INFERENCIA Y MODELOS ESTADÍSTICOS

Jacqueline Köhler C. y José Luis Jara V.

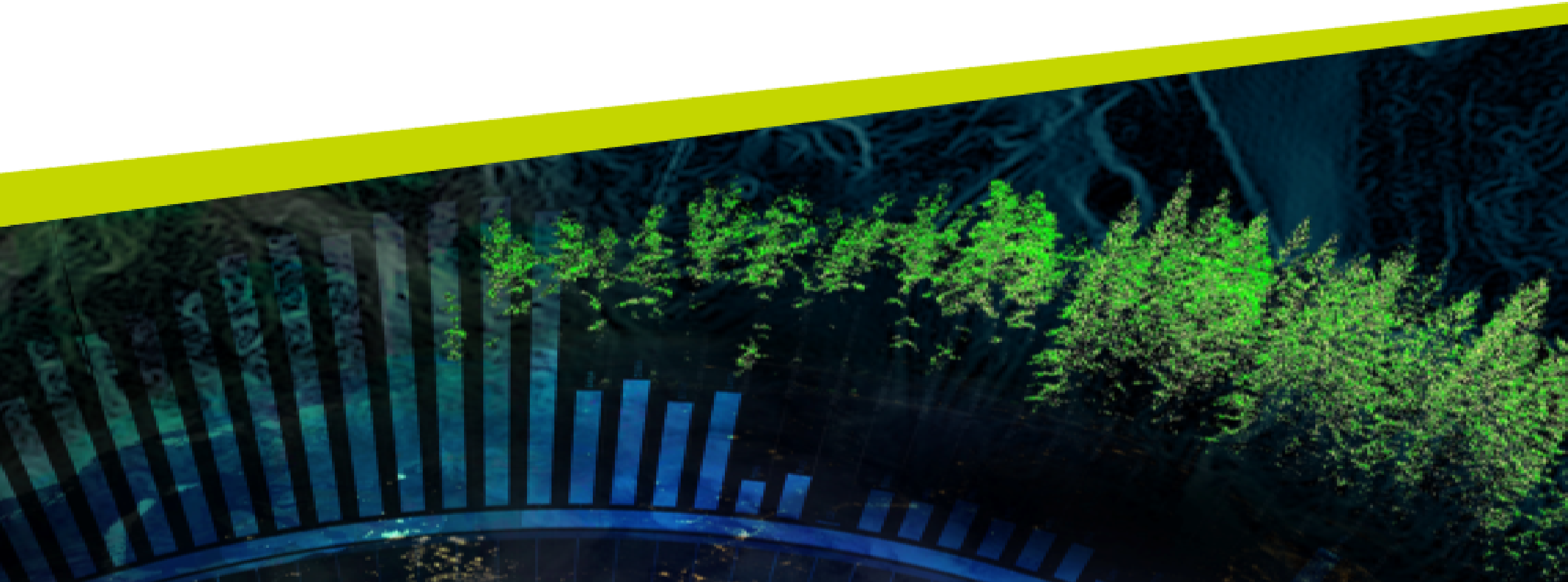


Tabla de contenido

Índice de tablas

Índice de figuras

Índice de scripts

CAPÍTULO 1. INTRODUCCIÓN

Este libro tiene como propósito acompañarte en el aprendizaje de las primeras nociones de inferencia estadística y de creación de modelos estadísticos. En este primer capítulo comenzaremos por buscar definiciones iniciales para los conceptos de inferencia y modelo, para luego abordar algunas nociones iniciales acerca de los datos empleados en estadística y algunas herramientas para que puedas empezar a usar el entorno de programación R, con el cual trabajaremos a lo largo de todo el texto. Te sugerimos, entonces, que lo instales junto con el entorno de desarrollo integrado RStudio.

1.1 INFERENCIA Y MODELOS ESTADÍSTICOS

La Real Academia Española (2014) define **inferencia** como "acción y efecto de inferir". Esto por sí solo no nos dice mucho, pero si buscamos también la definición de **inferir**, encontraremos que significa "deducir algo o sacarlo como conclusión de otra cosa". A partir de estas definiciones, y de acuerdo con Devore (2008, p. 5), podemos decir que la **estadística inferencial** es una rama de la estadística que busca obtener una conclusión para un conjunto de individuos o elementos (denominado **población**) a partir de información recolectada de un subconjunto de éste (llamado **muestra**).

Llegar a una definición de **modelo estadístico** puede ser bastante más complejo. Como nos muestra la figura 1.1, ¡un modelo puede ser muchas cosas diferentes! Veamos qué nos dice la Real Academia Española (2014):

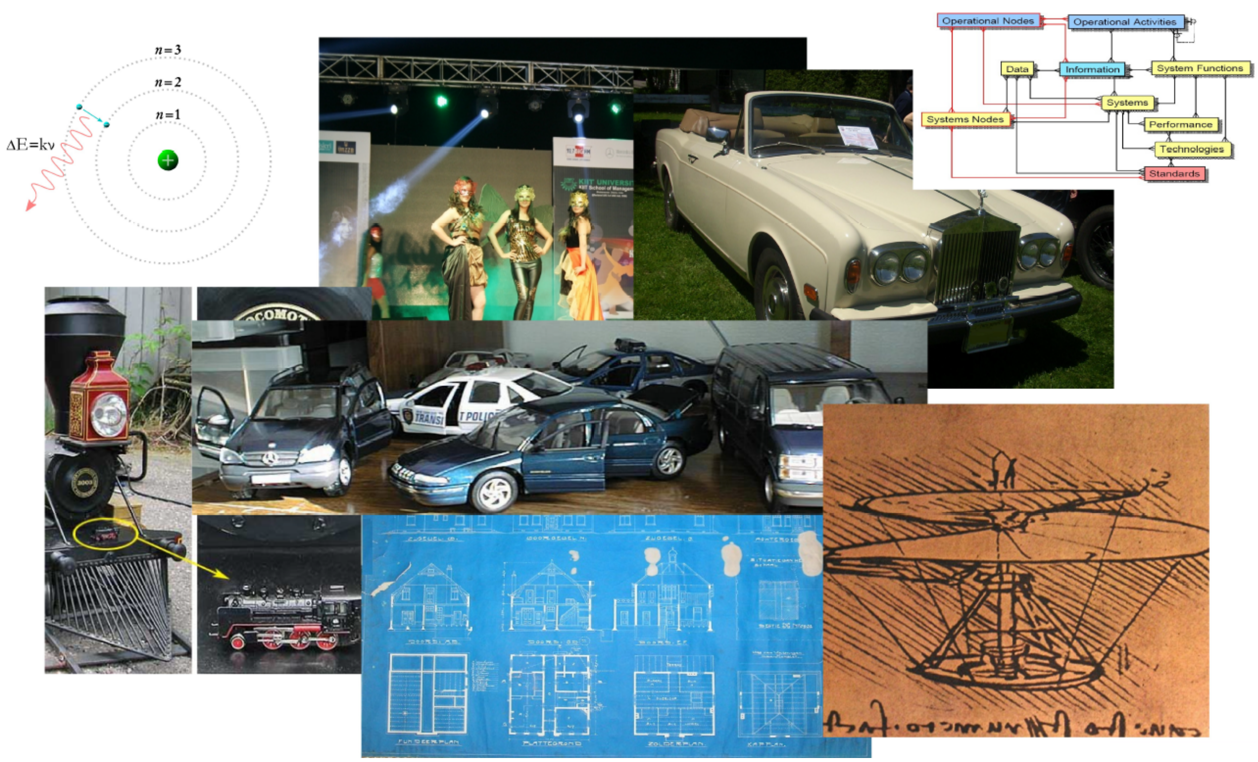


Figura 1.1: ejemplos de modelos.

1. Arquetipo o punto de referencia para imitarlo o reproducirlo.
2. En las obras de ingenio y en las acciones morales, ejemplar que por su perfección se debe seguir e imitar.
3. Representación en pequeño de alguna cosa.
4. Esquema teórico, generalmente en forma matemática, de un sistema o de una realidad compleja, como la evolución económica de un país, que se elabora para facilitar su comprensión y el estudio de su comportamiento.
5. Objeto, aparato, construcción, etc., o conjunto de ellos realizados con arreglo a un mismo diseño. *Auto modelo 1976. Lavadora último modelo.*
6. Vestido con características únicas, creado por determinado modista, y, en general, cualquier prenda de vestir que esté de moda.
7. En empresas, u. en aposición para indicar que lo designado por el nombre anterior ha sido creado como ejemplar o se considera que puede serlo. *Empresa modelo. Granjas modelo.*
8. Esc. Figura de barro, yeso o cera, que se ha de reproducir en madera, mármol o metal.
9. *Cuba* impreso (||hoja con espacios en blanco).
10. Persona que se ocupa de exhibir diseños de moda.
11. Persona u objeto que copia el artista.

Puede ser de ayuda tener en cuenta algunas definiciones e ideas que nos ofrece la literatura. Kaplan (2009), por ejemplo, señala que un modelo es una representación con un propósito particular. Pero otros contribuyen a enriquecer esta definición:

- Representación simplificada de la realidad en la que aparecen algunas de sus propiedades (Joly, 1988).
- Permiten estudiar de forma simple y comprensible una porción de la realidad (Ríos, 1995).
- Dejan cosas fuera y pueden llevar a conclusiones equivocadas (Kaplan, 2009).
- Resumen de manera conveniente, a juicio de los sus creadores, los aspectos más relevantes del fenómeno estudiado y sus relaciones (Mendez Ramírez, 1998).
- Están mediados por el diseño, es decir la forma de seleccionar y observar (o manipular) la realidad modelada (Mendez Ramírez, 2012).
- Son pequeños, económicos, seguros y fáciles de transportar, copiar y modificar (Kaplan, 2009).

Si a esta concepción del significado de la palabra modelo le agregamos nuevos conceptos fundamentales que nos ofrecen otros autores, podemos acercarnos un poco más a la idea de **modelo estadístico**:

- Modelo matemático de la regularidad estadística de los posibles resultados de la evolución de un fenómeno aleatorio (Mendez Ramírez, 1998).
- Distribución de probabilidad construida para poder hacer inferencias o tomar decisiones desde datos (Freedman, 2009).
- Descripción simple de un proceso probabilístico que puede haber dado origen a un conjunto de datos observados (McCullagh, 2002).
- Modelo estocástico que contiene parámetros desconocidos que deben ser estimados en base a suposiciones acerca del modelo y los datos (SAS Institute Inc., 2008).

Pero, ¿para qué sirven los modelos estadísticos? Diversos autores nos muestran que tales modelos son muy útiles en diversos contextos:

- Para describir (Kaplan, 2009) o resumir datos (Freedman, 2009).
- Para clasificar (Kaplan, 2009) o predecir (Freedman, 2009; Kaplan, 2009).
- Para anticipar el resultado de intervenciones (Freedman, 2009; Kaplan, 2009).

Ahora que hemos definido nuestros conceptos iniciales, veamos algunas definiciones y herramientas que nos permitan comenzar a explorarlos.

1.2 VARIABLES, PARÁMETROS Y ESTADÍSTICOS

Comencemos a partir de un ejemplo para ilustrar algunas ideas iniciales acerca de los datos. La tabla 1.1 muestra las primeras filas de la matriz de datos `ffaa`[1], que almacena información acerca de miembros activos de las Fuerzas Armadas de Chile. Cada columna de la matriz representa una **variable** o característica, mientras que cada fila corresponde a una **unidad de observación** o instancia. Así, cada fila de la tabla 1.1 almacena datos de una misma persona. El uso de **matrices de datos** para almacenar los datos es muy conveniente, pues nos ayuda a acceder a los datos y modificarlos más fácilmente. Por ejemplo, para agregar una nueva observación, basta con añadir una nueva fila a la matriz. Si queremos eliminar una característica, simplemente borramos la columna correspondiente. Para consultar una característica en particular de una observación, solo necesitamos conocer la fila y la columna correspondientes.

id	género	estatura	escalafón	servicio	antigüedad	rama
1	M	1,77	S	89,91	15	E
2	M	1,97	O	65,14	30	C
3	F	1,65	O	97,03	12	A
4	M	1,82	S	76,29	9	A
5	F	1,73	S	69,46	7	M
6	M	1,78	S	97,67	21	E
7	M	1,87	O	72,09	27	C
8	F	1,91	S	94,53	11	A
⋮	⋮	⋮	⋮	⋮	⋮	⋮
6051	M	1.72	S	86.48	17	E

Tabla 1.1: algunas filas de la matriz de datos `ffaa`.

Antes de comenzar a trabajar con los datos, tenemos que estar seguros de comprender cada uno de sus aspectos. Para ello, las siguientes preguntas pueden ser un excelente punto de partida:

- ¿A qué corresponde cada característica?
- ¿Cuáles son sus unidades de medición?
- ¿Qué valores puede tomar?

La tabla 1.2 muestra la descripción de las características presentes en la matriz de datos de la tabla 1.1. En ella se explica el significado de cada columna de la matriz junto a su rango de valores o a un listado de valores posibles.

Variable	Descripción
id	Identificador de la observación.
género	Género del estudiante (M: masculino, F: femenino).
estatura	Estatura (m).
escalafón	Escalafón al que pertenece (O: oficial, S: suboficial).
servicio	Evaluación de servicio (entre 0 y 100).
antigüedad	Años que lleva en servicio activo.
rama	Rama de las FF.AA. A la que pertenece (E: Ejército, M: Marina, A: Fuerza Aérea, C: Carabineros).

Tabla 1.2: descripción de las variables para el conjunto de datos `ffaa`.

Si estudias la tabla 1.2 con detalle, podemos notar diferencias interesantes entre las variables, más allá de su descripción. Por ejemplo, no todas ellas pueden tomar los mismos valores. Con esto aparece la noción de **tipos de variables**, los cuales podemos jerarquizar:

[1] Los datos aquí presentados son ficticios y han sido creados únicamente con fines pedagógicos.

- **Numéricas**: pueden tomar muchos valores numéricos, y son sensibles a operaciones aritméticas. Pueden separarse en:
  - **Continuas**: pueden tomar cualquier valor (en un intervalo) del conjunto de los reales. Por ejemplo, las variables `estatura` y `servicio` descritas en la tabla 1.2.
  - **Discretas**: no es posible que tomen cualquier valor (en un intervalo). Por ejemplo, podrían tomar únicamente valores enteros no negativos, como la variable `antigüedad` de la matriz de datos `ffaa`.
- **Categóricas**: solo pueden tomar un valor de entre un conjunto acotado. Cada posible valor se denomina **nivel**. Entre las variables categóricas es posible distinguir variables:
  - **Nominales**: no existe un orden natural entre los niveles. Ejemplos de variables nominales son `género` y `rama` de la matriz de datos `ffaa`.
  - **Ordinales**: existe un orden natural entre los niveles. Por ejemplo, la idea de jerarquía es evidente al distinguir entre oficiales y suboficiales en la variable `escalafón` de la tabla 1.1.

Tener diferentes tipos de variables significa que debemos medirlas con distintas clases de escalas, las cuales se distinguen por sus propiedades y los tipos de operaciones que permiten:

- **Escala nominal**: sirve solo para separar un conjunto de elementos en subclases excluyentes entre sí. Los valores no son más que nombres o estados, por lo que no podemos hacer operaciones aritméticas ni podemos establecer relaciones de orden.
- **Escala ordinal o de rangos**: esta escala, al igual que la nominal, permite separar un conjunto de elementos en subclases excluyentes entre sí. Una vez más, los valores son solo nombres o estados, por lo que tampoco podemos hacer operaciones aritméticas. Pero en este caso sí podemos establecer una relación de orden, aunque para ello es necesario que la variable tenga a lo menos tres niveles. A modo de ejemplo, si queremos una variable para medir el nivel de estudios de las personas en un grupo demográfico, podríamos considerar una escala ordinal con los niveles "ninguna", "básica completa", "media", "superior" y "postgrado". Aquí podemos apreciar claramente que los niveles están ordenados de manera creciente.
- **Escala de intervalo**: sirve para datos continuos o discretos con una gran cantidad de niveles. Además de la noción de orden de la escala ordinal, se cumple que la distancia entre dos valores cualesquiera de la escala es conocida y constante, por lo que podemos emplear operaciones aritméticas. Aunque el punto cero y la unidad de medida son arbitrarios, la razón entre dos intervalos es independiente de ambos elementos. Tomemos, por ejemplo, la escala Celsius de temperatura. El cero está dado por el punto de congelación del agua. La medida o tamaño se calcula en base a los puntos de congelación y ebullición del agua. Sin embargo, a pesar de estos parámetros arbitrarios, el cambio en la cantidad de calor es el mismo si aumentamos la temperatura de 10 a 15 grados Celsius, o de 25 a 30. Si miramos ahora la escala Fahrenheit de temperatura, los puntos fijos son diferentes a los empleados por la escala Celsius, por lo que el cero no significa lo mismo. Sin embargo, existe una transformación lineal que nos permite transformar una medida en una escala a su equivalente en otra escala.
- **Escala de razón**: cumple con todos los atributos de la escala de intervalos, pero además tiene su origen en un cero verdadero. Ejemplos de tales escalas son, por ejemplo, las que permiten medir la masa o la distancia. En una escala de razón, la diferencia entre dos puntos es independiente de la unidad de medida. Por ejemplo, si medimos la masa de dos objetos, la razón es constante independientemente de si empleamos kilogramos, libras u onzas (a diferencia de lo que ocurre con la temperatura usando las escalas Celsius y Fahrenheit).

La estadística usa los datos para responder diversas preguntas, muchas de las cuales se orientan a encontrar relaciones entre variables. Así, dos variables pueden ser:

1. **Independientes**: no existe asociación o relación entre las variables.
2. **Dependientes**: existe una asociación o relación entre las variables. Puede existir:
   - **Asociación positiva**: si una variable crece, la otra también lo hace.
   - **Asociación negativa**: si una variable crece, la otra decrece.

En el contexto de la estadística, decimos que un **parámetro** es cualquier número que describa una población en forma resumida, como por ejemplo la media poblacional. A su vez, un **estadístico** es "cualquier cantidad cuyo valor puede ser calculado a partir de datos muestrales" Devore (2008, p. 204), como por ejemplo la media, la mediana o la desviación estándar de un conjunto de datos observados. Si bien a primera vista ambos conceptos parecen similares, en realidad existe una diferencia importante entre ellos: el parámetro describe una población, mientras que el estadístico, al ser calculado a partir de una muestra, no es más que una **estimación puntual** del parámetro.

Si necesitas más ejemplos o quieres complementar lo aprendido, puedes consultar los textos de referencia para esta sección. Diez, Barr y Çetinkaya-Rundel (2017, pp. 9-19) describe los principales conceptos relativos a datos, tipos de variables y relaciones entre variables. En Dagnino (2014) puedes aprender más sobre escalas de medición.

1.3 CONOCIENDO R

R es un ambiente de software gratuito para estadística computacional y elaboración de gráficos. En esta sección conoceremos algunas herramientas que nos ayudarán a lo largo de este libro. Desde luego, estas breves páginas no pretenden ser un tutorial completo del lenguaje, sino más bien un punto de partida para que podamos aplicar los contenidos que aquí se abordan. Como ya señalamos, sugerimos el uso del entorno integrado de desarrollo RStudio, cuya documentación e instrucciones de instalación podemos consultar en RStudio (2021). En The R Foundation (s.f.) y Carchedi, De Mesmaeker y Vannoorenberghe (s.f.) podemos encontrar documentación acerca del lenguaje R y sus paquetes.

1.3.1 Importación de datos

Una de las primeras cosas que necesitamos conocer es cómo importar o cargar una matriz de datos (denominada *data frame* en R) desde un archivo de texto plano (.txt) o de valores separados por coma (.csv). Para lograrlo con éxito, debemos tener en cuenta algunas orientaciones para preparar los datos adecuadamente:

- La primera fila se usa para los nombres de las columnas o variables.
- La primera columna contiene los nombres de las observaciones, que deben ser únicos.
- Los nombres de las columnas deben respetar las convenciones de R:
  - No está permitido el uso de espacios ni símbolos especiales (?, $, *, +, #, (,), -, /, }, {, |, >, <, etc.). Solo se admite el uso de puntos (.) y guiones bajos (_).
  - Los nombres de variables no pueden comenzar con un dígito.
  - Los nombres de las columnas deben ser únicos.
- R es sensible a las mayúsculas.
- No puede haber filas en blanco.
- No debe tener comentarios.
- Los valores faltantes deben ser denotados mediante `NA`.
- Para columnas con fechas, se usa el formato mm/dd/aaaa.
- El archivo debe tener tener uno de los siguientes formatos, ejemplificados en la figura 1.2:
  - Extensión .txt con tabulaciones como delimitador y punto decimal para valores flotantes.
  - Extensión .csv en formato inglés, con comas (`,`) como delimitador y punto decimal para valores flotantes.
  - Extensión .csv en formato español, con punto y comas (`;`) como delimitador y coma decimal para valores flotantes.

El script 1.1 muestra las diferentes funciones para importar datos en R, donde las líneas que comienzan por `#` corresponden a comentarios. La línea 2 carga el conjunto de datos `mtcars`, disponible en R, mientras que las líneas 5, 9 y 16 importan datos desde archivos. Tanto `read.delim()` como `read.csv()` y `read.csv2()` se usan

```
id	género	estatura	escalafón	servicio	antigüedad	rama
1	M	1.77	S	89.91	15	E
2	M	1.97	O	65.14	30	C
3	F	1.65	O	97.03	12	A
4	M	1.82	S	76.29	9	A
5	F	1.73	S	69.46	7	M
6	M	1.78	S	97.67	21	E
7	M	1.87	O	72.09	27	C
8	F	1.91	S	94.53	11	A
```

(a) Texto plano delimitado por tabulaciones.

```
id,género,estatura,escalafón,servicio,antigüedad,rama
1,M,1.77,S,89.91,15,E
2,M,1.97,O,65.14,30,C
3,F,1.65,O,97.03,12,A
4,M,1.82,S,76.29,9,A
5,F,1.73,S,69.46,7,M
6,M,1.78,S,97.67,21,E
7,M,1.87,O,72.09,27,C
8,F,1.91,S,94.53,11,A
```

(b) Valores separados por comas (inglés).

```
id;género;estatura;escalafón;servicio;antigüedad;rama
1;M;1,77;S;89,91;15;E
2;M;1,97;O;65,14;30;C
3;F;1,65;O;97,03;12;A
4;M;1,82;S;76,29;9;A
5;F;1,73;S;69,46;7;M
6;M;1,78;S;97,67;21;E
7;M;1,87;O;72,09;27;C
8;F;1,91;S;94,53;11;A
```

(c) Valores separados por punto y comas (español).

Figura 1.2: formatos de archivo para importar datos en R.

de la misma forma, pudiendo recibir como argumento una llamada al selector de archivos (`file.choose()`), como en la línea 5, o la ruta completa para el archivo, como en la línea 9. En el caso de la línea 16, basta con proporcionar el nombre de archivo pues la función `setwd()` (línea 12) permite establecer el directorio de trabajo de R para la sesión. Las funciones `head()` y `tail()` (líneas 20 y 24) proporcionan una buena manera de inspeccionar los datos cargados, pues muestran por consola las primeras y últimas filas de la matriz de datos, respectivamente.

Script 1.1: sentencias para importar un conjunto de datos.

```r
1 # Cargar un conjunto de datos disponible en R.
2 datos1 <- mtcars
3
4 # Importar desde un archivo de texto plano delimitado por tabuladores.
5 datos2 <- read.delim(file.choose())
6
7 # Importar desde un archivo de valores separados por coma
8 # en formato inglés (figura 1.2 b).
9 datos3 <- read.csv("C:\\Inferencia\\ejemplo1-csv-eng.csv")
10
11 # Configurar carpeta de trabajo
12 setwd("C:\\Inferencia")
13
14 # Importar desde un archivo de valores separados por coma
15 # en formato español (figura 1.2 c).
16 datos4 <- read.csv2("ejemplo1-csv-esp.csv")
17
18 # Mostrar las primeras 6 filas del conjunto de datos
19 # almacenado en datos1.
20 head(datos1)
21
22 # Mostrar las últimas 6 filas del conjunto de datos
23 # almacenado en datos1.
24 tail(datos1)
```

1.3.2 Importación de paquetes

Si bien el entorno R básico incluye muchísimas funcionalidades, existe una enorme variedad de paquetes o colecciones que incorporan otras nuevas o mejoran las ya existentes.

Antes de usar un paquete por primera vez tenemos que instalarlo. Para ello, podemos usar la sentencia que se muestra en la línea 2 del script 1.2. Debemos tener en cuenta que la función `install.packages()` requiere que el nombre del paquete se escriba entre comillas.

Para poder usar un paquete, existen las sentencias `library()` (línea 5 del script 1.2) y `require()` (línea 8), que reciben como argumento el nombre del paquete (sin comillas). Si bien ambas sentencias pueden usarse indistintamente, se diferencian en que `library()` termina la ejecución con un mensaje de error si el paquete no está instalado, mientras que `require()` solo emite una advertencia.

Una forma elegante de evitar errores es verificar si un paquete se encuentra instalado antes de usarlo, para lo que podemos usar una combinación de las sentencias anteriores, como muestran las líneas 11 a 14 del script 1.2. Cabe destacar que la opción `dependencies = TRUE` en la línea 12 asegura que se instalen además aquellos paquetes que son requeridos por el que se desea instalar. Fijémonos que el lenguaje de programación R usa **argumentos con nombre**.

Script 1.2: instalar y cargar paquetes de R.

```
1  # Instalar un paquete.
2  install.packages("ggpubr")
3
4  # Primera forma de importar un paquete.
5  library(ggpubr)
6
7  # Segunda forma de importar un paquete.
8  require(ggplot2)
9
10 # Importar un paquete, instalándolo de ser necesario.
11 if(!require(dplyr)){
12   install.packages("dplyr", dependencies=TRUE)
13   require(dplyr)
14 }
```

1.3.3 Construcción de una matriz de datos

Consideremos la idea de construir una matriz de datos que contenga el nombre, la fecha de nacimiento y las calificaciones de los estudiantes en las tres evaluaciones de una asignatura. El script 1.3 crea esta matriz de datos en R con tres observaciones. En las líneas 2 a 4 crea un vector de strings con los nombres de los estudiantes y lo almacena en la variable `nombre`. De manera similar, en la línea 8 crea un vector de fechas. Debemos notar que para ello construye un vector de tres strings con las fechas en formato aaaa-mm-dd, el cual es entregado como argumento a la función `as.Date()` para que sean convertidos al formato de fecha. Las líneas 12 a 14 crean tres vectores de flotantes para las calificaciones obtenidas por los estudiantes. Hasta este punto, solo se tienen muchas variables con vectores de largo 3, los cuales deben ser combinados para formar una matriz de datos donde cada vector sea una columna. La función `data.frame()`, en las líneas 18 a 22, realiza esta tarea. Dicha función recibe como argumentos tantos vectores como variables tenga el conjunto de datos, y toma los nombres de las variables que los contienen como nombres de las columnas. Cabe destacar que, en la línea 23, `data.frame()` recibe un argumento adicional, el booleano `stringsAsFactors`, con valor falso. Esto se debe a que, si no se entrega este parámetro, R asume que su valor por defecto es verdadero, por lo que interpreta el vector de strings como una variable categórica y asigna un valor numérico a cada nivel.

La última línea del script 1.3 permite guardar la matriz de datos en un archivo de valores separados por comas (formato español). La función `write.csv2()` recibe como argumentos el nombre de la variable que contiene la matriz de datos y una cadena de caracteres con el nombre del archivo. El argumento `row.names = FALSE` indica que no deseamos guardar los nombres de las filas. Si queremos guardar nuestra matriz de datos en un archivo separado por comas en formato inglés, podemos hacerlo mediante la función `write.csv()`, que funciona del mismo modo que `write.csv2()`.

Script 1.3: construir un dataframe.

```
# Crear un vector de strings y guardarlo en la variable nombre.
nombre <- c("Alan Brito Delgado",
            "Zacarías Labarca del Río",
            "Elsa Payo Maduro")

# Crear un vector de fechas y guardarlo en la variable
# fecha_nacimiento.
fecha_nacimiento <- as.Date(c("2008-1-25", "2006-10-4", "2008-3-27"))

# Crear tres vectores de reales entre 1.0 y 7.0 y guardarlos
# en prueba_i, respectivamente.
prueba_1 <- c(5.5, 3.4, 4.5)
prueba_2 <- c(3.2, 4.7, 4.1)
prueba_3 <- c(4.8, 4.3, 5.1)

# Construir un data frame a partir de los vectores anteriores y
# guardarlo en la variable dataframe.
dataframe <- data.frame(nombre,
                        fecha_nacimiento,
                        prueba_1,
                        prueba_2,
                        prueba_3,
                        stringsAsFactors = FALSE)

# Guardar un dataframe en un archivo csv (formato español).
write.csv2(dataframe, "C:/Inferencia/Ejemplo.csv", row.names = FALSE)
```

1.3.4 Modificación de una matriz de datos

Muchas veces tendremos la necesidad de modificar la matriz de datos. Algunas tareas, como agregar o quitar una columna o un observación pueden hacerse de manera bastante sencilla, como ilustra el script 1.4.

Script 1.4: modificaciones sencillas de una matriz de datos.

```
# Leer un dataframe desde archivo csv.
datos <- read.csv2("C:/Inferencia/Ejemplo.csv", stringsAsFactors = FALSE)

# Eliminar del data frame la columna fecha_nacimiento.
dataframe$fecha_nacimiento <- NULL

# Agregar al data frame la columna edad.
dataframe$edad <- c(23, 25, 23)

```

```r
# Crear una nueva observación.
nueva <- data.frame(nombre="Elba Calao del Río",
                    prueba_1 = 6.4,
                    prueba_2 = 2.3,
                    prueba_3 = 4.6,
                    edad = 24)

# Agregar la nueva observación al data frame.
dataframe <- rbind(dataframe, nueva)

# Eliminar las primeras 3 observaciones del data frame.
dataframe <- dataframe[-c(1:3),]

# Guardar el dataframe en un archivo csv .
write.csv2(dataframe, "C:/Inferencia/Ejemplo_mod.csv", row.names = FALSE)
```

Sin embargo, también podemos vernos en la necesidad de realizar transformaciones más complejas. El paquete `dplyr` ofrece un conjunto de funciones que simplifica esta tarea:

- `filter()`: selecciona instancias (filas) de acuerdo a su valor.
- `arrange()`: modifica el orden de las filas.
- `select()`: permite seleccionar variables (características) por sus nombres, a la vez que las reordena.
- `mutate()`: permite agregar nuevas variables que se obtienen como funciones de otras ya existentes.

Para mostrar el uso de estas funciones (script 1.5) usaremos el conjunto de datos `iris`, disponible en R. Este contiene 150 observaciones pertenecientes a tres especies de una flor llamada iris: setosa, versicolor y virginica, para las cuales se registran el largo y ancho de sus sépalos y de sus pétalos (en centímetros). Puedes consultar otras funciones y ejemplos más detallados en Müller (2021) y Wickham y Grolemund (2017, cap. 5).

Script 1.5: modificación de una matriz de datos con el paquete `dplyr`.

```r
library(dplyr)

# Cargar dataframe iris incluido en R.
datos <- iris

# Seleccionar observaciones correspondientes a la especie versicolor.
versicolor <- datos %>% filter(Species == "versicolor")

# Seleccionar observaciones de la especie versicolor cuyos sépalos tengan una
# longitud igual o superior a 6 cm.
largas <- datos %>% filter(Species == "versicolor" & Sepal.Length >= 6)

# Seleccionar la especie y variables relativas a los pétalos.
petalos <- datos %>% select(Species, starts_with("Petal"))

# Seleccionar variables de ancho y la especie.
anchos <- datos %>% select(ends_with("Width"), Species)

# Agregar al conjunto de datos de los pétalos una nueva variable con la razón
# entre el largo y el ancho de éstos.
petalos <- petalos %>% mutate(Species, Petal.Width,
                              Petal.Ratio = Petal.Length / Petal.Width)

# Ordenar el conjunto de datos de pétalos en forma descendente según la razón
# de los pétalos.
petalos <- petalos %>% arrange(desc(Petal.Ratio))

# Ordenar el conjunto de datos de pétalos en forma ascendente según el largo de
```

```
# los pétalos.
petalos <- petalos %>% arrange(Petal.Length)
```

En el script 1.5 aparece frecuentemente el operador %>%, llamado *pipe* y definido en el paquete `magrittr`, cuya función es entregar un valor o el resultado de una expresión a la siguiente llamada a una función. En términos sencillos, la expresión `x %>% f` es equivalente a `f(x)`, y su utilidad es que simplifica la lectura de llamadas a funciones anidadas (Bache, 2014).

Otra transformación que se usa a menudo es la de pivotar la matriz de datos, cuyo efecto es el de "alrargar" o "ensanchar" la matriz. En el primer caso, se incrementa la cantidad de filas (observaciones) a la vez que se reduce la cantidad de columnas (variables). Para ello se usa la función `pivot_longer(cols, names_to, values_to)` del paquete `tidyr`, donde:

- `cols`: nombres de las columnas a pivotar.
- `names_to`: especifica el nombre de una nueva columna cuyos valores corresponden a los nombres de las columnas a pivotar.
- `values_to`: especifica el nombre de una nueva columna donde se almacenan los valores de las columnas a pivotar.

En el segundo caso se obtiene como resultado una reducción de la cantidad de filas junto al aumento de la cantidad de columnas. Para ello se usa la función `pivot_wider(names_from, values_from)`, también del paquete `tidyr`, donde:

- `names_from`: especifica el nombre de una variable desde la que se obtienen los nombres de las nuevas columnas.
- `values_from`: especifica el nombre de una variable desde donde se obtienen los valores de las nuevas columnas.

Veamos con un ejemplo el efecto de estas dos transformaciones. El script 1.6 comienza por crear una matriz de datos en que se registran los tiempos de ejecución (en milisegundos) para seis instancias de un problema con cuatro algoritmos diferentes. Las columnas de la matriz de datos original corresponden al identificador de la instancia y cada uno de los algoritmos. Así, la matriz de datos original tiene 6 filas y 5 columnas.

A continuación, se crea una nueva matriz de datos, `datos_largos`, que resulta de pivotar la original para "alargarla". Al ejecutar el script 1.6 podemos ver que nuestra nueva matriz de datos tiene solo tres columnas, pero que su cantidad de filas es 24. Si miramos con atención, veremos que ahora tenemos 4 filas por cada instancia, una por cada algoritmo (señalado en la columna `Algoritmo`) con su correspondiente tiempo de ejecución (columna `Tiempo`).

Por último, el script 1.6 crea otro conjunto de datos, `datos_anchos`, a partir de `datos_largos`. Al examinar este nuevo conjunto, se puede apreciar que es idéntico al creado inicialmente.

Script 1.6: modificación de una matriz de datos con el paquete `tidyr`.

```
library(dplyr)
library(tidyr)

# Crear el data frame.
Instancia <- 1:6
Quicksort <- c(23.2, 22.6, 23.4, 23.3, 21.8, 23.9)
Bubblesort <- c(31.6, 29.3, 30.7, 30.8, 29.8, 30.3)
Radixsort <- c(30.1, 28.4, 28.7, 28.3, 29.9, 29.1)
Mergesort <- c(25.0, 25.7, 25.7, 23.7, 25.5, 24.7)
datos <- data.frame(Instancia, Quicksort, Bubblesort, Radixsort, Mergesort)

# Mostrar las primeras filas de la matriz de datos.
cat("Datos originales\n")
print(head(datos))
cat("\n")

```

```r
# Convertir la matriz de datos a formato largo.
datos_largos <- datos %>% pivot_longer(c("Quicksort", "Bubblesort",
                                         "Radixsort", "Mergesort"),
                                       names_to = "Algoritmo",
                                       values_to = "Tiempo")

# Mostrar las primeras filas de la matriz de datos largos.
cat("Datos largos\n")
print(head(datos_largos))
cat("\n")

# Convertir la matriz de datos largos a formato ancho.
datos_anchos <- datos_largos %>% pivot_wider(names_from = "Algoritmo",
                                             values_from = "Tiempo")

# Mostrar las primeras filas de la matriz de datos largos.
cat("Datos anchos\n")
print(head(datos_anchos))
cat("\n")
```

Habrás notado que para poder usar las funciones de `tidyr` se requiere también el paquete `dplyr`. Una alternativa es cargar únicamente el paquete `tidyverse`, el cual los incluye a ambos (entre otros).

Puedes encontrar descripciones más extensas acerca del uso de la funciones del paquete `tidyverse`, junto con ejemplos más avanzados, en Wickham (2021).

En ocasiones puede ser necesario renombrar las columnas para que nos resulte más fácil comprender a qué variable corresponde. La función `rename()` del paquete `dplyr` nos permite hacer esta operación bastante sencilla. Sus argumentos son una lista de elementos de la forma `nuevo nombre = nombre original`. También podemos cambiar el tipo de una variable. Una conversión que nos será muy útil es de variable numérica a categórica, lo que se logra mediante la función `factor(x, levels, labels, ordered)`, donde:

- `x`: nombre de la variable a convertir.
- `levels`: argumento opcional con los posibles valores de la variable categórica.
- `labels`: argumento opcional con las etiquetas asociadas a cada valor.
- `ordered`: valor lógico que especifica si la variable es o no ordinal (falso por defecto).

Tomemos el conjunto de datos `mtcars` (incluido en R) para ejemplificar el uso de estas funciones. La tabla 1.3 muestra la descripción de estos datos. El script 1.7 modifica los nombres de las columnas para que sean más representativos y da formato de variable categórica a las variables que así lo requieren, asignando etiquetas adecuadas para cada nivel.

Variable	Descripción
mpg	Rendimiento, en millas / galón (EEUU).
cyl	Número de cilindros.
disp	Desplazamiento, en pulgadas cúbicas.
hp	Potencia, en caballos de fuerza brutos.
drat	Razón del eje trasero.
wt	Peso, en miles de libras.
qsec	Tiempo que tarda en recorrer un cuarto de milla partiendo desde el reposo, en segundos.
vs	Tipo de motor (0: en forma de V, 1: recto).
am	Tipo de transmisión (0: automática, 1: manual).
gear	Número de marchas hacia adelante.
carb	Número de carburadores.

Tabla 1.3: descripción de las variables para el conjunto de datos `mtcars`.

Script 1.7: modificación del conjunto de datos mtcars para facilitar su comprensión.

```r
library(dplyr)

# Cargar conjunto de datos.
datos <- mtcars

# Renombrar columnas.
datos <- datos %>% rename(Rendimiento = mpg, Cilindrada = cyl,
                          Desplazamiento = disp, Potencia = hp,
                          Eje = drat, Peso = wt, Cuarto_milla = qsec,
                          Motor = vs, Transmision = am, Cambios = gear,
                          Carburadores = carb)

# Dar formato categórico a las variables Motor y Transmision, renombrando
# sus niveles.
datos[["Motor"]] <- factor(datos[["Motor"]], levels = c(0, 1),
                           labels = c("V", "Recto"))

datos[["Transmision"]] <- factor(datos[["Transmision"]], levels = c(0, 1),
                                 labels = c("Automático", "Manual"))

# Dar formato ordinal a las variables Cilindrada y Cambios, renombrando
# sus niveles.
datos[["Cilindrada"]] <- factor(datos [["Cilindrada"]], levels = c(4, 6, 8),
                                labels = c("4 cilindros", "6 cilindros",
                                           "8 cilindros"),
                                ordered = TRUE)

datos[["Cambios"]] <- factor(datos[["Cambios"]], levels = c(3, 4, 5),
                             labels = c("3 cambios", "4 cambios", "5 cambios"),
                             ordered = TRUE)

write.csv2(datos , "C:/Inferencia/Mtcars.csv")
```

1.3.5 Fórmulas

Si bien hasta ahora solo tenemos una definición preliminar de lo que es un modelo estadístico, necesitamos conocer una herramienta para representarlos en R, pues son una parte fundamental del funcionamiento de este lenguaje.

Para entender de manera sencilla qué es una fórmula, podemos simplemente decir que permite capturar una expresión no evaluada, y que está asociada a un ambiente. Su sintaxis básica tiene la forma `variable independiente ~ variables dependientes`, lo que nos indica, entonces, que las fórmulas representan una relación entre variables.

Tomemos una vez más el conjunto de datos `iris`. Podríamos representar la asociación entre la especie de iris (variable independiente) y las dimensiones de sus pétalos (variables dependientes) como `Species ~ Petal.Length + Petal.Width`.

Extenderemos las nociones acerca del uso de fórmulas a medida que avancemos en nuestro aprendizaje, pero si quieres aprender más puedes consultar Willems (2017).

1.4 EJERCICIOS PROPUESTOS

1. Una encuesta reciente preguntó: "después de la jornada laboral usual, ¿cuántas horas dedica a relajarse o a realizar actividades que disfruta?" a una muestra de 580 chilenas y 575 chilenos. Se encontró que el número promedio de horas era de $1{,}30 \pm 0{,}30$ y $1{,}95 \pm 0{,}25$ para cada grupo, respectivamente.
   a) ¿Cómo sería una matriz de datos para este estudio? Muestra algunas filas de ella como ejemplos.
   b) ¿Cuál podría ser la población objetivo?
   c) ¿Qué se entendería por unidad de observación?
   d) ¿Qué tipo de variable sería "el número de horas dedicadas a distraerse después de la jornada laboral usual" que respondió cada persona entrevistada?
   e) ¿Existe alguna variable categórica? Si es así, ¿de qué tipo? ¿Con qué niveles?
   f) ¿Qué dato(s) correspondería(n) a un estadístico?
   g) ¿Cuál(es) sería(n) el(los) parámetro(s) en estudio?
   h) ¿Logra el estudio establecer que ser mujer chilena ocasiona tener menos horas dedicadas a distraerse después de la jornada laboral usual?
2. Investiga para qué sirven y cómo se usan los argumentos `row.names` y `col.names` en las funciones para importar datos desde archivos y la función `data.frame()`.
3. Construye en R una matriz de datos para almacenar las características de una muestra de servidores. Considera a lo menos una variable categórica y una variable numérica.
4. Investiga qué función (o funciones) ofrece R para guardar una matriz de datos en un archivo y úsala(s) para guardar la matriz de datos del ejercicio anterior.
5. Resuelve en R los siguientes ejercicios. Considera para ello el conjunto de datos nativo de R `chickwts`.
   a) ¿Cómo se puede cargar el conjunto de datos en la variable `pollos`?
   b) ¿Cómo se ve la estructura de la matriz de datos almacenada en `pollos`?
6. Muestra ejemplos de las distintas transformaciones que se pueden hacer a una matriz de datos usando para ello conjunto de datos nativo de R `ChickWeight`.

