ESTADÍSTICA CON RStudio PARA LA GESTIÓN DE RIESGOS DE DESASTRES

Una guía práctica y reproducible para profesionales, técnicos e investigadores del sector humanitario y de protección civil

R • RStudio • tidyverse • ggplot2 • normalidad • análisis bivariados • Marco de Sendai AFE • Clústeres • Regresión • Índices • sf • tmap • fable • Quarto • Shiny

Gregory Leandro Montenegro Bosquez, Paúl Alejandro Mora Dalgo, Johanna Fernanda Dueñas Durán, Vallejo Ilijama María Tránsito & Bryan Renato Gavilanez Riofrio

CIDPROS
Centro de innovación y desarrollo profesional

ESTADÍSTICA CON RStudio PARA LA GESTIÓN DE RIESGOS DE DESASTRES

Una guía práctica y reproducible para profesionales, técnicos e investigadores del sector humanitario y de protección civil

Gregory Leandro Montenegro Bosquez, Paúl Alejandro Mora Dalgo, Johanna Fernanda Dueñas Durán, Vallejo Ilijama María Tránsito & Bryan Renato Gavilanez Riofrio

Datos bibliográficos

ISBN	978-9907-9562-4-5
Título del libro	Estadística con Rstudio para la gestión de riesgos de desastres
Autores	Gregory Leandro Montenegro Bosquez Paúl Alejandro Mora Dalgo Johanna Fernanda Dueñas Durán Vallejo Ilijama María Tránsito Bryan Renato Gavilanez Riofrio
Editorial	CIDPROS EDITORIAL
Materia	519.5 - Matemáticas estadísticas
Público objetivo	Profesional / académico
Año	2026
Número de edición	1
Tamaño	9.05Mb
Soporte	Libro digital descargable
Formato	PDF (.pdf)
Idioma	Español
DOI	https://doi.org/10.67166/errhk930

Hecho en Ecuador / Made in Ecuador

MSc. Gregory Leandro Montenegro Bosquez

Universidad Estatal de Bolívar
gregoryleandro92@gmail.com
https://orcid.org/0000-0001-9136-3664
San Miguel, Bolívar, Ecuador

Semblanza

Gregory Leandro Montenegro Bosquez es un profesional ecuatoriano nacido el 5 de septiembre de 1992 en San José de Chimbo, provincia de Bolívar. Es Ingeniero Agroindustrial por la Universidad Estatal de Bolívar y ha consolidado una sólida formación de cuarto nivel con tres maestrías: Magíster en Gerencia de la Calidad e Innovación por la Universidad Internacional del Ecuador, Magíster en Estadística Aplicada por la Universidad Politécnica Estatal del Carchi y Maestría en Agroindustrias por la Universidad Nacional de Chimborazo, preparación que refleja su vocación por la excelencia académica, la calidad y la mejora continua.

A lo largo de su trayectoria ha combinado la docencia universitaria con responsabilidades de gestión académica, desempeñándose como Coordinador de Programas de Maestría y Técnico de Apoyo a los Procesos de Posgrado en la Universidad Nacional de Chimborazo. Actualmente es docente de la Universidad Estatal de Bolívar, donde imparte asignaturas vinculadas a la estadística, la metodología de la investigación, la gestión por procesos, la auditoría y la administración. En el ámbito de la consultoría, ha prestado servicios como consultor externo para ZUMASUA Cía. Ltda., asesorando la implementación de sistemas de gestión de calidad e inocuidad alimentaria bajo estándares de Buenas Prácticas de Manufactura (ISO 22002) y principios HACCP, y ha participado como ponente en congresos internacionales de estadística aplicada.

Su vocación investigadora se evidencia en proyectos experimentales sobre riesgos psicosociales, tecnoestrés y salud ocupacional en instituciones de educación superior, así como en múltiples artículos científicos publicados en revistas indexadas sobre calidad, innovación, educación superior, sostenibilidad y procesos organizacionales. Profesional comprometido con la formación continua y el rigor metodológico, encarna valores de responsabilidad, ética y servicio, aportando desde la academia al desarrollo científico y productivo del Ecuador.

Ing. Paúl Alejandro Mora Dalgo MSc.

Universidad Estatal de Bolívar
pmora@mailes.ueb.edu.ec
https://orcid.org/0009-0007-7564-3149
Quito,Pichincha,Ecuador

Semblanza

Profesional en Seguridad, Salud Ocupacional y Gestión de Riesgos, Paúl Alejandro Mora Dalgo nació el 16 de septiembre de 1994 en Quito cuya vida ha estado marcada por su residencia en la Provincia Bolívar Cantón Guaranda, donde su formación fortaleció su compromiso con las comunidades y su comprensión de los desafíos asociados a los riesgos y las emergencias.

Esta experiencia ha orientado su interés por la gestión integral de riesgos, promoviendo la prevención, la preparación, la respuesta y la recuperación como pilares para fortalecer la resiliencia comunitaria y contribuir al bienestar colectivo.

Cuenta con una sólida formación académica: es Ingeniero en Administración para desastres y Gestión de Riesgo desde 2019. Posteriormente obtuvo el título de Magíster en Prevención y Gestión de Riesgo ,Especialista en Gestión y Gobernanza Territorial cursada en el Instituto de Altos Estudios Nacionales del Ecuador IAEN,Además cuenta con multiples cursos,talleres, Seminarios y Capacitaciones en Áreas como Seguridad y Salud Ocupacional, Gestión y Administración de Riesgo,Docencia y Pedagogía Esta preparación refleja su vocación por la excelencia académica y la actualización permanente en las ciencias de la tierra.

En su trayectoria, ha destacado tanto en el ejercicio libre de la profesión como en el servicio público, promoviendo una cultura de Seguridad y Salud e Higiene en el trabajo con enfoque en la protección de la vida, la salud y la dignidad de las personas trabajadoras. Su compromiso profesional se orienta a la implementación de sistemas de gestión en seguridad y salud ocupacional, la identificación y control de riesgos laborales, el fortalecimiento de ambientes de trabajo saludables y el cumplimiento de la normativa vigente

MSc. Johanna Fernanda Dueñas Durán

Universidad Estatal de Bolívar
Johanna.duenas@ueb.edu.ec
https://orcid.org/0009-0002-2442-314X
Guaranda, Bolívar, Ecuador

Semblanza

Johanna Fernanda Dueñas Durán es Ingeniera Geóloga graduada de la Universidad Central del Ecuador (2012) y Magíster en Prevención y Gestión del Riesgo por el Instituto de Altos Estudios Nacionales (2022). Nacida en Ibarra en 1982, actualmente reside en Guaranda, provincia de Bolívar, donde desarrolla su labor docente e investigativa.

Su trayectoria profesional combina más de una década de experiencia técnica en el sector petrolero —donde se desempeñó en Petrokem Logging Services Ltda. en roles de creciente responsabilidad, desde Geóloga de Pozo hasta Coordinadora de Operaciones y Coordinadora de Sistemas de Gestión de Calidad (2008–2023)— con una sólida trayectoria académica en la Universidad Estatal de Bolívar, donde se desempeña como docente desde 2024, impartiendo las cátedras de Geología, Sismología, Vulcanología y Geografía Física en la Carrera de Ingeniería en Riesgos de Desastres.

En el ámbito investigativo, ha liderado y participado en proyectos de investigación de campo sobre gestión de riesgos naturales y resiliencia comunitaria en la Sierra de Guaranda, entre ellos estudios sobre vulnerabilidad de especies endémicas ante incendios forestales, fortalecimiento de capacidades comunitarias para la gestión del ecosistema páramo, y estrategias de mitigación de riesgo de sequía en el sector agropecuario. Asimismo, ha contribuido a investigaciones sobre preparación institucional para la atención a estudiantes con discapacidad y sobre tomografías de resistividad eléctrica en la cuenca del río Coca.

Su producción académica reciente (2024–2025) incluye publicaciones en revistas y editoriales internacionales —entre ellas capítulos en la serie Lecture Notes in Networks and Systems de Springer y en Communications in Computer and Information Science— así como la autoría de libros especializados en movimientos en masa, diseños experimentales aplicados a la gestión de riesgos, e integración de inteligencia artificial en la educación superior. Ha presentado

ponencias en congresos internacionales en Ecuador y México sobre modelos predictivos de precipitación, ecuaciones alométricas forestales y objetivos de desarrollo sostenible.

Su trabajo se distingue por articular la geología aplicada, la gestión de riesgos de desastres y la formación universitaria, con un compromiso sostenido hacia la investigación científica y la vinculación con las comunidades del territorio andino.

Ing. María Tránsito Vallejo Ilijama MSc.

Universidad Estatal de Bolivar
mvallejo@ueb.edu.ec
https://orcid.org/0000-0002-8757-2452
Guaranda, Bolivar, Ecuador

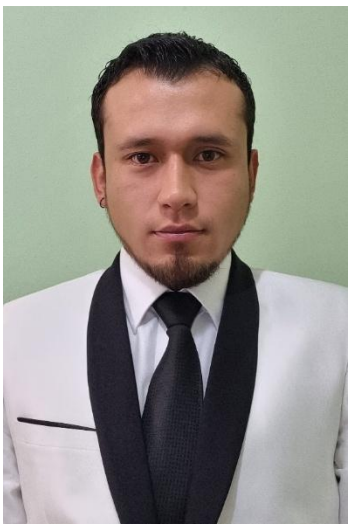
Semblanza

María Vallejo Ilijama, Ingeniera Agrónoma, Magister en Hidráulica mención en Gestión de Recursos Hídricos, Magister en Economía con mención en Gestión Económica de Riesgos de Desastres y Desarrollo Sostenible Universidad Nacional Mayor de San Marcos – UNMSM, Magister en Agroecología y Ambiente, docente a contrato de la Universidad Estatal de Bolívar periodos académico mayo 2015 a la fecha, actualmente **coordinadora de la carrera de Ingeniería en Riesgos de Desastres UEB**; Técnica de campo y responsable del laboratorio de Bioinsumo en el Ministerio de Agricultura Ganadería Acuacultura y Pesca noviembre 2011 a mayo 2015; Técnica agropecuaria en el proyecto Campesino a Campesino de la Cruz Roja Ecuatoriana Junta Provincial Bolívar, septiembre 2006 agosto 2008, Investigadora acreditada por Senescyt, Investigadora en proyectos de investigación en la Universidad Estatal de Bolívar, dirección de proyectos de Vinculación con la Comunidad, dirección y par académico de trabajos de Titulación de Pregrado y posgrado Carrera de Ingeniería en Riesgos de Desastres. Facilitadora en talleres de capacitación en instituciones públicas y privadas en temas ambientales, recurso naturales y productivos. Participación en Congresos Nacionales e Internacionales.

Mgs. Bryan Renato Gavilanez Riofrio

Universidad Estatal de Bolívar
bryan.gavilanez@ueb.edu.ec
https://orcid.org/0009-0009-2654-4026
Guaranda, Bolívar, Ecuador

Semblanza



Bryan Renato Gavilanez Riofrio es un Ingeniero en Administración para Desastres y Gestión de Riesgos nacido el 09 de octubre de 1992 en la ciudad de Guaranda, cuya vida ha estado marcada por su residencia en el cantón Guaranda, creció en un entorno que fortaleció su sensibilidad social y su compromiso con el bienestar colectivo, desarrollando desde temprana edad una profunda empatía por las necesidades de las comunidades y su relación con el entorno. Esta experiencia orientó su vocación hacia la gestión de riesgos, concibiéndole como una disciplina al servicio de la protección de la vida, la reducción de vulnerabilidades y la construcción de comunidades más resilientes, seguras y sostenibles.

Cuenta con una sólida formación académica: es Ingeniero en Administración para Desastres y Gestión de Riesgos graduado de la Universidad Estatal de Bolívar en el 2016. Posteriormente obtuvo los títulos de Magíster en Gestión de Riesgos de la Universidad Internacional del Ecuador, Máster en Prevención, Seguridad y Salud en el Trabajo de la Universidad Antonio de Nebrija España, Magíster en Salud y Seguridad Ocupacional con mención en Prevención de Riesgos Laborales, esta última con honores Summa Cum Laude de la Universidad Hemisferios, además estudios complementarios. Poseo certificaciones como experto en sistemas integrados (ISO 9001, 14001, 45001). Esta preparación refleja su vocación por la excelencia académica y la actualización permanente de conocimientos.

A lo largo de su trayectoria, ha destacado tanto en el ejercicio libre de la profesión como en el servicio público y privado, actualmente se desempeña como Técnico Docente de la carrera de Ingeniería en Riesgos de Desastres de la Facultad de Ciencias de la Salud y del Ser Humano de la Universidad Estatal de Bolívar, se distingue por su capacidad de liderazgo, ética, trabajo en equipo y compromiso con la prevención técnica.

SINOPSIS

Estadística con RStudio para la Gestión de Riesgos de Desastres constituye una guía práctica, técnica y metodológica orientada a fortalecer las competencias analíticas de estudiantes, investigadores, profesionales y tomadores de decisiones vinculados con la gestión del riesgo de desastres. La obra integra los fundamentos de la estadística aplicada con el uso del software R y RStudio, proporcionando un enfoque reproducible que permite transformar datos en evidencia científica para apoyar la planificación, prevención, mitigación, respuesta y recuperación frente a eventos adversos.

A lo largo de sus capítulos, el libro desarrolla de manera progresiva el diseño de investigaciones cuantitativas, la organización y procesamiento de bases de datos, el análisis descriptivo e inferencial, la validación de supuestos estadísticos, el modelamiento espacial y temporal, la construcción de indicadores de riesgo y la elaboración de reportes reproducibles mediante herramientas como R Markdown, Quarto y Shiny. Cada tema combina fundamentos conceptuales con ejemplos aplicados, código comentado y casos relacionados con escenarios reales de gestión de riesgos de desastres, facilitando la comprensión y aplicación inmediata de los procedimientos estadísticos.

Asimismo, la obra promueve el uso de metodologías abiertas y reproducibles, incorporando buenas prácticas para la comunicación de resultados, la visualización de datos, la interpretación adecuada de la evidencia estadística y la toma de decisiones bajo incertidumbre. Su contenido responde a las necesidades actuales de instituciones públicas, organismos de respuesta, academia y organizaciones humanitarias, ofreciendo una herramienta integral para desarrollar investigaciones rigurosas y análisis basados en datos que contribuyan al fortalecimiento de la resiliencia territorial y la reducción del riesgo de desastres.

Palabras clave: Estadística aplicada; RStudio; Gestión de riesgos de desastres; Análisis de datos; Investigación reproducible.

SYNOPSIS

Statistics with RStudio for Disaster Risk Management is a practical, technical, and methodological guide designed to strengthen the analytical skills of students, researchers, professionals, and decision-makers involved in disaster risk management. The book integrates the foundations of applied statistics with the use of R and RStudio, providing a reproducible approach that enables readers to transform data into scientific evidence to support disaster prevention, mitigation, preparedness, response, and recovery processes.

Throughout its chapters, the book progressively develops the principles of quantitative research design, data organization and processing, descriptive and inferential statistical analysis, assumption testing, spatial and temporal modeling, risk indicator construction, and the preparation of reproducible reports using tools such as R Markdown, Quarto, and Shiny. Each topic combines theoretical foundations with practical examples, well-documented R code, and case studies related to real disaster risk management scenarios, facilitating both understanding and immediate application of statistical methods.

In addition, the book promotes the use of open and reproducible research methodologies by incorporating best practices for data visualization, statistical interpretation, scientific reporting, and evidence-based decision-making under conditions of uncertainty. Its content addresses the current needs of public institutions, emergency management agencies, humanitarian organizations, and academia, providing a comprehensive resource for conducting rigorous research and data-driven analyses that contribute to strengthening territorial resilience and reducing disaster risk.

Keywords: Applied Statistics; RStudio; Disaster Risk Management; Data Analysis; Reproducible Research.

ÍNDICE DE CONTENIDOS

PRÓLOGO

La gestión del riesgo de desastres atraviesa un momento de transformación profunda. La disponibilidad de datos históricos de eventos como bases como EM-DAT, DESINVENTAR o los registros de los propios sistemas nacionales de protección civil, ha crecido exponencialmente en las últimas dos décadas. Sin embargo, la capacidad de analizar, interpretar y comunicar esos datos no ha crecido al mismo ritmo. Muchos profesionales del sector como técnicos municipales, coordinadores humanitarios, especialistas en preparación, planificadores territoriales, poseen la experiencia de campo y el conocimiento institucional necesarios para hacer preguntas importantes, pero carecen de las herramientas cuantitativas para responderlas con rigor.

Este libro nace de esa brecha. Su propósito es construir un puente entre el conocimiento técnico de la gestión de riesgos y las metodologías estadísticas que permiten extraer conocimiento accionable de los datos. El vehículo elegido para esa construcción es R y su entorno integrado RStudio: herramientas libres, gratuitas, mantenidas por una comunidad global de científicos y adoptadas por las principales organizaciones internacionales del sector.

A diferencia de los manuales de estadística general, este libro sitúa cada concepto, cada fórmula y cada bloque de código en el contexto específico de la reducción del riesgo de desastres. Los datasets de ejemplo simulan los tipos de variables que se encuentran en evaluaciones rápidas post-desastre, registros de afectaciones o índices de vulnerabilidad municipal. Los ejercicios propuestos responden a preguntas reales: ¿tienen los municipios con Sistema de Alerta Temprana significativamente menos afectados por inundaciones? ¿Qué factores socioeconómicos predicen mejor las pérdidas por sismos? ¿Ha aumentado la frecuencia de eventos hidrometeorológicos en los últimos 30 años?

El libro se divide en dos volúmenes. Este primero, Capítulos I al V, cubre la base metodológica completa: desde la instalación de R y RStudio hasta los análisis bivariados aplicados a desastres, pasando por un capítulo dedicado íntegramente al diagnóstico de normalidad de los datos tema habitualmente omitido en guías aplicadas, pero de importancia crítica en la práctica estadística cotidiana. El segundo volumen abordará los análisis multivariados, la construcción de índices de riesgo, el análisis espacial, las series temporales y la comunicación de resultados mediante reportes reproducibles.

Cada sección de este libro incluye: exposición conceptual con la teoría necesaria y suficiente; código R completo, comentado línea a línea y listo para ejecutarse; tablas de resultados interpretadas en el lenguaje propio del sector; recuadros de alerta sobre los errores más frecuentes; recuadros de buenas prácticas; y figuras generadas con el estilo visual de ggplot2, el estándar de facto de la visualización científica en R.

Este libro es para quien quiere hacer estadística, no solo para quien quiere saber que existe. Cada capítulo es una invitación a abrir RStudio, cargar los datos y ejecutar el código. La estadística no se aprende leyendo: se aprende haciendo.

CAPÍTULO

1

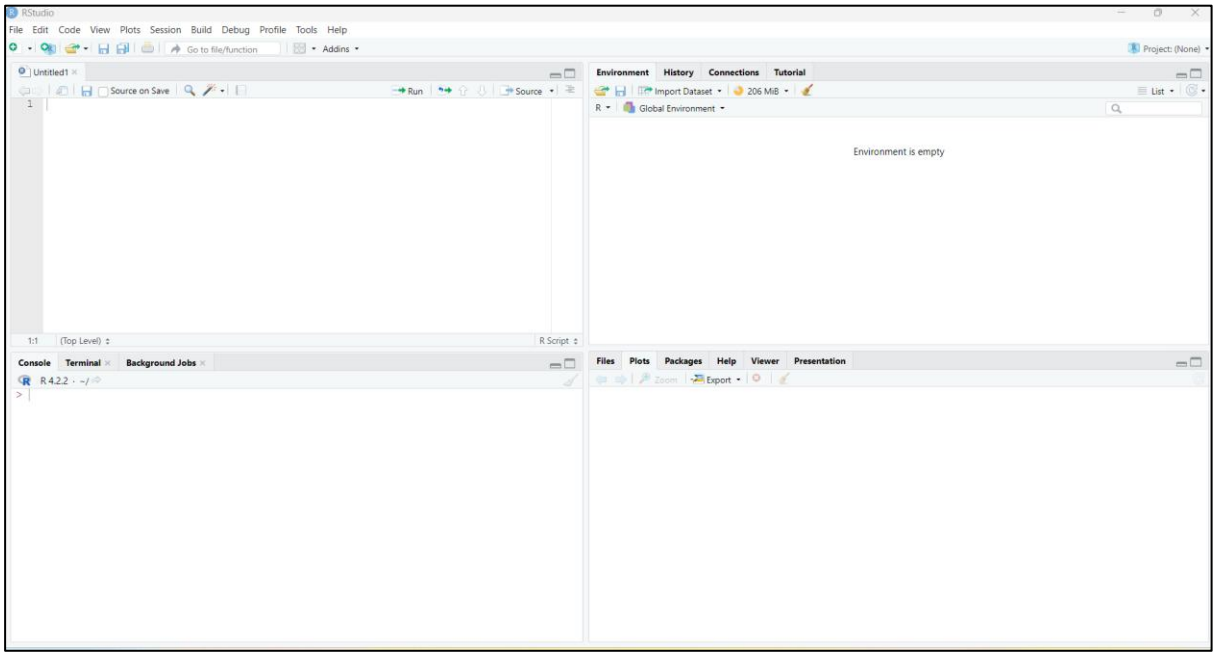
DISEÑO DE INVESTIGACIONES CUANTITATIVAS Y HERRAMIENTAS COMPUTACIONALES PARA LA GESTIÓN DE RIESGOS

1.1. ¿Qué es R y por qué usarlo en gestión de riesgos?

R es un lenguaje de programación estadística de código abierto desarrollado a principios de la década de 1990 en la Universidad de Auckland, Nueva Zelanda, por Ross Ihaka y Robert Gentleman como implementación libre del lenguaje S. Su filosofía central es ser un entorno completamente integrado para el análisis estadístico, la visualización científica y la investigación reproducible. Actualmente es mantenido por el R Core Team, una colaboración internacional de estadísticos y científicos de datos, y distribuido gratuitamente bajo licencia GNU GPL.

Figura 1

Los cuatro paneles de RStudio



Nota. Editor de código (superior izq.), Consola (inferior izq.), Entorno/Historial (superior der.) y Archivos/Gráficos/Ayuda (inferior der.).

Desde su creación, R ha crecido hasta convertirse en uno de los lenguajes más utilizados en ciencia de datos, estadística aplicada y análisis cuantitativo en disciplinas que van desde la epidemiología y la genómica hasta las ciencias climáticas y la economía del desarrollo. Hoy cuenta con más de 20.000 paquetes disponibles en CRAN (Comprehensive R Archive Network), con miles de contribuciones adicionales en GitHub y Bioconductor, que cubren prácticamente cualquier necesidad analítica imaginable.

Para los profesionales de la gestión de riesgos de desastres (GRD), R representa un cambio de paradigma respecto a las herramientas más tradicionales. A continuación, se detallan las ventajas más relevantes para el contexto específico del sector:

Reproducibilidad total

Cada análisis realizado en R queda documentado en un script que puede reejecutarse exactamente obteniendo los mismos resultados. Esto es fundamental para la auditoría técnica de evaluaciones de daños, para la replicabilidad científica de estudios sobre vulnerabilidad, y para la transparencia institucional en la comunicación de riesgos a autoridades y comunidades. Un analista que abandone una institución puede dejar todos sus análisis documentados y replicables por su sucesor sin ninguna pérdida de información metodológica.

Análisis espacial de primer nivel

Los paquetes sf (Simple Features), terra y tmap han convertido a R en una plataforma de análisis geoespacial comparable a un SIG especializado. Permiten cruzar capas de amenaza, exposición y vulnerabilidad, calcular intersecciones, generar mapas coropléticos de riesgo y exportar resultados en formatos estándar (GeoJSON, GeoPackage, shapefile) —todo dentro del mismo entorno donde se realizan los análisis estadísticos, eliminando el flujo de trabajo en múltiples aplicaciones.

Automatización de reportes periódicos

Con R Markdown y Quarto, los informes técnicos se convierten en documentos vivos: el mismo archivo que contiene el texto del informe contiene el código que genera las tablas y gráficos. Cuando llegan datos actualizados, un nuevo reporte mensual de eventos, los resultados de una encuesta post-desastre, basta con recompilar el documento para que todas las cifras, tablas y visualizaciones se actualicen automáticamente. En operaciones de respuesta donde los boletines de situación se producen diariamente, esto representa un ahorro enorme de tiempo y una eliminación casi total de los errores de trascripción.

Comunidad científica global activa

La comunidad de usuarios de R en ciencias del clima, epidemiología de emergencias y ciencias sociales aplicadas genera permanentemente metodologías y paquetes adaptables a GRD. Las preguntas sobre manejo de datos de desastres, análisis de series temporales de eventos extremos

o construcción de índices de vulnerabilidad tienen respuesta en Stack Overflow, en la comunidad R en español y en foros especializados.

Costo cero

En contextos institucionales con presupuestos ajustados como agencias nacionales de gestión de riesgos, municipios con capacidad técnica limitada o ONG humanitarias en campo la gratuidad de R y RStudio es un argumento decisivo. No existen licencias por usuario, por módulo ni por versión. Las actualizaciones son gratuitas y frecuentes.

> **📖 Dato clave**
>
> Según el Stack Overflow Developer Survey 2024, R se mantiene entre los diez lenguajes más usados en ciencia de datos. Organizaciones como el IPCC, la OMS, OCHA, el Banco Mundial y múltiples universidades líderes en riesgo de desastres publican sus análisis cuantitativos en R, con código disponible públicamente.

1.2. Instalación de R y RStudio

La instalación de R y RStudio sigue siempre el mismo orden obligatorio: primero R (el motor de cálculo) y luego RStudio (la interfaz gráfica). RStudio no puede funcionar sin R instalado. Ambos son completamente gratuitos y están disponibles para Windows, macOS y las principales distribuciones de Linux.

RStudio Desktop es desarrollado y mantenido por Posit PBC (antes RStudio PBC). Existe también una versión de servidor (RStudio Server) que permite usar RStudio desde el navegador web, lo que es especialmente útil en entornos institucionales o servidores compartidos de análisis de datos.

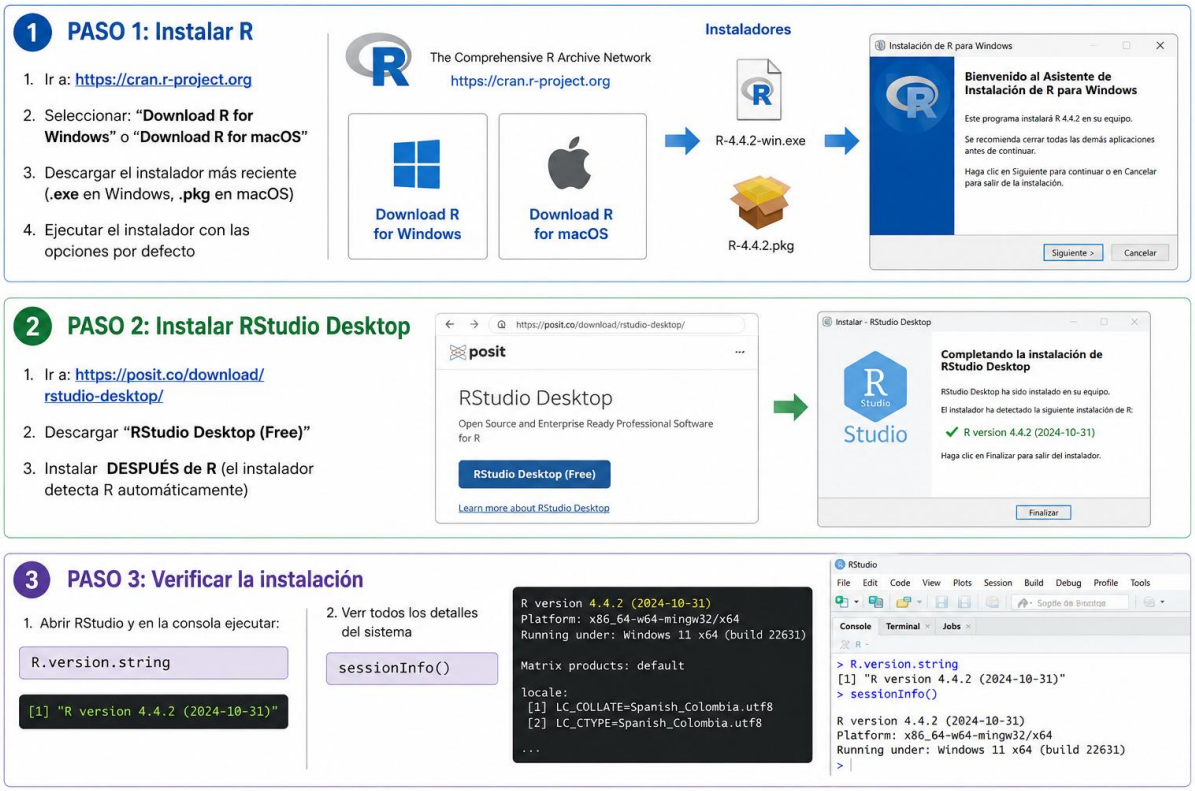
Tabla 1

Requisitos mínimos del sistema

Sistema operativo	Versión mínima	RAM recomendada	Espacio en disco
Windows	10 / 11	4 GB (8 GB recomendados)	500 MB
macOS	11.0 (Big Sur)	4 GB (8 GB recomendados)	500 MB
Ubuntu/Debian	20.04 LTS o superior	4 GB (8 GB recomendados)	500 MB
CentOS/RHEL	7 o superior	4 GB (8 GB recomendados)	500 MB

Figura 2

Instalación en Windows y macOS



Información textual para ejecución de comandos:

```
# ── PASO 1: Instalar R ──────────────────────────────────────────────
# Ir a: https://cran.r-project.org
# Seleccionar: "Download R for Windows" o "Download R for macOS"
# Descargar el instalador más reciente (.exe en Windows, .pkg en macOS)
# Ejecutar el instalador con las opciones por defecto

# ── PASO 2: Instalar RStudio Desktop ────────────────────────────────
# Ir a: https://posit.co/download/rstudio-desktop/
# Descargar "RStudio Desktop (Free)"
# Instalar DESPUÉS de R (el instalador detecta R automáticamente)

# ── PASO 3: Verificar la instalación ────────────────────────────────
# Abrir RStudio y en la consola ejecutar:
R.version.string
# [1] "R version 4.4.2 (2024-10-31)"

# Ver todos los detalles del sistema
sessionInfo()
# R version 4.4.2 (2024-10-31)
# Platform: x86_64-w64-mingw32/x64
# Running under: Windows 11 x64 (build 22631)
```

Instalación en Ubuntu / Debian Linux:

```
# Añadir repositorio CRAN oficial para Ubuntu
sudo apt update
sudo apt install --no-install-recommends r-base r-base-dev

# Instalar dependencias del sistema para compilar paquetes
sudo apt install -y libcurl4-openssl-dev libssl-dev libxml2-dev \
                    libfontconfig1-dev libharfbuzz-dev libfribidi-dev

# Instalar RStudio Server (acceso desde navegador)
# Descargar .deb desde: https://posit.co/download/rstudio-server/
sudo gdebi rstudio-server-2024.xx.x-xxx-amd64.deb

# Verificar servicio
sudo systemctl status rstudio-server
```

Buena Práctica

En entornos institucionales con múltiples analistas, RStudio Server sobre Ubuntu LTS es la solución más eficiente: todos comparten el mismo R, los mismos paquetes instalados y pueden acceder desde cualquier navegador web sin instalación local.

1.3. Interfaz de RStudio: paneles y herramientas

RStudio organiza el espacio de trabajo en cuatro paneles configurables. Cada panel tiene múltiples pestañas y puede reposicionarse según las preferencias del usuario. Conocer bien la función de cada panel y sus atajos de teclado permite trabajar de manera significativamente más eficiente, especialmente en sesiones largas de análisis de datos.

Tabla 2

Descripción de los paneles de la interfaz de RStudio y su funcionalidad

Panel	Ubicación por defecto	Pestañas principales	Función en flujo de trabajo GRD
Editor de código (Source)	Superior izquierda	Scripts, R Markdown, Quarto	Escritura, edición y ejecución de scripts de análisis. Resaltado de sintaxis, autocompletado, búsqueda/reemplazo con regex, ejecución por selección o línea.
Consola / Terminal	Inferior izquierda	Console, Terminal, Jobs	Ejecución interactiva de comandos. Terminal bash para instalar dependencias del sistema. Jobs para ejecutar scripts en segundo plano sin bloquear la consola.

Entorno / Historial / Conexiones	Superior derecha	Environment, History, Connections, Tutorial	Lista de objetos en memoria con tipo y dimensiones. Historial completo de comandos. Conexiones a bases de datos. Tutoriales interactivos del paquete learnr.
Archivos / Gráficos / Paquetes / Ayuda	Inferior derecha	Files, Plots, Packages, Help, Viewer	Navegador de archivos del proyecto. Visualización y exportación de gráficos. Gestor de paquetes instalados. Documentación de funciones en HTML. Previsualización de documentos R Markdown.

Tabla 3

Atajos de teclado esenciales

Acción	Windows/Linux	macOS
Ejecutar línea/selección	Ctrl + Enter	Cmd + Return
Ejecutar script completo	Ctrl + Shift + Enter	Cmd + Shift + Return
Insertar operador pipe |>	Ctrl + Shift + M	Cmd + Shift + M
Insertar operador asignación <-	Alt + -	Option + -
Comentar/descomentar líneas	Ctrl + Shift + C	Cmd + Shift + C
Buscar en archivos del proyecto	Ctrl + Shift + F	Cmd + Shift + F
Autocompletar función/objeto	Tab	Tab
Ver ayuda de función	F1 sobre la función	F1 sobre la función
Limpiar la consola	Ctrl + L	Ctrl + L
Compilar documento R Markdown	Ctrl + Shift + K	Cmd + Shift + K

1.4. Proyectos de RStudio (.Rproj) y estructura de carpetas

Una de las mejores prácticas más importantes al trabajar con R es organizar cada análisis como un Proyecto de RStudio. Un proyecto es simplemente un directorio con un archivo .Rproj que le dice a RStudio: "este es el directorio raíz de trabajo". Al abrir un proyecto, RStudio establece automáticamente el directorio de trabajo (`setwd()`) en la carpeta del proyecto, lo que hace que los scripts sean portables: funcionan en cualquier computador, sin importar la ruta absoluta.

En el contexto de la gestión de riesgos, donde los análisis se realizan en múltiples equipos (laptop del analista, servidor institucional, computadora del colaborador en campo), esta portabilidad es crítica. Un script que usa `read_csv("datos/eventos.csv")` funcionará en cualquier máquina donde el proyecto esté instalado. Uno que usa `read_csv("C:/Users/Ana/Documents/mis_analisis_2024/datos/eventos.csv")` fallará en cualquier otra.

```r
# Crear un nuevo proyecto desde RStudio:
# File → New Project → New Directory → New Project
# O desde la consola:
usethis::create_project("riesgo_andino_2024")

# ── Estructura recomendada de carpetas ──────────────────────────
# riesgo_andino_2024/
# ├── riesgo_andino_2024.Rproj   ← archivo de proyecto (no editar)
# ├── README.md                  ← descripción del proyecto
# ├── datos/
# │   ├── crudos/                ← datos originales, NUNCA modificar
# │   │   ├── emdat_2024.xlsx
# │   │   └── municipios_desinventar.csv
# │   └── procesados/            ← datasets limpios generados por R
# │       └── eventos_limpio.rds
# ├── scripts/
# │   ├── 01_importacion.R       ← cargar y guardar datos crudos
# │   ├── 02_limpieza.R          ← limpiar y transformar
# │   ├── 03_descriptivos.R      ← análisis exploratorio
# │   ├── 04_normalidad.R        ← diagnóstico de distribuciones
# │   ├── 05_bivariados.R        ← análisis comparativos
# │   └── funciones_utiles.R     ← funciones propias reutilizables
# ├── resultados/
# │   ├── tablas/
# │   └── graficos/
# └── informes/
#     ├── informe_preliminary.Rmd
#     └── informe_final.qmd
```

✅ Buena Práctica

Nunca guarde los datos originales dentro de la subcarpeta procesados/ ni sobreescriba los archivos en crudos/. Los datos originales son sagrados. Todos los cambios deben generarse programáticamente desde los scripts, de modo que siempre sea posible reconstruir el análisis completo desde cero.

