


Estadística y modelación de sistemas socio-ecológicos en R




Laboratorio
Nacional
de Ciencias
de la Sostenibilidad

Dra. Yosune Miquelajauregui Graf
yosune@iecologia.unam.mx


Objetivos del curso

1. Conocer los principios básicos de la estadística descriptiva e inferencial para su correcta aplicación
 2. Implementar algoritmos matemáticos para la modelación de sistemas socio-ecológicos
 3. Incorporar los conocimientos adquiridos al lenguaje de programación R
 4. Al finalizar el curso, los alumnos aplicarán los conocimientos adquiridos en el análisis de datos de tesis
 5. Aplicar la teoría en ejercicios prácticos
- 

Evaluación

1. Tareas y ejercicios en clase (20%)
 2. Asistencia y participación en clase (10%)
 3. Exposición de seminario (40%)
 4. Scripts (R markdowns) (30%)
- 

Plan del día

1. Introducción a R: tipo de objetos y manipulación de datos
 2. Instalación de R en los dispositivos de cómputo
 3. Instalación de editores : RStudio o Tinn-R
 4. Ejercicios
- 

Introducción a R



- R es un lenguaje de programación orientada a objetos
- Cualquier operación puede ser asignada a un objeto que queda guardado en la memoria de la sesión de R
- Es utilizado para realizar desde operaciones matemáticas simples hasta computaciones estadísticas complejas
- R también se utiliza para manipular base de datos, graficar e implementar modelos matemáticos

Introducción a R

Operaciones matemáticas simples

Sumas

```
3+2  
[1] 5
```

Restas

```
5-2  
[1] 3
```

Divisiones

```
6/3  
[1] 2
```

Multiplicaciones

```
6*3  
[1] 18
```

Introducción a R

Tipos de objetos

1. Constantes

```
a <- 6*3  
a  
[1] 18
```

2. Vectores

```
b <- c(1,4,8,16)  
b  
[1] 1 4 8 16
```

3. Hoja de datos

```
c1 <- data.frame(5:1,1:5)  
c1  
  X5.1 X1.5  
1     5     1  
2     4     2  
3     3     3  
4     2     4  
5     1     5
```

Introducción a R

Tipos de objetos

4. Matrices

```
m <- matrix (1:4,2,2)
```

```
m  
      [,1] [,2]  
[1,]    1    3  
[2,]    2    4
```

5. Listas

```
c1 <- data.frame(5:1,1:5)
```

```
c2 <- data.frame (1, 20:25)
```

```
mi.lista <- list(c1, c2)
```

```
mi.lista
```

```
[[1]]
```

```
  X5.1 X1.5
```

```
1    5    1
```

```
2    4    2
```

```
3    3    3
```

```
4    2    4
```

```
5    1    5
```

```
[[2]]
```

```
  X1 X20.25
```

```
1  1    20
```

```
2  1    21
```

```
3  1    22
```

```
4  1    23
```

```
5  1    24
```

```
6  1    25
```


Manipulación de datos

Obtener atributos de los objetos : la estructura

Función R : str ()

1.

```
a <- 6*3
```

```
a
```

```
[1] 18
```

```
str (a)
```

```
num 18
```

2.

```
b <- c(1,4,8,16)
```

```
b
```

```
[1] 1 4 8 16
```

```
str (b)
```

```
num [1:4] 1 4 8 16
```

3.

```
c1 <- data.frame(5:1,1:5)
```

```
c1
```

```
  X5.1 X1.5
```

```
1    5    1
```

```
2    4    2
```

```
3    3    3
```

```
4    2    4
```

```
5    1    5
```

```
str (c1)
```

```
'data.frame':  5 obs. of  2
```

```
variables:
```

```
$ X5.1: int  5 4 3 2 1
```

```
$ X1.5: int  1 2 3 4 5
```

Manipulación de datos

Obtener atributos de los objetos : el largo

Función R : length ()

1.

```
a <- 6*3
```

```
a
```

```
[1] 18
```

```
length (a)
```

```
[1] 1
```

