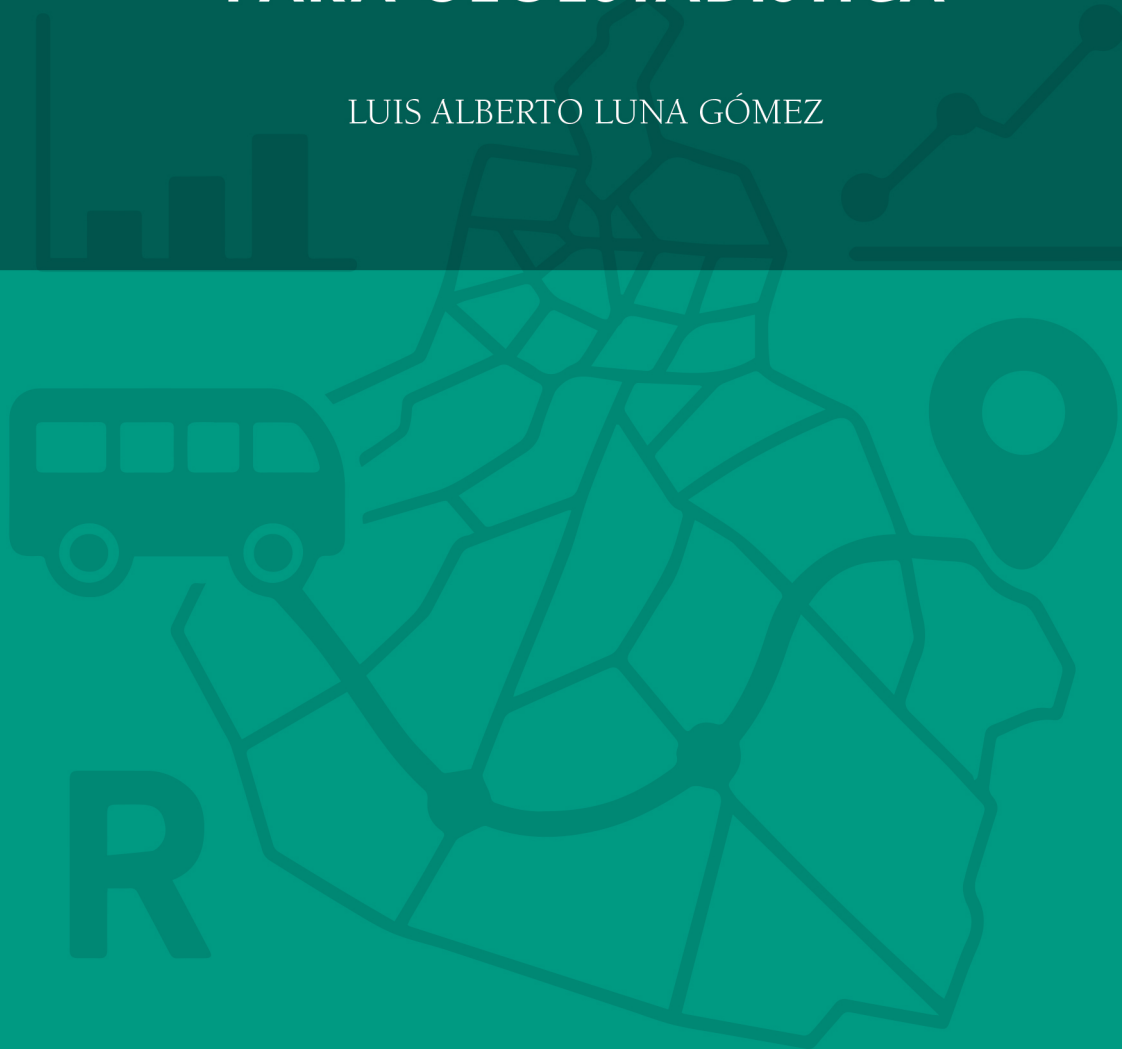


R **PARA GEOESTADÍSTICA**

LUIS ALBERTO LUNA GÓMEZ





UNIVERSIDAD AUTÓNOMA METROPOLITANA

DR. GUSTAVO PACHECO LÓPEZ
Rector General

DRA. ESTHELA IRENE SOTELO NÚÑEZ
Secretaria General

DR. ALFONSO MAURICIO SALES CRUZ
Rector de la Unidad Cuajimalpa

DRA. MARÍA DE LOURDES DELGADO NÚÑEZ
Secretaria de la Unidad Cuajimalpa

DRA. ESTHER MORALES FRANCO
Directora de la División
de Ciencias Sociales y Humanidades

DR. OSCAR FLORES JÁUREGUI
Secretario Académica DCSH

MTRO. LUIS EDUARDO HERNÁNDEZ HUERTA
Jefe de Difusión y Publicaciones DCSH

R
PARA GEOESTADÍSTICA

Catalogación realizada en la Biblioteca Dr. Miguel León Portilla

QA76.73.R3 Luna Gómez, Luis Alberto
L86
2025

R para Geoestadística [recurso electrónico] / Luis Alberto Luna Gómez -- Ciudad de México : UAM, Unidad Cuajimalpa, División de Ciencias Sociales y Humanidades, 2025.

Datos electrónicos (1 archivo pdf : 12 MB) . -- (Divulgación)

ISBN: 978-607-28-3643-3 (volumen)

ISBN: 978-607-28-2478-2 (colección)

1. R (Lenguaje de programación para computadoras) -- Manuales. 2. Geología -- Métodos estadísticos. 3. Ciencias sociales -- Estadísticas -- Investigaciones.

Clasificación Dewey: 005.133 L961 2025

R **PARA GEOESTADÍSTICA**

Luis Alberto Luna Gómez



1a. edición: 2026

© Luis Alberto Luna Gómez

D. R. © 2026, Universidad Autónoma Metropolitana
Prolongación Canal de Miramontes 3855
Rancho Los Colorines, Alcaldía Tlalpan
14386, Ciudad de México
Unidad Cuajimalpa/División de Ciencias Sociales y Humanidades
Avenida Vasco de Quiroga 4871
Col. Santa Fe Cuajimalpa
Alcaldía de Cuajimalpa de Morelos, 05348, Ciudad de México
www.dcsch.cua.uam.mx

Reservados todos los derechos.

Ninguna parte de esta obra puede ser reproducida o transmitida mediante ningún sistema o método electrónico o mecánico sin el consentimiento por escrito de los autores.
Los argumentos, ideas y opiniones presentados en cada uno de los capítulos que integran esta obra, son responsabilidad de cada uno de los autores.

ISBN: 978-607-28-3643-3 (volumen)
ISBN: 978-607-28-2478-2 (colección)

Corrección ortográfica y de estilo: Adriana Guerrero Tinoco
Diseño de portada e interiores y formación: Aarón González Cabrera
Elaborado por Cromberger Editores e Impresores

Hecho en México
Made in Mexico

ÍNDICE

Prefacio	15
Introducción	17
¿Qué hacer con la geoestadística?	17
Notas preliminares.	21
Capítulo 1. Caracterización de R	25
Capítulo 2 Elaboración y gestión de bancos de datos	57
Capítulo 3 Diseño de gráficos	83
Capítulo 4 Análisis de datos	101
Capítulo 5 Repaso y refuerzo para ordenar los datos.	131
Capítulo 6 Ejercicios de autoevaluación y consolidación	159

Capítulo 7	
Creación de mapas.	161
Capítulo 8	
Autoevaluación para generar un mapa.	181
Capítulo 9	
Bibliografía “R”ecomendada	185
Referencias	191
Anexos.	195

ÍNDICE DE FIGURAS

Figura 1. Neo descifra el código (imagen del filme <i>Matrix</i>)	18
Figura 2. Plataforma <i>RStudio</i>	28
Figura 3. Abrir <i>RStudio</i>	29
Figura 4. Dimensión de los procesos cognitivos	30
Figura 5. Representación de los datos	85
Figura 6. Diagrama de dispersión entre el costo del colectivo y costo de taxi registrada en la EOD, 2007.	88
Figura 7. Diagrama de dispersión del Índice de Rezago Social en la ZMVT, 2005-2010.	89
Figura 8. Diagrama de dispersión del Índice de Rezago Social en el Estado de México, 2005	90
Figura 9. Hora final y de inicio en viajes, EOD 2007	90
Figura 10. Gráfico de barras con el salario de los jugadores	91
Figura 11. Red.	92

