

Técnicas multivariantes en Investigación de Mercados

Roberto Gil

2026-10-25

Tabla de contenidos

1	Introducción	8
1.1	Bienvenidos a este sitio web...	9
1.2	¿Quién soy ...?	10
1.3	Vamos con la asignatura ...	11
1.4	Introducción a las herramientas de trabajo	12
1.4.1	Primer paso, instalar el lenguaje R	12
1.4.2	Segundo paso, instalar RStudio	12
1.4.2.1	Cuadrante superior izquierdo	13
1.4.2.2	Cuadrante inferior izquierdo:	13
1.4.2.3	Cuadrante superior derecho:	14
1.4.2.4	Cuadrante inferior derecho	14
1.4.3	Configuración inicial recomendada	15
1.5	¿Cómo verificar que todo ha ido bien?	15
1.5.1	El botón de ayuda en R	16
1.6	Preguntas y respuestas	16
1.7	Instalar y configurar el entorno de trabajo	19
1.7.1	Paso 1. Crear la carpeta del curso	19
1.7.2	Paso 2. Instalar y cargar los paquetes	23
1.7.3	Paquetes utilizados en la asignatura	23
1.7.4	Archivos de datos	28
1.7.5	La rutina de cada sesión	29
1.8	Practicando el <i>scripting</i>	29
1.8.1	Ejercicio: <i>script</i> 1: consola	29
1.8.2	Ejercicio: <i>script</i> 2: vectores y dataframes	30
1.8.3	Ejercicio: <i>script</i> 3: valores perdidos	32
1.8.4	Ejercicio: <i>script</i> 4: funciones generales	33
1.8.5	Ejercicio: Mi primer Quarto document	35
1.8.6	Ejercicio: <i>script</i> 5: carga de datos	36
1.8.7	Ejercicio: <i>script</i> 6: recuentos básicos	42
1.8.8	Ejercicio: <i>script</i> 7: uso de <i>pipes</i>	43
1.8.9	Ejercicio: <i>script</i> 8: personalización de tablas	43
1.8.10	Ejercicio: <i>script</i> 9: combinando estadísticos	44
1.8.11	Ejercicio: <i>script</i> 10: tipos de porcentajes	45
1.8.12	Ejercicio: <i>script</i> 11: gestión de etiquetas	46
1.9	¿Qué se ha aprendido hasta ahora?	46

2	Introducción al análisis	49
2.1	La exploración de datos	50
2.1.1	El investigador y su proceso de trabajo	51
2.1.2	Proceso	53
2.1.3	Escalas de medida	56
2.1.4	Relación instrumento de medida y base de datos	57
2.1.4.1	Múltiple en binario frente a menciones	58
2.2	Herramientas de exploración	60
2.2.1	Frecuencias	60
2.2.2	Histogramas, importancia de la visualización	65
2.2.3	Resumen estadístico	70
2.2.4	Medidas de tendencia central	72
2.2.5	Medidas de dispersión	73
2.2.6	Concepto de normalidad y relación con la muestra	73
2.2.6.1	Asimetría (<i>skewness</i>)	75
2.2.6.2	Curtosis (<i>kurtosis</i>)	75
2.2.6.3	Importancia en investigación de mercados	75
2.3	Preocupaciones del analista de datos	77
2.3.1	Valores perdidos (<i>missing values</i>)	77
2.3.1.1	¿Por qué faltan los datos?	78
2.3.1.2	Métodos de corrección	78
2.3.2	Valores atípicos (<i>outliers</i>)	79
2.3.2.1	Identificación visual	79
2.3.3	¿Por qué se producen?	82
2.3.3.1	Detección	82
2.3.3.2	Métodos de corrección	84
3	Inferencia e Investigación de Mercados	85
3.1	Introducción al proceso inferencial	86
3.1.1	La lógica de la inferencia estadística	86
3.2	Normalidad estadística	87
3.2.1	Herramientas de diagnóstico	88
3.2.1.0.1	La trampa de la muestra grande	88
3.2.1.1	Testar la normalidad	89
3.2.1.2	Otros diagnósticos visuales	91
3.2.1.2.1	Gráfico Q-Q estándar	92
3.2.1.3	Guía rápida para los informes	92
3.3	El procedimiento de linealización	93
3.3.0.1	Transformaciones comunes en R	93
3.4	Homogeneidad de varianzas (homoscedasticidad)	98
3.4.0.1	Test de Levene	98
3.4.0.2	Ratio de varianzas	100
3.5	Introducción a la inferencia estadística paramétrica	102
3.6	Inferencia paramétrica	103
3.6.1	Test t de Student para una muestra	103

3.6.2	Test t de Student para muestras independientes	105
3.6.2.1	El protocolo de decisión	105
3.6.2.2	Diagnóstico de homogeneidad de varianzas (Test de Levene)	106
3.6.2.3	Opción A: Varianzas iguales (Levene $p > 0.05$) . . .	106
3.6.2.4	Opción B: Varianzas diferentes (Levene $p \leq 0.05$) . .	107
3.6.3	Test de comparación de proporciones (Test Z)	108
3.6.3.1	Preparación de los datos	109
3.6.3.2	Ejecución del Test Z	109
3.6.4	Test t de Student para muestras dependientes (pareadas) . .	111
3.6.4.1	El protocolo de decisión	111
3.6.5	Correlación paramétrica de Pearson	115
3.6.6	Análisis de varianza de un factor (ANOVA one-way)	119
3.6.7	Análisis de varianza de dos factores (ANOVA two-way) . . .	121
3.7	Inferencia no paramétrica	123
3.7.1	Prueba Chi-cuadrado (χ^2)	123
3.7.1.1	Chi-cuadrado de bondad de ajuste (una muestra) . .	123
3.7.1.2	Chi-cuadrado de independencia (tabla de contingencia)	124
3.7.1.3	Medidas de asociación derivadas	126
3.7.2	Correlaciones no paramétricas	127
3.7.2.1	Correlación de Spearman (ρ)	127
3.7.2.2	Correlación de Kendall (τ)	128
3.7.3	Prueba de Wilcoxon — Muestras independientes (U de Mann-Whitney)	129
3.7.3.1	Diagnóstico de normalidad (Lilliefors y Shapiro-Wilk) .	130
3.7.3.2	Prueba W de Wilcoxon — U de Mann-Whitney . . .	132
3.7.4	Prueba de Wilcoxon — Muestras dependientes (V de Wilcoxon) .	135
3.7.4.1	Diagnóstico de normalidad de las diferencias	135
4	Análisis de Correspondencias Simple (CA)	138
4.1	Introducción	139
4.1.1	Tabla de frecuencias	139
4.1.2	Mapa cartesiano	140
4.2	La práctica del CA	144
4.3	Carga de los datos en forma de tabla	145
4.4	Prueba de homogeneidad (Chi-cuadrado)	146
4.5	Análisis de residuos y su tipología	147
4.6	Masas y perfiles: la base del cálculo de distancias	150
4.7	Estimación del modelo y resultados	153
4.7.1	Bloque de inercia (Principal Inertias)	154
4.7.2	Bloque de filas y columnas (Rows / Columns)	155
4.7.3	Visualización	156
4.8	Decálogo de interpretación del análisis de correspondencias . .	159
4.9	Diagnóstico estratégico para HATCO	161

4.10	Puntos suplementarios	161
4.10.1	Visualización del posicionamiento IDEAL	164
5	Análisis de Componentes Principales (PCA)	166
5.1	Introducción	167
5.1.1	Un ejemplo real: la escala E-S-QUAL	168
5.2	Caso de ejemplo: HBAT	169
5.2.1	La adecuación de los datos a la técnica	170
5.2.2	Preparación del entorno y datos	170
5.2.2.1	Diagnóstico de la matriz de correlaciones	172
5.3	Adecuación a los requisitos del análisis	173
5.3.1	El índice de Kaiser-Meyer-Olkin (KMO)	174
5.3.2	La prueba de esfericidad de Bartlett	174
5.3.3	Proceso iterativo de purificación (MSA individual)	175
5.4	Determinación del número de componentes	177
5.4.1	Criterio de los autovalores (<i>eigenvalues</i>) > 1	177
5.4.2	Porcentaje de la varianza explicada	177
5.4.3	Gráfico de sedimentación (<i>scree plot</i>)	178
5.5	Extracción y rotación de componentes	178
5.5.1	Interpretación de la salida de <code>psych::principal()</code> en R	181
5.5.1.1	Las nuevas dimensiones: RC1, RC2, RC3...	181
5.5.1.2	Los indicadores de calidad: h2, u2 y com	182
5.5.1.3	Comunalidades SPSS / unicidad R	182
5.5.2	Criterios de significación de las cargas factoriales	185
5.5.2.1	Relación entre carga factorial y tamaño muestral	185
5.5.2.2	Relación entre carga factorial y el número de variables	186
5.6	Interpretación de los nuevos componentes	186
5.7	Las puntuaciones factoriales	187
5.7.1	¿Qué son exactamente las puntuaciones?	187
5.7.2	Utilidad estratégica: puente hacia la segmentación (<i>cluster</i>)	188
5.7.2.1	Comprobación del cálculo de los <i>scores</i>	189
5.7.3	Representación gráfica	190
5.7.3.1	El mapa de individuos	190
5.7.3.2	¿Por qué es útil esta visualización?	190
5.8	¿PCA o FA?	192
5.8.1	Análisis de componentes principales (PCA)	192
5.8.2	Análisis factorial (FA o factores principales)	193
5.8.3	Comparativa de resultados PCA / FA	193
5.8.4	Comentarios finales	197
5.8.4.1	En PCA (análisis de componentes principales)	197
5.8.4.2	En FA (análisis de factores principales)	197
6	Análisis de Conglomerados (Cluster)	198
6.1	Introducción	199
6.2	Un ejemplo intuitivo acerca del lenguaje	199
6.2.1	Matriz de datos original (frecuencia de coincidencia)	200

6.2.2	El historial de conglomeración	201
6.2.2.1	Interpretación del historial	203
6.2.2.2	Representación visual con dendrograma	204
6.3	Pasos y objetivos del análisis <i>cluster</i>	204
6.4	Adecuación de la estructura de datos	205
6.4.1	Detección de <i>outliers</i>	205
6.4.1.1	La naturaleza del <i>outlier</i> en el análisis de grupos	205
6.4.1.2	Distancia de Mahalanobis	205
6.4.1.3	El criterio de significación estadística χ^2	206
6.4.1.4	Eliminar o mantener, ¿qué decisión tomar?	211
6.4.2	Medición de distancias	211
6.4.3	Estandarización	212
6.4.3.1	El método de estandarización en R	212
6.4.3.2	Ejemplo: cluster3.sav sin estandarizar	212
6.4.3.3	Ejemplo: cluster3.sav estandarizado	214
6.4.3.4	Consideraciones críticas en la estandarización	215
6.5	Análisis <i>cluster</i> , introducción	215
6.5.1	El método jerárquico: la exploración del número de grupos	216
6.5.2	El método no jerárquico (<i>K-means</i>): el ajuste de precisión	216
6.5.3	La estrategia combinada: lo mejor de ambos mundos	216
6.6	<i>Cluster</i> jerárquico	217
6.6.1	Métodos de vinculación (<i>Linkage</i>)	217
6.7	Implementación del <i>cluster</i> jerárquico con R	217
6.7.1	Determinación del número óptimo de grupos	220
6.7.2	Obtención de centroides finales	222
6.8	<i>Cluster</i> no jerárquico o <i>K-means</i>	223
6.8.1	La lógica del algoritmo <i>K-means</i>	223
6.9	Implementación del <i>cluster</i> no jerárquico (<i>K-means</i>) con R	224
6.9.1	Caracterización de los segmentos	224
6.9.2	Validación mediante ANOVA	225
6.9.3	Perfilado de los segmentos	226
6.10	Conclusión de la sesión	228
7	Análisis de Regresión Lineal	229
7.1	Introducción	230
7.1.1	Objetivos y fines del modelo	230
7.1.2	Determinación de las variables	231
7.2	Modelo de estimación	231
7.3	Planteamiento del problema a resolver	231
7.4	El proceso del análisis de regresión	233
7.4.1	Requisitos previos y tamaño de la muestra	233
7.4.2	Relaciones lineales	233
7.4.3	Descriptivos y correlaciones	234
7.5	Estimación del modelo: primera ejecución	237
7.5.1	Selección de las variables y método	237
7.5.2	Identificación de <i>outliers</i> y casos influyentes	238

7.5.3	Indicadores de ajuste	240
7.6	Diagnóstico del modelo	242
7.6.1	Significación global del modelo (Prueba F)	244
7.6.2	Significación de los parámetros individuales (Prueba t)	244
7.6.3	Interpretación de los coeficientes: B vs. β	244
7.6.4	Implicaciones de los coeficientes no significativos	246
7.6.4.1	El principio de parsimonia (la navaja de Ockham)	246
7.6.4.2	Eliminación hacia atrás (<i>Backward Elimination</i>)	246
7.6.4.3	El indicador clave, el <i>Adjusted R²</i>	246
7.6.4.4	Excepción: variables de control teórico	247
7.6.5	Correlaciones parciales y semiparciales	247
7.7	Validación de los supuestos básicos del modelo	248
7.7.1	Muestra	250
7.7.2	Residuos extremos	250
7.7.3	Independencia: Durbin-Watson	252
7.7.4	Linealidad	253
7.7.5	Homocedasticidad	254
7.7.6	Normalidad: gráfico Q-Q de los residuos	256
7.7.7	La decisión práctica	259

1 Introducción



El siguiente sitio web respalda los contenidos impartidos en la asignatura “Técnicas Multivariantes en Investigación de Mercados” y reproduce los cálculos estadísticos de las sesiones de organizadas dentro del máster oficial en Marketing e Investigación de mercados con R.

Así, este material organizado por sesiones, ayuda al alumno a ...

- introducir los conceptos básicos de R;
- ayudar a la carga de archivos;
- explicar los contenidos estadísticos además de aportar el código R para su cálculo;
- fijar el horizonte en la reproducibilidad de la investigación con el uso de **Quarto** (Allaire y Dervieux 2024).

Este es un documento escrito en lenguaje **markdown** y compilado usando la tecnología de **Quarto** (Allaire y Dervieux 2024), gestionado todo ello como un proyecto de Quarto en RStudio.

Si crees que no conoces **markdown**, estás equivocado. Lo estás usando sin darte cuenta desde hace años en múltiples sitios, pero hay uno que seguro usas y nunca lo habías pensado: WhatsApp. Prueba en un mensaje a escribir ****hola**** y verás el efecto. Negrita!!! o también **hola**, Cursiva!!!

El lenguaje *markdown* combina texto con *script* que ejecutan procesos. Estos *script* se localizan dentro de unidades de código de ejecución que se llaman *chunk* y que si el autor lo desea, se publican con un fondo gris para que el usuario los reconozca como tales. Un *chunk* se inicia con ````${r}```` al principio de la línea y se finaliza con ````` al principio de la línea también. Las líneas que quedan en medio de ambos marcadores (denominados *token*), forman el *script* de R. Los *token* (símbolos) de inicio y final del script, no se publican.

Esto que parece algo sin importancia, para los investigadores modernos es un lujo, porque podremos escribir y combinar textos, cuadros, gráficos y muchas cosas más en el mismo documento. No es el objetivo del curso aprender R, Rstudio, Quarto o Markdown, pero seguro que cuando llegue diciembre, te darás cuenta de que lo has aprendido sobre la marcha. Tu objetivo es analizar datos, y estas serán tus herramientas.

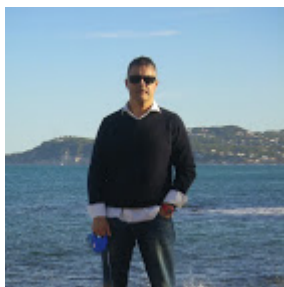
Sin más preámbulos, vamos allá. Comenzamos nuestro viaje. Vamos a descubrir esas herramientas de trabajo.

Publicación realizada utilizando Quarto 1.10.18.

1.1. Bienvenidos a este sitio web...

A diferencia de lo que suele ser habitual, y debido a las herramientas con las cuáles trabajamos, no vamos a disponer de las habituales diapositivas, sino que dispondremos de este sitio web donde se podrá ir alimentando toda la teoría y donde el alumno dispondrá de todos los materiales para poder trabajar.

1.2. ¿Quién soy ...?



Dirección de Proyectos de Análisis de Datos en TESI, doctor y profesor asociado en la Universitat de València (desde 1995), y en la Universidad Católica de Valencia (desde 2012).

Investigador y “blogger” en este sitio web InvestigaOnline.com.

Desde mis inicios en [TESI](#), he tenido la gran suerte de poder ejercitar y desarrollar todo aquello que pensamos era necesario para que los investigadores de mercado pudieran trabajar eficazmente. Además he tenido la inmensa fortuna de conocer a lo más granado de la IM en nuestro país, a aquellos que han hecho de esta industria mi pasión con sus consejos, sugerencias y advertencias. He desarrollado en TESI labores de marketing, de analista y de formación hasta que me introduje en el desarrollo de software como el nexo de unión entre la producción y el cliente final. Y de ahí han ido surgiendo Barbwin, Integra con sus módulos CatiNet, CawiNet y MobiNet, y ahora SegmentaNet, plataforma de análisis de datos y reporte gráfico online. En la actualidad, vuelta al análisis de datos y a la búsqueda de soluciones de visualización en analítica de datos.

Pero mi inquietud por la formación no desapareció nunca, y desde 1995 estoy como profesor asociado en la Universitat de València, fundamentalmente ejerciendo la docencia en el ámbito de la Investigación de Mercados y de la Distribución Comercial. Ahora también en la Universidad Católica de Valencia, comenzando un nuevo proyecto de “seducir” a los alumnos con este mundo de la investigación de mercados.

Y cuando todo lo demás me lo permite, me dedico a alimentar el blog blog.investigaonline.com. donde deixo mis opiniones, con la intención de ayudar a todo aquel que quiera saber sobre el mundo de la investigación de mercados y de forma más específica sobre la investigación online.

1.3. Vamos con la asignatura ...

Presentación y Evaluación de la asignatura (2026-2027)



- | | |
|---|--|
| <ul style="list-style-type: none">■ Asistencia<ul style="list-style-type: none">▶ 10%■ Casos prácticos<ul style="list-style-type: none">▶ Caso 1 - inferencia, 20%▶ Caso 2 - anaco, 15%▶ Caso 3 - afc, 15%▶ Caso 4 - cluster, 15%▶ Caso 5 - regresión, 15%■ Participación y colaboración en actividades junto con valoración personal del docente<ul style="list-style-type: none">▶ 10%■ Trabajo grupal<ul style="list-style-type: none">▶ Las actividades o casos se realizan en grupo.▶ Los grupos deberían ser de entre 4 y 5 personas.▶ Evaluación al grupo.■ Entrega de actividades<ul style="list-style-type: none">▶ Se realizará por medio de la plataforma virtual.▶ Se cerrará el plazo de entrega los jueves a las 23:59 | <ul style="list-style-type: none">■ Docencia<ul style="list-style-type: none">▶ 1ª sesión, introducción: mecánica de trabajo, herramientas (Quarto, R).▶ 2ª y 3ª sesiones inferencia, docencia teórica, exploración de datos.▶ 4ª a 8ª sesiones multivariante, con 2/3 aproximadamente de docencia teórica y 1/3 aproximadamente de docencia práctica de acompañamiento en el aula, lo cual no implica el posterior trabajo del alumno / grupo para completar las tareas■ Evaluación<ul style="list-style-type: none">▶ continúa a partir de las actividades indicadas con un 80% o más de asistencia▶ única por medio un examen final ...<ul style="list-style-type: none">— alumnos (o grupos de alumnos) que no hayan dispuesto al menos del 80% de asistencia y/o ...— alumnos (o grupos de alumnos) que no hayan superado los casos prácticos con una nota media de 5 o que al menos en dos casos tengan una nota por debajo de 4 puntos.■ Bibliografía<ul style="list-style-type: none">▶ Mirar en diapositivas por sesión |
|---|--|

Profesor/a: Roberto Gil

Este es el primer curso en el que migramos de la utilización de una aplicación que tradicionalmente se ha usado para el análisis de datos, el SPSS, pero que en la actualidad cada vez más está siendo sustituida por otras herramientas más específicas y de código abierto (*open source*) que de forma gratuita permiten el uso y cálculo de las más sofisticadas técnicas de proceso estadístico y que además, está siendo requisito en muchas contrataciones en el ámbito del proceso de datos. Y lo que es más importante en vuestra etapa de formación, usaréis las herramientas más buscadas en los perfiles de analista actuales.

La asignatura se estructura en base a:

- Una potente introducción estadística básica
- Una base aplicada sobre inferencia estadística
- Desarrollo de cuatro técnicas multivariantes básicas para un analista en Investigación en Marketing
 - Correspondencias
 - Componentes principales
 - Cluster o conglomerados
 - Regresión lineal múltiple

1.4. Introducción a las herramientas de trabajo

En esta primera sesión se acomete el trabajo básico de preparación del entorno de trabajo adecuado a nuestra necesidad de cálculo y escritura de informes:

1. instalar el *software* R;
2. instalar el *software* RStudio;
3. instalar los paquetes de trabajo;
4. cargar los bancos de datos necesarios para comenzar el trabajo;
5. acercarse al proceso de datos “real”, por medio de la tabulación.

Para comenzar la andadura en R son necesarias dos piezas fundamentales. **R** puede entenderse como **el motor** de un coche (realiza el trabajo duro: los cálculos y el análisis), mientras que **RStudio** es **el panel de control** o interfaz que permite la conducción el coche de forma cómoda.

1.4.1. Primer paso, instalar el lenguaje R

El lenguaje R es gratuito y de código abierto. Esta es una de las razones por las que las empresas están apostando por él.

1. Acceder a la página oficial: [CRAN \(Comprehensive R Archive Network\)](https://cran.r-project.org/).
2. Seleccionar el sistema operativo correspondiente:
 - **Download R for Windows:** hacer clic en “base” y luego en el enlace “Download R X.X.X for Windows”.
 - **Download R for macOS:** descargar el archivo `.pkg` correspondiente a la versión de procesador (en un Mac con chip Apple Silicon M1/M2/M3, la versión `arm64`; en un Mac con procesador Intel, la versión `x86_64`).
3. Ejecutar el instalador descargado y seguir los pasos por defecto (siguiente, siguiente, finalizar). No es necesario modificar ninguna configuración avanzada.

1.4.2. Segundo paso, instalar RStudio

RStudio es el entorno de desarrollo que facilita enormemente el trabajo diario. Es un proyecto empresarial, pero gratuito, lo que complementa muy bien a R y a otros lenguajes de programación como Python, Julia, Observable ...

1. Acceder a la página oficial: [Posit \(anteriormente RStudio\)](#).
2. Hacer clic en el botón Download RStudio Desktop correspondiente al sistema operativo.
3. Ejecutar el archivo descargado e instalarlo.
4. **Importante:** una vez instalado, abrir RStudio. Al ser la primera vez, el programa detectará automáticamente la versión de R instalada en el paso anterior.

Dar el salto a RStudio por primera vez puede parecer un poco desconcertante, sobre todo viniendo de la vista de datos tabular a la que acostumbran EXCEL / SPSS. Sin embargo, su diseño responde a una lógica clara y está pensado para facilitar el trabajo diario.

Al abrir RStudio y crear un nuevo *script* (mediante File > New File > R Script), la pantalla se divide en **cuatro cuadrantes principales**.

A continuación se describe cada uno de ellos:

1.4.2.1. Cuadrante superior izquierdo

Editor de Código (*Source*)

Es la mesa de trabajo principal: el lugar donde se escriben, editan y guardan los *scripts* (los archivos `.R` o los documentos `.qmd` de Quarto utilizados en esta asignatura).

- **¿Para qué sirve?** Para escribir el código de forma estructurada antes de ejecutarlo. Permite guardar el trabajo paso a paso, asegurando que el análisis sea 100% reproducible.
- **Dato útil:** también es el visor de datos. Al hacer clic en un *dataframe* cargado, se abre en esta ventana en formato de tabla (similar a la vista de datos de SPSS), aunque aquí es solo de lectura.

1.4.2.2. Cuadrante inferior izquierdo:

Consola (*Console*)

Es el “motor” interactivo de R: el lugar donde el código se ejecuta y cobra vida realmente.

- **¿Para qué sirve?** Al enviar una orden desde el Editor de Código (por ejemplo, calculando una media), el resultado de texto aparece aquí. También admite comandos rápidos escritos directamente en la consola, útiles para pruebas puntuales sin guardarlas en el *script*.
- **Dato útil:** este cuadrante incluye pestañas adicionales como *Terminal* (para comandos del sistema operativo) y *Background Jobs* (para procesos largos), aunque en el día a día del análisis de mercados la pestaña *Console* es la más relevante.

1.4.2.3. Cuadrante superior derecho:

Entorno (*Environment* / *History*)

Es la “memoria RAM” visual de la sesión de trabajo. En SPSS los datos se mostraban en una tabla frontal; en R, los datos se almacenan en el entorno, en objetos que no ven directamente, si no pides verlos.

- **¿Para qué sirve?** La pestaña *Environment* muestra un listado de todo lo cargado o creado en la sesión: bases de datos (*dataframes*), vectores, valores aislados y modelos estadísticos.
- **Dato útil:** desde aquí puede comprobarse rápidamente cuántas observaciones (filas) y variables (columnas) tiene cada base de datos. La pestaña *History* guarda un registro de los comandos ejecutados recientemente.

1.4.2.4. Cuadrante inferior derecho

El resto ... (*Files* / *Plots* / *Packages* / *Help*)

Este cuadrante es multiusos y resulta clave para la navegación y la visualización. Incluye varias pestañas de uso constante:

- **Files:** explorador de archivos integrado. Permite ver el contenido de la carpeta del proyecto, abrir *scripts* o localizar archivos de datos (*.sav*, *.csv*).
- **Plots:** el escaparate visual. Cada histograma, mapa de posicionamiento o dendrograma generado aparece aquí, y puede exportarse directamente como imagen o PDF.
- **Packages:** listado de todas las librerías instaladas (como *ggplot2* o *expss*), con la posibilidad de activarlas o actualizarlas desde aquí.

- **Help:** el salvavidas. Ante cualquier duda sobre el funcionamiento de un algoritmo o los parámetros de una prueba estadística, basta con escribir `?nombre_de_la_funcion` en la consola para que la documentación oficial se abra en esta pestaña.

1.4.3. Configuración inicial recomendada

Para evitar problemas futuros con la codificación de caracteres (especialmente con las tildes y la letra ñ), conviene configurar RStudio de la siguiente manera:

1. En el menú superior de RStudio, ir a **Tools > Global Options**.
2. En la pestaña **Code**, hacer clic en **Saving**.
3. Comprobar que en **Default text encoding** esté seleccionado **UTF-8**.
4. Pulsar **Apply** y **OK**.

1.5. ¿Cómo verificar que todo ha ido bien?

Para comprobar que todo se ha instalado correctamente, basta con abrir RStudio y localizar la **Consola** (la ventana de la izquierda o inferior izquierda), donde debería aparecer un texto de bienvenida con la versión de R.

Copiar y pegar la siguiente línea en la consola y pulsar *Enter*:

```
# Verify installation with a simple calculation  
2 + 2
```

```
[1] 4
```

Si el resultado que aparece es `[1] 4`, el entorno de trabajo está listo para comenzar.

1.5.1. El botón de ayuda en R

Es completamente normal encontrarse, al principio, con funciones cuyos parámetros resulten desconocidos o generen dudas. R dispone de un “botón de auxilio” integrado: al escribir un signo de interrogación ? pegado al nombre de cualquier función en la consola y pulsar *Enter*, se abre automáticamente la pestaña *Help* en el cuadrante inferior derecho de RStudio, con la documentación oficial, ejemplos de uso y detalles de los argumentos.

Por ejemplo, puede ejecutarse esta línea en la consola para comprobar cómo funciona el sistema de asistencia:

```
# Open help documentation for the mean function
?mean
```

1.6. Preguntas y respuestas

Antes de comenzar ...

¿Qué es R?

R es un entorno y lenguaje de programación con un enfoque al análisis estadístico. R nació como una reimplementación de *software* libre del lenguaje S, adicionado con soporte para ámbito estático. Se trata de uno de los pseudolenguajes de programación más utilizados en investigación científica, siendo además muy popular en los campos de aprendizaje automático (*machine learning*), minería de datos, investigación biomédica, bioinformática y matemáticas financieras. A esto contribuye la posibilidad de cargar diferentes bibliotecas o paquetes con funcionalidades de cálculo y de realización de gráficos.

¿Qué es RStudio?

RStudio es un entorno de desarrollo integrado (IDE) para el lenguaje de programación R, dedicado a la computación estadística y gráficos. Incluye una consola, editor de sintaxis que apoya la ejecución de código, así como herramientas para el trazado, la depuración y la gestión del espacio de trabajo. RStudio está disponible para Windows, Mac y Linux o para navegadores conectados a RStudio Server o RStudio Server Pro (Debian / Ubuntu, RedHat / CentOS, y SUSE Linux). RStudio tiene la

misión de proporcionar el entorno informático estadístico R. Permite un análisis y desarrollo para que cualquiera pueda analizar los datos con R.

¿Qué es un *script* de R base (.R)?

Los *scripts* son documentos de texto con la extensión de archivo .R, por ejemplo `mi_script.R`. Estos archivos son iguales a cualquier documento de texto, pero R los puede leer y ejecutar el código que contienen.

¿Qué es un vector?

Los vectores en R son objetos de una única dimensión que puede contener datos numéricos, cadena de caracteres o datos lógicos, entre otros. Esencialmente son uno de los elementos básicos en la estructura de los datos en R. Se crean con la estructura `c()`.

¿Qué es una matriz?

Una matriz en R es un conjunto de elementos del mismo tipo (numérico, carácter, lógico, etc.) organizado en filas y columnas. Las matrices en R se construyen con la función `matrix()`. Aunque con un ejemplo siempre es mucho más fácil comprender cómo funcionan las matrices.

¿Qué es un *dataframe* / *dataset*?

Un *dataframe* es lo que se conoce como un fichero de datos (*dataset*). Son estructuras para trabajar con datos de diferentes tipos (cadena, lógicos, aritméticos). Utiliza una estructura de datos tabular (como una matriz) pero que permite manipular distintos tipos de datos, de modo que una columna puede contener caracteres, otra números y otra variables lógicas. Son importantes para hacer tablas, cuadros, gráficas, análisis y modelos que tienen muchas variables estadísticas. Se crean con la estructura `data.frame()`. Mientras que el término *dataframe* hace referencia al nombre de la estructura en R, el concepto *dataset* hace referencia al conjunto de datos, archivo, fichero, etc.

i Analogía con la encuesta: vectores y matrices o *dataframes*

Dejando a un lado la abstracción informática, estos elementos pueden conectarse con la investigación de mercados tradicional. Un **vector** en R es el equivalente exacto a una columna entera de respuestas de un cuestionario (por ejemplo, las respuestas de todos los entrevistados a la pregunta de “edad”).

Por su parte, un ***dataframe*** no es más donde todas esas columnas (variables)

conviven alineadas fila por fila, representando cada fila a un individuo o consumidor diferente.

¿Qué es un paquete / librería?

Un paquete (*package*) es una colección de funciones, datos y código R que se almacenan en una carpeta conforme a una estructura bien definida, fácilmente accesible para R. En la web de R se puede consultar la lista de paquetes disponibles. Un paquete contiene funciones que las usamos para hacer nuestros cálculos.

¿Qué es un objeto?

La información que se manipula en R se estructura en forma de objetos. Para trabajar con R resulta importante conocer los principales tipos de objetos y sus propiedades básicas. En general, cada tipo de objeto viene definido por una serie de atributos. Las funciones genéricas (como por ejemplo `summary` o `plot`) reconocen estos atributos y llevan a cabo distintos tipos de acciones en función del tipo de objeto.

¿Qué es Markdown?

Markdown es un lenguaje de marcado ligero creado por John Gruber que trata de conseguir la máxima legibilidad y facilidad de publicación tanto en su forma de entrada como de salida, inspirándose en muchas convenciones existentes para marcar mensajes de correo electrónico usando texto plano. Markdown convierte el texto marcado en documentos XHTML utilizando `html2text` creado por Aaron Swartz. Para conocer los rudimentos del lenguaje puede consultarse el [siguiente enlace](#).

¿Qué es un documento Quarto (.qmd) con *scripting*?

Quarto es una herramienta usada por RStudio que permite una fácil creación de documentos, presentaciones dinámicas e informes de R. Markdown es un formato de sintaxis simple en documento de texto para crear documentos en HTML, PDF, y Word.

¿Qué es un *chunk*?

Los trozos de código R que se insertan en archivos markdown se denominan *chunk* y permiten hacer análisis estadísticos y mostrar los resultados en el documento final. Los *chunks* tienen diversas opciones que permiten una mayor flexibilidad en cómo se muestra el código y los resultados en el documento final.

¿Qué es un archivo HTML *self-contained*?

Los ficheros HTML *self-contained* son ficheros autónomos que residen en un solo archivo HTML. No puede incluir ningún otro archivo y deben funcionar sin una conexión de red. Un usuario debería poder guardar el archivo, abrirlo localmente y tener todo listo para trabajar.

```
# This is a minimal chunk: just one line of R code  
print('_aquí iría el código_')
```

```
[1] "_aquí iría el código_"
```

¿Qué es *inline code*?

A diferencia de los *chunks*, el *inline code* se inserta en el texto del archivo; de este modo se inserta aquí la fecha: 2026-10-25.

1.7. Instalar y configurar el entorno de trabajo

Con R y RStudio ya instalados, queda preparar el entorno en el que se trabajará durante todo el curso. Son solo dos pasos, y ambos consisten en copiar un bloque de código, pegarlo en la **consola** de RStudio y pulsar **Intro**:

1. **Crear la carpeta del curso** (una sola vez): crea la carpeta TMIM, descarga en ella los *scripts* y los datos del curso y la abre como proyecto de RStudio.
2. **Instalar y cargar los paquetes** (al comenzar cada sesión): instala los paquetes que falten y los carga en memoria.

! Antes de empezar

No crear carpetas a mano, no descomprimir archivos y no mover nada: los bloques de código se encargan de todo. Trabajar siempre con la carpeta que se crea en el paso 1, y nunca dentro de OneDrive, iCloud, Google Drive o similares.

1.7.1. Paso 1. Crear la carpeta del curso

Copiar el bloque completo con el icono  *copy* (esquina superior derecha), pegarlo en la consola y pulsar **Intro**.

```

# PROJECT FOLDER SETUP

↪ -----

# Copy this whole block into the RStudio Console and press
↪ Enter.

# It creates the same course folder on every computer (Windows,
↪ macOS, Linux),
# outside OneDrive/iCloud, downloads the course scripts and data
↪ into it
# and opens it as an RStudio project.
# It can be run again later to get new files: existing files are
↪ never overwritten.

project_name <- "TMIM"
site_url      <- "https://tmim2026.investigaonline.com/descargas"

# 1. Find the user's home folder (on Windows, avoid the OneDrive
↪ "Documents")
os_name       <- Sys.info()[["sysname"]]
home_dir      <- if (os_name == "Windows")
↪ Sys.getenv("USERPROFILE") else Sys.getenv("HOME")
home_dir      <- normalizePath(home_dir, winslash = "/", mustWork
↪ = TRUE)
project_dir   <- file.path(home_dir, project_name)

# 2. Create the project folder and its subfolders
for (sub_dir in c("", "data", "scripts", "output")) {
  dir.create(file.path(project_dir, sub_dir), showWarnings =
↪ FALSE, recursive = TRUE)
}

# 3. Create the RStudio project file (TMIM.Rproj) if it does not
↪ exist
rproj_file    <- file.path(project_dir, paste0(project_name,
↪ ".Rproj"))
if (!file.exists(rproj_file)) {
  writeLines(c(

```

```

    "Version: 1.0", "",
    "RestoreWorkspace: No",
    "SaveWorkspace: No",
    "AlwaysSaveHistory: Default", "",
    "EnableCodeIndexing: Yes",
    "UseSpacesForTab: Yes",
    "NumSpacesForTab: 2",
    "Encoding: UTF-8"
  ), rproj_file)
}

# 4. Download and unzip the course files (zip name ->
  ↪ destination folder)
course_zips <- c(
  "tmim-scripts.zip" = file.path(project_dir, "scripts"), #
  ↪ flat .R files
  "tmim-data.zip"     = project_dir                        #
  ↪ contains data/
)

for (zip_name in names(course_zips)) {
  zip_file <- file.path(tempdir(), zip_name)
  ok <- tryCatch({
    download.file(paste0(site_url, "/", zip_name), zip_file,
      ↪ mode = "wb", quiet = TRUE)
    TRUE
  }, error = function(e) FALSE, warning = function(w) FALSE)

  if (ok) {
    # overwrite = FALSE keeps the student's edited files
    ↪ (warnings just list skipped files)
    suppressWarnings(unzip(zip_file, exdir =
      ↪ course_zips[[zip_name]], overwrite = FALSE))
    message("Downloaded: ", zip_name)
  } else {

```

```

    message("Could not download ", zip_name, ". Check the
    ↪ internet connection and run this block again.")
  }
}

message("Project folder ready: ", project_dir)

# 5. Open the project in RStudio (RStudio will restart inside
    ↪ the project)
if (!requireNamespace("rstudioapi", quietly = TRUE))
  ↪ install.packages("rstudioapi")
if (rstudioapi::isAvailable()) {
  rstudioapi::openProject(project_dir)
} else {
  message("Open RStudio and use File > Open Project... to open:
  ↪ ", rproj_file)
}

```

Al terminar, RStudio se reinicia y abre el proyecto. Si pregunta si se desea guardar el espacio de trabajo (*Save workspace image*), responder **Don't Save**.

Para comprobar que todo ha ido bien:

- en la esquina superior derecha de RStudio aparece el nombre del proyecto, **TMIM**;
- en la pestaña **Files** (cuadrante inferior derecho) se ven el archivo **TMIM.Rproj** y las carpetas **data**, **scripts** y **output**.

Se obtiene así la siguiente estructura, idéntica en todos los ordenadores:

TMIM/

TMIM.Rproj	← archivo del proyecto
data/	← archivos de datos del curso
scripts/	← global.R y los scripts de cada sesión
output/	← resultados que se generen (tablas, gráficos...)

La carpeta TMIM se crea dentro de la carpeta personal del usuario: C:\Users\nombre\TMIM en Windows y /Users/nombre/TMIM en macOS.

Nuevos materiales durante el curso

Este bloque puede ejecutarse de nuevo en cualquier momento para descargar los *scripts* de las sesiones que se vayan publicando. Los archivos que ya existen en la carpeta no se sobrescriben, de modo que los cambios propios no se pierden.

1.7.2. Paso 2. Instalar y cargar los paquetes

R incorpora de serie las funciones básicas, pero la mayor parte de las técnicas del curso requieren **paquetes** adicionales. El *script* `global.R` (ya descargado en la carpeta `scripts`) se encarga de instalar los que falten y de cargarlos todos.

1. Reiniciar R desde el menú **Session > Restart R**.
2. Copiar esta línea en la consola y pulsar **Intro**:

```
# Install (if needed) and load every package used in the course
source("scripts/global.R")
```

La primera vez tarda unos minutos, porque descarga e instala todos los paquetes. Las siguientes veces solo los carga y tarda unos segundos. Si todo va bien, aparece el mensaje:

```
Environment ready! All packages are installed and loaded.
```

Si aparece un mensaje de error

Si algún paquete no se instala correctamente, `global.R` muestra un recuadro titulado **HOW TO FIX IT** con los pasos a seguir y una línea del tipo `install.packages(c("..."))` preparada para copiar y pegar en la consola. Basta con seguir esos pasos y volver a ejecutar `source("scripts/global.R")`. Si el error persiste, enviar al profesor una captura de pantalla del mensaje completo.

1.7.3. Paquetes utilizados en la asignatura

El contenido completo de `global.R` se muestra a continuación a título informativo. No es necesario copiarlo: el archivo ya está en la carpeta `scripts`.

```

# ENVIRONMENT SETUP SCRIPT
↪ -----

# How to run it:
# 1. Restart R first: Session > Restart R (so no package is in
↪ use)
# 2. In the Console: source("scripts/global.R")
# The script installs missing packages one by one, loads them
↪ all and,
# if something fails, prints the exact line you need to run to
↪ fix it.

# 1. Packages required for the course
↪ -----

required_packages <- c(
  "car", "psych", "Hmisc", "ca", "FactoMineR", "factoextra",
  ↪ "readr", "readxl",
  "tidyverse", "kableExtra", "sparkline", "fontawesome",
  ↪ "outliers", "nortest",
  "vcd", "corrplot", "gplots", "DT", "lmtest", "sjstats",
  ↪ "igraph", "smacof",
  "maditr", "expss", "NbClust", "effectsize", "GGally",
  ↪ "gtsummary", "mclust",
  "ggrepel", "lm.beta", "ppcor", "htmltools"
)

options(repos = c(CRAN = "https://cloud.r-project.org"))
os_name <- Sys.info()[["sysname"]] # "Windows", "Darwin"
↪ (macOS) or "Linux"

# 2. System checks
↪ -----

# Old R versions get no ready-made (binary) packages: most macOS
↪ errors come from here
if (getRversion() < "4.4.0") {
  warning("Your R version (", getRversion(), ") is too old. ",

```

```

        "Install the latest R from https://cran.r-project.org
        ↪ and run this script again.",
        call. = FALSE, immediate. = TRUE)
}

# Apple Silicon Mac running the Intel build of R (a frequent
↪ installation mistake)
if (os_name == "Darwin" && R.version$arch == "x86_64") {
  translated <- suppressWarnings(
    system("sysctl -n sysctl.proc_translated", intern = TRUE,
    ↪ ignore.stderr = TRUE)
  )
  if (identical(translated, "1")) {
    warning("You installed the Intel version of R on an Apple
    ↪ Silicon Mac (M1/M2/M3/M4). ",
            "Uninstall it and install the 'arm64' version from
            ↪ https://cran.r-project.org.",
            call. = FALSE, immediate. = TRUE)
  }
}

# Windows and macOS: use ready-made binaries, never compile from
↪ source
if (os_name %in% c("Windows", "Darwin")) {
  options(install.packages.compile.from.source = "never")
}

# 3. Helper: extract package names quoted inside an
↪ error/warning message -----
find_culprits <- function(msg_text) {
  pattern <- "[\'\\\"]([A-Za-z][A-Za-z0-9.]+)[\'\\\"]"
  hits <- regmatches(msg_text, gregexpr(pattern, msg_text))
  hits <- gsub("[\'\\\"]", "", unlist(hits))
  unique(hits[nchar(hits) > 1])
}

```

```

# 4. Install missing packages one by one
↪ -----
missing_packages <- setdiff(required_packages,
↪   rownames(installed.packages()))
failed_packages <- character(0)
culprit_packages <- character(0)
error_log        <- character(0)

if (length(missing_packages) > 0) {
  message("Installing ", length(missing_packages), " package(s).
↪   This may take a few minutes...")

  for (pkg in missing_packages) {
    message(" - ", pkg)
    collected <- character(0)

    withCallingHandlers(
      tryCatch(
        install.packages(pkg, quiet = TRUE),
        error = function(e) collected <- c(collected,
↪       conditionMessage(e))
      ),
      warning = function(w) {
        collected <- c(collected, conditionMessage(w))
        invokeRestart("muffleWarning")
      }
    )

    # The real test: is the package installed now?
    if (!pkg %in% rownames(installed.packages())) {
      failed_packages <- c(failed_packages, pkg)
      culprit_packages <- c(culprit_packages,
↪   find_culprits(collected))
      error_log        <- c(error_log, paste0("[install ", pkg,
↪   "] ", collected))
    }
  }
}

```

```

    }
  }

# 5. Load all packages one by one
↪ -----
for (pkg in setdiff(required_packages, failed_packages)) {
  load_error <- tryCatch({
    suppressPackageStartupMessages(suppressWarnings(library(pkg,
↪   character.only = TRUE)))
    NULL
  }, error = function(e) conditionMessage(e))

  if (!is.null(load_error)) {
    failed_packages <- c(failed_packages, pkg)
    culprit_packages <- c(culprit_packages,
↪   find_culprits(load_error))
    error_log <- c(error_log, paste0("[load ", pkg, "] ",
↪   load_error))
  }
}

# 6. Final report
↪ -----
if (length(failed_packages) == 0) {
  message("\n Environment ready! All packages are installed and
↪   loaded.")
} else {
  # Packages to reinstall: the failed ones plus any dependency
  ↪   named in the errors
  to_reinstall <- unique(c(failed_packages, culprit_packages))
  to_reinstall <- to_reinstall[to_reinstall %in%
↪   rownames(available.packages())]
  fix_line <- paste0('install.packages(c("'", paste(to_reinstall,
↪   collapse = "'", '"'), '"))')

```

```

↪ cat("\n=====
cat("  Some packages could not be installed or loaded:\n  ",
    paste(unique(failed_packages), collapse = ", "), "\n\n")
cat("HOW TO FIX IT:\n")
cat("  1. Restart R: Session > Restart R\n")
cat("  2. Copy and run this line in the Console:\n\n")
cat("    ", fix_line, "\n\n")
cat("  3. Run the script again:
↪ source(\"scripts/global.R\")\n")
cat("  4. If it still fails, send a screenshot of this message
↪ to the teacher.\n")

if (os_name == "Linux") {
  cat("\nLINUX (Ubuntu/Debian): some packages need system
↪ libraries. In a terminal:\n")
  cat("  sudo apt install libcurl4-openssl-dev libssl-dev
↪ libxml2-dev libfontconfig1-dev \\\n")
  cat("    libharfbuzz-dev libfribidi-dev libfreetype6-dev
↪ libpng-dev libtiff5-dev \\\n")
  cat("    libjpeg-dev libglpk-dev libgmp3-dev\n")
}

↪ cat("=====

# Full technical details, in case the teacher needs them
cat("\nTechnical details:\n")
cat(paste0("  ", unique(error_log), collapse = "\n"), "\n")
}

```

1.7.4. Archivos de datos

Los archivos de datos se descargan automáticamente en la carpeta **data** del proyecto durante el paso 1. Si por algún motivo la descarga hubiera fallado, pueden obtenerse también desde este sitio web o desde el Aula Virtual (**tmim-data.zip**); en ese caso, se debe descomprimir el archivo dentro de la carpeta **TMIM**, de modo que los datos

queden en **TMIM/data**.

Gracias al proyecto de RStudio, los datos se leen siempre con rutas relativas del tipo `"data/archivo.sav"`, iguales en todos los ordenadores.

Los archivos están en formato SPSS (**SAV**) porque es un formato tradicional en los ficheros de investigación. Aunque podrían transformarse, se ha optado por mantener este formato, ya que en la práctica profesional del análisis de mercados es habitual encontrarse con archivos **.sav**. A lo largo de la asignatura se mostrará también cómo cargar ficheros en otros formatos habituales, como **XLSX**, **CSV**, **TSV** o el formato estándar de R, **RDS**.

1.7.5. La rutina de cada sesión

Una vez configurado el entorno, el inicio de cada sesión de trabajo se reduce a:

1. Abrir RStudio. Se abre automáticamente el último proyecto utilizado (**TMIM**); si no fuera así, hacer doble clic sobre el archivo **TMIM.Rproj** o usar el menú **File > Recent Projects**.
2. Ejecutar en la consola `source("scripts/global.R")`.
3. Abrir el *script* de la sesión desde la pestaña **Files** (carpeta **scripts**) y comenzar a trabajar.

1.8. Practicando el *scripting*

1.8.1. Ejercicio: *script* 1: consola

Aunque se puede trabajar en consola, es más habitual trabajar con ficheros de instrucciones o *scripts*.

Se proponen a continuación estas líneas para escribir en la consola, una a una; al completar cada línea, se pulsa la tecla **INTRO** (también llamada **ENTER**). El resultado de la operación aparece directamente en la consola.

Por ejemplo:

- escribir `mean(2,3,4,5)` y pulsar **INTRO**.
- escribir `max(2,3,4,5)` y pulsar **INTRO**.

¿Qué está pasando? R no va bien fallos en la media y el máximo, ¿qué pasa aquí...? Igual la sintaxis no es la correcta.

Probar esto:

- escribir `mean(c(2,3,4,5))` y pulsar INTRO.
- escribir `max(c(2,3,4,5))` y pulsar INTRO.

Vaya, se recuperó

¿Qué ha pasado? R en ocasiones necesita una serie de valores, y no un solo valor. Para identificar una serie de valores se utiliza la función `c()`, que separa por comas los valores a trabajar. Se continúa...

1.8.2. Ejercicio: *script* 2: vectores y dataframes

Un primer ejemplo más completo para iniciarse en R consiste en crear el siguiente *script*, donde:

- se crean dos vectores
- se publica su contenido
- se unen esos dos vectores creando un *dataframe*
- se publica el *dataframe*
- se muestra la estructura del *dataframe*
- se calcula la media aritmética del vector (ya se hizo anteriormente)
- se calcula la media aritmética del campo correspondiente en el *dataframe*

El *script* se escribe ahora no en la consola, sino en un archivo nuevo: `file > new file > R script`

Se abre un lienzo en blanco.

Las siguientes líneas pueden copiarse con el icono  *copy* situado en la esquina superior derecha, y pegarse en ese lienzo en blanco.

```
# Create two vectors
x <- c(1, 2, 3, 1, 3, 1, 3, 3, 3)
y <- c(2, 1, 3, 4, 1, 2, 4, 1, 1)

# Create a dataframe
# where vector x will be called V1 and
```

```

# vector y will be called V2
df1 <- data.frame(V1 = x, V2 = y)

# Display the dataframe
cat("Dataframe: \n")
print(df1)

# Inspect its structure
cat("\n Dataframe structure: \n")
str(df1)

# Calculate means: of the vector and of the column
# corresponding to vector x in the dataframe to see
# that the result is the same

cat("\n Vector mean: ")
mean(x)

cat("\n Mean of column V1 in the dataframe: ")
mean(df1$V1)

```

Dataframe:

	V1	V2
1	1	2
2	2	1
3	3	3
4	1	4
5	3	1
6	1	2
7	3	4
8	3	1
9	3	1

Dataframe structure:

'data.frame': 9 obs. of 2 variables:

\$ V1: num 1 2 3 1 3 1 3 3 3

\$ V2: num 2 1 3 4 1 2 4 1 1

```
Vector mean: [1] 2.222222
```

```
Mean of column V1 in the dataframe: [1] 2.222222
```

El botón Run, situado en la parte superior derecha, permite ejecutar las líneas del fichero una a una; seleccionándolas todas, la opción Run all encadena la ejecución de todas ellas, una tras otra.

A partir de ahora, los sucesivos *scripts* incluirán menos explicación paso a paso: bastará con copiar, pegar y analizar qué instrucciones se están ejecutando.

1.8.3. Ejercicio: *script* 3: valores perdidos

Este segundo caso permite analizar la influencia de los valores perdidos (*missing values*) y su efecto en el cálculo de estimaciones estadísticas. En concreto:

- se crean de nuevo los vectores, añadiendo un valor perdido (NA)
- se publica su contenido
- se unen esos dos vectores creando un *dataframe*
- se publica el *dataframe*
- se repiten los cálculos estadísticos anteriores con una leve modificación

```
# --- Add a missing value (NA) to each vector ---
x <- c(x, NA)
y <- c(y, NA)

df1 <- data.frame(V1 = x, V2 = y)

# --- Mean calculation comparison ---
# By default, R cannot calculate a mean if any value is missing
↪ (NA):
# the result of any operation involving NA is NA,
↪ "contamination" by default
cat("Media de V1 con missing values: \n")
mean(df1$V1) # Will return NA

cat("\n--- *** ---\n")
```

```
# 'na.rm = TRUE' tells R to remove (ignore) NA values before
↪ calculating
cat("Media de V1 ignorando los missing values (na.rm = TRUE):
↪ \n")
mean(df1$V1, na.rm = TRUE) # Correct calculation
```

Media de V1 con missing values:

```
[1] NA
```

```
--- *** ---
```

Media de V1 ignorando los missing values (na.rm = TRUE):

```
[1] 2.222222
```

1.8.4. Ejercicio: *script 4: funciones generales*

Funciones de inspección (para conocer los datos)

Estas funciones evitan tener que abrir el *dataframe* a ciegas.

- `head(df1)`: Muestra las primeras 6 filas. Es el estándar para comprobar si los datos se han cargado bien.
- `tail(df1)`: Muestra las últimas 6 filas.
- `str(df1)`: (Estructura) Es, probablemente, la función más importante. Indica si una columna es numérica, texto (*character*) o categórica (*factor*).
- `glimpse(df1)`: (Del paquete *tidyverse/dplyr*) Es una versión más moderna y limpia de `str()`. Se recomienda encarecidamente para quienes empiezan, porque resulta mucho más fácil de leer.
- `summary(df1)`: Genera un resumen estadístico básico (media, mediana, mínimos y máximos) de cada columna. Es el sustituto rápido de los cuadros de estadísticos de SPSS.

Funciones de dimensiones y nombres

- `ncol(df1)`: Devuelve el número de columnas (útil cuando se combina con `nrow()` para saber el tamaño total).
- `dim(df1)`: Devuelve un vector con dos valores, n° filas / n° columnas.

- `names(df1)`: Es un alias de `colnames(df1)`. Es muy común encontrarlo en la literatura de R.

Funciones de sistema/entorno

- `ls()`: Lista todos los objetos cargados en la memoria actual del entorno; permite comprobar si hay basura acumulada.
- `rm(objeto)`: Para eliminar un objeto (limpiar la memoria).
- `getwd()`: (*get working directory*) Fundamental. Indica en qué carpeta está trabajando R. Sin esa referencia no es posible importar ni guardar archivos correctamente.

```
# --- Exploratory functions ---

# System info
Sys.time()      # Current time
Sys.Date()      # Current date
getwd()         # Check your working directory
ls()            # List all objects currently in memory

# Dataframe dimensions
nrow(df1)       # Number of rows
ncol(df1)       # Number of columns
dim(df1)        # Dimensions (rows, columns)

# Dataframe content
colnames(df1)   # Variable names
summary(df1)    # Basic statistics for all variables
head(df1, 3)    # First 3 rows
tail(df1, 3)    # Last 3 rows
```

```
[1] "2026-10-25 08:38:58 CET"
[1] "2026-10-25"
[1] "C:/Users/rober/Documents/R/rgs.tmim.sesiones"
[1] "culprit_packages"  "df1"          "error_log"
[4] "failed_packages"  "find_culprits" "load_error"
[7] "missing_packages" "os_name"       "pkg"
[10] "required_packages" "x"             "y"
```

```

[1] 10
[1] 2
[1] 10 2
[1] "V1" "V2"

      V1      V2
Min.   :1.000 Min.   :1.000
1st Qu.:1.000 1st Qu.:1.000
Median :3.000 Median :2.000
Mean   :2.222 Mean   :2.111
3rd Qu.:3.000 3rd Qu.:3.000
Max.   :3.000 Max.   :4.000
NAs    :1      NAs    :1

V1 V2
1  1  2
2  2  1
3  3  3

V1 V2
8  3  1
9  3  1
10 NA NA

```

1.8.5. Ejercicio: Mi primer Quarto document

A continuación se realiza un nuevo experimento. A diferencia de los archivos `.R` (que son puras instrucciones), el formato *Quarto* (`.qmd`) es el estándar para crear informes reproducibles. Este tipo de archivos combina texto explicativo (en formato Markdown) con trozos de código (*chunks*) que se ejecutan al generar el documento.

Conviene explorar las plantillas que RStudio ofrece:

- **File → New File → Quarto Document → Document → HTML** (ver variantes)
- **File → New File → Quarto Presentation → Presentation → ioslides** (ver variantes)

Consejo: conviene *renderizar* el documento resultante para ver cómo el código y el texto se convierten en un informe profesional.

1.8.6. Ejercicio: *script 5*: carga de datos

Aunque RStudio cuenta con menús visuales para importar datos, en el análisis profesional se prefiere cargar los datos mediante *scripts*, lo que garantiza que el proceso sea trazable y reproducible. Se utiliza el paquete **readr** para archivos de texto y el paquete **expss** para archivos **.sav** de SPSS.

