

Algunos temas de estadística
Implementados en R

Algunos temas de estadística Implementados en R

CARLOS ALBERTO RAMOS SOLER
SANDRA PATRICIA CÁRDENAS OJEDA

Universidad Pedagógica y Tecnológica de Colombia
Tunja
2010

Algunos temas de estadística implementados en R / Carlos Alberto Ramos Soler, Sandra Patricia Cárdenas Ojeda. – Tunja: Uptc, 2010.

142 p.: il. – (Colección investigación Uptc, no. 36) Incluye bibliografía
ISBN 978-958-660-168-9

1. Estadística en R.— I. Ramos Soler, Carlos Alberto.— II. Cárdenas Ojeda, Sandra Patricia. – III. Tit. – IV. Ser.

CDD 519.5/R147

Primera edición, 2010

300 ejemplares

Algunos temas de estadística implementados en R

ISBN 978-958-660-168-9

Colección investigación Uptc; n.º 36

© Carlos Alberto Ramos Soler

© Sandra Patricia Ojeda Cárdenas

© Universidad Pedagógica y Tecnológica de Colombia

Alfonso López Díaz, Rector

Wilson Alcides Valenzuela Pérez, Vicerrector Académico

Manuel Humberto Restrepo Domínguez, Director de Investigaciones

Resultado del proyecto de investigación “Capacitación sobre el manejo del paquete estadístico R y su implementación como material didáctico en el programa de Licenciatura de Matemáticas y Estadística” del Grupo de Investigación en Estadística GIE.

Libro financiado por la Dirección de Investigaciones de la Uptc.

Se autoriza la reproducción parcial o total, citando siempre la fuente.

Coordinación Editorial: Yolanda Romero A.

Corrección de Estilo: Luis Enrique Clavijo Morales

Impresión:

Grupo Imprenta y Publicaciones

Coordinador Pablo Alejandro Sánchez Pereira

UPTC - Avenida Central del Norte

Tels.: (0*8) 7422174/76. Fax - Ext.: 1530

imprenta.publicaciones@uptc.edu.co

Tunja - Boyacá - Colombia

CONTENIDO

	Pág.
PRESENTACIÓN	9
INTRODUCCIÓN	11
1 DESCARGA E INSTALACIÓN DE R	13
1.1 DESCARGA	13
1.2 INSTALACIÓN	16
1.3 ESTANDO EN R	16
1.3.1 Utilización de R	17
1.3.2 Mayúsculas y minúscula en R	17
2 FUNCIONES EN R	18
2.1 FUNCIONES ARITMÉTICAS	18
2.2 COMANDOS ESPECIALES	18
2.2.1 Comando help()	18
2.2.2 Asignación	19
2.2.3 Comando c()	19
2.2.4 Comando seq()	19
2.2.5 Comando rep()	19
2.3 VECTORES	20
2.3.1 Definición	20
2.3.2 Aritmética vectorial	20
2.3.3 Gráficas de funciones	21
2.3.4 Vectores lógicos	22
2.3.5 Vectores de caracteres	23
2.4 MATRICES	23
2.4.1 Definición	23
2.4.2 Matriz Identidad	24
2.5 OPERACIONES ARITMÉTICAS CON MATRICES	25
2.5.1 Multiplicación por un escalar	25
2.5.2 Multiplicación de matrices	26
2.5.3 Traspuesta de una matriz	26
2.5.4 Inversa de una matriz	27
2.5.5 Determinante	27

2.5.6 Valores y vectores propios	27
2.5.7 Comando apply()	28
2.5.8 Partición de matrices	28
2.6 EJERCICIOS	29
3 ESTADÍSTICA DESCRIPTIVA	31
3.1 TABLA DE DISTRIBUCIÓN DE FRECUENCIAS	31
3.2 FUNCIONES GRÁFICAS BÁSICAS PARA EL ANÁLISIS EXPLORATORIO DE DATOS	32
3.2.1 Diagrama de sectores	33
3.2.2 Diagrama de barras	33
3.2.3 Diagrama de tallos y hojas	34
3.2.4 Diagrama de Caja	34
3.2.5 Histograma	35
3.2.6 Dispersograma	35
3.3 MEDIDAS DE TENDENCIA CENTRAL	38
3.3.1 Media aritmética	38
3.3.2 Mediana	39
3.3.3 Moda	39
3.4 MEDIDAS DE DISPERSIÓN	39
3.4.1 Varianza	39
3.4.2 Desviación estándar	40
3.4.3 Cuantiles	40
3.5 MEDIDAS DE ASIMETRÍA	41
3.5.1 Sesgo	41
3.5.2 Curtosis	42
3.6 MEDIDAS DE ASOCIACIÓN	43
3.6.1 Covarianza	43
3.6.2 Correlación	43
3.7 EJERCICIOS	44
4 PROBABILIDAD	45
4.1 CONJUNTOS	45
4.1.1 Unión	46
4.1.2 Intersección	46
4.1.3 Diferencia	46
4.1.4 Diferencia simétrica	46
4.2 ESPACIO MUESTRAL	47
4.2.1 Probabilidad de eventos	47
4.2.2 Técnicas de conteo	47
4.2.3 Factorial	47
4.2.4 Combinaciones	48
4.2.5 Permutaciones	49
4.2.6 <code>rolldie(times,nsides=6,makespace=TRUE)</code>	50

4.2.7 tosscoin(times,makespace=TRUE)	50
4.2.8 urnsamples(x,size,replace=,ordered=)	51
4.2.9 Cálculo de una probabilidad y de probabilidades condicionales de eventos	51
4.3 EJERCICIOS	53
5 DISTRIBUCIONES	54
5.1 DISTRIBUCIONES DISCRETAS	54
5.2 DISTRIBUCIONES CONTINUAS	54
5.3 EJERCICIOS	65
6 INFERENCIA	67
6.1 GRÁFICO CUANTIL-CUANTIL (Q-Q PLOT)	67
6.2 PRUEBA EXACTA DE NORMALIDAD	69
6.3 PRUEBA KOLGOMOROV-SMIRNOV	69
6.4 HIPÓTESIS ESTADÍSTICA	70
6.4.1 Pruebas de hipótesis para la media	71
6.4.1.1 Prueba de hipótesis para la media varianza conocida	71
6.4.1.2 Prueba de hipótesis para la media con varianza desconocida y tamaño de muestra pequeño	73
6.4.2 Prueba de hipótesis para la homogeneidad de varianzas	74
6.4.3 Prueba de hipótesis para la Correlación/Asociación entre muestras pareadas	75
6.4.4 Prueba de hipótesis para el Sesgo	75
6.4.5 Prueba de hipótesis para la Curtosis	75
6.4.6 Prueba de Wilcoxon	76
6.5 EJERCICIOS	77
7 MUESTREO	78
7.1 TAMAÑO DE MUESTRA	78
7.2 MUESTREO ALEATORIO SIMPLE	79
7.3 MUESTREO SISTEMÁTICO	80
7.4 MUESTREO ESTRATIFICADO	81
7.5 EJERCICIOS	82
8 ANÁLISIS MULTIVARIADO	83
8.1 MATRIZ DE DATOS	83
8.2 VECTOR DE MEDIAS	84
8.3 MATRIZ DE VARIANZAS Y COVARIANZAS	85
8.4 MATRIZ DE CORRELACIONES	86
8.5 CÁLCULOS A PARTIR DE UN DATA FRAME	87
8.6 DISTANCIA DE MAHALANOBIS	89
8.7 ANÁLISIS GRÁFICO DE OBSERVACIONES MULTIVARIANTES	91
8.8 DISTRIBUCIÓN NORMAL MULTIVARIADA	92
8.9 ELIPSES DE CONFIANZA	93
8.10 EVALUACIÓN DE LA MULTINORMALIDAD	95
8.11 EJERCICIOS	101
9 ANÁLISIS DE REGRESIÓN	102

9.1 MODELO LINEAL	103
9.2 MODELOS LINEALES GENERALIZADOS (GLM)	105
9.3 EJERCICIOS	107
10 DISEÑO DE EXPERIMENTOS	108
10.1 CONSTRUCCIÓN DE VARIABLES CLASIFICATORIAS	108
10.2 ANÁLISIS DE VARIANZA	109
10.3 EJERCICIOS	118
11 IMPORTAR Y EXPORTAR DATOS EN R	120
11.1 IMPORTAR	120
11.2 EXPORTAR	121
BIBLIOGRAFÍA CITADA	123
BIBLIOGRAFÍA RECOMENDADA	124

PRESENTACIÓN

El Grupo de Investigación en Estadística (GIE), de la Universidad Pedagógica y Tecnológica de Colombia, cuenta entre sus líneas de investigación con una denominada «Aplicación de métodos o construcción de modelos estadísticos»; este libro es producto del desarrollo de esta línea, en lo que se refiere a la apropiación y enseñanza del *software* estadístico R en la Uptc y en la comunidad académica en general.

El libro consta de once capítulos, en los que se dan a conocer algunos comandos básicos y diversos ejemplos que ilustran las temáticas tratadas. Los capítulos 1 y 2 presentan lo relacionado con la descarga e instalación de R, además se mencionan determinadas funciones que involucran operaciones aritméticas entre números reales, vectores, matrices y ciertos comandos especiales, entre otros. Los capítulos del 3 al 10, aunque sus títulos coincidan con los diferentes campos de la estadística, como es el caso de Estadística Descriptiva, Probabilidad, Distribuciones, Inferencia, Muestreo, Análisis Multivariado, Análisis de regresión y Diseño de experimentos, solo presentan algunos tópicos relevantes que, a juicio de los autores, se pueden abordar mediante R, sin que sean los únicos.

Se espera que este libro sea oportuno para los diferentes usuarios de la estadística, y que permita un primer acercamiento con el R, de tal forma que se suscite en el usuario la curiosidad por explorar nuevas y más complejas temáticas.

Por último, es importante mencionar que el proyecto que culminó en este texto se desarrolló dentro del programa de Jóvenes Investigadores, apoyado por la Dirección de Investigaciones (DIN) de la Uptc.

INTRODUCCIÓN

R es un sistema para análisis estadísticos y gráficos, creado por Ross Ihaka y Robert Gentleman (1996); tiene una naturaleza doble, de programa y lenguaje de programación, y es considerado un dialecto del lenguaje S, creado por los Laboratorios AT&T Bell. S está disponible como el programa S-PLUS, comercializado por Insightful. R se distribuye gratuitamente bajo los términos de la GNU General Public Licence; su desarrollo y distribución son llevados a cabo por varios estadísticos conocidos, como el Grupo Nuclear de Desarrollo de R.

R es un conjunto integrado de programas para manipulación de datos, cálculo y gráficos. Entre otras características, dispone de:

- Almacenamiento y manipulación efectiva de datos.
- Operadores para cálculo sobre variables indexadas (Arrays), en particular matrices.
- Una amplia, coherente e integrada colección de herramientas para análisis de datos.
- Posibilidades gráficas para análisis de datos, que funcionan directamente sobre pantalla o impresora.
- Un lenguaje de programación bien desarrollado, simple y efectivo, que incluye condicionales, ciclos, funciones recursivas y posibilidad de entradas y salidas. (Debe destacarse que muchas de las funciones suministradas con el sistema están escritas en el lenguaje R).

El término “entorno” lo caracteriza como un sistema completamente diseñado y coherente, antes que como una agregación incremental de herramientas muy específicas e inflexibles, como ocurre frecuentemente con otros programas de análisis de datos.

R es en gran parte un vehículo para el desarrollo de nuevos métodos de análisis interactivo de datos. Como tal es muy dinámico, y las diferentes versiones no siempre son totalmente compatibles con las anteriores. Algunos usuarios prefieren los cambios debido a los nuevos

1. DESCARGA E INSTALACIÓN DE R

1.1 DESCARGA

La descarga del paquete R se puede realizar desde la página <http://www.cran.r-project.org>, haciendo clic en “Windows”, de la región denominada “Download and Install R”.

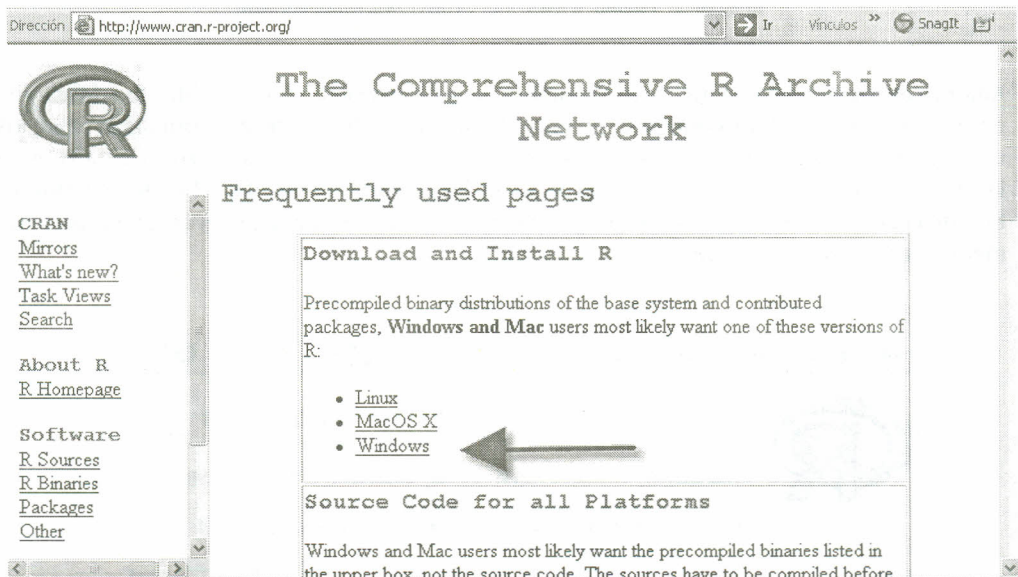


Imagen 1. Página www.cran.r-project.org

Luego de esto aparece una pantalla titulada “R for Windows”, en la cual se selecciona el subdirectorio base.



R for Windows

This directory contains binaries for a base distribution and packages to run on i386/x64 Windows.

Note: CRAN does not have Windows systems and cannot check these binaries for viruses. Use the normal precautions with downloaded executables.

Subdirectories:

- [base](#) Binaries for base distribution (managed by Duncan Murdoch)
- [contrib](#) Binaries of contributed packages (managed by Uwe Ligges)

Please do not submit binaries to CRAN. Package developers might want to contact Duncan Murdoch or Uwe Ligges directly in case of questions / suggestions related to Windows binaries.

You may also want to read the [R FAQ](#) and [R for Windows FAQ](#).

Last modified: April 4, 2004, by Friedrich Leisch

CRAN
[Mirrors](#)
[What's new?](#)
[Task Views](#)
[Search](#)

About R
[R Homepage](#)

Software
[R Sources](#)
[R Binaries](#)
[Packages](#)
[Other](#)

Imagen 2. Página www.cran.r-project.org/windows/base

Aparece enseguida una pantalla con el nombre de la última versión disponible para descargar, en este caso “R-2.11.1 for Windows”; en esta pantalla se puede encontrar información sobre los cambios de versión a versión que ha sufrido el paquete R, además se encuentra el instalador de la última versión; la descarga se realiza al pulsar el directorio “Download R-2.11.1 for Windows”. En este cuadro se puede ejecutar el programa directamente en el equipo, guardar en cualquier medio o cancelar la descarga.

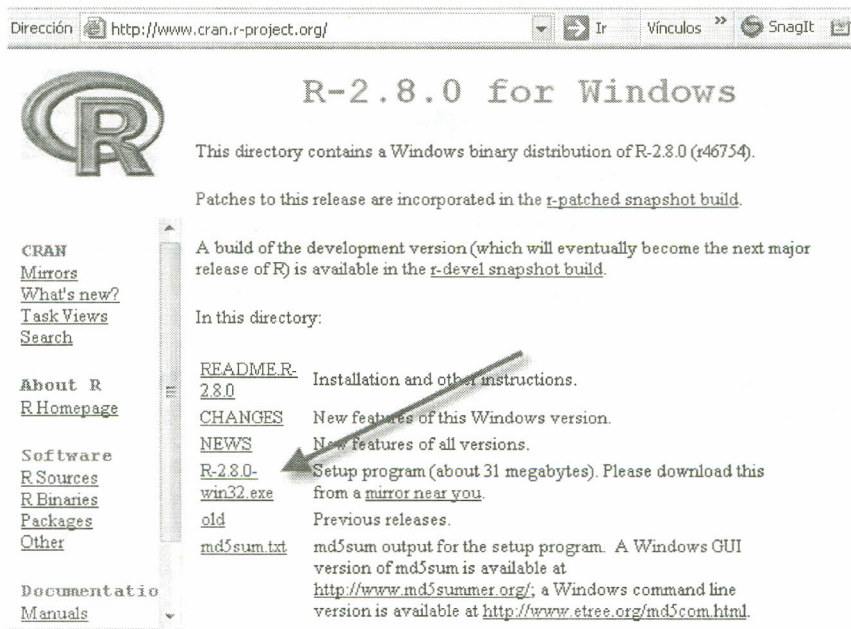


Imagen 3. Página www.cran.r-project.org/windows/base/R-2.11.1forwindows

Junto con R se incluyen algunos paquetes (llamados paquetes estándar), pero muchos otros están disponibles a través de Internet en <http://www.cran.r-project.org/>.

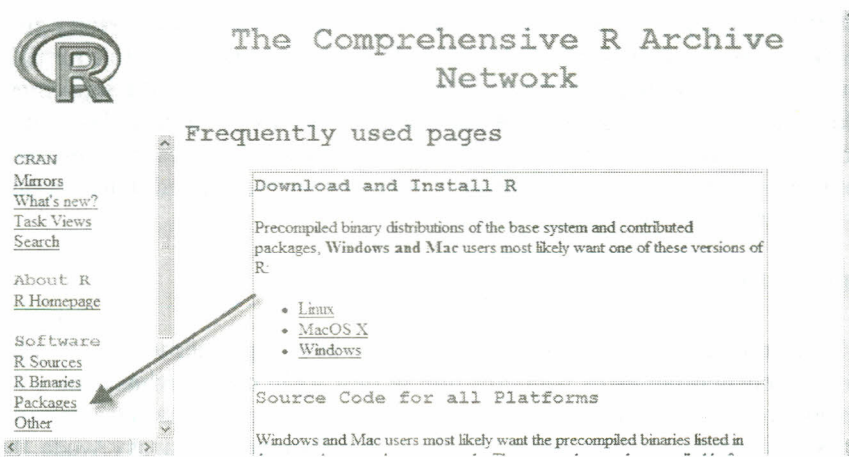


Imagen 4: Página www.cran.r-project.org/

En la parte inferior de esta página se pueden encontrar, según la necesidad del interesado, paquetes avanzados; aparece un listado con el nombre de cada uno de los paquetes y una pequeña descripción de su función. A continuación se presenta una parte de esa pantalla, aclarando que el listado de paquetes es solo el inicio y que este es bastante extenso.

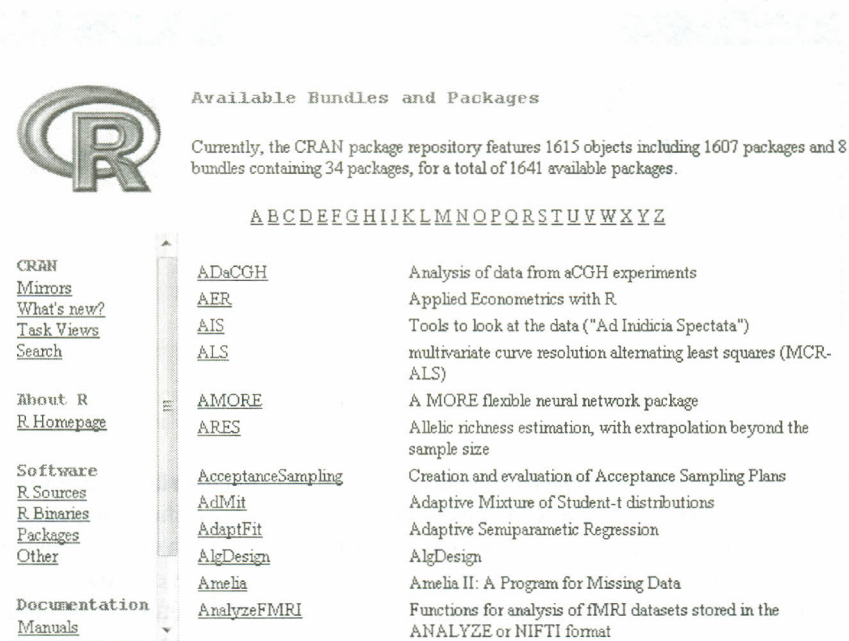


Imagen 5. Listado de paquetes disponibles en Internet

1.2 INSTALACIÓN

Haciendo clic en Inicio, se desplaza el mouse hasta la opción Mi PC, y se selecciona la ubicación del instalador (CD o memoria), si este fue descargado previamente de la Web y guardado en alguno de los medios mencionados; luego se da doble clic sobre el instalador (R-2.11.1 for Windows); enseguida se escoge el idioma en el cual se desea trabajar R; en las siguientes ventanas de diálogo que aparecen se debe hacer clic en aceptar hasta que aparezca la opción finalizar; con esto R estará instalado en el PC. En el proceso de instalación se crea un acceso directo en el escritorio que permite iniciar una sesión en R; alternativamente se puede iniciar sesión al hacer clic en Inicio / todos los programas / R / R 2.11.1

1.3 ESTANDO EN R

Ahora bien, como algunas rutinas necesitan de paquetes especializados, en la página de la imagen 5 se encuentran estos paquetes que se descargan como archivos zip y se pueden instalar abriendo una sesión en R; en la pantalla de R se encuentra una barra de herramientas que incluye las opciones Archivo, Editar, Visualizar, Misc, Paquetes, Ventanas y Ayuda. Un paquete se instala haciendo clic en la barra de herramientas en la opción Paquetes; al hacer clic se despliega un cuadro de diálogo del cual se selecciona “Instalar paquetes a partir de archivos zip locales”.

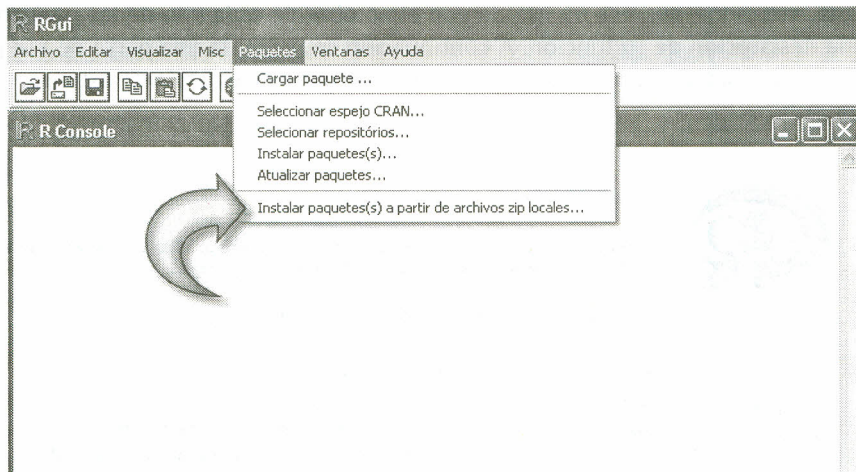


Imagen 6. Instalación de paquetes desde archivos

Esta opción despliega una ventana de diálogo en la cual se especifica la ubicación del archivo zip referente al paquete que se desea instalar; luego de esto, R especifica si el paquete ha sido instalado satisfactoriamente o no. Cuando se necesite hacer uso de cualquier paquete que haya sido cargado desde un archivo zip, es necesario invocarlo para que el programa utilice sus rutinas; esto se realiza en la barra de herramientas en el opción “Paquetes”, donde aparece una ventana de diálogo en la cual se debe seleccionar “cargar paquete”, esta muestra los paquetes disponibles en el momento, de allí se debe seleccionar el que sea necesario.

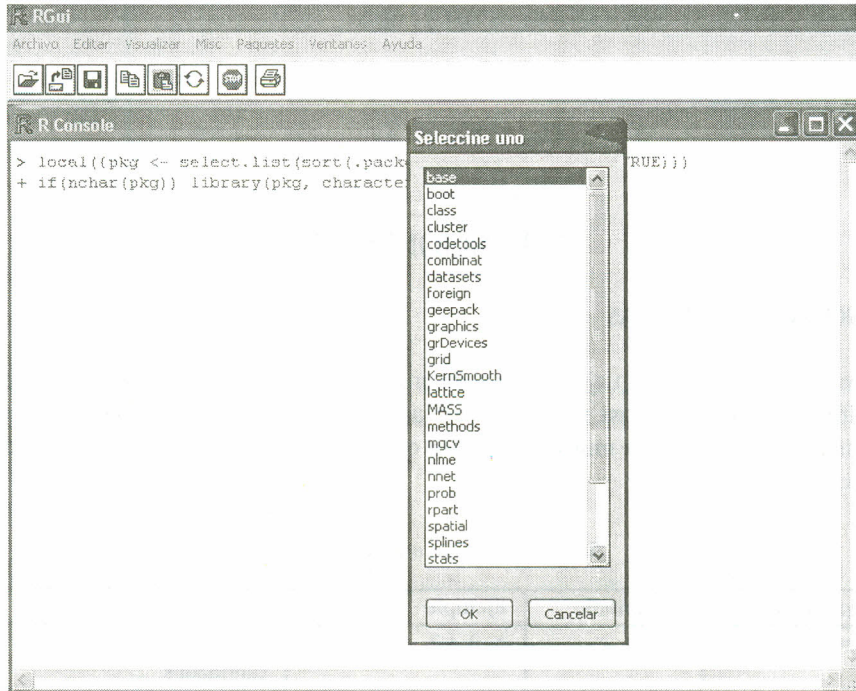


Imagen 7. Cargando un paquete en R

1.3.1 Utilización de R. Cuando R espera la entrada de órdenes, presenta el símbolo “>” para indicarlo. Para salir de R la orden es `q()`.

1.3.2 Mayúsculas y minúsculas en R. R es un lenguaje de expresiones con una sintaxis muy simple; distingue entre mayúsculas y minúsculas, de tal modo que `A` y `a` son símbolos distintos y se referirán, por tanto, a objetos distintos. Las órdenes elementales consisten en expresiones o en asignaciones; si una orden consiste en una expresión, se evalúa, se imprime y su valor se pierde; una asignación, por el contrario, evalúa una expresión, no la imprime y guarda su valor en una variable que, de necesitarse luego, se invoca colocando solo el nombre.

2. FUNCIONES EN R

2.1 FUNCIONES ARITMÉTICAS

El programa R utiliza un lenguaje similar al de una calculadora: por pantalla se puede digitar la operación que se requiere, así, se puede realizar desde una suma hasta calcular un logaritmo determinado. En la tabla 1 se muestran algunas operaciones con su respectiva instrucción y un ejemplo:

Tabla 1. Funciones aritméticas básicas

Operación	Símbolo	Ejemplo	Descripción
Suma y Resta	+ y -	> 3 + 2	Realiza la suma o resta de dos cantidades
Multiplicación	*	> 6 * 5	Realiza la multiplicación de dos cantidades
División	/	> 9 / 3	Realiza la división entre dos cantidades
Potenciación	^	> 3 ^ 5	Devuelve la potencia de un número
Raíz cuadrada	sqrt()	> sqrt(81)	Devuelve la raíz cuadrada de un número
Logaritmo	log(x,base)	> log(64,8)	Calcula el logaritmo de x en base "base"
Seno	sin()	> sin(radianes)	Calcula el seno para un ángulo determinado
Coseno	cos()	> cos(radianes)	Calcula el coseno para un ángulo determinado
Tangente	tan()	> tan(radianes)	Calcula la tangente para un ángulo determinado

2.2 COMANDOS ESPECIALES

2.2.1 Comando `help()`. Permite obtener ayuda sobre funciones específicas; se necesita tener el nombre de la función sobre la cual se desea obtener información; para utilizar esta ayuda se procede así:

`help(función)` o alternativamente `?función`

2.2.2 Asignación. Consiste en dar un nombre a un valor o a una determinada función, de tal manera que esta pueda ser utilizada más adelante en otras operaciones o con otras funciones más complicadas. La estructura para realizar la asignación es la siguiente:

```
Nombre<-valor o función
X<- 5 * 9 + 2
```

Nótese que en el anterior comando se escribe (<-) después del nombre al que se quiere asignar el valor o la función, esto es equivalente a utilizar (=); para ilustrar esto, en algunos comandos más adelante se trabaja con el (=) en lugar de (<-). Con lo anterior se ha asignado a un nombre específico un valor o función que se almacena en la memoria del computador, y si se quiere observar esta asignación en la pantalla se invoca colocando nuevamente el nombre de la asignación y dando un enter. La asignación puede realizarse también mediante la función assign(). Una forma equivalente de realizar la asignación anterior es

```
assign("X",5 * 9 + 2)
```

2.2.3 Comando c(). R utiliza diferentes estructuras de datos. La estructura más simple es el vector, que es una colección ordenada de números. Para crear un vector columna de cualquier dimensión, escriba los números dentro de los paréntesis del comando separados por comas; por ejemplo, si se desea un vector columna de tamaño cinco se procede así:

```
Y=c(1,2,3,4,5) o assign("Y",c(1,2,3,4,5))
```

2.2.4 Comando seq(). Permite generar sucesiones de números; para ello se debe indicar el número inicial, el número final y el incremento que se desee; la estructura para este comando es como sigue:

```
seq(valor inicial, valor final, incremento)
seq(1,5,0.5) "genera la sucesión de números del
1 al 5 con incrementos de 0.5"
```

```
>
> # Secuencia de 1 a 5 incrementando en 0.5
> seq(1,5,0.5)
[1] 1.0 1.5 2.0 2.5 3.0 3.5 4.0 4.5 5.0
>
```

Imagen 8. Salida R para generar secuencias

Si se necesita que esta sucesión se almacene dentro de un vector, lo anterior se escribe dentro del comando c() y se le asigna un nombre, para su utilización más adelante.

