Técnicas cuantitativas

Enric Aguilar y Benito Zaragozí

22-Mar-2021

Índice general

Antes de empezar

En este breve capítulo explicamos los aspectos básicos para que podáis reproducir los ejercicios y entregar las actividades utilizando R, Markdown y Rstudio. Aquí planteamos los conceptos básicos para preparar un documento que incluye explicaciónes, código y los resultados de análisis R. Se trata de una estratégia de ***programación literaria*** que en los últimos años está siendo cada vez más utilizada en el análisis de datos (Knuth, 1984; Xie, 2015).

0.1 Prerrequisitos

Para seguir las explicaciones de este curso será necesario instalar primero R y RStudio con los paquetes `knitr` y `rmarkdown`. A continuación mostramos la configuración general del sistema R utilizado.

```
sessionInfo()
```

```
## R version 4.0.2 (2020-06-22)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 20.04 LTS
##
## Matrix products: default
## BLAS/LAPACK: /usr/lib/x86_64-linux-gnu/openblas-pthread/libopenblasp-r0.3.8.so
##
## locale:
##  [1] LC_CTYPE=C.UTF-8       LC_NUMERIC=C           LC_TIME=C.UTF-8
##  [4] LC_COLLATE=C.UTF-8     LC_MONETARY=C.UTF-8    LC_MESSAGES=C
##  [7] LC_PAPER=C.UTF-8       LC_NAME=C              LC_ADDRESS=C
## [10] LC_TELEPHONE=C         LC_MEASUREMENT=C.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] nycflights13_1.0.1 kableExtra_1.2.1   corrplot_0.84      pander_0.6.3
##  [5] DiagrammeR_1.0.6.1 forcats_0.5.0      stringr_1.4.0      dplyr_1.0.2
```

```
##  [9] purrr_0.3.4        readr_1.4.0        tidyr_1.1.2        tibble_3.0.3
## [13] ggplot2_3.3.2      tidyverse_1.3.0
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_1.0.5          lubridate_1.7.9    visNetwork_2.0.9   utf8_1.1.4
##  [5] assertthat_0.2.1    digest_0.6.25      R6_2.4.1           cellranger_1.1.0
##  [9] backports_1.1.10    reprex_0.3.0       evaluate_0.14      highr_0.8
## [13] httr_1.4.2          pillar_1.4.6       rlang_0.4.7        readxl_1.3.1
## [17] rstudioapi_0.11     blob_1.2.1         rmarkdown_2.4      labeling_0.3
## [21] webshot_0.5.2       htmlwidgets_1.5.2  munsell_0.5.0      broom_0.7.1
## [25] compiler_4.0.2      modelr_0.1.8       xfun_0.18          pkgconfig_2.0.3
## [29] htmltools_0.5.0     tidyselect_1.1.0   bookdown_0.20      fansi_0.4.1
## [33] viridisLite_0.3.0   crayon_1.3.4       dbplyr_1.4.4       withr_2.3.0
## [37] grid_4.0.2          jsonlite_1.7.1     gtable_0.3.0       lifecycle_0.2.0
## [41] DBI_1.1.0           magrittr_1.5       scales_1.1.1       cli_2.0.2
## [45] stringi_1.5.3       farver_2.0.3       fs_1.5.0           xml2_1.3.2
## [49] ellipsis_0.3.1      generics_0.0.2     vctrs_0.3.4        RColorBrewer_1.1-2
## [53] tools_4.0.2         glue_1.4.2         hms_0.5.3          yaml_2.2.1
## [57] colorspace_1.4-1    rvest_0.3.6        knitr_1.30         haven_2.3.1
```

0.2 Tipos de ficheros

- Los archivos para producir documentos de RMarkdown tienen la extensión `.Rmd`.
- Los archivos deben abrirse con RStudio y se compilan haciendo clic en el botón knitr.
- El resultado es un documento en formato `.pdf`, `.html` o `.doc`.

0.3 Un ejemplo muy sencillo

Cread un fichero con el siguiente contenido

```
Hola, soy **R Markdown**

Aprende más sobre mi [aquí](http://rmarkdown.rstudio.com/).
```

al hacer clic en el botón `knitr a HTML` de Rstudio se crea un archivo `.html` con este contenido.

0.4 Estructura básica de un documento Markdown

Revisemos las partes más importantes del documento `Rmd`.

0.4.1 Cabecera

La cabecera está en la parte superior del documento dentro de estas dos líneas
---

```
---
title: "Write your title here"
author: "Write your name here"
date: "Write the date here"
output:
  pdf_document: default
  html_document: default
  word_document: default
---
```

En el encabezado del archivo debe escribir el título del documento, su nombre y la fecha. La declaración de salida se utiliza para la clase del documento final, `pdf`, `htmlo doc`. Para producir un PDF es necesario tener instalado un motor de LaTeX.

0.4.2 Insertar código R

A continuación, configuramos las opciones necesarias para imprimir el código R y la salida R en el documento final.

La primera línea es ocultar este fragmento de código en el documento final.

La segunda línea es para imprimir el código R y la salida R en el documento final.

0.4.3 Formatos de texto

El texto sin formato se escribe como en cualquier otro documento, como un documento de Word. Debe tener cuidado con las letras en cursiva o negrita y algunos caracteres especiales. Por ejemplo

- **Negrita**: escriba el texto entre **Negrita** o ___Negrita___
- *Cursiva*: escriba su texto entre*cursiva* o _Italica_
- Encabezados de sección

```
# Título 1
## Título 2
### Título 3
```

Cuantos más símbolos # escriba antes de su texto, menor será el tamaño de su título

Puede encontrar más información sobre cómo escribir en el siguiente archivo.

0.4.4 Generando el documento

Para compilar el archivo `.Rmd` y obtener su documento final, simplemente haga clic en el botón `Knit` y seleccione `Knit a PDF` para producir un archivo `.pdf`.

0.5 Ejercicio

- Utilizando Rstudio, crea tu primer documento con Rmarkdown. El documento debe mostrar los metadatos básicos (título, autor y fecha), un título de primer nivel (por ejemplo, 'Información de la sesión') y un recuadro con la información básica de vuestra sesión de R.

Capítulo 1

Introducción

Para cuando llega el momento de analizar los datos, la mayor parte del trabajo ya está hecho. Antes de tener un conjunto de datos se ha tenido que definir el problema de investigación, desarrollar e implementar un plan de muestreo, decidir las escalas de medidas y desarrollar un diseño de investigación. Si se ha hecho bien este trabajo, el análisis de los datos puele ser bastante sencillo.

En la mayoría de las investigaciones de ciencias sociales, el análisis de datos implica tres pasos principales, realizados aproximadamente en este orden:

1. Limpieza y organización de los datos para su análisis (preparación de datos)
2. Describir los datos (estadística descriptiva)
3. Prueba de hipótesis y modelos (estadística inferencial)

La **preparación de datos** implica verificar o registrar los datos, comprobar su exactitud, cargar los datos en el software de análisis, transformar los datos y crear una base de datos adecuada para el análisis que se vaya a realizar.

Las **estadísticas descriptivas** se utilizan para describir las características básicas de los datos en un estudio. Proporcionan resúmenes sencillos sobre la muestra y las medidas. Junto con el análisis gráfico simple, forman la base de prácticamente todos los análisis cuantitativos de datos. Las estadísticas descriptivas simplemente se *describen los datos.*

La **estadística inferencial** investiga preguntas, modelos e hipótesis. En muchos casos, las conclusiones de las estadística inferencial se extienden más allá de los datos inmediatos por sí solos. Por ejemplo, usamos estadísticas inferenciales para tratar de inferir de los datos de la muestra lo que piensa la población total. También utilizamos estadística inferencial para hacer juicios sobre la probabilidad de que una diferencia observada entre grupos sea confiable o si podría haber ocurrido por casualidad en este estudio. Por lo tanto, usamos estadísticas inferenciales para *inferir lo que sucede más allà de nuestros datos.*

En la mayoría de los estudios de investigación, la sección de análisis sigue estas tres fases de análisis. Las descripciones de cómo se prepararon los datos tienden a ser breves y a centrarse solo en los aspectos más exclusivos de su estudio, como las transformaciones de datos específicas que se realizan. Las estadísticas descriptivas que observa en realidad pueden ser voluminosas. En la mayoría de los estudios, las estadísticas descriptivas se seleccionan cuidadosamente y se organizan en tablas de resumen y gráficos que solo muestran la información más relevante o importante. Por lo general, el investigador vincula cada uno de los análisis inferenciales con preguntas o hipótesis de investigación específicas que se plantearon en la introducción, o toma nota de los modelos que se probaron y que surgieron como parte del análisis. En la mayoría de los informes de análisis, es especialmente importante ***"no dejar que los arboles nos impidan ver el bosque"***. Si se presentan demasiados detalles sobre el análisis, es posible que se pierda de vista el problema de la realidad que estemos estudiando.

Todos estos pasos del análisis se pueden realizar habitualmente en todos los paquetes estadísticos. En este curso nos centraremos en el uso de R, por lo que a continuación haremos una breve introducción.

1.1 R como lenguaje de programación

R fue creado en 1992 por Ross Ihaka y Robert Gentleman en la Universidad de Auckland, Nueva Zelanda. R es una implementación gratuita de código abierto del lenguaje de programación estadística **S** creado inicialmente en Bell Labs. En esencia, R es un lenguaje de programación funcional (sus principales funcionalidades giran en torno a la definición y ejecución de funciones). Sin embargo, ahora es compatible, y se usa comúnmente como un lenguaje de programación imperativo (enfocado en instrucciones sobre variables y estructuras de control de programación) y orientado a objetos (que involucra estructuras de objetos complejas).

En términos simples, hoy en día, la programación en R se enfoca principalmente en diseñar una serie de instrucciones para ejecutar una tarea, más comúnmente, cargar y analizar un conjunto de datos (Wickham and Grolemund, 2017).

Como tal, R se puede usar para programar creando secuencias de **instrucciones** que involucren **variables**, que son entidades con nombre que pueden almacenar valores. Ese será el tema principal de esta sesión práctica. Las instrucciones pueden incluir estructuras de flujo de control, como puntos de decisión (*if/else*) y bucles, que serán el tema de la próxima sesión práctica. Las instrucciones también se pueden agrupar en **funciones**, que también veremos en la próxima sesión práctica.

R es **interpretado**, no compilado. Lo que significa que un intérprete de R (si está utilizando R Studio, el intérprete de R simplemente está oculto en el backend y R Studio es el frontend que le permite interactuar con el intérprete) recibe una instrucción que escribe en R, la interpreta y la ejecuta .

Otros lenguajes de programación requieren que su código sea compilado en un ejecutable para ser ejecutado en un ordenador.

1.2 Utilizando RStudio

La interfaz de RStudio se divide en dos secciones principales. En el lado izquierdo, encontrará la *Consola*, así como el editor de secuencias de comandos R, cuando se está editando una secuencia de comandos. La *Consola* en una ventana de entrada/salida en el intérprete de R, donde se pueden escribir instrucciones y se muestra la salida calculada.

Por ejemplo, si escribís en la *Consola*

```
1 + 1
```

el intérprete de R entiende eso como una instrucción para sumar uno más uno, y produce el resultado (dado que los materiales para este módulo se crean en RMarkdown, la salida del cálculo siempre está precedida por '##').

```
## [1] 2
```

Fíjate cómo el valor de salida `2` está precedido por `[1]`, lo que indica que la salida está constituida por un solo elemento. Si la salida está constituida por más de un elemento, como la lista de números a continuación, cada fila de la salida está precedida por el índice del primer elemento de la salida.

```
##  [1]   1   4   9  16  25  36  49  64  81 100 121 144 169 196 225 256 289 324 361
## [20] 400
```

En el lado derecho, encontrarás dos grupos de paneles. En la parte superior derecha, el elemento principal es el panel *Entorno*, que es una representación del estado actual de la memoria del intérprete y, como tal, muestra todas las variables, conjuntos de datos y funciones almacenados. En la parte inferior derecha, encontrarás el panel *Archivos*, que muestra el sistema de archivos (archivos y carpetas del ordenador), así como el panel *Ayuda*, que le muestra las páginas de ayuda cuando sea necesario. Discutiremos los otros paneles más adelante.

1.3 Interpretación de valores

Cuando se escribe un valor en la *Consola*, el intérprete simplemente devuelve el mismo valor. En los ejemplos siguientes, `2` es un valor numérico simple, mientras que `"Valor de cadena"` es un valor textual, que en R se conoce como un valor de *carácter* y en programación también se conoce comúnmente como una *cadena* (abreviatura de *una cadena de caracteres*).

Ejemplo númerico

```
2
```

```
## [1] 2
```

Ejemplo de cadena

```
"String value"
```

```
## [1] "String value"
```

Tened en cuenta que los valores de los caracteres deben comenzar y terminar con comillas simples o dobles (`'o"`), que no forman parte de la información en sí. La Guía de estilo de Tidyverse sugiere usar siempre comillas dobles (`"`), así que serán las que usaremos en este curso.