Figura 12. Pirámide poblacional de los distritos que reporta la EOD 2017	96
Figura 13. Mapa conceptual	98
Figura 14. Simulación de la distribución normal	106
Figura 15. Probabilidad acumulada bajo la normal	108
Figura 16. Histograma de 500 valores al azar de una distribución normal	109
Figura 17. Distribución de viajes por motivo de trabajo en la ZMVM, 2007	118
Figura 18. Nube de puntos de la correlación anterior (0.7937)	119
Figura 19. Atitalaquia. Red de intensidad de población emigrante a los estados y EUA, 2010	125
Figura 20. Tula. Red de intensidad de población emigrante a los estados y EUA, 2010	125
Figura 21. Flujos computacionales	132
Figura 22. Ejecución de funciones	133
Figura 23. Esquema de trabajo	135
Figura 24. Gráfica de barras sin etiquetas en el eje x	136
Figura 25. Gráfica de barras con sus etiquetas en el eje x	137
Figura 26. Modelo secuencial de gráfica de barras	137
Figura 27. Visualización del vector	139
Figura 28. <i>Plot xy</i>	139
Figura 29. Distribución de vector	140

Figura 30. Perspectiva de elevación	142
Figura 31. Contorno de elevación	143
Figura 32. Imagen en tres dimensiones de elevación	143
Figura 33. Correlación de viajes por motivo de trabajo y regreso a casa en la EOD, 2007	144
Figura 34. Correlación entre viajes por motivo de trabajo y viajes por motivo de compras, EOD-2007	145
Figura 35. Variables contenidas en un <i>data.frame</i>	148
Figura 36. Distritos electorales nacionales, 2005	166
Figura 37. Sectores policiacos por Alcaldía en la CDMX, 2008	167
Figura 38. Ciudad de México: hospitales y red del metro	168
Figura 39. Ciudad de México: densidad poblacional en cada estación del metro, 1997	169
Figura 40. ZMVM: Construcciones, 2021.	172
Figura 41. ZMVM: Manzanas, 2021	173
Figura 42. ZMVM: Índice de Desarrollo Humano, 2000; Densidad en red del metro, 1997.	175
Figura 43. Ciudad de México, avenidas principales	177
Figura 44. Ciudad de México, avenidas primarias, secundarias y terciarias.	179

LISTA DE TABLAS

Tabla 1. Algunos comandos básicos para el uso del <i>software</i>	31
Tabla 1. Algunos comandos básicos para el uso del <i>software</i> (continuación)	32
Tabla 2. Objetos <i>R</i>	34
Tabla 3. Resumen. Tipología de objetos en <i>R</i>	53
Tabla 3. Resumen. Tipología de objetos en <i>R</i> (continuación)	54
Tabla 4. Gestión. Acceso a datos dentro de una estructura de datos . . .	54
Tabla 5. Índice de Rezago Social. Estado de México, 2000-2015	65
Tabla 5. Índice de Rezago Social. Estado de México, 2000-2015 (continuación).	66
Tabla 5. Índice de Rezago Social. Estado de México, 2000-2015 (continuación).	68
Tabla 6. Argumento y descripción para la selección de un conjunto de datos muestra	70
Tabla 7. Argumento y descripción para la selección de <i>tapply</i>	71

Tabla 8. Tipología de objetos en <i>R</i>	76
Tabla 9. Tipos de gráficos	83
Tabla 9. Tipos de gráficos (continuación)	84
Tabla 10. Funciones gráficas de alto nivel	85
Tabla 11. Funciones gráficas de bajo nivel	86
Tabla 12. Aplicación de test por tipo de variable	102
Tabla 13. Funciones para observar la tendencia de las observaciones	105
Tabla 14. Errores y conclusiones correctas en las pruebas de hipótesis	112
Tabla 15. Argumentos de la función <i>t.test</i>	113
Tabla 16. Argumentos de la función <i>nnet</i>	124
Tabla 17. Atitalaquia-Tula: población emigrante a los estados y EUA, 2010	126
Tabla 18. Comandos más comunes para instalación de complementos .	146
Tabla 19. Elementos para compilar un paquete	147
Tabla 20. Funciones para diseñar un mapa temático	165
Tabla 21. Función para comparar mapas temáticos con facetas	166
Tabla 22. Sintaxis para elaborar un mapa.	171

PREFACIO

Este libro es resultado de mi experiencia en la aplicación de recursos técnicos utilizados en diversas investigaciones de ciencias sociales para el conocimiento de la realidad. Actores sociales, tiempo y espacio son tres formas cuyas dimensiones en intensidad y magnitud se resisten a ser comprendidas. El científico social se vuelca con las herramientas que le permiten jerarquizar y enunciar su problema de investigación.

La observación es y seguirá siendo un arma muy poderosa. La perseverancia y obstinación que se tiene del objeto de estudio develarán, paso a paso, el pasado, presente y escenarios futuros. Una metáfora pertinente al respecto es el cuadro de Remedios Varo, *Mujer saliendo del psicoanalista*, con aquellas entrañas como diferentes capas del objeto que se develan en la práctica consciente y de autoconciencia.

He sostenido en mis publicaciones que no existe una pugna entre lo que se ha denominado métodos cualitativos y cuantitativos, antes bien, se tiene el deber de resolver el enigma y la mística que plantea el objeto de investigación, por lo cual, es fundamental la búsqueda en los recursos que le proporcionan la probabilidad estadística, pluralidad de sus actores y la mirada de las ciencias sociales, de la razón.

Grandes científicos asisten la pertinencia epistemológica del proceso. Algunos realizaron investigaciones en América Latina sobre el carácter del campesino mexicano (Fromm y Maccoby, 2007). Asimismo, Pierre Bourdieu (2004), en su libro *El baile de los solteros*, realiza una observación minuciosa en una comunidad de Francia, donde combina análisis estadístico a través de un cuestionario, entrevistas y observación participante. También está Díaz-Guerrero (2014), con su libro *Psicología del mexicano. Descubrimiento de la etnopsicología*.

Las anteriores investigaciones coinciden en que no existen métodos diferenciados, sino posturas epistemológicas dedicadas a la aproximación del fenómeno de estudio. Los senderos del conocimiento científico son complejos, recubiertos de prejuicios y formulaciones superficiales para confundir la conciencia. Marx (1999) nos advierte en su libro *El capital*, que para internarnos en la ciencia no hay caminos establecidos, quien aspire ascender por sus reveladoras cumbres debe prepararse a escalar la montaña abriéndose paso por veredas sinuosas.

Los positivistas ven los fenómenos sociales como hechos o elementos externos accesibles a través de los sentidos, mientras que el fenomenólogo desea entenderlos desde su perspectiva, comprensión, desde los motivos y creencias que no se develan a primera vista.

En cualquier caso, la hiper información con la que se cuenta actualmente nos conduce a establecer criterios de clasificación, codificación, interpretación y asociaciones; a través de un ordenador, con el *software R* podemos apoyarnos para registrar dichas prácticas.

INTRODUCCIÓN

¿QUÉ HACER CON LA GEOESTADÍSTICA?

Frente a una base de datos con muchas variables, columnas, registros y renglones, se manifiesta una lluvia de números y palabras que contienen información, pero ¿cómo generar conocimiento? ¿Qué podemos hacer al respecto?

Recordemos el filme *Matrix* (figura 1), la escena donde Neo, el protagonista, ve un código compuesto de números y al colocarse las gafas logra sintetizarlo, resumirlo. Similar a esta imagen que se acaba de describir es la geoestadística: ayuda a comprimir conjuntos de datos que parecieran infinitos. A través de métodos de concentración, el estudiante o investigador pueden enfocar su análisis, utilizar estadísticas acumulativas, crear tablas de resumen, realizar tablas cruzadas, probar diferentes pruebas de inferencia.