Carga desde CSV

```
# Note: library(readr) is not loaded here because
  ↳ source('global') has already
# been executed and it loads all the required packages

# Practice dataset hosted externally (Google Drive), used only
  ↳ for this exercise
# In later sessions we will load files from our local 'data/'
  ↳ project folder instead
# suppressMessages() hides the column-type detection message
  ↳ that read_csv prints by default
data <-
  ↳ read_csv("https://drive.google.com/uc?export=download&id=10StFMmg5fzIpfTZnzX9

# Show dataframe
cat('Dataframe: /n')
data

# Show 2 head rows
cat('Dataframe (2 head): /n')
head(data, 2)

#Show 2 tail rows
cat('Dataframe (2 tail): /n')
tail(data, 2)
```

Dataframe: /n# A tibble: 6 x 11

	x	y	z	low	high	value	name	color	from	to	weight
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>	<chr>	<chr>	<chr>	<dbl>
1	0	1.6	-34	-6	9.2	1	lemon	#d35400	lemon	olive	1

2	1	11	-23	6.7	15.3	10	nut	#2980b9	lemon	guava	1
3	2	20.4	6.8	2.8	38	19	olive	#2ecc71	lemon	fig	1
4	3	22.1	32.3	19.4	24.8	21	guava	#f1c40f	nut	olive	1
5	4	15.4	27.7	12.1	18.7	14	fig	#2c3e50	olive	pear	2
6	5	7.4	3.2	-11.8	26.6	6	pear	#7f8c8d	guava	pear	2

Dataframe (2 head): /n# A tibble: 2 x 11

	x	y	z	low	high	value	name	color	from	to	weight
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>	<chr>	<chr>	<chr>	<dbl>
1	0	1.6	-34	-6	9.2	1	lemon	#d35400	lemon	olive	1
2	1	11	-23	6.7	15.3	10	nut	#2980b9	lemon	guava	1

Dataframe (2 tail): /n# A tibble: 2 x 11

	x	y	z	low	high	value	name	color	from	to	weight
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>	<chr>	<chr>	<chr>	<dbl>
1	4	15.4	27.7	12.1	18.7	14	fig	#2c3e50	olive	pear	2
2	5	7.4	3.2	-11.8	26.6	6	pear	#7f8c8d	guava	pear	2

Carga desde ficheros SAV

```
# Note: library(expss) is not loaded here because
↪ source('global') has already
# been executed and it loads all the required packages

# Same idea, but reading an SPSS (.sav) file directly from a URL
# read_spss() comes from the 'expss' package and preserves
↪ variable/value labels
data <-
↪ read_spss("https://drive.google.com/uc?export=download&id=11q4pg2iWwWdV9mk5P4")

# Preview the first and last 5 rows
# Preview 2 head rows
cat('Dataframe (5 first): /n')
head(data, 5)

# Preview 2 tail rows
cat('Dataframe (5 last): /n')
tail(data, 5)
```

Dataframe (5 first): /n ESTU CUES CCAA PROV MUN TAMUNI CAPITAL DISTR SECCION ENTRE

1	3192	1	16	1	59	5	1	0	0	0	3	9	3	6	1
2	3192	2	16	1	59	5	1	0	0	0	3	2	2	7	1
3	3192	3	16	1	59	5	1	0	0	0	3	2	2	8	1
4	3192	4	16	1	59	5	1	0	0	0	3	9	2	8	1
5	3192	5	16	1	59	5	1	0	0	0	3	3	2	7	1
P402 P403 P404 P501 P502 P503 P504 P6 P6A01 P6A02 P6B P6C P6D P7 P7A P801															
1	1	1	1	7	7	8	8	1	2	0	1	2	3	3	1
2	1	1	1	7	7	7	7	1	1	0	0	2	3	1	NA
3	1	1	1	8	8	8	8	1	2	0	2	2	2	3	1
4	1	1	1	9	9	9	1	1	1	0	0	1	3	2	NA
5	1	1	1	9	5	8	8	1	2	0	1	2	3	4	1
P802 P803 P8A01 P8A02 P8A03 P8B01 P8B02 P8B03 P901 P902 P903 P904 P905 P906															
1	2	1	2	0	1	96	96	8	7	7	7	7	7	6	6
2	1	1	2	2	1	96	96	7	7	7	7	7	7	7	7
3	1	1	1	2	1	10	96	10	8	8	8	8	8	6	8
4	2	1	1	0	2	9	96	96	10	98	10	98	5	7	7
5	1	1	2	2	2	96	96	96	8	8	9	9	6	8	8
P907 P908 P1001 P1002 P1003 P11 P11A01 P11A02 P11B P11C P11D P12 P12A01															
1	6	7	3	2	3	2	96	96	0	0	0	2	96		
2	7	1	3	2	3	2	96	96	0	0	0	1	2		
3	8	2	2	3	3	2	96	96	0	0	0	1	1		
4	7	3	2	3	2	2	96	96	0	0	0	1	1		
5	8	4	2	3	2	2	96	96	0	0	0	1	1		
P12A02 P12B P12C P12D01 P12D02 P12D P1301 P1302 P1303 P1304 P1305 P14 P14A01															
1	96	0	0	NA	NA	NA	6	7	7	6	3	2	96		
2	0	2	3	97	97	9997	7	7	7	7	1	2	96		
3	0	2	3	25	0	25	7	7	7	7	2	2	96		
4	0	1	3	0	1	30	9	9	10	10	3	2	96		
5	0	2	3	0	4	120	5	6	8	8	4	2	96		
P14A02 P14B P14C P1501 P1502 P1503 P1504 P1505 P1506 P16 P17 P18 P18A P18B															
1	96	0	0	7	7	5	5	6	2	3	2	1	8	1	
2	96	0	0	7	7	9	7	7	1	4	8	2	0	0	
3	96	0	0	7	7	7	7	7	2	2	3	2	0	0	
4	96	0	0	8	10	4	8	9	3	2	8	1	10	2	
5	96	0	0	5	10	7	5	98	4	8	2	1	10	2	
P18C01 P18C02 P18C03 P18C04 P18C05 P18C06 P18C07 P18C08 P19 P20 P21 P21A01															
1	NA	NA	NA	1	NA	NA	NA	NA	2	2	2	0			

2	0	0	0	0	0	0	0	0	0	2	2	2	0									
3	0	0	0	0	0	0	0	0	0	2	2	2	0									
4	0	0	0	0	0	0	0	0	0	3	2	2	0									
5	0	0	0	0	0	0	0	0	0	2	2	2	0									
P21A02 P21A03 P21A04 P21B P22 P23 P24 P2501 P2502 P2503 P2601 P2602 P2603																						
1	0	0	0	0	3	1	2	10	10	8	1	2	2									
2	0	0	0	0	1	1	1	10	6	4	2	9	9									
3	0	0	0	0	1	8	2	10	10	10	1	2	2									
4	0	0	0	0	3	2	2	10	10	10	1	2	2									
5	0	0	0	0	1	1	2	10	8	10	2	2	2									
P2604 P27 P28 P29 P30 P30A P30AR P31 P32 P33 P34 P35 P35A P36 P37 P38 P39 P40																						
1	1	2	3	99	1	99	99	2	51	1	1	3	5	2	1	1	2	1				
2	2	1	3	2	1	3	3	1	37	1	1	3	13	1	2	1	1	1				
3	1	1	2	4	1	99	99	2	60	1	2	3	7	2	2	4	3	2				
4	1	1	1	2	1	98	98	1	53	1	1	3	11	1	1	4	1	1				
5	1	2	3	1	1	3	3	2	69	3	2	3	4	2	2	4	1	2				
P41 P42 P42A P43 P44 P45 P46 P46A P46B P46C P46D P47 P47A P47B P48 P4901																						
1	561	2	1	87	99	99	1	1	0	0	0	1	16	0	1	NA						
2	215	4	0	86	7	6	1	1	0	0	0	1	16	0	3	NA						
3	351	1	3	35	10	7	1	1	0	0	0	1	16	0	1	NA						
4	223	1	1	85	9	7	1	1	0	0	0	1	16	0	1	NA						
5	921	1	3	81	5	5	1	1	0	0	0	1	14	0	1	NA						
P4902 P4903 P4904 P5001 P5002 P5003 P5004 P5005 P5101 P5102 P5103 P5104 P5105																						
1	NA	NA	NA	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-				
2	NA	NA	NA	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-				
3	NA	NA	NA	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-				
4	NA	NA	NA	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-				
5	NA	NA	NA	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-				
P52 P53 P54 P55 I1 I2 I3 I4 I5 I6 I7 I8 I9 E101 E102 E103 E2 E3 E4 C1 C1A C2																						
1	2	2	2	2	7	NA	30	2	2	NA	12	NA	1	20	10	17	5	17	3	1	0	2
2	2	2	2	2	1	NA	11	NA	2	2	2	NA	NA	20	10	17	5	17	1	1	0	1
3	2	2	1	1	2	NA	10	NA	NA	1	NA	NA	NA	20	10	17	5	30	2	1	0	1
4	2	2	2	1	3	NA	31	NA	2	2	10	1	NA	20	10	17	5	18	2	1	0	1
5	2	2	2	2	4	NA	13	NA	NA	1	9	NA	NA	20	10	17	5	19	2	1	0	1
C2A C2B C3 C4 RECUERDO ESTUDIOS OCUMAR11 RAMA09 CONDICION11 ESTATUS PESO																						
1	1	0	1	0		99			5		5		4			7			2	0.925		
2	0	0	1	0		3			6		2		4			1			1	0.925		

3	0	0	1	0	99	5	3	2	8	2	0.925
4	0	0	1	0	98	6	2	4	2	1	0.925
5	0	0	1	0	3	3	9	4	8	5	0.925

Dataframe (5 last): /n

	ESTU	CUES	CCAA	PROV	MUN	TAMUNI	CAPITAL	DISTR	SECCION	ENT
2553	3192	2596	19	52	1	4	1	0	0	3 3 2 6
2554	3192	2597	19	52	1	4	1	0	0	3 2 2 7
2555	3192	2598	19	52	1	4	1	0	0	3 2 4 3
2556	3192	2599	19	52	1	4	1	0	0	3 2 3 5
2557	3192	2600	19	52	1	4	1	0	0	3 7 2 7

	P401	P402	P403	P404	P501	P502	P503	P504	P6	P6A01	P6A02	P6B	P6C	P6D	P7	P7A
2553	1	1	1	1	8	7	6	7	1	10	0	2	1	2	4	7
2554	3	2	1	2	6	9	5	6	2	96	96	0	0	0	0	NA
2555	2	2	2	2	5	3	7	7	1	0	2	0	0	0	0	NA
2556	2	2	2	2	4	6	2	3	2	96	96	0	0	0	0	NA
2557	2	2	2	2	3	4	3	6	1	0	2	0	0	0	0	NA

	P801	P802	P803	P8A01	P8A02	P8A03	P8B01	P8B02	P8B03	P901	P902	P903	P904
2553	1	2	2	1	0	0	4	96	96	7	7	8	8
2554	0	0	0	0	0	0	96	96	96	7	7	7	7
2555	0	0	0	0	0	0	96	96	96	5	5	5	5
2556	0	0	0	0	0	0	96	96	96	5	5	5	5
2557	0	0	0	0	0	0	96	96	96	98	98	98	98

	P905	P906	P907	P908	P1001	P1002	P1003	P11	P11A01	P11A02	P11B	P11C	P11D	P12
2553	8	8	8	5	1	3	1	2	96	96	0	0	0	2
2554	1	7	7	1	2	8	2	2	96	96	0	0	0	2
2555	2	5	5	1	2	3	2	1	1	0	1	0	0	2
2556	2	5	5	6	3	3	3	2	96	96	0	0	0	2
2557	2	98	98	1	2	3	2	2	96	96	0	0	0	2

	P12A01	P12A02	P12B	P12C	P12D01	P12D02	P12D	P1301	P1302	P1303	P1304	P1305
2553	96	96	0	0	NA	NA	NA	7	8	8	8	4
2554	96	96	0	0	NA	NA	NA	7	10	7	10	1
2555	96	96	0	0	NA	NA	NA	5	6	5	6	1
2556	96	96	0	0	NA	NA	NA	5	7	5	5	4
2557	96	96	0	0	NA	NA	NA	7	7	7	7	1

	P14	P14A01	P14A02	P14B	P14C	P1501	P1502	P1503	P1504	P1505	P1506	P16	P17
2553	2	96	96	0	0	7	7	7	6	7	4	3	8
2554	2	96	96	0	0	7	8	1	5	8	1	3	8
2555	2	96	96	0	0	4	6	1	1	3	1	3	8

2556	2	96	96	0	0	6	7	1	1	3	4	3	8					
2557	2	96	96	0	0	5	7	6	4	7	1	3	1					
P18 P18A P18B P18C01 P18C02 P18C03 P18C04 P18C05 P18C06 P18C07 P18C08 P19																		
2553	1	4	1	NA	NA	NA	NA	NA	1	NA	NA	8						
2554	2	0	0	0	0	0	0	0	0	0	0	8						
2555	2	0	0	0	0	0	0	0	0	0	0	1						
2556	2	0	0	0	0	0	0	0	0	0	0	2						
2557	2	0	0	0	0	0	0	0	0	0	0	2						
P20 P21 P21A01 P21A02 P21A03 P21A04 P21B P22 P23 P24 P2501 P2502 P2503																		
2553	2	2	0	0	0	0	0	3	1	3	10	8	8					
2554	3	2	0	0	0	0	0	3	8	1	10	1	98					
2555	2	2	0	0	0	0	0	1	8	3	10	1	1					
2556	3	2	0	0	0	0	0	1	1	3	10	1	1					
2557	2	2	0	0	0	0	0	3	2	2	10	5	98					
P2601 P2602 P2603 P2604 P27 P28 P29 P30 P30A P30AR P31 P32 P33 P34 P35																		
2553	2	2	2	8	1	1	5	5	0	0	1	32	2	2	3			
2554	2	2	2	2	1	3	7	1	1	1	1	19	2	2	3			
2555	2	2	2	2	1	3	7	1	1	1	2	38	1	1	3			
2556	1	1	1	1	2	4	5	5	0	0	2	20	2	2	3			
2557	2	2	2	2	1	1	8	1	1	1	1	44	1	1	3			
P35A P36 P37 P38 P39 P40 P41 P42 P42A P43 P44 P45 P46 P46A P46B P46C P46D																		
2553	12	3	1	4	2	4	271	2	3	62	98	1	1	1	0	0	0	
2554	6	2	2	4	2	6	592	1	1	84	98	1	1	1	0	0	0	
2555	9	2	2	4	2	1	224	1	1	85	99	99	1	1	0	0	0	
2556	6	1	2	4	2	6	740	1	3	45	98	1	1	1	0	0	0	
2557	5	1	2	4	1	1	2	1	1	84	99	7	1	1	0	0	0	
P47 P47A P47B P48 P4901 P4902 P4903 P4904 P5001 P5002 P5003 P5004 P5005																		
2553	1	19	0	1	1	NA	NA	NA	-	-	-	-	-	-				
2554	1	19	0	3	1	NA	NA	NA	-	-	-	-	-	-				
2555	1	5	0	1	1	NA	NA	NA	-	-	-	-	-	-				
2556	1	19	0	1	NA	NA	NA	1	-	-	-	-	-	-				
2557	1	19	0	1	NA	NA	NA	NA	-	-	-	-	-	-				
P5101 P5102 P5103 P5104 P5105 P52 P53 P54 P55 I1 I2 I3 I4 I5 I6 I7 I8 I9																		
2553	-	-	-	-	-	2	3	2	2	5	NA	NA	NA	NA	NA	1	NA	NA
2554	-	-	-	-	-	2	1	2	1	6	NA	5	2	1	1	14	NA	NA
2555	-	-	-	-	-	2	1	1	1	7	NA	1	NA	NA	NA	3	NA	NA
2556	-	-	-	-	-	2	1	2	1	3	NA	1	NA	1	NA	2	NA	1

2557	-	-	-	-	-	2	1	1	1	8	NA	NA	NA	NA	NA	1	NA	NA
	E101	E102	E103	E2	E3	E4	C1	C1A	C2	C2A	C2B	C3	C4	RECUERDO	ESTUDIOS	OCUMAR11		
2553	22	10	17	7	17	2	1	0	1	0	0	1	0	97	6	2		
2554	22	10	17	7	19	2	1	0	1	0	0	1	0	1	4	5		
2555	22	10	17	7	17	2	1	0	1	0	0	1	0	1	6	2		
2556	22	10	17	7	17	1	1	0	1	0	0	1	0	97	4	7		
2557	22	10	17	7	17	2	2	1	1	0	0	1	0	1	5	10		
	RAMA09	CONDICION11	ESTATUS	PESO														
2553	4		9		1	0.049												
2554	4		10		2	0.049												
2555	4		2		1	0.049												
2556	4		10		4	0.049												
2557	4		12		2	0.049												

1.8.7. Ejercicio: *script 6: recuentos básicos*

Existen dos formas de obtener frecuencias. La primera utiliza la sintaxis clásica de R; la segunda, más potente, se integra con el flujo de **expss** y muestra una función de este paquete pensada para la explotación de archivos de datos en investigación de mercados (tipo SPSS).

```
# Note: library(expss) is not loaded here because
↪ source('global') has already
# been executed and it loads all the required packages

# Basic frequency table using standard R syntax
table(data$P31)

# Basic frequency table using standard `expss` syntax
fre(data$P31)
```

Hombre	Mujer
1256	1301

Sexo de la persona entrevistada	Count	Valid percent	Percent	Responses, %	Cumulative
Hombre	1256	49.1	49.1	49.1	
Mujer	1301	50.9	50.9	50.9	
#Total	2557	100	100	100	
<NA>	0		0.0		

1.8.8. Ejercicio: *script 7: uso de pipes*

Se utiliza el operador `%>%` (*pipe*), que puede imaginarse como una cinta transportadora: el conjunto de datos se pasa a la función de celdas, luego a la de estadísticos y, finalmente, se pivota o publica.

```
# Calculate frequencies using pipes and expss
data %>%
  tab_cells(P31) %>%
  tab_stat_cases() %>%
  tab_pivot()
```

	#Total
Sexo de la persona entrevistada	
Hombre	1256
Mujer	1301
#Total cases	2557

1.8.9. Ejercicio: *script 8: personalización de tablas*

La legibilidad de las tablas puede mejorarse añadiendo totales y etiquetas personalizadas.

```
# Frequency table with total row positioned above
data %>%
  tab_cells(P31) %>%
  tab_stat_cases(total_row_position = "above", label = "Total
  ↪ cases") %>%
```

```

tab_pivot()

# Frequency table with column percentages
data %>%
  tab_cells(P31) %>%
  tab_stat_cpct(total_row_position = "above", label =
    ↪ "Percentage (%)") %>%
  tab_pivot()

```

			#Total
Sexo de la persona entrevistada			
#Total cases	Total cases		2557
Hombre	Total cases		1256
Mujer	Total cases		1301

			#Total
Sexo de la persona entrevistada			
#Total cases	Percentage (%)		2557
Hombre	Percentage (%)		49.1
Mujer	Percentage (%)		50.9

1.8.10. Ejercicio: *script 9: combinando estadísticos*

A menudo resulta útil ver tanto los casos absolutos como los porcentajes en una misma tabla, para facilitar la interpretación.

```

# Combining counts and column percentages in one table
data %>%
  tab_cells(P31) %>%
  tab_stat_cases(total_row_position = "above", label = "Cases")
  ↪ %>%
  tab_stat_cpct(label = "Column%") %>%
  tab_pivot(stat_position = "inside_columns")

```

	#Total	
	Cases	Column %
Sexo de la persona entrevistada		
#Total cases	2557	2557
Hombre	1256	49.1
Mujer	1301	50.9

1.8.11. Ejercicio: *script 10*: tipos de porcentajes

Se exploran a continuación las distintas formas de calcular porcentajes (columna, fila y tabla):

```
# Comparing different percentage calculations
data %>%
  tab_cells(P31) %>%
  tab_stat_cases() %>%
  tab_stat_cpct() %>% # Column percentage
  tab_stat_rpct() %>% # Row percentage
  tab_stat_tpct() %>% # Table percentage
  tab_pivot()
```

	#Total
Sexo de la persona entrevistada	
Hombre	1256.0
Mujer	1301.0
#Total cases	2557
Hombre	49.1
Mujer	50.9
#Total cases	2557
Hombre	100.0
Mujer	100.0
#Total cases	2557
Hombre	49.1
Mujer	50.9
#Total cases	2557

1.8.12. Ejercicio: *script* 11: gestión de etiquetas

Una parte vital del análisis de encuestas es la gestión de etiquetas. La instrucción `var_lab()` permite documentar qué significa la variable, y `val_lab()` permite asignar etiquetas a los valores numéricos (igual que en la vista de variables de SPSS).

```
# Assigning a variable label
var_lab(data$P31) <- 'Respondent gender'

# Assigning value labels to the numeric codes
val_lab(data$P31) <- c('Male' = 1, 'Female' = 2)

# Check the new labels
var_lab(data$P31)
val_lab(data$P31)
```

```
[1] "Respondent gender"
  Male Female
    1      2
```

1.9. ¿Qué se ha aprendido hasta ahora?

Se cierra así esta primera sección, en la que se ha pasado de ejecutar comandos aislados en la consola a construir el primer *script*. Se ha visto también que R es sensible a los valores perdidos (NA).

Esta primera inmersión en el ecosistema de R y RStudio puede resultar exigente. Es completamente normal que, en este punto, la sintaxis, los *scripts* o el uso del *pipe* (`%>%`) resulten algo extraños. Aprender R es como aprender un nuevo idioma: al principio cuesta memorizar el vocabulario y la gramática, pero con un poco de práctica las instrucciones empiezan a fluir con naturalidad.

Conviene tener presente el objetivo principal de este máster: **el propósito no es formar programadores informáticos, sino analistas de mercado estratégicos**. *RStudio* es simplemente el vehículo más potente y profesional que existe en la actualidad para transformar encuestas, hojas de cálculo y bases de datos en información de negocio. En esta sesión se ha encendido el motor, se ha configurado el

panel de mandos y se ha dado una vuelta de reconocimiento tabulando los primeros porcentajes. En los siguientes pasos de este camino, empieza la conducción real.

Los aspectos puramente técnicos de la instalación quedan atrás para centrarse en el **proceso de investigación, la inferencia y el análisis multivariante**. El recorrido pasa de la observación descriptiva a la toma de decisiones basada en evidencia estadística:

- **El investigador y su proceso de trabajo:** nos ayudará a organizar el trabajo con Proyectos de RStudio, a entender la relación vital entre el diseño de un cuestionario y la estructura de un *dataframe*, y a clasificar correctamente las variables para no cometer errores estadísticos.
- **Diagnóstico estadístico:** se pasará de los recuentos básicos a la estadística de precisión, aprendiendo a visualizar datos con histogramas y diagramas de caja, a detectar valores atípicos (*outliers*) o perdidos, y a validar los supuestos de normalidad y homocedasticidad.
- **Inferencia estadística paramétrica y no paramétrica:** se dará el salto definitivo, utilizando test de hipótesis (T de Student, ANOVA, Chi-cuadrado) para descubrir si las diferencias observadas entre clientes (por ejemplo, ¿compran más los hombres que las mujeres?) son reales o un simple efecto del azar.
- **Análisis de Correspondencias:** se entra en el terreno del posicionamiento estratégico, aprendiendo a construir mapas perceptuales para visualizar gráficamente dónde se ubica una marca respecto a los competidores según los atributos valorados por los consumidores.
- **Análisis de Componentes Principales (ACP):** se trabajará con cuestionarios complejos con decenas de preguntas, descubriendo cómo reducir esa “sábana” de datos a unas pocas dimensiones latentes (por ejemplo, resumiendo 20 preguntas en 3 factores clave: calidad, precio y servicio).
- **Análisis de Conglomerados (*Cluster Analysis*):** la herramienta reina del marketing. Se utilizarán algoritmos jerárquicos y de K-medias para segmentar el mercado, agrupando a los clientes en nichos homogéneos con necesidades y comportamientos similares.
- **Regresión Múltiple:** se modelizará usando la regresión múltiple, como herramienta de base ya usada de forma interna en el ACP, pero explicitando ahora su uso, con diagnóstico, validación y especificación de modelo de relación de datos.
- **Cierre:** se consolidará todo el conocimiento adquirido, aplicando las variables y segmentos creados a lo largo del curso para responder a los objetivos finales

de investigación que demanda un cliente real.

El recorrido no ha hecho más que empezar. Conviene guardar los *scripts*, revisar los conceptos básicos y prepararse para descubrir todo lo que los datos tienen que contar.

2 Introducción al análisis



2.1. La exploración de datos

Contenidos del tema



1. Exploración de los datos: introducción y herramientas básicas para la exploración de los datos. (Presentación de R / Quarto)
2. Herramientas de exploración: comprobación de la linealidad, normalidad y homoscedasticidad
3. Inferencia estadística paramétrica
4. Inferencia estadística no paramétrica

Bibliografía recomendada

- o Uriel, E. y Aldás, J. (2017). Análisis multivariante aplicado con R. Ediciones Paraninfo, SA.
- o Uriel, E. y Aldás, J. (2005). Análisis multivariante aplicado. Madrid: Thomson.
- o Hair, J.F., Anderson, R.E., Tatham, R.L. Y Black, W. (1999). Análisis multivariante. 4ª edición. Madrid: Prentice Hall.
- o James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). "An introduction to statistical learning" (Vol. 112). New York, NY: Springer.
- o De Vries, A., & Meys, J. (2015). R for Dummies. John Wiley & Sons.
- o Chapman, C., & Feit, E. M. (2015). R for marketing research and analytics. New York, NY: Springer.

5

Con los rudimentos del trabajo en R ya cubiertos, comienzan los contenidos propios de la asignatura, dedicada en primer término a una introducción a los procedimientos básicos de trabajo con datos provenientes de recogidas estructuradas o de investigaciones de mercado, predominantemente de encuesta, también llamado EDA (*exploratory data analysis*) Conviene recordar que esta no es una asignatura de R, sino una asignatura que explota datos estadísticos y que utilizará técnicas univariantes, bivariantes y multivariantes para obtener esa información. Os apporto una referencia de producción propia que os puede venir bien como documentación básica para las técnicas que se ven en la asignatura y otros que no - [EDA y más \(*exploratory data analysis*\)](#) - .

Los siguientes pasos permiten avanzar en los cálculos estadísticos y en la obtención de información significativa. Este documento pretende ser una guía ilustrativa y demostrativa de cómo se trabaja.

Antes de explorar las posibilidades del *software* con el que se va a trabajar, conviene entender cómo está organizado el trabajo. En la primera sesión se creó el proyecto **TMIM**, que es el punto de partida de todo el curso. Un proyecto de RStudio es, simplemente, una carpeta que RStudio reconoce como espacio de trabajo: al abrirlo, R sabe en qué carpeta debe buscar los datos y guardar los resultados, sin necesidad de indicar rutas completas.

El proyecto tiene la misma estructura en todos los ordenadores:

TMIM/

TMIM.Rproj	← archivo del proyecto (abre RStudio dentro del proyecto)
data/	← archivos de datos del curso
scripts/	← global.R y los scripts de cada sesión
output/	← resultados que se generen (tablas, gráficos...)

Estar dentro de un proyecto significa que todo el trabajo se desarrolla dentro de esa carpeta. Puede comprobarse de dos formas:

- en la esquina superior derecha de RStudio aparece el nombre **TMIM**;
- al ejecutar `getwd()` en la consola, R devuelve la ruta de la carpeta **TMIM**.

Gracias a ello, los ficheros se cargan siempre con una **ruta relativa** a la carpeta del proyecto, por ejemplo "**data/archivo.sav**". Esta ruta funciona igual en cualquier ordenador, sea Windows o Mac, y sea cual sea el nombre del usuario. Puede comprobarse el contenido de cada carpeta desde la pestaña **Files** del cuadrante inferior derecho.

Tip

... Si RStudio no abre el proyecto

RStudio abre automáticamente el último proyecto utilizado. Si no aparece **TMIM** en la esquina superior derecha, abrirlo desde **File > Recent Projects** o haciendo doble clic sobre el archivo **TMIM.Rproj** de la carpeta del curso.

Comienza así el contenido específico de la asignatura.

2.1.1. El investigador y su proceso de trabajo

Para realizar una investigación sólida es necesario dominar cinco pilares fundamentales:

1. **El contexto de la explotación de datos:** ¿por qué se analiza? Los datos tienen vida y propósito; no son meros números, sino la voz de quien los proporciona.
2. **El proceso de análisis:** se sigue un flujo lógico: desde la recolección y limpieza hasta la transformación y, finalmente, la interpretación estratégica.

3. **Escalas de medida:** se aprende a distinguir entre variables nominales, ordinales y de razón. En R, saber clasificar correctamente una variable es el paso previo obligatorio para elegir la técnica de análisis adecuada.
4. **Relación entre instrumento y base de datos:** se explora cómo el diseño de la encuesta (el instrumento) se traduce en la estructura del *dataframe*. Un buen diseño de cuestionario ahorra horas de trabajo en R.
5. **La exploración (EDA - *Exploratory Data Analysis*):** esta es la etapa más crítica. Antes de construir modelos complejos, es necesario “mirar” los datos, identificar patrones, detectar valores perdidos y entender las distribuciones.

Conviene dejar de ver una hoja de cálculo como una tabla inerte y empezar a verla como un conjunto de información esperando ser interpretada. Se trata de aprender a hacerle las preguntas correctas a los datos. **Una buena exploración no busca probar hipótesis, busca descubrir qué dicen los datos antes de procesarlos para informar lo que se quiere.**

La explotación de datos no comienza frente a la pantalla de R o RStudio, sino en la definición previa de la investigación. Este esquema refleja las etapas necesarias para garantizar que los datos obtenidos sean útiles para el análisis:

¿Dónde se inicia la explotación de los datos?



[3%]
SIS

1. **Definición metodológica:** El primer paso es determinar si el enfoque será cualitativo, para comprender fenómenos, o cuantitativo, para medir variables. Esta decisión condiciona toda la estrategia posterior.

2. **Selección de la técnica:** La elección entre observación directa, encuestas o registros automáticos depende del objetivo de estudio y de la fuente de información disponible.
3. **Diseño del instrumento:** Consiste en estructurar el soporte de recogida (cuestionarios, formularios o sistemas de registro). Un diseño preciso es fundamental para minimizar errores durante la captura de datos.
4. **Estrategia de análisis:** Una vez obtenidos los datos, se aplica la metodología de tratamiento estadístico. El éxito de esta etapa final depende directamente de la coherencia con la que se hayan ejecutado los tres pasos anteriores.

La calidad de los resultados está intrínsecamente ligada al rigor aplicado en cada una de estas fases previas.



2.1.2. Proceso

Una vez que los datos han sido recolectados y estructurados, el investigador debe iniciar una fase de análisis descriptivo que permita sintetizar la información. Las tareas fundamentales en esta etapa son:

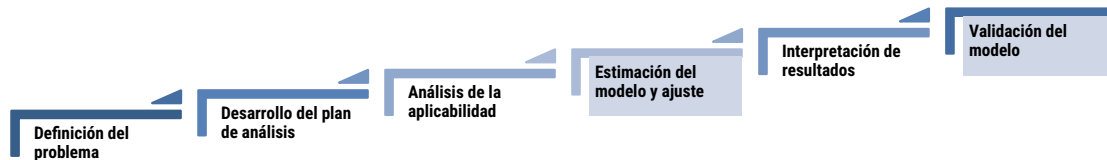
1. **Exploración de datos:** Consiste en el examen preliminar de las variables para detectar anomalías, valores perdidos o comportamientos inusuales antes de iniciar el análisis estadístico.
2. **Recuentos:** Es la tarea básica de contabilización de frecuencias, esencial para entender el tamaño de cada categoría en la muestra.

3. **Tablas marginales:** Permiten obtener una visión general de la distribución de una variable por sí misma, independientemente de las demás.
4. **Tablas cruzadas:** Es el método principal para analizar la relación entre dos variables, permitiendo identificar patrones o asociaciones entre ellas.
5. **Inferencia sobre tablas de contingencia:** Finalmente, se aplican pruebas estadísticas (como el test de Chi-cuadrado) para determinar si las asociaciones observadas en las tablas cruzadas son estadísticamente significativas o fruto del azar.

Este flujo de trabajo permite pasar de una base de datos bruta a un conocimiento consolidado, sirviendo como base necesaria para cualquier análisis de mayor complejidad.



Proceso de aplicación de una técnica bivalente (χ^2 , correlación ...) / multivalente (componentes, clúster...)



8

La profundidad del proceso de análisis varía según la complejidad de la técnica seleccionada. Aunque todo proyecto parte de una definición clara del problema, el flujo operativo se expande a medida que aumenta la sofisticación de las herramientas utilizadas:

- **Técnicas univariantes:** El proceso es directo. Tras la definición y el plan de análisis, se evalúa la aplicabilidad (comprobación de supuestos básicos) para proceder a la interpretación de los estadísticos descriptivos (medias, frecuencias, dispersión).
- **Técnicas bivalentes y multivariantes:** Este flujo es más exigente. Además de las etapas iniciales, requiere una fase central de **estimación y ajuste del modelo**, donde se cuantifican las relaciones o estructuras subyacentes. Tras

la interpretación, se añade un paso crítico: la **validación del modelo**, que garantiza que los resultados obtenidos son robustos, fiables y generalizables.

Mientras que en el análisis univariante se busca describir el comportamiento de una variable, en el análisis multivariante el objetivo es reducir la incertidumbre mediante la confirmación de modelos que expliquen la estructura de los datos con rigor estadístico.

Para operar con eficacia es necesario conectar los conceptos teóricos de la investigación con la arquitectura técnica de los bancos de datos. Esta fase se divide en dos ejes:

Consideraciones sobre las variables: el análisis estadístico comienza con la correcta definición de la variable. Es necesario comprender qué constituye un instrumento de medición, cómo se desglosa en preguntas y respuestas, y la distinción vital entre el **valor** (el dato numérico o texto) y su **etiqueta** (el significado que se le asigna). La codificación correcta es lo que permite que una variable pase de ser una respuesta en un cuestionario a un factor analizable en el *software*.

Consideraciones sobre los datos: aquí se aborda la naturaleza del archivo. No basta con disponer de información; es preciso entender cómo se estructura en filas (observaciones) y columnas (variables), qué formatos son óptimos para el intercambio entre plataformas y cómo gestionar los metadatos. La correcta diferenciación entre el dato bruto y el etiquetado de variables es lo que garantiza la transparencia y la calidad del informe final.

Dominar estos fundamentos permite evitar errores comunes en la importación y manipulación de *datasets*, asegurando que el análisis realizado en R sea preciso desde el primer momento.

2.1.3. Escalas de medida



Escalas de medida

- Criterio del tipo de métrica
 - No métricas, cualitativas.
 - **Nominales**: los números son etiquetas. Su objetivo es distinguir entre valores. Ni orden, ni diferencias, ni razón.
 - **Ordinales**: distingue entre valores y establece un orden entre ellos. Ni diferencia, ni razón.
 - Métricas, cuantitativas.
 - **Intervalo**: distinción, orden y diferencias. No hay razón; no cuantifica proporcionalmente.
 - **Razón**: distinción, orden, diferencias y razón; cuantifica proporcionalmente.
- Criterio de la continuidad de valores
 - Discretas.
 - Puede tomar sólo unos determinados valores en un intervalo.
 - Continuas.
 - Puede tomar todos los posibles valores en un intervalo.

10

El primer paso para analizar un dato es definir su naturaleza. El *software* necesita saber qué tipo de información está procesando para aplicar la prueba estadística correcta. Las variables se clasifican bajo dos criterios:

- 1. Según el tipo de métrica (Nivel de medición):
 - **Cualitativas (No métricas)**: Se centran en atributos.
 - Las variables **nominales** actúan como etiquetas de clasificación (ej. género).
 - Las variables **ordinales** incorporan además un sentido de jerarquía u orden (ej. nivel de satisfacción).
 - **Cuantitativas (Métricas)**: Permiten operaciones matemáticas precisas. Las variables de **intervalo** poseen un orden y una magnitud de diferencia constante (ej. temperatura en grados Celsius), mientras que las de **razón** incluyen un cero absoluto que permite cuantificar proporciones (ej. ingresos o edad).
- 2. Según la continuidad de valores:
 - **Discretas**: Representan valores contables que no admiten decimales intermedios (ej. número de hijos).
 - **Continuas**: Pueden asumir cualquier valor dentro de un rango determinado, lo que permite una medición de alta precisión (ej. tiempo de respuesta o peso).

La identificación correcta de estas escalas es el paso crítico que determina la capacidad de ejecutar modelos estadísticos. En R, esta clasificación es la que guía la elección entre utilizar un gráfico de barras, un histograma, un test no paramétrico o una regresión lineal.

2.1.4. Relación instrumento de medida y base de datos

La traducción de un cuestionario a una base de datos no siempre es lineal. La estructura final del *dataframe* depende directamente del diseño de la pregunta.

Por ejemplo en este extracto de cuestionario simulado ...

Relaciones entre instrumento de medida y base de datos ...



- P1. ¿Dispone usted de licencia de conducir para vehículos de tipo turismo?
 - 1. Sí
 - 2. No
- P2. ¿Desde cuándo posee esta licencia?
 - ____ (años)
- P3. De los siguientes elementos mecánicos, indiquenos si ha sufrido alguna ocasión alguna avería relacionada?
 - 1. Neumáticos
 - 2. Sistema de frenado
 - 3. Sistema de amortiguación
 - 4. Sistema de refrigeración
 - 5. Sistema eléctrico (incluye batería)
 - 6. Ninguna avería
- P4. Indiquenos su grado de acuerdo con las siguientes afirmaciones y para ello utilice una escala de 1 a 5, donde el 1 es "muy en desacuerdo", el 2 es "en desacuerdo", el 3 es "ni de acuerdo, ni en desacuerdo", el 4 es de "acuerdo" y el 5 es "muy de acuerdo".
 - 1. Conduzco siempre respetando los límites de velocidad que creo adecuados al tipo de vía utilizada.
 - 2. En ocasiones siento que la vía no se adapta a los límites de velocidad establecidos
 - 3. Siguiente ítem ...
 - 4. Siguiente ...
 - 5. ...

QN	P1	P2	P3_1	P3_2	P3_3	P3_4	P3_5	P3_6	P4_1	P4_2	P4_...
1	1	24	1	1	0	1	1	0	5	4	...
2	2	-	-	-	-	-	-	-	-	-	...
3	1	5	1	0	0	0	1	0	2	5	...
...	...	...	...	...	...	...	...	...	...	...	...
625	1	0	0	0	0	0	0	1	5	3	...

QN	P1	P2	P3_1	P3_2	P3_3	P3_4	P4_1	P4_2	P4_...
1	1	24	5	4	1	2	5	4	...
2	2	-	-	-	-	-	-	-	...
3	1	5	1	5			2	5	...
...	...	...	...	...	...	...	...	...	...
625	1	0	6				5	3	..

11

1. **Preguntas simples (P1, P2):** Son las más directas. Una pregunta se traduce habitualmente en una sola columna con el valor registrado (ej. el código de respuesta o el número de años).
2. **Preguntas de respuesta múltiple (P3):** Estas preguntas (como la de "averías sufridas") requieren una atención especial. Aunque en el cuestionario aparecen bajo un mismo enunciado, en la base de datos se transforman en **tantas columnas como opciones de respuesta** existan (P3_1, P3_2, etc.). Cada columna funciona como una variable binaria (0|1 o Sí|No), lo que permite al encuestado seleccionar varias opciones simultáneamente.
3. **Baterías o escalas (P4):** Aunque visualmente ocupan varias columnas (P4_1, P4_2...), su naturaleza es distinta a la multirespuesta. Aquí, cada columna representa el mismo concepto medido bajo la misma escala (ej. escala

Likert). A diferencia de la multirespuesta, estas columnas son intrínsecamente comparables entre sí y suelen analizarse conjuntamente como una unidad de medida. Una escala es un conjunto de ítems que tienen el mismo conjunto de respuestas posibles y que se consideran en conjunto. Si se trata de ítems de actitudes y el protocolo verbal es el grado de acuerdo, estamos ante una escala de tipo Likert, y su consideración o puntuación es agregada.

Entender esta distinción es vital para el analista: mientras que en una escala (P4) el objetivo es analizar la **consistencia interna** entre los ítems, en una multirespuesta (P3) el objetivo es **contabilizar la incidencia** de cada una de las opciones por separado.

2.1.4.1. Múltiple en binario frente a menciones

La forma en que se registran los datos no es neutral; determina qué se puede —y qué no se puede— analizar después. Las dos tablas de la diapositiva muestran la diferencia crítica al tratar una pregunta de respuesta múltiple (P3):

1. **La tabla superior (Variables binarias):** Es la forma estándar de codificar “presencia o ausencia”. Cada opción de respuesta se convierte en una columna (0 = no marcado, 1 = marcado). Es ideal para contabilizar cuántos encuestados eligieron cada elemento, pero **se pierde el orden en el que fueron seleccionados**.
2. **La tabla inferior (Sistema de menciones):** Aquí, el valor registrado corresponde al orden de elección (1ª mención, 2ª mención, etc.). Esta estructura es mucho más rica y potente:
 - **Conserva la jerarquía:** Permite identificar qué avería fue la más crítica o la que primero vino a la mente del usuario.
 - **Análisis de *Top of Mind* (TOM):** Es la única forma de aislar la primera respuesta para medir el recuerdo espontáneo o la prioridad máxima del consumidor.

i Ejercicio práctico

Para operar con datos, la teoría debe transformarse en una acción inmediata: clasificar. A continuación se analiza un banco de datos de hábitos digitales. El objetivo es identificar la escala de medida real de cada variable, superando la apariencia superficial de los datos.

Medición y escalas en el instrumento de medición



¿Nos atrevemos a averiguar la escala de la variable a partir de su definición?

- Cualitativas / no métricas
 - Escala nominal o categórica
 - Escala ordinal
- Cuantitativas / métricas
 - Escala de intervalo o proporción
 - Escala de razón o numérica

Variable	Descripción
age	Edad (en años)
educ	Años de estudios
usecomp	¿Utiliza habitualmente un ordenador personal en casa, el trabajo o en algún otro lugar? (1=Sí, 0=No)
usenet	Además de para consultar el correo, ¿Utiliza Internet para otra tarea? (1=Sí, 0=No)
usemail	¿Utiliza el correo electrónico? (1=Sí, 0=No)
emailhrs	¿Cuántas horas a la semana aproximadamente dedica a enviar y contestar correos electrónicos? (horas)
webhrs	Sin contar el correo electrónico, ¿cuántas horas dedica por semana a navegar por Internet (blogs, webs, chats, foros de discusión...) (horas)
nethrs	Total de horas navegación (emailhrs+nethrs, calculada)
netcat	Total de horas navegación (codificada) (1=4 horas o menos, 2=4-8 horas, 3=8-16 horas, 4= más de16 horas)
region	Regiones A, B, C.... (A=1, B=2, C=3 ...)

12

- **age (Edad en años):** Escala de razón. Tiene un cero absoluto y permite medir cuántas veces es mayor una edad que otra.
- **educ (Años de estudios):** Escala de razón o intervalo (según el contexto), pero en investigación de mercados suele tratarse como métrica para promedios.
- **usecomp, usenet, usemail (Binarias 1/0):** Escala nominal. Aunque los valores sean numéricos (0 o 1), estos números son meras etiquetas que indican presencia o ausencia.
- **emailhrs, webhrs, nethrs (Horas):** Escala de razón. Son variables continuas que permiten realizar cálculos aritméticos directos.
- **netcat (Categorizada 1-4):** Escala ordinal. Aunque derivan de una variable continua, al agruparlas en rangos, se pierde la precisión exacta y solo se conserva el orden o jerarquía.
- **region (A=1, B=2...):** Escala nominal. Los números asignados a las regiones no representan cantidad ni orden; solo sirven para diferenciar grupos.

El reto del analista: la presencia de números puede engañar. En R, si una escala nominal (como **region**) se clasifica erróneamente como de razón (numérica), el *software* intentará calcular una “media de regiones”, lo cual carece de sentido lógico. Clasificar correctamente es el primer paso hacia un análisis válido.

2.2. Herramientas de exploración

Para realizar una inferencia válida es necesario dominar primero la descripción básica de los datos; en R, este proceso se traslada a la consola mediante funciones específicas. El objetivo de este bloque es pasar de la exploración visual a la cuantificación estadística:

1. **La exploración del archivo:** antes de calcular nada, se realiza un diagnóstico de la estructura (filas, columnas, tipos de variables) y un análisis exploratorio inicial para detectar anomalías.
2. **La descripción estadística:** se toma la variable `age` como modelo para aprender a extraer los estadísticos fundamentales:
 - **Para variables categóricas (Nominales/Ordinales):** se utilizan **tablas de frecuencias**, que permiten conocer cuántos individuos responden a cada categoría y qué porcentaje representan sobre el total.
 - **Para variables continuas (Intervalo/Razón):** se utilizan **estadísticos descriptivos**: la media, mediana, desviación típica, mínimos, máximos y la MAD, que permiten resumir la distribución de la variable en unos pocos indicadores numéricos.

Cambio de paradigma

Mientras que en otros *software* el análisis se limita a “clicar” opciones, en R el análisis es **reproducible**. Cada tabla de frecuencias o estadístico obtenido queda guardado en el *script*, lo que permite que el informe completo se genere de manera automática y sin errores de manipulación manual. Esta es una de las grandes ventajas de trabajar con R y RStudio.

2.2.1. Frecuencias

Este es el siguiente paso, el primer *script* de trabajo del proyecto. Conviene explorar siempre el banco de datos cargado con la instrucción `glimpse` del paquete `dplyr`.

Frecuencias y porcentajes, medida básica

- Las frecuencias son la base de trabajo del investigador, cuántas veces sucede un evento.
- El resultado muestra diferentes columnas con el valor absoluto y el relativo, tomando como base todos los casos del banco de datos o solo los casos válidos.
- Se suele atender a los valores y no a las etiquetas, pero eso varía según paquetes y formas de usarla.

```
expss::: fre()
```

13

```
# Load the SPSS dataset into the 'gssnet1' object
# We use the 'expss' package because it perfectly retains
  ↳ variable and value labels from SPSS
gssnet1 <- expss::read_spss('data/gssnet1.sav')

# Get a quick, clean overview of the data structure (variables,
  ↳ data types, and first few values)
dplyr::glimpse(gssnet1)

# Display the dataframe in the console to inspect its actual
  ↳ content
head(gssnet1,10)
cat(' ... ')
tail(gssnet1,10)
```

Rows: 984

Columns: 11

```
$ age      <labelled> 30, 39, 72, 41, 24, 23, 27, 34, 45, 51, 46, 56, 46, 23, ~
$ educ     <labelled> 11, 9, 10, 13, 12, 12, 12, 10, 11, 12, 12, 6, 10, 10, 11~
$ usecomp  <labelled> 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0,~
$ usenet   <labelled> 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0,~
$ usemail  <labelled> 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0,~
```

\$ emailhrs <labelled> -1.00000000, -1.00000000, -1.00000000, -1.00000000, 1.00~
 \$ webhrs <labelled> -1.0, -1.0, -1.0, -1.0, 2.0, -9.0, 0.5, -1.0, -1.0, -1.0~
 \$ nethrs <labelled> -1.000000, -1.000000, -1.000000, -1.000000, 3.000000, -9~
 \$ netcat <labelled> -1, -1, -1, -1, 1, -9, 1, -1, -1, -1, -1, -1, -1, 1,~
 \$ region <labelled> 5, 7, 3, 3, 6, 5, 7, 4, 3, 5, 6, 7, 7, 7, 7, 2, 3, 3, 3,~
 \$ Zage <labelled> -0.97196706, -0.43378202, 1.53956312, -0.31418535, -1.33~

	age	educ	usecomp	usenet	usemail	emailhrs	webhrs	nethrs	netcat	region
1	30	11	0	0	0	-1	-1.0	-1.0	-1	5
2	39	9	0	0	0	-1	-1.0	-1.0	-1	7
3	72	10	0	0	0	-1	-1.0	-1.0	-1	3
4	41	13	0	0	0	-1	-1.0	-1.0	-1	3
5	24	12	1	1	1	1	2.0	3.0	1	6
6	23	12	1	1	0	-1	-9.0	-9.0	-9	5
7	27	12	1	1	1	1	0.5	1.5	1	7
8	34	10	0	0	0	-1	-1.0	-1.0	-1	4
9	45	11	0	0	0	-1	-1.0	-1.0	-1	3
10	51	12	0	0	0	-1	-1.0	-1.0	-1	5

Zage

1	-0.9719671
2	-0.4337820
3	1.5395631
4	-0.3141853
5	-1.3307571
6	-1.3905554
7	-1.1513621
8	-0.7327737
9	-0.0749920
10	0.2837980

...	age	educ	usecomp	usenet	usemail	emailhrs	webhrs	nethrs	netcat
975	61	20	1	1	1	2.0000000	4.0000000	6.000000	2
976	45	15	9	9	9	-9.0000000	-9.0000000	-3.000000	-3
977	40	18	9	9	9	-9.0000000	-9.0000000	-3.000000	-3
978	42	19	1	1	1	20.0000000	2.0000000	22.000000	4
979	40	12	1	1	1	0.3333333	10.0000000	10.333333	3
980	67	8	0	0	0	-1.0000000	-1.0000000	-1.000000	-1
981	49	14	1	1	1	1.0000000	0.0333333	1.033333	1
982	31	13	1	1	1	1.0000000	1.0000000	2.000000	1

983	71	12	0	0	0	-1.00000000	-1.00000000	-1.00000000	-1
984	34	14	1	1	0	-1.00000000	-9.00000000	-9.00000000	-9
	region		Zage						
975	8		0.8817814						
976	2		-0.0749920						
977	9		-0.3739837						
978	3		-0.2543870						
979	3		-0.3739837						
980	3		1.2405714						
981	3		0.1642014						
982	3		-0.9121687						
983	3		1.4797648						
984	5		-0.7327737						

Se calculan ahora las frecuencias de la variable **age**. Las frecuencias son la base de trabajo del investigador: cuántas veces sucede un evento. El resultado muestra diferentes columnas con el valor absoluto y el relativo, tomando como base todos los casos del banco de datos o solo los casos **válidos**.

Se utiliza el paquete **expss** para obtener todos los cálculos relacionados con la estadística básica paramétrica y con los cálculos de frecuencia.

```
# Generate a detailed frequency table for the 'age' variable
# The 'fre' function from the 'expss' package perfectly mimics
↪ SPSS frequency outputs,
# providing counts, percentages, valid percentages, and
↪ cumulative percentages
head(expss::fre(gssnet1$age),10)
cat(' ... ')
tail(expss::fre(gssnet1$age),10)
```

Age of respondent	Count	Valid percent	Percent	Responses, %	Cumulative responses, %
18	10	1.0	1.0	1.0	1.0
19	10	1.0	1.0	1.0	2.0
20	6	0.6	0.6	0.6	2.6
21	13	1.3	1.3	1.3	3.9
22	9	0.9	0.9	0.9	4.8

Age of respondent	Count	Valid percent	Percent	Responses, %	Cumulative responses, %
23	18	1.8	1.8	1.8	6
24	26	2.6	2.6	2.6	9
25	21	2.1	2.1	2.1	13
26	20	2.0	2.0	2.0	13
27	17	1.7	1.7	1.7	13

...

Age of respondent	Count	Valid percent	Percent	Responses, %	Cumulative responses, %
83	4	0.4	0.4	0.4	97
84	3	0.3	0.3	0.3	98
85	4	0.4	0.4	0.4	98
86	3	0.3	0.3	0.3	98
87	1	0.1	0.1	0.1	98
88	2	0.2	0.2	0.2	99
89	5	0.5	0.5	0.5	99
NA	4	0.4	0.4	0.4	100
#Total	984	100	100	100	
<NA>	0		0.0		

```
# Display only the first few rows (the head) of the frequency
↪ table
head(expss::fre(gssnet1$age))
```

Age of respondent	Count	Valid percent	Percent	Responses, %	Cumulative responses, %
18	10	1.0	1.0	1.0	1
19	10	1.0	1.0	1.0	2
20	6	0.6	0.6	0.6	2
21	13	1.3	1.3	1.3	4
22	9	0.9	0.9	0.9	4
23	18	1.8	1.8	1.8	6

```
# Print a visual separator (...) in the console for better
  ↳ readability
cat (' ... ')
```

...

```
# Display only the last few rows (the tail) of the frequency
  ↳ table
tail(expss::fre(gssnet1$age))
```

Age of respondent	Count	Valid percent	Percent	Responses, %	Cumulative responses, %
87	1	0.1	0.1	0.1	98
88	2	0.2	0.2	0.2	99
89	5	0.5	0.5	0.5	99
NA	4	0.4	0.4	0.4	100
#Total	984	100	100	100	
<NA>	0		0.0		

2.2.2. Histogramas, importancia de la visualización



La importancia del histograma

El gráfico histograma, ofrece una visión de conjunto que ninguna tabla puede replicar, por tres razones clave:

1. Detección de patrones visuales. El histograma revela la forma de la distribución de un vistazo.
2. Identificación de valores atípicos
3. Validación de supuestos: si el gráfico no se parece a una campana, el modelo podría no ser adecuado.

Las tablas de frecuencias indican cuántos casos hay, y el histograma muestra cómo se comportan y qué forma tienen los datos.

```
base::hist()
```

¿Las frecuencias nos cuentan toda la historia de los datos? No; a menudo las tablas de frecuencias ofrecen una visión “atomizada” de los datos: fila a fila, grupo a grupo. El **histograma**, en cambio, ofrece una **visión de conjunto** que ninguna tabla puede replicar, por tres razones clave:

1. **Detección de patrones visuales:** En una tabla de frecuencias, es difícil ver si los datos tienen forma de campana (normal), si están sesgados hacia la derecha, o si tienen “picos” inesperados (como una edad donde hay muchísima gente y en la siguiente casi nadie). El histograma revela la **forma** de la distribución de un vistazo.
2. **Identificación de valores atípicos (*Outliers*):** A veces, un error de carga de datos (alguien que puso edad 200 en lugar de 20) pasa desapercibido en una tabla larga, pero en un histograma es un punto solitario que “salta a la vista” lejos de la masa principal de datos.
3. **Validación de supuestos:** Muchos modelos estadísticos (como las regresiones que se verán más adelante) asumen que los datos siguen una distribución normal. El histograma es la primera prueba de fuego: si el gráfico no se parece a una campana, el modelo podría no ser adecuado.

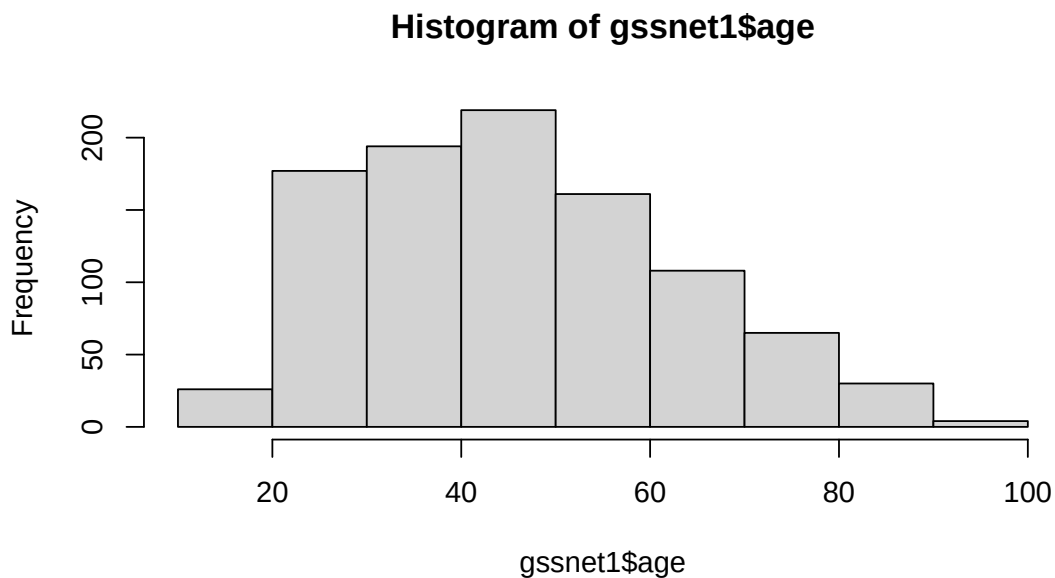
Las tablas de frecuencias indican *cuántos* casos hay, y el histograma muestra *cómo se comportan y qué forma tienen* los datos.

No conviene confiar en una media o una mediana sin haber visualizado antes el histograma. La media es una simplificación extrema; el histograma muestra la realidad de cómo se distribuye el mercado.

Al trabajar con variables numéricas (como la edad), las tablas de frecuencias suelen ser demasiado extensas para interpretarlas directamente. En estos casos, el **histograma** es la herramienta principal, ya que agrupa los datos en intervalos para revelar la forma de su distribución.

Por defecto, R calcula automáticamente el número de intervalos (llamados *bins* o *breaks*) para ofrecer una visión equilibrada. Sin embargo, es el investigador quien debe decidir el nivel de detalle necesario.

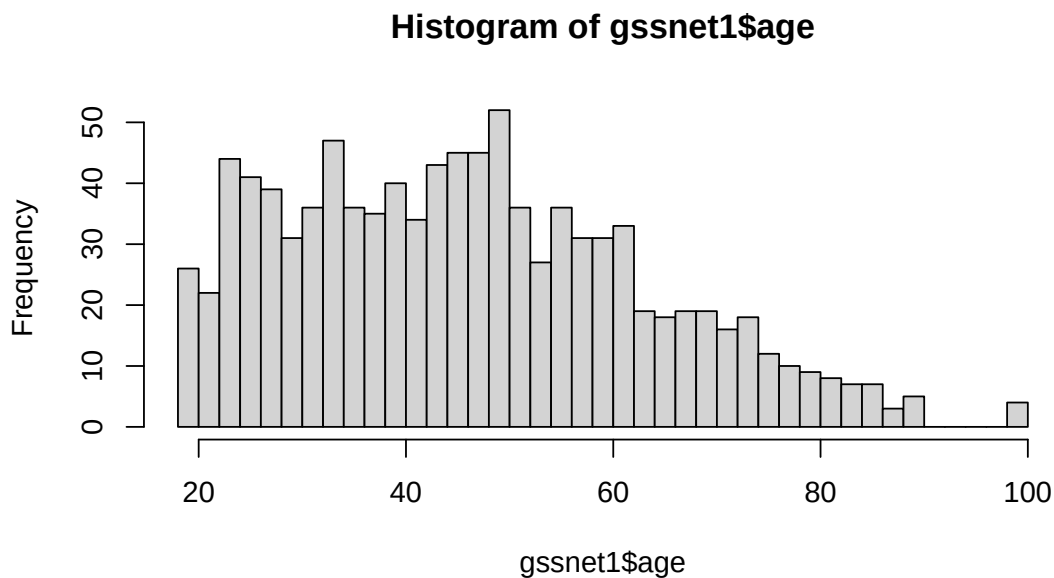
```
# Standard histogram: R chooses the number of breaks (bins)
↪ automatically
hist(gssnet1$age)
```



Si se necesita mayor precisión, el número de intervalos puede controlarse mediante el argumento **breaks**.

Al fijar un número de **breaks** muy elevado (como en el ejemplo siguiente), R intenta mostrar cada valor individual. Esto puede ser útil para ver la composición de los datos, aunque conviene tener cuidado: demasiados intervalos pueden “fragmentar” en exceso la información y dificultar la detección de tendencias.

```
# Customizing bins: Increasing detail with 'breaks'  
hist(gssnet1$age, breaks = 50)
```



Hasta ahora, el histograma muestra cómo se distribuyen realmente los datos de la muestra. Pero en investigación de mercados, a menudo es necesario saber si ese comportamiento es “normal” o si, por el contrario, presenta anomalías o sesgos que deben tenerse en cuenta antes de aplicar pruebas estadísticas avanzadas.

¿Por qué superponer una curva normal?, superponer una curva normal (o Campana de Gauss) sirve para tres propósitos críticos:

1. **Validación de supuestos:** muchos de los modelos que se utilizarán después (como las regresiones o los test t de Student) asumen que los datos siguen una distribución normal. Si al dibujar la curva los datos se alejan mucho de ella, conviene ser cauto con los resultados.
2. **Detección de sesgos:** si la curva normal es simétrica pero el histograma tiene una “cola” muy larga hacia un lado, los datos no están equilibrados y esa variable debe tratarse con técnicas específicas (por ejemplo, transformaciones logarítmicas).
3. **Referencia teórica:** la curva normal es el “estándar de oro”. Permite ver de forma intuitiva si la muestra es homogénea o si presenta valores extremos (datos que caen fuera de la campana) que podrían estar distorsionando la media.