2.2.5 Comando rep(). Permite repetir el mismo número tantas veces como se desee; si se requiere obtener el resultado del comando dentro de un vector, entonces se utiliza el comando c() con su asignación respectiva, ejemplo:

2.3 VECTORES

2.3.1 Definición. Un vector es todo elemento de un espacio vectorial. Los vectores surgen como elementos de ciertas estructuras matemáticas previamente definidas. R utiliza diferentes estructuras de datos; la estructura más simple es el vector, que R considera como una colección ordenada de números. Un número, por sí mismo, se considera un vector de longitud uno.

$Y=c(\text{rep}(x,\text{número de veces}))$ $\text{Unos}=c(\text{rep}(1,5))$ "repite el 1 cinco veces"	<pre>> > Unos=c(rep(1,5)) > Unos [1] 1 1 1 1 1 ></pre>
---	--

Imagen 9. Salida R para repeticiones

2.3.2 Aritmética vectorial. Los vectores pueden usarse en expresiones aritméticas, en cuyo caso las operaciones se realizan elemento tras elemento. Dos vectores que se utilizan en la misma expresión no tienen por qué ser de la misma longitud. Si no lo son, el resultado será un vector de la longitud del más largo, y el más corto será reciclado, repitiéndolo tantas veces como sea necesario (puede que no un número exacto de veces) hasta que coincida con el más largo.

Los operadores aritméticos elementales son los habituales $+$, $-$, $*$, $/$ y $^$, para elevar a una potencia. Además, están disponibles las funciones \log , \exp , \sin , \cos , \tan , $\sqrt{}$, entre otras; estas se utilizan en operaciones aritméticas, y, además, se aplican a todos los elementos del vector. Si se tiene un vector llamado X , es posible definir las siguientes funciones:

Tabla 2. Funciones aritméticas vectoriales

Comando	Descripción
$\text{sum}(x)$	Suma de los elementos del vector
$\text{prod}(x)$	Multiplca los elementos del vector
$\text{max}(x)$	Valor máximo del vector
$\text{min}(x)$	Valor mínimo del vector
$\text{range}(x)$	Rango del vector o $c(\text{min}(x),\text{max}(x))$
$\text{length}(x)$	Número de elementos del vector
$\text{sort}(x)$	Ordena de menor a mayor los elementos del vector
$\text{rev}(\text{sort}(x))$	Ordena de mayor a menor los elementos del vector
$\text{round}(\text{vector},n)$	Redondea los elementos del vector a n cifras decimales
$\text{cumsum}(x)$	Vector donde cada elemento es la suma de él y sus cifras anteriores

A continuación se ilustran algunos de los comandos anteriores con ejemplos:

Ejemplo: Determinar el número de elementos de un vector

B=c(6,1,2,5,4,8,7,9,6,3)	<pre> > > B=c(6,1,2,5,4,6,7,9,6,3) > length(B) [1] 10 > </pre>
--------------------------	--

Imagen 10. Salida R para determinar el tamaño de un vector

Ejemplo: Realizar la suma de cada número en un vector con los números que le anteceden.

```

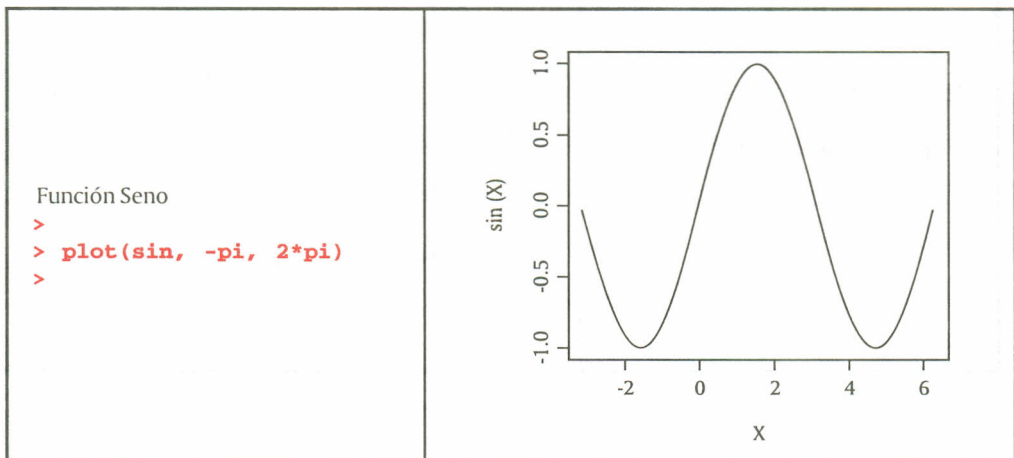
>
> X=c(1,2,3,4,5,6,7,8,9,10)
> cumsum(X)
[1] 1 3 6 10 15 21 28 36 45 55
>

```

Imagen 11. Salida R para el comando vectorial cumsum

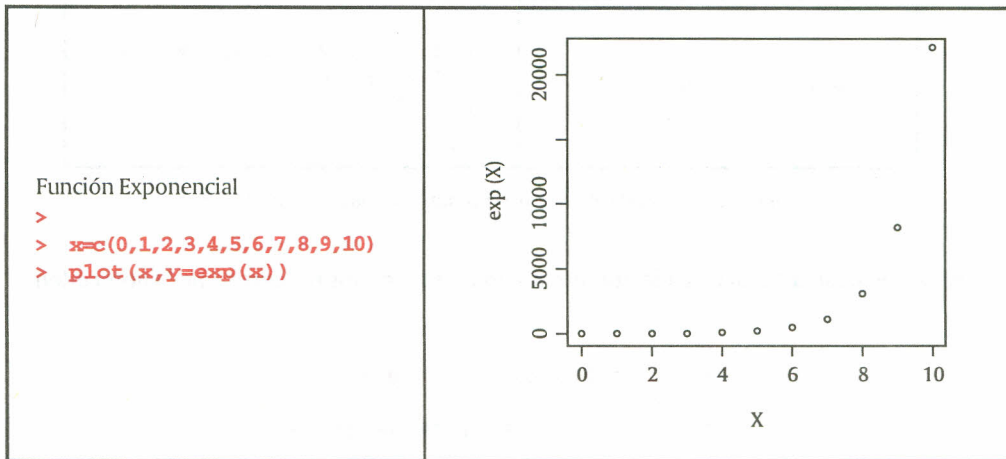
2.3.3 Gráficas de funciones. En R también es posible realizar gráficas de funciones conocidas como: seno (sin), coseno (cos), tangente (tan), exponenciales (exp) y casi cualquier clase de funciones; a continuación se muestran algunos ejemplos de estas gráficas.

Para realizar una gráfica de una función trigonométrica se utiliza el siguiente comando `plot(función, Rango)`; El argumento después de la función corresponde al rango, ejemplo:



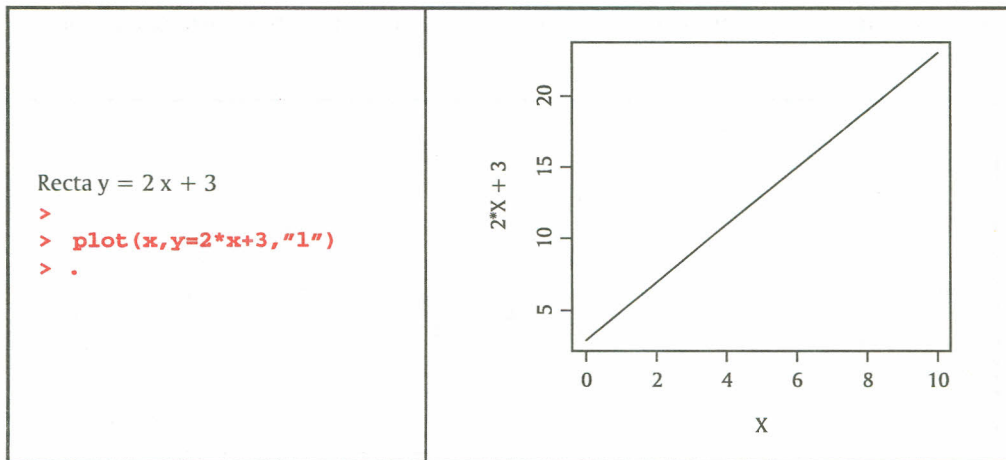
Imágenes 12 y 13. Salida R para creación de función Seno

Para las demás funciones es necesario construir un vector con los valores iniciales que se quieren evaluar mediante una función específica; el comando utilizado es `plot(vector inicial, función)`.



Imágenes 14 y 15. Salida R para creación de función Exponencial

Si el interés está en que los puntos de la gráfica aparezcan conectados mediante una línea, en el comando `plot()` se agrega la instrucción "l". Si se desea realizar la gráfica de la recta $y = 2x + 3$ utilizando el vector `x` del ejemplo anterior se procede así:



Imágenes 16 y 17. Salida R para creación de función lineal

2.3.4 Vectores lógicos. Los elementos de un vector lógico solo pueden tomar dos valores: FALSE (falso) y TRUE (verdadero). Los vectores lógicos aparecen al utilizar condiciones; los operadores lógicos son < (menor), = (menor o igual), > (mayor), >= (mayor o igual), == (igual), y != (distinto). Además, si `c1` y `c2` son expresiones lógicas, entonces `c1 & c2` es su intersección (conjunción), `c1 | c2` es su unión (disyunción) y `!c1` es la negación de `c1`. Por ejemplo, dado un vector `(1, 2, 3, 4, 5)` se establece la condición `x > 3`.

```

>
> X=c(1,2,3,4,5)
> mayores=x>3
> mayores
[1] FALSE FALSE FALSE TRUE TRUE
>

```

Imagen 18. Salida R para evaluación de valores lógicos

2.3.5 Vectores de caracteres. Las cadenas de caracteres, o frases, también son utilizadas en R; por ejemplo, para etiquetar gráficos. Una cadena de caracteres se construye escribiendo entre comillas la sucesión de caracteres que la define; por ejemplo: "Altura" o "Peso". La función `paste()` une todos los vectores de caracteres que se le suministran y construye una sola cadena de caracteres.

2.4 MATRICES

2.4.1 Definición. Si m y n son enteros positivos, entonces una matriz $m \times n$ (que se lee "m por n") es un arreglo rectangular. Una matriz $m \times n$ tiene m filas (líneas horizontales) y n columnas (líneas verticales).

$$A = [ij] = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix}$$

Imagen 19. Matriz

En R, una matriz es realmente un vector con un atributo adicional, dimensión (`dim`), el cual a su vez es un vector numérico de longitud 2, que define el número de filas y columnas de la matriz; además, el tamaño del vector debe ser igual al producto del número de filas por el número de columnas. Una matriz se puede crear con la función `matrix()`, teniendo los datos dentro de un vector, así:

`matrix(vector de datos, #filas, #columnas)`

Por defecto, R ordena los elementos del vector de datos en términos de vectores columna; si se desea realizar el ordenamiento del vector de datos por fila, se incorpora dentro del comando anterior la instrucción `byrow=T`.

<pre> > > X=c(1,2,3,4,5,6) > matrix(X,2,3) [,1] [,2] [,3] [1,] 1 3 5 [2,] 2 4 6 > </pre>	<pre> > > matrix(X,2,3,byrow=T) [,1] [,2] [,3] [1,] 1 2 3 [2,] 4 5 6 > . </pre>
--	--

Imágenes 20 y 21. Salida R para creación de matrices

En R existe otra forma para crear una matriz; la creación de dicha matriz es posible si se tiene un conjunto de vectores que, luego, con el comando `cbind()`, se encadenan en un arreglo rectangular, en donde cada columna representa un vector. Si se requiere que los vectores representen una fila de la matriz se utiliza el comando `rbind()`.

```

>
> x1=c(1,2,3)
> x2=c(4,5,6)
> x3=c(7,8,9)
> X=cbind(x1,x2,x3)
> X
      x1 x2 x3
[1,]  1  4  7
[2,]  2  5  8
[3,]  3  6  9
> .

```

Imagen 22. Salida R para creación de matrices con `cbind`

2.4.2 Matriz Identidad. Es aquella cuyos elementos en su diagonal principal son unos y los demás elementos son ceros. Para crear esta matriz en R se procede de la siguiente forma:

<p><code>diag(n)</code>, donde <code>n</code> indica el orden de la matriz</p>	<pre> > > Y=diag(3) > Y [,1] [,2] [,3] [1,] 1 0 0 [2,] 0 1 0 [3,] 0 0 1 > . </pre>
--	---

Imagen 23. Salida R para creación de matriz Identidad

También, al trabajar con la función `diag()`, si su argumento es una matriz, `diag(matriz)`, devuelve un vector formado por los elementos de la diagonal de esta. Si, por el contrario, su argumento es un vector (de longitud mayor que uno), `diag(vector)`, lo transforma en una matriz diagonal cuyos elementos en la diagonal principal son los elementos del vector. Por último, si se requiere

una matriz diagonal diferente a la identidad, entonces dentro del comando `diag()` se escribe primero el número que debe aparecer en la diagonal y luego el número que corresponde al orden de la matriz, así por ejemplo:

<p>Construcción de una matriz diagonal de tamaño cinco con el número cuatro en su diagonal</p>	<pre>> > diag(4,5) [,1] [,2] [,3] [,4] [,5] [1,] 4 0 0 0 0 [2,] 0 4 0 0 0 [3,] 0 0 4 0 0 [4,] 0 0 0 4 0 [5,] 0 0 0 0 4 > .</pre>
--	--

Imagen 24. Salida R para creación de matriz diagonal

2.5 OPERACIONES ARITMÉTICAS CON MATRICES

Las matrices pueden utilizarse en expresiones aritméticas, y el resultado es una matriz formada a partir de las operaciones elemento tras elemento de las matrices involucradas. Las dimensiones de los operadores deben ser iguales en general y coincidirán con la dimensión de la matriz resultado; luego se tiene:

A + B, realiza la suma elemento a elemento de las matrices A y B

A - B, realiza la resta elemento a elemento de las matrices A y B

A * B, realiza la multiplicación elemento a elemento de las matrices A y B

A / B, realiza la división elemento a elemento de las matrices A y B

$\begin{bmatrix} 2 & 5 & 6 \\ 3 & 2 & 1 \\ 8 & 4 & 5 \end{bmatrix} + \begin{bmatrix} 6 & 1 & 4 \\ 2 & 1 & 1 \\ 1 & 3 & 2 \end{bmatrix} = \begin{bmatrix} 8 & 6 & 10 \\ 5 & 3 & 2 \\ 9 & 7 & 7 \end{bmatrix}$	<pre>> > X=matrix(c(2,3,8,5,2,4,6,1,5),3) > Y=matrix(c(6,2,1,1,1,3,4,1,2),3) > Z=X+Y > Z [,1] [,2] [,3] [1,] 8 6 10 [2,] 5 3 2 [3,] 9 7 7 ></pre>
--	---

Imágenes 25 y 26. Salida R para suma de matrices

2.5.1 Multiplicación por un escalar. Los números suelen denominarse escalares; estos serán números reales, a no ser que se determine otra cosa; la multiplicación de un escalar por una matriz se realiza al multiplicar el escalar por cada uno de los elementos que componen la matriz. En R, la multiplicación de una matriz por un escalar se lleva a cabo mediante la siguiente secuencia:

<p>Teniendo en cuenta la matriz Z del ejemplo anterior: $L = 5 * Z$, realiza la multiplicación del escalar 5 por la matriz Z</p>	<pre>> L = 5*Z > L [,1] [,2] [,3] [1,] 40 30 50 [2,] 25 15 10 [3,] 45 35 35 ></pre>
--	--

Imagen 27. Salida R para la multiplicación por escalar

2.5.2 Multiplicación de matrices. Si $A = [a_{ij}]$ es una matriz $m \times n$ y $B = [b_{ij}]$ es una matriz $n \times p$, entonces, el producto AB es una matriz $m \times p$. Nótese que para que el producto esté definido, el número de columnas de la primera matriz debe ser igual al número de filas de la segunda matriz. En R, el operador “%*%” permite realizar el producto entre dos matrices; un ejemplo del uso de este operador se muestra a continuación:

$A = \begin{bmatrix} 1 & 2 \\ 3 & 2 \end{bmatrix} \text{ y } B = \begin{bmatrix} 1 & 2 & 0 \\ 2 & 1 & 1 \end{bmatrix}$	<pre>> > A=matrix(c(1,3,2,2),2) > B=matrix(c(1,2,2,1,0,1),2,3) > C=A%*%B > C [,1] [,2] [,3] [1,] 5 4 2 [2,] 7 8 2 ></pre>
--	---

Imágenes 28 y 29. Salida R para el producto de matrices

Una matriz de $n \times 1$ ó $1 \times n$ puede ser utilizada como un vector n dimensional en caso necesario. Análogamente, R puede usar automáticamente un vector en una operación matricial, convirtiéndolo en una matriz fila o una matriz columna cuando ello es posible.

2.5.3 Traspuesta de una matriz. La traspuesta de una matriz se forma al escribir sus columnas como filas. Por ejemplo, si A es la matriz de orden $m \times n$, entonces la traspuesta, denotada por A^t , es la matriz de orden $n \times m$. El comando que permite calcular la traspuesta de una matriz en R es $t(\text{nombre de la matriz})$.

<p style="text-align: center;">Matriz C</p> <pre>> C [,1] [,2] [,3] [1,] 5 4 2 [2,] 7 8 2 ></pre>	<p style="text-align: center;">Traspuesta de C</p> <pre>> t(C) [,1] [,2] [1,] 5 7 [2,] 4 8 [3,] 2 2 ></pre>
---	---

Imágenes 30 y 31. Salida R para obtener la traspuesta

2.5.4 Inversa de una matriz. Una matriz A $n \times n$ es invertible (o no singular) si hay una matriz B $n \times n$ tal que $AB = BA = I_n$, donde I_n es la matriz identidad de orden n . La matriz B se denomina inversa multiplicativa de A . La inversa de A se denota A^{-1} . Una matriz que no tiene inversa se denomina no invertible (o singular). El comando utilizado en R para encontrar la matriz inversa es `solve(matriz)`. A continuación se ilustra esto.

$\text{Sea } D = \begin{bmatrix} 1 & 3 & 4 \\ 2 & 5 & 1 \\ 3 & 1 & 2 \end{bmatrix}$	<pre>> > D=matrix(c(1,2,3,3,5,1,4,1,2),3) > Inv=solve(D) > Inv [,1] [,2] [,3] [1,] -0.19565217 0.04347826 0.36956522 [2,] 0.02173913 0.21739130 -0.15217391 [3,] 0.28260870 -0.17391304 0.02173913 > .</pre>
---	--

Imágenes 32 y 33. Salida R para calcular la matriz Inversa

2.5.5 Determinante. Toda matriz cuadrada puede asociarse con un número real denominado su determinante. Si A es una matriz cuadrada (de orden mayor o igual a 2), entonces el determinante de A es la suma de los elementos en el primer renglón de A multiplicados por sus cofactores. En R, el comando que permite calcular el determinante de una matriz cuadrada es `det(nombre de la matriz)`; recuerde que si el determinante de una matriz es cero, la matriz se denomina singular.

$\text{Sea } A = \begin{bmatrix} 4 & 1 \\ 3 & 2 \end{bmatrix}$	<pre>> > A=matrix(c(4,3,1,2),2) > det(A) [1] 5 ></pre>
--	--

Imágenes 34 y 35, Salida R para repeticiones

2.5.6 Valores y vectores propios. En álgebra lineal, los vectores propios, autovectores o eigenvalores de un operador lineal son los vectores no nulos, que cuando son transformados por el operador dan lugar a un múltiplo escalar de sí mismos, con lo que no cambian su dirección. Este escalar se recibe el nombre de valor propio, autovalor, valor característico o eigenvalor. A menudo, una transformación queda completamente determinada por sus vectores propios y valores propios. Un espacio propio, autoespacio o eigenespacio, es el conjunto de vectores propios con un valor propio común. En R, el comando que permite calcular los valores y vectores propios de una matriz es `eigen(matriz)`; si se desea que solo aparezcan los valores propios dentro del comando anterior luego de la matriz, se le da la instrucción `only.values=TRUE`, es decir: `eigen(matriz,only.values=TRUE)`, otra forma equivalente es `eigen(matriz)$val`. Por defecto, el comando anterior arroja los vectores normalizados, si se requiere se puede pedir que estos vectores estén sin normalizar, incluyendo dentro del comando la instrucción `EISPACK=TRUE`. Por ejemplo, al considerar la matriz A utilizada anteriormente, se tiene:

Vectores normalizados	Vectores sin normalizar
<pre>> > A=matrix(c(4,3,1,2),2) > eigen(A) \$Values [1] 5 1 \$vectors [,1] [,2] [1,] 0.7071068 -0.3162278 [2,] 0.7071068 0.9486833</pre>	<pre>> eigen(A,EISPACK=TRUE) \$values [1] 5 1 \$vectors [,1] [,2] [1,] 0.7071068 -0.3535534 [2,] 0.7071068 1.0606602</pre>

Imágenes 36 y 37: Salida R para valores y vectores propios

2.5.7 Comando `apply()`. Este comando permite aplicar una función específica a las filas o columnas de una matriz; esta selección se puede realizar en el comando, mediante la dimensión (1 = fila, 2 = columna); ejemplo:

<p>Apply (matriz, dimensión, función) Apply (B,2,sum), calcula la suma de cada columna de la matriz B</p>	<pre>> > B=matrix(c(1,2,3,4,5,6,7,8,9,10,11,12),4,3) > B [,1] [,2] [,2] [1,] 1 5 9 [2,] 2 6 10 [3,] 3 7 11 [4,] 4 8 12 > apply(B,2,sum) [1] 10 26 42 ></pre>
---	--

Imagen 38. Salida R para aplicación comando `apply`

2.5.8 Partición de matrices. En algunos casos se tienen variables dentro de un arreglo matricial y se desea trabajar solamente con una de estas variables; para esto se hace necesario particionar la matriz; en R es posible tomar un solo elemento de la matriz, una columna, una fila o un arreglo matricial de menor dimensión que la matriz inicial. A continuación se muestra el comando utilizado para realizar lo descrito anteriormente; considérese la siguiente matriz:

```
>
> A=matrix(c(1,2,3,4,5,6,7,8,9),3)
> A
      [,1] [,2] [,2]
[1,]    1    4    7
[2,]    2    5    8
[3,]    3    6    9
>
```

Imagen 39. Salida R para creación de matriz

A continuación se presentan algunas particiones de la matriz A:

```
> # Elemento primera fila, segunda columna
>
> A[1,2]
[1] 4
>
> # Tomar una columna de la matriz
> # Columna número 3
>
> A[,3]
[1] 7 8 9
>
> # Tomar una fila de la matriz
> # Fila número 2
>
> A[2,]
[1] 2 5 8
>
>
> # Tomar un arreglo matricial
> # filas 1 y 2, columnas 2y3.
>
> A[1:2.2:3]
      [,1] [,2]
[1,]    4    7
[2,]    5    8
```

Imagen 40. Salida R para particionar una matriz

A continuación se presentan ejercicios para ser desarrollados en R, y en el siguiente capítulo se presentan algunos temas de estadística descriptiva que pueden ser trabajados en R.

2.6 EJERCICIOS

2.6.1 Con los siguientes datos construya un vector y asígnele la letra N.

43 76 30 90 12 38 74 46 15 45 25 67 53

2.6.2 Genere una secuencia de números que inicie en 8 y termine en 100, con incrementos de 2.

2.6.3 Los siguientes datos hacen referencia al peso de algunos estudiantes encuestados:

58 54 40 47 60 61 43 50 42 49 53 39 41
51 70 48 43 55 44 60 52 46 49 42 52 48

Obtenga:

- a. El número de estudiantes encuestados
- b. La suma de los pesos de los estudiantes
- c. El peso menor y el mayor
- d. Ordene los pesos de menor a mayor

2.6.4 Dada la Matriz A

$$\begin{bmatrix} 2 & 7 & 0 \\ 4 & 3 & 1 \\ 6 & 8 & 2 \end{bmatrix}$$

- a. Construya esta matriz en R
- b. Multiplique la matriz A por el escalar -3
- c. Construya la matriz traspuesta de A
- d. Calcule A^{-1}
- e. Calcule el determinante de A
- f. Realice el producto matricial entre A y A^{-1}

3. ESTADÍSTICA DESCRIPTIVA

La estadística descriptiva es una parte de la estadística que se dedica al ordenamiento y tratamiento de la información para su presentación por medio de tablas y de representaciones gráficas, así como a la obtención de algunos parámetros útiles para la explicación de la información. En este contexto, R brinda muchas alternativas para calcular medidas descriptivas de una población o muestra; a continuación se presentan algunas de estas.

3.1 TABLA DE DISTRIBUCIÓN DE FRECUENCIAS

Se entiende como el agrupamiento de datos en categorías, el cual muestra el número de observaciones en cada categoría mutuamente excluyente. Cada una de estas categorías es llamada intervalo de clase; los intervalos de clase usados en la distribución de frecuencias deben ser iguales. Para determinar la amplitud de un intervalo de clase se utiliza la fórmula $\text{int} = (\text{valor más alto} - \text{valor más bajo}) / \text{número de clases}$. En R es posible construir la tabla de distribuciones de frecuencias de la siguiente manera:

```
>
> y=c(15,23.7,19.7,15.4,18.3,23,14.2,20.8,13.5,20.7,17.4,18.6,12.9,20.3,13.7)
>
> range(y) # valor mínimo y valor máximo
[1] 12.9 23.7
>
> rango=23.7-12.9
> rango
[1] 10.8
>
> n=6 # número de clases
>
> amplitud=rango/n # amplitud de clase
> amplitud
[1] 1.8
>
> table(cut(y,breaks=seq(12,24,2),right=TRUE))

(12,14] (14,16] (16,18] (18,20] (20,22] (22,24]
      3      3      1      3      3      2
>
```

Imagen 41. Salida R para creación de tabla de frecuencias

En el ejemplo anterior se hace necesario calcular primero la amplitud de clase, y de acuerdo con esta se puede llegar a modificar el inicio de la primera clase y el final de la última clase, mediante la instrucción `breaks`; la instrucción `right=TRUE` indica que el intervalo es cerrado a la derecha. Si se necesita determinar la tabla de frecuencias acumuladas, el comando `table` se escribe dentro del comando `cumsum()`, como se muestra a continuación.

```
>
> table(cut(y,breaks=seq(12,24,2),right=TRUE))

(12,14] (14,16] (16,18] (18,20] (20,22] (22,24]
      3      3      1      3      3      2
>
> cumsum(table(cut(y,breaks=seq(12,24,2),right=TRUE)))

(12,14] (14,16] (16,18] (18,20] (20,22] (22,24]
      3      6      7     10     13     15
```

Imagen 42. Salida R para creación de tabla de frecuencias acumuladas

Si el interés está en ver las tablas de frecuencia relativa y frecuencia relativa acumulada, entonces las tablas obtenidas con los comandos `table` y `cumsum` se dividen entre el tamaño del vector de datos; a continuación un ejemplo:

```
>
> table(cut(y,breaks=seq(12,24,2),right=TRUE)) # tabla de frecuencias

(12,14] (14,16] (16,18] (18,20] (20,22] (22,24]
      3      3      1      3      3      2
>
> table(cut(y,breaks=seq(12,24,2),right=TRUE))/length(y) # Frec Relativa

      (12,14]      (14,16]      (16,18]      (18,20]      (20,22]      (22,24]
0.20000000  0.20000000  0.06666667  0.20000000  0.20000000  0.13333333
>
> cumsum(table(cut(y,breaks=seq(12,24,2),right=TRUE))/length(y))# F R A
      (12,14]      (14,16]      (16,18]      (18,20]      (20,22]      (22,24]
0.2000000  0.4000000  0.4666667  0.6666667  0.8666667  1.0000000
>
```

Imagen 43. Salida R para creación de tabla de frecuencias relativas

3.2 FUNCIONES GRÁFICAS BÁSICAS PARA EL ANÁLISIS EXPLORATORIO DE DATOS

Los métodos gráficos proporcionan al investigador un conjunto de formas sencillas para examinar tanto las variables de manera individual como las relaciones entre ellas. Los métodos gráficos se distinguen según la cantidad de variables que se analizan; a continuación se presentan algunos gráficos para el análisis exploratorio de datos y su correspondiente algoritmo para ser trabajado en R.

3.2.1 Diagrama de sectores. Utilizado para variables de tipo cualitativas (también llamado circular). Se divide un círculo en tantas porciones como clases existan, de modo que a cada clase le corresponde un arco de círculo proporcional a su frecuencia absoluta o relativa.