Los elementos importantes en el análisis de datos son el resumen y las estadísticas descriptivas, estos proporcionan una forma abreviada de describir y condensar sus datos, sin olvidarse de señalar el procedimiento, el método para desarrollar un procedimiento analítico correcto.

En este libro, sin duda, el lector aprenderá a usar geoestadísticas de resumen, a describir sus datos en lugar de simplemente listar los contenidos. También verá la tabulación como método de la creatividad sociológica. Las tablas pueden dividir sus datos en partes manejables que muestran patrones que de otra manera no serían observables. Producir un resumen cartográfico es importante porque una impresión visual puede transmitir a un lector más información que valores numéricos. El mapa también funciona como un libro, hay que aprender a leer su escala, orientación, valores y relacionar sus diferentes símbolos para correlacionar el espacio con los problemas ambientales, la desigualdad

y sistemas de producción agrícola, industrial y el rubro financiero, con sus respectivas maneras de distribución, intercambio y consumo.

Figura 1. Neo descifra el código (imagen del filme *Matrix*)



Fuente: Fierro (2024).

Los estudiantes de ciencias sociales, en su paso por la licenciatura, adquieren mayor fortaleza en la comprensión de lecturas, búsqueda de información bibliográfica, orden de ideas y redacción. A diferencia de otras áreas de la ciencia, la poca habilidad y comprensión para elaborar un banco de datos y dominar su gestión se complica aún más cuando una base de datos está formada por varias tablas que tienen una o más variables que las relacionan.

Éste es el objetivo del libro: encaminar a los alumnos en el aprendizaje de un *software* del que poco se sabe, dado su reducido o nulo uso en las diferentes escuelas y universidades, para dejar de lado ese pretexto para justificar los vacíos metodológicos que se presentan como hoyos negros en la profesionalización de los futuros investigadores.

Por otro lado, la formulación de hipótesis y su contrastación a través de un test o un modelo se erigen en un espeluznante fantasma que apareció en las primaverales investigaciones de la antiguamente denominada física social, provocando miedo de invocar y utilizar la geoestadística, por tal motivo, mediante

el presente libro, pretendo exorcizar las imágenes y permitir que los alumnos aprendan de manera lúdica, con el uso de *R*, *RStudio* y *Rmarkdown*.

La obra busca motivar el conocimiento de la herramienta *R* con base en cinco objetivos fundamentales:

1. Que el alumno aprenda los conceptos básicos de programación para elaborar un vector que incluya funciones, operaciones, conjuntos, listas, apuntes y observaciones.
2. Que el alumno aprenda y desarrolle habilidades lógicas para realizar gestión en un banco de datos y consiga extraer observaciones con ciertas cualidades definidas, subconjuntos de la base y el diseño de algunas fórmulas.
3. Que el alumno obtenga la síntesis de datos por medio de gráficas, cartografía, tablas, frecuencias y medidas de tendencia central. Geoestadística básica para observar distribuciones, concentraciones y dispersiones.
4. Que el alumno aprenda a realizar algunos modelos y test para contrastar hipótesis.
5. Que el alumno aprenda a modelar variables y fenómenos a través del análisis espacial que permiten ciertas funciones del *software R*. Mapear coropletas, categóricos, burbujas, contorno, disimetría, puntos y cartogramas.

El orden del presente libro responde a los objetivos planteados. El capítulo 1 se aboca a la caracterización de *R*, *RStudios* y *Rmarkdown*, conceptos básicos, diseño de vectores, clases de estos, operaciones y gestión de datos con vectores. El capítulo 2 está orientado a la elaboración de bancos de datos, algunas funciones que acompañan esta labor, diseño de fórmulas y extracción de subconjuntos. El apartado 3 muestra cómo realizar una síntesis de datos a través del diseño de gráficos para observar concentraciones y dispersiones. El capítulo 4 presenta el camino para realizar una síntesis y análisis de datos a través de las medidas de tendencia central, distribución de probabilidad, test y modelos geoestadísticos. El bloque 5 y 6 engloba un repaso, refuerzo y práctica de ejercicios de autoevaluación que le permitirán al alumno consolidar los conocimientos adquiridos. El capítulo 7 expone cómo realizar un mapa por medio de *R*. Por otro lado, casi en la recta final, se presenta una autoevaluación que

permitirá afianzar los conocimientos en la creación de mapas en el capítulo 8, mientras que el apartado 9 presenta la bibliografía del *software* que se recomienda para que el alumno continúe con su formación y ahonde en algunos temas del libro. Finalmente, el apartado 10 contiene aplicaciones de *Rmarkdown* para elaborar una presentación y reporte de investigación.

NOTAS PRELIMINARES

Antiguamente, para la recolección de información y contrastación de hipótesis, el investigador contaba sólo con tarjetas de trabajo, fichas bibliográficas, cámara fotográfica, diario de campo, grabadora, calculadora, formulario y pruebas estadísticas, acudía a las bibliotecas, compraba bancos de datos del Instituto Nacional de Estadística y Geografía (INEGI), del Consejo Nacional de Población (CONAPO), acudía a los archivos del Instituto Nacional Indigenista (INI), Archivo General de la Nación, mapotecas, grupos sociales, entrevistas y lugares de los hechos sociales para recoger los datos.

En la actualidad, el investigador tiene el teléfono celular para captar imagen, audio, realizar anotaciones, tener acceso de manera gratuita a los bancos de datos y algunos *software* con licencia y libres que se han desarrollado para dar orden, jerarquizar y analizar la información con el fin de relacionar fenómenos que expliquen la realidad social, ventajas de la sociedad del conocimiento.

En el presente libro analizaremos, derivado de la aplicación de paquetes de cómputo, matrices y tablas de la Encuesta Origen Destino del año 2007 (EOD-2007), levantada por el INEGI a petición de los gobiernos del Estado de México y la Ciudad de México, los cuales conforman la importantísima Zona Metropolitana del Valle de México (ZMVM). Cabe señalar que los microdatos de la EOD 2017 se encuentran disponibles en línea.

Vamos a mostrar un panorama general de los datos de la EOD 2007. En abril de 2012, la Comisión de Derechos Humanos del Distrito Federal (CDHDF) y el Instituto de Políticas para el Transporte y el Desarrollo México (ITDP) firmaron un convenio específico de colaboración para la promoción, difusión e investigación del derecho a la movilidad en el Distrito Federal (Comisión Nacional de los Derechos Humanos [CNDH], 2012).

La cantidad de desplazamientos reportada en la EOD 2007 es de casi 22 millones, de los cuales, 83.5% se realiza al interior de la Ciudad de México y 16.5% proviene del Estado de México, principalmente de Ecatepec, Nezahualcóyotl y Naucalpan.

La Zona Metropolitana del Valle de México (ZMVM) se encontraba conformada por las 16 delegaciones de la Ciudad de México (CDMX), hoy alcaldías, 64 municipios del Estado de México y uno del estado de Hidalgo. El desplazamiento promedio es de una hora con 21 minutos.

La encuesta reporta que casi 45% de los viajes se realiza para regresar al hogar, 25.4% para ir al trabajo, 8.8% para ir a la escuela, casi 5% para ir de compras y un poco más de 5% para llevar y recoger a alguien.

En viajes por medio del transporte público, se reporta que 59% de estos se realizan en microbús, 27% en autobús, 9% en combi y 5% en vagoneta (CNDH, 2012). En la Evaluación Ciudadana de Servicios Públicos, el Metro y Metrobús reportan una mejor calificación, mientras que los taxis y peseros fueron los peor evaluados (Observatorio Ciudadano de la Ciudad de México [OCCM], 2009).

Vuelcos en los cambios de paradigma entre transporte y movilidad

Se decía que el transporte es el movimiento de vehículos, mientras que en la movilidad, el movimiento es de personas. Antes se expresaba en términos de número de viajes y pasajes, ahora está determinado por la posibilidad de relaciones, oportunidades y satisfacción de necesidades. Ayer se estipulaba por la eficacia, rapidez y fluidez de los vehículos, hoy se define por la accesibilidad de las personas a lugares y por la satisfacción de sus necesidades de bienes, productos y servicios. En la actualidad, el movimiento tiene un valor en sí mismo. Las personas eran vistas como un grupo heterogéneo de moléculas que se desplazan, ahora se pone mayor énfasis en la condición de género y edad de los viajantes.