2.

```
b <- c(1,4,8,16)
```

```
b
```

```
[1] 1 4 8 16
```

```
length(b)
```

```
[1] 4
```

3.

```
c1 <- data.frame(5:1,1:5)
```

```
c1
```

```
  X5.1 X1.5
```

```
1    5    1
```

```
2    4    2
```

```
3    3    3
```

```
4    2    4
```

```
5    1    5
```

```
length(c1[,2])
```

```
[1] 5
```

Manipulación de datos

Operaciones matemáticas entre distintos tipos de objetos

```
b <- c(1,4,8,16)
```

```
xx <- c(2,3,4,5)
```

```
bporxx <- b*xx
```

Resultado :

```
bporxx
```

```
[1] 2 12
```

```
32 80
```

```
bxx <- b/xx
```

Resultado:

```
bxx
```

```
[1] 0.500000
```

```
1.333333 2.000000
```

```
3.200000
```

```
b1 <- b*a
```

```
a <- 6*3
```

```
b <- c(1,4,8,16)
```

Resultado:

```
b1
```

```
[1] 18 72 144 288
```

```
b2 <- b-a
```

Resultado:

```
b2
```

```
[1] -17 -14 -10 -2
```

Manipulación de datos

Acceder a ciertos elementos del objeto

```
b <- c(1,4,8,16)
```

```
b[c(3,1,4)]  
[1] 8 1 16
```

```
c1 <- data.frame(5:1,1:5)
```

```
X5.1 X1.5  
1 5 1  
2 4 2  
3 3 3  
4 2 4  
5 1 5
```

Primer renglón

```
c1[1,]  
X5.1 X1.5  
1 5 1
```

Primeros 3 renglones

```
c1[1:3,]  
X5.1 X1.5  
1 5 1  
2 4 2  
3 3 3
```

Manipulación de datos

Acceder a ciertos elementos del objeto

```
c1 <- data.frame(5:1,1:5)
```

```
X5.1 X1.5
```

```
1    5    1
```

```
2    4    2
```

```
3    3    3
```

```
4    2    4
```

```
5    1    5
```

Primera

columna

```
c1[,1]
```

```
[1] 5 4 3 2 1
```

Segunda

columna

```
c1[,2]
```

```
[1] 1 2 3 4 5
```

Manipulación de datos

Acceder a ciertos elementos del objeto

mi.lista			[[2]]		
[[1]]			X1 X20.25		
X1 X1.5					
1	1	1	1	1	20
2	1	2	2	1	21
3	1	3	3	1	22
4	1	4	4	1	23
5	1	5	5	1	24
			6	1	25

Primer elemento de la lista

mi.lista[[1]]		
X1 X1.5		
1	1	1
2	1	2
3	1	3
4	1	4
5	1	5

Manipulación de datos

Añadir, reemplazar y eliminar elementos del objeto

```
b  
[1] 1 4 8 16
```

```
b[1] <- 5
```

```
b  
[1] 5 4 8 16
```

```
b[5:10] <- c(18:21,30:31)
```

```
b  
[1] 5 4 8 16 18 19 20 21 30 31
```

```
b <- b[-10]
```

```
b  
[1] 5 4 8 16 18 19 20 21 30
```

Manipulación de datos

Asignar nombres a las columnas de una hoja de datos

Función R : colnames ()

```
colnames(c1)<- c('Individuo','Orden')
```

```
c1 <- data.frame(5:1,1:5)
```

```
X5.1 X1.5
```

```
1  5  1
```

```
2  4  2
```

```
3  3  3
```

```
4  2  4
```

```
5  1  5
```

	Individuo	Orden
1	5	1
2	4	2
3	3	3
4	2	4
5	1	5

Manipulación de datos

Crear una nueva columna en la hoja de datos resultado de una operación matemática

```
c1 <- data.frame(5:1,1:5)
```

```
X5.1 X1.5
```

```
1 5 1
```

```
2 4 2
```

```
3 3 3
```

```
4 2 4
```

```
5 1 5
```

```
c1[,3]<- c1$Orden/7
```

```
colnames(c1)[3]= 'Tasa'
```

```
head (c1)
```

```
Individuo Orden Tasa
```

```
1 5 1 0.1428571
```

```
2 4 2 0.2857143
```

```
3 3 3 0.4285714
```

```
4 2 4 0.5714286
```

```
5 1 5 0.7142857
```

