Diplomado de Big Data,
Introducción a R

Índice

1. Instalación **2**

2. Instalación de paquetes **2**

3. Operaciones básicas **3**

4. Estadística **5**

5. Visualización **6**

6. Crear proyectos **9**

1. Instalación

R es un software para estadística computacional y graficos que es completamente gratis. Para descargarlo, sigue los siguientes pasos:

1. Abre el siguiente link https://www.r-project.org/.
2. Haz click en "CRAN" en la sección "Download" del lado izquierdo de la página.
3. Busca tu país y haz click en el primer link (estos son distintos repositorios).
4. En el primer recuadro "Download and Install R" haz click en tu sistema operativo.
5. En el caso de Windows, haga click en "base" para descargar el programa base. Luego, haga click en "Download R x.x.x for Windows", donde "x.x.x" es la versión actual de R. Para el caso de Mac OS, en la sección "Files" haga click en la versión mas reciente de R.
6. Una vez descargado, elija español e instale con las especificaciones que vienen por defecto.

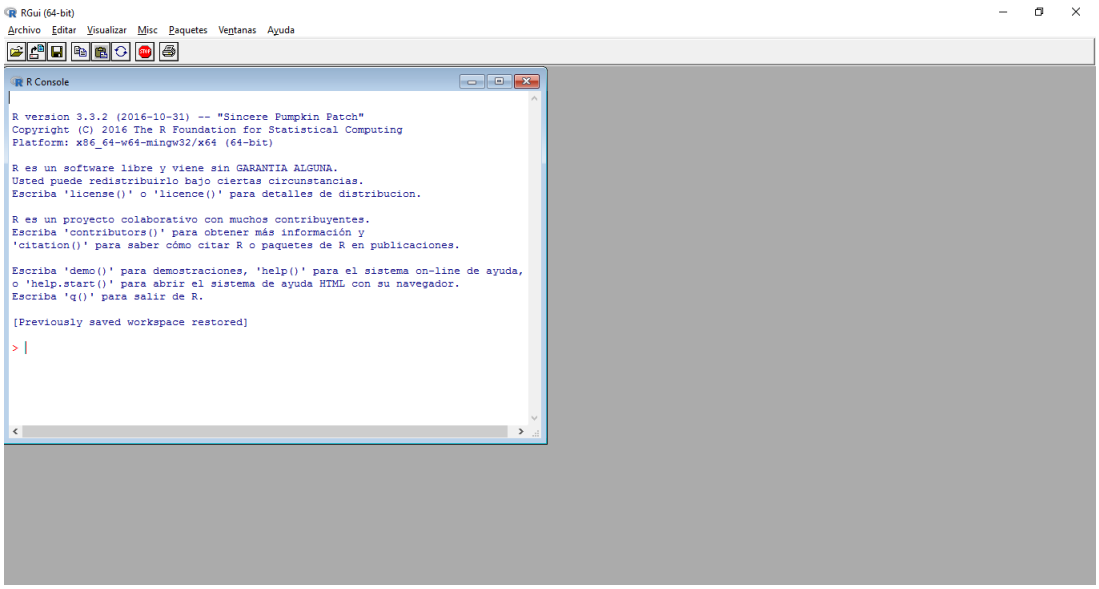
OBS: En el caso de Linux, se aconseja descargar R estudio.

2. Instalación de paquetes

Como R es un programa abierto, muchas personas y grupos de estudios han generado paquetes para poder hacer mas fácil la utilización de métodos estadísticos en R. Para poder usar un paquete, es necesario saber instalarlo y llamarlo desde la sesión actual en R. Para poder instalar un paquete nuevo siga los siguientes pasos:

1. Abra R desde el acceso directo en el escritorio.

   Una vez abierto R se vera de la siguiente manera



2. En el menú superior haga click en "paquetes", ubicado en la quinta posición.
3. Haga click en "instalar paquete(s) ....".
4. Se le abrirá una nueva ventana para que elija el *mirror* que es de donde se descargaran los paquetes, nuevamente busque y elija alguno de su país.