Todo lo que sigue a un símbolo `#` se considera un *comentario* y el intérprete lo ignora. Cada lenguaje de programación puede usar sus propios símbolos para identificar los comentarios. Por ejemplo cabe destacar la diferencia entre `#` en Markdown que identifica un título de primer nivel, mientras que los comentarios en un fichero de Rmarkdown se identifican entre `<` y `>`.

```
# Este comentario es ignorado por R. Solo sirve para documentar el código.
```

Como se ha mencionado anteriormente, el intérprete también comprende operaciones aritméticas simples sobre valores numéricos.

```
1 + 1
```

```
## [1] 2
```

Además, también hay una gran cantidad de funciones predefinidas, por ejemplo, raíz cuadrada: `sqrt`.

```
sqrt(2)
```

```
## [1] 1.414214
```

Las funciones se recopilan y almacenan en *bibliotecas* (a veces denominadas *paquetes*), que contienen funciones relacionadas. Las bibliotecas pueden variar desde la biblioteca `base`, que incluye la función `sqrt` anterior, hasta la biblioteca `rgdal`, que funciona como un puente hacia las funciones de la GDAL (Biblioteca de abstracción de datos geoespaciales), que es una importante librería en el mundo de los Sistemas de Información Geográfica. Por lo tanto, es mediante la creación de librerías que podemos extender las capacidades de R.

1.4 Variables

Una variable se puede definir usando un **identificador** (por ejemplo, `una_variable`) a la izquierda de un **operador de asignación** `<-`, seguido del *objeto* que se vinculará al identificador, como un **valor** (por ejemplo, `1`) que se

asignará a la derecha. Una vez realizada la asignación, el valor de la variable se puede probar/invocar simplemente especificando el **identificador**.

```
una_variable <- 1
una_variable
```

```
## [1] 1
```

Si escribes `una_variable <- 1` en la *Consola* de RStudio, aparece un nuevo elemento en el panel *Environment* (derecha-arriba), que representa la nueva variable en la memoria. La parte izquierda de la entrada contiene el identificador `una_variable`, y la parte derecha contiene el valor asignado a la variable`una_variable`, es decir, `1`.

No es necesario aportar un valor directamente. La parte derecha de la tarea puede ser una **llamada a una función**. En ese caso, la función se **ejecuta** en la entrada proporcionada y **el resultado se asigna a la variable**.

```
una_variable <- sqrt(4)
una_variable
```

```
## [1] 2
```

Observa cómo, al escribir `una_variable <- sqrt (4)` en la *Consola* de RStudio, el elemento en el panel *Environment* cambia para reflejar el nuevo valor asignado a `una_variable`, que ahora es el resultado de `sqrt (4)`, es decir 2.

En el siguiente ejemplo, se crea otra variable llamada `otra_variable` y se suma a `una_variable`, guardando el resultado en `suma_de_dos_variables`. La raíz cuadrada de esa suma se almacena en la variable `raiz_cuadrada_de_suma`.

```
otra_variable <- 4
otra_variable
```

```
## [1] 4
```

```
suma_de_dos_variables <- una_variable + otra_variable

raiz_cuadrada_de_suma <- sqrt(suma_de_dos_variables)
raiz_cuadrada_de_suma
```

```
## [1] 2.44949
```

1.5 Tipos basicos

1.5.1 Números

El tipo *numeric* representa números en general (tanto enteros como reales), pero R es capaz de distinguirlos utilizando funciones.

```
un_numero <- 1.41
is.numeric(un_numero)
```

```
## [1] TRUE
```

```
is.integer(un_numero)
```

```
## [1] FALSE
```

```
is.double(un_numero) # i.e., es real
```

```
## [1] TRUE
```

Operadores numéricos básicos.

Operador	Significado	Ejemplo	Resultado
+	Suma	`5+2`	7
-	Resta	`5-2`	3
*	Multiplicación	`5*2`	10
/	División	`5/2`	2.5
%/%	Div. de enteros	`5%/%2`	2
%%	Módulo	`5%%2`	1
^	Potencia	`5^2`	25

Algunas funciones predefinidas en R son:

```
abs(-2) # Valor absoluto
```

```
## [1] 2
```

```
ceiling(3.475) # Redondeo al alza
```

```
## [1] 4
```

```
floor(3.475) # Redondeo a la baja
```

```
## [1] 3
```

```
trunc(5.99) # Truncar
```

```
## [1] 5
```

```
log10(100) # Logaritmo en base 10
```

```
## [1] 2
```

```
log(exp(2)) # Logaritmo natural y exponencial
```

```
## [1] 2
```

Como en cualquier otro entorno, podéis utilizar paréntesis simples para

especificar el orden de ejecución. Si no se especifica, el orden predeterminado es: potencia, multiplicación y división, suma y resta al final.

```
un_numero <- 1
(un_numero + 2) * 3
```

```
## [1] 9
```

```
un_numero + (2 * 3)
```

```
## [1] 7
```

```
un_numero + 2 * 3
```

```
## [1] 7
```

R devuelve el resultado `NaN` (*Not a number*)cuando el resultado de una operación no es un número.

```
0/0
```

```
## [1] NaN
```

```
is.nan(0/0)
```

```
## [1] TRUE
```

No hay que confundir `NaN`con `NA` (*No Available*), que sirve para identificar datos faltantes.

1.5.2 Lógicos o booleanos

El tipo *lógico* codifica dos valores dicotómicos: Verdadero y Falso.

```
valor_logico <- TRUE
is.logical(valor_logico)
```

```
## [1] TRUE
```

```
isTRUE(valor_logico)
```

```
## [1] TRUE
```

```
as.logical(0) # se pueden convertir 1/0 a TRUE/FALSE
```

```
## [1] FALSE
```

Operadores lógicos básicos

Operador	Significado	Ejemplo	Resultado
==	Igual	`5==2`	FALSE
!=	No igual	`5!=2`	TRUE
>	Mayor que	`5>2`	TRUE

Operador	Significado	Ejemplo	Resultado
<	Menor que	`5<2`	FALSE
>=	Mayor o igual	`5>=2`	TRUE
<=	Menor o igual	`5<=2`	FALSE
!	No	`!TRUE`	FALSE
&	Y	`TRUE & FALSE`	FALSE
|	O	`TRUE | FALSE`	TRUE

1.5.3 Cádenas de caracteres

El tipo *character* representa objetos de texto, incluidos caracteres individuales y cadenas de caracteres (es decir, objetos de texto de más de un carácter, comúnmente denominados simplemente *cadenas* o *strings* en informática).

```
una_cadena <- "¡Hola!"
is.character(una_cadena)
```

```
## [1] TRUE
```

```
is.numeric(una_cadena)
```

```
## [1] FALSE
```

```
as.character(2) # Conversión de número a cadena (en inglés, hacer un 'cast')
```

```
## [1] "2"
```

```
as.numeric("2")
```

```
## [1] 2
```

```
as.numeric("¡Hasta luego!")
```

```
## Warning: NAs introduced by coercion
```

```
## [1] NA
```

1.6 Estructuras de datos

Estos tipos de datos más básicos se gestiónan habitualmente dentro de estructuras de datos más complejas. R tiene muchas otras estructuras de datos pero las más básicas comprenden:

- Vectores
- Listas
- Matrices
- *data frames* (se usa el término en inglés por comodidad).
- Factores

1.6.1 Vectores

1.6.2 Listas

1.6.3 Matrices

1.6.4 Data frames

1.6.5 Factores

1.7 Funciones

Hasta este punto ya habéis visto y usado varias funciones. Por ejemplo, la función `c()` se puede usar para **c**ombinar objetos en un vector. En general, todas las llamadas a funciones tienen el mismo aspecto: el nombre de una función siempre va seguido de paréntesis. A veces, los paréntesis incluyen argumentos como sucede también en el siguiente ejemplo:

```
# Crea el vector `z` a partir de la función seq (secuenciar).
z <- seq(from = 1, to = 5, by = 1)
```

En este ejemplo usamos una función llamada `seq()` para crear una secuencia que progresa por unidades, desde 1 a 5 (probad de modificar los parámetros para obtener otras secuencias distintas).

Si no estáis seguros de qué argumentos acepta una función, siempre se puede consultar la documentación de dicha función (todas las librerías *oficiales* suelen tener una buena documentación). Por ejemlo, supongamos que no estamos seguros de cómo funcionan los argumentos necesarios para `seq()`. Podemos escribir `?seq` en la consola y, al ejecutar este comando, la página de documentación para esa función aparece en el panel inferior derecho de RStudio. En la sección *Argumentos* está toda la información que buscamos. En la parte inferior de casi todas las páginas de ayuda, suele haber ejemplos sobre cómo utilizar las funciones correspondientes.

1.8 Tidyverse

Como se ha mencionado anteriormente, las librerias o *paquetes* son colecciones de funciones y/o conjuntos de datos. Las librerias se pueden instalar en R usando la función `install.packages()` o usando el menú `Herramientas> Instalar Librerias ...`en RStudio. Algunas librerias de R están relacionadas entre sí o forman parte de flujos de trabajo mayores. A dia de hoy la meta-librería Tidyverse contiene algunas de las librerías más utilizadas en el análisis de datos (Wickham et al., 2019). Solo por mencionar algunas:

- `ggplot2` para crear gráficos.
- `dplyr` para manipular datos (filtrar, seleccionar, agregar, sumarizar, etc).

- `tidyr` para organizar los datos de un modo que las otras librerias del **Tidyverse** puedan trabajar mejor.
- `readr` para importar tablas de datos a partir de formatos habituales (csv, tsv, o fwf). Permite minimizar la introducción de errores en la importación.
- `purrr` se utiliza para facilitar la automatización de tareas en R, mientras se escribe menos código.
- `stringr` facilita la manipulación de cadenas de texto (unir, separar, filtrar palabras en un documento, etc).
- `forcats` para trabajar con factores.
- `tibble`…

Se puede cargar una librería usando la función `library()`, como se muestra a continuación (tened en cuenta que el nombre de la biblioteca no está entrecomillado). Una vez que una librería está instalada en un ordenador, no es necesario que la instale nuevamente, pero cada secuencia de comandos o *script* debe cargar todas las librerías que utiliza. Una vez que se carga una librería, se pueden utilizar todas sus funciones.

```
library(tidyverse)
```

1.8.1 stringr

El siguiente código presenta un mínimo ejemplo del uso de las funciones de la librería `stringr`.

```
str_length("Tarragona")
```

```
## [1] 9
```

```
str_detect("Tarragona", "a")
```

```
## [1] TRUE
```

```
str_replace_all("Tarragona", "r", "R")
```

```
## [1] "TaRRagona"
```

1.8.2 El operador `pipe`

El operador `pipe` (tuberia) es útil para reducir el número de asignaciones en operaciones más complejas. Un *pipe* (`%>%`) toma el resultado de una función y lo pasa a la siguiente función como **primer argumento**, de este modo ya no hace falta repetir el resultado de la primera función en el código.

El siguiente código muestra un ejemplo sencillo. El número `2` se toma como entrada para el primer `pipe` que lo pasa como primer argumento a la función `sqrt`. El valor de salida 1.4142136 se toma como entrada para el segundo `pipe`, que lo pasa como primer argumento a la función `trunc`. Finalmente se devuelve la salida final “1”.

```
2 %>%
  sqrt() %>%
  trunc()
```

```
## [1] 1
```

```
sqrt(2) %>%
 round(digits = 2)
```

El primer paso de una secuencia de *pipes* puede ser un valor, una variable o una función que incluya argumentos. El siguiente código muestra una serie de ejemplos de diferentes formas de lograr el mismo resultado. Los ejemplos usan la función `round`, que también permite un segundo argumento `digits = 2`. Tened en cuenta que, cuando se utiliza el operador `%>%`, solo se proporciona el segundo argumento nominalmente a la función `round`, es decir, `round(digits = 2)`

```
# R básico, sin utilizar '%>%', pero en varios pasos
variable_temporal_a <- 2
variable_temporal_b <- sqrt(variable_temporal_a)
round(variable_temporal_b, digits = 2)

# R básico, sin utilizar '%>%', pero sin asignaciones
round(sqrt(2), digits = 2)

# Pipe a partir de un valor
2 %>%
  sqrt() %>%
  round(digits = 2)

# Pipe a partir de una variable
el_numero_dos <- 2
el_numero_dos %>%
  sqrt() %>%
  round(digits = 2)

# Pipe empezando por una función
sqrt(2) %>%
 round(digits = 2)
```

Una operación compleja creada mediante el uso de `%>%` se puede usar en el lado derecho de una asignación `<-`, de modo que se guarda resultado de toda la operación de la derecha.

```
raiz_cuadrada_de_dos <- 2 %>%
  sqrt() %>%
  round(digits = 2)
```

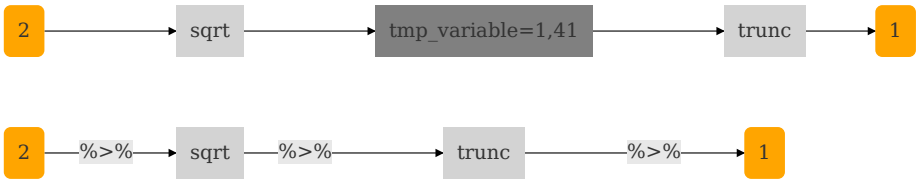


Figura 1.1: Ejecución del código sin o con `%>%`. Sin *pipes* los resultados intermedios tienen que guardarse en una variable en memoria.