Manipulación de datos

Crear una nueva columna en la hoja de datos resultado de una operación matemática

```
c1 <- data.frame(5:1,1:5)
```

```
X5.1 X1.5
```

```
1 5 1
```

```
2 4 2
```

```
3 3 3
```

```
4 2 4
```

```
5 1 5
```

```
c1[,3]<- c1$Orden/7
```

```
colnames(c1)[3]= 'Tasa'
```

```
head (c1)
```

```
Individuo Orden Tasa
```

```
1 5 1 0.1428571
```

```
2 4 2 0.2857143
```

```
3 3 3 0.4285714
```

```
4 2 4 0.5714286
```

```
5 1 5 0.7142857
```

Manipulación de datos

Importar y exportar archivos .csv o .txt

Función R: read.table (), read.csv(), write.table() y write.csv()

```
Incendio<-read.table("C:\\Users\\Usuario\\Documents\\CursoR\\Fire_intensity.txt", header = T)
```

```
write.csv(Incendio,file="C:\\Users\\Usuario\\Documents\\CursoR\\incendios.csv")
```



Manipulación de datos

Mostrar los primeros diez elementos del objeto

Función R: head ()

```
head(Incendio)
  Probability.mortality Stand.age Fire.intensity Region Stand.id Tree.id
1         0.04008590   Young    13.967749    A2      1         1      1
2         0.02371104   Young     8.118900    A2      1         2      2
3         0.02959688   Young    10.189049    A2      1         3      3
4         0.06637321   Young    23.751139    A2      1         4      4
5         0.07203268   Young    25.304217    A2      1         5      5
6         0.02844691   Young     9.429986    A2      1         6      6
```

Manipulación de datos

Mostrar los nombres de las columnas de la hoja de datos

Función R: names ()

```
> names(Incendio)
[1] "Probability.mortality" "Stand.age"
[3] "Fire.intensity"       "Region"
[5] "Stand.id"             "Tree.id"
```

Manipulación de datos

Obtener el rango de una variable continua en la hoja de datos

Función R: range()

Se utiliza el \$ para identificar una variable y trabajar con ella.

```
> range(Incendio$Fire.intensity)  
[1] NA NA
```

Manipulación de datos

Remover de la hoja de datos los valores con NA

Función R: na.omit ()

```
range(Incendio$Fire.intensity)
[1] 2.423678e-01 1.000000e+04
```

Manipulación de datos

Verificar que una variable sea factor o numérico

Funciones R: `is.factor ()` , `is.numeric ()`

```
is.factor(Incendio$Stand.id)  
[1] FALSE
```

```
is.numeric(Incendio$Stand.id)  
[1] TRUE
```


Manipulación de datos

Convertir objetos - números a factores y viceversa

Funciones R: `as.factor ()` , `as.numeric ()`

```
> is.factor(Incendio$Region)
[1] TRUE
> Incendio$Region<-as.numeric(Incendio$Region)
> is.factor(Incendio$Region)
[1] FALSE
```

Manipulación de datos

Crear subconjuntos de datos

```
# select variables v1, v2, v3
myvars <- c("v1", "v2", "v3")
newdata <- mydata[myvars]
```

Funciones R: subset () , which ()

Incendio <- Incendio[- which(Incendio\$Probability.mortality < 0),] **Eliminar negativos**

Incendio2 <- Incendio[which(Incendio\$Fire.intensity < 1000),] **Remover valores menores a 1000**

Incendio3 <-subset(Incendio2[,c(1,3,4)]) Seleccionar las variables 1,3 y 4

```
> names(Incendio3)
```

```
[1] "Probability.mortality" "Fire.intensity" "Region"
```

Manipulación de datos

Crear subconjuntos de datos

Funciones R: subset () , which ()