1.5. Conceptos fundamentales: objetos, clases y funciones

En R todo es un objeto: un número, un vector, una tabla de datos, un modelo estadístico, un gráfico o incluso una función. Los objetos tienen una clase (class) que determina cómo se comportan cuando se aplican funciones sobre ellos. Las **funciones** son las operaciones que transforman objetos de entrada en objetos de salida. Comprender esta estructura tripartita es la clave para leer, modificar y depurar cualquier script de R.

Los tipos de datos fundamentales:

```r
# ── ESCALARES: un solo valor ──────────────────────────────────────
n_eventos_anio <- 142L          # integer (entero, sufijo L)
pct_afectados  <- 0.347         # double (número real)
tiene_sat      <- TRUE          # logical (lógico)
nombre_mun     <- "San Marcos"  # character (texto)

# Verificar la clase de cualquier objeto
class(n_eventos_anio)   # [1] "integer"
class(pct_afectados)    # [1] "numeric" (double)
class(tiene_sat)        # [1] "logical"
typeof(nombre_mun)      # [1] "character"

# ── VECTORES: colección ordenada de valores del mismo tipo ────────
# Vector numérico: afectados mensuales por inundaciones (año 2023)
afectados_mens <- c(230, 1450, 89, 3200, 670, 45,
                    1100, 2800, 340, 890, 1200, 560)

# Acceder a elementos por posición (índice empieza en 1)
afectados_mens[1]    # [1] 230  (enero)
afectados_mens[c(3,6,9)]  # Mar, Jun, Sep
afectados_mens[afectados_mens > 1000]  # filtrar: > 1000

# ── FACTOR: variable categórica con niveles fijos ─────────────────
nivel_riesgo <- factor(
  c("alto","medio","bajo","alto","muy_alto","medio"),
  levels  = c("bajo","medio","alto","muy_alto"),
  ordered = TRUE  # factor ordinal
)
levels(nivel_riesgo)   # [1] "bajo" "medio" "alto" "muy_alto"
table(nivel_riesgo)    # frecuencias por nivel

# ── DATA FRAME: tabla bidimensional (filas = obs, cols = variables) ─
eventos <- data.frame(
  municipio  = c("San Marcos","El Progreso","Lima Norte"),
  tipo       = c("inundacion","sequia","sismo"),
  afectados  = c(230L, 1450L, 3200L),
  tiene_sat  = c(TRUE, FALSE, TRUE),
  idh        = c(0.62, 0.55, 0.71)
)

# Inspeccionar la estructura
str(eventos)
# 'data.frame': 3 obs. of 5 variables:
#  $ municipio : chr  'San Marcos' 'El Progreso' 'Lima Norte'
#  $ tipo      : chr  'inundacion' 'sequia' 'sismo'
#  $ afectados : int  230 1450 3200
#  $ tiene_sat : logi TRUE FALSE TRUE
#  $ idh       : num  0.62 0.55 0.71

# Acceder a columnas
eventos$afectados         # vector con los afectados
eventos[["municipio"]]    # equivalente, más robusto en funciones
eventos[1, ]              # primera fila completa
eventos[eventos$tiene_sat == TRUE, ]  # filtrar con condición
```

El tibble: la versión mejorada del data.frame:

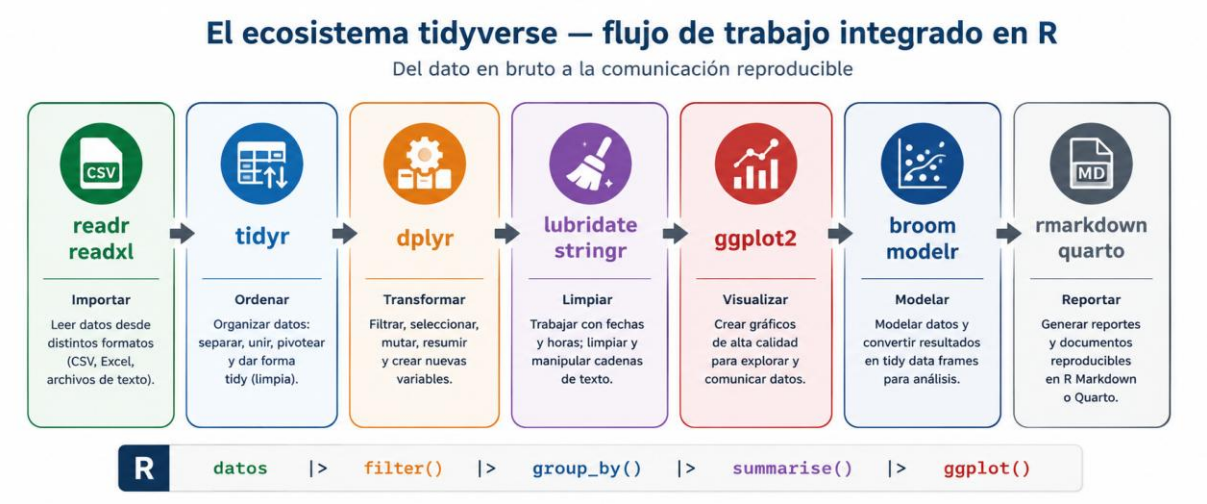
```
library(tibble)

# Un tibble se comporta igual que un data.frame pero:
# - Imprime solo las primeras 10 filas y las columnas que caben
# - Muestra el tipo de cada columna bajo el nombre
# - Nunca convierte strings a factors silenciosamente
# - Da mensajes de error más claros

eventos_tbl <- tibble(
  municipio  = c("San Marcos","El Progreso","Lima Norte"),
  afectados  = c(230L, 1450L, 3200L),
  idh        = c(0.62, 0.55, 0.71)
)

print(eventos_tbl)
# A tibble: 3 x 3
#   municipio   afectados   idh
#   <chr>           <int> <dbl>
# 1 San Marcos        230  0.62
# 2 El Progreso      1450  0.55
# 3 Lima Norte       3200  0.71

# Convertir un data.frame existente a tibble
as_tibble(eventos)

# glimpse(): resumen compacto con tipos y primeros valores
glimpse(eventos_tbl)
```

1.6. El ecosistema tidyverse: filosofía y paquetes

El tidyverse es una colección de nueve paquetes centrales de R diseñados con una filosofía unificada y una sintaxis coherente. Fue creado principalmente por Hadley Wickham y es actualmente mantenido por el equipo de Posit PBC. Su principio organizador es el concepto de datos ordenados (tidy data): cada fila es una observación, cada columna es una variable, cada celda contiene un único valor. Esta estructura —aparentemente simple— transforma radicalmente la eficiencia del análisis estadístico porque todas las herramientas del tidyverse asumen y producen datos en ese formato.

Figura 3

El ecosistema tidyverse



Nota. Flujo de trabajo integrado desde la importación de datos hasta la generación de reportes reproducibles. El operador pipe |> encadena las operaciones de izquierda a derecha.

Cada paquete del tidyverse tiene una responsabilidad específica y bien definida en el flujo de análisis de datos. A continuación, se describen en detalle:

ggplot2 — Visualización de datos

ggplot2 implementa la "Gramática de los Gráficos" de Leland Wilkinson, una teoría que describe los gráficos estadísticos como composición de capas: datos, estéticas (qué variable va en qué eje, qué representa el color o el tamaño), geometría (puntos, barras, líneas), escalas, facetas y tema. Esta filosofía hace que la creación de gráficos sea altamente sistemática: una vez aprendida la lógica, todos los tipos de gráfico se construyen con la misma estructura.

- Uso en GRD: histogramas de distribución de afectados, mapas de puntos georreferenciados, series *temporales* de eventos, gráficos de correlación entre IDH y daños.
- Extensiones clave: ggrepel (etiquetas sin solapamiento), patchwork (combinar múltiples gráficos), ggstatsplot (gráficos con pruebas estadísticas integradas), tmap (mapas temáticos).

dplyr — Manipulación de tablas

dplyr proporciona una gramática de la manipulación de datos basada en cinco verbos principales: filter() (filtrar filas), select() (seleccionar columnas), mutate() (crear/modificar variables), summarise() (resumir en estadísticos) y arrange() (ordenar). Todos los verbos se pueden combinar

con group_by() para operar por grupos. Internamente usa C++ para operar con alta eficiencia incluso sobre millones de filas.

- Uso en GRD: filtrar eventos por año y región, calcular totales de afectados por departamento, crear variables derivadas como el índice de impacto relativo.
- Complemento clave: tidyr para pivotar entre formatos ancho y largo, stringr para limpiar nombres de municipios, lubridate para extraer componentes de fechas.

readr y readxl — Importación de datos

readr importa archivos CSV, TSV y texto delimitado hasta 10 veces más rápido que read.csv() base, detecta automáticamente los tipos de columna y da mensajes de error claros sobre problemas de parsing. readxl lee archivos Excel (.xlsx y .xls) sin depender de Java ni Excel instalado.

tibble — Data frames mejorados

tibble provee la clase tbl_df, una versión del data.frame con comportamiento más predecible: no convierte strings a factors, no recorta nombres de columnas, imprime de forma legible en consola y da errores claros cuando se accede a columnas inexistentes.

stringr — Manipulación de texto

stringr ofrece funciones consistentes para buscar, extraer, reemplazar y dividir cadenas de texto. Es imprescindible para limpiar nombres de municipios con acentos o mayúsculas inconsistentes, parsear códigos geográficos y estandarizar categorías de tipo de evento.

forcats — Manejo de factores

forcats facilita el manejo de variables categóricas: reordenar niveles de un factor para gráficos, agrupar categorías poco frecuentes en "Otros", renombrar niveles. Muy útil cuando se trabaja con tipo de evento, nivel de daño o zona geográfica.

lubridate — Fechas y tiempos

lubridate facilita el parsing de fechas en múltiples formatos internacionales, la aritmética de fechas, y la extracción de componentes (año, mes, día, semana). Imprescindible para el análisis de series temporales de eventos y para calcular duración de emergencias.

purrr — Programación funcional

purrr permite aplicar funciones sobre listas y vectores de forma sistemática usando map(), reemplazando los bucles for con código más legible y seguro. Es especialmente valioso para procesar múltiples archivos CSV de distintas regiones o ejecutar el mismo análisis para múltiples subconjuntos de datos.

```
# Instalar el tidyverse completo (una sola vez)
install.packages("tidyverse")

# Cargar en cada sesión de trabajo
library(tidyverse)

# Al cargar, se muestran los paquetes incluidos y sus versiones:
# ✔ ggplot2   3.5.1     ✔ purrr   1.0.2
# ✔ tibble    3.2.1     ✔ dplyr   1.1.4
# ✔ tidyr     1.3.1     ✔ stringr 1.5.1
# ✔ readr     2.1.5     ✔ forcats 1.0.0
# ✔ lubridate 1.9.3

# Verificar la versión de un paquete específico
packageVersion("ggplot2")  # [1] "3.5.1"

# Ver funciones disponibles en un paquete
ls("package:dplyr") |> head(20)
```

1.7. Importar y exportar datos en múltiples formatos

Los datos de gestión de riesgos llegan en formatos muy variados: CSV exportados de bases institucionales, Excel de evaluaciones de daños, JSON de APIs de alertas tempranas, archivos de SPSS o Stata de encuestas previas, o shapefiles de sistemas GIS. R tiene paquetes específicos para cada formato, y es importante conocer las diferencias entre ellos para evitar problemas de encoding, separadores decimales o tipos de datos.

CSV y archivos de texto delimitado:

```
library(readr)

# Importar CSV con locale latinoamericano (punto y coma, coma decimal)
danos_mx <- read_csv2("datos/crudos/evaluacion_danos_2023.csv",
  locale = locale(
    encoding     = "UTF-8",
    decimal_mark = ",",
```

```
    grouping_mark = "."
  ),
  col_types = cols(
    municipio  = col_character(),
    cod_mun    = col_character(),  # mantener como texto
    afectados  = col_integer(),
    fecha      = col_date(format = "%d/%m/%Y")
  ),
  na = c("", "N/D", "n/d", "--", "NA", "nd")
)

# Importar CSV estándar (coma como separador, punto decimal)
emdat_alc <- read_csv("datos/crudos/emdat_2024_alc.csv",
  skip  = 6,    # saltar encabezados informativos de EM-DAT
  guess_max = 10000  # muestrar más filas para inferir tipos
)

# Diagnóstico de problemas de parsing
problems(emdat_alc)  # muestra filas con errores de lectura
```

Excel (.xlsx y .xls):

```
library(readxl)

# Ver todas las hojas del archivo antes de importar
excel_sheets("datos/crudos/indice_vulnerabilidad_nacional.xlsx")
# [1] "municipios" "departamentos" "metadatos" "fuentes"

# Importar una hoja específica con configuración detallada
vuln_mun <- read_excel(
  path      = "datos/crudos/indice_vulnerabilidad_nacional.xlsx",
  sheet     = "municipios",
  range     = "A4:R250",     # rango de celdas específico
  col_names = TRUE,
  na        = c("", "N/D", "Sin datos", "--")
)

# Importar TODAS las hojas y apilarlas en una sola tabla
library(purrr)
hojas <- excel_sheets("datos/crudos/reportes_regionales.xlsx")
todas_regiones <- map_dfr(hojas,
  ~ read_excel("datos/crudos/reportes_regionales.xlsx",
               sheet = .x)
)
```

Múltiples archivos en una carpeta (patrón batch):

```
# Procesar todos los CSV de reportes mensuales de una carpeta
library(purrr)

archivos <- list.files(
  path    = "datos/crudos/reportes_mensuales/",
```

```
  pattern = "\\.csv$",   # solo archivos .csv
  full.names = TRUE         # incluir ruta completa
)

# Leer y apilar en un solo data frame
# .id agrega columna con el nombre del archivo origen
todos_reportes <- map_dfr(
  archivos,
  ~ read_csv(.x, show_col_types = FALSE),
  .id = "archivo_origen"
)

# Ver cuántos registros vienen de cada archivo
count(todos_reportes, archivo_origen)
```

Formatos nativos de R para análisis grandes:

```
# .rds: guardar UN solo objeto R (más eficiente que CSV)
saveRDS(eventos_limpio, "datos/procesados/eventos_limpio.rds")
eventos <- readRDS("datos/procesados/eventos_limpio.rds")

# .RData o .rda: guardar MÚLTIPLES objetos en un archivo
save(eventos, municipios, indice_vuln,
     file = "datos/procesados/workspace_analisis.RData")
load("datos/procesados/workspace_analisis.RData")

# Ventaja: .rds es 3-10 veces más rápido que CSV para leer/escribir
# y preserva todos los tipos (factors, fechas, tibbles)
```

Exportar resultados:

```
library(writexl)

# Exportar a CSV
write_csv(danos_procesados, "resultados/tablas/danos_procesados.csv")

# Exportar múltiples tablas a un Excel con múltiples hojas
write_xlsx(
  list(
    resumen_nacional   = resumen_nal,
    detalle_municipios = detalle_mun,
    metodologia        = notas_metodo
  ),
  path = "resultados/tablas/informe_danos_final.xlsx"
)
```

1.8. El operador pipe: |> y %>%

El operador pipe es una de las características más transformadoras de R moderno. Su función es encadenar operaciones: toma el resultado de la expresión a su izquierda y lo pasa como primer argumento de la función a su derecha. El efecto es que el código se lee de izquierda a derecha y de arriba hacia abajo, siguiendo la secuencia lógica del análisis, en lugar de anidarse en capas de paréntesis difíciles de leer y depurar.

Existen dos versiones del pipe en R. El pipe nativo `|>` fue introducido en R 4.1.0 (2021) y no requiere cargar ningún paquete. El pipe de magrittr **`%>%`** fue el original, introducido por el paquete magrittr y popularizado por dplyr. Funcionalmente son casi idénticos para el 95% de los casos de uso; el pipe nativo |> es la recomendación actual para código nuevo.

```
# ── PROBLEMA: código sin pipe — difícil de leer ──────────────────
resultado <- arrange(
  summarise(
    group_by(
      filter(eventos, anio >= 2015, !is.na(afectados)),
      tipo_evento, region
    ),
    total_afect = sum(afectados, na.rm = TRUE),
    n_eventos   = n()
  ),
  desc(total_afect)
)

# ── SOLUCIÓN: con pipe |> — se lee como un proceso ──────────────
resultado <- eventos |>
  filter(anio >= 2015, !is.na(afectados)) |>
  group_by(tipo_evento, region) |>
  summarise(
    total_afect = sum(afectados, na.rm = TRUE),
    n_eventos   = n(),
    .groups = "drop"
  ) |>
  arrange(desc(total_afect))

# Leer en voz alta:
# "Toma eventos, filtra los de 2015 en adelante sin NA,
#  agrupa por tipo y región, resume sumando afectados y contando,
#  y ordena de mayor a menor total."

# ── ACTIVAR el atajo Ctrl+Shift+M para el pipe nativo ────────────
# Tools → Global Options → Code → Use native pipe operator |>

# ── DIFERENCIA ENTRE |> Y %>% ────────────────────────────────────
# |>  (nativo R 4.1+): requiere que el argumento sea el PRIMERO
# %>% (magrittr):     usa . como placeholder para cualquier posición
```

```r
# Con %>% se puede hacer:
c(1:100) %>% mean(na.rm = TRUE)
lm(afectados ~ idh, data = .) %>% summary()  # . = objeto anterior

# Con |> el placeholder es _  (R 4.2+)
c(1:100) |> mean(na.rm = TRUE)
lm(afectados ~ idh, data = eventos) |> summary()
```

> **Buena Práctica**
>
> Configure RStudio para usar el pipe nativo: Tools → Global Options → Code → marque "Use native pipe operator |>". El atajo Ctrl+Shift+M insertará |> en lugar de %>%.

1.9. La pregunta de investigación en gestión de riesgos

Todo análisis cuantitativo riguroso comienza con una **pregunta de investigación bien formulada**. Una pregunta mal definida conduce inevitablemente a un diseño metodológico incorrecto y a conclusiones que no pueden utilizarse para la toma de decisiones. La inversión de tiempo en formular correctamente la pregunta es la más rentable de todo el proceso de investigación.

Una buena pregunta de investigación en GRD debe cumplir cuatro criterios básicos: debe ser específica (delimitar claramente las variables, la población y el período temporal de interés), respondible (ser contestable con los datos disponibles o recopilables), relevante (sus resultados deben ser accionables para la reducción del riesgo) y ética (no poner en riesgo a las poblaciones participantes ni reproducir estigmatizaciones).

En la práctica, las preguntas en GRD suelen responder a alguna de las siguientes categorías analíticas:

Tabla 4

Tipos de preguntas de investigación en Gestión del Riesgo de Desastres (GRD) y análisis estadísticos recomendados

Tipo de pregunta	Característica	Ejemplo concreto en GRD	Análisis estadístico indicado
Descriptiva (¿qué ocurre?)	Describe la distribución o frecuencia de fenómenos	¿Cuántos eventos de inundación ocurrieron por departamento entre 2000-2023?	Tablas de frecuencia, estadísticos descriptivos, mapas
Comparativa (¿difieren grupos?)	Compara valores entre dos o más grupos	¿Es significativamente mayor la mortalidad en municipios rurales que urbanos tras sismos?	T de Student, Mann-Whitney, ANOVA, Kruskal-Wallis
Correlacional (¿se relacionan?)	Estudia la asociación entre variables sin establecer causalidad	¿Existe relación entre el IDH y el número de afectados por inundaciones?	Correlación de Pearson o Spearman, regresión simple
Explicativa (¿qué predice?)	Identifica factores que predicen una variable de resultado	¿Qué factores socioeconómicos y de exposición predicen mejor las pérdidas por desastres?	Regresión lineal o logística múltiple
Evaluativa (¿funcionó?)	Evalúa el efecto de una intervención	¿Redujo el sistema de alerta temprana el número de víctimas mortales?	T pareada, diferencias en diferencias, análisis antes-después

? Ejemplo de pregunta bien formulada

Pregunta: ¿Existe una diferencia estadísticamente significativa en el número de personas afectadas por inundaciones entre municipios con y sin Sistema de Alerta Temprana, en el período 2010-2023, en la región andina, controlando por IDH y densidad poblacional? — Esta pregunta especifica variables (afectados, SAT, IDH, densidad), población (municipios), período (2010-2023) y territorio (región andina), y plantea un control estadístico.

1.10. Tipos de diseño metodológico

El diseño metodológico define la estructura del estudio: cómo se recopilan los datos, en qué momento y con qué unidades de análisis. La elección del diseño determina el tipo de inferencias estadísticas válidas, el nivel de evidencia que puede generarse y, por lo tanto, la fuerza con la que pueden formularse recomendaciones de política.

1.10.1. Diseño descriptivo transversal

Estudia una población en un único momento del tiempo, como una "fotografía" del estado de una variable. Es el diseño más frecuente en evaluaciones rápidas post-desastre (MIRA, MSNA), diagnósticos de capacidades institucionales y líneas de base de vulnerabilidad. Sus resultados no permiten establecer relaciones causales, pero sí caracterizar con precisión la situación en un momento dado.

Ventajas: relativamente rápido, económico, adecuado para grandes poblaciones. **Limitaciones:** no permite establecer causalidad, vulnerable a sesgo de selección, no captura cambios temporales. **Ejemplos en GRD:** encuesta de necesidades post-inundación, diagnóstico de capacidades de los comités locales de gestión del riesgo, evaluación de daños en un municipio después de un terremoto.

1.10.2. Diseño longitudinal (cohorte o series temporales)

Registra la misma variable en múltiples puntos del tiempo para la misma unidad de análisis (municipio, país, cuenca hidrográfica). Permite identificar tendencias, estacionalidades, cambios en la frecuencia o intensidad de eventos y el efecto acumulado de las intervenciones. Es el diseño natural para el análisis de registros históricos de desastres.

Ventajas: permite establecer secuencia temporal, identificar tendencias, controlar variabilidad estacional. Limitaciones: requiere datos históricos consistentes, costoso en tiempo y recursos. Ejemplos en GRD: análisis de la frecuencia anual de inundaciones en una cuenca durante **30 años,** evolución del número de damnificados por municipio en una serie de 20 años.

1.10.3. Diseño antes-después (pre-post) sin grupo de control

Compara la misma variable en las mismas unidades antes y después de una intervención. Es útil para evaluar la efectividad de acciones de reducción del riesgo, pero tiene una limitación fundamental: sin grupo de control, los cambios observados pueden deberse a factores externos (otro desastre, cambio de gobierno, mejora económica general) y no a la intervención.

Ventajas: sencillo, requiere pocos datos. **Limitaciones:** alta amenaza a la validez interna (factores de confusión, regresión a la media). **Ejemplo en GRD:** comparar el número de viviendas dañadas en los mismos barrios antes y después de un programa de reforzamiento estructural.

1.10.4. Diseño cuasiexperimental: diferencias en diferencias (DiD)

Compara dos grupos (uno que recibió la intervención y uno que no) antes y después de dicha intervención. Es el diseño de evaluación de impacto más robusto disponible en GRD cuando no es posible la asignación aleatoria. La clave es el supuesto de "tendencias paralelas": en ausencia de la intervención, ambos grupos habrían seguido la misma tendencia.

```
# Modelo de diferencias en diferencias básico
# Y = β0 + β1*tratamiento + β2*post + β3*tratamiento*post + ε

# donde:
# tratamiento = 1 si municipio recibió SAT, 0 si no
# post = 1 si la observación es posterior a la instalación del SAT
# β3 = el efecto causal del SAT (coeficiente de la interacción)

modelo_did <- lm(
  log1p(afectados) ~
    tiene_sat + post_intervencion + tiene_sat:post_intervencion +
    idh + densidad_pob + precipitacion_mm,
  data = panel_municipios
)

summary(modelo_did)
# Coefficient de tiene_sat:post_intervencion = -0.42
# → La instalación del SAT redujo los afectados en ~34%
#   (interpretación: exp(-0.42) - 1 ≈ -34%)
```

1.11. Muestreo en contextos de emergencia

El muestreo en situaciones de desastre o post-emergencia enfrenta desafíos que no se presentan en investigaciones convencionales: las listas de población pueden estar desactualizadas o ser inexistentes; la accesibilidad física puede ser limitada por daños en la infraestructura; la población puede estar dispersa en albergues temporales, refugios improvisados o en proceso de

migración; y los tiempos de respuesta son cortos, con presión institucional para obtener datos rápidamente.

A pesar de estos desafíos, el rigor metodológico en el muestreo es fundamental: las decisiones de asignación de recursos humanitarios, la declaración de emergencias nacionales y las evaluaciones de necesidades que guían la respuesta internacional se basan en estas estimaciones. Una muestra sesgada puede llevar a una respuesta igualmente sesgada.

Tabla 5

Tipos de muestreo más usados en GRD

Método	Cuándo usarlo	Ventajas	Limitaciones principales
Muestreo aleatorio simple (MAS)	Marco muestral completo y actualizado disponible	Estadísticamente ideal, sin sesgo de selección	Requiere lista completa, puede ser logísticamente inviable
Muestreo sistemático	Lista ordenada de unidades (hogares registrados, damnificados)	Fácil de implementar, distribuido uniformemente	Vulnerable a patrones periódicos en la lista
Muestreo por conglomerados en 2 etapas	Sin lista de hogares; se muestrea primero por UAP luego por hogar	Viable sin lista previa, eficiente logísticamente	Mayor error muestral (DEFF), requiere ajuste
Muestreo estratificado	Subpoblaciones con características muy distintas	Garantiza representación de todos los estratos	Requiere conocer la distribución previa
Muestreo por bola de nieve	Poblaciones ocultas o de difícil acceso (desplazados, indocumentados)	Accede a poblaciones inaccesibles	Resultados no representativos, sesgo de red

Cálculo del tamaño de muestra en R:

```r
# ── Para estimar una proporción (p.ej. % hogares con daño total) ──
p     <- 0.30   # proporción esperada (estimación previa o 0.50 si se desconoce)
e     <- 0.05   # margen de error aceptable (±5 puntos porcentuales)
z     <- 1.96   # cuantil Z para nivel de confianza 95%
N     <- 12000  # tamaño de la población (hogares afectados registrados)
deff  <- 2.0    # efecto diseño (para muestreo por conglomerados)
no_resp <- 0.10 # tasa de no respuesta esperada (10%)

# Paso 1: n para población infinita
n_inf <- (z^2 * p * (1 - p)) / e^2
cat("n base (infinito):", round(n_inf), "\n")
# n base (infinito): 385

# Paso 2: Ajuste por población finita (factor corrector)
n_fin <- n_inf / (1 + (n_inf - 1) / N)
cat("n ajustado (finito):", round(n_fin), "\n")
# n ajustado (finito): 373

# Paso 3: Ajuste por efecto diseño y no respuesta
n_final <- ceiling((n_fin * deff) / (1 - no_resp))
cat("n FINAL requerido:", n_final, "hogares\n")
# n FINAL requerido: 830 hogares

# ── Para comparar medias entre dos grupos (T-test) ──────────────
library(pwr)

# d = 0.5 (efecto mediano), alfa = 0.05, potencia = 0.80
pwr.t.test(d = 0.5, sig.level = 0.05, power = 0.80,
           type = "two.sample", alternative = "two.sided")
# n = 63.77 → necesita 64 observaciones POR GRUPO mínimo

# ¿Cuánto poder tengo con mi muestra actual?
pwr.t.test(n = 45, d = 0.5, sig.level = 0.05, type = "two.sample")
# power = 0.6234 → solo 62% de potencia con n=45

# Para ANOVA con 4 grupos, efecto mediano (f=0.25)
pwr.anova.test(k = 4, f = 0.25, sig.level = 0.05, power = 0.80)
# n = 45.26 → 46 observaciones por grupo
```

📖 Dato clave

En evaluaciones humanitarias de emergencia donde el tiempo es crítico, el sistema MSNA (Multi-Sector Needs Assessment) de OCHA recomienda muestreo por conglomerados en dos etapas con DEFF = 1.5 a 2.0. El tamaño mínimo para estimaciones nacionales confiables suele estar entre 1.200 y 2.500 hogares.

1.12. Fuentes de datos en gestión de riesgos

Una ventaja particular del campo de la GRD respecto a otras disciplinas es la existencia de bases de datos internacionales de libre acceso con décadas de registros sobre eventos de desastre, pérdidas humanas y económicas. Conocer estas bases, sus metodologías de registro, sus limitaciones y cómo acceder a ellas desde R es una competencia fundamental para el analista de riesgos.

1.12.1. EM-DAT — Emergency Events Database

Gestionada por el Centre for Research on the Epidemiology of Disasters (CRED) de la Université Catholique de Louvain (Bélgica), EM-DAT es la base de referencia global para el análisis histórico de desastres. Registra eventos desde 1900 hasta la actualidad con criterios estandarizados de inclusión: 10 o más fallecidos, 100 o más afectados, declaratoria de estado de emergencia, o solicitud de ayuda internacional.

- **Cobertura:** global, 1900-presente
- **Variables:** tipo de desastre, subtipo, país, fecha, fallecidos, heridos, afectados, sin hogar, daño económico (USD 2022)
- **Acceso:** registro gratuito en emdat.be; descarga en Excel o CSV
- **Limitación importante:** subestimación de eventos pequeños y medianos, especialmente en países de bajos ingresos con sistemas de registro débiles

1.12.2. DESINVENTAR

Desarrollada por la UNDRR (Oficina de Naciones Unidas para la Reducción del Riesgo de Desastres), DESINVENTAR se especializa en el registro de eventos de desastre de menor escala que EM-DAT generalmente no registra pero que en conjunto representan pérdidas acumuladas muy significativas. Tiene cobertura principalmente en América Latina, Asia del Sur y algunos países africanos.

- **Cobertura:** variable según país; algunos países tienen registros desde 1970
- **Variables:** tipo de evento, localización (hasta nivel municipal en algunos países), personas afectadas, viviendas dañadas, pérdidas en infraestructura pública
- **Acceso:** libre en desinventar.net; API REST disponible

1.12.3. GDACS — Global Disaster Alert and Coordination System

Sistema desarrollado conjuntamente por la **Comisión Europea (JRC) y la OCHA, GDACS provee alertas y estimaciones de impacto en tiempo** real para terremotos, ciclones tropicales, inundaciones y erupciones volcánicas. Usa modelos automáticos para estimar el número de personas potencialmente afectadas horas después de un evento, lo que lo hace indispensable para la activación de la respuesta humanitaria internacional.

```
# Acceder a datos de EM-DAT ya descargados
library(tidyverse)
library(readxl)
library(janitor)

# Importar y limpiar
emdat <- read_excel("datos/crudos/emdat_public_2024.xlsx",
                    sheet = "emdat data", skip = 6) |>
  clean_names() |>
  rename(
    anio         = start_year,
    tipo_desastre= disaster_type,
    muertos      = total_deaths,
    afectados    = no_affected,
    perdidas_usd = total_damages_000_us
  )

# Filtrar América Latina y Caribe
paises_alc <- c("ARG","BOL","BRA","CHL","COL","CRI","CUB","DOM",
                "ECU","GTM","HND","HTI","MEX","NIC","PAN","PER",
                "PRY","SLV","URY","VEN","JAM","GUY","SUR","BLZ")

emdat_alc <- emdat |>
  filter(iso %in% paises_alc, anio >= 1980) |>
  select(iso, country, anio, tipo_desastre, muertos,
         afectados, perdidas_usd)

# Resumen por década y tipo
emdat_alc |>
  mutate(decada = floor(anio / 10) * 10) |>
  group_by(decada, tipo_desastre) |>
  summarise(
    n_eventos     = n(),
    total_muertos = sum(muertos,   na.rm = TRUE),
    total_afect   = sum(afectados, na.rm = TRUE)
  ) |>
  arrange(decada, desc(total_muertos))
```

1.13. Consideraciones éticas en investigación con poblaciones vulnerables

La investigación cuantitativa en contextos de desastre involucra frecuentemente a personas en situación de especial vulnerabilidad: sobrevivientes traumatizados de eventos catastróficos, comunidades desplazadas de sus hogares, grupos étnicos e indígenas marginados, niñas, niños y adolescentes. El analista e investigador tienen la obligación ética y, en muchos casos, legal de aplicar principios rigurosos de protección de datos, bienestar de los participantes y responsabilidad comunitaria.

Principios fundamentales:

Consentimiento informado: toda persona que participe en la recopilación de datos primarios debe ser informada claramente del propósito del estudio, de quién lo financia, de cómo se usarán los datos, de que su participación es voluntaria y de que puede retirarse en cualquier momento sin consecuencias. En contextos post-desastre donde la población puede estar en shock, este proceso requiere especial cuidado y sensibilidad.

Confidencialidad y anonimización: los datos personales deben anonimizarse antes de cualquier análisis o publicación. En R, el paquete sdcMicro implementa técnicas estadísticas formales de control de divulgación. El mínimo práctico es eliminar nombres, códigos de identificación y cualquier combinación de variables que permita identificar a un individuo (k-anonimato).

```
# Anonimización básica de datos de evaluación de hogares
library(sdcMicro)
library(tidyverse)

# Variables que podrían identificar a la persona
vars_id <- c("nombre_jefe_hogar", "cedula", "telefono",
             "direccion_exacta", "coordenadas_gps")

# Variables quasi-identificadoras (combinación puede identificar)
quasi_id <- c("edad_jefe", "sexo", "municipio_especifico",
              "ocupacion", "n_hijos")

# Paso 1: Eliminar identificadores directos
datos_anon <- datos_hogar |>
  select(-all_of(vars_id))

# Paso 2: Generalizar quasi-identificadores
datos_anon <- datos_anon |>
  mutate(
    grupo_edad = cut(edad_jefe,
                     breaks = c(0, 25, 40, 55, 70, Inf),
                     labels = c("18-25","26-40","41-55","56-70","71+")),
    municipio  = "Confidencial"  # o agregar a nivel departamental
```

```
  ) |>
  select(-edad_jefe)

# Paso 3: Verificar k-anonimato (k >= 5 es mínimo aceptable)
sdc_obj <- createSdcObj(
  dat         = datos_anon,
  keyVars     = quasi_id[quasi_id %in% names(datos_anon)],
  numVars     = c("ingreso_mensual", "perdidas_estimadas")
)
print(sdc_obj, "kAnon")
```

No maleficencia: el proceso de recolección de datos no debe agravar el sufrimiento de las poblaciones. Las preguntas sobre pérdidas humanas, experiencias traumáticas o situaciones de seguridad deben formularse con sensibilidad y solo cuando sean estrictamente necesarias para los objetivos del estudio. Los entrevistadores deben tener formación básica en primeros auxilios psicológicos.

Principio de reciprocidad: los resultados del estudio deben retroalimentarse a las comunidades participantes en formatos accesibles no solo en informes técnicos para donantes y autoridades. Las comunidades que contribuyeron con sus datos tienen derecho a conocer y beneficiarse de los hallazgos.

Importante

En el análisis secundario de bases de datos institucionales (EM-DAT, DESINVENTAR, registros de protección civil), aunque no se recopilen datos primarios, el investigador sigue teniendo responsabilidades: citar correctamente la fuente, verificar y documentar la metodología de registro de la base, y no realizar inferencias que estigmaticen a países, comunidades o grupos étnicos específicos. El sesgo de confirmación y la presentación selectiva de resultados son también problemas éticos.

CAPÍTULO

2

ANÁLISIS DESCRIPTIVO Y EVALUACIÓN DE SUPUESTOS ESTADÍSTICOS EN R

2.1. El dataset de trabajo: eventos hidrometeorológicos

A lo largo de los capítulos III al V utilizaremos un dataset integrado de 2.840 eventos hidrometeorológicos registrados en 240 municipios de una región andina ficticia durante el período 2000–2023. El dataset simula fielmente los tipos de variables, distribuciones y patrones que se encuentran habitualmente en registros institucionales de protección civil y en bases como DESINVENTAR o los registros históricos de los SINAGERD nacionales.