Ejemplo: En una encuesta realizada a 10 personas sobre la preferencia que tienen hacia algún deporte, los resultados son los siguientes: 4 optaron por fútbol; 2, por básquet; 3, por voleibol, y 1, por tenis; de acuerdo con los datos, la fracción del diagrama que le corresponde a cada deporte es: fútbol = 0.4, básquet = 0.3, voleibol = 0.3 y tenis = 0.1; con esta información se construyen los vectores necesarios para realizar el diagrama en R.

```
> X = c(0.4,0.3,0.3,0.1)
> nombres = c("Fútbol - 40%", "Básquet - 30%", "Voleibol - 30%", "Tenis - 10%")
```

El primer vector se refiere a la fracción del diagrama por deporte, y el segundo vector contiene el nombre del deporte y su respectivo porcentaje; con esto se procede a realizar el gráfico en R así:

```
>
> X=c(0.4,0.3,0.3,0.1)
> nombres=c("Futbol - 40%", "Basket - 30%", "Voleibol - 30%", "tenis - 10%")
> pie(X, labels=nombres)
>
```

Imagen 44. Salida R para la creación del diagrama de sectores

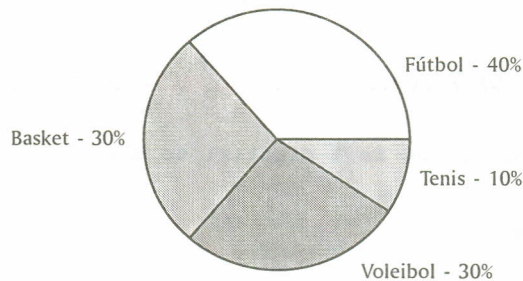


Imagen 45. Salida R diagrama de sectores

3.2.2 Diagrama de barras. Se utiliza para representar los caracteres cualitativos y cuantitativos discretos. En el eje horizontal, o eje de abscisas, se representan los datos o modalidades; en el eje vertical, o de ordenadas, se representan las frecuencias de cada dato o modalidad. Las frecuencias pueden ser absolutas, relativas y relativas acumuladas. Teniendo en cuenta los mismos datos del ejemplo de preferencias de deporte utilizado en el diagrama de sectores, el siguiente comando permite realizar el diagrama de barras en R:

```
barplot(X, names.arg = nombres)
```

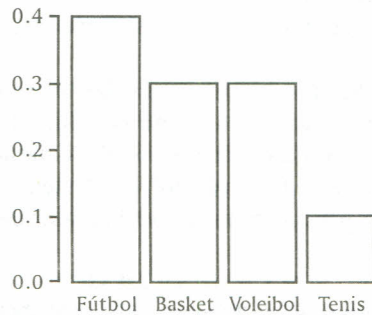


Imagen 46. Salida R diagrama de barras

X se refiere a la fracción de cada deporte, y **names.arg**, al nombre de la fracción en el vector del ejemplo anterior.

3.2.3 Diagrama de tallos y hojas. Utilizado para variables de tipo numérico; uno de los objetivos es descubrir un patrón de comportamiento de los datos, es decir, qué distribución de probabilidad pueden seguir los datos. Es aplicable para valores formados por al menos dos cifras, y su principio es que cada número se divide en dos partes, una llamada “Tallo”, y la otra, “ramas u hojas”. En R, el comando que permite realizar este diagrama es **stem(vector)**.

Ejemplo: Considerar los números 65, 57, 79, 69, 53, 63, 71, 81, 64, 85, 72, 59, 90, 51, 68. Los tallos serán las decenas, y las ramas serán las unidades; la instrucción en R es:

```
>
> T=c(65,57,79,69,53,71,81,64,85,72,59,90,51,68)
> stem(T)

The decimal point is 1 digit(s) to the right of the |

5 | 1379
6 | 34589
7 | 129
8 | 15
9 | 0
```

Imagen 47. Salida R diagrama de tallos y hojas

3.2.4 Diagrama de Caja. Utilizado para variables de tipo numérico. Es un gráfico representativo de las distribuciones de un conjunto de datos en cuya construcción se usan cinco medidas descriptivas de estos, a saber: mediana, primer cuartil, tercer cuartil, valor máximo y valor mínimo; permite identificar de forma individual observaciones que se alejan del resto de los datos; a estas observaciones se les conoce como valores atípicos. El comando que permite realizar este gráfico es **boxplot(vector)**.

Se tiene por ejemplo un vector G cuyos datos son: $>G=c(29, 78, 48, 29, 30, 44, 72, 73, 46, 82, 84, 71, 75, 84, 45, 45, 47, 35, 33, 54, 56, 33, 62, 63, 64, 36)$

`>boxplot(G)`

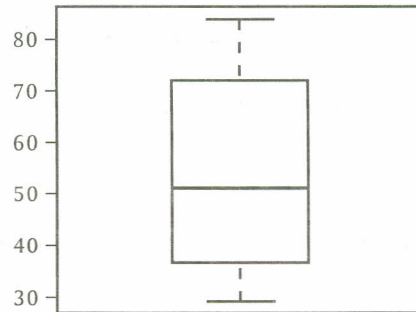


Imagen 48. Salida R diagrama de caja

3.2.5 Histograma. Es el gráfico estadístico que se utiliza para representar datos continuos cuando vienen agrupados en intervalos. Sobre cada uno de estos intervalos se levanta una franja tan ancha como el intervalo y de forma que su área sea proporcional a su frecuencia. El comando que permite realizar un histograma en R es `hist(vector)`.

Con los datos del diagrama de caja, el histograma correspondiente se obtiene así:

`>hist(G)`

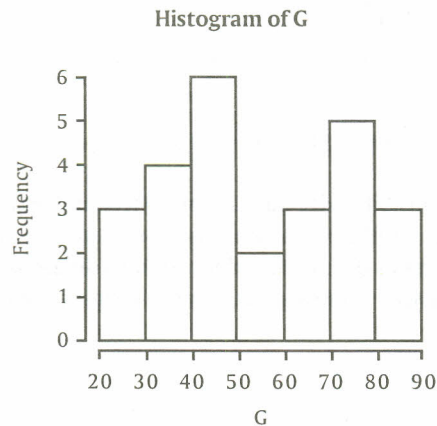


Imagen 49. Salida R histograma

3.2.6 Dispersograma. Gráfico bidimensional usado para variables cuantitativas. Consiste en dos ejes perpendiculares; en cada uno de ellos se ubican los valores de cada una de las variables.

Ejemplo: Los siguientes datos representan las calificaciones de matemáticas para una muestra aleatoria de 12 alumnos de primer grado de cierta universidad, junto con sus calificaciones de

una prueba de inteligencia que se les aplicó cuando aún eran alumnos del último año de bachillerato:

Calificación prueba de inteligencia (x) = 65, 50, 55, 65, 55, 70, 65, 70, 55, 70, 50, 55.

Calificación prueba de matemáticas (y) = 85, 74, 76, 90, 85, 87, 94, 98, 81, 91, 76, 74.

A continuación la sintaxis para realizar el diagrama de dispersión de estos conjuntos de datos y su respectivo gráfico.

```
>
> X=c(65,50,55,65,55,70,65,70,55,70,50,55)
> Y=c(85,74,76,90,85,87,94,98,81,91,76,74)
>
> plot(X,Y)
```

Imagen 50. Salida R creación de dispersograma

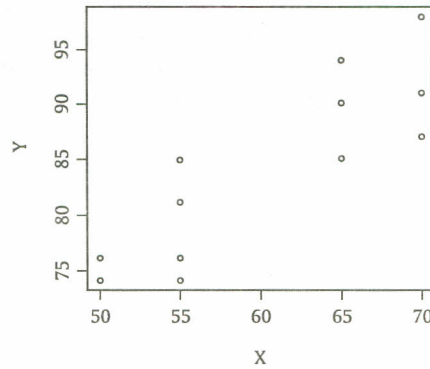


Imagen 51. Salida R dispersograma

Las funciones gráficas tienen posibilidades para ser modificadas, hasta ahora solo se ha definido el comando básico para realizar el gráfico con una instrucción. Para incluir opciones en las gráficas, estas se escriben después del nombre del conjunto de datos; la estructura por utilizar es **función-gráfica(datos, opción = parámetro)**; la tabla siguiente muestra algunas de estas opciones:

Tabla 3. Opciones gráficas

Opción	Descripción
main = "título"	Título principal; debe ser de tipo carácter
sub = "subtítulo"	Subtítulo (escrito en letra más pequeña)
xlab=", ylab="	Títulos en los ejes: deben ser variables de tipo carácter
xlim =, ylim =	Especifica los límites inferiores y superiores de los ejes
axes = TRUE	Si es FALSE no dibuja los ejes ni la caja del gráfico
col = "color"	Le da un color específico a los puntos o a las líneas

El tipo de gráfico también es posible determinarlo para los dispersogramas, de la siguiente forma:

Tabla 4. Opciones gráficas para el dispersograma

Opción	Descripción
type = "p"	Puntos
type = "l"	Líneas
type = "b"	Puntos conectados por líneas
type = "o"	Igual al anterior, pero las líneas están sobre los puntos
type = "h"	Líneas verticales
type = "s"	Escaleras, los datos se representan como la parte superior de las líneas verticales
type = "S"	Escaleras, los datos se representan como la parte inferior de las líneas verticales

También es posible cambiar los puntos por otra figura (como una de las que se aprecian en la imagen 52); para esto se introduce la opción **pch** en el comando **plot()**, cada figura aparecerá de acuerdo con el número que se especifique después del comando **pch=número**. Las figuras y su correspondiente número son:



Imagen 52. Opciones para el argumento pch

Con el fin de visualizar algunas de las figuras mencionadas, se presenta un diagrama de dispersión empleando la figura de diamante.

Ejemplo: El supervisor de mantenimiento de una línea de autobuses cree que existe una relación entre el costo anual de mantenimiento de las unidades y los años que llevan de operación. Considera que si tal relación existe podrá hacer un mejor pronóstico de presupuesto. Los datos tomados por el supervisor sobre 15 autobuses de la empresa se muestran a continuación:

x = c(8, 5, 3, 9, 11, 2, 1, 8, 12, 4, 7, 10, 6, 3, 9)
y = c(8.6, 6.8, 4.7, 7, 11, 2.2, 3.2, 6.5, 10.5, 5.6, 6.8, 10.8, 6.2, 5, 8)

```
> plot (x,y,main="Diagrama de dispersión",
+ xlab="Tiempo de operación",ylab="Costo de mantenimiento",
+ col="red",col.main="blue",pch=18)
```

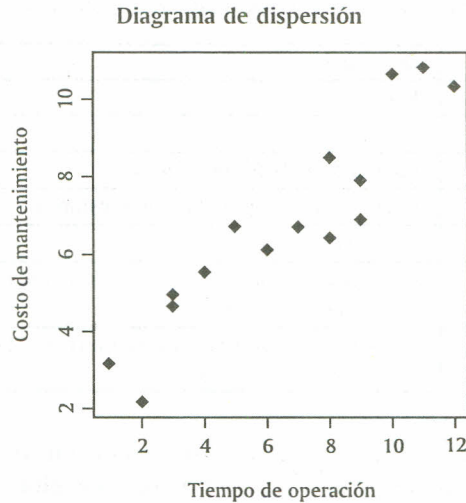


Imagen 53. Salida R argumentos gráficos

3.3 MEDIDAS DE TENDENCIA CENTRAL

Al describir grupos de observaciones, con frecuencia se desea describir el grupo con un solo número; desde luego, no se usará el valor más elevado ni el valor más pequeño como único representante, ya que solo representan los extremos y no los valores que generalmente tienen mayor ocurrencia en una población; sería más adecuado buscar un valor central. Las medidas que describen un valor típico en un grupo de observaciones suelen llamarse medidas de tendencia central. Es importante tener en cuenta que estas medidas se aplican a grupos más que a individuos. Un promedio es una característica de grupo, no individual. En los ejemplos para las medidas de tendencia central se utilizará el siguiente conjunto de datos localizados en el vector W:

```
>
> W=c(24,30,32,35,48,16,14,15,21,32,30,25,26,25,30,16,15,17,21,20,30)
>
```

Imagen 54. Salida R vector de trabajo

3.3.1 Media aritmética. La medida de tendencia central más utilizada que se puede elegir es el valor obtenido sumando las observaciones y dividiendo esta suma por el número de observaciones que hay en el grupo; en R, el comando que calcula directamente la media aritmética es: `mean(vector)`. La media aritmética en R se calcula como se muestra a continuación:

$\bar{x} = \frac{\sum_{i=1}^n x_i}{n}$	<pre> > > mean(W) [1] 24.85714 > > media=sum(W)/length(W) > media [1] 24.85714 > </pre>
--	---

Ecuación 1 e imagen 55. Salida R para la media aritmética

3.3.2 Mediana. Definida como el valor de la variable que deja el mismo número de datos antes y después que él.

```

>
>
median(W)
[1] 25
>

```

Imagen 56. Salida R para la mediana

3.3.3 Moda. Es el dato que más se repite en un conjunto de datos. Si existen dos datos que se repiten un número igual de veces, entonces, el conjunto será bimodal. Al utilizar el comando `table(vector)` se construye una tabla donde se aprecian cada uno de los valores diferentes de la variable y su frecuencia. Considere el vector “y”:

```

>
>
y=c(1,5,6,8,2,4,1,2,2,3,2,6,2)
> table(y)
y
1 2 3 4 5 6 8
2 5 1 1 1 2 1
>

```

Imagen 57. Salida R para la moda

En los datos anteriores se observa que el valor que más se repite es el 2, pues su frecuencia es 5; por lo tanto, este valor es la moda para este conjunto de datos.

3.4 MEDIDAS DE DISPERSIÓN

Se llaman medidas de dispersión a aquellas que permiten expresar la distancia de los valores de la variable a un cierto valor central, o que permiten identificar la concentración de los datos en un cierto sector del recorrido de la variable. Se trata del coeficiente para variables cuantitativas.

3.4.1 Varianza. Es el valor obtenido de sumar los cuadrados de las desviaciones de cada uno de los datos respecto a la media y dividir esta suma por el número de observaciones menos uno; en

R, el comando que calcula directamente la varianza es `var(vector)`, o bien, se puede programar así, teniendo en cuenta el vector de datos G dado en los ejemplos sobre tipos de gráficas, se tiene:

$\text{Var}(x) = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n - 1}$	<pre>> > Var=sum(c(G-mean(G))^2)/(length(G)-1) > var [1] 344.8185 > > var(G) [1] 344.8185 > .</pre>
--	---

Ecuación 2 e imagen 58. Salida R para la varianza

3.4.2 Desviación estándar. Es la raíz cuadrada de la varianza; el comando `sd(G)` hace el cálculo directo de la desviación estándar sobre un vector numérico.

$\text{desvest} = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n - 1}}$	<pre>> > sqrt(var(G)) [1] 18.56929 > > sd(G) [1] 18.56929 ></pre>
--	--

Ecuación 3 e imagen 59. Salida R para la desviación estándar

3.4.3 Cuantiles. Se usan con frecuencia en los datos para dividir las poblaciones en grupos. Por ejemplo, se puede utilizar el primer cuantil para determinar cuál valor deja un 25 por ciento de datos por debajo de él, esto se observa a continuación:

```
>
> quantile(G)
 0%   25%   50%   75%  100%
29.00 38.00 51.00 71.75 84.00
>
```

Imagen 60. Salida R para los cuantiles

Si el interés recae en calcular los percentiles de la variable, se utiliza la función

```
quantile(variable,seq(valor inicial,valor final, incremento))
```

Luego los percentiles para el vector G son:

```
>
> quantile(G,seq(0.1,0.9,0.1))
 10%  20%  30%  40%  50%  60%  70%  80%  90%
31.5  35.0  44.5  46.0  51.0  62.0  67.5  73.0  80.0
>
```

Imagen 61. Salida R para los percentiles

El comando **summary** permite calcular directamente y a la vez algunas de las medidas de tendencia central y de dispersión como: media, mediana, primer cuartil, tercer cuartil, valor mínimo y valor máximo de un conjunto de datos. Así como se muestra:

```
>
> x = c(8,5,3,9,11,2,1,8,12,4,7,10,6,3,9)
>
> summary(x)
  Min. 1st Qu. Median Mean 3rd Qu.  Max.
 1.000  3.500   7.000 6.533   9.000 12.000
>
```

Imagen 62. Salida R para obtener resumen general

3.5 MEDIDAS DE ASIMETRÍA

Comparan la forma que tiene la representación gráfica, bien sea el histograma o el diagrama de barras de la distribución, con la distribución normal.

3.5.1 Sesgo. Diremos que una distribución es simétrica cuando su mediana, su moda y su media aritmética coinciden. El sesgo mide la simetría de la distribución de un conjunto de datos; puede ser negativo, cero o positivo. Una fórmula para calcular este coeficiente de simetría es la siguiente:

$$\text{sesgo} = \frac{n}{(n-1)(n-2)} \sum_{i=1}^n \left(\frac{(x_i - \bar{x})}{s} \right)^2$$

Ecuación 4. Sesgo

El algoritmo de la fórmula anterior se puede escribir como sigue:

```

>
> x=c(29,78,48,29,30,44,72,73,46,82,84,71,75)
> n=length(x) # tamaño del vector
> n
[1] 13
> s=sd(x)# desviación estándar
> s
[1] 21.27837
> M=mean(x) # media
> M
[1] 58.53846
> sesgo=(n/((n-1)*(n-2)))*sum(((x-M)/s)^3)
> sesgo
[1] -0.3307138
>

```

Imagen 63. Salida R para el cálculo del sesgo

3.5.2 **Curtosis**. Mide la mayor o menor cantidad de datos que se agrupan en torno a la moda. Se definen tres tipos de distribuciones según su grado de curtosis: **distribución mesocúrtica**, presenta un grado de concentración medio alrededor de los valores centrales de la variable (el mismo que presenta una distribución normal); **distribución leptocúrtica**, presenta un elevado grado de concentración alrededor de los valores centrales de la variable; **distribución platicúrtica**, presenta un reducido grado de concentración alrededor de los valores centrales de la variable. Una fórmula para calcular la curtosis de un conjunto de datos es la siguiente:

$$curtosis = \left[\frac{n \times (n+1)}{(n-1) \times (n-2) \times (n-3)} \sum_{i=1}^n \left(\frac{(x_i - \bar{x})}{s} \right)^4 \right] - \frac{3 \times (n-1)^2}{(n-2) \times (n-3)}$$

Ecuación 5. Curtosis

Y el algoritmo para la curtosis está dado por:

```

>
> x=c(29,78,48,29,30,44,72,73,46,82,84,71,75)
> n=length(x) # tamaño del vector
> n
[1] 13
> s=sd(x)# desviación estándar
> s
[1] 21.27837
> M=mean(x) # media
> M
[1] 58.53846
> curtosis=((n*(n+1))/((n-1)*(n-2)*(n-3)))*sum(((x-M)/s)^4)-((3*(n-1)^2)/((n-2)*(n-3)))
> curtosis
[1] -1.702062
>

```

Imagen 64. Salida R para el cálculo de la curtosis

Las fórmulas anteriores son un poco engorrosas, pero se muestran con el fin de verificar que, en la mayoría de los casos, fórmulas dispendiosas son posibles de programar en R. El sesgo y la curtosis son posibles de calcular directamente luego de cargar el paquete **moments**, mediante los siguientes comandos:

```
skewness(vector de datos) para calcular el sesgo  
kurtosis(vector de datos) para calcular la curtosis
```

3.6 MEDIDAS DE ASOCIACIÓN

3.6.1 Covarianza. Es una medida de la intensidad de cierta asociación estadística entre dos variables. Como se menciona, es necesario contar con dos variables para calcular la covarianza. Los comandos utilizados para calcularla son `cov(x, y)` o `var(x, y)`.

Ejemplo: En el examen de una asignatura que consta de parte teórica y parte práctica, las calificaciones de nueve alumnos se muestran en la tabla siguiente. Determinar la covarianza entre la prueba teórica y la prueba práctica.

```
Teórica (x) 5 7 6 9 3 1 2 4 6  
Práctica (y) 6 5 8 6 4 2 1 3 7  
  
>  
> x=C(5,7,6,9,3,1,2,4,6)  
> y=C(6,5,8,6,4,2,1,3,7)  
> cov(x,y)  
[1] 4.541667  
>  
>  
> x=C(5,7,6,9,3,1,2,4,6)  
> y=C(6,5,8,6,4,2,1,3,7)  
> var(x,y)  
[1] 4.541667  
>
```

Imagen 65. Salida R para el cálculo de la covarianza

Si en lugar de tener dos vectores se tiene una matriz, este comando calcula la covarianza entre columnas de la matriz.

3.6.2 Correlación. Cuando dos fenómenos sociales, físicos o biológicos crecen o decrecen de forma simultánea y proporcional debido a factores externos, se dice que los fenómenos están positivamente correlacionados. Si uno crece en la misma proporción que el otro decrece, los dos fenómenos están negativamente correlacionados. El grado de correlación se calcula mediante un coeficiente de correlación aplicado a los datos de ambos fenómenos. Una correlación positiva perfecta tiene un coeficiente igual a 1, y para una correlación negativa perfecta es -1. La ausencia de correlación da como coeficiente 0.

Ejemplo: Teniendo en cuenta los datos del ejercicio del examen, la correlación entre estas dos variables es:

```

>
> x=c(5,7,6,9,3,1,2,4,6)
> y=c(6,5,8,6,4,2,1,3,7)
> cor(x,y)
[1] 0.7628535
>

```

Imagen 66. Salida R para el cálculo de la correlación

De la anterior medida se puede observar que se presenta una correlación positiva alta entre estas dos variables.

3.7 EJERCICIOS

3.7.1 En un estudio sobre la preferencia de bebidas calientes en una empresa se preguntó a 20 empleados su bebida favorita; los resultados son los siguientes: café (C), té (T) y aromática (A).

C T C A A T C A C C
A C T C C A A T C C

Determine los porcentajes para cada bebida y construya un diagrama de barras o un diagrama de sectores.

3.7.2 Al finalizar un curso de estadística descriptiva las notas de los 16 estudiantes son las siguientes:

4.1 2.0 3.6 4.5 3.1 3.2 3.6 2.8
3.1 4.0 3.5 3.1 4.7 3.1 4.2 3.9

- Construir la tabla de frecuencias acorde con la situación y su respectivo histograma.
- Calcular las medidas de tendencia central (media, moda y mediana).
- Analice la dispersión de los datos.
- Analice la simetría de la distribución de las notas.

3.7.3 En la empresa CARS se cree que la inversión realizada en atención al cliente está relacionada con el nivel de ventas de cada mes; de darse esta relación las directivas estarían dispuestas a incrementar el porcentaje de inversión en este aspecto. Los datos tomados por un supervisor en los últimos 10 meses se muestran a continuación:

Mes	1	2	3	4	5	6	7	8	9	10
Inversión en dólares	20	50	50	80	90	85	100	105	260	300
Ventas en dólares	1300	1500	1400	1550	1700	1900	1800	1850	1900	1800

- Construya el dispersograma para estas dos variables.
- Calcule la covarianza y la correlación para la inversión y las ventas de la empresa.
- Concluya respecto al gráfico y las medidas calculadas anteriormente.

4. PROBABILIDAD

La probabilidad mide la frecuencia con la que ocurre un resultado en un experimento bajo condiciones suficientemente estables. La teoría de la probabilidad se usa en áreas como la estadística, la matemática, la ciencia y la filosofía para sacar conclusiones sobre la posibilidad de sucesos potenciales y la mecánica subyacente de sistemas complejos. La teoría de la probabilidad constituye la base o fundamento de la estadística, ya que las inferencias que se hagan sobre la población o poblaciones en estudio se moverán dentro de unos márgenes de error controlado, que será medido en términos de probabilidad.

Las probabilidades son muy útiles, ya que pueden servir para desarrollar estrategias. Por ejemplo, algunos conductores parecen mostrar una mayor tendencia a aumentar la velocidad si creen que existe un riesgo pequeño de ser multados; los inversionistas estarán más interesados en invertir dinero si las posibilidades de ganar son buenas. El punto central en todos estos casos es la capacidad de cuantificar cuán probable es determinado evento.

Algunas funciones para calcular probabilidades en R requieren cargar paquetes adicionales a los estándar, como `combinat` y `prob`; por tanto, se recomienda cargar estos paquetes antes de utilizar comandos relacionados con probabilidad.

4.1 CONJUNTOS

La palabra conjunto implica la idea de una colección de objetos que se caracterizan por algo común. En matemática tiene el mismo significado, sólo que a estos objetos se les llama elementos o miembros del conjunto. La noción simple de una colección o conjunto de objetos es fundamental en la estructura básica de las matemáticas y fue Georg Cantor, en el año 1870, quien primero llamó la atención de los matemáticos a este respecto. En R, un conjunto se puede pensar como un vector donde cada uno de los elementos que lo componen son los elementos de un conjunto determinado.

En R se pueden trabajar las diferentes operaciones entre conjuntos. Considerando los siguientes conjuntos:

```
>
> A=c(0,1,2,3,4,5,6,7,8,9)
>
>
> B=c(3,6,9,12,15,18,21,24,27,30)
>
```

Imagen 67. Salida R para la determinación de conjuntos

las operaciones unión, intersección, diferencia y diferencia simétrica se llevan a cabo de la siguiente manera:

4.1.1 **Unión.** La unión de los conjuntos A y B es el conjunto formado por todos los elementos que pertenecen a A o a B o a ambos. Se denota por $A \cup B$ y el comando utilizado para calcularla es `union()`.

```
>
> A=c(0,1,2,3,4,5,6,7,8,9)
> B=c(3,6,9,12,15,18,21,24,27,30)
> union(A,B)
[1] 0 1 2 3 4 5 6 7 8 9 12 15 18 21 24 27 30
>
```

Imagen 68. Salida R para la unión de dos conjuntos

4.1.2 **Intersección.** Se define la intersección de dos conjuntos A y B, como el conjunto de elementos que son comunes a A y B, que se lee A intersección B, y el comando utilizado para calcularla es `intersect()`.

```
>
>
> intersect(A,B)
[1] 3 6 9
>
```

Imagen 69. Salida R para la intersección de dos conjuntos

4.1.3 **Diferencia.** Se denomina diferencia de dos conjuntos A y B al conjunto formado por todos los elementos que pertenecen a A, pero que no pertenecen a B. La diferencia se denota por $A - B$ y se lee A diferencia B o A menos B, y en R el comando utilizado para calcularla es `setdiff()`.

```
>
> setdiff(A,B)
[1] 0 1 2 4 5 7 8
>
> setdiff(B,A)
[1] 12 15 18 21 24 27 30
>
```

Imagen 70. Salida R para la diferencia de dos conjuntos

4.1.4 **Diferencia simétrica.** Se denomina diferencia simétrica de dos conjuntos A y B al conjunto formado por todos los elementos que pertenecen a la unión de A y B, pero que no pertenecen a la intersección de A y B. Se lee A diferencia simétrica B. En R se puede calcular la diferencia simétrica de dos conjuntos al realizar la unión entre las diferencias de los conjuntos, como se mostró en la operación anterior.

```

>
> union(setdiff(A,B),setdiff(B,A))
[1] 0 1 2 4 5 7 8 12 15 18 21 24 27 30
>

```

Imagen 71. Salida R para la diferencia simétrica de dos conjuntos

Dos conjuntos son iguales si los elementos que los conforman son los mismos; el comando `setequal()` permite verificar si dos conjuntos son iguales, utilizando los A y B se tiene,

```

> setequal(A,B)
[1] FALSE
>

```

Imagen 72. Salida R para la igualdad de dos conjuntos

Como el resultado es FALSE, estos dos conjuntos no son iguales.

4.2 ESPACIO MUESTRAL

Un espacio muestral es el conjunto de todos los resultados posibles de un evento o muestra. Es el conjunto de todos los resultados posibles de un experimento estadístico, y se representa con la letra S.

4.2.1 Probabilidad de eventos. La probabilidad de un evento A en un espacio muestral S es la suma de todos los puntos muestrales que conforman el evento A dividido en el tamaño de S; por lo tanto $0 < P(A) < 1$, $P() = 0$, y $P(S) = 1$, es decir, la probabilidad de que un evento ocurra está dada mediante un número que va desde de 0 a 1.

4.2.2 Técnicas de conteo. Son aquellas que son usadas para enumerar eventos difíciles de cuantificar; son principios que se usan para contar resultados que no se conocen o que son muy extensos. Se les denomina técnicas de conteo a: combinaciones y permutaciones, cabe destacar que estas proporcionan la información de todas las maneras posibles en que ocurre un evento determinado.

4.2.3 Factorial. El factorial de un número entero positivo se define como el producto de todos los números naturales anteriores o iguales a él. Se escribe $n!$, y se lee “n factorial” (por definición, el factorial de 0 es 1, así: $0! = 1$). Si se quiere calcular $7!$, su desarrollo y comando utilizado es:

$$7! = 7 \times 6 \times 5 \times 4 \times 3 \times 2 \times 1$$

```

>
> factorial(7)
[1] 5040
>

```

Imagen 73. Salida R para el cálculo de un factorial

4.2.4 Combinaciones. Una combinación es un arreglo de elementos en donde no interesa el lugar o posición que estos ocupan. En una combinación el objetivo está en formar grupos, y el contenido de estos, el número de combinaciones de n objetos tomados de r , se puede calcular mediante el comando `choose(n, r)`.