Cabe hacer una pequeña crítica de un proceso que se va complejizando: el de la movilidad y el derecho que todo sujeto tiene a ésta. Las reformas que modifican su paradigma se incorporan al proceso de cambios que se formulan como una ecuación que enaltece las libertades individuales, pero concentra el ingreso y la producción, lo cual, al poco andar, reparte riqueza y miseria

de manera altamente desigual. Asimismo, la movilidad es el fenómeno idóneo para observar la compresión del tiempo y el espacio, el capital aprovecha esta condición para apropiarse de los mercados a través de los flujos y fijos que requieren soportes materiales, infraestructura vial para la distribución de mercancías. Los desplazamientos conllevan la oportunidad de aprovechar la tecnología que proporcionan el GPS, el celular, las aplicaciones y el monitoreo de bicicletas, vehículos y personas con el fin de comprimir el tiempo y el espacio en los viajes.

El análisis de la movilidad muestra las características de las herramientas tecnológicas *R*, *RStudio* y *Rmarkdown*, instrumentos que permitieron observar y explicar los datos. Se presentan los resultados de una prueba de hipótesis espacio-distrital. Se realizan dichos análisis concretos con la realidad empírica porque los manuales de estadística muestran supuestos que están desapegados de la cotidianidad del estudiante de ciencias sociales. Asimismo, cuando se enseña el *software*, los alumnos carecen de los conocimientos para aplicar una hipótesis nula y una hipótesis alterna, por ello, con base en las necesidades docentes, se incorporan estos temas fundamentales de análisis en los que se aplican los modelos para que los alumnos encuentren en el ejemplo el asidero teórico aplicado en la práctica.

Se busca describir las funciones y rutinas del programa estadístico *R*, cuáles son las potencialidades y críticas. Entre los comentarios más recurrentes se encuentra que *R* es un programa libre en el que varias personas pueden desarrollar complementos, pero no están validados por la comunidad científica, a diferencia de los programas con una licencia, incluso en algunas publicaciones se solicita que se encamine la licencia para validar que todos los cálculos realizados en la investigación son confiables, sin embargo, *r-cran* es una organización a la que pocas instituciones están inscritas y muestran su reputación en la investigación, en el caso mexicano, tienen registro el Instituto de Posgraduados (IP) y el Instituto Tecnológico Autónomo de México (ITAM). Por tanto, *R* es un programa confiable y una gran cantidad de población está desarrollando su investigación con dicho *software*.

El análisis geoestadístico, como dimos en llamar un curso, permite ver que la labor técnica tiene una función práctica en la lucha por el poder, como sostienen Michael Hardt y Antonio Negri (2014) en su libro *Imperio*: toda herramienta es un arma, sabiéndola empuñar adecuadamente.

Las conclusiones nos remiten a que en las ciencias sociales el investigador posee mayor fortaleza y confianza en su conocimiento a partir de recordar algunas rutinas y aplicarlas para entender el complicado entorno de los grupos sociales, territorio y tiempo, que son las tres formas fundamentales.

Se confía en que el estudioso de las ciencias sociales encontrará en este libro un buen instrumento de estudio que le permitirá acercarse al análisis geoes-tadístico, apoyándose de los insumos que producen las nuevas tecnologías de la información y comunicación. Invito a todos los nuevos científicos sociales a adentrarse en estos programas dado que muestran la agilidad de representar los datos y dejar de lado esas abultadas presentaciones textuales, donde el interlocutor ha dejado de tomar importancia y confianza. Los nuevos investigadores de ciencia social no pueden perder la oportunidad de presentar sus datos objetivos en estos programas y renunciar a ser comunicadores de café, en su petulancia y supuesto saber, con la falsa confianza de comunicar una información adulterada, tal como realizan los grandes medios de comunicación; las falsas seguridades que se replican en la microescala con los confiados de la protocomunicación.

Finalmente, como dicen los teólogos de la liberación: “Más vale encender una luz que maldecir las tinieblas”. Más nos vale a los científicos sociales incorporar mayor pensamiento y menos especulación a la ciencia y, así, cambiar la realidad social, en lugar de andar especulando realidades artísticas y de café. Más sacrificio, dedicación y perseverancia en el análisis y menos representación en lugar del ser.

CAPÍTULO 1

CARACTERIZACIÓN DE R

El *software R* es una poderosa e innovadora herramienta de programación estadística y gráfica. Se descarga muy fácilmente de la página de internet: cran.r-project.org. Actualmente, está siendo integrada al análisis de datos por parte de los investigadores sociales.

La experiencia y práctica han demostrado idoneidad para el análisis de redes migratorias, movilidad urbana y análisis de datos censales. Asimismo, por más de un año ininterrumpido, con dicha herramienta se imparten cursos sobre problemas actuales para los alumnos de la licenciatura en Políticas Públicas de la Universidad Autónoma Metropolitana, México. El programa es libre, tiene a su disposición un sistema de recursos que lo complementan, hay más de 2,000 complementos disponibles *R* (Teetor, 2011). Compíte con todos los paquetes estadísticos comerciales.

R no es un paquete simple de utilizar, aunque *RStudio* y *Rmarkdown* le han permitido un uso más sencillo. Diversos autores que han realizado manuales y desarrollado la herramienta para su uso, concuerdan en la dificultad inicial para efectuar operaciones sencillas (Teetor, 2011; Adler, 2010; Aho, 2014; Beaujean, 2014; Pathak, 2014; Bilder y Loughin, 2015; Bivand et al., 2013; Gaetan y Guyon, 2010; Arnold y Tilton, 2015; Xie et al., 2018). Las operaciones sencillas parecen fáciles una vez que se sabe cómo realizarlas, no obstante, el cómo puede resultar exasperante.

En algunas ocasiones, por la ausencia de un símbolo en la sintaxis, no se logra la interacción en el proceso de ejecución, *Rmarkdown* emite una leyenda numerando la línea donde se encuentra el problema y, una vez que se han descargado las librerías oportunas, el escritorio funciona de manera inteligente,

por lo que permite completar la sintaxis, como ocurre actualmente cuando se escribe por medio de WhatsApp o un correo de Gmail, se completan las palabras y se corrige la ortografía. Sin embargo, no se debe tener exceso de confianza, porque la programación tiene un orden lógico para que haya un tejido (*knit*) lógico entre cada comando y función que estructuran el resultado del análisis.

Para ejemplificar lo anterior y reiterar que se debe ser muy claro en las órdenes que se ejecutan en el *software*, recurriré a un chiste. Una programadora se levanta un fin de semana, por la mañana, se viste y le dice a su esposo que va a salir a caminar: “¿Se te ofrece algo?”, a lo que el esposo responde que, por favor, compre dos litros de leche y, si hay huevo, le compre cuatro. Pasaron unos minutos y la programadora regresó con cuatro litros de leche, el marido le pregunta: “¿Por qué traes tanta leche y nada para desayunar?”, ella contesta: “Sí había huevo, luego entonces, compre cuatro litros de leche”.

En el año 2009 se estimó que entre uno y dos millones de personas hicieron uso del paquete (Aho, 2014). No se tiene certeza de la cantidad de usuarios debido a que es de acceso libre. Aho (2014) sostiene que el tráfico en internet para temas de *R* supera los debates para otros tipos de *software* estadístico. Aunado a lo anterior, las búsquedas recientes en Google Scholar muestran que el número de citas de *R* y sus complementos en artículos científicos ha ido en aumento desde el año 2005, mientras que para los más reconocidos, como Statistical Analysis Software (SAS) y Statistical Package Social Sciences (SPSS) han ido declinando (Aho, 2014).

El lenguaje *R* se basa en el antiguo idioma *S*, desarrollado en los Laboratorios Bell. Se trata de una estructura perfeccionada inicialmente en el Instituto Tecnológico de Massachusetts (MIT, por sus siglas en inglés). Ihaka y Gentleman utilizaron el nombre de *R* para reconocer la influencia de *S*, debido a que *R* y *S* son caracteres yuxtapuestos en el alfabeto (Adler, 2010).