1.9 Ejercicios

Crea un *script* para responder a cada pregunta. En las preguntas de la 4 a la 8 es posible que tengas que consultar la referencia de la librería `stringr` (stringr.tidyverse.org/reference) o a otras funciones de `R base`.

1. Escribe un fragmento de código utilizando el operador `%>%` que haga lo siguiente:
   - tomar como entrada el número `1632`
   - calcular el logaritmo en base 10
   - redondear el número a la baja
   - truncar la parte entera y
   - verificar si se trata de un número entero.
2. Escribe un fragmento de código utilizando el operador `%>%` que haga lo siguiente:
   - tomar como entrada el número `1632`
   - calcular la raíz cuadrada,
   - redondear al alza y quedarse con la parte entera, y
   - verificar que se trata de un entero.
3. Escribe un fragmento de código utilizando el operador `%>%` que haga lo siguiente:
   - tomar como entrada la cadena `1632`
   - convertirla a número
   - comprobar si el resultado es o no un número.
4. Escribe un fragmento de código utilizando el operador `%>%` que haga lo siguiente:
   - tomar como entrada la cadena `"-16.32"`
   - transformarla en un número
   - calcular el valor absoluto y truncarlo
   - comprobar si el resultado es o no un número.
5. Escribe un fragmento de código utilizando el operador `%>%` y la librería and the `stringr` que haga lo siguiente:
   - tomar la cadena `"Siempre r que r"` como entrada
   - transformar la cadena a mayúsculas.
6. Escribe un fragmento de código utilizando el operador `%>%` y la librería and the `stringr` que haga lo siguiente:

- tomar la cadena `"Siempre r que r"` como entrada
- truncarla para dejar solamente 'Siempre R'.

Capítulo 2

Estadística descriptiva

La estadística descriptiva es una rama de la estadística cuyo objetivo es resumir, describir y presentar una serie de valores o un conjunto de datos. Estas estadísticas pueden ser realmente útiles al analizar largas series de datos en las que resulte difícil reconocer algún patrón. En éste capítulo utilizaremos una muestra de datos sobre salarios (en euros) de una *población* de 100 trabajadores:

2186, *1218*, *1682*, *1816*, *1702*, *1447*, *2256*, *1453*, *2509*, *1469*, *2152*, *2643*, *806*, *1361*, *1433*, *1818*, *1358*, *172*, *280*, *2160*, *1347*, *609*, *1414*, *2107*, *2448*, *1285*, *1371*, *618*, *1730*, *1180*, *1728*, *1852*, *2018*, *1196*, *1752*, *642*, *1108*, *1074*, *293*, *1518*, *1603*, *1320*, *1879*, *1137*, *816*, *1716*, *1094*, *2222*, *1284*, *1828*, *1661*, *1108*, *2288*, *1821*, *1545*, *1638*, *1840*, *1545*, *4*, *1642*, *1316*, *1593*, *1791*, *2200*, *1136*, *2151*, *1668*, *2019*, *1960*, *1860*, *978*, *1455*, *1812*, *1023*, *1229*, *1790*, *1884*, *1732*, *1057*, *950*, *2256*, *1629*, *1544*, *1440*, *903*, *1806*, *1391*, *1409*, *1967*, *1911*, *2196*, *1262*, *1825*, *2196*, *945*, *1070*, *934*, *770*, *1540* y *1827*

De un primer vistazo es difícil (por no decir imposible) que podamos comprender los datos y tener una visión clara de los salarios de este grupo de personas. Las estadísticas descriptivas permiten resumir y así tener una mejor visión general de los datos. Por supuesto, al resumir los datos a través de una o varias medidas, inevitablemente se perderá parte de la información. Sin embargo, en muchos casos es mejor perder algo de información pero, a cambio, obtener una visión general. Podriamos decir que se trata de *ganar perspectiva*.

La estadística descriptiva es a menudo el primer paso y una parte importante en cualquier análisis estadístico. Permite comprobar la calidad de los datos detectando posibles valores atípicos (*outliers*), es decir, datos que parecen ser significativamente distintos del resto. También se puede utilizar estadística descriptiva para detectar errores de recopilación o codificación, determinar si están bien presentados, entre otras posibles aplicaciones.

Podemos distinguir dos típos básicos de estadísticos para describir un conjunto

de datos: de centralidad y de dispersión. Habitualmente, ambos tipos de medidas se utilizan juntos para resumir los datos de la forma más concisa.

2.1 Tendencia central

Las medidas de tendencia central permiten ver "dónde" se ubican los datos, alrededor de qué valores. En otras palabras, las medidas de ubicación permiten comprender cuál es la tendencia central o la "posición" de los datos en su conjunto. Entre las estadísticas más habituales de este tipo podemos distinguir:

- Mínimo y máximo
- Media
- Mediana
- Primer y tercer cuartil
- Moda

2.1.1 Mínimo y máximo

Mínimo (*min*) y máximo (*max*) son simplemente los valores más bajo y más alto de la muestra. Dada una muestra de 6 de estos salarios:

2186, *1218*, *1682*, *1816*, *1702* y *1447*

El mínimo es 1217.7 euros y el máximo es 2185.5 euros. Estas dos estadísticas básicas dan una idea clara sobre el los extremos de la muestra y su cálculo con R es bastante sencillo:

```
min(salarios_sel)
```

```
## [1] 1217.7
```

```
max(salarios_sel)
```

```
## [1] 2185.5
```

2.1.2 Media

La media o promedio, es probablemente la estadística más habitual. Da una idea de cuál es el valor medio, es decir, el valor central de los datos o, en otras palabras, su centro de gravedad. La media se encuentra sumando todos los valores y dividiendo esta suma por el número de observaciones (n):

$$Media = \overline{x} = \frac{\text{suma de todos los valores}}{\text{número de valores}} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

Dada nuestra muestra de 6 salarios presentada anteriormente, la media es:

$$\overline{x} = \frac{2186 + 1218 + 1682 + 1816 + 1702 + 1447}{6} = 1675,033$$

En conclusión, el tamaño medio de la nuestra muestra es 1675.03 euros (redondeado a 2 decimales). El cálculo con R también resulta bastante sencillo:

```
mean(salarios_sel)
```

```
## [1] 1675.033
```

2.1.3 Mediana

La mediana es otra medida de centralidad. La interpretación de la mediana es que hay tantas observaciones por debajo como por encima de la mediana. En otras palabras, el 50% de las observaciones se encuentran por debajo de la mediana y el 50% de las observaciones están por encima de la mediana.

La forma más fácil de calcular la mediana es primero ordenar los datos de menor a mayor (es decir, en orden ascendente) y luego tomar el punto medio como la mediana. A partir de los valores ordenados, para un número impar de observaciones, el punto medio es fácil de encontrar: es el valor con tantas observaciones abajo como arriba. Aún a partir de los valores ordenados, para un número par de observaciones, el punto medio está exactamente entre los dos valores medios. Formalmente, después de ordenar, la mediana es:

- si n (número de observaciones) es impar:

$$mediana(x) = x_{\frac{n+1}{2}}$$

- si n es par:

$$mediana(x) = \frac{1}{2}(x_{\frac{n}{2}} + x_{\frac{n}{2}+1})$$

donde el subíndice de x denota la numeración de los datos ordenados. El cálculo en R es de nuevo bastante sencillo:

```
median(salarios_sel)
```

```
## [1] 1691.85
```

2.1.4 Primer y tercer cuartil

El primer y tercer cuartil son similares a la mediana en el sentido de que también dividen las observaciones en dos partes, solo que estas partes no son iguales. Recordad que la mediana divide los datos en dos partes iguales (con el 50% de las observaciones por debajo y el 50% por encima de la mediana). El primer cuartil divide las observaciones de modo que haya un 25% de las observaciones *debajo de* este punto y un 75% **por encima** del primer cuartil. El tercer cuartil se calcula del mismo modo pero representa el valor con el 75% de las observaciones por debajo y el 25% de las observaciones por encima. Existen varios métodos para calcular el primer y tercer cuartil, pero por ejemplo se puede calcular siguiendo los siguientes pasos:

1. Ordenar los datos en orden ascendente

2. Calcular $0.25 \cdot n$ y $0.75 \cdot n$ (es decir, 0.25 y 0.75 veces el número de observaciones)
3. Redondear estos dos números al siguiente número entero `ceil`

Los pasos son los mismos para un número par e impar de observaciones. A continuación se muestra un ejemplo para el cálculo de estos valores sobre los salarios de nueve trabajadores:

```
quantile(salarios_sel)
```

```
##       0%      25%      50%      75%     100%
## 1217.700 1505.575 1691.850 1787.825 2185.500
```

Dado que obtenemos un vector de resultados, podemos seleccionar por posición para obtener el que necesitemos en cada caso.

2.1.5 Moda

La moda de una serie es el valor que aparece con mayor frecuencia. En otras palabras, es el valor que tiene el mayor número de ocurrencias. Dados los siguientes salarios:

1700, *1680*, *1710*, *1700*, *1820*, *1650*, *1700*, *1890* y *1670*

La moda es 1700 porque es el valor más común con 3 apariciones. Todos los demás valores aparecen solo una vez. En conclusión, la mayoría de los trabajadores de esta muestra perciben exactamente 1700 euros al mes.

Hay que tener en cuenta que es posible que una serie no tenga moda (p. Ej., *4*, *7*, *2* y *10*) o más de una moda (p. Ej., *4*, *2*, *2*, *8*, *11* y *11*). Los datos con dos modas a menudo se denominan bimodales y los datos con más de dos modos a menudo se denominan multimodales, a diferencia de las series con una moda, que se denominan unimodales.

A diferencia de las anteriores estadísticas (mínimo, máximo, media, mediana, primer y tercer cuartil) que solo se pueden calcular para variables cuantitativas, **la moda se puede calcular para variables cuantitativas y cualitativas**. Atendiendo al tipo de empleo que desarrollan los 9 trabajadores presentados anteriormente:

ingeniero, *ingeniero*, *ingeniero*, *ingeniero*, *asistente*, *asistente*, *asistente*, *ingeniero* y *camarero*

La moda es "ingeniero", por lo que la mayoría de los trabajadores de esta muestra son de uso ingenieros.

2.2 Dispersión

2.2.1 Rango

El rango es la diferencia entre el máximo y el mínimo:

$$rango = max - min$$

Dada nuestra muestra de salarios:

2186, *1218*, *1682*, *1816*, *1702* y *1447*

El rango es $2185.5 - 1217.7 = 967.8$ euros. El rango es muy sencillo de calcular y en algunos casos puede dar una buena idea de lo que podemos esperar de un conjunto de datos. En R se puede utilizar la función `range`

```
range(salarios_sel)
```

```
## [1] 1217.7 2185.5
```

No obstante, esta métrica no da ninguna información de la distribución interna del resto de medidas.

2.2.2 Desviación estándard

La desviación estándar es la medida de dispersión más común en estadística. Si tenemos que presentar una estadística que resuma la distribución de los datos, suele ser la desviación estándar. Como sugiere su nombre, la desviación estándar indica cuál es la desviación *normal* de los datos. De hecho, calcula la desviación promedio de la **media**. Cuanto mayor sea la desviación estándar, más dispersos estarán los datos. Por el contrario, cuanto menor es la desviación estándar, más se centran los datos alrededor de la media.

Existen dos fórmulas para calcular la desviación estándard en función de si nos enfrentamos a una muestra o a una población. Una población incluye a todos los miembros de un grupo específico, mientras que una muestra contiene algunas observaciones extraídas de la población, es decir, una parte o un subconjunto de la población. Por ejemplo, la población puede ser "**todas** personas que viven en España" y la muestra puede ser "**algunas** personas que viven en España".

La desviación estándar para una población, a partir de ahora σ, es:

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2}$$

Como se puede ver en la fórmula, la desviación estándar es en realidad la desviación promedio de los datos de su media μ. Hay calcular primero el cuadrado de la diferencia entre las observaciones y la media para evitar que las diferencias negativas se compensen con diferencias positivas.

Por ejemplo, imagine una población de solo 3 trabajadores:

770, *1540* y *1827*

La media es 1379 (redondeado a 1 decimal). La desviación estándar es entonces:

$$\sigma = \sqrt{\frac{1}{3}[(770.4 - 1379)^2 + (1540 - 1379)^2 + (1827 - 1379)^2]}$$

$$\sigma = 546,2$$

Por lo tanto, la desviación estándar para los salarios de estos trabajadores es de 546,2 euros. Esto significa que los salarios de los trabajadores de esta población se desvían de la media en 546,2 euros.