Ejemplo: Si se cuenta con 5 personas para seleccionar 2, sin importar el orden de selección, el número total de combinaciones posibles entre estas personas es de:

$n C r = \frac{n!}{(n-r)! r!}$	<pre>> > choose(5,2) [1] 10 ></pre>
--------------------------------	--

Ecuación 6 e imagen 74. Salida R para el cálculo de una combinación

Con lo que se tiene que existen 10 posibles combinaciones de 2 personas; ahora bien, si se desea establecer el espacio muestral, es decir, observar las 10 combinaciones, se utilizan los comandos dados en la siguiente ilustración; los nombres de las personas son Bibiana, Carlos, Sandra, Carmen y Reinaldo.

```
>
> # Vector con los nombres de las personas a combinar
>
> Per=c("Bibiana","Carlos","Sandra","Carmen","Reinaldo")
>
> # Comando para crear el espacio muestral asociado
>
> urnsamples(Per, size = 2)
      X1      X2
1 Bibiana Carlos
2 Bibiana  Sandra
3 Bibiana  Carmen
4 Bibiana Reinaldo
5 Carlos   Sandra
6 Carlos   Carmen
7 Carlos Reinaldo
8 Sandra   Carmen
9 Sandra Reinaldo
10 Carmen Reinaldo
>
```

Imagen 75. Salida R para determinar las posibles combinaciones

Obsérvese que el comando que permite construir el espacio muestral asociado con la combinatoria es `urnsamples(x, size)`, donde x es el vector que contiene los objetos por combinar y `size` se refiere a la cantidad de objetos tomados en cada grupo.

4.2.5 Permutaciones. Es todo arreglo de elementos en donde interesa el lugar o posición que ocupa cada uno de ellos. El número de permutaciones de n objetos tomados de r en r objetos se puede calcular mediante el comando `nsamp(n,r,ordered=TRUE)`.

Ejemplo: Si se cuenta con 5 personas y se quieren seleccionar 2, importando el orden de selección, el número total de permutaciones posibles es de:

${}_n P_r = \frac{n!}{(n-r)!}$	<pre> > > nsamp(5,2,ordered=TRUE) [1] 20 > > > n=5 > r=2 > Permutaciones=(factorial(n)/factorial(n-r)) > Permutaciones [1] 20 > . </pre>
--------------------------------	--

Ecuación 7 e imágenes 76 y 77. Salida R para el cálculo de una permutación

Con lo que se tiene que existen 20 permutaciones de 2 personas; ahora bien, si se desea establecer el espacio muestral, el procedimiento es (tomando los mismos nombres de las personas del ejemplo de combinatoria):

```

>
> urnsamples(Per,size=2,ordered=TRUE)

      X1      X2
1  Bibiana  Carlos
2  Carlos  Bibiana
3  Bibiana  Sandra
4  Sandra  Bibiana
5  Bibiana  Carmen
6  Carmen  Bibiana
7  Bibiana Reinaldo
8  Reinaldo Bibiana
9  Carlos   Sandra
10 Sandra  Carlos
11 Carlos  Carmen
12 Carmen  Carlos
13 Carlos  Reinaldo
14 Reinaldo Carlos
15 Sandra  Carmen
16 Carmen  Sandra
17 Sandra  Reinaldo
18 Reinaldo Sandra
19 Carmen  Reinaldo
20 Reinaldo Carmen
>

```

Imagen 78. Salida R para determinar las posibles permutaciones

El comando que permite construir el espacio muestral asociado con la permutación es el mismo utilizado para la combinación, adicionándole el argumento `ordered=TRUE`. Esta permutación también se puede realizar con el comando `permsn(x, r)`, donde `x` es el vector que contiene los objetos por permutar y `r` se refiere a la cantidad de objetos tomados en cada grupo.

Los paquetes `combinat` y `prob` permiten generar experimentos aleatorios, por tanto, si se desea desarrollar los experimentos aleatorios que se mencionan a continuación, se hace necesario que estos paquetes se carguen previamente. A continuación se especifica cómo realizar estos experimentos aleatorios y la forma de determinar su espacio muestral.

4.2.6 `rolldie(times, nsides=6, makespace=TRUE)` genera el espacio muestral para un experimento aleatorio consistente en el lanzamiento de un dado varias veces. Los argumentos utilizados son: `times`, número de lanzamientos deseados; `nsides`, número de caras del dado, y `makespace=`, que si es `FALSE`, solo genera el espacio muestral, y si es `TRUE`, genera una columna adicional con la probabilidad de cada punto muestral.

Ejemplo: Considere el lanzamiento de un dado que tiene 3 caras, numeradas con 1, 2 y 3, respectivamente, en 2 ocasiones.

```
>
> rolldie(2, nsides = 3, makespace = TRUE)
  x1 x2      probs
1  1  1 0.1111111
2  2  1 0.1111111
3  3  1 0.1111111
4  1  2 0.1111111
5  2  2 0.1111111
6  3  2 0.1111111
7  1  3 0.1111111
8  2  3 0.1111111
9  3  3 0.1111111
>
```

Imagen 79. Salida R para el experimento aleatorio "dados"

4.2.7 `tosscoin(times, makespace=TRUE)` genera el espacio muestral para un experimento aleatorio consistente en el lanzamiento de una moneda varias veces. Los argumentos utilizados son: `times`, número de lanzamientos deseados, y `makespace=`, que si es `FALSE`, solo genera el espacio muestral, y si es `TRUE`, genera una columna adicional con la probabilidad de cada punto muestral.

Ejemplo: Considere el lanzamiento de una moneda en dos ocasiones.

```

>
> tosscoin(2, makespace = FALSE)
  toss1 toss2
1      H      H
2      T      H
3      H      T
4      T      T
> tosscoin(2, makespace = TRUE)
  toss1  toss2 probs
1      H      H 0.25
2      T      H 0.25
3      H      T 0.25
4      T      T 0.25
>

```

Imagen 80. Salida R para el experimento aleatorio "monedas"

4.2.8 `urnsamples(x,size,replace=,ordered=)` crea un espacio muestral asociado con un experimento en el cual se muestrean objetos distinguibles de una urna. Los argumentos utilizados al interior de este comando son: `x`, un vector con el nombre de los objetos a ser muestreados; `size` indica el número de objetos de la muestra; `replace`, si es `TRUE` realiza un muestreo con reemplazamiento, si es `FALSE` realiza el muestreo sin reemplazamiento, y `ordered`, si es `TRUE` importa el orden en la selección de las muestras, si es `FALSE` no importa el orden de selección en las muestras. Los tamaños de cada uno de los escenarios anteriores con un vector `x` de tamaño `n` y tamaño de muestra `size` igual a `r`, son como sigue:

Replace	Ordered	Número de muestras
TRUE	TRUE	n^r
FALSE	TRUE	$\frac{n!}{(n-r)!}$
FALSE	FALSE	$\frac{n!}{(n-r)!} r!$
TRUE	FALSE	$\frac{(n-1+r)!}{(n-r)!} r!$

Imagen 81. Tamaños para los posibles escenarios del comando

4.2.9 **Cálculo de una probabilidad y de probabilidades condicionales de eventos.** En R es posible calcular la probabilidad de un evento, dado que se conoce el espacio muestral; además, es posible calcular la probabilidad de un evento, dado que otro evento ya ha ocurrido. El comando que permite calcular lo mencionado anteriormente es `prob(S,evento,given=evento)`, donde `S` hace referencia al espacio muestral, `evento` define el evento al cual se le calcula la probabilidad y `given=` se usa para determinar una probabilidad condicional, es decir, calcular la probabilidad de un evento, dado que ocurrió otro evento.

Ejemplo: Considere el experimento aleatorio (E.A.) lanzamiento de un dado en dos oportunidades; el espacio muestral está determinado mediante una asignación como se indica:

```
>
> s <- rolldie(times = 2, nsides=6, makespace = TRUE)
>
```

Imagen 82. Construcción experimento aleatorio

El tamaño del espacio muestral para este experimento es de 36, a continuación se muestran los 10 primeros puntos muestrales del espacio:

```
> s
      X1 X2      probs
1     1  1  0.02777778
2     2  1  0.02777778
3     3  1  0.02777778
4     4  1  0.02777778
5     5  1  0.02777778
6     6  1  0.02777778
7     1  2  0.02777778
8     2  2  0.02777778
9     3  2  0.02777778
10    4  2  0.02777778
```

Imagen 83. Salida R para los primeros 10 puntos del E.A.

En el espacio muestral construido, el primer lanzamiento es denotado por X1, y el segundo, por X2. Ahora bien, luego de generar el espacio muestral y de asignarle la letra S para el experimento, se procede a realizar el cálculo de algunas probabilidades de interés.

Probabilidad de que la suma de las caras superiores de los dados sea igual a 4.	<pre>> > probs(s, X1+X2 ==4) [1] 0.08333333 ></pre>
Probabilidad de que la suma de las caras superiores de los dados sea menor a 5.	<pre>> > probs(s, X1+X2 < 5) [1] 0.1666667 ></pre>
Probabilidad de que la suma de las caras superiores de los dados sea mayor a 8.	<pre>> > probs(s, X1+X2 > 8) [1] 0.2777778 ></pre>
Probabilidad de que en el primer lanzamiento caiga 5, dado que la suma de las caras superiores de los dados sea mayor a 9.	<pre>> > probs(s, X1 == 5, given = X1+X2 >9) [1] 0.3333333 ></pre>

Imágenes 84, 85, 86 y 87. Salida R cálculo de probabilidades

4.3 EJERCICIOS

4.3.1 Dados los conjuntos A y B, escríbalos en R y luego realice su unión, intersección, diferencia y diferencia simétrica.

$$A = \{1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25\}$$

$$B = \{8, 9, 10, 11, 12, 13, 14, 15, 16\}$$

4.3.2 En una carrera atlética los cuatro primeros en llegar fueron Andrés, Juan, Felipe y Pablo, sin ser este el orden de llegada, responda:

- De cuántas formas diferentes pudieron llegar los atletas, y
- Construya el espacio muestral.

4.3.3 Una empresa selecciona 3 empleados, entre 8 candidatos, para un curso de ascenso; la selección se realiza al azar y no importa el orden. Empleados (a, b, c, d, e, f, g, h)

- ¿Cuántos grupos diferentes se pueden conformar entre los empleados para tomar el curso de ascenso?
- Construya el espacio muestral.

4.3.4 Considere el lanzamiento en 3 ocasiones de un dado que tiene 6 caras, numeradas del 1 al 6.

- Construya el experimento aleatorio en R
- Calcule la probabilidad de que la suma de las caras superiores sea 14
- Calcule la probabilidad de que la suma de las caras superiores sea mayor a 10.

5. DISTRIBUCIONES

R contiene un amplio conjunto de tablas estadísticas; para cada distribución hay comandos que permiten calcular la función de distribución, $F(x) = P(X = x)$; la función de distribución inversa; la función de densidad, y la generación de números pseudoaleatorios de la distribución. El nombre que toma cada distribución en R y los argumentos (parámetros) que cada distribución posee se muestran a continuación.

5.1 DISTRIBUCIONES DISCRETAS

Tabla 5. Distribuciones discretas

Distribución	Fdp	Nombre en R	Argumentos adicionales
Binomial	$\binom{n}{x} p^x q^{n-x}$	binom	size, prob
Geométrica	$q^{x-1} p$	geom	prob
Hipergeométrica	$\frac{\binom{K}{x} \binom{N-K}{n-x}}{\binom{N}{n}}$	hyper	m,n,k
Binomial Negativa	$\binom{x-1}{k-1} q^{x-k} p^k$	nbinom	size, prob
Poisson	$\frac{\lambda^x e^{-\lambda}}{x!}$	pois	lambda

5.2 DISTRIBUCIONES CONTINUAS

Tabla 6. Distribuciones continuas

Distribución	Fdp	Nombre en R	Argumentos adicionales
Beta	$\frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1}$	beta	shape1, shape2, ncp

(sigue...)

Distribución	Fdp	Nombre en R	Argumentos adicionales
Cauchy	$(\pi\lambda)^{-1} \left\{ 1 + \left[\frac{x - \theta}{\lambda} \right]^2 \right\}^{-1}$	cauchy	location, scale
Ji cuadrado	$\frac{1}{\Gamma(v/2)2^{v/2}} x (v/2)^{-1} \exp(-x/2)$	chisq	df,ncp
Exponencial	$\frac{1}{\theta} e^{-x/\theta}$	exp	rate
F	$\frac{\Gamma\left(\frac{v_1+v_2}{2}\right) v_1^{v_1/2} v_2^{v_2/2}}{\Gamma(v_1/2)\Gamma(v_2/2)} f^{(v_1-2)/2} (v_2+v_1 f)^{-(v_1+v_2)/2}$	f	df1,df2,ncp
Gamma	$\frac{1}{\Gamma(\alpha)\theta^\alpha} x^{\alpha-1} \exp(-x/\theta)$	gamma	Shape, scale
Log normal	$\frac{1}{\sqrt{2\pi\sigma x}} \exp\left(-\frac{(\log(x)-\mu)^2}{2\sigma^2}\right)$	lnorm	meanlog, sdlog
Logística	$\frac{1}{s} \exp\left(\frac{x-m}{s}\right) \left(1 + \exp\left(\frac{x-m}{s}\right)\right)^{-2}$	logis	location, scale
Normal	$\frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right)$	norm	mean, sd
t de student	$\frac{\Gamma[(v+1)/2]}{\sqrt{\pi v}\Gamma(v/2)} \left[1 + (t^2/v)\right]^{-(v-1)/2}$	t	df, ncp
Uniforme	$\frac{1}{b-a}$	unif	min, max
Weibull	$\frac{\alpha}{\theta^\alpha} x^{\alpha-1} \exp[-(x/\theta)^\alpha]$	weibull	shape,scale

Con los nombres que recibe cada distribución en R se construyen varias funciones que permiten calcular medidas de interés con respecto a cada una de ellas; con esto se tiene que si al nombre de la distribución se le precede por la letra “d” se obtiene la función de densidad, si se precede por la letra “p” se obtiene la función de distribución, si se precede de la letra “q” se obtiene la función de distribución inversa y si se precede de la letra “r” se genera números pseudoaleatorios, Si se requiere conocer las características de los argumentos de cada función, se consulta la ayuda interactiva; por ejemplo, si se tiene alguna duda sobre los argumentos utilizados para generar números aleatorios de la distribución Normal o calcular probabilidades de esta distribución, en la consola de R se escribe el comando **help(rnorm)**, con lo que se obtiene:

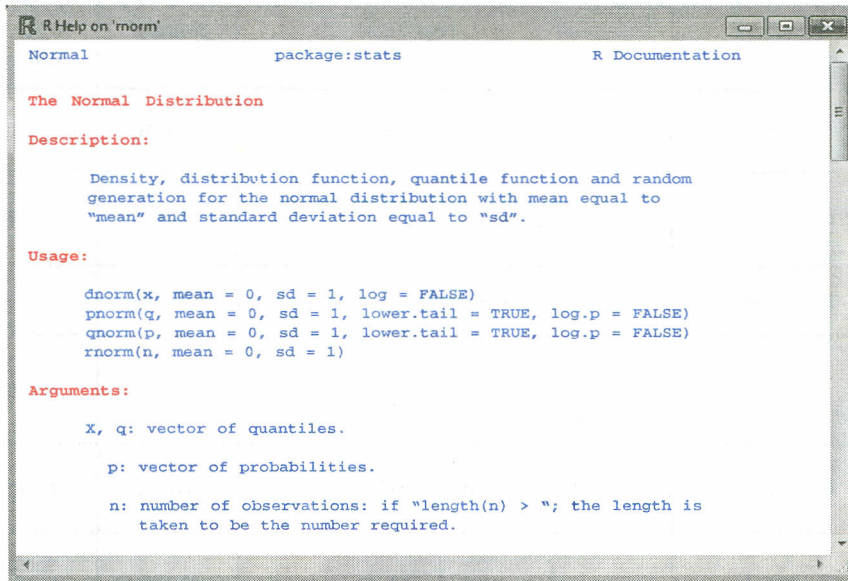


Imagen 88. Ayuda interactiva para la distribución Normal

Algunos ejemplos en los cuales se pueden usar las diferentes funciones son:

Ejemplo: Sea X una variable aleatoria que representa el número de máquinas que una compañía produce en un mes; por experiencia, la compañía sabe que la probabilidad de producir una máquina defectuosa es de $p = 0.16$; considere que el mes pasado la compañía produjo 6 máquinas. Determine cuál es la probabilidad de que de estas 6 máquinas producidas 2 sean defectuosas.

```

>
> dbinom(2,6,0.16)
[1] 0.1911826
>

```

Imagen 89. Salida R distribución binomial

Calcular la probabilidad de que la empresa fabricó a lo sumo 2 máquinas defectuosas.

```

>
> pbinom(2,6,0.16)
[1] 0.9439641
>

```

Imagen 90. Salida R distribución binomial

Para calcular los valores de la función de probabilidad se define un vector con el posible número de máquinas defectuosas (espacio muestral) y se procede como se indica en la siguiente imagen:


```

>
> # Máquinas defectuosas
> x=c(0,1,2,3,4,5,6)
>
> # tamaño de la muestra
> size=6
>
> # Probabilidad de éxito
> prob=0.16
>
> # Probabilidad de éxito
> fdist=dbinom(x,size,prob)
>
> cbind(x,fdist)
      x      fdist
[1,] 0 3.512980e-01
[2,] 1 4.014835e-01
[3,] 2 1.911826e-01
[4,] 3 4.855431e-02
[5,] 4 6.936330e-03
[6,] 5 5.284823e-04
[7,] 6 1.677722e-05
>

```

Imagen 91. Salida R distribución binomial

Para calcular la función de distribución se emplea:

```

>
> # Función de distribución
> # N° máquinas defectuosas
>
> Fdist=pbinom(x,size,prob)
>
> cbind(x,Fdist)
      x      Fdist
[1,] 0 0.3512980
[2,] 1 0.7527815
[3,] 2 0.9439641
[4,] 3 0.9925184
[5,] 4 0.9994547
[6,] 5 0.9999832
[7,] 6 1.0000000
>

```

Imagen 92. Salida R distribución binomial

Ejemplo: Se seleccionan 10 personas para un trabajo, de 20 ingenieros con doctorado; además, dentro del grupo de opcionados hay 5 ingenieros que fueron catalogados como los mejores. ¿Cuál es la probabilidad de que en el grupo de los 10 ingenieros seleccionados se incluyan cuatro de los cinco mejores.

```

>
> # N° a calcular la probabilidad
> x=4
>
> # N° que tienen la característica
> # "Ingeniero bueno"
> m=5
>
> # N° que no tienen la característica
> n=20-5
>
> # Tamaño de la muestra
> k=10
>
> # Probabilidad
> dhyper(x,m,n,k)
[1] 0.1354489
>

```

Imagen 93. Salida R distribución hipergeométrica

Ejemplo: Un gramo de material radioactivo emite partículas alfa a una tasa constante de 3 partículas por segundo; hallar $P(x=0)$, $P(x=1)$, ..., $P(x=7)$ y hallar la función de distribución.

```

>
> # Posibles valores de x
> x=c(0,1,2,3,4,5,6,7)
>
> # lambda
> lambda = 3
>
> # probabilidades
> fden=(dpois(x,lambda))
>
> # Función de distribución
> Fdist=(ppois(x, lambda))
>
> cdind(x,fden,Fdist)
      x      fden      Fdist
[1,] 0 0.04978707 0.04978707
[2,] 1 0.14936121 0.19914827
[3,] 2 0.22404181 0.42319008
[4,] 3 0.22404181 0.64723189
[5,] 4 0.16803136 0.81526324
[6,] 5 0.10081881 0.91608206
[7,] 6 0.05040941 0.96649146
[8,] 7 0.02160403 0.98809550
> .

```

Imagen 94. Salida R distribución Poisson

Ejemplo: Considere el lanzamiento de un dado hasta la primera aparición de un as; X es la variable aleatoria que cuenta el número de fallas hasta obtener un as. Determinar la probabilidad de que se necesiten 3 lanzamientos para obtener un as; determine la función de distribución para $x = (0, 1, 2, 3, 4)$. En el lanzamiento de un dado cada número tiene $1/6$, entonces la probabilidad de que salga un as en un lanzamiento es de 0.16.

```
>
> # Valores de x
> # Número de fallas
> x=c(0,1,2,3,4)
>
> # Probabilidades de un as
> prob=0.16
>
> # probabilidades
> fden=(dgeom(x,prob))
>
> # Función de distribución
> Fdist=(dgeom(x,prob))
>
> cdind(x,fden,Fdist)
      x      fden      Fdist
[1,] 0 0.16000000 0.1600000
[2,] 1 0.13440000 0.2944000
[3,] 2 0.11289600 0.4072960
[4,] 3 0.09483264 0.5021286
[5,] 4 0.07965942 0.5817881
>
```

Imagen 95. Salida R distribución geométrica

Se observa que la probabilidad de que se necesiten tres lanzamientos de un dado para obtener un as, es decir, fallar en los dos primeros lanzamientos, es de 0.1128.

Ejemplo: Considere el lanzamiento de un dado hasta obtener cierta cantidad de ases; X es el número de fallas hasta obtener un determinado número de ases y size es el número de éxitos. Determinar la probabilidad de que se necesiten 5 lanzamientos para obtener 3 ases.

```
>
> # Número de fallas
> x = 2
>
> # Número de éxitos
> size = 3
>
> # probabilidades de un as
> prob = (1/6)
>
> # Probabilidades
> dnbinom(x,size,prob)
[1] 0 0.01929012
>
```

Imagen 96. Salida R distribución binomial negativa

Considere que se necesitaron los siguientes lanzamientos de los dados para obtener los 3 ases especificados (3, 4, 5, 6, 7, 8, 9); construir la función de distribución.

```
>
> # Número de fallas
> x=c(0,1,2,3,4,5,6)
>
> # Número de éxitos
> size = 3
>
> # Probabilidades de un as
> prob = (1/6)
>
> # Función de distribución
> Fdist=pnbinom(x,size,prob)
>
> cbind(x,Fdist)
      x      Fdist
[1,] 0 0.00462963
[2,] 1 0.01620370
[3,] 2 0.03549383
[4,] 3 0.06228567
[5,] 4 0.09577546
[6,] 5 0.13484689
[7,] 6 0.17825959
>
```

Imagen 97. Salida R distribución binomial negativa

Debe tenerse en cuenta que si se realizan tres lanzamientos y se obtienen tres ases, el número de fallas es cero, por tanto, la probabilidad es de 0.00462963; como aparece en la tabla de la imagen anterior, la columna denominada x representa la columna de las fallas.

Ejemplo: Una fábrica de alimentos empaqueta productos cuyos pesos están distribuidos normalmente con media de 450 gramos y desviación estándar de 20 gramos. Determine la probabilidad de que un paquete escogido al azar pese exactamente 470 gramos, $P(X = 470) = ?$

```
>
> # P ( X = 470 )
> dnorm(470, mean=450, sd=20)
[1,] 0.01209854
>
```

Imagen 98. Salida R distribución Normal

Encuentre la probabilidad de que un paquete escogido al azar pese entre 425 y 486 gramos. $P(425 < X < 486) = ?$

$$P(425 < X < 486) = P(X < 486) - P(X < 425)$$

```

>
> # P ( X < 486 )
> P1 = pnorm(486, mean=450, sd=20)
> P1
[1] 0.9649697
>
> # P ( X < 425 )
> P2 = pnorm(425, mean=450, sd=20)
> P2
[1] 0.1056498
>
> # P ( X < 486 ) - P( X < 425)
> P1 - P2
[1] 0.8584199
>

```

Imagen 99. Salida R distribución Normal

Ejemplo: Encontrar la probabilidad de que una muestra aleatoria de 5 observaciones, de una población normal con varianza $\sigma^2 = 1$, tenga una $S^2 \geq 0.265$. La variable aleatoria asociada con este ejemplo es la siguiente:

$$X^2 = \frac{S^2 (n - 1)}{\sigma^2}$$

Ecuación 8. Distribución ji-cuadrado

Con los datos del ejemplo se calcula el valor para esta variable así:

$$X^2 = \frac{S^2 (n - 1)}{\sigma^2} = \frac{0.265 (5 - 1)}{1}$$

Ecuación 9. Distribución ji-cuadrado

Con lo que se tiene que calcular la probabilidad $P(x^2 > 1.06)$ o análogamente $1 - P(x^2 < 1.06)$, mediante la distribución ji cuadrado.

```

>
> # Grados de libertad (n-1)
> df=(5-1)
>
> # P(ji >= 1.06)
> pchisq(1.06,df,lower.tail=FALSE)
[1] 0.9005656
>
> # 1 - P(ji >= 1.06)
> 1 - pchisq(1.06,df)
[1] 0.9005656
>

```

Imagen 100. Salida R distribución ji-cuadrado

Se tiene una probabilidad de 0.90 de que el valor de la varianza muestral sea mayor o igual a 1.06.

Ejemplo: El gerente de una fábrica de cierto tipo de alimentos asegura que el peso promedio del producto que elabora es de 165.285 g. El encargado de control de calidad toma una muestra de 16 paquetes del producto y los pesa. Los resultados fueron los siguientes: 165, 158, 153, 162, 171, 175, 173, 169, 166, 170, 164, 177, 148, 167, 152, 149. Encuentre la probabilidad de $\bar{x} < 163.6875$. La distribución muestral asociada con este ejemplo es la siguiente:

$$t = \frac{X - \mu}{S / \sqrt{n}}$$

Ecuación 10. Distribución t

con $v = n - 1$ grados de libertad, con esto se tiene lo siguiente:

```
>
> # vector con los elementos de la muestra
>
> X=c(165,158,153,162,171,175,173,169,
+     166,170,164,177,148,167,152,149)
>
> # Tamaño muestral
> n = length(X)
> n
[1] 16
>
> # Media muestral
> M = mean(X)
> M
[1] 163.6875
>
> # Desviación estándar muestral
> s = sd(X)
> s
[1] 9.235574
>
> # Valor para la v.a.
> t= (M - 165.285) / (s / sqrt(n))
> t
[1] -0.6918898
```

Imagen 101. Salida R distribución t

Con lo que se tiene que calcular la probabilidad $P(t = -0.69189)$, mediante la distribución t,

```
>  
> # Grados de libertad n-1  
> df = 16 - 1  
>  
> # Probabilidades  
> pt(-0.691, df)  
[1] 0.2500601  
>
```

Imagen 102. Salida R distribución t

La probabilidad de que la media sea menor de 163.6875 es de 0.250.

Ejemplo: En una prueba sobre efectividad de dos tipos distintos de píldoras para dormir, A y B, se utilizarán dos grupos independientes de personas con insomnio. A un grupo de 40 personas se le administrará la píldora A y al otro grupo, de 60, se le administrará la B, y se registrará el número de horas de sueño de cada individuo participante en el estudio. Si se supone que el número de horas de sueño de quienes usan cada tipo de píldora se distribuye normalmente con $\sigma_1^2 = \sigma_2^2$, determinar la probabilidad:

$$P\left(\frac{S_1^2}{S_2^2} < 1.93\right)$$

Ecuación 11. Distribución F

La distribución muestral asociada con el ejemplo anterior es la siguiente:

$$F = \frac{S_1^2 \sigma_1^2}{S_2^2 \sigma_2^2}$$

Ecuación 12. Distribución F

Como para el caso se tiene que $\sigma_1^2 = \sigma_2^2$, entonces, 1.93 es un valor de una distribución F con 39 grados de libertad en el numerador y 59 grados de libertad en el denominador, el cual se puede calcular de la siguiente forma:

```

>
> # tamaño de muestra A
> n1 = 40
> # tamaño de muestra B
> n2 = 60
>
> # Grados de libertad numerador A
> df1 = 40-1
> # Grados de libertad denominador B
> df2 = 60-1
>
> # valor a probar
> x = 1.93
>
> # Probabilidad
> pf( x, df1, df2)
[1] 0.9890417
>

```

Imagen 103. Salida R distribución F

Se tiene una probabilidad del 98.9% de que el cociente entre las varianzas del número de horas de sueño de cada individuo participante en el estudio con las píldoras A y B es menor a 1.93.

CONSIDERACIÓN

En ocasiones se tiene un conjunto de datos del cual se desconoce su distribución, pero se desea inferir con estos datos acerca de una población; así que se hace necesario estudiar la distribución que sigue el conjunto de datos; este estudio se puede realizar de varias formas. Una es mediante un análisis descriptivo, utilizando comandos para calcular medidas de posición, dispersión y de forma `summary()`, `skewness()`, `kurtosis()`, `sd()` y `range()`, entre otras. Otra, mediante el uso de gráficas que permiten observar el comportamiento de los datos `stem()`, `boxplot()`, `hist()`. También es posible graficar la distribución empírica acumulada de los datos con el comando `plot(ecdf(vector de datos))`. Si se desea saber si los datos se ajustan a la distribución Normal es posible realizar un diagrama cuantil-cuantil (Q-Q) con el comando `qqnorm(vector de datos)` o realizar una prueba de hipótesis para comprobar la normalidad; las pruebas serán descritas con mayor profundidad en la sección de inferencia estadística.