El paquete *R* no tiene garantías, puesto que es libre. Su marco de derecho de autor es GNU. La Licencia Pública General (PGL, por sus siglas en inglés) permite a los usuarios compartir y cambiar las funciones y complementos de *R*. El INEGI ha diseñado rutinas para conocer los resultados de algunos censos; el Instituto Mora ha publicado un blog para reforzar su docencia para modelar datos y cada vez se suman más universidades que ofrecen cursos al respecto.

Beaujean (2014) reduce el estrés para los inexpertos, al señalar que el programa *R* es de mayor flexibilidad y facilidad que otros programas. Por ejemplo:

al ejecutar un modelo de regresión múltiple o el uso de diferentes conjuntos de predicción, se dispone de la oportunidad de utilizar los primeros resultados para subsecuentes análisis. Por otra parte, si el análisis se deja pendiente por algunos días, en el historial se pueden observar las secuencias utilizadas para obtener los resultados anteriores. Asimismo, como ya se mencionó, el escritorio funciona de manera inteligente.

RStudio

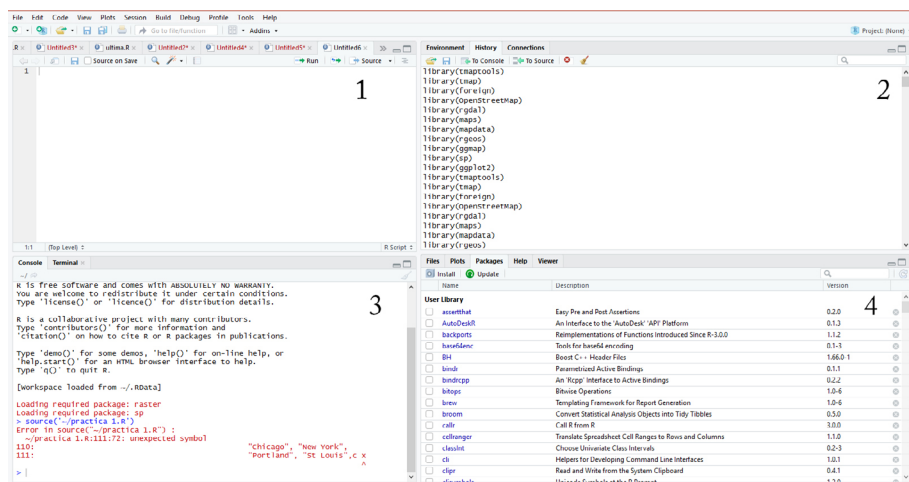
Se puede descargar *RStudio* a través del portal www.rstudio.com, para tener una mascarilla ordenada y amigable. Muestra los coeficientes y resultados en una pantalla. En lugar de abrir muchas ventanas, en la misma, con ayuda de *RStudio*, se puede contar con cuatro secciones: 1) *Note Block*; 2) *Environment-History*; 3) *Output*; 4) *Files-Plots-Packages-Help-Viewer*. Se puede alimentar la sintaxis, copiándola en posteriores análisis, o bien, obtener nuevamente los resultados de ésta (figura 2).

- *Source*. En la esquina superior izquierda de la pantalla se encuentra el editor de texto que permite trabajar con archivos de origen, *Note Block*, *cuadernillo de apuntes*. Se pueden capturar múltiples líneas de código, las cuales se encuentran numeradas, funciona, de cierta manera, como cualquier otro editor de texto, asimismo, como un común denominador en la tecnología, es inteligente, reconoce y resalta varios elementos, por ejemplo: utilizando diferentes colores para caracterizar el tipo de dato, ayuda a encontrar coincidencia de corchete en los guiones y, en caso de existir error, marca un tache (x) en color rojo en la línea donde se encuentra.
- *Console*. En la parte inferior izquierda, debajo de *Source*, tercer cuadrante, es donde se ejecuta todo el trabajo interactivo con *R*. Es idéntica a la consola *R*. Se observa que luego de que *R* evalúa una expresión, se mantiene el símbolo mayor que (>), que significa que el *software* está esperando una orden. Aguarda a que compruebes cosas, escribas o copies en la consola y posteriormente imprimas los códigos.
- *Workspace and history*. En la parte superior derecha se localiza el resumen indispensable de su área de trabajo, donde se pueden inspeccionar las variables creadas en la sesión, así como funciones y valores. También

es posible ver un historial de los comandos, más de 200 líneas de sintaxis capturadas.

- *Files, plots, package, help*. En la esquina inferior derecha se ubica el acceso a varias herramientas.
- Archivos (*Files*). Es posible examinar las carpetas y archivos de la computadora.
- Gráficos (*plots*). Muestra las tablas y gráficos.
- Paquetes (*package*). Conjunto de código autónomo que agrega funcionalidad a R, algo parecido a la forma en que un complemento agrega funcionalidad al paquete Excel de Microsoft.
- Ayuda (*help*). Puede navegar por el sistema de ayuda integrado de R.

Figura 2. Plataforma RStudio



En síntesis, *RStudio* es un editor de código, entorno de desarrollo que permite que R sea fácil y divertido:

1. Resalta con colores diferentes las palabras clave y las variables permitiendo una lectura ágil y sencilla;
2. Cada renglón va numerado, por lo que, si en alguna línea detecta inconsistencias en la sintaxis, cruza el número con una x de color rojo,

- asimismo, no concibe inconsistencia con los paréntesis, puesto que al momento de abrirlo, inmediatamente presenta el cierre de éste;
3. Permite el rápido acceso a *R Help*, con algunas características idóneas para explorar funciones y parámetros de la librería en uso. Cabe señalar que también propicia la accesibilidad para explorar las variables y sus valores.
 4. *RStudio* es compatible con Linux, Windows y Apple iOS.
 5. Disponible de forma gratuita, es un excelente complemento para *R*. Todos los ejemplos que se presentan en forma de imagen se encuentran en dicha plataforma.

Para abrir *RStudio*, dé clic en el ícono *RStudio* en el sistema de menú del ordenador o en el ícono que se encuentra en el escritorio, o bien, en los íconos de inicio (figura 3).

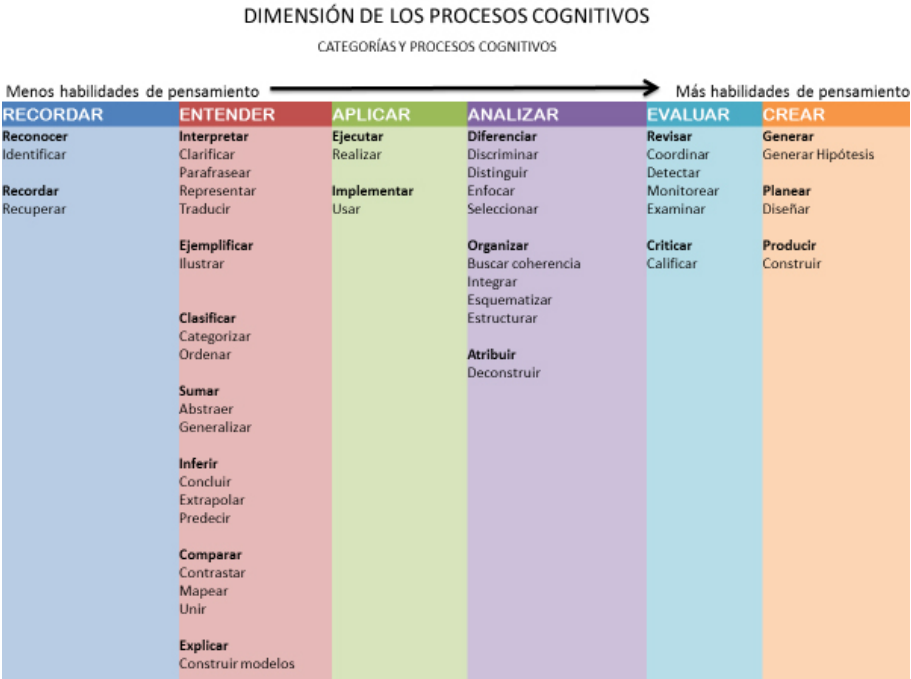
Figura 3. Abrir RStudio



El programa *RStudio* es indispensable para la investigación en ciencias sociales, pone a prueba todos los procesos cognitivos porque hay que recordar la sintaxis básica, entender el procedimiento y la posición de cada uno de sus componentes, aplicar la teoría en modelos, índices y algoritmos hasta analizar, evaluar y crear conjeturas y refutaciones para desarrollar el pensamiento y crear soluciones a problemas concretos de la realidad. Como dijo el compañero Mao Tse-Tung, “conocer la realidad es el primer paso para cambiarla” (figura 4).