...

2.2.3 Varianza

2.2.4 Coeficiente de variación

2.3 Correlación

Las correlaciones entre variables juegan un papel importante en un **análisis descriptivo**. Una correlación mide la **relación entre dos variables**, es decir, cómo están vinculadas entre sí. En este sentido, una correlación permite saber si dos variables evolucionan en la misma dirección, en sentido contrario y si son independientes.

En este capítulo, se muestra cómo calcular **coeficientes de correlación**, cómo realizar **pruebas de correlación** y cómo **visualizar** correlaciones entre variables usando R.

La correlación generalmente se calcula en dos variables *cuantitativas*, pero también se puede calcular en dos variables *cualitativas ordinales* (ver prueba de independencia de chi-cuadrado).

2.3.1 Datos

Usaremos el conjunto de datos `mtcars`. Este conjunto de datos viene cargado por defecto en R, de modo que se utiliza en numerosas demostraciones.

```
# mostrar las primeras cinco filas
head(mtcars, 5)
```

```
##                   mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
```

Las variables `vs` yam son variables categóricas, por lo que se eliminan para este artículo:

```
# Eliminar las variables vs y am
library(tidyverse)
dat <- mtcars %>%
select(-vs, -am)

# mostrar las primeras cinco filas
head(dat, 5)
```

```
##                    mpg cyl disp  hp drat    wt  qsec gear carb
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46    4    4
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02    4    4
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61    4    1
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44    3    1
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02    3    2
```

2.3.2 Coeficiente de correlación

La correlación entre 2 variables se calcula con la función `cor()`. Supón que queremos calcular la correlación entre caballos de potencia (`hp`) y millas por galón (`mpg`):

```
# Correlación de Pearson entre 2 variables
cor(dat$hp, dat$mpg)
```

```
## [1] -0.7761684
```

Hay que fijarse en que la correlación entre las variables X e Y es igual a la correlación entre las variables Y y X, por lo que el orden de las variables en la función `cor()` no importa.

La función `cor()` calcula por defecto la correlación de Pearson, por lo que si se quiere calcular la correlación por otro método, se puede agregar el argumento `method =" spearman "` a la función `cor()`:

```
# Correlación de Sperman entre 2 variables
cor(dat$hp, dat$mpg,
    method = "spearman")
```

```
## [1] -0.8946646
```

Hay varios métodos de correlación (se puede consultar la ayuda de la la función para saber más `?cor`):

- **Pearson** se usa a menudo para variables *cuantitativas continuas* que tienen una relación lineal.
- **Spearman** (es similar a Pearson pero se basa en los valores ordenados para cada variable en lugar de en los datos brutos) se usa a menudo para

evaluar relaciones que involucran variables cualitativas ordinales en las que la relación sea parcialmente lineal.
- **Kendall** se calcula a partir del número de pares concordantes y discordantes, se utiliza a menudo para variables ordinales cualitativas.

La función `cor ()` también permite calcular correlaciones para varios pares de variables a la vez:

```
# Correlaciones entre todas las variables
round(cor(dat),
      digits = 2 # redondeo a dos decimales
      )
```

```
##        mpg   cyl  disp    hp  drat    wt  qsec  gear  carb
## mpg   1.00 -0.85 -0.85 -0.78  0.68 -0.87  0.42  0.48 -0.55
## cyl  -0.85  1.00  0.90  0.83 -0.70  0.78 -0.59 -0.49  0.53
## disp -0.85  0.90  1.00  0.79 -0.71  0.89 -0.43 -0.56  0.39
## hp   -0.78  0.83  0.79  1.00 -0.45  0.66 -0.71 -0.13  0.75
## drat  0.68 -0.70 -0.71 -0.45  1.00 -0.71  0.09  0.70 -0.09
## wt   -0.87  0.78  0.89  0.66 -0.71  1.00 -0.17 -0.58  0.43
## qsec  0.42 -0.59 -0.43 -0.71  0.09 -0.17  1.00 -0.21 -0.66
## gear  0.48 -0.49 -0.56 -0.13  0.70 -0.58 -0.21  1.00  0.27
## carb -0.55  0.53  0.39  0.75 -0.09  0.43 -0.66  0.27  1.00
```

La correlación varía de **-1 a 1**, de modo que el signo nos indica la dirección de la relación (aumentan a la vez o son opuestas) y el valor nos indica la fuerza de la relación (más fuerte cuanto más alejado de 0).

Una **correlación negativa** implica que las dos variables consideradas varían en **direcciones opuestas**, es decir, si una variable aumenta la otra disminuye y viceversa. Por otro lado, una **correlación positiva** implica que las dos variables consideradas varían en la **misma dirección**, es decir, si una variable aumenta, la otra aumenta y si una disminuye, la otra también disminuye.

En cuanto a la fuerza de la relación: cuanto **más extremo** es el coeficiente de correlación (cuanto más cerca de -1 o 1), **más fuerte es la relación**. Esto también significa que una **correlación cercana a 0** indica que las dos variables son **independientes**, es decir, a medida que una variable aumenta, no hay tendencia en la otra variable a disminuir o aumentar.

Por ejemplo, la correlación de Pearson entre caballos de potencia (`hp`) y millas por galón (`mpg`) encontrada es -0.78, lo que significa que las 2 variables varían en dirección opuesta. Esto tiene sentido, los automoviles con más caballos de potencia suelen a consumir más combustible (hacen menos millas con el mismo combustible que los automóviles más potentes). Por el contrario, de la matriz de correlación vemos que la correlación entre millas por galón (`mpg`) y el tiempo para conducir un cuarto de milla (`qsec`) es 0.42, lo que significa que los automóviles rápidos (con un menor `qsec`) tienden a tener un peor rendimiento por galón (bajo `mpg`). De nuevo, esto tiene sentido, ya que los coches rápidos tienden a

consumir más combustible.

2.3.3 Test de correlación

Volver una vez leída la sección sobre test de hipótesis.

Hay que tener en cuenta que el valor p se basa en el coeficiente de correlación y en el tamaño de la muestra. Cuanto mayor sea el tamaño de la muestra y más extrema será la correlación (más cercana a -1 o 1). Con un tamaño de muestra pequeño, es posible obtener una correlación *relativamente* grande en la muestra (según el coeficiente de correlación), pero aún así encontrar una correlación no significativamente diferente de 0 en la población (según la prueba de correlación). Por este motivo, se recomienda realizar siempre un test de correlación antes de interpretar un coeficiente de correlación para evitar conclusiones erróneas.

A diferencia de una matriz de correlación que indica los coeficientes de correlación entre algunos pares de variables en la muestra, se utiliza un test de correlación para probar si la correlación (ρ) entre dos variables es significativamente diferente de 0 o no en la *población*.

En realidad, un coeficiente de correlación diferente de 0 en la muestra no significa que la correlación sea **significativamente** diferente de 0 en la población. Esto debe probarse con un **test de hipótesis**.

Las hipótesis (nula y alternativa) para el test de correlación son las siguientes:

- H_0: $\rho = 0$ (si no existe una relación lineal entre las dos variables)
- H_1: $\rho \neq 0$ (si existe una relación lineal entre las dos variables)

A través de esta prueba de correlación, lo que realmente estamos probando es si:

- La muestra contiene evidencia suficiente para rechazar la hipótesis nula y concluir que el coeficiente de correlación no es igual a 0, por lo que la relación existe en la población.
- La muestra no contiene suficiente evidencia de que el coeficiente de correlación no sea igual a 0, por lo que en este caso no rechazamos la hipótesis nula de no-relación entre las variables de la población.

Supongamos que queremos probar si el ratio del eje trasero (`drat`) está correlacionado con el tiempo necesario para conducir 1/4 de milla (`qsec`):

```
# Test de correlación de Pearson
test <- cor.test(dat$drat, dat$qsec)
test
```

```
##
##  Pearson's product-moment correlation
##
## data:  dat$drat and dat$qsec
## t = 0.50164, df = 30, p-value = 0.6196
```

```
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.265947  0.426340
## sample estimates:
##        cor
## 0.09120476
```

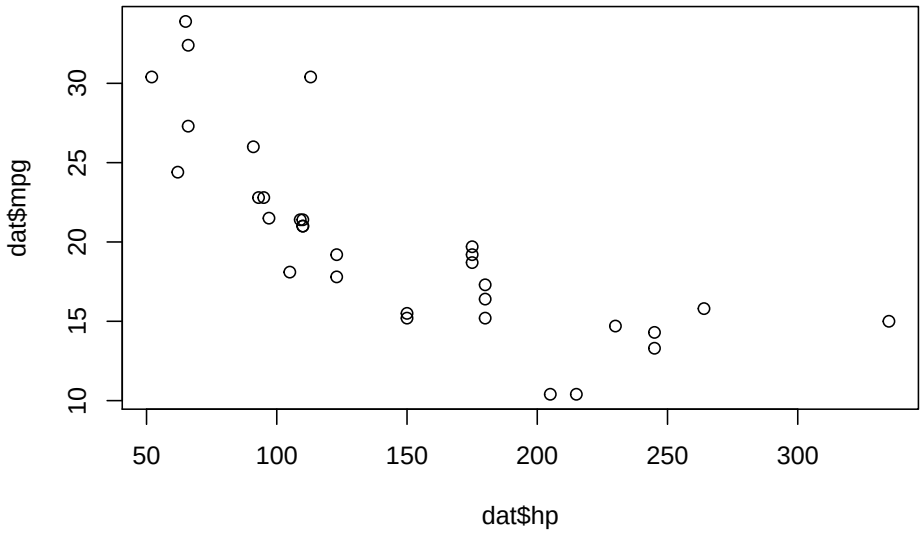
El p-valor de la prueba de correlación entre estas 2 variables es 0.62. Al nivel de significancia del 5%, no se rechaza la hipótesis nula de no correlación. Por lo tanto, concluimos que no rechazamos la hipótesis de que no existe una relación lineal entre las 2 variables. [1]

Esta prueba demuestra que incluso si el coeficiente de correlación es diferente de 0 (la correlación es 0.09 en la muestra), en realidad no es significativamente diferente de 0 en la población.

2.3.4 Visualizando correlaciones

Una buena forma de visualizar una correlación entre 2 variables es mediante un diagrama de dispersión. Por ejemplo:

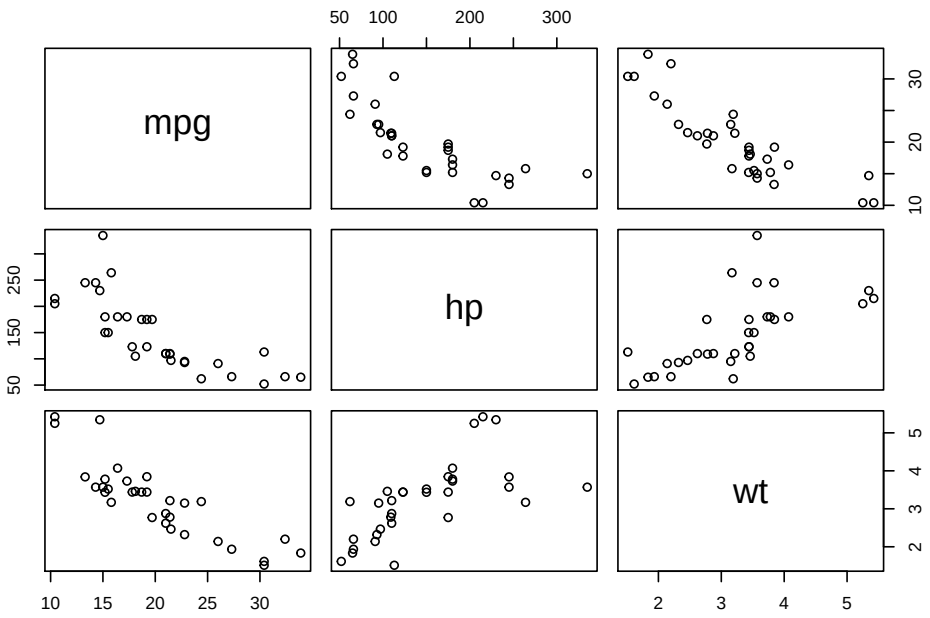
```
# Diagrama de dispersión con R base
plot(dat$hp, dat$mpg)
```



Para visualizar la relación entre más de 2 variablesse puede usar la función `pair()`. En este caso limitamos el ejemplo a tres variables:

