Introducción a R

Email: [delval@decsai.ugr.es](mailto:delval@decsai.ugr.es)

1. Matrices

\* Ejecuta los siguientes comandos.

matrix(data=5, nr=2, nc=2)

matrix(1:6, 2, 3)

matrix(1:6, 2, 3, byrow=TRUE)

\* Crea un vector z con los 30 primeros números y crea con el una matriz m con 3 filas y 10 columnas.

\* Escribe la tercera columna en un vector

\* Create in R the matrices

x =3 2l

−1 1

y =1 4 0

0 1 -1

Y calcula los efectos de los siguientes comandos

(a) x[1,]

(b) x[2,]

(c) x[,2]

(d) y[1,2]

(e) y[,2:3]

\* Transforma la matriz m que creaste en el ejercicio anterior en un array multidimensional. (Pista: averigua lo que puedas de la función dim().)

\* Crea un array de 5 x 5 y rellénalo con valores del 1 al 25. Investiga la función array(). Llama al array x

\* Escribe el array x en un vector y

\* Dadas las matrices m1 y m2 usa rbind() y cbind() para crear matrices nuevas utilizando estas funciones, llamalas M1 y M2. ¿En que se diferencian las matrices creadas?

m1 <- matrix(1, nr = 2, nc = 2)

m2 <- matrix(2, nr = 2, nc = 2)

\* El operador para el producto de dos matrices es ‘ %\* %’. Por ejemplo, considerando las dos matrices creadas en el ejercicio anterior utilízalo.

\* Usa la matriz M1 del ejercicio anterior y aplica la función t(). ¿qué hace esa función?

\* Ejecuta los siguientes comandos basados en la función diag() sobre las matrices creadas anteriormente m1 y m2. ¿Qué tipo de acciones puedes ejecutar con ella?

> diag(m1)

> diag(rbind(m1, m2) %\*% cbind(m1, m2))

> diag(m1) <- 10

> diag(3)

> v <- c(10, 20, 30)

> diag(v)

> diag(2.1, nr = 3, nc = 5)

\* Ordena la matriz x <- matrix(1:100, ncol=10):

a. en orden descendente por su segunda columna y asigna el resultado a una nueva matrix x1. Pista: función order()

b. en orden descendente por su segunda fila y asigna el resultado a una nueva matrix x2

c. Ordena solo la primera columna de x de forma descendente

\* Accede al dataset “women”.

* Primero confirma que los datos están ordenados de forma creciente según la altura (height) y el peso (weight) sin mirar los datos
* Crea una nueva variable “bmi”. Este valor responde a la siguiente fórmula: BMI = ( Weight in Pounds / (Height in inches) x (Height in inches) ) x 703
* Ordena el dataframe por el valor de bmi y lpor orden alfabético de la variable name

investiga las funciones is.unsorted(), sort() and order()

\* Crea los siguientes vectores:

# Box office Star Wars: In Millions (!) First element: US, Second element:

# Non-US

new\_hope = c(460.998007, 314.4)

empire\_strikes = c(290.475067, 247.9)

return\_jedi = c(309.306177, 165.8)

Los datos se corresponden con las ventas en millones de la trilogía de la guerra de las galaxias. El primer numero corresponde a las ventas en US y el segundo al resto de países.

1. Construye la matriz star\_wars\_matrix con esos vectores
2. Añádele nombres a las columnas y filas de la matriz según las descripciones dadas anteriormente de los datos
3. Calcula las ganacias mundiales de cada película y guardalas en un vector que se llame worldwide\_vector.
4. Añade éste ultimo vector como una columna nueva a la matriz star\_wars\_matrix y asigna el resultado a all\_wars\_matrix. Usa para ello la función [cbind()](http://www.rdocumentation.org/packages/base/functions/cbind" \t "_blank).
5. Calcula las ganancias totals en USA y fuera de USA para las tres películas. Puedes usar para ello la función [colSums()](http://www.rdocumentation.org/packages/base/functions/colSums" \t "_blank)
6. Calcula la media de ganancias para todas las películas fuera de los estados unidos. Asigna esa media la variable non\_us\_all.
7. Haz lo mismo pero solo par alas dos primeras películas . Asigna el resultado a la variable non\_us\_some.
8. Calcula cuantos visitantes hubo para cada película en cada área geográfica. Ya tienes las ganancias totales en star\_wars\_matrix. Asume que el precio de las entradas es de cinco euros/dólares (Nota: el numero total de visitantes para cada pelicula dividido por el precio del ticket te da el numero de visitantes)
9. Calcula la media de visitantes en territorio USA y en territorio noUS.
10. Subsetting matrices y arrays

\* Como hemos visto en teoría la sintásis para acceder tanto a matrices como a arrays bidimiensionales es la siguiente.

array[rows, columns]

Muchas funciones de R necesitan una matriz como dato de entrada. Si algo no funciona recuerda convertir el objeto a una matriz con la función

as.matrix(iris)

\* Crea un arrayi <- array(c(1:10),dim=c(5,2)). ¿Que información te dan los siguientes comandos?

dim(i);

nrow(i);

ncol(i)

\* Crea un array de dimensiones 5 filas y dos columnas y rellénalo con valores del 1-5 y del 5 al 1

\* ¿Qué hace el comando x[i] ¿. Comprueba que tienes en x antes

\* ¿y el comando x[i] <- 0?

\* Descárgate el fichero array\_datos.txt de PRADO (Datos/) e impórtalo en tu work space de R teniendo en cuenta que es un texto tabulado. Después crea un documento con los mismos datos pero en formato csv en vez de tab separated.