`Incendio <- Incendio[- which(Incendio$Probability.mortality < 0),]` **Eliminar negativos**

`Incendio2 <- Incendio[which(Incendio$Fire.intensity < 1000),]` **Remover valores menores a 1000**

Manipulación de datos

Crear subconjuntos de datos

Funciones R: subset () , which ()

Incendio3 <-subset(Incendio2[,c(1,3,4)]) Seleccionar las variables 1,3 y 4

```
> names(Incendio3)  
[1] "Probability.mortality" "Fire.intensity" "Region"
```

Manipulación de datos

Crear subconjuntos de datos utilizando operadores lógicos

Funciones R: subset () , which ()

Operador lógico	Descripción	Ejemplo
& (and)	Verdadero sólo si las dos comparaciones son verdaderas	>> x=5; >> (x>2) & (x<10) ans = 1
(or)	Verdadero si al menos una de las comparaciones es verdaderas	>> x=8; >> (x>2) (x<5) ans = 1
~ (not)	Niega el resultado de la comparación. Verdadero si la comparación es falsa.	>> x=8; >> ~ (x>10) ans = 1

Manipulación de datos

Crear subconjuntos de datos utilizando operadores lógicos

Funciones R: subset () , which ()

```
nuevosdatos <- Incendio[which(Incendio$Fire.intensity<1500  
                             & Incendio$Stand.age == "Young"), ]
```

```
> levels(nuevosdatos$Stand.age)  
[1] "Young"
```

Manipulación de datos

Crear una nueva variable basada en niveles de otra variable

Función R: ifelse ()

Generar una nueva variable : "Tipo de incendio" basada en la variable Fire.intensity (intensidad del incendio (kW/m)):

1. Superficial de baja intensidad SBI cuando Intensidad < 200
2. Superficial alta intensidad cuando Intensidad ≥ 200 y < 1500
3. Corona alta intensidad cuando Intensidad > 1500

Manipulación de datos

Crear una nueva variable basada en niveles de otra variable

Función R: ifelse ()

```
Incendio$TipoIncendio <-  
ifelse(Incendio$Fire.intensity<200,"SBI",ifelse(Incendio$Fire.intensity>=200 &  
Incendio$Fire.intensity<1500,"SAI","CAI"))
```

```
> levels(Incendio$TipoIncendio)  
[1] "CAI" "SAI" "SBI"
```


Manipulación de datos

Elegir entradas basadas en otros atributos

Función R: subset ()

```
nuevosdatos <- subset(Incendio, TipoIncendio == "SBI",  
                      select=c(Probability.mortality,TipoIncendio))
```

Manipulación de datos

Unir elementos de dos o más hojas de datos y matrices

Función R: cbind (), rbind () y merge ()

df1				
	id1	id2	c11	c12
1	11	21	1	a
2	12	22	2	b
3	13	23	3	c
4	14	24	4	d

df2				
	id1	id2	c21	c22
1	99	98	5	e
2	12	22	6	f
3	13	23	7	g
4	14	24	8	h

df3				
	id1	id2	c31	c32
1	999	998	9	i
2	9999	9998	10	j
3	13	23	11	k
4	14	24	12	l

Manipulación de datos

Unir elementos de dos o más hojas de datos y matrices

Función R: cbind (), rbind () y merge ()

```
> cbind(df1,df2,df3)
```

	id1	id2	c11	c12	id1	id2	c21	c22	id1	id2	c31	c32
1	11	21	1	a	99	98	5	e	999	998	9	i
2	12	22	2	b	12	22	6	f	9999	9998	10	j
3	13	23	3	c	13	23	7	g	13	23	11	k
4	14	24	4	d	14	24	8	h	14	24	12	l

Manipulación de datos

Unir elementos de dos o más hojas de datos y matrices

Función R: cbind (), rbind () y merge ()

```
> rbind(df1,df2,df3)
```

```
Error in match.names(clabs, names(xi)) :  
names do not match previous names
```