CAPÍTULO 2. EXPLORACIÓN DE DATOS

Siempre es bueno que nos familiaricemos con los datos y algunas de sus características antes de empezar a trabajar con ellos. Esto nos ayuda a decidir qué herramientas son las más adecuadas para dar respuesta a las preguntas que queramos responder. En este capítulo revisaremos las principales estadísticas descriptivas que nos ayudarán a resumir los datos para entenderlos mejor, así como diversos tipos de gráficos que nos permitirán representar los datos de modo que podamos comprenderlos de forma visual. Para ello, tomamos como base los conceptos expuestos en Diez y col. (2017, pp. 26-50), Field y col. (2012, pp. 19-27) y STDHA (s.f.), fuentes que puedes consultar si deseas saber más acerca de estos temas.

Para muchos de los ejemplos de este capítulo usaremos el conjunto de datos `mtcars` con las modificaciones realizadas en el script 1.7, cuyo diccionario de datos se muestra en la tabla 2.1.

Variable	Descripción
Rendimiento	Rendimiento, en millas / galón (EEUU).
Cilindrada	Número de cilindros (4 cilindros, 6 cilindros, 8 cilindros).
Desplazamiento	Desplazamiento, en pulgadas cúbicas.
Potencia	Potencia, en caballos de fuerza brutos.
Eje	Razón del eje trasero.
Peso	Peso, en miles de libras.
Cuarto_milla	Tiempo que tarda en recorrer un cuarto de milla partiendo desde el reposo, en segundos.
Motor	Tipo de motor (V, Recto).
Transmision	Tipo de transmisión (Automático, Manual).
Cambios	Número de cambios hacia adelante (3 cambios, 4 cambios, 5 cambios).
Carburadores	Número de carburadores.

Tabla 2.1: descripción de las variables para el conjunto de datos `mtcars`.

2.1 ESTADÍSTICAS DESCRIPTIVAS

Las estadísticas descriptivas son medidas que nos permiten sintetizar y, como su nombre lo indica, describir los datos. Estas pueden aplicarse tanto a una muestra como a una población. Cuando una de estas medidas se aplica a la muestra, corresponde a un **estimador puntual** de la misma medida para la población. Al ser una estimación, no es exacta, aunque la precisión tiende a aumentar mientras mayor sea el tamaño de la muestra.

Un concepto importante a tener en cuenta es la noción de **distribución**. En este capítulo se considera la **distribución de frecuencia**, que representa cuántas veces aparece cada valor para una variable en un conjunto de datos.

2.1.1 Estadísticas descriptivas para datos numéricos

Una de las estadísticas descriptivas más empleadas es la **media**, conocida en otros contextos como media aritmética o promedio. Denotamos la **media muestral** por $\overline{x}$, donde x corresponde al nombre de la variable, mientras que para la **media poblacional** empleamos la notación μ_x. Esta medida se calcula como muestra la ecuación 2.1, donde x_i son los n valores observados de la variable. Podemos entender la media como el punto de equilibrio de la distribución (Diez y col., 2017, p. 28). Así, la media corresponde a una **medida de tendencia central**.

$$\overline{x} = \frac{1}{n} \sum_{i=1}^{n} x_i \tag{2.1}$$

El script 2.1 muestra cómo usar la función `mean()` de R para calcular la media de diversas variables del conjunto de datos `mtcars`[1]. Como primer ejemplo, se calcula la media de la variable `Rendimiento`. A continuación se muestra cómo realizar esta operación para dos variables, señaladas por el índice de sus respectivas columnas. Luego, de manera similar, se calculan las medias para cuatro columnas consecutivas de la matriz de datos. En estos dos casos hacemos uso de la función `sapply()`, que permite aplicar una misma función (cualquiera) para múltiples columnas.

Script 2.1: uso de las funciones `mean()` y `sapply()`.

```r
# Cargar conjunto de datos.
datos <- read.csv2("C:/Inferencia/Mtcars.csv", stringsAsFactors = TRUE,
                   row.names = 1)

# Calcular la media para la variable Rendimiento.
media <- mean(datos[["Rendimiento"]])
cat("Rendimiento medio:", media, "\n\n")

# Calcular la media para la tercera y quinta columnas
# (variables Desplazamiento y Eje).
cat("Medias\n")
print(sapply(datos[c(3, 5)], mean))
cat("\n")

# Calcular la media para las columnas 3 a 6
# (variables Desplazamiento, Potencia, Eje y Peso).
cat("Medias\n")
print(sapply(datos[3:6], mean))
cat("\n")

# Calcular la media para la variable Rendimiento omitiendo valores faltantes.
print(mean(datos[["Rendimiento"]], na.rm = TRUE))
```

La función `mean()` devuelve `NA` (*not available*, es decir, no disponible) si existen valores faltantes en los datos de entrada. Para prevenir este error, se puede proporcionar un argumento adicional que descarte los valores faltantes, como muestra la última línea del script 2.1.

Una medida de tendencia central alternativa a la media es la **mediana**, que es, simplemente, el valor central de los valores previamente ordenados. Cuando no existe un valor central, vale decir, cuando el tamaño de la muestra es par, la mediana está dada por el promedio simple de los dos valores centrales. En R, la mediana se calcula con la función `median()`.

[1]Todas las demás funciones de R mencionadas en esta sección para las que no se proporcione un script se usan del mismo modo que `mean()`.

La **moda** es, simplemente, el valor más frecuente en el conjunto de datos. No obstante, tiene el problema de que puede haber múltiples modas. Dependiendo de la cantidad de modas, se habla de distribuciones **unimodales**, **bimodales** y **multimodales**.

Si bien R no cuenta con una función nativa para encontrar la moda, el paquete `modeest` ofrece la función `mfv()` que entrega el valor más frecuente de una variable. En caso de que dos (o más) valores sean los más frecuentes con igual cantidad de observaciones, los entrega todos en forma de vector.

Las medidas que hemos estudiado hasta ahora buscan describir el centro del conjunto de datos. No obstante, también es importante conocer su **variabilidad o dispersión**, pues así se puede saber qué tan semejantes (o diferentes) son las observaciones entre sí. Estas suelen calcularse en base a la **desviación** de las observaciones, que se entiende como la distancia entre una observación y la media del conjunto de datos. Las dos principales medidas de dispersión son la **varianza** y la **desviación estándar**, ambas basadas en los cuadrados de las distancias, ya que, por una parte, los valores grandes se incrementan más significativamente y, por otra, se opera solo con valores positivos, pues la dirección de la distancia no es de interés.

La varianza muestral se calcula como muestra la ecuación 2.2, donde x_i son los valores de cada una de las n observaciones. Cabe destacar que puede emplearse un subíndice para indicar el nombre de la variable, al igual que en el caso de la media.

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2 \tag{2.2}$$

La desviación estándar de la muestra se define como la raíz cuadrada de la varianza (2.3), medida que resulta de gran utilidad cuando se necesita saber cuán cercanos son los datos a la media, ya que se encuentra en la misma escala que la variable.

$$s = \sqrt{s^2} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2} \tag{2.3}$$

Al igual que en el caso de la media, podemos usar las fórmulas anteriores para obtener estimaciones puntuales de la varianza y la desviación estándar de la población, denotadas por σ^2 y σ, respectivamente.

Es importante considerar que, si bien la media y la desviación estándar permiten conocer el centro y la dispersión del conjunto de datos, respectivamente, la distribución de los puntos puede ser muy diferente, como ilustra la figura 2.1.

Las funciones de R para calcular la varianza y la desviación estándar son, respectivamente, `var()` y `sd()`.

Aunque menos empleado, el **rango** muestra los valores extremos, es decir, el mínimo y el máximo, de una variable. R ofrece la función `range()` para obtener ambos valores, además de `min()` y `max()` para obtenerlos por separado.

En párrafos anteriores vimos que la mediana es el valor central (o el promedio de los dos valores centrales) del conjunto de datos ordenado, ya sea una población o una muestra. Esto significa, entonces, que esta medida divide el conjunto de datos en dos mitades con igual cantidad de elementos. De manera similar, es posible dividir el conjunto de datos en segmentos más pequeños, por ejemplo en 4, 10 o 100 partes con igual cantidad de elementos. Cada fragmento del conjunto de datos dividido de esta forma recibe el nombre de **cuantil**. Algunas subdivisiones de uso frecuente reciben nombres especiales:

- **Percentiles**: dividen el conjunto de datos en 100 subconjuntos de igual tamaño.
- **Deciles**: dividen el conjunto de datos en 10 subconjuntos de igual tamaño.
- **Quintiles**: dividen el conjunto de datos en 5 subconjuntos de igual tamaño.
- **Cuartiles**: dividen el conjunto de datos en 4 subconjuntos de igual tamaño.

Los cuantiles (al igual que las otras subdivisiones antes mencionadas) se nombran de forma ascendente según el sentido de crecimiento del conjunto de datos. Así, el percentil 1 contiene a los valores más pequeños,

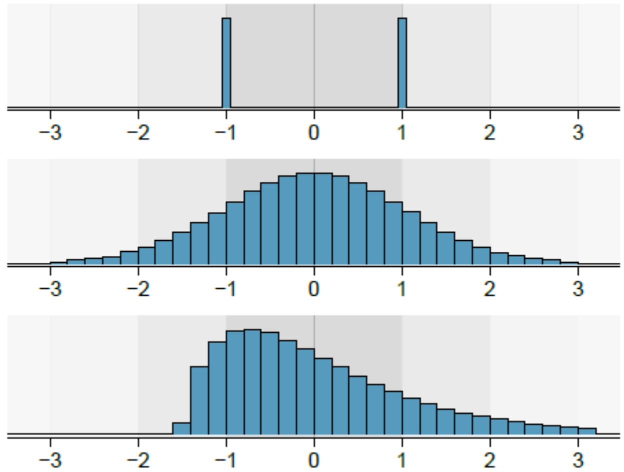


Figura 2.1: tres distribuciones de población muy distintas con media $\mu = 0$ y desviación estándar $\sigma = 1$. Fuente: Diez y col. (2017, p. 34).

mientras que el percentil 100, a los más grandes. Cabe destacar que la mediana corresponde al percentil 50 o al cuartil 2, y que nombrar al decil 3 es equivalente al percentil 30.

R proporciona la función `quantile()` para calcular cuantiles, que por defecto calcula los cuartiles, aunque su uso puede generalizarse mediante el parámetro adicional `probs`, como muestra el script 2.2. La función `seq()` genera una secuencia de números equiespaciados, y recibe como argumentos el inicio, el término y el incremento de la secuencia.

Script 2.2: cálculo de cuantiles con la función `quantile()`.

```
# Cargar conjunto de datos.
datos <- read.csv2("C:/Inferencia/Mtcars.csv", stringsAsFactors = TRUE,
                   row.names = 1)

# Cálculo de percentiles para la variable Rendimiento.
cat("Cuartiles:\n")
print(quantile(datos[["Rendimiento"]]))
cat("\n")

cat("Quintiles:\n")
print(quantile(datos[["Rendimiento"]], seq(0, 1, 0.2)))
cat("\n")

cat("Deciles:\n")
print(quantile(datos[["Rendimiento"]], seq(0, 1, 0.1)))
cat("\n")

cat("Percentiles:\n")
print(quantile(datos[["Rendimiento"]], seq(0, 1, 0.01)))
```

Ahora que conocemos los cuartiles, podemos introducir una nueva medida de variabilidad que usaremos a menudo, llamada **rango intercuartil** o IQR (por su sigla en inglés), dada por la ecuación 2.4, donde Q_1 y Q_3 corresponden a los cuartiles 1 y 3, respectivamente. Al igual que la varianza y la desviación estándar, mientras más disperso sea el conjunto de datos, mayor será el valor del IQR. En R, la función que calcula este estimador es `IQR()`.

$$IQR = Q_3 - Q_1 \tag{2.4}$$

Muchas veces los conjuntos de datos contienen lo que se conoce como **valores atípicos** o ***outliers***. Estos corresponden a observaciones que parecen estar fuera de rango o ser muy extremos con respecto al resto de los datos. Medidas como la media o la desviación estándar son muy sensibles a los valores atípicos, por lo que son propensas a errores ante la presencia de este tipo de observaciones. Para reducir el efecto de los valores extremos muchas veces necesitaremos medidas **robustas**, que son aquellas que proporcionan una estimación confiable aún ante la presencia de valores atípicos. En este escenario, la mediana resulta ser una buena medida de tendencia central y el IQR, una buena medida de dispersión.

Nos encontraremos frecuentemente con la necesidad de calcular varias medidas de tendencia central y de dispersión descritas en el apartado anterior. Por esta razón, R, y algunos de sus paquetes, ofrecen algunas funciones que calculan varios de estos estadísticos con una sola llamada. Tal es el caso de la función nativa `summary()`, que entrega la media, la mediana, el primer y el tercer cuartil, el mínimo y el máximo. Otra función que nos puede ser de mucha ayuda es `summarise()`, del paquete `dplyr`. Con ella podemos calcular varias de las medidas en una sola llamada, como muestra el script 2.3.

Script 2.3: uso de la función `summarise()` del paquete `dplyr`.

```r
library(dplyr)

datos <- read.csv2("C:/Inferencia/Mtcars.csv", stringsAsFactors = TRUE,
                   row.names = 1)

# Cálculo de varias medidas para la variable Potencia.
medidas_potencia <- datos %>% summarise(Media = mean(Potencia),
                                        Mediana = median(Potencia),
                                        Varianza = var(Potencia),
                                        IQR = IQR(Potencia))

print(medidas_potencia)
cat("\n")

# Cálculo de la media y la desviación estándar para las variables Peso y
# Cuarto_milla.
medidas_varias <- datos %>% summarise(Media_P = mean(Peso),
                                        Media_C = median(Cuarto_milla),
                                        SD_P = sd(Peso),
                                        SD_C = sd(Cuarto_milla))

print(medidas_varias)
cat("\n")
```

2.1.2 Estadísticas descriptivas para datos categóricos

Cuando queremos trabajar con datos categóricos, medidas como la media o la desviación estándar carecen de sentido. En consecuencia, necesitamos otros estadísticos para resumir el conjunto de datos.

Como primer estadístico para variables categóricas podemos mencionar la **frecuencia**, que corresponde a la cantidad de veces que podemos encontrar cada nivel de la variable en los datos. Otro estadístico importante corresponde a la **proporción**, que corresponde a la frecuencia relativa. En otras palabras, la proporción corresponde a frecuencia de un nivel de la variable dividida por la cantidad total de observaciones.

La mejor alternativa para este tipo de datos es la **tabla de contingencia**, también llamada **matriz de confusión** o **tabla de frecuencias**, donde cada fila representa la cantidad de veces en que ocurre una combinación de variables. También es posible usar porcentajes o proporciones en lugar de la cantidad de ocurrencia, en cuyo caso se habla de una **tabla de frecuencias relativas**. La tabla 2.2 muestra la tabla de contingencia (de frecuencias) para la variable `Cambios`. Se puede observar, por ejemplo, que el conjunto de datos contiene una muestra de 32 automóviles y que 15 de ellos tienen tres cambios.

3 cambios	4 cambios	5 cambios	Total
15	12	5	32

Tabla 2.2: tabla de contingencia para la cantidad de cambios de los automóviles.

Desde luego, podemos construir tablas de contingencia de manera bastante sencilla en R. El script 2.4 muestra dos formas de obtener la tabla 2.2. La primera es la función `table()` y la segunda, la función `xtabs()`. El funcionamiento de ambas es equivalente, aunque `xtabs()` muestra el nombre de la variable tabulada al imprimir los resultados y `table()` no lo hace. Las tablas entregadas por estas funciones no incluyen los totales por filas, pero la función `marginSums()` permite calcularlos y mostrarlos como un vector. A su vez, la función `addmargins()` permite calcular dichos totales e incorporarlos a la tabla. Para terminar, el las últimas sentencias del script 2.4 ilustran la manera de obtener las tablas de frecuencias relativas con proporciones y porcentajes, respectivamente.

Podemos ver que las llamadas a `table()` y a `xtabs()` son algo diferentes. La primera recibe como argumento la columna de la matriz de datos, es decir, un vector con los datos a tabular, mientras que la segunda recibe una fórmula en que no existe una variable dependiente y la variable categórica es la independiente.

Script 2.4: tabla de contingencia para la variable `Cambios`.

```r
# Cargar datos.
datos <- read.csv2("C:/Inferencia/Mtcars.csv", stringsAsFactors = TRUE,
                   row.names = 1)

# Crear tabla de contingencia para la variable gear.
contingencia <- table(datos[["Cambios"]])
cat("Tabla de contingencia generada con table():\n")
print(contingencia)
cat("\n")

# Otra forma de crear la misma tabla.
contingencia <- xtabs(~ Cambios, data = datos)
cat("Tabla de contingencia generada con xtabs():\n")
print(contingencia)
cat("\n")

# Calcular totales por fila y mostrarlos por separado.
totales <- marginSums(contingencia)
cat("Totales por fila:\n")
print(totales)
cat("\n")

# Calcular totales por fila y agregarlos a la tabla.
con_totales <- addmargins(contingencia, 1)
cat("Tabla de contingencia con totales por fila:\n")
print(con_totales)
cat("\n")

# Convertir a tabla de proporciones
proporciones <- prop.table(contingencia)
proporciones <- addmargins(proporciones, 1)
cat("Tabla de contingencia con proporciones:\n")
print(proporciones)
cat("\n")

```

```r
# Convertir a tabla de porcentajes con 2 decimales.
porcentajes <- round(prop.table(contingencia), 4) * 100
porcentajes <- addmargins(porcentajes)
cat("Tabla de contingencia con porcentajes:\n")
print(porcentajes)
cat("\n")
```

También podemos construir matrices de confusión para dos variables categóricas, como muestra la tabla 2.3 para las variables `Cambios` y `Transmisión`.

		Cambios			
		3 cambios	4 cambios	5 cambios	Total
Transmision	Automático	15	4	0	19
	Manual	0	8	5	13
	Total	15	12	5	32

Tabla 2.3: tabla de contingencia para las variables `Cambios` y `Transmisión`.

En ocasiones resulta útil determinar las proporciones por fila o por columna, que podemos obtener dividiendo el valor de una celda de la matriz por el total de su fila o columna, según corresponda. Así, el total de cada fila (o columna) es igual a 1. Puesto que las proporciones por fila y por columna no son equivalentes, debemos ser cuidadosos al escoger la más adecuada en cada caso. Las tablas 2.4 a 2.6 muestran las proporciones por fila, por columna y generales para la matriz de confusión de la tabla 2.3. La construcción en R de la tabla de contingencia y las tablas de proporciones para dos variables se muestra en el script 2.5.

		Cambios			
		3 cambios	4 cambios	5 cambios	Total
Transmision	Automático	0,7894737	0,2105263	0,0000000	1,0000000
	Manual	0,0000000	0,6153846	0,3846154	1,0000000

Tabla 2.4: tabla de proporciones con totales por fila para la tabla 2.3.

		Cambios		
		3 cambios	4 cambios	5 cambios
Transmision	Automático	1,0000000	0,3333333	0,0000000
	Manual	0,0000000	0,6666667	1,0000000
	Total	1,0000000	0,0000000	1,0000000

Tabla 2.5: tabla de proporciones con totales por columna para la tabla 2.3.

Script 2.5: tablas de contingencia y proporciones para dos variables.

```r
# Cargar datos.
datos <- read.csv2("C:/Inferencia/Mtcars.csv", stringsAsFactors = TRUE,
                   row.names = 1)

# Crear tabla de contingencia para las variables Transmision y gear.
contingencia <- table(datos[["Transmision"]], datos[["Cambios"]])
cat("Tabla de contingencia generada con table():\n")
print(contingencia)
cat("\n")

# Otra forma de crear la misma tabla.
contingencia <- xtabs(~ Transmision + Cambios, data = datos)
cat("Tabla de contingencia generada con xtabs():\n")
print(contingencia)
cat("\n")
```

		Cambios			
		3 cambios	4 cambios	5 cambios	Total
Transmision	Automático	0,46875	0,12500	0,00000	0,59375
	Manual	0,00000	0,25000	0,15625	0,40625
	Total	0,46875	0,37500	0,15625	1,00000

Tabla 2.6: tabla de proporciones con totales por fila y columna para la tabla 2.3.

```

# Proporciones con totales por fila.
proporciones_fila <- prop.table(contingencia, margin=1)
proporciones_fila <- addmargins(proporciones_fila, margin=2)
cat("Tabla de contingencia con proporciones totales por fila:\n")
print(proporciones_fila)
cat("\n")

# Proporciones con totales por columna.
proporciones_columna <- prop.table(contingencia, margin=2)
proporciones_columna <- addmargins(proporciones_columna, margin=1)
cat("Tabla de contingencia con proporciones totales por columna:\n")
print(proporciones_columna)
cat("\n")

# Proporciones con totales.
proporciones <- prop.table(contingencia)
proporciones <- addmargins(proporciones)
cat("Tabla de contingencia con proporciones totales:\n")
print(proporciones)
cat("\n")
```

Aunque no ocurre con frecuencia, podríamos necesitar una matriz de confusión para más de dos variables. Veamos ahora un ejemplo con tres variables: `Motor`, `Cambios` y `Transmisión`. Para ello, tomamos una de las variables (en este caso, `Motor`) y creamos una subtabla por cada uno de sus niveles. Cada subtabla muestra las frecuencias para la combinación de las dos variables restantes cuando `Motor` tiene el nivel correspondiente, como muestra la tabla 2.7. En R, podemos obtener estas tablas como muestra el script 2.6. Desde luego, esta misma idea puede extenderse para cuatro o más variables categóricas.

Motor = Recto

		Cambios		
		3 cambios	4 cambios	5 cambios
Transmision	Automático	3	4	0
	Manual	0	6	1

Motor = V

		Cambios		
		3 cambios	4 cambios	5 cambios
Transmision	Automático	12	0	0
	Manual	0	2	4

Tabla 2.7: tabla de contingencia para tres variables.

Script 2.6: matriz de confusión para tres variables.

```
# Cargar datos.
datos <- read.csv2("C:/Inferencia/Mtcars.csv", stringsAsFactors = TRUE,
                   row.names = 1)

```

```
# Convertir la variable Cambios en categórica.
datos[["Cambios"]] <- factor(datos[["Cambios"]])

# Crear tabla de contingencia para las variables Transmision,
# Cambios y Motor.
contingencia <- ftable(datos[["Transmision"]], datos[["Cambios"]],
                       datos[["Motor"]])

cat("Tabla de contingencia generada con ftable():\n")
print(contingencia)
cat("\n")

# Otra forma de crear la misma tabla.
xtabs(~ Cambios + Transmision + Motor, data = datos)
cat("Tabla de contingencia generada con xtabs():\n")
print(contingencia)
cat("\n")
```

2.1.3 Trabajando con datos agrupados

A menudo nos veremos en la necesidad de obtener estadísticas descriptivas de una variable separando las observaciones en grupos de acuerdo a una variable categórica. Para ello, el paquete `dplyr` ofrece la función `group_by()`, que podemos usar en conjunto con `summarise()`, como muestra el script 2.7. En dicho script, primero se agrupan las observaciones de acuerdo a la variable `Cambios`, y luego se efectúa una llamada a `summarise()` donde el primer argumento cuenta la cantidad de observaciones en el grupo actual y los argumentos restantes (que pueden ser tantos como se desee) corresponden a diferentes estadísticas descriptivas.

Script 2.7: estadísticas descriptivas para datos agrupados.

```
library(dplyr)

# Cargar datos.
datos <- read.csv2("C:/Inferencia/Mtcars.csv", stringsAsFactors = TRUE,
                   row.names = 1)

resumen <- group_by(datos, Cambios) %>%
  summarise(count = n(), mean(Rendimiento), median(Rendimiento),
            sd(Rendimiento), IQR(Rendimiento), mean(Potencia))

print(resumen)
```

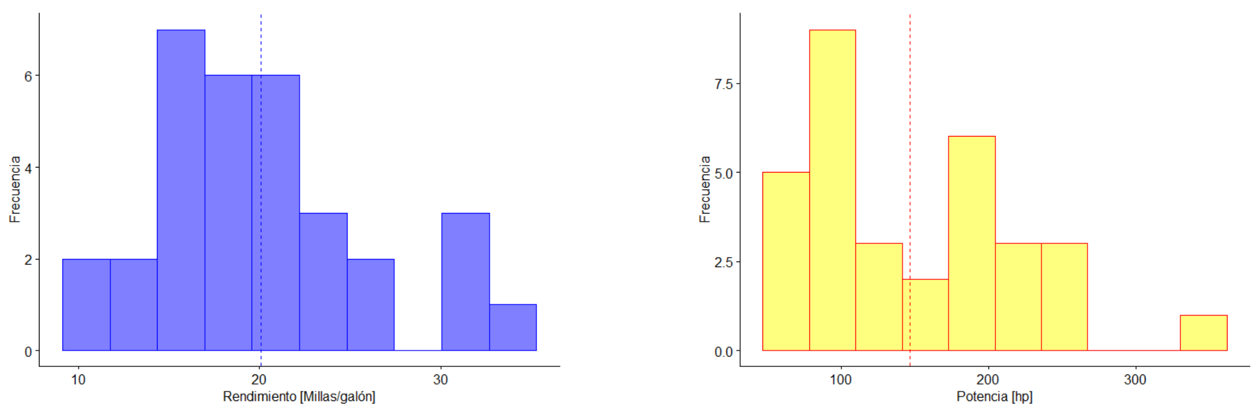
2.2 REPRESENTACIÓN GRÁFICA DE DATOS

En esta sección revisaremos diversos tipos de gráficos que resultan útiles al momento de estudiar un conjunto de datos disponibles, considerando su definición, su utilidad y cómo se construyen en R. Para crear gráficos en R usaremos el paquete `ggpubr`. Algunos de los principales parámetros que usaremos para crear y editar gráficos con este paquete son:

- `data`: un data frame.
- `x`: string con el nombre de la variable `x`.
- `y`: string(s) con el(los) nombre(s) de la(s) variable(s) a graficar.
- `color`: color de delineado.
- `fill`: color de relleno.
- `palette`: paleta de colores cuando existen múltiples grupos.
- `linetype`: tipo de línea a emplear.
- `add`: permite agregar elementos adicionales al gráfico, como barras de error o la media, entre otros.
- `title`: título del gráfico.
- `xlab`: rótulo del eje x. Puede ocultarse usando `xlab = FALSE`.
- `ylab`: rótulo del eje y. Puede ocultarse usando `ylab = FALSE`.

2.2.1 Una variable numérica

El **histograma** resulta muy útil si queremos representar una única variable numérica y la muestra es grande. Podemos decir que este gráfico muestra una aproximación a la **densidad** (o distribución de frecuencias) para la variable, para lo que tenemos que dividir el rango de valores posibles en intervalos (generalmente iguales) y luego contar la cantidad de observaciones en cada intervalo. Para construir el gráfico, creamos una barra por cada intervalo, cuya altura (o longitud) es proporcional a la cantidad de observaciones en el intervalo representado. La figura 2.2 muestra histogramas creados con el script 2.8.



(a) Distribución cercana a la simétrica.

(b) Distribución desviada a la izquierda.

Figura 2.2: dos histogramas.

Script 2.8: histogramas para las variables `Rendimiento` y `Potencia`.

```
library(ggpubr)

# Cargar datos.
datos <- read.csv2("C:/Inferencia/Mtcars.csv", stringsAsFactors = TRUE,
                   row.names = 1)

# Histograma para la variable Rendimiento.
g1 <- gghistogram(datos,
                  x = "Rendimiento",
                  bins = 10,
                  add = "mean",
```

```
                     xlab = "Rendimiento [Millas/galón]",
                     ylab = "Frecuencia",
                     color = "blue",
                     fill = "blue")

print(g1)

# Histograma para la variable Potencia.
g2 <- gghistogram(datos,
                     x = "Potencia",
                     bins = 10,
                     add = "mean",
                     xlab = "Potencia [hp]",
                     ylab = "Frecuencia",
                     color = "red",
                     fill = "yellow")

print(g2)
```

A medida que avancemos en este libro, veremos que es muy importante conocer la **distribución de frecuencias** de una variable. Al observar la figura 2.2b, podemos ver que la frecuencia es mayor para potencias más bajas, pues las barras de la izquierda del gráfico son, en general, algo más altas que las de la derecha. Podría decirse que las observaciones se concentran a la izquierda y que hay una cola que se prolonga hacia la derecha. Cuando esto ocurre, decimos que la la distribución está **desviada a la izquierda**, o que hay **asimetría negativa**. Análogamente, podría darse que la distribución estuviese desviada a la derecha o, equivalentemente, que presenta asimetría positiva. En el caso de la figura 2.2a, el histograma es más **simétrico**, pues las observaciones se aglomeran hacia el centro y hay colas tanto a la izquierda como a la derecha. Para ilustrar mejor la idea de la simetría, podemos revisar una vez más la figura 2.1, donde la población central es perfectamente simétrica y la inferior presenta asimetría positiva.

Otra ventaja de los histogramas es que permiten identificar modas de una variable, las cuales corresponden a barras que sean más prominentes que las de su entorno. Ambos ejemplos de la figura 2.2 son bimodales, pues tienen dos modas claramente identificables. Si bien es cierto que en ambos casos hay un único valor más frecuente (moda), podemos ver apreciar que existen dos "cumbres" o máximos locales.

Otro gráfico que usaremos a menudo es el de **gráfico de caja**. Es muy útil, pues su construcción considera 5 estadísticos para representar el conjunto de datos y además facilita la identificación de datos atípicos. La figura 2.3 muestra este gráfico para la variable `Potencia`, el cual fue creado con el script 2.9.

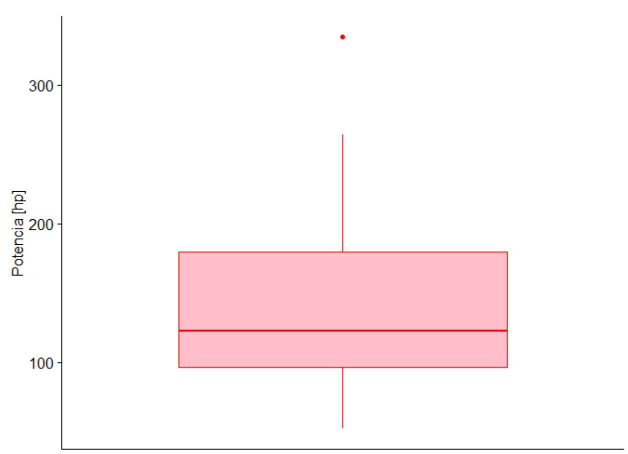


Figura 2.3: gráfico de caja para la variable `Potencia`.

Los extremos inferior y superior del rectángulo o caja de la figura 2.3 corresponden, respectivamente, al primer y al tercer cuartil, mientras que la línea horizontal al interior de la caja denota la mediana. Así, la caja engloba el 50 % central de los datos, y su altura corresponde al rango intercuartil. Las barras que se extienden por sobre y por debajo de la caja, llamadas bigotes, capturan aquellos datos fuera de la caja central y que estén situados a no más de 1,5 veces el IQR. Cualquier observación que esté más allá de la caja y los bigotes se representa como un punto, el cual podría tratarse de una observación atípica.

Script 2.9: gráfico de caja para la variable `Potencia`.

```
library(ggpubr)

# Cargar datos.
datos <- read.csv2("C:/Inferencia/Mtcars.csv", stringsAsFactors = TRUE,
                   row.names = 1)

g <- ggboxplot(datos[["Potencia"]],
               color = "red",
               fill = "pink",
               ylab = "Potencia [hp]")

g <- g + rremove("x.ticks")
g <- g + rremove("x.text")
g <- g + rremove("x.title")

print(g)
```

2.2.2 Una variable categórica

Si queremos representar una única variable categórica, lo más adecuado es usar un **gráfico de barras**, pues cada barra es tan larga como la proporción de valores presentes en cada nivel de la variable. La figura 2.4 muestra el gráfico de barras correspondiente a la tabla 2.2, elaborado mediante el script 2.10.

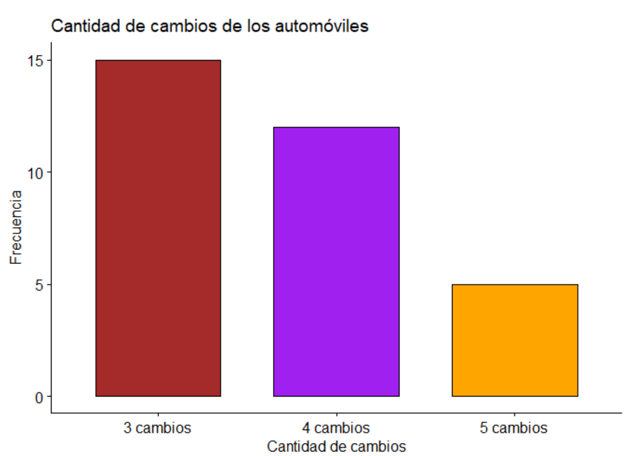


Figura 2.4: gráfico de barras para la variable `Cambios`.

Script 2.10: gráfico de barras para la variable `Cambios`.

```
library(ggpubr)

```

```r
# Cargar datos.
datos <- read.csv2("C:/Inferencia/Mtcars.csv", stringsAsFactors = TRUE,
                   row.names = 1)

# Crear la tabla de frecuencias para la variable Cambios y convertirla a
# data frame.
contingencia <- as.data.frame(xtabs(~ Cambios, data = datos))

# Crear el gráico de barras.
g <- ggbarplot(contingencia,
               x = "Cambios",
               y = "Freq",
               fill = c("brown", "purple", "orange"),
               title = "Cantidad de cambios de los automóviles",
               xlab = "Cantidad de cambios",
               ylab = "Frecuencia")

print(g)
```

Otra alternativa para representar una única variable categórica es el **gráfico de torta**, que se presenta en la figura 2.5 y se construye en R como muestra el script 2.11.

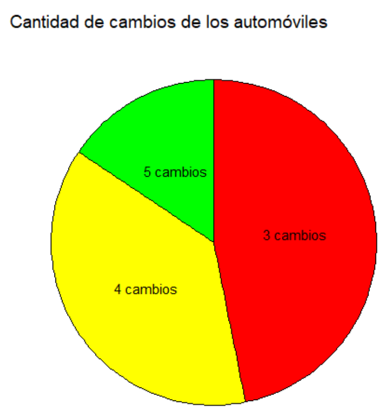


Figura 2.5: gráfico de torta para la variable `Cambios`.

Script 2.11: gráfico de torta para la variable `Cambios`.

```r
library(ggpubr)

# Cargar datos.
datos <- read.csv2("C:/Inferencia/Mtcars.csv", stringsAsFactors = TRUE,
                   row.names = 1)

# Crear la tabla de frecuencias y convertirla a data frame.
contingencia <- as.data.frame(xtabs(~ Cambios, data = datos))

# Crear gráfico de torta.
g <- ggpie(contingencia,
           x = "Freq",
           label = "Cambios",
           fill = c("red", "yellow", "green"),
           title = "Cantidad de cambios de los automóviles",
           lab.pos = "in")

```

```
print(g)
```

2.2.3 Dos variables numéricas

Los **gráficos de dispersión** son adecuados en este caso. Se caracterizan porque muestran información caso a caso, ya que cada punto del gráfico corresponde a una observación. Por ejemplo, el gráfico de la figura 2.6, creado mediante el script 2.12, muestra este tipo de gráfico para las variables `Rendimiento` y `Peso`.

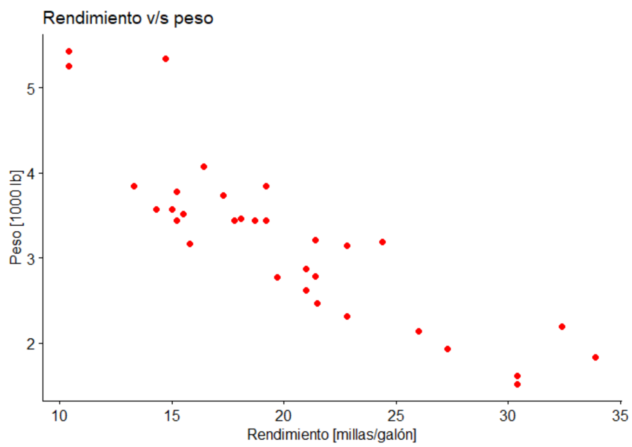


Figura 2.6: gráfico de dispersión para las variables `Rendimiento` y `Peso`.

Script 2.12: gráfico de dispersión para las variables `Rendimiento` y `Peso`.

```
library(ggpubr)

# Cargar datos.
datos <- read.csv2("C:/Inferencia/Mtcars.csv", stringsAsFactors = TRUE,
                   row.names = 1)

# Crear gráfico de dispersión.
g <- ggscatter(datos,
               x = "Rendimiento",
               y = "Peso",
               color = "red",
               title = "Rendimiento v/s peso",
               xlab = "Rendimiento [millas/galón]",
               ylab = "Peso [1000 lb]")

print(g)
```

Los gráficos de dispersión también son muy útiles para identificar si dos (o más) variables están relacionadas. La figura 2.7 (creada mediante el script 2.13) muestra tres gráficos de dispersión diferentes: en el de la izquierda, se aprecia que las variables `Peso` y `Cuarto_milla` son independientes, pues no hay una tendencia definida en la organización de los puntos. En el gráfico del centro, en cambio, podemos ver que la potencia tiende a aumentar a medida que también lo hace el peso, por lo que ambas variables están positivamente asociadas. Por último, el gráfico de la derecha nos muestra que las variables `Peso` y `Rendimiento` presentan asociación negativa, puesto que a medida que la primera aumenta, la segunda disminuye.

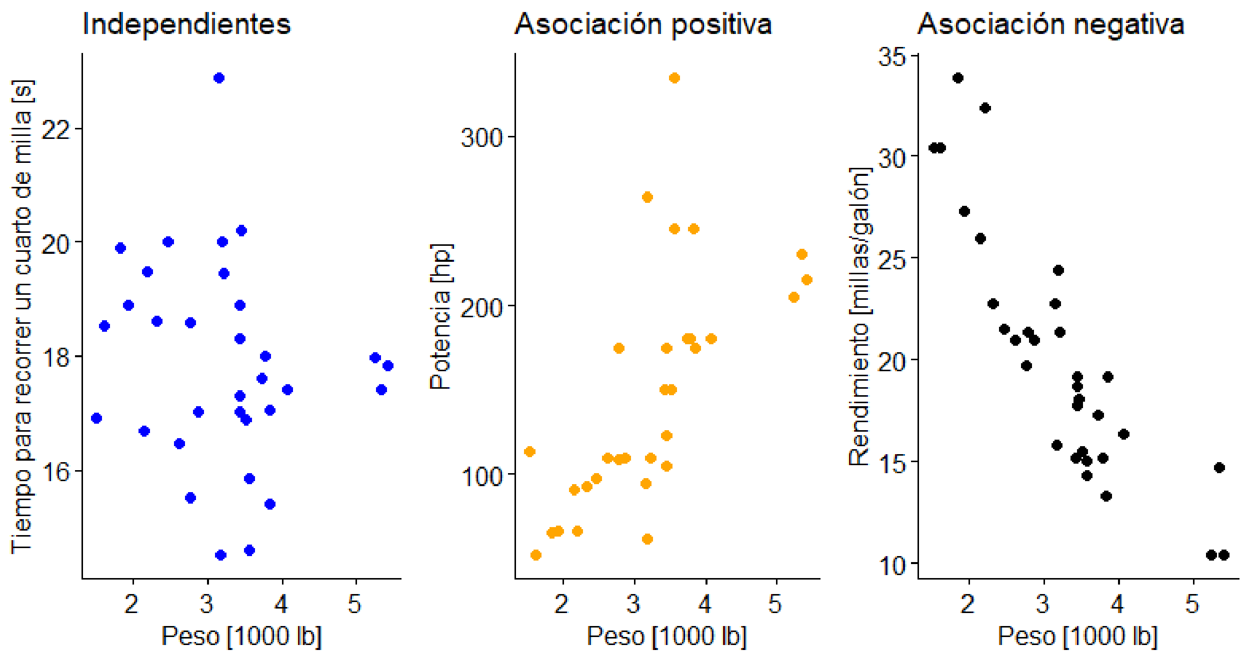


Figura 2.7: gráficos de dispersión con diferentes tipos de asociación entre las variables.

Script 2.13: gráficos de dispersión con diferentes tipos de asociación entre las variables.

```
library(ggpubr)

# Cargar datos.
datos <- read.csv2("C:/Inferencia/Mtcars.csv", stringsAsFactors = TRUE,
                   row.names = 1)

# Gráfico para variables independientes.
g1 <- ggscatter(datos,
                x = "Peso",
                y = "Cuarto_milla",
                color = "blue",
                title = "Independientes",
                xlab = "Peso [1000 lb]",
                ylab = "Tiempo para recorrer un cuarto de milla [s]")

# Gráfico para variables con asociación positiva.
g2 <- ggscatter(datos,
                x = "Peso",
                y = "Potencia",
                color = "orange",
                title = "Asociación positiva",
                xlab = "Peso [1000 lb]",
                ylab = "Potencia [hp]")

# Gráfico para variables con asociación negativa.
g3 <- ggscatter(datos,
                x = "Peso",
                y = "Rendimiento",
                color = "black",
                title = "Asociación negativa",
                xlab = "Peso [1000 lb]",
```

```
                 ylab = "Rendimiento [millas/galón]")

# Crear figura con tres gráficos.
g  <- ggarrange(g1 ,g2 ,g3, ncol = 3, nrow = 1, common.legend = TRUE)

print(g)
```

2.2.4 Dos variables categóricas

Similares al gráfico de barras para una variable categórica, los **gráficos de barras apiladas, agrupadas y estandarizadas** permiten visualizar la matriz de confusión entre dos variables y encontrar posibles relaciones entre ellas. La figura 2.8, creada con el script 2.14, ejemplifica esta familia de gráficos usando para ello las variables `Cambios` y `Motor`.

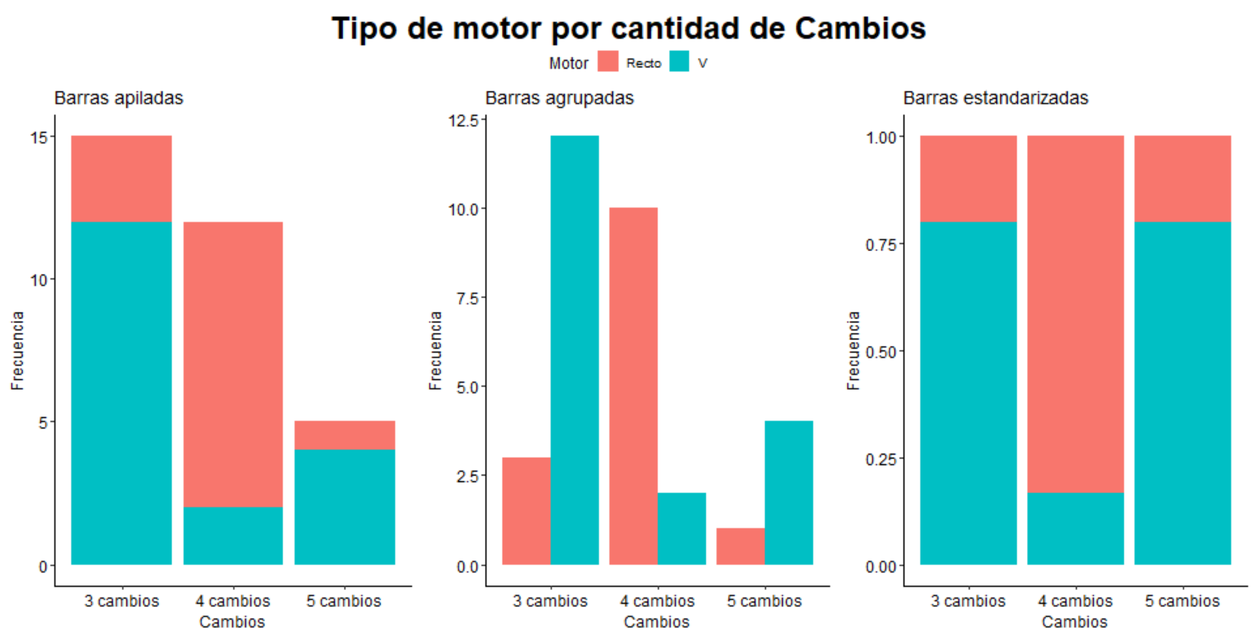


Figura 2.8: gráficos de barras para las variables `Cambios` y `Motor`.