! Importante

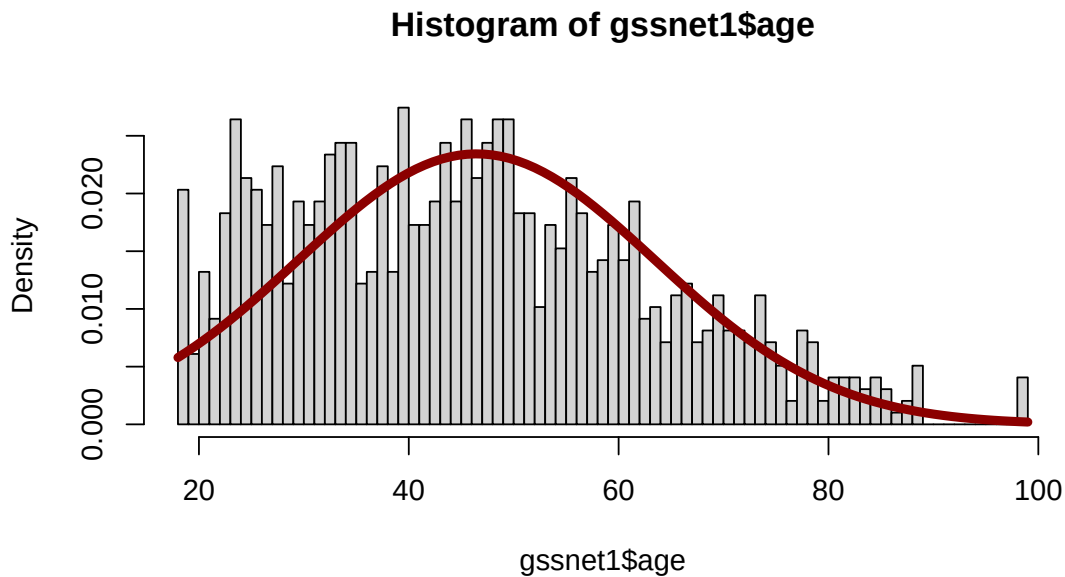
No es preocupante que el histograma no encaje perfectamente con la curva roja. En la vida real, los datos de mercado raramente son “perfectos”. Lo importante no es la perfección, sino identificar si la divergencia es pequeña o si la distribución está totalmente distorsionada.

```
# 1. Calculate the mean and standard deviation of the variable
# We need these two parameters to build a theoretical normal
  ↳ curve that matches our data.
# 'na.rm = TRUE' ensures that missing values do not cause the
  ↳ calculation to fail.
media      <- mean(gssnet1$age, na.rm = TRUE)
desviacion <- sd(gssnet1$age, na.rm = TRUE)

# 2. Create the histogram in probability density mode
# Setting 'probability = TRUE' is mandatory here; it scales the
  ↳ Y-axis to show density
# instead of raw frequencies. This is required so the normal
  ↳ curve fits over the bars.
# The 'breaks' argument customizes the number of bins based on
  ↳ the variable's range.
hist(gssnet1$age,
     breaks = max(gssnet1$age, na.rm = TRUE) - min(gssnet1$age,
  ↳ na.rm = TRUE),
     probability = TRUE
)

# 3. Overlay the theoretical normal distribution curve
# We use 'dnorm' to draw the ideal bell curve based on our
  ↳ specific 'mu' and 'sigma'.
# 'add = TRUE' tells R to draw this curve directly on top of the
  ↳ existing histogram.
curve(dnorm(x, mean = media, sd = desviacion),
     add = TRUE,
     col = "darkred",
```

```
lwd = 5) # Added line width (lwd = 2) to make the curve  
↪ stand out better
```



2.2.3. Resumen estadístico

[?] %
[?]

Resumen estadístico: descriptivos

- Es la forma más rápida de obtener un diagnóstico de una columna o de todo el dataframe.
- Obtiene la “foto fija” numérica de las variables.
- Trabaja con las variables individualmente

`base:summary()` o `psych::describe()`

Hasta ahora se ha explorado la forma de los datos (histogramas). A continuación se obtiene la “foto fija” numérica de las variables. En R existen varias formas de

obtener estos estadísticos; se empieza con la opción nativa (`summary`) y después se presentan herramientas más avanzadas.

Opción estándar: `summary()`

Es la forma más rápida de obtener un diagnóstico de una columna o de todo el dataframe.

```
# Generate a basic statistical summary (min, 1st quartile,
  ↪ median, mean, 3rd quartile, max)
# for a single variable ('age')
summary(gssnet1$age)
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
18.00	33.00	45.00	46.47	58.00	99.00

```
# Generate a summary for every variable in the entire dataframe
# This is a great way to get a quick, global overview of all
  ↪ your data at once
summary(gssnet1)
```

age		educ		usecomp		usenet	
Min.	:18.00	Min.	: 0.00	Min.	:0.0000	Min.	:0.0000
1st Qu.	:33.00	1st Qu.	:12.00	1st Qu.	:0.0000	1st Qu.	:0.0000
Median	:45.00	Median	:14.00	Median	:1.0000	Median	:1.0000
Mean	:46.47	Mean	:13.74	Mean	:0.7896	Mean	:0.7896
3rd Qu.	:58.00	3rd Qu.	:16.00	3rd Qu.	:1.0000	3rd Qu.	:1.0000
Max.	:99.00	Max.	:20.00	Max.	:9.0000	Max.	:9.0000

usemail		emailhrs		webhrs		nethrs	
Min.	:0.0000	Min.	:-9.00	Min.	:-9.000	Min.	: -9.000
1st Qu.	:0.0000	1st Qu.	:-1.00	1st Qu.	:-1.000	1st Qu.	: -1.000
Median	:1.0000	Median	: 1.00	Median	: 1.000	Median	: 3.000
Mean	:0.7419	Mean	: 3.31	Mean	: 2.937	Mean	: 6.795
3rd Qu.	:1.0000	3rd Qu.	: 4.00	3rd Qu.	: 5.000	3rd Qu.	:10.000
Max.	:9.0000	Max.	:50.00	Max.	:70.000	Max.	:118.000

netcat	region	Zage
Min. : -9.0000	Min. : 1.000	Min. : -1.68955
1st Qu.: -1.0000	1st Qu.: 3.000	1st Qu.: -0.79257
Median : 1.0000	Median : 5.000	Median : -0.07499
Mean : 0.2337	Mean : 5.117	Mean : 0.00000
3rd Qu.: 3.0000	3rd Qu.: 7.000	3rd Qu.: 0.70239
Max. : 4.0000	Max. : 9.000	Max. : 2.55613
		NAs : 4

Opción avanzada: `describe()`

Si se necesita una precisión técnica mayor (para investigación de mercados avanzada), el paquete `psych` ofrece una tabla completa con medidas robustas.

```
# Advanced descriptive statistics
psych::describe(gssnet1$age)
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
X1	1	984	46.47	17.02	45	45.49	17.79	18	99	81	0.45	-0.42	0.54

Para tomar decisiones no basta con calcular: hay que saber qué medir. Estos indicadores se clasifican en dos familias:

2.2.4. Medidas de tendencia central

¿Dónde está el “centro”?

Ayudan a entender el valor típico de la muestra.

- **Media:** El promedio aritmético. Es el “centro de gravedad”, aunque conviene tener cuidado: si existen pocos casos con valores muy altos (como ingresos millonarios), la media se verá distorsionada.
- **Mediana:** El valor que deja al 50 % de la muestra por debajo y al 50 % por encima. Es **robusta**: no se deja engañar por valores extremos.
- **Error estándar de la media:** Indica la precisión de la media muestral respecto a la población real.

2.2.5. Medidas de dispersión

¿Cómo de variados son los datos?

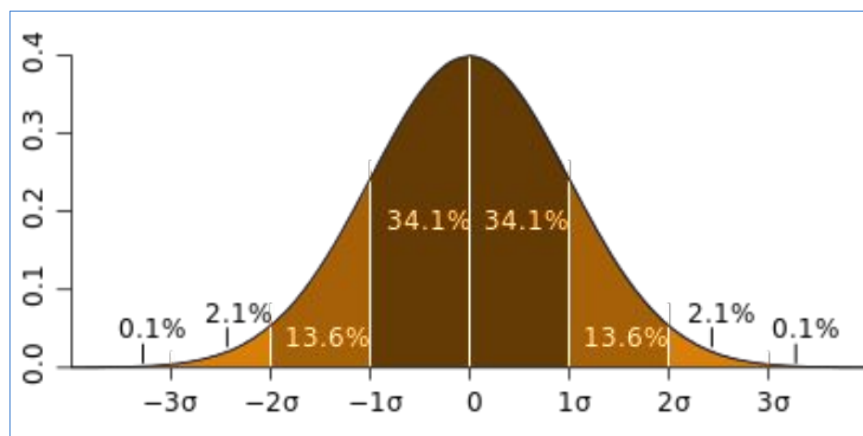
Indican si los datos están muy concentrados en torno a la media o si hay mucha variabilidad.

- **Desviación Estándar:** La medida más clásica. En una distribución normal, permite predecir dónde se encuentra la mayoría de los datos.
- **Varianza:** La desviación al cuadrado. Técnicamente crucial, aunque menos intuitiva para interpretar en marketing.
- **Median Absolute Deviation (MAD):** La compañera de la mediana. Mide la dispersión real ignorando el ruido de los casos extremos. Es una herramienta clave para detectar la variabilidad real.
- **Rango e Intervalos (Min, Max, Quartiles):** Indican dónde empiezan y terminan los datos. El **Rango Intercuartílico (IQR)** es especialmente útil porque ofrece el rango del “50 % central” de los datos, eliminando los extremos.

2.2.6. Concepto de normalidad y relación con la muestra

Se introduce ahora un concepto muy importante: la distribución normal y su relevancia en la investigación de mercados y en el trabajo con muestras.

La distribución normal, o Campana de Gauss, no es solo un gráfico teórico: es la herramienta que permite a los investigadores de mercados realizar **inferencias**, pasando de “lo que dice la muestra” a “lo que probablemente ocurre en todo el mercado”.



Para un valor dado en una muestra aleatoria con un error distribuido normal, la imagen de arriba representa la proporción de muestra que puede caer entre 0, 1, 2 y 3 desviaciones estándar por encima y por debajo del valor real.

16

¿Por qué es tan relevante en la investigación?

1. **La ley de los errores:** En una muestra aleatoria, la mayoría de las observaciones se agrupan en torno al centro (el valor real o la media). Los errores de medición se distribuyen de forma simétrica: es tan probable que una estimación sea un poco mayor como un poco menor que el valor real.
2. **El poder de las desviaciones (σ):** La imagen muestra cómo se distribuye la probabilidad en función de la desviación estándar (σ):
 - **Dentro de $\pm 1\sigma$:** Se encuentra aproximadamente el **68 %** de los datos. Es la “zona central” o lo más común.
 - **Dentro de $\pm 2\sigma$:** Se encuentra el **95 %** de los casos. En investigación de mercados, este es el estándar de confianza habitual.
 - **Más allá de $\pm 3\sigma$:** Solo queda el 0.2 % de los datos (la suma de las colas extremas). Un dato que cae aquí es una anomalía estadística muy improbable.

Esta curva permite cuantificar la **incertidumbre**. Al reportar una media de satisfacción o intención de compra, gracias a la distribución normal puede añadirse un margen de error y afirmar con confianza: *“existe un 95 % de seguridad de que el valor real del mercado se encuentra en este intervalo”*.

Sin este modelo, se estarían dando números aislados sin saber qué tan fiables son. La curva normal es lo que transforma una simple medición en un dato científicamente robusto y generalizable.

Una vez visualizado el histograma, es necesario cuantificar qué tan lejos está la distribución de la forma ideal (la curva normal). Para ello se utilizan dos indicadores:

2.2.6.1. Asimetría (*skewness*)

La asimetría indica si la distribución es equilibrada o si está “estirada” hacia uno de los lados.

- **Simétrica (valor 0):** Media, mediana y moda coinciden. Es la forma perfecta de la curva normal.
- **Asimetría positiva:** La cola se extiende hacia la derecha (valores altos). Es muy común en datos de ingresos o tiempo de navegación: la mayoría está en valores bajos, pero hay unos pocos casos extremos que elevan la media.
- **Asimetría negativa:** La cola se extiende hacia la izquierda (valores bajos). Indica que la mayoría de los casos se concentran en la parte alta.

2.2.6.2. Curtosis (*kurtosis*)

La curtosis no mide hacia dónde se inclina la curva, sino qué tan “puntiaguda” es y, sobre todo, **cuánta importancia tienen los valores extremos (outliers)**.

- **Mesocúrtica (valor 0):** Tiene la forma de la distribución normal estándar.
- **Leptocúrtica (Curtosis positiva):** La curva es más estrecha y puntiaguda. Indica una mayor concentración de datos cerca del centro, pero también una mayor presencia de valores extremos en las colas.
- **Platicúrtica (Curtosis negativa):** La curva es más “achataada”. Indica una mayor dispersión de los datos; no hay una concentración tan clara en el centro.

2.2.6.3. Importancia en investigación de mercados

Muchos de los tests estadísticos utilizados en el máster (como las pruebas de medias) asumen que los datos son simétricos y con una curtosis moderada. Si al calcular la asimetría o la curtosis se obtienen valores muy elevados, esto indica que: 1. la muestra puede tener anomalías (*outliers*) que conviene investigar; 2. quizás sea necesario aplicar una transformación matemática a la variable o utilizar tests no paramétricos más robustos.

En R, el diagnóstico es inmediato: al ejecutar la función `describe()` del paquete `psych`, aparecen estos valores. Un valor de asimetría o curtosis que supere el doble de su error estándar es una señal de alerta clara: la variable no se comporta de forma “normal”.

```
# Generate detailed descriptive statistics for the 'age'
  ↪ variable
# The 'describe' function from the 'psych' package is an
  ↪ excellent alternative to SPSS,
# providing a comprehensive summary that includes mean, standard
  ↪ deviation, median,
# as well as skewness and kurtosis (essential for assessing
  ↪ normality)
psych::describe(gssnet1$age)
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
X1	1	984	46.47	17.02	45	45.49	17.79	18	99	81	0.45	-0.42	0.54

Podemos ver con un ejemplo donde los datos están forzados, las diferentes situaciones.

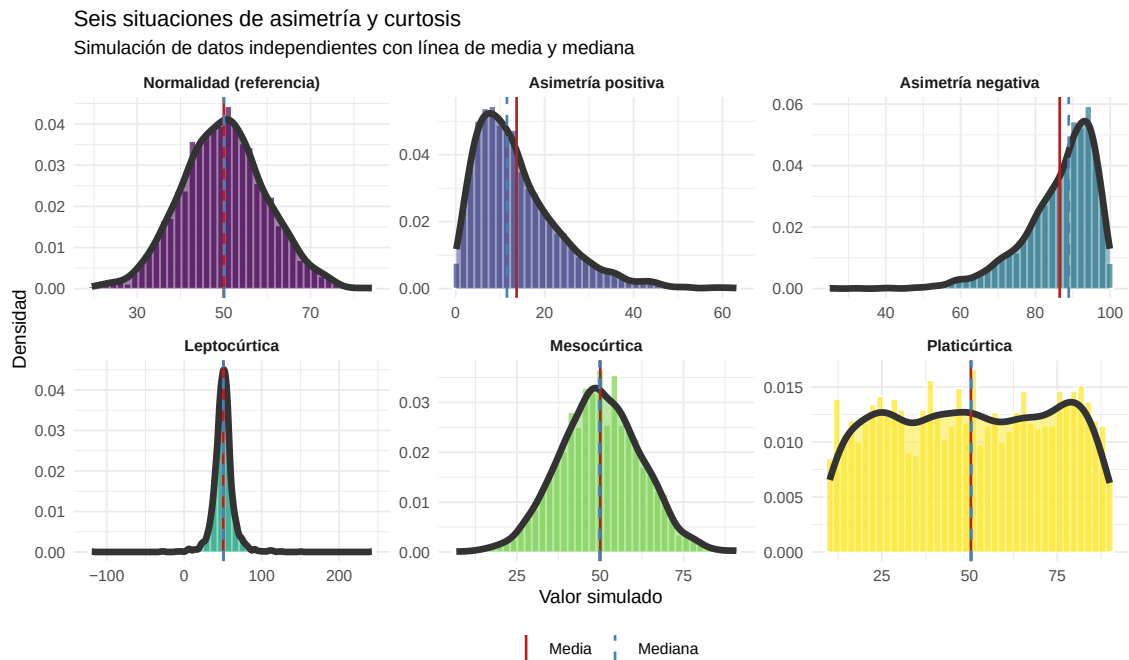


Figura 2.1: Seis situaciones de asimetría y curtosis

A tibble: 6 x 5

scenario	mean_val	median_val	skewness	kurtosis
<fct>	<dbl>	<dbl>	<dbl>	<dbl>
1 Normalidad (referencia)	50.0	50.0	0.0403	-0.0176
2 Asimetría positiva	13.7	11.5	1.31	2.18
3 Asimetría negativa	86.5	88.9	-1.32	2.44
4 Leptocúrtica	50.5	50.5	0.714	33.4
5 Mesocúrtica	50.0	49.8	0.0158	-0.00864
6 Platicúrtica	50.4	50.5	-0.0255	-1.22

- La asimetría indica **hacia dónde se desplaza el sesgo**, mientras que la curtosis indica **la peligrosidad de los extremos**.
- Resulta útil aplicar la regla empírica: no se busca el cero perfecto, sino la ausencia de sesgos significativos que invaliden el análisis.

2.3. Preocupaciones del analista de datos

2.3.1. Valores perdidos (*missing values*)

En investigación, es casi imposible trabajar con bases de datos perfectas. Los valores perdidos (*Missing Values*) son una realidad cotidiana. Sin embargo, no todos los vacíos tienen la misma naturaleza, y la forma de tratarlos determina la validez de las conclusiones.

[?]%

Tareas del investigador: análisis de los valores perdidos

- Depuración de la base de datos
 - Búsqueda de errores, valores fuera de rango permitido, etc.
- Análisis de los valores perdidos
 - Buscar la aleatoriedad de los datos perdidos
 - Patrones en los valores perdidos, mayor importancia que la cuantía
- Diagnóstico
 - Cuantificación, ver su impacto en el banco de datos
 - Revisión del patrón en otras variables (crosstab)
 - Uso de pruebas t para media en muestra independientes (más adelante, sesión 2)
 - Correlaciones dicotomizadas o binarizadas con variables ficticias (más adelante)
- Tratamiento, diferentes métodos
 - Eliminación de todos los casos
 - Eliminación por parejas
 - Imputación de valores medios
 - Imputación de valores aleatorios
 - Imputación de valores aleatorios condicionados
 - Imputación por regresión

2.3.1.1. ¿Por qué faltan los datos?

Antes de actuar, conviene investigar. Lo más importante no es *cuántos* datos faltan, sino el **patrón** de esa ausencia:

- **Aleatoriedad:** ¿faltan los datos al azar? En ese caso, el impacto suele ser mínimo.
- **Patrones sistemáticos:** ¿faltan datos en una variable específica según el perfil del encuestado? Aquí reside el peligro: si un segmento ignora sistemáticamente una pregunta, los resultados presentarán un **sesgo** importante que debe corregirse.

El primer paso es siempre limpiar: eliminar errores de registro y detectar valores fuera de rango. A continuación se cuantifica el impacto. Herramientas como las tablas cruzadas (*crosstabs*) permiten ver si existe una correlación entre tener datos perdidos en una variable y el comportamiento en otras variables.

2.3.1.2. Métodos de corrección

Si el volumen de pérdida es significativo, es necesario aplicar un método de imputación para “rellenar” los huecos:

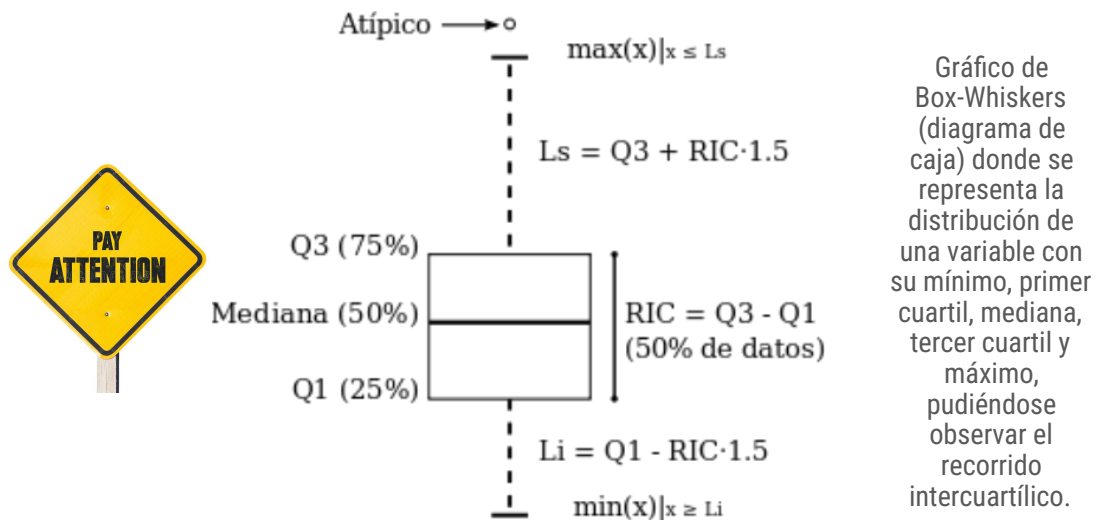
- **Eliminación:** La forma más drástica. Consiste en eliminar casos o variables incompletas (con la consecuencia de reducir la muestra).
- **Imputación simple:** Sustituir por la media o valores aleatorios. Útil, pero puede reducir la variabilidad real de los datos.
- **Imputación avanzada:** Métodos basados en regresión o imputación múltiple, que estiman el valor perdido a partir de la información disponible del mismo encuestado.

La imputación no es magia: es una estimación. Conviene comparar siempre los resultados **antes y después** de la imputación. Si la diferencia es drástica, significa que los datos perdidos ocultaban un comportamiento del mercado que merece ser analizado por sí solo.

2.3.2. Valores atípicos (*outliers*)

2.3.2.1. Identificación visual

Diagrama de caja



18

El diagrama de caja o *Box-Whisker* es una representación visual que resume una variable a través de sus cuartiles. Mientras que el histograma muestra la “forma” de toda la distribución, el *boxplot* es más “quirúrgico”: está diseñado específicamente para detectar **valores atípicos** (*outliers*).

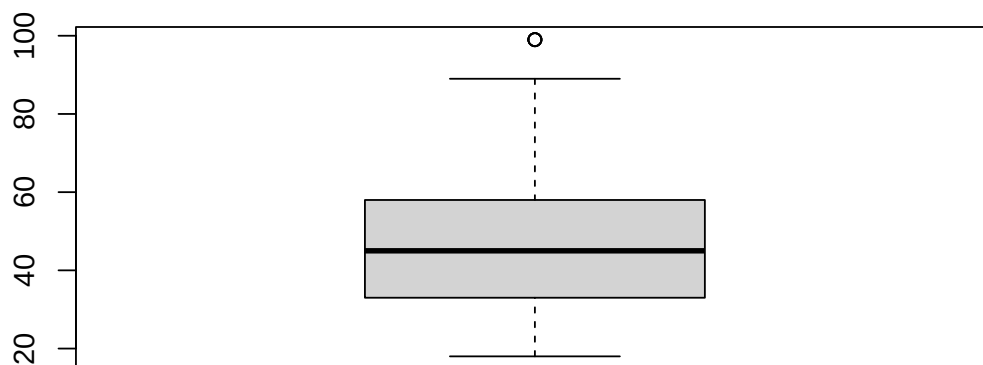
💡 Componentes clave del gráfico

- **La Caja:** Representa el 50 % central de los datos. El borde inferior es el primer cuartil (Q1) y el superior es el tercero (Q3). La línea central es la **mediana**.
- **Los Bigotes (*Whiskers*):** Marcan el límite hasta donde consideramos que los datos son “normales”.
- **Los Puntos (*outliers*):** Cualquier dato que caiga fuera de los límites calculados ($L_s = Q3 + 1.5 * RIC$) ó ($Li = Q1 - 1.5 * RIC$) aparece como un punto aislado.

R permite generar esta radiografía de los datos con una única función:

```
# Create a standard box-and-whisker plot for the 'age' variable
```

```
# This visualization is crucial for understanding the data
↪ spread and quickly
# identifying outliers
boxplot(gssnet1$age)
```



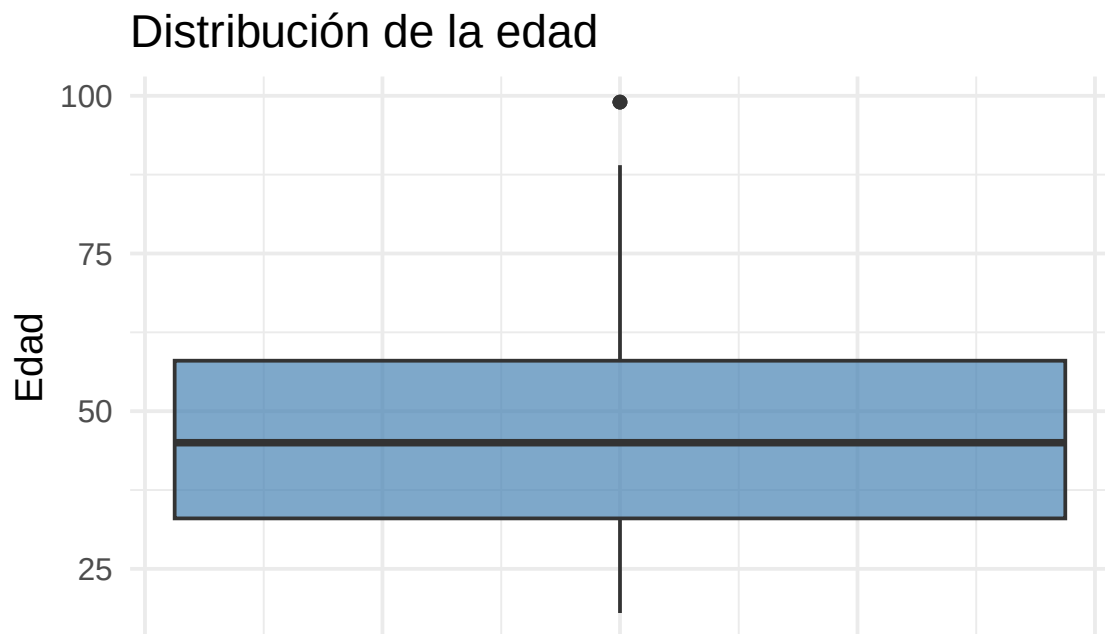
💡 Tip

Un punto fuera de los bigotes no debe eliminarse sin más. ¿Es un error de registro (por ejemplo, una edad imposible)? ¿O es un cliente real con un comportamiento extremo? Los *outliers* a menudo contienen los casos más interesantes de los datos de mercado.

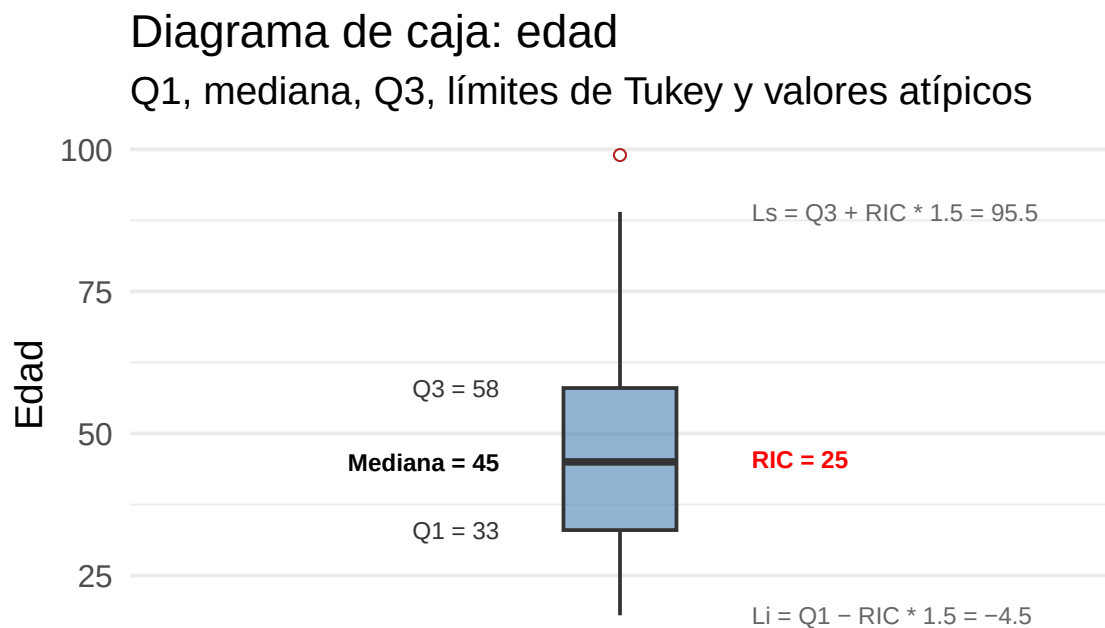
La instrucción anterior usa el gráfico base de R, pero el universo **tidyverse** tiene otro paquete, **ggplot2**, que es el más utilizado para graficar. Aunque su sintaxis es menos sencilla, cuando se conoce, es poderosísima. Los modelos de IA la utilizan cuando se les indica con gran acierto.

```
# Simple boxplot for the 'age' variable
ggplot2::ggplot(gssnet1, aes(y = age)) +
  geom_boxplot(fill = "steelblue", alpha = 0.7) +
  labs(title = "Distribución de la edad", y = "Edad") +
  theme_minimal(base_size = 16) +
```

```
theme(axis.text.x = element_blank(),
      axis.ticks.x = element_blank()
    )
```



Igual que hicimos antes con las distribuciones normales, aquí hacemos un ejemplo de *boxplot* con datos forzados y poniendo etiquetas. No entregamos el código.



2.3.3. ¿Por qué se producen?

Como indicábamos, un *outlier* es un caso con un valor tan extremo que se aleja del comportamiento global de la muestra. Su detección es vital porque, aunque sean pocos, pueden distorsionar drásticamente el cálculo de medias y la precisión de los modelos.

Antes de tomar cualquier decisión, conviene clasificar el *outlier*:

- **Errores:** Datos mal registrados o inconsistencias durante la carga.
- **Sesgos intencionados:** Respuestas deliberadamente falsas o poco serias en encuestas.
- **Errores de muestreo:** Individuos que no pertenecen realmente al perfil de mercado que estamos estudiando.
- **Outliers legítimos:** Casos reales, aunque extremos. Son los “clientes de alto valor” o “usuarios intensivos” que, lejos de ser eliminados, suelen ser los más interesantes para el negocio.

2.3.3.1. Detección

- **Univariante:** Se detectan valores extremos en una sola variable (por ejemplo, alguien que gasta mucho más que el resto). Mediante estandarización.
- **Bivariante y multivariante:** Se detectan anomalías en la combinación de variables (por ejemplo, alguien con una edad muy baja pero unos ingresos muy altos). Mediante medidas de distancias multivariantes (lo vemos más adelante).

El *outlier* plantea una pregunta, no una solución automática. Eliminar un dato es una medida irreversible que requiere una justificación clara: ¿este dato extremo es una mentira del sistema o una verdad incómoda del mercado?

Para decidir qué es un valor extremo, los datos se estandarizan (transformándolos a una distribución con media 0 y desviación 1). Como regla general, un caso que se aleja más de 3 desviaciones típicas del centro es estadísticamente sospechoso.

¿Cómo podría, por ejemplo, estandarizarse la variable `age` de `gssnet1` (aunque no es necesario, pero a título ilustrativo)?

```
# --- Step 1: Get the mean and standard deviation of 'age' ---  
# Compute arithmetic mean
```

```

z_mu = mean(gssnet1$age, na.rm = TRUE)
# Compute standard deviation
z_sd = sd(gssnet1$age, na.rm = TRUE)

# --- Step 2: Apply the Z-score formula: Z = (x - mean) / sd ---
# Each value tells us how many standard deviations a
  ↪ respondent's age
# is above (positive) or below (negative) the sample mean
gssnet1$valor_z <- round((gssnet1$age - z_mu) / z_sd, 3)

# Note: fre() works best with categorical/discrete variables.
# On a continuous variable like 'valor_z' it will list one row
  ↪ per unique
# value, so the table will be long - useful only to spot-check
  ↪ extreme values

tail(fre(abs(gssnet1$valor_z)),10)

```

Age of respondent	Count	Valid percent	Percent	Responses, %	Cumulative responses, %
2.146	4	0.4	0.4	0.4	97.9
2.205	3	0.3	0.3	0.3	98.2
2.263	4	0.4	0.4	0.4	98.6
2.322	3	0.3	0.3	0.3	98.9
2.381	1	0.1	0.1	0.1	99.0
2.44	2	0.2	0.2	0.2	99.2
2.498	5	0.5	0.5	0.5	99.7
3.086	4	0.4	0.4	0.4	100.0
#Total	984	100	100	100	
<NA>	0		0.0		

Y se caracterizan ahora sus valores...

```
psych::describe(gssnet1$valor_z)
```

```

vars  n mean sd median trimmed  mad   min  max range skew kurtosis   se
X1    1 984   0  1  -0.09  -0.06 1.05 -1.67  3.09  4.76 0.45   -0.42 0.03

```

2.3.3.2. Métodos de corrección

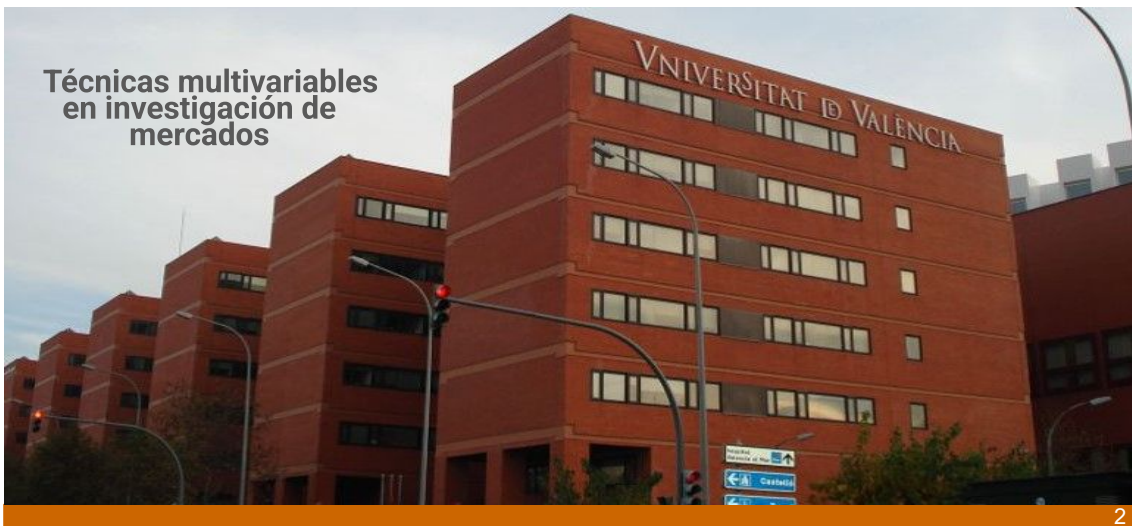
La detección de un valor atípico no implica automáticamente su eliminación. Si el outlier es un **error de registro** —una edad de 187 años, un gasto negativo, un código fuera de rango—, la corrección es directa: se sustituye por el valor correcto si puede recuperarse, o se trata como dato perdido (NA) si no es posible verificarlo.

Si, en cambio, el valor extremo corresponde a un **caso real**, las opciones son más matizadas:

- **Conservarlo:** cuando el objetivo es describir la realidad tal como es, eliminarlo sesgaría los resultados.
- **Winsorización:** sustituir el valor extremo por el percentil más próximo aceptable (p. ej., el percentil 95), reduciendo su influencia sin eliminarlo.
- **Transformación de la variable:** aplicar logaritmo o raíz cuadrada comprime la escala y reduce el peso de los extremos.
- **Análisis separado:** tratar los casos extremos como un segmento propio, especialmente relevante en mercados de lujo o de comportamiento de nicho.

En investigación de mercados conviene recordar que los *outliers* a menudo contienen los casos más interesantes: el gran comprador, el detractor radical o el cliente evangelizador suelen esconderse precisamente en los extremos de la distribución.

3 Inferencia e Investigación de Mercados



Tenemos las bases para introducirnos en el mundo de la inferencia estadística, es decir, responder a esas preguntas que planteábamos como esenciales:

- Existen diferencias
- Existe relación

Vamos con ello.

3.1. Introducción al proceso inferencial



¿Qué vamos a hacer?

Inferencia estadística ¿qué es?

- Inferencia estadística
- Pruebas paramétricas vs pruebas no paramétricas
- Para elegir entre pruebas paramétricas y no paramétricas a la hora de explotar los datos, hemos de comprobar si se cumplen, básicamente, los supuestos:
- Nivel de medición de intervalo / razón (métrica);
 - Independencia de las observaciones;
 - Normalidad;
 - En algunos casos, el proceso de linealización, puede ayudar a entender la normalidad de la distribución
 - Cuando se comparan grupos
 - Homocedasticidad entre los grupos generados en una relación;
 - Independencia de los grupos;

El proceso de una prueba inferencia, contraste estadístico

- Las pruebas para comprobarlo se basan en la inferencia estadística y su procedimiento es:
 - Generar una hipótesis (hipótesis experimental vs. hipótesis nula)
 - Ajustar un modelo a los datos (elección de la técnica)
 - Calcular el test asociado (varianza explicada / varianza no explicada por el modelo)
 - Concluir acerca de la verosimilitud de la hipótesis experimental atendiendo al resultado del test (valor crítico, nivel de significación, nivel de significación crítico)
 - 0.05 (el más habitual)
 - 0.01 (en ocasiones ...)
 - 0.001 (muy extremo ...)
 - Decisión:
 - Rechazar H_0
 - No poder rechazar H_0

19

3.1.1. La lógica de la inferencia estadística

Hasta ahora se ha resumido lo que ocurre en la muestra. La **inferencia estadística** es el paso definitivo que permite saltar de los datos observados a conclusiones sobre la realidad del mercado. Aquí no se busca solo medir, sino verificar si lo observado es fruto de un patrón real o de simple azar.

Cuando hablamos de inferencia, todo contraste sigue un proceso riguroso de cuatro etapas:

1. **Definición de hipótesis:** Se formula una hipótesis experimental (lo que se pretende demostrar) frente a la **hipótesis nula** (H_0), que asume que no hay efecto alguno (todo es azar).
2. **Elección del modelo:** Se selecciona la técnica adecuada en función de la naturaleza de los datos. Para ello es necesario validar los supuestos:
 - **Nivel de medida:** ¿son datos métricos o categóricos?
 - **Normalidad:** ¿siguen una distribución de Gauss?
 - **Homocedasticidad:** ¿tienen los grupos la misma variabilidad?
3. **Cálculo del test:** Se evalúa qué parte de la varianza es explicada por el modelo y qué parte es ruido. Aquí se obtiene el **valor crítico** (*p-value*).

4. **Toma de decisiones:** El resultado se compara con un nivel de significación (usualmente 0.05).

- Si $p < 0.05$: se rechaza H_0 y se acepta el efecto como estadísticamente significativo.
- Si $p > 0.05$: no puede rechazarse H_0 , lo que implica que no hay evidencia suficiente para afirmar que el efecto existe.

La clave del analista es elegir la herramienta correcta. Si los datos cumplen los supuestos (normalidad, homocedasticidad), se utilizan **pruebas paramétricas** (más potentes). Si no se cumplen, se recurre a **pruebas no paramétricas**, más flexibles y “robustas” ante datos que no siguen una distribución perfecta.

Es vital insistir en que el 0.05 no es un “número mágico”, sino un **criterio de convención científica**. En marketing, un 0.05 ofrece la tranquilidad de que, si el estudio se repitiera 100 veces, solo en 5 de ellas se habría afirmado erróneamente que existe un efecto real.

En R se aprenderá a extraer el *p-value* de forma específica, lo que obliga a ser conscientes de **qué test se está realizando** en cada momento. Esto evita el error común de aplicar pruebas automáticas sin entender la hipótesis que hay detrás.

3.2. Normalidad estadística



Test paramétricos para el contraste de la normalidad univariante

- Test de Kolmogorov-Smirnov, con la corrección de Lilliefors para muestras pequeñas, y Shapiro-Wilk para muestras grandes:
 - Compara los valores de la distribución real con una distribución normal que tuviera la misma media y desviación típica.
 - Hipótesis nula (H_0): la distribución es normal.
- Advertencia:
 - Con muestras grandes hay una fuerte tendencia a rechazar la hipótesis nula, aunque los desvíos de la normalidad sean pequeños.
- Realizar siempre los test:
 - También obtener siempre los gráficos para tomar una decisión bien informada respecto al alcance de los problemas de normalidad.

Un ejemplo muy básico:

- Banco de datos para los contrastes: fib2.sav
 - Partamos de los datos de la medida de los niveles de higiene entre los participantes (público) del FIB de Benicàssim el primer, el segundo y el tercer día del mismo.
- Escala utilizada:
 - Valor métrico entre 0 (poca higiene corporal) a 5 (buena higiene corporal).

Antes de utilizar técnicas estadísticas avanzadas, es necesario validar si la variable sigue una distribución normal. Este es el supuesto base de muchas pruebas paramétricas (como el test t o el ANOVA). Para verificarlo no basta con la intuición visual; son necesarios contrastes estadísticos formales.

3.2.1. Herramientas de diagnóstico

- **Test de Shapiro-Wilk:** Es el estándar de referencia para evaluar la normalidad. Bajo H_0 , los datos siguen una distribución normal. Es el más potente para detectar desviaciones reales, especialmente en muestras pequeñas o medianas ($n < 200$).
- **Test de Kolmogorov-Smirnov con corrección de Lilliefors:** Alternativa habitual que ajusta el test K-S original para el caso —el más frecuente en la práctica— en que la media y la desviación típica se estiman a partir de la propia muestra. En términos de potencia es algo inferior a Shapiro-Wilk, pero su uso está muy extendido y es la opción que ofrecen la mayoría de los programas estadísticos de forma predeterminada.

3.2.1.0.1. La trampa de la muestra grande

Al aumentar el tamaño muestral, **ambos tests** se vuelven extremadamente sensibles: en bases de datos con cientos o miles de casos, cualquier desviación mínima e irrelevante produce $p < 0.05$. No debe confundirse *significación estadística* con *relevancia práctica*.

Por eso, la decisión sobre normalidad nunca debe basarse únicamente en un p-valor. El protocolo recomendado es:

1. **Ejecutar el test** (Shapiro-Wilk como primera opción; Lilliefors como contraste adicional).
2. **Visualizar:** histograma y gráfico Q-Q. Si el test rechaza la normalidad pero el gráfico muestra una campana razonable, prevalece el criterio del investigador.
3. **Decidir:** si el sesgo es real y relevante, se opta por pruebas no paramétricas o por transformaciones de los datos.

En investigación de mercados, las variables casi nunca son perfectamente normales. El objetivo no es la perfección, sino la **suficiencia**. Ante la duda, el gráfico revela lo que el p-valor oculta.

3.2.1.1. Testar la normalidad

Kolmogorov-Smirnov (Lilliefors) y Shapiro-Wilk

Una vez ejecutados los tests de normalidad, llega el momento de la verdad: ¿pueden aplicarse técnicas paramétricas, o es necesario buscar alternativas? En ambos tests, la **hipótesis nula** (H_0) es que “**los datos siguen una distribución normal**”.

- **Si el $p\text{-value} > 0.05$:** no hay evidencia suficiente para rechazar la hipótesis nula. Por tanto, **se asume normalidad**.
- **Si el $p\text{-value} < 0.05$:** se rechaza la normalidad. La distribución se aleja significativamente de la campana de Gauss.

Si vamos con el ejemplo, al aplicar los tests a la variable `dia1` (higiene corporal, día 1 del FIB):

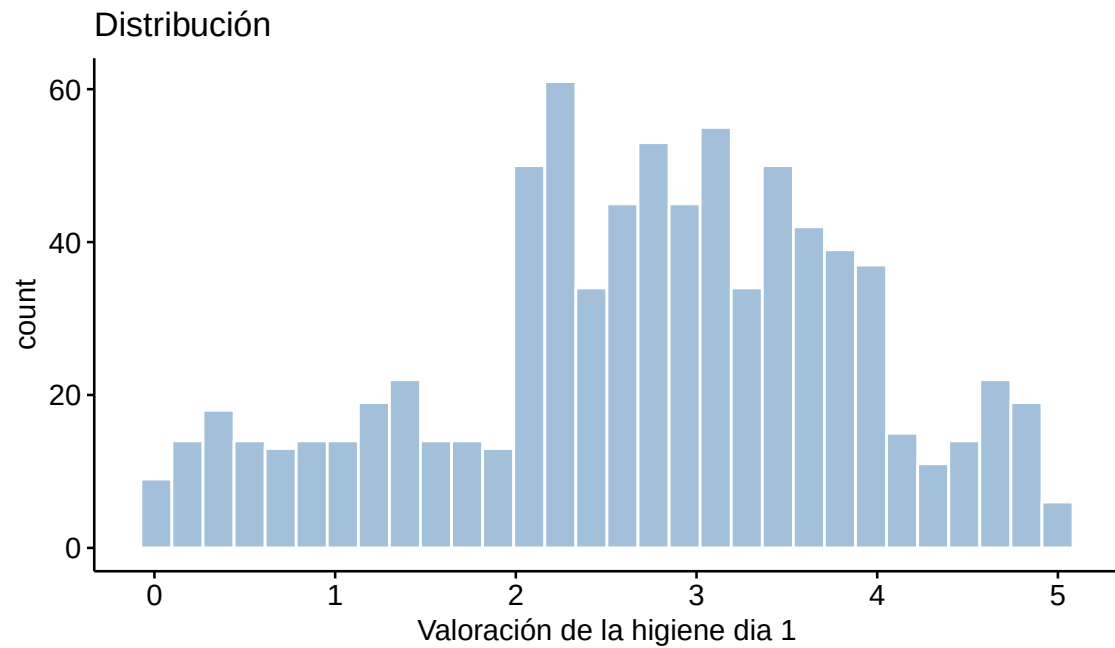
```
cat("H0 → los datos siguen una distribución normal\n\n")

# --- Load the FIB hygiene dataset ---
fib2 <- expss::read_spss('data/fib2.sav')

# --- Lilliefors test (Kolmogorov-Smirnov with Lilliefors
  ↪ correction) ---
# H0: the variable follows a normal distribution
nortest::lillie.test(fib2$dia1)

# --- Shapiro-Wilk test ---
# H0: the variable follows a normal distribution (most powerful
  ↪ test for small/medium n)
stats::shapiro.test(fib2$dia1)

# Basic gghistogram, no overlay
ggpubr::gghistogram(
  fib2, x = "dia1",
  fill = "steelblue", color = "white",
  title = "Distribución"
)
```



$H_0 \rightarrow$ los datos siguen una distribución normal

Lilliefors (Kolmogorov-Smirnov) normality test

```
data: fib2$dia1
```

D = 0.053023, p-value = 1.288e-05

Shapiro-Wilk normality test

```
data: fib2$dia1
```

W = 0.97795, p-value = 1.043e-09

Las conclusiones que se extraen de estos resultados representan uno de los aprendizajes más importantes para un investigador o analista de datos. Es un caso clásico de “**significación estadística frente a relevancia práctica**”.

Ambos tests plantean como hipótesis nula (H_0) que los datos provienen de una distribución normal.

- En el test de **Lilliefors**, el *p-value* es 0.00001288 (< 0.05).
- En el test de **Shapiro-Wilk**, el *p-value* es 0.000000001043 (< 0.05).

Conclusión técnica: se rechaza la hipótesis nula en ambos casos. Estadísticamente hablando, la variable `dia1` **no sigue una distribución perfectamente normal**.

Aquí es donde entra la experiencia analítica. La muestra es de **810 casos**, un tamaño muestral bastante grande. Como se ha comentado, cuando la muestra es grande, los tests de bondad de ajuste (especialmente Shapiro-Wilk) se vuelven **hipersensibles**. Tienen tanto “poder estadístico” que detectan cualquier mínima imperfección o asimetría microscópica en la campana de Gauss, devolviendo un *p-value* significativo que indica “no es normal”.

Sin embargo, hay un detalle crucial en el test de Shapiro-Wilk: **el valor de su estadístico es $W = 0.97795$** . El estadístico W oscila entre 0 y 1, donde 1 significa una normalidad absoluta. Un valor de **0.978** indica que, en la práctica, la curva es extremadamente parecida a una normal. El test la rechaza por purismo matemático, pero visualmente se trata de una campana muy decente.

Con estos resultados sobre $n = 810$, el investigador de mercados puede tomar las siguientes decisiones:

1. **Ignorar el alarmismo del *p-value*:** no debe descartarse automáticamente el uso de técnicas paramétricas (como el Test t o el ANOVA).
2. **Revisión visual obligatoria:** el siguiente paso inmediato es dibujar el histograma y el gráfico Q-Q (*Q-Q plot*). Si el histograma tiene forma de campana y no hay valores atípicos (*outliers*) extremos y deformantes, puede confiarse en la forma de los datos.
3. **El escudo del Teorema Central del Límite:** con 810 casos, la muestra es tan grande que el Teorema Central del Límite protege los análisis paramétricos. Las pruebas de comparación de medias funcionarán perfectamente y serán robustas aunque la distribución original presente esa leve desviación detectada por Lilliefors y Shapiro.

3.2.1.2. Otros diagnósticos visuales

Para complementar el histograma se utilizan los gráficos **Q-Q (*Quantile-Quantile Plot*)**. Su lógica es sencilla: representan los cuantiles de los datos frente a los cuantiles de una distribución normal teórica.

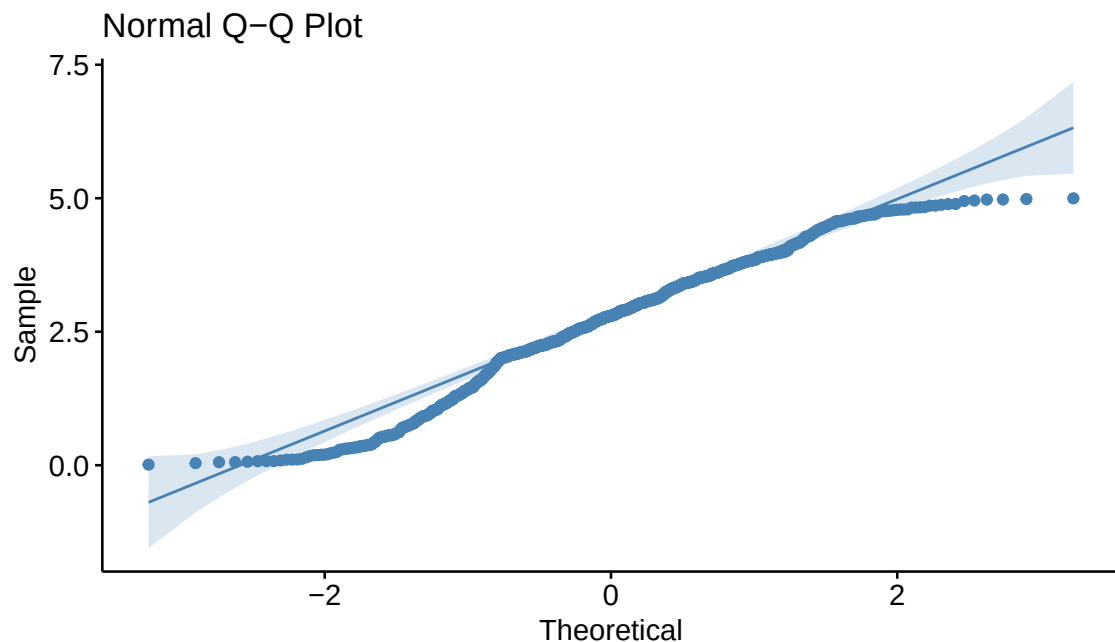
- **Si los puntos siguen la línea diagonal:** Los datos se ajustan a la normalidad.

- **Si los puntos se curvan en los extremos:** Indica colas más pesadas (más *outliers* de los esperados) o una asimetría clara.

3.2.1.2.1. Gráfico Q-Q estándar

Es la primera parada para validar la normalidad.