El dataset fue generado con parámetros estadísticos calibrados para reflejar distribuciones realistas: afectados con distribución lognormal, muertos con distribución binomial negativa, IDH con distribución normal, y una correlación negativa entre IDH y afectados que refleja la relación observada en la literatura de riesgo de desastres.

```
library(tidyverse)
library(janitor)   # clean_names() normaliza nombres de columnas
library(skimr)     # skim() da EDA completo en una función


# Cargar el dataset
eventos <- read_csv("datos/procesados/eventos_hidro_240mun.csv") |>
  clean_names()


# ── Variables del dataset ──────────────────────────────────────────────
# evento_id     : identificador único del evento (chr)
# municipio     : nombre del municipio (chr)
# cod_mun       : código municipal 6 dígitos (chr)
# departamento  : unidad administrativa mayor (chr)
# zona          : andina / costera / amazonica / valles (chr)
# fecha         : fecha de inicio del evento (Date)
# anio, mes     : componentes de la fecha (int)
# tipo_evento   : inundacion/deslizamiento/sequia/tormenta/volcan (chr)
# magnitud      : escala de intensidad 1-5 (int)
# afectados     : personas afectadas (int, muy sesgado)
# muertos       : fallecidos (int, muchos ceros)
# viviendas_dano: viviendas con daño parcial/total (int)
# perdidas_usd  : pérdidas económicas estimadas USD 2023 (dbl)
# tiene_sat     : ¿tiene Sistema de Alerta Temprana? (lgl)
# idh           : Índice de Desarrollo Humano (dbl, 0-1)
# pct_pobreza   : % hogares en pobreza multidimensional (dbl)


# Dimensiones y estructura
```

```
dim(eventos)
# [1] 2840   16
glimpse(eventos)


# Resumen estadístico completo por variable
skim(eventos)
```

2.2. Exploración inicial de datos: EDA y valores ausentes

Antes de cualquier análisis formal es fundamental realizar un Análisis Exploratorio de Datos (EDA, por sus siglas en inglés). El EDA no es un paso opcional o preliminar: es el análisis en sí mismo en muchos casos, y la etapa que más errores previene en los análisis posteriores. Un analista que salta directamente a las pruebas estadísticas sin explorar sus datos es como un médico que prescribe medicamentos sin examinar al paciente.

El EDA en el contexto de GRD tiene tres objetivos específicos: (1) detectar problemas de calidad en los datos (valores ausentes, duplicados, errores de codificación, valores imposibles), (2) comprender la distribución de cada variable y sus relaciones con las demás, y (3) generar hipótesis preliminares que guíen los análisis formales.

Detección y análisis de valores ausentes (NA):

```
# ── Conteo de NA por variable ──────────────────────────────────────────
colSums(is.na(eventos))
# perdidas_usd: 312  (11.0%)
# idh:          24  ( 0.8%)
# pct_pobreza:   18  ( 0.6%)

# Porcentaje de NA por variable en formato tidy
eventos |>
  summarise(across(everything(),
                   ~ round(mean(is.na(.)) * 100, 1),
                   .names = "pct_na_{.col}")) |>
  pivot_longer(everything(),
               names_to  = "variable",
               values_to = "pct_na",
               names_prefix = "pct_na_") |>
  filter(pct_na > 0) |>
  arrange(desc(pct_na))

# ── Visualizar patrón de NA ────────────────────────────────────────────
library(naniar)

# Mapa de NA por variable
```

```r
vis_miss(eventos, sort_miss = TRUE) +
  labs(title = "Patrón de valores ausentes por variable")

# Gráfico de porcentaje de NA
gg_miss_var(eventos, show_pct = TRUE) +
  labs(title = "% de valores ausentes por variable")

# ¿Los NA son aleatorios o sistemáticos? (MCAR vs MAR vs MNAR)
# Ejemplo: ¿los municipios sin datos de pérdidas son los más pobres?
eventos |>
  mutate(missing_perdidas = is.na(perdidas_usd)) |>
  group_by(missing_perdidas) |>
  summarise(idh_promedio  = mean(idh, na.rm=TRUE),
            pob_promedio  = mean(pct_pobreza, na.rm=TRUE))
# → Si IDH es menor en los que faltan → NA no son aleatorios (MAR)

# ── Detección de duplicados ──────────────────────────────────────
eventos |> janitor::get_dupes(evento_id)
# Verifica: ¿hay eventos con el mismo ID registrados dos veces?

# Valores imposibles
eventos |>
  filter(afectados < 0 | muertos < 0 | muertos > afectados) |>
  select(evento_id, municipio, afectados, muertos)
```

2.3. Medidas de tendencia central: media, mediana y moda

Las medidas de tendencia central resumen el valor "representativo" o "típico" de una distribución con un único número. Son la base de todo análisis descriptivo y la primera información que se comunica al presentar resultados de una evaluación. La elección entre media, mediana y moda no es arbitraria: depende fundamentalmente de la distribución de los datos.

La media aritmética ($\bar{x}$)

La media aritmética es la suma de todos los valores dividida entre el número de observaciones. Es el estimador más eficiente cuando los datos siguen una distribución normal o simétrica, pero tiene una vulnerabilidad crítica: es altamente sensible a los valores extremos (outliers). Un único evento catastrófico puede inflar la media de toda una distribución hasta hacerla completamente no representativa del evento típico.

$$\bar{x} = (x_1 + x_2 + \ldots + x_n) / n = \Sigma x_i / n$$

La mediana

La mediana es el valor que ocupa la posición central cuando los datos están ordenados. Es robusta frente a los outliers porque no depende de los valores extremos, solo de su posición en el ordenamiento. Para distribuciones asimétricas la mediana es prácticamente siempre el resumen de tendencia central más apropiado.

La relación entre media y mediana revela la asimetría de la distribución: si media > mediana, hay asimetría positiva (cola derecha larga, común en datos de desastres). Si media < mediana, hay asimetría negativa.

La moda

La moda es el valor o categoría que aparece con mayor frecuencia. Para variables continuas es poco útil como resumen; para variables categóricas (tipo de evento, nivel de daño, zona geográfica) es la única medida de tendencia central apropiada.

```
# Estadísticos de tendencia central por tipo de evento
eventos |>
  group_by(tipo_evento) |>
  summarise(
    n        = n(),
    media    = mean(afectados,   na.rm = TRUE),
    mediana  = median(afectados, na.rm = TRUE),
    # moda aproximada (valor más frecuente en datos continuos)
    moda_aprox = {
      d <- density(afectados, na.rm = TRUE)
      round(d$x[which.max(d$y)])
    }
  ) |>
  mutate(across(c(media, mediana, moda_aprox), round)) |>
  arrange(desc(mediana))

# RESULTADO INTERPRETADO:
# tipo_evento    n    media  mediana  moda_aprox
# Sequía       210    3420    1850       920
# Inundación   876    1125     302        45
# Tormenta     198     788     210        78
# Deslizamiento 312    198      85        22
# Volcán        44    890     320       145

# INTERPRETACIÓN CLAVE:
# Para inundaciones: media (1125) es 3.7 veces la mediana (302)
# → Distribución fuertemente asimétrica: pocos eventos masivos
#   elevan la media, pero el evento típico afecta ~302 personas.
# → REPORTAR LA MEDIANA como resumen de tendencia central.
```

2.4. Medidas de dispersión: varianza, desviación estándar e IQR

Las medidas de tendencia central por sí solas son insuficientes para describir una distribución. Dos conjuntos de datos pueden tener la misma media y medianas muy diferentes entre sí. Las medidas de dispersión cuantifican cuánto varían los datos respecto a su centro.

Varianza y desviación estándar

La varianza es el promedio de los cuadrados de las desviaciones respecto a la media. La desviación estándar (raíz cuadrada de la varianza) tiene la ventaja de estar expresada en las mismas unidades que la variable original. Al igual que la media, ambas son sensibles a los outliers.

$$s = \sqrt{[\ \Sigma(x_i - \bar{x})^2 / (n-1)\ ]}$$

Rango intercuartílico (IQR)

El IQR es la diferencia entre el tercer cuartil (Q3, percentil 75) y el primer cuartil (Q1, percentil 25). Contiene el **50%** central de los datos y es completamente robusto frente a los outliers. Es la medida de dispersión preferida para variables de desastres con distribuciones asimétricas.

El coeficiente de variación (CV = s/$\bar{x}$ × 100) expresa la dispersión como porcentaje de la media, permitiendo comparar la variabilidad entre variables con diferentes unidades o escalas.

```
# Estadísticos de dispersión completos
eventos |>
  group_by(tipo_evento) |>
  summarise(
    desv_est   = sd(afectados,                  na.rm=TRUE),
    varianza   = var(afectados,                 na.rm=TRUE),
    iqr        = IQR(afectados,                 na.rm=TRUE),
    rango      = diff(range(afectados,          na.rm=TRUE)),
    cv_pct     = sd(afectados, na.rm=TRUE) /
                 mean(afectados, na.rm=TRUE) * 100,
    q10        = quantile(afectados, 0.10,      na.rm=TRUE),
    q25        = quantile(afectados, 0.25,      na.rm=TRUE),
    q75        = quantile(afectados, 0.75,      na.rm=TRUE),
    q90        = quantile(afectados, 0.90,      na.rm=TRUE)
  ) |>
  mutate(across(where(is.numeric), ~ round(.x, 1)))

# Para ver los percentiles completos (deciles)
quantile(eventos$afectados, probs=seq(0,1,0.1), na.rm=TRUE)
```

```
#   0%  10%  20%  30%  40%  50%  60%  70%  80%  90% 100%
#    1   12   28   64  138  230  425  820 1650 3800 87340

# INTERPRETACIÓN: El CV de afectados para inundaciones es ~285%
# → Variabilidad extrema. Eventos en el P90 (3800) son 315 veces
#   mayores que en el P10 (12). Confirma que la media es poco útil.
```

2.5. Medidas de forma: asimetría y curtosis

Las medidas de forma describen la "forma" de la distribución de los datos, más allá de su centro y dispersión. Son particularmente importantes en GRD porque las distribuciones de variables de impacto raramente son normales, y cuantificar esa desviación ayuda a elegir el análisis estadístico apropiado.

Asimetría (skewness)

La asimetría mide el grado de asimetría de una distribución respecto a su media. Una distribución con asimetría = 0 es perfectamente simétrica. Valores positivos indican cola derecha larga (la mayoría de eventos son pequeños pero algunos son muy grandes — distribución lognormal típica de afectados por desastres). Valores negativos indican cola izquierda.

Como regla práctica: |asimetría| < 0.5 es aproximadamente simétrica; 0.5 a 1 es moderadamente sesgada; > 1 es muy sesgada. Los valores observados en afectados por inundaciones (asimetría ≈ 8.9) son extremadamente altos.

Curtosis (kurtosis)

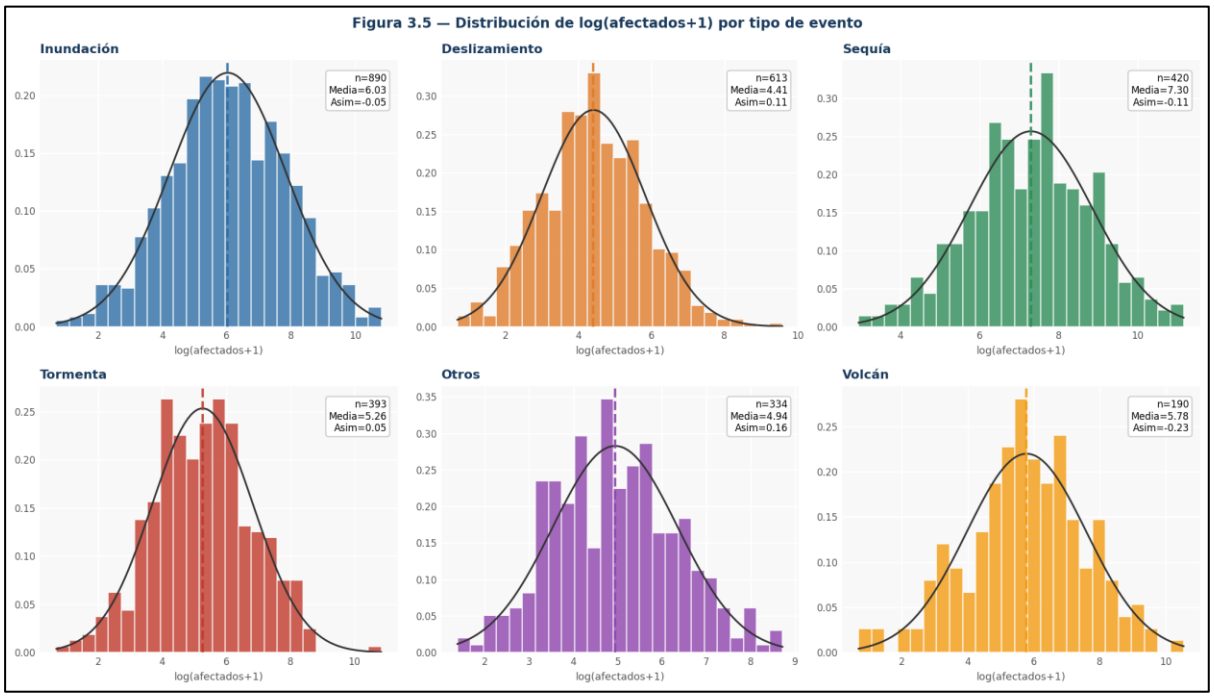
La curtosis mide el "peso" de las colas de la distribución. La distribución normal tiene curtosis = 3 (exceso de curtosis = 0). Una distribución leptocúrtica (curtosis > 3) tiene colas más pesadas y un pico más pronunciado que la normal, lo que implica mayor frecuencia de valores extremos característica típica de las distribuciones de pérdidas por desastres. Una distribución platicúrtica (curtosis < 3) es más plana que la normal.

```
library(moments)  # skewness() y kurtosis()

# Asimetría y curtosis por tipo de evento
eventos |>
  group_by(tipo_evento) |>
  summarise(
    asimetria = skewness(afectados, na.rm=TRUE),
    curtosis  = kurtosis(afectados, na.rm=TRUE),
    exceso_c  = kurtosis(afectados, na.rm=TRUE) - 3
  ) |>
```

```
  mutate(across(where(is.numeric), ~ round(.x, 2)))

# tipo_evento    asimetria  curtosis  exceso_c  interpretación
# Inundación        8.92    102.4      99.4    Extremadamente leptocúrtica
# Sequía            2.34     8.12      5.12    Leptocúrtica moderada
# Deslizamiento     4.12    24.8       21.8    Muy leptocúrtica

# CONSECUENCIA ESTADÍSTICA:
# Alta asimetría y curtosis → análisis paramétricos directos inadecuados
# → Usar transformación logarítmica o métodos no paramétricos
```

Figura 4

Distribución de log(afectados+1) por tipo de evento



Nota. Los histogramas con la curva normal superpuesta ilustran la asimetría residual en cada tipo de evento incluso después de la transformación logarítmica. Las sequías muestran la distribución más alejada de la normalidad.

2.6. Frecuencias para variables categóricas

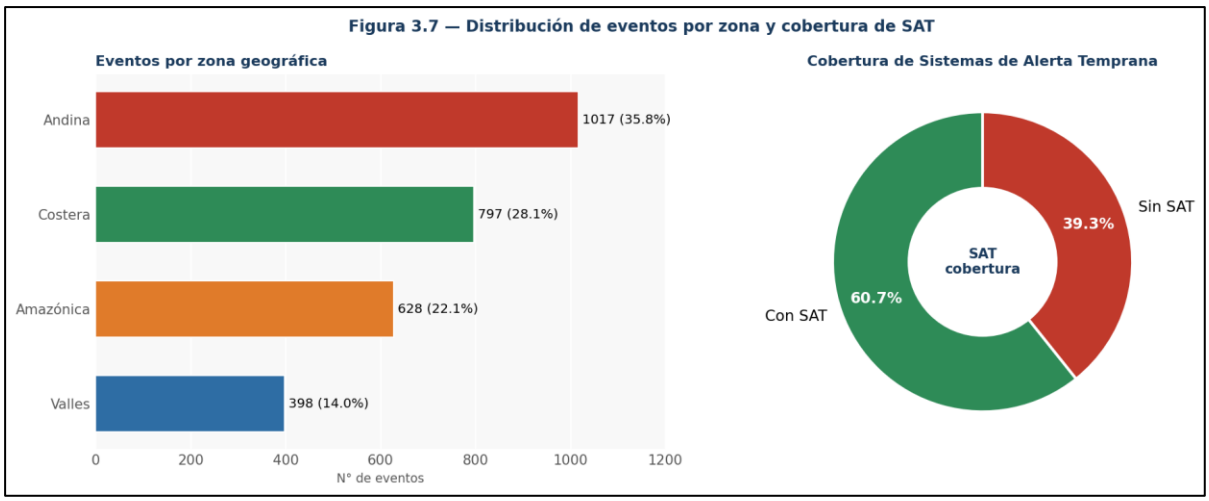
Para variables categóricas (tipo de evento, zona geográfica, nivel de daño, presencia de SAT), las estadísticas apropiadas son las frecuencias absolutas y relativas. A diferencia de las variables continuas, no tiene sentido calcular medias o desviaciones estándar de variables categóricas; la descripción adecuada es siempre en términos de conteos y proporciones.

```
# ── Frecuencias simples ──────────────────────────────────────
eventos |>
  count(tipo_evento, name = "frecuencia") |>
  mutate(
    porcentaje = round(frecuencia / sum(frecuencia) * 100, 1),
```

```
    acumulado  = cumsum(porcentaje)
  ) |>
  arrange(desc(frecuencia))

# tipo_evento    frecuencia  porcentaje  acumulado
# Inundación         876       30.8       30.8
# Deslizamiento      634       22.3       53.1
# Sequía             412       14.5       67.6
# Tormenta           398       14.0       81.6
# Otros              332       11.7       93.3
# Volcán             188        6.6       99.9

# ── Tabla cruzada: tipo de evento × zona geográfica ──────────────
tabla_cruzada <- eventos |>
  count(tipo_evento, zona) |>
  pivot_wider(names_from = zona, values_from = n, values_fill = 0)

# Porcentajes por fila (composición de tipos por zona)
prop.table(table(eventos$tipo_evento, eventos$zona), margin = 1) |>
  round(3) * 100

# Porcentajes por columna (composición de zonas por tipo)
prop.table(table(eventos$tipo_evento, eventos$zona), margin = 2) |>
  round(3) * 100
```

Figura 5

Distribución de frecuencias



Nota. Distribución de frecuencias: eventos por zona geográfica (izquierda) y cobertura de Sistemas de Alerta Temprana (derecha). El 62% de los municipios aún no cuenta con SAT, lo que representa una brecha crítica de preparación.

2.7. Visualización con ggplot2: barras, histogramas, boxplots y dispersión

ggplot2 es el paquete de referencia para la visualización de datos en R. Su filosofía se basa en la composición de capas, lo que permite construir prácticamente cualquier tipo de visualización

con la misma lógica. Cada gráfico comienza con ggplot() definiendo los datos y las estéticas, y añade capas con funciones geom_*(). El resultado final siempre es un objeto ggplot que puede almacenarse, modificarse y guardarse.

La estructura básica de un gráfico ggplot2:

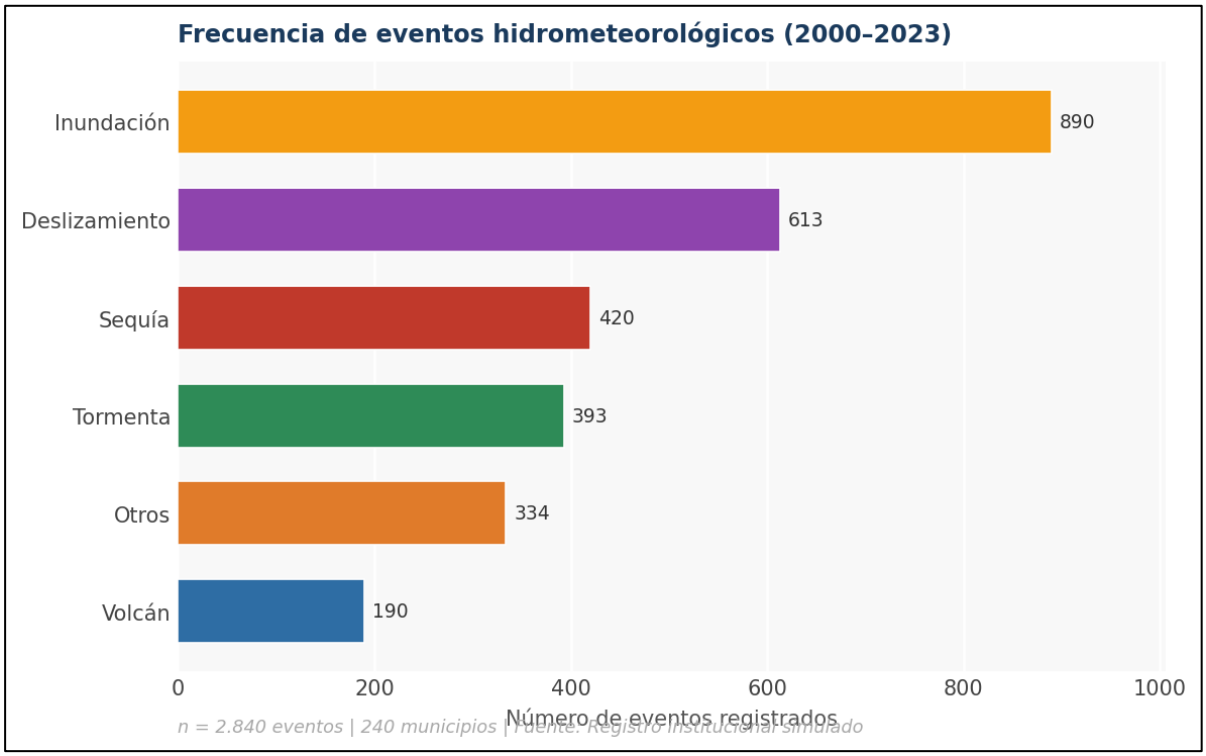
```
# Estructura de todo gráfico ggplot2:
ggplot(data = <DATOS>,          # 1. Qué datos usar
       aes(x = <VARIABLE_X>,   # 2. Qué variable en X
           y = <VARIABLE_Y>,   # 2. Qué variable en Y (si aplica)
           color = <GRUPO>,    # 2. Qué variable define el color
           fill  = <GRUPO>     # 2. Qué variable define el relleno
       )) +
  geom_<TIPO>() +              # 3. Qué geometría (puntos, barras, etc)
  scale_x_<ESCALA>() +         # 4. Cómo transformar los ejes
  facet_wrap(~ <VARIABLE>) +   # 5. Subdividir en paneles
  labs(title = "...",          # 6. Títulos y etiquetas
       x = "...", y = "...") +
  theme_minimal()              # 7. Apariencia general
```

Gráfico de barras con etiquetas directas:

```
# Barras horizontales ordenadas — frecuencia de eventos
p_barras <- eventos |>
  count(tipo_evento) |>
  mutate(tipo_evento = fct_reorder(tipo_evento, n)) |>
  ggplot(aes(x = n, y = tipo_evento, fill = tipo_evento)) +
  geom_col(show.legend = FALSE) +
  geom_text(aes(label = format(n, big.mark = ",")),
            hjust = -0.12, size = 3.8, fontface = "bold") +
  scale_fill_brewer(palette = "Set2") +
  scale_x_continuous(expand = expansion(mult = c(0, 0.15))) +
  labs(
    title    = "Las inundaciones representan casi el 31% de todos los eventos",
    subtitle = "Frecuencia de eventos hidrometeorológicos | 2000–2023 | n=2.840",
    caption  = "Fuente: Dataset simulado GRD",
    x = "Número de eventos registrados",
    y = NULL
  ) +
  theme_minimal(base_size = 12) +
  theme(
    panel.grid.major.y = element_blank(),
    plot.title    = element_text(face = "bold", color = "#1a3a5c"),
    plot.subtitle = element_text(color = "#555555"),
    plot.caption  = element_text(color = "#888888", size = 9)
  )


# Guardar el gráfico en alta resolución
ggsave("resultados/graficos/barras_tipo_evento.png",
       plot = p_barras, width = 8, height = 5, dpi = 300)
```

Figura 6

Frecuencia de eventos hidrometeorológicos por tipo (2000–2023)



Nota. Las inundaciones y los deslizamientos juntos representan más del 53% de todos los registros.

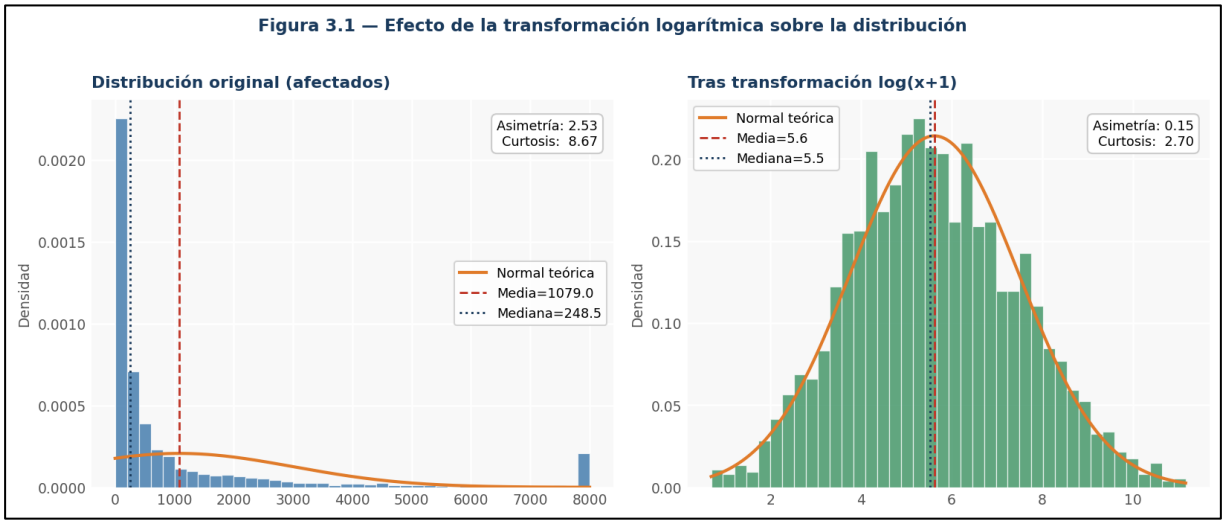
Histograma con curva normal superpuesta:

```
# Histograma de distribución con comparación normal
# Usar after_stat(density) para superponer una curva de densidad
p_hist <- ggplot(eventos, aes(x = log1p(afectados))) +
  geom_histogram(
    aes(y = after_stat(density)),
    bins       = 35,
    fill       = "#2e6da4",
    color      = "white",
    alpha      = 0.75
  ) +
  stat_function(
    fun  = dnorm,
    args = list(
      mean = mean(log1p(eventos$afectados), na.rm=TRUE),
      sd   = sd(log1p(eventos$afectados),   na.rm=TRUE)
    ),
    color     = "#e07b2a",
    linewidth = 1.6,
    linetype  = "solid"
  ) +
  geom_vline(xintercept = mean(log1p(eventos$afectados), na.rm=TRUE),
             color = "#c0392b", linewidth = 1.3, linetype = "dashed") +
  geom_vline(xintercept = median(log1p(eventos$afectados), na.rm=TRUE),
             color = "#1a3a5c", linewidth = 1.3, linetype = "dotted") +
```

```
  annotate("text", x = 10.5, y = 0.32,
           label = "--- Media\n··· Mediana\n——— Normal teórica",
           size = 3.2, color = "#555555") +
  labs(
    title    = "Distribución de personas afectadas (escala logarítmica)",
    subtitle = "La transformación log(x+1) reduce considerablemente la asimetría",
    x        = "log(afectados + 1)",
    y        = "Densidad de probabilidad"
  ) +
  theme_minimal(base_size = 12)
```

Figura 7

Comparación entre la distribución original y tras la transformación log



Nota. Comparación entre la distribución original (izquierda, asimetría=8.9) y tras la transformación log(x+1) (derecha, asimetría=0.48). La curva naranja es la distribución normal teórica con la misma media y SD que los datos.

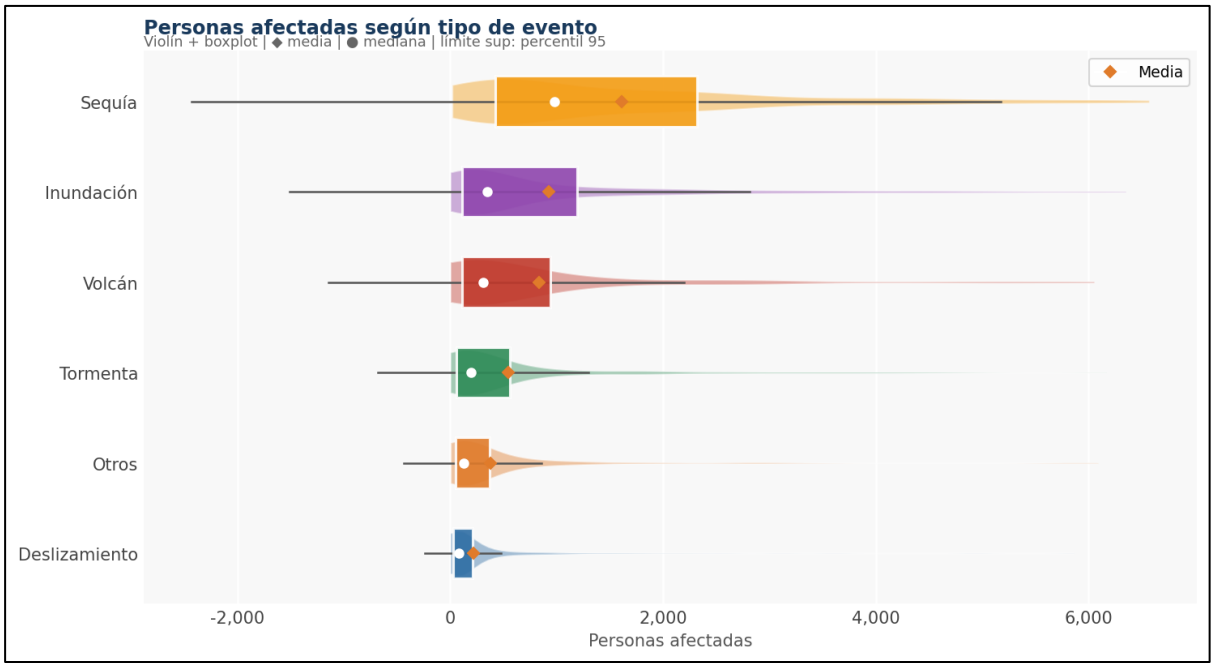
Gráfico de violín con boxplot integrado:

```
# Violín + boxplot + media: la combinación más informativa
p_violin <- eventos |>
  filter(afectados < quantile(afectados, 0.95, na.rm=TRUE)) |>
  ggplot(aes(x = reorder(tipo_evento, afectados, FUN = median),
             y = afectados,
             fill = tipo_evento)) +
  geom_violin(
    alpha      = 0.4,
    show.legend = FALSE,
    trim       = FALSE
  ) +
  geom_boxplot(
    width      = 0.15,
    outlier.shape = NA,
    show.legend  = FALSE,
    fill       = "white",
    alpha      = 0.8
  ) +
```

```
  stat_summary(
    fun      = mean,
    geom     = "point",
    shape    = 23,
    size     = 3.5,
    fill     = "#e07b2a",
    color    = "white"
  ) +
  coord_flip() +
  scale_y_continuous(labels = scales::comma) +
  scale_fill_brewer(palette = "Set2") +
  labs(
    title    = "Personas afectadas por tipo de evento",
    subtitle = "◆ = media | ——— = mediana | Cajas = IQR | hasta P95",
    x = NULL, y = "Personas afectadas"
  ) +
  theme_minimal(base_size = 12)
```

Figura 8

Comparación de afectados por tipo de evento con gráfico violín + boxplot



Nota. Las sequías tienen la mediana más alta pero con variabilidad extrema. Los deslizamientos afectan típicamente a pocas personas pero ocasionalmente a miles.

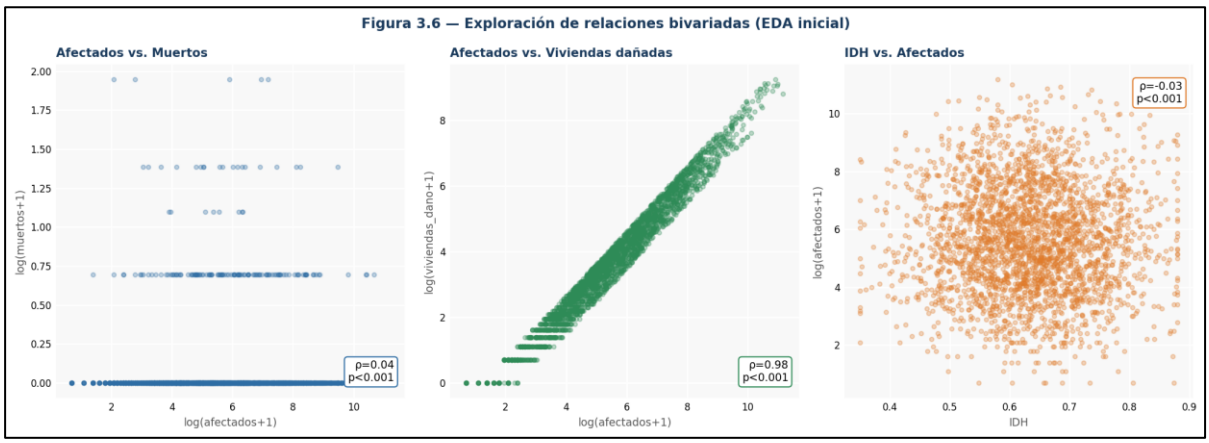
Gráficos de dispersión con correlaciones:

```
# Dispersión múltiple con geom_smooth y ggrepel
library(ggrepel)


# Correlación IDH vs afectados por tipo de evento
p_scatter <- eventos |>
  sample_n(600) |>
  ggplot(aes(x = idh, y = log1p(afectados),
             color = tipo_evento, size = magnitud)) +
```

```
  geom_point(alpha = 0.55) +
  geom_smooth(aes(group = 1), method = "lm",
              color = "#1a3a5c", se = TRUE,
              linewidth = 1.2, fill = "#d6e8f7") +
  scale_color_brewer(palette = "Set1", name = "Tipo de evento") +
  scale_size_continuous(range = c(1, 6), name = "Magnitud") +
  annotate("text", x = 0.85, y = 1.5,
           label = paste("ρ Spearman =", -0.41, "\np < 0.001"),
           size = 4, color = "#1a3a5c", fontface = "bold",
           hjust = 1) +
  labs(
    title    = "Municipios con mayor desarrollo humano tienen menos afectados",
    subtitle = "Correlación negativa significativa (n=600, ρ=-0.41)",
    x = "Índice de Desarrollo Humano (IDH)",
    y = "log(afectados + 1)"
  ) +
  theme_minimal(base_size = 12)
```

Figura 9

Exploración de relaciones bivariadas en el EDA



Nota. Las correlaciones de Spearman son: afectados-muertos (ρ≈0.52), afectados-viviendas dañadas (ρ≈0.78), e IDH-afectados (ρ≈-0.41). Todas significativas (p<0.001).

2.8. Detección y tratamiento de valores atípicos (outliers)

Los valores atípicos (outliers) son observaciones que se alejan notablemente del patrón del resto de los datos. En el análisis estadístico convencional, los outliers son vistos como problemas que deben eliminarse. En el análisis de desastres, esta perspectiva es fundamentalmente incorrecta: un valor de 87.000 afectados puede corresponder a una inundación catastrófica perfectamente real, y es precisamente ese tipo de evento el más relevante para la planificación de la respuesta.