El **gráfico de barras apiladas**, a la izquierda en la figura 2.8, muestra tres barras cuya altura corresponde a la frecuencia de la cantidad de cambios, al igual que en la figura 2.4, pero ahora cada barra está subdividida en secciones de distinto color para cada tipo de motor. La altura de cada sección está dada por la frecuencia del tipo de motor para la cantidad de cambios representada en la barra.

Similar al anterior, el gráfico de la derecha en la figura 2.8, que corresponde a un **gráfico de barras estandarizadas**, muestra barras de igual altura para cada cantidad de cambios representando claramente los cambios en la proporción de cada tipo de motor por la cantidad de cambios. Se puede apreciar que los automóviles con 3 y 5 cambios tienen mayoritariamente motores en forma de V, ambas en igual proporción, mientras que el uso de motores rectos se da principalmente en automóviles de 4 cambios.

El **gráfico de barras agrupadas**, al centro en la figura 2.8, es equivalente al de la izquierda, pero en lugar de dividir una barra en segmentos, muestra barras contiguas para cada tipo de motor.

Script 2.14: gráficos de barras para las variables `Cambios` y `Motor`.

```r
library(ggpubr)

# Cargar datos.
datos <- read.csv2("C:/Inferencia/Mtcars.csv", stringsAsFactors = TRUE,
                   row.names = 1)

# Crear tabla de contingencia para las variables Motor y Cambios,
# y guardarla como data frame.
tabla <- xtabs(~ Motor + Cambios, data = datos)
contingencia <- as.data.frame(tabla)

# Crear gráfico de barras segmentadas.
g1 <- ggplot(contingencia, aes(fill = Motor, y = Freq, x = Cambios))
g1 <- g1 + geom_bar(position = "stack", stat = "identity")
g1 <- g1 + labs(y = "Frecuencia") + ggtitle("Barras apiladas")
g1 <- g1 + theme_pubr()

# Crear gráfico de barras agrupadas.
g2 <- ggplot(contingencia, aes(fill = Motor, y = Freq, x = Cambios))
g2 <- g2 + geom_bar(position = "dodge", stat = "identity")
g2 <- g2 + labs(y = "Frecuencia") + ggtitle("Barras agrupadas")
g2 <- g2 + theme_pubr()

# # Crear gráfico de barras segmentadas estandarizado.
g3 <- ggplot(contingencia, aes(fill = Motor, y = Freq, x = Cambios))
g3 <- g3 + geom_bar(position = "fill", stat = "identity")
g3 <- g3 + labs(y = "Frecuencia") + ggtitle("Barras estandarizadas")
g3 <- g3 + theme_pubr()

# Crear una figura que contenga los tres gráficos.
g <- ggarrange(g1, g2, g3, nrow = 1, common.legend = TRUE)

# Agregar un título común en negrita y con fuente de 24 puntos.
titulo <- text_grob("Tipo de motor por cantidad de Cambios",
                    face = "bold", size = 24)

g <- annotate_figure(g, top = titulo)

# Guardar la figura en formato png con tamaño 960 x 480 pixeles.
ggexport(g, filename = "C:/Inferencia/f-barras-2.png",
         height = 480, width = 960)
```

Similar al gráfico de barras para dos variables, el **gráfico de mosaico** permite representar una tabla de contingencia. Para ello, divide un área en regiones y el área de cada región es proporcional al porcentaje de observaciones que representa. La figura 2.9, creada con el script 2.15 ejemplifica el uso de este tipo de gráficos, usando para ello las variables `Cambios` y `Motor`. En ella, el ancho de cada columna es proporcional a la cantidad de automóviles que tienen la correspondiente cantidad de cambios, mientras que la altura de cada barra de las columnas refleja la proporción de automóviles con un determinado tipo de motor.

Si nos fijamos bien en la figura 2.9, podemos ver claramente que los vehículos con 5 cambios son, por mucho, los menos frecuentes y que los con 3 cambios son algo más frecuentes que los que tienen 4 cambios. Del mismo modo, podemos ver que, para vehículos de 3 y 5 cambios, la proporción de vehículos con motor recto es la misma, y mucho menor que la de aquellos con motor en forma de V. Sin embargo, este último no es muy frecuente en automóviles con 4 cambios.

Cabe destacar que, para este tipo de gráfico, se requiere emplear el paquete `ggmosaic`.

Script 2.15: gráfico de mosaico para las variables `Cambios` y `Motor`.

```r
library(ggmosaic)
```

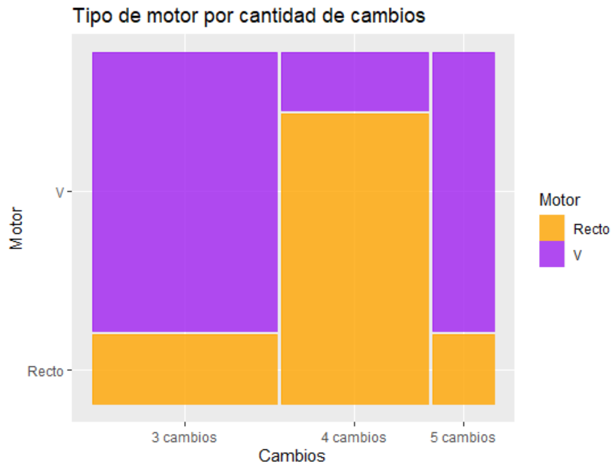


Figura 2.9: gráfico de mosaico para las variables `Cambios` y `Motor`.

```

# Cargar datos.
datos <- read.csv2("C:/Inferencia/Mtcars.csv", stringsAsFactors = TRUE,
                   row.names = 1)

# Crear tabla de contingencia para las variables gear y vs,
# y guardarla como data frame.
tabla <- xtabs(~ Cambios + Motor, data = datos)
contingencia <- as.data.frame(tabla)

# Crear gráfico de mosaico.
g <- ggplot(data = contingencia)
g <- g + geom_mosaic(aes(weight = Freq, x = product(Cambios), fill = Motor))

g <- g + labs(y = "Motor", x = "Cambios",
              title = "Tipo de motor por cantidad de cambios")

g <- g + scale_fill_manual(values=c("orange", "purple"))

print(g)
```

2.2.5 Una variable numérica y otra categórica

Desde luego, también es importante poder comparar diferentes grupos de observaciones de acuerdo a una característica categórica, para lo cual los gráficos pueden ser de gran ayuda. Por ejemplo, la figura 2.10, creada mediante el script 2.16 muestra un gráfico de cajas para la variable `Rendimiento` agrupada por el número de cambios de los automóviles.

Script 2.16: gráfico de cajas por grupo.

```
library(ggpubr)

# Cargar datos.
datos <- read.csv2("C:/Inferencia/Mtcars.csv", stringsAsFactors = TRUE,
```

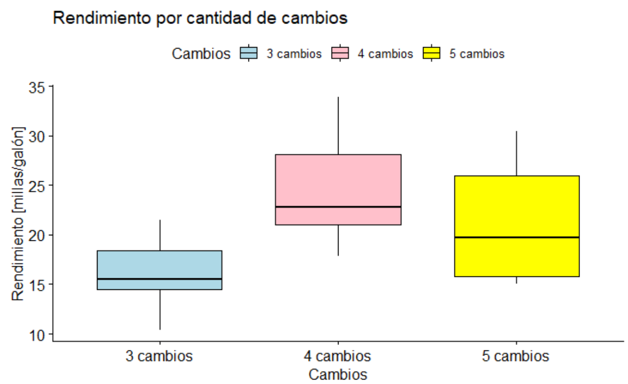


Figura 2.10: gráfico de cajas por grupo.

```
                     row.names = 1)

g <- ggboxplot(datos, x = "Cambios",
          y = "Rendimiento",
          palette = c("light blue", "pink", "yellow"),
          fill = "Cambios",
          title = "Rendimiento por cantidad de cambios",
          xlab = "Cambios",
          ylab = "Rendimiento [millas/galón]")

print(g)
```

Una buena alternativa, si la cantidad de observaciones es pequeña, es el **gráfico de tiras**, similar al gráfico de dispersión. El script 2.17 construye este gráfico para la variable `Rendimiento` agrupada según los niveles de la variable `Cambios`, obteniéndose como resultado la figura 2.11.

Script 2.17: gráfico de tiras.

```
library(ggpubr)

# Cargar datos.
datos <- read.csv2("C:/Inferencia/Mtcars.csv", stringsAsFactors = TRUE,
                     row.names = 1)

g <- ggstripchart(datos, x = "Cambios",
                  y = "Rendimiento",
                  palette = c("blue", "red", "dark green"),
                  color = "Cambios",
                  title = "Rendimiento por cantidad de cambios",
                  xlab = "Cambios",
                  ylab = "Rendimiento [millas/galón]")

print(g)
```

2.3 EJERCICIOS PROPUESTOS

1. ¿Cuándo es apropiado utilizar un gráfico de puntos para revisar datos?

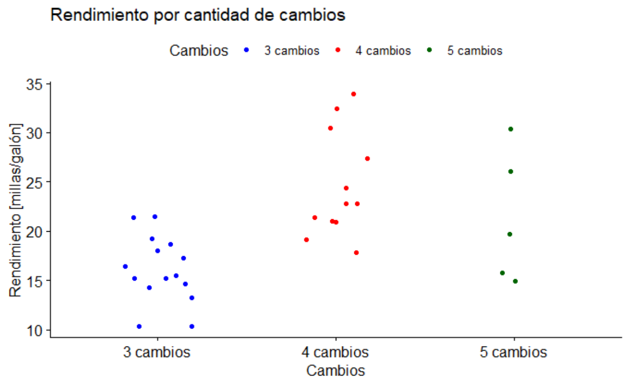
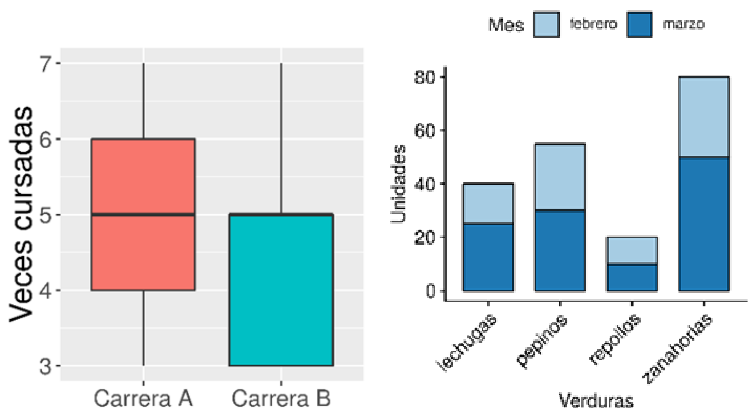


Figura 2.11: gráfico de tiras.

2. ¿Cuándo la mediana caracteriza mejor a un conjunto de datos que la media?
3. Da ejemplos de tres variables que posiblemente tengan una distribución simétrica, con asimetría positiva y con asimetría negativa, respectivamente. Justifique bien cada caso.
4. Describe un estudio en que posiblemente los datos recolectados tengan una distribución bimodal.
5. ¿Qué tipo de información buscada llevaría a utilizar un gráfico de dispersión?
6. ¿Por qué es importante conocer una medida de dispersión (variabilidad) de un conjunto de datos? Dé un ejemplo para clarificar su respuesta.
7. Considera la representación de la figura 2.12a de los datos obtenidos al tomar muestras aleatorias de estudiantes de dos carreras de la Facultad de Ingeniería para estudiar si el número de veces que se cursan las tres asignaturas de física en el Módulo Básico de Ingeniería depende de la carrera de los alumnos. Compara las distribuciones de ambos grupos. ¿En qué se parecen y en qué se diferencian?



(a) Ejercicio 7. (b) Ejercicio 8.

Figura 2.12: gráficos para los ejercicios propuestos.

8. El gráfico de la figura 2.12b muestra las unidades de verduras vendidas en uno de los kioskos de la Universidad durante los meses anteriores. Construye la tabla de contingencia correspondiente a los datos que se representan. ¿Qué mes tuvo mayores ventas? ¿En qué proporción?
9. ¿Cómo se puede generar la secuencia 0.00, 0.25, 0.50, ..., 2.75, 3.00 en R?
10. Resuelve en R los siguientes ejercicios. Considera para ello el conjunto de datos nativo de R `chickwts`.
    a) ¿Qué son los cuartiles y cómo se pueden obtener para los pesos de los pollitos reportados en la columna `weight`?
    b) ¿Cómo obtener los cuartiles del ejercicio anterior por cada tipo de alimento en la columna `feed`?
    c) ¿Cómo obtener un histograma de los pesos de los pollitos?

d) ¿Cómo se obtiene un gráfico de cajas para comparar los pesos de los pollitos por tipo de alimento suministrado?

11. Investiga acerca del uso del paquete de R `ggplot2` para la creación de gráficos.

CAPÍTULO 3. VARIABLES ALEATORIAS Y DISTRIBUCIONES DE PROBABILIDAD

Los conceptos que estudiaremos en este capítulo pueden resultar algo difíciles de entender, por lo que si necesitas más material, puedes consultar las fuentes en que se basa este capítulo: Diez y col. (2017, pp. 104-157) y Freund y Wilson (2003, pp. 104-106).

Definimos como **variable aleatoria** una variable o un proceso cuyo resultado sea numérico. Dichas variables se nombran con letras mayúsculas, y denotamos sus posibles valores por la letra minúscula correspondiente, acompañada de un subíndice. Las variables aleatorias tienen una **distribución de probabilidad**, la cual define la probabilidad de que ocurran los diferentes valores que dicha variable puede tomar.

3.1 VARIABLES ALEATORIAS

La definición de **variable aleatoria continua** es en realidad bastante sencilla: es una variable que puede tomar cualquiera de los infinitos valores posibles dentro de un intervalo.

Una **variable aleatoria discreta**, en cambio, solo puede tomar un conjunto finito de valores. Un ejemplo típico de variable aleatoria puede ser el lanzamiento de un dado. Si el dado está bien balanceado, tendremos igual probabilidad de obtener cualquiera de las caras. Pero es sabido que algunos tramposos fabrican dados adulterados para favorecer, por ejemplo, la obtención de valores 1 y 6. Una distribución aleatoria de la variable lanzamiento de un dado adulterado (X) podría ser la que se presenta en la tabla 3.1.

i	1	2	3	4	5	6	Total
x_i	1	2	3	4	5	6	-
$P(X = x_i)$	0.250	0.125	0.125	0.125	0.125	0.250	1.000

Tabla 3.1: distribución de probabilidad para el lanzamiento de un dado adulterado.

El **valor esperado**, denotado como $E(X)$ o μ, corresponde al resultado promedio de una variable aleatoria. Para una variable aleatoria discreta, se calcula sumando los valores posibles ponderados por su probabilidad, como muestra la ecuación 3.1.

$$E(X) \equiv \mu = \sum_{i=1}^{n} x_i P(X = x_i) \tag{3.1}$$

También podemos calcular qué tan alejado podría estar el valor obtenido del valor esperado por medio de la varianza general, denotada por $Var(X)$ o σ^2, que se calcula como la media de los cuadrados de la diferencia con respecto a la media ponderada según la probabilidad de ocurrencia, como muestra la ecuación 3.2. Una vez más, la desviación estándar corresponde a la raíz cuadrada de la varianza.

$$Var(X) \equiv \sigma^2 = \sum_{i=1}^{n} (x_i - \mu)^2 P(X = x_i) \tag{3.2}$$

En R, el paquete `DiscreteRV` permite trabajar con variables aleatorias discretas, como se ejemplifica en el script 3.1 (Cross, 2017).

Script 3.1: variables aleatorias discretas en R.

```r
library(discreteRV)

# Crear una variable discreta para representar el dado
# adulterado de la tabla 3.1.
resultados <- 1:6
probabilidades = c(0.25, 0.125, 0.125, 0.125, 0.125, 0.25)
X <- RV(outcomes = resultados, probs = probabilidades)

# Calcular el valor esperado.
esperado <- E(X)
cat("Valor esperado:", esperado, "\n")

# Calcular la varianza.
varianza <- V(X)
cat("Varianza:", varianza, "\n")

# Calcular la desviación estándar.
desviacion <- SD(X)
cat("Desviación estándar:", desviacion, "\n")
```

Para ayudarnos a entender mejor la noción de distribución de probabilidad, veamos la figura 3.1 (obtenida mediante el script 3.2). Ella nos muestra, de izquierda a derecha, las distribuciones de probabilidad para el puntaje total obtenido al lanzar 5, 10 y 20 dados, respectivamente.

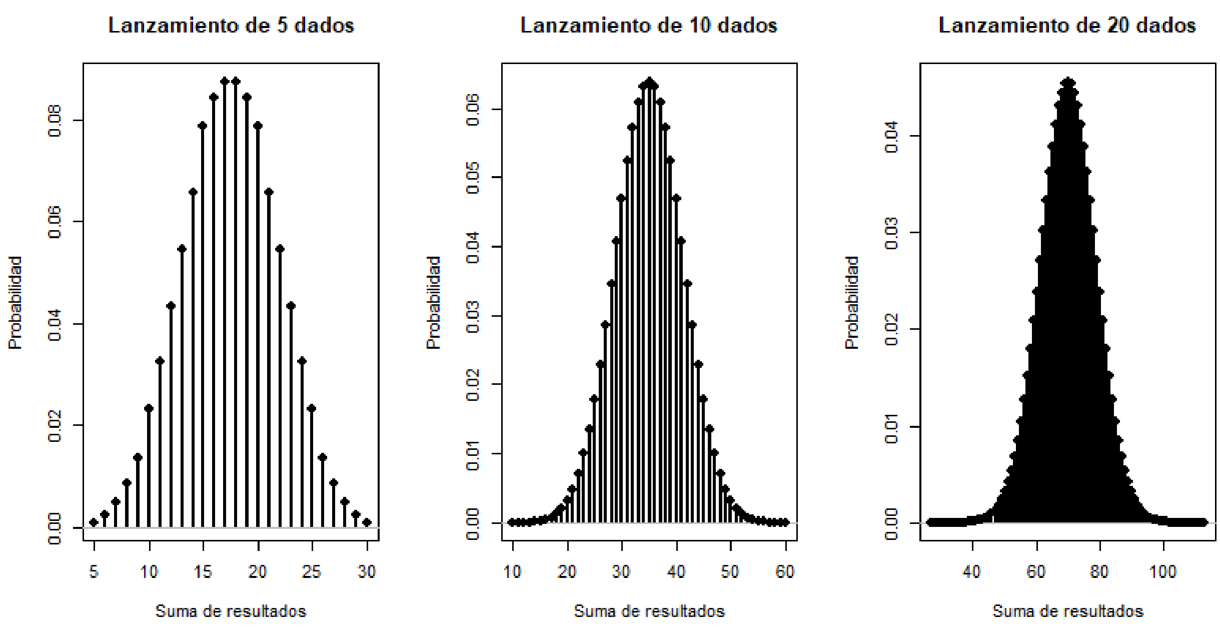


Figura 3.1: distribución de probabilidad para varios lanzamientos de un dado cargado.

Script 3.2: histogramas de variables aleatorias discretas en R.

```r
library(discreteRV)
library(ggpubr)

# Crear una variable discreta para representar el dado
# adulterado de la tabla 4.1.
resultados <- 1:6
```

```r
probabilidades = c(0.25, 0.125, 0.125, 0.125, 0.125, 0.25)
X <- RV(outcomes = resultados, probs = probabilidades)

# Crear vector con los resultados de 5 lanzamientos del dado.
lanzar_5 <- SofIID(X, n=5)

# Crear vector con los resultados de 10 lanzamientos del dado.
lanzar_10 <- SofIID(X, n=10)

# Crear vector con los resultados de 20 lanzamientos del dado.
lanzar_20 <- SofIID(X, n=20)

# Graficar los resultados.
par(mfrow=c(1, 3))

plot(lanzar_5,
     main="Lanzamiento de 5 dados",
     xlab="Suma de resultados",
     ylab="Probabilidad")

plot(lanzar_10,
     main="Lanzamiento de 10 dados",
     xlab="Suma de resultados",
     ylab="Probabilidad")

plot(lanzar_20,
     main="Lanzamiento de 20 dados",
     xlab="Suma de resultados",
     ylab="Probabilidad")
```

Conocer la distribución de probabilidad de una variable discreta nos ayuda a hacer estimaciones útiles. A modo de ejemplo, supongamos que un ingeniero de software debe crear un programa que resuelva un problema (siempre con instancias del mismo tamaño) con un tiempo de respuesta no mayor a 25 segundos. El histograma de la figura 3.2 muestra los tiempos de ejecución obtenidos para 500 pruebas de la solución propuesta, donde se observa que 30 de ellas tardaron en realidad más de 25 segundos, con un rango que va de 0 a 30 segundos. Así, podemos estimar la probabilidad de que el tiempo de ejecución sea mayor a 25 segundos dividiendo la cantidad de observaciones que cumplen este criterio por la cantidad total de instancias, como muestra la ecuación 3.3.

$$P(X > 25) = \frac{30}{500} = 0.06 \tag{3.3}$$

Frecuentemente resulta más adecuado expresar o modelar un fenómeno como una combinación de dos o más variables aleatorias. Por ejemplo, un jugador de baloncesto puede anotar canastas de uno, dos o tres puntos dependiendo de si encesta con un tiro libre, un lanzamiento desde dentro del área o desde fuera del área. Así, se tienen tres variables aleatorias:

1. X: Anotaciones por tiro libre.
2. Y: Anotaciones desde dentro del área.
3. Z: Anotaciones desde fuera del área.

Podemos representar el total de puntos anotados por el jugador como la suma de los puntos anotados de las tres formas posibles, lo que corresponde a una **combinación lineal** de las variables X, Y y Z. La fórmula general de una combinación lineal de n variables está dada por la ecuación 3.4, donde cada X_i corresponde a una variable aleatoria y cada c_i es una constante conocida.

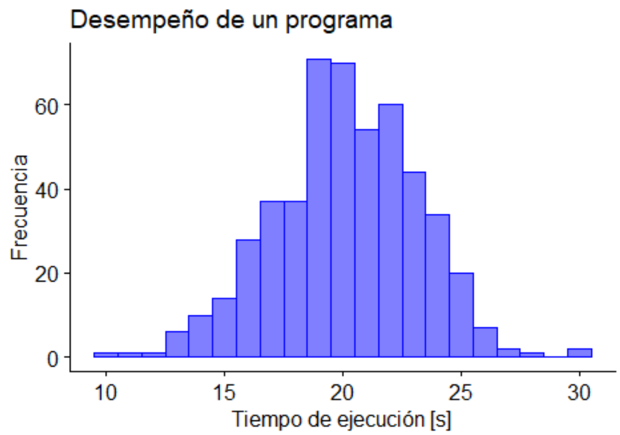


Figura 3.2: histograma para el desempeño del programa.

$$\sum_{i=1}^{n} c_i X_i \tag{3.4}$$

Cuando las variables de una combinación lineal son independientes[1], podemos calcular el valor esperado y la varianza de la combinación lineal usando las ecuaciones 3.5 y 3.6. Una vez más, la desviación estándar está dada por la raíz cuadrada de la varianza.

$$E\left(\sum_{i=1}^{n} c_i X_i\right) = \sum_{i=1}^{n} c_i E(X_i) \tag{3.5}$$

$$Var\left(\sum_{i=1}^{n} c_i X_i\right) = \sum_{i=1}^{n} c_i^2 Var(X_i) \tag{3.6}$$

Por supuesto, en R también podemos trabajar con combinaciones lineales de variables aleatorias discretas, como muestra el script 3.3.

Script 3.3: combinación lineal de variables aleatorias discretas en R.

```
library(discreteRV)

# Crear una variable discreta para representar el dado adulterado de la tabla
# 3.1, y calcular su valor esperado, varianza y desviación estándar.
resultados <- 1:6
probabilidades = c(0.25, 0.125, 0.125, 0.125, 0.125, 0.25)
X <- RV(outcomes = resultados, probs = probabilidades)
esperado_x <- E(X)
varianza_x <- V(X)
desviacion_x <- SD(X)
cat("E(X):", esperado_x, "\n")
cat("V(X):", varianza_x, "\n")
cat("SD(X):", desviacion_x, "\n\n")

# Crear una variable aleatoria para un dado balanceado, y calcular su valor
# esperado, varianza y desviación estándar.
Y <- RV(outcomes = resultados, probs = 1/6)
```

[1] Si las variables no son independientes, se requieren métodos más complejos fuera del alcance de este libro.

```
esperado_y <- E(Y)
varianza_y <- V(Y)
desviacion_y <- SD(Y)
cat("E(Y):", esperado_y, "\n")
cat("V(Y):", varianza_y, "\n")
cat("SD(Y):", desviacion_y, "\n\n")

# Crear una combinación lineal de variables aleatorias, y calcular su valor
# esperado, varianza y desviación estándar.
Z <- 0.5 * X + 0.5 * Y
esperado_z <- E(Z)
varianza_z <- V(Z)
desviacion_z <- SD(Z)
cat("E(Z):", esperado_z, "\n")
cat("V(Z):", varianza_z, "\n")
cat("SD(Z):", desviacion_z)
```

Al examinar con mayor detención los gráficos de la figura 3.1 podemos apreciar que, a medida que se efectúan más lanzamientos del dado, el histograma se asemeja cada vez más a una curva continua, la cual recibe el nombre de **función de densidad de probabilidad**, o simplemente **distribución** o **densidad**.

Las distribuciones tienen la propiedad de que el área total bajo la curva siempre es 1, lo que resulta muy útil al momento de calcular probabilidades, pues basta con calcular el área bajo la curva del segmento deseado. Volviendo al ejemplo del desempeño del programa, presentado en la figura 3.2, el tiempo de ejecución es en realidad una variable continua. Así, la probabilidad de que el tiempo de ejecución sea mayor a 25 segundos corresponde al área coloreada en el gráfico de la figura 3.3, con un valor de 0,048[2].

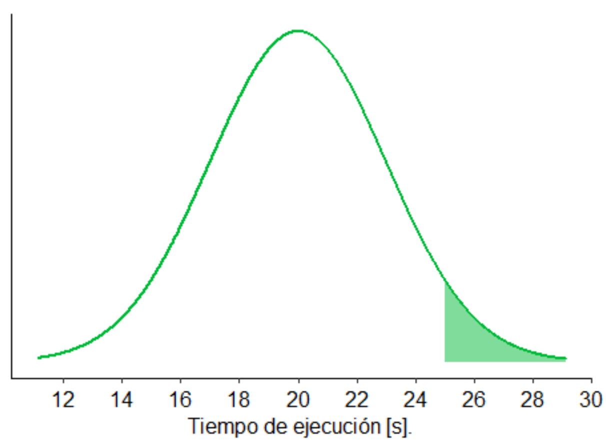


Figura 3.3: distribución para el desempeño del programa.

3.2 DISTRIBUCIONES CONTINUAS

Existen múltiples funciones de distribución continua que son de uso frecuente en estadística, las cuales se describen a continuación.

[2] El cálculo de esta probabilidad se aborda en el siguiente apartado

3.2.1 Distribución normal

También conocida como **distribución gaussiana**, la **distribución normal** es la más ampliamente empleada en estadística, pues muchas variables se acercan a esta distribución. Se caracteriza por ser unimodal y simétrica, con forma de campana. La figura 3.3 ejemplifica esta distribución.

La distribución normal se usa para modelar diversos fenómenos y podemos ajustarla mediante dos parámetros:

- μ: la media, que desplaza el centro de la curva a lo largo del eje x.
- σ: la desviación estándar, que modifica qué tan dispersos están los datos con respecto a la media.

Así, denotamos este tipo de distribución por $N(\mu, \sigma)$. La figura 3.4, creada mediante el script 3.4, muestra dos ejemplos superpuestos de distribución normal: $N(\mu = 0, \sigma = 1)$ en azul y $N(\mu = 10, \sigma = 6)$ en rojo.

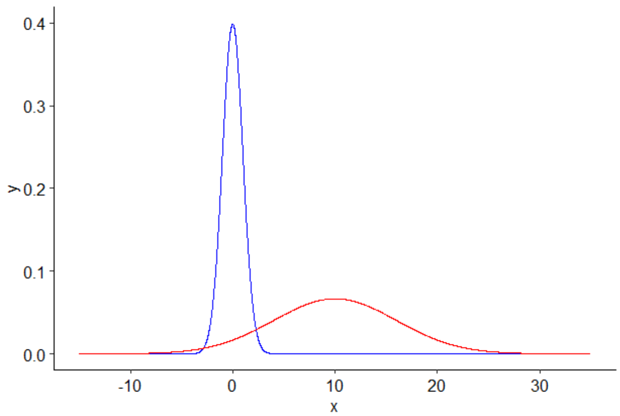


Figura 3.4: dos ejemplos superpuestos de distribución normal.

Script 3.4: graficando dos ejemplos de distribución normal.

```r
library(ggpubr)

# Generar valores para una distribución normal con media 0 y
# desviación estándar 1.
media <- 0
desv_est <- 1
x <- seq(-15, 35, 0.01)
y <- dnorm(x, mean = media, sd = desv_est)
normal_1 <- data.frame(x, y)

# Repetir el proceso para una distribución normal con media 10
# y desviación estándar 6.
media <- 10
desv_est <- 6
x <- seq(-15, 35, 0.01)
y <- dnorm(x, mean = media, sd = desv_est)
normal_2 <- data.frame(x, y)

# Graficar ambas distribuciones.
g <- ggplot(normal_1, aes(x, y)) + geom_line(color = "blue")
g <- g + geom_line(data = normal_2, color = "red")
g <- g + theme_pubr()

print(g)
```

Antes de continuar, fijémonos en las líneas 8 y 16 del script 3.4, donde se usa la función `dnorm(x, mean, sd)`. Esta función calcula la densidad de una distribución normal. Además de `dnorm()`, R nos ofrece otras funciones que también resultan de mucha ayuda:

- `pnorm(q, mean, sd, lower.tail)`: permite encontrar percentiles, los cuales corresponden a la **función de distribución acumulada** (es decir, la probabilidad de que la variable tome valores menores o iguales que un valor dado), a partir de las probabilidades.
- `qnorm(p, mean, sd, lower.tail)`: encuentra el percentil para las probabilidades dadas en `p`, por lo que es la función inversa de `pnorm()`.
- `rnorm(n, mean, sd)`: genera aleatoriamente `n` observaciones de la distribución normal especificada.

Los argumentos de esta familia de funciones son:

- `x, q`: vector de cuantiles (percentiles).
- `p`: vector de probabilidades.
- `mean`: media de la distribución normal.
- `sd`: desviación estándar de la distribución normal.
- `lower.tail`: valor lógico que señala cuál de los dos extremos o colas de la distribución emplear.
- `n`: tamaño del vector resultante.

Es importante señalar que, por defecto, `lower.tail` toma el valor verdadero, con lo que `pnorm()` y `qnorm()` operan con la cola inferior de la distribución. Si, en cambio, `lower.tail = FALSE`, dichas funciones operan con la cola superior (es decir, `pnorm()` nos entrega la probabilidad de que la variable tome valores mayores que un valor dado).

Una **regla empírica** muy útil al momento de trabajar con distribuciones normales es la llamada regla 68-95-99.7, ilustrada en la figura 3.5, la cual establece que:

- Cerca de 68 % de las observaciones se encuentran a una distancia de una desviación estándar de la media.
- Alrededor de 95 % de las observaciones se encuentran a una distancia de dos desviación estándar de la media.
- Aproximadamente 99.7 % de las observaciones se encuentran a una distancia de tres desviación estándar de la media.

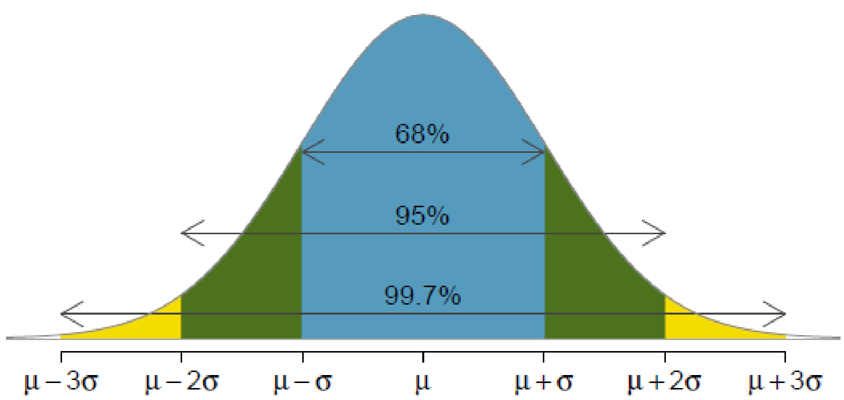


Figura 3.5: regla empírica de la distribución normal. Fuente: Diez y col. (2017, p. 136).

Muchas pruebas estadísticas operan bajo el supuesto de que los datos siguen una distribución normal. Como se insinuó en párrafos anteriores, la normalidad es siempre una aproximación, por lo que debemos verificar que el supuesto de una distribución normal sea aceptable. Una buena herramienta para ello es el **gráfico cuantil-cuantil**, también llamado **gráfico Q-Q**, que se muestra en la figura 3.6 y que podemos construir en R como muestra el script 3.5. En él podemos distinguir los siguientes elementos: un grupo de puntos, una recta y una región coloreada. Los puntos corresponden a las observaciones, mientras que la recta representa la distribución normal. En consecuencia, mientras más se asemeje el patrón que forman los puntos a la recta, más parecida será la distribución a la normal. La banda coloreada establece el margen aceptable para suponer normalidad en el conjunto de datos. Así, para el conjunto de datos de la figura 3.6 sería imprudente aceptar el supuesto de normalidad.

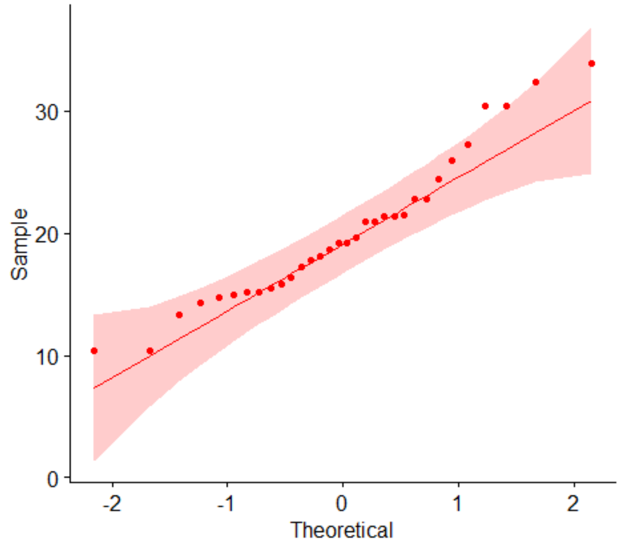


Figura 3.6: gráfico cuantil-cuantil.

Script 3.5: creación de un gráfico cuantil-cuantil.

```
library(ggpubr)

# Cargar datos.
datos <- read.csv2("C:/Inferencia/Mtcars.csv", stringsAsFactors = TRUE,
                   row.names = 1)

# Gráfico Q-Q para la variable Rendimiento.
g <- ggqqplot(datos,
              x = "Rendimiento",
              color = "red")

print(g)
```

3.2.2 Distribución Z

Al trabajar con distribuciones, especialmente las simétricas, a menudo usaremos **técnicas de estandarización** para determinar qué tan usual o inusual es un determinado valor en una escala única. Así, para la distribución normal usamos como estandarización la **distribución Z** o **distribución normal estándar**, que no es más que una distribución normal centrada en 0 y con desviación estándar 1, que podemos obtener de manera bastante sencilla como muestra la ecuación 3.7.

$$Z = \frac{x - \mu}{\sigma} \tag{3.7}$$

Al aplicar la ecuación 3.7 a una observación x en una distribución normal obtenemos, entonces, su **valor**

z, que determina cuán por encima o por debajo de la media (en términos de la desviación estándar) se encuentra dicha observación x. Así, observaciones cuyos valores z sean negativos estarán por debajo de la media. Análogamente, un valor Z positivo indica que la observación está por sobre la media. Mientras mayor sea el valor absoluto de su valor z ($|z|$), más inusual será la observación.

3.2.3 Distribución chi-cuadrado

También llamada **ji-cuadrado** o χ^2, la distribución **chi-cuadrado** (Devore, 2008) se usa para caracterizar valores siempre positivos y habitualmente desviados a la derecha. El único parámetro de esta distribución corresponde a los **grados de libertad**, usualmente representada por la letra griega ν, que son una estimación de la cantidad de observaciones empleadas para calcular un estimador. Otra forma de entender esta idea es como la cantidad de valores que pueden cambiar libremente en un conjunto de datos. Como ejemplo, supongamos que necesitamos una muestra de tres elementos cuya media sea 10. Una vez escogidos los primeros dos, solo queda una posibilidad para el tercero de modo que se cumpla con la media deseada. Así, solo los dos primeros valores pueden cambiar libremente, por lo que se tienen dos grados de libertad.

Esta distribución está relacionada con la ya conocida distribución Z, pues si sumamos los cuadrados de k variables aleatorias independientes que siguen una distribución Z, dicha suma sigue una distribución χ^2 con k grados de libertad:

$$\sum_{i=1}^{k} Z_i^2 \sim \chi^2(\nu = k) \tag{3.8}$$

La media de la distribución χ^2 es $\mu = \nu$, y su desviación estándar, $\sigma = 2\nu$.

Las funciones de R para esta distribución, similares a las descritas para la distribución normal, son:

- `dchisq(x, df)`.
- `pchisq(q, df, lower.tail)`.
- `qchisq(p, df, lower.tail)`.
- `rchisq(n, df)`.

Donde:

- `x, q` son vectores de cuantiles (enteros no negativos).
- `p` es un vector de probabilidades.
- `n` es la cantidad de observaciones.
- `df` son los grados de libertad.
- `lower.tail` es análogo al de la función `pnorm`.

La figura 3.7 muestra un ejemplo de la distribución χ^2.

3.2.4 Distribución t de Student

Ampliamente empleada cuando se trabaja con muestras pequeñas, la **distribución t de Student**, o simplemente **distribución t**, tiene, al igual que la distribución χ^2, los grados de libertad como único parámetro. A

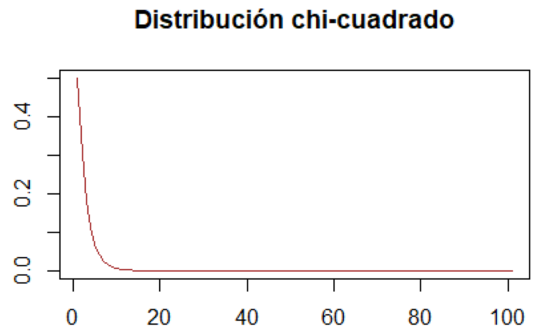


Figura 3.7: ejemplo de distribución χ^2 con 2 grados de libertad.

medida que los grados de libertad aumentan, esta distribución se asemeja cada vez más a la normal, aunque sus colas son más gruesas, como ilustra la figura 3.8.

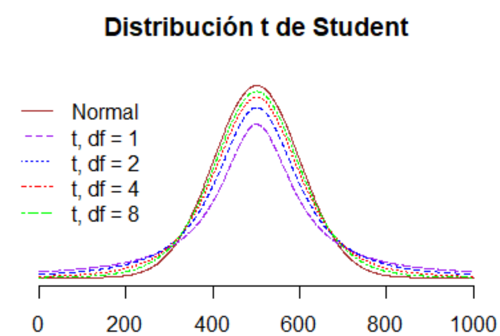


Figura 3.8: ejemplo de distribuciones t.

La distribución t se encuentra relacionada con las distribuciones vistas anteriormente de acuerdo a la ecuación 3.9, donde Z es una distribución normal estándar y $\chi^2(\nu)$ es una distribución χ^2 con ν grados de libertad.

$$Z\sqrt{\frac{v}{\chi^2(v)}} \sim t(\nu) \tag{3.9}$$

La media de la distribución t, para $\nu > 1$, es $\mu = 0$. Su desviación estándar, para $\nu > 2$, está dada por la ecuación 3.10.

$$\sigma = \sqrt{\frac{\nu}{\nu - 2}} \tag{3.10}$$

Las funciones de R para esta distribución, cuyos argumentos son análogos a los que hemos visto para las distribuciones anteriores, son:

- `dt(x, df)`.
- `pt(q, df, lower.tail)`.

- `qt(p, df, lower.tail)`.
- `rt(n, df)`.

3.2.5 Distribución F

Otra distribución que usaremos a lo largo de este libro es la **distribución F**, ampliamente usada para comparar varianzas. La distribución F se relaciona con las anteriores de acuerdo a la ecuación 3.11, donde $\chi_1^2(\nu_1)$ y $\chi_2^2(\nu_2)$ son dos distribuciones χ^2 con ν_1 y ν_2 grados de libertad, respectivamente. Un ejemplo de una distribución F se puede encontrar en la figura 3.9.

$$\frac{\frac{X_1^2(\nu_1)}{\nu_1}}{\frac{X_2^2(\nu_2)}{\nu_2}} \sim F(\nu_1, \nu_2) \tag{3.11}$$

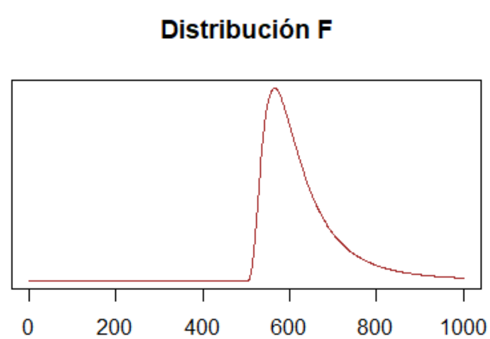


Figura 3.9: ejemplo de una distribución F.