El comando `moment(x, order=, central=, absolute=)` permite calcular un vector de datos con los momentos muestrales de un orden específico; los argumentos utilizados son: `x` es el vector que contiene el conjunto de datos, `order=` se refiere al momento que se desea calcular, `central=` se relaciona con un valor lógico (si es TRUE, calcula el momento central; si es FALSE, calcula los momentos respecto al origen) y `absolute=` se relaciona con un valor lógico (si es TRUE, determina el momento aplicándole el valor absoluto a cada uno de los componentes del vector). Para poder trabajar con este comando es necesario cargar previamente el paquete `moments`. Por ejemplo, el momento de orden uno alrededor del origen coincide con la media, a continuación se muestra cómo calcular este primer momento en R.


```

>
> X=c(29,78,48,29,30,44,72,73,46,82,84,71,75,84,45,45)
>
> # Momento de orden uno alrededor del origen
> moment(X,order=1,central=FALSE,absolute=TRUE)
[1] 58.4375
>
> # Media del vector X
> mean(X)
[1] 58.4375

```

Imagen 104. Salida R momentos

Además, con el comando

`all.moments(x,order.max=,central=,Absolute=)` es posible calcular varios momentos a la vez en un conjunto de datos; los argumentos son los mismos utilizados en el comando `moment` y se le adiciona el argumento `orden.max=`, que fija hasta qué momento se desea calcular; teniendo en cuenta el vector X utilizado en el ejemplo anterior, se calculan los momentos con un orden máximo de cuatro.

```

>
> all.moments(X, order.max = 4, central = FALSE, absolute = TRUE)
[1] 1.000000e+00 5.843750e+01 3.817937e+03 2.690324e+05 1.988036e+07
>

```

Imagen 105. Salida R momentos

5.3 EJERCICIOS

5.3.1 Un jugador de tejo revienta mecha dos veces de cada cinco que lanza. Si en una partida dicho jugador lanza 15 veces,

- Calcule la probabilidad de que reviente tres veces mecha.
- Calcule la probabilidad de que reviente por lo menos una vez.
- Calcule la probabilidad de que el jugador reviente más de diez veces durante el juego.

5.3.2 El número de vehículos que llegan a un peaje durante una hora sigue una distribución de Poisson de parámetro $\lambda = 15$.

- Calcule la probabilidad de que sólo llegue un vehículo.
- Calcule la probabilidad de que lleguen más de 5 vehículos.

5.3.3 Sea Z una variable aleatoria normal estándar, es decir, $Z \sim N(0, 1)$; calcule las siguientes probabilidades:

- a. $P(Z \leq 1.64)$
- b. $P(Z \geq 0.5)$
- c. $P(Z \leq -1)$
- d. $P(-1.2 \leq Z \leq 2)$
- e. $P(Z \geq -0.2)$
- f. $P(Z \geq 1.96)$

5.3.4 Un estudio realizado en una empresa para determinar el coeficiente intelectual de los empleados arrojó que dicha medida se distribuye normal con parámetros $\mu = 99$ y $\sigma = 6$.

- a. Calcule la probabilidad de que un empleado seleccionado al azar tenga un coeficiente intelectual por debajo de 90.
- b. Calcule el porcentaje de empleados cuyo coeficiente intelectual está comprendido entre 97 y 108.

5.3.5 Genere una muestra de tamaño 20 de las siguientes poblaciones:

- a. Distribución Binomial con parámetro $p = 0.35$
- b. Distribución Poisson con parámetro $\lambda = 23$
- c. Distribución Normal con parámetros $\mu = 25$ y $\sigma = 3.3$

6. INFERENCIA

Generalmente, cuando en una investigación se analiza una población es casi imposible tomarla en su conjunto, individuo por individuo, ya sea por cuestiones económicas o de accesibilidad, entre otras; por tanto, se hace necesario seleccionar una muestra representativa, de un tamaño manejable, la cual es utilizada para sacar conclusiones de la población de interés. Al realizar el proceso anterior se está utilizando *estadística inferencial*. Dentro de este contexto, la estadística inferencial involucra el uso de un estadístico para obtener una conclusión o inferencia sobre su parámetro correspondiente; por ejemplo: se puede utilizar el estadístico (media muestral) como estimador de (media poblacional).

Las muestras tienen un impacto directo en las decisiones que se toman, por tanto, se hace necesario realizar estas inferencias de manera correcta; en muchas aplicaciones prácticas se trabaja con estadísticos que contemplan dentro de sus supuestos básicos que la distribución muestral de los datos se ajuste a una Normal. Teniendo en cuenta este supuesto, se presentan algunas gráficas y pruebas para verificar el ajuste de un conjunto de datos.

6.1 GRÁFICO CUANTIL-CUANTIL (Q-Q PLOT)

Este tipo de gráfico compara los cuartiles de un conjunto de datos dado contra los cuantiles de la distribución Normal; mediante esta comparación es posible identificar si el conjunto de datos se aproxima a la distribución normal. En R, el comando que permite realizar este tipo de gráfico es `qqnorm(vector)`.

Ejemplo: Verificar que el siguiente conjunto de datos sigue una distribución aproximadamente normal.

```
>
> # vector de datos
> Y=c(9,5,8,10,7,5,6,4,1,1,4,4,5,0,10)
>
> # gráfico Q-Q
> qqnorm(Y)
>
```

Imagen 106. Salida R gráfico Q-Q

A través de estas instrucciones se produce un gráfico de la forma:

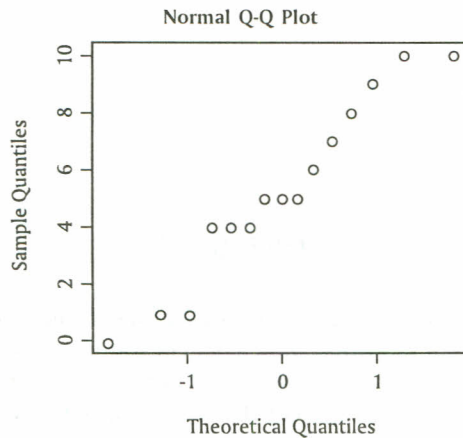
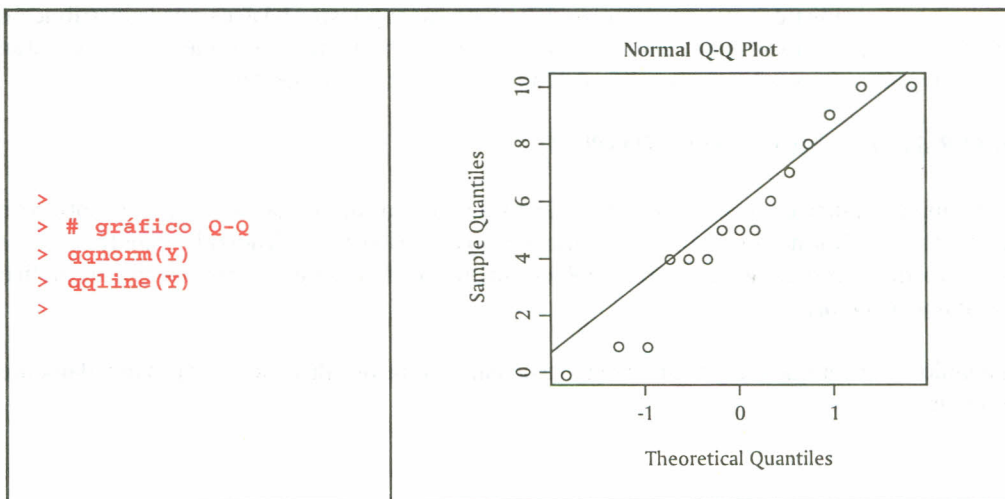


Imagen 107. Salida R gráfico Q-Q

Si se escribe el comando `qqline()`, este añade una recta que pasa por los cuantiles de la distribución y de los datos.



```
>
> # gráfico Q-Q
> qqnorm(Y)
> qqline(Y)
>
```

Imágenes 108 y 109. Salida R gráfico Q-Q

Se espera que, si el conjunto de datos pertenece o se aproxima a la distribución normal, los puntos estén sobre la línea. Como se puede apreciar, los datos del ejemplo pertenecen a una distribución normal.

Además, el comando `qqplot(vector1, vector2)` representa los cuantiles del vector1 sobre los cuantiles del vector2 para comparar sus distribuciones respectivas.

6.2 PRUEBA EXACTA DE NORMALIDAD

Una prueba exacta para probar la normalidad de un conjunto de datos es la **Shapiro-wilks**; en R es posible realizarla mediante el comando `shapiro.test(vector)`. La prueba arroja el estadístico W (Shapiro – Wilk) y el p valor, es decir, el valor más bajo de significancia al cual se puede rechazar la hipótesis nula. La hipótesis nula de la prueba es:

H₀: Los datos siguen una distribución normal

Ejemplo: Los valores sobre las longitudes en micras de 50 filamentos de la producción de una máquina son los siguientes:

102	98	93	100	98	105	115	110	99	120
115	130	100	86	95	103	105	92	99	134
116	118	89	102	128	99	119	128	110	130
112	114	106	114	100	116	108	113	106	105
120	106	110	100	106	117	109	108	105	106

Imagen 110. Datos de 50 filamentos

Determinar si los datos de la muestra provienen de una distribución aproximadamente normal. Para lo cual se procede de la siguiente manera:

```
>
> # Vector de datos
> X=c(102,115,116,112,120,98,130,118,114,106
+      93,100,89,106,110,100,86,102,114,100,
+      98,95,128,100,106,105,103,99,116,117,
+      115,105,119,108,109,110,92,128,113,108,
+      99,99,110,106,105,120,134,130,105,106)
>
> # prueba Shapiro - Wilks
> shapiro.test(X)
```

```
Shapiro-Wilk normality test
```

```
data: X
W = 0.9759, p-value = 0.3954
```

Imagen 111. Salida R Prueba Shapiro-Wilks

6.3 PRUEBA KOLGOMOROV-SMIRNOV

Si se sospecha que la distribución de un conjunto de datos pertenece a una distribución diferente a la distribución normal y se desea comprobar a qué distribución se ajusta, se usa la prueba de **Kolmogorov-Smirnov**, que permite contrastar un conjunto de datos contra cualquier distribución; el comando por utilizar es:

`ks.test(vector, "distribución", parámetros, alternative=c("two.sided", "less", "greater"))`; los argumentos utilizados son: `vector`, se refiere al vector con el conjunto de datos; `"distribución"`, hace referencia a la función de distribución con la cual se desee contrastar (esta distribución debe estar escrita como se vio en la sección de distribuciones); `parámetros`, son los parámetros de cada distribución, y `alternative`, hace referencia al tipo de prueba que se desee calcular, es decir, prueba a dos colas `"two.sided"`: prueba a cola izquierda, `"less"`, y prueba a cola derecha, `"greater"`.

Ejemplo: Se cree que las fallas de la máquina empacadora en la empresa Empacar Ltda. sigue una distribución de Poisson; comprobar este supuesto con los datos de los últimos 15 días de producción de la empresa; por experiencia del año anterior, se tiene que en promedio una máquina de la empresa presenta 5 fallas diariamente.

```
>
> # Fallas
> X=c(3,3,2,4,1,6,7,5,6,8,5,6,8,7,5)
>
> # Prueba Kolgomorov - smirnov
> ks.test(X, "ppois", lambda=5, alternative="two.sided")
```

```
One-sample Kolmogorov-Smirnov test

data: X
D = 0.2826, p-value = 0.1820
alternative hypothesis: two-sided
```

Imagen 112. Salida R Prueba Kolmogorov-Smirnov

El contraste Kolgomorov-Smirnov también se puede utilizar para comparar dos conjuntos de datos a fin de verificar si pertenecen aproximadamente a una misma distribución.

6.4 HIPÓTESIS ESTADÍSTICA

Una hipótesis estadística es un enunciado acerca de la distribución de probabilidad de una variable aleatoria y de sus parámetros. Las hipótesis estadísticas involucran a menudo una o más características de la distribución, como, por ejemplo, forma o independencia de la variable aleatoria. A continuación se muestran los pasos por seguir para realizar una hipótesis estadística:

1. Expresar la hipótesis nula
2. Expresar la hipótesis alternativa
3. Especificar el nivel de significancia
4. Determinar el tamaño de la muestra
5. Establecer los valores críticos que determinan las regiones de rechazo y las de no rechazo.
6. Determinar la prueba estadística.
7. Recolectar los datos y calcular el valor del estadístico de la muestra para la prueba apropiada.

8. Determinar si la prueba estadística ha sido en la zona de rechazo o en una de no rechazo.
9. Determinar la decisión estadística.
10. Expresar la decisión estadística en términos del problema.

6.4.1 Pruebas de hipótesis para la media. Cuando se van a realizar pruebas de hipótesis relativas a la media poblacional se debe saber si la varianza poblacional σ^2 es conocida o desconocida, ya que la distribución subyacente al estadístico de prueba será la normal estándar, si la varianza es conocida, y t-student, si la varianza es desconocida.

6.4.1.1 Prueba de hipótesis para la media, varianza conocida. Cuando la varianza σ^2 es conocida, las pruebas de hipótesis se basan en el estadístico de prueba:

$$Z = \frac{\bar{X} - \mu}{\sigma / \sqrt{n}}$$

Ecuación 13. Estadístico de prueba

Existen tres tipos de pruebas que se pueden plantear con respecto a la media:

1. Prueba cola a derecha $H_0 : \mu = \mu_0$
 $H_1 : \mu > \mu_0$
2. Prueba cola a izquierda $H_0 : \mu = \mu_0$
 $H_1 : \mu < \mu_0$
3. Prueba a dos colas $H_0 : \mu = \mu_0$
 $H_1 : \mu \neq \mu_0$

Para este caso, el valor o los valores críticos se obtienen de la distribución normal estándar, es decir, de la distribución normal con media igual a cero y varianza uno. Mediante el siguiente ejemplo se muestra cómo se puede realizar esta prueba paso a paso.

Ejemplo: Suponga una variable aleatoria P que se designa para el peso de un pasajero de avión; el interés está en conocer el peso promedio de todos los pasajeros. Como hay limitaciones de tiempo y dinero para pesarlos a todos, se toma una muestra de 36 pasajeros, de la cual se obtiene una media muestral $\bar{x} = 160$ libras. Suponga además que la distribución de los pasajeros es aproximadamente normal con desviación estándar $\sigma = 30$ libras, con un nivel de significancia de 0.05. ¿Se puede concluir que el peso promedio de todos los pasajeros es menor que 170 libras?

Las hipótesis planteadas para este ejercicio son:

$$H_0 : \mu \geq 170 \quad \text{vs} \quad H_1 : \mu < 170$$

```

>
> # media muestral
> X = 160
>
> # Mu
> M = 170
>
> # desviación estándar
> d = 30
>
> # Tamaño de la muestra
> n = 36
>
>
> # Zona de rechazo
> # Una Z con (alfa = 0.05)
> Zcritico = qnorm(0.05,mean=0,sd=1)
> Zcritico
[1] -1.644854
>
> # Estadístico de prueba
> Zcalculado = (X - M) / (d / sqrt(n))
> Zcalculado
[1] -2
>
> # Decisión rechazar Hipótesis mula si
> Zcalculado < Zcritico
[1] TRUE

```

Imágenes 113 y 114. Salida R Prueba de hipótesis para la media

Los puntos críticos se determinan de acuerdo con la hipótesis alternativa H_1 ; es así que para la hipótesis $H_1: \mu > \mu_0$ el punto crítico es una $Z_{(1-\alpha)}$, para la hipótesis $H_1: \mu < \mu_0$ el punto crítico es una $Z_{(\alpha)}$ y para la hipótesis $H_1: \mu \neq \mu_0$ se tiene que los puntos críticos son $Z_{(1-\alpha/2)}$ y $Z_{(\alpha/2)}$.

Esta misma prueba es posible realizarla en R mediante un comando directo; para poder realizarla es necesario cargar los paquetes PASWR, e1071, class y MASS; el comando utilizado para esto es:

```
z.test(x, y = NULL, alternative = "two.sided", mu = 0, sigma.x = NULL,
sigma.y = NULL, conf.level = 0.95)
```

Los argumentos utilizados son: **x**, vector numérico; **y**, vector numérico en caso de realizar una prueba de diferencias de medias; **alternative**, se selecciona el tipo de prueba deseado; **mu**, especifica el valor de la media o la diferencia de medias; **sigma.x** y **sigma.y**, número que representa la desviación estándar de **x** o **y**; **conf.level**, es el nivel de significancia al cual se realiza la prueba.

Ejemplo: Los siguientes conjuntos de datos representan la producción en toneladas de dos fincas; probar si los promedios de producción no difieren en más de dos toneladas; las producciones en toneladas se distribuyen de manera normal con desviaciones estándar 0.5.


```

>
> # Datos de producción en toneladas
> x = c(7.8, 6.6, 6.5, 7.4, 7.3, 7, 6.4, 7.1, 6.7, 7.6, 6.8)
> y = c(4.5, 5.4, 6.1, 6.1, 5.4, 5, 4.1, 5.5)
>
> # Prueba para diferencias de medias
> z.test(x, sigma.x=0.5, y, sigma.y=0.5, mu=2)

Two-sample z-Test

data: x and y
z = -1.0516, p-value = 0.293
alternative hypothesis: true difference in means is not equal to 2
95 percent confidence interval:
 1.300323 2.211040
sample estimates:
mean of x mean of y
 7.018182 5.262500

```

Imagen 115. Salida R prueba Z diferencias de medias

6.4.1.2 Prueba de hipótesis para la media con varianza desconocida y tamaño de muestra pequeño.

Cuando la varianza σ^2 no es conocida, las pruebas de hipótesis se basan en el siguiente estadístico de prueba que sigue una distribución t - student con n - 1 grados de libertad.

$$t = \frac{\bar{X} - \mu}{S / \sqrt{n}}$$

Ecuación 14. Estadístico de prueba

El comando utilizado en R para realizar esta prueba es `t.test(x,y=NULL,alternative="", mu=,paired=,var.equal=,conf.level=)`, donde x se refiere al vector de datos, `alternative` selecciona el tipo de prueba deseada, `mu` es el valor de la media y `conf.level` es el nivel de significancia al cual se realiza la prueba. Los siguientes parámetros son utilizados dentro de la prueba si se desea comparar los promedios de dos muestras independientes: `mu` especifica la diferencia entre los promedios; si las muestras son pareadas, entonces `paired=TRUE`; si se asumen varianzas iguales en las muestras, entonces `var.equal=TRUE`.

Ejemplo: En una muestra de 15 bolsas de arroz de un kilo de la molinera “El Bello Arroz” se encontró la siguiente información sobre el peso de ellas:

987, 997, 1006, 965, 1009, 968, 1007, 999, 1006, 1099, 1000, 997, 985, 1002, 1069

Si un cliente demanda a la compañía por no presentar el peso de la referencia, realice una prueba de hipótesis y resuelva el conflicto con una confiabilidad del 95%. La hipótesis alternativa para este caso sería $H_1 : \mu \neq 1000$ gramos, con lo que se tiene el siguiente desarrollo:

```

>
> # Pesajes
> x=c(987,997,1006,965,1009,968,1007,
+ 999,1006,1099,1000,997,985,1002,1069)
>
> # prueba de Hipótesis
> t.test(X,alternative="two.sided",
+       mu=1000,conf.level=0.95)

One Sample t-test

data: X
t = 0.7152, df = 14, p-value = 0.4862
alternative hypothesis: true mean is not equal to 1000
95 percent confidence interval:
 987.2075 1025.5925
sample estimates:
mean of x
 1006.4

```

Imagen 116. Salida R prueba t

6.4.2 Prueba de hipótesis para la homogeneidad de varianzas. Si se requiere contrastar la igualdad de varianzas se puede utilizar el comando

```
var.test(x,y,ratio=1,alternative=c("two.sided","less","greater"), conf.level=),
```

que desarrolla una prueba F para comparar las varianzas de dos muestras provenientes de distribuciones normales. El parámetro ratio se refiere al cociente hipotético de las varianzas de las poblaciones.

```

>
> # Muestras de la distribución normal
> x = rnorm(50, mean = 0, sd = 2)
> y = rnorm(30, mean = 1, sd = 1)
>
> # Prueba F
> var.test(x, y, ratio=1)

F test to compare two variances

data: x and y
F = 3.1209, num df = 49, denom df = 2, p-value = 0.001583
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 1.568018 5.871735
sample estimates:
ratio of variances
 3.120912

```

Imagen 117. Salida R prueba F

6.4.3 Prueba de hipótesis para la Correlación/Asociación entre muestras pareadas. Si se quiere verificar el grado de correlación/asociación entre muestras se puede utilizar la función:

```
cor.test(x, y, alternative = c("two.sided", "less", "greater"),
        method = c("pearson", "kendall", "spearman"),
        exact = NULL, conf.level = 0.95)
```

Donde **method** se refiere al tipo de método por el cual se calcula el coeficiente de correlación utilizado en la prueba; las opciones son "pearson" (coeficiente de correlación de Pearson), "kendall" (tao de kendall) y "spearman" (rho de Sperman).

En R también es posible realizar pruebas de hipótesis para el Sesgo y Curtosis de un conjunto de datos; para esto se hace necesario que previamente a la prueba se cargue el paquete **moments**.

6.4.4 Prueba de hipótesis para el Sesgo. Bajo supuesto de normalidad, un conjunto de datos debe tener una distribución simétrica, para lo cual el sesgo debe ser igual a cero; esta última afirmación es la hipótesis nula para la prueba que realiza R. La instrucción que permite realizar esta prueba es la siguiente:

```
agostino.test(x, alternative=c("two.sided", "less", "greater"))
```

Ejemplo: Generar 1000 datos de la distribución Normal estándar y verificar si el sesgo es igual o aproximadamente igual a cero (hipótesis nula).

```
>
> # Aleatorios de Normal estándar
> x = rnorm(1000)
>
> skewness(x)
[1] 0.07151397
>
> agostino.test(x)

      D'agostino skewness test

data:  x
skew = 0.0715, z = 0.6121, p-value = 0.5404
alternative hypothesis: data have a skewness
```

Imagen 118. Salida R prueba para el sesgo

En la prueba anterior, cuando no se especifica el tipo de prueba que se desee, R por defecto realiza una prueba a dos colas. Los resultados muestran la no existencia de evidencia estadística para rechazar la hipótesis nula.

6.4.5 Prueba de hipótesis para la Curtosis. Bajo supuestos de normalidad, un conjunto de datos debe tener curtosis igual a 3; por tanto, la hipótesis nula para la prueba es que la curtosis es igual a 3. La instrucción que permite realizar esta prueba es la siguiente:

```
anscombe.test(x, alternative=c("two.sided", "less", "greater"))
```

```

>
> # Aleatorios de Normal estándar
> x = rnorm(1000)
>
> kurtosis(x)
[1] 3.171290
>
> anscombe.test(x,alternative="two.sided")

      Anscombe-Glynn kurtosis test

data:  x
kurt = 3.1713, z = 1.1467, p-value = 0.2515
alternative hypothesis: kurtosis is not equal to 3

```

Imagen 119. Salida R prueba para la curtosis

Ejemplo: Generar 1000 datos de la distribución Normal estándar y verificar si la curtosis es igual o aproximadamente igual a tres (hipótesis nula).

Como se ha indicado, una de las condiciones de aplicación de los contrastes anteriores es la normalidad. Si esta falta, es posible utilizar otro tipo de pruebas para contrastar parámetros de una o de dos muestras; entre estos contraste se cuenta con el contraste de Wilcoxon (o de Mann-Whitney), que solo presupone que la distribución común es continua.

6.4.6 Prueba de Wilcoxon. Si la prueba se realiza con solo una muestra o se tienen dos conjuntos de datos y las muestras de estos conjuntos son pareadas, la prueba de rangos Wilcoxon tiene como hipótesis nula que la distribución de la muestra (en el caso de una muestra) o de las dos muestras (dos muestras pareadas) es simétrica sobre su media (μ). Por otra parte, si las muestras no son pareadas, la prueba tiene como hipótesis nula que las distribuciones de las muestras difieren por la localización de la media. A continuación se presenta el comando y los argumentos utilizados para realizar esta prueba.

```

wilcox.test(x, y=NULL, alternative=c("two.sided", "less", "greater"),
mu = 0, paired=, exact = NULL, correct = TRUE, conf.int = FALSE, conf.level= )

```

x , y se refieren a los vectores de datos si $y=NULL$, la prueba es para una sola muestra; μ se refiere a un número específico sobre el parámetro usado en la prueba de hipótesis; **exact** es indicación lógica que determina si el p valor es calculado; **correct** es indicación lógica si se aplica una corrección continua en la distribución normal usada para calcular el p valor; **conf.int**, si es igual TRUE, calcula un intervalo de confianza para el parámetro de localización; **conf.level** determina el nivel de confianza para el intervalo.

Existen más pruebas de hipótesis que se desarrollan y que no tienen una instrucción específica en R; pero como se vio en la primera prueba de hipótesis, estas se pueden escribir con facilidad paso a paso, ya que, como se ha mencionado, R es un lenguaje sencillo.

6.5 EJERCICIOS

6.5.1 Los siguientes datos hacen parte de una investigación sobre el crecimiento y desarrollo de niños de cierta comunidad indígena.

21 45 35 28 42 30 31 32 36 48 41
22 26 37 40 29 28 27 28 32 33 35

Verificar que el conjunto de datos sigue una distribución aproximadamente normal, mediante pruebas graficas (Q–Q PLOT) y prueba exacta de normalidad (Shapiro).

6.5.2 Se registraron los siguientes datos, en segundos, sobre lo que tardan algunos hombres y mujeres en resolver ciertos ejercicios de matemáticas en la universidad, los cuales fueron seleccionados aleatoriamente.

Hombres	Mujeres
$n_1 = 12$	$n_2 = 15$
$\bar{x}_1 = 19$	$\bar{x}_2 = 22$
$S_1^2 = 1.8$	$S_2^2 = 2$

Suponga que los tiempos para los dos grupos se distribuyen normalmente y que las varianzas son iguales, aunque desconocidas. Pruebe la hipótesis referente a que no existe diferencia entre el tiempo promedio de solución de los ejercicios entre hombres y mujeres, a un nivel de confianza del 95%.

7. MUESTREO

El muestreo estadístico es un procedimiento empleado para obtener una o más muestras de una población, a fin de realizar estimaciones a algunos de sus parámetros. Una vez establecido un marco muestral representativo de la población, se realiza el muestreo, es decir, se procede a la selección de los elementos de la muestra. Existen diferentes técnicas para obtener una muestra.

Para realizar algunos procesos relacionados con muestreo en R es necesario cargar los paquetes PASWR, e1071, class, MASS. Las siguientes rutinas se realizan luego de haber cargado estos paquetes.

7.1 TAMAÑO DE MUESTRA

En estadística el tamaño de la muestra es el número de sujetos extraídos de una población que la componen, necesarios para que los datos obtenidos sean representativos de la población. Bajo ciertas condiciones, en R es posible determinar el tamaño muestral; el comando que se presenta a continuación sirve para determinarlo:

```
nsize(b, sigma = NULL, p = 0.5, conf.level= , type="pi")
```

donde **b** es el error máximo permisible; **sigma** se refiere a la desviación estándar poblacional (de ser conocida); **p** es la estimación de la proporción poblacional, que no se requiere en caso de que se use el tipo **mu**; **conf.level** se refiere al nivel de confianza deseado, y **type** indica el parámetro (media "mu" o proporción "pi") bajo el cual se estima el tamaño de la muestra.

```
> # Error máximo permitido 5%
> b = 0.05
>
> # Confiabilidad del 94%
> conf.level = 0.94
>
> # Estimación de la proporción
> p = 0.048
>
> # Tamaño de muestra
> nsize(b, p, conf.level,type="pi")
```

```
The required sample size (n) to estimate the population
proportion of successes with a 0.95 confidence interval
so that the margin of error is no more than 0.05 is 87 .
```

Imagen 120. Salida R Tamaño de muestra

7.2 MUESTREO ALEATORIO SIMPLE

Uno de los objetivos del muestreo estadístico es precisamente tomar muestras aleatorias de una población; el siguiente comando es útil para seleccionar muestras mediante la técnica denominada Muestreo Aleatorio Simple (MAS), una vez definido el tamaño de muestra:

```
sample(x, size, replace = FALSE, prob = NULL)
```

donde x es el vector del cual se seleccionan los elementos, que puede ser de tipo numérico o puede contener los nombres de los caracteres, en este último caso cada uno de los elementos debe ir entre comillas; $size$ es un entero positivo que indica el número de elementos por seleccionar del vector x ; $replace$ es la indicación lógica para realizar un muestreo con reemplazamiento (TRUE) o sin reemplazamiento (FALSE), y $prob$ es un vector con las probabilidades correspondientes a cada uno de los elementos del vector x .