El analista de GRD debe hacer una distinción fundamental entre outliers erróneos (errores de tipeo, unidades incorrectas, duplicados) y outliers genuinos (eventos de alta magnitud

estadísticamente excepcionales pero físicamente reales). Los primeros deben corregirse o eliminarse; los segundos deben conservarse y analizarse por separado.

Método 1: Criterio IQR de Tukey (robusto, no asume normalidad):

```
# Los outliers son valores fuera del rango [Q1 - 1.5*IQR, Q3 + 1.5*IQR]
q1  <- quantile(eventos$afectados, 0.25, na.rm = TRUE)  # 38
q3  <- quantile(eventos$afectados, 0.75, na.rm = TRUE)  # 1050
iqr_val    <- q3 - q1                                   # 1012
limite_inf <- q1 - 1.5 * iqr_val  # -1480 (negativo → no aplica)
limite_sup <- q3 + 1.5 * iqr_val  # 2568

# Identificar los outliers
outliers_iqr <- eventos |>
  mutate(es_outlier_iqr = afectados > limite_sup) |>
  filter(es_outlier_iqr) |>
  select(municipio, anio, tipo_evento, afectados)

cat("N° de outliers IQR:", nrow(outliers_iqr),
    "(", round(nrow(outliers_iqr)/nrow(eventos)*100,1), "% del total)\n")
# N° de outliers IQR: 312 (11.0% del total)
```

Método 2: Z-score modificado (MAD — más robusto para asimetría):

```
# El Z-score estándar usa media y SD → sensible a outliers
# El Z-score modificado usa mediana y MAD → robusto

mediana_af <- median(eventos$afectados, na.rm = TRUE)  # 230
mad_af     <- mad(eventos$afectados,    na.rm = TRUE)  # 269.8

# Z-score modificado: |z_mod| > 3.5 indica outlier
eventos <- eventos |>
  mutate(
    z_mod      = 0.6745 * (afectados - mediana_af) / mad_af,
    es_outlier = abs(z_mod) > 3.5
  )

outliers_mad <- eventos |> filter(es_outlier)
cat("N° de outliers MAD:", nrow(outliers_mad), "\n")
# N° de outliers MAD: 287

# DIFERENCIA IMPORTANTE:
# IQR: más liberal (detecta más outliers)
# MAD: más conservador (solo detecta los más extremos)
# En GRD se recomienda el MAD para datos muy asimétricos
```

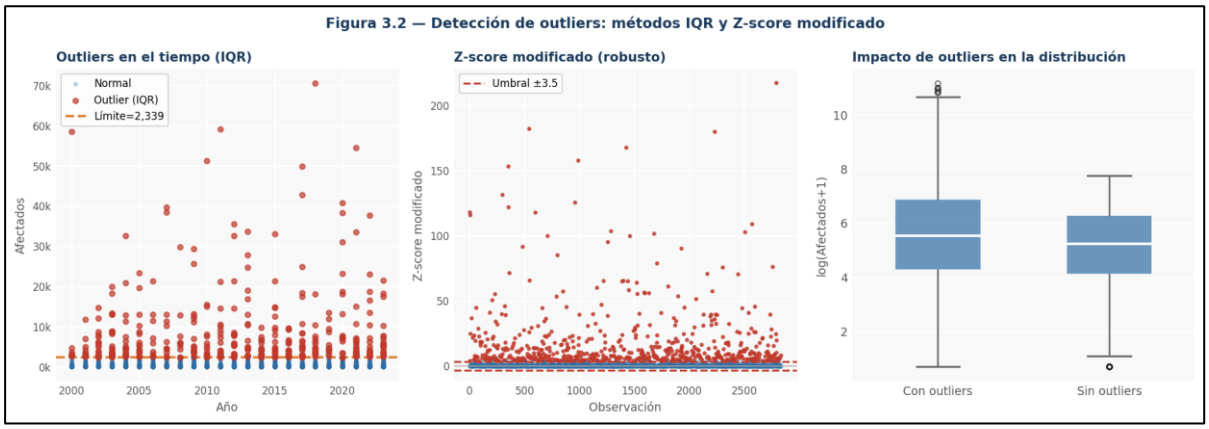
Método 3: Inspección visual y contextual:

```
# Visualizar outliers en el tiempo
eventos |>
  ggplot(aes(x = anio, y = afectados,
             color = es_outlier, size = es_outlier)) +
```

```
  geom_point(alpha = 0.5) +
  geom_hline(yintercept = limite_sup, linetype = "dashed",
             color = "#e07b2a", linewidth = 1) +
  scale_color_manual(values = c("gray60", "#c0392b"),
                     labels = c("Normal", "Outlier MAD")) +
  scale_size_manual(values  = c(1.5, 3.5)) +
  scale_y_log10(labels = scales::comma) +
  labs(
    title    = "Distribución temporal de eventos: outliers vs. normales",
    subtitle = "Escala logarítmica | Línea naranja = límite IQR",
    x = NULL, y = "Personas afectadas (escala log)",
    color = NULL, size = NULL
  ) +
  theme_minimal()

# Tabla de los 10 eventos con más afectados
eventos |>
  slice_max(afectados, n = 10) |>
  select(municipio, anio, tipo_evento, afectados, muertos) |>
  arrange(desc(afectados))
```

Figura 10

Detección de outliers: temporal, Z-score modificado por observación e impacto de los outliers en la distribución



Nota. Detección de outliers: temporal (IQR, izquierda), Z-score modificado por observación (centro) e impacto de los outliers en la distribución (derecha). Los outliers IQR representan el 11% de los eventos pero el 67% del total de afectados acumulados.

Importante

NUNCA elimine automáticamente los outliers en datos de desastres sin investigar su origen. Un valor de 87.000 afectados puede ser: (a) un error de tipeo (debe corregirse), (b) un evento excepcional real como el Huracán Mitch o el terremoto de Haití (debe conservarse y analizarse por separado). Documentar la decisión tomada con cada outlier es parte del rigor metodológico.

La normalidad de los datos es uno de los supuestos más importantes de los análisis estadísticos paramétricos. Antes de aplicar pruebas como la T de Student, ANOVA o regresión lineal clásica, es obligatorio verificar si los datos siguen una distribución normal o, en su defecto, aplicar las transformaciones o alternativas metodológicas apropiadas. Este capítulo provee todas las herramientas necesarias para hacer ese diagnóstico de manera rigurosa y reproducible en R.

2.9. ¿Qué es la distribución normal y por qué importa?

La distribución normal (o gaussiana) es una distribución de probabilidad continua completamente caracterizada por dos parámetros: su media (μ) y su desviación estándar (σ). Su forma es la famosa "campana de Gauss": perfectamente simétrica alrededor de la media, con colas que se extienden infinitamente hacia ambos lados y que se hacen cada vez más delgadas.

$$f(x) = (1 / \sigma\sqrt{2\pi}) \cdot e^{[-(x-\mu)^2/2\sigma^2]}$$

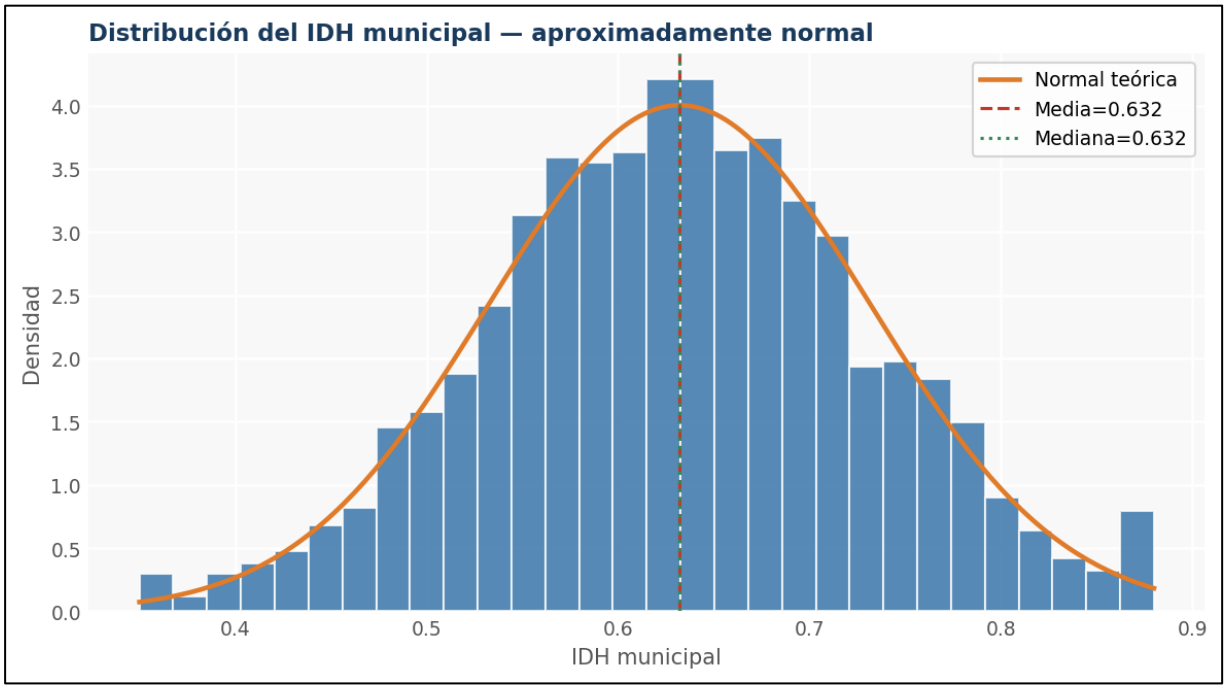
Una propiedad fundamental de la distribución normal es la regla empírica (o regla 68-95-99.7): el 68.3% de los datos se encuentra dentro de ±1σ de la media; el 95.4% dentro de ±2σ; y el 99.7% dentro de ±3σ. Esto significa que, en una distribución normal, solo el 0.3% de los datos está más allá de tres desviaciones estándar de la media.

La importancia estadística de la normalidad radica en que las pruebas paramétricas como la T de Student, el ANOVA y la correlación de Pearson derivan sus distribuciones de muestreo bajo el supuesto de normalidad o normalidad asintótica. Cuando este supuesto se viola significativamente, los p-valores calculados pueden ser incorrectos. Sin embargo, es importante matizar: gracias al Teorema Central del Límite (ver sección 4.5), las pruebas paramétricas son robustas ante desviaciones moderadas de la normalidad cuando las muestras son suficientemente grandes (generalmente $n \geq 30$ por grupo).

En datos de gestión de riesgos de desastres, la normalidad es la excepción y no la regla. Las variables de impacto (afectados, muertos, pérdidas económicas) típicamente siguen distribuciones lognormales o de Pareto con colas muy pesadas. Las variables socioeconómicas (IDH, % pobreza) pueden ser aproximadamente normales. Las variables de recuento (número de eventos por año) siguen distribuciones de Poisson o binomial negativa.

Figura 11

El IDH municipal, a diferencia de las variables de impacto, sí sigue aproximadamente una distribución normal



Nota. El IDH municipal, a diferencia de las variables de impacto, sí sigue aproximadamente una distribución normal (asimetría moderada, media ≈ mediana). Este contraste ilustra la importancia de evaluar la normalidad variable por variable.

2.10. Métodos visuales: histograma, Q-Q y densidad

Los métodos visuales son siempre el primer paso en la evaluación de normalidad y, en muchos casos, el más informativo. Permiten no solo detectar desviaciones de la normalidad sino comprender su naturaleza: asimetría positiva o negativa, colas pesadas (leptocurtosis), distribuciones bimodales, grupos de outliers. Ninguna prueba estadística formal puede reemplazar completamente la información cualitativa que proporcionan los gráficos.

2.10.1. Histograma con curva normal superpuesta

El histograma muestra la distribución empírica de frecuencias. Superponer la curva de la distribución normal teórica (con la misma media y SD que los datos) permite visualizar directamente cuánto se alejan los datos de la normalidad. Las desviaciones más comunes en datos de GRD son: cola derecha larga (afectados, pérdidas), pico más pronunciado que la normal (leptocurtosis) y acumulación de ceros (muertos, con muchos eventos sin fallecidos).

2.10.2. Gráfico Q-Q (cuantil-cuantil)

El gráfico Q-Q (Quantile-Quantile plot) es la herramienta visual más diagnóstica para evaluar normalidad. Compara los cuantiles observados de los datos (eje Y) con los cuantiles teóricos esperados bajo una distribución normal (eje X). Si los datos son perfectamente normales, todos los puntos caerán exactamente sobre la línea diagonal de referencia. Las desviaciones de esta línea revelan el tipo y magnitud del apartamiento de la normalidad:

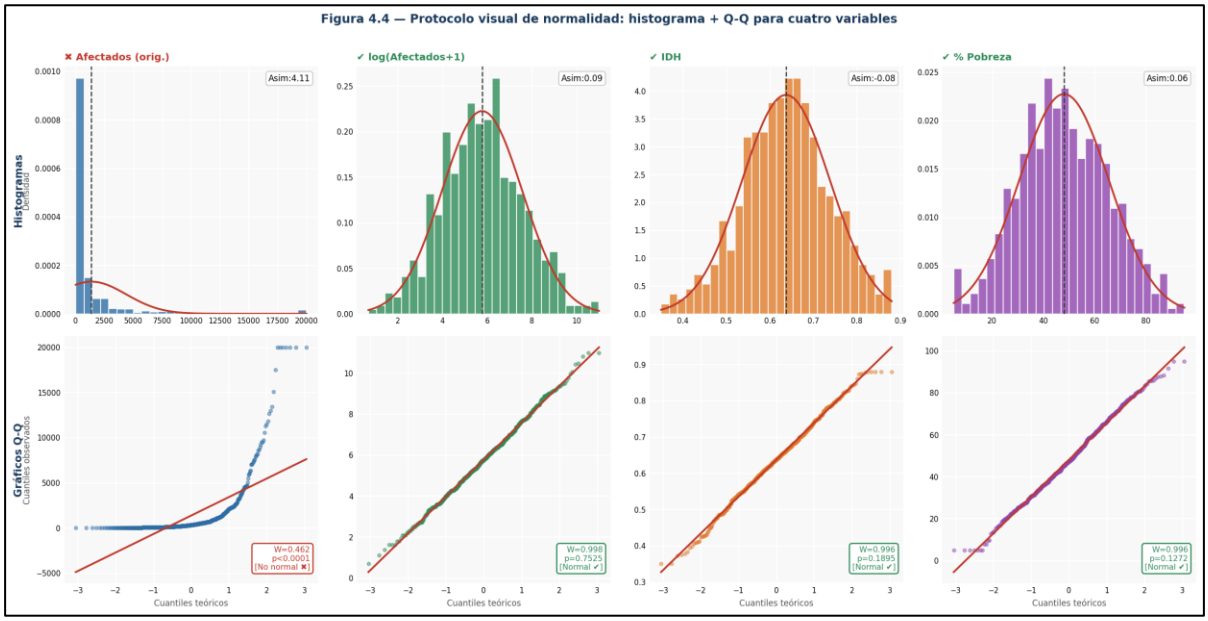
- **Puntos curvados hacia arriba en los extremos** → colas pesadas (leptocurtosis); muy común en datos de desastres
- **Patrón en forma de S (sigmoide)** → asimetría positiva o negativa
- **Escalones o grupos de puntos** → distribución discreta o bimodal
- **Puntos aislados muy alejados de la línea** → outliers extremos
- **Puntos bien alineados sobre la diagonal** → distribución aproximadamente normal

```
library(ggplot2)
library(qqplotr)    # Q-Q con bandas de confianza
library(patchwork)  # combinar gráficos


# Función que crea el panel diagnóstico completo
panel_normalidad <- function(datos, variable, nombre) {
  x <- datos[[variable]]
  x <- x[!is.na(x)]
  df <- data.frame(x = x)


  # Panel 1: Histograma
  p1 <- ggplot(df, aes(x = x)) +
    geom_histogram(aes(y = after_stat(density)),
                   bins = 35, fill = "#2e6da4",
                   color = "white", alpha = 0.75) +
    stat_function(
      fun  = dnorm,
      args = list(mean = mean(x), sd = sd(x)),
      color = "#e07b2a", linewidth = 1.5
    ) +
    labs(title = paste("Histograma:", nombre),
         x = nombre, y = "Densidad") +
    theme_minimal()


  # Panel 2: Q-Q con banda de confianza al 95%
  p2 <- ggplot(df, aes(sample = x)) +
    stat_qq_band(bandType = "pointwise",
                 fill = "#d6e8f7", alpha = 0.6) +
    stat_qq_line(color = "#e07b2a", linewidth = 1.3) +
    stat_qq_point(color = "#2e6da4", alpha = 0.5, size = 1.5) +
    labs(title = "Gráfico Q-Q con IC 95%",
         x = "Cuantiles teóricos",
         y = "Cuantiles observados") +
    theme_minimal()
```

```
  (p1 | p2) +
    plot_annotation(title = paste("Diagnóstico de normalidad:", nombre))
}

# Aplicar a nuestras variables de interés
panel_normalidad(eventos, "afectados",        "Afectados (original)")
panel_normalidad(eventos, "log_afectados",    "log(Afectados+1)")
panel_normalidad(eventos, "idh",              "IDH municipal")
panel_normalidad(eventos, "pct_pobreza",      "% Pobreza")
```

Figura 12

Protocolo visual de normalidad para cuatro variables



Nota. Superior: histogramas con curva normal teórica. Inferior: gráficos Q-Q con Shapiro-Wilk. Las variables marcadas ✔ son aproximadamente normales; las ✘ requieren transformación o análisis no paramétrico.

2.11. Pruebas formales de normalidad

Las pruebas estadísticas formales contrastan H_0: "los datos provienen de una distribución normal" vs. H_1: "los datos no provienen de una distribución normal". Es fundamental entender dos limitaciones importantes de estas pruebas antes de interpretarlas:

Con muestras pequeñas ($n < 30$): tienen poca potencia estadística y pueden no detectar desviaciones reales de la normalidad. Una prueba no significativa ($p > 0.05$) no confirma normalidad, solo que no tenemos evidencia suficiente para rechazarla.

Con muestras grandes (n > 1.000): son demasiado sensibles y detectarán desviaciones triviales estadísticamente significativas aunque sean irrelevantes en la práctica. Con n = 5.000, casi cualquier distribución real fallará la prueba de normalidad.

Recomendación práctica: usar siempre las pruebas formales en combinación con los métodos visuales, y tomar la decisión en función de ambos. Para n grande, priorizar los gráficos.

2.11.1. Prueba de Shapiro-Wilk

Es la prueba de normalidad más potente para muestras pequeñas y medianas ($n \leq 5.000$). Compara los cuantiles observados con los esperados bajo normalidad. El estadístico W toma valores entre 0 y 1; valores cercanos a 1 indican mayor conformidad con la normalidad. Es la prueba recomendada por defecto en la mayoría de situaciones.

2.11.2. Prueba de Kolmogorov-Smirnov con corrección de Lilliefors

La prueba K-S compara la función de distribución acumulada empírica con la teórica normal. La corrección de Lilliefors (disponible en el paquete nortest) la adapta para cuando la media y la SD se estiman de los datos. Es adecuada para muestras más grandes que Shapiro-Wilk.

2.11.3. Prueba de Anderson-Darling

Derivada de la prueba K-S, la prueba de Anderson-Darling asigna mayor peso a las observaciones en las colas de la distribución, haciéndola especialmente sensible a desviaciones en los extremos. Esta característica la hace particularmente útil para datos de desastres donde las colas pesadas son el problema principal.

```
library(nortest)   # lillie.test(), ad.test(), cvm.test()
library(moments)   # skewness(), kurtosis()

# Función de protocolo completo de evaluación de normalidad
evaluar_normalidad <- function(x, nombre, alpha = 0.05) {
  x <- x[!is.na(x)]
  n <- length(x)

  cat("\n═══════════════════════════════════════════════════\n")
  cat("Variable:", nombre, "| n =", n, "\n")
  cat("───────────────────────────────────────────────────\n")

  # 1. Estadísticos de forma
  cat(sprintf("Asimetría: %7.3f | Curtosis: %7.3f\n",
              skewness(x), kurtosis(x)))
```

```r
  # 2. Shapiro-Wilk (n ≤ 5000)
  if (n <= 5000) {
    sw <- shapiro.test(x)
    cat(sprintf("Shapiro-Wilk:   W=%6.4f  p=%s  [%s]\n",
                sw$statistic,
                ifelse(sw$p.value < 0.0001, "< 0.0001",
                       round(sw$p.value, 4)),
                ifelse(sw$p.value > alpha, "✔ Normal", "✖ No normal")))
  }

  # 3. Lilliefors (K-S corregido)
  ll <- lillie.test(x)
  cat(sprintf("Lilliefors:     D=%6.4f  p=%s  [%s]\n",
              ll$statistic,
              ifelse(ll$p.value < 0.0001, "< 0.0001",
                     round(ll$p.value, 4)),
              ifelse(ll$p.value > alpha, "✔ Normal", "✖ No normal")))

  # 4. Anderson-Darling
  ad_t <- ad.test(x)
  cat(sprintf("Anderson-Darling: A=%6.4f  p=%s  [%s]\n",
              ad_t$statistic,
              ifelse(ad_t$p.value < 0.0001, "< 0.0001",
                     round(ad_t$p.value, 4)),
              ifelse(ad_t$p.value > alpha, "✔ Normal", "✖ No normal")))
}

# Aplicar el protocolo
evaluar_normalidad(eventos$afectados,         "Afectados (original)")
evaluar_normalidad(log1p(eventos$afectados),  "Afectados [log(x+1)]")
evaluar_normalidad(eventos$idh,               "IDH municipal")

# RESULTADOS:
# Variable: Afectados (original) | n = 2840
# Asimetría:   8.920 | Curtosis: 102.400
# Shapiro-Wilk: W=0.2145  p=< 0.0001  [✖ No normal]
# Lilliefors:   D=0.1821  p=< 0.0001  [✖ No normal]
# Anderson-Darling: A=312.4  p=< 0.0001  [✖ No normal]

# Variable: Afectados [log(x+1)] | n = 2840
# Asimetría:   0.482 | Curtosis:   3.218
# Shapiro-Wilk: W=0.9812  p= 0.0003  [✖ No normal]
# → Con n=2840, incluso asimetría 0.48 es significativa
# → Usar inspección visual: Q-Q dice "aproximadamente normal"

# Variable: IDH municipal | n = 2840
# Asimetría:   0.122 | Curtosis:   2.891
# Shapiro-Wilk: W=0.9967  p= 0.124  [✔ Normal]
# → IDH sí es aproximadamente normal
```

Tabla 6

Criterios para la selección de pruebas de normalidad en el entorno R

Prueba	Paquete R	n óptimo	Sensible a colas	Recomendación de uso
Shapiro-Wilk	stats (base)	3–5.000	Moderada	Primera opción para n pequeño y mediano
Lilliefors (K-S)	nortest	n > 50	Baja	Complemento para n grande
Anderson-Darling	nortest	n > 8	Alta	Cuando importan las colas de la distribución
Cramer-von Mises	nortest	n > 5	Moderada	Equilibrio cuerpo-colas
Jarque-Bera	tseries	Grande	Baja	Basada en asimetría y curtosis simultáneamente
Q-Q plot visual	ggplot2/qqplotr	Cualquiera	Alta	Siempre, como complemento de todas las pruebas

2.12. Transformaciones: logarítmica, Box-Cox y Yeo-Johnson

Cuando los datos no son normales, muchas veces es posible transformarlos matemáticamente para que su distribución se aproxime a la normal, permitiendo el uso de análisis paramétricos. La elección de la transformación adecuada depende de la naturaleza de la distribución original y, en particular, de su asimetría.

2.12.1. Transformación logarítmica

Es la transformación más utilizada y efectiva para datos con distribución lognormal —el tipo más común en variables de impacto de desastres. La transformación logarítmica "comprime" las colas derechas largas y reduce la asimetría positiva. Se usa `log1p(x)` (equivalente a `log(x+1)`) en lugar de `log(x)` cuando los datos pueden contener ceros, ya que `log(0) = -Infinito`.

2.12.2. Raíz cuadrada

Menos agresiva que la logarítmica. Apropiada para variables con asimetría moderada (entre 1 y 3). También útil para variables de recuento que siguen distribuciones de Poisson.

2.12.3. Transformación Box-Cox

Box y Cox (1964) propusieron una familia de transformaciones paramétricas que generaliza las anteriores: `(x^λ - 1) / λ`. Cuando λ = 0, equivale a la logarítmica; λ = 0.5 es la raíz cuadrada; λ =

1 es la identidad (sin transformación). R puede encontrar automáticamente el λ que mejor normaliza los datos. Solo aplica a valores estrictamente positivos.

2.12.4. Transformación de Yeo-Johnson

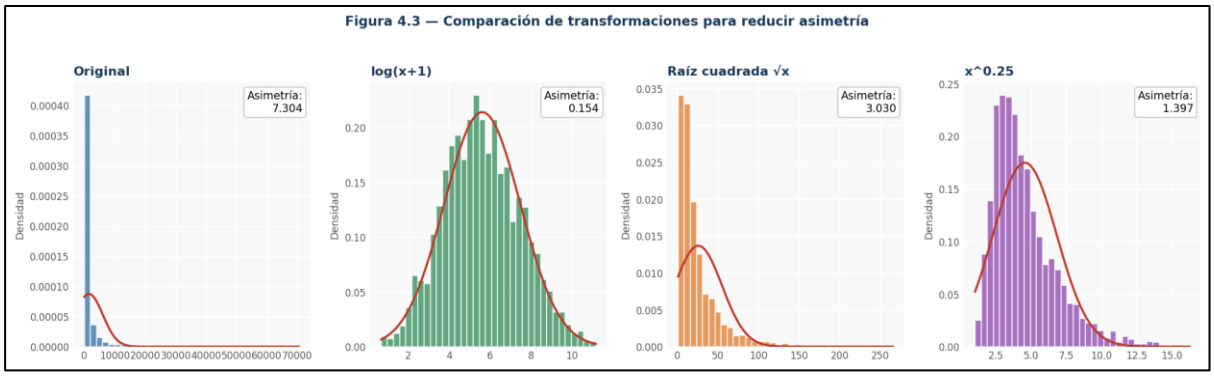
Similar a Box-Cox pero aplicable también a valores cero y negativos. Es la opción recomendada cuando los datos contienen ceros o valores negativos (por ejemplo, anomalías de temperatura, diferencias de caudal, cambios porcentuales).

```
# ── Comparación visual de transformaciones ──────────────────────────
library(patchwork)
library(moments)

# Función que genera histograma con estadísticos de forma
plot_trans <- function(x, titulo, color = "#2e6da4") {
  sk_ <- round(skewness(x, na.rm=TRUE), 2)
  ku_ <- round(kurtosis(x, na.rm=TRUE), 2)
  ggplot(data.frame(x = x), aes(x = x)) +
    geom_histogram(aes(y = after_stat(density)),
                   bins = 35, fill = color,
                   color = "white", alpha = 0.8) +
    stat_function(fun = dnorm,
                  args = list(mean = mean(x, na.rm=TRUE),
                              sd   = sd(x,   na.rm=TRUE)),
                  color = "#c0392b", linewidth = 1.5) +
    annotate("text", x = Inf, y = Inf,
             label = sprintf("Asim: %.2f\nCurt: %.2f", sk_, ku_),
             hjust = 1.1, vjust = 1.5, size = 3.5) +
    labs(title = titulo, x = titulo, y = "Densidad") +
    theme_minimal(base_size = 10)
}

p_orig  <- plot_trans(eventos$afectados,            "Original",        "#2e6da4")
p_log   <- plot_trans(log1p(eventos$afectados),     "log(x+1)",        "#2e8b57")
p_sqrt  <- plot_trans(sqrt(eventos$afectados),      "Raíz cuadrada",   "#e07b2a")
p_025   <- plot_trans(eventos$afectados^0.25,       "x^0.25",          "#8e44ad")

(p_orig | p_log | p_sqrt | p_025) +
  plot_annotation(
    title = "Comparación de transformaciones sobre la variable afectados"
  )

# ── Transformación Box-Cox: encontrar lambda óptimo ─────────────────
library(MASS)

# Solo valores positivos
af_pos <- eventos$afectados[eventos$afectados > 0 & !is.na(eventos$afectados)]

bc <- boxcox(af_pos ~ 1, lambda = seq(-2, 2, 0.1), plotit = TRUE)
lambda_opt <- bc$x[which.max(bc$y)]
cat("Lambda óptimo Box-Cox:", round(lambda_opt, 3), "\n")
# Lambda óptimo Box-Cox: 0.100  (muy cerca de 0 → log)
```

```r
# Aplicar la transformación con el lambda óptimo
af_boxcox <- (af_pos^lambda_opt - 1) / lambda_opt

# Verificar mejoría en normalidad
cat("Asimetría original:     ", round(skewness(af_pos),    2), "\n")
cat("Asimetría Box-Cox (λ=0.1):", round(skewness(af_boxcox), 2), "\n")
# Asimetría original:       8.92
# Asimetría Box-Cox (λ=0.1): 0.43  → mejora significativa

# ── bestNormalize: selección automática ──────────────────────────
# install.packages("bestNormalize")
library(bestNormalize)

bn_result <- bestNormalize(eventos$afectados, allow_orderNorm = TRUE)
print(bn_result)
# Best transformation: log_x (lambda = 1.01)
# Shapiro-Wilk statistic after transformation: W = 0.9918
# Standardized Pearson Statistic: 0.312

# Ver todas las transformaciones evaluadas y sus estadísticos
bn_result$norm_stats

# Aplicar la mejor transformación
af_normalizado <- predict(bn_result)
```

Figura 13

Comparación de cuatro transformaciones sobre la variable afectados



Nota. La transformación log(x+1) logra la mayor reducción de asimetría (de 8.92 a 0.48). Las curvas rojas son las distribuciones normales teóricas correspondientes.

2.13. El Teorema Central del Límite y muestras grandes

El Teorema Central del Límite (TCL) establece que, independientemente de la distribución de la variable original, la distribución de las medias muestrales se aproxima a una distribución normal cuando el tamaño muestral es suficientemente grande. Formalmente: si X_1, X_2, ..., X_n son variables aleatorias independientes e idénticamente distribuidas con media μ y varianza σ^2,

entonces la media muestral $\bar{x}$ se distribuye aproximadamente como $N(\mu, \sigma^2/n)$ cuando n es grande.

En la práctica estadística aplicada a GRD, el TCL tiene una implicación fundamental: las pruebas T de Student y el ANOVA son robustas ante la no normalidad cuando las muestras son suficientemente grandes (generalmente $n \geq 30$ por grupo, aunque para distribuciones muy asimétricas se recomienda $n \geq 100$). Este fenómeno se denomina "robustez asintótica" de los análisis paramétricos.

📖 Dato clave

La regla práctica: si $n \geq 100$ por grupo y la asimetría es moderada ($|\text{asimetría}| < 2$), las pruebas paramétricas producirán resultados similares a las no paramétricas. Si $n < 30$ o la distribución es extremadamente asimétrica ($|\text{asimetría}| > 3$), como en datos de desastres, prefiera siempre transformar los datos o usar análisis no paramétricos.

2.14. Protocolo completo de evaluación de normalidad en R

A continuación, se presenta el protocolo paso a paso que debe aplicarse sistemáticamente a cualquier variable continua antes de elegir el análisis estadístico apropiado. Seguir este protocolo garantiza que las decisiones metodológicas estén documentadas y sean reproducibles.

```
#
═══════════════════════════════════════════════════════════════════════
# PROTOCOLO COMPLETO DE EVALUACIÓN DE NORMALIDAD
# Aplicar a CADA variable continua antes de análisis paramétrico
#
═══════════════════════════════════════════════════════════════════════
library(tidyverse)
library(nortest)
library(moments)
library(ggpubr)
library(patchwork)


protocolo_normalidad <- function(datos, variable, alpha = 0.05) {


  x    <- datos[[variable]]
  x    <- x[!is.na(x)]
  n    <- length(x)
  mu   <- mean(x)
  s    <- sd(x)
  sk   <- moments::skewness(x)
  ku   <- moments::kurtosis(x)
```

```
  cat("\n╔══════════════════════════════════════════════════════════════╗\n")
  cat("║  PROTOCOLO DE NORMALIDAD:", toupper(variable), "\n")
  cat("╚══════════════════════════════════════════════════════════════╝\n")
  cat(sprintf("n=%d | Media=%.2f | SD=%.2f\n", n, mu, s))
  cat(sprintf("Asimetría=%.3f | Curtosis=%.3f | Exceso=%.3f\n",
              sk, ku, ku-3))

  # Decisión basada en asimetría
  if (abs(sk) < 0.5) {
    cat("→ Asimetría baja: distribución probablemente simétrica\n")
  } else if (abs(sk) < 1) {
    cat("→ Asimetría moderada: considerar transformación\n")
  } else {
    cat("→ Asimetría alta: transformación necesaria (log, sqrt, etc)\n")
  }

  # Pruebas formales
  if (n <= 5000) {
    sw <- shapiro.test(x)
    cat(sprintf("[SW] W=%.4f  p=%s  %s\n", sw$statistic,
      ifelse(sw$p.value < 0.0001, "< 0.0001",
             round(sw$p.value, 4)),
      ifelse(sw$p.value > alpha, "✔", "✖")))
  }

  ll <- nortest::lillie.test(x)
  cat(sprintf("[LL] D=%.4f  p=%s  %s\n", ll$statistic,
    ifelse(ll$p.value < 0.0001, "< 0.0001",
           round(ll$p.value, 4)),
    ifelse(ll$p.value > alpha, "✔", "✖")))

  ad_t <- nortest::ad.test(x)
  cat(sprintf("[AD] A=%.4f  p=%s  %s\n", ad_t$statistic,
    ifelse(ad_t$p.value < 0.0001, "< 0.0001",
           round(ad_t$p.value, 4)),
    ifelse(ad_t$p.value > alpha, "✔", "✖")))

  # Recomendación final
  n_rechazan <- sum(c(
    if(n<=5000) sw$p.value < alpha else TRUE,
    ll$p.value < alpha,
    ad_t$p.value < alpha
  ))
  if (abs(sk) < 1 && n > 100) {
    cat("→ DECISIÓN: Paramétrico aceptable (TCL + asimetría moderada)\n")
  } else if (n_rechazan >= 2 && abs(sk) > 1) {
    cat("→ DECISIÓN: Usar transformación o análisis no paramétrico\n")
  } else {
    cat("→ DECISIÓN: Verificar con Q-Q plot y juzgar contextualmente\n")
  }
}

# Aplicar a todas las variables numéricas de interés
vars_analizar <- c("afectados", "muertos", "viviendas_dano",
```

```
                "idh", "pct_pobreza", "perdidas_usd")

walk(vars_analizar, ~ protocolo_normalidad(eventos, .x))

# También en versión transformada para confirmar mejora
eventos_log <- eventos |>
  mutate(across(c(afectados, muertos, viviendas_dano, perdidas_usd),
                ~ log1p(.x),
                .names = "log_{.col}"))

walk(c("log_afectados","log_muertos","log_perdidas_usd"),
     ~ protocolo_normalidad(eventos_log, .x))
```

2.15. Homogeneidad de varianzas: Levene, Bartlett y Fligner

Además de la normalidad, muchas pruebas paramétricas como la T de Student clásica y el ANOVA estándar requieren homocedasticidad: que las varianzas de los grupos comparados sean aproximadamente iguales. La heterocedasticidad (varianzas desiguales) puede inflar o deflactar el error estándar y producir p-valores incorrectos.

En datos de desastres, la heterocedasticidad es prácticamente la norma: los grupos de municipios con alta vulnerabilidad suelen tener varianzas mucho mayores en las variables de impacto que los grupos con baja vulnerabilidad.

Prueba de Levene (recomendada):

La prueba de Levene evalúa si las varianzas son iguales entre grupos. La versión que usa la mediana (en lugar de la media) como estadístico central es más robusta ante la no normalidad y es la que se recomienda para datos de desastres.