5. Al seleccionar el *mirror*, se le abrirá una nueva ventana con todos los paquetes disponibles, busque el que desea instalar, selecciónelo y haga click en "OK".

6. Si es la primera vez que instala paquetes, le preguntara para hacer una librería en su computador, responda que si. Luego, le preguntara por la dirección, responda que si a menos que quiera cambiarla.

3. Operaciones básicas

R funciona como una consola interactiva, osea que uno le entrega comandos y los ejecuta de inmediato. Estos comandos se ingrersan en la ventana llamada consola, que se debe seleccionar y al escribir el texto se colocará donde debe ir:

```
>
```

Por ejemplo, para hacer aritmética, basta escribir la operación deseada y presionar `enter`:

```
> 2+2
```

Se definen variables de manera intuitiva, y no hay complicaciones con el tipo de datos. Ojo que esta flexibilidad es un arma de doble filo, pues se puede estar trabajando con datos de tipo distinto al que uno cree, lo que puede generar muchos errores:

```
> a = 4      # Definir 'a' como una variable con el número 4
> a = 2+2    # Lo mismo
> a = "hola" # Definir 'a' como el texto "hola"
```

Todo lo que va desde el símbolo `#` es un comentario, osea que R lo ignora. Solo sirve para que después el código sea más fácil de leer. Podemos ver el contenido de la variable simplemente ejecutándola:

```
> a
[1] "hola"
```

Los objetos matemáticos usados son los vectores y matrices. Primero, los vectores se pueden definir de varias maneras, y lo mismo vale para el acceso a sus elementos.

```
>x=c(2,7,5) # Definir un vector con tres números
>x
[1] 2 7 5
> y=seq(from=4,length=3,by=3) # Definir un vector con tres números
> y
[1]  4  7 10
> x+y # Las operaciones entre vectores se realizan término a término
[1]  5  9 13
> x/y
[1] 0.2500000 0.2857143 0.3000000
> x^y
[1]     1   128 59049
> x[2] # Los vectores se enumeran desde el 1, osea que x es: 1->2, 2->7, 3->5
[1] 2
> 2:3 # Esto entrega simplemente una lista de números desde 2 hasta 3
[1] 2 3
> x[2:3] # Se pueden usar listas de índices para acceder a dichos elementos
[1] 2 3
> x[-2] # También se pueden sacar elementos
[1] 1 3
```

```
> x[-c(1,2)]
[1] 3
```

Por lo general las matrices se generan a través de vectores o con comandos especiales que las generan.

```
> v1 = c(1,2,3,4) # Creamos tres vectores
> v2 = c(2,3,4,5)
> v3 = c(3,4,5,6)
> mat1 = matrix(c(1,2,3,4,5,6), 3,2) # Ojo con la forma en que ordena el vector, lo pone por columna
> mat2 = cbind(v1,v2,v3) # Los agrupa como columnas
> mat3 = rbind(v1,v2,v3) # Los agrupa como filas
> mat1
     [,1] [,2]
[1,]    1    4
[2,]    2    5
[3,]    3    6
> mat2
     v1 v2 v3
[1,]  1  2  3
[2,]  2  3  4
[3,]  3  4  5
[4,]  4  5  6
> mat3
   [,1] [,2] [,3] [,4]
v1    1    2    3    4
v2    2    3    4    5
v3    3    4    5    6
> z=matrix(seq(1,12),4,3) # Podemos usar el comando sea que vimos antes
> z
     [,1] [,2] [,3]
[1,]    1    5    9
[2,]    2    6   10
[3,]    3    7   11
[4,]    4    8   12
> z[2,2] # Se accede a los elementos de la matriz por índices, igual que los vectores
[1] 6
> z[3:4,2:3]
     [,1] [,2]
[1,]    7   11
[2,]    8   12
> z[,2:3] # Para usar todos los elementos asociados a un índices, basta con dejarlo en blanco
     [,1] [,2]
[1,]    5    9
[2,]    6   10
[3,]    7   11
[4,]    8   12
> z[,1]
[1] 1 2 3 4
> z[,1,drop=FALSE] # Ponemos drop=FALSE para que el objeto que entrega sea una matrriz, no un vector
     [,1]
[1,]    1
[2,]    2
```

```
[3,]    3
[4,]    4
> dim(z)
[1] 4 3
```