El sistema *RStudio* incluye varios componentes. *R* se refiere a un lenguaje, el intérprete de códigos escritos en *R* y un sistema para diseñar gráficos.

Figura 4. Dimensión de los procesos cognitivos



Fuente: elaboración propia con base en Gallardo et al. (2012).

Generar sintaxis es una forma lógica de apropiarnos del conocimiento, teniendo a más habilidades del pensamiento, crear desde las ciencias sociales ecuaciones y algoritmos que nos ayuden a entender la realidad para modificarla.

Iniciar por recordar, no olvidar para crear, como se muestra en el proceso cognitivo de la figura 4.

Conceptos básicos y diseño de vectores

Entender un lenguaje de computación es similar a aprender a hablar una lengua extranjera. Cuando se visita un país extranjero, es posible entender algunas frases, sin realmente concebir cómo se estructura el lenguaje. La metáfora acontece con algunas relaciones de simples elementos de *R*. A continuación, se muestran comandos básicos en la tabla 1.

Tabla 1. Algunos comandos básicos para el uso del software

Concepto	Descripción	Sintaxis
Comentario	Se utiliza para realizar anotaciones que permitan ahorrar tiempo y esfuerzo, porque el registro de los procesos es simultáneo. El símbolo conduce a realizar comentarios sobre las operaciones, hipótesis y avances de investigación, el color que le asigna el programa a estos es el verde claro. #Geoadística en <i>R</i> .	#
Asignar	Es el símbolo <i>menor que</i> , seguido de un guión medio. Asigna un operador, equivale a utilizar igual (=), pero es mejor reservar el signo igual para definir valores en los argumentos. Confíere lo que está a la derecha de la flecha para el objeto a la izquierda de la flecha.	<-
Conjunto	La función de conjunto relaciona los argumentos dentro de una función. Usando <code>c()</code> en conjunto con (<-) asigna la unión de objetos dentro de un nuevo objeto. Por ejemplo, utilizando la numeración de Fibonacci: <code>Fibonacci<-c(1,1,2,3,5,8,13,21,...)</code>	<code>c()</code>

**Tabla 1. Algunos comandos básicos para el uso del software
(continuación)**

Concepto	Descripción	Sintaxis
Ayuda	La función ayuda (<i>help</i>) regresa información sobre una función. Se puede escribir la función o el símbolo de interrogación. Ejemplo: help(summary) ?summary	help() ?...
Ejemplo	Se utiliza para mostrar un ejemplo de la función. example(mean)	example()
Modelo lineal	Modelo estadístico que mide la varianza (r) entre una variable dependiente y una o más variables independientes.	lm()
Base de datos	La función <i>read</i> permite montar una base de datos separada por comas. Modificando la extensión se pueden cargar bases de datos en diferentes formatos, por ejemplo: texto (.txt), Excel (.xlsx), SPSS (.spss), DBF (.dbf). read.txt read.xlsx read.spss read.dbf	read.csv()
Instalar complemento	Función que sirve para instalar librerías que permiten al usuario realizar análisis estadísticos, de redes y cartográficos con mayor desarrollo y precisión.	install. packages()
Correlación	La correlación es el valor r que va de -1 a 1, cuyo valor p , en caso de ser menor a 0.05, permite aceptar la hipótesis alterna y en caso de ser mayor a 0.05, permite no descartar la hipótesis nula.	cor()
Librería	Funciona para abrir la biblioteca de un complemento. Cada que hacemos uso de éste, debemos llamar primero la biblioteca a la cual pertenece.	library()

Concepto	Descripción	Sintaxis
Agrupación	Función que llama y agrupa expresiones (respectivamente).	({
Indexación	Indexa.	[[[
Componente	Variable que pertenece a una base de datos o elemento de un vector, podemos ubicarlo al colocar del lado izquierdo el nombre de dicha base y del lado derecho nos arroja el objeto o la variable contenida.	\$
Exponencial	Eleva el número a la potencia señalada a la derecha del capuchón.	^
Suma y resta	Ambas son operaciones aritméticas.	- +
Operador secuencial	Arroja de manera ordenada un conjunto iniciando por el número de la derecha y terminando por el número de la izquierda, ejemplo: 5:20, nos arroja el conjunto de números que va del 5 hasta el 20: 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20.	:
Operador especial	Operador representado por el símbolo de porcentaje, permite localizar un registro en la gestión de datos, especificando por este operador con características particulares.	%any%
Multiplicar y Dividir	Multiplica y divide. Operaciones aritméticas.	* /
Ordenar	Ordenan y comparan un par de cifras.	< > <= >= == !=
Negación	Negación.	!
Conjuntar	Y	&& &&&
Conjuntar	O	
Dependencia	Dependencia en fórmulas.	~
Nombrar	Nombra lo que está en la derecha.	-> ->>
Nombrar	Nombra lo que está en la izquierda.	<- <<-

Fuente: elaboración propia con base en Teetor (2011) y Cotton (2013). Actualmente, se puede montar una base de datos sin necesidad de un vector, es cuestión de seguir la rutina de Import Dataset localizada en la sección Environment de *RStudio*.

Cuando se precisa comprender un nuevo lenguaje realmente bien, se necesita entender sobre la sintaxis y la gramática: conjugación de verbos, propiedad de los artículos, estructura de las sentencias, entre otros elementos. Es similar con las estructuras de *R*. Se requiere entender cómo programar efectivamente en *R*, por ello, mostramos en este libro sintaxis y gramática fundamental del paquete.

En *R*, cada objeto (tabla 2) tiene un modo que indica cómo se almacena en la memoria: como número, cadena de caracteres, lista y función. A continuación, se muestra el objeto, con su ejemplo y tipo de dato.

Tabla 2. Objetos *R*

Objeto	Ejemplo	Modo
Número	3.1416	Numérico
Vector de números	c(3.1416, 0.752)	Numérico
Cadena de caracteres	"Taberna de Moe"	Carácter
Vector de cadena de caracteres	c("Lili", "Bety", "Boo")	Carácter
Factor	factor(c("MT", "CA", "NX"))	Numérico
Lista	list("Moenia", "Lola", "Miranda")	Lista
Data frame	data.frame(x=5:7,y=c("D","M","D"))	Lista
Función	print	Función

Cuando se ingresa una expresión en el paquete y presionas *CTRL+ENTER*, *R* evaluará la expresión, mostrará los resultados, los imprimirá. A continuación, realizar las siguientes operaciones en *RStudio* y contrastar con los valores [1] expresados.

2+3+4	<i>CTRL+ENTER</i>
[1] 9	
1+2*3	
[1] 7	
(1+2)*3	
[1] 9	
c(1,1,2,3,5,8,13,21)	
[1] 1 1 2 3 5 8 13 21	

Estos son algunos vectores que contienen operaciones, suma y multiplicaciones, el último vector es un conjunto de los primeros ocho números de Fibonacci.