Para $\nu_2 > 2$, la media de esta distribución está dada por la ecuación 3.12, y la desviación estándar corresponde a la ecuación 3.13 para $\nu_2 > 4$.

$$\mu = \frac{\nu_2}{\nu_2 - 2} \tag{3.12}$$

$$\sigma = \sqrt{\frac{2\nu_2^2(\nu_1 + \nu_2 - 2)}{\nu_1(\nu_2 - 2)^2(\nu_2 - 4)}} \tag{3.13}$$

Las funciones de R para esta distribución son:

- `df(x, df1, df2)`.
- `pf(q, df1, df2, lower.tail)`.
- `qf(p, df1, df2, lower.tail)`.
- `rf(n, df1, df2)`.

Donde `df1` como `df2` corresponden a grados de libertad y los argumentos restantes son los mismos que ya hemos visto anteriormente.

3.3 DISTRIBUCIONES DISCRETAS

Al igual que con las variables aleatorias continuas, también existen diversas distribuciones discretas de uso frecuente en estadística.

3.3.1 Distribución de Bernoulli

Una **variable aleatoria de Bernoulli** es aquella en que cada intento individual tiene solo dos resultados posibles: "éxito", que ocurre con una probabilidad p y se representa habitualmente con un 1, y "fracaso", que ocurre con probabilidad $q = 1 - p$ y suele representarse por un 0. La selección de qué resultado se considera como éxito o fracaso suele ser arbitraria. Para ilustrar esta idea, si dos personas lanzan una moneda al aire para sortear al ganador, cada una de ellas considerará una cara diferente de la moneda como un éxito.

Otro ejemplo que nos puede ayudar es el de lanzar varios dados de 20 caras, donde el éxito corresponda a obtener un 20 como resultado. Cada uno de ellos tiene una **probabilidad de éxito** (obtener 20) $p = 0.05$ y una **probabilidad de fracaso** (obtener otro valor) $q = 1 - p = 0.95$. Los lanzamientos de los dados son **independientes**, pues un dado no afecta a los demás.

Definimos la **proporción de la muestra** para una distribución de Bernoulli, $\hat{p}$, como la cantidad de éxitos dividida por la cantidad de intentos. Mientras mayor sea la cantidad de intentos, más cercano será el valor de $\hat{p}$ a la probabilidad real de éxito p.

Al igual que la distribución normal, la distribución de Bernoulli puede resumirse expresando su media ($\mu = p$) y su desviación estándar. Esta última está dada por la ecuación 3.14.

$$\sigma = \sqrt{p(1-p)} \tag{3.14}$$

El paquete `extraDistr` de R ofrece 4 funciones, similares a las ya conocidas, para la distrubución de Bernoulli:

- `dbern(x, prob)`.
- `pbern(q, prob, lower.tail)`.
- `qbern(p, pro, lower.tail)`.
- `rbern(n, prob)`.

3.3.2 Distribución geométrica

La **distribución geométrica** describe la cantidad de intentos que debemos realizar hasta obtener un éxito para variables de Bernoulli **independientes e idénticamente distribuidas**, es decir, que no se afectan unas a otras y cada una con igual probabilidad de éxito.

La probabilidad de obtener un éxito al n-ésimo intento está dada por la ecuación 3.15, donde podemos ver que las probabilidades en esta distribución decrecen exponencialmente rápido, como ilustra la figura 3.10. La media y la desviación estándar de la distribución geométrica están dadas, respectivamente, por las ecuaciones 3.16 y 3.17.

$$\Pr(\text{primer éxito al } n\text{-ésimo intento}) = (1-p)^{n-1}p \tag{3.15}$$

$$\mu = \frac{1}{p} \tag{3.16}$$

$$\sigma = \sqrt{\frac{1-p}{p^2}} \tag{3.17}$$

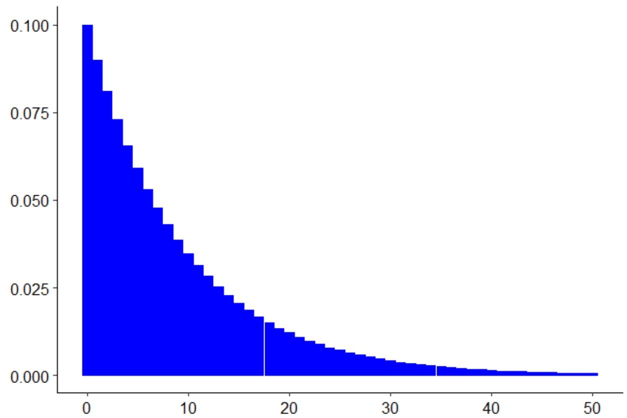


Figura 3.10: distribución geométrica para obtener un valor específico lanzando un dado de 20 caras.

Para entender mejor la utilidad de la distribución geométrica, consideremos la pregunta: ¿cuántas veces tenemos que lanzar un dado de 20 caras para obtener un 1? Anteriormente vimos que la probabilidad de éxito en este caso es $p = 0.05$. El valor esperado, representado por la media, sería en este caso el que se presenta en la ecuación 3.18.

$$\mu = \frac{1}{p} = \frac{1}{0.05} = 20 \tag{3.18}$$

Una vez más, R ofrece funciones similares a las presentadas anteriormente para trabajar con distribuciones geométricas:

- `dgeom(x, prob)`.
- `pgeom(q, prob, lower.tail)`.
- `qgeom(p, prob, lower.tail)`.
- `rbern(n, prob)`.

3.3.3 Distribución binomial

A diferencia de la distribución geométrica, la **distribución binomial** describe la probabilidad de tener exactamente k éxitos en n intentos independientes de Bernoulli con probabilidad de éxito p, cuya función de probabilidad está dada por la ecuación 3.19, donde:

- $\binom{n}{k} = \frac{n!}{k!(n-k)!}$ corresponde a la cantidad de formas de obtener k éxitos en un total de n intentos.
- $p^k(1-p)^{n-k}$ es la probabilidad de tener un único éxito en solo una de las $\binom{n}{k}$ maneras posibles.

.

$$f(k,n,p) = \Pr(k;n,p) = \Pr(X = k) = \binom{n}{k} p^k (1-p)^{n-k} \tag{3.19}$$

La media y la desviación estándar de la distribución binomial están dadas por las ecuaciones 3.20 y 3.21, respectivamente. Un ejemplo de esta distribución se presenta en la figura 3.11

$$\mu = np \tag{3.20}$$

$$\sigma = \sqrt{np(1-p)} \tag{3.21}$$

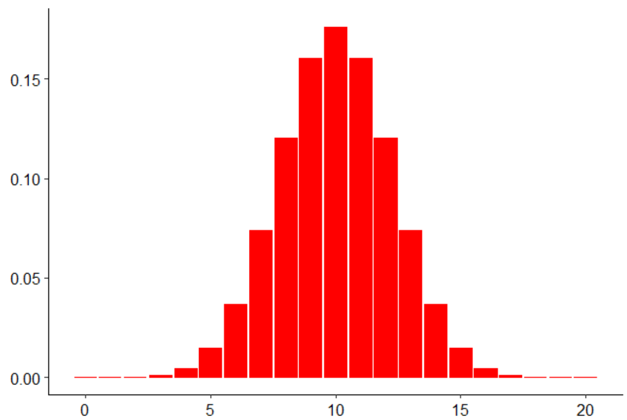


Figura 3.11: distribución binomial con $\mu = 400$ y $\sigma = 15.4019$.

Antes de decidir usar la distribución binomial, tenemos que verificar cuatro condiciones:

1. Los intentos son independientes.
2. La cantidad de intentos (n) es fija.
3. El resultado de cada intento puede ser clasificado como éxito o fracaso.
4. La probabilidad de éxito (p) es la misma para cada intento.

Las funciones que ofrece R para trabajar con esta distribución son:

- `dbinom(x, size, prob)`.
- `pbinom(x, size, prob)`.
- `qbinom(p, size, prob)`.
- `rbinom(n, size, prob)`.

Donde:

- `x` es un vector numérico.
- `p` es un vector de probabilidades.
- `n` es la cantidad de observaciones.
- `size` corresponde al número de intentos.
- `prob` es la probabilidad de éxito de cada intento.

En la figura 3.11 podemos observar que, en cierto modo, la distribución binomial se asemeja a la distribución normal: ambas son simétricas, aunque la distribución binomial no tiene la forma de campana de la distribución normal. Esta similitud ofrece una importante ventaja, pues en ocasiones es posible usar la distribución normal para estimar probabilidades binomiales, evitando así el uso de la compleja fórmula de la ecuación 3.19. Formalmente, esta aproximación es válida cuando el tamaño de la muestra, n, es lo suficientemente grande para que tanto np como $n(1-p)$ sean mayores o iguales que 10. En este caso, los parámetros de la distribución normal aproximada son los mismos de la distribución binomial (ecuaciones 3.20 y 3.21).

3.3.4 Distribución binomial negativa

La **distribución binomial negativa** es algo más general que la binomial, pues describe la probabilidad de encontrar el k-ésimo éxito al n-ésimo intento. Como señalan Diez y col. (2017, p. 155), "en el caso binomial, en general se tiene una cantidad fija de intentos y se considera la cantidad de éxitos. En el caso binomial negativo, se examina cuántos intentos se necesitan para observar una cantidad fija de éxitos y se requiere que la última observación sea un éxito"[3].

Como adelanta la comparación anterior, antes de decidir usar la distribución binomial negativa tenemos que verificar cuatro condiciones:

1. Los intentos son independientes.
2. El resultado de cada intento puede ser clasificado como éxito o fracaso.
3. La probabilidad de éxito (p) es la misma para cada intento.
4. El último intento debe ser un éxito.

La función de probabilidad para esta distribución, ejemplificada en la figura 3.12, está dada por la ecuación 3.22. La varianza y la desviación estándar están dadas por las ecuaciones 3.23 y 3.24 (Devore, 2008, p. 120).

$$\Pr(k\text{-ésimo éxito al } n\text{-ésimo intento}) = \binom{n-1}{k-1} p^k (1-p)^{n-k} \tag{3.22}$$

$$\mu = \frac{k(1-p)}{p} \tag{3.23}$$

$$\sigma = \sqrt{\frac{k(1-p)}{p^2}} \tag{3.24}$$

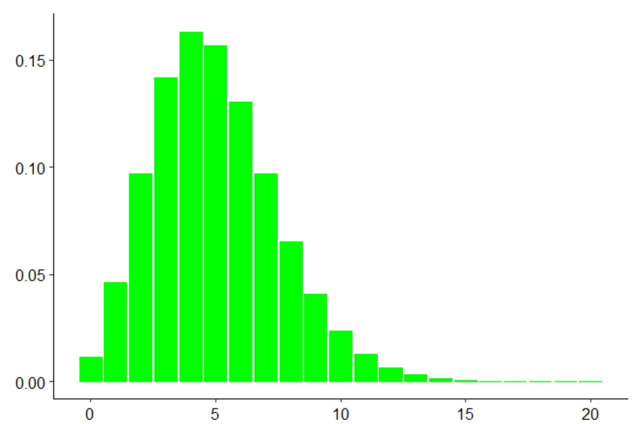


Figura 3.12: ejemplo de distribución binomial negativa.

Nuevamente, R dispone de cuatro funciones que permiten trabajar con esta distribución:

- `dnbinom(x, size, prob)`.
- `pnbinom(q, size, prob, lower.tail)`.
- `qnbinom(p, size, prob, lower.tail)`.
- `rnbinom(n, size, prob)`.

Donde:

[3]Traducción libre de los autores.

- `x, q` son vectores de cuantiles (enteros no negativos).
- `p` es un vector de probabilidades.
- `n` es la cantidad de observaciones.
- `size` corresponde al número (no negativo) de intentos.
- `prob` es la probabilidad de éxito de cada intento.
- `lower.tail` es análogo al de la función `pnorm`.

3.3.5 Distribución de Poisson

Útil para estimar la cantidad de eventos en una población grande en un lapso de tiempo dado, por ejemplo, la cantidad de contagios de influenza entre los habitantes de Santiago en una semana, la **distribución de Poisson** (figura 3.13) tiene una función de probabilidad definida por la ecuación 3.25, donde λ es la tasa o cantidad de eventos que se espera observar en un lapso de tiempo dado y k puede tomar cualquier valor entero no negativo. La media de esta distribución está dada por λ y la desviación estándar, por $\sqrt{\lambda}$.

$$\Pr(\text{observar } k \text{ eventos}) = \frac{\lambda^k e^{-\lambda}}{k!} \tag{3.25}$$

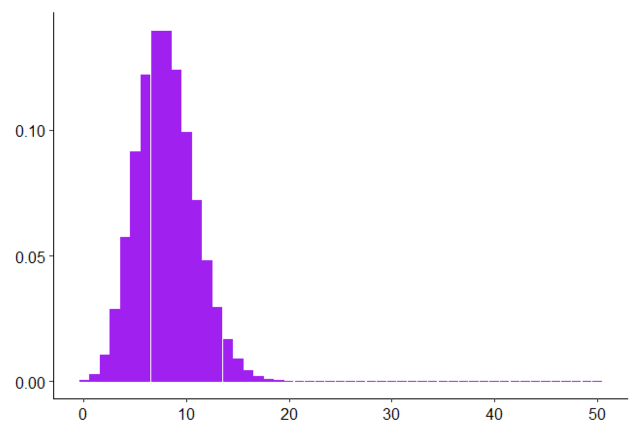


Figura 3.13: ejemplo de distribución de Poisson.

Las funciones de R para esta distribución son:

- `dpois(x, lambda)`.
- `ppois(q, lambda, lower.tail)`.
- `qqpois(p, lambda, lower.tail)`.
- `rpois(n, lambda)`.

Donde:

- `x, q` son vectores de cuantiles (enteros no negativos).
- `p` es un vector de probabilidades.
- `n` es la cantidad de observaciones.
- `lambda` es un vector no negativo de medias.
- `lower.tail` es análogo al de la función `pnorm`.

3.4 EJERCICIOS PROPUESTOS

1. Da un ejemplo de variable aleatoria (novedosa) que puedas observar en tus compañeros y que tenga una función de densidad de probabilidad discreta.
2. Para la variable anterior, ¿cuál sería el valor esperado? ¿Cuál sería la varianza? ¿Cómo te imaginas su función de densidad de probabilidad?
3. Lista tres nombres distintos con que también se llama a las funciones de densidad de probabilidad.
4. Si una variable aleatoria tiene una función de densidad de probabilidad con media igual a 30 y desviación estándar de 3, ¿por qué podría ocurrir que la probabilidad de que la variable tome el valor 30 sea nula, es decir, $P(X = 30) = 0$?
5. Según el Reporte Mensual de Empleo, las siguientes son las estadísticas (media $\pm$ desviación estándar) para las seis variables relevantes que se han estudiado en los últimos cinco años:
   a) Número de personas despedidas: 64.675 ± 8.321.
   b) Número de personas renunciadas: 118.543 ± 17.936.
   c) Número de personas jubiladas: 97.092 ± 11.147.
   d) Número de empleos creados: 24.715 ± 10.832.
   e) Número de personas contratadas: 301.345 ± 27.261.
   f) Número de personas entrando a la fuerza de trabajo: 26.444 ± 29.440.

   Con esta información, calcula la media y la desviación estándar de:
   a) Caída neta del empleo: $(d) - (a) - (b) - (c)$.
   b) Subida neta del empleo: $(e) - (a) - (b) - (c)$.
   c) Caída neta del desempleo: $(a) + (b) + (e) + (f)$.
   d) Vacancia del empleo: $(d) - (e)$.
6. ¿Qué significa que cierto valor de una variable aleatoria, usualmente con distribución normal, tenga valor $Z = 1,5$?
7. Según la regla empírica, ¿entre qué estaturas se podría encontrar al 95 % de los estudiantes varones del Departamento de Ingeniería Informática de la Universidad de Santiago de Chile, si esta variable sigue una distribución $N(\mu = 171, \sigma = 3)$?
8. ¿Qué información entrega un Gráfico Q-Q? ¿Para qué se usa?
9. La probabilidad de que un estudiante universitario chileno seleccionado al azar sea VIH positivo es 0,013. ¿Cuáles serían la media y la desviación estándar de esta variable?
10. En promedio, ¿a cuántos estudiantes universitarios se debería revisar hasta encontrar a un VIH positivo?
11. Si el Departamento de Salud de una Universidad chilena controla a 50 estudiantes por día durante una semana de clases (lunes a viernes), ¿cuál sería el número promedio de VIH positivos detectados cada día? ¿Con qué varianza?
12. Si la Universidad del ejercicio anterior dispone de 10 paquetes de tratamiento de VIH para estudiantes, ¿cómo podría saber a cuántos estudiantes debería examinar para poder asignarlos (suponiendo que todo estudiante VIH positivo acepta el tratamiento)?
13. Muestra un ejemplo novedoso de una variable aleatoria relacionada que podría seguir una distribución de Poisson.

CAPÍTULO 4. FUNDAMENTOS PARA LA INFERENCIA

En el capítulo 1 se definen los conceptos de población, entendido como todo el conjunto de interés, y muestra, que es un subconjunto de la población. También se introducen las nociones de parámetro, correspondiente a un valor que resume la población (por ejemplo la media de la población, μ), y de estadístico, como valor que resume una muestra (por ejemplo, la media muestral, $\overline{x}$). La **inferencia estadística** tiene por objeto entender cuán cerca está el estadístico del parámetro real de la población. En este capítulo conoceremos los principios necesarios para la inferencia estadística, con base en Diez y col. (2017, pp. 168-202) y Field y col. (2012, pp. 40-47).

4.1 ESTIMADORES PUNTUALES

Como ya dijimos, los parámetros y los estadísticos son valores que resumen, respectivamente, una población y una muestra. En consecuencia, podemos decir que un estadístico corresponde a un **estimador puntual** de un parámetro. El valor de un estimador puntual cambia dependiendo de la muestra que usemos para obtenerlo. Así, por más que su valor se acerque al parámetro de la población, difícilmente será igual a este último. Sin embargo, el estimador tiende a mejorar a medida que aumentamos el tamaño de la muestra, por efecto de la **ley de los grandes números**. Para ilustrar este fenómeno, consideremos la **media móvil**, que es una secuencia de medias muestrales en que cada una de ellas toma un elemento más de la población que su antecesora. La figura 4.1, elaborada con el script 4.1, ejemplifica este fenómeno.

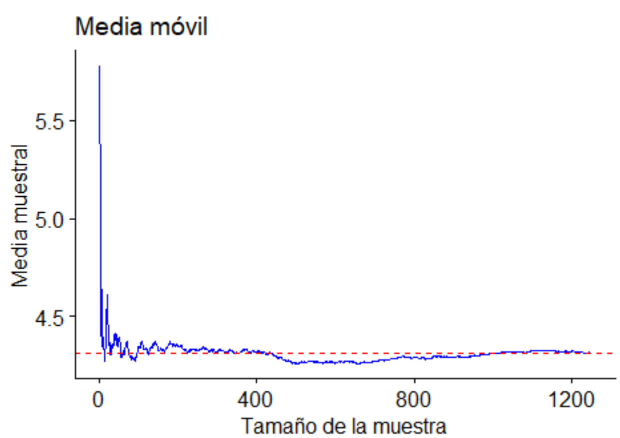


Figura 4.1: medias obtenidas al agregar a la muestra un elemento cada vez.

Script 4.1: representación gráfica de la media móvil.

```
library(ggpubr)

# Establecer la semilla para generar números aleatorios.
set.seed(9437)

# Generar aleatoriamente una población de tamaño 1500
# (en este caso, con una distribución cercana a la normal).
```

```
poblacion <- rnorm(n = 1500, mean = 4.32, sd = 0.98)

# Calcular la media de la población.
media_poblacion <- mean(poblacion)
cat("Media de la población:", media_poblacion, "\n")

# Tomar una muestra de tamaño 1250.
tamano_muestra <- 1250
muestra <- sample(poblacion, tamano_muestra)

# Calcular las medias acumuladas (es decir, con muestras de
# 1, 2, 3, ... elementos).
n <- seq(along = muestra)
media <- cumsum(muestra) / n

# Crear una matriz de datos con los tamaños y las medias muestrales.
datos <- data.frame(n, media)

# Graficar las medias muestrales.
g <- ggline(data = datos,
             x = "n",
             y = "media",
             plot_type = "l",
             color = "blue",
             main = "Media móvil",
             xlab = "Tamaño de la muestra",
             ylab = "Media muestral")

# Añadir al gráfico una recta con la media de la población.
g <- g + geom_hline(aes(yintercept = media_poblacion),
                    color = "red", linetype = 2)

print(g)
```

Para determinar qué tan adecuado es un estimador, necesitamos saber cuánto cambia de una muestra a otra. Si esta variabilidad es pequeña, es muy probable que la estimación sea buena. Podemos estudiar la variabilidad de la muestra con ayuda de la **distribución muestral**, que representa la distribución de estimadores puntuales obtenidos con **todas** las diferentes muestras de igual tamaño de una misma población. La figura 4.2 (construida con el script 4.2) representa las medias para diferentes muestras de una población, aunque solo una selección aleatoria de todas las posibles muestras, incluyendo además una línea vertical roja que señala la media de la población. Podemos destacar que las medias muestrales tienden a aglutinarse en torno a la media poblacional, pues de acuerdo al **teorema del límite central**, la distribución de $\overline{x}$ se aproxima a la normalidad. Esta aproximación mejora a medida que aumenta el tamaño de la muestra.

Script 4.2: distribución de la media muestral.

```
library(ggpubr)

# Establecer la semilla para generar números aleatorios.
set.seed(94)

# Generar aleatoriamente una población de tamaño 1500
# (en este caso, con una distribución cercana a la normal).
poblacion <- rnorm(n = 1500, mean = 4.32, sd = 0.98)

# Calcular la media de la población.
media_poblacion <- mean(poblacion)
cat("Media de la población:", media_poblacion, "\n")

```

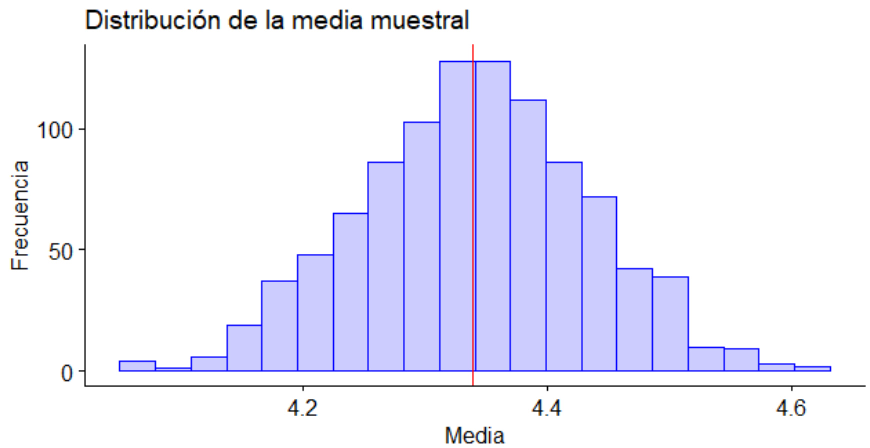


Figura 4.2: distribución muestral de la media para muestras con 100 observaciones.

```
# Tomar 1000 muestras de tamaño 100. Quedan almacenadas
# como una matriz donde cada columna es una muestra.
tamano_muestra <- 100
repeticiones <- 1000

muestras <- replicate(repeticiones,
                      sample(poblacion, tamano_muestra))

# Calcular medias muestrales y almacenar los resultados
# en forma de data frame.
medias <- colMeans(muestras)
medias <- as.data.frame(medias)

# Construir un histograma de las medias muestrales.
g <- gghistogram(data = medias,
                 x = "medias",
                 bins = 20,
                 title = "Distribución de la media muestral",
                 xlab = "Media",
                 ylab = "Frecuencia",
                 color = "blue",
                 fill = "blue",
                 alpha = 0.2)

# Agregar línea vertical con la media de la población.
g <- g + geom_vline(aes(xintercept = media_poblacion),
                    color = "red", linetype = 1)

print(g)
```

4.2 MODELOS ESTADÍSTICOS

Ahora que hemos conocido más conceptos, podemos definir con precisión qué es un **modelo estadístico**. En el capítulo 1 dijimos que un modelo es simplemente una representación y que los modelos estadísticos pueden emplearse para diversos propósitos:

- Describir o resumir datos.
- Clasificar objetos o predecir resultados.
- Anticipar los resultados de intervenciones (en ocasiones).

Más formalmente, un modelo estadístico es una descripción de un **proceso probabilístico** con **parámetros desconocidos** que deben ser **estimados** en base a **suposiciones** y un conjunto de datos **observados**. En general, tiene la forma dada en la ecuación 4.1:

$$y_i = (\text{modelo}) + \varepsilon_i \tag{4.1}$$

Donde:

- y_i es el i-ésimo valor observado de la variable respuesta Y (también llamada variable de salida o variable dependiente).
- modelo es el resultado de una función determinista basada en un conjunto de argumentos.
- ε_i es el error, correspondiente a la **variación natural**, y no a una equivocación, existente entre los valores observados y los valores pronosticados por el modelo. También recibe los nombres de variación no sistemática, variación aleatoria, residuos o incluso, residuales.

El error ε_i en la ecuación 4.1 se relaciona entonces con la calidad del modelo. Mientras menor sea el error, mejor será el modelo. Por el contrario, un error grande es señal de un modelo fallido, que no describe bien los datos, no ayuda a predecirlos bien, o no ayuda a su correcta clasificación.

La media y la proporción, y cualquier estadístico en general, son, en sí mismos, modelos estadísticos, aunque bastante simples.

4.3 ERROR ESTÁNDAR

En el capítulo 2 conocimos la desviación estándar como medida que estima la distancia de las observaciones respecto de la media. El **error estándar**, denotado usualmente por $SE_{\hat{\theta}}$ o $\sigma_{\hat{\theta}}$, corresponde a la desviación estándar de la distribución de un estimador muestral $\hat{\theta}$ de un parámetro θ. Por ejemplo, el error estándar de la media, es decir la desviación estándar de la distribución de las medias de todas las posibles muestras de n observaciones independientes, se calcula de acuerdo a la ecuación 4.2.

$$SE_{\overline{x}} = \frac{s}{\sqrt{n}} \tag{4.2}$$

En esta ecuación y los párrafos siguientes deberíamos hablar de σ y no s

Donde s es la desviación estándar de la muestra (ecuación 2.3) y n corresponde al tamaño de la muestra. En esta ecuación queda en evidencia que el error estándar de la media disminuye a medida que el tamaño de la muestra aumenta. Un método confiable que podemos usar para asegurar que las observaciones sean independientes es realizar un muestreo aleatorio simple[1] que abarque menos del 10 % de la población.

Volviendo a la ecuación para calcular el error estándar de la media muestral (equación 4.2), ¡debemos tener cuidado antes de usarla! Ya hemos mencionado antes que la distribución de las medias muestrales tiende a ser cercana a la normal, por lo que en dicho caso es posible usar el **modelo normal**, sustentado en el teorema del límite central. Las condiciones que deben cumplirse para usar este modelo y que, en consecuencia, el error estándar sea preciso, son:

[1]Es decir, una muestra en que todos los elementos de la población tengan igual probabilidad de ser escogidos. Las técnicas de muestreo se abordan con más detalle en capítulos posteriores.

1. Las observaciones de la muestra son independientes.
2. La muestra es grande (en general $n \geq 30$).
3. La distribución de la muestra no es significativamente asimétrica. Esto último suele además relacionarse con la presencia de valores atípicos. Mientras mayor sea el tamaño de la muestra, más se puede relajar esta condición.

Si no se cumplen las condiciones anteriores, debemos considerar otras opciones: para muestras pequeñas, se deben considerar métodos alternativos, y si la distribución de la muestra presenta una asimetría significativa, entonces tendremos que incrementar el tamaño de la muestra para compensar el efecto de la desviación.

4.4 INTERVALOS DE CONFIANZA

Hasta ahora sabemos que un estimador puntual es un único valor (obtenido a partir de una muestra) que, como su nombre indica, estima un parámetro de la población. Por ende, dicho valor rara vez es exacto. En consecuencia, lo lógico sería establecer un rango de valores plausibles para el parámetro estimado, que llamaremos **intervalo de confianza**, y que se construye en torno al estimador puntual. Dado que el error estándar representa la desviación estándar asociada al estimador, tiene sentido que lo usemos como guía en este proceso.

Recordemos que en el capítulo 3 vimos una regla empírica para la distribución normal (figura 3.5), la cual señala que (para distribuciones normales) alrededor de 95 % de las veces el estimador puntual se encontrará en un rango de 2 errores estándar del parámetro. Es decir, al considerar un intervalo de confianza de dos errores estándar (4.3), tendremos 95 % de **confianza** de haber capturado el parámetro real.

$$\overline{x} \pm 2 \cdot SE_{\overline{x}} \tag{4.3}$$

Podemos generalizar la ecuación 4.3 para calcular el intervalo de confianza para la media con cualquier **nivel de confianza** como muestra la ecuación 4.4.

$$\overline{x} \pm z^* \cdot SE_{\overline{x}} \tag{4.4}$$

El término z^* en la ecuación 4.4 corresponde, usualmente, al valor z tal que el área bajo la curva normal estándar comprendida entre $-z^*$ y z^* es igual al nivel de confianza deseado. La expresión $z^* \cdot SE$ recibe el nombre de **margen de error**.

Tomemos como ejemplo un **nivel de confianza** (que, por razones que veremos en la sección siguiente, denotaremos por $1 - \alpha$) de 90 % (es decir, $1 - \alpha = 0,9$). Eso significa, entonces, que nuestro intervalo de confianza excluye el 5 % del área correspondiente a la cola inferior (es decir, el percentil con valor 0,05) e igual porcentaje del área correspondiente a la cola superior (que, como la distribución Z es simétrica, es igual al área anterior). Puesto que conocemos el percentil, $(1 - \alpha)/2 = 0,05$, en R podemos usar la llamada `qnorm(0.05, mean = 0, sd = 1, lower.tail = FALSE)` y obtenemos $z^* = 1,64$. Es importante indicar que en esta llamada estamos en realidad trabajando con la cola superior para que z^* sea positivo. Si hacemos la llamada para la cola inferior, obtenemos $z^* = -1,64$.

Es importante destacar que, una vez más, debemos ser cuidadosos al interpretar un intervalo de confianza del x % (x = $1 - \alpha$). Su significado es, sencillamente, “se tiene x % de certeza de que el parámetro de la población se encuentra entre...” (Diez y col., 2017, p. 180), es decir, que, en promedio, x % de los intervalos de confianza que se construyan en torno a un estadístico, con muestras de un tamaño fijo, capturarán el verdadero valor del parámetro. Esto **no es equivalente** a decir que el valor del parámetro tiene una ”probabilidad de x %” de estar entre los valores del intervalo calculado, lo que sería incorrecto. Por otra parte, los intervalos de confianza no dicen nada acerca de observaciones individuales, sino que solo hablan del parámetro en cuestión.

4.5 PRUEBAS DE HIPÓTESIS

Supongamos que un banco ha desarrollado un nuevo sistema computacional para gestionar sus transacciones. El nuevo sistema (N) se ha puesto a prueba durante un mes, funcionando (con iguales condiciones de hardware) en paralelo con el sistema antiguo (A) y el banco ha llevado un registro del tiempo que tarda cada sistema en efectuar cada transacción. El gerente ha determinado que autorizará la migración al nuevo sistema únicamente si este es más rápido que el antiguo para procesar las transacciones. Se sabe que el sistema antiguo tarda en promedio $\mu_A = 530$ milisegundos en procesar una transacción. Para el sistema nuevo se han registrado $n = 1.600$ transacciones, realizadas en un tiempo promedio de $\overline{x}_N = 527,9$ [ms] con desviación estándar $s_N = 48$ [ms].

Una primera aproximación para tomar la decisión puede ser investigar si existe diferencia en los tiempos de ejecución de ambos sistemas, lo que puede expresarse en torno a dos **hipótesis** (palabra que la Real Academia Española (2014) define como "Suposición de algo posible o imposible para sacar de ello una consecuencia") que compiten entre sí:

H_0: El nuevo sistema, en promedio, tarda lo mismo que el antiguo en procesar las transacciones, es decir: $\mu_N = \mu_A$.
H_A: Los sistemas requieren, en promedio, cantidades de tiempo diferentes para procesar las transacciones, es decir: $\mu_N \neq \mu_A$

La primera hipótesis, H_0, recibe el nombre de **hipótesis nula** y suele representar una postura escéptica, es decir, que no hay cambios, por lo que **¡la hipótesis nula siempre se formula como una igualdad!**. La segunda (H_A), llamada **hipótesis alternativa**, representa en cambio una nueva perspectiva. Esta primera aproximación corresponde a una **prueba bilateral** o de dos colas, pues la diferencia puede ser en ambos sentidos: H_0 no parece correcta si $\mu_N < \mu_A$ o si $\mu_N > \mu_A$.

Como en este caso conocemos el valor de $\mu_A = 530$ [ms], también podríamos escribir la formulación matemática de las hipótesis de la siguiente manera:

H_0: $\mu_N = 530$
H_A: $\mu_N \neq 530$

En este planteamiento, "530" recibe el nombre de **valor nulo**, pues representa el valor del parámetro cuando se cumple la hipótesis nula.

Una aproximación más cercana al problema descrito puede ser investigar si el nuevo sistema es efectivamente **más rápido** que el antiguo. En este caso, se habla de una **prueba unilateral** o de una cola, pues solo interesa saber si el tiempo promedio empleado por el nuevo sistema es menor que el empleado por el sistema antiguo. Las hipótesis, en este caso, serían:

H_0: El nuevo sistema tarda, en promedio, lo mismo que el antiguo en procesar las transacciones, es decir: $\mu_N = \mu_A$.
H_A: El nuevo sistema tarda, en promedio, menos que el antiguo en procesar las transacciones, es decir: $\mu_N < \mu_A$

Obviamente en otros casos podría interesar solamente si valor alternativo es mayor que el valor nulo.

Teniendo las hipótesis planteadas, sigue decidir si la hipótesis nula parece o no plausible a través de una **prueba de hipótesis**. El marco para la prueba de hipótesis es **escéptico**: no se rechaza la hipótesis nula a menos que haya suficiente evidencia para rechazarla en favor de la hipótesis alternativa. Esta idea es muy parecida a la expresada en la expresión de uso común "se presume inocente mientras no se demuestre lo contrario". Sin embargo, el que no se logre rechazar H_0 **no significa aceptarla** como verdadera o como correcta sin más. Por eso se usa un lenguaje bastante peculiar, señalando que *se falla al rechazar* H_0 o bien que *se rechaza* H_0 *en favor de* H_A. Retomando la analogía con la expresión anterior, que no haya pruebas suficientes para la culpabilidad, no significa que una persona sea en verdad inocente.

Volvamos al escenario del ejemplo para la prueba de hipótesis bilateral (es decir, aquella en que solo queremos ver si hay diferencias en el tiempo de procesamiento de transacciones entre ambos sistemas del banco). El valor de $\overline{x}_N = 527,9$ [ms] es, en efecto, distinto de $\mu_A = 530$ [ms]. No obstante, al ser una estimación puntual, como ya hemos estudiado, esta diferencia podría deberse simplemente a la muestra escogida, por lo que el parámetro real μ_N podría ser igual a μ_A [ms]. En consecuencia, resulta útil calcular el intervalo de confianza para $\overline{x}_N$.

Comencemos por determinar el error estándar:

$$SE_{\overline{x}} = \frac{s_N}{\sqrt{n}} = \frac{48}{\sqrt{1600}} = 1,2$$

Ahora fijemos un nivel de confianza, por ejemplo 95 %, y usemos el valor z^* correspondiente para calcular el intervalo de confianza:

$$\overline{x}_N \pm z^* \cdot SE_{\overline{x}} = 527,9 \pm 1,96 \cdot 1,2 = [525,548; 530,252]$$

Como el parámetro del sistema antiguo ($\mu_A = 530$ [ms]) cae (a penas) dentro de este intervalo, se puede suponer que no existe una diferencia significativa entre los tiempos promedio requeridos por ambos sistemas, por lo que no se rechaza H_0. Así, tenemos un 95 % de confianza en que no existe una diferencia entre los tiempos que requieren ambos sistemas para procesar transacciones. Sin embargo, esta decisión es un tanto apresurada ya que el resultado está cerca del borde de rechazo y, en este caso, lo lógico sería investigar más (hacer crecer la muestra).

Revisemos ahora el caso planteado con hipótesis alternativa unilateral (es decir, queremos ver si el nuevo sistema es, en efecto, más rápido). Manteniendo nuestro nivel de confianza $1-\alpha = 0,95$, en este caso debemos considerar los valores menores a $\mu_A = 530$ [ms] para el cálculo de z^*. En otras palabras, el 5 % que descartamos corresponde únicamente a la cola superior. Así, nuestro valor para z^* está dado por la llamada `qnorm(0.05, mean = 0, sd = 1, lower.tail = FALSE)`, obteniéndose $z^* = 1,64$ (aprox.) por lo que se tiene que la cota superior es:

$$\overline{x}_N - z^* \cdot SE_{\overline{x}} = 527,9 - 1,64 \cdot 1,2 = 529.874$$

Luego, el intervalo de confianza va desde "cualquier valor" bajo la media observada en la muestra hasta el valor calculado arriba, por lo que el intervalo con 95 % confianza sería: $[-\infty; 529,874]$.

Ahora el valor $\mu_A = 530$ [ms] cae (apenas) fuera del intervalo y podemos decir que existe evidencia de que el nuevo sistema tarda en promedio menos tiempo que el antiguo en procesar las transacciones.

Ahora bien, siempre que se prueban hipótesis podemos cometer un error al momento de decidir si rechazar o no la hipótesis nula. Afortunadamente, la estadística ofrece herramientas para cuantificar cuán frecuentes son dichos errores. Existen cuatro posibles escenarios, los cuales se presentan en la tabla 4.1. El **error tipo I** corresponde a rechazar H_0 cuando en realidad es verdadera, mientras que el **error tipo II** corresponde a no rechazarla cuando en realidad H_A es verdadera.

		Conclusión de la prueba	
		No rechazar H_0	Rechazar H_0 en favor de H_A
Verdad	H_0 verdadera	Decisión correcta	Error tipo I
	H_A verdadera	Error tipo II	Decisión correcta

Tabla 4.1: posibles escenarios para una prueba de hipótesis.

Como ya hemos señalado, la prueba de hipótesis se basa en no rechazar H_0 a menos que se tenga evidencia contundente. Por regla general, no se desea cometer el error de rechazar incorrectamente la hipótesis nula (error tipo I) en más de 5 % de los casos. Esto corresponde a un **nivel de significación** de 0,05, denotado por $\alpha = 0,05$. Si usamos un intervalo de confianza de 95 % para evaluar una prueba de hipótesis en que la hipótesis nula es verdadera, cometeremos un error tipo I cada vez que el estimador puntual esté a 1,96 o más errores estándar del parámetro de la población. Esto puede ocurrir un 5 % de las veces (2,5 % en cada cola de la distribución para el caso bilateral). Del mismo modo, un intervalo de confianza del 99 % es equivalente a un nivel de significación $\alpha = 0,01$.

El intervalo de confianza es de mucha ayuda para decidir si rechazar o no H_0. No obstante, no aporta información directa acerca de cuán fuerte es la evidencia para la decisión tomada.

4.5.1 Prueba formal de hipótesis con valores p

Antes de que la computación se hiciera masiva, las personas tenían dos procedimientos posibles para decidir una prueba de hipótesis. El primero es el realizado en la sección anterior, esto es, calcular el intervalo con $(1 - \alpha)$ % de confianza de acuerdo a los estadísticos observados en una muestra y revisar si el valor nulo cae o no dentro de este intervalo. El otro procedimiento clásico, que podemos encontrar en muchos libros y sitios en Internet, es estimar a qué valor z corresponde la media observada en la distribución normal estandarizada que define el valor nulo y el error estándar: si este estadístico z es mayor que z^*, entonces el estadístico cae en una "zona de rechazo" de H_0; en caso contrario ($|z| < z^*$), se falla en rechazar la hipótesis nula.

Si bien estos procedimientos siguen siendo útiles, su diseño respondía a la existencia de **tablas de probabilidad** en que se tabulaban probabilidades para algunos valores de percentiles de uso común, como 90 %, 95 %, 0,975 % o 0,99 %.

Con la llegada de los computadores, y en particular de entornos como R, es posible obtener probabilidades (casi) exactas para cualquier percentil. Esto hizo que un tercer método para decidir una prueba de hipótesis haya ido ganando popularidad: el uso del **valor p**, también llamado **p-valor**, que es definido por Diez y col. (2017, p. 186) como "la probabilidad de observar datos al menos tan favorables como la muestra actual para la hipótesis alternativa, si la hipótesis nula es verdadera". De esta forma, un p-valor permite cuantificar cuán fuerte es la evidencia en contra de la hipótesis nula (y en favor de la hipótesis alternativa).

Consideremos ahora el escenario de la hipótesis unilateral del ejemplo, con un nivel de significación $\alpha = 0,05$, bajo el supuesto de que H_0 es verdadera y que la muestra a su vez tiene una distribución cercana a la normal. Recordemos que $\overline{x}_N = 527,9$ [ms] y $s_N = 48$ [ms] en $n = 1600$ observaciones. Esta distribución se vería como muestra la figura 4.3, creada mediante el script 4.3.

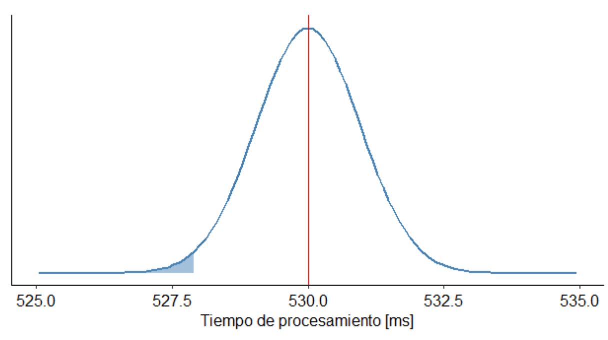


Figura 4.3: probabilidad de encontrar una media igual o menor que $\overline{x} = 527,9$ [ms] en la distribución muestral con $\mu_{\overline{x}} = 530$ y $\sigma_{\overline{x}} = 1,2$.

En este punto, resulta importante hacer una aclaración en relación al valor p. El área bajo la sección de la curva con valores menores o iguales a un estimador se calcula usando para ello el **valor z**, definido en la ecuación 4.5, como **estadístico de prueba**.

$$z = \frac{\text{estimador puntual} - \text{valor nulo}}{SE_{\text{estimador puntual}}} = \frac{\hat{\theta} - \theta_0}{SE_{\hat{\theta}}} \tag{4.5}$$

Un **estadístico de prueba** es un estadístico de resumen que resulta especialmente útil para evaluar hipótesis o calcular el valor p. El valor z se usa cuando el estimador puntual se acerca a la normalidad, aunque existen otros estadísticos de prueba adecuados para otros escenarios.