Acá nos encontramos con el primer caso en que el tipo de dato es importante. Solo las matrices tienen dimensiones, por lo que pedirle dimensiones a un vector dará un resultado inesperado:

```
> dim(z[,1])
NULL
> dim(z[,1, drop = FALSE])
[1] 4 1
```

Se puede ver que por defecto R dice que la dimensión de un vector es `NULL`, y en cambio si hacemos que el objeto que entrega sea de nuevo una matriz, ahí sí puede ver su dimensión. Se puede veer el tipo de dato de una variable con el comando `class`:

```
> class(z[,1])
[1] "integer"
> class(z[,1,drop=FALSE])
[1] "matrix"
```

En caso de dudas, R posee una amplia documentación interna. A esta se accede con `?` o `help()` y luego el comando sobre el que se desea leer:

```
>?class
>help(class)
```

Además, se pueden borrar todas las variables almacenadas con:

```
> rm(list=ls())
```

4. Estadística

Para esta parte, cargaremos una de las tantas bases de datos que R trae por defecto llamada `iris`. Esta contiene información sobre tipos de rosas. Se puede ver una descripción más detallada con el comando de ayuda:

```
> ?iris
```

Es posible también cargar datos desde archivos de texto (.txt y .csv) a una variable llamada `datos` con

```
> datos = read.table(file.choose(), header=T)
```

donde la instrucción `header=T` dice que la primera fila del archivo contiene los nombres de las columnas de datos.

Notemos primero que los datos se almacenan en un tipo especial llamado `data frame`, veamos los primeros datos y un resumen de ellos:

```
> class(iris)
[1] "data.frame"
> names(iris) # Entrega los nombres de las variables en un vector
[1] "Sepal.Length" "Sepal.Width"  "Petal.Length" "Petal.Width"  "Species"
> head(iris) # Muestra las primeras entradas de datos
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1         3.5          1.4         0.2  setosa
```

```
2          4.9         3.0          1.4         0.2  setosa
3          4.7         3.2          1.3         0.2  setosa
4          4.6         3.1          1.5         0.2  setosa
5          5.0         3.6          1.4         0.2  setosa
6          5.4         3.9          1.7         0.4  setosa
> summary(iris)
  Sepal.Length    Sepal.Width     Petal.Length    Petal.Width           Species
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100   setosa     :50
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300   versicolor :50
 Median :5.800   Median :3.000   Median :4.350   Median :1.300   verginica  :50
 Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500
```

Los `data frame` se pueden acceder de la misma forma que una matriz, por lo que podemos extraer una columna cualquiera, guardarla como un nuevo vector y sacar algunos indicadores:

```
> columna = iris[,2] # Extraemos la segunda variable
> columna[1:20] # Vemos los primeros 20 datos
  [1] 3.5 3.0 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 3.7 3.4 3.0 3.0 4.0 4.4 3.9 3.5
 [20] 3.8
> mean(columna) # Promedio
[1] 3.057333
> median(columna) # Mediana
[1] 3
> var(columna) # Varianza
[1] 0.1899794
> sd(columna) # Desviación estándar
[1] 0.4358663
> cor(columna, iris[,2]) # Correlación entre dos variables
[1] 1
```