El intérprete interactivo de R imprime el objeto revirando una expresión. Cualquier número que se ingrese en el computador es interpretado como un vector. Un vector es una colección de números ordenados. El [1] significa el primer ítem arrojado en el renglón uno. Cada uno de los casos tiene sólo un elemento en el vector. Se puede construir un vector largo utilizando la función de conjunto, `c(...)`. Por ejemplo, realice la siguiente sintaxis en el *software*.

```
c(1,2,3,4,5)
[1] 1 2 3 4 5
```

Mostraremos un conjunto con los primeros cincuenta números, es un ejemplo de vector que tiene múltiples líneas, con el operador de secuencia se produce un vector integrado por la numeración del 1 al 50. Realice la actividad.

```
1:50
[1] 1 2 3 4 5 6 7 8 9 10 11 12
[13] 13 14 15 16 17 18 19 20 21 22 23 24
[25] 25 26 27 28 29 30 31 32 33 34 35 36
[37] 37 38 39 40 41 42 43 44 45 46 47 48
[49] 49 50
```

Cuando se corre una operación con dos vectores, R unirá los elementos de los dos vectores e imprimirá un vector. Realice las siguientes operaciones y compruebe el resultado [1].

```
c(1,2,3,4)+c(10,20,30,40)
[1] 11 22 33 44
```

```
c(1,2,3,4)*c(10,20,30,40)
[1] 10 40 90 160
```

Los operadores que llevan a cabo todo el trabajo son llamados funciones:

```
exp(1)
[1] 2.718282
cos(3.14)
[1] -0.9999987
log(1)
[1] 0
```

En cada uno de los ejemplos, la función sólo tomó un argumento. Muchas funciones requieren más que un solo argumento. Se pueden especificar los argumentos por el nombre. Mostraremos un ejemplo y le suplicamos al lector acompañar la rutina:

```
log(x=64, base=4)
[1] 3
```

Asimismo, se puede omitir el nombre de las funciones:

```
log(64,4)
[1] 3
```

Al usar *R*, con frecuencia encontrará los seis tipos básicos de vectores:

1. Lista;
2. Matrices;
3. Array;
4. Factores;
5. Data frame, y
6. Fórmulas.

R incluye varias formas diferentes de crear un nuevo vector, la función *c* es de mucha ayuda porque combina argumentos en un solo vector. Un vector es una secuencia que de forma lógica organiza, proporciona, aplica, analiza, evalúa y crea información. Comúnmente, contiene tres elementos principales:

- a) Nombre
- b) Comando, menor que, seguido de guión medio (§)
- c) Información

Replique los siguientes ejemplos, los cuales se ejecutarán después de oprimir las teclas *ctrl+enter*:

- `luna<-1`
- `eclipse<-2`
- `cromosoma_x<-1`

Realizaremos una operación para comprobar el ejercicio con los vectores. Únicamente con símbolos aritméticos.

```
luna*eclipse+cromosoma_x
[1] 3
```

- `if(1>2) "yes" else "no"`
- ```
if(1>2) "yes" else "no"
[1] "no"
```

Es imprescindible reiterar que para imprimir las operaciones, cálculo o contenido de un vector en *RStudio* es necesario escribir el nombre del vector seguido de *ctrl+enter*, por ejemplo, si deseamos imprimir la información que tiene el vector uno, se debe realizar la siguiente rutina.

```
luna →ctrl y enter, al mismo tiempo.
[1] 1
```

## Clases de vectores

Cada tipo de vector viene precedido de un símbolo que anteriormente se conocía como numeral o gato, en la actualidad, se le nombra de manera anglosajona, por el uso de la aplicación Twitter, como *hashtag*, que equivale a decir “etiqueta”. Realice los ejemplos que se expresan a continuación.

### #Clase de vector numérico

```
Aries<-c(1,2,3,4,5)
Aries
[1] 1 2 3 4 5
```

```
Tomate<-c(0.90, 0.60, 0.90)
Tomate
[1] 0.9 0.6 0.9
```

De forma predeterminada, los números en las expresiones *R* se interpretan como números de punto flotante, precisión doble, incluso cuando ingresa números enteros simples. Veamos lo anteriormente señalado con la función *typeof*. Realice los siguientes ejemplos:

```
typeof(1)
[1] "double"
```

Si realmente buscas un entero, se puede anotar la siguiente función.

```
typeof(1:1)
[1] "integer"
typeof(as.integer(1, "integer"))
[1] "integer"
```

### **#Vector carácter**

```
Tauro<-c("este es un objeto")
Tauro
[1] este es un objeto
```

```
Números<-c("uno", "dos", "tres", "cuatro", "cinco")
Números
[1] "uno" "dos" "tres" "cuatro" "cinco"
```

### **#Vector secuencial**

```
seq(from=5, to=25, by=5)
[1] 5 10 15 20 25
```

```
wi<-20:30
wi
[1] 20 21 22 23 24 25 26 27 28 29 30
```

Se puede manipular la longitud de un vector señalando el atributo por medio de la función *length*, como veremos en el siguiente ejemplo, donde se manipula la dimensión del vector *wi* que originalmente comprende once observaciones.

```
length(wi)<-5
wi
[1] 20 21 22 23 24
length(wi)<-10
wi
[1] 20 21 22 23 24 NA NA NA NA NA
```

En todos los ejemplos anteriores hemos examinado estructuras de datos basadas en un solo tipo. En *R* es posible construir conglomerados más complicados utilizando múltiple información. El *software* posee una forma para mezclar objetos diferentes, llamados listas.

Las listas son sutilmente diferentes que en otros lenguajes de cómputo. Pueden contener una selección heterogénea de objetos. Nombrar cada componente en una lista. Los elementos pueden ser referidos por ubicación o nombre.

### **#Vector lista**

```
Capricornio<-list(c(14,15,17,14,16), "este es un objeto", "esta es una lista")
```

```
[[1]]
[1] 14 15 17 14 16
[[2]]
[1] "este es un objeto"
[[3]]
[1] "esta es una lista"
```

```
Mao<-list(thing="anteojos", size="4.0")
Mao
$thing
[1] "anteojos"
$size
[1] "4.0"
```

**#Una lista puede contener otra lista**

```
Tse<-list("esta lista refiere otra lista", Mao)
```

```
Tse
```

```
[[1]]
```

```
[1] "esta lista refiere otra lista"
```

```
[[2]]
```

```
[[2]]$thing
```

```
[1] "anteojos"
```

```
[[2]]$size
```

```
[1] "4.0"
```

```
lula<-list(12,11,10,9,8,7,6,5,4,3,2,1,"libre")
```

```
lula
```

```
[[1]]
```

```
[1] 12
```

```
[[2]]
```

```
[1] 11
```

```
[[3]]
```

```
[1] 10
```

```
[[4]]
```

```
[1] 9
```

```
[[5]]
```

```
[1] 8
```

```
[[6]]
```

```
[1] 7
```

```
[[7]]
```

```
[1] 6
```

```
[[8]]
```

```
[1] 5
```

```
[[9]]
[1] 4
```

```
[[10]]
[1] 3
```

```
[[11]]
[1] 2
```

```
[[12]]
[1] 1
```

```
[[13]]
[1] "libre"
```

## Operaciones

### #Suma de vectores

```
manzana<-3
peras<-4
manzana+peras
[1] 7
```

### #Valores especiales: NA, Inf

```
v<-c(1,2,3)
v
[1] 1 2 3
```

```
length(v)<-4
v
[1] 1 2 3 NA
```

```
2^1024
[1] Inf
```

```
1/0
```

```
[1] Inf
```

### #Coerción

```
x<-c(1,2,3,4,5)
```

```
x
```

```
[1] 1 2 3 4 5
```

```
typeof(x)
```

```
[1] "double"
```

```
class(x)
```

```
[1] "numeric"
```

```
x[2]<-"sombrero"
```

```
x
```

```
[1] "1" "sombrero" "3"
```

```
[4] "4" "5"
```

```
typeof(x)
```

```
[1] "character"
```

```
class(x)
```

```
[1] "character"
```

### #Operadores

```
1+19
```

```
[1] 20
```

```
5*4
```

```
[1] 20
```

R también incluye notación para otras operaciones matemáticas, envolviendo módulos, exponentes y división entera.

```
41%%21
```

```
[1] 20
```

```
20^1
```

```
[1] 20
```

*R* es un lenguaje interpretado. Cuando se ingresan expresiones en la plataforma *RStudio* y se corre el guión, un programa dentro del sistema *R* llama al intérprete y ejecuta los códigos escritos. Unlike C, C++, y Java, no necesita compilar sus programas dentro de un lenguaje. Otros ejemplos de lenguajes interpretados son LISP, Perl y JavaScript.