Script 4.3: cálculo del valor p para una prueba de una cola.

```
library(ggpubr)

# Generar una muestra donde la media cumpla con la hipótesis nula.
set.seed(872)

media_poblacion_antiguo <- 530
media_muestra_nuevo <- 527.9
desv_est <- 48
n <- 1600
error_est <- desv_est / sqrt(n)

x <- seq(media_poblacion_antiguo - 5.2 * error_est,
         media_poblacion_antiguo + 5.2 * error_est,
         0.01)

y <- dnorm(x, mean = media_poblacion_antiguo, sd = error_est)

datos <- data.frame(x, y)

# Graficar la muestra.
g <- ggplot(data = datos, aes(x))

g <- g + stat_function(fun = dnorm,
                       args = list(mean = media_poblacion_antiguo,
                                   sd = error_est),
                       colour = "steelblue", size = 1)

g <- g + ylab("")
g <- g + scale_y_continuous(breaks = NULL)
g <- g + scale_x_continuous(name = "Tiempo de procesamiento [ms]")
g <- g + theme_pubr()

# Colorear el área igual o menor que la media observada.
g <- g + geom_area(data = subset(datos,
                                 x < media_muestra_nuevo),
                   aes(y = y),
                   colour = "steelblue",
                   fill = "steelblue",
                   alpha = 0.5)

# Agregar una línea vertical para el valor nulo.
g <- g + geom_vline(aes(xintercept = media_poblacion_antiguo),
                    color = "red", linetype = 1)

print(g)

# Calcular el valor Z para la muestra.
Z <- (media_muestra_nuevo - media_poblacion_antiguo) / error_est

```

```
50 # Calcular el valor p.
51 p_1 <- pnorm(Z, lower.tail = TRUE)
52 
53 cat("Valor p: ", p_1, "\n")
54 
55 # También se puede calcular el valor p directamente a partir de la
56 # distribución muestral definida por el valor nulo y el error
57 # estándar.
58 p_2 <- pnorm(media_muestra_nuevo, mean = media_poblacion_antiguo,
59              sd = est_err)
60 
61 cat("Valor p: ", p_2)
```

El valor p, en este caso $p = 0,040$, corresponde al área coloreada en la figura 4.3, y se calcula en la línea 51 del script 4.3. Esto nos indica, en este caso, que si H_0 fuera verdadera y el nuevo sistema tarda en promedio lo mismo que el antiguo en procesar las transacciones, la probabilidad de encontrar una media de a lo más 527,9 [ms] para una muestra de 1.600 transacciones es de 4 %, lo que sería bastante poco frecuente.

Cuanto menor sea el valor p, más fuerte será la evidencia en favor de H_A por sobre H_0. Y aquí la ventaja de usar este método para decidir: el valor p se puede **comparar directamente** con el nivel de significación α, y si p es menor que el nivel de significación se considera evidencia suficiente para rechazar la hipótesis nula en favor de la hipótesis alternativa. En este ejemplo, $p = 0,040 < \alpha = 0,05$, por lo que se falla al rechazar H_0 en favor de H_A. Pero como se dijo cuando usamos intervalos de confianza, el valor p está cerca del valor α y convendría ser menos tajante en la decisión y evaluar la posibilidad de ampliar la muestra para conseguir evidencia más definitiva.

Siempre es recomendable formular la conclusión de la prueba de hipótesis en lenguaje llano, para facilitar su comprensión. Así, en este caso concluimos que los datos sugieren que el nuevo sistema tarda menos que el antiguo en procesar transacciones, pero que es necesario hacer un estudio con más observaciones para tener un diagnóstico más definitivo.

Volvamos nuevamente al escenario de la prueba de hipótesis bilateral para el ejemplo, manteniendo el nivel de significación $\alpha = 0,05$. Puesto que en este caso nos interesa la diferencia en ambas direcciones, ya que la evidencia en ambas direcciones es favorable para H_A, debemos considerar el área bajo las dos colas de la curva normal, a diferencia del caso de la prueba de hipótesis unilateral en que solo se consideramos la cola correspondiente a la dirección de interés de la diferencia. Dado que el modelo normal es simétrico, el área bajo ambas colas es la misma (figura 4.4, script 4.4). El valor p, entonces, ahora es igual a dos veces el área de la cola inferior, es decir, $p = 0,080$. Puesto que $p > \alpha$, se falla en rechazar H_0. Es decir, no hay evidencia suficiente para concluir que existe una diferencia entre los tiempos promedio requeridos por ambos sistemas para procesar transacciones.

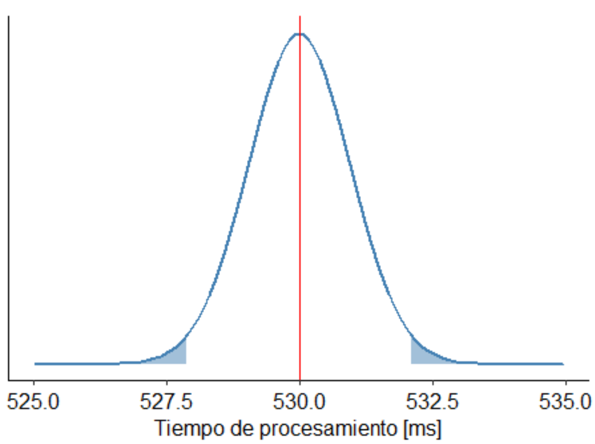


Figura 4.4: cuando la prueba de hipótesis es bilateral, se deben colorear ambas colas.

Script 4.4: cálculo del valor p para una prueba de dos colas.

```
library(ggpubr)

# Generar una muestra donde la media cumpla con la hipótesis nula.
set.seed(208)

media_poblacion_antiguo <- 530
media_muestra_nuevo <- 527.9
desv_est <- 48
n <- 1600
error_est <- desv_est / sqrt(n)

x <- seq(media_poblacion_antiguo - 5.2 * error_est,
         media_poblacion_antiguo + 5.2 * error_est,
         0.01)

y <- dnorm(x,
           mean = media_poblacion_antiguo,
           sd = error_est)

dataframe <- data.frame(x, y)

# Graficar la muestra.
g <- ggplot(data = dataframe, aes(x))

g <- g + stat_function(fun = dnorm,
                       args = list(mean = media_poblacion_antiguo,
                                   sd = error_est),
                       colour = "steelblue", size = 1)

g <- g + ylab("")
g <- g + scale_y_continuous(breaks = NULL)
g <- g + scale_x_continuous(name = "Tiempo de procesamiento [ms]")
g <- g + theme_pubr()

# Colorear el área igual o menor que la media observada.
g <- g + geom_area(data = subset(dataframe,
                                 x < media_muestra_nuevo),
                   aes(y = y),
                   colour = "steelblue",
                   fill = "steelblue",
                   alpha = 0.5)

# Calcular el área bajo la cola inferior.
area_inferior <- pnorm(media_muestra_nuevo,
                       mean = media_poblacion_antiguo,
                       sd = desv_est)


# Colorear igual área en la cola restante.
corte_x <- qnorm(1 - area_inferior,
                 mean = media_poblacion_antiguo,
                 sd = desv_est)

g <- g + geom_area(data = subset(dataframe,
                                 x > corte_x),
                   aes(y = y),
                   colour = "steelblue",
                   fill = "steelblue",
```

```r
                    alpha = 0.5)

# Agregar una línea vertical para el valor nulo.
g <- g + geom_vline(aes(xintercept = media_poblacion_antiguo),
                     color = "red", linetype = 1)

print(g)

# Calcular el valor Z para la muestra.
Z <- (media_muestra_nuevo - media_poblacion_antiguo) / error_est

# Calcular el valor p (recordando ahora que la hipótesis es bilateral).
p <- 2 * pnorm(Z, lower.tail = TRUE)

cat("Valor p: ", p)
```

Un punto importante que debemos tener en cuenta es que **las pruebas unilaterales** se usan cuando se desea verificar un incremento o un decremento, pero no ambas. No obstante, esta decisión debe tomarse siempre **antes de examinar los datos**, pues de lo contrario se duplica la probabilidad de cometer errores de tipo I y se está cayendo en **prácticas poco éticas**.

4.5.2 El efecto del nivel de significación

Hemos visto que el nivel de significación (α) representa la proporción de veces en que se cometería un error de tipo I (es decir, rechazar H_0 en favor de H_A, cuando H_0 es en realidad verdadera). Si resulta costoso o peligroso cometer un error de este tipo, debemos requerir evidencia más fuerte para rechazar la hipótesis nula (es decir, reducir la probabilidad de que esto ocurra), lo que podemos lograr usando un valor más pequeño para el nivel de significación, por ejemplo, $\alpha = 0,01$. Sin embargo, esto necesariamente **aumentará** la probabilidad de cometer un error de tipo II.

Si, por el contrario, el costo o el peligro de cometer un error de tipo II (no rechazar H_0 cuando en realidad H_A es verdadera) es mayor, debemos escoger un nivel de significación más elevado (por ejemplo, $\alpha = 0,10$).

Así, **el nivel de significación seleccionado para una prueba siempre debe reflejar las consecuencias de cometer errores de tipo I o de tipo II**.

4.6 INFERENCIA PARA OTROS ESTIMADORES

Hasta ahora, solo hemos considerado la media como estimador para la inferencia. No obstante, muchos de los conceptos que hemos visto en este capítulo pueden aplicarse, con algunas ligeras modificaciones, usando otros estimadores.

4.6.1 Estimadores puntuales con distribución cercana a la normal

En realidad existen múltiples estimadores puntuales, además de la media, cuya distribución muestral es cercana a la normal si las muestras son lo suficientemente grandes, tales como las proporciones y la diferencia de medias. Si bien veremos con detalle la prueba de hipótesis con estos estimadores puntuales en capítulos posteriores, es importante contar con algunas orientaciones generales.

Un supuesto importante que debemos tener en cuenta es que el estimador puntual $\hat{\theta}$ debe ser **insesgado**. Esto significa que la distribución muestral de $\hat{\theta}$ tiene su centro en el valor del parámetro θ que estima. En otras palabras, un estimador insesgado (como la media) tiende a proveer una estimación cercana al parámetro real.

En términos generales, el intervalo de confianza para un estimador puntual insesgado cuya distribución es cercana a la normal (como la media, las proporciones o la diferencia de medias) está dado por la ecuación 4.6, donde z^* se escoge de manera tal que se condiga con el nivel de confianza seleccionado y y la lateralidad de la hipótesis alternativa. Como se dijo anteriormente, el valor $z^* \cdot SE_{\hat{\theta}}$ se denomina "margen de error". Debemos recordar que la ecuación 4.2 corresponde al error estándar de la media, pero los errores estándar para otros estimadores puntuales se estiman de manera diferente a partir de los datos.

$$\hat{\theta} \pm z^* \cdot SE_{\hat{\theta}} \tag{4.6}$$

El método de prueba de hipótesis usando valores p puede generalizarse para otros estimadores puntuales con distribución cercana a la normal. Para ello, Diez y col. (2017, p. 199) señalan que se debemos considerar los siguientes pasos:

> Prueba de hipótesis usando el modelo normal:
>
> 1. Formular las hipótesis nula (H_0) y alternativa (H_A) en lenguaje llano y luego en notación matemática.
> 2. Identificar un estimador puntual (estadístico) adecuado e insesgado para el parámetro de interés.
> 3. Verificar las condiciones para garantizar que la estimación del error estándar sea razonable y que la distribución muestral del estimador puntual siga aproximadamente una distribución normal.
> 4. Calcular el error estándar. Luego, graficar la distribución muestral del estadístico bajo el supuesto de que H_0 es verdadera y sombrear las áreas que representan el valor p.
> 5. Usando el gráfico y el modelo normal, calcular el valor p para evaluar las hipótesis y escribir la conclusión en lenguaje llano.

4.6.2 Estimadores con otras distribuciones

Existen métodos de construcción de intervalos de confianza y prueba de hipótesis adecuados para aquellos casos en que el estimador puntual o el estadístico de prueba no son cercanos a la normal (por ejemplo, si la muestra es pequeña, se tiene una mala estimación del error estándar o el estimador puntual tiene una distribución distinta a la normal). No obstante, la selección de métodos alternativos debe hacerse siempre teniendo en cuenta la distribución muestral del estimador puntual o del estadístico de prueba.

Una consideración importante es que **siempre debemos verificar el cumplimiento de las condiciones requeridas por una herramienta estadística**, pues de lo contrario las conclusiones pueden ser erradas y carecerán de validez.

4.7 EJERCICIOS PROPUESTOS

1. ¿Es correcto afirmar que, si se lanza un dado una y otra vez, la media móvil simple del número de puntos que aparecen en la cara superior crece monótonamente? Justifica tu respuesta.
2. ¿Es correcto afirmar que, si se lanza un dado una y otra vez, la proporción de veces que aparece un número impar de puntos (1, 3 o 5) en la cara superior es siempre 0,5? Justifica su respuesta.
3. Si se calcula la media de diez muestras distintas extraídas de la misma población, ¿se espera ver el mismo valor cada vez? ¿Cómo se llama a este fenómeno?
4. Completa las siguiente oraciones:
   a) Una estimación __________ __________ es un __________ calculado con datos de una muestra como aproximación del valor desconocido de un __________ de la población en estudio.
   b) $\overline{X}$ o $\overline{x}$ se usan para denotar la __________ __________, que es una estimación puntual de μ, la __________ __________.
5. Se sabe que una prueba para medir el coeficiente intelectual de jóvenes de 18 años produce puntuaciones que siguen una distribución $\mathcal{N}(\mu = 100, \sigma^2 = 100)$.
   a) Dibuja el histograma de la distribución muestral de medias para muestras de tamaño 25 de esta población.
   b) Una de las muestras anteriores presentó $\overline{x} = 95$ y $s = 15$. Determina el intervalo con 95 % de confianza para este caso.
   c) Con otra de las muestras se pudo determinar que su intervalo con 99 % confianza era $[90, 26; 105, 74]$. ¿Qué significa esto?
   d) El intervalo anterior, ¿es más grande o más pequeño que uno con 90 % de confianza?
6. Una empresa de tecnología quiere promocionar un software especializado para almacenar y recuperar imágenes médicas digitales. Con esta idea, está financiando un estudio para determinar el tiempo (en segundos) que necesita un grupo de médicos para recuperar imágenes desde sus propios registros en sus portátiles personales y desde la base de datos central con el software ofrecido y una conexión a la Web.
   a) Enuncia las hipótesis nula y alternativa (en castellano común).
   b) Identifica la variable aleatoria que se va a estudiar, el parámetro de interés y el correspondiente estadístico.
   c) Enuncia, más formalmente, las hipótesis nula y alternativa para este caso.
   d) Supón que el intervalo con 95 % confianza para el tiempo de recuperación promedio de una imagen digital desde la base de datos central resultó ser $[24; 36]$ [s]. ¿Qué decisión tomarías ante la hipótesis nula: la media del tiempo de recuperación de una imagen digital con el nuevo software es de 25 segundos? En este caso, ¿cuál podría ser la hipótesis alternativa?
   e) Para el intervalo de confianza anterior, ¿cuál sería un error de tipo I?
   f) Conociendo el intervalo de confianza anterior, ¿es posible cometer un error de tipo II? Explica.
7. Si una hipótesis nula es falsa, aumentar el nivel de significación para un tamaño de muestra dado, ¿reduce la probabilidad de rechazarla?
8. Qué significa que un estadístico tenga un valor p de 0,025?
9. Si una hipótesis nula es rechazada a un nivel de significación de 0,01, ¿será rechazada a un nivel de significación 0,05? Explica.
10. Si una hipótesis nula es rechazada por una prueba unilateral (una cola), ¿será también rechazada por una prueba bilateral (dos colas)? Explica.
11. Acabas de leer un artículo que hace la siguiente aseveración: "*a 95 % confidence interval for mean reaction time is from 0.25 to 0.29 seconds. Thus, about 95 % of individuals will have reaction times in this interval.*" Comenta.
12. Da el ejemplo de un estudio en que es más dañino cometer un error tipo II que un error tipo I.
13. Lista las condiciones que deben verificarse para asegurar que el TLC (teorema del límite central) está rigiendo y es posible hacer una prueba de hipótesis o calcular un intervalo de confianza.
14. Si para un estudio de una determinada variable aleatoria numérica es igualmente dañino cometer errores de tipo I como errores tipo II:
    a) Dibuja la distribución de una muestra de tamaño 16 (un diagrama de caja, por ejemplo) para la que el contraste de hipótesis con nivel de significación 0,05 sea confiable.

b) Dibuja la distribución de una muestra de tamaño 30 en que se requiera de un nivel de significación más exigente ($\alpha < 0,05$) para hacer el contraste de hipótesis más confiable.
c) Dibuja la distribución de una muestra en que es mejor no confiar en el contraste de hipótesis con métodos estudiados hasta ahora.

15. Si un estudio sobre el tiempo promedio de búsqueda y recuperación de imágenes médicas con dos tecnologías distintas reporta: "existe una diferencia significativa (p <0,02) entre el tiempo invertido con la tecnología A ($33 \pm 4[s]$) que con la tecnología B ($30 \pm 6[s]$)", ¿significa que se debe adoptar la tecnología B? ¿Por qué?
16. Explica por qué se incrementa la probabilidad de cometer errores tipo I al cambiar de una prueba de hipótesis bilateral a otra unilateral.

CAPÍTULO 5. INFERENCIA CON MEDIAS MUESTRALES

En el capítulo 4 conocimos los principios de la inferencia y definimos los principales conceptos involucrados. En dicho capítulo conocimos el modelo normal, es decir, que la distribución muestral de la media sigue aproximadamente una distribución normal, supuesto que en general se cumple si la muestra tiene a lo menos 30 observaciones.

Veremos que diversas pruebas estadísticas consideran el modelo normal, aunque otras consideran estadísticos (estimaciones puntuales) diferentes que siguen otras distribuciones que ya conocimos en el capítulo 3.

En este capítulo veremos nuestras primeras pruebas estadísticas, las cuales nos permitirán inferir acerca de una o dos medias muestrales. Para ello nos basaremos principalmente en las explicaciones que ofrecen Diez y col. (2017, pp. 219-239) y Meena (2020).

5.1 PRUEBA Z

Como ya adelantamos, la prueba Z es adecuada para inferir acerca de las medias con una o dos muestras, aunque aquí solo veremos el primer caso. Para poder usarla, debemos **verificar el cumplimiento** de algunas condiciones, muchas de las cuales están asociadas al modelo normal que conocimos en el capítulo anterior:

- La muestra debe tener al menos 30 observaciones. Si la muestra tiene menos de 30 observaciones, se debe conocer la varianza de la población.
- Las observaciones deben ser independientes, es decir que la elección de una observación para la muestra no influye en la selección de las otras.
- La población de donde se obtuvo la muestra sigue aproximadamente una distribución normal.

Esta prueba resulta adecuada si queremos **asegurar** o **descartar** que la media de la población tiene un cierto **valor hipotético**. Supongamos que queremos saber si, en promedio, las utilidades mensuales de una pequeña empresa son de 20 millones de pesos y que el gerente general, Esteban Quito, nos ha informado que la desviación estándar para las utilidades es de 2,32 millones de pesos. El Sr. Quito nos ha proporcionado una muestra, obtenida mediante muestreo aleatorio simple, con las utilidades (en millones de pesos) reportadas para 20 meses, que se muestra en la tabla 5.1.

Obs.	Utilidad [M$]	Obs.	Utilidad [M$]	Obs.	Utilidad [M$]	Obs.	Utilidad [M$]
1	19,33	6	22,22	11	22,55	16	29,68
2	29,37	7	31,26	12	20,69	17	29,27
3	29,14	8	26,92	13	24,68	18	26,72
4	32,10	9	31,40	14	28,74	19	27,08
5	25,04	10	17,66	15	26,85	20	20,62

Tabla 5.1: muestra para el ejemplo de prueba Z con una muestra.

El Sr. Quito nos ha dicho que debemos ser muy exigentes con respecto a nuestras conclusiones, por lo que se decide usar un nivel de significación $\alpha = 0,01$ (es decir, un nivel de confianza de 99 %).

Comencemos por formular nuestras hipótesis:

H_0: la media de las utilidades mensuales de la empresa (μ) es de 20 millones de pesos, es decir: $\mu = 20$ [M$].

H_A: las utilidades mensuales de la empresa son, en promedio, distintas de 20 millones de pesos, es decir: $\mu \neq 20$ [M$].

Ahora debemos verificar el cumplimiento de las condiciones para poder usar la prueba Z. En cuanto a la primera condición, el enunciado nos indica que, si bien la muestra tiene solo 20 observaciones, la desviación estándar de la población es conocida, por lo que se verifica su cumplimiento.

También podemos comprobar en el enunciado que las observaciones son independientes entre sí, pues fueron obtenidas mediante muestreo aleatorio simple. Si bien no estamos seguros que que esta muestra considere menos del 10 % de las observaciones, podemos suponerlo razonablemente.

En cuanto a la distribución de la muestra, el gráfico Q-Q de la figura 5.1 (obtenido mediante el script 5.1) nos muestra que no se observan valores atípicos. Otra forma de comprobar esta condición es mediante la prueba de Shapiro-Wilk (Parada, 2019), que podemos realizar en R mediante la función `shapiro.test(x)`, donde `x` es un vector con las observaciones de la muestra. La hipótesis nula de esta prueba es que la muestra fue extraída desde una distribución normal (por ende, la hipótesis alternativa es que la distribución detrás de la muestra es diferente a la normal). Al ejecutar el script, podemos ver que el valor p obtenido es $p = 0,244$, muy superior a nuestro nivel de significación, por lo que podemos suponer con relativa confianza que la población de donde proviene la muestra sigue una distribución muestral.

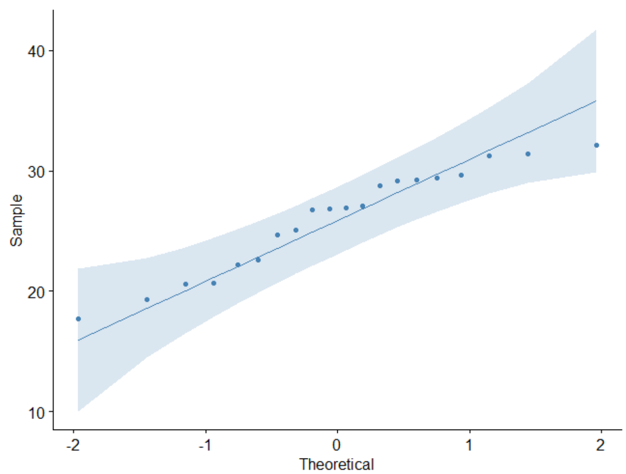


Figura 5.1: gráfico Q-Q para la muestra de la tabla 5.1.

Puesto que hemos comprobado que se cumplen todas las condiciones, podemos hacer una prueba Z para una muestra. Comencemos por calcular ahora el **estadístico de prueba** como ya hemos estudiado, usando para ello la ecuación 3.7:

$$Z = \frac{\overline{x} - \mu}{\sigma} = \frac{26,066 - 20}{2,32} = 2.6147$$

Con este resultado calculamos el valor p. Debemos recordar que las funciones de R (al igual que las antiguas tablas de probabilidades) nos entregan la probabilidad asociada al área correspondiente a una sola cola de la distribución, por lo que debemos multiplicar el resultado por 2 para considerar ambas colas si, como en este caso, se trata de una prueba bilateral. Al hacer la llamada `2 * pnorm(2.6147, lower.tail = FALSE)`, obtenemos que $p = 0,009 < 0.01$[1], con lo que se rechaza la hipótesis nula en favor de la hipótesis alternativa. Sin embargo, debemos ser cuidadosos puesto que el valor p es bastante cercano al nivel de significación establecido, por lo que sería prudente evaluar los resultados con una muestra más grande. Así, concluimos que los datos **sugieren** que, en promedio, las utilidades mensuales de la empresa difieren de los 20 millones de pesos establecidos.

[1]Más precisamente, $p = 0.008931758$, pero, por convención, p-valores suelen reportarse con tres decimales.

Desde luego, gracias a R podemos realizar esta prueba simplemente con una llamada a la función `z.test(x, mu, stdev, alternative, conf.level)`, disponible en el paquete `TeachingDemos`, donde:

- `x`: vector con las observaciones de la muestra.
- `mu`: valor nulo.
- `stdev`: desviación estándar de la población.
- `alternative`: tipo de hipótesis alternativa. Puede tomar los valores ‘‘`two.sided`’’ (hipótesis bilateral), ‘‘`less`’’ (hipótesis unilateral que la media de la población es menor que el valor nulo) o ‘‘`greater`’’ (hipótesis unilateral que la media de la población es mayor que el valor nulo).
- `conf.level`: nivel de confianza.

El script 5.1 muestra el desarrollo de este ejemplo en forma manual y luego, en la línea 42, usando la función `z.test()`. El resultado que se obtiene al usar esta función es el que se muestra en la figura 5.2, idéntico al obtenido en nuestro desarrollo previo.

```
     One Sample z-test

data:  media
z = 2.6147, n = 1.00, Std. Dev. = 2.32, Std. Dev. of the sample mean = 2.32, p-value = 0.008932
alternative hypothesis: true mean is not equal to 20
99 percent confidence interval:
 20.09008 32.04192
sample estimates:
mean of media
       26.066
```

Figura 5.2: resultado de la prueba Z para una muestra.

Script 5.1: prueba Z para una muestra.

```
1 library(TeachingDemos)
2 library(ggpubr)
3
4 # Ingresar los datos.
5 muestra <- c(19.33, 29.37, 29.14, 32.10, 25.04, 22.22, 31.26, 26.92,
6              31.40, 17.66, 22.55, 20.69, 24.68, 28.74, 26.85, 29.68,
7              29.27, 26.72, 27.08, 20.62)
8
9 # Establecer los datos conocidos.
10 desv_est <- 2.32
11 n <- length(muestra)
12 valor_nulo <- 20
13
14 # Crear gráfico Q-Q para verificar la distribución de la muestra,
15 datos <- data.frame(muestra)
16
17 g <- ggqqplot(datos, x = "muestra", color = "SteelBlue")
18 print(g)
19
20 # Verificar distribución muestral usando la prueba de normalidad
21 # de Shapiro-Wilk.
22 normalidad <- shapiro.test(muestra)
23 print(normalidad)
24
25 # Fijar un nivel de significación.
26 alfa <- 0.01
27
28 # Calcular la media de la muestra.
```

```r
cat("\tPrueba Z para una muestra\n\n")
media <- mean(muestra)
cat("Media =", media, "M$\n")

# Calcular el estadístico de prueba.
Z <- (media - valor_nulo) / desv_est
cat("Z =", Z, "\n")

# Calcular el valor p.
p <- 2 * pnorm(Z, lower.tail = FALSE)
cat("p =", p, "\n")

# Hacer la prueba Z con R.
prueba <- z.test(media, mu = valor_nulo, alternative = "two.sided",
                 stdev = desv_est, conf.level = 1-alfa)

print(prueba)
```

5.2 PRUEBA T DE STUDENT

En la práctica, rara vez podemos conocer la desviación estándar de la población y a menudo nos encontraremos con muestras pequeñas, por lo que la prueba Z no es muy utilizada.

En el caso de la media, el teorema del límite central se cumple para datos normales, es decir, independientemente del tamaño de la muestra, la media muestral tendrá una distribución cercana a la normal siempre que las observaciones sean independientes y provengan de una distribución cercana a la normal. Sin embargo, cuando el conjunto de datos es pequeño, resulta muy difícil comprobar el cumplimiento de estas condiciones.

En el capítulo 3 conocimos la distribución t de Student, o simplemente distribución t. Vimos que un aspecto destacado de esta distribución, siempre centrada en 0 y definida únicamente por los grados de libertad (ν) como parámetro, es su semejanza con la distribución normal pese a que sus colas son algo más gruesas. Este grosor adicional de las colas tiene como consecuencia que, para la distribución t, es más probable que una observación esté a más de dos desviaciones estándares de la media que en el caso de la distribución normal. Este fenómeno permite que la estimación del error estándar sea más certera que al usar la distribución normal cuando el conjunto de datos es pequeño.

La prueba t de Student, basada en la distribución t, es en consecuencia la alternativa más ampliamente empleada para inferir acerca de una o dos medias muestrales.

5.2.1 Prueba t para una muestra

Aunque la prueba t no opera bajo el supuesto de normalidad, aún así requiere verificar algunas condiciones para poder usarla:

1. Las observaciones son independientes entre sí.
2. Las observaciones provienen de una distribución cercana a la normal.

Podemos ver que estas condiciones son casi las mismas que para la prueba Z, excepto por el hecho de que no limitan el tamaño de la muestra para que sea mayor a 30. La ventaja evidente de eliminar esta restricción es que la distribución t permite su uso para muestras pequeñas, pero es igualmente adecuada cuando la muestra es grande. Esto se debe a que la forma de la distribución t es regulada por los grados de libertad y, a medida que aumentan, más se parece a una distribución normal. Este parámetro, al trabajar con medias de muestras de tamaño n, siempre estará dado por $\nu = n - 1$.

Tomemos el siguiente problema para ilustrar la prueba de hipótesis para la media de una muestra usando el modelo t: un ingeniero en Informática necesita determinar si el tiempo promedio que tarda una implementación dada de un algoritmo en resolver un problema, sabiendo que el algoritmo siempre se ejecuta en las mismas condiciones (misma máquina, igual disponibilidad de recursos de hardware y tamaño constante de las instancias), es inferior a 500 milisegundos. Para ello, ha seleccionado aleatoriamente 15 instancias del problema y registrado el tiempo de ejecución del algoritmo (en milisegundos) para cada una de ellas, como muestra la tabla 5.2.

Obs.	**t [ms]**	**Obs.**	**t [ms]**	**Obs.**	**t [ms]**
1	411,5538	6	388,6731	11	418,1169
2	393,2753	7	430,0382	12	408,4110
3	445,8905	8	469,4734	13	463,3733
4	411,4022	9	409,5844	14	407,0908
5	498,8969	10	442,0800	15	516,5222

Tabla 5.2: tiempo de ejecución para las instancias de la muestra.

El primer paso es formular las hipótesis:

H_0: el tiempo promedio que tarda el algoritmo en resolver una instancia del problema es igual a 500 milisegundos.
H_A: el tiempo promedio que tarda el algoritmo en resolver una instancia del problema es inferior a 500 milisegundos.

Recordemos que μ_0 es el valor nulo, por lo que en este caso $\mu_0 = 500$ [ms]. Matemáticamente, las hipótesis anteriores pueden formularse como:

Denotando como μ al tiempo medio que tarda la implementación del algoritmo en resolver una instancia cualquiera del problema:
H_0: $\mu = \mu_0$, esto es $\mu = 500$
H_A: $\mu < \mu_0$, es decir $\mu < 500$

Ahora debemos verificar que se cumplen las condiciones necesarias para usar la distribución t:

- Como las muestras fueron elegidas al azar, se puede asumir que son independientes.
- El gráfico de la figura 5.3 muestra que es válido suponer una distribución cercana a la normal. Si bien los puntos de la muestra no forman una recta, no se observan valores atípicos que se alejen de la región aceptable.

La media de la muestra es de $\overline{x} = 434,2921$, y la desviación estándar, $s = 38,0963$.

En este caso, el estadístico de prueba es el estadístico T, el cual sigue una distribución t con $\nu = n - 1$ grados de libertad y está dado por la ecuación 5.1, donde la subexpresión $(s/\sqrt{n})$ corresponde al error estándar de la media (cuando no se conoce la desviación estándar de la población, σ).

$$T = \frac{\overline{x} - \mu_0}{\frac{s}{\sqrt{n}}} \tag{5.1}$$

Así, para el ejemplo tenemos que:

$$T = \frac{434,2921 - 500}{\frac{38,0963}{\sqrt{15}}} = -6,6801$$

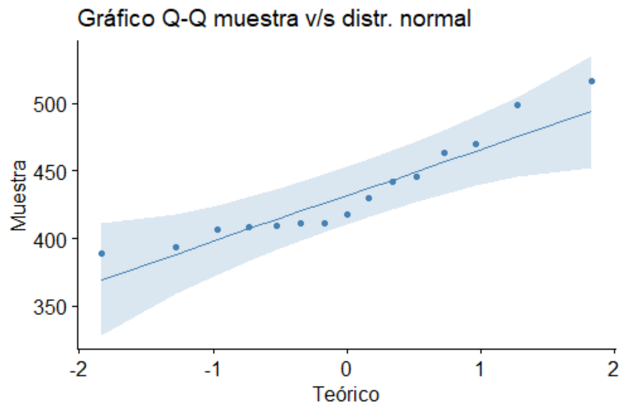


Figura 5.3: gráfico para comprobar el supuesto de normalidad.

A partir de este resultado, obtenemos el valor p con ayuda de la función `pt()`, obteniéndose $p = 5,219 \cdot 10^{-6}$, o simplemente, como dicta la convención, $p < 0,001$.

La fórmula para construir el intervalo de confianza usando la distribución t es ligeramente diferente al caso normal, como muestra la ecuación 5.2. Para este ejemplo consideraremos un nivel de confianza de 97,5 % (es decir, un nivel de significación $\alpha = 0,025$).

$$\overline{x} \pm t^*_\nu \cdot SE \tag{5.2}$$

Fijémonos en que en la ecuación 5.2 aparece el nuevo valor t^*_ν, el cual se obtiene a partir del nivel de confianza y la distribución t con ν grados de libertad (en este caso, $\nu = 14$), usando para ello una tabla de distribución t o la función `qt()` en R. Como puede verse al ejecutar el script 5.2, en este caso $t^*_\nu = 2,1448$.

Para el cálculo del error estándar, nuevamente se emplea la ecuación 4.2:

$$SE_{\overline{x}} = \frac{s}{\sqrt{n}} = \frac{38,0963}{\sqrt{15}} = 9,8364$$

Así, el intervalo de confianza está dado por:

$$(-\infty, t^*_{nu} * SE_{\overline{x}}] = (-\infty, 2,1448 * 9,8364] = (-\infty, 455,3892]$$

Una vez más, R permite realizar esta prueba de manera rápida y sencilla, gracias a la función `t.test(x, alternative, mu, conf.level)`, donde:

- `x`: vector no vacío de valores numéricos (la muestra).
- `alternative`: tipo de prueba de hipótesis. Los posibles valores son `"two.sided"` (prueba bilateral), `"greater"` (hipótesis unilateral que la media de la población es mayor que el valor nulo) o `"less"` (hipótesis unilateral que la media de la población es menor que el valor nulo).
- `mu`: valor nulo.
- `conf.level`: nivel de confianza.

El script 5.2 muestra el desarrollo en R para este ejemplo, incluyendo la construcción del gráfico de la figura 5.3, con iguales resultados al realizar la prueba paso a paso y con la función `t.test()`.

A partir de los resultados podemos observar que el valor p obtenido es muy pequeño, dando a entender que, si se cumple el supuesto de que la verdadera media es $\mu = 500$ [ms] (hipótesis nula), sería muy improbable obtener una media muestral de $\overline{x} = 434,2921$. Además, el valor p es muchísimo menor que el nivel de significación, por lo que la evidencia a favor de H_A es muy fuerte. En consecuencia, se rechaza H_0 en favor de H_A. Se puede afirmar, con 97,5 % de confianza, que el tiempo promedio que tarda el algoritmo en resolver una instancia del problema es inferior a 500 milisegundos.

Script 5.2: prueba t para una muestra.

```
library(ggpubr)

# Cargar los datos.
tiempo <- c(411.5538, 393.2753, 445.8905, 411.4022, 498.8969,
            388.6731, 430.0382, 469.4734, 409.5844, 442.0800,
            418.1169, 408.4110, 463.3733,407.0908, 516.5222)

# Establecer los datos conocidos.
n <- length(tiempo)
grados_libertad <- n - 1
valor_nulo <- 500


# Verificar si la distribución se acerca a la normal.
g <- ggqqplot(data = data.frame(tiempo),
              x = "tiempo",
              color = "steelblue",
              xlab = "Teórico",
              ylab = "Muestra",
              title = "Gráfico Q-Q muestra v/s distr. normal")

print(g)

# Fijar un nivel de significación.
alfa <- 0.025

# Calcular el estadístico de prueba.
cat("\tPrueba t para una muestra\n\n")
media <- mean(tiempo)
cat("Media =", media, "M$\n")
desv_est <- sd(tiempo)
error <- desv_est / sqrt(n)
t <- (media - valor_nulo) / error
cat("t =", t, "\n")

# Calcular el valor p.
p <- pt(t, df = grados_libertad, lower.tail = TRUE)
cat("p =", p, "\n")

# Construir el intervalo de confianza.
t_critico <- qt(alfa, df = grados_libertad, lower.tail = FALSE)
superior <- media + t_critico * error
cat("Intervalo de confianza = (-Inf, ", superior, "]\n", sep = "")

# Aplicar la prueba t de Student con la funcuón de R.
prueba <- t.test(tiempo,
                 alternative = "less",
                 mu = valor_nulo,
                 conf.level = 1 - alfa)

print(prueba)
```

5.2.2 Prueba t para dos muestras pareadas

Para esta prueba, supongamos ahora que el ingeniero en Informática del ejemplo anterior tiene dos algoritmos diferentes (A y B) que, en teoría, deberían tardar lo mismo en resolver un problema. Para ello, probó ambos algoritmos con 35 instancias del problema (elegidas al azar) de igual tamaño y registró los tiempos de ejecución (en milisegundos) de ambos algoritmos bajo iguales condiciones para cada una de ellas, además de calcular la diferencia en los tiempos de ejecución, como muestra la tabla 5.3. El ingeniero desea comprobar si efectivamente el rendimiento de ambos algoritmos es equivalente.

instancia	t_A **[ms]**	t_B **[ms]**	**dif [ms]**	**instancia**	t_A **[ms]**	t_B **[ms]**	**dif [ms]**
1	436,5736	408,5142	28,0594	19	438,5959	458,2536	-19,6577
2	470,7937	450,1075	20,6862	20	439,7409	474,9863	-35,2454
3	445,8354	490,2311	-44,3957	21	464,5916	496,0153	-31,4237
4	470,9810	513,6910	-42,7100	22	467,9926	485,8112	-17,8186
5	485,9394	467,6467	18,2927	23	415,3252	457,4253	-42,1001
6	464,6145	484,1897	-19,5752	24	495,4094	483,3700	12,0394
7	466,2139	465,9334	0,2805	25	493,7082	510,7131	-17,0049
8	468,9065	502,6670	-33,7605	26	433,1082	467,5739	-34,4657
9	473,8778	444,9693	28,9085	27	445,7433	482,5621	-36,8188
10	413,0639	456,3341	-43,2702	28	515,2049	453,5986	61,6063
11	496,8705	501,1443	-4,2738	29	441,9420	385,9391	56,0029
12	450,6578	471,7833	-21,1255	30	472,1396	548,7884	-76,6488
13	502,9759	441,1206	61,8553	31	451,2234	467,2533	-16,0299
14	465,6358	544,1575	-78,5217	32	476,5149	494,7049	-18,1900
15	437,6397	447,8844	-10,2447	33	440,7918	451,9716	-11,1798
16	458,8806	432,4108	26,4698	34	460,1070	522,3699	-62,2629
17	503,1435	477,1712	25,9723	35	450,1008	444,1270	5,9738
18	430,0524	482,4828	-52,4304				

Tabla 5.3: tiempos de ejecución de cada algoritmo para las instancias de la muestra.

Para este ejemplo, tenemos dos tiempos de ejecución diferentes para cada instancia del problema: uno con cada algoritmo. En consecuencia, los datos están **pareados**. Es decir, cada observación de un conjunto tiene una correspondencia o conexión especial con exactamente una observación del otro. Una forma de uso común para examinar datos pareados es usar la diferencia entre cada par de observaciones, para lo cual podemos usar la técnica de la distribución t (también llamada prueba t de Student) vista en la sección anterior.

La media de las diferencias es $\overline{x}_{dif} = -12,08591$ y la desviación estándar es $s_{dif} = 36,08183$.

Una vez más, comenzamos por formular las hipótesis:

H_0: la media de las diferencias en los tiempos de ejecución es igual a 0.
H_A: la media de las diferencias en los tiempos de ejecución es distinta de 0.

Que matemáticamente se expresan como:

Denotando la media de las diferencias en los tiempos de ejecución necesitados por ambos algoritmos para cualquier instancia del problema como μ_{dif}*:*
H_0: $\mu_{dif} = 0$
H_A: $\mu_{dif} \neq 0$

Como siguiente paso, verificamos el cumplimiento de las condiciones. Como las instancias fueron escogidas al azar, se puede suponer razonablemente que las observaciones son independientes, pues además el conjunto de instancias posibles es muy grande (o infinito) y las 35 seleccionadas no superan el 10 % de la población. Además, al aplicar una prueba de normalidad de Shapiro-Wilk (ver script 5.3, línea 23) se obtiene $p = 0,357$, con lo que podemos concluir que la diferencia en los tiempos de ejecución se acerca razonablemente a una distribución normal. En consecuencia, podemos proceder con la prueba t de Student. El ingeniero no necesita ser especialmente riguroso, por lo que usaremos un nivel de confianza del 95 %.

En este caso, la función `t.test()` de R permite efectuar la prueba de dos maneras diferentes (con idéntico resultado), como muestra el script 5.3. La primera de ellas (línea 30) es aplicar la prueba t directamente a las diferencias, tal como en la sección anterior (es decir, una prueba t para una muestra). La segunda (línea 39) consiste en entregar a la función ambas muestras por separado e indicarle que están pareadas. En este caso, la llamada tiene la forma `t.test(x, y, paired, alternative, mu, conf.level)`, donde los argumentos son:

- `x`: vector de valores numéricos para la primera muestra).
- `y`: vector de valores numéricos para la segunda muestra).
- `paired`: booleano (por defecto falso) que, cuando es verdadero, indica que ambas muestras están pareadas.
- `alternative`: tipo de prueba de hipótesis.
- `mu`: valor nulo.
- `conf.level`: nivel de confianza.

Script 5.3: inferencia con la media de las diferencias entre dos muestras pareadas usando la distribución t.

```
1  # Cargar los datos.
2  instancia <- seq(1, 35, 1)
3  
4  t_A <- c(436.5736, 470.7937, 445.8354, 470.9810, 485.9394,
5           464.6145, 466.2139, 468.9065, 473.8778, 413.0639,
6           496.8705, 450.6578, 502.9759, 465.6358, 437.6397,
7           458.8806, 503.1435, 430.0524, 438.5959, 439.7409,
8           464.5916, 467.9926, 415.3252, 495.4094, 493.7082,
9           433.1082, 445.7433, 515.2049, 441.9420, 472.1396,
10          451.2234, 476.5149, 440.7918, 460.1070, 450.1008)
11 
12 t_B <- c(408.5142, 450.1075, 490.2311, 513.6910, 467.6467,
13          484.1897, 465.9334, 502.6670, 444.9693, 456.3341,
14          501.1443, 471.7833, 441.1206, 544.1575, 447.8844,
15          432.4108, 477.1712, 482.4828, 458.2536, 474.9863,
16          496.0153, 485.8112, 457.4253, 483.3700, 510.7131,
17          467.5739, 482.5621, 453.5986, 385.9391, 548.7884,
18          467.2533, 494.7049, 451.9716, 522.3699, 444.1270)
19 
20 diferencia <- t_A - t_B
21 
22 # Verificar si la distribución se acerca a la normal.
23 normalidad <- shapiro.test(diferencia)
24 print(normalidad)
25 
26 # Fijar un nivel de significación.
27 alfa <- 0.05
28 
29 # Aplicar la prueba t de Student a la diferencia de medias.
30 prueba_1 <- t.test(diferencia,
31                    alternative = "two.sided",
32                    mu = valor_nulo,
33                    conf.level = 1 - alfa)
34 
35 print(prueba_1)
36 
37 # Otra alternativa puede ser aplicar la prueba t de Student
38 # para dos muestras pareadas.
39 prueba_2 <- t.test(x = t_A,
40                    y = t_B,
```

```
                      paired = TRUE,
                      alternative = "two.sided",
                      mu = valor_nulo,
                      conf.level = 1 - alfa)

print(prueba_2)
```

Los resultados para esta prueba son:

- El valor para el estadístico de prueba T es $t = -1,9816$.
- Se consideran $df = 34$ grados de libertad para la distribución t.
- El valor p obtenido es $p = 0,05565$.
- El intervalo de confianza obtenido es $[-24,4804542; 0,3086313]$.
- La media de la muestra es $\overline{x} = -12,08591$.