```
# 1. Standard Q-Q Plot
ggpubr::ggqqplot(fib2$dial1,
  main = "Normal Q-Q Plot",
  color = "steelblue")
```



3.2.1.3. Guía rápida para los informes

1. **¿Muestra pequeña ($n < 50$)?:** Usar **Shapiro-Wilk**. Es el más potente para detectar desviaciones reales de la normalidad con pocos datos.
2. **¿Muestra grande ($n > 200$)?:** Los tests formales —tanto Shapiro-Wilk como Lilliefors— se vuelven hipersensibles: cualquier desviación mínima produce $p < 0.05$, aunque sea irrelevante en la práctica. **Confiar principalmente en el histograma y el gráfico Q-Q**, y usar los tests solo como referencia complementaria.

3. **¿Duda eterna?**: Como siempre, **el histograma**. Un $p < 0.05$ con una distribución visualmente simétrica y acampanada no obliga a abandonar las pruebas paramétricas.

3.3. El procedimiento de linealización

El procedimiento de linealización



- Cuando una distribución no es normalmente estadística, es decir, se rechaza la H_0 de normalidad mediante las pruebas de KS y/o SW, se puede intentar linealizar.
- Hablamos en este caso de un proceso de linealización univariante, que suaviza la escala.
- Usamos la transformación de variables, realizando cálculos muy sencillos:
 - Obtener la raíz cuadrada
 - `sqrt()`
 - Obtener el logaritmo base 10
 - `log10()`
 - Obtener el logaritmo neperiano
 - `log()`
- Son fórmulas que se aplican directamente sobre los datos.
- El procedimiento creará nuevas variables con las que trabajar y recalcular para ver si ese proceso las habilita.



21

Cuando los datos no siguen una distribución normal, no siempre es necesario renunciar a las técnicas paramétricas. A menudo, la falta de normalidad es solo un efecto visual provocado por una escala de medida desproporcionada. La **linealización** es el proceso de aplicar transformaciones matemáticas a las variables para “suavizar” su distribución y aproximarla a la normalidad.

3.3.0.1. Transformaciones comunes en R

En R, estas transformaciones se aplican directamente sobre la columna del *dataframe*, creando nuevas variables que conservan la información original pero con una escala más apta para el análisis estadístico:

- **Raíz cuadrada (`sqrt()`)**: Reduce la variabilidad en datos con una asimetría positiva moderada.
- **Logaritmo base 10 (`log10()`)**: Muy útil para variables con rangos muy amplios (ej. ingresos o población).

- **Logaritmo neperiano ($\log()$):** El estándar en modelización económica y de marketing para estabilizar la varianza.



El procedimiento de linealización ...

La transformación de datos (como la raíz cuadrada, el logaritmo natural o el logaritmo base 10) se utiliza para modificar la forma de la distribución de una variable para que se ajuste mejor a los supuestos de las pruebas paramétricas:

- Corrige la Asimetría: Las transformaciones logarítmicas son especialmente efectivas para reducir la asimetría (sesgo) positiva, haciendo que la distribución se acerque a la simetría de la distribución normal.
- Estabiliza la Varianza (Homocedasticidad): Algunas transformaciones también ayudan a que las varianzas sean más homogéneas, un supuesto clave en ANOVA y regresión.

No es correcto usar la variable en su modo original en la inferencia paramétrica solo porque su versión transformada superó la prueba de normalidad.

Si la variable original no es normal y se usa una prueba paramétrica, los resultados serán inválidos o, al menos, poco fiables. La transformación se realiza precisamente para cumplir el supuesto de normalidad y es la variable transformada la que debe usarse en la inferencia paramétrica.

La variable transformada se utiliza para el cálculo estadístico (**inferencia paramétrica**), y la variable original (o la transformación inversa) se utiliza para la interpretación y reporte de los resultados. No debes usar la variable original para el cálculo si no cumple los supuestos de normalidad y tu muestra es pequeña.

22

```
# --- Step 1: Create transformed copies of 'dia1' in the
  ↳ dataframe ---
fib2$dia1_sqrt    <- sqrt(fib2$dia1)
fib2$dia1_log     <- log(fib2$dia1 + 1)  # +1 avoids errors when
  ↳ there are zero values
fib2$dia1_log10   <- log10(fib2$dia1 + 1) # +1 avoids errors when
  ↳ there are zero values

# --- Step 2: Re-run Shapiro-Wilk to check if normality has
  ↳ improved ---
stats::shapiro.test(fib2$dia1_sqrt)
```

Shapiro-Wilk normality test

```
data:  fib2$dia1_sqrt
W = 0.9165, p-value < 2.2e-16
```

```
stats::shapiro.test(fib2$dia1_log)
```

Shapiro-Wilk normality test

data: fib2\$dia1_log

W = 0.89497, p-value < 2.2e-16

```
stats::shapiro.test(fib2$dia1_log10)
```

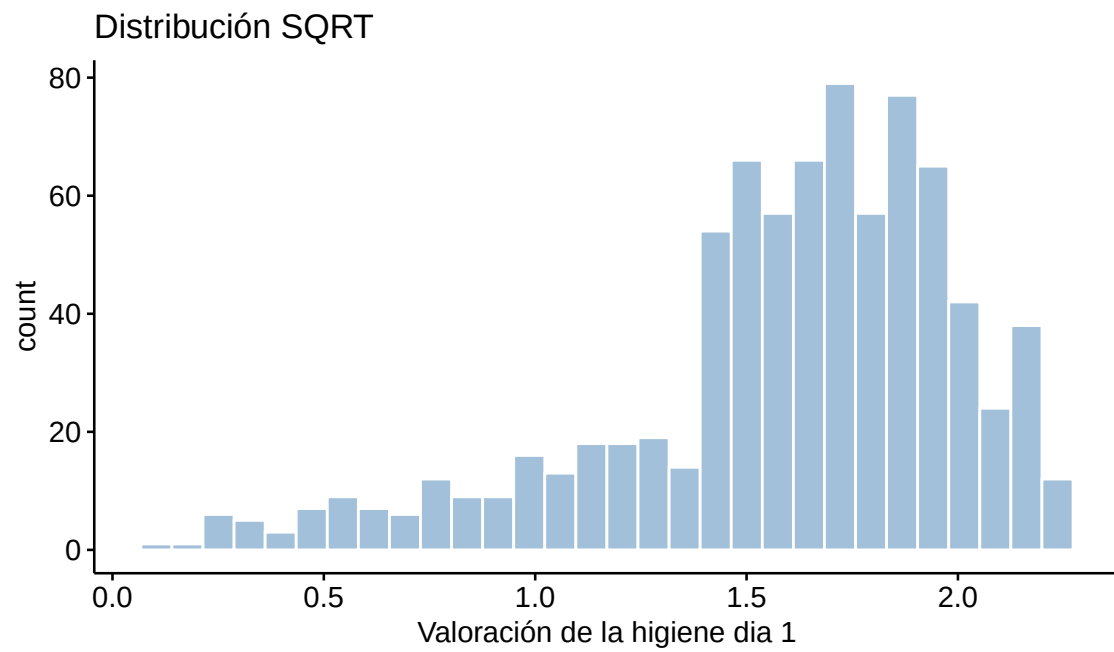
Shapiro-Wilk normality test

data: fib2\$dia1_log10

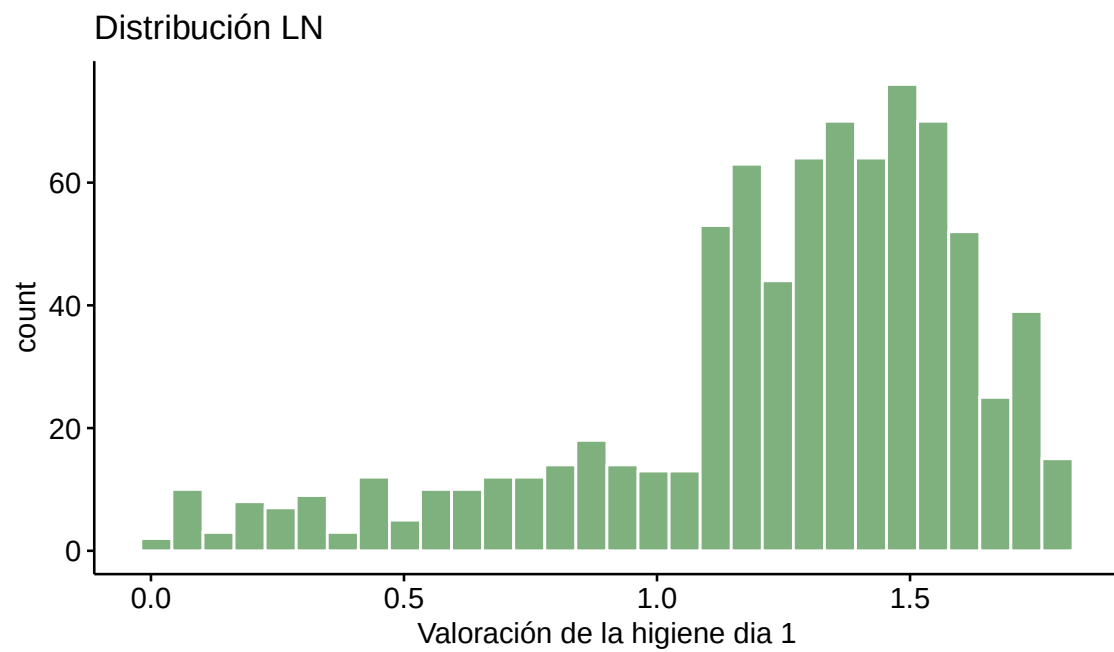
W = 0.89497, p-value < 2.2e-16

```
# --- Step 3: Compare the three transformed distributions
↪ visually ---
# par(mfrow = c(1, 3)) arranges the next 3 plots side by side in
↪ one row
par(mfrow = c(1, 3))

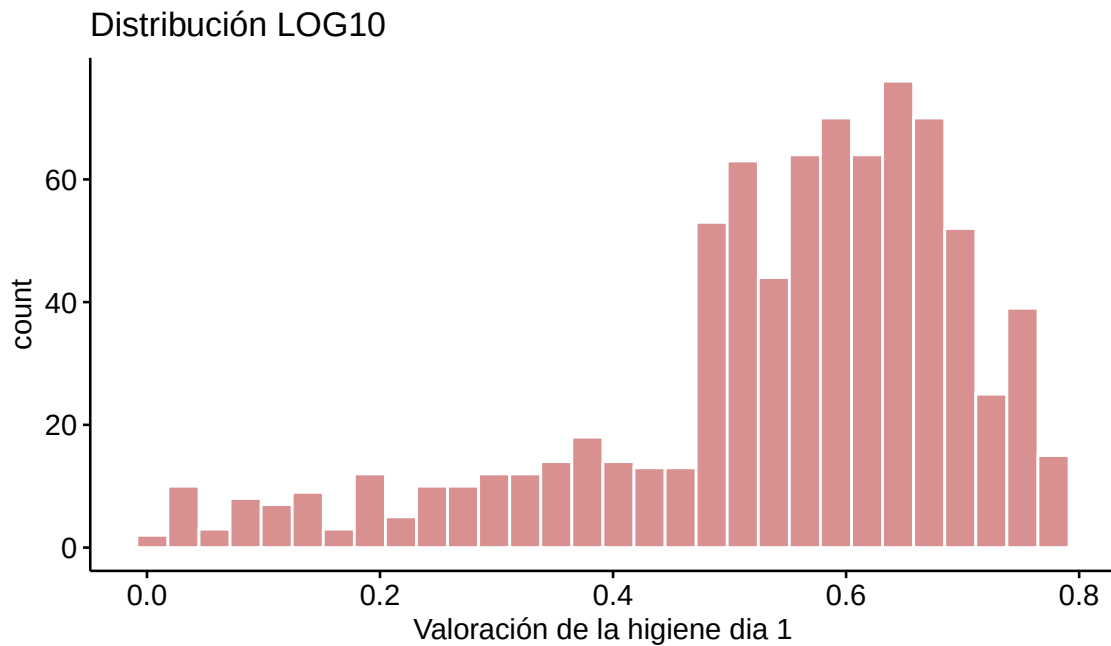
ggpubr::gghistogram(
  fib2, x = "dia1_sqrt",
  fill = "steelblue", color = "white",
  title = "Distribución SQRT"
)
```



```
ggpubr::gghistogram(
  fib2, x = "dia1_log",
  fill = "darkgreen", color = "white",
  title = "Distribución LN"
)
```



```
ggpubr::gghistogram(
  fib2, x = "dia1_log10",
  fill = "firebrick", color = "white",
  title = "Distribución LOG10"
)
```



```
# Reset the plotting layout back to a single plot per figure
par(mfrow = c(1, 1))
```

Nota importante: al realizar estas transformaciones, **cambia la interpretación de la variable**. La media de una variable logarítmica no se lee en las mismas unidades que la original. Por eso, el procedimiento correcto siempre es: transformar, realizar el análisis paramétrico y, después —si es necesario—, interpretar los resultados en el contexto de la escala original.

3.4. Homogeneidad de varianzas (homoscedasticidad)

La comprobación de la homogeneidad de varianzas, también conocida como homoscedasticidad, es un pilar innegociable antes de aplicar técnicas de inferencia estadística paramétrica como el test t de Student o el ANOVA. Estas pruebas matemáticas asumen desde su origen que los grupos que se comparan tienen una variabilidad similar; es decir, que sus datos se dispersan de forma parecida en torno a su propia media. Si se ignora este paso y resulta que las varianzas son radicalmente distintas, los cálculos del error estándar se distorsionan.

Esto resulta especialmente peligroso en investigación de mercados, ya que aumenta drásticamente la probabilidad de cometer un falso positivo (afirmar que hay diferencias estratégicas entre dos segmentos de clientes cuando en realidad no las hay). Validar la homoscedasticidad garantiza que los grupos se comparan bajo unas reglas del juego justas y equilibradas. Sin este diagnóstico previo, cualquier decisión de negocio basada en diferencias de medias carecerá de validez y rigor científico.

3.4.0.1. Test de Levene



Test paramétricos para el contraste de la homoscedasticidad: test de Levene

- Test de Levene
 - Debe realizarse con dos variables: una variable que determina el grupo y otra variable de la cual se mide su dispersión (métrica) su media (o mediana...)
- Hipótesis nula:
 - Homoscedasticidad, es decir, que las varianzas (dispersiones) obtenidas en los grupos a testar, son iguales
 - Con muestras grandes hay una fuerte tendencia a rechazar la hipótesis nula, aunque las varianzas no sean muy distintas.
- Consejo:
 - Realizar siempre el test, pero también obtener siempre los gráficos para tomar una decisión bien informada.
 - Si no hay nada que hacer, realizar las transformaciones logarítmicas (LG10 ó LN) o potenciales (SQRT).

```
# --- Levene's test for homogeneity of variances ---  
# H0: Variances are equal. H1: Variances are different.
```

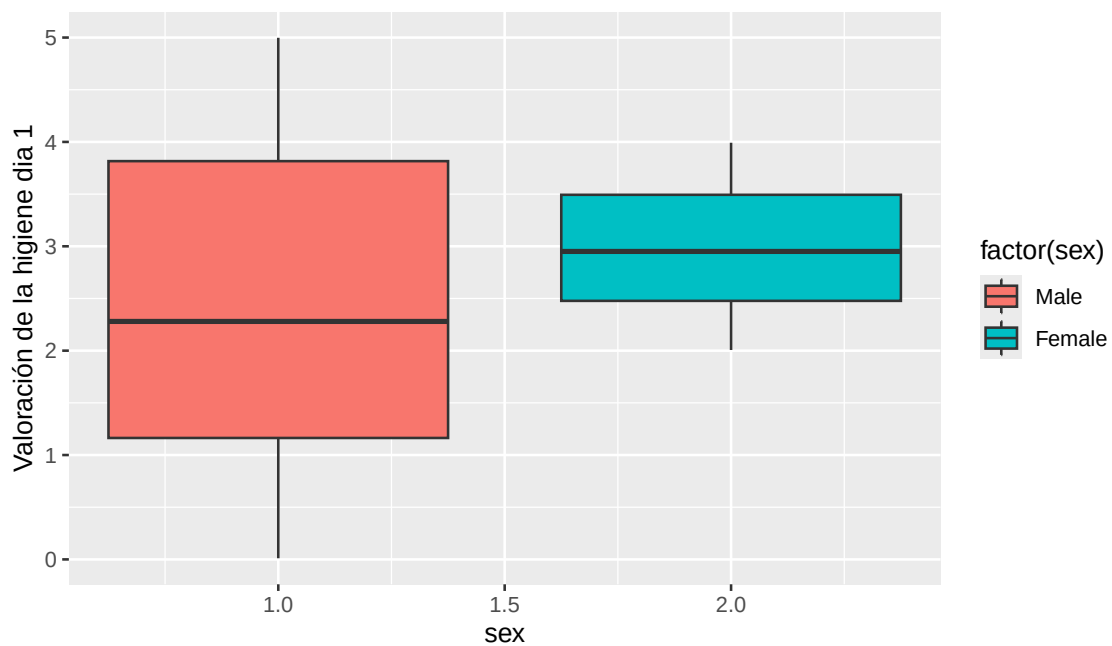
```
car::leveneTest(dia1 ~ factor(sex), data = fib2)
```

Levene's Test for Homogeneity of Variance (center = median)

```
      Df F value    Pr(>F)
group  1  388.54 < 2.2e-16 ***
      808
```

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
# --- Visual proof: compare the spread of 'dia1' between sex
  ↳ groups ---
ggplot2::ggplot(fib2, aes(x = sex, y = dia1, fill =
  ↳ factor(sex))) +
  geom_boxplot()
```



Este análisis se suele complementar con gráficos de caja e histogramas, así como con el cálculo de los ratios de varianza, una medida que **SPSS no proporciona**.

3.4.0.2. Ratio de varianzas



Otra forma de medir la homoscedasticidad

- Ratio de varianzas
 - Otra forma más directa y simplificada de testar la homoscedasticidad
 - Comprobar si la ratio de las varianzas de la variable dependiente en los dos grupos fuera menor que 2 (asumimos en este caso la homoscedasticidad independientemente del test de Levene).
 - Cociente entre las varianzas calculadas

24

Para calcular el **ratio de varianzas** (también conocido como *F-test ratio*) de forma manual en R, basta con dividir la varianza del grupo que tiene mayor dispersión entre la del grupo con menor dispersión. Esto da un índice de cuántas veces es más grande la variabilidad de un grupo respecto al otro.

```
# 1. Calculate variances for each group
var_male    <- var(fib2$dial[fib2$sex == 1], na.rm = TRUE)
var_female  <- var(fib2$dial[fib2$sex == 2], na.rm = TRUE)

# 2. Calculate the ratio (Var_max / Var_min)
# We use max() so the ratio is always >= 1, making it easier to
↪ interpret
variance_ratio <- max(var_male, var_female) / min(var_male,
↪ var_female)

# 3. Print the result
cat("Variance of Male group:", round(var_male, 3), "\n")
```

Variance of Male group: 2.237

```
cat("Variance of Female group:", round(var_female, 3), "\n")
```

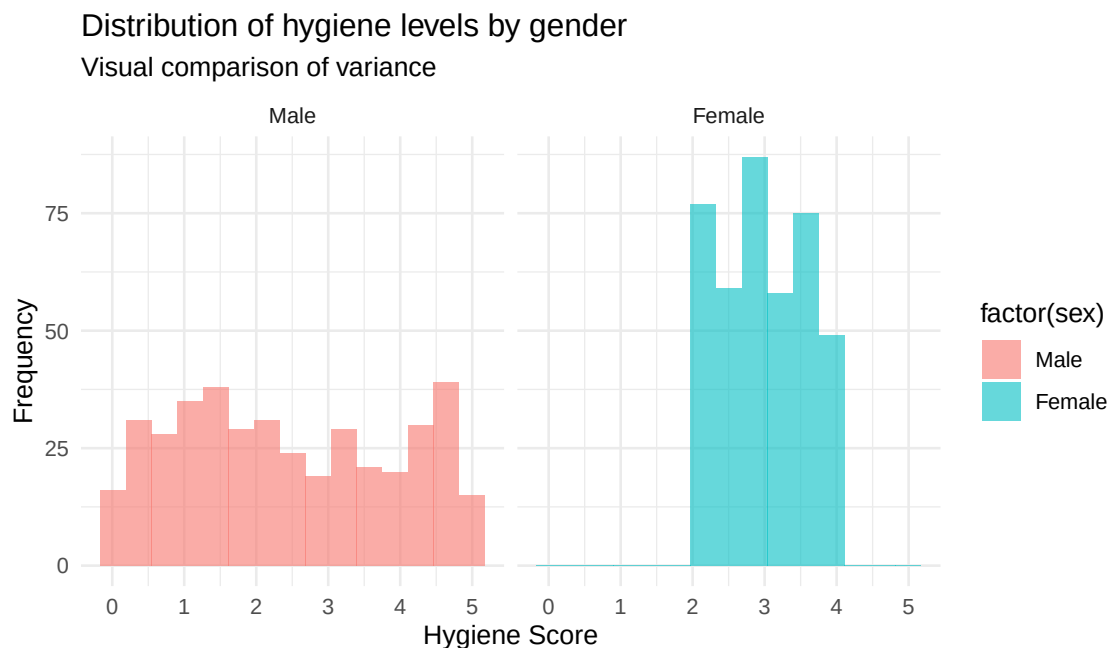
Variance of Female group: 0.34

```
cat("Variance Ratio:", round(variance_ratio, 3), "\n")
```

Variance Ratio: 6.571

Si el ratio es cercano a **1**, significa que las varianzas son prácticamente idénticas. Si el ratio es **mayor a 2**, ya hay una diferencia considerable. Es un indicador de magnitud (tamaño del efecto), mientras que el test de Levene indica si esa diferencia es estadísticamente “confiable”.

```
# Visual proof: Histograms for comparison
ggplot2::ggplot(fib2, aes(x = dia1, fill = factor(sex))) +
  geom_histogram(bins = 15, alpha = 0.6, position = "identity")
  ↪ +
  facet_wrap(~factor(sex)) +
  theme_minimal() +
  labs(title = "Distribution of hygiene levels by gender",
       subtitle = "Visual comparison of variance",
       x = "Hygiene Score", y = "Frequency")
```



El **ratio de varianzas** es un indicador de magnitud. Mientras que el test de Levene indica si la diferencia es *estadísticamente significativa*, el ratio indica *cuántas veces más* grande es la variabilidad de un grupo respecto al otro.

Como norma general, si este ratio supera un valor de **2.0**, la heterocedasticidad se considera lo suficientemente marcada como para desaconsejar el uso de pruebas paramétricas clásicas.

3.5. Introducción a la inferencia estadística paramétrica

Una vez descritos los datos y validada su distribución, el siguiente nivel es responder a las preguntas estratégicas de cualquier investigación. La **inferencia estadística** es el conjunto de técnicas que permite realizar afirmaciones sobre una población a partir de los datos observados en una muestra.

En esta sesión se abordan cuatro tipos fundamentales de contrastes que todo investigador debe dominar:

1. **Test t para una muestra:** se utiliza para comparar un resultado actual frente a un valor de referencia, un objetivo histórico o un estándar poblacional conocido.
2. **Test t para muestras independientes:** es la herramienta para comparar grupos distintos (ej. ¿se comportan igual los clientes de la Región A frente a los de la Región B?).
3. **Test t para muestras dependientes (o relacionadas):** permite medir la eficacia de una acción comparando el mismo grupo en dos momentos temporales diferentes (ej. antes y después de una campaña publicitaria).
4. **Correlación de Pearson:** es el primer paso para detectar asociaciones; permite saber si, cuando una variable aumenta, la otra también lo hace (o decrece), cuantificando la fuerza de esa relación.

El enfoque de esta sesión: no se trata de “clicar” opciones, sino de identificar **qué pregunta estadística corresponde a cada problema planteado**. Una vez identificada la técnica, se ejecuta el modelo en R, se interpreta su significación y, sobre todo, se evalúa si los resultados obtenidos son robustos.

Se pasa así de la observación simple a la **toma de decisiones basada en evidencia estadística**.

```
# Load the dataset (expss::read_spss preserves variable/value  
  ↳ labels from SPSS)  
gssft1 <- expss::read_spss('data/gssft1.sav')
```

3.6. Inferencia paramétrica

3.6.1. Test t de Student para una muestra

t-test para una muestra (media)



- **Objetivo**
 - Deseamos saber si nuestra muestra procede de una población cuya media es conocida.
 - **Alternativamente**
 - Queremos saber si una determinada variable de nuestra muestra difiere o no significativamente de un valor concreto que tiene significado relevante.
 - **Advertencia**
 - Tenemos una sola muestra (cajas de cereales, niños prematuros).
 - Tenemos un valor de referencia respecto al que comparar.
 - **Procedimiento:**
 - Mirar siempre los estadísticos descriptivos
 - Hipótesis nula:
 - **la media es igual al valor de referencia**
 - **Ejemplos reales ...**
 - Control de calidad: la muestra de cajas de cereales seleccionadas de la cadena de producción, ¿pesan realmente 250 gramos?
 - ¿Tienen los niños nacidos prematuramente un coeficiente intelectual promedio estándar de 100?
-
-
- **Nuestro ejemplo (sobre gssft1.sav):**
 - ¿Trabajan los licenciados universitarios una media de 40 horas semanales?
 - Variable:
 - hrs1: número de horas trabajadas la semana pasada

25

El test t para una muestra es la primera herramienta de inferencia. Se utiliza cuando es necesario saber si la media de una muestra es estadísticamente diferente de un **valor de referencia** (o valor poblacional).

Antes de ejecutar el test, la hoja de ruta es:

1. **Descriptivos:** conviene visualizar siempre la media y el histograma. Si los datos están muy sesgados o tienen *outliers* extremos, el test t pierde potencia.
2. **Planteamiento de Hipótesis:**
 - **Hipótesis Nula (H_0):** la media de la población es igual al valor de referencia ($\mu = \mu_0$).

- **Hipótesis Alternativa (H_1):** la media es diferente ($\mu \neq \mu_0$).

En R, la función `stats::t.test()` es extremadamente flexible. A continuación se definen todos sus parámetros para que el código sea transparente y no quede ninguna decisión estadística “oculta”:

```
# Defining the target value (reference mean)
target_mean <- 40

# Executing the one-sample t-test
stats::t.test(
  gssft1$hrs1,          # The variable to analyze
  mu                    = target_mean,  # The reference value
  alternative = "two.sided",  # Two-tailed: looking for any
    ↪ difference
  conf.level  = 0.95      # Standard confidence level
    ↪ (95 %)
)
```

One Sample t-test

```
data:  gssft1$hrs1
t = 14.069, df = 438, p-value < 2.2e-16
alternative hypothesis: true mean is not equal to 40
95 percent confidence interval:
 46.22201 48.24268
sample estimates:
mean of x
 47.23235
```

Interpretación para el investigador: al ejecutar el código, R devuelve tres datos críticos:

- **t-statistic:** indica cuántas desviaciones estándar se aleja la media muestral del valor objetivo.
- **p-value:** si es menor que 0.05, se rechaza la H_0 . Se concluye que la diferencia observada es estadísticamente significativa y no fruto del azar.

- **Confidence Interval (95 % CI):** indica el rango donde, con un 95 % de probabilidad, se encuentra la verdadera media poblacional. Si el valor de referencia (40) no está dentro de este rango, hay una prueba extra de que la media es significativamente distinta.

3.6.2. Test t de Student para muestras independientes

t-test para 2 muestras independientes

[?] %
SIS

- **Objetivo**
 - Tenemos una variable que genera dos grupos formados por distintos individuos.
 - Queremos saber si pertenecer a uno u otro grupo, genera respuestas distintas en la variable experimental
- **Requisitos**
 - Variable métrica
 - Independencia de los grupos (de facto)
 - Datos normales (validar)
 - Homoscedasticidad
- **Procedimiento:**
 - Mirar siempre los estadísticos descriptivos
 - Comprobar condiciones (normalidad -previo-, homoscedasticidad -en el propio test-)
 - Hipótesis nula
 - **la media es igual en los dos grupos**
- **Ejemplos:**
 - Actitud respecto a que se suban los impuestos sobre el tabaco entre fumadores y no fumadores
 - Nivel de conocimiento de una asignatura entre los grupos de dos profesores distintos



- **Ejemplo (gssnet2.sav).**
 - ¿Es el uso de internet el mismo entre hombres y mujeres?
 - Lo analizaremos con dos variables, el número de horas dedicadas a leer y contestar el correo electrónico (emailhrs) y a navegar por la web (webhrs)

26

Este test permite determinar si existe una diferencia estadísticamente significativa entre las medias de dos grupos distintos (ej. ¿consumen más internet los hombres que las mujeres?).

3.6.2.1. El protocolo de decisión

Para aplicar correctamente este test se siguen tres pasos obligatorios:

1. **Diagnóstico de varianzas (Levene):** se comprueba si los grupos tienen dispersiones similares. Si el *p-value* es > 0.05 , se asumen **varianzas iguales** (`var.equal = TRUE`). Si es ≤ 0.05 , se utiliza la **Corrección de Welch** (`var.equal = FALSE`).
2. **Ejecución del test:** se utiliza la sintaxis de fórmulas `variable_metrica ~ variable_factor`.

3. **Tamaño del efecto:** no basta con saber si hay diferencia; se calcula el r de Cohen (o *eta-cuadrado* mediante la fórmula de la t y los grados de libertad) para ver si esa diferencia es relevante en el mundo real.

3.6.2.2. Diagnóstico de homogeneidad de varianzas (Test de Levene)

Antes de comparar medias, se comprueba si la variabilidad es similar en ambos grupos.

```
# Loadin data
gssnet2 <- expss::read_spss("data/gssnet2.sav")

# Convert sex to factor with descriptive labels
gssnet2$sex_factor <- factor(gssnet2$sex, labels = c("Hombre",
  ↪ "Mujer"))

# Levene's test for homogeneity of variances
car::leveneTest(emailhrs ~ sex_factor, data = gssnet2)
```

Levene's Test for Homogeneity of Variance (center = median)

	Df	F value	Pr(>F)
group	1	0.0988	0.7534
	982		

3.6.2.3. Opción A: Varianzas iguales (Levene $p > 0.05$)

Se utiliza `var.equal = TRUE`.

```
# t-test assuming equal variances
test_t_equal <- stats::t.test(emailhrs ~ sex_factor, data =
  ↪ gssnet2,
                                var.equal = TRUE,
                                conf.level = 0.95)

print(test_t_equal)
```

Two Sample t-test

data: emailhrs by sex_factor

t = 0.20832, df = 982, p-value = 0.835

alternative hypothesis: true difference in means between group Hombre and group Mujer

95 percent confidence interval:

-0.9158859 1.1334347

sample estimates:

mean in group Hombre mean in group Mujer

3.371057

3.262282

```
# Effect size: Cohen's r
t_val <- test_t_equal$statistic
df_val <- test_t_equal$parameter
cat("Effect size (r):", round(sqrt((t_val^2) / ((t_val^2) +
↪ df_val))), 3), "\n")
```

Effect size (r): 0.007

3.6.2.4. Opción B: Varianzas diferentes (Levene $p \leq 0.05$)

Se utiliza var.equal = FALSE (Corrección de Welch).

```
# t-test with Welch correction (unequal variances)
test_t_welch <- stats::t.test(emailhrs ~ sex_factor, data =
↪ gssnet2,
                                var.equal = FALSE,
                                conf.level = 0.95)
print(test_t_welch)
```

Welch Two Sample t-test

data: emailhrs by sex_factor

t = 0.20682, df = 893.88, p-value = 0.8362

```
alternative hypothesis: true difference in means between group Hombre and group Mujer
95 percent confidence interval:
-0.9234377  1.1409865
sample estimates:
mean in group Hombre  mean in group Mujer
          3.371057          3.262282
```

```
# Effect size: Cohen's r
t_val <- test_t_welch$statistic
df_val <- test_t_welch$parameter
cat("Effect size (r):", round(sqrt((t_val^2) / ((t_val^2) +
↪ df_val))), 3), "\n")
```

Effect size (r): 0.007

3.6.3. Test de comparación de proporciones (Test Z)

[C%]
ΣiΣ

Test Z para proporciones

Cuando se trabaja con variables categóricas (como el grupo de pertenencia A, B o C), ya no se comparan medias, sino **proporciones**. El Test Z permite determinar si la diferencia entre las proporciones observadas en distintos grupos es estadísticamente significativa.

3.6.3.1. Preparación de los datos

Se utiliza el archivo `gssz`, que incluye grupos de pertenencia (`grupo`), sexo (`sexo`) y edad (`edad`).

```
# Load dataset
gssz <- expss::read_spss('data/gssz.sav')

# Contingency table: sex vs group membership
contingency_sex_group <- table(gssz$sexo, gssz$grupo)
print(contingency_sex_group)
```

	A	B	C
Hombre	217	54	71
Mujer	90	146	47

```
# Row percentages
round(prop.table(contingency_sex_group, 1) * 100, 2)
```

	A	B	C
Hombre	63.45	15.79	20.76
Mujer	31.80	51.59	16.61

3.6.3.2. Ejecución del Test Z

Se quiere verificar si la proporción de pertenencia al **Grupo A** es distinta entre **Hombres** y **Mujeres**.

```
# Successes (Group A counts) and totals by sex
group_a_counts <- c(contingency_sex_group["Hombre", "A"],
                    contingency_sex_group["Mujer", "A"])

sex_totals <- c(sum(contingency_sex_group["Hombre", ]),
               sum(contingency_sex_group["Mujer", ]))
```

```
# Proportion test (Z-test for two proportions)
stats::prop.test(x = group_a_counts, n = sex_totals, correct =
  ↪ FALSE)
```

2-sample test for equality of proportions without continuity correction

```
data:  group_a_counts out of sex_totals
X-squared = 62.062, df = 1, p-value = 3.328e-15
alternative hypothesis: two.sided
95 percent confidence interval:
 0.2419910 0.3909725
sample estimates:
   prop 1    prop 2 
0.6345029 0.3180212
```

Interpretación para el investigador:

- **Hipótesis nula (H_0):** la proporción de pertenencia al Grupo A es igual para hombres que para mujeres.
- **p-value:** si el valor es < 0.05 , se rechaza la igualdad.
- **Confidence interval:** si el intervalo no incluye el valor 0, la diferencia entre las proporciones es estadísticamente significativa.

3.6.4. Test t de Student para muestras dependientes (pareadas)



t-test para 2 muestras relacionadas

- **Objetivo**
 - Tenemos dos muestras formadas por los mismos individuos
 - Queremos saber si la variable manipulada genera diferencias en la variable observada antes y después del experimento
- **Procedimiento**
 - Mirar siempre los estadísticos descriptivos
 - Paired (dependent) sample t-test
- **Hipótesis nula:** media igual antes y después del experimento
- **Requisitos**
 - Datos normales (normalidad estadística)
 - Variable métrica
- **Ejemplos:**
 - Actitud hacia el tabaco de fumadores antes y después de un documental sobre sus efectos
 - Conocimiento de una asignatura antes y después de un curso de repaso



- **Ejemplo (endorph1.sav):**
 - Dale *et al.* (1987) postulan que los corredores de una maratón son capaces de continuar corriendo pese al cansancio y el dolor porque sus niveles de β -endorfina crecen y producen una sensación de bienestar.
 - Midieron los niveles de esta variable antes y después de una media maratón
 - (Variables: after, before), a un total de 11 corredores

28

Este test se aplica cuando se mide la **misma variable sobre los mismos sujetos en dos momentos o condiciones diferentes**. La pregunta de investigación es: ¿ha cambiado significativamente la medida tras la intervención?

El ejemplo proviene del dataset `endorph1`, que recoge los niveles de endorfinas de un grupo de sujetos **antes** y **después** de una sesión de ejercicio físico. Cada fila es el mismo individuo en dos momentos distintos — de ahí el término *pareadas*.

3.6.4.1. El protocolo de decisión

Se siguen tres pasos:

1. **Descriptivos de la diferencia:** se calcula la media, desviación típica e intervalo de confianza de la diferencia **before** - **after**. Si la diferencia media es cercana a cero, hay pocas razones para esperar significación.
2. **Diagnóstico de normalidad:** lo que debe ser normal es la **distribución de las diferencias** (aunque en la práctica se comprueba cada condición por separado), mediante Lilliefors y Shapiro-Wilk.
3. **Ejecución del test y tamaño del efecto:** Con `paired = TRUE`, R calcula internamente la diferencia y aplica el test t sobre ella.

```
# Loading data
endorph1 <- expss::read_spss("data/endorph1.sav")

# --- Step 1: Descriptive statistics of the difference ---
before_scores <- endorph1$before
after_scores <- endorph1$after

diff_mean <- mean(before_scores - after_scores, na.rm = TRUE)
diff_sd <- sd(before_scores - after_scores, na.rm = TRUE)
diff_se <- diff_sd / sqrt(length(before_scores)) # Standard
↪ error of the mean difference
ci_lower <- diff_mean - (2 * diff_se)
ci_upper <- diff_mean + (2 * diff_se)

cat("Mean difference:", round(diff_mean, 3), "\n")
```

Mean difference: -18.736

```
cat("Std. deviation: ", round(diff_sd, 3), "\n")
```

Std. deviation: 8.33

```
cat("95% CI: [", round(ci_lower, 3), ",", round(ci_upper, 3),
↪ "]" "\n")
```

95% CI: [-23.759 , -13.713]

```
# --- Step 2: Normality tests for each condition ---
# Lilliefors (Kolmogorov-Smirnov with Lilliefors correction)
print(nortest::lillie.test(before_scores))
```

Lilliefors (Kolmogorov-Smirnov) normality test

data: before_scores

D = 0.17365, p-value = 0.4668

```
print(nortest::lillie.test(after_scores))
```

Lilliefors (Kolmogorov-Smirnov) normality test

data: after_scores

D = 0.12988, p-value = 0.87

```
# Shapiro-Wilk (recommended for n < 50)
print(stats::shapiro.test(before_scores))
```

Shapiro-Wilk normality test

data: before_scores

W = 0.86876, p-value = 0.07472

```
print(stats::shapiro.test(after_scores))
```

Shapiro-Wilk normality test

data: after_scores

W = 0.96441, p-value = 0.8252

```
# --- Step 3: Paired t-test ---
paired_ttest <- stats::t.test(
  before_scores,
  after_scores,
  alternative = "two.sided", # Two-tailed: we test for any
  ↪ difference
  paired      = TRUE,       # KEY: same subjects measured
  ↪ twice
  conf.level  = 0.95
```

```
)

print(paired_ttest)
```

Paired t-test

```
data: before_scores and after_scores
t = -7.4602, df = 10, p-value = 2.159e-05
alternative hypothesis: true mean difference is not equal to 0
95 percent confidence interval:
 -24.33236 -13.14037
sample estimates:
mean difference
 -18.73636
```

```
# --- Step 4: Effect size (Cohen's r) ---
t_stat      <- paired_ttest[["statistic"]][["t"]]
df_paired    <- paired_ttest[["parameter"]][["df"]]
effect_size <- sqrt((t_stat^2) / ((t_stat^2) + df_paired))

cat("Effect size (r):", round(effect_size, 3), "\n")
```

Effect size (r): 0.921

```
cat("Interpretation: small >= 0.10 | medium >= 0.30 | large >=
↪ 0.50\n")
```

Interpretation: small >= 0.10 | medium >= 0.30 | large >= 0.50

Interpretación para el investigador: conviene fijarse en tres elementos del output:

- **p-value:** si es < 0.05 , la diferencia antes/después es estadísticamente significativa — el ejercicio ha tenido un efecto real sobre los niveles de endorfinas.

- **Intervalo de confianza (95 % CI):** si el intervalo **no contiene el cero**, la diferencia es significativa. El cero representaría “ningún cambio”.
- **Tamaño del efecto (r):** indica si esa diferencia es relevante en términos prácticos, más allá de la significación estadística. Un p-value significativo con un efecto pequeño puede ser poco útil en la toma de decisiones.



El efecto ...

- Introduciendo el tamaño del efecto (effect size)
 - Que el estadístico sea significativo, no quiere decir que el tamaño del efecto que mide sea importante
 - El tamaño del efecto es una medida objetiva y estandarizada de la magnitud del efecto observado.
 - Implica el grado en que son diferentes las mediciones o puntajes entre los pares.
- Cohen (1988, 1992) ha planteado la siguiente clasificación (bastante aceptada) de cómo clasificar el tamaño del efecto (basado en el coeficiente de correlación de Pearson).
 - $r \geq .10$ (efecto pequeño) (1% de la varianza total)
 - $r \geq .30$ (efecto medio) (9% de la varianza total)
 - $r \geq .50$ (efecto grande) (25% de la varianza total)
- Una forma de obtenerlo la proporcionan Rosenthal (1991; p.19) o Rosnow y Rosenthal (2005; p.328)

29

3.6.5. Correlación paramétrica de Pearson



Medidas de asociación: correlación

- Coeficiente de correlación de Pearson (anxiety.sav)
 - Es, simplemente, una estandarización de la covarianza, para eliminar el efecto de las unidades de medidas diferentes en las dos variables y, así, queda acotado entre -1 y 1.
- Ejemplo.
 - Veamos cómo se relacionan las notas de un conjunto de estudiantes en un examen (nota 0-100), con el número de horas de estudio y con el nivel de ansiedad previo al examen (medido con un test psicológico que puntúa 0-100).
 - Covarianza: grado en que cuando una variable se desvía de su media, la otra variable lo hace de manera similar.
- Procedimiento:
 - Gráfico para evaluar la linealidad de la relación.
 - Coeficiente de Pearson
 - (Hipótesis nula: $r=0$, ausencia de relación)

30

La correlación de Pearson mide la **fuerza y dirección de la relación lineal** entre

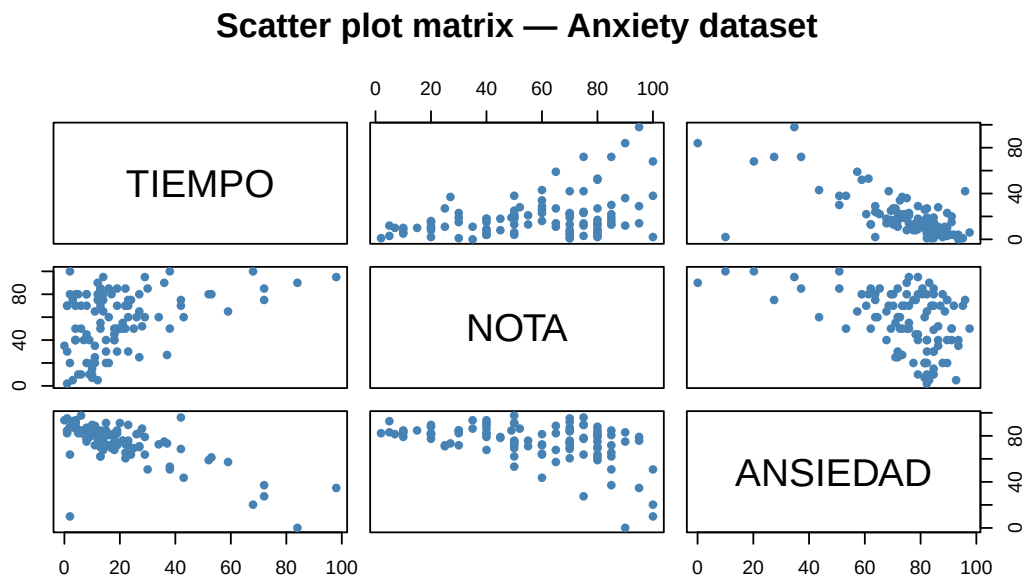
dos variables métricas. Antes de interpretar el coeficiente es fundamental verificar que la relación sea efectivamente lineal, de ahí que el análisis se acompañe siempre de una representación gráfica.

El ejemplo utiliza el dataset `anxiety`, que recoge tres variables de un grupo de estudiantes: tiempo de estudio (TIEMPO), nota obtenida (NOTA) y nivel de ansiedad (ANSIEDAD).

```
# Loading data
anxiety <- expss::read_spss("data/anxiety.sav")

# --- Step 1: Select variables of interest ---
anxiety_data <- dplyr::select(anxiety, TIEMPO, NOTA, ANSIEDAD)

# --- Step 2: Scatter plot matrix to assess linearity visually
↪ ---
plot(anxiety_data,
     main = "Scatter plot matrix - Anxiety dataset",
     col = "steelblue",
     pch = 16)
```



```
# --- Step 3: Bivariate Pearson correlation (TIEMPO vs NOTA) ---
pearson_tiempo_nota <- stats::cor.test(
  anxiety$TIEMPO,
  anxiety$NOTA,
  alternative = "two.sided",
  method      = "pearson",
  conf.level   = 0.95
)

print(pearson_tiempo_nota)
```

Pearson's product-moment correlation

```
data: anxiety$TIEMPO and anxiety$NOTA
t = 4.3434, df = 101, p-value = 3.343e-05
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.2200938 0.5481602
sample estimates:
      cor
0.3967207
```

```
# --- Step 4: Full correlation matrix with p-values ---
pearson_matrix <- Hmisc::rcorr(as.matrix(anxiety_data), type =
  ↪ "pearson")

cat("--- Correlation coefficients (r) ---\n")
```

--- Correlation coefficients (r) ---

```
print(round(pearson_matrix[["r"]], 3))
```

	TIEMPO	NOTA	ANSIEDAD
TIEMPO	1.000	0.397	-0.709

NOTA	0.397	1.000	-0.441
ANSIEDAD	-0.709	-0.441	1.000

```
cat("\n--- Sample sizes (n) ---\n")
```

```
--- Sample sizes (n) ---
```

```
print(pearson_matrix[["n"]])
```

	TIEMPO	NOTA	ANSIEDAD
TIEMPO	103	103	103
NOTA	103	103	103
ANSIEDAD	103	103	103

```
cat("\n--- p-values ---\n")
```

```
--- p-values ---
```

```
print(round(pearson_matrix[["P"]], 4))
```

	TIEMPO	NOTA	ANSIEDAD
TIEMPO	NA	0	0
NOTA	0	NA	0
ANSIEDAD	0	0	NA

Interpretación para el investigador: la función `Hmisc::rcorr()` devuelve tres matrices que deben leerse de forma conjunta:

- **r:** el coeficiente de correlación. Valores cercanos a ± 1 indican relación fuerte; cercanos a 0, relación débil o inexistente.
- **n:** el número de pares válidos usados en cada correlación (importante cuando hay valores perdidos).
- **P:** el p-value de cada coeficiente. Si es < 0.05 , la correlación es estadísticamente significativa. La diagonal siempre será NA (correlación de una variable consigo misma).

3.6.6. Análisis de varianza de un factor (ANOVA one-way)



Análisis de varianza

ANOVA, del inglés Analysis of Variance

- Extensión natural del test t cuando se necesita comparar más de dos grupos simultáneamente.
- ¿Qué pregunta responde?
 - ¿Existen diferencias significativas entre las medias de tres o más grupos independientes, o las diferencias observadas pueden atribuirse al azar?
- ¿Por qué no repetir el test t varias veces?
 - Comparar los grupos de dos en dos infla el error de tipo I: si se realizan tres comparaciones al 95 % de confianza, la probabilidad acumulada de obtener al menos un falso positivo supera ya el 14 %. ANOVA resuelve este problema con una única prueba global.

Lógica del contraste

- ANOVA descompone la variabilidad total de los datos en dos fuentes:
 - Variabilidad entre grupos: debida al efecto de la variable de agrupación (el factor).
 - Variabilidad dentro de los grupos: debida al azar o a diferencias individuales no controladas.
 - Si la variabilidad entre grupos es mucho mayor que la variabilidad interna, hay evidencia de que al menos un grupo difiere del resto.
- Supuestos necesarios
 - Normalidad de los residuos en cada grupo.
 - Homogeneidad de varianzas entre grupos (homocedasticidad).
 - Independencia de las observaciones.

El ANOVA de un factor es la extensión natural del test t cuando se quiere comparar **más de dos grupos**. La pregunta de investigación es: ¿difieren significativamente las medias de la variable métrica entre los grupos definidos por el factor?

Se utiliza el dataset **hatco**. La variable dependiente es **X10** (valoración del servicio) y el factor de agrupación es **X14** (tipo de compra: nueva, recompra modificada, recompra directa).

Nota importante: en R, la variable de agrupación **debe ser de tipo factor**. Si se pasa como numérica, `stats::aov()` no funcionará correctamente. Conviene convertirla siempre antes de ejecutar el modelo.

```
#Loadind data
hatco <- expss::read_spss("data/hatco.sav")

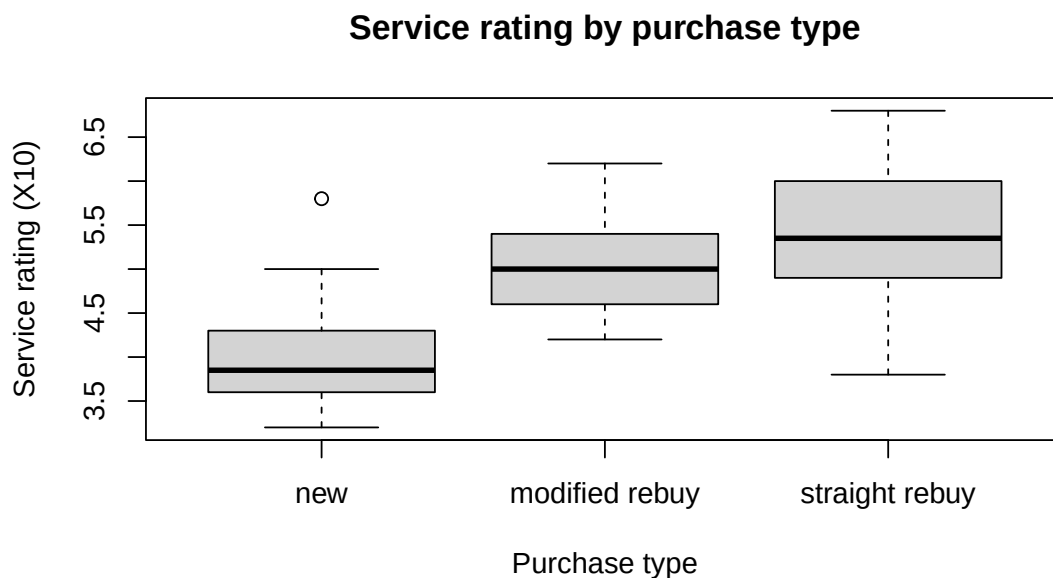
# --- Step 1: Convert grouping variables to factor with
#   ↳ descriptive labels ---
hatco <- hatco %>%
  dplyr::mutate(
    purchase_type = factor(X14, labels = c("new", "modified",
    ↳ rebuy", "straight rebuy")),
```

```

    firm_type      = factor(X13, labels = c("type 1", "type
    ↪ 2"))
  )

# --- Step 2: Boxplot to visualize group differences ---
boxplot(hatco$X10 ~ hatco$purchase_type,
        col = "lightgray",
        main = "Service rating by purchase type",
        xlab = "Purchase type",
        ylab = "Service rating (X10)")

```



```

# --- Step 3: One-way ANOVA ---
anova_one_way <- stats::aov(X10 ~ purchase_type, data = hatco)

# Summary table: F-statistic and p-value
summary(anova_one_way)

```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
purchase_type	2	39.01	19.503	56.54	<2e-16 ***
Residuals	97	33.46	0.345		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
# Group means
stats::model.tables(anova_one_way, "means")
```

Tables of means

Grand mean

4.771

	purchase_type		
	new	modified	rebuy
	3.929	5.003	5.394
rep	34.000	32.000	34.000

Interpretación para el investigador: el output de `summary()` ofrece la tabla ANOVA clásica. El dato clave es el **p-value de la F** (columna `Pr(>F)`):

- Si $p < 0.05$: al menos un par de grupos tiene medias significativamente distintas.
- Si $p \geq 0.05$: no hay evidencia de diferencias entre grupos.

Cuando el ANOVA es significativo, el siguiente paso sería un **test post-hoc** (Tukey, Bonferroni) para identificar *qué pares de grupos* difieren. Esto se abordará en sesiones posteriores.

3.6.7. Análisis de varianza de dos factores (ANOVA two-way)

El ANOVA de dos factores introduce un segundo factor de agrupación, lo que permite analizar no solo el efecto de cada factor por separado, sino también su posible **interacción**: ¿el efecto del tipo de compra sobre la valoración del servicio es diferente según el tipo de empresa?

```
# --- Two-way ANOVA: main effects of purchase_type and firm_type
↪ ---
anova_two_way <- stats::aov(X10 ~ purchase_type + firm_type,
↪ data = hatco)
```

```
# Summary table
summary(anova_two_way)
```

```

              Df Sum Sq Mean Sq F value Pr(>F)
purchase_type  2  39.01  19.503   57.424 <2e-16 ***
firm_type      1   0.85   0.854    2.514  0.116
Residuals     96  32.61   0.340
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
# Group means for each factor
stats::model.tables(anova_two_way, "means")
```

Tables of means
Grand mean

4.771

```

purchase_type
      new modified rebuy straight rebuy
      3.929          5.003          5.394
rep 34.000          32.000          34.000

```

```

firm_type
      type 1 type 2
      4.863 4.679
rep 50.000 50.000

```

Interpretación para el investigador: `summary()` muestra ahora una fila por cada factor. Cada p-value se interpreta de forma independiente: un factor puede ser significativo aunque el otro no lo sea. Para testar la **interacción** entre ambos factores se utilizaría la sintaxis `purchase_type * firm_type` en lugar de `purchase_type + firm_type`.

3.7. Inferencia no paramétrica

Cuando los datos no cumplen el supuesto de normalidad — habitual en escalas de valoración, variables ordinales o muestras pequeñas — se recurre a los **contrastes no paramétricos**. Estos tests trabajan sobre rangos en lugar de sobre los valores originales, lo que los hace robustos frente a distribuciones asimétricas o con valores extremos.

3.7.1. Prueba Chi-cuadrado (χ^2)

3.7.1.1. Chi-cuadrado de bondad de ajuste (una muestra)

Este test compara la distribución observada en una muestra con una distribución teórica de referencia. La pregunta es: ¿se distribuyen los datos de forma similar a lo esperado?

```
# --- Observed frequencies vs. expected (theoretical)
↪ frequencies ---
observed_freq <- c(28, 47, 80, 82, 47, 35)
expected_freq <- c(30, 50, 75, 75, 50, 30)

# rescale.p = TRUE: R normalizes expected_freq to proportions
↪ automatically
chi2_goodness <- stats::chisq.test(observed_freq, p =
  ↪ expected_freq, rescale.p = TRUE)
print(chi2_goodness)
```

Chi-squared test for given probabilities

data: observed_freq

X-squared = 1.9941, df = 5, p-value = 0.85

3.7.1.2. Chi-cuadrado de independencia (tabla de contingencia)



Tablas de contingencia

Tablas resumen de frecuencias:

- Marginales
- Cruzadas
- Observado / Esperado
- Residuos
- Residuos ajustados (normalizados)

Prueba de diferentes formas de tabulación en los datos (hatco.sav).

- ¿Cuántas empresas grandes tienen su departamento de compras centralizado? 40
- ¿Qué porcentaje de las empresas con departamento de compra centralizado son grandes? 80%
- ¿Qué porcentaje de las empresas grandes tiene departamento de compras centralizado? 100%
- ¿Qué porcentaje de empresas tiene departamento de compras centralizado y son empresas grandes? 40%

Normalmente en un estudio de mercado, se hacen tablas cruzadas de todos los datos de profiling (clasificación) con todas las preguntas del cuestionario (u otros campos presentes en la base de datos) en la encuesta.

En estas tablas, el cliente suele exigir χ^2 para las tablas que cruzan con nominal, **prueba Z de proporciones** cuando lo que quieren es comparar los porcentajes de las categorías entre los grupos o **t-test** (medias) para las tablas que reportan la media.

32

Este es el uso más frecuente en investigación de mercados: determinar si dos variables categóricas están **asociadas** o son independientes entre sí.



Tablas de contingencia y test χ^2

- Objetivo:
 - Las tablas de contingencia evalúan la dependencia de una variable no métrica respecto a otra variable no métrica
- Procedimiento:
 - Se basa en frecuencias, luego los datos no han de cumplir requisito alguno de normalidad, pero cada individuo solo puede estar en una celda y no debería haber en ninguna celda menos de **5 casos esperados** (pérdida de potencia de la prueba)
 - Hay que interpretar primero la dirección de la influencia potencial mediante los correspondientes porcentajes
 - Chi-square test
- Hipótesis nula:
 - las dos variables son independientes
- Ejemplo (manners1.sav):
 - Hay personas que dan prioridad a las buenas maneras y otras a que la gente se exprese libremente, aunque lo haga de manera poco educada.
 - ¿Son las mujeres más proclives que los hombres a dar más importancia a las buenas maneras que a expresarse libremente o ambas variables son independientes?
- Variables:
 - sexo, (hombre=1, mujer=2)
 - freedman (1=prefiero dar a la gente libertad de expresión, tolerando las malas formas; 2=prefiero dar prioridad a las buenas formas, aún restringiendo la libertad de expresión)

33

```
# Loading data
data2 <- expss::read_spss("data/manners.sav")

# --- Step 1: Build the contingency table ---
```

```
contingency_table <- table(data2$freedman, data2$sex)
print(contingency_table)
```

	Male	Female
Don't know	28	36
No Answer	1	10
Allowing people freedom of expression/tolerate bad manners	268	232
Enforcing good manners/limit freedom of expression	195	240

```
# --- Step 2: Chi-square test ---
# Without Yates correction (recommended for tables larger than
  ↪ 2x2)
chi2_no_yates <- stats::chisq.test(contingency_table, correct =
  ↪ FALSE)
print(chi2_no_yates)
```

Pearson's Chi-squared test

```
data: contingency_table
X-squared = 14.951, df = 3, p-value = 0.001859
```

```
# With Yates continuity correction (applies automatically to 2x2
  ↪ tables)
chi2_yates <- stats::chisq.test(contingency_table, correct =
  ↪ TRUE)
print(chi2_yates)
```

Pearson's Chi-squared test

```
data: contingency_table
X-squared = 14.951, df = 3, p-value = 0.001859
```

3.7.1.3. Medidas de asociación derivadas

Un Chi-cuadrado significativo indica que hay asociación, pero no indica **cuánta**. Para eso se calculan medidas de efecto como la V de Cramér o el coeficiente de contingencia.

```
# --- Association measures: Cramer's V, contingency coefficient,  
↪ etc. ---  
association_stats <- vcd::assocstats(contingency_table)  
print(association_stats)
```

	X ²	df	P(> X ²)
Likelihood Ratio	16.138	3	0.0010623
Pearson	14.951	3	0.0018586

Phi-Coefficient : NA
Contingency Coeff.: 0.121
Cramer's V : 0.122

Interpretación para el investigador: `vcd::assocstats()` devuelve varias medidas. La más utilizada en investigación de mercados es la **V de Cramér**: valores próximos a 0 indican asociación débil; próximos a 1, asociación fuerte. Para tablas 2×2, el **coeficiente phi** es equivalente.

3.7.2. Correlaciones no paramétricas



Medidas de asociación: correlación

- Coeficiente de correlación de Spearman (grades.sav)
 - No paramétrico
 - Se utiliza cuando ambas variables son ordinales
 - Acotado +1, -1
 - Funciona primero generando un ranking de los datos y luego calculando un coeficiente de Pearson entre esos ranking
- Coeficiente de correlación Tau-b de Kendall (grades.sav)
 - Mismo funcionamiento que el de Spearman, pero es más adecuado si hay muchos rankings empatados
- Ejemplo
 - ¿Están relacionadas las notas de matemáticas de los estudiantes en el selectivo con la de la asignatura de estadística de primero de carrera?
 - En ambos casos las notas siguen el sistema británico: A, B, C, D, E, F.
 - Son ordinales porque, por ejemplo, el rango de notas para B es 60%-70%, mientras que para A es 70%-100%

36

Cuando las variables no son métricas o no cumplen el supuesto de normalidad, la correlación de Pearson se sustituye por alternativas basadas en rangos.

El dataset **grades** contiene las notas de estadística (**Estadística**) y la puntuación de selectividad (**Selectivo**) de un grupo de alumnos.

3.7.2.1. Correlación de Spearman (ρ)

Mide la correlación entre los **rangos** de las dos variables. Es la alternativa más común a Pearson cuando los datos son ordinales o presentan distribución no normal.

```
# Loading data
grades <- expss::read_spss("data/grades.sav")

# --- Spearman rank correlation ---
spearman_result <- stats::cor.test(
  grades$Estadística,
  grades$Selectivo,
  alternative = "two.sided",
  method      = "spearman",
  exact       = NULL,
```

```

    conf.level = 0.95
)

print(spearman_result)

```

Spearman's rank correlation rho

```

data:  grades$Estadística and grades$Selectivo
S = 1418, p-value = 0.02243
alternative hypothesis: true rho is not equal to 0
sample estimates:
      rho
0.4546021

```

3.7.2.2. Correlación de Kendall (τ)

Alternativa más conservadora a Spearman, especialmente recomendada con muestras pequeñas o cuando hay muchos empates en los rangos.

```

# --- Kendall's tau-b correlation ---
kendall_result <- stats::cor.test(
  grades$Estadística,
  grades$Selectivo,
  alternative = "two.sided",
  method      = "kendall",
  exact       = NULL,
  conf.level  = 0.95
)

print(kendall_result)

```

Kendall's rank correlation tau

```

data:  grades$Estadística and grades$Selectivo

```

```

z = 2.1794, p-value = 0.0293
alternative hypothesis: true tau is not equal to 0
sample estimates:
      tau
0.3539614

```

Interpretación para el investigador: tanto ρ de Spearman como τ de Kendall se interpretan igual que el r de Pearson en cuanto a signo y magnitud, pero sus valores no son directamente comparables entre sí. En general, τ suele ser numéricamente más pequeño que ρ para los mismos datos — eso es normal y esperado.

3.7.3. Prueba de Wilcoxon — Muestras independientes (U de Mann-Whitney)

U de Mann-Whitney y W de Wilcoxon [equivalente a t-test para muestras independientes]



Descripción del test	Algunos comentarios
- Situación - Mismo uso que los t-test para medias en muestras independientes, es decir, queremos saber si una variable toma valores (rangos en este caso) distintos en dos subgrupos. - Sin embargo, la variable dependiente no sigue una distribución normal. - La hipótesis nula es que no hay diferencias de rangos. - Test disponibles - Wilcoxon suma de rangos (*) (Wilcoxon, 1945) - Mann-Whitney (*) (Mann y Whitney, 1947) - Otras pruebas disponibles - Kolmogorov-Smirnov Z - Moses Reacciones Extremas (extreme reactions)	- Para sustituir al test t (comparar medianas/localización): U de Mann-Whitney / W de Wilcoxon (misma prueba, aunque con resultado de estadístico diferente que no de resultado de inferencia). Es la alternativa directa. - Para saber si las distribuciones difieren en CUALQUIER aspecto (forma, dispersión, etc.): Usa la prueba de Kolmogorov-Smirnov para dos muestras. Es más general, pero a menudo menos potente para detectar sólo diferencias de medianas. - Para saber si un grupo tiene más valores extremos que otro (comparar dispersión): Usa la prueba de Moses de Reacciones Extremas. Es el análogo no paramétrico de un test de Levene (igualdad de varianzas), no de un test t.

34

Es el equivalente no paramétrico del test t para muestras independientes. Compara si las distribuciones de dos grupos independientes son similares, sin exigir que los datos sigan una distribución normal.

El dataset `bdi` recoge puntuaciones de depresión (BDI) de dos grupos de sujetos (droga: 1 = alcohol, 2 = éxtasis) medidos en dos días de la semana (`sunbdi` = domingo, `wedbdi` = miércoles).

3.7.3.1. Diagnóstico de normalidad (Lilliefors y Shapiro-Wilk)

Antes de decidir qué prueba aplicar, es imprescindible verificar si los datos cumplen el supuesto de normalidad. La regla de decisión es siempre la misma: la **hipótesis nula (H₀) afirma que la distribución es normal**. Un p-valor inferior a 0,05 lleva a rechazar H₀ y a concluir que los datos *no* se distribuyen con normalidad — lo que justifica el uso de una prueba no paramétrica.

Se combinan dos contrastes de normalidad para mayor robustez:

- **Lilliefors** (corrección de Kolmogórov-Smirnov): más potente cuando los parámetros de la distribución no son conocidos de antemano y se estiman a partir de la propia muestra.
- **Shapiro-Wilk**: especialmente recomendado para muestras pequeñas ($n < 50$). Es el contraste de referencia habitual en investigación aplicada.

Se ejecutan los ocho contrastes resultantes de cruzar los cuatro vectores (grupo × día) con los dos tests:

```
bdi <- expss::read_spss('data/bdi.sav')
# --- Step 1: Split data by group and measurement day ---
alcohol_sunday <- bdi %>% dplyr::filter(droga == 1) %>%
  ↪ dplyr::pull(sunbdi)
ecstasy_sunday <- bdi %>% dplyr::filter(droga == 2) %>%
  ↪ dplyr::pull(sunbdi)
alcohol_wednes <- bdi %>% dplyr::filter(droga == 1) %>%
  ↪ dplyr::pull(wedbdi)
ecstasy_wednes <- bdi %>% dplyr::filter(droga == 2) %>%
  ↪ dplyr::pull(wedbdi)

# --- Step 2: Normality tests ---
# Lilliefors
print(nortest::lillie.test(alcohol_sunday))
print(nortest::lillie.test(ecstasy_sunday))
print(nortest::lillie.test(alcohol_wednes))
print(nortest::lillie.test(ecstasy_wednes))

# Shapiro-Wilk
```

```
print(shapiro.test(alcobol_sunday))
print(shapiro.test(ecstasy_sunday))
print(shapiro.test(alcobol_wednes))
print(shapiro.test(ecstasy_wednes))
```

Lilliefors (Kolmogorov-Smirnov) normality test

data: alcobol_sunday
D = 0.27585, p-value = 0.02986

Lilliefors (Kolmogorov-Smirnov) normality test

data: ecstasy_sunday
D = 0.16992, p-value = 0.5687

Lilliefors (Kolmogorov-Smirnov) normality test

data: alcobol_wednes
D = 0.23469, p-value = 0.1226

Lilliefors (Kolmogorov-Smirnov) normality test

data: ecstasy_wednes
D = 0.30502, p-value = 0.00894

Shapiro-Wilk normality test

data: alcobol_sunday
W = 0.81064, p-value = 0.01952

Shapiro-Wilk normality test

```
data:  ecstasy_sunday  
W = 0.95947, p-value = 0.7798
```

Shapiro-Wilk normality test

```
data:  alcohol_wednes  
W = 0.94114, p-value = 0.5658
```

Shapiro-Wilk normality test

```
data:  ecstasy_wednes  
W = 0.75347, p-value = 0.003933
```

Lectura de los resultados. Cada test devuelve un estadístico (D en Lilliefors, W en Shapiro-Wilk) y su p-valor asociado. La atención debe centrarse en el p-valor: si es **mayor que 0,05**, no hay evidencia para rechazar la normalidad; si es **menor que 0,05**, la normalidad queda descartada para ese grupo y ese momento de medición.

En este caso, los resultados muestran p-valores por debajo de 0,05 en la mayoría de los subgrupos — en especial en el grupo de éxtasis. Cuando al menos uno de los grupos no cumple la normalidad, la comparación paramétrica (test t) deja de ser válida y debe sustituirse por su alternativa no paramétrica. Esa alternativa es la prueba de Wilcoxon para muestras independientes, también conocida como U de Mann-Whitney.

3.7.3.2. Prueba W de Wilcoxon — U de Mann-Whitney

Una vez diagnosticada la falta de normalidad, se aplica el contraste no paramétrico. La prueba de Wilcoxon para muestras independientes transforma las puntuaciones originales en rangos y compara la distribución de esos rangos entre los dos grupos. La **hipótesis nula** es que las distribuciones de ambos grupos son idénticas (o equivalentemente, que la diferencia de localización entre ellas es cero).