N.B. Marca error ya que los nombres de las columnas son distintos

Manipulación de datos

Unir elementos de dos o más hojas de datos y matrices

Función R: cbind (), rbind () y merge ()

```
> rbind(df1[,1],df2[,1],df3[,1])
```

	[,1]	[,2]	[,3]	[,4]
[1,]	11	12	13	14
[2,]	99	12	13	14
[3,]	999	9999	13	14

Manipulación de datos

```
> rbind(df1[,1:2],df2[,1:2],df3[,1:2])
```

	id1	id2
1	11	21
2	12	22
3	13	23
4	14	24
5	99	98
6	12	22
7	13	23
8	14	24
9	999	998
10	9999	9998
11	13	23
12	14	24

Manipulación de datos

Unir elementos de dos o más hojas de datos y matrices

Función R: cbind (), rbind () y merge ()

```
> rbind(df1[,1],df2[,1],df3[,1])
```

	[,1]	[,2]	[,3]	[,4]
[1,]	11	12	13	14
[2,]	99	12	13	14
[3,]	999	9999	13	14

Manipulación de datos

Unir elementos de dos o más hojas de datos y matrices

Función R: cbind (), rbind () y merge ()

```
m1 <- matrix (1:4,2,2)  
m2 <- matrix (5:8,2,2)
```

m1

	[,1]	[,2]
[1,]	1	3
[2,]	2	4

m2

	[,1]	[,2]
[1,]	5	7
[2,]	6	8

Manipulación de datos

Unir elementos de dos o más hojas de datos y matrices

Función R: cbind (), rbind () y merge ()

```
m1 <- matrix (1:4,2,2)
m2 <- matrix (5:8,2,2)
```

```
m3<- cbind (m1,m2)
```

```
m3
      [,1] [,2] [,3] [,4]
[1,]    1    3    5    7
[2,]    2    4    6    8
```

Manipulación de datos

Unir elementos de dos o más hojas de datos y matrices

Función R: cbind (), rbind () y merge ()

```
m1 <- matrix (1:4,2,2)
```

```
m2 <- matrix (5:8,2,2)
```

```
m4<- rbind (m1,m2)
```

m4	[,1]	[,2]
[1,]	1	3
[2,]	2	4
[3,]	5	7
[4,]	6	8

Manipulación de datos

Unir elementos de dos o más hojas de datos y matrices

Función R: cbind (), rbind () y merge ()

```
> merge(df1,df2,,all = TRUE)
```

	id1	id2	c11	c12	c21	c22
1	11	21	1	a	NA	NA
2	12	22	2	b	6	f
3	13	23	3	c	7	g
4	14	24	4	d	8	h
5	99	98	NA	NA	5	e

```
> merge(df1,df2, by = "id1")
```

	id1	id2.x	c11	c12	id2.y	c21	c22
1	12	22	2	b	22	6	f
2	13	23	3	c	23	7	g
3	14	24	4	d	24	8	h