En este caso, la media de las diferencias está dentro del intervalo de confianza, y además el valor p es mayor que el nivel de significación, por lo que se falla al rechazar la hipótesis nula. Pero, nuevamente, el resultado está cerca del borde de significación. En consecuencia, se puede afirmar con 95 % de confianza que pareciera no haber diferencia entre los tiempos de ejecución de ambos algoritmos, aunque sería necesario conseguir una muestra más grande para tener mayor certeza.

5.2.3 Prueba t para dos muestras independientes

En este caso, la prueba t se usa para comparar las medias de dos poblaciones en que las observaciones con que se cuenta no tienen relación con ninguna de las otras observaciones, ni influyen en su selección, ni en la misma ni en la otra muestra. En este caso la inferencia se hace sobre la diferencia de las medias: $\mu_1 - \mu_2 = d_0$, donde d_0 es un valor hipotético fijo para la diferencia. Usualmente se usa $d_0 = 0$, en cuyo caso las muestras podrían provenir de dos poblaciones distintas con igual media, o desde la misma población. Para ello, la prueba usa como estimador puntual la diferencia de las medias muestrales $(\overline{x}_1 - \overline{x}_2)$. Así, el estadístico T en este caso toma la forma de la ecuación 5.3.

$$T = \frac{(\overline{x}_1 - \overline{x}_2) - d_0}{SE_{(\overline{x}_1 - \overline{x}_2)}} \tag{5.3}$$

Al usar la distribución t de Student para la diferencia de medias, se deben cumplir los siguientes requisitos:

1. Cada muestra cumple las condiciones para usar la distribución t.
2. Las muestras son independientes entre sí.

Veamos el funcionamiento de esta prueba con un ejemplo. El doctor E. L. Matta Sanno desea determinar si una nueva vacuna A es más efectiva que otra vacuna B, a fin de inmunizar a la población mundial contra una terrible enfermedad. Para ello, ha reclutado a un grupo de 28 voluntarios en diferentes países, 15 de los cuales (seleccionados al azar) recibieron la vacuna A y los 13 restantes, la vacuna B. La tabla 5.4 muestra, para cada voluntario, la concentración de anticuerpos (en microgramos por cada mililitro de sangre) al cabo de un mes de recibir la vacuna.

Las hipótesis a formular en este caso son:

H_0: no hay diferencia entre la efectividad promedio de ambas vacunas.
H_A: la vacuna A es, en promedio, más efectiva que la B.

En lenguaje matemático:

Anticuerpos [mg/ml]	
Vacuna A	Vacuna B
6,04	5,32
19,84	3,31
8,62	5,68
13,02	5,73
12,20	4,86
14,78	5,68
4,53	2,93
26,67	5,48
3,14	6,10
19,14	2,56
10,86	7,52
13,13	7,41
6,34	4,02
11,16	
7,62	

Tabla 5.4: Concentración de anticuerpos de los pacientes vacunados.

Si μ_A y μ_B son la concentraciones medias de anticuerpos presentes en personas luego de un mes de recibir la vacuna A y B, respectivamente, entonces:
H_0: $\mu_A = \mu_B$
H_A: $\mu_A > \mu_B$

Como es habitual, debemos ahora verificar el cumplimiento de las condiciones. Ambas muestras son independientes entre sí, pues son diferentes voluntarios y fueron designados aleatoriamente a cada grupo. Además, se puede asumir que las observaciones son independientes, pues cada muestra es significativamente menor a la población total a vacunar. En cuanto al supuesto de normalidad para cada muestra, al aplicar a cada una la prueba de Shapiro-Wilk (script 5.4, líneas 13 y 15) se obtiene, respectivamente, $p = 0,428$ y $p = 0,445$. En ambos casos el valor p es bastante alto, por lo que podemos concluir que ambas muestras provienen de poblaciones que se distribuyen de forma aproximadamente normal. Puesto que hemos verificado las condiciones, podemos llevar a cabo la prueba t para dos muestras independientes.

Ahora bien, como las muestras son algo pequeñas, sería prudente proceder con algo más de cautela. Además, en este escenario, un error tipo I (rechazar H_0 cuando es verdadera) implicaría reducir innecesariamente la cantidad de vacunas disponibles y retrasar el proceso de vacunación, poniendo en riesgo a todos los habitantes del planeta. Un error tipo II, en cambio, podría causar que se continúe el uso indistinto de ambas vacunas retrasando ligeramente el efecto inmune en la población. En consecuencia, el error tipo I es más grave, por lo que el nivel de significación debiese ser aún más exigente. En consecuencia, optaremos por $\alpha = 0,01$.

Al aplicar la prueba t (script 5.4), obtenemos que la diferencia entre las medias es $6,683$ [mg/ml] y que el intervalo de confianza es $[2,2739; \infty)$. Además, el valor p es $p < 0.001$, muy inferior al nivel de significación $\alpha = 0,01$. Esto significa que la evidencia en favor de H_A es muy fuerte, por lo rechazamos la hipótesis nula. En consecuencia, podemos concluir con 99 % de confianza que la vacuna A es, en promedio, mejor que la vacuna B (produce una mayor concentración media de anticuerpos en las personas vacunadas con ella que la producida por la vacuna B).

Script 5.4: prueba t para dos muestras independientes.

```
1 library(ggpubr)
2
3 # Cargar los datos.
4 vacuna_A <- c(6.04, 19.84, 8.62, 13.02, 12.20, 14.78, 4.53, 26.67,
5               3.14, 19.14, 10.86, 13.13, 6.34, 11.16, 7.62)
6
7 vacuna_B <- c(5.32, 3.31, 5.68, 5.73, 4.86, 5.68, 2.93, 5.48, 6.10,
```

```
              2.56, 7.52, 7.41, 4.02)

# Verificar si las muestras se distribuyen de manera cercana
# a la normal.
normalidad_A <- shapiro.test(vacuna_A)
print(normalidad_A)
normalidad_B <- shapiro.test(vacuna_B)
print(normalidad_B)

# Fijar un nivel de significación.
alfa <- 0.01

# Aplicar la prueba t para dos muestras independientes.
prueba <- t.test(x = vacuna_A,
                 y = vacuna_B,
                 paired = FALSE,
                 alternative = "greater",
                 mu = 0,
                 conf.level = 1 - alfa)

print(prueba)

# Calcular la diferencia entre las medias.
media_A <- mean(vacuna_A)
media_B <- mean(vacuna_B)
diferencia <- media_A - media_B
cat("Diferencia de las medias =", diferencia, "[mg/ml]\n")
```

Si estás leyendo atentamente, te habrás dado cuenta que ¡no hemos definido el error estándar para cuando tenemos dos muestras! En este caso, *SE* se construye a partir del error estándar de cada muestra, como se aprecia en la ecuación 5.4. En este escenario, la determinación de los grados de libertad es más compleja, por lo que se recomienda usar programas estadísticos o, en su defecto, escoger el menor valor entre $n_1 - 1$ y $n_2 - 1$.

$$SE_{\overline{x}_1 - \overline{x}_2} = \sqrt{SE^2_{\overline{x}_1} + SE^2_{\overline{x}_2}} = \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}} \tag{5.4}$$

Se puede lograr un mejor ajuste de la distribución t si se sabe con certeza que las desviaciones estándares de ambas poblaciones son casi iguales. En este caso, se puede usar una **varianza agrupada** (s_p^2, del inglés *pooled variance*) que reemplaza tanto a s_1^2 como a s_2^2 en la ecuación 5.4. Esta varianza agrupada se calcula como muestra la ecuación 5.5 y, en este caso, se consideran $n_1 + n_2 - 2$ grados de libertad.

$$s_p^2 = \frac{s_1^2 \cdot (n_1 - 1) + s_2^2 \cdot (n_2 - 1)}{n_1 + n_2 - 2} \tag{5.5}$$

Por defecto, R utiliza la corrección de Welch para la prueba t de Student de la diferencia de dos medias, variante considerada más segura, que en general entrega resultados muy similares a la versión original de la prueba cuando las muestras tienen varianzas similares. No obstante, los resultados son bastante mejores cuando los tamaños de las muestras y sus desviaciones estándares son muy diferentes (Kassambara, 2019a). La corrección de Welch calcula el error estándar como muestra la ecuación 5.4, pero ajusta los grados de libertad de acuerdo a la ecuación 5.6.

$$\nu = \frac{\left(\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}\right)^2}{\frac{s_1^4}{n_1^2(n_1-1)} + \frac{s_2^4}{n_2^2(n_2-1)}} \tag{5.6}$$

5.3 EJERCICIOS PROPUESTOS

1. Investiga acerca de la prueba de Kolmogorov-Smirnov y explica cómo puede usarse para verificar si una distribución se asemeja a la normal. Compara esta prueba con la de Shapiro-Wilk.
2. Para confirmar que el tiempo que requieren los estudiantes de ingeniería para desarrollar una guía de ejercicios de Cálculo I es de dos horas, se eligió aleatoriamente a 16 estudiantes de esta asignatura y se les pidió anotar el tiempo [min.] invertido en la tarea. Los resultados fueron los siguientes: 140,6; 133,3; 142,4; 86,4; 129,9; 110,8; 133,2; 129,1; 142,5; 150,2; 141,6; 111,0; 127,2; 137,9; 131,9; 121,9.
   a) Enuncia las hipótesis nula y alternativa a contrastar.
   b) Analiza si es razonable en este caso considerar que los datos cumplen las condiciones para usar una prueba t de Student.
   c) Independientemente del resultado anterior, aplica la prueba propuesta y obtenga un intervalo de confianza y un valor p.
   d) Usando un nivel de significación adecuado, entrega una conclusión para la cuestión planteada.
3. El departamento de control de calidad de un importante laboratorio requiere analizar la concentración de ingredientes activos presente en una muestra de 10 botellas diferentes de detergente líquido que ellos seleccionaron aleatoriamente en el último mes. Como se sospecha que esta concentración depende del catalizador que se use, la mitad del contenido de cada botella fue sometida a un catalizador, y la otra mitad a otro catalizador. En orden por botella seleccionada, los resultados fueron:
   - Catalizador 1: 62,9; 67,2; 67,4; 67,4; 67,2; 64,6; 69,6; 65,7; 68,2; 72,0.
   - Catalizador 2: 66,8; 69,3; 69,6; 67,3; 68,8; 68,4; 68,6; 70,3; 69,6; 71,7.

   a) Como primer paso, el departamento de control de calidad necesita saber si la concentración media de concentraciones de ingredientes activos depende del catalizador elegido.
   b) Propón las hipótesis nula y alternativa que permitan responder el problema planteado con una prueba t de Student.
   c) Muestra que es razonable considerar que estos datos cumplen las condiciones para usar la prueba propuesta y fija un nivel de significación apropiado.
   d) Aplica la prueba propuesta y obtenga un intervalo de confianza y un valor p.
   e) ¿Cuál sería tu respuesta al departamento de control mencionado?
4. Una fábrica de detectores de radón recibió consultas de sus clientes sobre si era conveniente comprar su nuevo modelo de detectores Radolmes+®para reemplazar los antiguos aparatos Radolmes®en su poder. Si bien los técnicos están seguros que la inversión es conveniente, la gerencia decidió hacer un estudio previo a la recomendación. Para esto, se introdujeron en una tómbola oscura los números de serie de los aparatos producidos en los últimos meses de ambos modelos y se seleccionaron 26 números sin mirar y girando la tómbola cinco veces entre cada selección, resultando escogidos 12 aparatos Radolmes y 14 aparatos Radolmes+. Luego, cada detector seleccionado se expuso a 100 pCi/l de radón. Las lecturas resultantes fueron las siguientes:
   - Radolmes: 105,6; 100,1; 90,9; 105,0; 91,2; 99,6; 96,9; 107,7; 96,5; 103,3; 91,3; 92,4.
   - Radolmes+: 98,9; 94,3; 95,9; 107,7; 102,0; 94,2; 100,6; 98,5; 99,1; 101,3; 94,4; 103,6; 95,3; 106,7.

   a) ¿Qué hipótesis nula y alternativa se deberían docimar[2] con una prueba t de Student para responder a la inquietud planteada?
   b) ¿Cumplen los datos obtenidos las condiciones para usar esta prueba t de Student?
   c) Aplicando la prueba t de Student para este caso, obtén un intervalo de confianza y un valor p.
   d) ¿Qué aconsejarías a los directivos de la fábrica?

[2]Término que suele ocuparse en estadística como sinónimo de "probar".

CAPÍTULO 6. PODER ESTADÍSTICO

En el capítulo 4 estudiamos el procedimiento para someter hipótesis a prueba, junto con los errores de decisión que podríamos cometer:

- Error tipo I: rechazar H_0 en favor de H_A cuando H_0 es en realidad verdadera.
- Error tipo II: no rechazar H_0 en favor de H_A cuando H_A es en realidad verdadera.

Allí conocimos el nivel de significación, α, como herramienta para representar y, de alguna manera, controlar la probabilidad de cometer un error de tipo I, con lo que la preocupación se centra en controlar la ocurrencia de esta clase de errores, desviando la atención de los errores de tipo II. Esto se debe a que la hipótesis nula representa el *status quo*, es decir, mantener las cosas y creencias tal como están y, por ende, cuando no se rechaza H_0, no suele requerirse tomar ninguna acción. En contraste, la hipótesis alternativa describe un cambio de condiciones, por lo que rechazar H_0 en favor de H_A usualmente conlleva un esfuerzo, mayor costo, para adaptarse o aprovechar las nuevas condiciones.

Sin embargo, en el capítulo 4 también vimos que el valor de α debe ser acorde con las consecuencias de cometer errores tanto de tipo I como de tipo II, ¡pero no sabemos cómo se relaciona el nivel de significación con los errores de tipo II!

Así como el nivel de significación α corresponde a la probabilidad de cometer errores de tipo I, definimos ahora β como la probabilidad de cometer errores de tipo II. α y β están relacionados: **para un tamaño fijo de la muestra: al reducir $\boldsymbol{\beta}$, $\boldsymbol{\alpha}$ aumenta, y viceversa**. Este fenómeno se evidencia con mayor fuerza mientras más pequeña sea la muestra. No obstante, en la práctica resulta más interesante conocer la probabilidad de **no** cometer errores de tipo II. Esto nos lleva a un nuevo concepto: el **poder estadístico** de una prueba de hipótesis, dado por $1-\beta$, que se define como **la probabilidad de correctamente rechazar H_0 cuando es falsa**.

Otra forma de entender la noción de poder de una prueba es qué tan propensa es esta para distinguir un efecto real de una simple casualidad, lo que nos lleva a la noción de **tamaño del efecto**, que corresponde a una cuantificación de la diferencia entre dos grupos, o del valor observado con respecto al valor nulo.

En el capítulo 5 conocimos la prueba t para inferir acerca de dos medias. En este contexto, el tamaño del efecto corresponde a qué tan grande es la diferencia real entre ambas. Si quieres aprender más sobre estos conceptos, puedes consultar las fuentes en las que se basa este capítulo: Diez y col. (2017, pp. 239-245) y Freund y Wilson (2003, pp. 123-138).

6.1 PODER, NIVEL DE SIGNIFICACIÓN Y TAMAÑO DE LA MUESTRA

En la introducción de este capítulo vimos que el poder corresponde a la probabilidad de **no** cometer un error de tipo II, y que está muy relacionado con el tamaño de la muestra. También mencionamos que existe una relación entre el poder y el nivel de significación, la cual exploraremos en esta sección.

La figura 6.1 (creada mediante el script 6.1) muestra cuatro curvas de poder para la prueba t de Student de una muestra con desviación estándar $s = 1$ y valor nulo $\mu_0 = 0$. En ella, el tamaño del efecto está representada en la misma escala de la variable, aunque en la sección siguiente veremos otra alternativa. La curva roja considera $\alpha = 0,05$ y $n = 6$; la azul, $\alpha = 0,01$ y $n = 6$; la verde, $\alpha = 0,05$ y $n = 10$, y la naranja, $\alpha = 0,01$ y $n = 10$. En ella podemos observar que:

- El poder de la prueba aumenta mientras mayor es el tamaño del efecto (en este caso, la distancia entre el valor nulo y la media de la muestra).
- A medida que el tamaño del efecto disminuye (es decir, el estimador se acerca al valor nulo), el poder se aproxima al nivel de significación.
- Usar un valor de α más exigente (menor), manteniendo constante el tamaño de la muestra, hace que la curva de poder sea más baja para cualquier tamaño del efecto (lo que verifica la relación entre α y β).
- Usar una muestra más grande aumenta el poder de la prueba para cualquier tamaño del efecto distinto de 0.

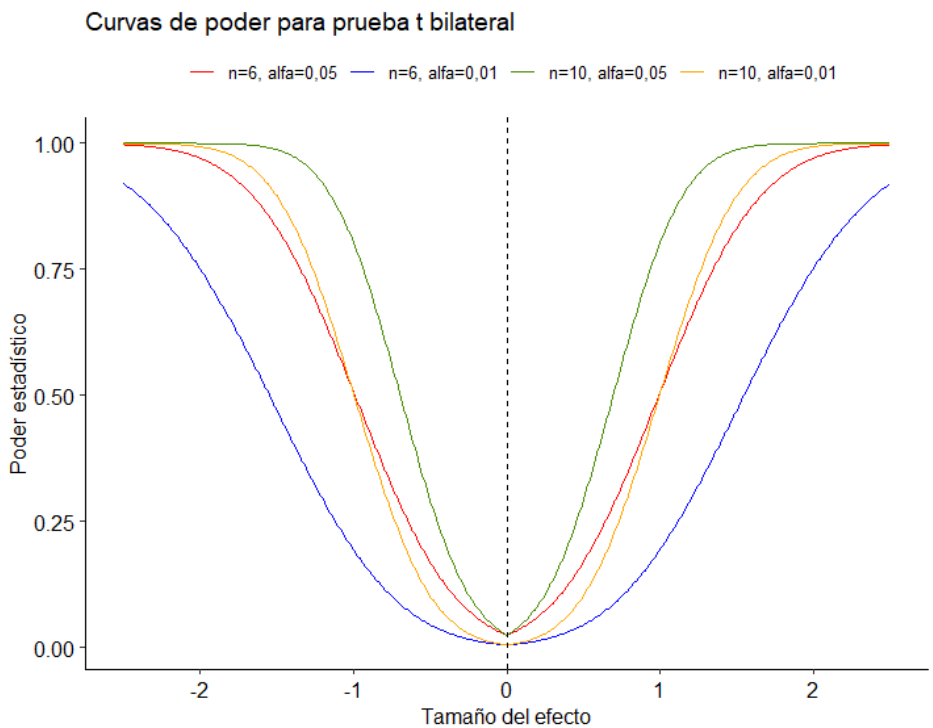


Figura 6.1: poder estadístico para prueba t bilateral.

De manera similar, la figura 6.2 considera las mismas muestras y los mismos niveles de significación que la figura 6.1, pero ahora para una prueba t unilateral. En ella se aprecia que la gran desventaja de las pruebas unilaterales es que el poder tiende a cero a medida que el tamaño del efecto aumenta en sentido contrario a la hipótesis alternativa, por lo que no sería posible detectar una diferencia en el sentido opuesto aunque fuese muy grande (pues no hay una región de rechazo en dicho sentido). El script empleado para la construcción de la figura 6.2 es idéntico al script 6.1, excepto porque el argumento `alternative` toma como valor `‘‘one.sided’’` en las llamadas a `power.t.test()`.

Script 6.1: poder estadístico para prueba t bilateral.

```
library(ggpubr)
library(tidyverse)

# Generar un vector con un rango de valores para la efecto
# de medias.
efecto <- seq(-2.5, 2.5, 0.01)

# Calcular el poder para una prueba t bilareral, para cada tamaño
# del efecto, asumiendo una muestra con desviación estándar igual a 1.
# Se consideran 4 escenarios para calcular el poder:
# 1. Una muestra de tamaño 6 y nivel de significación 0.05.
```

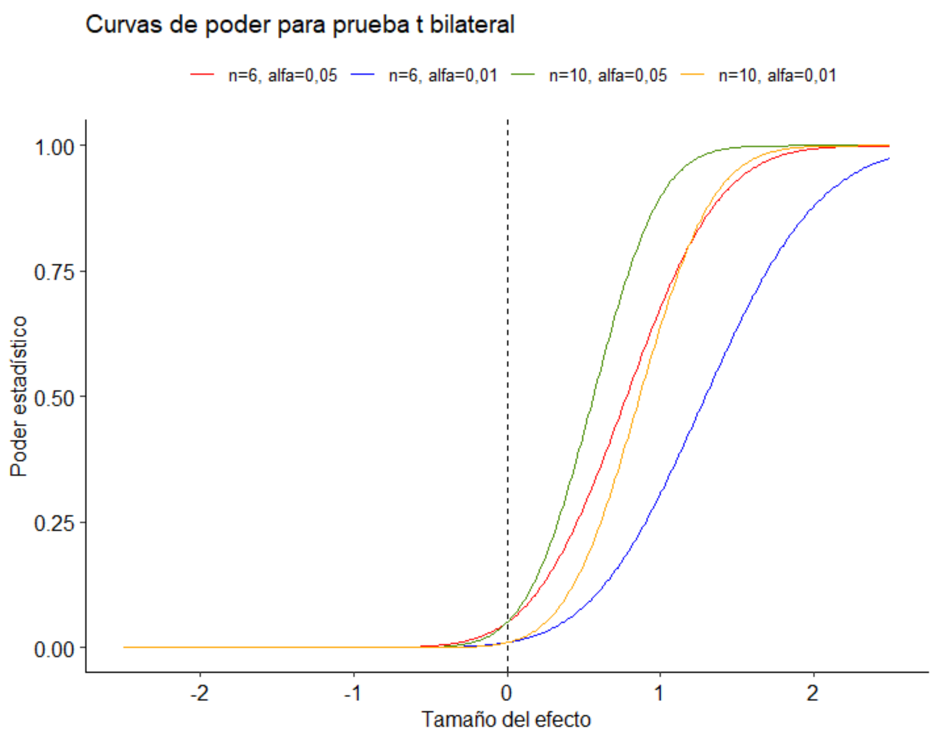


Figura 6.2: poder estadístico para prueba t unilateral.

```
# 2. Una muestra de tamaño 6 y nivel de significación 0.01.
# 3. Una muestra de tamaño 10 y nivel de significación 0.05.
# 4. Una muestra de tamaño 10 y nivel de significación 0.01.
n_6_alfa_05 <- power.t.test(n = 6,
                      delta = efecto,
                      sd = 1,
                      sig.level = 0.05,
                      type = "one.sample",
                      alternative = "two.sided")$power

n_6_alfa_01 <- power.t.test(n = 6,
                      delta = efecto,
                      sd = 1,
                      sig.level = 0.01,
                      type = "one.sample",
                      alternative = "two.sided")$power

n_10_alfa_05 <- power.t.test(n = 10,
                       delta = efecto,
                       sd = 1,
                       sig.level = 0.05,
                       type = "one.sample",
                       alternative = "two.sided")$power

n_10_alfa_01 <- power.t.test(n = 10,
                             delta = efecto,
                             sd = 1,
                             sig.level = 0.01,
                             type = "one.sample",
                             alternative = "two.sided")$power
```

```r

# Construir matriz de datos en formato ancho.
datos <- data.frame(efecto, n_6_alfa_05, n_6_alfa_01,
                    n_10_alfa_05, n_10_alfa_01)

# Llevar a formato largo.
datos <- datos %>% pivot_longer(!"efecto",
                                names_to = "fuente",
                                values_to = "poder")

# Formatear fuente como variable categórica.
niveles <- c("n_6_alfa_05", "n_6_alfa_01", "n_10_alfa_05",
             "n_10_alfa_01")

etiquetas <- c("n=6, alfa=0,05", "n=6, alfa=0,01", "n=10, alfa=0,05",
               "n=10, alfa=0,01")

datos[["fuente"]] <- factor(datos[["fuente"]], levels = niveles,
                            labels = etiquetas)

# Graficar las curvas de poder.
g <- ggplot(datos, aes(efecto, poder,  colour = factor(fuente)))
g <- g + geom_line()
g <- g + labs(colour = "")
g <- g + ylab("Poder estadístico")
g <- g + xlab("Tamaño del efecto")

g <- g + scale_color_manual(values=c("red", "blue", "chartreuse4",
                                     "orange"))

g <- g + theme_pubr()
g <- g + ggtitle("Curvas de poder para prueba t bilateral")
g <- g + geom_vline(xintercept = 0, linetype = "dashed")

print(g)
```

La figura 6.3 muestra las curvas de poder para una prueba t unilateral y otra bilateral, ambas para una muestra de tamaño 6, desviación estándar $s = 1$ y $\alpha = 0,05$. En ella se evidencia claramente la ventaja de las pruebas unilaterales: cuando el tamaño del efecto aumenta en el sentido de la hipótesis alternativa, el poder es mayor que para una prueba bilateral.

Es deseable que las pruebas que se empleen para docimar hipótesis tengan un alto poder y, si hay más de una prueba disponible, se debe escoger la más poderosa. No obstante, los cálculos del poder suelen ser altamente complejos. Afortunadamente, la teoría permite en muchos casos conocer la prueba con mayor poder posible ante cualquier hipótesis alternativa, nivel de significación y tamaño de muestra (siempre que se cumplan las condiciones de base). Estas pruebas reciben el nombre de **uniformemente más poderosas**, y tal es el caso de la prueba t de Student.

6.2 TAMAÑO DEL EFECTO

El problema que podríamos tener al considerar el tamaño del efecto en la misma escala de la variable estudiada, como hemos hecho hasta ahora, es que esta escala varía de variable en variable. Para poder hacer comparaciones con mayor libertad, existen diferentes **medidas estandarizadas de efecto** que podemos

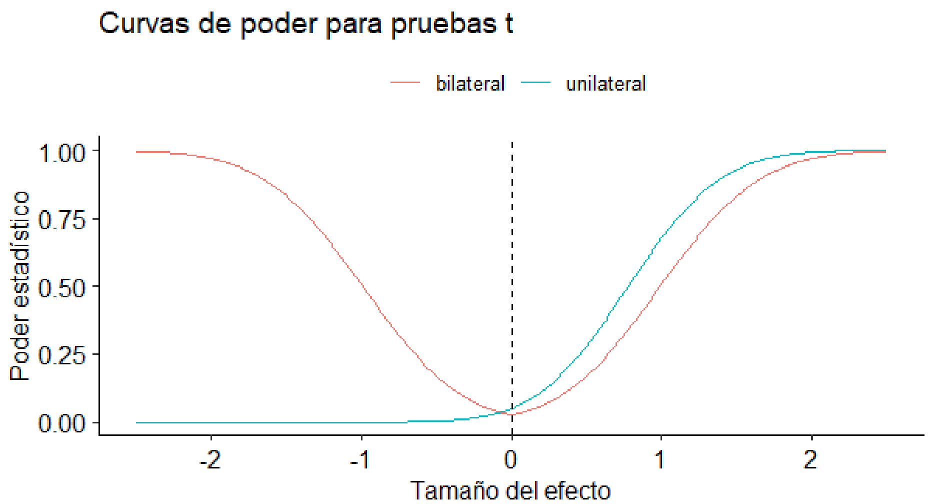


Figura 6.3: poder estadístico para pruebas t.

usar. Puesto que hasta ahora solo hemos estudiado la prueba t de Student, en esta sección conoceremos la llamada ***d* de Cohen** (Kassambara, 2019b), una medida estándar ampliamente empleada para el tamaño del efecto con esta prueba.

En términos generales, se considera que $d = 0,2$ es un efecto pequeño (imperceptible a simple vista), $d = 0,5$ es un efecto mediano (probablemente perceptible a simple vista) y $d = 0,8$, un efecto grande (definitivamente perceptible a simple vista).

En el caso de la prueba t de una muestra, la d de Cohen se calcula como muestra la ecuación 6.1, donde:

- $\overline{x}$: media muestral.
- μ_0: media teórica para el contraste (valor nulo).
- s: desviación estándar de la muestra con $n - 1$ grados de libertad.

$$d = \frac{\overline{x} - \mu_0}{s} \tag{6.1}$$

Para la prueba t de diferencia de dos medias (también llamada prueba t para dos muestras independientes o, simplemente, prueba t independiente), si el tamaño de la muestra es mayor a 50 elementos, se calcula como muestra la ecuación 6.2, y para muestras pequeñas se aplica un factor de corrección, como indica la ecuación 6.3, donde:

- $\overline{x}_1$, $\overline{x}_2$: medias muestrales de cada grupo.
- n_1 y n_2 son los tamaños de ambas muestras.
- s_p: desviación estándar agrupada, dada por la ecuación 6.4[1].

$$d = \frac{\overline{x}_1 - \overline{x}_2}{s_p} \tag{6.2}$$

$$d = \frac{\overline{x}_1 - \overline{x}_2}{s_p} \cdot \frac{n_1 + n_2 - 3}{n_1 + n_2 - 2,25} \tag{6.3}$$

$$s_p = \sqrt{\frac{\sum(x - \overline{x}_1)^2 + \sum(x - \overline{x}_2)^2}{n_1 + n_2 - 2}} \tag{6.4}$$

[1]Note que esta corresponde a la raíz de la varianza agrupada descrita en 5.5

En el caso de la variante de Welch para la prueba t independiente, la fórmula para el cálculo de la d de Cohen es ligeramente diferente, como puede apreciarse en la ecuación 6.5.

$$d = \frac{\overline{x}_1 - \overline{x}_2}{\sqrt{\frac{s_1^2 + s_2^2}{2}}} \tag{6.5}$$

Por último, las ecuaciones 6.6 y 6.7 muestran cómo se calcula la d de Cohen en el caso de la prueba t con muestras pareadas grandes ($n > 50$) y pequeñas, respectivamente, donde D corresponde a las diferencias entre las observaciones pareadas.

$$d = \frac{\overline{x}_D}{s_D} \tag{6.6}$$

$$d = \frac{\overline{x}_D}{s_D} \cdot \frac{n_1 - 2}{n_1 - 1,25} \tag{6.7}$$

6.3 PODER, TAMAÑO DEL EFECTO Y TAMAÑO DE LA MUESTRA

Mencionamos en páginas anteriores que el poder puede también entenderse como qué tan propensa es una prueba estadística para distinguir un efecto real de una simple casualidad, y que podemos cuantificar este efecto.

Una gran ventaja del poder estadístico es que nos sirve para determinar el tamaño adecuado de la muestra para detectar un cierto tamaño del efecto. La figura 6.4, elaborada con el script 6.2, muestra el aumento del poder estadístico a medida que el tamaño de la muestra aumenta (para un tamaño del efecto y nivel de significación fijos). En ella se aprecia que, a medida que el tamaño de la muestra crece, el poder estadístico también crece asintóticamente a 1, valor que equivale a tener la certeza de rechazar la hipótesis nula si esta es falsa.

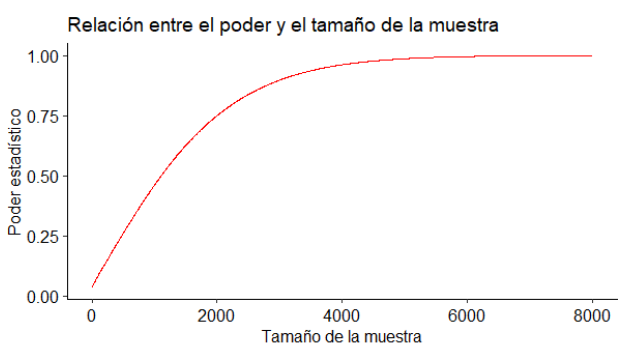


Figura 6.4: aumento del poder estadístico a medida que crece el tamaño de la muestra (manteniendo fijos el tamaño del efecto y el nivel de significación).

Script 6.2: aumento del poder estadístico a medida que crece el tamaño de la muestra.

```
library(ggpubr)

# Generar un vector con un rango para el tamaño de la muestra.
n <- seq(5, 8000, 5)
```

```r

# Definir constantes
desv_est <- 6
alfa <- 0.05
tam_efecto <- 0.5

# Se calcula el poder con que se detecta el tamaño del efecto para
# cada tamaño de la muestra, asumiendo una prueba bilateral para
# una sola muestra.
poder <- power.t.test(n = n,
                      delta = tam_efecto,
                      sd = desv_est,
                      sig.level = alfa,
                      type = "two.sample",
                      alternative = "two.sided")$power

# Crear un data frame.
datos <- data.frame(n, poder)

# Graficar la curva de poder.
g <- ggplot(datos, aes(n, poder))
g <- g + geom_line(colour = "red")
g <- g + ylab("Poder estadístico")
g <- g + xlab("Tamaño de la muestra")
g <- g + theme_pubr()
g <- g + ggtitle("Relación entre el poder y el tamaño de la muestra")

print(g)
```

6.4 CÁLCULO TEÓRICO DEL PODER

Como ya hemos mencionado a lo largo de este capítulo, el poder es la probabilidad de correctamente rechazar H_0 cuando es falsa, lo que equivale a la probabilidad de distinguir un efecto real de una mera casualidad. Ahora veremos algunos ejemplos de cómo podemos usar el poder.

Lola Drones, estudiante de computación, ha diseñado dos nuevos algoritmos (A y B) que resuelven un mismo problema como parte de su trabajo de titulación. Lola desea saber si existe diferencia entre los tiempos de ejecución de ambos algoritmos. Para ello, ha decidido realizar una prueba t con muestras pareadas, con un nivel de significación $\alpha = 0,05$, usando para ello 36 instancias del problema de tamaño fijo que se ejecutan bajo iguales condiciones con cada algoritmo. Además, Lola ya sabe que la diferencia en el tiempo de ejecución sigue una distribución normal con desviación estándar $\sigma = 12$ milisegundos. Así, Lola ha formulado las siguientes hipótesis:

H_0: $\mu_{(A_i - B_i)} = 0$, es decir que la media de las diferencias en el tiempo de ejecución necesitado por los algoritmos A y B, para cada posible instancia i, es cero

H_A: $\mu_{(A_i - B_i)} \neq 0$

La figura 6.5 muestra cómo sería la distribución de la muestra (media de las diferencias en los tiempos de ejecución) si la hipótesis nula (H_0) fuese cierta, con las áreas correspondientes a la región de rechazo de H_0 coloreadas.

Supongamos por un momento que, en realidad, el algoritmo B es en promedio 4 milisegundos más rápido que el algoritmo A. En este caso tendríamos que la media de las diferencias es de -4 [ms], correspondiente

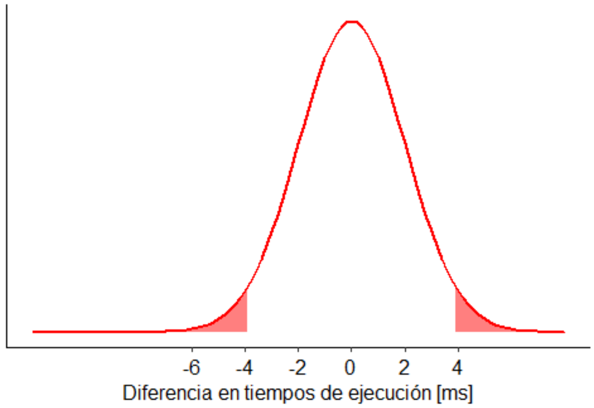


Figura 6.5: distribución de la diferencia de medias del tiempo de ejecución, señalando zonas de rechazo de la hipótesis nula.

al tamaño del efecto. En este caso, su distribución sería como muestra la figura 6.6 (ver script 6.3) en color azul. Al superponer esta nueva curva a la que ya teníamos bajo el supuesto de que la hipótesis nula fuera verdadera, vemos que el área de la curva real que se situaría dentro de la región de rechazo de la curva teórica es aquella coloreada en azul. Esta área corresponde al poder de la prueba t, que en este caso es de 0,516 de acuerdo al análisis teórico (ver script 6.1, líneas 77–86). Puesto que el poder corresponde a la probabilidad de **no** cometer un error de tipo II, de acuerdo al resultado obtenido se tiene que $\beta = 0,484$. ¡Lola no sería capaz de detectar una diferencia de -4 [ms] casi la mitad de las veces!

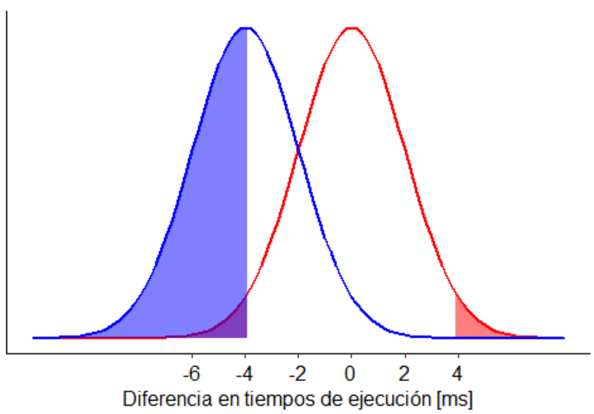


Figura 6.6: región de rechazo de la hipótesis nula en la distribución cuando el programa B es, en promedio, 4 milisegundos más rápido que el programa A.

Script 6.3: cálculo teórico del poder.

```
library(ggpubr)
library(pwr)

# Fijar valores conocidos.
sigma <- 12
alfa <- 0.05
n <- 36

# Calcular el error estándar.
SE <- sigma / sqrt(n)

# Gráficar la distribución muestral de la media de las diferencias si
```

```
# la hipótesis nula fuera verdadera.
x <- seq(-6 * SE, 4 * SE, 0.01)
y <- dnorm(x, mean = media_nula, sd = SE)
g <- ggplot(data = data.frame(x, y), aes(x))

g <- g + stat_function(
  fun = dnorm,
  args = list(mean = media_nula, sd = SE),
  colour = "red", size = 1)

g <- g + ylab("")
g <- g + scale_y_continuous(breaks = NULL)
g <- g + scale_x_continuous(name = "Diferencia en tiempos de ejecución [ms]",
                            breaks = seq(-6, 4, 2))

g <- g + theme_pubr()

# Colorear la región de rechazo de la hipótesis nula.
media_nula <- 0
Z_critico <- qnorm(alfa/2, mean = media_nula, sd = SE, lower.tail = FALSE)
q_critico_inferior <- media_nula - Z_critico
q_critico_superior <- media_nula + Z_critico

g <- g + geom_area(data = subset(df, x < q_critico_inferior),
                   aes(y = y),
                   colour = "red",
                   fill = "red",
                   alpha = 0.5)

g <- g + geom_area(data = subset(df, x > q_critico_superior),
                   aes(y = y),
                   colour = "red",
                   fill = "red",
                   alpha = 0.5)

print(g)

# Superponer la distribución muestral de la media de las diferencias
# si la la diferencia de medias fuera -4.
g <- g + stat_function(
  fun = dnorm,
  args = list(mean = media_efecto, sd = SE),
  colour = "blue", size = 1)

# Colorear la región de la nueva curva situada en la región de
# rechazo de la curva original.
x1 <- seq(-6 * SE, 4 * SE, 0.01)
y1 <- dnorm(x, mean = media_efecto, sd = SE)
g <- g + geom_area(data = subset(data.frame(x1, y1),
                                 x < q_critico_inferior),
                   aes(x = x1, y = y1),
                   colour = "blue",
                   fill = "blue",
                   alpha = 0.5)

g <- g + geom_area(data = subset(data.frame(x1, y1),
                                 x > q_critico_superior),
                   aes(x = x1, y = y1),
                   colour = "blue",
```

```
                    fill = "blue",
                    alpha = 0.5)
print(g)

# Calcular el poder de acuerdo al análisis teórico.
poder <- pnorm(q_critico_inferior,
               mean = media_efecto,
               sd = SE,
               lower.tail = TRUE)
+ pnorm(q_critico_superior,
        mean = media_efecto,
        sd = SE,
        lower.tail =  FALSE)

cat("Poder = ", poder, "\n")

# Calcular la probabilidad de cometer un error tipo II.
beta <- 1 - poder_teorico
cat("Beta = ", beta, "\n")
```

6.5 CÁLCULO DEL PODER EN R

Desde luego, si trabajamos con R, podemos usar funciones para calcular el poder. Como primera alternativa, R trae incorporada la función `power.t.test(n, delta, sd, sig.level, power, type, alternative)` (empleada en los scripts 6.1 y 6.2), donde:

- `n`: tamaño de la muestra (por cada grupo, si corresponde).
- `delta`: diferencia observada entre las medias, o entre la media muestral y el valor nulo, no estandarizada.
- `sd`: desviación estándar observada.
- `sig.level`: nivel de significación.
- `power`: poder de la prueba.
- `type`: tipo de prueba t de Student (“`two.sample`” para diferencia de medias, “`one.sample`” para una sola muestra o “`paired`” para dos muestras pareadas).
- `alternative`: tipo de hipótesis alternativa (“`one.sided`” si es unilateral, “`two.sided`” si es bilateral).

Esta función entrega como resultado un objeto con diversos elementos (que podemos indexar del mismo modo que las columnas de una matriz de datos), entre los que se incluyen los 5 primeros argumentos definidos para la función.

Si revisamos con detenimiento los argumentos de la función `power.t.test()`, veremos que ¡recibe el poder como uno de sus argumentos! Esto no parece tener sentido... ¿o sí?

Como ya hemos visto existe una relación entre: poder, tamaño de la muestra, tamaño del efecto y nivel de significación. A esta combinación de elementos debemos añadir también la desviación estándar, aunque no estudiaremos las matemáticas subyacentes.

En realidad, para usar la función `power.t.test()` siempre debemos señalar el tipo de prueba t con el que estamos trabajando y si la hipótesis alternativa es de una o dos colas. Esta función nos permite calcular cualquiera de los demás argumentos (tamaño de la muestra, tamaño del efecto, desviación estándar, nivel de significación o poder estadístico) para la prueba en cuestión a partir de los 4 argumentos restantes. Así, al argumento que queremos calcular se le asigna el valor `NULL` en la llamada.