```

# --- Wilcoxon rank-sum test (Mann-Whitney U) ---
# Sunday: alcohol vs ecstasy
wilcox_sunday <- stats::wilcox.test(alcohol_sunday,
  ↪  ecstasy_sunday,
                                paired      = FALSE,
                                ↪  # Independent groups
                                alternative = "two.sided",
                                mu          = 0,
                                conf.int    = TRUE,
                                conf.level  = 0.95)

print(wilcox_sunday)

```

Wilcoxon rank sum exact test

```

data:  alcohol_sunday and ecstasy_sunday
W = 64.5, p-value = 0.2882
alternative hypothesis: true location shift is not equal to 0
95.9 percent confidence interval:
-1  5
sample estimates:
difference in location
1

```

```

# Wednesday: alcohol vs ecstasy
wilcox_wednes <- stats::wilcox.test(alcohol_wednes,
  ↪  ecstasy_wednes,
                                paired      = FALSE,
                                alternative = "two.sided",
                                mu          = 0,
                                conf.int    = TRUE,
                                conf.level  = 0.95)

print(wilcox_wednes)

```

Wilcoxon rank sum exact test

```

data:  alcohol_wednes and ecstasy_wednes
W = 96, p-value = 0.0001299
alternative hypothesis: true location shift is not equal to 0
96.3 percent confidence interval:
 18 29
sample estimates:
difference in location
      23.5

```

Interpretación de los resultados. El output de R incluye el estadístico W (suma de rangos del primer grupo), el p-valor y, si se solicita, el intervalo de confianza para la diferencia de medianas (pseudomediana de Hodges-Lehmann). Conviene tener en cuenta que R denomina a este estadístico W, mientras que otros programas estadísticos utilizan U; los valores numéricos difieren, pero el **p-valor es matemáticamente equivalente** y la conclusión estadística es idéntica.

- Si $p < 0.05$: se rechaza H_0 y se concluye que los dos grupos presentan distribuciones significativamente distintas — los perfiles de depresión del grupo de alcohol y del grupo de éxtasis difieren en ese día de la semana.
- Si $p \geq 0.05$: no hay evidencia suficiente para afirmar que las distribuciones sean distintas.

La interpretación debe realizarse para cada día de medición por separado, ya que el estado del sujeto puede variar entre el domingo (normalmente el día de mayor consumo) y el miércoles (varios días después).

3.7.4. Prueba de Wilcoxon — Muestras dependientes (V de Wilcoxon)

Prueba de rangos de Wilcoxon [equivalentes a t-test para muestras dependientes]



Descripción de la prueba

- Situación
 - Mismo uso que los de muestras relacionadas t-test, es decir queremos saber si una variable toma valores medios distintos en un mismo subgrupo pero medida en dos momentos del tiempo.
 - Sin embargo, la variable dependiente no sigue una distribución normal.
 - La hipótesis nula es que no hay diferencias de rangos.
- Test disponibles
 - Wilcoxon signed-rank test (*)
 - Otros test disponibles
 - Test de signos, igual al de Wilcoxon, pero en lugar de tener en cuenta el tamaño de la diferencia, sólo tiene en cuenta el signo de la misma. Su poder de prueba es mínimo y solo tiene sentido para tamaños muestrales muy pequeños (6 o menos)
 - Test de McNemar, es útil si los datos son nominales, más que ordinales. Normalmente dos variables dicotómicas que están relacionadas.
 - Test "marginal homogeneity", extensión del de McNemar para variables ordinales; es prácticamente idéntico a Wilcoxon.

Lógica de la prueba

- Ejemplo (bdidroga.sav).
 - Una de las consecuencias del consumo de drogas es la depresión.
 - Queremos saber si los individuos que han tomado una de las drogas, tienen mayores o menores niveles de depresión el domingo que el miércoles, de acuerdo con la escala BDI (Beck Depression Inventory).
- ¿Cómo se hace?
 - no se ordena la variable dependiente;
 - se calcula la diferencia para cada individuo siempre 2^a-1^a ;
 - si la diferencia es 0, se excluye;
 - se ordenan los casos atendiendo a la diferencia e ignorando el signo;
 - los empates se tratan igual que en el test anterior;
 - se separa los rangos según tengan signo positivo o negativo;
 - se suma por separado los rangos positivos y negativos.
- En nuestro caso, queremos saber si hay diferencias entre los consumidores de alcohol en miércoles y domingo y los consumidores de éxtasis en miércoles y domingo. Cada droga es independiente de la otra, cada dos pruebas, una prueba por droga.



35

Cuando se quiere comparar dos mediciones del **mismo sujeto** en momentos distintos — en lugar de comparar dos grupos diferentes —, el diseño es dependiente o apareado. En este caso, cada individuo actúa como su propio control, lo que elimina la variabilidad entre sujetos y hace el análisis más potente.

El equivalente no paramétrico del test t para muestras pareadas es la **prueba de los rangos con signo de Wilcoxon**. A diferencia del caso anterior, aquí se trabaja dentro de cada grupo (alcohol o éxtasis) y se comparan las puntuaciones del domingo con las del miércoles para los mismos individuos.

3.7.4.1. Diagnóstico de normalidad de las diferencias

Para el test t pareado, el supuesto que debe cumplirse no es la normalidad de cada variable por separado, sino la **normalidad de las diferencias** dentro de cada par. Cuando ese supuesto no se sostiene — o cuando el tamaño muestral es pequeño y no puede verificarse con fiabilidad —, la alternativa no paramétrica es la prueba de Wilcoxon pareada.

En los grupos de este estudio, los tamaños muestrales son reducidos y los diagnósticos anteriores ya han mostrado desviaciones de la normalidad. Esto refuerza la decisión de emplear la versión no paramétrica también para las comparaciones intra-grupo.

```

# --- Wilcoxon signed-rank test (paired) ---
# Alcohol group: Sunday vs Wednesday
wilcox_alcohol_paired <- stats::wilcox.test(alcohol_sunday,
  ↪ alcohol_wednes,

paired      =
  ↪ TRUE,    # Same
  ↪ subjects, two
  ↪ occasions
alternative =
  ↪ "two.sided",
mu          = 0,
conf.int    =
  ↪ TRUE,
conf.level  =
  ↪ 0.95)

print(wilcox_alcohol_paired)

```

Wilcoxon signed rank exact test

data: alcohol_sunday and alcohol_wednes

V = 0, p-value = 0.007812

alternative hypothesis: true location shift is not equal to 0

96.5 percent confidence interval:

-18.0 -6.5

sample estimates:

LABEL: Depression BDI domingo

VALUES:

-13

```

# Ecstasy group: Sunday vs Wednesday
wilcox_ecstasy_paired <- stats::wilcox.test(ecstasy_sunday,
  ↪ ecstasy_wednes,

paired      =
  ↪ TRUE,

```

```

alternative =
  ↪ "two.sided",
mu          = 0,
conf.int    =
  ↪ TRUE,
conf.level  =
  ↪ 0.95)

print(wilcox_ecstasy_paired)

```

Wilcoxon signed rank exact test

```

data:  ecstasy_sunday and ecstasy_wednes
V = 47, p-value = 0.04492
alternative hypothesis: true location shift is not equal to 0
95.3 percent confidence interval:
 2.0 10.5
sample estimates:
LABEL: Depression BDI domingo
VALUES:
7.5

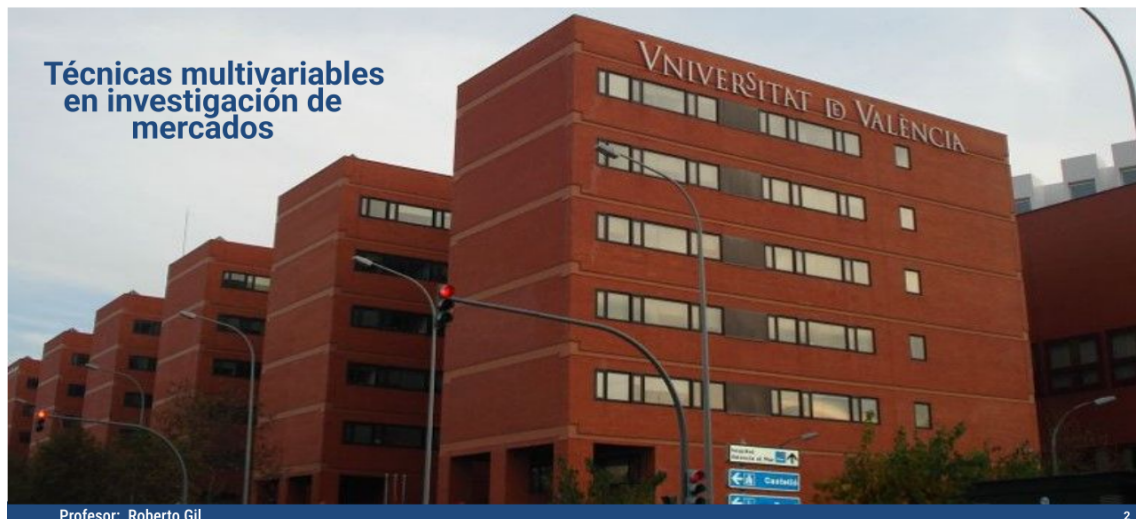
```

Interpretación de los resultados. El estadístico que devuelve R en este caso se denomina V y corresponde a la **suma de rangos positivos** de las diferencias (domingo — miércoles). El p-valor se interpreta de la misma forma que en cualquier contraste de hipótesis:

- Si $p < 0.05$: se rechaza H_0 y se concluye que existe un cambio estadísticamente significativo en las puntuaciones de depresión entre el domingo y el miércoles dentro de ese grupo. Dicho de otro modo, el patrón de consumo tiene un impacto detectable en el estado de ánimo a lo largo de la semana.
- Si $p \geq 0.05$: no hay evidencia de que el estado de ánimo varíe significativamente entre los dos momentos de medición.

Al igual que en el caso independiente, el estadístico V de R no coincide numéricamente con el que ofrecen otros programas estadísticos, pero el **p-valor es equivalente** y la conclusión no varía. El intervalo de confianza al 95 % para la pseudomediana de las diferencias permite cuantificar también la magnitud del cambio observado.

4 Análisis de Correspondencias Simple (CA)



Profesor: Roberto Gil

2

Contenidos del tema / Índice



1. Análisis de correspondencias (simple)

1. Introducción a los mapas de dispersión
2. Planteamiento del problema a resolver
3. La práctica del análisis de correspondencias
4. Solución en SPSS
5. Decálogo de interpretación de un análisis de correspondencias
6. Introducción de filas /columnas suplementarias

Bibliografía recomendada

- Hair, J.F., Anderson, R.E., Tatham, R.L. Y Black, W. (1999). Análisis multivariante. 4ª edición. Madrid: Prentice Hall.
- Uriel, E. y Aldás, J. (2005). Análisis multivariante aplicado. Madrid: Thomson.
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). "An introduction to statistical learning" (Vol. 112). New York, NY: Springer.
- De Vries, A., & Meys, J. (2015). R for Dummies. John Wiley & Sons.
- Chapman, C., & Feit, E. M. (2015). R for marketing research and analytics. New York, NY: Springer.

4.1. Introducción

El análisis de correspondencias (CA) es una técnica de interdependencia cuyo objetivo es la representación geométrica de las relaciones bidimensionales derivadas de una tabla de contingencia.

Análisis de correspondencias



Profesor/a: Roberto Gil

7

El análisis de correspondencias trabaja con variables categóricas o no métricas (frecuencias, recuentos) y permite pasar del aparente caos de una tabla muy grande para que visualmente se aprecien relaciones entre los datos a la claridad de un mapa. Su principal ventaja es que ayuda a visualizar la asociación o similitud entre los niveles de las variables de fila y columna basándose en el concepto de distancia (distancia χ^2).

Antes de adentrarnos en el desarrollo de la técnica, vamos a recordar, ajustar unos conceptos ya conocidos, pero que aquí toman una gran relevancia para el entendimiento de la técnica.

4.1.1. Tabla de frecuencias

Antes de abordar la ejecución matemática y visual del análisis de correspondencias, es fundamental repasar cuatro conceptos estadísticos sobre los que se asienta esta técnica:

- **Frecuencia observada y esperada:** la *frecuencia observada* es el recuento real de casos empíricos en la tabla de contingencia (las respuestas directas de la encuesta). La *frecuencia esperada*, por su parte, es el recuento teórico que debería encontrarse en cada celda si no existiera absolutamente ninguna relación (independencia estadística) entre las variables de fila y columna.
- **Residuo:** es la diferencia matemática entre lo observado en la realidad y lo esperado de forma teórica. Un residuo positivo señala una “atracción” o asociación mayor de lo normal entre dos categorías, mientras que un residuo negativo indica una falta de vinculación o “repulsión”.
- **Prueba χ^2 (Chi-cuadrado):** es el contraste estadístico que evalúa el conjunto total de los residuos de la tabla. Permite confirmar si las discrepancias entre lo observado y lo esperado son lo suficientemente grandes como para descartar que sean fruto del azar, validando así que existe una asociación significativa en el mercado.

4.1.2. Mapa cartesiano

Por otro lado, tenemos los mapas:

- **Un mapa cartesiano:** consiste en la proyección geométrica bidimensional (el plano formado por los ejes coordenados) de las relaciones ocultas en la tabla de contingencia. En este mapa, la inercia y los residuos se traducen en distancias y ángulos, lo que permite leer visualmente el posicionamiento competitivo de forma intuitiva.

Algunas consideraciones sobre la representación gráfica de dos variables en un plano cartesiano. Un mapa cartesiano tiene dos ejes principales. Si hablamos de geografía:

- X -> latitud
- Y -> longitud

X no tiene por qué estar medida en la misma escala que Y, de modo que solo se interpretan distancias relativas, no valores absolutos.

Bien a título de ejemplo y como ilustración de lo que luego veremos, vamos a reflexionar sobre este ejemplo. Pensemos en un mapa geográfico ...

- ¿Qué muestra? Distancias
- ¿Qué unidad de medida presenta? kilómetros pero con gradaciones

- ¿Podríamos a partir de las distancias kilométricas crear un mapa?

Es el ejemplo clásico de “a partir de las distancias entre ciudades, reconstruir su mapa”:

	Albacete	Alicante	Almería	Avila	Badajoz	Barcelona	Bilbao	Burgos	Cáceres	Cádiz	Castellón	Ciudad Real	Córdoba	A Coruña	Cuenca	Gerona	Granada	Guadalajara	Huelva	Huesca	Jáen	León	Lérida	Lugo
Albacete	0	171	369	366	525	540	646	488	504	617	256	207	354	860	142	640	363	309	596	495	264	584	515	578
Alicante	171	0	294	537	696	515	817	659	675	688	231	378	525	1031	313	615	353	480	703	570	415	855	499	653
Almería	369	294	0	663	664	809	958	808	651	484	525	407	332	1172	511	909	1166	621	516	830	228	896	862	899
Avila	366	537	663	0	318	717	401	245	229	418	512	256	487	538	282	817	134	173	512	490	435	255	558	358
Badajoz	525	696	664	318	0	1022	494	536	89	342	808	318	272	772	558	1122	438	439	251	798	376	496	868	676
Barcelona	540	515	809	717	1022	0	620	583	918	1284	284	811	968	1118	562	100	868	563	1140	274	804	784	156	468
Bilbao	646	817	958	401	494	620	0	158	605	1058	467	585	795	644	562	720	829	396	939	322	730	359	464	152
Burgos	488	659	808	245	536	583	158	0	447	900	524	427	637	535	404	683	671	238	781	359	572	201	427	115
Cáceres	504	675	651	229	89	918	605	447	0	369	701	324	319	683	451	1018	485	355	323	694	423	407	762	595
Cádiz	617	688	484	618	342	1284	1058	900	369	0	873	464	263	1072	708	1384	335	721	219	1060	367	796	1128	999

1-10 of 47 rows

Previous 1 2 3 4 5 Next

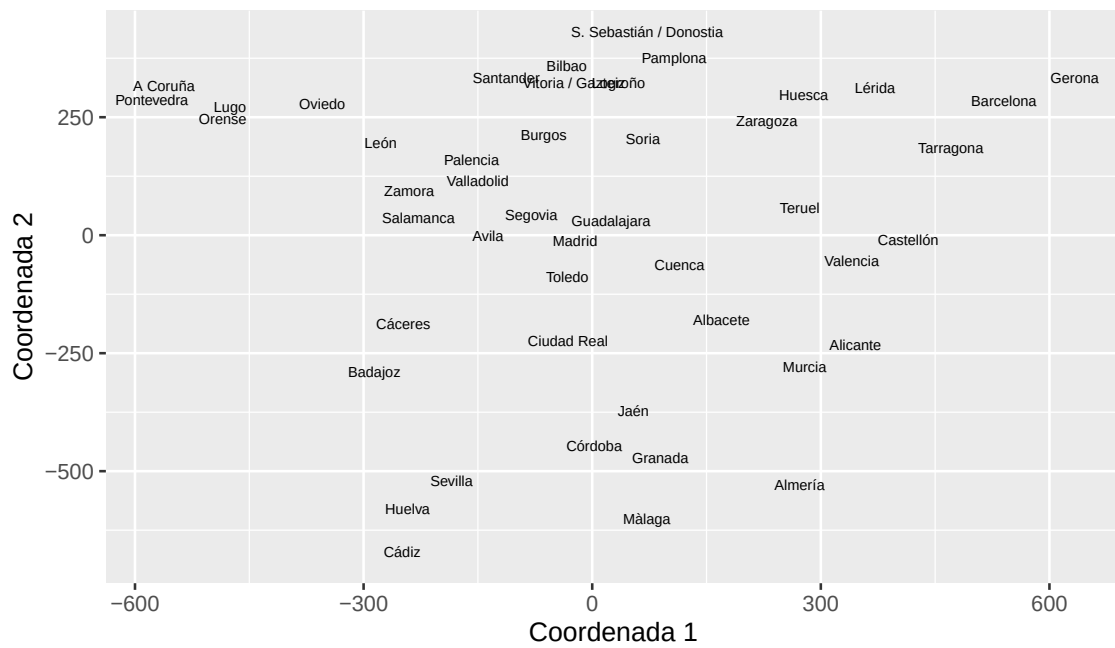
Existe una función denominada `stats::cmdscale()` (escalado multidimensional o *MDS*) permite reconstruir un mapa de posiciones relativas a partir, únicamente, de una matriz de distancias entre puntos.

	V1	V2
Albacete	-191.618357819543	1669.667202318786
Alicante	-243.34869310662	345.52212765675
Almeria	-541.249487539699	272.148367135375
Avila	-11.8509256588769	-136.666667332418
Badajoz	-101.71856155019	-285.898356366496
Barcelona	272.267303220385	540.37867869414
Bilbao	346.864449092666	-33.2748884521945
Burgos	201.721613450526	-63.6626520005146
Caceres	-199.913810622398	-247.774768116137
Caliz	-682.761931068887	-249.279373038894
Cantabria	-20.4518311433532	414.721297163423
Catala Real	-234.28621005638	-31.821099466741
Cordoba	-637.982673844109	2.6452599021289
A Coruña	305.62779384778	-561.796768488956
Cuenca	-74.7795917197863	114.6753353884952
Girona	323.681088894808	633.1617821451186
Granada	-483.083095758497	89.4225831587641
Guadalajara	19.0914814799559	24.8976297031649
Huelva	-590.288704317652	-242.29295403389
Huesca	285.533913057468	277.568199843608
Jala	-183.826951320945	54.0614602854802
León	184.126237043996	-277.597991548761
Lerida	300.927645201848	371.106516615383
Lugo	311.907478700067	34.5429921749425
Lugo	260.591256998907	-475.283115457202
Madrid	-24.3326211623248	-22.456326888159
Malaga	-611.589420105635	71.7578156757567
Marcia	-289.354852105555	278.840352754438
Orense	235.713480159707	-484.899585226979
Oviedo	266.109323023678	-354.1835453630094
Palencia	148.637056841212	-158.2366004737118
Pamplona	363.40342494489	107.515495580244
Pontevedra	275.540188044227	-577.48877021914
Salamanca	24.4468917726808	-228.085778484228
S. Sebastian / Donostia	420.097123948761	72.1922025112948
Santander	231.426478588097	-112.69767862553
Segovia	51.0345478476111	-79.8674817982113
Sevilla	-531.118655372795	-184.2844825747037
Soria	191.817915710133	66.6686915759172
Tarazona	174.161478680073	476.6373204536
Tenali	46.9524004182968	272.4454518427903
Tobelo	-99.7605681223795	-32.6081747371256
Valencia	-45.5002157969184	340.552354469889
Valladolid	104.308438762838	-130.623962581145
Vitoria / Gasteiz	312.602702422716	-23.7096507135494
Zamora	82.6370321302629	-240.2382382688158
Zaragoza	229.89812716736	229.162028175629

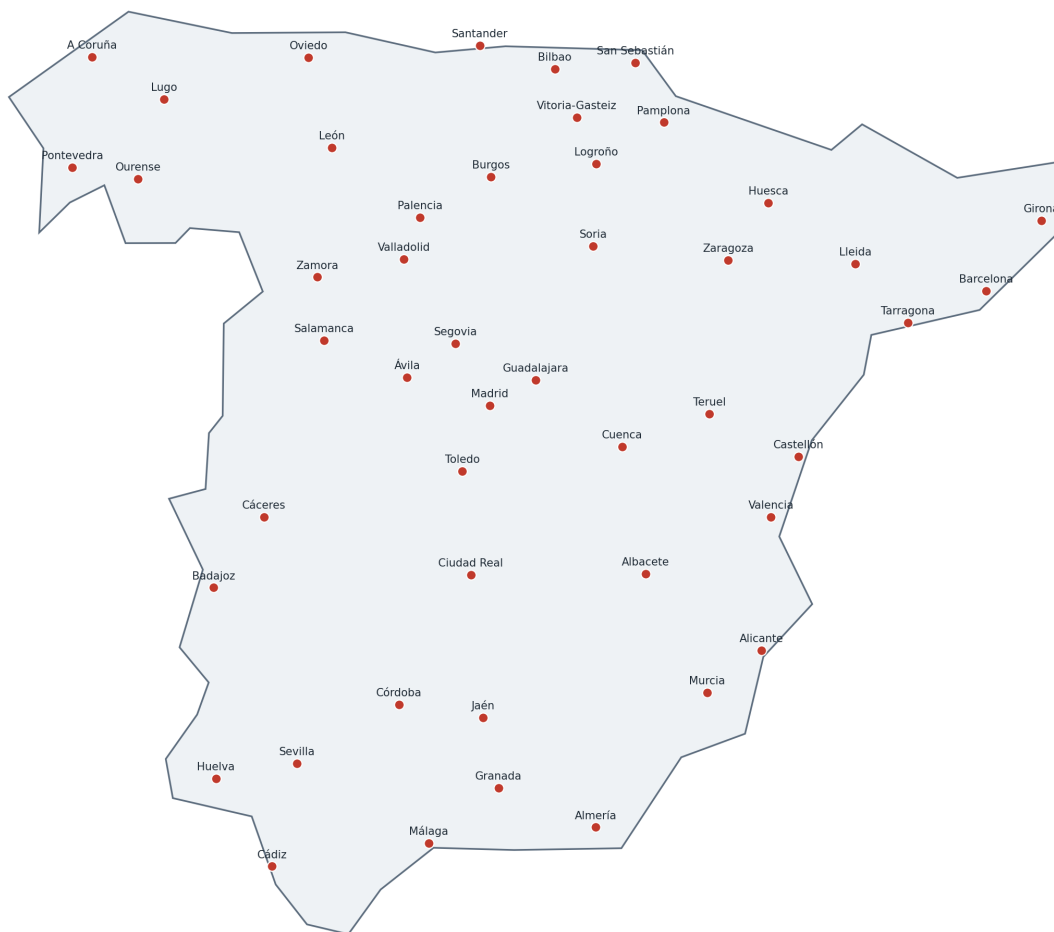
\$names

```
[1] "points" "eig"      "x"       "ac"      "GOF"
```

Una vez tenemos las coordenadas, que es la parte mágica de cálculo, ya todo es más simple. Solo debemos buscar la forma de hacer un gráfico de puntos X,Y.



Capitales de provincia de la España peninsular



Estamos listos, hemos sentado las bases para entender nuestra primera técnica avan-

zada.

4.2. La práctica del CA

Supongamos la siguiente situación para la empresa imaginaria HATCO, donde existe un problema desde la perspectiva del Director de Marketing de HATCO, que ha identificado 9 competidores principales (empresas A a I) y ha pedido a sus clientes que asocien a estas empresas (incluyendo HATCO) con 8 atributos competitivos:

- x1: Rapidez del servicio
- x2: Nivel de precios
- x3: Flexibilidad de precios
- x4: Imagen del fabricante
- x5: Calidad del servicio
- x6: Imagen de los vendedores
- x7: Calidad del producto
- x8: Tamaño de la empresa

Cómo resultado de eestio, ¿qué tenemos? una tabla de frecuencias de este modo.

Del caos de los datos a la claridad de un mapa



ATRIBUTO	EMPRESA									
	HATCO	A	B	C	D	E	F	G	H	I
X1 Rapidez del servicio	16	13	8	13	9	17	15	16	6	12
X2 Nivel de precios	14	14	10	11	11	14	12	13	10	14
X3 Flexibilidad de precios	6	6	14	10	11	8	7	4	14	4
X4 Imagen del fabricante	15	18	9	2	3	15	16	7	8	8
X5 Calidad del servicio	15	14	6	4	4	15	14	13	7	13
X6 Imagen de los vendedores	7	18	13	4	9	16	14	5	4	16
X7 Calidad del producto	4	3	1	13	9	6	3	18	2	10
X8 Tamaño de la empresa	15	16	15	11	11	14	16	12	14	14

Imagina que eres el Director de Marketing de HATCO. Recibes esta tabla con datos de 10 empresas y 8 atributos. Y te preguntas ...

- ¿Quiénes son nuestros verdaderos competidores?
- ¿En qué nos diferenciamos? ¿Cuál es nuestra mayor competencia?
- ¿Hay algún hueco en el mercado que no estemos viendo?

Mirando esta tabla...

Profesor/a: Roberto Gil

8

Las preguntas estratégicas que plantea el caso son:

1. ¿Quiénes son los verdaderos competidores de HATCO?
2. ¿En qué se diferencia HATCO?

3. ¿Existe algún hueco de mercado sin cubrir?

Conviene ser cuidadoso respecto al sentido del análisis en función de la representación de los datos. En este ejemplo conviene verificar teóricamente que:

- los poseedores de los atributos son comparables respecto a esos atributos;
- las empresas A, B, ..., I son efectivamente competidoras de HATCO;
- su diferenciación se basa en los atributos x1 a x8;
- el listado de atributos es exhaustivo y no deja fuera ninguno relevante para la caracterización de las empresas.

4.3. Carga de los datos en forma de tabla

A continuación se cargan las librerías necesarias y se construye la matriz de datos. En este caso, los cálculos del análisis de correspondencias se realizarán con el paquete *ca* (escuela de Greenacre) Una de las necesidades de este paquete es que el *input* sea una tabla. No basta con que sea un *dataframe*. Este se debe convertir a objeto de clase *table*.

```
# Build the HATCO contingency table directly as a matrix
↪ (attributes x companies)
hatco_table <- read_csv("data/hatco_table.csv")

# Need to adapt dataframe to table
hatco_table <- hatco_table %>%
  column_to_rownames(var = "atributo") %>% # use first column
  ↪ as rownames
  as.matrix() %>%
  as.table()

# Show the object as table
hatco_table
```

	hatco	A	B	C	D	E	F	G	H	I
x1	16	13	8	13	9	17	15	16	6	12
x2	14	14	10	11	11	14	12	13	10	14
x3	6	6	14	10	11	8	7	4	14	4

x4	15	18	9	2	3	15	16	7	8	8
x5	15	14	6	4	4	15	14	13	7	13
x6	7	18	13	4	9	16	14	5	4	16
x7	4	3	1	13	9	6	3	18	2	10
x8	15	16	15	11	11	14	16	12	14	14

4.4. Prueba de homogeneidad (Chi-cuadrado)

Antes de crear el mapa y adentrarnos en lo que es el CA, es necesario confirmar empíricamente que existe dependencia estadística entre las filas (atributos) y las columnas (empresas). La hipótesis nula (H_0) es la independencia: las empresas no se diferencian por estos atributos. Recordemos que esta prueba ya la hemos visto anteriormente como la forma en la que el investigador determina que las columnas y las filas de una tabla de contingencia y por tanto que las dos dimensiones medidas no son independientes. Si rechazamos la (H_0), estaremos afirmando que existe una relación de dependencia entre las dos dimensiones.

```
# Chi-square test of homogeneity on the contingency table
chi2_homogeneity <- stats::chisq.test(hatco_table)
cat("Chi-square test - Test de homogeneidad: \n")
print(chi2_homogeneity)
```

Chi-square test - Test de homogeneidad:

Pearson's Chi-squared test

data: hatco_table

X-squared = 122.6, df = 63, p-value = 1.033e-05

Como nos iremos acostumbrando en todas las pruebas que hagamos, al obtener un $p\text{-value} < 0.05$, se rechaza H_0 , por lo que existe una asociación significativa entre las marcas y los atributos, y puede procederse a crear el mapa. Si no pudiera rechazarse la H_0 , ¿convendría seguir adelante? No hay una respuesta única: es probable que no aparezcan relaciones de mucho peso, pero las relaciones se representarían en el plano de igual forma.

¿Qué marcas o qué atributos determinan las diferencias? Esta es la primera pregunta a resolver, y la respuesta comienza con el análisis de los residuos.

4.5. Análisis de residuos y su tipología

Una vez confirmado que existe una relación significativa, el primer paso es explorar los residuos. Estos representan la diferencia entre las frecuencias observadas en la muestra y las frecuencias que se esperarían encontrar si las variables fueran totalmente independientes (frecuencias esperadas).

Recordemos que tenemos creado el objeto `ch2_homogeneity` que tiene diferentes resultados que no hemos explorado. Uno de ellos son las frecuencias esperadas.

```
# Expected frequency
chi2_homogeneity$expected
```

	hatco	A	B	C	D	E	F
x1	13.513514	14.98237	11.163337	9.988249	9.841363	15.423032	14.247944
x2	13.297297	14.74266	10.984724	9.828437	9.683901	15.176263	14.019976
x3	9.081081	10.06816	7.501763	6.712103	6.613396	10.364277	9.574618
x4	10.918919	12.10576	9.019976	8.070505	7.951821	12.461810	11.512338
x5	11.351351	12.58519	9.377203	8.390129	8.266745	12.955347	11.968273
x6	11.459459	12.70505	9.466510	8.470035	8.345476	13.078731	12.082256
x7	7.459459	8.27027	6.162162	5.513514	5.432432	8.513514	7.864865
x8	14.918919	16.54054	12.324324	11.027027	10.864865	17.027027	15.729730

	G	H	I
x1	12.925969	9.547591	13.366627
x2	12.719154	9.394830	13.152761
x3	8.686251	6.415981	8.982374
x4	10.444183	7.714454	10.800235
x5	10.857814	8.019976	11.227967
x6	10.961222	8.096357	11.334900
x7	7.135135	5.270270	7.378378
x8	14.270270	10.540541	14.756757

El siguiente paso, responde a la definicion de **residuo directo**. Diferencia entre la frecuencia observada y la esperada.

```
# Direct residuals: observed - expected
chi2_homogeneity$observed - chi2_homogeneity$expected
```

	hatco		A	B	C	D	E
x1	2.48648649	-1.98237368	-3.16333725	3.01175088	-0.84136310	1.57696827	
x2	0.70270270	-0.74265570	-0.98472385	1.17156287	1.31609871	-1.17626322	
x3	-3.08108108	-4.06815511	6.49823737	3.28789659	4.38660400	-2.36427732	
x4	4.08108108	5.89424207	-0.01997650	-6.07050529	-4.95182139	2.53819036	
x5	3.64864865	1.41480611	-3.37720329	-4.39012926	-4.26674501	2.04465335	
x6	-4.45945946	5.29494712	3.53349001	-4.47003525	0.65452409	2.92126910	
x7	-3.45945946	-5.27027027	-5.16216216	7.48648649	3.56756757	-2.51351351	
x8	0.08108108	-0.54054054	2.67567568	-0.02702703	0.13513514	-3.02702703	

	F	G	H	I
x1	0.75205640	3.07403055	-3.54759107	-1.36662750
x2	-2.01997650	0.28084606	0.60517039	0.84723854
x3	-2.57461810	-4.68625147	7.58401880	-4.98237368
x4	4.48766157	-3.44418331	0.28554642	-2.80023502
x5	2.03172738	2.14218566	-1.01997650	1.77203290
x6	1.91774383	-5.96122209	-4.09635723	4.66509988
x7	-4.86486486	10.86486486	-3.27027027	2.62162162
x8	0.27027027	-2.27027027	3.45945946	-0.75675676

Para identificar con precisión qué celdas concretas generan la asociación, se calculan los **residuos estandarizados**. Estos residuos funcionan como una puntuación Z, lo que permite realizar comparaciones estadísticas directas entre celdas:

```
# Raw residuals from the chi-square object
round(chi2_homogeneity$residuals, 3)
```

	hatco	A	B	C	D	E	F	G	H	I
x1	0.676	-0.512	-0.947	0.953	-0.268	0.402	0.199	0.855	-1.148	-0.374
x2	0.193	-0.193	-0.297	0.374	0.423	-0.302	-0.539	0.079	0.197	0.234
x3	-1.022	-1.282	2.373	1.269	1.706	-0.734	-0.832	-1.590	2.994	-1.662
x4	1.235	1.694	-0.007	-2.137	-1.756	0.719	1.323	-1.066	0.103	-0.852
x5	1.083	0.399	-1.103	-1.516	-1.484	0.568	0.587	0.650	-0.360	0.529
x6	-1.317	1.486	1.148	-1.536	0.227	0.808	0.552	-1.801	-1.440	1.386

x7	-1.267	-1.833	-2.080	3.188	1.531	-0.861	-1.735	4.067	-1.425	0.965
x8	0.021	-0.133	0.762	-0.008	0.041	-0.734	0.068	-0.601	1.066	-0.197

Cuando estos residuos se corrigen de un determinado modo, se da lugar a una puntuación Z directa donde:

- **Valores > 1.96:** Indican que la frecuencia observada es significativamente superior a la esperada (hay “exceso” de casos en esa celda).
- **Valores < -1.96:** Indican que la frecuencia observada es significativamente inferior a la esperada (hay “defecto” de casos en esa celda).

Estos valores permiten señalar las celdas que realmente “rompen” la hipótesis de independencia con un nivel de confianza del 95 %.

La corrección de Haberman (1973) ajusta el residuo de Pearson estandarizado dividiéndolo por su **error estándar real**, que es menor que 1 (de ahí que el residuo estandarizado simple infravalore la significatividad). Esta es la corrección que hace que el residuo siga aproximadamente una distribución normal estándar (0, 1), por lo que el criterio de “significativo si $|\text{valor}| > 1.96$ ” solo es válido sobre `$stdres`, no sobre `$residuals`.

```
# Standardized residuals, rounded for easier interpretation
round(chi2_homogeneity$stdres, 3)
```

	hatco	A	B	C	D	E	F	G	H	I
x1	0.775	-0.591	-1.074	1.076	-0.303	0.464	0.229	0.978	-1.293	-0.428
x2	0.221	-0.223	-0.337	0.421	0.476	-0.349	-0.620	0.090	0.222	0.267
x3	-1.140	-1.440	2.619	1.394	1.872	-0.826	-0.931	-1.769	3.282	-1.853
x4	1.393	1.923	-0.007	-2.373	-1.949	0.818	1.497	-1.199	0.114	-0.960
x5	1.225	0.454	-1.234	-1.688	-1.651	0.648	0.666	0.733	-0.400	0.598
x6	-1.491	1.692	1.286	-1.711	0.252	0.922	0.626	-2.032	-1.601	1.567
x7	-1.399	-2.038	-2.273	3.467	1.664	-0.960	-1.922	4.481	-1.546	1.065
x8	0.024	-0.155	0.873	-0.009	0.047	-0.856	0.079	-0.693	1.211	-0.228

4.6. Masas y perfiles: la base del cálculo de distancias

Antes de que el algoritmo calcule los ejes del mapa, el análisis de correspondencias transforma las frecuencias absolutas (los recuentos) en frecuencias relativas.

Para entender de dónde salen las coordenadas, conviene definir tres conceptos estructurales:

- **Masa (Mass):** es la importancia relativa o el “peso” de cada categoría respecto al total de la muestra (porcentaje sobre el gran total). En el mapa, una categoría con mucha masa tiene mayor capacidad para “tirar” del centro de gravedad hacia sí misma. Existen masas de fila (peso de cada atributo) y masas de columna (peso de cada empresa).
- **Perfil de fila (Row profile):** muestra cómo se distribuyen las respuestas de un atributo específico entre todas las marcas. Se calcula dividiendo la frecuencia de cada celda entre el total de su fila (es el equivalente al porcentaje de fila). La suma de cada perfil de fila siempre es 1 (o 100 %).
- **Perfil de columna (Column profile):** indica cómo se compone la valoración de una marca específica a través de todos los atributos. Se calcula dividiendo la celda entre el total de su columna (porcentaje de columna).

Si el perfil de una marca es idéntico al perfil medio del mercado, esa marca se sitúa exactamente en el centro del mapa (0,0).

En el análisis de correspondencias, la famosa “distancia χ^2 ” no es más que la diferencia matemática entre el perfil de una categoría y el perfil promedio (su masa).

A continuación se obtienen estos cálculos matriciales en R utilizando la función `prop.table()` sobre la tabla de contingencia original:

Advertencia

Estos cálculos no son necesarios, pues los ofrecerá también el modelo (con otro formato), pero de alguna forma quiero que se entienda cual es su origen.

```
# View input table
hatco_table
margin.table(hatco_table)
```

```

# 1. Calculate and display masses (marginal proportions)
# Row masses: weight of each attribute
row_masses <- prop.table(margin.table(hatco_table, 1))
cat("\n\n\n--- Row Masses (Attributes) ---\n")
round(row_masses, 3)

# Column masses: weight of each company
col_masses <- prop.table(margin.table(hatco_table, 2))
cat("\n\n\n--- Column Masses (Companies) ---\n")
round(col_masses, 3)

# 2. Calculate row profiles (sum of each row = 1)
# margin = 1 indicates calculations across rows
row_profiles <- prop.table(hatco_table, margin = 1)
cat("\n\n\n--- Row Profiles | rows sum 1---\n")
round(row_profiles, 3)

# 3. Calculate column profiles (sum of each column = 1)
# margin = 2 indicates calculations down columns
col_profiles <- prop.table(hatco_table, margin = 2)
cat("\n\n\n--- Column Profiles | columns sum 1---\n")
round(col_profiles, 3)

```

	hatco	A	B	C	D	E	F	G	H	I
x1	16	13	8	13	9	17	15	16	6	12
x2	14	14	10	11	11	14	12	13	10	14
x3	6	6	14	10	11	8	7	4	14	4
x4	15	18	9	2	3	15	16	7	8	8
x5	15	14	6	4	4	15	14	13	7	13
x6	7	18	13	4	9	16	14	5	4	16
x7	4	3	1	13	9	6	3	18	2	10
x8	15	16	15	11	11	14	16	12	14	14

[1] 851

--- Row Masses (Attributes) ---

	x1	x2	x3	x4	x5	x6	x7	x8
	0.147	0.145	0.099	0.119	0.123	0.125	0.081	0.162

--- Column Masses (Companies) ---

hatco	A	B	C	D	E	F	G	H	I
0.108	0.120	0.089	0.080	0.079	0.123	0.114	0.103	0.076	0.107

--- Row Profiles | rows sum 1---

	hatco	A	B	C	D	E	F	G	H	I
x1	0.128	0.104	0.064	0.104	0.072	0.136	0.120	0.128	0.048	0.096
x2	0.114	0.114	0.081	0.089	0.089	0.114	0.098	0.106	0.081	0.114
x3	0.071	0.071	0.167	0.119	0.131	0.095	0.083	0.048	0.167	0.048
x4	0.149	0.178	0.089	0.020	0.030	0.149	0.158	0.069	0.079	0.079
x5	0.143	0.133	0.057	0.038	0.038	0.143	0.133	0.124	0.067	0.124
x6	0.066	0.170	0.123	0.038	0.085	0.151	0.132	0.047	0.038	0.151
x7	0.058	0.043	0.014	0.188	0.130	0.087	0.043	0.261	0.029	0.145
x8	0.109	0.116	0.109	0.080	0.080	0.101	0.116	0.087	0.101	0.101

--- Column Profiles | columns sum 1---

	hatco	A	B	C	D	E	F	G	H	I
x1	0.174	0.127	0.105	0.191	0.134	0.162	0.155	0.182	0.092	0.132
x2	0.152	0.137	0.132	0.162	0.164	0.133	0.124	0.148	0.154	0.154
x3	0.065	0.059	0.184	0.147	0.164	0.076	0.072	0.045	0.215	0.044
x4	0.163	0.176	0.118	0.029	0.045	0.143	0.165	0.080	0.123	0.088
x5	0.163	0.137	0.079	0.059	0.060	0.143	0.144	0.148	0.108	0.143
x6	0.076	0.176	0.171	0.059	0.134	0.152	0.144	0.057	0.062	0.176
x7	0.043	0.029	0.013	0.191	0.134	0.057	0.031	0.205	0.031	0.110
x8	0.163	0.157	0.197	0.162	0.164	0.133	0.165	0.136	0.215	0.154

Vistos por tanto los cálculos que en una tabla se pueden hacer, pasamos a realizar

la estimación del modelo y a la aplicación de una primera modelización.

4.7. Estimación del modelo y resultados

El paquete `ca`, *correspondence analysis*, ofrece una salida de resultados basada en la escuela francesa, desarrollada por Michael Greenacre, uno de los autores que más ha contribuido al desarrollo de esta técnica.

```
# Execute Correspondence Analysis
ca_greenacre <- ca::ca(hatco_table)

# Display the summary (inertia, chi-square, and decomposition)
# This provides the 'scree plot' information in table format
summary(ca_greenacre)
```

Principal inertias (eigenvalues):

dim	value	%	cum%	scree plot
1	0.076541	53.1	53.1	*****
2	0.047813	33.2	86.3	*****
3	0.015291	10.6	96.9	***
4	0.002658	1.8	98.8	
5	0.000806	0.6	99.3	
6	0.000576	0.4	99.7	
7	0.000381	0.3	100.0	

Total: 0.144066 100.0

Rows:

	name	mass	qlt	inr	k=1	cor	ctr	k=2	cor	ctr
1	x1	147	619	41	107	289	22	115	330	40
2	x2	145	527	8	61	469	7	-21	58	1
3	x3	99	991	231	23	2	1	-578	989	689
4	x4	119	901	132	-356	789	196	133	111	44

5		x5		123	816	70		-106	138	18		235	677	142	
6		x6		125	372	129		-232	358	87		46	14	6	
7		x7		81	991	367		792	961	665		139	30	33	
8		x8		162	772	22		-43	93	4		-115	678	45	

Columns:

		name	mass	qlt	inr	k=1	cor	ctr	k=2	cor	ctr				
1		hatc		108	433	62		-130	206	24		137	228	42	
2		A		120	928	86		-283	772	125		127	156	40	
3		B		89	942	115		-233	294	63		-346	648	224	
4		C		80	975	180		535	882	299		-174	93	50	
5		D		79	863	88		268	445	74		-260	418	111	
6		E		123	812	29		-124	456	25		110	356	31	
7		F		114	954	52		-232	810	80		98	144	23	
8		G		103	963	204		465	762	292		239	201	123	
9		H		76	797	128		-109	49	12		-425	748	289	
10		I		107	446	56		65	55	6		171	390	66	

La salida de este comando permite validar si el mapa obtenido es una representación fiel de la realidad o si se está perdiendo demasiada información.

4.7.1. Bloque de inercia (Principal Inertias)

Esta tabla inicial indica cuántas “capas” de información (dimensiones) tiene la tabla y cuánta importancia tiene cada una.

- **dim (Dimension):** El número del eje. El primer eje (1) siempre es el que más información captura, el segundo (2) el siguiente, y así sucesivamente.
- **value (Eigenvalue/Inercia):** Es el valor propio de la dimensión. Representa la cantidad de varianza (inercia) que ese eje es capaz de explicar de la tabla original.
- **% (Percentage of inertia):** indica la proporción de la inercia total que captura ese eje. En investigación de mercados, se busca que los dos primeros ejes sumen un porcentaje elevado (idealmente > 70-80 %).
- **cum% (Cumulative percentage):** la suma acumulada de los porcentajes anteriores. Ayuda a decidir cuántas dimensiones son necesarias para entender el fenómeno.

- **scree plot:** Una representación visual (histograma de puntos) de la importancia relativa de cada eje.

4.7.2. Bloque de filas y columnas (Rows / Columns)

Esta sección es crucial para entender el comportamiento de cada marca o atributo individualmente. Ambos bloques tienen las mismas columnas:

- **mass (Masa):** Indica el peso relativo del punto en la muestra total. Una masa de 0.150 significa que esa categoría representa el 15 % de los datos. Los puntos con mucha masa tienen más “fuerza” para atraer los ejes hacia ellos.
- **qlt (Quality):** es la calidad de representación (de 0 a 1000). Indica qué tan bien representado está ese punto en el mapa actual (usualmente las dos primeras dimensiones). Si un punto tiene un **qlt** muy bajo (ej. < 100), su posición en el gráfico es engañosa y no deben extraerse conclusiones estratégicas sobre él.
- **inr (Inertia):** es la contribución del punto a la inercia total de la tabla. Indica cuánto “caos” o diferenciación aporta ese punto específico al mercado. Una marca muy distinta al resto tendrá un **inr** alto.
- **k=1, k=2 (Coordinates):** Son las coordenadas exactas del punto en el eje 1 y eje 2. Es lo que R utiliza para dibujar el punto en el mapa cartesiano.
- **cor (Correlation / Cos²):** Mide la asociación entre el punto y el eje específico. Un **cor** alto en el eje 1 significa que ese punto es el que mejor define lo que ese eje representa (ej. el eje del precio).
- **ctr (Contribution):** indica cuánto contribuye ese punto a la construcción de ese eje concreto. Ayuda a ponerle “nombre” a los ejes (ej. “si las marcas con más **ctr** en el eje 1 son las más caras, ese eje puede denominarse ‘eje de precio’”).

Al analizar la tabla, conviene seguir este orden lógico:

1. Revisar el **% acumulado** para saber si un mapa de 2 dimensiones es suficiente.
2. Observar el **qlt** de las marcas de interés; si es alto, puede confiarse en su posición en el mapa.
3. Utilizar el **ctr** para identificar qué variables son las responsables de que el mapa tenga esa forma y no otra.

4.7.3. Visualización

El paquete `ca` incluye su propia función `plot()`, que permite un control muy preciso sobre el escalado (simétrico, asimétrico, etc.) de los ejes visuales. Dado que se va a representar un mapa cartesiano, es importante conocer primero las coordenadas; no se entra aquí en el detalle del cálculo matemático para su obtención.

```
# Inspect the column and row coordinates computed by ca()
cat("\n\n")
cat("Coordenadas de columna =====:\n")
ca_greenacre$colcoord
cat("\n\n")
cat("Coordenadas de fila =====:\n")
ca_greenacre$rowcoord
cat("\n\n")
cat("\n\nRetenemos solo las que nos interesan ...\n\n")
# Inspect the column and row coordinates computed by ca() only
↪ Dim 1 & 2
cat("Coordenadas de columna 1 & 2 =====:\n")
ca_greenacre$colcoord[,1:2]
cat("\n\n")
cat("Coordenadas de fila 1 & 2 =====:\n")
ca_greenacre$rowcoord[,1:2]
```

```
Coordenadas de columna =====:
      Dim1      Dim2      Dim3      Dim4      Dim5      Dim6
hatco -0.4703327  0.6261240 -1.6935356  0.29465670 -1.7292066  0.4655331
A      -1.0217346  0.5801388  0.5338336 -0.07823246  1.1100906  1.6894406
B      -0.8432000 -1.5832948  0.7964316  0.26209544  0.2874627 -0.1659282
C       1.9333842 -0.7935744 -0.1648613  1.59150233 -0.5015929  1.0023065
D       0.9699347 -1.1891134  1.1762320  0.24157388 -0.3215290  0.1038722
E      -0.4498224  0.5024514  0.2356379  1.01941582  0.2256877 -2.0473172
F      -0.8381141  0.4468266 -0.2145714  0.65967745  0.3110986  0.1309981
G       1.6806871  1.0920696 -0.6758338 -0.77145803  1.5096436 -0.3224750
H      -0.3922857 -1.9443407 -1.6117961 -1.83337906  0.3985224 -0.4838085
```

I	0.2334232	0.7842522	1.4674729	-1.62005491	-1.4611537	-0.1711086
	Dim7					
hatco	-0.4460250					
A	-1.1173385					
B	1.3363279					
C	0.3955033					
D	-1.4550986					
E	-0.9157056					
F	1.6885279					
G	0.4368324					
H	-0.4346288					
I	0.5078109					

Coordenadas de fila =====:

	Dim1	Dim2	Dim3	Dim4	Dim5	Dim6
x1	0.38833337	0.52453784	-0.48064888	2.0327480	-0.7993536	-0.3021486
x2	0.21906639	-0.09762105	-0.05138317	-0.4795627	-1.3748826	0.7323587
x3	0.08369285	-2.64119673	-0.34019450	0.2493870	0.6347806	-1.2085960
x4	-1.28554439	0.61037038	-0.89764463	0.4026826	1.7762794	0.8684848
x5	-0.38378371	1.07394894	-0.73674199	-1.3240795	-0.2697071	-1.8311119
x6	-0.83693026	0.21106121	2.47591880	0.1896399	0.1476687	-0.2958822
x7	2.86346220	0.63690241	0.48320469	-0.4465221	1.4975360	0.2144778
x8	-0.15394237	-0.52485592	-0.23761538	-0.7752912	-0.3939105	1.2342381
	Dim7					
x1	0.53731697					
x2	-1.78938738					
x3	-0.32626826					
x4	-0.72333444					
x5	0.28708482					
x6	0.08839279					
x7	-0.08239776					
x8	1.59105249					

Retenemos solo las que nos interesan ...

Coordenadas de columna 1 & 2 =====:

	Dim1	Dim2
hatco	-0.4703327	0.6261240
A	-1.0217346	0.5801388
B	-0.8432000	-1.5832948
C	1.9333842	-0.7935744
D	0.9699347	-1.1891134
E	-0.4498224	0.5024514
F	-0.8381141	0.4468266
G	1.6806871	1.0920696
H	-0.3922857	-1.9443407
I	0.2334232	0.7842522

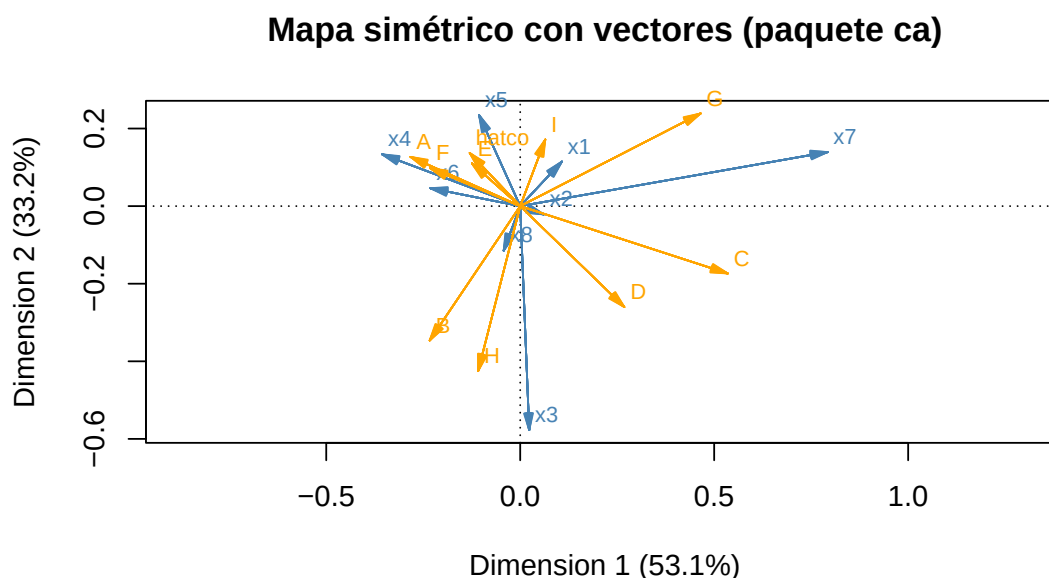
Coordenadas de fila 1 & 2 =====:

	Dim1	Dim2
x1	0.38833337	0.52453784
x2	0.21906639	-0.09762105
x3	0.08369285	-2.64119673
x4	-1.28554439	0.61037038
x5	-0.38378371	1.07394894
x6	-0.83693026	0.21106121
x7	2.86346220	0.63690241
x8	-0.15394237	-0.52485592

Conocidas las coordenadas, se representa el mapa visual.

```
# Generate the base plot using the ca package, with arrows for
↔ rows and columns
plot(ca_greenacre,
     main = "Mapa simétrico con vectores (paquete ca)",
     map = "symmetric",
     col = c("steelblue", "orange"),
     col.lab = c("steelblue", "orange"),
     arrows = c(TRUE, TRUE) # TRUE for rows, TRUE for columns
```

)



Interpretación estratégica:

Al observar el mapa, el punto **IDEAL** suele situarse cerca del origen. Esto ocurre porque un perfil que es “bueno en todo” no presenta sesgo hacia ninguna dimensión específica: no es especialmente “barato” ni especialmente “rápido” en comparación con el resto, sino que es máximo en todos los atributos. Para HATCO, la clave no es estar necesariamente cerca del centro (donde el IDEAL se ubica por equilibrio), sino verificar que el ángulo formado entre el vector de **HATCO** y el del **IDEAL** sea lo más pequeño posible en las dimensiones que el mercado más valora.

4.8. Decálogo de interpretación del análisis de correspondencias

Al interpretar el mapa cartesiano, se aplican estas reglas fundamentales:

1. **El origen (0,0):** representa el “perfil promedio” del mercado. Los puntos cercanos al centro son poco discriminatorios (no destacan especialmente en nada o son percibidos como la media del sector).
2. **Distancia al origen:** cuanto más lejos está una marca o un atributo del centro, más fuerte es su diferenciación y más contribuye a la inercia del modelo.

3. **Proximidad intra-grupo:** la proximidad física entre dos marcas (puntos del mismo color) indica que los consumidores las perciben como similares y, por tanto, como competidores directos.
4. **Ángulos inter-grupo (Regla de Oro):** para relacionar una marca con un atributo, no basta con mirar la distancia física; se observa el **ángulo** que forman al trazar líneas desde el origen (0,0):
 - **Ángulo agudo ($< 90^\circ$):** fuerte asociación positiva (ej. HATCO y la empresa A están muy asociadas a los atributos x4 y x6).
 - **Ángulo recto (90°):** independencia. La marca no destaca ni a favor ni en contra de ese atributo.
 - **Ángulo obtuso/llano ($> 90^\circ$):** oposición o asociación negativa. La marca se define por *carecer* de ese atributo (ej. la empresa G está diametralmente opuesta al atributo x3).
5. **Calidad de representación (qlt):** antes de interpretar la posición de un punto, conviene comprobar su calidad. Un punto mal representado en el plano bidimensional puede situarse en un lugar engañoso. Como norma general, no deben extraerse conclusiones sobre puntos con $qlt < 100$.
6. **Contribución a los ejes (ctr):** los puntos con mayor **ctr** en un eje son los que “dan forma” a ese eje y permiten nombrarlo. Identificarlos es el primer paso para interpretar qué concepto subyace en cada dimensión.
7. **Los ejes no tienen escala única:** las coordenadas de filas y columnas no están en la misma unidad. La distancia *entre una marca y un atributo* solo puede interpretarse con la regla del ángulo, no midiendo píxeles en el mapa.
8. **Simetría del mapa:** el mapa simétrico (`map = "symmetric"`) es el estándar en investigación de mercados porque coloca filas y columnas en el mismo espacio visual. Es el más intuitivo para el cliente, aunque no permite comparar distancias euclídeas entre filas y columnas directamente.
9. **Número de dimensiones:** la decisión sobre cuántos ejes retener debe basarse en el porcentaje de inercia acumulado. Un mapa de dos dimensiones que explica más del 80 % de la inercia es una representación fiel; por debajo del 60 %, conviene reportar con cautela.
10. **El mapa es un resumen, no la verdad absoluta:** el análisis de correspondencias *reduce* la realidad multidimensional a un plano. Siempre puede existir información relevante en las dimensiones no representadas. El mapa debe complementarse con la lectura de los estadísticos numéricos (`summary()`), especialmente el **qlt** y el **ctr**.

4.9. Diagnóstico estratégico para HATCO

A partir del mapa obtenido, pueden identificarse los siguientes patrones:

- **El eje del prestigio (X-Dimensión 1 — derecha):** HATCO, A, E y F dominan el cuadrante asociado a la imagen del fabricante (x4), calidad de servicio (x5) e imagen de vendedores (x6). Son las marcas *premium*. HATCO debe proteger esta reputación como ventaja competitiva sostenible.
- **El eje de eficiencia de producto (X-Dimensión 1 — izquierda):** la empresa G domina en solitario la calidad pura del producto (x7), distanciándose de la “imagen comercial”. Representa un posicionamiento técnico claramente diferenciado.
- **El eje del precio (Y-Dimensión 2 — abajo):** las empresas B y H compiten puramente en flexibilidad de precios (x3), siendo las opciones “negociadoras” o económicas del mercado.
- **Oportunidades de mercado:** los cuadrantes con escasa presencia de marcas pero con atributos relevantes señalan huecos estratégicos que HATCO podría evaluar para una extensión de su propuesta de valor.

4.10. Puntos suplementarios

En la investigación estratégica, a menudo resulta necesario proyectar perfiles teóricos (como el “servicio ideal” o un “competidor de referencia”) sin que estos influyan en la construcción de los ejes. En el paquete `ca`, esto se gestiona mediante los argumentos `supcol` (para columnas suplementarias) o `suprow` (para filas suplementarias).

A continuación se proyecta un “Competidor IDEAL” para visualizar la distancia entre la percepción actual de HATCO y la perfección teórica. El IDEAL se define como un perfil de columna con el valor máximo posible en todos los atributos (15 menciones en cada atributo), de modo que no presenta ningún sesgo hacia ninguna dimensión particular.