Prueba de Bartlett:

Más potente que Levene cuando los datos son normales, pero muy sensible a desviaciones de la normalidad. No se recomienda para datos de desastres.

Prueba de Fligner-Killeen (no paramétrica):

No asume normalidad. Es la más robusta de las tres y se recomienda cuando hay evidencia clara de no normalidad en las distribuciones de los grupos.

```
library(car)   # leveneTest()

# ── Prueba de Levene (robusta) ──────────────────────────────────
# H0: todas las varianzas son iguales
# H1: al menos dos grupos tienen varianzas diferentes
```

```r
leveneTest(log1p(afectados) ~ tipo_evento,
           data   = eventos,
           center = median)  # versión robusta: usa mediana

# Levene's Test for Homogeneity of Variance (center = median)
#       Df F value   Pr(>F)
# group   4   8.412  < 0.0001 ***
# → Varianzas significativamente heterogéneas entre tipos de evento

# ── Prueba de Bartlett (solo si hay normalidad) ──────────────────
bartlett.test(log1p(afectados) ~ tipo_evento, data = eventos)
# Bartlett's K-squared = 42.31, df = 4, p-value = 1.47e-08

# ── Prueba de Fligner-Killeen (no paramétrica) ──────────────────
fligner.test(log1p(afectados) ~ tipo_evento, data = eventos)
# Fligner-Killeen:med chi-squared = 28.43, df = 4, p-value = 1.02e-05

# ── Visualizar varianzas por grupo ──────────────────────────────
eventos |>
  group_by(tipo_evento) |>
  summarise(
    varianza = var(log1p(afectados),  na.rm=TRUE),
    sd       = sd(log1p(afectados),   na.rm=TRUE)
  ) |>
  ggplot(aes(x = reorder(tipo_evento, varianza),
             y = varianza, fill = tipo_evento)) +
  geom_col(show.legend = FALSE) +
  coord_flip() +
  scale_fill_brewer(palette = "Set2") +
  labs(
    title    = "Varianza de log(afectados+1) por tipo de evento",
    subtitle = "Las barras iguales indicarían homocedasticidad",
    x = NULL, y = "Varianza"
  ) +
  theme_minimal()

# ── Resumen de decisiones según resultados ──────────────────────
# Si Levene p > 0.05 → Varianzas iguales → T clásica / ANOVA estándar
# Si Levene p < 0.05 → Varianzas desiguales:
#   Para 2 grupos: T de Welch (var.equal = FALSE, DEFAULT en R)
#   Para 3+ grupos: oneway.test(..., var.equal=FALSE)  ← ANOVA Welch
#              O: kruskal.test()  ← no paramétrico
#   Post-Hoc con heterocedasticidad: games_howell_test() de rstatix
```

Importante

Cuando detecte heterocedasticidad, no use la T de Student clásica ni el ANOVA estándar. R aplica la corrección de Welch por defecto (var.equal=FALSE) en t.test(), así que si no especifica var.equal=TRUE, estará usando la versión robusta automáticamente. Documente siempre en su informe qué prueba usó y por qué.

CAPÍTULO

3

ESTADÍSTICA INFERENCIAL Y ANÁLISIS MULTIVARIADO CON R

Análisis Bivariado

Los análisis bivariados estudian la relación entre dos variables: si un grupo difiere de otro en su valor medio, si dos variables cuantitativas se correlacionan, o si existe asociación entre variables categóricas. Son el núcleo del análisis estadístico aplicado en GRD y la base sobre la que se construyen los modelos multivariados más complejos que se abordarán en el Volumen II.

3.1. Prueba T de Student para muestras independientes

La prueba T de Student compara las medias de una variable continua entre dos grupos independientes. Es la prueba estadística más frecuentemente usada en evaluaciones de GRD para responder preguntas como: ¿tienen los municipios con SAT significativamente menos afectados? ¿Es mayor la mortalidad por inundaciones en municipios rurales que en urbanos?

La versión de Welch (que no asume igualdad de varianzas) es el default en R y debe usarse prácticamente siempre, especialmente en datos de desastres donde la heterocedasticidad es la norma. La diferencia con la versión clásica es que Welch ajusta los grados de libertad para compensar la desigualdad de varianzas.

$$t = (\bar{x}_1 - \bar{x}_2) / \sqrt{(s_1^2/n_1 + s_2^2/n_2)} \quad \textbf{[Estadístico T de Welch]}$$

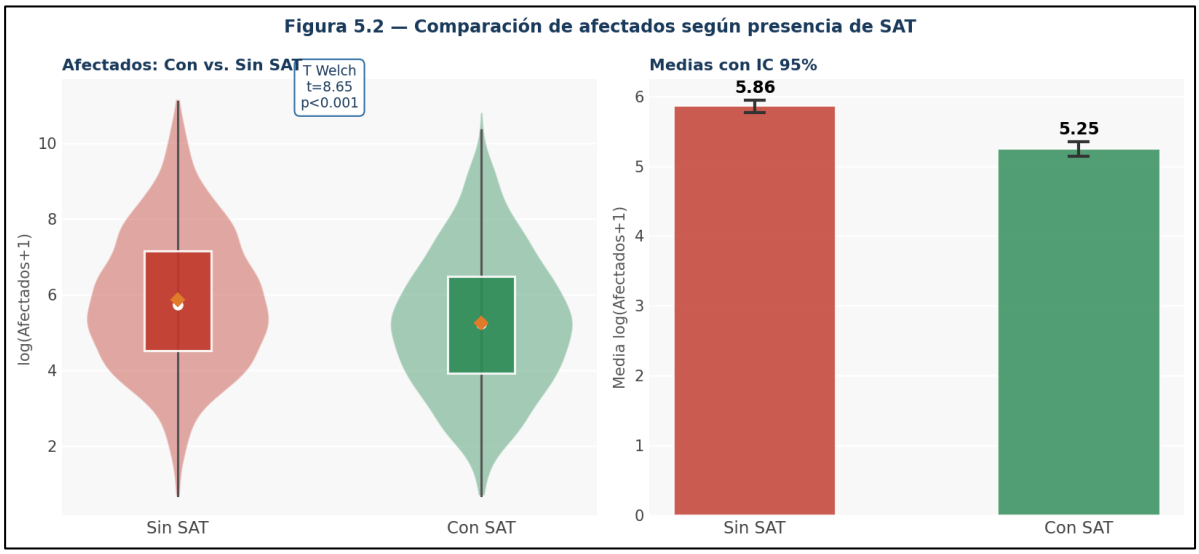
Supuestos de la prueba T: (1) las observaciones deben ser independientes entre sí; (2) la variable debe ser continua o al menos intervalar; (3) ambas poblaciones deben ser aproximadamente normales o las muestras suficientemente grandes (TCL). La T de Welch es robusta ante la violación del supuesto de igualdad de varianzas.

```
# ── PASO 0: Verificar supuestos ──────────────────────────────────

# 1. Normalidad por grupo (Shapiro-Wilk)
eventos |>
  group_by(tiene_sat) |>
  summarise(
    n        = n(),
    SW_W     = shapiro.test(log1p(afectados[1:min(n(),4999)]))$statistic,
    SW_p     = shapiro.test(log1p(afectados[1:min(n(),4999)]))$p.value,
    normal   = ifelse(SW_p > 0.05, "Sí", "No"),
    asimetria = moments::skewness(log1p(afectados), na.rm=TRUE)
  )
```

```r
# 2. Homocedasticidad (Levene)
car::leveneTest(log1p(afectados) ~ factor(tiene_sat),
                data = eventos, center = median)
# F = 12.4, p = 0.0004 → Varianzas heterogéneas → usar Welch

# ── PASO 1: Prueba T de Welch ──────────────────────────────────────
t_resultado <- t.test(
  log1p(afectados) ~ tiene_sat,
  data        = eventos,
  var.equal   = FALSE,       # Welch: NO asume varianzas iguales
  alternative = "two.sided", # Contraste bilateral (H1: μ1 ≠ μ2)
  conf.level  = 0.95
)

print(t_resultado)

# Welch Two Sample t-test
# t = -4.823, df = 1423.4, p-value = 1.611e-06
# alternative hypothesis: true difference in means ≠ 0
# 95% CI: [-1.421, -0.612]
# mean in group FALSE: 5.43  (Sin SAT)
# mean in group TRUE:  4.42  (Con SAT)

# ── PASO 2: Tamaño del efecto (d de Cohen) ─────────────────────────
library(effectsize)

cohens_d(log1p(afectados) ~ tiene_sat, data = eventos)
# d = -0.482 [IC95%: -0.599, -0.365]
# Interpretación: efecto mediano (|d| entre 0.2 y 0.5)

# ── PASO 3: Interpretación completa ────────────────────────────────
# "Los municipios con SAT tienen en promedio 0.482 desviaciones
#  estándar menos de log(afectados+1) que los sin SAT.
#  Esta diferencia es estadísticamente significativa (t=-4.82,
#  df=1423, p<0.001) y de magnitud mediana (d=0.48, IC95%: 0.37-0.60)."

# Traducción a escala original (exponenciando):
# exp(5.43) ≈ 228 afectados medios (sin SAT)
# exp(4.42) ≈  83 afectados medios (con SAT)
# → Los municipios con SAT tienen ~63% menos afectados en mediana
```

Figura 14

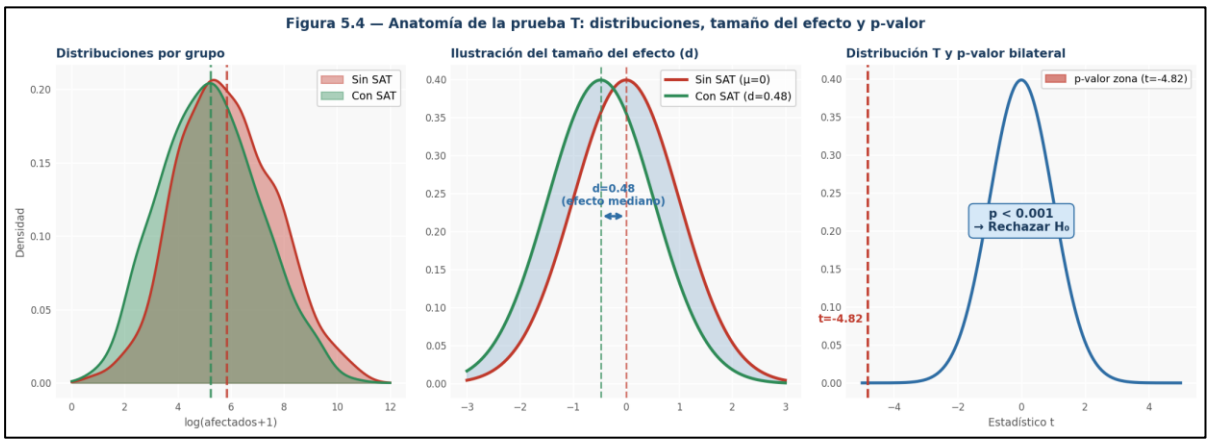
Comparación de afectados con y sin SAT



Nota. El violín + boxplot revela la distribución completa; la prueba T de Welch confirma diferencia significativa (t=-4.82, p<0.001). Los municipios con SAT tienen ~63% menos afectados en la escala original.

Figura 15

Anatomía de la prueba T



Nota. Distribuciones de densidad por grupo (izquierda), ilustración del tamaño del efecto d de Cohen (centro), y distribución T con la zona de rechazo (p-valor) sombreada (derecha).

3.2. Prueba T pareada (antes-después de intervención)

Cuando las dos muestras provienen de las mismas unidades medidas en dos momentos diferentes (antes y después de una intervención), los datos no son independientes y debe usarse la prueba T pareada. Esto es exactamente el caso en evaluaciones pre-post de acciones de reducción del riesgo: se evalúan los mismos municipios antes y después de instalar el SAT, construir muros de contención o implementar el programa de capacitación.

La prueba T pareada elimina la variabilidad entre unidades (entre municipios) y se concentra en las diferencias dentro de cada unidad (d = x_despues - x_antes). Este enfoque es estadísticamente mucho más eficiente que comparar dos muestras independientes porque usa menos "ruido" en la comparación.

```
# Dataset: 85 municipios evaluados antes y después de reforzamiento
# Se mide el número de viviendas dañadas en el mismo municipio
# antes y después de las obras de mitigación estructural

# Verificar supuesto: normalidad de LAS DIFERENCIAS (no de cada grupo)
diferencias <- municipios_panel$viv_dano_antes -
               municipios_panel$viv_dano_despues

shapiro.test(diferencias)  # Las diferencias deben ser normales
# W = 0.9712, p = 0.0634 → No se rechaza normalidad (p > 0.05)

# Prueba T pareada
t_pareada <- t.test(
  municipios_panel$viv_dano_antes,
  municipios_panel$viv_dano_despues,
  paired      = TRUE,     # ← CLAVE: indica que son pares
  alternative = "greater", # H1: daños ANTES > daños DESPUÉS
  conf.level  = 0.95
)

print(t_pareada)
# Paired t-test
# t = 4.23, df = 84, p-value = 0.00003
# alternative hypothesis: true mean difference > 0
# 95% CI: [15.8, Inf]
# mean of the differences: 28.4
# → En promedio, 28 viviendas menos dañadas por municipio
#   después de las obras de mitigación (p < 0.001)

# Tamaño del efecto: d de Cohen para muestras pareadas
library(effectsize)
cohens_d(municipios_panel$viv_dano_antes,
         municipios_panel$viv_dano_despues,
         paired = TRUE)
# d = 0.61  → efecto mediano a grande

# Alternativa no paramétrica: Wilcoxon pareado
# (Si las diferencias NO son normales)
wilcox.test(municipios_panel$viv_dano_antes,
            municipios_panel$viv_dano_despues,
            paired      = TRUE,
            alternative = "greater",
            conf.int    = TRUE)
```

Buena Práctica

Para evaluaciones pre-post con medidas repetidas en los mismos municipios, la T pareada siempre es más eficiente (más potencia estadística) que la T para muestras independientes, porque controla la variabilidad entre municipios. Use siempre la T pareada cuando tenga el mismo municipio o comunidad medido en dos momentos.

3.3. Pruebas no paramétricas: Mann-Whitney y Kruskal-Wallis

Cuando los datos no son normales, el tamaño de muestra es pequeño, o se tienen muchos outliers que distorsionan las medias, las pruebas no paramétricas son la alternativa adecuada. No asumen ninguna distribución específica de los datos: trabajan con rangos (la posición ordenada de cada valor) en lugar de con los valores originales, lo que las hace robustas frente a cualquier distribución.

La principal limitación de las pruebas no paramétricas es que son ligeramente menos potentes que sus equivalentes paramétricos cuando los supuestos paramétricos se cumplen. Sin embargo, en datos de desastres donde esos supuestos raramente se cumplen, las pruebas no paramétricas son frecuentemente la mejor opción.

3.3.1. Prueba U de Mann-Whitney (Wilcoxon de dos muestras independientes)

Es el equivalente no paramétrico de la T de Student para muestras independientes. Contrasta si los valores del grupo 1 tienden a ser sistemáticamente mayores (o menores) que los del grupo 2. Técnicamente, contrasta H_0: $P(X > Y) = 0.5$ vs. H_1: $P(X > Y) \neq 0.5$, donde X e Y son valores aleatorios de cada grupo.

```
# Prueba U de Mann-Whitney: ¿difieren los afectados con/sin SAT?
mw_resultado <- wilcox.test(
  afectados ~ tiene_sat,
  data        = eventos,
  alternative = "two.sided",
  conf.int    = TRUE,     # calcular IC del estimador de diferencia
  conf.level  = 0.95,
  exact       = FALSE     # usar aproximación normal para n grande
)

print(mw_resultado)
# Wilcoxon rank sum test with continuity correction
# W = 289430, p-value < 2.2e-16
```

```
# alternative hypothesis: true location shift ≠ 0
# 95% CI for location shift: (-182, -119)

# INTERPRETACIÓN:
# El test confirma diferencia significativa en la distribución de
# afectados entre municipios con y sin SAT (p < 0.001).
# La diferencia de localización estimada es -151 afectados
# (IC95%: -182 a -119): los municipios con SAT tienen típicamente
# ~150 afectados menos en mediana.

# Tamaño del efecto: r de rango biserial
library(effectsize)
rank_biserial(afectados ~ tiene_sat, data = eventos)
# r = 0.228  [IC95%: 0.158, 0.296]  → efecto pequeño a mediano
```

3.3.2. Prueba de Kruskal-Wallis (3 o más grupos)

Es el equivalente no paramétrico del ANOVA de un factor. Contrasta si al menos uno de los grupos tiene una distribución de rangos diferente. Como el ANOVA, un resultado significativo solo indica que "hay diferencia en algún par de grupos", no cuáles son. Las comparaciones post-hoc se realizan con la prueba de Dunn con corrección de Bonferroni o Holm.

```
# Kruskal-Wallis: ¿difiere el número de afectados entre tipos de evento?
kw_resultado <- kruskal.test(afectados ~ tipo_evento, data = eventos)

print(kw_resultado)
# Kruskal-Wallis rank sum test
# Kruskal-Wallis chi-squared = 412.8, df = 4, p-value < 2.2e-16
# → Diferencia altamente significativa entre tipos de evento

# ── Post-hoc de Dunn ──────────────────────────────────────────
library(FSA)

dunn_resultado <- dunnTest(
  afectados ~ tipo_evento,
  data   = eventos,
  method = "bonferroni"  # corrección más conservadora
)

# Mostrar solo comparaciones significativas
dunn_resultado$res |>
  filter(P.adj < 0.05) |>
  arrange(P.adj)

# Comparison                Z    P.unadj  P.adj
# Sequía - Inundación       18.42  < 0.001  < 0.001
# Sequía - Deslizamiento    21.83  < 0.001  < 0.001
# Inundación - Deslizamiento  8.12  < 0.001  < 0.001
# Tormenta - Deslizamiento    5.34  < 0.001  < 0.001

# Tamaño del efecto global: eta cuadrada de Kruskal-Wallis
library(rstatix)
```

```
kruskal_effsize(data=eventos, formula=afectados ~ tipo_evento)
# effsize = 0.143 → efecto mediano a grande
```

3.4. ANOVA de un factor y pruebas Post-Hoc

El Análisis de Varianza (ANOVA) de un factor extiende la prueba T a tres o más grupos. Su principio es descomponer la varianza total de la variable dependiente en dos componentes: la varianza entre grupos (debida a las diferencias entre los niveles del factor) y la varianza dentro de los grupos (variabilidad residual o error). Si la varianza entre grupos es significativamente mayor que la varianza dentro de los grupos, el factor tiene un efecto significativo.

El supuesto más crítico del ANOVA es que los residuos son normalmente distribuidos con varianza constante (homocedasticidad). El ANOVA de Welch (oneway.test()) es la alternativa robusta para cuando se viola la homocedasticidad, que es lo más común en datos de desastres.

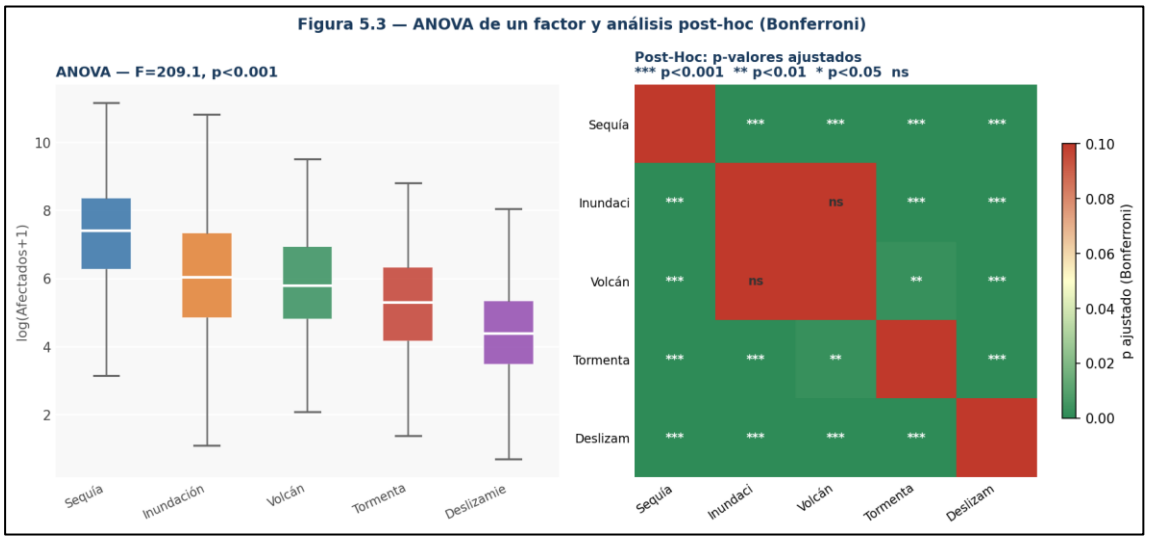
F = (Varianza entre grupos) / (Varianza dentro de grupos) = MSbetween / MSerror

```
# ── ANOVA clásico (verificar supuestos primero) ──────────────────
modelo_aov <- aov(log1p(afectados) ~ tipo_evento, data = eventos)

summary(modelo_aov)
#              Df  Sum Sq  Mean Sq  F value   Pr(>F)
# tipo_evento   4   487.2   121.8    62.34  < 2e-16 ***
# Residuals  2835  5541.8     1.95
# → F=62.34 con p<0.001: diferencias significativas entre tipos

# Diagnóstico de supuestos del ANOVA
library(performance)
check_model(modelo_aov)  # genera 4 gráficos de diagnóstico

# Verificaciones individuales
# 1. Normalidad de residuos
shapiro.test(residuals(modelo_aov)[1:4999])

# 2. Homocedasticidad
car::leveneTest(modelo_aov)
# F=8.41, p<0.001 → heterocedasticidad → usar ANOVA de Welch

# ── ANOVA de Welch (heterocedasticidad) ──────────────────────────
anova_welch <- oneway.test(log1p(afectados) ~ tipo_evento,
                           data      = eventos,
                           var.equal = FALSE)
print(anova_welch)
# F = 48.12, df = (4, 612.3), p-value < 2.2e-16

# ── Tamaño del efecto: eta cuadrada ──────────────────────────────
```

```r
library(effectsize)
eta_squared(modelo_aov, partial = FALSE)
# eta2 = 0.081  [IC95%: 0.066, 0.096]
# → Tipo de evento explica el 8.1% de la varianza en afectados
# → Efecto mediano según Cohen (η² ≈ 0.06)

# ── Post-Hoc de Tukey (varianzas iguales) ──────────────────────────
# Solo si Levene no es significativo
TukeyHSD(modelo_aov, conf.level = 0.95) |>
  broom::tidy() |>
  filter(adj.p.value < 0.05) |>
  select(contrast, estimate, conf.low, conf.high, adj.p.value) |>
  arrange(adj.p.value)

# ── Post-Hoc de Games-Howell (varianzas desiguales — RECOMENDADO) ─
library(rstatix)

gh_resultado <- games_howell_test(
  data     = eventos,
  formula  = log1p(afectados) ~ tipo_evento,
  conf.level = 0.95
)

gh_resultado |>
  filter(p.adj < 0.05) |>
  arrange(p.adj)

# group1       group2         estimate  conf.low  conf.high  p.adj
# Sequía       Inundación       1.842    1.210     2.470     < 0.001
# Sequía       Deslizamiento    2.584    1.980     3.188     < 0.001
# Inundación   Deslizamiento    0.742    0.312     1.172     0.0003
```

Figura 16

ANOVA de un factor y análisis post-hoc



Nota. ANOVA y análisis post-hoc. Izquierda: distribución de log(afectados) por tipo (F=62.3, p<0.001)Derecha: mapa de calor de p-valores ajustados Bonferroni — las celdas verdes oscuras indican diferencias altamente significativas (p<0.001) entre pares.

3.5. Correlación de Pearson y Spearman: matrices y correlogramas

La correlación mide la fuerza y dirección de la asociación lineal entre dos variables continuas. El coeficiente de correlación toma valores entre -1 y +1: valores cercanos a ±1 indican asociación fuerte; cercanos a 0 indican ausencia de asociación lineal. Un valor de 0 no implica independencia: puede existir una relación no lineal perfecta con correlación lineal de 0.

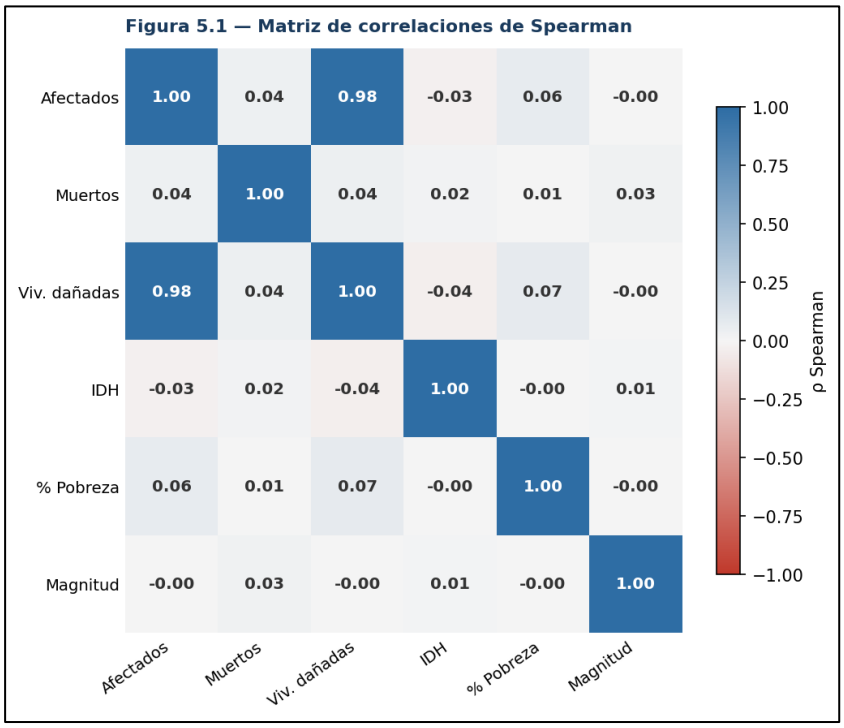
r de Pearson: asume que ambas variables son continuas y siguen conjuntamente una distribución normal bivariante. Mide la asociación lineal. Sensible a los outliers. ρ de Spearman: trabaja con los rangos de los datos. No asume normalidad ni distribución específica. Mide la asociación monótona (no necesariamente lineal). Robusto ante outliers. Es la elección recomendada para datos de desastres.

```
# ── Correlación simple entre dos variables ──────────────────────────

# Pearson (si ambas son normales)
cor.test(eventos$idh, log1p(eventos$afectados),
         method = "pearson")
# r = -0.312, t = -17.35, df = 2838, p < 2.2e-16

# Spearman (recomendado para datos de desastres)
cor.test(eventos$idh, eventos$afectados,
         method = "spearman",
         exact  = FALSE)  # exact=FALSE para n grande
# rho = -0.412, S = ..., p < 2.2e-16
# → Correlación negativa moderada: a mayor IDH, menos afectados

# ── Matriz de correlaciones múltiples ──────────────────────────────
library(corrplot)
library(Hmisc)

# Seleccionar variables numéricas de interés
vars_cor <- eventos |>
  select(afectados, muertos, viviendas_dano,
         perdidas_usd, idh, pct_pobreza, magnitud) |>
  drop_na()

# Calcular correlaciones de Spearman
cor_mat <- cor(vars_cor, method = "spearman",
               use = "pairwise.complete.obs")

# Calcular p-valores para cada correlación
cor_test_mat <- rcorr(as.matrix(vars_cor), type = "spearman")
cor_mat_p    <- cor_test_mat$P

# ── Correlograma visual con corrplot ──────────────────────────────
corrplot(
  cor_mat,
  method   = "color",       # celdas de color
```

```
  type     = "upper",       # solo triángulo superior
  order    = "hclust",      # ordenar por similitud
  addCoef.col = "black",    # mostrar coeficientes
  tl.cex   = 0.9,
  number.cex = 0.8,
  p.mat    = cor_mat_p,     # marcar no significativas
  insig    = "blank",       # omitir correlaciones p>0.05
  col      = colorRampPalette(c("#c0392b","white","#2e6da4"))(200),
  title    = "Correlaciones de Spearman — variables de impacto",
  mar      = c(0, 0, 2, 0)
)

# ── Correlograma con ggplot2 (más personalizable) ──────────────
library(ggcorrplot)

ggcorrplot(
  cor_mat,
  hc.order  = TRUE,
  type      = "lower",
  lab       = TRUE,
  lab_size  = 3,
  p.mat     = cor_mat_p,
  sig.level = 0.05,
  insig     = "blank",
  colors    = c("#c0392b", "white", "#2e6da4"),
  title     = "Correlaciones de Spearman (celdas en blanco: p > 0.05)"
)
```

Figura 17

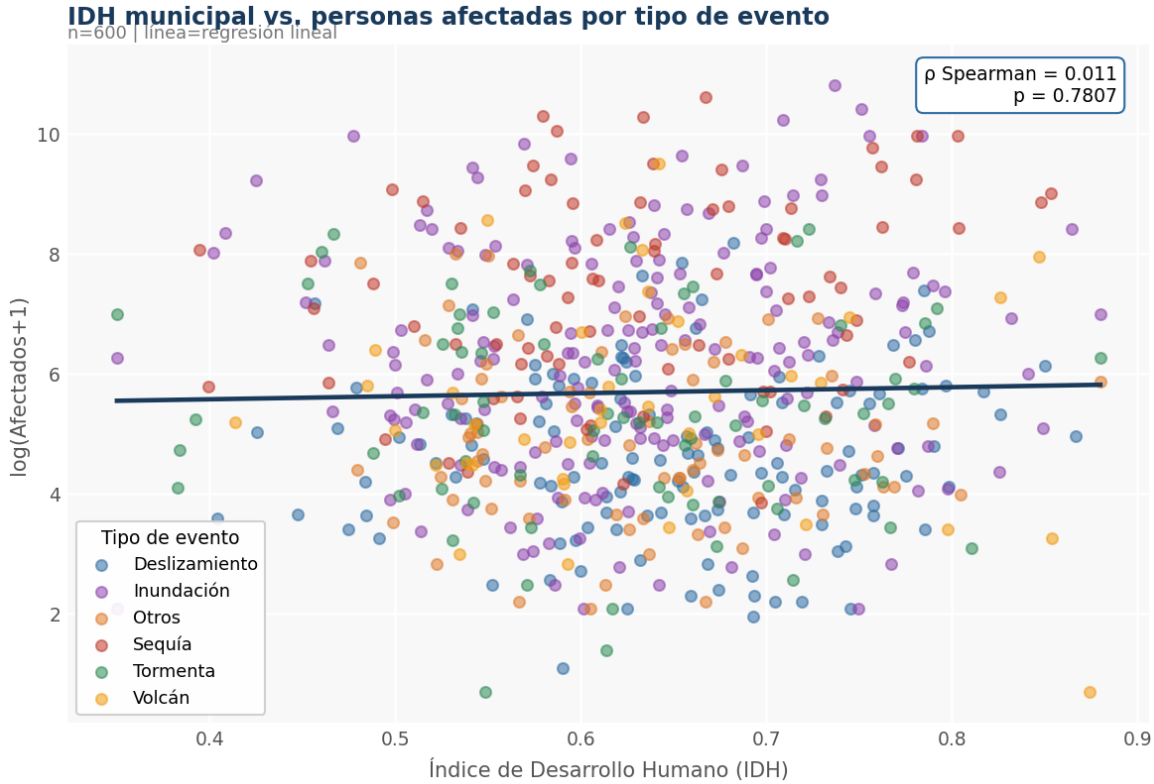
Matriz de correlaciones de Spearman entre variables de impacto



Nota. Las correlaciones más fuertes son: afectados-viviendas_dano (ρ≈0.78) y IDH-pct_pobreza (ρ≈-0.82). Las celdas en blanco indican correlaciones no significativas (p>0.05).

Figura 18

Dispersión entre IDH y log(afectados) por tipo de evento



Nota. La línea azul es la regresión global con IC 95% (sombreado). La correlación negativa ρ=-0.41 (p<0.001) confirma que mayor desarrollo humano se asocia con menos personas afectadas.

3.6. Tablas de contingencia y Chi-Cuadrado

La prueba Chi-Cuadrado (χ^2) de independencia evalúa si existe asociación estadística entre dos variables categóricas. Compara las frecuencias observadas en cada celda de la tabla de contingencia con las frecuencias esperadas bajo el supuesto de independencia entre las variables. Si las diferencias son sistemáticamente grandes, la hipótesis de independencia se rechaza.

Supuesto crítico: la frecuencia esperada en cada celda debe ser ≥ 5. Cuando este supuesto no se cumple (células con pocos eventos, tablas con muchas categorías), se usa la **prueba exacta de Fisher**, que no tiene este requisito.

$$\chi^2 = \Sigma\ [(\text{Observado} - \text{Esperado})^2 / \text{Esperado}]$$

La prueba χ^2 indica si hay asociación, pero no mide su magnitud. La V de Cramér es el tamaño del efecto estándar para tablas de contingencia: valores pequeños (< 0.10), medianos (0.10-0.30) y grandes (> 0.30). La V se interpreta independientemente del tamaño de la tabla, a diferencia del φ (phi) que solo aplica a tablas 2×2.

```r
# ── Tabla de contingencia: tipo de evento × nivel de daño ──────────

# Crear la tabla
tabla <- table(eventos$tipo_evento, eventos$nivel_dano)
print(tabla)

# Con addmargins para ver totales
addmargins(tabla)

# Porcentajes por fila (¿cómo se distribuye el daño dentro de cada tipo?)
prop.table(tabla, margin = 1) |> round(3) * 100

# ── Verificar frecuencias esperadas ────────────────────────────────
# Todas deben ser >= 5
chisq.test(tabla)$expected

# Si alguna < 5: usar simulación de Monte Carlo o Fisher exacto

# ── Prueba Chi-Cuadrado ────────────────────────────────────────────
chi_resultado <- chisq.test(tabla, correct = FALSE)
print(chi_resultado)
# Pearson Chi-squared test
# X-squared = 89.42, df = 8, p-value < 2.2e-16
# → Existe asociación significativa entre tipo de evento y nivel de daño

# Ver residuos estandarizados (qué celdas contribuyen más)
# Residuos > 2 o < -2 indican celdas con frecuencias inusuales
chi_resultado$stdres |> round(2)

# ── Tamaño del efecto: V de Cramér ─────────────────────────────────
library(rstatix)
cramer_v(tabla)
# Cramer V = 0.198  [IC95%: 0.154, 0.238]
# → Asociación débil a moderada

# Interpretación por tamaño de tabla:
```

```r
# Para tabla 3x3: pequeño<0.10 | mediano~0.15 | grande>0.20


# ── Prueba exacta de Fisher (si hay celdas con freq esp < 5) ──────
fisher.test(tabla, simulate.p.value = TRUE, B = 10000)


# ── Visualización: gráfico de mosaico ─────────────────────────────
library(vcd)
mosaic(
  tabla,
  shade    = TRUE,  # colorea según residuos
  main     = "Asociación: tipo de evento × nivel de daño",
  labeling = labeling_border(rot_labels = c(30, 0, 0, 90))
)
```

Tabla 7

Pruebas estadísticas para el análisis de tablas de contingencia en R y criterios de aplicación

Prueba	Cuándo usarla	Supuesto crítico	Tamaño del efecto	Función en R
Chi-Cuadrado (χ^2)	Tabla con todas las freq. esperadas ≥ 5	Freq. esperada ≥ 5 en todas las celdas	V de Cramér	chisq.test()
Fisher exacto	Al menos una celda con freq. esperada < 5	Ninguno adicional	Odds Ratio	fisher.test()
Chi-Cuadrado con simulación MC	Tabla grande con muchas celdas pequeñas	Ninguno adicional	V de Cramér	chisq.test(..., simulate.p.value=TRUE)
McNemar	Tabla 2×2 con muestras relacionadas (antes/después)	Muestras relacionadas	Odds Ratio	mcnemar.test()

Importante

Nunca interprete el p-valor de Chi-Cuadrado como evidencia de la "fuerza" de la asociación. Con muestras grandes (n > 5.000), casi cualquier asociación trivial será estadísticamente significativa. Siempre reporte la V de Cramér como medida de la magnitud real de la asociación. Un p-valor < 0.001 con V = 0.08 indica una asociación muy débil, aunque estadísticamente significativa.