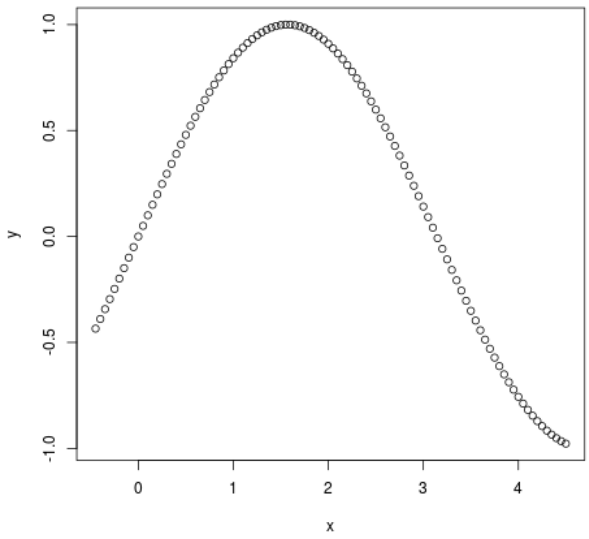
5. Visualización

Para esta sección, seguiremos usando la base de datos `iris` que viene en R. Para hacer la sección más independiente, cargamos primero una columna de datos

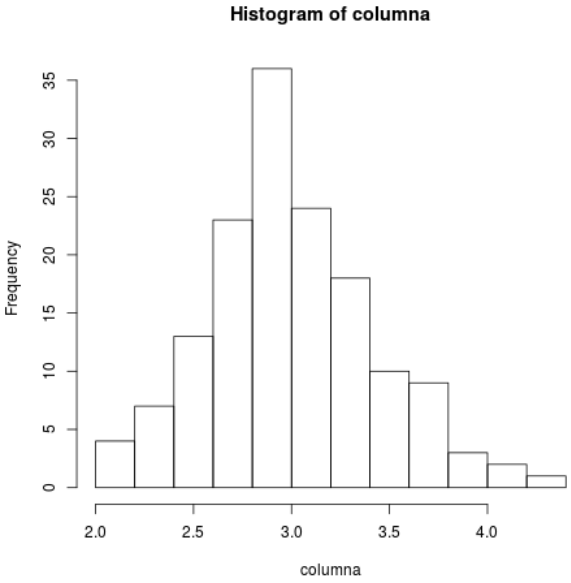
```
> columna = iris[,2]
```

y sobre ella haremos los gráficos.

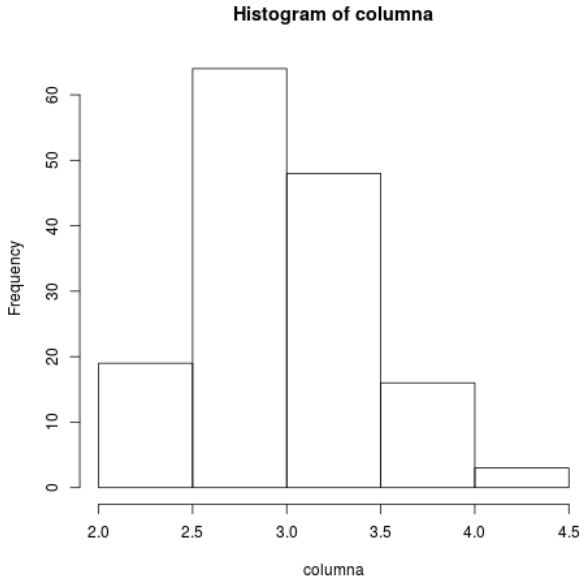
```
> x = 1:100/20 - 0.5
> y = sin(x)
> plot(x,y) # Se pueden graficar un conjunto de puntos con las coordenadas x,y
```



