datos en experiencias posteriores.



## **CAPÍTULO 4. BIBLIOGRAFÍA**

Christian Salas. (2008). ¿por qué comprar un programa estadístico si existe r?. Recuperado desde [http:](http://www.scielo.org.ar/scielo.php?script=sci_arttext&pid=S1667-782X2008000200007)

[//www.scielo.org.ar/scielo.php?script=sci\\_arttext&pid=S1667-782X2008000200007](http://www.scielo.org.ar/scielo.php?script=sci_arttext&pid=S1667-782X2008000200007)

R Proyect. (2015, Agosto). R data import/export. Recuperado desde <https://cran.r-project.org/doc/manuals/>

[r-release/R-data.html](https://cran.r-project.org/doc/manuals/r-release/R-data.html)