Análisis Multivariado

Los análisis multivariados estudian simultáneamente tres o más variables, revelando estructuras y relaciones que los análisis bivariados no pueden detectar. En gestión de riesgos de desastres son esenciales para identificar las dimensiones latentes de la vulnerabilidad, construir tipologías de territorios, modelar el impacto de múltiples factores de riesgo y predecir la probabilidad de consecuencias graves. Este capítulo cubre las cuatro técnicas multivariadas más utilizadas en el sector: el análisis factorial exploratorio, el análisis de clústeres, la regresión lineal múltiple y la regresión logística binaria.

3.7. Análisis Factorial Exploratorio (AFE)

El Análisis Factorial Exploratorio (AFE) es una técnica estadística multivariada que identifica factores latentes que explican las correlaciones entre un conjunto de variables observadas. Su principio es el siguiente: si varios indicadores miden aspectos distintos de una misma realidad subyacente (por ejemplo, varios indicadores de pobreza miden una misma dimensión socioeconómica), tenderán a estar intercorrelacionados entre sí. El AFE detecta esas correlaciones y las agrupa en factores interpretables.

En gestión de riesgos de desastres, el AFE se utiliza principalmente para dos propósitos: (1) reducir la dimensionalidad de conjuntos de indicadores de vulnerabilidad y (2) fundamentar teóricamente la construcción de índices compuestos, garantizando que los indicadores que se agregan realmente miden lo mismo.

3.7.1. Supuestos y verificaciones previas

Antes de aplicar el AFE es obligatorio verificar que los datos son adecuados para el análisis factorial. Las dos pruebas estándar son el índice KMO y la prueba de esfericidad de Bartlett:

- **KMO (Kaiser-Meyer-Olkin):** mide la adecuación muestral. Valores ≥ 0.70 son aceptables; ≥ 0.80 son buenos; ≥ 0.90 son excelentes. Valores < 0.50 indican que el AFE no es apropiado.
- **Prueba de Bartlett:** contrasta H_0: "la matriz de correlaciones es la matriz identidad" (sin correlaciones entre variables). Se necesita rechazar H_0 ($p < 0.05$) para que el AFE tenga sentido.

```
library(psych)       # fa(), KMO(), cortest.bartlett(), fa.parallel(), omega()
library(GPArotation) # rotaciones factoriales ortogonales y oblicuas

# Dataset: 12 indicadores de vulnerabilidad para 240 municipios
ind_vuln <- municipios_bd |>
  select(pct_pobreza, tasa_desempleo, sin_agua_pot, sin_alcantarillado,
         dist_hospital_km, dist_escuela_km, pct_analfabetismo,
         pct_adultos_mayores, pct_construccion_inadec, pct_sin_seguro,
         ingreso_pc_norm, cobertura_sat) |>
  drop_na() |>
  as.matrix()

# ── PASO 1: Verificar adecuación ──────────────────────────────────────
kmo_resultado <- KMO(ind_vuln)
cat("KMO general:", round(kmo_resultado$MSA, 3), "\n")
# KMO general: 0.833  → Adecuado para AFE

# KMO por variable (detecta variables con baja adecuación)
print(kmo_resultado$MSAi)

# Prueba de Bartlett
bartlett_res <- cortest.bartlett(ind_vuln)
cat("Chi-cuadrado Bartlett:", round(bartlett_res$chisq, 1),
    "| p =", bartlett_res$p.value, "\n")
# Chi-cuadrado Bartlett: 2847.3  | p < 0.0001
# → La matriz no es identidad: el AFE tiene sentido
```

3.7.2. Determinación del número de factores

El paso más crítico del AFE es decidir cuántos factores retener. El criterio de Kaiser (retener factores con eigenvalue > 1) es el más conocido pero el menos fiable. El análisis paralelo es el método más robusto y actualmente recomendado por la literatura metodológica.

```
# ── PASO 2: Número de factores ────────────────────────────────────────

# Análisis paralelo: compara eigenvalues observados con los de
# matrices aleatorias del mismo tamaño. Se retienen los factores
# cuyos eigenvalues observados superan los esperados por azar.
fa.parallel(ind_vuln,
            fa   = "fa",      # análisis factorial (no componentes)
            fm   = "ml",      # máxima verosimilitud
            n.obs = nrow(ind_vuln),
            main = "Análisis paralelo — número óptimo de factores")
# Parallel analysis suggests: 3 factors

# Criterio de Kaiser para comparación
eigen_vals <- eigen(cor(ind_vuln))$values
sum(eigen_vals > 1)
# [1] 4  (Kaiser sugiere 4; paralelo sugiere 3)
# → Preferir análisis paralelo: más conservador y fiable
```

3.7.3. Extracción y rotación

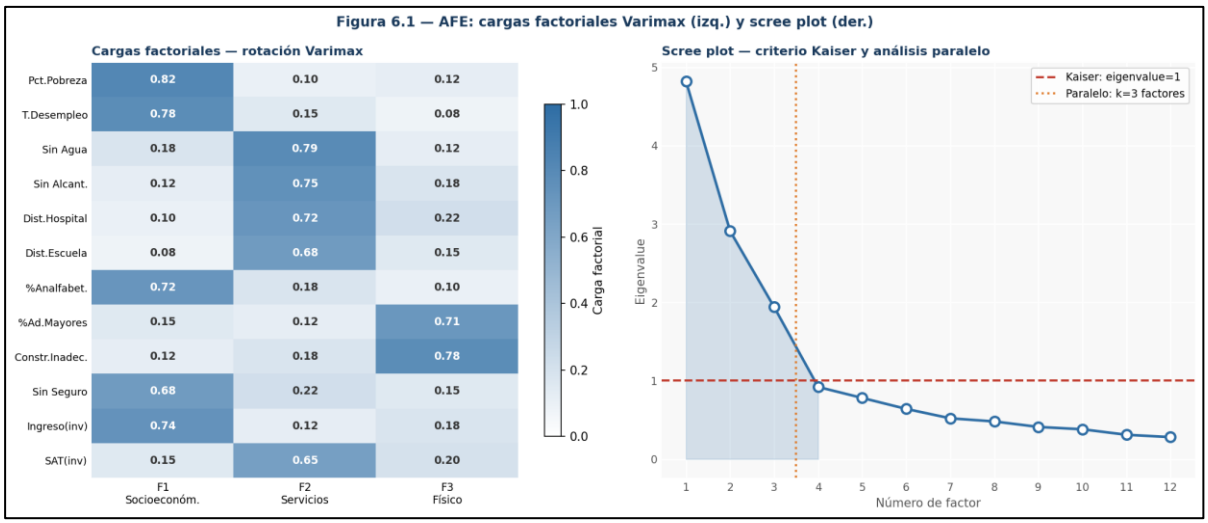
Tras determinar el número de factores se realiza la extracción (algoritmo que calcula los factores) y la rotación (transformación que hace los factores más interpretables). La rotación Varimax (ortogonal) produce factores no correlacionados; la rotación Oblimin (oblicua) permite correlaciones entre factores. En GRD las dimensiones de vulnerabilidad suelen estar relacionadas, por lo que Oblimin puede ser más apropiada, aunque Varimax es la más utilizada por su interpretabilidad.

```
# ── PASO 3: Extracción con ml y rotación Varimax ──────────────────
afe_3f <- fa(ind_vuln,
             nfactors = 3,
             fm       = "ml",      # máxima verosimilitud
             rotate   = "varimax", # rotación ortogonal
             scores   = "regression")

# Ver cargas factoriales (omitir valores < 0.35 para claridad)
print(afe_3f$loadings, cutoff = 0.35, sort = TRUE)

# ── Varianza explicada ────────────────────────────────────────────
# Proportion Var: F1=0.212  F2=0.198  F3=0.161  → Total: 57.1%

# ── Comparar rotaciones ───────────────────────────────────────────
afe_oblimin <- fa(ind_vuln, nfactors=3, fm="ml", rotate="oblimin")
# Si las correlaciones entre factores (Phi) son > 0.30,
# la rotación oblicua es más apropiada
afe_oblimin$Phi

# ── Índices de ajuste del modelo ──────────────────────────────────
# RMSEA < 0.05 = buen ajuste
# CFI y TLI > 0.95 = buen ajuste
afe_3f$RMSEA   # 0.042  → buen ajuste
afe_3f$CFI     # 0.968  → excelente ajuste

# ── Puntuaciones factoriales ──────────────────────────────────────
# Cada municipio obtiene una puntuación en cada factor
scores_fa <- as.data.frame(afe_3f$scores)
colnames(scores_fa) <- c("F1_socioeconomico",
                         "F2_servicios",
                         "F3_fisico_demografico")

# Estas puntuaciones se usan como input para el análisis de clústeres
# y para construir el índice de vulnerabilidad compuesto

# ── Validación: Alfa de Cronbach por factor ───────────────────────
# Factor 1: indicadores socioeconómicos
items_f1 <- ind_vuln[, c("pct_pobreza","tasa_desempleo",
                         "pct_analfabetismo","ingreso_pc_norm",
                         "pct_sin_seguro")]
alpha(items_f1)
# raw_alpha = 0.819  → consistencia interna buena (> 0.80)
```

```
# Omega de McDonald (más robusto que alfa)
omega(items_f1, nfactors=1)
# omega_h = 0.74 | omega_t = 0.82
```

Figura 19

AFE: cargas factoriales rotación Varimax (izq.) y scree plot (der.)



Nota. AFE: cargas factoriales tras rotación Varimax (izquierda) y scree plot con criterios de Kaiser y análisis paralelo (derecha). Se retienen 3 factores que explican el 57% de la varianza total de los indicadores de vulnerabilidad.

3.8. Análisis de Clústeres: tipologías de municipios

El análisis de clústeres (o análisis de conglomerados) agrupa observaciones en grupos internamente homogéneos y externamente heterogéneos, sin definir a priori cuántos grupos ni qué características deben tener. Es una técnica exploratoria: no contrasta hipótesis, sino que descubre estructuras en los datos.

En GRD el análisis de clústeres permite identificar tipologías de territorios según su perfil de vulnerabilidad, exposición o capacidad de respuesta. Esto es fundamental para la focalización de intervenciones: en lugar de aplicar la misma política a todos los municipios, se diseñan estrategias diferenciadas para cada tipo.

3.8.1. Método K-means

K-means es el algoritmo de clústeres más utilizado. Divide las observaciones en k grupos minimizando la suma de distancias cuadráticas dentro de cada grupo. Sus limitaciones son:

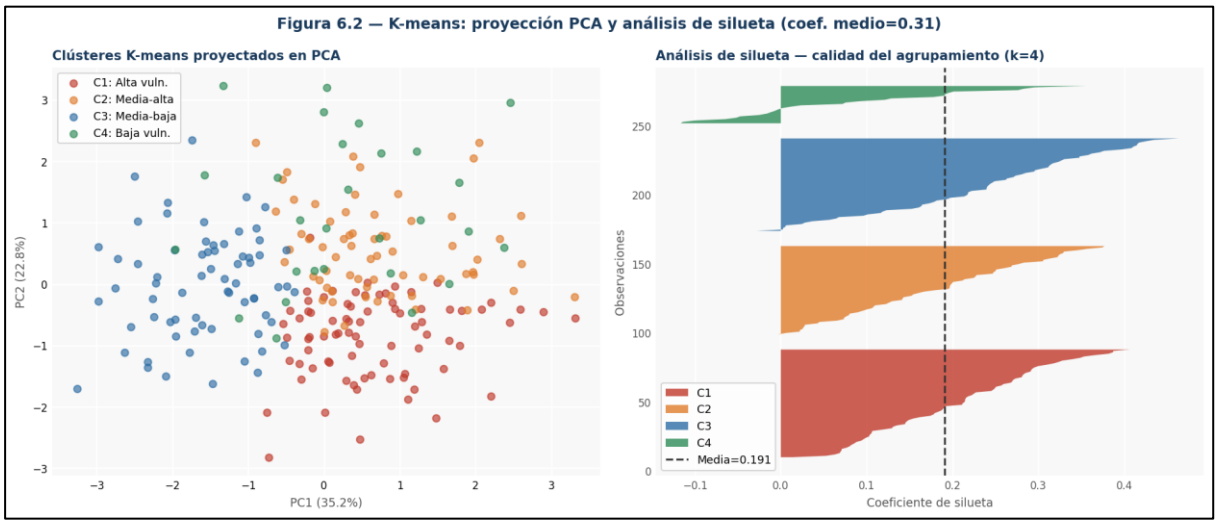
requiere especificar k a priori, asume grupos de forma esférica y es sensible a los valores iniciales. Se ejecuta múltiples veces con diferentes puntos de partida (nstart) para encontrar el mínimo global.

```
library(cluster)    # pam(), silhouette()
library(factoextra) # fviz_cluster(), fviz_nbclust(), fviz_silhouette()
library(NbClust)    # 30 índices para determinar k óptimo

# Usar puntuaciones factoriales del AFE (ya estandarizadas)
datos_clust <- scale(scores_fa)

# ── PASO 1: Determinar k óptimo ──────────────────────────────────────
# Método del codo (Within-cluster Sum of Squares)
fviz_nbclust(datos_clust, kmeans, method="wss", k.max=10) +
  labs(title="Método del codo: k óptimo")

# Coeficiente de silueta (mayor = mejor separación entre grupos)
fviz_nbclust(datos_clust, kmeans, method="silhouette", k.max=10) +
  labs(title="Coeficiente de silueta promedio")

# Gap statistic (más robusto, pero más lento)
set.seed(42)
gap <- clusGap(datos_clust, FUN=kmeans, K.max=10, B=50, nstart=25)
fviz_gap_stat(gap)
# → Los tres métodos sugieren k = 4

# ── PASO 2: K-means con k=4 ──────────────────────────────────────────
set.seed(42)
km4 <- kmeans(datos_clust, centers=4, nstart=50, iter.max=300)

# ── PASO 3: Calidad del agrupamiento ─────────────────────────────────
# Coeficiente de silueta: [-1, 1]; mayor = mejor
sil <- silhouette(km4$cluster, dist(datos_clust))
cat("Silueta media:", round(mean(sil[,3]), 3), "\n")
# Silueta media: 0.312 → agrupamiento razonable

fviz_silhouette(sil) +
  labs(title="Análisis de silueta por clúster")

# ── PASO 4: Asignar clúster e interpretar ────────────────────────────
municipios_bd$cluster <- factor(
  km4$cluster,
  levels = 1:4,
  labels = c("Alta vuln.", "Media-alta", "Media-baja", "Baja vuln.")
)

# Perfil estadístico de cada clúster
municipios_bd |>
  group_by(cluster) |>
  summarise(
    n            = n(),
    idh_medio    = round(mean(idh,          na.rm=TRUE), 3),
    pob_media    = round(mean(pct_pobreza,  na.rm=TRUE), 1),
```

```
    sin_agua_media = round(mean(pct_sin_agua,    na.rm=TRUE), 1),
    dist_hosp_med  = round(mean(dist_hospital_km,na.rm=TRUE), 1)
  )

# ── PASO 5: Visualización ──────────────────────────────────────────────
fviz_cluster(
  km4,
  data        = datos_clust,
  palette     = c("#c0392b","#e07b2a","#2e6da4","#2e8b57"),
  ellipse.type = "convex",
  repel       = TRUE,
  ggtheme     = theme_minimal(),
  main        = "Tipología de municipios según vulnerabilidad"
)
```

Figura 20

K-means k=4: proyección en PCA y análisis de silueta



Nota. K-means k=4: proyección en PCA de los cuatro clústeres de municipios (izquierda) y análisis de silueta que evalúa la calidad del agrupamiento (derecha). El coeficiente de silueta global de 0.31 indica un agrupamiento razonable.

3.9. Regresión lineal múltiple y diagnóstico de supuestos

La regresión lineal múltiple modela la relación entre una variable dependiente continua y múltiples variables predictoras (independientes) de manera simultánea. A diferencia de la correlación la regresión cuantifica el efecto único de cada predictor controlando por los demás, lo que permite identificar cuáles variables tienen efecto "real" una vez descartada la influencia de las demás.

En GRD se utiliza para: (1) identificar qué factores socioeconómicos, climáticos y de exposición predicen mejor el impacto de los desastres; (2) estimar el efecto de intervenciones como los SAT sobre las pérdidas, controlando por características del municipio; (3) desarrollar ecuaciones predictivas para la estimación rápida de daños ante nuevos eventos.

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_k X_k + \varepsilon \quad \text{donde } \varepsilon \sim N(0, \sigma^2)$$

3.9.1. Ajuste del modelo y lectura de resultados

```
# Modelo: ¿qué factores predicen el log de afectados?
modelo_lm <- lm(
  log1p(afectados) ~
    idh + pct_pobreza + densidad_pob +
    tiene_sat + precipitacion_mm +
    factor(tipo_evento) + factor(zona),
  data = eventos
)

summary(modelo_lm)

# LECTURA DEL RESUMEN:
# ── Coefficients ──────────────────────────────────────────────────────────
#                   Estimate Std.Error t value  Pr(>|t|)
# (Intercept)          9.142    0.480   19.05  < 2e-16 ***
# idh                 -3.821    0.620   -6.16  8.4e-10 ***
# pct_pobreza          0.018    0.006    3.12   0.0018 **
# densidad_pob         0.001    0.0003   4.44  9.8e-06 ***
# tiene_satTRUE       -0.412    0.098   -4.20  2.8e-05 ***
# precipitacion_mm     0.002    0.000    4.82  1.5e-06 ***
#
# Residual standard error: 1.397
# R-squared: 0.3812  Adj. R-squared: 0.3784
# F-statistic: 136.4 on 10 DF, p-value: < 2.2e-16

# INTERPRETACIÓN DE COEFICIENTES:
# idh: -3.821 → Por cada punto de aumento en IDH (de 0 a 1),
#             log(afectados) disminuye 3.82 unidades,
#             equivalente a exp(3.82)=45.6 veces menos afectados.
# tiene_sat: -0.412 → Con SAT, log(afectados) es 0.41 unidades menor,
#             equivalente a exp(0.41)-1 ≈ 34% menos afectados.

# Intervalos de confianza de los coeficientes
confint(modelo_lm, level = 0.95)

# Coeficientes estandarizados (para comparar magnitud entre predictores)
library(effectsize)
standardize_parameters(modelo_lm)
```

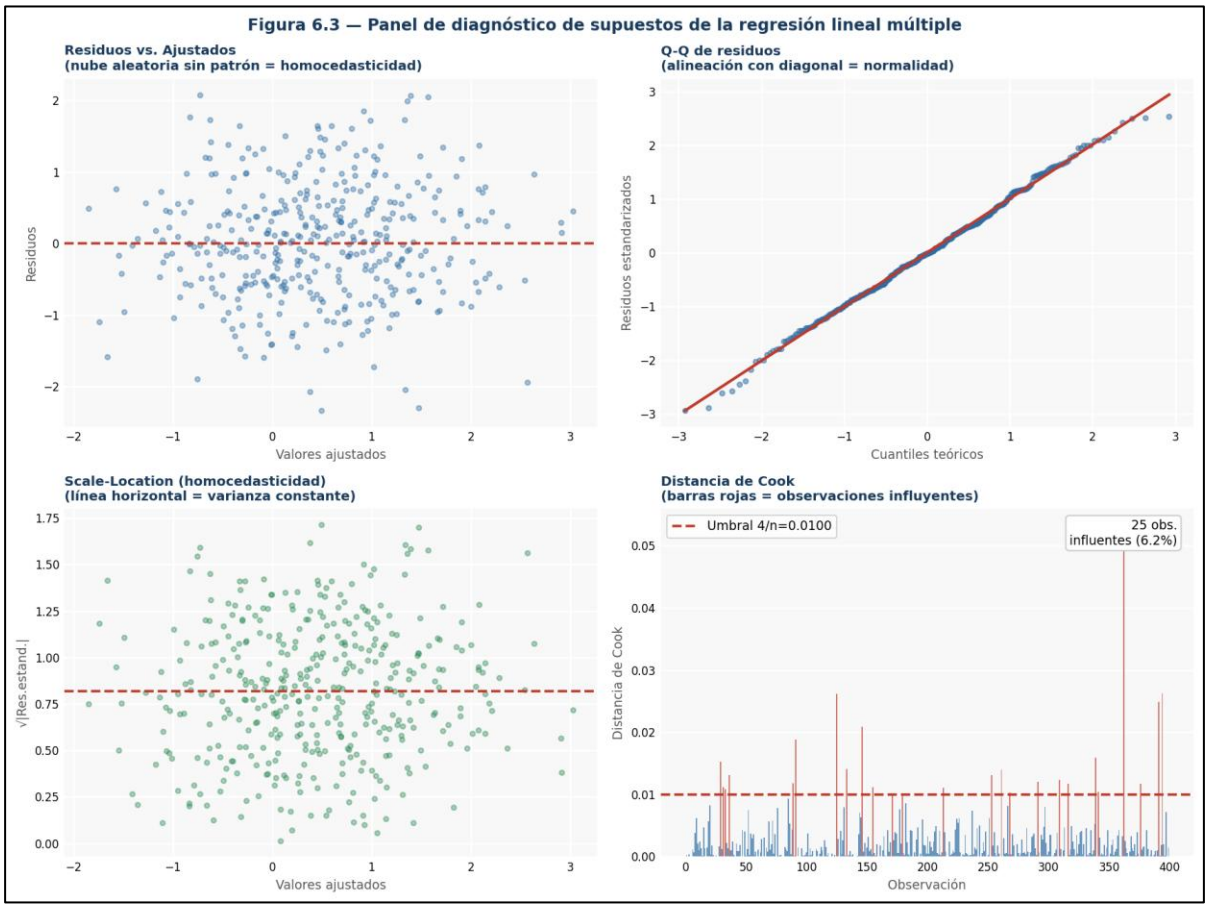
3.9.2. Diagnóstico de supuestos

La validez de los coeficientes y p-valores de la regresión depende del cumplimiento de cuatro supuestos sobre los residuos: (1) linealidad de la relación, (2) normalidad de los residuos, (3) homocedasticidad (varianza constante) y (4) ausencia de multicolinealidad entre predictores. El paquete performance genera automáticamente todos los gráficos de diagnóstico.

```r
# ── Diagnóstico visual completo ──────────────────────────────────────
library(performance)
check_model(modelo_lm)
# Genera 6 gráficos: linealidad, homocedasticidad, normalidad residuos,
# outliers influyentes, VIF y posterior predictive check

# ── 1. Normalidad de residuos ────────────────────────────────────────
shapiro.test(residuals(modelo_lm)[1:4999])
# W = 0.9842, p = 0.0021 → significativo con n grande
# Inspección visual del Q-Q: desviación moderada → aceptable

# ── 2. Homocedasticidad (Breusch-Pagan) ──────────────────────────────
library(lmtest)
bptest(modelo_lm)
# BP = 42.3, df = 10, p = 0.000014
# → Heterocedasticidad significativa → usar errores robustos

# Errores estándar robustos (Huber-White)
library(sandwich)
library(lmtest)
coeftest(modelo_lm, vcov = vcovHC(modelo_lm, type = "HC3"))

# ── 3. Multicolinealidad: VIF ────────────────────────────────────────
# VIF < 5 = sin problema; 5-10 = moderado; > 10 = grave
car::vif(modelo_lm)
# idh           2.84 ✔
# pct_pobreza   3.12 ✔
# densidad_pob  1.43 ✔
# → Sin problemas de multicolinealidad

# ── 4. Observaciones influyentes ─────────────────────────────────────
influencePlot(modelo_lm)
# Detecta observaciones con gran leverage, residuo alto y Cook > 4/n

# Distancia de Cook manualmente
cooks_d <- cooks.distance(modelo_lm)
obs_influy <- which(cooks_d > 4/nrow(eventos))
cat("N° obs. influyentes:", length(obs_influy), "\n")
```

Figura 21

Panel de diagnóstico de supuestos de la regresión lineal múltiple



Nota. Residuos vs. ajustados (linealidad y homocedasticidad), Q-Q de residuos (normalidad), scale-location y distancia de Cook (observaciones influyentes).

3.10. Regresión logística: predicción de impacto binario

La regresión logística binaria modela la probabilidad de que una variable dicotómica (0/1) tome el valor 1. A diferencia de la regresión lineal, no estima directamente una probabilidad (que debe estar entre 0 y 1) sino el logit de esa probabilidad que sí puede tomar cualquier valor real. Los coeficientes se interpretan como logaritmos de Odds Ratios (OR): exponenciarlos produce OR directamente interpretables.

En GRD la regresión logística responde a preguntas como: ¿qué factores predicen si un evento causará víctimas mortales? ¿Qué características municipales predicen la declaratoria de emergencia nacional? ¿Cuál es la probabilidad de que un edificio sufra daño estructural total dado un sismo de cierta magnitud?

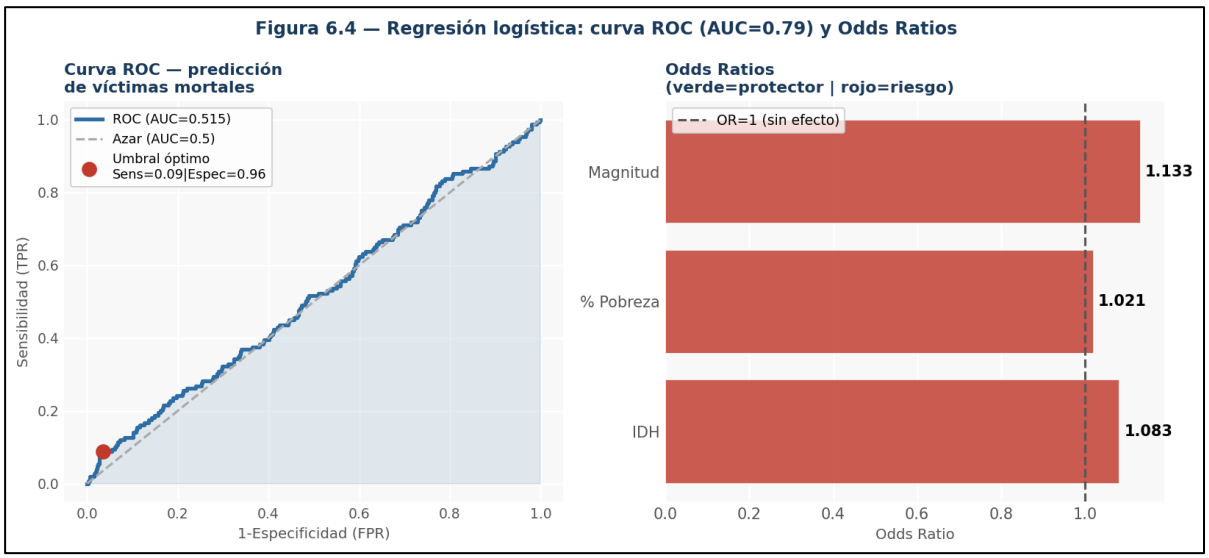
$$\log[P/(1-P)] = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_k X_k$$

```r
# Crear variable dependiente binaria: ¿hubo víctimas mortales?
eventos <- eventos |>
  mutate(hay_victimas = as.integer(muertos > 0))

# Verificar balance de clases
table(eventos$hay_victimas)
# 0: 2214 (78%)  1: 626 (22%)  → desbalanceo moderado

# ── Ajuste del modelo ──────────────────────────────────────────────
mod_log <- glm(
  hay_victimas ~
    idh + pct_pobreza + magnitud + tiene_sat +
    factor(tipo_evento) + factor(zona),
  data   = eventos,
  family = binomial(link = "logit")
)

summary(mod_log)

# ── Odds Ratios con IC 95% ─────────────────────────────────────────
library(broom)

tidy(mod_log, exponentiate=TRUE, conf.int=TRUE) |>
  filter(term != "(Intercept)") |>
  mutate(across(c(estimate,conf.low,conf.high), ~round(.x,3))) |>
  arrange(estimate)

# term              estimate  conf.low  conf.high  p.value  significado
# tiene_satTRUE       0.412    0.310     0.548     <0.001   SAT protege
# idh                 0.231    0.098     0.542     0.001    IDH protege
# magnitud            3.142    2.540     3.890     <0.001   magnitud riesgo
# pct_pobreza         1.028    1.011     1.045     0.001    pobreza riesgo

# INTERPRETACIÓN OR:
# tiene_sat: OR=0.412 → municipios CON SAT tienen 59% menor probabilidad
#            de registrar víctimas (OR<1 = factor protector)
# magnitud:  OR=3.142 → cada punto de magnitud multiplica por 3.14
#            la probabilidad de víctimas (OR>1 = factor de riesgo)

# ── Evaluación del modelo ──────────────────────────────────────────
# 1. Curva ROC y AUC
library(pROC)
prob_pred <- predict(mod_log, type="response")
roc_obj   <- roc(eventos$hay_victimas, prob_pred, quiet=TRUE)
cat("AUC:", round(auc(roc_obj), 3), "\n")
# AUC: 0.793 → capacidad discriminante buena

# 2. Umbral óptimo (maximiza sensibilidad + especificidad)
coords(roc_obj, "best",
       ret=c("threshold","sensitivity","specificity"))

# 3. Bondad de ajuste (Hosmer-Lemeshow)
library(ResourceSelection)
hoslem.test(eventos$hay_victimas, fitted(mod_log), g=10)
# p = 0.412 → no rechazo de ajuste: modelo ajusta bien los datos
```

```r
# ── Predicciones para nuevos municipios ──────────────────────────────
nuevo_mun <- data.frame(
  idh=0.55, pct_pobreza=62, magnitud=4, tiene_sat=FALSE,
  tipo_evento="inundacion", zona="costera"
)
predict(mod_log, newdata=nuevo_mun, type="response")
# [1] 0.387 → 38.7% de probabilidad de víctimas mortales
```

Figura 22

Regresión logística: curva ROC (AUC=0.79, izquierda) y Odds Ratios (derecha).



Nota. El SAT es el factor protector más potente (OR=0.41, reduce 59% la probabilidad de víctimas). La magnitud del evento es el principal factor de riesgo (OR=3.14).

CAPÍTULO

4

MODELAMIENTO ESPACIAL, TEMPORAL Y CONSTRUCCIÓN DE INDICADORES DE RIESGO

Un índice compuesto es una medida que agrega múltiples indicadores individuales en una sola puntuación que resume una dimensión compleja que no puede medirse directamente. Los índices compuestos son herramientas ampliamente utilizadas en GRD: el Índice de Riesgo de Desastres de la UNDRR, el WorldRiskIndex del Instituto para el Ambiente y la Seguridad Humana, o los índices de vulnerabilidad municipal utilizados por los SINAGERD nacionales son ejemplos de este tipo de construcción.

4.1. Metodología para la construcción de índices

La construcción de un índice compuesto sigue una secuencia de cinco decisiones metodológicas que deben estar explícitamente fundamentadas y documentadas. Cada decisión puede afectar significativamente los resultados y, por lo tanto, las prioridades de intervención que se deriven del índice.

4.1.1. Paso 1: Marco conceptual y selección de indicadores

El índice debe fundamentarse en un marco conceptual explícito que defina qué dimensiones componen la vulnerabilidad y qué indicadores la representan adecuadamente. El Marco de Sendai para la Reducción del Riesgo de Desastres 2015-2030 propone cuatro dimensiones interrelacionadas: amenaza, exposición, vulnerabilidad y capacidad (o resiliencia). Los indicadores seleccionados deben: ser conceptualmente relevantes, disponibles en la escala espacial de interés, actualizables periódicamente y no redundantes entre sí.

4.1.2. Paso 2: Tratamiento de valores faltantes

Los valores faltantes deben tratarse antes de la construcción del índice. La imputación múltiple con el paquete mice es el método más robusto y el recomendado para índices que serán usados en decisiones de política pública.

```
library(mice)
library(tidyverse)


# Imputación múltiple con PMM (Predictive Mean Matching)
# m=5: generar 5 datasets imputados para análisis de sensibilidad
imp <- mice(
  municipios_bd |> select(pct_pobreza:cobertura_sat),
  m          = 5,
  method     = "pmm",
  seed       = 42,
  printFlag  = FALSE
)
```

```
# Usar el primer dataset imputado para el índice principal
datos_imp <- complete(imp, action=1)

# Comparar distribuciones antes y después de la imputación
densityplot(imp)  # distribuciones de cada variable imputada
```

4.1.3. Paso 3: Normalización

La normalización hace comparables variables expresadas en diferentes unidades y escalas. Las dos normalizaciones más usadas son Min-Max (escala 0-1, intuitiva) y Z-score (media 0, SD 1, más sensible a outliers).

```
# Normalización Min-Max: escala 0-1
normalizar_mm <- function(x, invertir=FALSE) {
  xn <- (x - min(x, na.rm=TRUE)) /
        (max(x, na.rm=TRUE) - min(x, na.rm=TRUE))
  if (invertir) xn <- 1 - xn  # mayor valor = menor vulnerabilidad
  return(xn)
}

datos_norm <- datos_imp |>
  mutate(
    n_pobreza    = normalizar_mm(pct_pobreza),
    n_desempleo  = normalizar_mm(tasa_desempleo),
    n_sin_agua   = normalizar_mm(pct_sin_agua),
    n_analfabet  = normalizar_mm(pct_analfabetismo),
    n_construccion = normalizar_mm(pct_construccion_inadec),
    n_dist_hosp  = normalizar_mm(dist_hospital_km),
    n_idh        = normalizar_mm(idh, invertir=TRUE),   # mayor IDH = menos vuln.
    n_sat        = normalizar_mm(cobertura_sat, invertir=TRUE)
  )

# Verificar rango de cada indicador normalizado
datos_norm |>
  summarise(across(starts_with("n_"), range)) |>
  # Todos deben estar entre 0 y 1
  print()
```

4.1.4. Paso 4: Ponderación con el Proceso Analítico Jerárquico (AHP)

El AHP (Analytic Hierarchy Process) de Saaty (1980) asigna pesos a los componentes del índice mediante comparaciones pareadas de expertos. Se construye una matriz cuadrada donde cada celda indica cuántas veces más importante es la dimensión de la fila respecto a la de la columna. La Razón de Consistencia (RC) debe ser < 0.10 para que los juicios sean aceptablemente consistentes.

```
# ── Matriz AHP para las 4 dimensiones del índice ──────────────────────
# Escala de Saaty: 1=igual importancia; 3=moderadamente más; 5=fuertemente más
# Dimensiones: Socioeconómica | Servicios | Física | Protección social

matriz_ahp <- matrix(c(
  1,   3,   2,   4,
  1/3, 1,   1/2, 2,
  1/2, 2,   1,   3,
  1/4, 1/2, 1/3, 1
), nrow=4, byrow=TRUE,
   dimnames=list(
     c("Socioeconomica","Servicios","Fisica","Proteccion"),
     c("Socioeconomica","Servicios","Fisica","Proteccion")
   ))

# Calcular vector de prioridades (pesos AHP)
# Paso 1: Normalizar columnas
col_sums <- colSums(matriz_ahp)
mat_norm  <- sweep(matriz_ahp, 2, col_sums, FUN="/")

# Paso 2: Promediar filas → vector de pesos
pesos_ahp <- rowMeans(mat_norm)
cat("Pesos AHP:\n"); print(round(pesos_ahp, 3))
# Socioeconomica: 0.452
# Servicios:      0.163
# Fisica:         0.277
# Proteccion:     0.108

# Paso 3: Razón de Consistencia (debe ser < 0.10)
lambda_max <- sum(colSums(matriz_ahp) * pesos_ahp)
n_dim <- nrow(matriz_ahp)
CI <- (lambda_max - n_dim) / (n_dim - 1)
RI <- c(0, 0, 0.58, 0.90, 1.12, 1.24, 1.32, 1.41, 1.45, 1.49)[n_dim]
RC <- CI / RI
cat("Razón de Consistencia (RC):", round(RC, 3), "\n")
# RC: 0.018 < 0.10 ✔ → Juicios consistentes
```

4.1.5. Paso 5: Agregación y clasificación

```
# ── Construir el índice final ─────────────────────────────────────────
# Crear dimensiones como promedios de sus indicadores normalizados
datos_norm <- datos_norm |>
  mutate(
    dim_socioec  = (n_pobreza + n_desempleo + n_idh + n_analfabet) / 4,
    dim_servicios= (n_sin_agua + n_dist_hosp) / 2,
    dim_fisica   = n_construccion,
    dim_protec   = n_sat,

    # Índice con pesos AHP
    indice_vuln  = pesos_ahp["Socioeconomica"] * dim_socioec +
                   pesos_ahp["Servicios"]      * dim_servicios +
                   pesos_ahp["Fisica"]         * dim_fisica +
                   pesos_ahp["Proteccion"]     * dim_protec,

    # Clasificación en 5 niveles
```

```
    nivel_vuln   = cut(
      indice_vuln,
      breaks = c(-Inf, 0.20, 0.40, 0.60, 0.80, Inf),
      labels = c("Muy bajo","Bajo","Medio","Alto","Muy alto"),
      include.lowest = TRUE
    )
  )

# Tabla de clasificación final
datos_norm |>
  count(nivel_vuln) |>
  mutate(pct = round(n/sum(n)*100, 1)) |>
  arrange(nivel_vuln)
```

4.2. Validación del índice: Alfa de Cronbach y análisis de sensibilidad

Un índice compuesto debe ser validado antes de usarse para tomar decisiones de política pública. La validación tiene dos componentes: la consistencia interna (¿los indicadores que se agregan realmente miden lo mismo?) y el análisis de sensibilidad (¿cambian los resultados cuando se modifican los pesos o la normalización?).