```
# Add a supplementary column 'IDEAL' with equal mentions across
↪ all attributes
hatco_extended <- as.table(matrix(
  c(
    16, 13, 8, 13, 9, 17, 15, 16, 6, 12, 15, #x1
```

```

14, 14, 10, 11, 11, 14, 12, 13, 10, 14, 15, #x2
6, 6, 14, 10, 11, 8, 7, 4, 14, 4, 15, #x3
15, 18, 9, 2, 3, 15, 16, 7, 8, 8, 15, #x4
15, 14, 6, 4, 4, 15, 14, 13, 7, 13, 15, #x5
7, 18, 13, 4, 9, 16, 14, 5, 4, 16, 15, #x6
4, 3, 1, 13, 9, 6, 3, 18, 2, 10, 15, #x7
15, 16, 15, 11, 11, 14, 16, 12, 14, 14, 15 #x8
),
nrow = 8,
byrow = TRUE,
dimnames = list(
  c("x1", "x2", "x3", "x4", "x5", "x6", "x7", "x8"),
  c("hatco", "A", "B", "C", "D", "E", "F", "G", "H", "I",
    ↪ "IDEAL")
)
)
)
hatco_extended

```

	hatco	A	B	C	D	E	F	G	H	I	IDEAL
x1	16	13	8	13	9	17	15	16	6	12	15
x2	14	14	10	11	11	14	12	13	10	14	15
x3	6	6	14	10	11	8	7	4	14	4	15
x4	15	18	9	2	3	15	16	7	8	8	15
x5	15	14	6	4	4	15	14	13	7	13	15
x6	7	18	13	4	9	16	14	5	4	16	15
x7	4	3	1	13	9	6	3	18	2	10	15
x8	15	16	15	11	11	14	16	12	14	14	15

```

# Estimate the model identifying the 11th columnn as
↪ supplementary
# The algorithm ignores this column for inertia calculation but
↪ projects it afterwards
ca_sup <- ca::ca(hatco_extended, supcol = 11)

# Summary including supplementary points

```

```
# 'IDEAL' appears in a separate section of the output
summary(ca_sup)
```

Principal inertias (eigenvalues):

dim	value	%	cum%	scree plot
1	0.076541	53.1	53.1	*****
2	0.047813	33.2	86.3	*****
3	0.015291	10.6	96.9	***
4	0.002658	1.8	98.8	
5	0.000806	0.6	99.3	
6	0.000576	0.4	99.7	
7	0.000381	0.3	100.0	

Total:		0.144066	100.0	

Rows:

	name	mass	qlt	inr	k=1	cor	ctr	k=2	cor	ctr
1	x1	147	619	41	107	289	22	115	330	40
2	x2	145	527	8	61	469	7	-21	58	1
3	x3	99	991	231	23	2	1	-578	989	689
4	x4	119	901	132	-356	789	196	133	111	44
5	x5	123	816	70	-106	138	18	235	677	142
6	x6	125	372	129	-232	358	87	46	14	6
7	x7	81	991	367	792	961	665	139	30	33
8	x8	162	772	22	-43	93	4	-115	678	45

Columns:

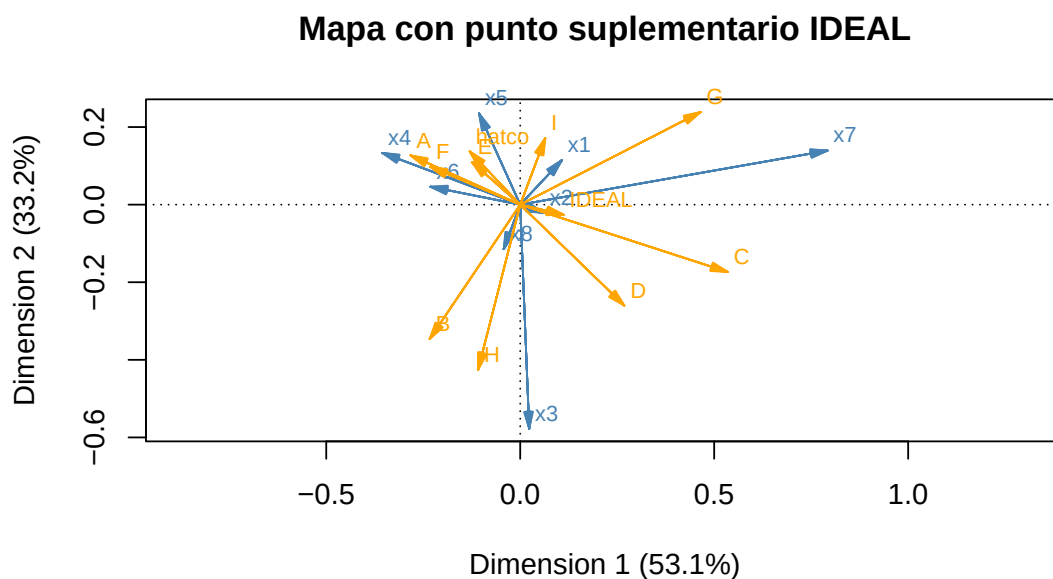
	name	mass	qlt	inr	k=1	cor	ctr	k=2	cor	ctr
1	hatc	108	433	62	-130	206	24	137	228	42
2	A	120	928	86	-283	772	125	127	156	40
3	B	89	942	115	-233	294	63	-346	648	224
4	C	80	975	180	535	882	299	-174	93	50
5	D	79	863	88	268	445	74	-260	418	111

6		E		123	812	29		-124	456	25		110	356	31	
7		F		114	954	52		-232	810	80		98	144	23	
8		G		103	963	204		465	762	292		239	201	123	
9		H		76	797	128		-109	49	12		-425	748	289	
10		I		107	446	56		65	55	6		171	390	66	
11		(*)IDEA		<NA>	289	<NA>		112	274	<NA>		-26	15	<NA>	

4.10.1. Visualización del posicionamiento IDEAL

Al graficar, el paquete `ca` diferencia automáticamente los puntos activos de los suplementarios (generalmente mediante símbolos o colores distintos) para que el analista no olvide que ese punto no ha sido considerado en el cómputo del mapa, sino que solo ha sido “invitado” a la representación”.

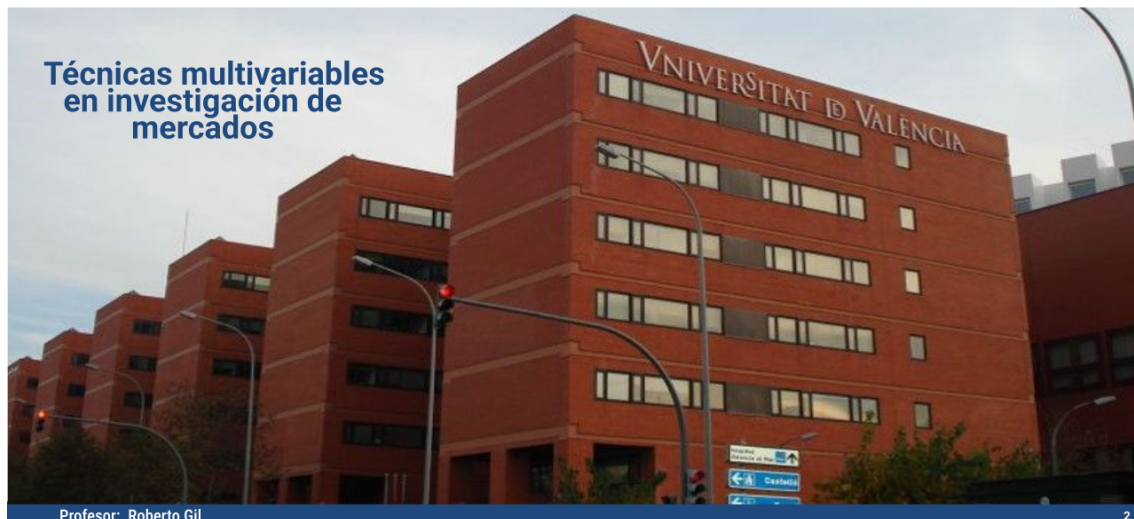
```
# Plot with supplementary point
plot(ca_sup,
      main = "Mapa con punto suplementario IDEAL",
      col    = c("steelblue", "orange", "seagreen"),
      col.lab = c("steelblue", "orange"),
      map     = "symmetric",
      arrows  = c(TRUE, TRUE)
)
```



i En resumen

El análisis de correspondencias no es un fin en sí mismo, sino una herramienta de diálogo con el cliente. El mapa traduce la complejidad de una tabla de contingencia en un lenguaje visual inmediato, permitiendo en segundos a la pregunta más importante de cualquier estrategia competitiva: **¿quién es esta empresa para sus clientes y quiénes son sus verdaderos rivales?**

5 Análisis de Componentes Principales (PCA)



Profesor: Roberto Gil

2

Contenidos del tema / Índice



- **Análisis de componentes principales - ACP**
 - ▶ Introducción
 - ▶ Planteamiento del problema a resolver
 - ▶ La práctica del análisis de componentes principales
 - ▶ Solución en SPSS / R

Bibliografía recomendada

- Hair, J.F., Anderson, R.E., Tatham, R.L. Y Black, W. (1999). Análisis multivariante. 4ª edición. Madrid: Prentice Hall.
- Uriel, E. y Aldás, J. (2005). Análisis multivariante aplicado. Madrid: Thomson.
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). "An introduction to statistical learning" (Vol. 112). New York, NY: Springer.
- Chapman, C., & Feit, E. M. (2015). R for marketing research and analytics. New York, NY: Springer.

5.1. Introducción

Introducción



Profesor/a: Roberto Gil

3

El análisis de componentes principales (PCA) se sitúa dentro de las técnicas multivariantes de **interdependencia**, lo que significa que no se distingue entre variables dependientes e independientes; el objetivo es entender la estructura de relaciones de todo el conjunto de datos simultáneamente.

En el ámbito de la investigación de mercados, es habitual enfrentarse a cuestionarios con decenas de atributos (valoraciones de marca, hábitos de consumo o estilos de vida). Trabajar con tantas variables de forma aislada dificulta la toma de decisiones. El PCA actúa como una herramienta de **reducción de la dimensionalidad**, y permite:

1. **Simplificar la complejidad:** se transforma un conjunto amplio de variables métricas, habitualmente correlacionadas entre sí, en un número menor de variables ficticias llamadas **componentes**.
2. **Garantizar la independencia:** a diferencia de las variables originales, los componentes resultantes son ortogonales (no están correlacionados), lo que elimina la redundancia de información.
3. **Identificar dimensiones latentes:** el PCA ayuda a descubrir conceptos no observados directamente. Por ejemplo, a partir de variables como “limpieza”, “amabilidad del personal” y “rapidez de respuesta”, el PCA puede revelar una dimensión única de “Calidad de Servicio”.

4. **Optimizar análisis posteriores:** estas nuevas dimensiones (puntuaciones factoriales o *scores*) sirven como *input* perfecto para técnicas de segmentación (*cluster analysis*) o modelos de regresión, evitando problemas de multicolinealidad.

5.1.1. Un ejemplo real: la escala E-S-QUAL

Para entender cómo el PCA se aplica en la vanguardia de la investigación de mercados, resulta útil observar el proceso de creación de escalas de medida profesionales como la **E-S-QUAL**. En *marketing*, conceptos como la “calidad de servicio electrónico” no pueden medirse con una sola pregunta; son **fenómenos multidimensionales**.

Este flujo de trabajo, basado en Parasuraman et al. (2005), resume el camino que se recorrerá en esta sesión:

1. **Generación de ítems:** se parte de una batería amplia de preguntas (a veces más de 100) que intentan cubrir todos los ángulos del fenómeno.
2. **Refinamiento y purificación:** es aquí donde el PCA es el protagonista. Mediante el análisis de las correlaciones y la eliminación de variables que no aportan valor (bajo MSA o bajas comunalidades), se “limpia” la escala.
3. **Identificación de dimensiones:** el PCA permite agrupar los ítems finales en dimensiones críticas. En el caso de E-S-QUAL, lo que empezó como una masa de datos terminó definiendo cuatro pilares: Eficiencia, Cumplimiento, Disponibilidad del Sistema y Privacidad.
4. **Validación:** una vez reducida la dimensionalidad, las puntuaciones de estos componentes se convierten en las variables que realmente explican la satisfacción y lealtad del cliente.

Reflexión para el analista: el PCA no es solo un algoritmo matemático; es el filtro que permite separar el ruido de la encuesta de las señales estratégicas del mercado. Sin esta reducción, resulta imposible construir modelos de gestión accionables.

5.2. Caso de ejemplo: HBAT

```
# Load the HBAT dataset; expss::read_spss preserves variable and
↪ value labels from SPSS
hbat <- expss::read_spss("data/hbat.sav")

cat("Sample of working file:\n")
utils::head(hbat, 10)

# Select the metric perception variables for the analysis (x6 to
↪ x18)
# These 13 variables represent customer perceptions about the
↪ company
data_pca <- hbat %>%
  dplyr::select(x6, x7, x8, x9, x10, x11, x12, x13, x14, x15,
  ↪ x16, x17, x18)

cat("\nSample of working file with selected variables:\n")
utils::head(data_pca, 10)
```

Sample of working file:

	id	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	x17	x18	x19
1	1	2	0	1	1	1	8.5	3.9	2.5	5.9	4.8	4.9	6.0	6.8	4.7	4.3	5.0	5.1	3.7	8.2
2	2	3	1	0	0	0	8.2	2.7	5.1	7.2	3.4	7.9	3.1	5.3	5.5	4.0	3.9	4.3	4.9	5.7
3	3	3	0	1	1	1	9.2	3.4	5.6	5.6	5.4	7.4	5.8	4.5	6.2	4.6	5.4	4.0	4.5	8.9
4	4	1	1	1	1	0	6.4	3.3	7.0	3.7	4.7	4.7	4.5	8.8	7.0	3.6	4.3	4.1	3.0	4.8
5	5	2	0	1	0	1	9.0	3.4	5.2	4.6	2.2	6.0	4.5	6.8	6.1	4.5	4.5	3.5	3.5	7.1
6	6	1	1	0	1	0	6.5	2.8	3.1	4.1	4.0	4.3	3.7	8.5	5.1	9.5	3.6	4.7	3.3	4.7
7	7	1	1	1	1	0	6.9	3.7	5.0	2.6	2.1	2.3	5.4	8.9	4.8	2.5	2.1	4.2	2.0	5.7
8	8	2	0	1	1	0	6.2	3.3	3.9	4.8	4.6	3.6	5.1	6.9	5.4	4.8	4.3	6.3	3.7	6.3
9	9	2	1	1	1	0	5.8	3.6	5.1	6.7	3.7	5.9	5.8	9.3	5.9	4.4	4.4	6.1	4.6	7.0
10	10	1	0	1	1	0	6.4	4.5	5.1	6.1	4.7	5.7	5.7	8.4	5.4	5.3	4.1	5.8	4.4	5.5
		x20	x21	x22	x23															
1		8.0	8.4	65.1		1														
2		6.5	7.5	67.1		0														
3		8.4	9.0	72.1		1														

```

4  6.0 7.2 40.1  0
5  6.6 9.0 57.1  0
6  6.3 6.1 50.1  0
7  7.8 7.2 41.1  0
8  5.8 7.7 56.1  0
9  7.5 8.2 56.1  1
10 5.9 6.7 59.1  0

```

Sample of working file with selected variables:

```

      x6  x7  x8  x9 x10 x11 x12 x13 x14 x15 x16 x17 x18
1  8.5 3.9 2.5 5.9 4.8 4.9 6.0 6.8 4.7 4.3 5.0 5.1 3.7
2  8.2 2.7 5.1 7.2 3.4 7.9 3.1 5.3 5.5 4.0 3.9 4.3 4.9
3  9.2 3.4 5.6 5.6 5.4 7.4 5.8 4.5 6.2 4.6 5.4 4.0 4.5
4  6.4 3.3 7.0 3.7 4.7 4.7 4.5 8.8 7.0 3.6 4.3 4.1 3.0
5  9.0 3.4 5.2 4.6 2.2 6.0 4.5 6.8 6.1 4.5 4.5 3.5 3.5
6  6.5 2.8 3.1 4.1 4.0 4.3 3.7 8.5 5.1 9.5 3.6 4.7 3.3
7  6.9 3.7 5.0 2.6 2.1 2.3 5.4 8.9 4.8 2.5 2.1 4.2 2.0
8  6.2 3.3 3.9 4.8 4.6 3.6 5.1 6.9 5.4 4.8 4.3 6.3 3.7
9  5.8 3.6 5.1 6.7 3.7 5.9 5.8 9.3 5.9 4.4 4.4 6.1 4.6
10 6.4 4.5 5.1 6.1 4.7 5.7 5.7 8.4 5.4 5.3 4.1 5.8 4.4

```

5.2.1. La adecuación de los datos a la técnica

Antes de proceder con el análisis, es necesario verificar dos condiciones previas: que el banco de datos cumple los requisitos mínimos de tamaño y que las variables muestran un grado suficiente de correlación entre sí. Si no se satisfacen estas condiciones, el PCA no conseguirá reducir la dimensionalidad de forma eficiente.

La primera verificación es la **ratio observaciones/variables**: se recomienda disponer de al menos 5 observaciones por variable, y preferiblemente 10. La segunda consiste en inspeccionar la **matriz de correlaciones de Pearson** para comprobar si existen suficientes coeficientes que superen el umbral de 0.30 (Hair et al., 2006).

5.2.2. Preparación del entorno y datos

Se trabaja con el *dataset* `hbat`, que contiene 13 variables de percepción (x6 a x18) que se intentarán reducir a dimensiones de gestión.

```

# Display data structure dimensions
cat("Data structure dimensions:\n")
dim(data_pca)

# Calculate the observations-to-variables ratio
cat("\nObservations / variables ratio:\n")
dim(data_pca)[1] / dim(data_pca)[2]

# Preliminary inspection: Pearson correlation matrix
# If there are no significant correlations (> 0.30), PCA is not
  ↪ justifiable
cat("\nPearson correlation matrix:\n")
round(stats::cor(data_pca), 3)

```

Data structure dimensions:

```
[1] 100 13
```

Observations / variables ratio:

```
[1] 7.692308
```

Pearson correlation matrix:

	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15
x6	1.000	-0.137	0.096	0.106	-0.053	0.477	-0.152	-0.401	0.088	0.027
x7	-0.137	1.000	0.001	0.140	0.430	-0.053	0.792	0.229	0.052	-0.027
x8	0.096	0.001	1.000	0.097	-0.063	0.193	0.017	-0.271	0.797	-0.074
x9	0.106	0.140	0.097	1.000	0.197	0.561	0.230	-0.128	0.140	0.059
x10	-0.053	0.430	-0.063	0.197	1.000	-0.012	0.542	0.134	0.011	0.084
x11	0.477	-0.053	0.193	0.561	-0.012	1.000	-0.061	-0.495	0.273	0.046
x12	-0.152	0.792	0.017	0.230	0.542	-0.061	1.000	0.265	0.107	0.032
x13	-0.401	0.229	-0.271	-0.128	0.134	-0.495	0.265	1.000	-0.245	0.023
x14	0.088	0.052	0.797	0.140	0.011	0.273	0.107	-0.245	1.000	0.035
x15	0.027	-0.027	-0.074	0.059	0.084	0.046	0.032	0.023	0.035	1.000
x16	0.104	0.156	0.080	0.757	0.184	0.424	0.195	-0.115	0.197	0.069
x17	-0.493	0.271	-0.186	0.395	0.334	-0.378	0.352	0.471	-0.170	0.094
x18	0.028	0.192	0.025	0.865	0.276	0.602	0.272	-0.073	0.109	0.106
	x16	x17	x18							

x6	0.104	-0.493	0.028
x7	0.156	0.271	0.192
x8	0.080	-0.186	0.025
x9	0.757	0.395	0.865
x10	0.184	0.334	0.276
x11	0.424	-0.378	0.602
x12	0.195	0.352	0.272
x13	-0.115	0.471	-0.073
x14	0.197	-0.170	0.109
x15	0.069	0.094	0.106
x16	1.000	0.407	0.751
x17	0.407	1.000	0.497
x18	0.751	0.497	1.000

5.2.2.1. Diagnóstico de la matriz de correlaciones

En una matriz de $k \times k$ variables, el número de correlaciones únicas (la diagonal superior o inferior, sin contar la diagonal principal de 1s) viene dado por la fórmula:

$$\frac{k^2 - k}{2}$$

Para revelar cuántas de estas son significativas (por ejemplo, con un $|r| > 0.30$ según los criterios de Hair), se puede usar una lógica de filtrado sobre la matriz. Si el número de correlaciones significativas es bajo, el PCA no conseguirá reducir la información de forma eficiente.

```
# Extract only the upper triangle to avoid counting each pair
↪ twice
unique_cors <-
↪ stats::cor(data_pca)[upper.tri(stats::cor(data_pca))]

cat("\nNumber of unique correlation coefficients:\n")
total_unique <- length(unique_cors)
print(total_unique)

cat("\nNumber of significant coefficients (|r| >= 0.30):\n")
```

```
significant <- sum(abs(unique_cors) >= 0.30)
print(significant)

cat("\n% of significant coefficients (|r| >= 0.30):\n")
pct_significant <- round((significant / total_unique) * 100, 0)
print(pct_significant)
```

Number of unique correlation coefficients:

```
[1] 78
```

Number of significant coefficients ($|r| \geq 0.30$):

```
[1] 21
```

% of significant coefficients ($|r| \geq 0.30$):

```
[1] 27
```

Tras ejecutar el conteo, se observa que un número sustancial de correlaciones únicas supera el umbral de 0.30. Aunque no es una matriz extremadamente densa, este ~30 % justifica proceder con el análisis. Sin embargo, este dato “tibio” ya anticipa que, probablemente, algunas variables no encajarán bien en la estructura común y deberán ser eliminadas en el proceso de **depuración de la escala**.

En investigación de mercados, bajo los criterios de Hair et al., un 30 % de correlaciones significativas (con $|r| > 0.30$) suele ser el umbral mínimo aceptable para que el PCA tenga sentido. Si fuera menor, se estarían intentando resumir variables que no tienen nada en común, lo que daría lugar a una solución factorial muy pobre donde cada variable “va por libre”.

5.3. Adecuación a los requisitos del análisis

Una vez que la inspección visual sugiere que existe relación entre las variables, debe formalizarse estadísticamente si el banco de datos es apto para un PCA. No basta con que existan correlaciones significativas; estas deben ser lo suficientemente fuertes y el conjunto de datos lo suficientemente consistente. Este proceso se apoya en dos indicadores clásicos que actúan como “llave de paso” antes de proceder a la extracción de componentes:

5.3.1. El índice de Kaiser-Meyer-Olkin (KMO)

El KMO es una medida de la **adecuación muestral**. Compara los coeficientes de correlación observados con los coeficientes de correlación parcial. Si las correlaciones parciales son pequeñas en comparación con las correlaciones originales, el índice será alto, lo cual es la señal idónea.

Siguiendo la clasificación de Kaiser y Rice (1974), el valor obtenido se interpreta así:

- **KMO > 0.90:** Maravilloso.
- **0.80 - 0.90:** Meritorio.
- **0.70 - 0.80:** Medio.
- **0.60 - 0.70:** Mediocre (aceptado como límite inferior en investigación de mercados).
- **< 0.50:** Inaceptable. En este caso el análisis no debe continuar sin antes depurar las variables.

5.3.2. La prueba de esfericidad de Bartlett

Esta prueba evalúa la aplicabilidad del análisis desde una perspectiva global de la matriz. Contrasta la **hipótesis nula** (H_0) de que la matriz de correlaciones es, en realidad, una **matriz identidad** (una matriz donde no hay correlación entre ninguna variable, solo de cada variable consigo misma).

- **Si el p-value ≤ 0.05 :** se rechaza la H_0 . Existe evidencia de que hay una estructura de correlación significativa y puede aplicarse el PCA con garantías.
- **Si el p-value > 0.05 :** no puede asegurarse que las variables estén relacionadas. En este escenario, el PCA no conseguirá reducir la dimensionalidad de forma efectiva porque no hay “patrones comunes” que agrupar.

El KMO informa sobre la **calidad de la relación** entre las variables para ser reducidas, mientras que Bartlett confirma la **existencia** de dicha relación.

```
# KMO: measures the proportion of variance that is common among
↪ variables
# The diagonal of the output shows individual MSA values per
↪ variable
```

```

cat("KMO (Kaiser-Meyer-Olkin) and individual MSA values:\n")
psych::KMO(data_pca)

# Bartlett's test of sphericity:
# H0 = the correlation matrix is an identity matrix (no
  ↪ correlations at all)
# A p-value < 0.05 is required to proceed with PCA
cat("\nBartlett's test of sphericity:\n")
psych::cortest.bartlett(stats::cor(data_pca), n =
  ↪ nrow(data_pca))

```

KMO (Kaiser-Meyer-Olkin) and individual MSA values:

Kaiser-Meyer-Olkin factor adequacy

Call: psych::KMO(r = data_pca)

Overall MSA = 0.61

MSA for each item =

	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	x17	x18
	0.87	0.62	0.53	0.89	0.81	0.45	0.59	0.88	0.53	0.31	0.86	0.44	0.53

Bartlett's test of sphericity:

\$chisq

[1] 948.9831

\$p.value

[1] 8.81589e-150

\$df

[1] 78

5.3.3. Proceso iterativo de purificación (MSA individual)

Siguiendo el flujo de trabajo de “purificación” del esquema de Parasuraman, no basta con el KMO global. Es necesario revisar los **MSA individuales** (la diagonal de la matriz KMO). Aquellas variables que no alcancen un mínimo (habitualmente 0.50) están “ensuciando” el modelo y deben eliminarse una a una, en orden de MSA más bajo a más alto, recalculando el KMO tras cada eliminación.

Se observa que **x15** tiene el MSA más bajo (por debajo de 0.50). Se procede a eliminarla y se recalcula:

```
# Remove x15 - lowest individual MSA value, below the 0.50
↪ threshold
data_pca <- data_pca %>%
  dplyr::select(-x15)

# Recalculate KMO to check whether the overall adequacy has
↪ improved
cat("KMO after removing x15:\n")
psych::KMO(data_pca)
```

KMO after removing x15:

Kaiser-Meyer-Olkin factor adequacy

Call: psych::KMO(r = data_pca)

Overall MSA = 0.61

MSA for each item =

	x6	x7	x8	x9	x10	x11	x12	x13	x14	x16	x17	x18
	0.88	0.62	0.53	0.89	0.80	0.45	0.59	0.88	0.53	0.86	0.44	0.53

Tras esta primera eliminación, se evidencia que x17 tiene el MSA más bajo entre las variables restantes (aún por debajo de 0.50). Se reitera el proceso:

```
# Remove x17 - now the lowest MSA in the remaining set
data_pca <- data_pca %>%
  dplyr::select(-x17)

# Recalculate KMO
cat("KMO after removing x17:\n")
psych::KMO(data_pca)
```

KMO after removing x17:

Kaiser-Meyer-Olkin factor adequacy

Call: psych::KMO(r = data_pca)

Overall MSA = 0.65

MSA for each item =

x6	x7	x8	x9	x10	x11	x12	x13	x14	x16	x18
0.51	0.63	0.52	0.79	0.78	0.62	0.62	0.75	0.51	0.76	0.67

Nótese —y este es el motivo por el que se elimina de una en una— que x11 se ha recuperado. No obstante, hay que vigilar esta variable, que emite señales de *incoherencia* con el resto del conjunto de datos.

5.4. Determinación del número de componentes

Una de las decisiones más críticas en el PCA es determinar cuántos componentes deben retenerse. No se trata de una decisión arbitraria, sino que debe estar respaldada por criterios estadísticos que equilibren la **parsimonia** (retener pocos componentes) y la **capacidad explicativa** (no perder demasiada información).

Para tomar esta decisión se recurre a tres criterios fundamentales:

5.4.1. Criterio de los autovalores (*eigenvalues*) > 1

El autovalor asociado a un componente principal mide la cantidad total de varianza que dicho componente es capaz de recoger. Dado que en el PCA se trabaja con variables estandarizadas (donde cada variable original tiene una varianza de 1), solo tiene sentido retener aquellos componentes que expliquen, al menos, **la varianza de una sola variable**.

Lógica: si un componente tiene un autovalor menor que 1, aporta menos información que una variable original aislada y, por tanto, no cumple su función de síntesis.

5.4.2. Porcentaje de la varianza explicada

Aunque el objetivo principal sea la reducción de datos, este proceso conlleva inevitablemente una pérdida de información (varianza).

Umbral recomendado: en el ámbito del *marketing* y la investigación de mercados, se suele aceptar una solución que recoja **al menos el 60 % de la varianza original** (Hair et al., 2006). Ganar interpretabilidad no debe costar una pérdida excesiva de precisión.

5.4.3. Gráfico de sedimentación (*scree plot*)

Es la representación visual de los autovalores de cada componente ordenados de mayor a menor. Al analizar esta gráfica, se busca la **regla del codo** (*elbow rule*):

Interpretación: la extracción debe detenerse en el punto en que la curva se aplana significativamente. A partir de ese “codo”, añadir nuevos componentes no incrementa de forma relevante la varianza explicada, y la varianza de los factores únicos empieza a dominar sobre la de los factores comunes. Se retiene el número de componentes anterior al codo.

Determinar el número de componentes es un ejercicio de compromiso. Un modelo con demasiados componentes no reduce la complejidad, mientras que uno con muy pocos puede estar ignorando dimensiones estratégicas del mercado vitales para la toma de decisiones.

5.5. Extracción y rotación de componentes

Se utiliza el método de **Componentes Principales** con rotación **Varimax**. La rotación ortogonal Varimax simplifica la interpretación al minimizar el número de variables con cargas altas en un solo factor.

```
# Compute eigenvalues from the correlation matrix of the
↪ purified dataset
# (x15 and x17 have been removed at this point)
eigen_values <- eigen(stats::cor(data_pca))$values

# Retain components with eigenvalue >= 1 (Kaiser's rule)
# This replicates the default extraction behaviour of SPSS
n_factors_auto <- sum(eigen_values >= 1)
```

```

cat("Components retained (eigenvalue >= 1):", n_factors_auto,
    ↪  "\n\n")

# Run PCA with Varimax rotation
pca_final <- psych::principal(data_pca, nfactors =
    ↪  n_factors_auto, rotate = "varimax")
print(pca_final)

```

Components retained (eigenvalue >= 1): 4

Principal Components Analysis

Call: psych::principal(r = data_pca, nfactors = n_factors_auto, rotate = "varimax")

Standardized loadings (pattern matrix) based upon correlation matrix

	RC1	RC2	RC3	RC4	h2	u2	com
x6	0.00	-0.01	-0.03	0.88	0.77	0.232	1.0
x7	0.06	0.87	0.05	-0.12	0.78	0.223	1.1
x8	0.02	-0.02	0.94	0.10	0.89	0.107	1.0
x9	0.93	0.12	0.05	0.09	0.88	0.119	1.1
x10	0.14	0.74	-0.08	0.01	0.58	0.424	1.1
x11	0.59	-0.06	0.15	0.64	0.79	0.213	2.1
x12	0.13	0.90	0.08	-0.16	0.86	0.141	1.1
x13	-0.09	0.23	-0.25	-0.72	0.64	0.359	1.5
x14	0.11	0.05	0.93	0.10	0.89	0.108	1.1
x16	0.86	0.11	0.08	0.04	0.77	0.234	1.1
x18	0.94	0.18	0.00	0.05	0.91	0.086	1.1

	RC1	RC2	RC3	RC4
SS loadings	2.89	2.23	1.86	1.77
Proportion Var	0.26	0.20	0.17	0.16
Cumulative Var	0.26	0.47	0.63	0.80
Proportion Explained	0.33	0.26	0.21	0.20
Cumulative Proportion	0.33	0.59	0.80	1.00

Mean item complexity = 1.2

Test of the hypothesis that 4 components are sufficient.

The root mean square of the residuals (RMSR) is 0.04
with the empirical chi square 19.51 with prob < 0.3

Fit based upon off diagonal values = 0.98

Al revisar la salida, se observa que **x11** presenta cargas cruzadas en los componentes RC1 y RC4. El indicador **com** (Complejidad de Hoffman), que se detalla en la siguiente sección, confirma el problema técnico: esta variable no carga de forma limpia en un solo componente y debe eliminarse para mejorar la interpretabilidad del modelo.

```
# Remove x11 due to cross-loadings (confirmed by high 'com'
↪ complexity index)
data_pca <- data_pca %>%
  dplyr::select(-x11)

# Recompute eigenvalues on the updated dataset before re-running
↪ PCA
# (the eigenvalue structure changes after removing a variable)
eigen_values <- eigen(stats::cor(data_pca))$values
n_factors_auto <- sum(eigen_values >= 1)
cat("Components retained after removing x11 (eigenvalue >= 1):",
    ↪ n_factors_auto, "\n\n")

# Re-run PCA with the purified 10-variable dataset
pca_final <- psych::principal(data_pca, nfactors =
    ↪ n_factors_auto, rotate = "varimax")
print(pca_final)
```

Components retained after removing x11 (eigenvalue >= 1): 4

Principal Components Analysis

Call: psych::principal(r = data_pca, nfactors = n_factors_auto, rotate = "varimax")

Standardized loadings (pattern matrix) based upon correlation matrix

	RC1	RC2	RC3	RC4	h2	u2	com
x6	0.03	-0.01	-0.02	0.89	0.80	0.20	1.0
x7	0.06	0.87	0.05	-0.14	0.78	0.22	1.1

```

x8    0.02 -0.02  0.94  0.10  0.89  0.11  1.0
x9    0.93  0.10  0.06  0.08  0.89  0.11  1.0
x10   0.16  0.74 -0.08  0.04  0.58  0.42  1.1
x12   0.14  0.90  0.08 -0.17  0.86  0.14  1.1
x13  -0.10  0.23 -0.26 -0.73  0.66  0.34  1.5
x14   0.10  0.05  0.93  0.08  0.89  0.11  1.0
x16   0.89  0.10  0.09  0.07  0.81  0.19  1.1
x18   0.93  0.17  0.00  0.01  0.89  0.11  1.1

```

```

                RC1  RC2  RC3  RC4
SS loadings      2.59 2.22 1.85 1.41
Proportion Var   0.26 0.22 0.18 0.14
Cumulative Var    0.26 0.48 0.67 0.81
Proportion Explained 0.32 0.28 0.23 0.17
Cumulative Proportion 0.32 0.60 0.83 1.00

```

Mean item complexity = 1.1

Test of the hypothesis that 4 components are sufficient.

The root mean square of the residuals (RMSR) is 0.04
 with the empirical chi square 17.86 with prob < 0.085

Fit based upon off diagonal values = 0.98

5.5.1. Interpretación de la salida de `psych::principal()` en R

Al ejecutar el análisis en R, se obtiene una matriz que combina las cargas factoriales con indicadores de calidad del ajuste, todo integrado en una sola vista.

5.5.1.1. Las nuevas dimensiones: RC1, RC2, RC3...

- **¿Qué significan?:** son los **Componentes Rotados** (del inglés *Rotated Components*). El número que los acompaña indica el orden de extracción.
- **Equivalencia en SPSS:** corresponden a las columnas de la “Matriz de componentes rotados”.

- **Interpretación:** los valores que se observan son las **cargas factoriales** (correlación entre la variable original y el componente). Se buscan cargas > 0.50 para asignar una variable a un componente, siguiendo a Hair et al. (2006).

5.5.1.2. Los indicadores de calidad: h^2 , u^2 y com

Estas tres columnas finales conforman el diagnóstico técnico de cada variable en el modelo:

1. h^2 (Comunalidad):

- Es la proporción de la varianza de la variable que es explicada por los componentes retenidos.
- **Equivalencia en SPSS:** es la columna “**Extracción**” de la tabla de **Comunalidades**.
- **Umbral:** se desean valores superiores a **0.50**. Una h^2 baja indica que la variable no comparte información con las demás y “ensucia” el análisis.

2. u^2 (Unicidad):

- Es el inverso de la comunalidad $(1 - h^2)$.
- Representa la varianza específica de la variable que no ha podido ser capturada por los componentes comunes. Cuanto más alto sea u^2 , menos útil es esa variable para definir conceptos generales.

3. com (Complejidad de Hoffman):

- Este índice es una aportación de R que no aparece de forma directa en SPSS. Mide cuántos componentes “comparten” la carga de esa variable.
- **Interpretación:** un valor cercano a 1 indica que la variable carga de forma limpia en un solo componente; un valor cercano a 2 o superior indica una variable “conflictiva” que carga en varios componentes a la vez.
- **Utilidad:** es el indicador clave para detectar **cargas cruzadas**. Una complejidad alta es señal inequívoca de que la variable debe eliminarse para mejorar la interpretabilidad del modelo (como ocurrió con x11).

5.5.1.3. Comunalidades SPSS / unicidad R

SPSS muestra la comunalidad, inverso de la unicidad. R también la calcula y se puede observar su relación con este sencillo *script*.

```
# Communality and uniqueness are complementary: they always sum
↪ to 1
cat("Communality (h2):\n")
print(pca_final$communality)

cat("\nUniqueness (u2):\n")
print(pca_final$uniquenesses)

cat("\nVerification - communality + uniqueness = 1:\n")
print(pca_final$communality + pca_final$uniquenesses)
```

Communality (h2):

	x6	x7	x8	x9	x10	x12	x13	x14
	0.7979811	0.7796785	0.8937300	0.8895521	0.5848385	0.8595670	0.6609157	0.8907841
	x16	x18						
	0.8060437	0.8940676						

Uniqueness (u2):

	x6	x7	x8	x9	x10	x12	x13	x14
	0.2020189	0.2203215	0.1062700	0.1104479	0.4151615	0.1404330	0.3390843	0.1092159
	x16	x18						
	0.1939563	0.1059324						

Verification - communality + uniqueness = 1:

x6	x7	x8	x9	x10	x12	x13	x14	x16	x18
1	1	1	1	1	1	1	1	1	1

Aunque resulte improbable que falle en esta etapa, conviene testar de nuevo los indicadores básicos de adecuación sobre el conjunto de datos ya depurado, para disponer de documentación completa del proceso:

```
# Final dimensions of the purified dataset
cat("Final data structure dimensions:\n")
dim(data_pca)

cat("\nFinal observations / variables ratio:\n")
```

```

dim(data_pca)[1] / dim(data_pca)[2]

# Final KMO and individual MSA values
cat("\nFinal KMO and MSA:\n")
psych::KMO(data_pca)

# Final Bartlett's test
cat("\nFinal Bartlett's test of sphericity:\n")
psych::cortest.bartlett(stats::cor(data_pca), n =
  ↪ nrow(data_pca))

```

Final data structure dimensions:

```
[1] 100 10
```

Final observations / variables ratio:

```
[1] 10
```

Final KMO and MSA:

Kaiser-Meyer-Olkin factor adequacy

Call: psych::KMO(r = data_pca)

Overall MSA = 0.67

MSA for each item =

	x6	x7	x8	x9	x10	x12	x13	x14	x16	x18
	0.61	0.64	0.52	0.69	0.81	0.63	0.74	0.53	0.83	0.72

Final Bartlett's test of sphericity:

\$chisq

```
[1] 502.9739
```

\$p.value

```
[1] 1.130404e-78
```

\$df

```
[1] 45
```

Para obtener una vista más limpia de la matriz de cargas —replicando el comportamiento visual de SPSS—, se utiliza el argumento `cutoff` para suprimir cargas

pequeñas:

```
# Rotated component matrix - suppress loadings below 0.30 to  
↪ match SPSS output style  
print(pca_final$loadings, cutoff = 0.3, sort = TRUE)
```

Loadings:

	RC1	RC2	RC3	RC4
x9	0.933			
x16	0.886			
x18	0.931			
x7		0.868		
x10		0.743		
x12		0.898		
x8			0.940	
x14			0.933	
x6				0.892
x13				-0.730

	RC1	RC2	RC3	RC4
SS loadings	2.589	2.216	1.846	1.406
Proportion Var	0.259	0.222	0.185	0.141
Cumulative Var	0.259	0.481	0.665	0.806

5.5.2. Criterios de significación de las cargas factoriales

En la práctica habitual, se suelen aceptar cargas factoriales superiores a **0.50** como indicador de una relación sólida. Sin embargo, la evidencia científica (Hair et al., 2006; Stevens, 1992) indica que la “importancia” de una carga no es un valor absoluto, sino que depende directamente del rigor estadístico impuesto por el tamaño de la muestra (n).

5.5.2.1. Relación entre carga factorial y tamaño muestral

A medida que aumenta el tamaño de la muestra, el error estándar disminuye, lo que permite aceptar como significativas cargas factoriales más pequeñas:

- Para muestras pequeñas ($n = 50$), se necesitan cargas muy potentes (**0.75**) para asegurar que la relación no es fruto del azar.
- En el *dataset* actual (`hbat.sav`), con $n = 100$, el umbral de significación estadística se sitúa en **0.55**.
- Si se trabajara con muestras grandes ($n > 350$), una carga de **0.30** ya podría considerarse significativa.

5.5.2.2. Relación entre carga factorial y el número de variables

Stevens (1992) sugiere que el punto de corte debe ajustarse también en función del número de variables analizadas (p), para mantener un nivel de significación estable (generalmente $\alpha = 0.01$):

Variables (p)	Carga mínima ($n = 50$)	Carga mínima ($n = 100$)
10	0.722	0.512
20	0.754	0.534
30	0.780	0.552

Implicación estratégica: dado que la escala original tiene 13 variables y 100 casos, el criterio de Stevens sitúa el punto de corte ideal ligeramente por encima de **0.51**. Esto refuerza la decisión de haber sido estrictos en la purificación inicial de la escala.

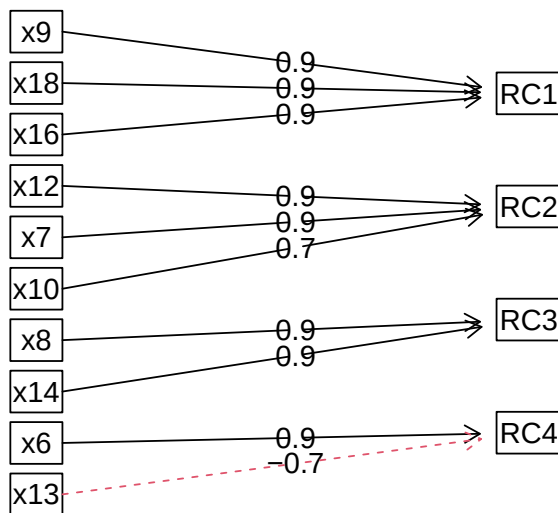
5.6. Interpretación de los nuevos componentes

A partir de la matriz rotada, se identifican las dimensiones subyacentes:

- **C1: Servicio postventa:** (x9: Resolución de quejas, x18: Rapidez de entrega, x16: Pedidos).
- **C2: Actividad comercial:** (x12: Imagen vendedores, x7: *e-Commerce*, x10: Publicidad).
- **C3: Soporte técnico y postventa:** (x8: Apoyo técnico, x14: Garantía).
- **C4: Valor de producto:** (x6: Calidad producto, x13: Competitividad precios).

```
# Factor diagram - visual summary of component loadings
psych::fa.diagram(pca_final, main = "Análisis de componentes
↪ principales")
```

Análisis de componentes principales



5.7. Las puntuaciones factoriales

Tras identificar las dimensiones latentes (como “Servicio postventa” o “Calidad de producto”), el siguiente paso es cuantificar cómo puntúa cada cliente en cada una de esas nuevas dimensiones. Para ello se calculan los *scores* o puntuaciones factoriales.

5.7.1. ¿Qué son exactamente las puntuaciones?

Si el PCA es un modelo donde cada componente se explica mediante una combinación lineal de las variables originales, la **puntuación factorial** es el valor numérico que obtiene cada individuo (encuestado) en ese nuevo eje o dimensión.

- Cada cliente de la base de datos deja de tener solo sus respuestas originales (x6, x7...) para poseer 4 nuevos valores correspondientes a los 4 componentes extraídos.

- Estas puntuaciones están **estandarizadas**: tienen media 0 y varianza 1. Por tanto, un cliente con un *score* de +1.5 en “Servicio postventa” está muy por encima de la media de satisfacción en esa área, mientras que uno con -2.0 representa un punto de fricción crítico para la empresa.

5.7.2. Utilidad estratégica: puente hacia la segmentación (*cluster*)

La creación de estas variables no es un fin en sí misma, sino el combustible para análisis avanzados en la toma de decisiones:

1. **Segmentación de mercado (análisis *cluster*):** es la aplicación principal. En lugar de segmentar clientes por variables aisladas (que suelen estar correlacionadas y “pesan” doble en el algoritmo), se utilizan los *scores*. Al ser ortogonales (independientes entre sí), se garantiza que cada dimensión de segmentación es única, lo que permite identificar grupos de clientes con necesidades claramente diferenciadas (ej. “los sensibles al precio” vs. “los que valoran el soporte técnico”).
2. **Modelos de regresión:** si se quisiera predecir la “Intención de compra” (x19) utilizando las percepciones, usar las 13 variables originales causaría problemas de **multicolinealidad**. Al usar los *scores*, este problema técnico se resuelve de raíz, obteniendo coeficientes mucho más estables y fiables.
3. **Mapas de posicionamiento:** es posible cruzar dos componentes (ej. Eje X: Calidad vs. Eje Y: Precio) y situar a cada cliente (o la media de un segmento) en un mapa bidimensional para visualizar la ventaja competitiva de la marca.

```
# Extract factor scores and append them to the original dataset
↪ with descriptive names
hbat_scores <- cbind(hbat, pca_final$scores) %>%
  dplyr::rename(
    Post_Sale_Service = RC1,
    Commercial_Activity = RC2,
    Technical_Support = RC3,
    Product_Value = RC4
  )
```

```
# Inspect the new score variables for the first few cases
hbat_scores %>%
  dplyr::select(id, Post_Sale_Service:Product_Value) %>%
  utils::head()
```

	id	Post_Sale_Service	Commercial_Activity	Technical_Support	Product_Value
1	1	0.2946283	0.7669530	-1.893546629	0.7011501
2	2	1.1387716	-1.6624231	-0.575796984	0.5755343
3	3	0.6665769	0.5805422	0.017132131	1.6109803
4	4	-0.9182291	-0.2457565	1.229367774	-1.2576711
5	5	-0.3199601	-0.8337491	-0.001359746	0.4406027
6	6	-0.6679233	-1.0533799	-1.329738840	-0.9836773

5.7.2.1. Comprobación del cálculo de los *scores*

```
# 1. Obtain the regression weights from the PCA output
weights <- as.matrix(pca_final$weights)

# 2. Standardise the purified data (mandatory step before
  ↪ applying weights)
# scale() subtracts the mean and divides by the standard
  ↪ deviation for each column
data_scaled <- scale(data_pca)

# 3. Manual calculation for the first record (Row 1)
# Multiply row 1 of the standardised data by the weights for RC1
score_manual_1 <- sum(data_scaled[1, ] * weights[, "RC1"])

# 4. Compare with the automatically computed score
score_automatic_1 <- pca_final$scores[1, "RC1"]

cat(
  "Manual calculation (standardised data):", score_manual_1,
  "\nResult from psych::principal():", score_automatic_1,
  "\n"
```

)

Manual calculation (standardised data): 0.2946283

Result from `psych::principal()`: 0.2946283

5.7.3. Representación gráfica

5.7.3.1. El mapa de individuos

Al visualizar un mapa cartesiano (*biplot* o gráfico de individuos) tras un PCA, se proyectan los encuestados sobre el nuevo espacio vectorial definido por los componentes:

- El eje X (horizontal) representa las puntuaciones del **Componente 1**.
- El eje Y (vertical) representa las puntuaciones del **Componente 2**.

La posición exacta de un cliente en ese mapa es el reflejo directo de sus *scores*. Un individuo situado en el extremo derecho del mapa tiene un *score* muy alto (positivo) en el primer componente.

5.7.3.2. ¿Por qué es útil esta visualización?

1. **Detección de *outliers*:** los puntos que quedan muy aislados en los márgenes del mapa son individuos con perfiles extremos (puntuaciones factoriales muy altas o muy bajas), lo que permite identificar casos atípicos que podrían distorsionar el análisis.
2. **Identificación visual de *clústeres*:** si los puntos forman nubes o grupos naturales en el mapa, se está ante la evidencia visual de que existen segmentos de mercado con percepciones similares. El análisis de conglomerados (*cluster*) simplemente pondrá nombre y límites matemáticos a esas agrupaciones que ya se intuyen mediante los *scores*.
3. **Relación con las variables (vectores):** al superponer las variables originales (vectores), puede observarse hacia dónde “tira” cada componente. Si un individuo está situado en la misma dirección que el vector “Calidad de Producto”, se deduce que ese cliente puntúa alto en esa percepción sin necesidad de consultar la base de datos original.

```

# 1. Prepare the data frame with scores
scores_df <- as.data.frame(pca_final$scores)
scores_df$client_id <- row.names(scores_df) # Label using row
↵ numbers

# 2. Scatter plot of individuals in the two-component space
ggplot2::ggplot(scores_df, ggplot2::aes(x = RC1, y = RC2, label
↵ = client_id)) +
  ggplot2::geom_hline(yintercept = 0, linetype = "dashed", color
↵ = "grey70") +
  ggplot2::geom_vline(xintercept = 0, linetype = "dashed", color
↵ = "grey70") +
  ggplot2::geom_point(color = "#2E5077", size = 2, alpha = 0.6)
↵ +
  ggrepel::geom_text_repel(
    size = 3, color = "grey30",
    max.overlaps = 20, segment.color = "grey80"
  ) +
  ggplot2::theme_minimal() +
  ggplot2::labs(
    title = "Mapa de individuos",
    subtitle = "Posicionamiento de cada encuestado en el espacio
↵ factorial",
    x = "Puntuaciones componente 1 (RC1)",
    y = "Puntuaciones componente 2 (RC2)"
  )

```

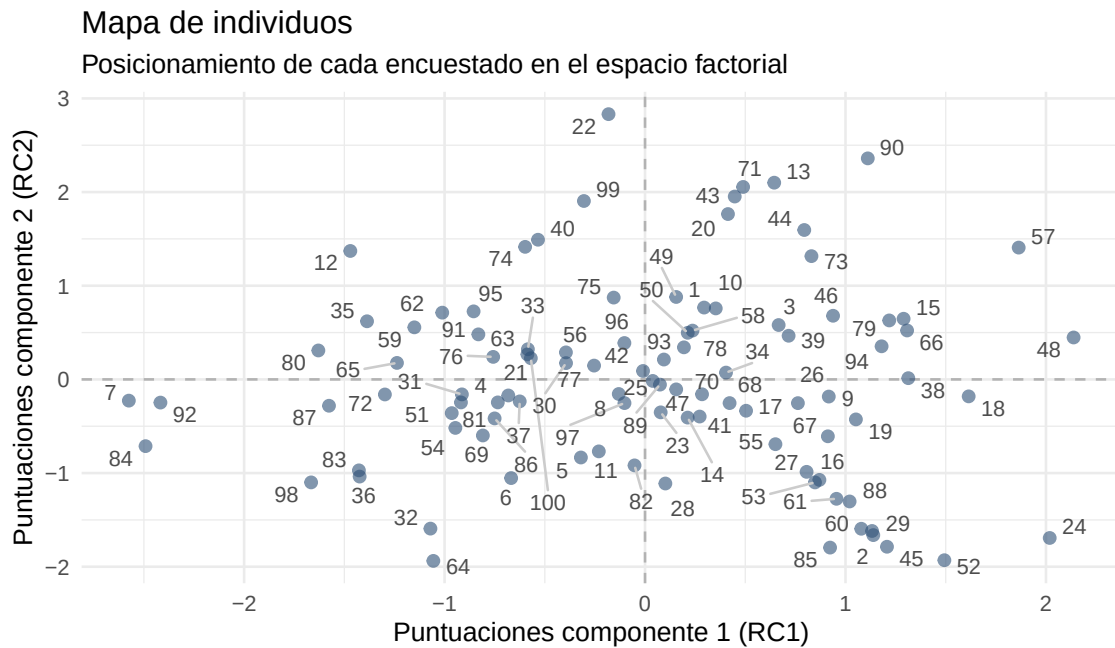


Figura 5.1: Mapa de posicionamiento de los clientes basado en puntuaciones factoriales

5.8. ¿PCA o FA?

El analista de mercados debe distinguir entre estas dos aproximaciones. Aunque ambas buscan reducir datos, su punto de partida difiere de forma fundamental:

5.8.1. Análisis de componentes principales (PCA)

Es un enfoque **matemático**. Su objetivo es explicar la **máxima varianza total** posible.

- **Filosofía:** asume que la varianza de una variable es “pura” (comunalidad inicial = 1).
- **Uso en *marketing*:** es la técnica estándar para reducir cuestionarios largos a unos pocos índices operativos (puntuaciones) que posteriormente se utilizarán para segmentar. Se busca síntesis.

5.8.2. Análisis factorial (FA o factores principales)

Es un enfoque **estadístico y causal**. Su objetivo es explicar la **covarianza** (la varianza compartida).

- **Filosofía:** asume que las variables observadas son solo indicadores de un **factor latente** no observado (ej. “la Inteligencia” o “la Personalidad”). La varianza se divide en varianza común y varianza única (error). Las communalidades iniciales son menores a 1.
- **Uso en investigación:** es típico en psicología o sociología, donde el interés no es reducir datos, sino identificar la estructura teórica subyacente que causa las respuestas.

5.8.3. Comparativa de resultados PCA / FA

Para contrastar la diferencia, se ejecutan ambos modelos sobre el mismo *dataset* depurado (`data_pca`). Se puede observar cómo cambian las cargas factoriales y las communalidades (h^2).