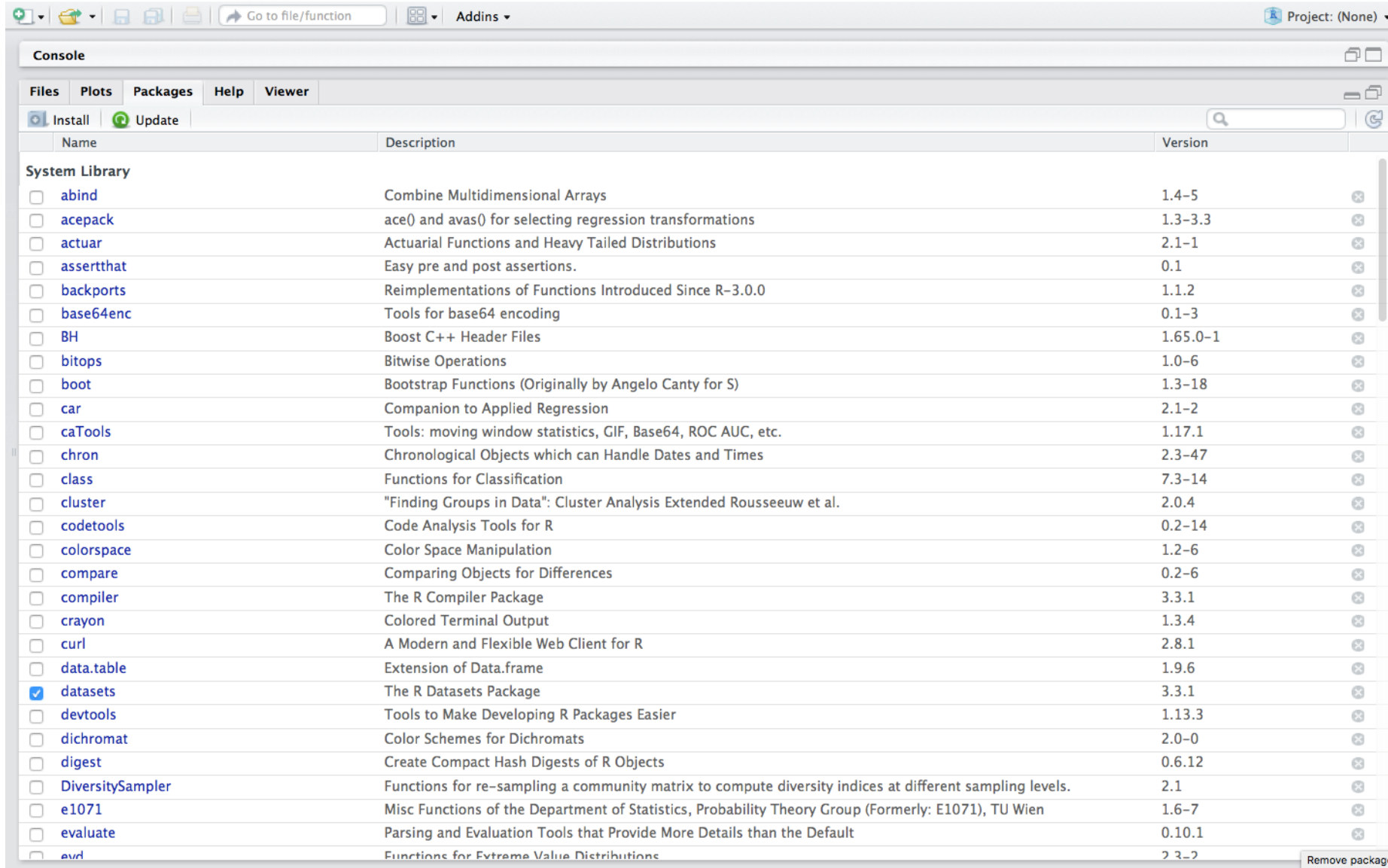
Instalación de R

1. Instalación de R : <https://www.r-project.org/>
2. Instalación un editor en R : RStudio (<https://www.rstudio.com/>),
Tinn-R (<https://sourceforge.net/projects/tinn-r/>)
3. Instalación de paquetería en R

Paquetería de R

1. Un paquete es un banco de funciones implementadas en R.
2. Cuando se abre una sesión en R, una serie de paquetes son activados automáticamente.
3. Para activar el paquete de interés es necesario cargarlo en la sesión de trabajo de R con la función `library ()`.

Paquetería de R



The screenshot displays the RStudio interface, specifically the 'Packages' tab. The top toolbar includes icons for file operations and a 'Go to file/function' search bar. The 'Project: (None)' dropdown is visible in the top right. The 'Console' window is open at the top. Below it, the 'Packages' tab is active, showing a list of installed and available packages. The 'System Library' section is expanded, showing a list of packages with their names, descriptions, and versions. The 'datasets' package is checked, indicating it is installed. The 'Remove package' button is visible at the bottom right of the package list.