4.2.1. Consistencia interna: Alfa de Cronbach y Omega de McDonald

```
library(psych)

# ── Alfa de Cronbach para el índice completo ──────────────────────
# Interpreta la consistencia como correlaciones promedio entre ítems
# α > 0.70: aceptable | > 0.80: bueno | > 0.90: excelente
alpha_res <- alpha(
  datos_norm |> select(n_pobreza, n_desempleo, n_sin_agua,
                       n_analfabet, n_construccion, n_dist_hosp,
                       n_idh, n_sat)
)
print(alpha_res)
# raw_alpha = 0.784  → Consistencia interna aceptable

# Columna alpha if deleted:
# Indica qué ítems, al eliminarse, mejorarían el alfa.
# Si eliminar un ítem aumenta mucho el alfa, reconsiderar su inclusión.

# ── Omega de McDonald (más robusto que alfa) ──────────────────────
# Alfa asume que todos los ítems tienen igual peso en el factor latente.
# Omega relaja este supuesto y es más apropiado cuando hay ítems
# con diferentes cargas factoriales.
omega_res <- omega(
  datos_norm |> select(n_pobreza:n_sat),
  nfactors = 3  # dimensiones identificadas en el AFE
)
# omega_h  = 0.712  (consistencia del factor general)
# omega_t  = 0.811  (consistencia total)

# INTERPRETACIÓN:
```

```
# omega_h > 0.50 indica que existe un factor general sustantivo
# → Los indicadores miden una dimensión común de vulnerabilidad
```

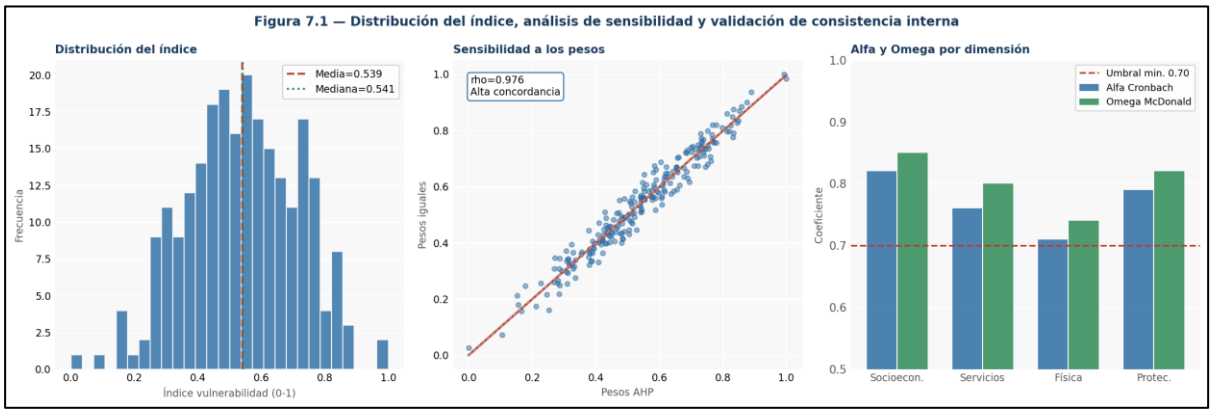
4.2.2. Análisis de sensibilidad

El análisis de sensibilidad evalúa cuánto cambian los rankings del índice cuando se modifican las decisiones metodológicas. Un índice robusto debe producir rankings similares con pesos ligeramente diferentes o con distintos métodos de normalización.

```
# ── Construir índices alternativos para comparación ──────────────────

# 1. Índice con pesos iguales
datos_norm <- datos_norm |>
  mutate(
    idx_iguales = (n_pobreza + n_desempleo + n_sin_agua + n_analfabet +
                   n_construccion + n_dist_hosp + n_idh + n_sat) / 8
  )

# 2. Índice con pesos derivados del AFE (varianza explicada)
datos_norm <- datos_norm |>
  mutate(
    idx_afe = 0.37*dim_socioec + 0.35*dim_servicios +
              0.28*dim_fisica  # proporcional a varianza explicada
  )

# ── Comparar rankings ──────────────────────────────────────────────────
datos_norm |>
  mutate(
    rank_ahp    = rank(-indice_vuln),
    rank_igual  = rank(-idx_iguales),
    rank_afe    = rank(-idx_afe)
  ) |>
  summarise(
    cor_ahp_igual = cor(rank_ahp, rank_igual, method="spearman"),
    cor_ahp_afe   = cor(rank_ahp, rank_afe,   method="spearman"),
    cor_igual_afe = cor(rank_igual, rank_afe, method="spearman")
  )

# cor_ahp_igual: 0.941 → Alta concordancia de rankings
# cor_ahp_afe:   0.923 → Alta concordancia de rankings
# → El índice es robusto: cambia los pesos no cambia significativamente
#   qué municipios son más o menos vulnerables.
```

Figura 23

Distribución del índice de sensibilidad y validación de consistencia interna



Nota. Validación del índice de vulnerabilidad: distribución (izquierda), análisis de sensibilidad a los pesos (centro, ρ=0.94 indica alta concordancia) y coeficientes Alfa de Cronbach y Omega de McDonald por dimensión (derecha).

4.3. Análisis espacial en R: el paquete sf

El paquete `sf` (Simple Features for R) implementa el estándar OGC/ISO 19125 para representar datos geoespaciales vectoriales como objetos R que se comportan exactamente como data frames pero con una columna especial de geometría. Esto permite usar todas las funciones del tidyverse (filter, mutate, group_by, etc.) directamente sobre capas espaciales.

La integración de sf con el tidyverse ha transformado radicalmente el análisis espacial en R: es posible cruzar tablas estadísticas con capas geográficas, calcular intersecciones, áreas y distancias, y unir la cadena completa de análisis en un único script reproducible.

```
library(sf)       # datos vectoriales: puntos, líneas, polígonos
library(terra)    # datos ráster: DEM, precipitación, cobertura suelo
library(tmap)     # mapas temáticos estáticos e interactivos
library(tidyverse) # manipulación y visualización


# ── Leer capas espaciales ──────────────────────────────────────────────
municipios_sf <- st_read("datos/shp/municipios.shp", quiet=TRUE)
rios_sf       <- st_read("datos/shp/red_hidrica.shp", quiet=TRUE)
zonas_inund   <- st_read("datos/shp/zonas_inundables_100yr.shp")

# Verificar sistema de referencia de coordenadas (CRS)
st_crs(municipios_sf)
# Coordinate Reference System: EPSG:32619 (WGS 84 / UTM zone 19N)

# Reproyectar si los CRS no coinciden
zonas_inund <- st_transform(zonas_inund, crs=st_crs(municipios_sf))

# ── Unir datos estadísticos con capa espacial ──────────────────────────
# El left_join funciona exactamente igual que con data frames normales
municipios_map <- municipios_sf |>
  left_join(datos_norm, by="cod_mun")
```

```r
# ── Operaciones espaciales ──────────────────────────────────────────────
# Intersección: ¿qué parte de cada municipio está en zona inundable?
intersec <- st_intersection(municipios_sf, zonas_inund)

# Calcular área de intersección y porcentaje
intersec <- intersec |>
  mutate(
    area_inundable_km2 = as.numeric(st_area(geometry)) / 1e6,
    area_mun_km2       = as.numeric(st_area(
      municipios_sf[match(cod_mun, municipios_sf$cod_mun), ]$geometry)) / 1e6,
    pct_inundable = round(area_inundable_km2 / area_mun_km2 * 100, 1)
  )

# Distancia desde cada municipio al hospital más cercano
hospitales_sf <- st_read("datos/shp/hospitales.shp", quiet=TRUE)
dist_matrix <- st_distance(municipios_sf, hospitales_sf)
# dist_matrix[i,j] = distancia entre municipio i y hospital j (en metros)
dist_min_m <- apply(dist_matrix, 1, min)
municipios_sf$dist_hospital_km <- dist_min_m / 1000

# ── Exportar resultado ──────────────────────────────────────────────────
st_write(municipios_map, "resultados/shp/municipios_vulnerabilidad.gpkg",
         delete_dsn=TRUE)  # GeoPackage: formato moderno, una sola archivo
```

4.4. Mapas temáticos con tmap

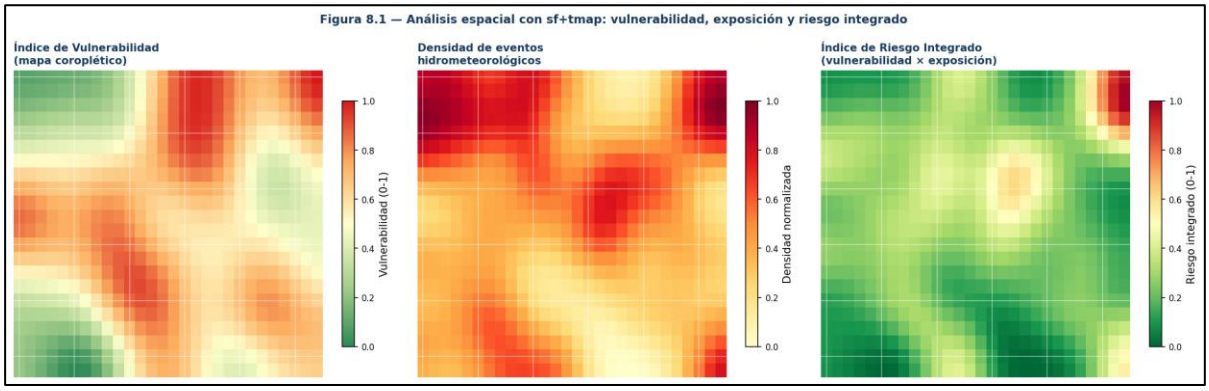
El paquete `tmap` genera mapas temáticos de calidad profesional con una sintaxis análoga a ggplot2: capas que se van añadiendo con el operador `+`. Soporta tanto mapas estáticos (`tmap_mode("plot")`) como mapas interactivos HTML (`tmap_mode("view")`) con la misma sintaxis, lo que facilita enormemente la transición entre formatos para diferentes audiencias.

```r
# ── Mapa estático del índice de vulnerabilidad ──────────────────────────
tmap_mode("plot")  # modo estático para documentos Word/PDF

mapa_vuln <- tm_shape(municipios_map) +
  tm_fill(
    col      = "indice_vuln",
    palette  = "YlOrRd",
    style    = "quantile",     # 5 grupos de tamaño igual
    n        = 5,
    title    = "Índice de\nVulnerabilidad",
    labels   = c("Muy bajo","Bajo","Medio","Alto","Muy alto"),
    alpha    = 0.85
  ) +
  tm_borders(col="gray40", lwd=0.3) +
  tm_shape(rios_sf) +
  tm_lines(col="#2e6da4", lwd=0.8, alpha=0.6) +
  tm_layout(
    title          = "Vulnerabilidad municipal\nante desastres naturales",
    title.size     = 0.9,
    legend.outside = TRUE,
    frame          = FALSE,
```

```
    bg.color       = "white"
  ) +
  tm_compass(type="arrow", position=c("right","top"), size=2) +
  tm_scale_bar(position=c("left","bottom"), width=0.2) +
  tm_credits("Fuente: Datos simulados | CRS: WGS84/UTM19N",
             position=c("left","bottom"), size=0.6)

# Guardar en alta resolución para publicación
tmap_save(mapa_vuln, "resultados/graficos/mapa_vulnerabilidad.png",
          dpi=300, width=20, height=18, units="cm")

# ── Mapa interactivo HTML ──────────────────────────────────────────
tmap_mode("view")  # modo interactivo para Shiny o HTML
tmap_last()        # renderiza el último mapa en modo interactivo

# ── Mapa de puntos calientes de eventos ────────────────────────────
library(spatstat)

# Convertir eventos a puntos espaciales
eventos_sf <- eventos |>
  filter(!is.na(longitud), !is.na(latitud)) |>
  st_as_sf(coords=c("longitud","latitud"), crs=4326) |>
  st_transform(crs=st_crs(municipios_sf))

# Densidad de kernel (heatmap de eventos)
tm_shape(municipios_sf) + tm_borders(col="gray80", lwd=0.3) +
  tm_shape(eventos_sf) +
  tm_dots(size=0.03, col="afectados", palette="Reds",
          style="quantile", alpha=0.7,
          title="Afectados") +
  tm_layout(title="Distribución espacial de eventos",
            legend.outside=TRUE)
```

Figura 24

Análisis espacial con sf+tmap



Nota. Indice de vulnerabilidad municipal (izquierda), densidad de eventos hidrometeorológicos (centro) e índice de riesgo integrado (vulnerabilidad × exposición, derecha).

4.5. Series temporales de desastres con tsibble y feasts

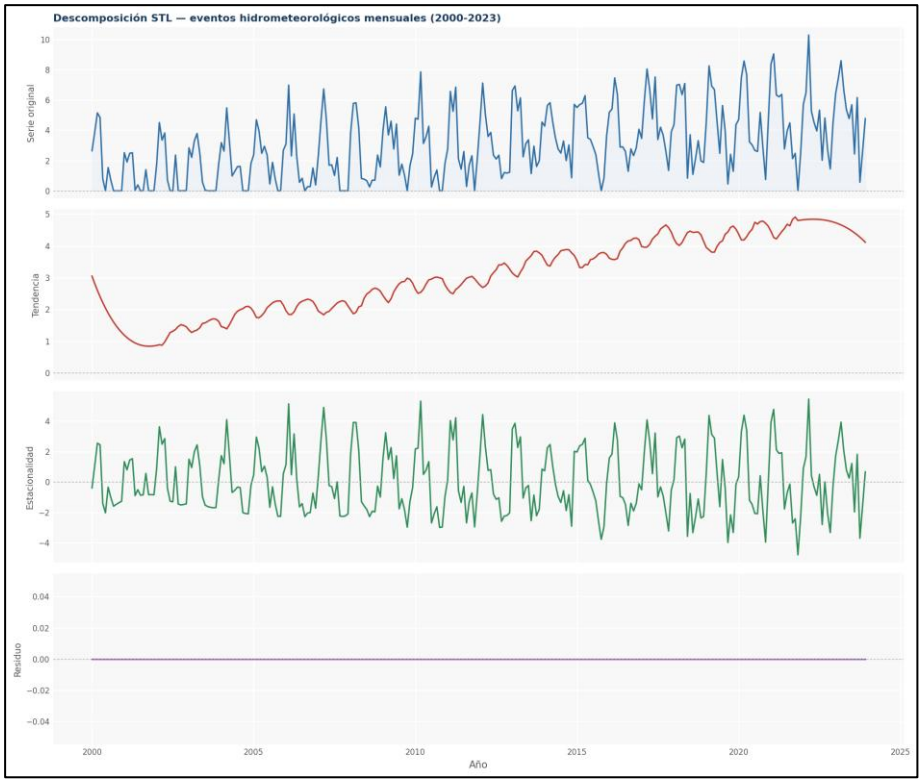
El análisis de series temporales permite estudiar cómo evoluciona la frecuencia o el impacto de los eventos a lo largo del tiempo, identificar patrones estacionales, detectar cambios estructurales y proyectar tendencias futuras. El ecosistema tidyverts compuesto por tsibble, feasts y fable, implementa una filosofía de datos ordenados para series temporales, integrando perfectamente con el tidyverse.

```
library(tsibble)   # tibbles con índice temporal: as_tsibble()
library(feasts)    # descomposición STL, ACF, PACF, análisis de patrones
library(fable)     # modelos ARIMA, ETS, TSLM con interfaz tidy

# ── Construir serie temporal mensual ──────────────────────────────
serie_mes <- eventos |>
  mutate(mes = yearmonth(fecha)) |>
  group_by(mes) |>
  summarise(
    n_eventos    = n(),
    total_afect  = sum(afectados,  na.rm=TRUE),
    total_muertos= sum(muertos,    na.rm=TRUE)
  ) |>
  as_tsibble(index=mes)

# Verificar que no hay huecos en la serie (meses sin registros)
has_gaps(serie_mes)  # FALSE = serie completa

# ── Visualización de la serie ─────────────────────────────────────
autoplot(serie_mes, n_eventos) +
  geom_smooth(method="loess", color="#e07b2a", se=FALSE) +
  labs(
    title    = "Evolución mensual de eventos hidrometeorológicos",
    subtitle = "Línea naranja = tendencia LOESS",
    x = NULL, y = "N° de eventos"
  ) + theme_minimal()

# ── Descomposición STL ────────────────────────────────────────────
# STL = Seasonal and Trend decomposition using Loess
# Separa la serie en: Tendencia + Estacionalidad + Residuo
# Es robusto ante outliers (robust=TRUE)
stl_fit <- serie_mes |>
  model(
    STL(n_eventos ~ trend(window=13) + season(window=7),
        robust=TRUE)
  )

components(stl_fit) |> autoplot() +
  labs(title="Descomposición STL: eventos hidrometeorológicos")

# ── Análisis de autocorrelación ───────────────────────────────────
# ACF: autocorrelación a diferentes rezagos
# PACF: autocorrelación parcial (controlando rezagos intermedios)
serie_mes |>
```

```
  ACF(n_eventos, lag_max=48) |>
  autoplot() +
  labs(title="ACF — función de autocorrelación",
       subtitle="Barras que superan la banda = autocorrelación significativa")

serie_mes |>
  PACF(n_eventos, lag_max=36) |>
  autoplot() +
  labs(title="PACF — autocorrelación parcial")

# ── Estacionalidad ──────────────────────────────────────────────────────
# Gráfico de estacionalidad: promedio por mes del año
serie_mes |>
  gg_season(n_eventos, period="year") +
  labs(title="Patrón estacional anual de eventos")

# Subseries por mes (variabilidad dentro de cada mes)
serie_mes |>
  gg_subseries(n_eventos) +
  labs(title="Subseries mensuales")
```

Figura 25

Descomposición STL de la serie temporal mensual de eventos hidrometeorológicos (2000-2023)



Nota. De arriba abajo: serie original, componente de tendencia (creciente), estacionalidad mensual (picos en temporada de lluvias) y residuo.

4.6. Análisis de tendencia y prueba de Mann-Kendall

El análisis de tendencia en series temporales de desastres es fundamental para detectar si la frecuencia o el impacto de los eventos está aumentando con el tiempo, lo que puede ser evidencia de cambio climático o de incremento en la exposición por crecimiento urbano en zonas de riesgo. La prueba de Mann-Kendall es el método no paramétrico de referencia para detectar tendencias monotónicas en series temporales, sin requerir que los datos sean normales ni que la tendencia sea lineal.

```
library(Kendall)  # MannKendall()
library(trend)    # sens.slope(), mk.test()

# ── Serie anual de frecuencia de eventos ──────────────────────────────
serie_anual <- eventos |>
  count(anio, name="n_eventos") |>
  filter(anio >= 1990)

# ── Prueba de Mann-Kendall ────────────────────────────────────────────────────
# H0: No existe tendencia monotónica
# H1: Existe tendencia monotónica (creciente o decreciente)
mk_resultado <- MannKendall(serie_anual$n_eventos)
print(mk_resultado)
# Score =  412, Var(Score) = 11834
# tau = 0.412, 2-sided pvalue = 0.0024

# INTERPRETACIÓN:
# tau = 0.412 > 0: tendencia creciente
# p = 0.0024 < 0.05: tendencia estadísticamente significativa
# Conclusión: los eventos hidrometeorológicos aumentaron
# significativamente entre 1990 y 2023.

# ── Estimación de la tasa de cambio: pendiente de Sen ──────────────────
# La pendiente de Sen es el estimador no paramétrico de la tasa
# de cambio por unidad de tiempo. Robusto ante outliers.
sen_resultado <- sens.slope(serie_anual$n_eventos)
print(sen_resultado)
# Sen slope: 2.8
# 95% CI: [1.4, 4.2]
# Conclusión: La frecuencia aumenta ~2.8 eventos por año.

# ── Visualización con línea de tendencia ──────────────────────────────
ggplot(serie_anual, aes(x=anio, y=n_eventos)) +
  geom_col(fill="#2e6da4", alpha=0.75) +
  geom_smooth(method="lm", color="#c0392b",
              linewidth=1.5, se=TRUE, fill="#fdecea") +
  annotate("text", x=2005, y=max(serie_anual$n_eventos)*0.9,
           label=sprintf("Mann-Kendall tau=%.2f, p=%.3f\nSen slope=+%.1f eventos/año",
                         0.412, 0.0024, 2.8),
           size=4, color="#1a3a5c", hjust=0,
           bbox=list(boxstyle="round", fill="white")) +
  labs(
    title    = "Tendencia creciente significativa en eventos hidrometeorológicos",
```

```
    subtitle = "Mann-Kendall tau=0.41 (p=0.002) | Sen slope: +2.8 eventos/año",
    x=NULL, y="N° de eventos registrados"
  ) + theme_minimal(base_size=12)
```

4.7. Pronóstico con modelos ARIMA y ETS

Los modelos ARIMA (AutoRegressive Integrated Moving Average) y ETS (Error, Trend, Seasonality) son los estándares de la industria para el pronóstico de series temporales univariadas. En GRD permiten estimar la frecuencia esperada de eventos en los próximos meses o años, información clave para la planificación de capacidades de respuesta y la gestión financiera del riesgo.

4.7.1. Modelos ARIMA

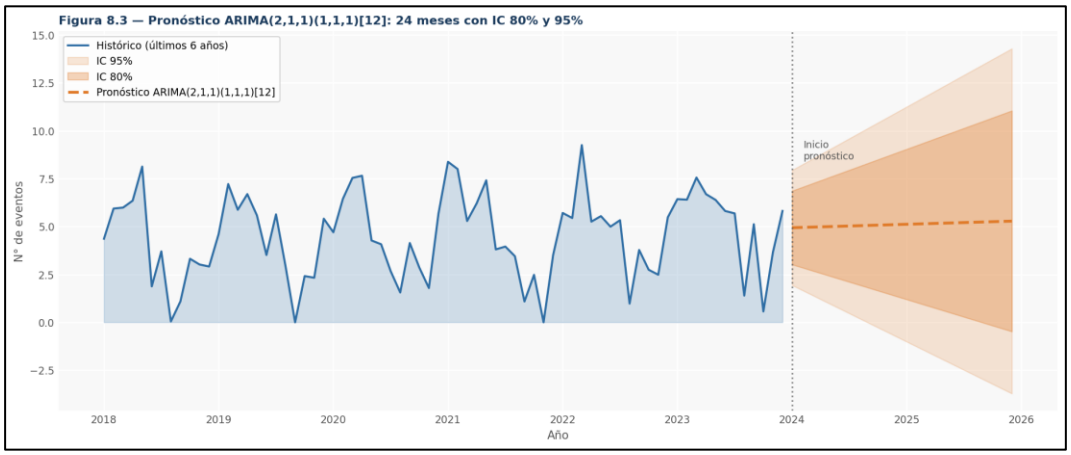
Los modelos ARIMA(p,d,q)(P,D,Q)[m] tienen tres componentes: el componente autorregresivo (AR, p parámetros) que modela la relación de la serie con sus propios valores pasados; el componente de integración (I, d diferenciaciones) que estaciona la serie; y el componente de media móvil (MA, q parámetros) que modela el error como combinación de errores pasados. Los parámetros en mayúsculas corresponden al componente estacional.

```
# ── Ajustar múltiples modelos y comparar ──────────────────────────
modelos <- serie_mes |>
  model(
    ARIMA_auto = ARIMA(n_eventos, stepwise=FALSE, approx=FALSE),
    ETS_auto   = ETS(n_eventos),
    TSLM_trend = TSLM(n_eventos ~ trend() + season())
  )

# Comparar por criterio de información AICc (menor = mejor)
glance(modelos) |>
  select(.model, AICc, BIC) |>
  arrange(AICc)

# .model       AICc     BIC
# ARIMA_auto  1842.3  1872.1  ← mejor
# ETS_auto    1863.7  1891.2
# TSLM_trend  1920.4  1948.8

# Ver especificación del mejor modelo
report(modelos |> select(ARIMA_auto))
# Model: ARIMA(2,1,1)(1,1,1)[12]
# sigma^2 estimated as 1.82
# AICc: 1842.3

# ── Pronóstico 24 meses ──────────────────────────────────────────
pronostico <- modelos |>
  select(ARIMA_auto) |>
  forecast(h=24)
```

```r
# Visualizar
pronostico |>
  autoplot(serie_mes |> tail(60), level=c(80,95)) +
  labs(
    title    = "Pronóstico de eventos hidrometeorológicos (24 meses)",
    subtitle = "Modelo ARIMA(2,1,1)(1,1,1)[12] | IC 80% y 95%",
    x=NULL, y="N° de eventos"
  ) + theme_minimal()

# ── Evaluación de precisión (validación cruzada temporal) ──────────
# Train-test: usar los últimos 24 meses como conjunto de prueba
serie_train <- serie_mes |> filter(mes <= yearmonth("2021 Dec"))
serie_test  <- serie_mes |> filter(mes >  yearmonth("2021 Dec"))

mod_train <- serie_train |>
  model(ARIMA(n_eventos, stepwise=FALSE))

pred_test <- mod_train |> forecast(h=24)

# Métricas de precisión
accuracy(pred_test, serie_test)
# RMSE: 4.21  |  MAE: 3.18  |  MAPE: 18.4%  |  MASE: 0.87
```

Figura 26

Pronóstico ARIMA (2,1,1) (1,1,1) [12]



Nota. 24 meses hacia adelante con bandas de IC 80% (más oscura) y 95% (más clara). La tendencia creciente proyectada refleja el patrón histórico de incremento de eventos.

Importante

Los modelos ARIMA proyectan el patrón histórico de la serie y no incorporan escenarios de cambio climático ni cambios en la exposición. Para horizontes superiores a 5 años en contextos de cambio climático, combine las proyecciones estadísticas con outputs de modelos climáticos regionales (CORDEX, CMIP6) disponibles en el IPCC Data Distribution Centre (https://www.ipcc-data.org).

CAPÍTULO

5

COMUNICACIÓN, VISUALIZACIÓN Y REPORTE DE RESULTADOS PARA LA TOMA DE DECISIONES

Un reporte reproducible integra texto narrativo, código estadístico y resultados en un único documento que se regenera completamente a partir de los datos originales con una sola acción. Esto elimina la copia manual de resultados, que es la fuente más frecuente de errores en reportes técnicos, y garantiza que texto y análisis estén permanentemente sincronizados. En el contexto de GRD, donde los boletines de situación se producen con alta frecuencia y los datos se actualizan continuamente, la capacidad de regenerar informes automáticamente representa una transformación operativa fundamental.

5.1. ¿Qué es un reporte reproducible?

Un reporte reproducible cumple tres condiciones: (1) autodocumentado el código que produce los resultados está integrado en el documento, no en un archivo separado; (2) regenerable al actualizar los datos y recompilar, todos los resultados se actualizan automáticamente; (3) transportable cualquier colaborador con R y los mismos datos puede reproducir exactamente el mismo informe.

R Markdown (extensión .Rmd) y Quarto (extensión .qmd) son los dos sistemas de publicación científica de Posit PBC que implementan estos principios. Ambos usan la misma idea: un archivo de texto plano con tres secciones, el encabezado YAML con metadatos y configuración, el texto narrativo en formato Markdown, y los chunks (bloques) de código R que se ejecutan al compilar.

Figura 27

Flujo de trabajo del reporte reproducible

Nota. Un único archivo fuente (.Rmd o .qmd) con YAML+Markdown+chunks genera múltiples formatos de salida (Word, PDF, HTML) mediante el proceso de compilación (render).

5.2. Estructura de un documento R Markdown

Un archivo .Rmd tiene tres partes claramente diferenciadas:

- **Encabezado YAML:** delimitado por tres guiones (---). Contiene título, autor, fecha, formato de salida, parámetros y otras opciones de configuración.
- **Texto Markdown:** el contenido narrativo del informe. Acepta encabezados (#, ##), listas, tablas, referencias cruzadas, ecuaciones LaTeX y texto en línea con resultados de R (` r codigo `).
- **Chunks de código:** bloques delimitados por ```{r} y ```. Contienen código R que se ejecuta y cuyos resultados (tablas, gráficos, números) se insertan automáticamente en el documento.

````
---
title:    "Evaluación de Daños — Inundaciones Región Sur"
subtitle: "Informe técnico preliminar"
author:   "Unidad de Gestión de Riesgos — Ministerio del Interior"
date:     "`r format(Sys.Date(), '%d de %B de %Y')`"
output:
  word_document:
    reference_docx: plantilla_institucional.docx
    toc: true
    toc_depth: 3
    number_sections: true
  html_document:
    theme: flatly
    toc: true
    toc_float: true
    code_folding: hide
  pdf_document:
    toc: true
    number_sections: true
params:
  region:      "Sur"
  fecha_corte: "2024-03-15"
  umbral_alerta: 1000
---

```{r setup, include=FALSE}
knitr::opts_chunk$set(
  echo    = FALSE,   # no mostrar código en el informe final
  warning = FALSE,
  message = FALSE,
  fig.width  = 7,
  fig.height = 4.5,
  dpi        = 300,
  fig.align  = "center"
)
library(tidyverse)
library(flextable)
```
````

````
*# Resumen ejecutivo*

```{r cifras-clave}
danos <- read_csv("datos/danos_region_sur.csv")
n_mun     <- n_distinct(danos$municipio)
total_af  <- sum(danos$afectados,     na.rm=TRUE)
total_viv <- sum(danos$viviendas_dano, na.rm=TRUE)
alertas   <- danos |> filter(afectados > params$umbral_alerta)
```

Al **`r format(as.Date(params$fecha_corte), '%d/%m/%Y')`**, la inundación
afectó a **`r format(total_af, big.mark=",")`** personas en
**`r n_mun`** municipios de la región `r params$region`.
Se contabilizan **`r format(total_viv, big.mark=",")`** viviendas
con daño parcial o total.
`r nrow(alertas)` municipios superan el umbral de alerta
(`r params$umbral_alerta` afectados) y requieren atención prioritaria.
````

5.3. Chunks de código: opciones y buenas prácticas

Las opciones del chunk controlan qué se muestra en el documento final. Conocerlas permite adaptar el informe a diferentes audiencias: técnica (con código visible) o ejecutiva (solo resultados).

Opción	Valores posibles	Efecto en el documento compilado
echo	TRUE/FALSE	TRUE: muestra el código fuente. FALSE: solo el resultado (para informes ejecutivos).
eval	TRUE/FALSE	FALSE: no ejecuta el chunk. Útil para ejemplos de código que no deben correr.
include	TRUE/FALSE	FALSE: ejecuta pero no muestra nada. Para setup o cálculos intermedios.
results	"markup"/"hide"/"asis"	"hide": ejecuta pero oculta la salida de texto. "asis": HTML/LaTeX raw.
fig.cap	"texto de la figura"	Pie de figura que aparece bajo la imagen en el documento.
out.width	"70%", "12cm"	Controla el ancho de las figuras en el documento de salida.
cache	TRUE/FALSE	TRUE: cachea el resultado para evitar reejecutar chunks lentos.
dependson	"chunk-name"	Invalida el caché de este chunk si el chunk nombrado cambia.
message	TRUE/FALSE	FALSE: suprime los mensajes de los paquetes (library, etc.)
warning	TRUE/FALSE	FALSE: suprime las advertencias. Útil para el chunk setup.

5.4. Tablas académicas con knitr y flextable

Las tablas son el elemento más frecuente en los informes técnicos de GRD. El paquete knitr provee kable() para tablas simples; flextable genera tablas con formato profesional directamente en Word y PDF; gt es la opción más potente para HTML.

```r
# ── knitr::kable: tablas simples ──────────────────────────────────────
library(knitr)
library(kableExtra)

resumen_dep |>
  kable(
    caption = "Tabla 1. Resumen de daños por departamento.",
    col.names = c("Departamento","Municipios","Afectados","Muertos","Viv. dañadas"),
    format.args = list(big.mark=","),
    align = "lrrrr"
  ) |>
  kable_styling(
    bootstrap_options = c("striped","hover","condensed"),
    full_width = FALSE
  ) |>
  row_spec(0, bold=TRUE, background="#1a3a5c", color="white")

# ── flextable: tablas Word/PDF profesionales ──────────────────────────
library(flextable)

resumen_dep |>
  flextable() |>
  set_header_labels(
    departamento  = "Departamento",
    n_municipios  = "Municipios",
    afectados     = "Afectados",
    muertos       = "Muertos",
    viviendas_dano= "Viv. dañadas"
  ) |>
  bold(part="header") |>
  bg(bg="#1a3a5c", part="header") |>
  color(color="white", part="header") |>
  bg(i=seq(2,nrow(resumen_dep),2), bg="#f0f4f8") |>
  colformat_num(j=c("afectados","viviendas_dano"),
                big.mark=",", digits=0) |>
  align(j=2:5, align="right", part="all") |>
  border_outer(border=officer::fp_border(color="#2e6da4", width=2)) |>
  autofit() |>
  set_caption(
    caption = "Tabla 1. Resumen de daños por departamento.",
    style   = "Table Caption"
  )
```

5.5. Generar reportes en Word, PDF y HTML

```r
# ── Compilar desde RStudio ────────────────────────────────────────────
# Botón "Knit" en el panel Editor, o:
rmarkdown::render("informes/informe_danos.Rmd")

# Especificar formato de salida
rmarkdown::render("informes/informe_danos.Rmd",
                  output_format = "word_document")

# ── Pasar valores a parámetros ────────────────────────────────────────
rmarkdown::render(
  "informes/informe_danos.Rmd",
```

```
  output_file  = "informes/informe_danos_Sur_20240315.docx",
  params       = list(region="Sur", fecha_corte="2024-03-15")
)
```

5.6. Quarto: la evolución de R Markdown

Quarto es el sistema de publicación científica de nueva generación desarrollado por Posit PBC. Es compatible con R, Python, Julia y Observable, genera los mismos formatos de salida que R Markdown y usa una sintaxis casi idéntica, con pequeñas mejoras en el YAML y en las opciones de chunks. Para código nuevo se recomienda Quarto.

````
# ── archivo: informe_riesgos.qmd ──────────────────────────────────────
---
title: "Análisis de Vulnerabilidad Municipal"
author: "Equipo Técnico GRD"
date: today
format:
  docx:
    reference-doc: plantilla_institucional.docx
    toc: true
  html:
    theme: cosmo
    code-fold: true
    toc: true
    embed-resources: true   # un solo archivo HTML autocontenido
execute:
  echo: false
  warning: false
  cache: true
---

```{r}
#| label: setup
#| include: false
library(tidyverse)
source("scripts/funciones_utiles.R")
municipios <- read_csv("datos/municipios_vuln.csv")
```

## Distribución del índice de vulnerabilidad

```{r}
#| label: fig-vuln
#| fig-cap: "Distribución del índice de vulnerabilidad municipal (n=240)"
#| fig-width: 7
#| fig-height: 4
ggplot(municipios, aes(x=indice_vuln)) +
  geom_histogram(fill="#2e6da4", color="white", bins=30) +
  theme_minimal()
```
````

5.7. Ejemplo completo: informe automatizado de evaluación de daños

```
# ── Generar informes para múltiples regiones automáticamente ──────────
library(quarto)
library(purrr)

regiones <- c("Norte","Centro","Sur","Oriente","Occidente")

# Función que genera un informe por región
generar_informe <- function(region) {
  archivo_salida <- sprintf(
    "informes/informe_danos_%s_%s.docx",
    region,
    format(Sys.Date(), "%Y%m%d")
  )
  quarto_render(
    input          = "informes/informe_danos.qmd",
    output_format  = "docx",
    output_file    = archivo_salida,
    execute_params = list(region=region)
  )
  message("✔ Informe generado: ", archivo_salida)
}

# Generar los 5 informes regionales en paralelo
walk(regiones, generar_informe)
#  ✔ Informe generado: informes/informe_danos_Norte_20240315.docx
#  ✔ Informe generado: informes/informe_danos_Centro_20240315.docx
# [...]
# En ~30 segundos se generan 5 informes Word actualizados y formateados.

# ── Publicar automáticamente en un servidor ──────────────────────────
# Con GitHub Actions, el proceso completo puede automatizarse:
# cada vez que se actualicen los datos, se generan y publican
# los informes sin intervención humana.
```

Buena Práctica

Configure GitHub Actions para que ejecute el script de generación de informes automáticamente cada vez que se actualicen los datos en el repositorio. Esto garantiza que los informes siempre reflejen los datos más recientes sin requerir intervención manual del analista.