```
# 1. PCA - uses 1s on the diagonal of the correlation matrix
acp_res <- psych::principal(data_pca, nfactors = 4, rotate =
  ↪ "varimax")

# 2. EFA / PAF - estimates communalities iteratively (method =
  ↪ 'pa')
afp_res <- psych::fa(data_pca, nfactors = 4, rotate = "varimax",
  ↪ fm = "pa")

cat("Principal component analysis (PCA):\n")
print(acp_res)

cat("\nPrincipal axis factoring (FA):\n")
print(afp_res)
```

Principal component analysis (PCA):

Principal Components Analysis

Call: `psych::principal(r = data_pca, nfactors = 4, rotate = "varimax")`

Standardized loadings (pattern matrix) based upon correlation matrix

	RC1	RC2	RC3	RC4	h2	u2	com
x6	0.03	-0.01	-0.02	0.89	0.80	0.20	1.0
x7	0.06	0.87	0.05	-0.14	0.78	0.22	1.1
x8	0.02	-0.02	0.94	0.10	0.89	0.11	1.0
x9	0.93	0.10	0.06	0.08	0.89	0.11	1.0
x10	0.16	0.74	-0.08	0.04	0.58	0.42	1.1
x12	0.14	0.90	0.08	-0.17	0.86	0.14	1.1
x13	-0.10	0.23	-0.26	-0.73	0.66	0.34	1.5
x14	0.10	0.05	0.93	0.08	0.89	0.11	1.0
x16	0.89	0.10	0.09	0.07	0.81	0.19	1.1
x18	0.93	0.17	0.00	0.01	0.89	0.11	1.1

	RC1	RC2	RC3	RC4
SS loadings	2.59	2.22	1.85	1.41
Proportion Var	0.26	0.22	0.18	0.14
Cumulative Var	0.26	0.48	0.67	0.81
Proportion Explained	0.32	0.28	0.23	0.17
Cumulative Proportion	0.32	0.60	0.83	1.00

Mean item complexity = 1.1

Test of the hypothesis that 4 components are sufficient.

The root mean square of the residuals (RMSR) is 0.04
with the empirical chi square 17.86 with prob < 0.085

Fit based upon off diagonal values = 0.98

Principal axis factoring (FA):

Factor Analysis using method = pa

Call: psych::fa(r = data_pca, nfactors = 4, rotate = "varimax", fm = "pa")

Standardized loadings (pattern matrix) based upon correlation matrix

	PA1	PA2	PA3	PA4	h2	u2	com
x6	0.05	-0.06	0.02	0.65	0.43	0.575	1.0
x7	0.07	0.78	0.03	-0.14	0.63	0.367	1.1
x8	0.02	-0.02	0.89	0.12	0.81	0.191	1.0
x9	0.91	0.12	0.06	0.10	0.86	0.140	1.1
x10	0.17	0.53	-0.05	-0.05	0.32	0.683	1.2

x12	0.12	0.97	0.06	-0.15	0.99	0.011	1.1
x13	-0.10	0.21	-0.21	-0.58	0.44	0.559	1.6
x14	0.10	0.05	0.88	0.10	0.80	0.201	1.1
x16	0.79	0.12	0.09	0.10	0.66	0.340	1.1
x18	0.92	0.17	0.01	0.01	0.88	0.115	1.1

	PA1	PA2	PA3	PA4
SS loadings	2.39	1.95	1.63	0.85
Proportion Var	0.24	0.19	0.16	0.08
Cumulative Var	0.24	0.43	0.60	0.68
Proportion Explained	0.35	0.29	0.24	0.12
Cumulative Proportion	0.35	0.64	0.88	1.00

Mean item complexity = 1.1

Test of the hypothesis that 4 factors are sufficient.

df null model = 45 with the objective function = 5.3 with Chi Square = 502.97
 df of the model are 11 and the objective function was 0.08

The root mean square of the residuals (RMSR) is 0.01

The df corrected root mean square of the residuals is 0.02

The harmonic n.obs is 100 with the empirical chi square 0.44 with prob < 1
 The total n.obs was 100 with Likelihood Chi Square = 7.45 with prob < 0.76

Tucker Lewis Index of factoring reliability = 1.033

RMSEA index = 0 and the 90 % confidence intervals are 0 0.074

BIC = -43.21

Fit based upon off diagonal values = 1

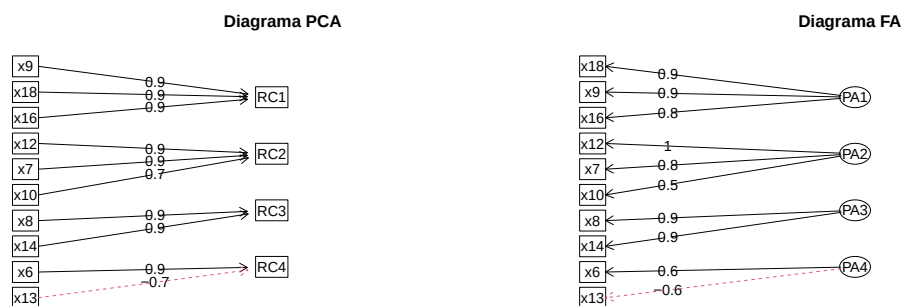
Measures of factor score adequacy

	PA1	PA2	PA3	PA4
Correlation of (regression) scores with factors	0.97	0.99	0.94	0.74
Multiple R square of scores with factors	0.93	0.97	0.88	0.55
Minimum correlation of possible factor scores	0.86	0.95	0.76	0.11

Al visualizar ambos modelos con `psych::fa.diagram()`, el estilo de las flechas comunica la naturaleza de la carga factorial:

1. **Color rojo y línea punteada:** indica una **carga factorial negativa**. La flecha hacia **x13** (Competitividad de precios) es roja porque tiene una correlación negativa con su componente: a medida que aumenta el valor del componente, el valor percibido en x13 disminuye (o viceversa).
2. **Color negro y línea sólida:** indica una **carga factorial positiva**. La mayoría de variables (x6, x9, x12...) muestran este comportamiento, lo que significa que “suman” positivamente a la definición del componente.
3. **Grosor de la flecha:** representa la **magnitud** (el valor absoluto) de la carga. Cuanto más gruesa, más importante es esa variable para definir ese factor.

```
# Visual comparison of PCA and FA diagrams side-by-side
psych::fa.diagram(acp_res, main = "Diagrama PCA")
psych::fa.diagram(afp_res, main = "Diagrama FA")
```



Si se observa que **x13 (Competitividad de precios)** presenta una flecha roja, el mensaje estratégico es claro: ese componente se mueve de forma inversa al precio competitivo. Los clientes que perciben a HBAT como una empresa de altísima calidad y servicio (puntuaciones altas en el componente) tienden a puntuar bajo en “precios competitivos” (perciben que la empresa es cara).

Es un indicador visual excelente del concepto de **polaridad de los ítems** y de cómo una dimensión puede estar formada por variables que se mueven en direcciones opuestas.

La dirección de las flechas no es meramente estética; representa la **relación de causalidad** (o dirección de formación del dato) entre lo que se observa (las preguntas del cuestionario) y lo que se calcula (los componentes o factores).

5.8.4. Comentarios finales

5.8.4.1. En PCA (análisis de componentes principales)

Dirección: Indicador $\rightarrow$ Componente

En el PCA, el componente es una **combinación lineal** de las variables originales. El componente no “existe” por sí mismo antes de los datos, sino que se “fabrica” sumando la aportación de cada variable original.

- **Lógica:** las variables originales son las que **causan** o **componen** el componente.
- **Metáfora:** al preparar una receta combinada, los ingredientes individuales van hacia la batidora para formar el producto final. Si se altera un ingrediente, el resultado cambia. El componente es un “resumen” de los datos.
- **PCA (flechas hacia adentro):** las variables “empujan” para crear un índice resumen. Se denomina **modelo formativo**.

5.8.4.2. En FA (análisis de factores principales)

Dirección: Factor $\rightarrow$ Indicador

En el análisis factorial se trabaja con el concepto de **variable latente**. Se asume que el factor (ej. “Satisfacción”) existe en la mente del consumidor y es lo que **provoca** que responda de una determinada manera a las preguntas.

- **Lógica:** el factor es la **causa** de las respuestas observadas. Si un cliente está muy satisfecho (factor), eso “causará” que puntúe alto en “calidad”, “limpieza” y “precio”.
- **El error (Unicidad):** como el factor causa la respuesta pero no la explica al 100 %, cada flecha que sale del factor hacia el indicador va acompañada de un margen de “error” o “especificidad” que representa lo que el factor no pudo explicar de esa pregunta.
- **FA (flechas hacia afuera):** el concepto latente “se refleja” en las respuestas de los clientes. Se denomina **modelo reflectivo**.

6 Análisis de Conglomerados (Cluster)



Profesor: Roberto Gil

2

Contenidos del tema / Índice



■ Herramientas para la segmentación del mercado

- ▶ Introducción
- ▶ Planteamiento del problema a resolver
- ▶ La práctica del análisis de conglomerados
- ▶ Solución en SPSS / R

Bibliografía recomendada

- Hair, J.F., Anderson, R.E., Tatham, R.L. Y Black, W. (1999). Análisis multivariante. 4ª edición. Madrid: Prentice Hall.
- Uriel, E. y Aldás, J. (2005). Análisis multivariante aplicado. Madrid: Thomson.
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). "An introduction to statistical learning" (Vol. 112). New York, NY: Springer.
- Chapman, C., & Feit, E. M. (2015). R for marketing research and analytics. New York, NY: Springer.

6.1. Introducción

Introducción



Profesor/a: Roberto Gil

3

El análisis de conglomerados es un conjunto de técnicas multivariantes utilizadas para clasificar a un conjunto de individuos en grupos homogéneos. A diferencia del análisis discriminante, en el **análisis *cluster*** los grupos son desconocidos *a priori*; el objetivo es precisamente determinarlos.

Este análisis presenta un marcado carácter **exploratorio** y se fundamenta en dos pilares:

1. **Interdependencia:** no se distingue entre variables dependientes e independientes.
2. **Determinación de grupos:** se busca la máxima homogeneidad *intra-grupo* (dentro del grupo) y la máxima heterogeneidad *inter-grupo* (entre grupos).

6.2. Un ejemplo intuitivo acerca del lenguaje

Para entender cómo funciona el algoritmo de forma visual, es posible imaginar que se compara cómo se escriben los números del 1 al 10 en 11 idiomas diferentes. El criterio de “distancia” o “similitud” es el número de veces que dos idiomas coinciden en la primera letra del número.

E	N	DA	DU	G	FR	SP	I	PL	H	FI
one	en	en	een	eins	un	uno	uno	jeden	egy	yksi
two	to	to	twee	zwei	deux	dos	due	dwa	kétó	kaksi
three	tre	tre	drie	drei	trois	tres	tre	trzy	három	kolme
four	fire	fire	vier	vier	quatre	cuatro	quattro	cztery	negy	neua
five	fem	fem	vijf	funf	cinq	cinco	cinque	piec	öt	viisi
six	seks	seks	zes	sechs	six	seis	sei	szesc	hat	kunsi
seven	syu	syv	zeven	sieben	sept	siete	sette	siedem	hét	seitsemän
eight	atte	otte	acht	acht	huit	ocho	otto	osiem	nyolc	kahdeksan
nine	ni	ni	negen	neun	neuf	nueve	nove	dziewiec	kilenc	yhdeksän
ten	ti	ti	tien	zehn	diz	diez	dieci	dziesięc	tíz	kymmenen

6.2.1. Matriz de datos original (frecuencia de coincidencia)

A continuación se recrea la matriz de similitudes que representa la semejanza de los idiomas entre sí:

El proceso de forma intuitiva

	E	N	DA	DU	G	FR	SP	I	P	H	FI
E	10										
N	8	10									
DA	8	9	10								
DU	3	5	4	10							
G	4	6	5	5	10						
FR	4	4	4	1	3	10					
SP	4	4	5	1	3	8	10				
I	4	4	5	1	3	9	9	10			
P	3	3	4	0	2	5	7	6	10		
H	1	2	2	2	1	0	0	0	0	10	
FI	1	1	1	1	1	1	1	1	1	2	10

- Disponemos de una enumeración de la traducción de los números hasta en 11 idiomas diferentes
- La matriz de datos, es una especie de matriz de distancias, y representa el número de ocasiones en que en cada idioma el número comienza por la misma letra que en el idioma con que se compara (`cluster1.sav`).

```
# Similarity matrix: value = number of matching first letters
↪ across digits 1-10
languages_data <- data.frame(
```

```

Language = c("EN", "NO", "DA", "DU", "GE", "FR", "SP", "IT",
  ↪  "PO", "HU", "FI"),
EN = c(10, 8, 8, 3, 4, 4, 4, 4, 3, 1, 1),
NO = c(8, 10, 9, 5, 6, 4, 4, 4, 3, 2, 1),
DA = c(8, 9, 10, 4, 5, 4, 5, 5, 4, 2, 1),
DU = c(3, 5, 4, 10, 5, 1, 1, 1, 0, 2, 1),
GE = c(4, 6, 5, 5, 10, 3, 3, 3, 2, 1, 1),
FR = c(4, 4, 4, 1, 3, 10, 8, 9, 5, 0, 1),
SP = c(4, 4, 5, 1, 3, 8, 10, 9, 7, 0, 1),
IT = c(4, 4, 5, 1, 3, 9, 9, 10, 6, 0, 1),
PL = c(3, 3, 4, 0, 2, 5, 7, 6, 10, 0, 1),
HU = c(1, 2, 2, 2, 1, 0, 0, 0, 0, 10, 2),
FI = c(1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 10)
)

cat("Similarity matrix:\n")
print(languages_data)

```

Similarity matrix:

	Language	EN	NO	DA	DU	GE	FR	SP	IT	PL	HU	FI
1	EN	10	8	8	3	4	4	4	4	3	1	1
2	NO	8	10	9	5	6	4	4	4	3	2	1
3	DA	8	9	10	4	5	4	5	5	4	2	1
4	DU	3	5	4	10	5	1	1	1	0	2	1
5	GE	4	6	5	5	10	3	3	3	2	1	1
6	FR	4	4	4	1	3	10	8	9	5	0	1
7	SP	4	4	5	1	3	8	10	9	7	0	1
8	IT	4	4	5	1	3	9	9	10	6	0	1
9	PO	3	3	4	0	2	5	7	6	10	0	1
10	HU	1	2	2	2	1	0	0	0	0	10	2
11	FI	1	1	1	1	1	1	1	1	1	2	10

6.2.2. El historial de conglomeración

El proceso jerárquico muestra cómo se van fusionando los casos. En las primeras etapas se unen los elementos con menor “distancia” (o mayor similitud):

1. **Etapas 1:** se unen los idiomas con mayor parecido.
2. **Etapas 2:** se unen nuevamente los idiomas con mayor parecido o se incorpora un nuevo idioma al grupo ya formado.
3. **Etapas sucesivas:** los grupos pequeños se van integrando en estructuras mayores (ej. el grupo escandinavo: Noruego y Danés).
4. **Resultado final:** un único grupo que engloba a todos los individuos.

Para que el algoritmo pueda trabajar con la matriz de similitudes, primero debe transformarse en una matriz de distancias. En este caso se resta la coincidencia máxima (10) menos el valor observado. Una distancia de 0 indica identidad total.

```
# 1. Convert similarity to distance (10 = maximum similarity →
  ↪ distance 0)
distances_lang <- stats::as.dist(10 -
  ↪ dplyr::select(languages_data, -Language))

cat("Distance matrix:\n")
print(distances_lang)

# 2. Hierarchical clustering with Ward's method
# 'ward.D2' matches SPSS exact algorithm for squared Euclidean
  ↪ distances
model_lang <- stats::hclust(distances_lang, method = "ward.D2")

# 3. Agglomeration schedule
agg_schedule <- data.frame(
  stage      = seq_len(length(model_lang$height)),
  element_1 = model_lang$merge[, 1],
  element_2 = model_lang$merge[, 2],
  distance  = model_lang$height
)

cat("\nAgglomeration schedule:\n")
print(agg_schedule)
```

Distance matrix:

EN NO DA DU GE FR SP IT PL HU

```

NO  2
DA  2  1
DU  7  5  6
GE  6  4  5  5
FR  6  6  6  9  7
SP  6  6  5  9  7  2
IT  6  6  5  9  7  1  1
PL  7  7  6 10  8  5  3  4
HU  9  8  8  8  9 10 10 10 10
FI  9  9  9  9  9  9  9  9  9  8

```

Agglomeration schedule:

	stage	element_1	element_2	distance
1	1	-2	-3	1.000000
2	2	-6	-8	1.000000
3	3	-7	2	1.732051
4	4	-1	1	2.236068
5	5	-9	3	4.898979
6	6	-4	-5	5.000000
7	7	4	6	7.576279
8	8	-10	-11	8.000000
9	9	7	8	12.078316
10	10	5	9	13.614355

6.2.2.1. Interpretación del historial

La tabla anterior permite explicar la lógica secuencial del algoritmo:

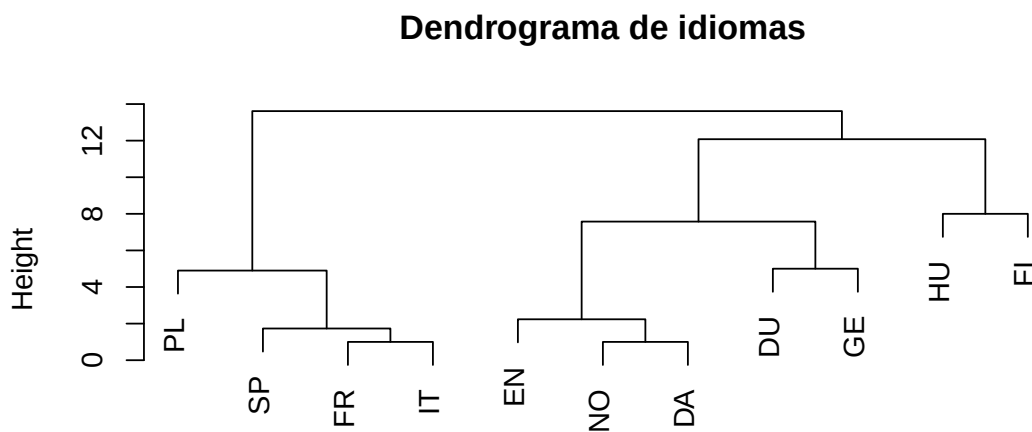
1. **Etapas iniciales:** se fusionan los elementos con los coeficientes más bajos (menor distancia). Por ejemplo, en la primera etapa se unen los idiomas con mayor coincidencia.
2. **Estructura del coeficiente:** el coeficiente (distancia) aumenta en cada etapa. Un salto brusco en este valor suele indicar que se está forzando la unión de grupos que son muy diferentes entre sí.
3. **De casos a grupos:** en las primeras etapas se unen casos (números negativos en R), y en etapas avanzadas aparecen fusiones con grupos ya formados (números positivos), indicando que un nuevo idioma se une a un grupo preexistente

o que dos grupos se fusionan.

6.2.2.2. Representación visual con dendrograma

Para cerrar la explicación intuitiva, se visualiza el “árbol” resultante:

```
# Dendrogram of the language similarity example
plot(model_lang, main = "Dendrograma de idiomas", xlab = "", sub
     ↵ = "")
```



6.3. Pasos y objetivos del análisis *cluster*

Para que la segmentación sea exitosa, se deben seguir estos pasos técnicos:

- **Establecer un criterio de similitud:** generalmente mediante una matriz de distancias (euclídea, Manhattan, etc.).
- **Algoritmo de clasificación:** elegir entre métodos jerárquicos (aglomerativos) o no jerárquicos (*K-means*).
- **Representación:** uso de **dendrogramas** para visualizar las uniones y decidir el número óptimo de grupos.

- **Caracterización y perfilado:** determinar qué variables definen cada grupo y validar si existen diferencias significativas mediante pruebas de contraste (ANOVA / Chi-cuadrado).

6.4. Adecuación de la estructura de datos

6.4.1. Detección de *outliers*

En la investigación de mercados, la presencia de valores atípicos es una de las mayores amenazas para la validez de una segmentación. Un *outlier* no es simplemente un “dato raro”; es una observación que procede de una estructura de población distinta a la que se desea modelizar o que introduce un sesgo desproporcionado en el cálculo de los centroides.

6.4.1.1. La naturaleza del *outlier* en el análisis de grupos

El análisis de conglomerados (especialmente el jerárquico y *K-means*) se basa en el cálculo de centros gravitacionales (medias). Si se incluyen sujetos con comportamientos extremos, el algoritmo intenta “acomodar” esos puntos, resultando en:

1. **Desplazamiento de centros:** el grupo no representará al cliente promedio, sino a un promedio distorsionado.
2. **Estructuras artificiales:** se crearán grupos de $N = 1$ o $N = 2$ que carecen de valor comercial o estadístico.
3. **Reducción del tamaño del efecto:** la varianza interna aumentará, haciendo que los grupos parezcan menos cohesionados de lo que realmente son.

6.4.1.2. Distancia de Mahalanobis

Mientras que la distancia euclídea es una medida de “línea recta” entre dos puntos, la **Distancia de Mahalanobis** (D^2) es una medida de distancia estadística. Su ventaja crítica reside en que es **invariante a la escala** y tiene en cuenta las **correlaciones** entre las variables.

Si dos variables de segmentación están muy correlacionadas (ej. Gasto en tienda y Gasto *online*), la distancia euclídea las contaría como dos dimensiones separadas,

“inflando” la importancia de ese comportamiento. Mahalanobis, en cambio, utiliza la matriz de varianzas-covarianzas (Σ) para ponderar el espacio:

$$D^2 = (x - \bar{x})^T S^{-1} (x - \bar{x})$$

Donde S^{-1} es la inversa de la matriz de covarianza de la muestra. Conceptualmente, mide la distancia de un caso al centroide del grupo en unidades de desviación típica multivariante.

6.4.1.3. El criterio de significación estadística χ^2

Para determinar si una D^2 es excesiva, se recurre a la distribución Chi-cuadrado (χ^2). Dado que la suma de variables normales al cuadrado sigue una distribución χ^2 , puede asignarse una probabilidad a cada distancia observada.

- **Grados de libertad** (df): se corresponden exactamente con el número de variables métricas introducidas en el análisis.
- **Umbral crítico** ($p < 0.001$): a diferencia del estándar 0.05 usado en contrastes de hipótesis, en la detección de *outliers* multivariantes se debe ser mucho más estricto. Solo se eliminan casos donde la probabilidad de que el sujeto pertenezca a la misma distribución que el resto sea prácticamente nula.

Antes de abordar el análisis *cluster*, se valida la no existencia de atípicos multivariantes:

```
# Load the HATCO dataset
hatco <- expss::read_spss('data/hatco.sav')
utils::head(hatco, 10)

# Define segmentation variables and remove missing values
# Mahalanobis distance cannot be computed in the presence of NAs
hatco_subset <- hatco %>%
  dplyr::select("X1", "X2", "X3", "X4", "X5", "X6", "X7") %>%
  tidyr::drop_na()

# Mahalanobis distance for each observation relative to the
↪ multivariate centroid
```

```

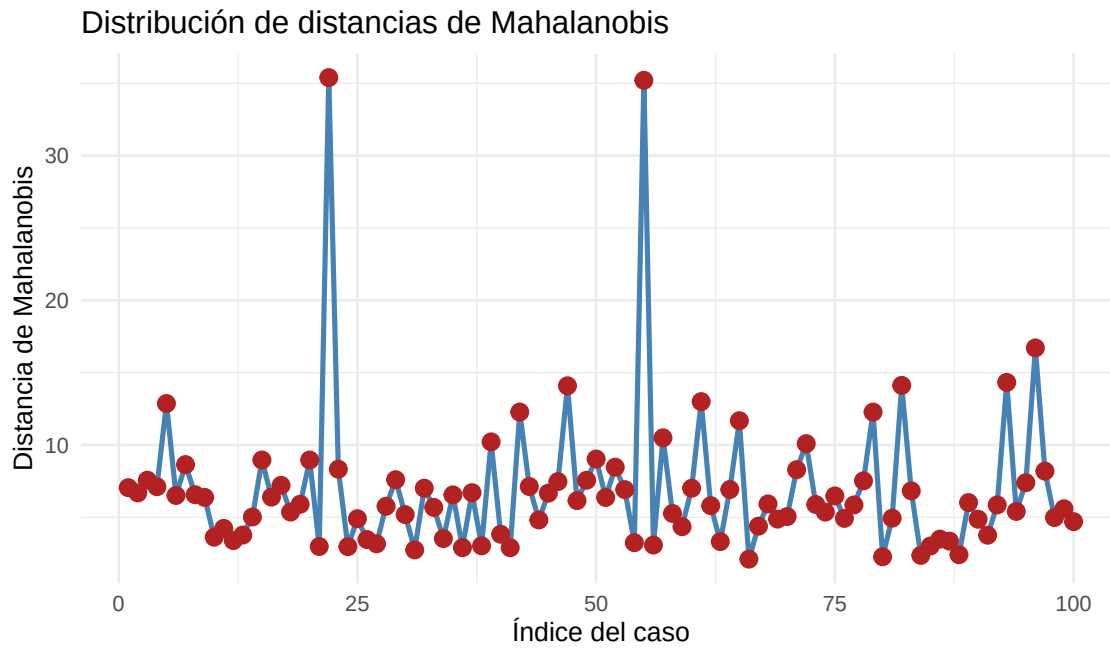
mahalanobis_values <- stats::mahalanobis(
  x      = hatco_subset,
  center = colMeans(hatco_subset),
  cov    = stats::cov(hatco_subset)
)

cat("Mahalanobis distance summary:\n")
summary(mahalanobis_values)

# Visualize the distribution of distances across observations
mahalanobis_df <- data.frame(
  observation_index = seq_len(length(mahalanobis_values)),
  distance_value    = mahalanobis_values
)

ggplot2::ggplot(mahalanobis_df, ggplot2::aes(x =
  ↪ observation_index, y = distance_value)) +
  ggplot2::geom_line(color = "steelblue", linewidth = 1) +
  ggplot2::geom_point(color = "firebrick", size = 3) +
  ggplot2::theme_minimal() +
  ggplot2::labs(
    title = "Distribución de distancias de Mahalanobis",
    x      = "Índice del caso",
    y      = "Distancia de Mahalanobis"
  )

```



	ID	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14
1	1	4.1	0.6	6.9	4.7	2.4	2.35	5.2	0	32	4.2	1	0	1	1
2	2	1.8	3.0	6.3	6.6	2.5	4.00	8.4	1	43	4.3	0	1	0	1
3	3	3.4	5.2	5.7	6.0	4.3	2.70	8.2	1	48	5.2	0	1	1	2
4	4	2.7	1.0	7.1	5.9	1.8	2.30	7.8	1	32	3.9	0	1	1	1
5	5	6.0	0.9	9.6	7.8	3.4	4.60	4.5	0	58	6.8	1	0	1	3
6	6	1.9	3.3	7.9	4.8	2.6	1.90	9.7	1	45	4.4	0	1	1	2
7	7	4.6	2.4	9.5	6.6	3.5	4.50	7.6	0	46	5.8	1	0	1	1
8	8	1.3	4.2	6.2	5.1	2.8	2.20	6.9	1	44	4.3	0	1	0	2
9	9	5.5	1.6	9.4	4.7	3.5	3.00	7.6	0	63	5.4	1	0	1	3
10	10	4.0	3.5	6.5	6.0	3.7	3.20	8.7	1	54	5.4	0	1	0	2

Mahalanobis distance summary:

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
2.108	4.322	5.912	6.930	7.531	35.391

Se calcula ahora la significación de esa medida de distancia y se obtiene su correspondiente gráfico:

```
# Convert Mahalanobis distances to chi-square p-values
# df = number of metric variables used in the distance
↪ calculation
n_vars <- 7
```

```

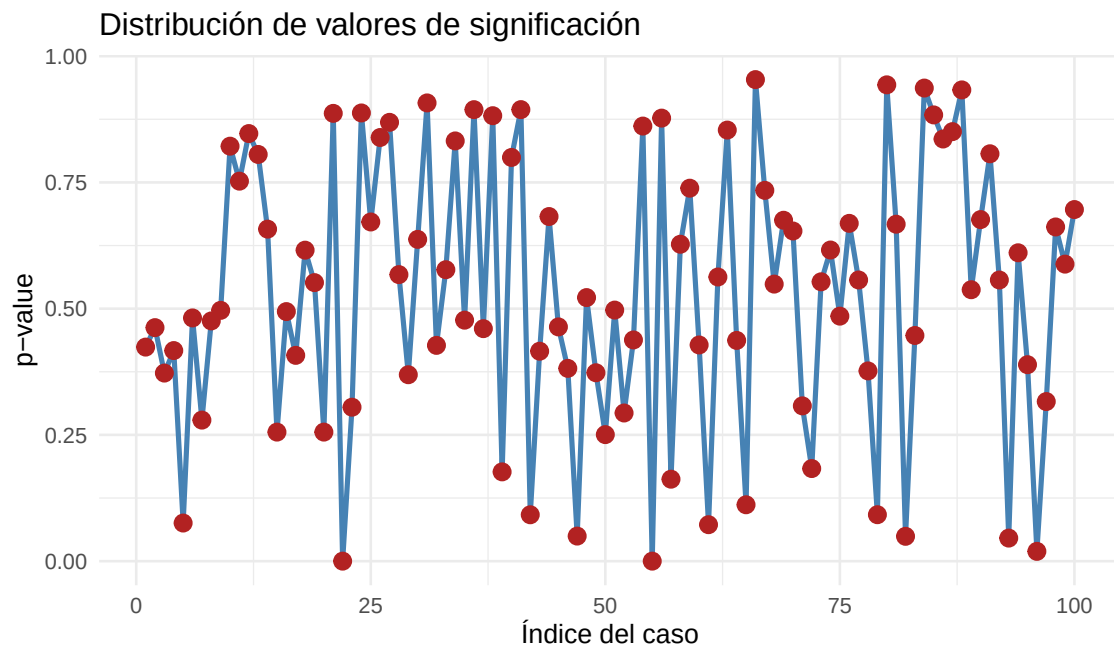
p_values <- stats::pchisq(
  mahalanobis_values,
  df          = n_vars,
  lower.tail = FALSE
)

cat("Chi-square significance values summary:\n")
summary(p_values)

p_values_df <- data.frame(
  observation_index = seq_len(length(p_values)),
  distance_value    = p_values
)

ggplot2::ggplot(p_values_df, ggplot2::aes(x = observation_index,
↵  y = distance_value)) +
  ggplot2::geom_line(color = "steelblue", linewidth = 1) +
  ggplot2::geom_point(color = "firebrick", size = 3) +
  ggplot2::theme_minimal() +
  ggplot2::labs(
    title = "Distribución de valores de significación",
    x      = "Índice del caso",
    y      = "p-value"
  )

```



Chi-square significance values summary:

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
9.440e-06	3.758e-01	5.501e-01	5.336e-01	7.420e-01	9.536e-01

A continuación se muestra el resultado para identificar y retener los casos válidos:

```
# Flag observations where p < 0.001 as multivariate outliers
outlier_report <- hatco_subset %>%
  dplyr::mutate(
    mahalanobis = mahalanobis_values,
    p_value     = p_values,
    is_outlier  = dplyr::if_else(p_value < 0.001, "Yes", "No")
  )

cat("Outlier count:\n")
table(outlier_report$is_outlier)

# Cases flagged for potential exclusion
outliers_detected <- outlier_report %>%
  dplyr::filter(is_outlier == "Yes")
print(outliers_detected)
```

```
# Retain only valid cases for the segmentation phase
hatco_subset <- outlier_report %>%
  dplyr::filter(is_outlier == "No")
```

Outlier count:

No Yes

98 2

	X1	X2	X3	X4	X5	X6	X7	mahalanobis	p_value	is_outlier
1	3.4	0.4	8.3	2.5	1.2	1.7	5.2	35.39056	9.443981e-06	Yes
2	3.8	0.8	8.7	2.9	1.6	2.1	5.6	35.20278	1.024445e-05	Yes

6.4.1.4. Eliminar o mantener, ¿qué decisión tomar?

No todos los *outliers* deben borrarse automáticamente. El analista debe considerar:

- **Error de datos:** si es un error de grabación, se corrige o elimina.
- **Sujetos excepcionales:** si el sujeto es real pero único (ej. una empresa con una facturación que duplica al segundo), su inclusión romperá la lógica de los grupos pequeños. En segmentación de mercados, es preferible **tratar a los outliers como un segmento separado** (“Segmento de excepciones”) y realizar el análisis *cluster* con el resto de la muestra para obtener grupos accionables y estables.

6.4.2. Medición de distancias

- **Distancia euclídea:** es recomendable cuando los datos están distribuidos de manera relativamente uniforme, no hay valores atípicos extremos que puedan distorsionar los resultados y se busca una medida de distancia más balanceada.
- **Distancia euclídea al cuadrado:** es recomendable cuando hay valores atípicos individuales de los que se desea minimizar la influencia, se busca una mayor separación entre los *clusters* y se quiere dar más peso a las diferencias grandes entre observaciones (es el estándar en el método de Ward).

6.4.3. Estandarización

Una vez detectados y tratados los valores atípicos mediante la distancia de Mahalanobis, el siguiente paso imperativo antes de ejecutar el algoritmo de segmentación es la **estandarización de las variables** para medir sus distancias.

- Las medidas de distancia son muy sensibles a las escalas de medición.
- Trabajar con diferentes escalas hace que las distancias entre grupos se vean muy afectadas por la métrica (ej. euros vs. años) y no resulte clara la diferenciación.
- Por tanto, es obligatorio estandarizar si las variables no están medidas en la misma escala.

6.4.3.1. El método de estandarización en R

Se utiliza la función `scale()`, que transforma todas las variables a una escala común con **media 0 y desviación típica 1**:

$$Z = \frac{x - \bar{x}}{s}$$

6.4.3.2. Ejemplo: cluster3.sav sin estandarizar

Para evidenciar las diferencias entre trabajar con estandarización o sin ella, se calcula primero una matriz de distancias del fichero `cluster3.sav` utilizando los datos brutos:

```
# Load example dataset with variables on very different scales
data_cluster3 <- expss::read_spss('data/cluster3.sav')
utils::head(data_cluster3, 10)

# Squared Euclidean distance matrix WITHOUT standardization
distances_raw <- stats::dist(
  dplyr::select(data_cluster3, activos, trabajadores),
  method = "euclidean"
)^2

cat("Distance matrix (raw, non-standardized):\n")
```

```
print(format(as.matrix(distances_raw), scientific = FALSE))
```

```

id   activos trabajadores
1 e1 1.000e+10          100
2 e2 1.005e+10           90
3 e3 1.000e+10          200
4 e4 1.005e+10          190
5 e5 2.000e+10          200
6 e6 2.005e+10          190
7 e7 2.000e+10          100
8 e8 2.005e+10           90
Distance matrix (raw, non-standardized):
  1                2                3
1 "                0" " 2500000000000100" "                10000"
2 " 2500000000000100" "                0" " 2500000000012100"
3 "                10000" " 2500000000012100" "                0"
4 " 2500000000008101" "                10000" " 250000000000100"
5 "10000000000000000000" " 99002500000000000000" "10000000000000000000"
6 "10100250000000000000" "10000000000000000000" "10100250000000000000"
7 "10000000000000000000" " 99002500000000000000" "10000000000000000000"
8 "10100250000000000000" "10000000000000000000" "10100250000000000000"
  4                5                6
1 " 2500000000008101" "10000000000000000000" "10100250000000000000"
2 "                10000" " 99002500000000000000" "10000000000000000000"
3 " 2500000000000100" "10000000000000000000" "10100250000000000000"
4 "                0" " 99002500000000000000" "10000000000000000000"
5 " 99002500000000000000" "                0" " 2500000000000100"
6 "10000000000000000000" " 2500000000000100" "                0"
7 " 99002500000000000000" "                10000" " 2500000000008101"
8 "10000000000000000000" " 2500000000012100" "                10000"
  7                8
1 "10000000000000000000" "10100250000000000000"
2 " 99002500000000000000" "10000000000000000000"
3 "10000000000000000000" "10100250000000000000"
4 " 99002500000000000000" "10000000000000000000"
5 "                10000" " 2500000000012100"

```

```

6 "      250000000000008101" "      10000"
7 "      0" "      25000000000000100"
8 "      25000000000000100" "      0"

```

6.4.3.3. Ejemplo: cluster3.sav estandarizado

A continuación se aplica el cálculo de los *Z-scores*:

```

# Standardize variables to Z-scores (mean = 0, sd = 1)
data_cluster3$z_activos      <- scale(data_cluster3$activos)
data_cluster3$z_trabajadores <-
  ↪ scale(data_cluster3$trabajadores)

cat("Extended data table (with Z-scores):\n")
print(data_cluster3)

# Squared Euclidean distance matrix WITH standardization
distances_std <- stats::dist(
  dplyr::select(data_cluster3, z_activos, z_trabajadores),
  method = "euclidean"
)^2

cat("\nDistance matrix (standardized):\n")
print(as.matrix(distances_std))

```

Extended data table (with Z-scores):

	id	activos	trabajadores	z_activos	z_trabajadores
1	e1	1.000e+10	100	-0.9400797	-0.8376949
2	e2	1.005e+10	90	-0.9307256	-1.0238493
3	e3	1.000e+10	200	-0.9400797	1.0238493
4	e4	1.005e+10	190	-0.9307256	0.8376949
5	e5	2.000e+10	200	0.9307256	1.0238493
6	e6	2.005e+10	190	0.9400797	0.8376949
7	e7	2.000e+10	100	0.9307256	-0.8376949
8	e8	2.005e+10	90	0.9400797	-1.0238493

Distance matrix (standardized):

	1	2	3	4	5	6	7
1	0.00000000	0.03474096	3.46534653	2.80701819	6.96525904	6.34192982	3.49991250
2	0.03474096	0.00000000	4.19315680	3.46534653	7.65807018	6.96525904	3.49965434
3	3.46534653	4.19315680	0.00000000	0.03474096	3.49991250	3.56965259	6.96525904
4	2.80701819	3.46534653	0.03474096	0.00000000	3.49965434	3.49991250	6.27193157
5	6.96525904	7.65807018	3.49991250	3.49965434	0.00000000	0.03474096	3.46534653
6	6.34192982	6.96525904	3.56965259	3.49991250	0.03474096	0.00000000	2.80701819
7	3.49991250	3.49965434	6.96525904	6.27193157	3.46534653	2.80701819	0.00000000
8	3.56965259	3.49991250	7.72806843	6.96525904	4.19315680	3.46534653	0.03474096

8
1 3.56965259
2 3.49991250
3 7.72806843
4 6.96525904
5 4.19315680
6 3.46534653
7 0.03474096
8 0.00000000

6.4.3.4. Consideraciones críticas en la estandarización

- **Ponderación implícita:** al estandarizar se otorga **el mismo peso específico** a todas las variables en la formación de los grupos. Si desde el análisis de negocio se considera que una variable es el doble de importante que el resto, deberían multiplicarse sus puntuaciones Z por 2 tras la estandarización.
- **Interpretación:** es fundamental recordar que el *cluster* se ejecuta sobre los datos estandarizados, pero para la **caracterización o perfilado final** de los grupos siempre se debe volver a las métricas originales para que las medias de los segmentos tengan sentido de negocio (ej. “el grupo 1 gasta 400€”, en lugar de “el grupo 1 tiene un Z de 0.8”).

6.5. Análisis *cluster*, introducción

A la hora de clasificar, el analista debe decidir qué tipo de algoritmo utilizar. Aunque ambos buscan crear grupos homogéneos, su funcionamiento y objetivos son diferen-

tes. En la práctica real de la investigación de mercados, se suelen utilizar de forma secuencial.

6.5.1. El método jerárquico: la exploración del número de grupos

Este algoritmo es de naturaleza **ascendente** y **exploratoria**. Al principio, cada cliente es un “grupo de una sola persona”. El algoritmo busca a los dos clientes más parecidos y los une. Luego busca la siguiente pareja más cercana, y así sucesivamente hasta que todos los individuos acaban formando parte de un único gran grupo.

- **¿Cuál es su gran ventaja?** No es necesario saber de antemano cuántos grupos crear. El análisis ofrece un “árbol de fusiones” (dendrograma) que permite visualizar cómo se estructuran los datos y decidir el número óptimo de segmentos.
- **Criterios de unión:** existen diferentes formas de medir la “cercanía” entre grupos (Vecino más cercano, Vinculación promedio, Método de Ward).

6.5.2. El método no jerárquico (*K-means*): el ajuste de precisión

A diferencia del anterior, este método es **confirmatorio**. El analista debe indicar al *software* exactamente cuántos grupos desea crear.

1. **Inicio:** el algoritmo coloca “centros” provisionales en el espacio.
2. **Asignación:** cada cliente se asigna al centro que tiene más cerca.
3. **Recálculo:** los centros se mueven para situarse en el promedio real de las personas asignadas a ese grupo.
4. **Optimización:** el proceso se itera hasta que los grupos son lo más compactos posible y ningún caso cambia de conglomerado.

6.5.3. La estrategia combinada: lo mejor de ambos mundos

En un flujo de trabajo profesional, lo habitual es seguir este orden de tres pasos:

1. **Fase exploratoria (Jerárquico):** ejecutar un *cluster* jerárquico para determinar cuántos grupos naturales existen en el mercado.

2. **Fijación de centros:** una vez decidido el número de grupos (por ejemplo, 3), se calculan los centros exactos de esos grupos.
3. **Fase de optimización (*K-means*):** utilizar esos centros como semillas para ejecutar un análisis de *K-means*, “recolocando” a los clientes situados en fronteras dudosas para lograr segmentos finales estables.

6.6. *Cluster* jerárquico

Como se ha visto, surge una pregunta clave: ¿cuándo se decide que dos grupos son “vecinos” y deben unirse? La respuesta depende del método de agrupación (o *linkage*) elegido.

6.6.1. Métodos de vinculación (*Linkage*)

- **Vínculo simple (*Single Linkage*):** utiliza al “vecino más cercano”. Basta con que dos casos de grupos distintos estén cerca para que los grupos se fusionen. Suele producir un problema llamado “encadenamiento”.
- **Vínculo completo (*Complete Linkage*):** utiliza al “vecino más lejano”. La distancia se define por los dos individuos más alejados entre sí. Es muy exigente y tiende a producir grupos compactos.
- **Vínculo promedio (*Average Linkage*):** calcula la distancia media entre todos los pares de individuos de dos grupos. Tiende a combinar grupos con varianzas pequeñas y es muy equilibrado.
- **Método de Ward:** es el más utilizado en investigación de mercados. A diferencia de los anteriores, minimiza la **varianza total dentro de los grupos**. Produce grupos extremadamente homogéneos y de tamaños muy similares.

6.7. Implementación del *cluster* jerárquico con R

Se utiliza la función estándar `stats::hclust()`, indicando el método deseado en el argumento `method`. La función `stats::cutree()` permite obtener la pertenencia de cada caso al grupo a partir de la solución jerárquica.

```

# Re-load HATCO dataset (fresh copy for the cluster analysis
↪ pipeline)
hatco <- expss::read_spss('data/hatco.sav')

# 1. Squared Euclidean distance matrix (required for Ward's
↪ method)
distances <- stats::dist(
  dplyr::select(hatco, X1, X2, X3, X4, X5, X6, X7),
  method = "euclidean"
)^2

# 2. Hierarchical models
# 'ward.D2' aligns with SPSS squared Euclidean implementation
model_ward <- stats::hclust(distances, method = "ward.D2")
model_avg <- stats::hclust(distances, method = "average")

# 3. Two-group solutions
cat("RESULTS: WARD'S METHOD\n\n")
cat("Frequency table - 2-group solution:\n")
table(stats::cutree(model_ward, k = 2))

cat("\nRESULTS: AVERAGE LINKAGE\n\n")
cat("Frequency table - 2-group solution:\n")
table(stats::cutree(model_avg, k = 2))

# Assign group membership to the dataset
hatco$cluster_ward <- stats::cutree(model_ward, k = 2)
hatco$cluster_avg <- stats::cutree(model_avg, k = 2)

# Cross-tabulation to assess method stability
cat("\nMethod stability cross-tabulation:\n")
table(hatco$cluster_ward, hatco$cluster_avg)

# Dendrograms
plot(model_ward, main = "Dendrograma - Método Ward", xlab =
↪ "", sub = "")

```

```
plot(model_avg, main = "Dendrograma - Vínculo Promedio", xlab =
  ↪ "", sub = "")
```

RESULTS: WARD'S METHOD

Frequency table - 2-group solution:

1	2
54	46

RESULTS: AVERAGE LINKAGE

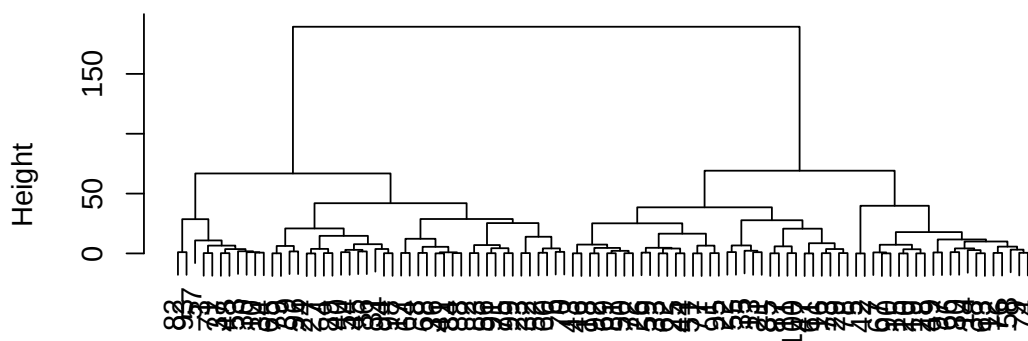
Frequency table - 2-group solution:

1	2
54	46

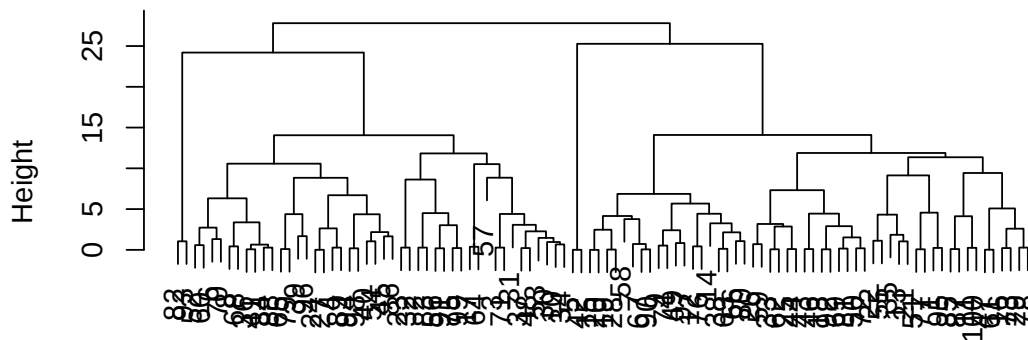
Method stability cross-tabulation:

	1	2
1	54	0
2	0	46

Dendrograma – Método Ward



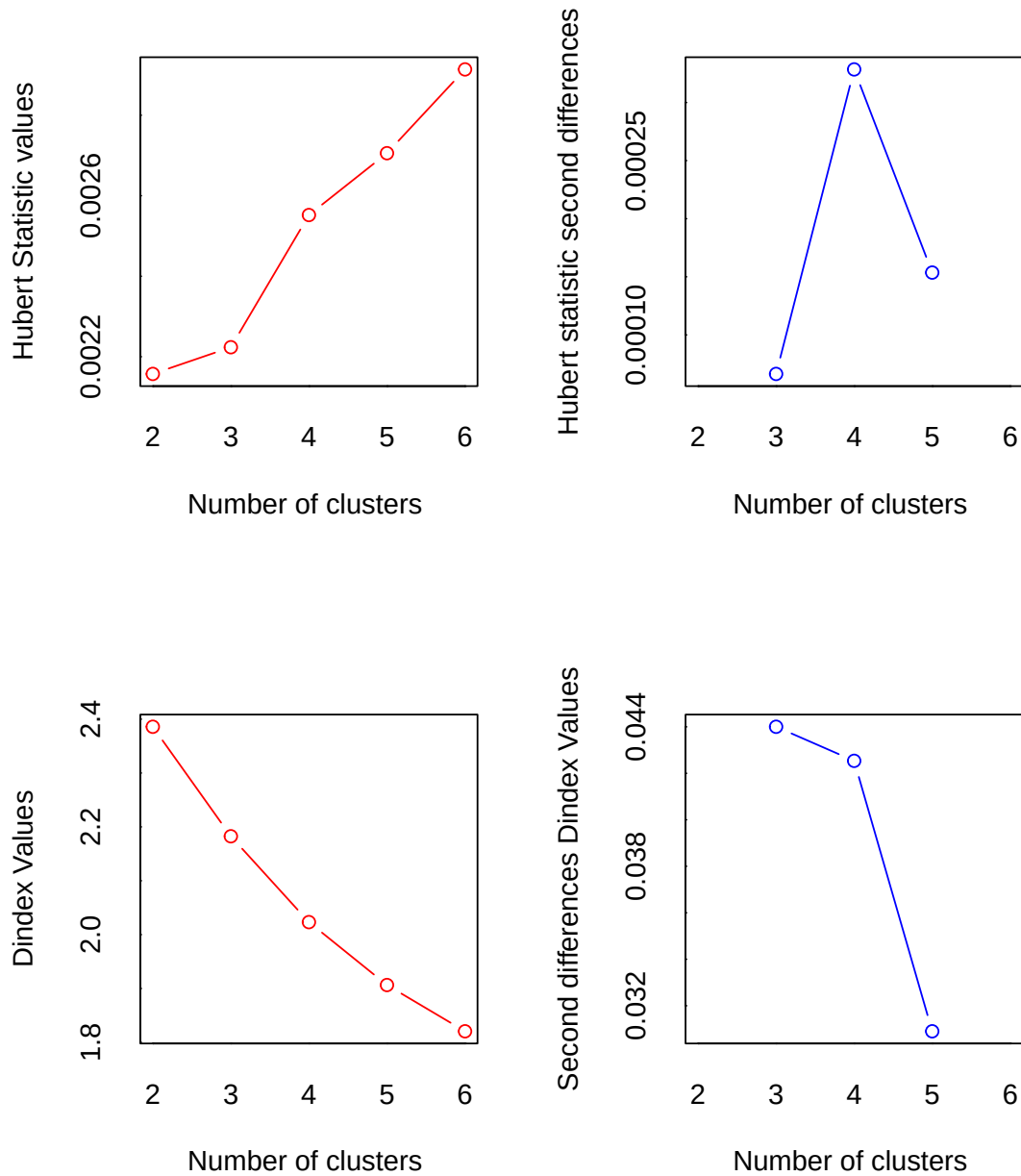
Dendrograma – Vínculo Promedio



6.7.1. Determinación del número óptimo de grupos

Para trascender la inspección visual del dendrograma, se utilizan criterios objetivos. El paquete `NbClust` ejecuta simultáneamente hasta **30 índices diferentes** (como Calinski-Harabasz o Silhouette) y aplica una “regla de mayoría” para sugerir el número óptimo de conglomerados.

```
# Consensus of 30 cluster validity indices to determine the
  ↳ optimal k
cat("Optimal number of groups - NbClust consensus:\n")
res_nb <- NbClust::NbClust(
  data      = dplyr::select(hatco, X1, X2, X3, X4, X5, X6, X7),
  distance  = "euclidean",
  min.nc    = 2,
  max.nc    = 6,
  method    = "ward.D2",
  index     = "all"
)
```



Optimal number of groups - NbClust consensus:

*** : The Hubert index is a graphical method of determining the number of clusters.
 In the plot of Hubert index, we seek a significant knee that corresponds to a significant increase of the value of the measure i.e the significant index second differences plot.

*** : The D index is a graphical method of determining the number of clusters.
 In the plot of D index, we seek a significant knee (the significant

second differences plot) that corresponds to a significant increase in the measure.

```
*****
```

```
* Among all indices:  
* 12 proposed 2 as the best number of clusters  
* 3 proposed 3 as the best number of clusters  
* 5 proposed 4 as the best number of clusters  
* 1 proposed 5 as the best number of clusters  
* 3 proposed 6 as the best number of clusters
```

```
***** Conclusion *****
```

```
* According to the majority rule, the best number of clusters is 2
```

```
*****
```

6.7.2. Obtención de centroides finales

Una vez que el análisis jerárquico ha ayudado a decidir el número de grupos (por ejemplo, $k = 2$), el siguiente paso estratégico es calcular los **centroides**. Estos vectores promedio servirán como “semilla” para inicializar el algoritmo *K-means*, garantizando una solución mucho más estable.

```
# Calculate mean of each metric variable per Ward group to  
↪ establish initial centroids  
initial_centroids <- hatco %>%  
  dplyr::group_by(cluster_ward) %>%  
  dplyr::summarise(  
    X1 = mean(X1, na.rm = TRUE),  
    X2 = mean(X2, na.rm = TRUE),  
    X3 = mean(X3, na.rm = TRUE),  
    X4 = mean(X4, na.rm = TRUE),  
    X5 = mean(X5, na.rm = TRUE),  
    X6 = mean(X6, na.rm = TRUE),
```

```

    X7 = mean(X7, na.rm = TRUE)
  ) %>%
  dplyr::select(X1, X2, X3, X4, X5, X6, X7) # Keep only metric
↪ values

cat("Initial centroids (group means from hierarchical
↪ solution):\n")
print(round(initial_centroids, 2))

```

Initial centroids (group means from hierarchical solution):

A tibble: 2 x 7

	X1	X2	X3	X4	X5	X6	X7
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	4.31	1.63	8.88	4.96	2.95	2.53	5.94
2	2.58	3.22	6.73	5.58	2.88	2.82	8.19

6.8. Cluster no jerárquico o *K-means*

6.8.1. La lógica del algoritmo *K-means*

El proceso *K-means* sigue una lógica de reasignación constante:

1. **Selección de semillas:** el algoritmo coloca k puntos en el espacio de datos (idealmente, los centroides calculados en la fase jerárquica).
2. **Asignación de casos:** cada sujeto de la muestra se asigna al centroide más cercano.
3. **Recálculo de centros:** los nuevos centros se recalculan basándose en la media real de los sujetos asignados.
4. **Optimización iterativa:** el proceso se repite hasta alcanzar el **criterio de convergencia** (cuando ningún sujeto cambia de grupo o el movimiento de los centros es estadísticamente insignificante).

6.9. Implementación del *cluster* no jerárquico (*K-means*) con R

Se procede a realizar el análisis confirmatorio alimentando el algoritmo con los centros iniciales. La función `stats::kmeans()` ejecuta las iteraciones necesarias hasta la convergencia.

```
# K-means initialized with the centroids from the hierarchical
  ↪ solution
# 'iter.max' defines the maximum number of passes to ensure
  ↪ convergence
set.seed(12345) # Set seed for perfect reproducibility
model_km <- stats::kmeans(
  x      = dplyr::select(hatco, X1, X2, X3, X4, X5, X6, X7),
  centers = initial_centroids,
  iter.max = 99
)

# Append optimized cluster membership to the dataset
hatco <- hatco %>%
  dplyr::mutate(cluster_km = as.factor(model_km$cluster))

cat("Final segment distribution (K-means):\n")
table(hatco$cluster_km)
```

Final segment distribution (K-means):

```
1 2
52 48
```

6.9.1. Caracterización de los segmentos

La **caracterización** consiste en analizar los centros finales del *K-means*. Estos valores indican la “identidad” del grupo: qué lo define y lo diferencia del resto.

```
# Final cluster centers - the 'identity card' of each segment
cat("Final K-means centroids:\n")
print(round(model_km$centers, 2))
```

Final K-means centroids:

	X1	X2	X3	X4	X5	X6	X7
1	4.38	1.58	8.9	4.93	2.96	2.53	5.90
2	2.57	3.21	6.8	5.60	2.87	2.82	8.13

6.9.2. Validación mediante ANOVA

Para asegurar que la caracterización es estadísticamente sólida, se realiza un análisis de varianza (ANOVA) sobre cada variable activa. Si $p < 0.05$, la variable discrimina correctamente entre grupos.

```
# ANOVA for each segmenting variable: tests whether the variable
# significantly differentiates between the identified clusters
cat("ANOVA - X1:\n")
summary(stats::aov(X1 ~ cluster_km, data = hatco))

cat("\nANOVA - X2:\n")
summary(stats::aov(X2 ~ cluster_km, data = hatco))
```

ANOVA - X1:

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
cluster_km	1	81.56	81.56	87.72	2.88e-15 ***
Residuals	98	91.12	0.93		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

ANOVA - X2:

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
cluster_km	1	66.46	66.46	86.75	3.73e-15 ***
Residuals	98	75.07	0.77		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

6.9.3. Perfilado de los segmentos

El **perfilado** es la fase final donde se cruza la pertenencia a los grupos (variable `cluster_km`) con variables descriptivas o de clasificación que **no participaron** en la creación del *cluster*:

1. **Variables categóricas:** se utiliza la prueba de **Chi-cuadrado** (χ^2).
2. **Variables métricas:** se utiliza el **ANOVA**.

Para obtener tablas con formato profesional se emplea el paquete **expss**, que incluye la significación estadística de forma integrada mediante letras de contraste.