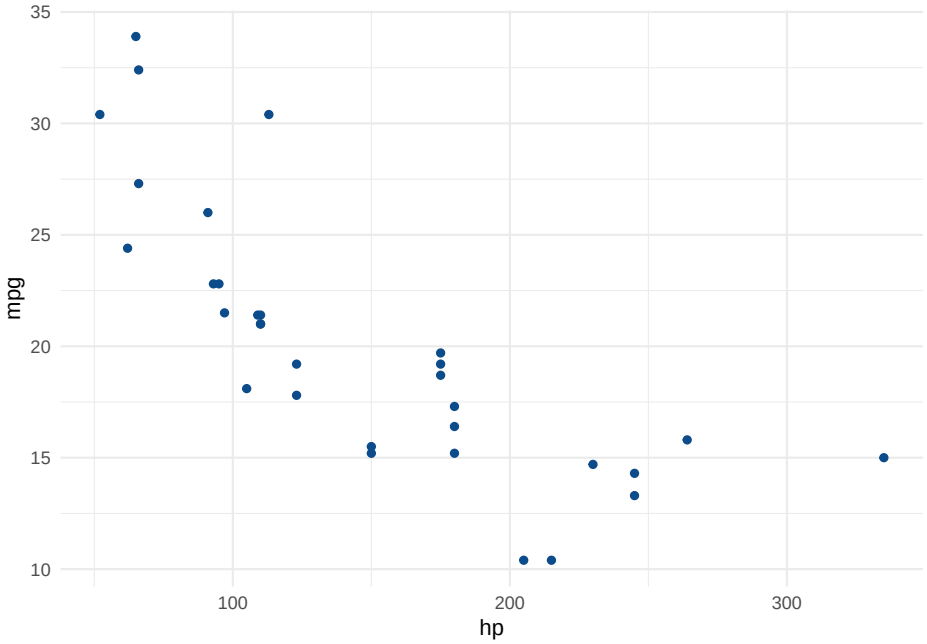
[1] Es importante recordar que probamos una relación *lineal* entre las dos variables ya que usamos la correlación de Pearson. Puede darse el caso de que exista una relación entre las dos variables en la población, pero esta relación puede no ser lineal.

```r
# Multiples diagramas de dispersión
pairs(dat[, c(1, 4, 6)])
```



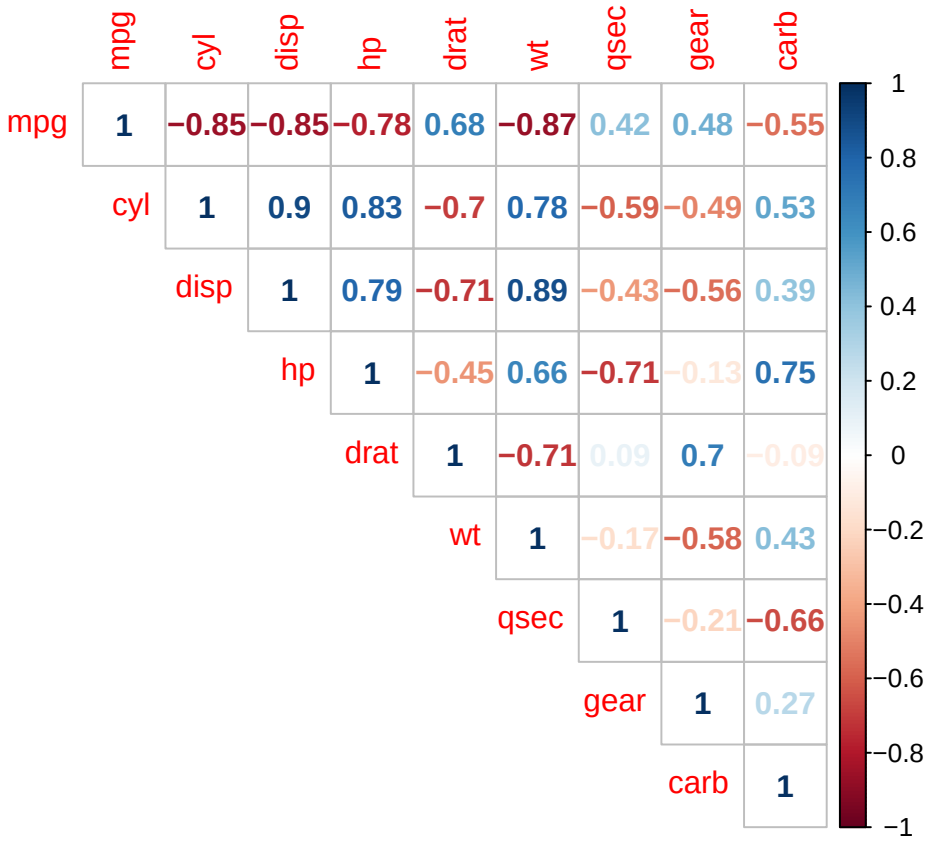
Por otra parte, existen numerosas librerías de R que permiten generar este tipo de gráficos con distintas opciones. Por ejemplo con `ggplot2`:

```r
# Diagrama de dispersión con ggplot2
library(ggplot2)

ggplot(dat) +
 aes(x = hp, y = mpg) +
 geom_point(colour = "#0c4c8a") +
 theme_minimal()
```



O representaciones más modernas como con la librería `corrplot`:

```
# improved correlation matrix
library(corrplot)

corrplot(cor(dat),
         method = "number",
         type = "upper" # show only upper side
         )
```



2.4 Ejercicios

1. Estableced en clase un debate sobre variables de carácter territorial que consideréis que pueden estar correlacionadas y en qué medida.

Capítulo 3

Estadística inferencial

La **estadísticas descriptiva** es la rama de las estadística que tiene como objetivo **describir y resumir un conjunto de datos** de la mejor manera posible, es decir, con la menor pérdida de información posible. Con la estadística descriptiva no hay incertidumbre, porque describimos solo el grupo de observaciones en las que decidimos trabajar y no se intenta generalizar las características observadas o estudiar un grupo más grande a partir de un conjunto de datos limitado.

Por otro lado, **La estadística inferencial** es la rama de la estadística que utiliza una muestra aleatoria de datos tomados de una población para hacer inferencias, es decir, **sacar conclusiones sobre la *población* de interés**. En otras palabras, la información de la muestra se utiliza para hacer generalizaciones sobre el *parámetro* de interés en la población.

Las dos herramientas más importantes utilizadas en estadística inferencial son los test de hipótesis y los intervalos de confianza.

3.1 Distribuciones de probabilidad

Una distribución de probabilidad es una función que describe la probabilidad de obtener los posibles valores que puede asumir una variable aleatoria. En otras palabras, los valores de la variable varían según la distribución de probabilidad subyacente.

Supongamos que seleccionamos una muestra aleatoria de personas y medimos la altura de los sujetos. A medida que vamos midiendo las alturas, podemos crear una distribución de alturas. Este tipo de distribución es útil cuando necesita saber qué resultados son más probables, la dispersión de los valores potenciales y la probabilidad de resultados diferentes. Por lo tanto se puede utilizar distribuciones de probabilidad para realizar inferencias.

3.1.1 Funciones de R para distribuciones de probabilidad

R tiene varias funciones para trabajar con cada distribución de probabilidad. Hay un nombre de raíz, por ejemplo, el nombre de raíz para la distribución normal es `norm`. Esta raíz tiene como prefijo una de las letras:

- `p` para "probabilidad", la función de distribución acumulativa.
- `q` para "cuantil", el inverso de la función de distribución acumulativa.
- `d` para "densidad", la función de densidad (p. f. o p. d. f.)
- `r` para "aleatorio", una variable aleatoria que tiene la distribución especificada

Para la distribución normal, estas funciones son `pnorm`, `qnorm`, `dnorm` y `rnorm`, mientras que para la distribución binomial, estas funciones son `pbinom`, `qbinom`, `dbinom` y `rbinom`.

Para una distribución continua (como la normal), las funciones más útiles para resolver problemas que involucran cálculos de probabilidad son las funciones `p` y `q`, porque la densidad por la función `d` solo se puede usar para calcular probabilidades a través de integrales.

Para una distribución discreta (como la binomial), la función `d` calcula la densidad (p. F.), Que en este caso es una probabilidad

$f(x) = P(X = x)$

y por lo tanto es útil para calcular probabilidades.

R tiene funciones para manejar muchas distribuciones de probabilidad. La siguiente tabla proporciona los nombres de las funciones para cada distribución y un enlace a la documentación en línea que es la referencia autorizada sobre cómo se utilizan las funciones. Pero no lea la documentación en línea todavía. Primero, pruebe los ejemplos de las secciones que siguen a la tabla.

library(readr) probability_distribution_functions_in_r <- read_csv("probability-distribution-functions-in-r.csv")

Funciones para distribución de probabilidades in R.

Distribution

p

q

d

r

Beta

pbeta

qbeta

dbeta

rbeta

Binomial

pbinom

qbinom

dbinom

rbinom

Cauchy

pcauchy

qcauchy

dcauchy

rcauchy

Chi-Square

pchisq

qchisq

dchisq

rchisq

Exponential

pexp

qexp

dexp

rexp

F

pf

qf

df

rf

Gamma

pgamma

qgamma

dgamma

rgamma

Geometric

pgeom

qgeom

dgeom

rgeom

Hypergeometric

phyper

qhyper

dhyper

rhyper

Logistic

plogis

qlogis

dlogis

rlogis

Log Normal

plnorm

qlnorm

dlnorm

rlnorm

Negative Binomial

pnbinom

qnbinom

dnbinom

rnbinom

Normal

pnorm

qnorm

dnorm

rnorm

Poisson

ppois

qpois

dpois

rpois

Student t

pt

qt

dt

rt

Studentized Range

ptukey

qtukey

dtukey

rtukey

Uniform

punif

qunif

dunif

runif

Weibull

pweibull

qweibull

dweibull

rweibull

Wilcoxon Rank Sum Statistic

pwilcox

qwilcox

dwilcox

rwilcox

Wilcoxon Signed Rank Statistic

psignrank

qsignrank

dsignrank

rsignrank

3.1.2 La distribución binomial

3.1.3 La distribución normal

3.2 Introduccion a los test de hipótesis

Primero cabe preguntarse por qué intentaríamos hacer inferencias sobre un parámetro de una población basadonos en una muestra, en lugar de simplemente recopilar datos para toda la población, calcular estadísticas que nos interesan y tomar decisiones basadas en eso. La principal razón por la que utilizamos una muestra en lugar de toda la población es porque recopilar datos sobre toda la población es más compicado o en ocasiones impracticable por varios motivos (complejidad, coste, limitación de tiempo, entre muchos otros motivos). [1]

> El objetivo general de una prueba de hipótesis es sacar conclusiones para confirmar o refutar una creencia sobre una población basándonos en un grupo más pequeño de observaciones.

Las pruebas de hipótesis tienen muchas aplicaciones prácticas. Aquí ponemos algunos ejemplos:

1. Una media: supongamos que a un político le gustaría le gustaría probar si el salario medio de los trabajadores españoles es diferente de 1200 euros.
2. Dos medias:
   - Muestras independientes: supongamos que a un profesor le gustaría probar la idoneidad de un nuevo método docente midiendo la nota media para los alumnos de un grupo de control y los del grupo en el que se ha introducido la novedad.
   - Muestras pareadas o relacionadas: supongamos que el mismo profesor quisiera probar la idoneidad de este nuevo método de enseñanza pero lo hiciera midiendo los conocimientos de los alumnos antes y después de la explicación.
3. Una proporción: supongamos que a un analísta político quisiera comprobar si la proporción de ciudadanos que van a votar por un candidato específico es inferior al 25%.

[1] Por ejemplo, una investigación podría consistir en conocer si la población de la provincia de Tarragona está satisfecha con el nuevo plan de movilidad. Si pudiéramos preguntar a toda la población en un período de tiempo razonable, no haríamos ninguna estadística inferencial. No obstante, aún habria que decidir que preguntas se les hace para entender mejor el motivo de su grado de satisfacción, complicando y encareciendo aún más la encuesta.

4. Dos proporciones: supongamos que a un geógrafo le gustaría probar si la proporción de visitantes a una playa es diferente entre jóvenes y personas mayores.
5. Una variación: supongamos que un ingeniero quisiera probar si una batería tiene una variabilidad en el timpo de carga menor que la indicada en la descripción técnica.
6. Dos variaciones: supongamos que, en una fábrica, dos líneas de producción funcionan independientemente una de la otra. El gerente querría probar si los costes del mantenimiento semanal de estas dos cadenas de producción tienen la misma variación.

Por supuesto, hay muchísimas más aplicaciones potenciales y muchas preguntas de investigación pueden responderse gracias a una prueba de hipótesis.

Por lo general, **las pruebas de hipótesis se utilizan para responder preguntas de investigación en análisis confirmatorios**. Los análisis confirmatorios se refieren a análisis estadísticos donde las hipótesis — deducidas de la teoría — se definen de antemano (preferiblemente antes de la recopilación de datos). En este enfoque, la investigadora tiene una idea específica sobre las variables en consideración y está tratando de ver si su idea, especificada como hipótesis, está respaldada por datos.