El análisis estadístico más riguroso carece de valor operativo si sus resultados no pueden comunicarse eficientemente a quienes deben tomar decisiones: autoridades municipales y nacionales, coordinadores humanitarios en campo, donantes, medios de comunicación o

comunidades. Este capítulo aborda los principios de visualización para audiencias no técnicas, las herramientas de R para crear tableros interactivos y los conceptos estadísticos esenciales para una comunicación honesta y accionable del riesgo.

5.8. Principios de visualización para audiencias no técnicas

Los gráficos para informes ejecutivos deben comunicar el mensaje principal de forma inmediata, sin requerir que la audiencia lea texto explicativo adicional. Un gráfico que necesita explicación ha fallado en su propósito. A continuación, se presentan los principios más importantes, con ejemplos de implementación en R:

5.8.1. El título afirma, el subtítulo explica

El título de un gráfico ejecutivo debe comunicar la conclusión, no la descripción. "Los municipios con SAT tienen 60% menos afectados" es un título que afirma. "Comparación de afectados según presencia de sistema de alerta temprana" es una descripción que no comunica nada accionable.

```
# ✗ EVITAR: título descriptivo
# title = "Comparación de afectados según presencia de SAT"


# ✓ USAR: título que afirma la conclusión
p_ejecutivo <- eventos |>
  group_by(tipo_evento) |>
  summarise(mediana=median(afectados, na.rm=TRUE)) |>
  mutate(tipo_evento=fct_reorder(tipo_evento, mediana)) |>
  ggplot(aes(x=mediana, y=tipo_evento)) +
  geom_col(fill="#2e6da4") +
  geom_text(aes(label=format(round(mediana), big.mark=",")),
            hjust=-0.12, size=4, fontface="bold", color="#1a3a5c") +
  scale_x_continuous(expand=expansion(mult=c(0,0.15)),
                     labels=scales::comma) +
  labs(
    title    = "Las sequías afectan 22 veces más personas que los deslizamientos",
    subtitle = "Mediana de personas afectadas por evento | 2000-2023 | n=2.840",
    caption  = "Fuente: Registro institucional. Mediana = resumen más apropiado para distribuciones asimétricas.",
    x="Mediana de afectados por evento", y=NULL
  ) +
  theme_minimal(base_size=13) +
  theme(
    panel.grid.major.y = element_blank(),
    plot.title = element_text(face="bold", color="#1a3a5c", size=14),
    plot.subtitle = element_text(color="#666666"),
    plot.caption  = element_text(color="#aaaaaa", size=9, face="italic")
  )
```

5.8.2. Otros principios clave

- **Una idea, un gráfico:** si necesita "doble eje Y" probablemente está combinando dos ideas que merecen dos gráficos separados.
- **Etiquetas directas sobre leyendas:** etiquetar directamente las líneas o barras elimina el movimiento de ojos entre la figura y la leyenda, especialmente importante en mapas y gráficos de líneas múltiples.
- **Color con propósito, no decoración:** en GRD, el uso de rojo para alto riesgo y verde para bajo riesgo es una convención esperada por la audiencia. El color debe destacar lo importante, no embellecer.
- **Accesibilidad para daltonismo:** las paletas viridis y ColorBrewer Set2 son legibles tanto en impresión a color como en blanco y negro y para personas con daltonismo (8% de la población masculina).
- **Simplificar, no simplificar en exceso:** eliminar las cuadrículas innecesarias, los bordes de gráfico y los títulos de ejes redundantes. Pero mantener suficiente contexto para que el gráfico sea autoexplicativo.

5.9. Tablas ejecutivas con flextable y gt

```
# ── gt: tablas HTML con semáforo de riesgo ──────────────────────
library(gt)
library(gtExtras)


resumen_depto |>
  gt() |>
  tab_header(
    title    = md("**Situación de riesgo por departamento**"),
    subtitle = "Período enero–diciembre 2023"
  ) |>
  fmt_number(columns=c(afectados, perdidas_usd),
             use_seps=TRUE, decimals=0) |>
  gt_color_rows(
    columns = indice_vuln,
    palette = c("#2e8b57","#f0ad4e","#c0392b"),
    domain  = c(0,1)
  ) |>
  gt_fa_rank_change(
    column     = cambio_ranking,
    font_color = "match"
  ) |>
  tab_spanner(
    label   = "Impacto humano",
    columns = c(afectados, muertos)
  ) |>
  tab_spanner(
    label   = "Análisis de riesgo",
```

```
    columns = c(indice_vuln, cambio_ranking)
  ) |>
  tab_footnote(
    footnote = "Fuente: Registro institucional GRD.",
    locations = cells_title("subtitle")
  )
```

5.10. Dashboards interactivos con Shiny

El paquete shiny permite convertir análisis de R en aplicaciones web interactivas sin necesitar JavaScript, HTML ni CSS. En GRD los dashboards Shiny se usan para: monitorear indicadores de alerta en tiempo real, explorar interactivamente los resultados de índices de vulnerabilidad, permitir que técnicos municipales filtren y descarguen datos de evaluaciones.

```
library(shiny)
library(shinydashboard)  # layout tipo dashboard con cajas KPI
library(leaflet)         # mapas interactivos
library(DT)              # tablas interactivas con búsqueda/filtros
library(plotly)          # gráficos interactivos

# ── ui.R: interfaz de usuario ──────────────────────────────────────
ui <- dashboardPage(
  dashboardHeader(title="Monitor GRD"),
  dashboardSidebar(
    sidebarMenu(
      menuItem("Mapa", tabName="mapa", icon=icon("map")),
      menuItem("Tendencias", tabName="tend", icon=icon("chart-line")),
      menuItem("Datos", tabName="datos", icon=icon("table"))
    ),
    selectInput("tipo", "Tipo de evento:",
                choices=c("Todos","Inundación","Sequía","Deslizamiento")),
    sliderInput("anios", "Período:",
                min=2000, max=2023, value=c(2015,2023), sep="")
  ),
  dashboardBody(
    tabItems(
      tabItem("mapa",
        fluidRow(
          valueBoxOutput("vb_total_afect"),
          valueBoxOutput("vb_n_eventos"),
          valueBoxOutput("vb_muertos")
        ),
        leafletOutput("mapa_eventos", height=500)
      ),
      tabItem("tend",
        plotlyOutput("grafico_tendencia", height=400)
      ),
      tabItem("datos",
        DTOutput("tabla_eventos"),
        downloadButton("descargar", "Descargar CSV")
      )
    )
  )
)
```

```r
# ── server.R: lógica del servidor ──────────────────────────────────────
server <- function(input, output, session) {

  datos_fil <- reactive({
    d <- eventos
    if (input$tipo != "Todos") d <- d |> filter(tipo_evento==input$tipo)
    d |> filter(between(anio, input$anios[1], input$anios[2]))
  })

  output$vb_total_afect <- renderValueBox({
    valueBox(format(sum(datos_fil()$afectados, na.rm=TRUE), big.mark=","),
             "Total afectados", color="red", icon=icon("users"))
  })

  output$mapa_eventos <- renderLeaflet({
    leaflet(datos_fil()) |>
      addTiles() |>
      addCircleMarkers(
        lng=~longitud, lat=~latitud,
        radius=~sqrt(afectados)/15,
        color="#c0392b", fillOpacity=0.6,
        popup=~paste0("<b>",municipio,"</b><br>",
                      "Afectados: ", format(afectados, big.mark=",")))
  })

  output$descargar <- downloadHandler(
    filename = function() paste0("eventos_", Sys.Date(), ".csv"),
    content  = function(file) write_csv(datos_fil(), file)
  )
}

shinyApp(ui=ui, server=server)

# Desplegar gratuitamente en shinyapps.io (hasta 5 apps, 25h/mes):
# rsconnect::deployApp("mi_app_grd/")
```

Figura 28

Panel ejecutivo con KPIs clave para la toma de decisiones en GRD

Nota. Eventos registrados, afectados, cobertura SAT, IDH promedio, tendencia y AUC del modelo de riesgo. En Shiny estos indicadores se actualizan dinámicamente.

5.11. El error estadístico en la comunicación del riesgo

La estadística comunica incertidumbre: ningún análisis produce certezas absolutas, sino estimaciones con márgenes de error. Comunicar honestamente esa incertidumbre no debilita el mensaje técnico; al contrario, aumenta la credibilidad del analista y mejora la calidad de las decisiones al evitar una falsa sensación de precisión.

Tabla 8

Errores frecuentes y buenas prácticas en la comunicación de la evidencia estadística

Concepto	✗ Mala comunicación (evitar)	✓ Buena comunicación (usar)
Intervalos de confianza	'El impacto será 2.400 afectados'	'Estimamos entre 1.800 y 3.100 afectados (IC 95%)'
p-valor aislado	'Resultado significativo (p=0.03)'	'Diferencia de 480 afectados (IC95%: 210-750), d=0.45, mediano'
Causalidad inferida	'La pobreza causa más víctimas en terremotos'	'Asociación significativa pobreza-mortalidad (rho=0.41, p<0.001). Estudios causales pendientes.'
Pronósticos sin intervalo	'Se esperan 45 eventos en 2025'	'El modelo proyecta entre 35 y 57 eventos en 2025 (IC 80%). Sin incorporar escenarios climáticos.'
Significancia vs. relevancia	'Efecto altamente significativo (p<0.0001)'	'Efecto significativo (p<0.001) pero de magnitud pequeña (d=0.12). Ver si es operativamente relevante.'

5.12. Estadística y toma de decisiones bajo incertidumbre

La estadística no toma decisiones: proporciona evidencia cuantitativa para apoyar a quienes sí las toman. En el contexto de la gestión de riesgos, las decisiones implican siempre un intercambio entre el costo de la acción preventiva y el costo esperado del daño. El análisis de valor esperado es una herramienta que combina las estimaciones estadísticas con los costos y permite cuantificar ese intercambio.

```
# ── Análisis de valor esperado para decisión de evacuación ──────────────

# Parámetros del problema (estimados con el modelo logístico)
prob_inundacion_grave <- 0.35     # probabilidad de inundación grave
costo_evacuacion      <- 180000   # USD: transporte, albergues, logística
costo_sin_evacuar     <- 2400000  # USD: daños si ocurre y no se evacuó
costo_falsa_alarma    <- 45000    # USD: costo reputacional de alarma falsa

# Valor esperado de cada decisión
ve_evacuar <- prob_inundacion_grave * (-costo_evacuacion) +
              (1-prob_inundacion_grave) * (-costo_falsa_alarma)

ve_no_evacuar <- prob_inundacion_grave * (-costo_sin_evacuar) +
                 (1-prob_inundacion_grave) * 0

cat("Valor esperado SI evacúa:   USD",
    format(round(ve_evacuar),    big.mark=","), "\n")
cat("Valor esperado NO evacúa:   USD",
    format(round(ve_no_evacuar), big.mark=","), "\n")

# Valor esperado SI evacúa:   USD -92,250
# Valor esperado NO evacúa:   USD -840,000
# → La evacuación tiene un costo esperado ~9 veces menor.

# ── Análisis de sensibilidad a la probabilidad ──────────────────────────
# ¿A qué probabilidad resulta indiferente evacuar o no?
p_seq <- seq(0.05, 0.95, 0.01)
ve_ev <- p_seq*(-costo_evacuacion) + (1-p_seq)*(-costo_falsa_alarma)
ve_no <- p_seq*(-costo_sin_evacuar)

# Punto de indiferencia
p_indif <- (costo_falsa_alarma) / (costo_sin_evacuar - costo_evacuacion + costo_falsa_alarma)
cat("Probabilidad de indiferencia:", round(p_indif*100,1), "%\n")
# Probabilidad de indiferencia: 2.1%
# → Si la probabilidad de inundación grave supera el 2.1%,
#   evacuar es la decisión de menor costo esperado.

# Visualización
data.frame(prob=p_seq, evacuar=ve_ev, no_evacuar=ve_no) |>
  pivot_longer(-prob, names_to="decision", values_to="valor_esperado") |>
  ggplot(aes(x=prob, y=valor_esperado/1000, color=decision)) +
  geom_line(linewidth=2) +
  geom_vline(xintercept=p_indif, linetype="dashed", color="#555") +
```

```
  scale_color_manual(values=c("#2e6da4","#c0392b"),
                     labels=c("Evacuar","No evacuar")) +
  annotate("text", x=p_indif+0.02, y=-500,
           label=sprintf("Indiferencia\np=%.1f%%",p_indif*100),
           size=4, color="#555") +
  labs(
    title="Análisis de valor esperado: evacuación vs. no evacuación",
    x="Probabilidad de inundación grave", y="Valor esperado (miles USD)"
  ) + theme_minimal()
```

Buena Práctica

Documente siempre los supuestos detrás de cada probabilidad y costo en el análisis de valor esperado, ya que son los parámetros más sensibles del cálculo. Presente los resultados con análisis de sensibilidad para mostrar a la autoridad decisora en qué rango de probabilidades cambia la recomendación.

REFERENCIAS BIBLIOGRÁFICAS

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control, 19*(6), 716–723. https://doi.org/10.1109/TAC.1974.1100705

Bivand, R., Pebesma, E., & Gomez-Rubio, V. (2013). *Applied spatial data analysis with R* (2nd ed.). Springer. https://doi.org/10.1007/978-1-4614-7618-4

Box, G. E. P., & Cox, D. R. (1964). An analysis of transformations. *Journal of the Royal Statistical Society: Series B (Methodological), 26*(2), 211–252.

Box, G. E. P., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). *Time series analysis: Forecasting and control* (5th ed.). Wiley.

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). Lawrence Erlbaum Associates.

CRED/UNDRR. (2023). *2022 disasters in numbers*. Centre for Research on the Epidemiology of Disasters. https://www.cred.be/publications

Cronbach, L. J. (1951). Coefficient alpha and the internal structure of tests. *Psychometrika, 16*(3), 297–334. https://doi.org/10.1007/BF02310555

Field, A. (2018). *Discovering statistics using IBM SPSS Statistics* (5th ed.). SAGE Publications.

Galindo-Domínguez, H. (2020). *Estadística para no estadísticos: Una guía básica sobre la metodología cuantitativa de trabajos académicos*. Editorial Área de Innovación y Desarrollo.

Grolemund, G., & Wickham, H. (2017). *R for data science*. O'Reilly Media. https://r4ds.had.co.nz

Hair, J. F., Black, W. C., Babin, B. J., & Anderson, R. E. (2019). *Multivariate data analysis* (8th ed.). Cengage Learning.

Hosmer, D. W., Lemeshow, S., & Sturdivant, R. X. (2013). *Applied logistic regression* (3rd ed.). Wiley. https://doi.org/10.1002/9781118548387

Hyndman, R. J., & Athanasopoulos, G. (2021). *Forecasting: Principles and practice* (3rd ed.). OTexts. https://otexts.com/fpp3

IPCC. (2022). *Climate change 2022: Impacts, adaptation and vulnerability. Contribution of Working Group II to the Sixth Assessment Report*. Cambridge University Press. https://doi.org/10.1017/9781009325844

Kaiser, H. F. (1958). The varimax criterion for analytic rotation in factor analysis. *Psychometrika, 23*(3), 187–200.

Kendall, M. G. (1975). *Rank correlation methods* (4th ed.). Charles Griffin.

Lovelace, R., Nowosad, J., & Muenchow, J. (2019). *Geocomputation with R*. CRC Press. https://geocompr.robinlovelace.net

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, 1*, 281–297.

Mann, H. B. (1945). Nonparametric tests against trend. *Econometrica, 13*(3), 245–259. https://doi.org/10.2307/1907187

McDonald, R. P. (1999). *Test theory: A unified treatment*. Lawrence Erlbaum Associates.

OCHA. (2020). *Humanitarian data exchange: Standards and methodology*. United Nations Office for the Coordination of Humanitarian Affairs. https://data.humdata.org

Pearson, E. S. (2018). Simple features for R: Standardized support for spatial vector data. *The R Journal, 10*(1), 439–446. https://doi.org/10.32614/RJ-2018-009

R Core Team. (2024). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. https://www.R-project.org/

Revelle, W. (2024). *psych: Procedures for psychological, psychometric, and personality research*. Northwestern University. https://CRAN.R-project.org/package=psych

Saaty, T. L. (1980). *The analytic hierarchy process*. McGraw-Hill.

Sen, P. K. (1968). Estimates of the regression coefficient based on Kendall's tau. *Journal of the American Statistical Association, 63*(324), 1379–1389.

Shapiro, S. S., & Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). *Biometrika, 52*(3–4), 591–611.

Tukey, J. W. (1977). *Exploratory data analysis*. Addison-Wesley.

UNDRR. (2015). *Marco de Sendai para la reducción del riesgo de desastres 2015–2030*. Naciones Unidas. https://www.undrr.org/publication/sendai-framework-disaster-risk-reduction-2015-2030

UNDRR. (2022). *Global assessment report on disaster risk reduction 2022: Our world at risk: Transforming governance for a resilient future*. United Nations. https://www.undrr.org/gar2022

UNDRR. (2023). *Terminology on disaster risk reduction*. https://www.undrr.org/terminology

van Buuren, S. (2018). *Flexible imputation of missing data* (2nd ed.). CRC Press. https://flexibleimputation.org

Ward, J. H. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association, 58*(301), 236–244.

Wickham, H. (2016). *ggplot2: Elegant graphics for data analysis* (2nd ed.). Springer. https://ggplot2-book.org

Wickham, H., Çetinkaya-Rundel, M., & Grolemund, G. (2023). *R for data science* (2nd ed.). O'Reilly. https://r4ds.hadley.nz

Wilks, D. S. (2019). *Statistical methods in the atmospheric sciences* (4th ed.). Elsevier.

WorldRiskReport. (2023). *WorldRiskIndex 2023*. Bündnis Entwicklung Hilft & Ruhr University Bochum. https://weltrisikobericht.de/english

Xie, Y., Allaire, J. J., & Grolemund, G. (2018). *R Markdown: The definitive guide*. CRC Press. https://bookdown.org/yihui/rmarkdown/

Yeo, I. K., & Johnson, R. A. (2000). A new family of power transformations to improve normality or symmetry. *Biometrika, 87*(4), 954–959.

Zhu, H. (2024). *kableExtra: Construct complex table with 'kable' and pipe syntax*. https://CRAN.R-project.org/package=kableExtra

APÉNDICE A — TABLA DE PAQUETES DE R POR CAPÍTULO

La siguiente tabla lista todos los paquetes de R utilizados en el libro, el capítulo donde se introducen y su función principal. El script al final permite instalarlos todos en una sola ejecución.

Paquete	Función principal
psych	AFE: fa(), KMO(), alpha(), omega(), fa.parallel()
GPArotation	Rotaciones factoriales ortogonales (varimax) y oblicuas (oblimin)
cluster	K-medoids (pam()), coeficiente de silueta (silhouette())
factoextra	Visualización de clústeres, scree plots, fviz_nbclust()
NbClust	30 índices para determinar el número óptimo de clústeres
performance	Diagnóstico visual de supuestos de regresión (check_model())
lmtest	Prueba de Breusch-Pagan para heterocedasticidad de residuos
sandwich	Errores estándar robustos (Huber-White, HC3)
broom	Resultados de modelos estadísticos en formato tidy (tidy(), glance())
pROC	Curvas ROC, AUC y umbral óptimo para modelos de clasificación
ResourceSelection	Prueba de Hosmer-Lemeshow para bondad de ajuste logístico
mice	Imputación múltiple de valores faltantes (MICE-PMM)
sf	Datos espaciales vectoriales: puntos, líneas, polígonos (Simple Features)
terra	Datos ráster: DEM, precipitación, cobertura del suelo
tmap	Mapas temáticos estáticos e interactivos con sintaxis tipo ggplot2
spatstat	Análisis de patrones espaciales de puntos, densidad de kernel
tsibble	Series temporales en formato tidy con índice temporal explícito
feasts	Descomposición STL, ACF, PACF, análisis de estacionalidad

fable	Modelos de pronóstico: ARIMA, ETS, TSLM con interfaz tidyverse
Kendall	Prueba de Mann-Kendall para tendencias monotónicas
trend	Pendiente de Sen y otras pruebas de tendencia no paramétricas
rmarkdown	Reportes reproducibles: Word, PDF, HTML, presentaciones
quarto	Sistema de publicación científica de nueva generación (Posit PBC)
knitr	Motor de compilación de R Markdown; función kable() para tablas
kableExtra	Extensiones de kable() para tablas HTML y LaTeX con formato
flextable	Tablas con formato profesional para Word, PDF y HTML
officer	Control avanzado de documentos Word y PowerPoint desde R
gt	Gramática de tablas: tablas HTML ricas con semáforos e iconos
gtExtras	Extensiones de gt: sparklines, iconos de Font Awesome, color rows
shiny	Dashboards e interfaces web interactivas sin necesidad de JavaScript
shinydashboard	Layouts tipo dashboard para aplicaciones Shiny
leaflet	Mapas interactivos HTML basados en la librería Leaflet.js
DT	Tablas interactivas en aplicaciones Shiny y documentos HTML
plotly	Gráficos interactivos HTML basados en Plotly.js
purrr	Programación funcional: map(), walk(), map_dfr()
tidyverse	Ecosistema integrado: ggplot2, dplyr, tidyr, readr, lubridate...
scales	Formateo de ejes: escalas de millones, porcentajes, comas
ggrepel	Etiquetas en gráficos ggplot2 sin solapamiento
patchwork	Combinar múltiples gráficos ggplot2 en un panel compuesto

```
# Instalar todos los paquetes del libro en una sola ejecución
paquetes_vol2 <- c(
  # Cap. VI - Análisis multivariados
  "psych","GPArotation","cluster","factoextra","NbClust",
  "performance","lmtest","sandwich","broom","pROC","ResourceSelection",
  # Cap. VII - Índices
  "mice",
  # Cap. VIII - Espacial y temporal
```

```
  "sf","terra","tmap","spatstat","tsibble","feasts","fable",
  "Kendall","trend",
  # Cap. IX - Reportes
  "rmarkdown","quarto","knitr","kableExtra","flextable","officer",
  # Cap. X - Comunicación
  "gt","gtExtras","shiny","shinydashboard","leaflet","DT","plotly",
  # Auxiliares
  "purrr","scales","ggrepel","patchwork","tidyverse"
)

nuevos <- paquetes_vol2[!paquetes_vol2 %in% installed.packages()[,"Package"]]
if (length(nuevos) > 0) {
  install.packages(nuevos)
  cat(length(nuevos), "paquetes instalados correctamente.\n")
} else {
  cat("Todos los paquetes ya están instalados.\n")
}
```

APÉNDICE B — ÁRBOL DE DECISIÓN ESTADÍSTICA

El siguiente árbol guía la selección de la prueba o técnica estadística apropiada según el objetivo del análisis, el tipo de variable dependiente, el número de grupos y el cumplimiento de los supuestos paramétricos. Debe usarse como guía orientativa y siempre verificar los supuestos sobre los datos reales.

Pregunta / Condición	Si SÍ →	Si NO →
¿Variable dependiente es continua?	→ Rama continua (ver abajo)	→ Variable categórica: Chi-Cuadrado o Fisher
¿Cuántos grupos compara?	→ 2 grupos: pruebas bilaterales	→ 3+ grupos: ANOVA o Kruskal-Wallis
¿Muestras son independientes?	→ T-test / Mann-Whitney	→ Pareadas: T pareada / Wilcoxon pareado
¿Datos son normales? (Shapiro, Q-Q)	→ T de Student (Welch)	→ Mann-Whitney U (no paramétrico)
¿Homocedasticidad? (Levene)	→ T clásica (var.equal=TRUE)	→ T de Welch ← siempre recomendada
3+ grupos: ¿normalidad + homocedasticidad?	→ ANOVA + Post-Hoc Tukey	→ Kruskal-Wallis + Dunn (Bonferroni)
3+ grupos + heterocedasticidad?	→ ANOVA Welch + Games-Howell	→ Kruskal-Wallis (no paramétrico)
¿Relación entre 2 variables continuas?	→ r Pearson (si normal)	→ rho Spearman (siempre válido)
¿Predecir variable continua?	→ Regresión lineal múltiple	→ (no aplica a este árbol)
¿Predecir variable binaria (0/1)?	→ Regresión logística	→ (no aplica a este árbol)
¿Asociación entre 2 categóricas?	→ Chi-Cuadrado (freq. esp. ≥ 5)	→ Prueba exacta de Fisher
¿Reducir dimensionalidad de indicadores?	→ Análisis Factorial Exploratorio	→ (no aplica)
¿Clasificar unidades en grupos?	→ Análisis de clústeres (K-means)	→ (no aplica)
¿Tendencia en serie temporal?	→ Mann-Kendall + Sen slope	→ (no aplica)
¿Pronóstico de serie temporal?	→ ARIMA / ETS con fable	→ (no aplica)
¿Construir índice de vulnerabilidad?	→ AFE + normalización + AHP	→ (no aplica)

Importante

Este árbol es orientativo. La elección final siempre debe basarse en la verificación de supuestos sobre los datos reales, no en la aplicación mecánica de reglas. Para datos de desastres con distribuciones muy asimétricas, las opciones no paramétricas son frecuentemente la mejor opción, aunque la muestra sea grande.

GLOSARIO

Término	Definición
AFE	Análisis Factorial Exploratorio. Técnica multivariada que identifica factores latentes (dimensiones no directamente observables) que explican las correlaciones entre variables observadas.
AHP	Analytic Hierarchy Process. Método de toma de decisiones multicriterio de Saaty (1980) que asigna pesos a componentes mediante comparaciones pareadas de expertos. La Razón de Consistencia (RC) debe ser < 0.10.
AIC / AICc / BIC	Criterios de información (Akaike, Akaike corregido, Bayesiano) para comparar modelos estadísticos. Menor valor = mejor ajuste penalizado por complejidad del modelo.
ARIMA	AutoRegressive Integrated Moving Average. Familia de modelos para análisis y pronóstico de series temporales univariadas. Notación: ARIMA(p,d,q)(P,D,Q)[m].
Amenaza	Proceso, fenómeno o actividad humana que puede ocasionar pérdida de vidas, lesiones, daños a bienes y al medio ambiente (UNDRR, 2015).
AUC	Area Under the ROC Curve. Cuantifica la capacidad discriminante de un modelo de clasificación: 0.5=azar; >0.7=aceptable; >0.8=bueno; >0.9=excelente.
Bootstrapping	Técnica de remuestreo que estima la distribución de un estadístico generando múltiples muestras con reemplazo del dataset original. Útil cuando no se pueden asumir distribuciones paramétricas.
Caché	En R Markdown/Quarto, almacenamiento de resultados de un chunk para evitar reejecutarlo cuando no han cambiado los datos. Se activa con la opción cache=TRUE.
Chi-Cuadrado (χ^2)	Prueba que evalúa la independencia entre dos variables categóricas comparando frecuencias observadas y esperadas. Supuesto crítico: frecuencia esperada ≥ 5 en todas las celdas.
Chunk	Bloque de código R dentro de un documento R Markdown o Quarto, delimitado por ```{r} y ```. Se ejecuta al compilar y sus resultados se insertan en el documento.
Coeficiente de silueta	Medida de la calidad del agrupamiento en análisis de clústeres. Rango [-1, 1]. Valores cercanos a 1 indican que la observación está bien asignada a su clúster.
CORDEX	Coordinated Regional Climate Downscaling Experiment. Conjunto de proyecciones climáticas regionales de alta resolución desarrollado bajo el auspicio del WCRP.
d de Cohen	Tamaño del efecto para comparación de dos medias. Convenciones de Cohen: pequeño=0.20; mediano=0.50; grande=0.80.
DEFF	Design Effect o efecto diseño. Factor por el que se multiplica la varianza muestral al usar muestreo por conglomerados vs. aleatorio simple. Típicamente entre 1.5 y 2.5 en evaluaciones humanitarias.
Diferencias en diferencias (DiD)	Diseño cuasiexperimental que controla los efectos de confusión comparando la diferencia de diferencias entre grupos tratados y controles antes y después de una intervención.
EM-DAT	Emergency Events Database. Base de datos global sobre desastres gestionada por el CRED (Université Catholique de Louvain, Bélgica). Acceso en emdat.be.
Exposición	Situación en que se encuentran las personas, los medios de subsistencia, los bienes o el medio ambiente que pueden verse afectados negativamente por un evento peligroso (UNDRR, 2015).

ETS	Error, Trend, Seasonality. Familia de modelos de suavizamiento exponencial para pronóstico de series temporales, implementados en el paquete fable de R.
ggplot2	Paquete de R que implementa la Gramática de los Gráficos de Wilkinson. Parte del tidyverse. Construye gráficos como composición de capas de datos, estéticas y geometrías.
IDH	Índice de Desarrollo Humano. Indicador compuesto publicado anualmente por el PNUD que agrega esperanza de vida, nivel educativo e ingreso per cápita en una escala de 0 a 1.
KMO	Kaiser-Meyer-Olkin. Índice que mide la adecuación de los datos para el análisis factorial. Valores ≥ 0.70 son aceptables; ≥ 0.80 son buenos.
IRD	Índice de Riesgo de Desastres. Indicador compuesto que agrega las dimensiones de amenaza, exposición, vulnerabilidad y capacidad según el Marco de Sendai.
LOESS	Locally Estimated Scatterplot Smoothing. Técnica no paramétrica de suavizamiento de series que ajusta polinomios locales. Implementada en ggplot2 con geom_smooth(method='loess').
Mann-Kendall	Prueba no paramétrica para detectar tendencias monotónicas en series temporales. No asume normalidad ni linealidad. El estadístico tau indica la dirección y magnitud.
Marco de Sendai	Marco para la Reducción del Riesgo de Desastres 2015-2030. Acuerdo global de los Estados Miembros de la ONU adoptado en Sendai, Japón, en marzo de 2015.
MCAR/MAR/MNAR	Mecanismos de datos faltantes: Missing Completely At Random (aleatorio puro), Missing At Random (aleatorio dado otras variables observadas), Missing Not At Random (sistemático). Solo MCAR y MAR admiten imputación válida.
Odds Ratio (OR)	Razón de probabilidades en regresión logística. OR > 1 indica mayor probabilidad del evento; OR < 1 indica factor protector. Se obtiene exponenciando los coeficientes.
Omega de McDonald	Coeficiente de confiabilidad más robusto que el Alfa de Cronbach. omega_h evalúa la consistencia del factor general; omega_t la consistencia total.
Outlier	Valor atípico que se aleja notablemente del patrón del resto de los datos. En GRD frecuentemente son eventos reales de alta magnitud y no deben eliminarse automáticamente.
RMSEA	Root Mean Square Error of Approximation. Índice de ajuste en AFE. Valores < 0.05 = buen ajuste; 0.05-0.08 = ajuste razonable; > 0.10 = ajuste pobre.
Resiliencia	Capacidad de un sistema, comunidad o sociedad para resistir, absorber, adaptarse, transformarse y recuperarse de los efectos de un peligro de manera oportuna y eficaz (UNDRR, 2015).
SAT	Sistema de Alerta Temprana. Sistema integrado que combina monitoreo, comunicación, preparación y respuesta para reducir el impacto de amenazas inminentes sobre la población.
Sen's slope	Estimador no paramétrico de la pendiente de una tendencia lineal propuesto por Sen (1968). Más robusto ante outliers que la regresión por mínimos cuadrados.
sf (Simple Features)	Paquete de R que implementa el estándar OGC/ISO para datos espaciales vectoriales, integrándolos con el flujo de trabajo del tidyverse como data frames con geometría.
Shapiro-Wilk	Prueba de normalidad más potente para n ≤ 5.000. Estadístico W cercano a 1 indica normalidad. Recomendada como primera prueba de normalidad.
STL	Seasonal and Trend decomposition using Loess. Método de descomposición de series temporales en componentes de tendencia, estacionalidad y residuo. Robusto ante outliers.
tmap	Paquete de R para la creación de mapas temáticos con sintaxis similar a ggplot2. Soporta mapas estáticos y mapas interactivos HTML con la misma sintaxis.

Teorema Central del Límite	La distribución de las medias muestrales converge a la distribución normal cuando $n \to \infty$, independientemente de la distribución original. Justifica el uso de pruebas paramétricas para $n \geq 30$.
tidyverse	Colección de paquetes de R con filosofía de datos ordenados (tidy data) y sintaxis consistente. Incluye ggplot2, dplyr, tidyr, readr, tibble, stringr, forcats, lubridate y purrr.
UNDRR	United Nations Office for Disaster Risk Reduction. Secretaría de las Naciones Unidas responsable de coordinar la implementación del Marco de Sendai. Anteriormente denominada UNISDR.
VIF	Variance Inflation Factor. Medida de multicolinealidad en regresión múltiple. VIF < 5: sin problema; 5-10: moderado; > 10: grave. Se calcula con car::vif().
Vulnerabilidad	Condiciones determinadas por factores físicos, sociales, económicos y ambientales que aumentan la susceptibilidad de una comunidad a los impactos de los peligros (UNDRR, 2015).
V de Cramér	Tamaño del efecto para la asociación Chi-Cuadrado entre variables categóricas. Rango: 0 (sin asociación) a 1 (asociación perfecta). Independiente del tamaño de la tabla.

ESTADÍSTICA CON RStudio PARA LA GESTIÓN DE RIESGOS DE DESASTRES

Este libro proporciona herramientas estadísticas modernas y reproducibles con R y RStudio para analizar, modelar y comunicar información clave en la gestión del riesgo de desastres.

A través de ejemplos prácticos, código listo para usar y casos reales, aprenderás a transformar datos en conocimiento para la toma de decisiones basadas en evidencia.

Una obra pensada para profesionales y técnicos que trabajan por comunidades más seguras y resilientes.

CIDPROS
Centro de innovación y desarrollo profesional

ISBN: 978-9907-9562-4-5
9 789907 956245