1. Factors

\* Dado x = c(1, 2, 3, 3, 5, 3, 2, 4, NA), ¿cuáles son los levels de factor(x)?

1. 1, 2, 3, 4, 5
2. NA
3. 1, 2, 3, 4, 5, NA

\* Dado  x <- c(11, 22, 47, 47, 11, 47, 11) y la ejecución de la sentencia

factor(x, levels=c(11, 22, 47), ordered=TRUE) ¿cuál es el cuarto elemento de la salida?

a. 11  
b. 22  
c. 47

\* Para el factor  z <- c("p", "a" , "g", "t", "b"), reemplaza el tercer elemento de z por "b".

a. factor(z[3]) <- "b"  
b. levels(z[3]) <- "b"  
c. z[3] <- "b"

\* Dado z <- factor(c("p", "q", "p", "r", "q")) escribe una expresión de R que cambie el level "p" a "w"

\* Usa el dataset “iris”

* escribe la expresión necesaria para convertir la variable “Sepal.Length” en un factor con cinco niveles (levels) . Pista( mira la función table() y la función cut().
* escribe la expresión necesaria para generar una tabla de frecuencias con dos filas y tres columnas . Las filas deben referirse a si la variable “Sepal.length” es menor que 5 y las columnas a las diferentes expecies. El resultado debe ser:

setosa versicolor virginica

FALSE 30 49 49

TRUE 20 1 1

\* El factor responses se define como:

responses <- factor(c("Agree", "Agree", "Strongly Agree", "Disagree", "Agree")),

sin embargo nos damos cuenta que tiene un nuevo nivel, "Strongly Disagree", que no estaba presente cuando se creó. Añade el nuevo nivel al factor y conviértelo en un factor ordenado de la siguiente forma:

Levels: Strongly Agree < Agree < Disagree < Strongly Disagree

\* Dado el factor:

x <- factor(c("high", "low", "medium", "high", "high", "low", "medium"))

escribe la expresión en R que permita dar valores numéricos únicos para los distintos niveles (levels) de x según el siguiente esquema:

level high => value 1  
level low => value 2  
level medium => value 3

Pista: investiga la función unique() y los parámetros de data.frame()

1. Acceso y selección de secciones de un data frames

La sintaxis general para acceder a un data frame es

my\_frame[rows, columns]

\* Vamos a trabajar con un ejemplo que viene por defecto en la instalación de R USArrests. Este data frame contiene la información para cada estado Americano de las tasas de criminales (por 100.000 habitantes). Los datos de las columnas se refieren a Asesinatos, violaciones y porcentaje de la población que vive en áreas urbanas. Los datos son de 1973. Contesta a las siguientes preguntas sobre los datos

- Las dimensiones del dataframe

- La longitud del dataframe (filas o columnas)

- Numero de columnas

- ¿Cómo calcularías el número de filas?

- Obtén el nombre de las filas y las columnas para este data frame

- échale un vistazo a los datos, por ejemplo a las seis primeras filas

- Ordena de forma decreciente las filas de nuestro data frame según el porcentaje de población en el área urbana. Para ello investiga la función order () y sus parámetros.

- ¿Podrías añadir un segundo criterio de orden?, ¿cómo?

- Muestra por pantalla la columna con los datos de asesinato

- Muestra las tasas de asesinato para el segundo, tercer y cuarto estado

- Muestra las primeras cinco filas de todas las columnas

- Muestra todas las filas para las dos primeras columnas

- Muestra todas las filas de las columnas 1 y 3

- Muestra solo las primeras cinco filas de las columnas 1 y 2

- Extrae las filas para el índice Murder

Vamos con expresiones un poco mas complicadas:…

-¿Que estado tiene la menor tasa de asesinatos? ¿qué línea contiene esa información?, obtén esa informaciónn

¿Que estados tienen una tasa inferior al 4%?, obtén esa informaciónn

¿Que estados estan en el cuartil superior (75) en lo que a poblacion en zonas urbanas se refiere?

\* Vamos a trabajar con otro dataframe. Descarga el fichero student.txt de la plataforma PRADO, almacena la información en una variable llamada “students”. Ten en cuenta que los datos son tab-delimited y tienen un texto para cada columna. Comprueba que R ha leído correctamente el fichero imprimiendo el objeto en la pantalla

> students

-Imprime solo los nombres de la columnas

-Llama a la columna height solo

-¿Cuantas observaciones hay en cada grupo?. Utiliza la función table(). Este commando se puede utilizar para crear tablas cruzadas (cross-tabulations)

-Crea nuevas variables a partir de los datos que tenemos. Vamos a crear una variable nueva “sym” que contenga M si el genero es masculino y F si el genero es femenino. Busca en la ayuda información sobre la función ifelse(). Crea una segunda variable “colours” cuyo valor será “Blue” si el estudiante es de kuopio y “Red” si es de otro sitio.

- Con los datos anteriores de height y shoesize y las nuevas variables crea un nuevo data.frame que se llame students.new

- Comprueba que la clase de student.new es un dataframe

- Crea dos subsets a partir del dataset student. Dividelo dependiendo del sexo. Para ello primero comprueba que estudiantes son hombres (male). Pista: busca información sobre la función which.

-Basándote en esa selección dada por which() toma solo esas filas del dataset student para generar el subset stundent.male

- Repite el procedimiento para seleccionar las estudiantes mujeres (females)

- Utiliza la function write.table() para guarder el contenido de student.new en un archivo.