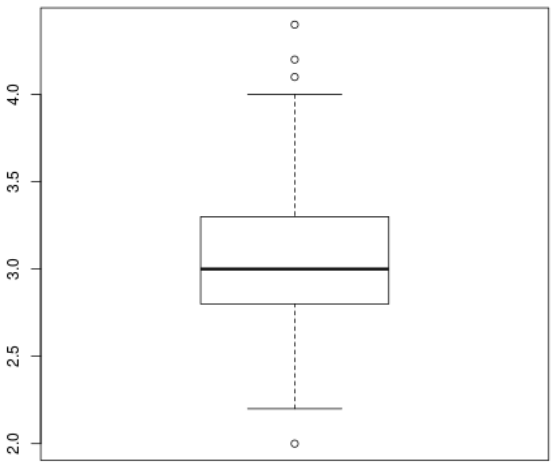

```
> hist(columna) # Histograma
```



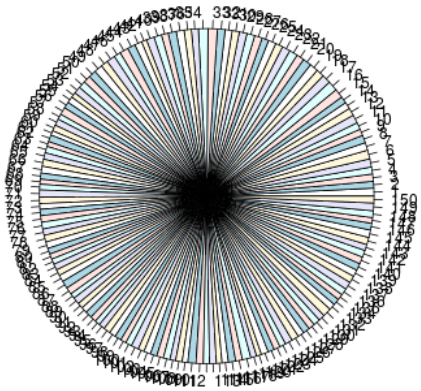

```
> hist(data[,1], 10) # Histograma, el número genera la cantidad de bins
```




```
> boxplot(data[,2]) # Gráfico de caja
```



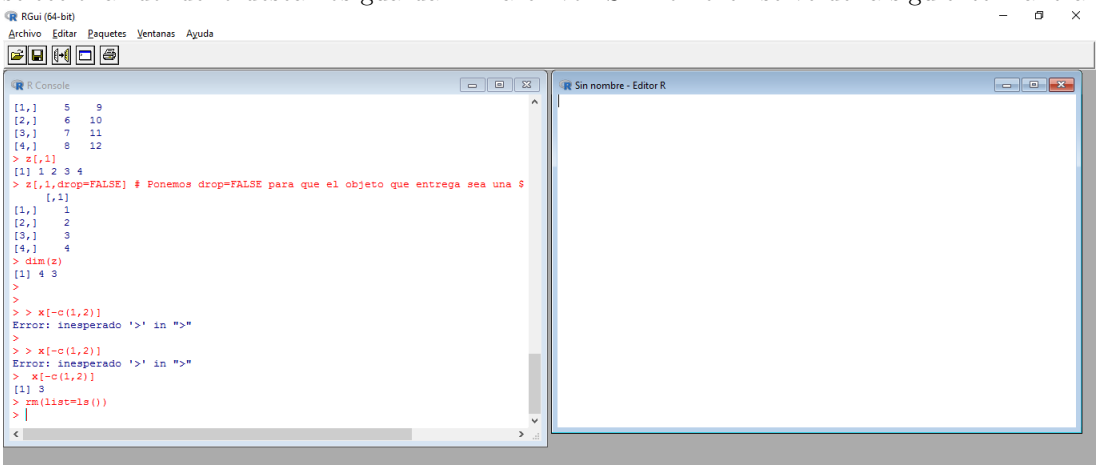

```
> pie(data[,2]) # Gráfico de torta
```



6. Crear proyectos

La base de los proyectos en R son los script, que básicamente son blocks de nota o archivos de texto plano. La idea es escribir varias lineas de código y luego tener la opción de ejecutar linea por linea o el archivo completo. A diferencia de la consola, en la cual solo puedes ejecutar linea por linea mientras escribes y no puedes editar las lineas escritas anteriormente.

Para crear un script nuevo debes ir al menú superior y hacer click en "Archivo→Nuevo script". Al hacer ésto, se abrirá una pantalla en blanco llamada "Sin nombre" y que al guardar deberemos darle uno y seleccionar donde lo deseamos guardar. El archivo "Sin nombre" se ve de la siguiente manera.



Una vez que tengamos un script con el proyecto que deseamos correr, debemos usar el comando `Ctrl+r` (o `F5`). que ejecuta la linea en donde esta el cursor o las lineas seleccionadas. En el caso de querer ejecutar el script completo, podemos seleccionarlo todo (usando el *mouse* o apretando `Ctrl+a`) y luego apretar `Ctrl+r`.