Ejemplo: Se tienen 100 empresas del sector industrial; de estas se necesita tomar una muestra aleatoria de tamaño 17, con el fin de realizar un estudio de factibilidad por parte de una proveedora de insumos; tener en cuenta que una empresa no puede aparecer en más de una oportunidad y que todas las empresas tienen la misma probabilidad de salir en la muestra; considere que las 100 empresas están numeradas de 1 a 100.

```
>
> # Numeración de las empresas
> x = c ( 1:100)
>
> sample (x, size=17, replace=FALSE, prob=NULL )
[1,] 34  2 65 37 49  4 56 16 64 99 80 33 85 19 28 72 74
>
```

Imagen 121. Salida R MAS

Ejemplo: Se desea seleccionar, de los 6 ingenieros que tiene la empresa, 2 para cargos directivos; se cree que los ingenieros Pedro y Luis tienen una probabilidad de ser seleccionados de 0.3, mientras que los demás tienen una probabilidad de 0.1; determinar aleatoriamente cuáles ingenieros son seleccionados.

```
>
> # Ingenieros
> x=c("Pedro","Luis","Edwin","Juan","Diego","Ronal")
>
> # Selección de ingenieros
> sample (x, size=2, prob=c(0.3,0.3,0.1,0.1,0.1,0.1 ) )
[1,] "Luis"    "Pedro"
>
```

Imagen 122. Salida R MAS

En el ejercicio anterior, si se quisiera observar todas las posibles muestras, se utilizaría el comando $SRS(x, n)$, donde n es el tamaño de la muestra; esto se muestra a continuación:

```

>
> # Ingenieros
> x=c("Pedro", "Luis", "Edwin", "Diego", "Ronal")
>
> # Posibles muestras
> SRS(x, 2)
      [,1] [,2]
[1,] "Pedro" "Luis"
[2,] "Pedro" "Edwin"
[3,] "Luis" "Edwin"
[4,] "Pedro" "Diego"
[5,] "Luis" "Diego"
[6,] "Edwin" "Diego"
[7,] "Pedro" "Ronal"
[8,] "Luis" "Ronal"
[9,] "Edwin" "Ronal"
[10,] "Diego" "Ronal"
>

```

Imagen 123. Salida R Posibles muestras

Para utilizar algunas otras técnicas de muestreo mediante comandos específicos es necesario cargar los paquetes **sampling**, **Ipsolve** y **MASS**.

7.3 MUESTREO SISTEMÁTICO

Se utiliza cuando el universo o población es de gran tamaño o se extiende en el tiempo. Primero hay que identificar las unidades y relacionarlas con el calendario (cuando proceda) o en lista, numeradas en orden. El comando que permite extraer una muestra mediante este método es **UPsystematic(pik, eps=1e-6)**, donde **pik** es el vector de probabilidades de inclusión; **pik = c(rep(n/N, N))**, donde **n** es el tamaño de la muestra, **N** es el tamaño de la población, y **eps** es un valor de control que por defecto es $1e-6$.

Ejemplo: En el archivo de la biblioteca se encuentran 150 tesis de grado numeradas en orden, de las cuales se desea revisar si cuentan con artículo; para ello se extrae una muestra de tamaño 15 mediante el método sistemático. Determinar cuáles son las tesis seleccionadas.

```

>
> # Tamaño de la población
> N = 150
>
> # Tamaño de la muestra
> n = 15
>
> # Probabilidad de inclusión
> pik = c(rep(n/N, N))
>
> # Selección de la muestra
> s=UPsystematic(pik)
>
> # Elementos seleccionados
> (1:length(pik))[s==1]
[1] 8 18 28 38 48 58 68 78 88 98 108 118 128 138 148
>

```

Imagen 124. Salida R Muestreo sistemático

El comando `UPsystematic()` genera un vector con 1 y 0; 1 significa que el objeto es seleccionado, y 0, que no fue seleccionado; el comando `(1:length(pik))[s==1]` permite ver los elementos seleccionados dentro del vector que fue creado (los unos del vector).

7.4 MUESTREO ESTRATIFICADO

Consiste en la división previa de la población de estudio en grupos o clases que se suponen homogéneos respecto a la característica que se va a estudiar. Para poder realizar un muestreo estratificado en R es necesario que los estratos sean del mismo tamaño. El comando que permite tomar muestras mediante esta técnica es: `balancedstratification(X, strata, pik)`, donde `X` es el vector con los elementos que conforman la población, `X = cbind(c(elementos))`; `strata` es el vector que especifica los estratos, y `pik` es el vector de probabilidades de inclusión; `pik = rep(n/m, N)`, donde `n` es el tamaño de la muestra por estrato, `m` es el tamaño del estrato y `N` es el tamaño de la población.

Ejemplo: Se tienen 15 elementos clasificados en 3 estratos diferentes; determinar una muestra aleatoria de tamaño 6.

```
>
> # Vector de estratificación
> strata=c(1,1,1,1,1,2,2,2,2,2,3,3,3,3)
>
> # Elementos de la población numerados en orden
> # y ordenados de acuerdo al estrato.
> X=cbind(c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15))
>
> # Probabilidades de inclusión
> pik=rep(2/5, 15)
>
> # Selección de la muestra
> s=balancedstratification(X,strata,pik)
```

Imagen 125. Salida R Muestreo Estratificado

Luego de este último comando, en la consola aparece una descripción de la selección de los elementos de la muestra en cada estrato y en conglomerado; para visualizar los elementos seleccionados se escribe el comando `(1:length(pik))[s==1]` en la consola de R.

```
QUALITY OF BALANCING
TOTALS HorvitzThompson_estimators Relative_deviation
1      2                          2      0.000000e+00
2      2                          2      2.220446e+14
3      2                          2      -2.220446e+14
4     120                         115     -4.166667e+00
>
> # Elementos de la muestra
> (1:length(pik))[s==1]
[1] 3 4 6 9 11 13
>
```

Imagen 126. Salida R Muestreo Estratificado

Así se tiene que los elementos 3 y 4 pertenecen al estrato 1; los elementos 6 y 9, al estrato 2, y los elementos 11 y 13, al estrato 3. Es de notar que este comando sólo se puede utilizar cuando el tamaño de los estratos es igual.

7.5 EJERCICIOS

7.5.1 Determine el tamaño de muestra bajo las siguientes condiciones: error máximo permitido (4%), confiabilidad (93%), con estimación de la media (25) y desviación estándar (1.2).

7.5.2 La Dirección de la universidad está interesada en conocer la opinión de los estudiantes frente al cambio de profesores, para lo cual decide tomar una muestra aleatoria de tamaño 35; considere que los 223 estudiantes tienen asignados códigos entre 1 y 223. Determine cuáles códigos (estudiantes) aparecen en la muestra.

7.5.3 En la empresa JB quieren seleccionar 2 administradores, de los cinco que hay, para asumir cargos fuera del país; se cree que por la experiencia los tres primeros administradores tienen una probabilidad de ser seleccionados de 0.25; el cuarto, una de 0.15, y el quinto, una de 0.1. Asigne un nombre a cada uno de los administradores y determine aleatoriamente cuáles son seleccionados.

7.5.4 En la alcaldía se encuentran las 240 hojas de vida de los trabajadores, numeradas en orden; se quiere revisar si estas cuentan con toda la documentación requerida, para lo cual se extrae una muestra de tamaño 48 mediante el método sistemático. Determinar cuáles son las hojas de vida seleccionadas.

8. ANÁLISIS MULTIVARIADO

El análisis de datos multivariantes comprende el estudio estadístico de varias variables medidas en elementos de una población, con los siguientes objetivos: 1) Resumir los datos mediante un pequeño conjunto de nuevas variables, construidas como transformaciones de las originales, con la mínima pérdida de información; 2) Encontrar grupos en los datos, si existen; 3) Clasificar nuevas observaciones en grupos definidos, y 4) Relacionar dos conjuntos de variables (Peña, 2002).

8.1 MATRIZ DE DATOS

Si se tiene p variables numéricas en un conjunto de n elementos, cada una de estas p se denomina variable univariante, y el conjunto de las p variables se denomina una variable multivariante. Para construir esta matriz en R existen varias formas; una es mediante el comando `cbin()`, visto en la sección correspondiente a matrices.

Una matriz de datos también se puede construir mediante las hojas de datos (**data frames**), que son estructuras similares a una matriz, en donde cada columna puede ser de un tipo distinto a las otras. Las hojas de datos son apropiadas para describir “matrices de datos” donde cada fila representa a un individuo, y cada columna, una variable, variables que pueden ser numéricas o categóricas. Considere las siguientes variables: género, peso, edad y estatura, medidas en 5 personas, a partir de las cuales se construye la siguiente matriz de datos.

```
>
> # Variables
> Género=c("Hombre", "Mujer", "Hombre", "Mujer", "Mujer")
> Estatura=c(170,160,162,168,160)
> Peso=c(70,50,65,60,62)
> Edad=c(27,26,32,40,21)
>
> # matriz de datos
> D=data.frame(Género,Estatura,Peso,Edad)
> D
  Género Estatura Peso Edad
1 Hombre      170   70   27
2 Mujer      160   50   26
3 Hombre      162   65   32
4 Mujer      168   60   40
5 Mujer      160   62   21
>
```

Imagen 127. Salida R Construcción de un data frame

Para describir los datos multivariantes primero se debe realizar una descripción de cada una de las variables por separado y luego las relaciones que se presentan entre ellas. Es así como para calcular algunas estadísticas sobre cada variable se usa el comando `summary()`.

```
>
> # Estadísticas descriptivas por variable
> summary(D)
  Género      Estatura      Peso      Edad
Hombre:2  Min.   :160  Min.   :50.0  Min.   :21.0
Mujer :3  1st Qu.:160  1st Qu.:60.0  1st Qu.:26.0
         Median :162  Mean   :62.0  Median :27.0
         Mean   :164  Mean   :61.4  Mean   :29.2
         3rd Qu.:168  3rd Qu.:65.0  3rd Qu.:32.0
         Max.   :170  Max.   :70.0  Max.   :40.0
>
```

Imagen 128. Salida R Resumen data frame

A continuación se presentan algunos conceptos y comandos útiles para realizar el análisis exploratorio de observaciones. Existen varias formas para calcular medidas de interés en este tipo de análisis, las cuales dependen de la presentación de la matriz de datos.

8.2 VECTOR DE MEDIAS

Es la medida de centralización más usada para describir datos multivariantes; es el vector constituido por los promedios de cada una de las variables. Cuando la matriz de datos es construida con el comando `cbind()` se cuenta con dos procedimientos para calcular el vector de medias. El primero es de forma matricial, tal como se muestra a continuación:

$$\text{vecmedias} = \frac{1}{n} X' \mathbf{1}$$

Ecuación 15. Vector de medias

donde $\mathbf{1}$ representa un vector de unos con dimensión igual al tamaño de la muestra n , y X es la matriz de datos (para el caso es la matriz de datos traspuesta).

Ejemplo: Dadas las variables x_1 , x_2 y x_3 , determinar el vector de medias.

```

>
> # Variables
> x1=c(23,15,46,25,32)
> x2=c(65,70,59,71,68)
> x3=c(173,168,159,150,154)
>
> # Matriz de datos
> X= cbind(x1,x2,x3)
> X
      x1 x2  x3
[1,] 23 65 173
[2,] 15 70 168
[3,] 46 59 159
[4,] 25 71 150
[5,] 32 68 154
>
>
> # Tamaño muestral
> n = 5
>
> # Vector de unos
> unos = c(rep(1,n))
>
> # Vector de medias
> vecmedias = (1/n)*t(X)%*%unos
> vecmedias
      [,1]
x1  28.2
x2  66.6
x3 160.8
>

```

Imágenes 129 y 130. Salida R Vector de medias

El segundo procedimiento es mediante el comando `apply(datos, 1 ó 2, mean)`; recuerde que si se deja 1 se calcula la función por filas, y si se deja 2 se calcula la función por columnas.

```

>
> vecmedias=apply(X,2,mean)
> vecmedias
      x1  x2  x3
28,2 66.6 160.8
>

```

Imagen 131. Salida R Vector de medias

8.3 MATRIZ DE VARIANZAS Y COVARIANZAS

Esta matriz permite determinar, por una parte, la variabilidad respecto a la media de cada una de las variables, y, por otra, la relación lineal por pares de variables, si esta existe. Al igual que en el cálculo del vector de medias, existen varias opciones. Cuando la matriz de datos es construida con el comando `cbind()`, se cuenta con dos procedimientos para calcular la matriz de varianzas y covarianzas; el primero es de forma matricial, tal como se indica en la expresión siguiente:

$$\widehat{S} = \frac{1}{n-1} X^t P X$$

$$P = I - \frac{1}{n} 11^t$$

Ecuación 16. Matriz de varianzas

Con la matriz identidad de orden n y el vector $\mathbf{1}$ de dimensión n .

Ejemplo: Considere las variables x_1, x_2, x_3 y x_4 y construya la matriz de varianzas y covarianzas.

```
>
> # Variables
> x1=c(23,15,46,25,32)
> x2=c(65,70,59,71,68)
> x3=c(173,168,159,150,154)
>
> # Matriz de datos
> X= cbind(x1,x2,x3)
>
> # Tamaño de la muestra
> n = 5
>
> # Vector de unos
> unos = c(rep(1,5))
>
> # Matriz identidad
> I = diag(5)
>
> # Matriz P
> P = I-(1/n)*unos%*%t(unos)
>
> # Matriz de varianzas y covarianzas
> cov = (1/(n-1))*t(X)%*%P%*%X
> cov
      x1      x2      x3
x1 135.70 -45.15 -45.45
x2 -45.15  23.30  -9.60
x3 -45.45  -9.60  91.70
>
```

Imagen 132. Salida R Matriz de varianzas

El segundo procedimiento es mediante el comando `cov(nombre de la matriz de datos)`.

```
>
> # Matriz de varianzas y covarianzas
> cov(X)
      x1      x2      x3
x1 135.70 -45.15 -45.45
x2 -45.15  23.30  -9.60
x3 -45.45  -9.60  91.70
>
```

Imagen 133. Salida R Matriz de varianzas

8.4 MATRIZ DE CORRELACIONES

La dependencia por pares entre las variables se mide por la matriz de correlación R , matriz cuadrada y simétrica con unos en su diagonal principal y fuera de ella los coeficientes de correlación lineal entre pares de variables. Esta matriz se puede calcular de forma matricial de la siguiente forma:

$$R = D^{-1/2} S D^{-1/2}$$

Ecuación 17. Matriz de correlaciones

donde D corresponde a la matriz diagonal formada por los elementos de la diagonal principal de la matriz de varianzas y covarianzas muestrales S .

Ejemplo: Para la ilustración considere los mismos datos utilizados en los ejemplos del vector de medias y matriz de varianzas y covarianzas.

```

>
> # Matriz de varianzas y covarianzas
> S = cov(X)
>
> # Elementos de la diagonal de S
> E = diag(S)
>
> Matriz diagonal "diagonal(S)"
> D = diag(E)
>
> # Matriz de correlaciones
> R = solve(sqrt(D))%*%S%*%solve(sqrt(D))
> R
           [,1]      [,2]      [,3]
[1,]  1.0000000 -0.8029525 -0.4074359
[2,] -0.8029525  1.0000000 -0.2076867
[3,] -0.4074359 -0.2076867  1.0000000
>

```

Imagen 134. Salida R Matriz de correlaciones

Para realizar el cálculo directo de la matriz de correlaciones se recurre al comando `cor(matriz de datos)`.

```

>
> # Calculo directo de la matriz de correlaciones
> R = cor(X)
> R
           x1      x2      x3
x1  1.0000000 -0.8029525 -0.4074359
x2 -0.8029525  1.0000000 -0.2076867
x3 -0.4074359 -0.2076867  1.0000000
>

```

Imagen 135. Salida R Matriz de correlaciones

8.5 CÁLCULOS A PARTIR DE UN DATA FRAME

Un data frame puede estar formado tanto de variables cualitativas como cuantitativas; por esta razón se hace necesario que las variables cualitativas sean excluidas al momento de determinar el vector de medias, la matriz de varianzas y la matriz de correlaciones de cada matriz de datos.

Ejemplo: Considere las siguientes cuatro variables: edad (x1), peso (x2), estatura en cm (x3) y genero (x4, H=hombre y M=mujer).

```

> # Variables
> x1=c(23,15,46,25,32)
> x2=c(65,70,59,71,68)
> x3=c(173,168,159,150,154)
> x4=c("H","M","M","H","H")
>
> # Matriz de datos
> X=data.frame(x1,x2,x3,x4)
> X
  
```

	x1	x2	x3	x4
1	23	65	173	H
2	15	70	168	M
3	46	59	159	M
4	25	71	150	H
5	32	68	154	H

Imagen 136. Salida R Matriz de datos data frame

Con esta matriz se puede determinar el vector de medias, la matriz de varianzas y la de correlaciones; se debe tener en cuenta que para realizar estos cálculos se debe obviar la variable x_4 .

```

> # Vector de medias
> vecmed = mean(X[,-c(4)])
> vecmed
  
```

	x1	x2	x3
	28.2	66.6	160.8

```

>
> # Matriz de varianzas
> mvar = var(X[,-c(4)])
> mvar
  
```

	x1	x2	x3
x1	135.70	-45.15	-45.45
x2	-45.15	23.30	-9.60
x3	-45.45	-9.60	91.70

```

>
> # matriz de correlaciones
> mcor = cor(X[,-c(4)])
> mcor
  
```

	x1	x2	x3
x1	1.0000000	-0.8029525	-0.4074359
x2	-0.8029525	1.0000000	-0.2076867
x3	-0.4074359	-0.2076867	1.0000000

Imagen 137: Salida R Matriz de datos data frame

Como se puede apreciar en el ejemplo anterior, se hizo necesario eliminar la cuarta variable para el cálculo de algunas medidas numéricas de interés para la matriz de datos; de ser necesario se puede eliminar más de una columna; esto se hace escribiendo el número de la columna por eliminar dentro del argumento $X[, -c(\text{variables a ser eliminadas})]$; las variables deben estar separadas por comas.

8.6 DISTANCIA DE MAHALANOBIS

Se define la distancia de mahalnobis entre un punto y su vector de medias por:

$$d_i^2 = [(x_i - \bar{x})' S^{-1} (x_i - \bar{x})]$$

Ecuación 18. Distancia de mahalnobis

La distancia de mahalnobis se calcula a través del comando **mahalanobis(x, center, cov)**, donde **x**, matriz de datos; **center**, vector de medias, y **cov**, matriz de varianzas y covarianzas.

Ejemplo: En la siguiente tabla se presentan medidas antropométricas tomadas a 15 trabajadores del sector alfarero del municipio de Ráquira (Boyacá); las variables en estudio son: estatura (EST), alcance lateral con asimiento (ALA), alcance frontal con asimiento (AFA), altura vertical con asimiento (AVA) y piso-codo (PC).

*Tabla 7. Datos alfareros
Tomado de estudio de alfareros de Boyacá Grupo Taller 11*

Observaciones	EST	ALA	AFA	AVA	PC
1	148	74	74	185	92
2	160	81	81	200	102
3	140	72	71	176	92
4	176	84	84	213	113
5	160	80	82	198	105
6	162	80	71	196	99
7	166	89	86	207	105
8	144	73	72.5	180.5	92
9	160	84	83	201	98
10	163	82	86	204	103
11	150	74	76	184	95
12	172	86	84	215	110
13	158	82	79	202	101
14	158	76	77	194	105
15	158	82	80	197	100

Inicialmente se procede a construir la matriz de datos y a determinar el vector de medias y la matriz de varianzas y covarianzas.

```

>
> # Variables
> Est=c(148,160,140,176,160,162,166,144,160,163,150,172,158,158)
> ALA=c(74,81,72,84,80,80,89,73,84,82,74,86,82,76,82)
> AFA=c(74,81,71,84,82,71,86,72.5,83,86,76,84,79,77,80)
> AVA=c(185,200,176,213,198,196,207,180.5,201,204,184,215,202,194,197)
> PC=c(92,102,92,113,105,99,105,92,98,103,95,110,101,105,100)
>
> # Matriz de datos
> X = cbind(EST,ALA,AFA,AVA,PC)
>
> # Vector de medias
> Vecmed = apply(X,2,mean)
>
> # Matriz de varianzas y covarianzas
> S = cov(X)

```

Imagen 138. Salida T Distancia de mahalanobis

Luego de esto se procede a calcular la distancia de mahalanobis para cada observación, y por último se presenta una matriz con la información de cada individuo con su correspondiente distancia.

```

>
> # distancia de mahalanobis
> di = mahalanobis(X, Vecmed, S)
> # Observaciones y distancias correspondientes
> Xdi = cbind(X,di)
> Xdi

```

	EST	ALA	AFA	AVA	PC	di
[1,]	148	74	74.0	185.0	92	3.9692477
[2,]	160	81	81.0	200.0	102	0.3449645
[3,]	140	72	71.0	176.0	92	5.6556945
[4,]	176	84	84.0	213.0	113	6.0571089
[5,]	160	80	82.0	198.0	105	2.6372598
[6,]	162	80	71.0	196.0	99	10.5634604
[7,]	166	89	86.0	207.0	105	8.5471896
[8,]	144	73	72.5	180.5	92	2.2789303
[9,]	160	84	83.0	201.0	98	4.1267558
[10,]	163	82	86.0	204.0	103	4.7876563
[11,]	150	74	76.0	184.0	95	3.9927099
[12,]	172	86	84.0	215.0	110	4.6368720
[13,]	158	82	79.0	202.0	101	6.7518356
[14,]	158	76	77.0	194.0	105	4.5135475
[15,]	158	82	80.0	197.0	100	1.1367675

```

>

```

Imagen 139. Salida R Distancia de mahalanobis

8.7 ANÁLISIS GRÁFICO DE OBSERVACIONES MULTIVARIANTES

Un primer paso en el análisis multivariante es representar gráficamente las variables individualmente; en segundo lugar es conveniente construir diagramas de dispersión de las variables por parejas; esto se puede realizar mediante el comando `pairs(datos,...)`. A continuación se presenta un ejemplo con los datos de los trabajadores alfareros.

```
>  
> # Diagrama de dispersión  
> pairs(x, pch=5, col="blue",  
+ main="Gráficos de dispersión bivalente")  
>
```

Imagen 140. Salida R Construcción de diagrama bivalente

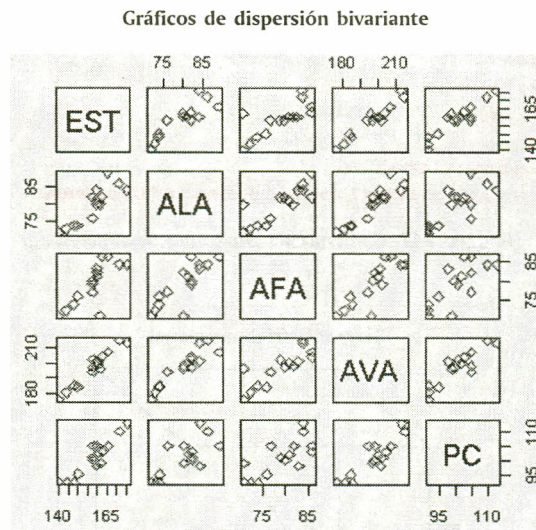


Imagen 141. Salida R Diagrama de dispersión bivalente

Este tipo de gráfico permite observar relaciones existentes entre las variables y la presencia de datos atípicos. Cuando se trabaja con tres o cuatro variables, la función `coplot()` puede ser más apropiada. Si `a` y `b` son vectores numéricos y `c` es un vector numérico o un factor (todos de la misma longitud), entonces, la orden `coplot(a ~ b / c)` produce diagramas de dispersión de `a` sobre `b` para cada valor de `c`. Si `c` es un factor, esto significa que `a` se representa sobre `b` para cada nivel de `c`. Si `c` es un vector numérico, entonces se agrupa en intervalos, y para cada intervalo se representa `a` sobre `b` para los valores de `c` dentro del intervalo. El número y tamaño de los intervalos puede controlarse con el argumento `given.values` de la función `coplot()`. La función `co.intervals()` también es útil para seleccionar intervalos. Asimismo, es posible utilizar dos variables condicionantes con una orden como `coplot(a ~ b / c + d)`, que produce diagramas de `a` sobre `b` para cada intervalo de condicionamiento de `c` y `d`.

Gráficos de dispersión 3 - variante: Cuando se tienen tres variables numéricas es posible realizar un diagrama de dispersión con ellas mediante el siguiente comando:

```
scatterplot3d(x, y=NULL, z=NULL, color=par("col"), pch=NULL,
  main=NULL, sub=NULL, xlim=NULL, ylim=NULL, zlim=NULL,
  xlab=NULL, ylab=NULL, zlab=NULL,...)
```

En el anterior comando sólo se presentan algunos argumentos; para mayor información se puede consultar la ayuda interactiva. Para realizar este diagrama es necesario que previamente se cargue el paquete `scatterplot3d`.

Ejemplo: Considere las variables x, y, z para realizar un diagrama de dispersión tri-dimensional.

```
>
> # Variables
> x = c(1,5,7,9,12)
> y = c(12,3,4,20,7)
> z = c(5,21,16,2,13)
>
> # Diagrama tri-dimensional
> scatterplot3d(x,y,z,color=3,pch=15,main="Diagrama tri-dimensional")
```

Imagen 142. Construcción Diagrama tri-dimensional

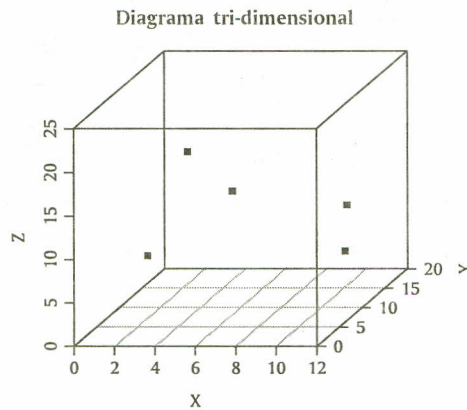


Imagen 143. Salida R Diagrama tri-dimensional

8.8 DISTRIBUCIÓN NORMAL MULTIVARIADA

El vector aleatorio p -dimensional x tiene distribución normal p -variante con vector de medias $\hat{\mu} = (\hat{\mu}_1, \hat{\mu}_2, \hat{\mu}_3, \dots, \hat{\mu}_p)$ y matriz de covarianzas $\hat{\Sigma}$ de tamaño $p \times p$, por ello tiene como función de densidad conjunta a:

$$f_x(x) = \frac{1}{(2\pi)^{p/2} |\Sigma|^{1/2}} \exp \left[-\frac{1}{2} (X - \mu)' \Sigma^{-1} (X - \mu) \right]$$

Ecuación 19. Distribución Normal

Es posible generar datos aleatorios de una distribución p-variante con el comando `mvrnorm(n=#, mu, Sigma)`, donde **n** indica el número de observaciones que se desean; **mu** es el vector de medias, y **sigma**, la matriz de varianzas y covarianzas.

Ejemplo: Si se quiere generar 6 observaciones de una distribución 5-variante con vector de medias $\mu = (2, 3, 4, 5, 6)$ y matriz de varianzas y covarianzas igual a la identidad, se procede así:

```
>
> # Número de observaciones
> n = 6
>
> # Vector de medias
> mu = c(2,3,4,5,6)
>
> # Matriz de varianzas y covarianzas
> sigma = diag(5)
>
> # Normal 5-variante
> mvrnorm(n, mu, sigma)
      [,1] [,2] [,3] [,4] [,5]
[1,] 2.7439461 2.195412 4.160518 2.546877 5.532961
[2,] 0.8959809 2.289943 4.077959 6.917811 6.342687
[3,] 1.2166718 4.192020 2.870380 3.957813 6.321503
[4,] 2.4007936 2.104596 4.764110 5.553305 6.567440
[5,] 2.5109468 4.529655 3.716143 6.777010 6.213452
[6,] 3.3561570 1.673395 3.469901 5.473750 5.037210
>
```

Imagen 144. Salida R Normal multivariante

8.9 ELIPSES DE CONFIANZA

Un caso particular de la distribución normal multivariante se presenta cuando $p=2$, con lo que se genera la distribución normal bivariada, utilizada en muchas aplicaciones de la vida cotidiana; a continuación se muestra cómo construir las coordenadas de elipses de confianza del $(1-a)100\%$ para un conjunto de n observaciones de una distribución normal bivariada; previamente se debe haber cargado el paquete **ellipse**; el comando utilizado es:

```
ellipse(x, centre, level = 0.95, npoints = )
```

Los argumentos utilizados son: **x**, matriz de correlaciones; **centre**, vector con las coordenadas del centro de la elipse (vector de medias); **level**, indica el nivel de confianza para la región, y

npoints indica el número de parejas ordenadas (puntos de la elipse). Para graficar esta elipse, el comando anterior se escribe dentro del comando **plot()**, así:

```
plot(ellipse(x, centre, level = 0.95, npoints = ))
```

Si la matriz de correlaciones es igual a la matriz identidad, entonces, la gráfica corresponderá a una circunferencia; a continuación se presentan ejemplos de los casos expuestos anteriormente.

Ejemplo: Dada una distribución normal bivalente con vector de medias μ y matriz de correlaciones R , construir una elipse de confianza del 92% para la distribución, donde,

$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \text{ y } R = \begin{bmatrix} 1 & 0.35 \\ 0.35 & 1 \end{bmatrix}$$

Imagen 145. Vector de medias y matriz de varianzas

```
>
> # centro de la elipse
> cen = c(0,0)
> # matriz de correlaciones
> R = matrix(c(1,0.35,0.35,1),2)
>
> # Elipse
> plot(ellipse(R,center=cen,level=0.92,npoints=100))
```

Imagen 146. Creación de elipse de confianza

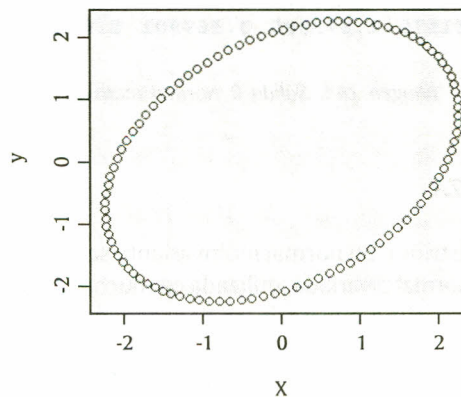


Imagen 147. Salida R elipse de confianza

Ejemplo: Dada una distribución normal bivalente con vector de medias μ y matriz de correlaciones R , construir una elipse de confianza del 96% para la distribución.