Podemos utilizar al menos tres métodos diferentes para realizar una prueba de hipótesis comparando:

1. la estadística de prueba con el **valor crítico**.
2. el ***p*-valor** con el nivel de significancia α.
3. el parámetro objetivo con el **intervalo de confianza**.

Estos enfoques puede diferir en algunos aspectos pero tienen muchos puntos en común. El uso de uno u otro método es a menudo una cuestión de elección personal o de contexto.

Para los tres métodos, explicaré los pasos necesarios para realizar una prueba de hipótesis desde un punto de vista general y los ilustraré con la siguiente situación: [2].

> Supongamos que un político quisiera comprobar si el salario medio de los trabajadores españoles es diferente de 1.200 euros.

En la mayoría de las pruebas de hipótesis, la prueba que vamos a utilizar como ejemplo a continuación requiere algunas condiciones. En esta sección asumimos que se cumplen todos los supuestos pero más adelante hablaremos de esto.

[2]Puede ver más o menos pasos en otros artículos o libros de texto, dependiendo de si estos pasos son detallados o concisos. Sin embargo, la prueba de hipótesis debe seguir el mismo proceso independientemente del número de pasos

3.2.1 Comparando la estadística de prueba con el valor crítico.

Este metodo consiste en reproducir los siguientes 4 pasos:

1. Establecer la **hipótesis nula y alternativa**
2. Calcular la **estadística de prueba**
3. Encontrar el **valor crítico**
4. **Concluir** e interpretar los resultados

3.2.1.1 Estableciendo la hipótesis nula y alternativa

Una prueba de hipótesis primero requiere una suposición sobre un fenómeno o hipótesis, que se deriva de la teoría y la pregunta de investigación.

Dado que una prueba de hipótesis se utiliza para confirmar o refutar una creencia previa, necesitamos **formular nuestra creencia de modo que haya una hipótesis nula y una alternativa**. Esas hipótesis deben ser **mútuamente excluyentes**, lo que significa que no pueden ser verdaderas al mismo tiempo. En el contexto, las hipótesis nula y alternativa son así:

- Hipótesis nula $H_0 : \mu = 1200$
- Hipótesis alternativa $H_1 : \mu \neq 1200$

Al plantear la hipótesis nula y alternativa, tenga en cuenta los siguientes tres puntos:

1. *Siempre estamos interesados en la población y no en la muestra.* Esta es la razón por la que H_0 y H_1 siempre se escribirán en términos de población y no en términos de muestra (en este caso, μ y no $\overline{x}$).
2. *La suposición que nos gustaría probar es a menudo la hipótesis alternativa.* Si quisieramos probar si el salario medio de los trabajadores españoles es inferior a 1200 euros, habríamos establecido que $H_0 : \mu = 1200$ (o equivalentemente, $H_0 : \mu \geq 1200$) y $H_1 : \mu < 1200$. ***No hay que confundir la hipótesis nula con la alternativa, o las conclusiones serán diametralmente opuestas***.
3. *La hipótesis nula es a menudo el status quo.* Por ejemplo, suponiendo que un empresario quiere probar si el nuevo logo de su marca es mejor valorado que el logo anterior. El *status quo* es que los dos logos sean igualmente valorados. Suponiendo que un valor mayor es mejor, entonces se escribirá $H_0 : \mu_{nuevo} = \mu_{viejo}$ (o equivalentemente, $H_0 : \mu_{nuevo} - \mu_{viejo} = 0$) y $H_1 : \mu_{nuevo} > \mu_{viejo}$ (o equivalentemente, $H_0 : \mu_{nuevo} - \mu_{viejo} > 0$). Por el contrario, si cuanto más bajo mejor, habríamos escrito $H_0 : \mu_{nuevo} = \mu_{viejo}$ (o equivalentemente, $H_0 : \mu_{nuevo} - \mu_{viejo} = 0$) y $H_1 : \mu_{nuevo} < \mu_{viejo}$ (o equivalentemente, $H_0 : \mu_{nuevo} - \mu_{viejo} < 0$).

3.2.1.2 Calcular la estadística de prueba

La **estadística de prueba** (o **t-stat**) es una métrica que indica **qué tan extremas son las observaciones en comparación con la hipótesis nula**. Cuanto mayor sea el *t*-stat (en valor absoluto), más extremas serán las observaciones.

Hay varias fórmulas para calcular el t-stat, con una fórmula para cada tipo de prueba de hipótesis: una o dos medias, una o dos proporciones, una o dos varianzas. Esto significa que hay una fórmula para calcular el t-stat para una prueba de hipótesis en una media, otra fórmula para una prueba en dos medias, otra para una prueba en una proporción, etc. [3] La única dificultad en este segundo paso es elegir la fórmula adecuada. Tan pronto como se sepa qué fórmula utilizar según el tipo de prueba, simplemente debe aplicársele a los datos. Afortunadamente, las fórmulas para las pruebas de hipótesis en una y dos medias, y una y dos proporciones siguen la misma estructura.

Calcular la estadística de prueba para estas pruebas es similar a *escalar* una variable aleatoria (un proceso también conocido como "estandarización" o "normalización") que consiste en restar la media de esa variable aleatoria y dividir el resultado por la desviación estándar:

$$Z = \frac{X - \mu}{\sigma}$$

Para estas 4 pruebas de hipótesis (una/dos medias y una/dos proporciones), calcular el estadístico de prueba es como escalar el estimador (calculado a partir de la muestra) correspondiente al parámetro de interés (en la población). Así que básicamente restamos el parámetro objetivo del estimador puntual y luego dividimos el resultado por el error estándar (que es equivalente a la desviación estándar, pero para un estimador).

Si esto no está claro, así es como se calcula la estadística de prueba (t_{obs}) en nuestro ejemplo (asumiendo que se desconoce la varianza de la población):

$$t_{obs} = \frac{\bar{x} - \mu}{\frac{s}{\sqrt{n}}}$$

dónde:

- $\overline{x}$ es la media de la muestra (es decir, el estimador)
- μ es la media bajo la hipótesis nula (es decir, el parámetro objetivo)
- s es la desviación estándar de la muestra
- n es el tamaño de la muestra
- ($\frac{s}{\sqrt{n}}$ es el error estándar)

[3] Incluso hay diferentes fórmulas dentro de cada tipo de prueba, dependiendo de si se cumplen o no algunos supuestos.

Suponiendo que en nuestro caso tenemos una media muestral de 1150 euros ($\overline{x} = 1150$), una desviación estándar muestral de 200 euros ($s = 200$) y un tamaño de muestra de 30 trabajadores ($n = 30$) y, teniendo en cuenta que la media poblacional (la media bajo la hipótesis nula) es 1200 euros ($\mu = 1200$), el t-stat quedaría así:

$$t_{obs} = \frac{\bar{x} - \mu}{\frac{s}{\sqrt{n}}} = \frac{1150 - 1200}{\frac{200}{\sqrt{30}}} = -1.369306$$

Aunque las fórmulas son diferentes según el parámetro que esté probando, el valor encontrado para la estadística de prueba nos da una indicación de cuán extremas son nuestras observaciones.

Recordemos este valor de `-1.369306` porque se volverá a utilizar al final de este test para compararlo con el valor crítico.

3.2.1.3 Encontrando el valor crítico

Aunque el `t-stat` nos da una indicación como de extremas son nuestras observaciones, necesitamos comparar este valor con un umbral o **valor crítico**, que viene dado por una ***distribución de probabilidad***.

De la misma manera que la fórmula para calcular el `t-stat` es diferente para cada parámetro de interés, la distribución de probabilidad subyacente en la que se basa el *valor crítico* también es diferente para cada parámetro objetivo. Esto significa que, además de elegir la fórmula apropiada para calcular el `t-stat`, también necesitamos seleccionar la distribución de probabilidad apropiada dependiendo del parámetro que estemos probando.

Afortunadamente, solo hay 4 distribuciones de probabilidad diferentes para las pruebas de hipótesis cubiertas aquí (recordemos que son una/dos medias, una/dos proporciones y una/dos varianzas):

1. Distribución normal estándar:
    - prueba en una y dos medias con varianzas de población conocidas.
    - prueba en dos muestras donde se conoce la varianza de la diferencia entre las 2 muestras σ_D^2
    - prueba en una y dos proporciones (dado que se cumplen algunos supuestos).
2. Distribución de Student:
    - prueba en una y dos medias con *varianza(s) de población desconocida(s).
    - prueba en dos muestras donde la varianza de la diferencia entre las 2 muestras σ_D^2 es *desconocida*.
3. Distribución Chi-cuadrado:
    - prueba en una varianza.
4. Distribution de fisher:
    - prueba en dos varianzas.

Cada distribución de probabilidad tiene sus propios parámetros, definiendo su forma y/o ubicación. Los parámetros de una distribución de probabilidad pueden verse como si fuesen marcadores de ADN; lo que significa que la distribución está completamente definida por su(s) parámetro(s).

Volviendo a nuestra investigación, la distribución de probabilidad subyacente de una prueba en una media es la distribución Normal estándar o de Student, dependiendo de si la varianza de la *población* (no la varianza de la muestra) es conocida o no:

- Si se conoce la varianza de la población →, se usa la distribución Normal estándar
- Si la varianza de la población es *desconocida* →, se utiliza la distribución de Student

Si no se proporciona explícitamente la varianza de la población, se puede suponer que es desconocida, ya que no se puede calcular basándonos en una muestra. Si pudiera calcularlo, eso significaría que tiene acceso a toda la población y, en este caso, no tiene sentido realizar una prueba de hipótesis (simplemente podría usar algunas estadísticas descriptivas para confirmar o refutar su creencia dicha hipótesis. En nuestro ejemplo, no se especifica la varianza de la población, por lo que se supone que es desconocida. Por lo tanto, usaremos la distribución de Student.

La distribución Student tiene un parámetro que la define: el número de grados de libertad. El número de grados de libertad depende del tipo de prueba de hipótesis. Por ejemplo, el número de grados de libertad para una prueba en una media es igual al número de observaciones menos uno $(n-1)$. Sin ir demasiado lejos en los detalles, el -1 proviene del hecho de que hay una cantidad que se estima (es decir, la media). Siendo el tamaño de la muestra igual a 30 en nuestro ejemplo, los grados de libertad son iguales a $n-1=30-1=29$.

Por último, para encontrar el valor crítico también es necesario conocer el **nivel de significancia** α, que es la **probabilidad de rechazar erróneamente la hipótesis nula aunque en realidad sea verdadera**. En este sentido, es un error de tipo I (en contraposición al error de tipo II) que aceptamos para poder sacar conclusiones sobre una población a partir de un subconjunto de ella.

En muchas aplicaciones el nivel de significancia se suele establecer en el 5%. En cambio, en algunos campos (como la medicina o la ingeniería, entre otros), el nivel de significancia también se establece a veces en el 1% para disminuir la tasa de error. Es mejor especificar el nivel de significancia *antes* de realizar una prueba de hipótesis para evitar la tentación de establecer el nivel de significancia de acuerdo con los resultados (la tentación es aún mayor cuando los resultados están al borde de ser significativos). En nuestro caso, tomamos $\alpha = 5\% = 0.05$.

Además, queremos probar si el salario medio de los trabajadores españoles es **diferente** de 1200 euros. Si quisiéramos probar que el salario medio fuera inferior a 1200 euros ($H_1 : \mu < 1200$) o superior a 1200 ($ H_1: >1200$),

habríamos realizado una prueba unilateral. Asegúrese de realizar la prueba correcta (bilateral o unilateral) porque tiene un impacto en cómo encontrar el valor crítico.

Ahora que conocemos la distribución apropiada (distribución de Student), su parámetro (grados de libertad (gl) = 29), el nivel de significancia ($\alpha = 0.05$) y la dirección (bilateral), tenemos todo lo que necesitamos para calcular el valor crítico. Podríamos localizar este valor en la tabla estadística correspondiente o directamente lo podríamos calcular con R.