Name	Description	Version
<input type="checkbox"/> abind	Combine Multidimensional Arrays	1.4-5
<input type="checkbox"/> acepack	ace() and avas() for selecting regression transformations	1.3-3.3
<input type="checkbox"/> actuar	Actuarial Functions and Heavy Tailed Distributions	2.1-1
<input type="checkbox"/> assertthat	Easy pre and post assertions.	0.1
<input type="checkbox"/> backports	Reimplementations of Functions Introduced Since R-3.0.0	1.1.2
<input type="checkbox"/> base64enc	Tools for base64 encoding	0.1-3
<input type="checkbox"/> BH	Boost C++ Header Files	1.65.0-1
<input type="checkbox"/> bitops	Bitwise Operations	1.0-6
<input type="checkbox"/> boot	Bootstrap Functions (Originally by Angelo Canty for S)	1.3-18
<input type="checkbox"/> car	Companion to Applied Regression	2.1-2
<input type="checkbox"/> caTools	Tools: moving window statistics, GIF, Base64, ROC AUC, etc.	1.17.1
<input type="checkbox"/> chron	Chronological Objects which can Handle Dates and Times	2.3-47
<input type="checkbox"/> class	Functions for Classification	7.3-14
<input type="checkbox"/> cluster	"Finding Groups in Data": Cluster Analysis Extended Rousseeuw et al.	2.0.4
<input type="checkbox"/> codetools	Code Analysis Tools for R	0.2-14
<input type="checkbox"/> colorspace	Color Space Manipulation	1.2-6
<input type="checkbox"/> compare	Comparing Objects for Differences	0.2-6
<input type="checkbox"/> compiler	The R Compiler Package	3.3.1
<input type="checkbox"/> crayon	Colored Terminal Output	1.3.4
<input type="checkbox"/> curl	A Modern and Flexible Web Client for R	2.8.1
<input type="checkbox"/> data.table	Extension of Data.frame	1.9.6
<input checked="" type="checkbox"/> datasets	The R Datasets Package	3.3.1
<input type="checkbox"/> devtools	Tools to Make Developing R Packages Easier	1.13.3
<input type="checkbox"/> dichromat	Color Schemes for Dichromats	2.0-0
<input type="checkbox"/> digest	Create Compact Hash Digests of R Objects	0.6.12
<input type="checkbox"/> DiversitySampler	Functions for re-sampling a community matrix to compute diversity indices at different sampling levels.	2.1
<input type="checkbox"/> e1071	Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien	1.6-7
<input type="checkbox"/> evaluate	Parsing and Evaluation Tools that Provide More Details than the Default	0.10.1
<input type="checkbox"/> evd	Functions for Extreme Value Distributions	2.3-2


Paquetería de R

4. Si el paquete no se encuentra ahí, es necesario bajarlo e instalarlo desde CRAN

Algunos paquetes de interés:

- a) Modelos lineales mixtos y generalizados: MASS, nmle, lme4
- b) Análisis multivariados: vegan, ade4, cluster
- c) Interactuar con otros programas: R2WinBUGS, Brugs
- d) Funciones de utilidad general: car, Design, Hmisc, gregmisc, gmodles, foreign
- e) Análisis espaciales: geoR, sp, spatial
- f) Análisis genéticos: adegenet, ape, apTreeshape, genetics
- g) Análisis dendrocronológicos: dpIR

Ejercicio 1

1. Importar Datos1.csv
 2. Verificar estructura e identificar variables
 3. Calcular una nueva variable llamada “Porcentaje de habitantes en pobreza (PHP)”
 4. Crear una nueva variable llamada “Categorías de pobreza” con respecto al porcentaje de habitantes con pobreza: Bajo (PHP <20), Moderado (PHP >20 y PHP <60) y Alto (PHP >60)
- 

Ejercicio 2

1. Importar Datos2.csv
2. Verificar estructura
3. Asignar nombres a las columnas 2 y 3: “Edad de la infraestructura” y “Proporción de viviendas sin drenaje”
4. Verificar el rango de la variable ”Proporción de viviendas sin drenaje”
5. Crear un nuevo objeto llamado “tmp” que contenga solamente los valores de la variable proporción de viviendas sin drenaje mayores que 0.5 y los elementos de la variable edad de la infraestructura mayores a 11”
6. Eliminar el primer y último renglón