```
# Descriptive label for the cluster variable
hatco <- expss::apply_labels(hatco, cluster_km = "K-means
  ↪ cluster")

# --- Categorical variable profiling (Chi-square) ---
# tab_last_sig_cases() adds significance letters between column
  ↪ proportions
cat("--- CATEGORICAL VARIABLES PROFILING ---\n")
hatco %>%
  expss::tab_cells(X8, X11) %>%
  expss::tab_cols(cluster_km) %>%
  expss::tab_stat_cases() %>%
  expss::tab_last_sig_cases() %>%
  expss::tab_pivot() %>%
  expss::set_caption("Categorical profiling - Chi-square
  ↪ significance")

# --- Metric variable profiling (ANOVA / T-test) ---
cat("\n--- METRIC VARIABLES PROFILING ---\n")
hatco %>%
  expss::tab_cells(X9, X10) %>%
  expss::tab_cols(cluster_km) %>%
  expss::tab_stat_mean_sd_n() %>%
  expss::tab_last_sig_means() %>%
  expss::tab_pivot() %>%
```

```
expss::set_caption("Metric profiling - mean comparison  
↪ significance")
```

--- CATEGORICAL VARIABLES PROFILING ---

Categorical profiling — Chi-square significance		
	K-means cluster	
	1	2
tamaño de la empresa		
pequeña	50.0	10.0
grande	2.0	38.0
#Chi-squared p-value	<0.05	
#Total cases	52.0	48.0
especificación de las compras		
especificacion	2.0	38.0
valor total	50.0	10.0
#Chi-squared p-value	<0.05	
#Total cases	52.0	48.0

--- METRIC VARIABLES PROFILING ---

Metric profiling — mean comparison significance		
	K-means cluster	
	1	2
	A	B
nivel de uso		
Mean	49.2 B	42.7
Std. dev.	9.0	7.7
Unw. valid N	52.0	48.0
nivel de satisfacción		
Mean	5.1 B	4.4
Std. dev.	0.8	0.8
Unw. valid N	52.0	48.0

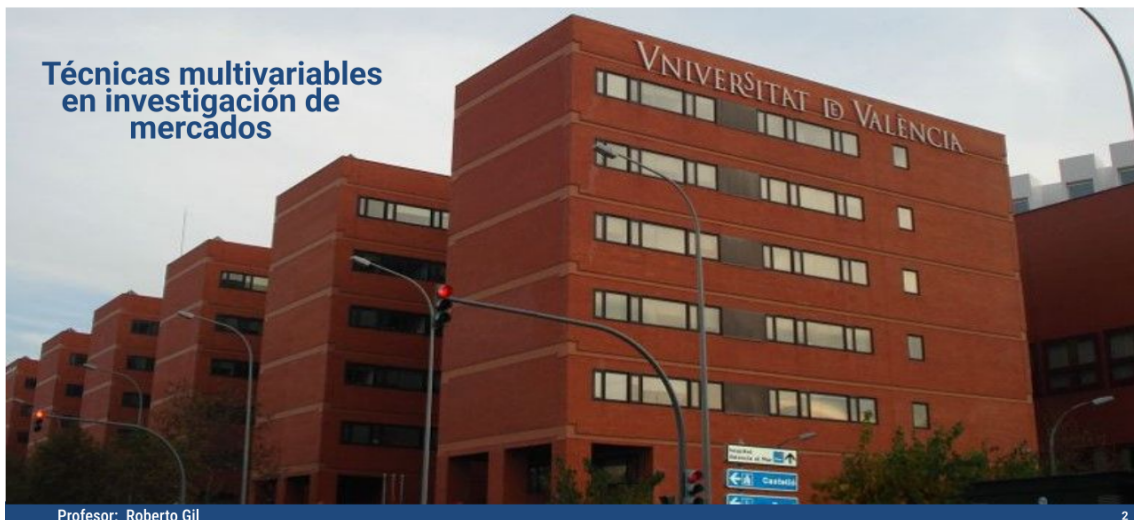
Al interpretar estas tablas, las **letras de significación** son el dato clave. Si un porcentaje o media tiene una letra diferente a la de otra columna, existe una **diferencia estadísticamente significativa** entre esos grupos.

6.10. Conclusión de la sesión

Con este proceso se ha completado el flujo de trabajo profesional del análisis de conglomerados:

1. **Limpieza:** detección de atípicos (Mahalanobis) para garantizar la calidad de los datos de partida.
2. **Preparación:** estandarización de variables para que las distancias no dependan de la escala de medición.
3. **Exploración:** *cluster* jerárquico (Ward) para determinar el número natural de grupos presentes en el mercado.
4. **Optimización:** *K-means* inicializado con los centroides jerárquicos para obtener segmentos finales estables y bien delimitados.
5. **Caracterización:** análisis de los centros finales del *K-means* para identificar la “identidad” de cada segmento mediante las variables activas.
6. **Perfilado:** descripción de cada segmento mediante variables de clasificación que no intervinieron en la construcción del *cluster*, con contraste de significación estadística integrado.

7 Análisis de Regresión Lineal



Profesor: Roberto Gil

2

Contenidos del tema / Índice



- **Explicación y predicción de variables métricas.**
 - ▶ Introducción
 - ▶ Planteamiento del problema a resolver
 - ▶ La práctica del análisis de regresión lineal múltiple por mínimos cuadrados ordinarios.
 - ▶ Solución en SPSS / R

Bibliografía recomendada

- Hair, J.F., Anderson, R.E., Tatham, R.L. Y Black, W. (1999). Análisis multivariante. 4ª edición. Madrid: Prentice Hall.
- Uriel, E. y Aldás, J. (2005). Análisis multivariante aplicado. Madrid: Thomson.
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). "An introduction to statistical learning" (Vol. 112). New York, NY: Springer.
- Chapman, C., & Feit, E. M. (2015). R for marketing research and analytics. New York, NY: Springer.



7.1. Introducción

El análisis de regresión múltiple es una **técnica de dependencia** fundamental en la modelización de datos. Se utiliza cuando el objetivo es explicar el comportamiento de una variable métrica (la variable dependiente) a partir de la influencia de varias variables independientes (que pueden ser métricas o, mediante transformaciones, no métricas como las dicotómicas).

Aunque es ampliamente aplicable a series temporales, en investigación de mercados se trabaja habitualmente con datos transversales para modelizar comportamientos y percepciones en un momento dado.

7.1.1. Objetivos y fines del modelo

El uso de la regresión en investigación de mercados persigue dos grandes fines:

- **Fines explicativos:** entender la estructura y relaciones subyacentes. Por ejemplo, *¿qué relación existe entre el gasto mensual de un cliente y su edad, nivel de ingresos y frecuencia de visita a la tienda?*
- **Fines predictivos:** estimar un valor futuro o desconocido. Por ejemplo, *estimar las ventas totales de un empleado en función de su experiencia previa, territorio asignado y horas de formación.*

7.1.2. Determinación de las variables

La selección de las variables es una de las tareas más críticas, ya que condiciona por completo la validez del resultado.

- **La variable dependiente** (Y): define el enfoque del problema. Su elección determina qué es exactamente lo que se intenta entender o predecir.
- **Las variables independientes** (X_k): deben seleccionarse basándose en la teoría o la lógica de negocio. Un exceso de variables independientes puede generar un problema de multicolinealidad, ocultando la contribución real e individual de cada variable al modelo.

7.2. Modelo de estimación

La estimación se realizará por **mínimos cuadrados ordinarios** (MCO u *OLS* por sus siglas en inglés, *Ordinary Least Squares*). El objetivo de este método es encontrar la línea (o hiperplano, en el caso múltiple) que minimice la suma de los cuadrados de las diferencias entre los valores reales observados y los valores predichos por el modelo.

La ecuación general de la regresión lineal múltiple se expresa formalmente como:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_k X_k + \epsilon$$

Donde:

- Y es la variable dependiente.
- β_0 es la constante o intersección.
- $\beta_1 \dots \beta_k$ son los coeficientes de regresión parcial asociados a cada variable independiente.
- $X_1 \dots X_k$ son las variables explicativas.
- ϵ es el término de error o residuo (lo que el modelo no logra explicar).

7.3. Planteamiento del problema a resolver

Para ilustrar de forma práctica todo el proceso, se utilizan datos reales procedentes de 82 institutos de la ciudad de Chicago (IL).

A modo de configuración inicial, se carga el conjunto de datos que se empleará en toda la sesión:

```
# Load dataset - institutoschicago.sav must be in the data/  
  ↪ project folder  
school_data <- haven::read_sav("data/institutoschicago.sav")  
  
# Remove records with missing values on any of the model  
  ↪ variables  
school_data <- school_data %>%  
  tidyr::drop_na(SELECTIVO, ASISTENCIA, RENTABAJA, EXTRANJERO,  
  ↪ TAMCLASE)  
  
# Apply descriptive labels for cleaner output in tables and  
  ↪ summaries  
school_data <- expss::apply_labels(  
  school_data,  
  SELECTIVO = "Percentage of passing students",  
  ASISTENCIA = "Attendance rate",  
  RENTABAJA = "Low income families rate",  
  EXTRANJERO = "Foreign students rate",  
  TAMCLASE = "Average class size"  
)
```

El contexto es relevante para interpretar correctamente los resultados: Chicago es una ciudad con una alta proporción de población afroamericana y de migrantes latinos.

Objetivo del caso: establecer qué explica que los alumnos procedentes de distintos institutos tengan un nivel de desempeño diferente, medido mediante un examen común (la selectividad).

Definición de las variables:

- **Variable dependiente:**
 - SELECTIVO: porcentaje de aprobados en selectividad.
- **Variables explicativas o independientes:**

- **ASISTENCIA:** porcentaje de estudiantes que acuden regularmente al instituto.
- **TAMCLASE:** número medio de alumnos en la clase donde recibieron formación.
- **EXTRANJERO:** porcentaje de alumnos procedentes de familias extranjeras con bajo dominio del idioma.
- **RENTABAJA:** porcentaje de alumnos procedentes de hogares de renta baja.

7.4. El proceso del análisis de regresión

Para que los resultados de una regresión sean fiables, no basta con ejecutar un comando. Debe seguirse un protocolo riguroso que garantice que los datos cumplen con los requisitos estadísticos.

7.4.1. Requisitos previos y tamaño de la muestra

Antes de comenzar, debe asegurarse que el diseño de investigación es sólido. Un error común es intentar explicar un fenómeno con demasiadas variables para muy pocos casos.

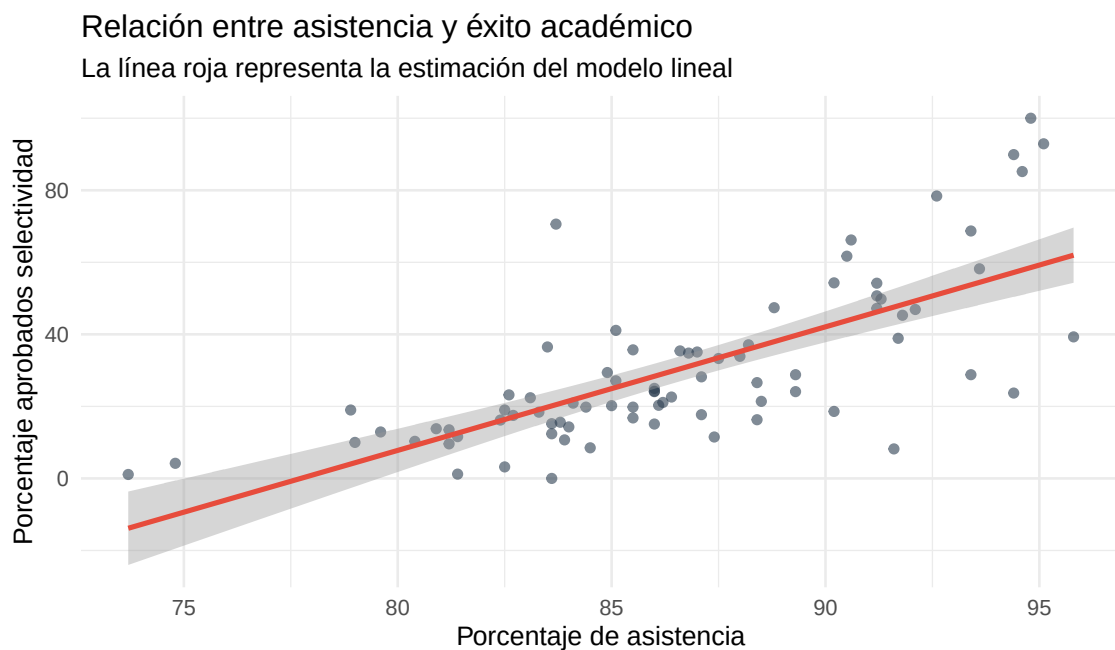
- **Proporción casos/variables:** como regla general, se necesita un mínimo de **15 a 20 observaciones por cada variable independiente** incluida en el modelo.
- **En el caso práctico:** se dispone de 82 institutos (casos, 80 con todos los datos) y 4 variables independientes (**ASISTENCIA**, **RENTABAJA**, **EXTRANJERO**, **TAMCLASE**). La ratio es de aproximadamente 20:1, lo cual es ideal para obtener resultados generalizables.

7.4.2. Relaciones lineales

Antes de pasar a los diagnósticos complejos, es vital visualizar si la relación asumida como lineal realmente lo parece en los datos.

```
# Scatter plot: Attendance vs Academic Success
```

```
# The linear fit (geom_smooth) offers a first visual check of
↪ linearity
ggplot2::ggplot(school_data, ggplot2::aes(x = ASISTENCIA, y =
↪ SELECTIVO)) +
  ggplot2::geom_point(color = "#2c3e50", alpha = 0.6) +
  ggplot2::geom_smooth(method = "lm", color = "#e74c3c", se =
↪ TRUE) +
  ggplot2::theme_minimal() +
  ggplot2::labs(
    title    = "Relación entre asistencia y éxito académico",
    subtitle = "La línea roja representa la estimación del
↪ modelo lineal",
    x        = "Porcentaje de asistencia",
    y        = "Porcentaje aprobados selectividad"
  )
```



7.4.3. Descriptivos y correlaciones

Antes de interpretar la capacidad explicativa conjunta del modelo, es una práctica fundamental analizar el comportamiento individual de las variables y sus relaciones bivariadas (uno a uno).

En primer lugar, se revisan los estadísticos descriptivos básicos (media, desviación típica y tamaño muestral N) para asegurar que los datos tienen sentido, trabajando ya únicamente con los casos válidos (sin perdidos).

```
# Subset with only the variables used in the regression
vars_regression <- school_data %>%
  dplyr::select(SELECTIVO, ASISTENCIA, RENTABAJA, EXTRANJERO,
  ↪ TAMCLASE)

# Descriptive statistics (psych::describe replicates SPSS's
  ↪ Descriptives output)
descriptives_table <- psych::describe(vars_regression) %>%
  ↪ round(2)
print(descriptives_table)
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew
SELECTIVO	1	80	30.41	22.29	23.45	27.26	16.61	0.0	100.0	100.0	1.24
ASISTENCIA	2	80	86.60	4.68	86.05	86.61	4.60	73.7	95.8	22.1	-0.10
RENTABAJA	3	80	80.81	15.31	85.65	83.45	7.86	23.1	98.6	75.5	-1.69
EXTRANJERO	4	80	4.06	6.35	0.35	2.68	0.52	0.0	22.3	22.3	1.52
TAMCLASE	5	80	17.91	4.41	17.90	18.15	4.60	4.8	27.2	22.4	-0.52
			kurtosis	se							
SELECTIVO			1.07	2.49							
ASISTENCIA			-0.23	0.52							
RENTABAJA			2.61	1.71							
EXTRANJERO			1.11	0.71							
TAMCLASE			0.12	0.49							

En segundo lugar, se analiza la **matriz de correlaciones de Pearson**. Esta tabla actúa como una “radiografía previa” al modelo múltiple y permite evaluar dos aspectos cruciales:

1. **Correlaciones con la variable dependiente:** se busca que las variables independientes tengan una correlación estadísticamente significativa ($p < 0.05$) con SELECTIVO. Sin correlación bivariada, difícilmente aportarán al modelo múltiple.

2. **Correlaciones entre las variables independientes:** proporciona una primera pista sobre posibles problemas de multicolinealidad. Si dos variables explicativas están fuertemente correlacionadas entre sí (por encima de 0.80), es probable que ambas estén midiendo aspectos muy similares y compitan por explicar la misma parte de la varianza.

```
# Pearson correlation matrix with p-values and sample sizes
# Hmisc::rcorr requires a matrix object as input
cor_matrix <- Hmisc::rcorr(as.matrix(vars_regression))

cat("--- Pearson correlation coefficients ---\n")
print(round(cor_matrix$r, 3))

cat("\n--- Significance (p-value) ---\n")
print(round(cor_matrix$P, 4))
```

--- Pearson correlation coefficients ---

	SELECTIVO	ASISTENCIA	RENTABAJA	EXTRANJERO	TAMCLASE
SELECTIVO	1.000	0.721	-0.714	-0.123	0.511
ASISTENCIA	0.721	1.000	-0.518	-0.010	0.593
RENTABAJA	-0.714	-0.518	1.000	0.103	-0.440
EXTRANJERO	-0.123	-0.010	0.103	1.000	0.036
TAMCLASE	0.511	0.593	-0.440	0.036	1.000

--- Significance (p-value) ---

	SELECTIVO	ASISTENCIA	RENTABAJA	EXTRANJERO	TAMCLASE
SELECTIVO	NA	0.0000	0.0000	0.2772	0.000
ASISTENCIA	0.0000	NA	0.0000	0.9304	0.000
RENTABAJA	0.0000	0.0000	NA	0.3642	0.000
EXTRANJERO	0.2772	0.9304	0.3642	NA	0.748
TAMCLASE	0.0000	0.0000	0.0000	0.7480	NA

7.5. Estimación del modelo: primera ejecución

7.5.1. Selección de las variables y método

En esta sesión se utiliza el método *Enter* (Introducir). Esto significa que se fuerza a todas las variables seleccionadas a entrar en el modelo simultáneamente. Es el método más robusto cuando existe una base teórica sólida que justifica la inclusión de dichas variables.

En R, el motor de la regresión lineal es la función `stats::lm()` (*linear model*). A diferencia de otros programas, la ejecución no arroja decenas de tablas de golpe, sino que guarda toda la información en un objeto que posteriormente se explora con `summary()`.

```
# Estimate the multiple regression model using OLS
# Syntax: dependent_variable ~ X1 + X2 + ... + Xk
model_chicago <- stats::lm(
  SELECTIVO ~
    ASISTENCIA +
    RENTABAJA +
    EXTRANJERO +
    TAMCLASE,
  data = school_data
)

# First look at the full model output
cat("Model results (first run):\n")
summary(model_chicago)
```

Model results (first run):

Call:

```
stats::lm(formula = SELECTIVO ~ ASISTENCIA + RENTABAJA + EXTRANJERO +
  TAMCLASE, data = school_data)
```

Residuals:

LABEL: Percentage of passing students

VALUES:

-34.845, -6.649, -1.301, 6.234, 52.897

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
(Intercept)	-110.5677	37.5578	-2.944	0.00431	**
ASISTENCIA	2.2045	0.4106	5.370	8.51e-07	***
RENTABAJA	-0.6517	0.1136	-5.739	1.90e-07	***
EXTRANJERO	-0.2593	0.2300	-1.127	0.26316	
TAMCLASE	0.2119	0.4162	0.509	0.61222	

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 12.86 on 75 degrees of freedom

Multiple R-squared: 0.6839, Adjusted R-squared: 0.6671

F-statistic: 40.57 on 4 and 75 DF, p-value: < 2.2e-16

7.5.2. Identificación de *outliers* y casos influyentes

Un modelo de regresión es muy sensible a casos extremos o al incumplimiento de ciertas reglas matemáticas. Antes de dar por válidos los coeficientes, se deben realizar dos tipos de comprobaciones:

- **Detección de casos influyentes con los residuos:** ¿existe algún instituto tan “especial” que esté arrastrando la línea de regresión hacia él, falseando el resultado general?
- **Validación de supuestos sobre el error:** ¿se comportan los residuos de forma aleatoria, normal y constante?

Para detectar estos casos perturbadores, se utilizan dos métricas complementarias:

- **Distancia de Mahalanobis:** mide la distancia de un caso respecto al centro de la nube de puntos de todas las variables independientes. Ayuda a identificar casos con combinaciones de variables explicativas muy inusuales (*outliers* multivariantes).
- **Distancia de Cook:** es, quizás, la medida más importante para el analista. No solo indica si un caso es “raro”, sino si tiene **influencia** en el modelo. Si se

eliminará un caso con una distancia de Cook alta, los coeficientes cambiarían significativamente.

¿Cómo proceder si se encuentra un *outlier*? No debe eliminarse por sistema. Primero se comprueba si es un error de grabación de datos. Si el dato es correcto, el investigador debe decidir si ese caso pertenece realmente a la población que se desea estudiar o si es una excepción que requiere un análisis separado. En el caso de Chicago, si un instituto tiene una configuración de alumnos extremadamente inusual, su eliminación podría hacer que el modelo sea mucho más robusto para explicar el comportamiento del resto de centros.

Estos indicadores — especialmente la distancia de Cook — se calculan con el modelo ya estimado:

```
# Extract the independent variable matrix once (used for
↪ Mahalanobis calculation)
indep_matrix <- as.matrix(
  school_data[, c("ASISTENCIA", "RENTABAJA", "EXTRANJERO",
↪ "TAMCLASE")]
)

# Append influence diagnostics to the dataframe
school_data <- school_data %>%
  dplyr::mutate(
    mahalanobis = stats::mahalanobis(
      x      = indep_matrix,
      center = colMeans(indep_matrix),
      cov    = stats::cov(indep_matrix)
    ),
    cook_d = stats::cooks.distance(model_chicago)
  )

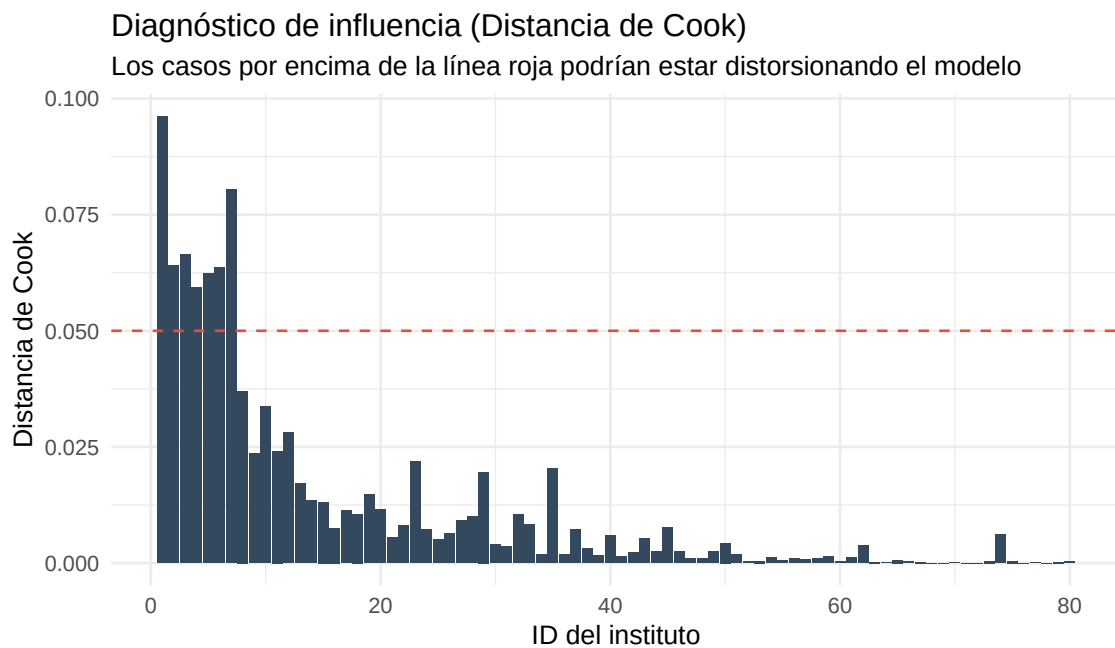
# Visualize Cook's Distance
# Common threshold: 4/n (here 4/82 = 0.049); cases above the red
↪ line warrant inspection
cutoff <- 4 / nrow(school_data)

ggplot2::ggplot(
```

```

school_data,
ggplot2::aes(x = seq_len(nrow(school_data)), y = cook_d)
) +
ggplot2::geom_bar(stat = "identity", fill = "#34495e") +
ggplot2::geom_hline(yintercept = cutoff, color = "#e74c3c",
↪ linetype = "dashed") +
ggplot2::theme_minimal() +
ggplot2::labs(
  title    = "Diagnóstico de influencia (Distancia de Cook)",
  subtitle = "Los casos por encima de la línea roja podrían
↪ estar distorsionando el modelo",
  x        = "ID del instituto",
  y        = "Distancia de Cook"
)

```



7.5.3. Indicadores de ajuste

Con el modelo estimado, la atención se centra en los indicadores que miden la “bondad del ajuste”:

```

# Full model summary - focus here on R2, Adjusted R2,
  ↪ F-statistic and Residual SE
cat("Model results (goodness of fit):\n")
summary(model_chicago)

# Multiple R: correlation between observed and predicted values
# Not shown directly in summary() - computed as sqrt(R2)
multiple_r <- sqrt(summary(model_chicago)$r.squared)
cat("Multiple correlation (R):", round(multiple_r, 4), "\n")

```

Model results (goodness of fit):

Call:

```
stats::lm(formula = SELECTIVO ~ ASISTENCIA + RENTABAJA + EXTRANJERO +
  TAMCLASE, data = school_data)
```

Residuals:

LABEL: Percentage of passing students

VALUES:

-34.845, -6.649, -1.301, 6.234, 52.897

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
(Intercept)	-110.5677	37.5578	-2.944	0.00431	**
ASISTENCIA	2.2045	0.4106	5.370	8.51e-07	***
RENTABAJA	-0.6517	0.1136	-5.739	1.90e-07	***
EXTRANJERO	-0.2593	0.2300	-1.127	0.26316	
TAMCLASE	0.2119	0.4162	0.509	0.61222	

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 12.86 on 75 degrees of freedom

Multiple R-squared: 0.6839, Adjusted R-squared: 0.6671

F-statistic: 40.57 on 4 and 75 DF, p-value: < 2.2e-16

Multiple correlation (R): 0.827

- **Coeficiente de correlación múltiple (R):** mide el grado de asociación lineal entre el valor real observado de la variable dependiente (**SELECTIVO**) y los valores predichos por el modelo. Su valor oscila entre 0 y 1. En R no aparece etiquetado como R en el `summary()`; se obtiene como la raíz cuadrada del R^2 , tal y como se ha calculado manualmente.
- **Coeficiente de determinación (R^2):** indica la proporción de la varianza de la variable dependiente que es explicada por el conjunto de las independientes.
- **Coeficiente de determinación ajustado ($Adjusted R^2$):** medida más conservadora y realista. Penaliza la inclusión de variables que no aportan valor real al modelo. Es el valor que debe reportarse habitualmente en investigación de mercados.
- **Estadístico F (ANOVA):** si el p-valor asociado es menor a 0.05, puede afirmarse que **el modelo es significativo en su conjunto**; es decir, que al menos una de las variables explica el éxito en selectividad mejor que el simple azar.
- **Error típico de la estimación ($Residual standard error$):** indica, en las mismas unidades que la variable dependiente (porcentaje de aprobados), cuánto se equivoca de media el modelo al realizar una predicción. Cuanto menor sea este valor, más precisas serán las estimaciones.

Además, la tabla de coeficientes no estandarizados proporciona los valores exactos para construir la **ecuación de predicción**:

$$SELECTIVO = \beta_0 + (\beta_1 \cdot ASISTENCIA) + (\beta_2 \cdot RENTABAJA) + (\beta_3 \cdot EXTRANJERO) + (\beta_4 \cdot T)$$

donde las β_k son las estimaciones de los parámetros.

7.6. Diagnóstico del modelo

Tras auditar la calidad del modelo, se entra en la fase de interpretación de resultados. Deben responderse tres preguntas: ¿es el modelo útil en su conjunto?, ¿qué variables son significativas?, ¿cuál es la que más influye?

Una vez validado que el modelo funciona de forma global, se desciende al detalle de los coeficientes (β). Se distinguen dos tipos de información:

```
# Model summary displayed again here as the starting point for
↪ coefficient interpretation
cat("Model results (coefficient interpretation):\n")
summary(model_chicago)

# Multiple R (repeated for interpretive completeness)
multiple_r <- sqrt(summary(model_chicago)$r.squared)
cat("Multiple correlation (R):", round(multiple_r, 4), "\n")
```

Model results (coefficient interpretation):

Call:

```
stats::lm(formula = SELECTIVO ~ ASISTENCIA + RENTABAJA + EXTRANJERO +
  TAMCLASE, data = school_data)
```

Residuals:

LABEL: Percentage of passing students

VALUES:

-34.845, -6.649, -1.301, 6.234, 52.897

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
(Intercept)	-110.5677	37.5578	-2.944	0.00431	**
ASISTENCIA	2.2045	0.4106	5.370	8.51e-07	***
RENTABAJA	-0.6517	0.1136	-5.739	1.90e-07	***
EXTRANJERO	-0.2593	0.2300	-1.127	0.26316	
TAMCLASE	0.2119	0.4162	0.509	0.61222	

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 12.86 on 75 degrees of freedom

Multiple R-squared: 0.6839, Adjusted R-squared: 0.6671

F-statistic: 40.57 on 4 and 75 DF, p-value: < 2.2e-16

Multiple correlation (R): 0.827

- **Coefficientes no estandarizados (B / Estimate):** representan el cambio

en la variable dependiente por cada unidad que aumenta la independiente, manteniendo el resto constantes. Se usan para la **ecuación de predicción**.

- **Coeficientes estandarizados (Beta):** son fundamentales para el **objetivo explicativo**. Al estar en la misma escala (desviaciones típicas), permiten comparar qué variable tiene mayor peso relativo en el modelo. No se ofrecen directamente en el `summary()` base y deben calcularse explícitamente.

7.6.1. Significación global del modelo (Prueba F)

La primera parada es la tabla ANOVA. La prueba F contrasta la hipótesis nula de que todos los coeficientes de las variables independientes son cero.

En la salida se muestra el resultado:

F-statistic: 40.57 on 4 and 75 DF, p-value: < 2.2e-16

Si p-valor < 0.05: se rechaza la hipótesis nula. El modelo es significativamente mejor que usar simplemente la media para predecir. Existe, al menos, una relación lineal real.

7.6.2. Significación de los parámetros individuales (Prueba t)

No todas las variables que se introducen en el modelo tienen por qué ser relevantes. La prueba **t de Student** evalúa la contribución individual de cada variable manteniendo las demás constantes.

- **Hipótesis:** la variable X_i no ayuda a explicar Y .
- **Decisión:** si el p-valor asociado a la t es menor de 0.05, se concluye que esa variable es un predictor significativo para el éxito en selectividad.
- Cada variable tiene su propio contraste. Si el p-valor ($Pr(>|t|)$) es inferior a 0.05, esa variable contribuye significativamente al modelo.

7.6.3. Interpretación de los coeficientes: B vs. β

Para entender el “peso” de las variables, R ofrece dos tipos de coeficientes:

- **Coeficientes no estandarizados (B / Estimate):** expresados en las unidades originales (ej: por cada 1 % que sube la asistencia, los aprobados suben un B %). Son útiles para la **ecuación de predicción**.

- **Coefficientes estandarizados (Beta):** expresados en desviaciones típicas. Al estar todos en la misma escala, permiten **comparar la importancia relativa**. La variable con el β absoluto más alto es la que tiene mayor impacto en el modelo.

```
# Standardized coefficients (Beta) via lm.beta::lm.beta()
# summary() here shows both B (Estimate) and the standardized
  ↪ Beta side by side
model_beta <- lm.beta::lm.beta(model_chicago)
summary(model_beta)
```

Call:

```
stats::lm(formula = SELECTIVO ~ ASISTENCIA + RENTABAJA + EXTRANJERO +
  TAMCLASE, data = school_data)
```

Residuals:

LABEL: Percentage of passing students

VALUES:

-34.845, -6.649, -1.301, 6.234, 52.897

Coefficients:

	Estimate	Standardized	Std. Error	t value	Pr(> t)	
(Intercept)	-110.56772	NA	37.55778	-2.944	0.00431	**
ASISTENCIA	2.20450	0.46328	0.41055	5.370	8.51e-07	***
RENTABAJA	-0.65173	-0.44757	0.11356	-5.739	1.90e-07	***
EXTRANJERO	-0.25926	-0.07389	0.22996	-1.127	0.26316	
TAMCLASE	0.21188	0.04195	0.41624	0.509	0.61222	

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 12.86 on 75 degrees of freedom

Multiple R-squared: 0.6839, Adjusted R-squared: 0.6671

F-statistic: 40.57 on 4 and 75 DF, p-value: < 2.2e-16

7.6.4. Implicaciones de los coeficientes no significativos

7.6.4.1. El principio de parsimonia (la navaja de Ockham)

En ciencia de datos, se busca el modelo más simple posible que explique la realidad. Una variable que **no es significativa** ($p > 0.05$) es, estadísticamente hablando, **ruido**. No aporta información real para predecir, ocupa espacio y consume grados de libertad. Mantenerla en el modelo desvirtúa la ecuación y dificulta la interpretación estratégica.

7.6.4.2. Eliminación hacia atrás (*Backward Elimination*)

Debe seguirse este protocolo estricto:

- Observar los p-valores de las variables no significativas.
- Identificar la que tiene el p-valor más alto (la “peor” predictora).
- Eliminar SOLAMENTE esa variable.
- Volver a ejecutar el modelo (reespecificar).

Las variables independientes suelen estar correlacionadas entre sí. En ocasiones, una variable parece “no significativa” únicamente porque otra variable le está “robando” la explicación (multicolinealidad leve).

Efecto Lázaro: al eliminar la variable con el peor p-valor, es posible que otra variable que antes no era significativa se vuelva significativa ($p < 0.05$) al tener más varianza disponible para explicarse.

7.6.4.3. El indicador clave, el *Adjusted R²*

¿Cómo asegurar que se ha acertado al eliminarla? Se observa el *Adjusted R²*. El R^2 normal siempre bajará ligeramente al quitar variables (es un principio matemático). Sin embargo, si el *Adjusted R²* sube o se mantiene igual, significa que la decisión fue correcta: se ha eliminado ruido reteniendo la mejor información.

7.6.4.4. Excepción: variables de control teórico

Existe un escenario donde no debe eliminarse una variable aunque carezca de significación: cuando la teoría o el diseño del experimento exigen su inclusión. Ejemplo: si se mide el efecto de una campaña de *marketing* en ventas y se incluye el “Precio” como variable de control. Si el precio no resulta significativo en esa muestra concreta, no debe eliminarse; de lo contrario, el modelo podría acusarse de sesgo por omisión. Se retiene para afirmar: “controlando por el efecto del precio, la campaña tuvo un efecto positivo”.

7.6.5. Correlaciones parciales y semiparciales

En investigación de mercados no basta con saber si una variable es significativa; se busca cuantificar su **aportación única**. Dado que las variables suelen estar relacionadas entre sí, el R^2 no es simplemente la suma de las correlaciones individuales. Para desgranar esta información se utilizan:

1. **Correlación parcial:** mide la relación entre una variable independiente y la dependiente, eliminando el efecto que las otras variables independientes tienen sobre *ambas*.
2. **Correlación semiparcial (o Part):** mide la relación eliminando el efecto de las otras independientes solo sobre la variable explicativa actual. **Es la medida más pura de aportación**, ya que su cuadrado indica exactamente cuánto aumentaría el R^2 si se añadiera esa variable en último lugar al modelo.

```
# Semi-partial (Part) correlations - unique contribution of each
↪ predictor
part_corr <- ppcor::spcor(vars_regression)

# Partial correlations - controlling for all other predictors on
↪ both variables
partial_corr <- ppcor::pcor(vars_regression)

# Summary table: zero-order, partial and part correlations with
↪ unique R2 contribution
contribution_table <- data.frame(
  Variable      = names(vars_regression)[-1],
```

```

Cor_Order_0 = cor_matrix$r[1, -1],
Partial      = partial_corr$estimate[1, -1],
Part         = part_corr$estimate[1, -1]
) %>%
  dplyr::mutate(Contribution_R2 = Part^2) # Unique% of R²
↪ explained by each predictor

print(contribution_table[, -1])

```

	Cor_Order_0	Partial	Part	Contribution_R2
ASISTENCIA	0.7208610	0.52695673	0.34858381	0.121510671
RENTABAJA	-0.7137334	-0.55240542	-0.37257096	0.138809118
EXTRANJERO	-0.1229632	-0.12909458	-0.07319028	0.005356817
TAMCLASE	0.5108685	0.05867719	0.03304565	0.001092015

Al interpretar esta tabla, el analista puede concluir con rigor:

- **La variable dominante:** aquella con la correlación semiparcial (**Part**) más alta es la que aporta más información exclusiva.
- **Redundancia:** si la correlación de orden cero es muy alta pero la parcial es muy baja, la variable es redundante (su información ya la están aportando otras).
- **Utilidad estratégica:** el valor **Contribution_R2** revela qué porcentaje del éxito en selectividad se explica *únicamente* por esa variable y por ninguna otra.

7.7. Validación de los supuestos básicos del modelo

Una vez analizada la capacidad explicativa, debe certificarse que el modelo cumple con los “cuatro pilares” de la regresión lineal. Si estos supuestos fallan, las pruebas de significancia (F y t) pierden su validez y las conclusiones podrían ser erróneas.

La validación se realiza interrogando los residuos (el error cometido) en busca de patrones que no deberían existir:

0. **Muestra:** relación entre el número de indicadores y casos.
1. **Residuos extremos:** *outliers* en los residuos.

2. **Independencia:** los errores no deben estar relacionados entre sí (Durbin-Watson).
3. **Linealidad:** la relación entre las variables debe ser rectilínea.
4. **Homocedasticidad:** la varianza del error debe ser constante (sin “embudos”).
5. **Normalidad:** los errores deben seguir una distribución normal con media cero.

Para estas comprobaciones, con el modelo ya calculado, se calcula una serie de variables diagnósticas:

1. **PRE (Valor predicho):** se obtiene con `stats::predict()`.
2. **RES (Residuo bruto):** se obtiene con `stats::residuals()`.
3. **ZPR (Valor predicho estandarizado):** se calcula tipificando los valores predichos con `scale()`.
4. **ZRE (Residuo estandarizado):** `stats::rstandard()` calcula el residuo estandarizado corregido por el apalancamiento o *leverage*.
5. **SRE (Residuo estudentizado):** se obtiene con `stats::rstudent()` (equivalente al residuo eliminado estudentizado).

```
# Calculate and append diagnostic variables using Base R
↪ functions
# as.vector() on scale() avoids matrix formatting issues inside
↪ the dataframe
school_data <- school_data %>%
  dplyr::mutate(
    pred_val   = stats::predict(model_chicago),
    resid_val  = stats::residuals(model_chicago),
    z_pred     =
      ↪ as.vector(scale(stats::predict(model_chicago))),
    z_resid    = stats::rstandard(model_chicago),
    stud_resid = stats::rstudent(model_chicago)
  )

# Inspect the first rows to verify the new diagnostic columns
school_data %>%
  dplyr::select(INSTITUTO, SELECTIVO, pred_val, resid_val,
  ↪ z_pred, z_resid, stud_resid) %>%
  utils::head()
```

```
# A tibble: 6 x 7
```

	INSTITUTO	SELECTIVO	pred_val	resid_val	z_pred	z_resid	stud_resid
	<chr>	<labelled>	<dbl>	<labelle>	<dbl>	<labelle>	<labelled>
1	FLOWER VOC HIGH	70.6	17.7	52.89734	-0.689	4.169925	4.725946
2	JONES COLL PREP HIGH	85.2	56.7	28.49938	1.43	2.283293	2.351211
3	YNG WOMN LDRSHP CHRT	28.8	59.8	-31.02755	1.60	-2.477262	-2.567996
4	NORTHSIDE COLL PREP	100.0	87.8	12.19111	3.11	1.064933	1.065899
5	PERSPECTIVES CHRTR H	23.7	49.2	-25.53578	1.02	-2.057522	-2.104002
6	NORTH LAWDALE CHRTR	8.2	43.0	-34.84521	0.686	-2.765504	-2.898777

Con estas variables calculadas, pueden construirse las visualizaciones para analizar la presencia de atípicos, la linealidad, la homocedasticidad y la normalidad.

7.7.1. Muestra

Este indicador ejerce una gran influencia sobre la significatividad de la relación:

- Con muestras más pequeñas se corre el riesgo de sobreajuste (*overfitting*, resultados no generalizables fuera de esa muestra).
- Con muestras muy grandes, habrá que asegurarse de que, además de resultados estadísticamente significativos, las relaciones obtenidas tienen cierta intensidad, pues existe la tendencia a que pequeños coeficientes resulten significativos estadísticamente de forma artificial.

Las reglas empíricas habituales son:

- 10 a 15 casos por indicador (una regla sencilla que puede simplificar en exceso la complejidad del problema).
- Según Green (1991), para k parámetros (variables independientes):
 - Cuando el foco se pone en el ajuste global del modelo: $50 + (8 \times k)$.
 - Cuando el foco se pone en la significatividad de los parámetros: $104 + k$.
 - Cuando el foco se centra en ambos análisis: el mayor de los anteriores.

7.7.2. Residuos extremos

Un *outlier* no es un caso que deba eliminarse inmediatamente, sino un caso que debe analizarse de manera particular:

- Puede ser un error en la entrada de datos: se modifica si es posible o se elimina si no hay opción.
- Puede ser un valor correcto pero explicable: se analizan las variables que lo explican.
- Puede ser un valor correcto pero inexplicable: se recomienda modelizar con y sin el valor para analizar su efecto.

Los *outliers* en una regresión se analizan sobre los residuos estandarizados o estudentizados. Una de las formas más potentes de detectar casos problemáticos es observar cuántas desviaciones típicas se aleja su residuo de la media (cero). En una distribución normal, es extremadamente raro encontrar casos más allá de 3 desviaciones típicas. Se aplican las siguientes reglas:

- 95 % deben estar entre -1.96 y +1.96.
- 95.45 % deben estar entre -2 y 2.
- 99 % deben estar entre -2.58 y +2.58.
- 99.78 % deben estar entre -3 y 3.

Si esos porcentajes observados son inferiores, existe evidencia de que el nivel de error del modelo no es aceptable (ajusta pobremente). Si un caso supera ± 2.58 , debe ser analizado.

- **Caso atípico:** aquel cuyo residuo estandarizado está fuera del intervalo $[-3, +3]$.
- **Caso a revisar:** aquellos ubicados entre 2 y 3 desviaciones típicas, especialmente si la muestra es pequeña.

```
# Frequency table of absolute standardised residuals
# Head and tail give a view of both low-residual (well-fitted)
#   and high-residual cases
utils::head(expss::fre(abs(school_data$z_resid)))
utils::tail(expss::fre(abs(school_data$z_resid)))
```

Percentage of passing students	Count	Valid percent	Percent	Responses, %	Cumulative
0.00783596844282617	1	1.2	1.2	1.2	
0.0191362365958814	1	1.2	1.2	1.2	
0.0299952853719144	1	1.2	1.2	1.2	
0.0400238883072977	1	1.2	1.2	1.2	

Percentage of passing students	Count	Valid percent	Percent	Responses, %	Cumulative
0.0466790187951101	1	1.2	1.2	1.2	
0.0474318937348801	1	1.2	1.2	1.2	
Percentage of passing students	Count	Valid percent	Percent	Responses, %	Cumulative
2.28329347648761	1	1.2	1.2	1.2	
2.47726205922403	1	1.2	1.2	1.2	
2.7655035970376	1	1.2	1.2	1.2	
4.16992537304343	1	1.2	1.2	1.2	
#Total	80	100	100	100	
<NA>	0		0.0		

7.7.3. Independencia: Durbin-Watson

Este estadístico garantiza que el error de un caso (ej. un instituto) no está condicionado por el error del anterior. En el caso de Chicago, dado que los datos no tienen un orden temporal lógico, es altamente probable que la independencia se cumpla, pero es obligatorio reportarlo.

- **Valor ideal:** 2.0.
- **Rango aceptable:** entre 1.5 y 2.5.

```
# Durbin-Watson test for serial autocorrelation in the residuals
dw_result <- lmtest::dwtest(model_chicago)
print(dw_result)
```

Durbin-Watson test

data: model_chicago

DW = 1.6611, p-value = 0.05996

alternative hypothesis: true autocorrelation is greater than 0

7.7.4. Linealidad

Se posiciona `z_pred` (predichos estandarizados) en el eje X y `z_resid` (residuos estandarizados) en el eje Y.

Resultado visual esperado:

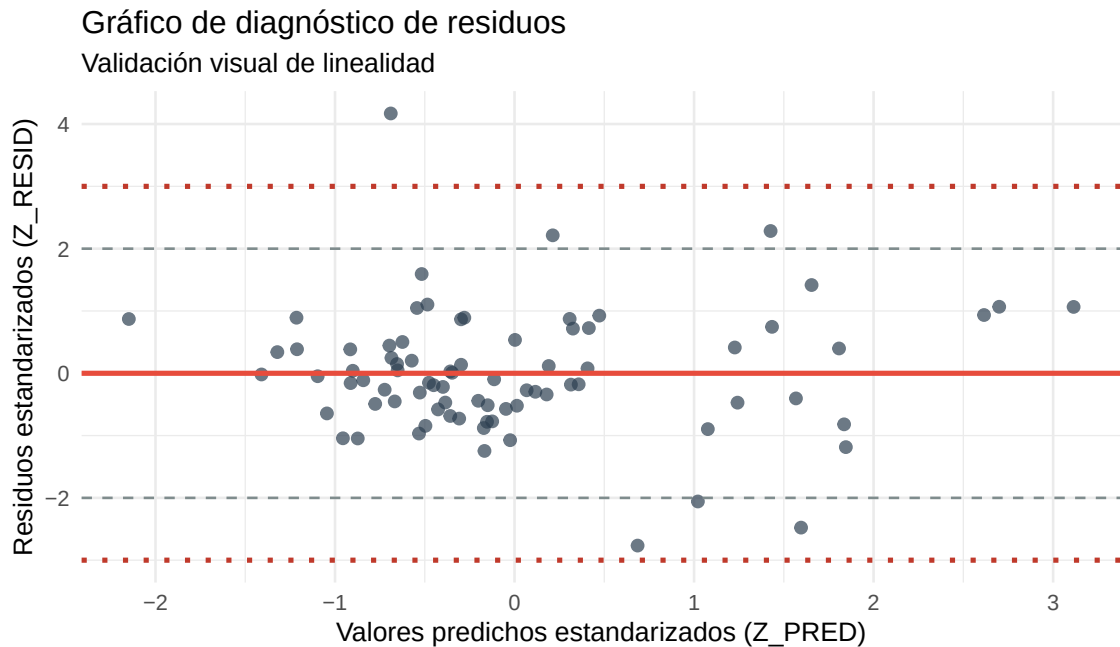
- **Correcto:** una nube de puntos sin forma, como un enjambre de abejas.
- **Incorrecto:** una forma de “U” o de “U invertida” (parábola), que indica que la relación original era curva y se ha forzado una línea recta, violando el supuesto de linealidad.

Al evaluar la linealidad, se observa la **tendencia central** de la nube de puntos de izquierda a derecha:

- **Lo ideal:** si se traza una línea que pase por el centro de la nube, debería ser una línea plana horizontal que coincida con el cero.
- **El problema:** si esa línea central dibuja una “U” o una onda, el modelo está fallando de forma sistemática (subestimando en los extremos y sobreestimando en el medio, o viceversa).

```
# Residual diagnostic plot: ZPRED vs ZRESID
# For linearity: assess the SHAPE of the central trend (should
↪ be flat at zero)
ggplot2::ggplot(school_data, ggplot2::aes(x = z_pred, y =
↪ z_resid)) +
  ggplot2::geom_point(color = "#2c3e50", alpha = 0.7, size = 2)
↪ +
  ggplot2::geom_hline(yintercept = 0, color = "#e74c3c",
↪ linewidth = 1) +
  ggplot2::geom_hline(yintercept = c(-2, 2), color = "#7f8c8d",
↪ linetype = "dashed") +
  ggplot2::geom_hline(yintercept = c(-3, 3), color = "#c0392b",
↪ linetype = "dotted",
                        linewidth = 1) +
  ggplot2::theme_minimal() +
  ggplot2::labs(
    title = "Gráfico de diagnóstico de residuos",
    subtitle = "Validación visual de linealidad",
```

```
x      = "Valores predichos estandarizados (Z_PRED)",
y      = "Residuos estandarizados (Z_RESID)"
)
```



7.7.5. Homocedasticidad

Se utiliza el mismo gráfico ZPRED vs ZRESID, pero ahora el foco cambia: en lugar de observar la tendencia central, se observa la **amplitud vertical** de la nube de puntos a medida que se avanza de izquierda a derecha.

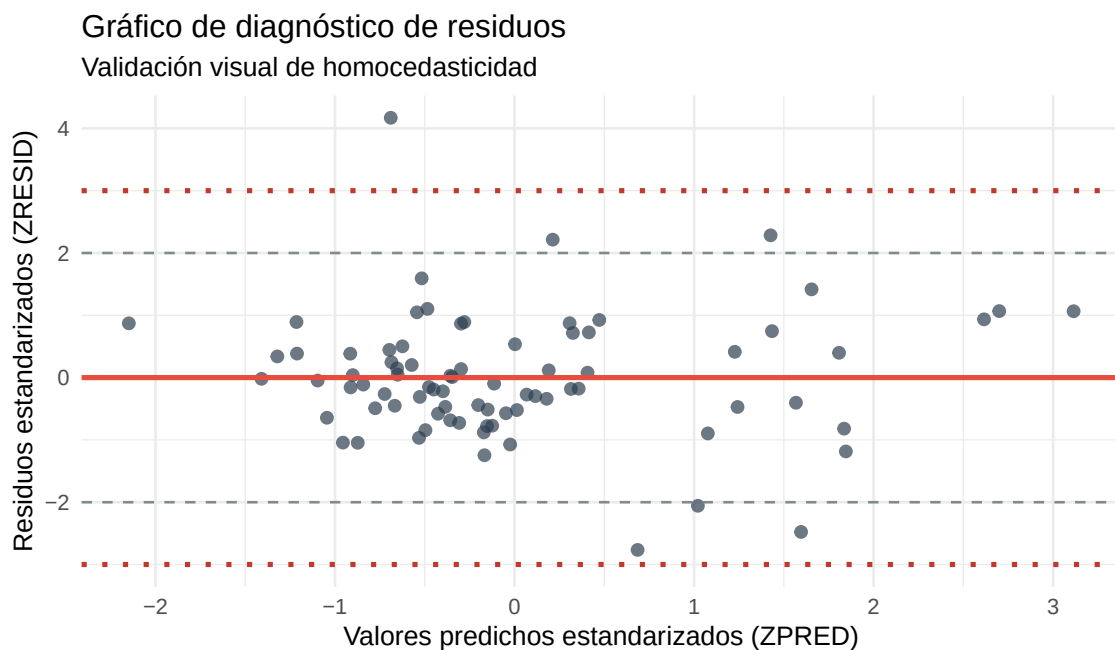
- **Lo ideal:** la nube de puntos debe parecer un “tubo” o un rectángulo de ancho constante. El error es igual de ancho para predicciones bajas que para predicciones altas.
- **El problema:** si la nube comienza muy estrecha y se va abriendo como un embudo o un megáfono, se está ante un escenario de heterocedasticidad. Significa que, para valores bajos, el modelo es muy preciso, pero a medida que los valores crecen, el modelo se vuelve cada vez más impreciso.

```
# Same ZPRED vs ZRESID plot reused with a different interpretive
↪ lens:
```

```

# For homoscedasticity: assess the WIDTH (vertical spread) of
↪ the cloud,
# not its central trend - it should remain constant across all
↪ predicted values
ggplot2::ggplot(school_data, ggplot2::aes(x = z_pred, y =
↪ z_resid)) +
  ggplot2::geom_point(color = "#2c3e50", alpha = 0.7, size = 2)
↪ +
  ggplot2::geom_hline(yintercept = 0, color = "#e74c3c",
↪ linewidth = 1) +
  ggplot2::geom_hline(yintercept = c(-2, 2), color = "#7f8c8d",
↪ linetype = "dashed") +
  ggplot2::geom_hline(yintercept = c(-3, 3), color = "#c0392b",
↪ linetype = "dotted",
                        linewidth = 1) +
ggplot2::theme_minimal() +
ggplot2::labs(
  title = "Gráfico de diagnóstico de residuos",
  subtitle = "Validación visual de homocedasticidad",
  x = "Valores predichos estandarizados (ZPRED)",
  y = "Residuos estandarizados (ZRESID)"
)

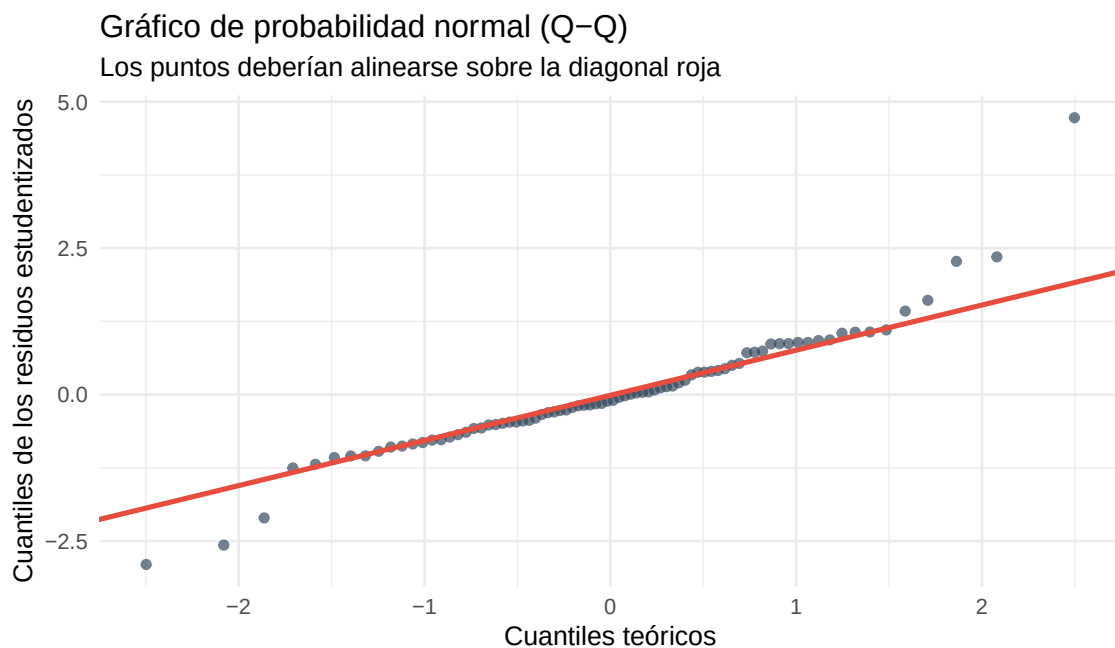
```



7.7.6. Normalidad: gráfico Q-Q de los residuos

Para evaluar si los errores siguen una distribución normal, se enfrentan los cuantiles teóricos de una distribución normal perfecta con los cuantiles reales de los residuos estudentizados:

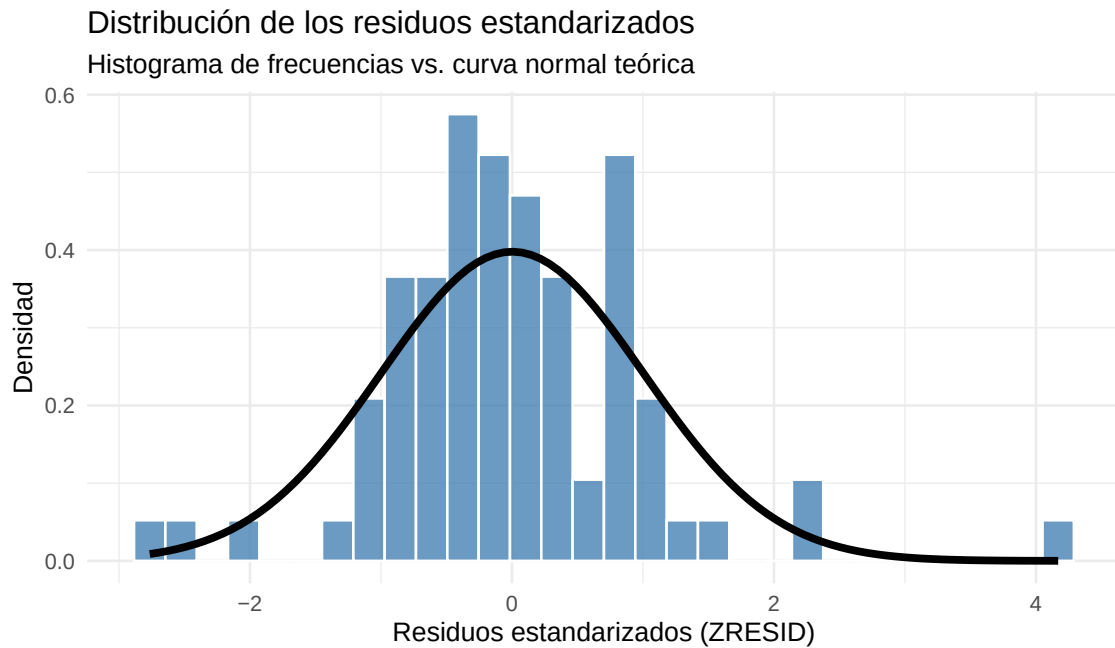
```
# Normal Q-Q plot for residual normality
# Points should lie close to the diagonal; systematic deviations
  ↪ signal non-normality
ggplot2::ggplot(school_data, ggplot2::aes(sample = stud_resid))
  ↪ +
  ggplot2::stat_qq(color = "#34495e", alpha = 0.7) +
  ggplot2::stat_qq_line(color = "#e74c3c", linewidth = 1) +
  ggplot2::theme_minimal() +
  ggplot2::labs(
    title    = "Gráfico de probabilidad normal (Q-Q)",
    subtitle = "Los puntos deberían alinearse sobre la diagonal
      ↪ roja",
    x        = "Cuantiles teóricos",
    y        = "Cuantiles de los residuos estudentizados"
  )
```



Evaluar la normalidad combinando la inspección visual (histograma con curva nor-

mal) con las pruebas analíticas (Shapiro-Wilk y Lilliefors) representa la forma más rigurosa de validar este supuesto:

```
# Visual validation: histogram of standardised residuals with
  ↳ normal curve overlay
# after_stat(density) scales the y-axis to density so the curve
  ↳ fits properly
ggplot2::ggplot(school_data, ggplot2::aes(x = z_resid)) +
  ggplot2::geom_histogram(
    ggplot2::aes(y = ggplot2::after_stat(density)),
    fill = "steelblue",
    color = "white",
    alpha = 0.8
  ) +
  ggplot2::stat_function(
    fun = dnorm,
    args = list(
      mean = mean(school_data$z_resid, na.rm = TRUE),
      sd = sd(school_data$z_resid, na.rm = TRUE)
    ),
    color = "black",
    linewidth = 1.5
  ) +
  ggplot2::theme_minimal() +
  ggplot2::labs(
    title = "Distribución de los residuos estandarizados",
    subtitle = "Histograma de frecuencias vs. curva normal
      ↳ teórica",
    x = "Residuos estandarizados (ZRESID)",
    y = "Densidad"
  )
```



```
# Analytical normality tests on the standardised residuals
# Shapiro-Wilk: most powerful for n < 100; Lilliefors: more
  ↪ conservative
stats::shapiro.test(school_data$z_resid)
nortest::lillie.test(school_data$z_resid)
```

Shapiro-Wilk normality test

```
data: school_data$z_resid
W = 0.93954, p-value = 0.0009098
```

Lilliefors (Kolmogorov-Smirnov) normality test

```
data: school_data$z_resid
D = 0.079702, p-value = 0.2389
```

¿Por qué existe aparente contradicción en los resultados? Se trata de una diferencia de **potencia estadística** entre ambas pruebas, en interacción con el tamaño de la muestra ($n = 80$):

- **Shapiro-Wilk** ($p < 0.05$): indica que se **RECHAZA** la normalidad. Esta prueba es extremadamente sensible y potente, especialmente para muestras menores de 100 casos. Es capaz de detectar desviaciones muy sutiles de la campana ideal.
- **Lilliefors** ($p > 0.05$): indica que **NO se puede RECHAZAR** la normalidad. Incluso con la corrección de Lilliefors, esta prueba es mucho más conservadora y suele pasar por alto desviaciones moderadas.

Al observar el gráfico, el estadístico de Shapiro-Wilk es matemáticamente preciso. La distribución de los residuos **no dibuja una campana perfecta**:

- Se aprecia un pico central muy agudo (apuntamiento o curtosis alta).
- Se observa una ligera asimetría (la cola derecha posee mayor densidad aparente que la izquierda).

Shapiro-Wilk detecta estas “imperfecciones” geométricas, mientras que Lilliefors evalúa que la forma general es lo suficientemente próxima a una normal como para no rechazar la hipótesis.

7.7.7. La decisión práctica

En un entorno de investigación de mercados profesional, el comentario que debe trasladarse al informe es el siguiente:

Aunque la prueba de Shapiro-Wilk (más sensible) detecta desviaciones significativas de la normalidad perfecta ($p = 0.0009$), la prueba de Lilliefors ($p = 0.2389$) y la inspección visual del histograma sugieren que estas desviaciones son moderadas. Dado que el tamaño de la muestra es suficientemente amplio ($n = 80$) y que la estimación por Mínimos Cuadrados Ordinarios es un método robusto frente a violaciones leves del supuesto de normalidad, **se decide dar por válido el modelo y continuar con la interpretación**. No obstante, esta ligera asimetría advierte que el modelo podría ser susceptible de mejora si se identifican y tratan casos atípicos o si se transforma alguna variable.

En la práctica empírica, estas asimetrías suelen estar causadas por aquellos 2 o 3 institutos que presentan un comportamiento anómalo (los *outliers* detectados en pasos anteriores con la distancia de Cook o con residuos superiores a 3 desviaciones).

Allaire, JJ, y Christophe Dervieux. 2024. *quarto: R Interface to 'Quarto' Markdown Publishing System*. <https://CRAN.R-project.org/package=quarto>.