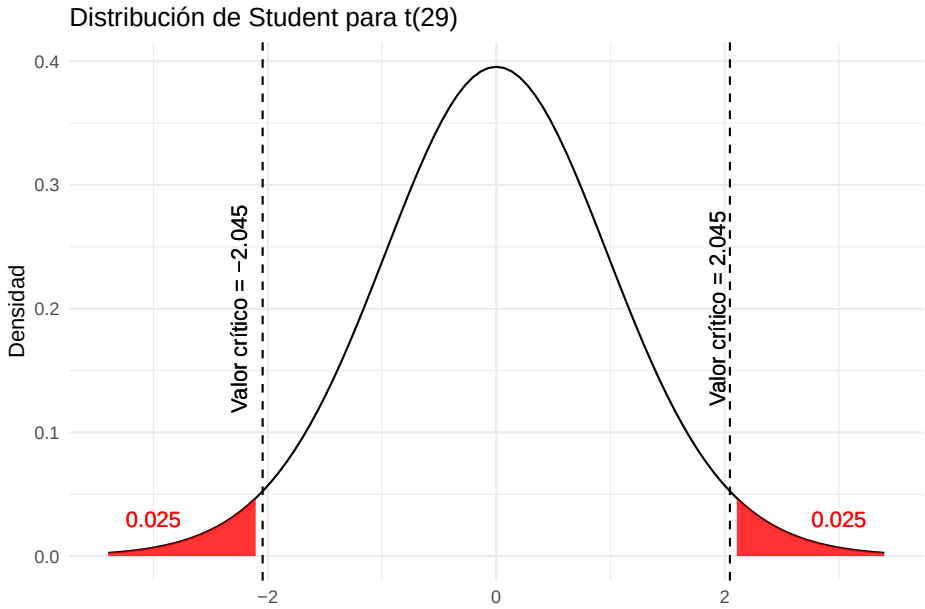
Al observar la fila df = 29 y la columna $t_{.}025$ en la tabla de distribución de Student, encontramos un valor crítico de:

$$t_{n-1;\alpha/2} = t_{29;0.025} = 2.04523$$

Tomamos $t_{\alpha/2} = t_{.}025$ y no $t_{\alpha} = t_{.}05$ ya que el nivel de significancia es 0.05 y estamos haciendo una prueba bilateral (de dos lados; $ H_1: 1200$), por lo que la tasa de error de 0.05 debe dividirse en 2 para encontrar el valor crítico a la derecha de la distribución. Dado que la distribución de Student es simétrica, el valor crítico a la izquierda de la distribución es simplemente: -2.04523.

Visualmente, la tasa de error de 0.05 se divide en dos partes:

- 0,025 a la izquierda de -2,04523 y
- 0,025 a la derecha de 2,04523



Al igual que en el apartado anterior, cabe recordar estos valores críticos de -2,045 y 2,045 el último paso.

Las áreas sombreadas en rojo en el gráfico anterior también se conocen como regiones de rechazo.

Estos valores críticos también se pueden encontrar en R, gracias a la función `qt()`:

```
qt(0.025, df = 29, lower.tail = TRUE)
```

```
## [1] -2.04523
```

```
qt(0.025, df = 29, lower.tail = FALSE)
```

```
## [1] 2.04523
```

Como se ha visto en el tema sobre distribuciones de probabilidad, la función `qt()` se usa para la distribución de Student (`q` significa cuantil y `t` para Student). Cabe recordar que hay otras funciones que acompañan a las diferentes distribuciones:

- `qnorm()` para la distribución Normal
- `qchisq()` para la distribución Chi-cuadrado
- `qf()` para la distribución de Fisher

3.2.1.4 Conclusión e interpretación de los resultados

Las únicas dos posibilidades al concluir una prueba de hipótesis son:

1. Rechazo de la hipótesis nula, o
2. No rechazo de la hipótesis nula

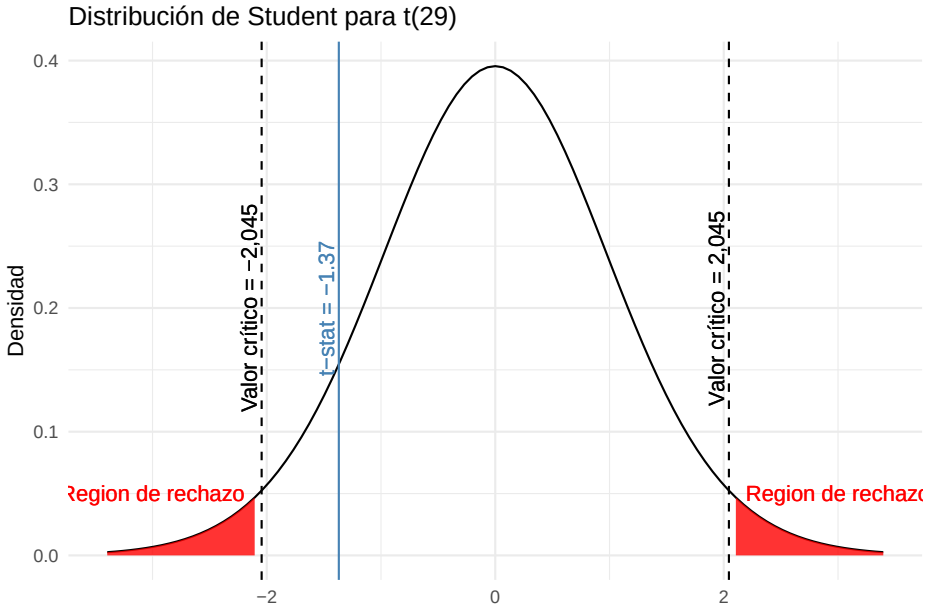
En nuestro ejemplo sobre los salarios de los españoles, recordamos que hemos determinado que el:

- el `t-stat` es -1.369306, y
- los valores críticos son -2.04523 y 2.04523

Recordemos que:

- el **t-stat da una indicación de cuán extrema es nuestra muestra** en comparación con la hipótesis nula
- los **valores críticos son el umbral a partir del cual el t-stat se considera *demasiado* extremo**

Para comparar el `t-stat` con los valores críticos de manera gráfica:



Los dos valores críticos forman las regiones de rechazo (las áreas sombreadas en rojo):

- de $-\infty$ a -2.045, y
- de 2.045 a ∞

Si el ***t-stat* se encuentra dentro de una de estas regiones, rechazamos la hipótesis nula**. Por el contrario, si **t-stat *no* se encuentra dentro de ninguna de las regiones, no rechazamos la hipótesis nula**.

Como podemos ver en el gráfico anterior, el *t-stat* es menos extremo que el valor crítico. En conclusión, no rechazamos la hipótesis nula de que $\mu = 1200$.

Esta es la conclusión en términos estadísticos, pero no tienen sentido sin una interpretación adecuada. Por tanto, es una buena práctica interpretar también el resultado en el contexto del problema:

> Con un nivel de significancia del 5%, no rechazamos la hipótesis de que el salario medio de los trabajadores españoles es de 1200 euros.

¿Qué significa esto realmente? Dicho de otro modo:

> "nosotros *no rechazamos* la hipótesis nula" y "nosotros *no rechazamos* la hipótesis de que el salario medio de los trabajadores españoles es igual a 1200 euros". No escribimos "*aceptamos o estamos de acuerdo con* la hipótesis nula" ni "el salario medio de los trabajadores españoles es de 1200 euros".

En los test de hipótesis, llegamos a una conclusión sobre la población a partir de una muestra. Por tanto, siempre existe cierta incertidumbre y no podemos

decir que estemos seguros al 100% de que nuestra conclusión sea correcta.

Quizás sea el caso de que el salario medio de los trabajadores españoles sea en realidad diferente a 1200 euros, pero **no lo pudimos demostrar** con los datos disponibles. Si tuviéramos más observaciones hubiéramos rechazado la hipótesis nula (dado que todo lo demás es igual, un tamaño de muestra más grande implica un `t-stat` más extremo). O puede darse el caso de que, incluso con más observaciones, no hubiéramos rechazado la hipótesis nula porque el salario de los trabajadores españoles en realidad se acerca a los 1200 euros. Con los datos disponibles no podemos distinguir entre estas dos posibilidades. Simplemente debemos admitir que no encontramos suficiente evidencia en contra de la hipótesis de partida, pero tampoco concluimos que la media sea igual a 1200 euros.

3.2.2 Comparando el *p*-valor con el nivel de significancia α

Este método consiste en los siguientes pasos:

1. Enunciar las **hipótesis nula y alternativa**
2. Calcular la **estadística de prueba** (`t-stat`).
3. Calcular el ***p*-valor**
4. **Concluir** e interpretar los resultados

En este segundo método que utiliza el valor p, los dos primeros pasos son similares a los del primer método, mientras que la interpretación de los resultados tiene algunos puntos en común.

3.2.2.1 Establecer las hipótesis

Las hipótesis de investigación (nula y alternativa) siguen siendo las mismas:

- $H_0 : \mu = 1200$
- $H_1 : \mu \neq 1200$

3.2.2.2 Calcular la estadística de prueba

Cabe recordar que la fórmula del estadístico t es diferente según el tipo de prueba de hipótesis (una o dos medias, una o dos proporciones, una o dos varianzas). En nuestro caso de una sola media con varianza desconocida, tenemos que:

$$t_{obs} = \frac{\bar{x} - \mu}{\frac{s}{\sqrt{n}}} = \frac{1150 - 1200}{\frac{200}{\sqrt{30}}} = -1.369306$$

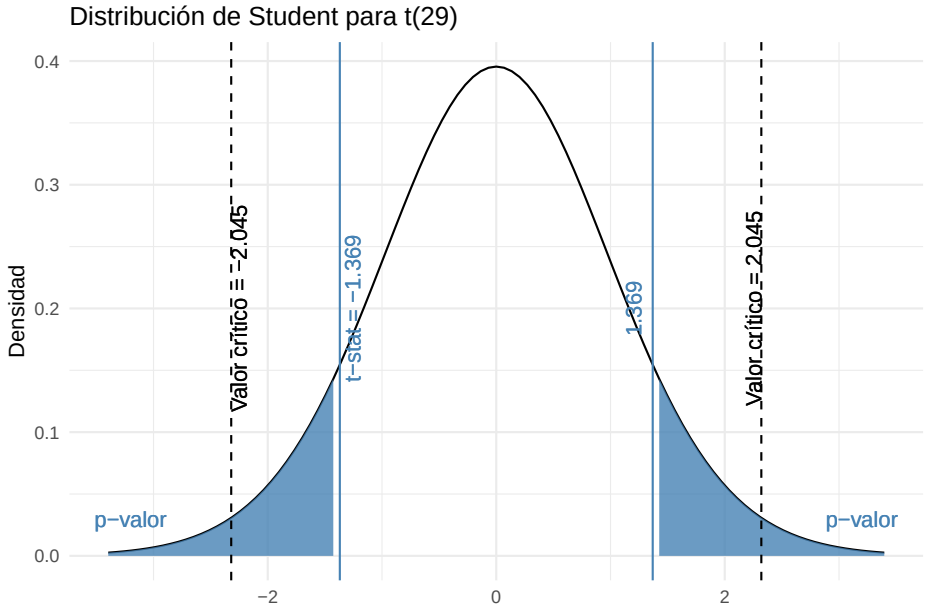
3.2.2.3 Calculo del valor *p*

El ***p*-valor** es la probabilidad (de 0 a 1) de observar una muestra al menos tan extrema como la que observamos si la hipótesis nula fuera cierta. Dicho de otro

modo: **¿cómo de probable es la hipótesis nula?**. También se define como el nivel de significancia más pequeño para el cual los datos indican el rechazo de la hipótesis nula.

Formalmente, el valor p es el área más allá del estadístico de prueba. Como estamos haciendo una prueba bidireccional, el valor p es, por lo tanto, la suma del área por encima de 1,369306 y por debajo de -1,369306.

Visualmente, el valor p es la suma de las dos áreas sombreadas en azul en la siguiente gráfica:



El valor p se puede obtener también con tablas estadísticas o es posible calcularlo con precisión en R con la función `pt()`:

```
p_val <- pt(-1.369306, df = 29, lower.tail = TRUE) + pt(1.369306, df = 29, lower.tail =
p_val
```

```
## [1] 0.1814156
```

```
# Que es lo mismo que...
p_val <- 2 * pt(1.369306, df = 29, lower.tail = FALSE)
p_val
```

```
## [1] 0.1814156
```

El valor p es 0.1814, que indica que hay un 18.14% de probabilidad de observar una muestra al menos tan extrema como la observada si el hipótesis nula eran verdaderas. Esto ya nos da una pista sobre si nuestro t-stat es demasiado extremo o no (y, por lo tanto, si nuestra hipótesis nula es probable o no).

Como la función `qt()` para encontrar el valor crítico, usamos `pt()` para encontrar el valor p porque la distribución subyacente es la distribución de Student. En otros casos se utilizarían las funciones `pnorm ()`, `pchisq ()` y `pf ()` para las otras distribuciones mencionadas anteriormente (Normal, Chi-cuadrado y Fisher).

3.2.2.4 Concluir e interpretar los resultados

Finalmente, hay que comparar el valor p que acabamos de calcular con el nivel de significancia α. Como para todas las pruebas estadísticas:

- Si el ***p*-valor es menor** que α (p-valor$<$ 0.05), entonces H_0 es poco probable $\rightarrow$ **rechazamos** la hipótesis nula.
- Si el ***p*-valor es mayor que o igual** a α (p-valor $\geq$ 0.05), entonces H_0 es probable $\rightarrow$ **no podemos rechazar** la hipótesis nula.

No importa si tomamos en consideración el p-valor exacto (es decir, 0.1814) o el acotado (0.05 $<p$-valor $<$0.10), es mayor que 0.05, entonces no rechazamos la hipótesis nula. En el contexto del problema, no rechazamos la hipótesis nula de que el salario medio de los trabajadores españoles es igual a 1200 euros.