Recordemos que en el ejemplo de la sección anterior, Lola Drones desea usar una prueba t bilateral para dos muestras pareadas a fin de determinar si hay diferencia entre los tiempos de ejecución promedio de ambos algoritmos. Para ello, ha considerado $n = 36$ y $\alpha = 0,05$, sabiendo que $sd = 12$ [ms]. Las líneas 4 a 14 del script 6.4 muestran cómo calcular el poder para este ejemplo si se desea detectar un tamaño del efecto (δ) de 4 [ms], obteniéndose como resultado que el poder es de 0.494 (y $\beta = 1 - poder = 0,506$), ligeramente diferente al obtenido en forma teórica debido a errores de redondeo.

¿Cuántas instancias debería usar Lola para lograr un poder de 0,9, manteniendo $\alpha = 0,05$, $sd = 12$ [ms] y $\delta = 4$ [ms]? Las líneas 17–28 del script 6.4 muestran cómo hacer este cálculo, obteniéndose como resultado $n = 97$. Como el tamaño de la muestra siempre debe ser un entero positivo, la línea 27 aproxima el resultado al entero superior.

Otra alternativa es usar la función `pwr.t.test(n, d, sig.level, power, type, alternative)` (ver script 6.4, líneas 37–63), incluida en el paquete `pwr`, donde:

- `n`: tamaño de la muestra (por cada grupo, si corresponde).
- `d`: tamaño del efecto (d de Cohen).
- `sig.level`: nivel de significación.
- `power`: poder de la prueba.
- `type`: tipo de prueba t de Student (“`two.sample`” para diferencia de medias, “`one.sample`” para una sola muestra o “`paired`” para dos muestras pareadas).
- `alternative`: tipo de hipótesis alternativa (“`greater`” o “`less`” si es unilateral, “`two.sided`” si es bilateral).

Debemos fijarnos en que, si bien esta función opera de manera similar a `power.t.test()`, en este caso la desviación estándar y la diferencia son reemplazadas por el tamaño del efecto que podemos cuantificar, como ya vimos, mediante la d de Cohen. Sin embargo, debemos tener cuidado, pues la función `pwr.t.test()` solo es adecuada para una muestra, dos muestras pareadas o cuando ambas muestras tienen igual tamaño. En el caso de la prueba t para dos muestras independientes con diferentes tamaños, debemos usar, en cambio, la función `pwr.t2n.test(n1, n2, d, sig.level, power, alternative)`.

Script 6.4: cálculo del poder en R.

```
1 library(pwr)
2 
3 # Fijar valores conocidos.
4 n <- 36
5 diferencia <- 4
6 desv_est <- 12
7 alfa <- 0.05
8 poder <- 0.9
9 
10 # Calcular el poder usando la función power.t.test().
11 cat("Cálculo del poder con power.t.test()\n")
12 
13 resultado <- power.t.test(n = n,
14                           delta = diferencia,
15                           sd = desv_est,
16                           sig.level = alfa,
17                           power = NULL,
18                           type = "paired",
19                           alternative = "two.sided")
20 
21 print(resultado)
22 
23 # Cálculo del tamaño de la muestra usando la función power.t.test().
24 cat("Cálculo del tamaño de la muestra con power.t.test()\n")
25 
26 resultado <- power.t.test(n = NULL,
27                           delta = diferencia,
28                           sd = desv_est,
29                           sig.level = alfa,
```

```
                          power = poder,
                          type = "paired",
                          alternative = "two.sided")

n <- ceiling(resultado[["n"]])
cat("n = ", n, "\n")

# Calcular el tamaño del efecto (d de Cohen).
d <- (4 / desv_est) * ((n - 2) / (n - 1.25))

# Calcular el poder usando la función pwr.t.test().
cat("\n\nCálculo del poder con pwr.t.test()\n")

resultado <- pwr.t.test(n = n,
                        d = d,
                        sig.level = alfa,
                        power = NULL,
                        type = "paired",
                        alternative = "two.sided")

print(resultado)

# Cálculo del tamaño de la muestra usando la función pwr.t.test().
cat("\nCálculo del tamaño de la muestra con pwr.t.test()\n")

resultado <- pwr.t.test(n = NULL,
                        d = d,
                        sig.level = alfa,
                        power = poder,
                        type = "paired",
                        alternative = "two.sided")

n <- ceiling(resultado[["n"]])
cat("n = ", n, "\n")
```

6.6 EJERCICIOS PROPUESTOS

1. Define con tus propias palabras lo que es el tamaño del efecto.
2. Un estudio sobre en el tiempo que necesitan los estudiantes para resolver una guía de ejercicios de Cálculo I, comparó un grupo de estudiantes que cursaban la asignatura por primera vez con un grupo que la cursaba en segunda ocasión. Sabiendo que este tiempo se distribuye normalmente en ambos casos, con varianza similar, dibuja cómo se verían los datos si el efecto de repetir la asignatura sobre el tiempo requerido para resolver la guía fuera "grande" y si este efecto fuera "pequeño, pero positivo".
3. Investiga cómo se calcula y cómo se interpreta la medida g de Hedges para el tamaño del efecto, e indica en qué casos es adecuada.
4. ¿Por qué se necesita conocer el tamaño del efecto?
5. ¿Cómo se relaciona el tamaño del efecto con la significación estadística?
6. ¿Por qué sería útil determinar un tamaño de muestra apropiado?
7. Explica en tus palabras lo que se muestra en la figura 6.4.
8. Ante algunas acusaciones de colusión, el Tribunal de la Libre Competencia quiere estudiar dos compañías del mercado de los seguros de automóviles. En base a datos del gremio de las aseguradoras, se puede asumir que el precio de las primas estándares para diferentes marcas de vehículos sigue una distribución

aproximadamente normal con desviación estándar de \$16.000. Fija los otros parámetros del estudio y determina qué tamaño debería tener la muestra de automóviles para detectar una diferencia de \$10.000 en el precio medio de las compañías bajo sospecha.

CAPÍTULO 7. INFERENCIA CON PROPORCIONES MUESTRALES

En el capítulo5 conocimos las pruebas Z y t de Student para contrastar hipótesis con una y dos medias. Ahora estudiaremos los métodos de Wald y de Wilson para inferir acerca de una y dos proporciones, basándonos para ello en los textos de Diez y col. (2017, pp. 274-286), NIST/SEMATECH (2013, pp. 7.2.4, 7.2.4.1), Pértega y Pita (2004), Champely, Ekstrom, Dalgaard, Gill, Weibelzahl, Anandkumar, Ford, Volcic y de Rosario (2020) y Kabacoff (2017).

7.1 MÉTODO DE WALD

En el capítulo 3 vimos que, cuando queremos responder preguntas del tipo "¿qué proporción de la ciudadanía apoya al gobierno actual?", estamos hablando de una variable aleatoria que sigue una distribución binomial. En general, no conocemos la **probabilidad de éxito $\boldsymbol{p}$** de la población, por lo que tenemos que usar el estimador puntual (correspondiente a la proporción de éxito de la muestra), denotado por $\hat{p}$. Este estimador se distribuye de manera cercana a la normal cuando se cumplen las siguientes condiciones:

1. Las observaciones de la muestra son independientes.
2. Se cumple la **condición de éxito-fracaso**, que establece que se espera observar al menos 10 observaciones correspondientes a éxito y al menos 10, correspondientes a fracasos. Matemáticamente, $np \geq 10$ y $n(1-p) \geq 10$.

Así, si la distribución muestral de $\hat{p}$ cumple con las condiciones anteriores, se dice que es cercana a la normalidad con media $\mu = p$, desviación estándar $\sigma = \sqrt{p(1-p)}$ y error estándar dado por la ecuación 7.1.

$$SE_{\hat{p}} = \sqrt{\frac{p(1-p)}{n}} \tag{7.1}$$

7.1.1 Método de Wald para una proporción

El **método de Wald** permite construir intervalos de confianza y contrastar hipótesis bajo el supuesto de normalidad para una proporción. Consideremos el siguiente ejemplo: Aquiles Baeza, ingeniero en informática, desea conocer qué proporción de las ejecuciones de un algoritmo de ordenamiento para instancias con 100.000 elementos (bajo iguales condiciones de hardware y sistema) tardan menos de 25 segundos. Para ello, registró los tiempos de ejecución para 150 instancias generadas de manera aleatoria, encontrando que 64 % de dichas instancias fueron resueltas en un tiempo menor al señalado.

Si bien no conocemos la probabilidad real de éxito para la población, sabemos que $\hat{p} = 0,64$. Así, si se cumplen las condiciones para que la distribución de $\hat{p}$ sea cercana a la normal, podemos construir un intervalo de confianza para la verdadera proporción muestral.

En el enunciado del ejemplo nos indican que las instancias del problema fueron escogidas de manera aleatoria y sabemos que éstas representan menos del 10 % del total de instancias posibles, con lo que se verifica la independencia de las observaciones. Por otra parte, nos dicen que la proporción de éxito es $\hat{p} = 0,64$, por lo que esperamos encontrar $0,64 \cdot 150 = 96$ instancias que tardan menos de 25 segundos y $(1 - 0,64) \cdot 150 = 54$ fracasos (instancias que tardan 25 segundos o más), con lo que se cumple la condición de éxito-fracaso. En consecuencia, podemos asumir que la distribución muestral de $\hat{p}$ sigue aproximadamente a la normal.

Podemos estimar el error estándar usando la ecuación 7.1, reemplazando p por el estadístico $\hat{p}$:

$$SE = \sqrt{\frac{\hat{p}(1-\hat{p})}{n}} = \sqrt{\frac{0,64(1-0,64)}{150}} = 0,0392$$

Con ello, construimos el intervalo de confianza para un nivel de significación $\alpha = 0,05$ usando la ecuación general (4.6) con $\hat{p}$ como estimador puntual:

$$\hat{p} \pm z^* \cdot SE \rightarrow 0,64 \pm 1,96 \cdot 0,0392 \rightarrow [0,5632; 0.7168]$$

Este intervalo significa que tenemos 95 % de confianza que la proporción de instancias (de 100.000 elementos) del problema que el algoritmo ordena en menos de 25 segundos se encuentra entre 56,32 % y 71,6 %.

Desde luego, también podemos usar el modelo normal en el contexto de la prueba de hipótesis para una proporción. Para ello, se deben cumplir las condiciones de independencia y éxito-fracaso que ya verificamos para construir el intervalo de confianza, pero en este caso tenemos que verificar la segunda condición con el valor nulo, denotado p_0. Una vez verificadas ambas condiciones, el error estándar y el estadístico Z que permiten determinar el p-valor se calculan usando las ecuaciones 7.2 y 7.3, respectivamente.

$$SE = \sqrt{\frac{p_0(1-p_0)}{n}} \tag{7.2}$$

$$Z = \frac{\hat{p} - p_o}{SE} \tag{7.3}$$

Supongamos ahora, volviendo a nuestro ejemplo, que Baeza afirma que más del 70 % de las instancias de tamaño 100.000 se ejecutan en menos de 25 segundos. Sin embargo, su jefe no está seguro, por lo que decide comprobarlo mediante una prueba de hipótesis con un nivel de significación $\alpha = 0,05$ (recordemos que $n = 150$ y $\hat{p} = 0,64$):

H_0: el 70 % de las instancias se ejecutan en menos de 25 segundos.
H_A: más del 70 % de las instancias se ejecutan en menos de 25 segundos.

De acuerdo a las hipótesis formuladas por el jefe de Baeza, el valor nulo es $p_0 = 0,7$, con lo que estas pueden formularse matemáticamente como:

Denotando como p a la proporción de todas las instancias de tamaño 100.000 que se ejecutan en menos de 25 segundos y considerando el valor hipotético $p_0 = 0,7$ para este parámetro:
H_0: p = p_0
H_A: p >p_0

Ya antes habíamos comprobado que se verifica la independencia de las observaciones. Además, considerando que el valor nulo fuese verdadero esperaríamos encontrar $0,7 \cdot 150 = 105$ éxitos y $(1-0,7) \cdot 150 = 45$ fracasos, ambos valores mayores que 10, por lo que la condición de éxito-fracaso se verifica.

Con ello, podemos calcular el estadístico de prueba:

$$SE = \sqrt{\frac{p_0(1-p_0)}{n}} = \sqrt{\frac{0,7(1-0,7)}{150}} = 0,0374$$

$$Z = \frac{\hat{p} - p_o}{SE} = \frac{0,64 - 0,7}{0,0374} = -1,6043$$

El valor p asociado, calculado en R mediante la llamada a la función `pnorm(-1.6042, lower.tail = FALSE)`, es $p = 0,9456$. En consecuencia, la evidencia no es suficiente para rechazar la hipótesis nula, por lo que se concluye, con 95 % de confianza, que no es cierto que el algoritmo se ejecute en menos de 25 segundos para más del 70 % de las instancias de tamaño 100.000.

R no ofrece esta prueba, como función. Sin embargo, podemos hacerla como muestra el script 7.1 para nuestro ejemplo.

Script 7.1: método de Wald para una proporción.

```
# Fijar valores conocidos
n <- 150
p_exito <- 0.64
alfa <- 0.05
valor_nulo <- 0.7

# Construcción del intervalo de confianza.
error_est <- sqrt((p_exito * (1 - p_exito)) / n)
Z_critico <- qnorm(alfa / 2, lower.tail = FALSE)
inferior <- p_exito - Z_critico * error_est
superior <- p_exito + Z_critico * error_est
cat("Intervalo de confianza = [", inferior, ", ", superior, "]\n", sep = "")

# Prueba de hipótesis.
error_est_hip <- sqrt((valor_nulo * (1 - valor_nulo)) / n)
Z <- (p_exito - valor_nulo) / error_est_hip
p <- pnorm(Z, lower.tail = FALSE)
cat("Hipótesis alternativa unilateral\n")
cat("Z =", Z, "\n")
cat("p =", p)
```

7.1.2 Método de Wald para dos proporciones

También podemos usar el método de Wald para estudiar la **diferencia entre las proporciones** de dos poblaciones, considerando para ello como estimador puntual la diferencia $\hat{p}_1 - \hat{p}_2$.

De manera similar a lo que ya vimos para una única proporción, también en este caso debemos verificar ciertas condiciones antes de poder aplicar el modelo normal:

1. Cada proporción, por separado, sigue el modelo normal.
2. Las dos muestras son independientes una de la otra.

El error estándar para la diferencia entre dos proporciones muestrales está dado por la ecuación 7.4, donde p_1 y p_2 corresponden a las proporciones de las poblaciones, y n_1 y n_2, a los tamaños de las muestras. La construcción del intervalo de confianza se realiza, una vez más, con la ecuación general 4.6.

$$SE_{\hat{p}_1 - \hat{p}_2} = \sqrt{SE^2_{\hat{p}_1} + SE^2_{\hat{p}_2}} = \sqrt{\frac{p_1(1-p_1)}{n_1} + \frac{p_2(1-p_2)}{n_2}} \tag{7.4}$$

A modo de ejemplo, supongamos que la Facultad de Ingeniería de una prestigiosa universidad desea determinar si la tasa de reprobación de estudiantes que rinden la asignatura de programación por primera vez es igual para hombres y mujeres. Para ello, se examina la situación final de los estudiantes que rindieron la asignatura durante el segundo semestre de 2017. Para una muestra de 48 hombres (de un total de 632), se encontró que 26 de ellos reprobaron la asignatura. De manera similar, para una muestra de 42 mujeres (de un total de 507), se encontraron 20 reprobaciones[1], con ambas muestras tomadas de manera aleatoria.

Como ya es habitual, comencemos por verificar las condiciones de normalidad para cada una de las muestras. En ambos casos, las observaciones son independientes entre sí, pues provienen de personas diferentes que representan a menos del 10 % de la población. Además, los datos entregados evidencian que en ambos casos se cumple la condición de éxito-fracaso. Adicionalmente, ambas muestras son independientes entre sí, pues ambas categorías se excluyen mutuamente. Con esto último se verifican entonces las condiciones de normalidad para la diferencia de proporciones.

Sean $\hat{p}_1$ y $\hat{p}_2$ las proporciones de éxito muestrales (considerando en este contexto la reprobación como éxito) para hombres y mujeres, respectivamente:

$$\hat{p}_1 = 26/48 = 0,5417$$

$$\hat{p}_2 = 20/42 = 0,4762$$

$$\hat{p}_1 - \hat{p}_2 = 0,5417 - 0,4762 = 0,0655$$

El error estándar puede estimarse como:

$$SE_{\hat{p}_1-\hat{p}_2} = \sqrt{\frac{p_1(1-p_1)}{n_1} + \frac{p_2(1-p_2)}{n_2}} = \sqrt{\frac{0,5417(1-0,5417)}{48} + \frac{0,4762(1-0,4762)}{42}} = 0,1054$$

Suponiendo un nivel de significación $\alpha = 0,05$, el intervalo de confianza corresponde a:

$$\hat{p}_1 - \hat{p}_2 \pm z^* SE_{\hat{p}_1-\hat{p}_2} \rightarrow 0,0655 \pm 1,96 \cdot 0,1054 \rightarrow [-0,1411; 0,2721]$$

En consecuencia, podemos afirmar con 95 % de confianza que la diferencia en la tasa de reprobación de la asignatura de programación para hombres y mujeres varía entre -14,11 % y 27,21 %.

Desde luego, también podemos realizar pruebas de hipótesis en este escenario. Para el ejemplo tenemos que:

H_0: no hay diferencia en la tasa de reprobación de hombres y mujeres.
H_A: las tasas de reprobación son diferentes para hombres y mujeres.

Matemáticamente:

Denotando como p_1 y p_2 a las proporciones de hombres y mujeres, respectivamente, que reprueban la asignatura de programación la primera vez que la cursan:
H_0: $p_1 - p_2 = 0$
H_A: $p_1 - p_2 \neq 0$

Ya verificamos las condiciones para operar bajo el supuesto de normalidad cuando construimos el intervalo de confianza. Sin embargo, **cuando la hipótesis nula supone que no hay diferencia entre las proporciones**, la verificación de la condición de éxito-fracaso y la estimación del error estándar se realizan usando para ello la **proporción agrupada**, dada por la ecuación 7.5, donde $\hat{p}_1 n_1$ y $\hat{p}_2 n_2$ representan la cantidad de éxitos en la primera y segunda muestra, respectivamente.

[1] Los datos aquí presentados son ficticios, creados únicamente con fines pedagógicos.

$$\hat{p} = \frac{\text{número de éxitos}}{\text{número de casos}} = \frac{\hat{p}_1 n_1 + \hat{p}_2 n_2}{n_1 + n_2} \tag{7.5}$$

Así, en este caso tenemos:

$$\hat{p} = \frac{\hat{p}_1 n_1 + \hat{p}_2 n_2}{n_1 + n_2} = \frac{0,5417 \cdot 48 + 0,4762 \cdot 42}{48 + 42} = 0,5111$$

En consecuencia, en el caso de los hombres esperamos encontrar $\hat{p}n_1 > 24$ éxitos (reprobaciones) y $(1-\hat{p})n_1 > 23$ fracasos. Del mismo modo, para las mujeres esperamos $\hat{p}n_2 > 21$ éxitos y $(1-\hat{p})n_2 > 20$ fracasos, con lo que se verifican las condiciones para emplear el modelo normal.

El error estándar se calcula, como ya mencionamos, usando la proporción agrupada:

$$SE = \sqrt{\frac{\hat{p}(1-\hat{p})}{n_1} + \frac{\hat{p}(1-\hat{p})}{n_2}} = \sqrt{\frac{0,5111 \cdot (1-0,5111)}{48} + \frac{0,5111 \cdot (1-0,5111)}{42}} = 0,1056$$

El estimador puntual para la diferencia es $\hat{p}_1 - \hat{p}_2 = 0,0655$, con lo cual el estadístico de prueba está dado por:

$$Z = \frac{\text{estimador puntual} - \text{valor nulo}}{SE} = \frac{0,0655 - 0}{0,1056} = 0,6203$$

En consecuencia, el valor p correspondiente es $p = 0,5351$. Puesto que el valor p es mayor que $\alpha = 0,05$, se falla en rechazar la hipótesis nula. Así, podemos decir con 95 % de confianza que no existe evidencia suficiente para concluir que hay diferencia en la tasa de reprobación de hombres y mujeres para el primer curso de programación.

El script 7.2 muestra el desarrollo de este ejemplo en R.

Script 7.2: método de Wald para la diferencia entre dos proporciones (ejemplo 1).

```
# Fijar valores conocidos
n_hombres <- 48
n_mujeres <- 42
exitos_hombres <- 26
exitos_mujeres <- 20
alfa <- 0.05
valor_nulo <- 0

# Calcular probabilidades de éxito.
p_hombres <- exitos_hombres / n_hombres
p_mujeres <- exitos_mujeres / n_mujeres

# Estimar la diferencia.
diferencia <- p_hombres - p_mujeres

# Construcción del intervalo de confianza.
error_hombres <- (p_hombres * (1 - p_hombres)) / n_hombres
error_mujeres <- (p_mujeres * (1 - p_mujeres)) / n_mujeres
error_est <- sqrt(error_hombres + error_mujeres)
Z_critico <- qnorm(alfa / 2, lower.tail = FALSE)
inferior <- diferencia - Z_critico * error_est
superior <- diferencia + Z_critico * error_est
cat("Intervalo de confianza = [", inferior, ", ", superior, "]\n", sep = "")

# Prueba de hipótesis.
```

```
p_agrupada <- (exitos_hombres + exitos_mujeres) / (n_hombres + n_mujeres)
error_hombres <- (p_agrupada * (1 - p_agrupada)) / n_hombres
error_mujeres <- (p_agrupada * (1 - p_agrupada)) / n_mujeres
error_est_hip <- sqrt(error_hombres + error_mujeres)
Z <- (diferencia - valor_nulo) / error_est_hip
p <- 2 * pnorm(Z, lower.tail = FALSE)
cat("Hipótesis alternativa bilateral\n")
cat("Z =", Z, "\n")
cat("p =", p)
```

Cuando contrastamos hipótesis para la **diferencia entre dos proporciones con un valor nulo distinto de 0**, el procedimiento es ligeramente diferente. En este caso, la comprobación de la condición de éxito-fracaso se realiza de manera independiente para ambas muestras y el error estándar se calcula, como ya se estudió para los intervalos de confianza, mediante la ecuación 7.4.

Supongamos ahora que la Facultad de Ingeniería de la Universidad anterior ha decidido replicar el estudio realizado para el curso de programación, esta vez para una asignatura de física. No obstante, las autoridades están convencidas de que la tasa de reprobación es 10 % mayor para los hombres y que, incluso, la diferencia podría ser mayor. Desean comprobar con un nivel de confianza de 95 % y para ello, seleccionaron aleatoriamente a 89 de los 1.023 hombres y a 61 de las 620 mujeres de la cohorte correspondiente al primer semestre de 2019. En las muestras se encuentran, respectivamente, 45 y 21 reprobaciones.

Las hipótesis son, en este caso:

H_0: la tasa de reprobación de los hombres es exactamente 10 % más alta que la de las mujeres.
H_A: la tasa de reprobación de los hombres es más de 10 % más alta que la de las mujeres.

Matemáticamente:

Denotando como p_1 y p_2 a las proporciones de hombres y mujeres, respectivamente, que reprueban la asignatura de física estudiada la primera vez que la cursan:
H_0: $p_1 - p_2 = 0,1$
H_A: $p_1 - p_2 > 0,1$

Al igual que en los ejemplos previos, las observaciones de cada muestra son independientes entre sí pues corresponden a menos del 10 % de la población y fueron escogidos aleatoriamente. A su vez, los datos proporcionados indican que se cumple la condición de éxito-fracaso para cada muestra. Como ambas muestras pertenecen a grupos diferentes de estudiantes, son independientes entre sí. En consecuencia, se cumplen las condiciones para operar bajo el modelo normal.

En el caso de los hombres, la tasa de éxito se estima como:

$$\hat{p}_1 = \frac{45}{89} = 0,5056$$

Análogamente, para las mujeres tenemos:

$$\hat{p}_2 = \frac{21}{61} = 0,3443$$

Con lo que el estimador puntual para la diferencia es:

$$\hat{p}_1 - \hat{p}_2 = 0,5056 - 0,3443 = 0,1613$$

Ahora calculamos el error estándar:

$$SE = \sqrt{\frac{0,5056(1-0,0,5056)}{89} + \frac{0,3443(1-0,3443)}{61}} = 0,0807$$

Con lo cual podemos calcular el estadístico de prueba:

$$Z = \frac{\text{estimador puntual} - \text{valor nulo}}{SE} = \frac{0,1613 - 0,1}{0,0807} = 0,7596$$

Con lo que se puede obtener el valor p, correspondiente a $p = 0.2237 > \alpha = 0,05$.

En consecuencia, se falla en rechaza H_0 en favor de H_A, por lo que concluimos, con 95 % de confianza, que la tasa de reprobación de los hombres es 10 % superior a la de las mujeres para el curso de física.

En R, esta prueba puede realizarse como muestra el script 7.3.

Script 7.3: método de Wald para la diferencia entre dos proporciones (ejemplo 2).

```
# Fijar valores conocidos
n_hombres <- 89
n_mujeres <- 61
exitos_hombres <- 45
exitos_mujeres <- 21
alfa <- 0.05
valor_nulo <- 0.1

# Calcular probabilidades de éxito.
p_hombres <- exitos_hombres / n_hombres
p_mujeres <- exitos_mujeres / n_mujeres

# Estimar la diferencia.
diferencia <- p_hombres - p_mujeres

# Prueba de hipótesis.
p_agrupada <- (exitos_hombres + exitos_mujeres) / (n_hombres + n_mujeres)
error_hombres <- (p_hombres * (1 - p_hombres)) / n_hombres
error_mujeres <- (p_mujeres * (1 - p_mujeres)) / n_mujeres
error_est <- sqrt(error_hombres + error_mujeres)
Z <- (diferencia - valor_nulo) / error_est
p <- pnorm(Z, lower.tail = FALSE)
cat("Hipótesis alternativa bilateral\n")
cat("Z =", Z, "\n")
cat("p =", p)
```

7.2 MÉTODO DE WILSON

El método de Wald, tratado en la sección anterior, es el método que tradicionalmente se ha usado y el que aparece en la mayoría de los libros clásicos de inferencia estadística. Sin embargo, el método está siendo muy criticado hoy en día debido a que hace importantes simplificaciones matemáticas en su procedimiento y ya hay evidencia empírica que ha demostrado sus limitaciones (Agresti & Coull, 1998).

Gracias al aumento del poder de cómputo y la disponibilidad de software estadístico, han surgido diversas alternativas, entre las cuales destaca el **método de Wilson** (junto con algunas variaciones), considerado el más robusto por diversos autores (Agresti & Coull, 1998; Brown y col., 2001; Devore, 2008; Wallis, 2013). Este método opera del mismo modo que el de Wald, aunque las fórmulas empleadas para estimar la proporción en la muestra y el error estándar son diferentes.

En R, podemos hacer esta prueba usando la función `prop.test(x, n, p, alternative, conf.level, ...)`, cuyos principales parámetros son:

- `x`: cantidad de éxitos en la muestra.
- `n`: tamaño de la muestra.
- `p`: valor nulo (por defecto, `p=NULL`).
- `alternative`: tipo de hipótesis alternativa, por defecto bilateral (`alternative=‘‘two.sided’’`), y valores `‘‘less’’` y `‘‘greater’’` para hipótesis unilaterales.
- `conf.level`: nivel de confianza (`conf.level=0.95` por defecto).

El script 7.4 muestra el uso de esta función con el mismo ejemplo que usamos para presentar la prueba de Wald para una proporción. Del mismo modo, el script 7.5 usa la función `prop.test()` para el primer ejemplo del método de Wald para la diferencia entre dos proporciones. Sin embargo, esta función tiene la limitante de que, al trabajar con dos proporciones, no permite establecer un valor nulo distinto de cero para la diferencia.

Script 7.4: método de Wilson para una proporción.

```r
# Fijar valores conocidos
n <- 150
p_exito <- 0.64
alfa <- 0.05
valor_nulo <- 0.7

# Calcular cantidad de éxitos.
exitos <- p_exito * n

# Prueba de Wilson en R.
prueba <- prop.test(exitos, n = n, p = valor_nulo,
                    alternative = "greater", conf.level = 1 - alfa)

print(prueba)
```

Script 7.5: método de Wilson para la diferencia entre dos proporciones.

```r
# Fijar valores conocidos (hombres, mujeres)
n <-c(48, 42)
exitos <- c(26, 20)
alfa <- 0.05

# Prueba de Wilson en R.
prueba <- prop.test(exitos, n = n, alternative = "two.sided",
                    conf.level = 1 - alfa)

print(prueba)
```

7.3 PODER Y PRUEBAS DE PROPORCIONES

En el capítulo anterior conocimos el poder estadístico y vimos que está relacionado con el nivel de significación, el tamaño de la muestra y el tamaño del efecto que queremos detectar.

R base nos ofrece la función `power.prop.test(n, p1, p2, sig.level, power, alternative)`, donde:

- `n`: número de observaciones por cada grupo.
- `p1`: probabilidad de éxito en un grupo.
- `p2`: probabilidad de éxito en otro grupo.

- `sig.level`: nivel de significación.
- `power`: poder de la prueba.
- `alternative`: tipo de hipótesis alternativa (`“one.sided”` si es unilateral, `“two.sided”` si es bilateral).

Al igual que vimos en el capítulo anterior para la función `power.t.test()`, recibe cuatro de los primeros argumentos y al restante debe asigrnársele el valor `NULL`. Como resultado, retorna un objeto que incluye el valor calculado para el argumento faltante.

Una vez más, el paquete `pwr` de R nos ofrece varias funciones que podemos usar como alternativa:

- `pwr.p.test(h, n, sig.level, power, alternative)`: para pruebas con una única proporción.
- `pwr.2p.test(h, n, sig.level, power, alternative)`: para pruebas con dos proporciones donde ambas muestras son de igual tamaño.
- `pwr.2p2n.test(h, n1, n2, sig.level, power, alternative)`: para pruebas con dos proporciones y muestras de diferente tamaño.

Donde:

- `h`: tamaño de efecto.
- `n`, `n1`, `n2`: tamaño(s) de la(s) muestra(s).
- `sig.level`: nivel de significación.
- `power`: poder.
- `alternative`: tipo de hipótesis alternativa (`“two.sided”`, `“less”` o `“greater”`).

El funcionamiento de esta familia de funciones es igual al que ya conocimos en el capítulo anterior para la función `pwr.t.test()`. Se entrega el parámetro `alternative` y todos los demás excepto uno (al cual debe asignarse explícitamente el valor `NULL`). Como resultado, la función calcula dicho valor.

El tamaño del efecto puede calcularse como muestra la ecuación 7.6, implementada en R en la función `ES.h(p1, p2)` del paquete `pwr`.

$$h = 2\arcsin(\sqrt{p_1}) - 2\arcsin(\sqrt{p_2}) \tag{7.6}$$

En el caso de una única proporción, los autores del paquete `pwr` sugieren usar $p_2 = 0,5$ (Champely y col., 2020).

Otra función que nos puede ser de ayuda es `bsamsize(p1, p2, fraction, alpha, power)`, del paquete `Hmisc`. En el caso de una prueba de Wilson con dos muestras, calcula los tamaños de cada grupo dados los siguientes argumentos:

- `p1`: probabilidad de la población para el grupo 1.
- `p2`: probabilidad del grupo 2.
- `fraction`: fracción de las observaciones en el grupo 1 ($n1/(n1 + n2)$).
- `alpha`: nivel de significación.
- `power`: poder deseado.

7.4 EJERCICIOS PROPUESTOS

1. ¿En qué condiciones la distribución muestral de una proporción tiene comportamiento aproximadamente normal?
2. ¿Cómo se calcula la desviación estándar de la distribución muestral de las proporciones bajo estas condiciones (según el método de Wald)?
3. ¿Cómo se calcula un intervalo de confianza para la verdadera proporción (según el método de Wald)?

4. El patrón de un gran fundo de nogales está preocupado porque se ha detectado la presencia de una plaga en varios árboles. Si bien existe un pesticida para el parásito, este es bastante caro y su aplicación solo se justifica económicamente si más del 20 % de los árboles está infectado. En consecuencia, el patrón ha decidido estimar la extensión de la infestación revisando una muestra aleatoria de 200 nogales (una porción bastante pequeña de los más de 20.000 árboles en el fundo). En base a lo anterior, determina:
   a) ¿Cuál es la variable dicotómica (experimento Bernulli) en este caso?
   b) ¿Cuál es el parámetro de interés?
   c) ¿Qué estimador existe para este parámetro?
   d) ¿Qué hipótesis respondería las dudas del patrón del fundo?
5. En el experimento del ejercicio anterior se encontró que 45 árboles de la muestra estaban infectados:
   a) ¿Se puede asumir que esta proporción muestral sigue el modelo normal?
   b) Independientemente de la respuesta anterior, obtén un intervalo con 95 % confianza para la verdadera proporción de árboles infectados en el fundo.
   c) ¿Qué recomendarías al patrón del fundo?
6. Como el patrón sigue con dudas, ahora pregunta: ¿cuántos árboles debería revisar en una muestra para estar 99 % confiado que más del 20 % de los árboles están infectados, con solo 10 % de probabilidades de equivocarse si la verdadera proporción fuera 18 %? ¿Cómo se puede calcular esto? ¿Cuál debiera ser la respuesta a la pregunta del patrón?
7. ¿En qué condiciones la distribución muestral de la diferencia de dos proporciones tiene comportamiento aproximadamente normal?
8. ¿Cómo se calcula el error estándar de la diferencia entre dos proporciones (según el método de Wald)?
9. ¿Cómo se calcula un intervalo de confianza para la verdadera diferencia entre dos proporciones (según el método de Wald)?
10. Un laboratorio homeopático acaba de lanzar un tónico que asegura que ayuda a prevenir el resfrío durante el periodo invernal, con igual eficacia tanto en mujeres como en hombres. Para comprobar esta promesa, el laboratorio está realizando un estudio de la eficacia del producto en una muestra aleatoria de 100 mujeres y 200 hombres:
    a) ¿Cuál es el parámetro de interés y qué estimador se podría usar?
    b) ¿Qué hipótesis se deberían docimar para comprobar o refutar la homogeneidad de la eficacia del tónico para el resfrío?
11. El estudio anterior encontró que, durante las semanas de prueba, 38 mujeres y 102 hombres presentaron síntomas de resfrío. ¿Es homogénea la eficacia del producto con un nivel de significación de 0,05?
12. ¿Qué poder tuvo la prueba anterior?
13. ¿Qué tamaño deberían tener las muestras aleatorias de mujeres y hombres (manteniendo la proporción del ejemplo) para conseguir un poder de 0,85 con 99 % de confianza?
14. Las fórmulas presentadas en la sección 7.1, se conocen colectivamente como “Método de Wald”, el que ya no es recomendado por académicos del área. Usando la bibliografía citada, ¿cuáles son las fórmulas del método de Wilson para estimar el error estándar de la proporción y su extensión a la prueba de hipótesis e intervalos de confianza?
15. Investiga para qué sirve y cómo funciona el parámetro `correct` (verdadero por defecto) de la función `prop.test()` de R.

CAPÍTULO 8. INFERENCIA NO PARAMÉTRICA CON PROPORCIONES

Si eres una persona observadora, habrás notado que el título de este capítulo lleva la frase **no paramétrica** para referirse a inferencias con proporciones, pero ¿qué significa esto?

En el capítulo 5 conocimos las pruebas Z y t de Student. Ambas formulan hipótesis relativas al parámetro μ de una distribución normal (o la diferencia $\mu_1 - \mu_2$ de dos distribuciones normales). Así estas pruebas (y otras que se verán más adelante) hacen una fuerte suposición acerca de la distribución que subyace a las poblaciones estudiadas, lo que permite inferir sobre los parámetros de esas distribuciones. Lo mismo ocurre con las pruebas de Wald y Wilson estudiadas en el capítulo 7, las cuales contrastan hipótesis en torno a un cierto valor para el parámetro p de una población que sigue una distribución binomial (o la diferencia de los parámetros $p_1 - p_2$ de dos de estas poblaciones).

En este capítulo conoceremos algunas pruebas para inferir acerca de proporciones cuyas hipótesis nula y alternativa **no mencionan parámetro alguno**. Es más, **ninguna de ellas hace alguna suposición sobre la distribución de la población** desde donde proviene la muestra analizada. Es por esta razón que a estas pruebas (y a otras que se abordan en capítulos posteriores) se les denomina **no paramétricas** o **libres de distribución**.

Las pruebas no paramétricas nos ofrecen una ventaja evidente: **son menos restrictivas** que las pruebas paramétricas, porque imponen menos supuestos a las poblaciones para poder trabajar con ellas. Asegurar que una población sigue una distribución normal o binomial, por ejemplo, puede ser una tarea difícil y, en la práctica, no es infrecuente encontrarse con conjuntos de datos que no parecen seguir alguna de estas distribuciones. Pero... si las pruebas no paramétricas parecen tan ventajosas, ¿por qué no usarlas siempre? Por dos grandes razones:

- Las pruebas no paramétricas **nos entregan menos información**. Como veremos en este capítulo para el caso de las proporciones, estas pruebas se limitan a trabajar con hipótesis del tipo "las poblaciones muestran las mismas proporciones" versus "las poblaciones muestran proporciones distintas", pero **ninguna indica cuáles serían esas proporciones** en realidad, ni siquiera si es mayor en una o en la otra.
- Cuando sí se cumplen las condiciones para aplicar una prueba paramétrica, las versiones no paramétricas presentan **menor poder estadístico** y, en consecuencia, suelen necesitar muestras de mayor tamaño para detectar diferencias significativas que pudieran existir entre las poblaciones comparadas.

Como ya hemos dicho, en este capítulo conoceremos algunas pruebas no paramétricas para estudiar la relación entre dos variables categóricas, con base en Diez y col. (2017, pp. 286-302), Pértega y Pita (2004), Glen (2016a) y Mangiafico (2016).

8.1 PRUEBA CHI-CUADRADO DE PEARSON

Conocida también como **Prueba χ^2 de Asociación**, la **prueba chi-cuadrado de Pearson** sirve para inferir con proporciones cuando disponemos de dos variables categóricas y una de ellas es dicotómica (es decir, tiene solo dos niveles). En este caso, podemos registrar las frecuencias observadas para las posibles combinaciones de ambas variables mediante una tabla de contingencia o matriz de confusión, como ya estudiamos en el capítulo 2. En adelante, nos referiremos a cada una de estas combinaciones como un grupo.

Debemos verificar algunas condiciones antes de poder usar la prueba chi-cuadrado:

1. Las observaciones deben ser independientes entre sí.
2. Debe haber a lo menos 5 observaciones esperadas en cada grupo.

La primera de estas condiciones ya la hemos encontrado antes, mientras que explicaremos la segunda a medida que avancemos en el estudio de la prueba chi-cuadrado.

Si bien en esta sección estamos hablando de una una única prueba, que sigue siempre el mismo procedimiento, es común encontrarla como tres pruebas diferentes:

- Prueba χ^2 de homogeneidad.
- Prueba χ^2 de bondad de ajuste
- Prueba χ^2 de independencia.

La diferencia entre ellas es **conceptual** (no matemática) y tiene relación con cómo se miren las variables y las poblaciones involucradas en el problema.

8.1.1 Prueba chi-cuadrado de homogeneidad

Esta prueba resulta adecuada si queremos determinar si **dos poblaciones** (la variable dicotómica) presentan **las mismas proporciones en los diferentes niveles de una variable categórica**.

Por ejemplo, supongamos que la Sociedad Científica de Computación (SCC) ha realizado una encuesta a 300 programadores con más de 3 años de experiencia de todo el país, escogidos al azar, y les ha preguntado cuál es su lenguaje de programación favorito. La tabla 8.1 muestra las preferencias para cada lenguaje, separadas en programadores (varones) y programadoras (mujeres). ¿Son similares las preferencias de lenguaje de programación entre hombres y mujeres?

Lenguaje	**C**	**Java**	**Python**	**Ruby**	**Otro**	**Total**
Programadores	42	56	51	27	24	200
Programadoras	25	24	27	15	9	100
Total	67	80	78	42	33	300

Tabla 8.1: tabla de frecuencias para el lenguaje de programación favorito de la muestra.

Si fuera cierto que ambas poblaciones tienen las mismas preferencias, esperaríamos encontrar proporciones similares en las muestras, pese a la variabilidad. En consecuencia, necesitamos determinar si las diferencias entre las cantidades observadas y las esperadas son lo suficientemente grandes como para proporcionar evidencia convincente de que las preferencias son disímiles. La tabla 8.2 muestra las frecuencias esperadas para cada lenguaje de programación bajo este supuesto, calculadas mediante la ecuación 8.1, donde:

- n_i: total de observaciones en la fila i.
- n_j: total de observaciones en la columna j.
- n: tamaño de la muestra.

$$E_{i,j} = \frac{n_i \cdot n_j}{n} \tag{8.1}$$

Ahora que ya sabemos cómo determinar la cantidad de observaciones esperadas en cada grupo, podemos verificar que, para cada caso, este valor es mayor que 5. Adicionalmente, es razonable suponer la muestra representa menos del 10 % de los programadores del país y sabemos que fue seleccionada de manera aleatoria, por lo que podemos proceder con la prueba χ^2 de homogeneidad.

Lenguaje	C	Java	Python	Ruby	Otro	Total
Programadores	44,7	53,3	52,0	28,0	22,0	200
Programadoras	22,3	26,7	26,0	14,0	11,0	100
Total	67	80	78	42	33	300

Tabla 8.2: frecuencias esperadas si hombres y mujeres tienen las mismas preferencias.

Las hipótesis a contrastar son:

H_0: programadores hombres y mujeres tienen las mismas preferencias en lenguaje de programación favorito (ambas poblaciones muestras las mismas proporciones para cada lenguaje estudiado).
H_A: programadores hombres y mujeres tienen preferencias distintas en lenguajes de programación favorito.

Recordemos que la primera aproximación para construir un estadístico de prueba adecuado está dada por la ecuación 4.5, que reproducimos aquí:

$$Z = \frac{\text{estimador puntual} - \text{valor nulo}}{SE_{\text{estimador puntual}}}$$

Podríamos usar esta fórmula de la diferencia estandarizada para cada uno de los grupos, donde:

- El estimador puntual corresponde a la frecuencia observada para el grupo.
- El valor nulo corresponde a la frecuencia esperada para el grupo.
- El error estándar del estimador puntual es la raíz cuadrada del valor nulo.

Así, para los programadores (varones) en C se tiene:

$$Z_{H\_C} = \frac{42 - 44,7}{\sqrt{44,7}} = -0,404$$

Al repetir el procedimiento para cada grupo, se obtienen los valores Z presentados en la tabla 8.3.

Lenguaje	C	Java	Python	Ruby	Otro
Programadores	-0,404	0,370	-0,139	-0,189	0,426
Programadoras	0,572	-0,523	0,196	0,267	-0,603

Tabla 8.3: valor Z para cada grupo.