$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \text{ y } R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Imagen 148. Vector de medias y matriz de varianzas

```
>
> # centro de la elipse
> cen = c(0,0)
> # matriz de correlaciones
> R = diag(2)
>
> # Elipse
> plot(ellipse(R,center=cen,level=0.96,npoints=100))
```

Imagen 149. Construcción elipse de confianza

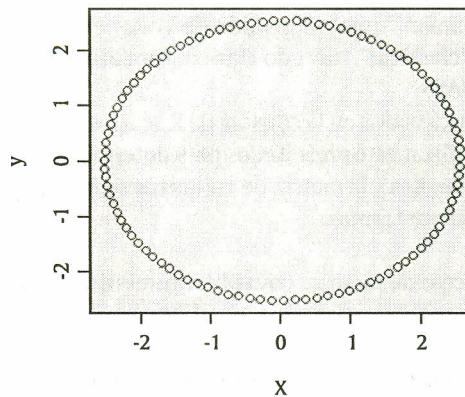


Imagen 150. Salida R elipse de confianza

Las gráficas anteriores pueden ser modificadas por medio de los parámetros gráficos anteriormente descritos (títulos y colores, entre otros).

8.10 EVALUACIÓN DE LA MULTINORMALIDAD

Un primer paso para probar la multinormalidad de un conjunto de observaciones es analizar cada una de las variables por separado, advirtiendo que esto no es suficiente, puesto que si solo se hiciera esto se estaría dejando de lado la asociación lineal entre las variables.

Datos atípicos: son aquellas observaciones que parecen haberse generado de manera distinta a las demás. Un primer procedimiento para identificar este tipo de observaciones es mediante gráficos y cálculo de distancias entre observaciones (distancia de mahalánobis) a fin de verificar si algún punto está alejado del conjunto de observaciones. Las consecuencias de una sola observación atípica pueden ser graves, entre estas se encuentran distorsión en promedios y

desviaciones estándar de las variables; por tanto, y como la distancia de mahalanobis está directamente relacionada con el vector de medias y la matriz de varianzas y covarianzas, puede no llegar a reflejar correctamente las observaciones atípicas (efecto de enmascaramiento). Una propuesta para obviar este problema es utilizar estimadores robustos, que son diseñados para verse poco afectados por cierta contaminación de atípicos (Peña, 2002).

Los estimadores robustos permiten realizar estimaciones para el vector de medias y la matriz de varianzas y covarianzas; estas estimaciones no se ven tan afectadas por la presencia de datos atípicos, y al utilizarlas para determinar la distancia de mahalanobis, esta refleja realmente los posibles alejamientos de un dato o un conjunto de datos de la población bajo estudio. El comando que permite realizar dichas estimaciones es:

```
cov.rob(x, cor=FALSE, method=c("mve", "mcd", "classical"))
```

Los argumentos utilizados en este comando son: **x**, matriz de datos; **cor**= función lógica por defecto FALSE, si es TRUE devuelve junto con los resultados la matriz de correlaciones; **method**= se refiere al método por el cual se realizan las estimaciones, en este caso los métodos implementados en R se llaman **"mve"** (Elipsoide de Volumen Mínimo); **"mcd"**, Covarianza de Determinante Mínimo, y **"classical"**, método clásico. Para utilizar este comando se debe cargar previamente el paquete MASS.

Al aplicar cualquiera de los métodos en la consola de R se aprecian todos los resultados como un solo objeto, y si se desea utilizar estos resultados para determinar la distancia de mahalanobis se necesita que el vector de medias y la matriz de varianzas sean objetos independientes, para lo cual se procede de la siguiente forma:

```
vector de medias: cov.rob(argumentos)$center
matriz de varianzas: cov.rob(argumentos)$cov
```

Ejemplo: Consiste en la generación de observaciones provenientes de dos distribuciones multinormales con distintos parámetros, con el fin de comparar los estimadores robustos frente a los estimadores usuales (vector de medias y matriz de varianzas y covarianzas muestrales).

Generación de muestras:

M1: muestra aleatoria de tamaño $n = 25$ de una distribución normal 3-variante con

$\mu_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \text{ y } \Sigma_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	<pre>> mu1 = c(0,0,0) > sigmal = diag(3) > x1 = mvrnorm(n=25,mu1,sigmal)</pre>
---	---

Imágenes 151 y 152. Construcción de muestras aleatorias M1

M2: muestra aleatoria de tamaño n=5 de una distribución normal 3-variante con

$\mu_2 = \begin{bmatrix} 15 \\ 85 \\ 70 \end{bmatrix} \text{ y } \Sigma_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	<pre>> mu2 = c(15,85,70) > sigma2 = diag(3) > x2 = mvrnorm(n=5,mu2,sigma2)</pre>
--	---

Imágenes 153 y 154. Construcción de muestras aleatorias M2

Creación de la matriz de datos. Se unen las dos muestras aleatorias dentro de un mismo arreglo mediante el comando `rbind()`.

```
> X=rbind(x1,x2)
> X
      [,1]      [,2]      [,3]
[1,] 0.430314213 -2.46499251 1.48551076
[2,] -0.812518531 -0.33530859 1.01089672
[3,] 1.536175155 -0.44371158 0.04336103
[4,] -2.103800963 -0.11578560 -1.02276904
[5,] 0.206273353 0.06576601 0.36081157
[6,] -1.381900544 -0.07797751 0.24924408
[7,] -0.871928839 0.88283780 0.17276662
[8,] -1.213167012 1.39217469 1.29961514
[9,] -1.005834566 0.39247735 0.60899808
[10,] -1.381900544 -0.07797751 0.24924408
[11,] 0.003762816 0.83687569 0.57117769
[12,] 1.147156783 -1.05345334 -0.82417008
[13,] -1.926147757 -1.20268938 1.78945174
[14,] 0.237389932 -0.22885753 -0.55974433
[15,] 0.139029616 1.22639374 -1.78587836
[16,] -0.531334715 0.72616816 -0.05103377
[17,] 0.213814969 0.97092876 0.7420808
[18,] -0.555377515 0.66231270 -1.15219277
[19,] 1.046889659 -0.78080671 0.26134942
[20,] -0.070029371 1.12885487 -0.51858915
[21,] -0.636398694 -1.61744701 0.48155900
[22,] -0.058228944 0.23079044 1.08871860
[23,] -1.036050912 0.85473871 0.91852479
[24,] 0.588420690 0.01574196 0.53243638
[25,] -1.409747460 -0.05213258 1.18323715
>
```

Imagen 155. Construcción matriz comando `rbind`

Estimadores usuales vector de medias, matriz de varianzas y covarianzas:

```
>
> vmedusual=apply(X,2,mean)
> vmedusual
[1] 2.25199 14.17923 12.06242
>
> covusual=cov(X)
> covusual
      [,1]      [,2]      [,3]
[1,] 35.04836 187.4725 155.9492
[2,] 187.47248 1029.9764 857.0561
[3,] 155.94925 857.0561 715.3208
>
```

Imagen 156. Salida R estimadores usuales

Estimador robusto (Elipsoide de Volumen Mínimo):

```
>
> vmedMVE=cov.rob(X,method="mve")$center
> vmedMVE
[1] -0.3000781 0.1959372 0.2842068
>
> covMVE=cov.rob(X,method="mve")$cov
> covMVE
      [,1]      [,2]      [,3]
[1,] 0.7664138 -0.1609999 -0.3356907
[2,] -0.1609999 0.9497991 -0.2514190
[3,] -0.3356907 -0.2514190 0.7692446
>
```

Imagen 157. Salida R estimador mve

Estimador robusto (Covarianza de Determinante Mínimo):

```
>
> vmedMCD=cov.rob(X,method="mcd")$center
> vmedMCD
[1] -0.2732624 0.1934830 0.3164735
>
> covMCD=cov.rob(X,method="mcd")$cov
> covMCD
      [,1]      [,2]      [,3]
[1,] 0.77968740 -0.08684056 -0.3867783
[2,] -0.08684056 0.68510753 -0.1274671
[3,] -0.38677827 -0.12746706 0.7446783
>
```

Imagen 158. Salida R estimadores mcd

Cálculo de los cuadrados de la distancia de mahalanobis para cada uno de los estimadores:

```
>
> # Distancias estimador usual
> diusual=mahalanobis(X,vmedusual,covusual)
>
> # Distancias estimador MVE
> diMVE=mahalanobis(X,vmedMVE,covMVE)
>
> # Distancias estimador MCD
> diMCD=mahalanobis(X,vmedMCD,covMCD)
```

Imagen 159. Salida R Distancia de mahalanobis para los estimadores

Gráficas para las distancias calculadas con cada uno de los estimadores

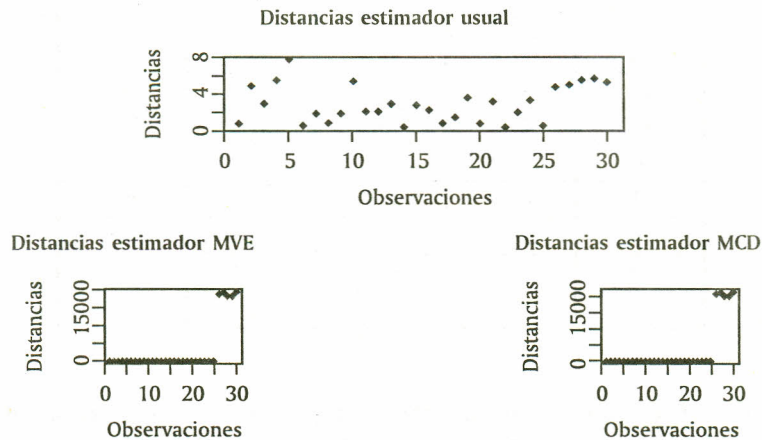


Imagen 160. Salida R Gráfica de las distancias de mahalanobis

Se observa claramente en los gráficos de las distancias de mahalanobis para los estimadores MVE y MCD que las observaciones con las que se contaminó el primer conjunto de datos están alejadas de este, mientras que en el gráfico para las distancias calculadas con el estimador usual estas observaciones se pueden llegar a confundir dentro del conjunto. El ejemplo anterior permitió verificar la eficacia de los estimadores robustos en la detección de datos atípicos cuando la matriz de datos es contaminada a propósito con datos provenientes de una distribución diferente a los datos iniciales de la matriz.

Ejemplo: Ahora se aplicarán los estimadores robustos a un conjunto de datos trabajados por Díaz (2002, p. 74), en un ejercicio en el que mediante diferentes procedimientos determina que las observaciones 9, 12 y 20 son datos potencialmente atípicos. En la siguiente tabla se muestran los datos sobre longitud de huesos registrados en 20 jóvenes a los 8, 8.5, 9 y 9.5 años, respectivamente (Rencher, 1995, p. 90, citado por Díaz):

Tabla 8. Datos sobre longitud de huesos

# obs	8 años(x_1)	8.5 años(x_2)	9 años(x_3)	9.5 años(x_4)
1	47.5	48.8	49.0	49.7
2	46.4	47.3	47.7	48.4
3	46.3	46.8	47.8	48.5
4	45.1	45.3	46.1	47.2
5	47.6	48.5	48.9	49.3
6	52.5	53.2	53.3	53.7
7	51.2	53.0	54.3	54.5
8	49.8	50.0	50.3	52.7
9	48.1	50.8	52.3	54.4
10	45.0	47.0	47.3	48.3
11	51.2	51.4	51.6	51.9
12	48.5	49.2	53.0	55.5
13	52.1	52.8	53.7	55.0
14	48.2	48.9	49.3	49.8
15	49.6	50.4	51.2	51.8
16	50.7	51.7	52.7	53.3
17	47.2	47.7	48.4	49.5
18	53.3	54.6	55.1	55.3
19	46.2	47.5	48.1	48.4
20	46.3	47.6	51.3	51.8

En la siguiente gráfica se observa un diagrama de dispersión de las distancias de mahalanobis tanto con los estimadores usuales como con los robustos (MVE, MCD):

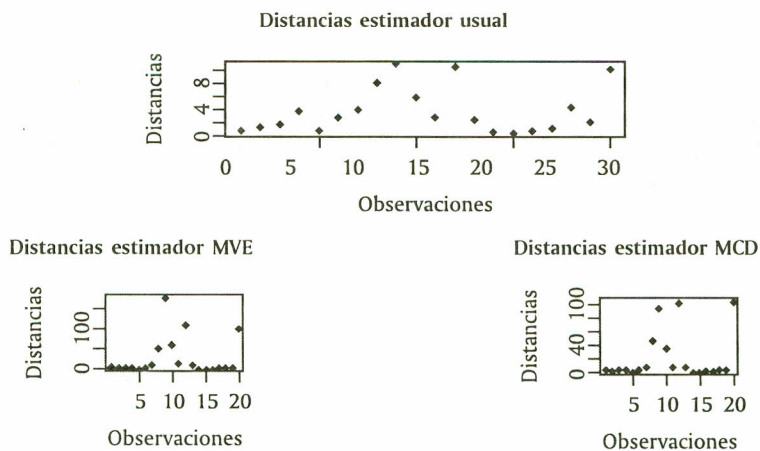


Imagen 161. Salida R Gráfica de las distancias de mahalanobis para los jóvenes

En los gráficos correspondientes a los estimadores robustos se identifican 5 posibles valores atípicos: los tres encontrados por Díaz (observaciones 9, 12 y 20) y dos observaciones adicionales (8 y 10) que surgen al utilizar los dos estimadores robustos.

8.11 EJERCICIOS

8.11.1 Los siguientes datos hacen referencia al seguimiento que la Secretaría de Salud viene realizando a 15 niños de una zona marginal de la ciudad.

Nombre	Edad (años)	Estatura (m)	Peso (kg)
José	12	1.4	48
Pedro	14	1.8	77
María	14	1.32	35
Carlos	16	1.6	40
Lucía	8	1.2	35
Maritza	9	1.4	35
Mariela	17	1.51	48
Mariana	15	1.56	52
Gabriela	12	1.3	45
Jesús	16	1.65	60
Oscar	15	1.7	62
David	9	1.2	30
Tania	12	1.4	40
Liliana	15	1.6	48
Lina	17	1.56	57

- Introduzca estos datos en R como un data frame.
- Construya: Vector de medias, matriz de varianzas y covarianzas, matriz de correlaciones.
- Calcule la distancia de mahalanobis para cada observación.
- Realice el análisis gráfico multivariante para el ejercicio.

8.11.2 Genere 10 observaciones de una distribución 6–variante con vector de medias $\mu = (5, 8, 2, 11, 3, 20)$ y matriz de varianzas y covarianzas igual a la identidad.

9. ANÁLISIS DE REGRESIÓN

La regresión es una técnica estadística utilizada para simular la relación existente entre dos o más variables. Por lo tanto, se puede emplear para construir un modelo que permita predecir el comportamiento de una variable dada. En la construcción de un modelo en R, el operador \sim se utiliza para definir una fórmula. La forma para un modelo lineal ordinario es:

var respuesta \sim **ope_1 term_1 ope_2 term_2 ...**

donde **var respuesta** es un vector o una matriz que definen, respectivamente, la o las variables respuesta; **ope_i** es un operador, bien + o bien -, que implica la inclusión o exclusión, respectivamente, de un término en el modelo. El primer (ope_1), de ser +, es opcional, no es completamente necesario; **term_i** es un término de uno de los siguientes tipos:

- una expresión vectorial, una expresión matricial o el número 1
- un factor
- una expresión de fórmula consistente en factores, vectores o matrices conectados mediante operadores de fórmula.

En todos los casos, cada término define una colección de columnas que deben ser añadidas o eliminadas de la matriz del modelo. Un 1 significa un término independiente y está incluido siempre, a no ser que sea eliminado explícitamente. A continuación se muestran algunos ejemplos de modelos estadísticos, teniendo en cuenta que y , x_0 , x_1 , x_2 , ... son variables numéricas, que X es una matriz y que A especifica los factores:

- $y \sim x$ o $y \sim 1 + x$ Ambos definen el mismo modelo de regresión lineal de y sobre x . El primero contiene el término independiente implícito y el segundo, explícito.
- $y \sim 0 + x$, $y \sim -1 + x$ o $y \sim x - 1$ Regresión lineal de y sobre x sin término independiente; esto es, que pasa por el origen de coordenadas.
- $\log(y) \sim x_1 + x_2$ Regresión múltiple de la variable transformada, $\log(y)$, sobre x_1 y x_2 (con un término independiente implícito).
- $y \sim \text{Poly}(x, 2)$ o $y \sim 1 + x + I(x^2)$ Regresión polinomial de y sobre x de segundo grado. La primera forma utiliza polinomios ortogonales y la segunda utiliza potencias de modo explícito.
- $y \sim X + \text{Poly}(x, 2)$ Regresión múltiple de y con un modelo matricial consistente en la matriz X , términos polinomiales en x de segundo grado.
- $y \sim A$ Análisis de varianza de entrada simple de y , con clases determinadas por A .
- $y \sim A * x$, $y \sim A/x$ ó $y \sim A/(1-x) - 1$ Modelos de regresión lineal simple separados de y sobre x para cada nivel de A . La última forma produce estimaciones explícitas de tantos términos independientes y pendientes como niveles tiene A .

9.1 MODELO LINEAL

El comando `lm()` es utilizado para ajustar modelos lineales mediante mínimos cuadrados:

`lm(formula, data, model = TRUE)`

Los argumentos utilizados en el comando `lm()` son: **fórmula** se refiere a la fórmula utilizada en la construcción del modelo a ajustar; **data**, al argumento opcional si los datos provienen de un `data.frame` que contiene las variables involucradas en el modelo; **model**, al argumento lógico, si es `TRUE` arroja los componentes del modelo ajustado. Además, hay más argumentos que se pueden consultar en la ayuda interactiva. El comando `lm()` arroja algunos resultados simples; las funciones `summary` y `anova` son utilizadas para obtener resumen y tabla de análisis de varianza, respectivamente, del modelo ajustado.

Ejemplo: El supervisor de mantenimiento de una línea de autobuses cree que existe una relación entre el costo anual de mantenimiento de las unidades y los años que llevan de operación, y considera que si tal relación existe podrá hacer un mejor pronóstico de presupuesto. Los datos tomados por el supervisor sobre 15 autobuses de la empresa se muestran en la siguiente tabla, x =tiempo de operación en años, y =costo de mantenimiento:

Tabla 9. Datos sobre autobuses

x	8	5	3	9	11	2	1	8	12	4	7	10	6	3	9
y	8.6	6.8	4.7	7	11	2.2	3.2	6.5	10.5	5.6	6.8	10.8	6.2	5	8

Modelo ajustado:

```
> # Tiempo de operación (años)
> x=c(8,5,3,9,11,2,1,8,12,4,7,10,6,3,9)
>
> # Costo de mantenimiento
> y=c(8.6,6.8,4.7,7,11,2.2,3.2,6.5,10.5,
+     5.6,6.8,10.8,6.2,5,8)
> # Modelo lineal
> lm( y ~ x, model = TRUE)
```

```
Call:
lm(formula = y ~ x, model = TRUE)
```

```
Coefficients:
(Intercept)          x
      2.215         0.711
```

Imagen 162. Salida R Modelo ajustado

Como se observa, el comando `lm()` arroja resultados sencillos; de necesitar aspectos más específicos del modelo se deben usar funciones extractoras de información del modelo; la descripción de algunas funciones es:

- *formula*(lm(modelo)): Extrae la fórmula del modelo.
- *coefficients*(lm(modelo)): Extrae la matriz de coeficientes de regresión. Forma reducida: *coef*(lm(modelo)).
- *summary*(lm(modelo)): Imprime un resumen estadístico completo de los resultados del análisis de regresión.
- *anova*(lm(modelo)): Compara un submodelo con un modelo externo y produce una tabla de análisis de varianza.
- *residuals*(lm(modelo)): Extrae la matriz de residuos, ponderada si es necesario. La forma reducida es *resid*(lm(modelo)).
- *plot*(lm(modelo)): Crea cuatro gráficos que muestran los residuos, los valores ajustados y otros gráficos de diagnósticos para examinar la calidad del modelo.
- *predict*(lm(modelo)): El resultado es un vector o matriz de valores predichos correspondiente a los valores de las variables de los datos.
- *deviance*(lm(modelo)): Suma de cuadrados residual, ponderada si es lo apropiado.
- *step*(lm(modelo)): Selecciona un modelo apropiado añadiendo o eliminando términos y preservando las jerarquías. Se devuelve el modelo que en este proceso tiene el máximo valor de AIC (Criterio de información de Akaike).

Teniendo en cuenta estas funciones extractoras, se presentan algunas ilustraciones para el modelo ajustado, usando el ejemplo de mantenimiento de la línea de autobuses. Para obtener un resumen del modelo se utiliza el comando `summary()`:

```
> # Resumen del modelo
> summary( lm( y ~ x, model = TRUE ) )

Call:
lm(formula = y ~ x, model = TRUE)

Residuals:
    Min       1Q   Median       3Q      Max
-1.6139 -0.5029  0.2744  0.6747  1.4751

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.21458     0.56945   3.889  0.00186 **
x            0.71103     0.07778   9.141 5.05e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.9953 on 13 degrees of freedom
Multiple R-squared:  0.8654,    Adjusted R-squared:  0.855
F-statistic: 83.57 on 1 and 13 DF,  p-value: 5.045e-07
```

Imagen 163. Salida R Resumen del Modelo ajustado

Para obtener la tabla de análisis de varianza se utiliza el comando `anova()`, como se muestra:


```

> # Tabla de análisis de varianza para el modelo
> anova( lm( y ~ x, model = TRUE ) )
Analysis of Variance Table

Response: y
      Df Sum Sq Mean Sq F value    Pr(>F)
x       1  82.779   82.779   83.567 5.045e-07 ***
Residuals 13  12.877    0.991
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Imagen 164. Salida R Anova del Modelo ajustado

Mediante el comando plot() se generan cuatro gráficos en una ventana interactiva, en la cual se pasa de un gráfico a otro con hacer un clic o dar un enter.

```

> # Gráfico para analizar el modelo
> plot( lm(y~x,model=TRUE) )

```

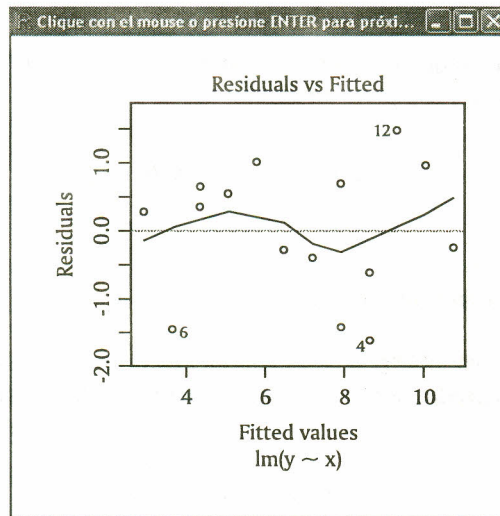


Imagen 165. Salida R Gráficos para el Modelo ajustado

9.2 MODELOS LINEALES GENERALIZADOS (GLM)

Cuando la variable respuesta es discreta o categórica, el modelo lineal clásico no es apropiado. Nelder y Wedderburn, en 1972, extendieron la teoría de los modelos lineales a una familia más amplia: la familia exponencial de densidades, denominándolos Modelos Lineales Generalizados (GLM), (Demétrio, 2001). Los GLM pueden ser usados cuando se tiene un único vector aleatorio Y asociado a un conjunto de variables explicativas o covariables X_1, X_2, \dots, X_p . Un GLM se define a través de tres componentes (Demétrio):

Componente aleatorio: Representado por un conjunto de variables aleatorias independientes Y_1, Y_2, \dots, Y_n provenientes de una misma distribución que hace parte de la familia exponencial de densidades. La familia exponencial de densidades fue propuesta por Pitman, Koopman y Darmois (Demétrio). Una distribución pertenece a esta familia si su función de densidad se puede llevar a la forma:

$$f(y, \theta, \phi) = \exp \left[\frac{1}{a(\phi)} (y\theta - b(\theta) + c(y, \phi)) \right]$$

Ecuación 20. Familia exponencial de densidades

Componente sistemático: Para un GLM se considera un conjunto pequeño de parámetros $\beta_1, \beta_2, \dots, \beta_p$ tal que la combinación lineal de los $\hat{\alpha}$'s es igual a:

$$\eta_i = X_i' \beta$$

Ecuación 21. Componente sistemático

Función de enlace: Función que relaciona la media con el predictor lineal, es decir, enlaza el componente aleatorio con el componente sistemático:

$$g(\mu_i) = \eta_i = X_i' \beta$$

Ecuación 22. Función de enlace

Siendo $g(\cdot)$ una función monótona derivable y $\mu_{it} = E(Y_i)$.

Cada distribución para la variable respuesta admite una variedad de funciones de enlace para conectar la media con el predictor lineal. La siguiente tabla recopila las que están disponibles automáticamente en R.

Tabla 10: Funciones de enlace

Nombre de la familia	Función de enlace
binomial	logit, probit, cloglog
gaussian	Identity
Gamma	identity, inverse, log
inverse.gaussian	1/mu ^ 2
poisson	identity, log, sqrt
quasi	logit, probit, cloglog, identity, inverse, log, 1/mu ^ 2

El comando `glm()` permite ajustar un modelo lineal generalizado y tiene la siguiente estructura.

`glm(formula, family=familia(link = "función de enlace"), data =)`

Los argumentos utilizados en el comando anterior son: **formula**, que especifica el modelo; **family**, que especifica la familia, y la función de enlace (**link= ""**) que se desee utilizar, y **data=**, que determina el conjunto de variables presentes en el modelo; estas pueden provenir de un **data.frame**, lista u otro ámbito permitido. Estos no son los únicos parámetros disponibles; para mayor información consulte la ayuda interactiva **help(glm)**.

9.3 EJERCICIOS

9.3.1 En una encuesta realizada a 13 familias de la región se observaron las variables: número de integrantes de la familia (N) y gasto en alimentación por familia en miles (G); los resultados se muestran en la siguiente tabla:

N	3	2	5	4	6	3	2	4	5	5	6	4	3
G	150	120	180	180	210	160	90	150	200	150	225	170	100

- Obtenga el modelo lineal que explica el gasto en alimentación de las familias en función de su tamaño.
- Obtenga un resumen estadístico completo de los resultados del análisis de regresión.
- Calcule los residuales y los valores predichos.

9.3.2 Un Supermercado ha decidido abrir una sucursal en la ciudad y decide estudiar el número de cajas registradoras que va a instalar, para evitar colas innecesarias a la hora de pagar. Para ello se obtuvieron los siguientes datos procedentes de otras sucursales en diferentes ciudades acerca del número de cajas registradoras (variable C) y del tiempo medio de espera en segundos (variable T).

N.º cajas	10	11	12	14	22	12	18	20
Tiempo	58	49	33	42	20	32	26	23

- Obtenga la ecuación que explica la relación entre estas variables.
- Obtenga la tabla de análisis de varianza.
- Construya gráficos que le permitan realizar el diagnóstico del modelo construido.
- Si se instalan 9 cajas registradoras, ¿cuál será el tiempo de espera?

10. DISEÑO DE EXPERIMENTOS

El diseño de un experimento es la secuencia completa de pasos previstos para asegurar que los datos apropiados se obtendrán de modo que permitan un análisis objetivo que conduzca a deducciones válidas con respecto al problema establecido.

Al igual que en el análisis de regresión, en el diseño de experimentos también se utilizan modelos que requieren fórmulas para el análisis; a continuación, algunas fórmulas utilizadas para definir modelos en el diseño de experimentos:

- $a + b$: efectos de a y b
- X : si X es una matriz, describe un efecto aditivo para cada una de las columnas.
- $a : b$: efecto interactivo entre a y b
- $a * b$: efectos aditivos e interactivos entre a y b
- $\text{poly}(a, n)$: polinomios de a hasta grado n
- $\wedge n$: incluye todas las interacciones hasta el nivel n
- $b\%in\%a$: los efectos de b están anidados en a
- $a - b$: remueve el efecto de b
- $- 1: y \sim x - 1$ regresión a través del origen (igual para $y \sim x + 0$ o $y \sim 0 + x$)
- $1: 1$ ajusta un modelo sin efectos (solo el intercepto)
- $\text{Offset}(\dots)$: agrega un efecto al modelo sin estimar los parámetros

Como se observa en la mayoría de las fórmulas para determinar cierto modelo, en un diseño de experimentos se trabaja con efectos que clasifican la variable respuesta; para construir estas variables clasificatorias es necesario construir un vector de caracteres o numérico que contenga la respectiva clasificación para cada elemento de la variable respuesta.

10.1 CONSTRUCCIÓN DE VARIABLES CLASIFICATORIAS

Para construir las variables clasificatorias es necesario construir un vector de caracteres o numérico que contenga la respectiva clasificación para cada elemento de la variable respuesta. Dado que la anterior construcción puede ser un poco dispendiosa en caso de tener gran cantidad de datos, se presentan a continuación varias opciones para construir variables de clasificación.

Diseños desbalanceados: En este caso podemos crear un vector mediante el comando $c()$ y dentro de él utilizar el comando $\text{rep}()$ para generar la cantidad respectiva de niveles.

Ejemplo: Si se tiene un diseño con tres niveles (a, b y c) de tamaños 2, 3 y 4, respectivamente, la forma para crear el vector con estos niveles está dada por:

```
>
> niveles=c(rep("a",2),rep("b",3),rep("c",4))
>
> niveles
[1] "a" "a" "b" "b" "b" "c" "c" "c" "c"
>
```

Imagen 166. Salida R generación de niveles

En algunos casos es conveniente modificar los atributos del vector generado; para el caso del vector cuyos atributos son caracteres, el cambio se realiza a través de la instrucción:

```
niv = as.factor(niveles)
```

Diseños balanceados: En este caso, como el número de réplicas por nivel es igual, se puede utilizar el comando `gl(n,k,length=n*k,labels=, ordered=FALSE)`, donde **n** determina el número de niveles, **k** determina el número de réplicas por nivel, **length = n*k** determina el tamaño del vector, **labels** es un vector opcional en el cual se puede especificar el nombre de cada uno de los niveles y **ordered** por defecto es FALSE, en caso de ser TRUE ordena los factores de menor a mayor, en caso de ser numéricos, y en orden alfabético en caso de ser caracteres.

Ejemplo: Considere que se necesita generar un vector que contenga 5 niveles con 3 réplicas por nivel.