El resultado obtenido ha sido el mismo que en el primer método. Evidentemente, debería dar lo mismo si se usan los mismos datos y con el mismo nivel de significancia.

3.2.3 Comparación del parámetro objetivo con el intervalo de confianza

Este método consiste en calcular primero el intervalo de confianza y comparar sobre éste el parámetro objetivo (el parámetro bajo la hipótesis nula). Podemos distinguir tres pasos:

1. Enunciar las **hipótesis nula y alternativa**
2. Calcular el **intervalo de confianza**
3. **Concluir** e interpretar los resultados

También se pueden apreciar varias similitudes con los métodos anteriores.

3.2.3.1 Enunciar las hipótesis

Nuevamente, las hipótesis nula y alternativa siguen siendo las mismas:

- $H_0 : \mu = 1200$
- $H_1 : \mu \neq 1200$

3.2.3.2 Calcular el intervalo de confianza

Al igual que los test de hipótesis, los intervalos de confianza son una herramienta bien conocida en la estadística inferencial. El **intervalo de confianza** es un

procedimiento de estimación que produce un **intervalo que contiene el parámetro verdadero con una cierta probabilidad**.

De la misma manera que existe una fórmula para cada tipo de prueba de hipótesis al calcular las estadísticas de la prueba, existe una fórmula para cada tipo de intervalo de confianza. La fórmula para calcular un intervalo de confianza en una media μ (con varianza poblacional desconocida):

$$(1-\alpha)\%\text{ IC para } \mu = \overline{x} \pm t_{\alpha/2,n-1}\frac{s}{\sqrt{n}}$$

donde $t_{\alpha/2,n-1}$ se encuentra en la tabla de distribución de Student o se puede calcular con R (y es similar al valor crítico encontrado en el primer método).

Dados nuestros datos y con $\alpha = 0.05$, tenemos que:

$$\begin{aligned} 95\%\text{ IC para } \mu &= \bar{x} \pm t_{\alpha/2,n-1}\frac{s}{\sqrt{n}} \\ &= 1150 \pm 2.045\frac{200}{\sqrt{30}} \\ &= [1075,33; 1224,67] \end{aligned}$$

El intervalo de confianza del 95% para μ es [1075,33; 1224,67] euros. **¿Qué significa este intervalo de confianza del 95%?**

Sabemos que este procedimiento de estimación tiene una probabilidad del 95% de producir un intervalo que contenga la media verdadera μ. En otras palabras, **si construimos muchos intervalos de confianza** (con diferentes muestras del mismo tamaño), **el 95% de ellos incluirá la media de la población** (el verdadero parámetro). Del mismo modo el 5% de estos intervalos de confianza no cubrirán la media real.

Si desea disminuir este último porcentaje, puede disminuir el nivel de significancia (por ejemplo $\alpha = 0.01$). En igualdad de condiciones, esto disminuirá el rango del intervalo de confianza y, por lo tanto, aumentará la probabilidad de que incluya el parámetro verdadero.

3.2.3.3 Conclusión e interpretación de los resultados

Finalmente, hay comparar el intervalo de confianza con el valor del parámetro objetivo (el valor cuestionado por la hipótesis nula):

- Si el **intervalo de confianza no incluye** el valor hipotético, H_0 es poco probable $\rightarrow$ **rechazamos** la hipótesis nula.
- Si el **intervalo de confianza incluye** el valor hipotético, H_0 es probable $\rightarrow$, **no rechazamos** la hipótesis nula

En nuestro ejemplo:

- el valor hipotético es 1200 (desde $ H_0: = 1200$)

- 1200 se incluye en el intervalo de confianza del 95%, ya que va de 1075,33 a 1224,67 euros
- Entonces **no rechazamos** la hipótesis nula de que el salario medio de los trabajadores españoles sea de 1200 euros.

Por supuesto, la conclusión es equivalente a la que se había llegado por los otros dos métodos. Esto debe ser así, ya que usamos los mismos datos y el mismo nivel de significancia α para los tres métodos.

3.3 Test de hipótesis en R: cálculo e informes

3.4 Un ejemplo sobre brecha salarial entre géneros

3.5 ANOVA

3.6 Algunas consideraciones finales

3.6.1 ¿Cuando no se necesita inferencia?

Hemos analizado varios ejemplos sobre cómo realizar inferencias estadísticas: realización de test de hipótesis y construcción de intervalos de confianza. Antes de empezar a realizar un experimento, siempre es necesario realizar un análisis exploratorio de los datos. Este primer vistazo siempre puede ayudar a intuir sobre lo que los métodos estadísticos como los intervalos de confianza y las pruebas de hipótesis pueden decirnos (y lo que no pueden). En los apartados anteriores hemos querido explicar cómo funciona la inferencia pero no nos hemos preguntado si era realmente necesaria.

Consideremos un ejemplo. Supongamos que estamos interesados en la siguiente pregunta: De *todos* los vuelos que salen de un aeropuerto de la ciudad de Nueva York, ¿los vuelos de Hawaiian Airlines están en el aire por más tiempo que los vuelos de Alaska Airlines? Además, supongamos que los vuelos de 2013 son una muestra representativa de todos esos vuelos. Entonces podemos usar el dataframe `flights` disponible en el paquete `nycflights13` para responder nuestra pregunta. Filtremos este dataframe para incluir solo a Hawaiian y Alaska Airlines usando sus códigos de "operador" "HA" y "AS":

```
library(tidyverse)
library(nycflights13)
flights_sample <- flights %>%
  filter(carrier %in% c("HA", "AS"))
```

Hay dos posibles métodos de inferencia estadística que podríamos utilizar para responder a estas preguntas. Primero, podríamos construir un intervalo de confianza del 95% para la diferencia en las medias poblacionales $\mu_{HA} - \mu_{AS}$,

donde μ_{HA} es el tiempo de vuelo medio de todos los vuelos de Hawaiian Airlines y μ_{AS} es el tiempo medio de vuelo de los vuelos de Alaska Airlines. Luego podríamos verificar si la totalidad del intervalo es mayor que 0, sugiriendo que $\mu_{HA} - \mu_{AS} > 0$, o, en otras palabras, sugiriendo que $\mu_{HA} > \mu_{AS}$. En segundo lugar, podríamos realizar una prueba de hipótesis de la hipótesis nula $H_0 : \mu_{HA} - \mu_{AS} = 0$ frente a la hipótesis alternativa $H_A : \mu_{HA} - \mu_{AS} > 0$.

Construyamos primero una visualización exploratoria como acabamos de sugerir. Dado que `air_time` es numérico y `carrier` es categórico, un diagrama de caja (*boxplot*) puede mostrar la relación entre estas dos variables (ver la Figura 3.1).

```
ggplot(data = flights_sample, mapping = aes(x = carrier, y = air_time)) +
  geom_boxplot() +
  labs(x = "Carrier", y = "Air Time")
```

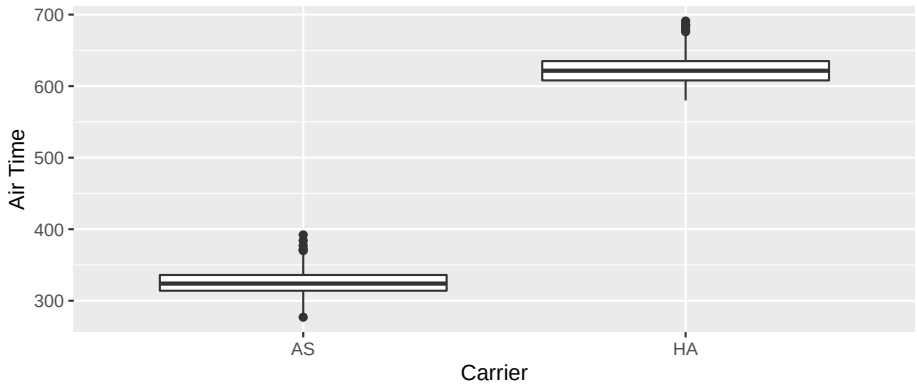


Figura 3.1: Air time for Hawaiian and Alaska Airlines flights departing NYC in 2013.

Esto es lo que nos gusta llamar momentos en los que "no se necesita un doctorado en estadística". No es necesario ser un experto en estadísticas para saber que Alaska Airlines y Hawaiian Airlines tienen horarios aéreos *significativamente* diferentes. ¡Los dos diagramas de caja ni siquiera se superponen! La construcción de un intervalo de confianza o la realización de una prueba de hipótesis, francamente, no proporcionaría mucha más información que la Figura 3.1.

Investiguemos por qué observamos una diferencia tan clara entre estas dos aerolíneas que utilizan la manipulación de datos. Primero agrupemos por las filas de `vuelos_muestra` no solo por`transportista` sino también por destino `dest`. Posteriormente, calcularemos dos estadísticas resumidas: el número de observaciones usando `n ()` y el tiempo medio de transmisión:

```
flights_sample %>%
  group_by(carrier, dest) %>%
```

```
  summarize(n = n(), mean_time = mean(air_time, na.rm = TRUE))
```

```
## # A tibble: 2 x 4
## # Groups:   carrier [2]
##   carrier dest      n mean_time
##   <chr>   <chr> <int>     <dbl>
## 1 AS      SEA     714      326.
## 2 HA      HNL     342      623.
```

Resulta que desde la ciudad de Nueva York en 2013, Alaska solo voló a "SEA" (Seattle) desde la ciudad de Nueva York (NYC) mientras que Hawaiian solo voló a "HNL" (Honolulu) desde Nueva York. Dada la clara diferencia en la distancia entre la ciudad de Nueva York y Seattle y la ciudad de Nueva York a Honolulu, no es sorprendente que observemos tiempos de vuelo tan diferentes (_ estadísticamente significativamente diferentes_, de hecho) en los vuelos.

Este es un claro ejemplo de que no es necesario hacer nada más que un simple análisis exploratorio de datos utilizando visualización de datos y estadísticas descriptivas para llegar a una conclusión adecuada. Por lo tanto, es recomendable empezar siempre por realizar un análisis exploratorio con estadísticas descriptivas antes de aplicar inferencia estadística.

3.6.2 Problemas con los p-valores

Además de los muchos malentendidos comunes sobre las pruebas de hipótesis y los valores de p que hemos comentado al explicar la interpretación de las pruebas de hipótesis, otra consecuencia desafortunada del uso ampliado de los valores de p y las pruebas de hipótesis es un fenómeno conocido como "p-hacking", que es es el acto de "seleccionar" sólo los resultados que son "estadísticamente significativos" y descartar los que no lo son, aunque sea a expensas de las ideas científicas. Hay muchos artículos escritos recientemente sobre malentendidos y problemas con los valores de p. Le recomendamos que consulte algunos de ellos:

1. Malentendidos de los valores de p
2. Qué debate más nerd sobre los valores de p sobre la ciencia y cómo solucionarlo
3. Los estadísticos emiten una advertencia sobre el uso indebido de los valores de p
4. No puede confiar en lo que lee sobre nutrición
5. Una letanía de problemas con valores p

Tales problemas se estaban volviendo tan recurrentes que la Asociación Estadounidense de Estadística (ASA) emitió una declaración en 2016 titulada, "Declaración de la ASA sobre la importancia estadística y los valores de p" con seis principios subyacentes al uso e interpretación adecuados de los valores de p. La ASA publicó esta guía sobre los valores de p para mejorar la conducta y la interpretación de la ciencia cuantitativa y para informar el creciente énfasis

en la reproducibilidad de la investigación científica.

Quizás el uso de intervalos de confianza para la inferencia estadística permita evitar ciertos malentendidos. Sin embargo, en muchos campos todavía se usan exclusivamente valores de p para la inferencia estadística y esta es una razón para incluirlos en este texto.

3.7 Ejercicios

Capítulo 4

El muestreo estadístico

Some *significant* applications are demonstrated in this chapter.

4.1 Example one

4.2 Example two

Capítulo 5

Regresiones

We have finished a nice book.

Capítulo 6

Estadística multivariante

We have finished a nice book.

Bibliografía

Knuth, D. E. (1984). Literate Programming. *The Computer Journal*, 27(2):97–111.

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L. D., François, R., Grolemund, G., Hayes, A., Henry, L., Hester, J., Kuhn, M., Pedersen, T. L., Miller, E., Bache, S. M., Müller, K., Ooms, J., Robinson, D., Seidel, D. P., Spinu, V., Takahashi, K., Vaughan, D., Wilke, C., Woo, K., and Yutani, H. (2019). Welcome to the tidyverse. *Journal of Open Source Software*, 4(43):1686.

Wickham, H. and Grolemund, G. (2017). *R for Data Science: Import, Tidy, Transform, Visualize, and Model Data*. O'Reilly Media, Inc., 1st edition.

Xie, Y. (2015). *Dynamic Documents with R and knitr*. Chapman and Hall/CRC, Boca Raton, Florida, 2nd edition. ISBN 978-1498716963.