Pero necesitamos transformar estos estadísticos por cada grupo en un único estadístico de prueba. Para ello, se considera la suma de sus cuadrados, pues así todos los valores son positivos y las diferencias significativas se incrementan aún más (como en el caso de la varianza). Así, se define el estadístico de prueba χ^2, definido en la ecuación 8.2, donde m y n son, respectivamente, la cantidad de filas y la cantidad de columnas de la tabla de frecuencias, sin considerar los totales (puede ser útil en este punto repasar lo que aprendimos en el capítulo 3 sobre la distribución χ^2).

$$\chi^2 = \sum_{i=1}^{m}\sum_{j=1}^{n} Z_{ij}^2 = \sum_{i=1}^{m}\sum_{j=1}^{n} \frac{(\text{cantidad observada} - \text{cantidad esperada})^2}{\text{cantidad esperada}} \tag{8.2}$$

Para el ejemplo tenemos entonces:

$$\chi^2 = (-0.404)^2 + (0.370)^2 + (-0.139)^2 + (-0.189)^2 + (0.426)^2 + (0.572)^2 + \\ +(-0.523)^2 + (0.196)^2 + (0.267)^2 + (-0.603)^2 = 1,611$$

Como estamos sumando $m \cdot n$ valores Z al cuadrado, **el estadístico χ^2 sigue una distribución chi-cuadrado, con $\nu = (m-1) \cdot (n-1)$ grados de libertad**. En el ejemplo, $\nu = (2-1) \cdot (5-1) = 4$.

El valor p para la prueba chi-cuadrado está dado por el área bajo la curva de la distribución chi-cuadrado con valores mayores al obtenido para el estadístico de prueba. En este caso, gracias a la llamada en R `pchisq(1.611, df = 4, lower.tail = FALSE)`, obtenemos que $p = 0,807$. Suponiendo un nivel de significación $\alpha = 0,05$, $p > \alpha$, por lo que se falla al rechazar la hipótesis nula. Es decir, no hay evidencia suficientemente fuerte que sugiera, con 95 % de confianza, que programadores hombres y mujeres prefieran lenguajes de programación distintos.

En R, podemos realizar la prueba chi-cuadrado de homogeneidad como muestra el script 8.1, usando para ello la función `chisq.test(x)`, donde `x` corresponde a la matriz de confusión.

Al ejecutar el script, debemos tener en cuenta que el valor p obtenido usando R es ligeramente diferente debido a los redondeos aplicados en la tabla 8.2 y al resolver la ecuación 8.2.

Script 8.1: prueba chi-cuadrado de homogeneidad.

```r
# Crear tabla de contingencia.
programadores <- c(42, 56, 51, 27, 24)
programadoras <- c(25, 24, 27, 15, 9)

tabla <- as.table(rbind(programadores, programadoras))

dimnames(tabla) <- list(sexo = c("programadores", "programadoras"),
                        lenguajes = c("C", "Java", "Python", "Ruby", "Otro"))

print(tabla)

# Hacer prueba chi-cuadrado de homogeneidad.
prueba <- chisq.test(tabla)
print(prueba)
```

8.1.2 Prueba chi-cuadrado de bondad de ajuste

Esta prueba **permite comprobar si una distribución de frecuencias observada se asemeja a una distribución esperada**. Usualmente se emplea para comprobar si una muestra es representativa de la población (NIST/SEMATECH, 2013, p. 1.3.5.15).

Para entender mejor esta idea, supongamos ahora que una gran empresa de desarrollo de software cuenta con una nómina de 660 programadores, especialistas en diferentes lenguajes de programación. El gerente ha seleccionado un subconjunto de 55 programadores, supuestamente de forma aleatoria, para enviarlos a cursos de perfeccionamiento en sus respectivos lenguajes, pero el sindicato lo ha acusado de "seleccionar estas personas a conveniencia de los intereses mezquinos de la gerencia, impidiendo que el grupo sea representativo a fin de asegurar una mejora en la productividad de toda la empresa". Ante el inminente riesgo de movilizaciones, el gerente necesita demostrar que el grupo seleccionado es una muestra representativa de sus programadores.

La tabla 8.4 muestra la cantidad de especialistas en cada lenguaje, tanto para la nómina de la empresa como para la muestra seleccionada.

Como ya es habitual, comencemos por verificar las condiciones. Puesto que la muestra representa menos del 10 % de la población y fue elegida de manera aleatoria, las observaciones son independientes entre sí.

La segunda condición resulta algo más compleja. Comencemos por calcular la proporción de programadores de la nómina (población) especialista en cada lenguaje. Para el caso de C, tenemos:

Lenguaje	**C**	**Java**	**Python**	**Ruby**	**Otro**
Nómina	236	78	204	76	66
Muestra	17	9	14	10	5

Tabla 8.4: frecuencias por lenguaje de programación para la toda la nómina y para la muestra.

$$P_C = \frac{n_C}{n} = \frac{236}{660} = 0,358$$

En consecuencia, esperaríamos la misma proporción de especialistas en C en la muestra, es decir:

$$E_C = P_C \cdot n = 0,358 \cdot 55 = 19,690$$

Repitiendo este proceso para los lenguajes restantes, obtenemos las proporciones para la población y valores esperados para la muestra que se presentan en la tabla 8.5. En ella podemos ver que para cada grupo se esperan más de 5 observaciones, por lo que se verifica la segunda condición.

Lenguaje	**C**	**Java**	**Python**	**Ruby**	**Otro**
Proporciones nómina	0,358	0,118	0,309	0,115	0,100
Valores esperados muestra	19,690	6,490	16,995	6,325	5,500

Tabla 8.5: proporciones de la población y valores esperados de la muestra.

En este ejemplo, las hipótesis a contrastar son:

H_0: las proporciones de especialistas en cada lenguaje son las mismas para la nómina y la muestra.
H_A: las proporciones de especialistas en cada lenguaje son diferentes en la nómina que en la muestra.

En este caso, se puede proceder de igual manera que para la prueba de bondad de ajuste, como muestra el script 8.2. Para este ejemplo, el valor p resultante es $p = 0,461$, por lo que se falla al rechazar la hipótesis nula con un nivel de significación $\alpha = 0,05$. En consecuencia, podemos concluir con 95 % de confianza que la muestra seleccionada es, en efecto, representativa de la nómina de programadores de la empresa, por lo que la acusación del sindicato no tiene fundamentos.

Script 8.2: prueba chi-cuadrado de bondad de ajuste.

```
# Crear tabla de contingencia.
nomina <- c(236, 78, 204, 76, 66)
muestra <- c(17, 9, 14, 10, 5)

tabla <- as.table(rbind(nomina, muestra))

dimnames(tabla) <- list(grupo = c("Nómina", "Muestra"),
                        lenguajes = c("C", "Java", "Python", "Ruby", "Otro"))

print(tabla)

# Verificar si se esperan más de 5 observaciones por cada grupo.
n_nomina <- sum(nomina)
n_muestra <- 55
proporciones <- round(nomina/n_nomina, 3)
esperados <- round(proporciones * n_muestra, 3)
print(esperados)

# Hacer prueba chi-cuadrado de homogeneidad.
prueba <- chisq.test(tabla, correct = FALSE)
print(prueba)
```

8.1.3 Prueba chi-cuadrado de independencia

Esta prueba permite **determinar si dos variables categóricas, de una misma población, son estadísticamente independientes** o si, por el contrario, están relacionadas.

Tomemos en este caso como ejemplo que un micólogo desea determinar si existe relación entre la forma del sombrero de los hongos y si estos son o no comestibles. Para ello, tras recolectar una muestra de 8.120 hongos, obtiene la tabla de contingencia que se muestra en la tabla 8.6[1].

		Forma del sombrero					
		Campana	**Convexo**	**Hundido**	**Nudoso**	**Plano**	**Total**
Clase	**Comestible**	404	1.948	32	228	1.596	4.208
	Venenoso	48	1.708	0	600	1.556	3.912
	Total	452	3.656	32	828	3.152	8.120

Tabla 8.6: tabla de contingencia para las características de los hongos.

Una vez más, comencemos por verificar las condiciones. Podemos suponer que la muestra fue obtenida de manera aleatoria, ya que se trata de un estudio publicado en una revista científica, y, desde luego, representa menos del 10 % de la población mundial de hongos. En consecuencia, se verifica la condición de independencia de las observaciones en las muestras.

Ahora debemos determinar cuántas observaciones esperaríamos tener en cada grupo si las variables fueran independientes. En este caso, la frecuencia esperada para cada celda está dado por la ecuación 8.3, donde:

- n_i: total de observaciones en la fila i.
- n_j: total de observaciones en la columna j.
- n: tamaño de la muestra.

$$E_{i,j} = \frac{n_i \cdot n_j}{n} \tag{8.3}$$

De acuerdo a esto, la cantidad de hongos comestibles con sombrero en forma de campana que esperaríamos encontrar es:

$$E_{1,1} = \frac{4.208 \cdot 452}{8.120} = 234,238$$

Si replicamos este cálculo para cada celda de nuestra matriz de confusión, se obtienen los valores esperados que se presentan en la tabla 8.7. Podemos ver que todos los valores esperados superan las 5 observaciones, por lo que podemos proceder con la prueba χ^2 de independencia.

		Forma del sombrero				
		Campana	**Convexo**	**Hundido**	**Nudoso**	**Plano**
Clase	**Comestible**	234,238	1.894,636	16,583	429,092	1.633,450
	Venenoso	217,762	1.761,364	15,417	398,908	1.518,550

Tabla 8.7: frecuencias esperadas para los hongos.

En este caso, las hipótesis a docimar son:

H_0: las variables clase y forma del sombrero son independientes.
H_A: las variables clase y forma del sombrero están relacionadas.

[1] Datos obtenidos desde el conjunto de datos Mushroom, disponible en https://archive.ics.uci.edu/ml/datasets/mushroom (última visita: 26-04-2021).

Al ejecutar la prueba en R (script 8.3) obtenemos que el valor para el estadístico de prueba es $\chi^2 = 485,64$, con $\nu = 4$ grados de libertad y un valor p $< 2,2 \cdot 10^{-16}$. Aún para un nivel de significación muy exigente, como $\alpha = 0,01$, el valor p obtenido nos permite rechazar la hipótesis nula en favor de la hipótesis alternativa. En consecuencia, concluimos con 99 % de confianza que las variables clase y forma del sombrero están relacionadas (son dependientes).

Script 8.3: prueba chi-cuadrado de independencia.

```
# Crear tabla de contingencia.
comestible <- c(404, 1948, 32, 228, 1596)
venenoso <- c(48, 1708, 0, 600, 1556)

tabla <- as.table(rbind(comestible, venenoso))

dimnames(tabla) <- list(tipo = c("comestible", "venenoso"),
                        sombrero = c("campana", "convexo", "hundido",
                                     "nudoso", "plano"))


print(tabla)

# Hacer prueba chi-cuadrado de independencia.
prueba <- chisq.test(tabla)
cat("\nLa prueba internamente calcula los valores esperados:\n")
esperados <- round(prueba[["expected"]], 3)
print(esperados)

cat("\nResultado de la prueba:\n")
print(prueba)
```

8.2 PRUEBAS PARA MUESTRAS PEQUEÑAS

Hemos visto que la prueba χ^2 nos pide que las observaciones esperadas para cada grupo sean a lo menos 5. Sin embargo, hay escenarios donde esta condición no se cumple, por lo que debemos recurrir a alguna alternativa.

8.2.1 Prueba exacta de Fisher

La **prueba exacta de Fisher** es una alternativa a la prueba χ^2 de independencia en el caso de que **ambas variables sean dicotómicas**. Así, las hipótesis a contrastar son:

H_0: las variables son independientes.
H_A: las variables están relacionadas.

En este escenario, las frecuencias de la muestra pueden resumirse en una tabla de contingencia de 2×2, como muestra la tabla 8.8.

Si se asume independencia entre ambas variables y los totales por filas y columnas son fijos, la **probabilidad exacta de observar el conjunto de frecuencias de la tabla 8.8** está dada por la ecuación 8.4,

		Variable 1		
		Presente	**Ausente**	**Total**
Variable 2	**Presente**	a	b	a+b
	Ausente	c	d	c+d
	Total	a+c	b+d	n

Tabla 8.8: tabla de contingencia para dos variables categóricas con dos niveles cada una.

correspondiente a la función de distribución hipergeométrica.

$$p = \frac{(a+b)!(c+d)!(a+c)!(b+d)!}{n!a!b!c!d!} \tag{8.4}$$

La prueba lleva en su nombre la palabra **exacta** porque internamente construye todas las tablas posibles con los mismos totales marginales que recibe como entrada y, para cada una de ellas, determina la probabilidad exacta de observarla. El valor p corresponde en este caso a la suma de las probabilidades de todas las tablas con probabilidad menor o igual que la tabla dada.

Para entender mejor esta prueba, supongamos que un controvertido estudio desea determinar si dos vacunas, Argh y Grrr, son igualmente efectivas para inmunizar a la población ante una mordida de vampiro. Para ello, los investigadores reclutaron a 17 voluntarios de todo el mundo, de los cuales 6 recibieron la vacuna Argh y los 11 restantes, la Grrr. Al cabo de tres meses, sometieron a cada uno de los participantes a una mordida de vampiro y observaron que ninguno de los voluntarios que recibieron la vacuna Argh resultó afectado, mientras que 5 de los que recibieron la vacuna Grrr se convirtieron en vampiros, como resume la tabla 8.9.

		Vacuna		
		Argh	**Grrr**	**Total**
Resultado	**Vampiro**	0	5	5
	Humano	6	6	12
	Total	6	11	17

Tabla 8.9: tabla de contingencia con los contagios producidos en el experimento.

La probabilidad de observar un conjunto de frecuencias con los mismos totales por fila y por columna, si las variables son realmente independientes está dada por:

$$p = \frac{(a+b)!(c+d)!(a+c)!(b+d)!}{n!a!b!c!d!} = \frac{5!12!6!11!}{17!0!5!6!6!} = 0,075$$

Son cinco las posibles tablas (además de la obtenida) con iguales valores marginales, como podemos ver en la tabla 8.10.

Calculando las probabilidades para cada una de ellas de acuerdo a la ecuación 8.4, se tiene que:

- Probabilidad para la tabla 8.10a: 0,001.
- Probabilidad para la tabla 8.10b: 0,320.
- Probabilidad para la tabla 8.10c: 0,027.
- Probabilidad para la tabla 8.10d: 0,400.
- Probabilidad para la tabla 8.10e: 0,178.

Así, el valor p está dado por la suma de las probabilidades de las tablas con probabilidad menor o igual a la de los datos observados:

$$p = 0,075 + 0,001 + 0,027 = 0,103$$

		Vacuna		
		Argh	**Grrr**	**Total**
Resultado	**Infectado**	5	0	5
	Sano	1	11	12
	Total	6	11	17

(a)

		Vacuna		
		Argh	**Grrr**	**Total**
Resultado	**Infectado**	1	4	5
	Sano	5	7	12
	Total	6	11	17

(b)

		Vacuna		
		Argh	**Grrr**	**Total**
Resultado	**Infectado**	4	1	5
	Sano	2	10	12
	Total	6	11	17

(c)

		Vacuna		
		Argh	**Grrr**	**Total**
Resultado	**Infectado**	2	3	5
	Sano	4	8	12
	Total	6	11	17

(d)

		Vacuna		
		Argh	**Grrr**	**Total**
Resultado	**Infectado**	3	2	5
	Sano	3	9	12
	Total	6	11	17

(e)

Tabla 8.10: tablas con los mismos valores marginales que los obtenidos.

Considerando un nivel de significación $\alpha = 0,05$, se falla al rechazar la hipótesis nula. En consecuencia, se concluye con 95 % de confianza que no hay una asociación estadísticamente significativa entre la cantidad de nuevos vampiros y la vacuna recibida.

En R, podemos llevar a cabo esta prueba mediante la función `fisher.test(x, conf.level)`, donde `x` corresponde a la tabla de contingencia y `conf.level`, al nivel de confianza. El script 8.4 muestra el su uso para el ejemplo (con una pequeña diferencia en el valor p obtenido debido a los redondeos efectuados en el cálculo anterior).

Script 8.4: prueba exacta de Fisher.

```r
# Construir la tabla de contingencia.
vacuna <- c(rep("Argh", 6), rep("Grrr", 11))
resultado <- c(rep("Humano", 12), rep("Vampiro", 5))
datos <- data.frame(resultado, vacuna)
tabla <- xtabs(~., datos)
print(tabla)

# Aplicar prueba exacta de Fisher.
alfa <- 0.05
prueba <- fisher.test(tabla, 1-alfa)
```

```
print(prueba)
```

8.2.2 Prueba de mcNemar

Esta prueba resulta apropiada cuando una misma característica, con respuesta dicotómica, se mide en dos ocasiones diferentes para los mismos sujetos (muestras pareadas) y queremos determinar si se produce o no un cambio significativo entre ambas mediciones. Una vez más, podemos registrar las frecuencias en una matriz de confusión como la que vimos en 8.8. En ella, podemos ver que las celdas a y d corresponde a instancias en que no hay cambios. La celda b en dicha tabla representa a las instancias que cambian de `Presente` a `Ausente` y la celda c, a instancias que cambian de `Ausente` a `Presente`.

Las hipótesis asociadas a la prueba de mcNemar son:

H_0: **no** hay cambios significativos en las respuestas.
H_A: **sí** hay cambios significativos en las respuestas.

Puesto que nos interesa medir los cambios, solo nos sirven las celdas b y c de la tabla de contingencia. La cantidad de instancias en que se producen cambios es $b+c$ y, de acuerdo a la hipótesis nula, se esperaría que $(b+c)/2$ cambien en un sentido y que las $(b+c)/2$ restantes lo hicieran en sentido contrario. Así, b y c cuentan respectivamente los éxitos y los fracasos de una distribución binomial de $b+c$ intentos con probabilidad de éxito igual a $1/2$. Cuando $(b+c) > 10$, esta distribución binomial se asemeja a una distribución normal con la misma media $((b+c)/2)$ y desviación estándar $\sqrt{(b+c)/4}$, a partir de la cual se puede obtener un estadístico z. Sin embargo, la mayoría de los paquetes de software para estadística (incluido R) reportan el cuadrado de dicho estadístico (e ignoran completamente los casos en que hay 10 o menos cambios entre las mediciones), el cual sigue una distribución χ^2 con un grado de libertad y se calcula como muestra la ecuación 8.5 (Agresti, 2019).

$$\chi^2 = \frac{(b-c)^2}{b+c} \tag{8.5}$$

Puesto que los datos siguen una distribución binomial (discreta), pero se está usando como aproximación la distribución chi-cuadrado (continua), suele emplearse un **factor de corrección de continuidad** propuesta por Frank Yates en 1934. El estadístico de prueba con la corrección de Yates se calcula en realidad como muestra la ecuación 8.6.

$$\chi^2 = \frac{(|b-c|-1)^2}{b+c} \tag{8.6}$$

Para ilustrar el funcionamiento de la prueba de mcNemar, suponga que un cientista de datos ha construido dos modelos para predecir, a partir de las notas obtenidas en cursos previos, si sus estudiantes aprobarán o no la asignatura de aprendizaje automático. Al probar sus modelos con los 25 estudiantes del semestre anterior, observó que predijeron el resultado final de cada estudiante como muestra la tabla 8.11 y se resume en la matriz de confusión de la tabla 8.12.

El cientista de datos desea saber si existe diferencia entre el desempeño de ambos algoritmos, por lo que decide emplear la prueba de mcNemar. Al calcular el estadístico de prueba (con el factor de corrección), obtiene:

$$\chi^2 = \frac{(|5-7|-1)^2}{5+7} = \chi^2 = \frac{(5-7)^2}{5+7} = 0,083$$

Alumno	Modelo 1	Modelo 2
1	Correcto	Correcto
2	Correcto	Correcto
3	Correcto	Correcto
4	Correcto	Correcto
5	Correcto	Correcto
6	Correcto	Correcto
7	Correcto	Correcto
8	Correcto	Correcto
9	Correcto	Correcto
10	Correcto	Incorrecto
11	Correcto	Incorrecto
12	Correcto	Incorrecto
13	Correcto	Incorrecto
14	Correcto	Incorrecto
15	Correcto	Incorrecto
16	Correcto	Incorrecto
17	Incorrecto	Incorrecto
18	Incorrecto	Incorrecto
19	Incorrecto	Incorrecto
20	Incorrecto	Incorrecto
21	Incorrecto	Correcto
22	Incorrecto	Correcto
23	Incorrecto	Correcto
24	Incorrecto	Correcto
25	Incorrecto	Correcto

Tabla 8.11: resultados de la predicción para cada estudiante con ambos modelos.

		Modelo 1		
		Correcto	**Incorrecto**	**Total**
Modelo 2	**Correcto**	9	5	14
	Incorrecto	7	4	11
	Total	16	9	25

Tabla 8.12: tabla de contingencia con las predicciones de los resultados finales de los estudiantes.

El valor p está dado por el área bajo la cola superior de la distribución chi-cuadrado, que en R puede calcularse como `pchisq(0.083, 1, lower.tail = FALSE)`, obteniéndose que $p = 0,773$. En consecuencia, se falla al rechazar la hipótesis nula (para un nivel de significación $\alpha = 0,05$) y se concluye que no hay diferencia en el desempeño de ambos clasificadores.

La función de R para esta prueba, que por defecto aplica el factor de corrección, es `mcNemar.test(x)`, donde `x` corresponde a la tabla de contingencia. El script 8.5 muestra su aplicación para el ejemplo dado.

Script 8.5: prueba de mcNemar.

```r
# Construir la tabla de contingencia.
alumno <- seq(1:25)
modelo_1 <- c(rep("Correcto", 16), rep("Incorrecto", 9))
modelo_2 <- c(rep("Correcto", 9), rep("Incorrecto", 11), rep("Correcto", 5))
datos <- data.frame(alumno, modelo_2, modelo_1)
tabla <- table(modelo_2, modelo_1)
print(tabla)

# Aplicar prueba de McNemar.
```

```
prueba <- mcnemar.test(tabla)
print(prueba)
```

8.3 PRUEBA Q DE COCHRAN

La **prueba Q de Cochran** es una extensión de la prueba de mcNemar, adecuada cuando la variable de respuesta es dicotómica y la variable independiente tiene más de dos observaciones pareadas (cuando ambas variables son dicotómicas, esta prueba es equivalente a la de mcNemar). Como tal, debería estar incluida en la sección precedente, pero le dedicaremos una sección aparte pues la explicación requiere de algunos conceptos importantes que no hemos estudiado aún.

Veamos esta prueba por medio de un ejemplo. Elsa Capunta, estudiante de un curso de algoritmos, tiene como tarea determinar si existe una diferencia significativa en el desempeño de tres metaheurísticas que buscan resolver el el problema del vendedor viajero. Para ello, el profesor le ha proporcionado los datos presentados en la tabla 8.13, donde la columna `instancia` identifica cada instancia del problema empleada para evaluar las metaheurísticas y las restantes columnas indican si la metaheurística en cuestión encontró (1) o no (0) la solución óptima para dicha instancia.

Instancia	**Simulated Annealing**	**Colonia de hormigas**	**Algoritmo genético**
1	0	0	1
2	1	0	0
3	0	1	1
4	0	0	1
5	0	0	1
6	0	1	1
7	0	0	0
8	1	0	1
9	0	0	0
10	0	1	1
11	0	0	1
12	0	0	0
13	1	0	0
14	0	0	1
15	0	1	1

Tabla 8.13: resultados de las metaheurísticas para cada instancia con ambos modelos.

Las hipótesis contrastadas por la prueba Q de Cochran son:

H_0: la proporción de "éxitos" es la misma para todos los grupos.
H_A: la proporción de "éxitos" es distinta para al menos un grupo.

Como ya debemos suponer, esta prueba también requiere que se cumplan algunas condiciones:

1. La variable de respuesta es dicotómica.
2. La variable independiente es categórica.
3. Las observaciones son independientes entre sí.
4. El tamaño de la muestra es lo suficientemente grande. Glen (2016a) sugiere que $n \cdot k \geq 24$, donde n es el tamaño de la muestra (la cantidad de instancias, para el ejemplo) y k, la cantidad de niveles en la variable independiente.

El estadístico de prueba se calcula como muestra la ecuación 8.7, donde:

- b: cantidad de bloques.
- k: cantidad de bloques (niveles de la variable independiente).
- x_j: total de éxitos en la columna j.
- x_i: total de éxitos en la fila i.
- N: número total de éxitos.

$$Q = k(k-1)\frac{\sum_{j=1}^{k}\left(x_j - \frac{N}{k}\right)^2}{\sum_{i=1}^{b} x_i(k - x_i)} \tag{8.7}$$

Podemos ver que los cálculos que se llevan a cabo para esta prueba son complejos, por lo que suele hacerse mediante software. En R, esta prueba está implementada en la función `cochran.qtest(formula, data, alpha = 0.05)` del paquete `RVAideMemoire`, donde:

- `formula`: fórmula de la forma `respuesta` $\sim$ `independiente | bloques`.
- `data`: matriz de datos en formato largo.
- `alpha`: nivel de significación.

Al ejecutar el script 8.6, obtenemos el resultado que se muestra en la figura 8.1. Tenemos que el valor p es $p = 0,028$, menor que el nivel de significación $\alpha = 0,05$, por lo que rechazamos la hipótesis nula en favor de la hipótesis alternativa. En consecuencia, Elsa concluye con 95 % de confianza que al menos una de las metaheurísticas tiene un desempeño diferente a las demás.

```
Cochran's Q test

data:  resultado by metaheuristica, block = instancia
Q = 7.1667, df = 2, p-value = 0.02778
alternative hypothesis: true difference in probabilities is not equal to 0
sample estimates:
proba in group annealing  proba in group genetico  proba in group hormigas
               0.2000000                0.6666667                0.2666667

        Pairwise comparisons using Wilcoxon sign test

         annealing genetico
genetico   0.09814        -
hormigas   1.00000  0.09375

P value adjustment method: fdr
```

Figura 8.1: resultado de la prueba Q de Cochran.

Todo buen estudiante sabe que Elsa debe entregar en su tarea una respuesta más detallada que la que hemos obtenido hasta ahora, pues el profesor esperaría un análisis de las diferencias.

En este punto, debemos mencionar que la hipótesis nula de la prueba Q de Cochran no es específica, sino que comprueba la igualdad de todas las proporciones. Esta clase de hipótesis nula suele llamarse **ómnibus** (en ocasiones también colectiva o global). Así, se dice que la prueba Q de Cochran es una prueba ómnibus porque tiene esta clase de hipótesis nula, con la dificultad de que solo detecta si existe al menos bloque con una proporción de "éxito" diferente. Sin embargo, de ser afirmativa la respuesta, no nos dice qué grupos presentan diferencias (Lane, s.f.). Desde luego, existen métodos para responder a esta última pregunta, llamados **pruebas *post-hoc***, o también ***a posteriori***. Reciben este nombre porque se realizan una vez que se ha concluido gracias a la prueba ómnibus que existen diferencias significativas.

Algo importante que debemos recordar: **solo haremos un procedimiento post-hoc si la prueba ómnibus rechaza la hipótesis nula** en favor de la hipótesis alternativa. Además, el procedimiento post-hoc realizado debe considerar el mismo nivel de significación que la prueba ómnibus.

En el caso de la prueba Q de Cochran, el procedimiento post-hoc consiste en efectuar pruebas de mcNemar entre cada par de bloques. R nos permite hacer esto mediante la función `pairwiseMcnemar(formula, data, method)` del paquete `rcompanion`, donde `formula` y `data` son las mismas que para la prueba Q de Cochran y `method` nos permite determinar el método para ajustar los valores p de las comparaciones. Pero... ¿por qué querríamos ajustar los valores p?

Como explican Goeman y Solari (2014), cuando contrastamos hipótesis acotamos la probabilidad de cometer errores tipo I por medio del nivel de significación α. Sin embargo, cuando hacemos múltiples contrastes de hipótesis simultáneamente, cada uno de ellos tendrá una probabilidad α de cometer un error de tipo I. Esto se traduce en un **incremento de la probabilidad de cometer este tipo de errores** a medida que aumenta la cantidad de hipótesis contrastadas y, en consecuencia, en una reducción del poder estadístico.

Muchos factores de corrección tienen por objeto distribuir el nivel de significación empleado para la prueba ómnibus en cada prueba de pares de bloques. El método más sencillo para ajustar los valores p es la **corrección de Bonferroni**. Como explica la ayuda de R, esta corrección simplemente multiplica el valor p obtenido en cada prueba por la cantidad de pruebas realizadas. En general, no se recomienda el uso del método de Bonferroni, especialmente si el número de grupos es alto, pues es considerado muy **conservador**, lo que significa que mantiene la probabilidad de cometer un error tipo I más baja que el nivel de significación establecido (y es, por ende, más propensa a cometer errores tipo II).

Otra alterativa es la **corrección de Holm** (Glen, 2016b), con mayor poder estadístico que la de Bonferroni. Esta corrección comienza por efectuar las pruebas entre pares de bloques y luego ordena los valores p en forma creciente. A continuación, se calcula el factor de Holm, HB, para cada par de bloques, dado por la ecuación 8.8, donde:

- α: nivel de significación.
- N: cantidad de comparaciones efectuadas.
- i: importancia de la comparación (posición en la lista de valores p ordenados).

$$HB_i = \frac{\alpha}{N - i + 1} \tag{8.8}$$

Luego, se compara el valor p con su respectivo factor de Holm y, si el valor p es menor, se considera que existe una diferencia significativa. R implementa esta corrección de manera ligeramente diferente, de modo que el valor p ajustado pueda ser comparado con el nivel de significación original.

Si has estado leyendo de manera atenta, habrás notado que en el resultado entregado por `cochran.qtest()` para el ejemplo (figura 8.1) aparece otro procedimiento post-hoc adecuado para la prueba Q de Cochran, aunque no lo presentaremos aquí pues se basa en una prueba que estudiaremos en capítulos posteriores.

El script 8.6 incluye también los procedimientos *post-hoc* mediante pruebas de mcNemar usando las correcciones de Holm y Bonferroni, obteniéndose los resultados que se muestran en la figura 8.2.

Podemos ver en la figura 8.2 que, aún cuando la prueba Q de Cochran indica que existen diferencias significativas entre las metaheurísticas, ninguno de los procedimientos post-hoc ha detectado diferencias significativas entre pares de bloques. En consecuencia, la respuesta que Elsa debe dar a su profesor es que la evidencia no es lo suficientemente fuerte para poder afirmar que existen diferencias entre las metaheurísticas, pero que podría ser adecuado hacer un estudio con una muestra mayor puesto que los resultados de la prueba Q de Cochran y de los procedimientos post-hoc son contradictorios.

Script 8.6: prueba Q de Cochran.

```
library(tidyverse)
library(RVAideMemoire)
library(rcompanion)

# Crear matriz de datos.
instancia <- 1:15
```

```
Cochran's Q test

Procedimiento post-hoc con corrección de Bonferroni
$Test.method
   Test
1 exact

$Adustment.method
      Method
1 bonferroni

$Pairwise
                Comparison Successes Trials p.value p.adjust
1 annealing - genetico = 0         2     11  0.0654   0.1960
2 annealing - hormigas = 0         3      7       1   1.0000
3  genetico - hormigas = 0         6      6  0.0313   0.0939


Procedimiento post-hoc con corrección de Holm
$Test.method
   Test
1 exact

$Adustment.method
  Method
1   holm

$Pairwise
                Comparison Successes Trials p.value p.adjust
1 annealing - genetico = 0         2     11  0.0654   0.1310
2 annealing - hormigas = 0         3      7       1   1.0000
3  genetico - hormigas = 0         6      6  0.0313   0.0939
```

Figura 8.2: resultados de los procedimientos post-hoc.

```r
annealing <- c(0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0)
hormigas <- c(0, 0, 1, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1)
genetico <- c(1, 0, 1, 1, 1, 1, 0, 1, 0, 1, 1, 0, 0, 1, 1)
datos <- data.frame(instancia, annealing, hormigas, genetico)

# Llevar matriz de datos a formato largo.
datos <- datos %>% pivot_longer(c("annealing", "hormigas", "genetico"),
                                names_to = "metaheuristica",
                                values_to = "resultado")

datos[["instancia"]] <- factor(datos[["instancia"]])
datos[["metaheuristica"]] <- factor(datos[["metaheuristica"]])

# Hacer prueba Q de Cochran.
prueba <- cochran.qtest(resultado ~ metaheuristica | instancia,
                        data = datos, alpha = 0.05)

print(prueba)

# Procedimiento post-hoc con corrección de Bonferroni.
```

```r
post_hoc_1 <- pairwiseMcnemar(resultado ~ metaheuristica | instancia,
                              data = datos, method = "bonferroni")

cat("\nProcedimiento post-hoc con corrección de Bonferroni\n")
print(post_hoc_1)

# Procedimiento post-hoc con corrección de Holm.
post_hoc_2 <- pairwiseMcnemar(resultado ~ metaheuristica | instancia,
                              data = datos, method = "holm")

cat("\nProcedimiento post-hoc con corrección de Holm\n")
print(post_hoc_2)
```

8.4 EJERCICIOS PROPUESTOS

1. Explica cómo se calcula el estadístico χ^2.
2. Menciona las condiciones para que una prueba de hipótesis χ^2 sea válida.
3. Da un ejemplo en que se requiera utilizar una prueba χ^2 de homogeneidad. ¿Qué hipótesis nula y alternativa serían docimadas?
4. Da un ejemplo en que se requiera utilizar una prueba χ^2 de bondad de ajuste. ¿Qué hipótesis nula y alternativa serían docimadas?
5. Da un ejemplo en que se requiera utilizar una prueba χ^2 de independencia. ¿Qué hipótesis nula y alternativa serían docimadas?
6. Un estudio clínico reclutó a 32 pacientes con fatiga crónica para determinar si un tratamiento basado en inyecciones intramusculares de magnesio es efectivo para esta condición. De los 15 pacientes que recibieron estas inyecciones, seleccionados de manera aleatoria, 12 reportaron sentirse mejor (80 %), mientras que solo 3 pacientes de los 17 que recibieron inyecciones placebo (18 %) reportaron mejorías.
   a) ¿Se cumplen las condiciones para aplicar una prueba exacta de Fisher al problema enunciado?
   b) ¿Cuáles serían las hipótesis nula y alternativa para esta prueba?
   c) Independientemente de la respuesta anterior, aplica la prueba usando R y luego de forma manual (Ayuda: hay 16 tablas que mantienen los totales marginales en el enunciado).
   d) ¿A qué conclusión lleva este procedimiento?
7. Antes del debate de candidatos presidenciales, una encuesta consultó a 1.000 auditores si apoyaban o no una reforma constitucional para permitir matrimonio igualitario, encontrando 705 personas a favor y 295 en contra. Luego de que estas personas escucharon el debate, 663 se manifestaron a favor y 337 en contra de la reforma. 73 encuestados cambiaron de opinión de en contra a en apoyo de la medida, mientras que 115 cambiaron su opinión a favor para estar en contra.
   a) ¿Se cumplen las condiciones para aplicar una prueba de mcNemar al problema enunciado?
   b) ¿Cuáles serían las hipótesis nula y alternativa si usamos esta prueba?
   c) Independientemente de la respuesta anterior, aplica la prueba usando R y luego de forma manual.
   d) ¿A qué conclusión lleva este procedimiento?
8. Con palabras propias ¿qué es una prueba ómnibus? ¿Con qué otros nombres se les conoce?
9. Con palabras propias ¿qué es una prueba post-hoc? ¿Cuándo se aplican?
10. Con palabras propias, cuando hay más de dos grupos ¿por qué es problemático hacer múltiples pruebas entre pares de esos grupos?
11. Las autoridades de la universidad desean conocer si las semanas de receso (sin actividades docentes) ayuda o no al descanso del estudiantado. Para eso seleccionaron 20 estudiantes de forma aleatoria y les consultaron si se sentían “cansada/o” o “descansada/o” en tres ocasiones: el lunes, miércoles y viernes de la primera semana de receso del semestre. Los resultados se muestran en la siguiente tabla, donde 0 representa cansancio y 1 descanso.

Estudiante	Lunes	Miércoles	Viernes
1	1	1	1
2	0	1	1
3	0	0	1
4	0	1	0
5	1	0	0
6	0	1	1
7	0	1	1
8	0	0	1
9	0	1	1
10	0	1	0
11	1	1	0
12	1	1	1
13	0	0	0
14	1	0	1
15	0	1	1
16	0	1	0
17	0	0	1
18	0	1	1
19	1	0	1
20	0	1	1

a) ¿Hay diferencias entre los tres periodos de tiempo sin actividades? No olvide enunciar las hipótesis, seleccionar una prueba para docimarlas y verificar si se cumplen las condiciones necesarias para realizar la prueba seleccionada.

b) Si hay diferencias, ¿entre qué periodos se encuentran? No olvide justificar el procedimiento post-hoc seguido si corresponde.

REFERENCIAS

Agresti, A. (2019). An introduction to categorical data analysis (3.ª ed.). John Wiley & Sons, Inc.

Agresti, A. & Coull, B. A. (1998). Approximate is better than "exact" for interval estimation of binomial proportions. The American Statistician, 52(2), 119-126.

Bache, S. (2014). Introducing magrittr. Consultado el 7 de abril de 2021, desde https://cran.r-project.org/web/packages/magrittr/vignettes/magrittr.html

Brown, L. D., Cai, T. T. & DasGupta, A. (2001). Interval estimation for a binomial proportion. Statistical science, 16(2), 101-117.

Carchedi, N., De Mesmaeker, D. & Vannoorenberghe, L. (s.f.). RDocumentation. Consultado el 2 de abril de 2021, desde https://www.rdocumentation.org/

Champely, S., Ekstrom, C., Dalgaard, P., Gill, J., Weibelzahl, S., Anandkumar, A., Ford, C., Volcic, R. & de Rosario, H. (2020). pwr: Basic Functions for Power Analysis. Consultado el 1 de octubre de 2021, desde https://cran.r-project.org/web/packages/pwr/vignettes/pwr-vignette.html

Cross, J. (2017). Discrete Random Variables. Consultado el 9 de abril de 2021, desde https://rpubs.com/jcross/discreteRV

Dagnino, J. (2014). Tipos de datos y escalas de medida. Revista Chilena de Anestesia, 42(2), 109-111.

Devore, J. L. (2008). Probabilidad y Estadística para Ingeniería y Ciencias (7.ª ed.). CENAGE Learning.

Diez, D., Barr, C. D. & Çetinkaya-Rundel, M. (2017). OpenIntro Statistics (3.ª ed.). https://www.openintro.org/book/os/.

Field, A., Miles, J. & Field, Z. (2012). Discovering statistics using R. SAGE Publications Ltd.

Freedman, D. A. (2009). Modelización. Cambridge University Press.

Freund, R. J. & Wilson, W. J. (2003). Statistical Methods (2.ª ed.). Academic Press.

Glen, S. (2016a). Cochran's Q Test. Consultado el 9 de octubre de 2021, desde https://www.statisticshowto.com/cochrans-q-test/

Glen, S. (2016b). Holm-Bonferroni Method: Step by Step. Consultado el 7 de mayo de 2021, desde https://www.statisticshowto.com/holm-bonferroni-method/

Goeman, J. J. & Solari, A. (2014). Multiple hypothesis testing in genomics. Statistics in Medicine, 33(11), 1946-1978.

Joly, F. (1988). La Cartografía. Oikos-Tau, S.A. Ediciones.

Kabacoff, R. I. (2017). Power Analysis. Consultado el 1 de octubre de 2021, desde https://www.statmethods.net/stats/power.html

Kaplan, D. (2009). Statistical Modeling: A Fresh Approach. Consultado el 8 de marzo de 2019, desde http://works.bepress.com/daniel_kaplan/38

Kassambara, A. (2019a). Practical Statistics in R II - Comparing Goups: Numerical Variables. Datanovia.

Kassambara, A. (2019b). T-test Effect Size using Cohen's d Measure. Consultado el 27 de abril de 2021, desde https://www.datanovia.com/en/lessons/t-test-effect-size-using-cohens-d-measure/#cohens-d-for-paired-samples-t-test

Lane, D. (s.f.). Online Statistics Education: A Multimedia Course of Study. Consultado el 4 de mayo de 2021, desde https://onlinestatbook.com/

Mangiafico, S. S. (2016). Cochran's Q Test for Paired Nominal Data. Consultado el 9 de octubre de 2021, desde https://rcompanion.org/handbook/H_07.html

McCullagh, P. (2002). What Is a Statistical Model? The Annals of Statistics, 30(5), 1225-1267.

Meena, S. (2020). Statistics for Analytics and Data Science: Hypothesis Testing and Z-Test vs. T-Test. Consultado el 22 de septiembre de 2021, desde https://www.analyticsvidhya.com/blog/2020/06/statistics-analytics-hypothesis-testing-z-test-t-test/#h2_1

Mendez Ramírez, I. (1998). Empirismo, método científico y estadística. Revista de Geografia Agricola (Mexico).

Mendez Ramírez, I. (2012).
Método Científico: aspectos epistemológicos y metodológicos para el uso de la Estadística. SaberEs, 4.
Müller, K. (2021). dplyr. Consultado el 10 de septiembre de 2021, desde https://dplyr.tidyverse.org/
NIST/SEMATECH. (2013). e-Handbook of Statistical Methods.
Consultado el 29 de abril de 2021, desde http://www.itl.nist.gov/div898/handbook/
Parada, L. F. (2019). Prueba de normalidad de Shapiro-Wilk.
Consultado el 22 de septiembre de 2021, desde https://rpubs.com/F3rnando/507482
Pértega, S. & Pita, S. (2004).
Asociación de variables cualitativas: El test exacto de Fisher y el test de Mcnemar. Consultado el 29 de abril de 2021, desde https://www.fisterra.com/mbe/investiga/fisher/fisher.asp#Mcnemar
Real Academia Española. (2014). Diccionario de la lengua española (23.ª ed.).
Consultado el 30 de marzo de 2021, desde https://dle.rae.es
Ríos, S. (1995). Modelización. Alianza Ediciones.
RStudio. (2021). RStudio. Consultado el 2 de abril de 2021, desde https://rstudio.com/products/rstudio/
SAS Institute Inc. (2008). SAS/STAT® 9.2 User's Guide.
STDHA. (s.f.). Descriptive Statistics and Graphics - Easy Guides - Wiki - STHDA. Consultado el 31 de marzo de 2021, desde http://www.sthda.com/english/wiki/descriptive-statistics-and-graphics
The R Foundation. (s.f.). Documentation.
Consultado el 2 de abril de 2021, desde https://www.r-project.org/other-docs.html
Wallis, S. (2013). Binomial confidence intervals and contingency tests: mathematical fundamentals and the evaluation of alternative methods. Journal of Quantitative Linguistics, 20(3), 178-208.
Wickham, H. (2021). tidyr.
Consultado el 10 de septiembre de 2021, desde https://r4ds.had.co.nz/index.html
Wickham, H. & Grolemund, G. (2017). R for Data Science. https://r4ds.had.co.nz/index.html.
Willems, K. (2017). Formulas in R Tutorial.
Consultado el 11 de septiembre de 2021, desde https://dplyr.tidyverse.org/