```
> # N° Niveles
> n = 5
>
> # N° replicas
> k = 3
>
> # Nombre de cada nivel
> nombres = c("A","B","C","D","E")
>
> # Generación de niveles
> gl(n,k,length=n*k,labels=nombres)
> niveles
[1] A A A B B B C C C D D D E E E
LEVELS: A B C D E
```

Imagen 167. Salida R Generación de niveles comando gl

10.2 ANÁLISIS DE VARIANZA

La tabla de análisis de varianza resume el conocimiento acerca de la variabilidad en las observaciones del experimento, el comando utilizado en R para obtener un análisis de varianza es:

```
aov(formula, data = NULL, contrasts = NULL)
```

Los argumentos utilizados en el comando anterior son: **formula**, que especifica el modelo; **data**, en caso de ser un data.frame que incluye las variables presentes en el modelo, si las variables han sido construidas previamente se deja **data=NULL**. Al igual que los modelos de regresión, en el análisis de varianza se utilizan comandos extractores de información; algunos de ellos se presentan aquí:

- **summary.aov()**: Muestra un resumen del análisis de varianza (Tabla ANOVA).
- **model.tables()**: Construye una tabla en la cual se encuentran los efectos de cada uno de los tratamientos.
- **TukeyHSD()**: Crea un conjunto de intervalos de confianza sobre las diferencias entre las medias de los niveles; los intervalos son calculados a través de rangos estudentizados (método de diferencias verdaderamente significativas de Tukey), además, si al comando se le precede con el comando **plot(TukeyHSD)** se genera una gráfica para los intervalos construidos en la prueba.

Es de notar que algunos de los comandos extractores de información utilizados en el análisis de regresión también se pueden utilizar en el análisis de varianza (**predict** y **residuals**, entre otros). Ahora un ejemplo del libro *Diseño de experimentos*, de Robert O Kuehl (2001, p. 38).

Ejemplo: La vida de anaquel de las carnes almacenadas es el tiempo que un corte previamente empacado es sano, nutritivo y vendible. Un paquete normal expuesto al aire ambiental tiene una vida aproximada de 48 horas, después de las cuales la carne comienza a deteriorarse por contaminación de microbios, degradación del calor y encogimiento. El empaque al vacío es efectivo para suprimir el desarrollo de microbios, sin embargo, continúan siendo un problema los otros aspectos. Algunos estudios sugieren las atmósferas controladas de gas, como alternativa de los empaques actuales. Dos atmósferas que prometen combinar la capacidad de suprimir el desarrollo de microbios con la conservación de las cualidades de la carne son: 1) dióxido de carbono puro CO, y 2) mezclas de monóxido de carbono. En la siguiente tabla se muestran los datos obtenidos para el número de bacterias sicotrópicas ($\log(N^0/cm^2)$) en muestras de carne almacenadas en cuatro condiciones de empaque.

Tabla 11: Datos bacterias sicotrópicas

Condiciones de empaque	Bacterias ($\log(N^0/cm^2)$)
Empaque comercial (comercial)	7.66, 6.98, 7.80
Empaque al vacío (vacío)	5.36, 5.44, 5.80
Mezcla de gases (COON)	7.41, 7.33, 7.04
Dióxido de carbono (CO)	3.51, 2.91, 3.66

Con los datos se procede a correr el diseño de experimentos, así:

```

>
> # Factores de clasificación
> nombres=c("comercial","vacío","COON","CO")
> empaques=gl(4,3,length=12,labels=nombres)
>
> # Variables respuesta
> Bacterias=c(7.66,6.98,7.80,5.26,5.44,5.8,7.41,7.33,7.04,3.51,2.91,3.66)
>
> # Diseño de experimento
> aov(Bacterias~empaques)
Call:
  aov(formula = Bacterias ~ empaques)

Terms:
                empaques Residuals
Sum of Squares  32.8728    0.9268
Deg. of Freedom      3         8

Residual standard error: 0.3403674
Estimated effects may be unbalanced

```

Imagen 168. Salida R Diseño de experimentos

Como se aprecia en la imagen anterior, los resultados mostrados por el comando `aov()` son sencillos, por lo que si se requiere mayor información acerca del modelo para el diseño de experimentos se necesita trabajar con algunos otros comandos extractores; por ejemplo, la tabla de análisis de varianza se construye mediante el comando `anova()`. Para trabajar con los comandos extractores se utiliza una asignación para el modelo.

```

>
> # Asignación para el modelo
> Mbact = aov(Bacterias~empaques)
>
> # Tabla de análisis de varianza
> anova(Mbact)

Analysis of Variance Table

Responce: Bacterias
      Df Sum Sq Mean Sq F value    Pr(>F)
empaques  3  32.873   10.958   94.584 1.376e06 ***
Residuals  8   0.927    0.116
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Imagen 169. Salida R Modelo para el diseño

A continuación se construye una tabla en la cual se encuentran los efectos de cada uno de los tratamientos.

```

>
> # Efectos de los tratamientos
> model.tables(Mbact)
Tables of effects

empaques
empaques
comercial      vacío      COON      CO
      1.58      -0.40      1.36      -2.54

```

Imagen 170. Salida R Efectos de los tratamientos

Luego se construyen los intervalos de Tukey (método de diferencias verdaderamente significativas).

```

>
> # Prueba de Tukey
> TukeyHSD(Mbact)
Tukey multiple comparisons of means
95% family-wise confidence level

Fit: aov(formula = Bacterias ~ empaques)

$empaques
      diff      lwr      upr      p adj
vacío-comercial -1.98 -2.869962 -1.090038 0.0004549
COON-comercial  -0.22 -1.109962  0.669962 0.8563618
CO-comercial    -4.12 -5.009962 -3.230038 0.0000020
COON-vacío      1.76  0.870038  2.649962 0.0010160
CO-vacío        -2.14 -3.029962 -1.250038 0.0002639
CO-COON         -3.90 -4.789962 -3.010038 0.0000031

```

Imagen 171: Salida R Prueba de Tukey

Gráfica de los intervalos de Tukey `plot(TukeyHSD(Mbact))`

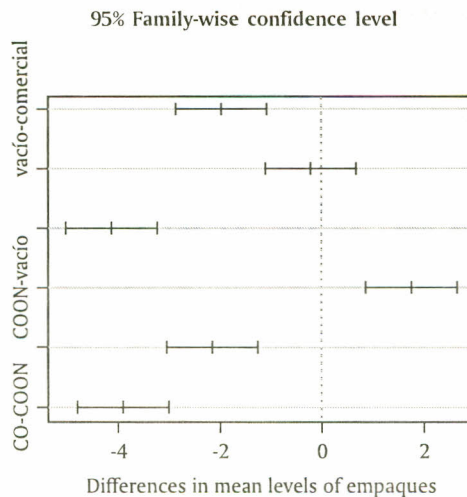


Imagen 172. Salida R Gráficos sobre los intervalos de Tukey

Como se ha mencionado, los comandos extractores utilizados en el análisis de regresión se pueden utilizar en el diseño de experimentos como se ilustra:

```
>
> # valores ajustados
> predict(Mbact)
  1    2    3    4    5    6    7    8    9   10   11   12
7.48 7.48 7.48 5.50 5.50 5.50 7.26 7.26 7.26 3.36 3.36 3.36
>
> # residuales
> residuals(Mbact)
  1    2    3    4    5    6    7    8    9   10   11   12
0.18 -0.50 0.32 -0.24 -0.06 0.30 0.15 0.07 -0.22 0.15 -0.45 0.30
```

Imagen 173. Salida R Comandos extractores

Una forma alternativa para elaborar la tabla de análisis de varianza y realizar las comparaciones entre tratamientos es mediante la creación de un archivo de texto (ver capítulo 11, importar y exportar datos). Para su elaboración, estando en la consola de R, en el menú Archivo, opción Nuevo Script, se despliega una ventana de edición donde se escribe la información de los tratamientos y los correspondientes valores de la variable respuesta, de manera que en la primera fila se ubique un nombre que se refiera a los tratamientos junto con un nombre para la variable que se está midiendo en el diseño de experimentos; con los datos del ejemplo de la vida de anaquel de las carnes, se tiene:

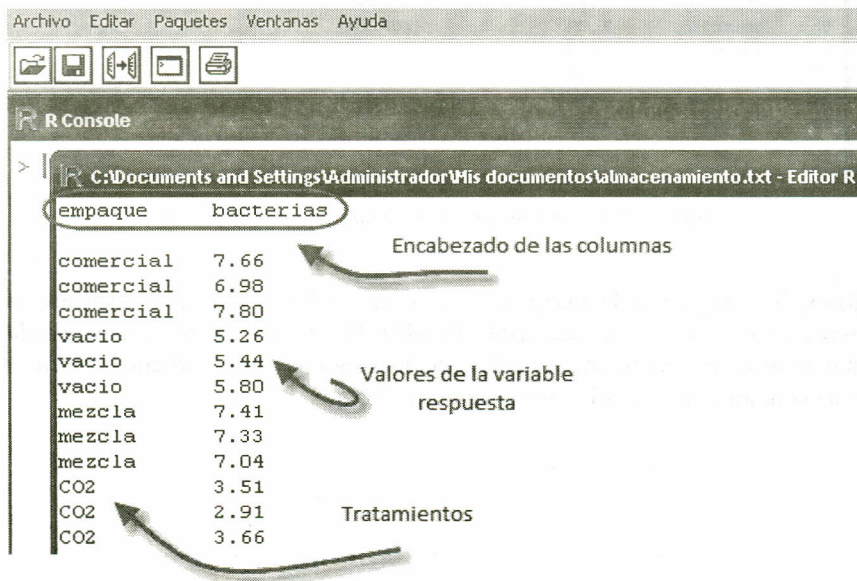
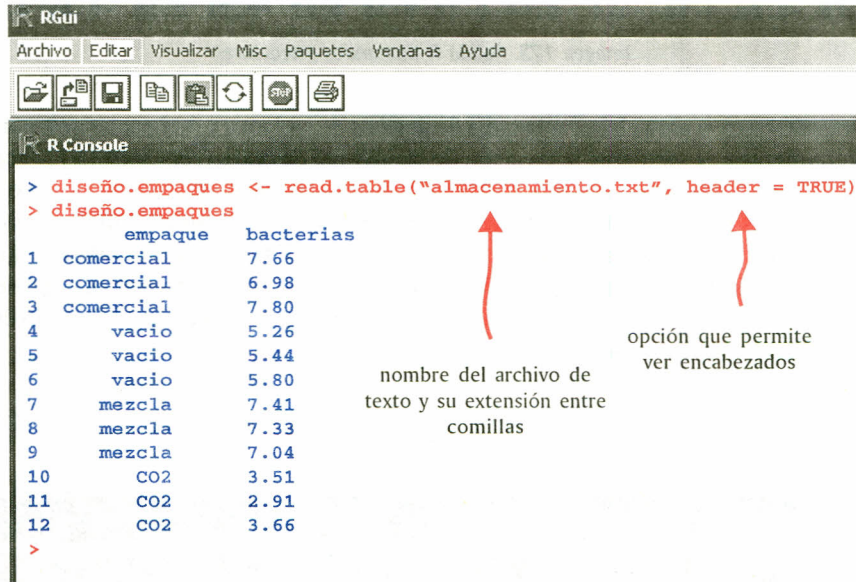


Imagen 174. Creación archivo de texto

Para guardar el archivo de texto, R emplea, por defecto, a **Mis documentos**; para este caso el nombre escogido fue `almacenamiento.txt`.

Ahora, en la consola de R, se debe importar el archivo tal como se muestra en la sección 11.1 importar; como el archivo de texto se encuentra en **Mis documentos** no es necesario cambiar de directorio para importarlo. Para elaborar la tabla con los datos del archivo de texto, que permitirá la elaboración del análisis de varianza y las comparaciones entre tratamientos, se recomienda un nombre que haga alusión a los datos contenidos en el archivo; para el ejemplo, la tabla se denominó `diseño.empaques`, así:



```
> diseño.empaques <- read.table("almacenamiento.txt", header = TRUE)
> diseño.empaques
  empaque bacterias
1 comercial  7.66
2 comercial  6.98
3 comercial  7.80
4 vacio      5.26
5 vacio      5.44
6 vacio      5.80
7 mezcla    7.41
8 mezcla    7.33
9 mezcla    7.04
10 CO2      3.51
11 CO2      2.91
12 CO2      3.66
>
```

nombre del archivo de texto y su extensión entre comillas

opción que permite ver encabezados

Imagen 175. Creación de tabla con datos de archivo de texto

Es posible que se tenga más de una tabla en la consola de R para realizar el análisis de varianza, de manera que se utiliza el comando `attach(nombre de la tabla)`, como se muestra en la imagen 176; además, después del comando `attach`, al digitar el nombre que se asignó a los tratamientos, R presenta el nombre de los diferentes niveles del factor.

```

Archivo  Editar  Visualizar  Misc  Paquetes  Ventanas  Ayuda
[Icons]

R Console
> attach(diseño.empaques)

> diseño.empaques
  empaque  bacterias
1  comercial  7.66
2  comercial  6.98
3  comercial  7.80
4   vacio     5.26
5   vacio     5.44
6   vacio     5.80
7   mezcla   7.41
8   mezcla   7.33
9   mezcla   7.04
10  CO2       3.51
11  CO2       2.91
12  CO2       3.66

> empaque
[1] comercial comercial comercial vacio vacio
[6] vacio mezcla mezcla mezcla CO2
[11] CO2 CO2
Levels: CO2 comercial mezcla vacio

```

Imagen 176. Invocar tabla de datos en la consola de R

Una vez se haya invocado la tabla con los datos por procesar, se crea un objeto con los resultados de la tabla de análisis de varianza mediante el comando `anova(lm(variable respuesta ~ tratamientos))`, donde el comando `lm()` se utilizó para ajustar modelos lineales, en el capítulo 9 (análisis de regresión).

```

RGui
Archivo  Editar  Visualizar  Misc  Paquetes  Ventanas  Ayuda
[Icons]

R Console
> attach(diseño.empaques)
> anova.diseño.empaques = anova(lm(bacterias~empaques))
> anova.diseño.empaques
Analysis of variance Table

Response: bacterias
      Df Sum Sq Mean Sq F value    Pr(>F)
empaques  3  32.873  10.9576   94.584 1.376e-06 ***
Residuals  8   0.927   0.1158
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
>

```

Imagen 177. Tabla de análisis de varianza para el ejemplo de las carnes

Para realizar las comparaciones en pares de todos los tratamientos se tiene el método de Tukey (o la denominada de Diferencia Honestamente Significativa, HSD) y pruebas como la de Student–Newman-Keuls (SNK), la de la Mínima Diferencia Significativa (Least Significant Difference) y la de Intervalos Múltiples de Duncan, entre otras. Para ejecutar en R las pruebas mencionadas, una alternativa es descargar el paquete `agricolae_1.0-9.zip` (Procedimientos Estadísticos para Investigación en Agricultura), de la página <http://cran.r-project.org>, en el que además del archivo `.zip` está el manual de referencia `agricolae.pdf`, que muestra otros procedimientos del diseño experimental.

El comando que se requiere es **nombre de la prueba**(y, “`trt`”, `DFerror`, `MSerror`, `alpha = 0.05`, `group=TRUE`, `main = NULL`), donde:

nombre de la prueba hace referencia a la sintaxis para las pruebas de Duncan, Tukey y Student–Newman-Keuls, la cual corresponde a `duncan.test`, `HSD.test` y `SNK.test`, respectivamente.

`y`, en este argumento se escribe el modelo, ya sea mediante los comandos `aov` o `lm`.

`trt` es el nombre dado a los tratamientos.

`DFerror` hace referencia a los grados de libertad del error.

`MSerror`, valor del cuadrado medio del error.

`Alpha`, valor del nivel de significancia de la prueba; por defecto, R trabaja con 0.05.

`Group`, este argumento requiere de las opciones `FALSE` o `TRUE`. Mediante la opción `FALSE`, R arroja los valores de las diferencias entre cada par de tratamientos, el p-valor y el correspondiente intervalo de confianza; con la opción `TRUE` se asigna letras a los tratamientos, donde tratamientos con la misma letra indican que no hay diferencias entre ellos.

`main` permite asignar un título a los resultados que arroja R; este título se debe escribir entre comillas.

Con los datos del ejemplo de la vida de anaquel de las carnes, una vez se corre la tabla de análisis de varianza, la prueba de Tukey se puede llevar a cabo especificando el modelo, el nombre de los tratamientos, el argumento `group=FALSE`, y un título para los resultados, así:

```

R Console
> HSD.test(lm(bacterias-empaque), "empaque", group=FALSE, main="PRUEBA DE TUKEY")

Study: PRUEBA DE TUKEY

HSD Test for bacterias

Mean Square Error: 0.11585

Empaque, means

      bacterias  std.err replication
CO2           3.36 0.2291288         3
comercial     7.48 0.2532456         3
mezcla        7.26 0.1123981         3
vacio         5.50 0.1587451         3

alpha: 0.05 ; Df Error: 8
Critical Value of Studentized Range: 4.52881

Comparison between treatments means

      Difference  pvalue sig      LCL      UCL
comercial - CO2    4.12 0.000002 ***  3.230038  5.009962
mezcla - CO2      3.90 0.000003 ***  3.010038  4.789962
vacio - CO2       2.14 0.000264 ***  1.250038  3.029962
comercial - mezcla 0.22 0.856362      -0.669962  1.109962
comercial - vacio  1.98 0.000455 ***  1.090038  2.869962
mezcla - vacio    1.76 0.001016 **   0.870038  2.649962

```

Imagen 178. Prueba de Tukey para el ejemplo de las carnes

En el ejemplo no se indicó el valor para alfa, de manera que por defecto empleo el valor 0.05; observando los resultados, al comparar las medias por pares de tratamientos no se presentan diferencias entre tipo de empaque comercial y mezcla de gases, pues para la diferencia entre estos tratamientos el p-valor es de 0.8564.

De manera similar se procede para las pruebas de Duncan y SNK.

Cuando la prueba por emplear es la de Mínima Diferencia Significativa, el comando es:

```

LSD.test(y, trt, DFerror, MSerror, alpha = 0.05, p.adj=c("none", "holm", "hochberg",
«bonferroni», "BH", "BY", «fdr»), group=TRUE, main = NULL)

```

Es de notar que a diferencia de las pruebas anteriores se inserta un nuevo argumento denominado **p.adj**, que indica el método para ajustar el p-valor; al especificar la opción "none", R arroja el valor empleando la distribución t-Student.

```

R Console
> LSD.test(lm(bacterias-empaque), "empaque", group=FALSE, p.adj=c("none"),
+ main = "PRUEBA DE MÍNIMA DIFERENCIA SIGNIFICATIVA")

Study: PRUEBA DE MÍNIMA DIFERENCIA SIGNIFICATIVA

LSD t Test for bacterias

Mean Square Error: 0.11585

empaque, means and individual ( 95 % ) CI

      bacterias  std.err  replication  LCL  UCL
CO2          3.36  0.2291288         3  2.831628  3.888372
comercial    7.48  0.2532456         3  6.896015  8.063985
mezcla       7.26  0.1123981         3  7.000810  7.519190
vacio        5.50  0.1587451         3  5.133933  5.866067

alpha: 0.05 ; Df Error: 8
Critical Value of t: 2.306004

Comparison between treatments means

      Difference  pvalue  sig  LCL  UCL
comercial - CO2      4.12  0.000000  ***  3.479141  4.760859
mezcla - CO2         3.90  0.000001  ***  3.259141  4.540859
vacio - CO2          2.14  0.000057  ***  1.499141  2.780859
comercial - mezcla    0.22  0.451410         -0.420859  0.860859
comercial - vacio     1.98  0.000100  ***  1.339141  2.620859
mezcla - vacio        1.76  0.000225  ***  1.119141  2.400859

```

Imagen 179. Prueba LSD para el ejemplo de las carnes

Mediante esta prueba se observa que con nivel de significancia igual a 0.05 no hay diferencias nuevamente entre los tratamientos comercial y mezcla de gases.

10.3 EJERCICIOS

10.3.1 Suponga que se tiene un diseño de experimentos desbalanceados con cinco niveles de clasificación, llamados a, b, c, d y e, de tamaños 8, 12, 10, 15 y 9, respectivamente; construya un vector que sirva para la clasificación de los datos.

10.3.2 La Gulupa es una fruta de gran acogida debido a su aporte nutricional; en un estudio anterior se estableció que en el estado cero (verde medio a verde oscuro pálido), la fruta alcanza su máximo peso total; con el fin de verificar este antecedente se construyó y aplicó un diseño de experimentos que arrojó los siguientes datos.

ESTADO DE MADURACIÓN	REPETICIÓN	PESO (g)
CERO	1	60,2
	2	63,4
	3	44,8
UNO	1	56
	2	57,4
	3	41,4
DOS	1	55,4
	2	43
	3	50,5
TRES	1	59
	2	45
	3	58
CUATRO	1	55
	2	46,3
	3	50,8
CINCO	1	62,3
	2	56,3
	3	47,8

Con esta información construya en R el diseño de experimentos y determine si existen o no evidencias estadísticas para pensar que el peso en los diferentes estados de maduración de la fruta difiere significativamente.

11. IMPORTAR Y EXPORTAR DATOS EN R

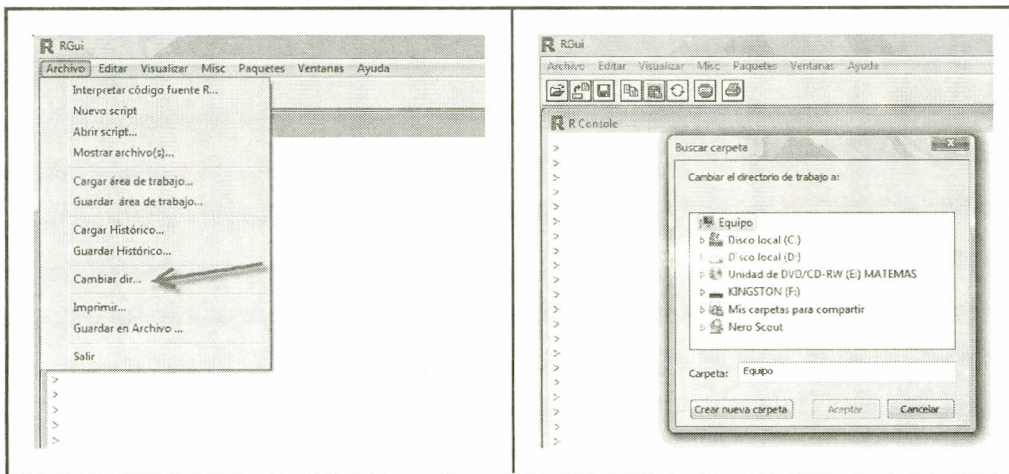
11.1 IMPORTAR

Cuando se cuenta con bases de datos, suelen estar construidas en programas diferentes a R, por lo que se hace necesario que sean importadas desde archivos predeterminados. La lectura de archivos en R usa comandos sencillos que requieren aspectos específicos y bastante estrictos. La manera más fácil de importar datos de un programa determinado es pasarlos a un block de notas y desde allí importarlos mediante el comando:

```
read.table("file", header = FALSE,...)
```

Algunos de los argumentos utilizados en el comando anterior son: **file**, que define el nombre y la extensión del archivo que se va a importar; por ejemplo si se el archivo proviene de un block de notas la extensión es txt, y **header=**, argumento lógico que indica si el nombre de las variables está presente en la primera línea del archivo. Para mayor información de los argumentos que pueden ser utilizados en el comando `read.table()` consulte la ayuda interactiva de R.

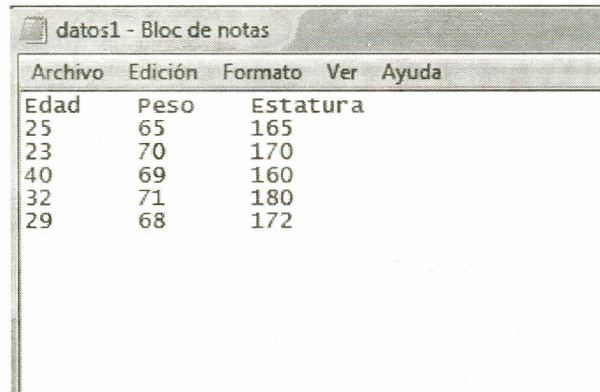
Para que el comando anterior lea los archivos correctamente es necesario que previamente se cambie el directorio en la barra de herramientas, es decir, se direcciona la búsqueda del archivo.



Imágenes 180 y 181. Cambiando el directorio en R

Cuando se importan datos mediante este comando, son almacenados en una hoja de datos (data frame), y, de ser necesario, sus atributos pueden ser cambiados mediante algunos comandos: `as.vector()`, `as.matrix()`, `as.factor()`,...

Ejemplo: Se tiene un conjunto de datos en un block de notas con el nombre `datos1`, el cual contiene información de cinco individuos:



Edad	Peso	Estatura
25	65	165
23	70	170
40	69	160
32	71	180
29	68	172

Imagen 182. Datos en block de notas

Al utilizar el conjunto de datos anterior la asignación se realiza así:

```
>
> datos = read.table("datos1.txt",header=TRUE)
> datos
  Edad Peso Estatura
1   25   65     165
2   23   70     170
3   40   69     160
4   32   71     180
5   29   68     172
```

Imagen 183. Salida R importando datos

11.2 EXPORTAR

Cuando se desea exportar datos a otros programas se debe cambiar el directorio al lugar del PC en el cual se quiere almacenar la información; el comando utilizado para exportar datos desde R es:

```
write.table(x, file = "", append = FALSE, quote = TRUE, sep = " ",
           eol = "\n", na = "NA", dec = ".", row.names = TRUE,
           col.names = TRUE, qmethod = c("escape", "double"))
```

Los argumentos utilizados en el comando anterior son: **x** es el objeto por exportar, preferiblemente una matriz o un data frame; **file** representa el nombre del archivo con su respectiva extensión, correspondiente al programa al cual se exporta (en el caso de exportar a Excel la extensión es xls). Para mayor información con respecto a los demás parámetros consulte la ayuda interactiva.

Ejemplo:

```
>
> # variables
> id = c(1,2,3,4,5,6)
> edad = c( 15, 35, 26, 45, 23, 45)
> genero = c("h","m","h","m","m","h")
> # Hoja de datos
> datos = data.frame(id,edad,genero)
> datos
  id edad genero
1  1  15      h
2  2  35      m
3  3  26      h
4  4  45      m
5  5  23      m
6  6  45      h
>
> # Exportar datos
> write.table(datos,file="datos.xls")
```

Imagen 184. Salida R exportando datos

BIBLIOGRAFÍA CITADA

Demétrio, C. G. B. (2001). *Modelos lineares generalizados em experimentação agrônômica*. Piracicaba, Brasil. [En línea] Disponible en <http://www.lce.esalq.usp.br/clarice/Apostila.pdf> [Recuperado el 30-08-2010].

Díaz Monroy, Luis Guillermo (2002). *Estadística Multivariada: Inferencia y Métodos*. Facultad de Ciencias, Universidad Nacional de Colombia, Bogotá.

Ihaka R. & Gentleman R. (1996). R: A Language for Data Analysis and Graphics. *Journal of Computational and Graphical Statistics* 5: 299–314.

Kuehl, Robert O. (2001). *Diseño de experimentos*, 2ª edición. México: Thomson, 661 p.

Peña, Daniel. (2002). *Análisis de datos multivariantes*. España: McGraw-Hill.

R Development Core Team (2008). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL <http://www.R-project.org>.

BIBLIOGRAFÍA RECOMENDADA

Paradis Emmanuel. R para Principiantes. Institut des Sciences de l'Évolution. Universit Montpellier II. F-34095 Montpellier cdex 05 France.

García, J. E.; Bacherro, J. M. (2005). *Estadística Descriptiva y nociones de probabilidad*. España: Universidad de Valencia, Editorial Thomsom Editores.

Richard A. Becker; John M. Chambers and Allan R. Wilks (1988). The New S Language. Chapman & Hall, New York. This book is often called the "Blue Book".

Venables, W. N.; Smith, D. M. (2007). *The R Development Core Team Notes on R: A Programming Environment for Data Analysis and Graphics Version 2.6.1*.

Este libro se terminó de imprimir en Imprenta
y Publicaciones de la Universidad Pedagógica
y Tecnológica de Colombia, en diciembre de
2010, con una edición de 300 ejemplares.
Tunja, Boyacá, Colombia