Todos los programas de *R* son compuestos de una serie de expresiones. Esas expresiones se obtienen de la forma de llamar las funciones. El intérprete de *R* comienza por analizar cada expresión, traducir la succión sintáctica de una manera funcional. Después, *R* sustituye objetos por símbolos. Finalmente, *R* evalúa cada expresión, regresando un objeto. Por expresión compleja se entiende un proceso iterativo, en algunos casos especiales, el paquete no evalúa todos los argumentos de una función.

## Gestión de datos con vectores y matriz

### #Indexar un vector

Para obtener un fragmento de nuestros datos, por ejemplo, cuando tenemos toda la información sobre un tema determinado de la Ciudad de México, pero nos interesa específicamente la Alcaldía Benito Juárez, debemos indexar la información contenida en una base de datos, lo que se consigue a través de la función de corchetes, `[ ]`.

```
v<-100:119
```

```
v
```

```
[1] 100 101 102 103 104 105
```

```
[7] 106 107 108 109 110 111
```

```
[13] 112 113 114 115 116 117
```

```
[19] 118 119
```

```
v[5]
[1] 104
```

```
v[1:5]
[1] 100 101 102 103 104
```

```
v[c(1,6,11,16)]
[1] 100 105 110 115
```

```
l<-list(a=1, b=2, c=3, d=4, e=5, f=6, g=7, h=8, i=9, j=10)
l
```

```
$a
[1] 1
```

```
$b
[1] 2
```

```
$c
[1] 3
```

```
$d
[1] 4
```

```
$e
[1] 5
```

```
$f
[1] 6
```

```
$g
[1] 7
```

```
$h
[1] 8
```



```
$i
[1] 9

$j
[1] 10

l[1:3]

$a
[1] 1

$b
[1] 2

$c
[1] 3
```

### #Matrix

Una matriz es un vector con dos dimensiones, renglones y columnas. Se utiliza para exponer datos de un solo tipo que responden a dos características. Se genera con la función: *matrix*.

Como ejemplo, elaboramos un vector de matriz con el nombre *m*, cuyas características son tres columnas y cuatro filas. Assignaremos a los registros los nombres: “observacion1”, “observacion2”, “observacion3” y “observacion4”, y a las columnas (variables) los nombres “variable1”, “variable2” y “variable3”.

```
m<-matrix(data = c(101:112), nrow = 4, ncol = 3,
dimnames=list(c(“observacion1”, “observacion2”, “observacion3”, “observacion4”), c(“variable1”, “variable2”, “variable3”)))
```

```
m
variable1 variable2 variable3
observación1 101 105 109
observación2 102 106 110
observación3 103 107 111
observación4 104 108 112
```

Asimismo, es posible gestionar una matriz desde otra estructura de datos, usando la función *as.matrix*. Las matrices se implementan como vectores, no como un vector de vectores (o como una lista de vectores). Los subíndices de matriz se utilizan para hacer referencia a elementos y no reflejan la forma en que se almacenan los datos. A diferencia de otras clases, las matrices no tienen un atributo de clase explícito.

Como un juego hemos aprendido a realizar vectores con diferentes tipos de datos, numérico y texto, fundamentalmente. También revisamos cómo solicitar, gestionar información por la posición en que se localizan los registros.

Iniciamos la captura de datos que responde a dos características (columnas y renglones) matrices, el nivel de complejidad aumenta, sin embargo, conservamos el patrón de la estructura con base en un solo vector.

De la matriz *m* vamos a solicitar toda la información que contiene el renglón número tres, *observacion3*. Observar y replicar el ejercicio. Verán que el modo de gestionar información es muy parecido al de buscar elementos de una lista, sin embargo, como los datos en la matriz responden a dos características, debemos señalar primero a qué observación y, posteriormente, a qué número de variable, ambos valores separados por una coma. En el caso de requerir toda la información de un registro, colocamos el número y la coma; por lo contrario, toda la información de una variable, colocamos la coma y el número.

```
m[3,]
[1] 103 106 109 112
```

```
m[3,4]
[1] 112
```

```
m[1:2,1:2]
[,1] [,2]
[1,] 101 104
[2,] 102 105
```

Asimismo, podemos realizar una estructura por medio de la función *array*, cuya extensión de un vector conforma más de dos dimensiones. Los vectores se usan para representar datos multidimensionales.

```
Ya<-array(data=1:24, dim=c(3,4))
```

```
Ya
```

```
 [,1] [,2] [,3] [,4]
[1,] 1 4 7 10
[2,] 2 5 8 11
[3,] 3 6 9 12
```

Vamos a presentar otro ejemplo de *array*, donde incorporamos una dimensión que expresa dos tablas, cada una con tres observaciones y cuatro variables.

```
treseis<-array(c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,
16,17,18,19,20,21,22,23,24), dim = c(3,4,2))
treseis
, , 1
```

```
 [,1] [,2] [,3] [,4]
[1,] 1 4 7 10
[2,] 2 5 8 11
[3,] 3 6 9 12
```

```
, , 2
```

```
 [,1] [,2] [,3] [,4]
[1,] 13 16 19 22
[2,] 14 17 20 23
[3,] 15 18 21 24
```

Al seleccionar un subconjunto, *R* forzará automáticamente el resultado al número más apropiado de dimensiones. Si selecciona un subconjunto de elementos que corresponde a una matriz, *R* devolverá un objeto de matriz; si selecciona un subconjunto que corresponde a un vector, *R* devolverá un objeto vectorial. Para deshabilitar este comportamiento, puede usar la opción *drop = FALSE*. Observe y replique los siguientes ejercicios.

```

luna<-array(data=c(101:124), dim=c(2,3,4))

, , 1

[,1] [,2] [,3]
[1,] 101 103 105
[2,] 102 104 106

, , 2

[,1] [,2] [,3]
[1,] 107 109 111
[2,] 108 110 112

, , 3

[,1] [,2] [,3]
[1,] 113 115 117
[2,] 114 116 118

, , 4

[,1] [,2] [,3]
[1,] 119 121 123
[2,] 120 122 124

class(luna[1,1,])
[1] "integer"
class(luna[1,,])
[1] "matrix"
class(luna[1:2,1:2,1:2])
[1] "array"
class(luna[1,1,1, drop=FALSE])
[1] "array"

```

Al analizar datos, es bastante común encontrar valores categóricos. Por ejemplo, supongamos que tiene un conjunto de observaciones sobre raza. Puede representar la raza como una matriz de caracteres.

```
raza<-c("blanco", "negro", "asiatico", "hispano",
 "nativo", "negro", "asiatico", "hispano")
```

Ésta es una forma perfectamente válida de representar la información, pero puede volverse ineficiente si está trabajando con un nombre con gran cantidad de caracteres o una gran cantidad de observaciones. *R* proporciona una mejor manera de representar valores categóricos mediante el uso de factores. Un factor es una colección ordenada de artículos. Los diferentes valores que puede tomar el factor se llaman niveles.

Recodifiquemos el vector *raza* como un factor:

```
raza<-factor(c("blanco", "negro", "asiatico", "hispano",
 "nativo", "negro", "asiatico", "hispano"))
```

La función *levels* muestra todos los niveles de un factor:

```
levels(raza)
[1] "asiatico" "blanco" "hispano" "nativo"
[5] "negro"
```

La impresión de un factor muestra información ligeramente diferente a la de un vector de caracteres. En particular, observe que las comillas no se muestran y que los niveles se imprimen explícitamente:

```
raza
[1] blanco negro asiatico hispano nativo
[6] negro asiatico hispano
5 Levels: asiatico blanco hispano ... negro
```

En el ejemplo de *raza*, el orden no importa. Sin embargo, a veces contamos con datos de tipo ordinal, donde el orden de los factores es importante para

un problema específico. Por ejemplo, suponga que ha realizado una encuesta de degustación, ofreciendo un producto nuevo y preguntado a los encuestados: ¿Cómo les parece la mezcla de “hotcakes con avena”? Además, suponga que permite que los encuestados brinden las siguientes respuestas: “muy delicioso”, “delicioso”, “desagradable”, “muy desagradable”.

Hay muchas formas de representar esta información en *R*. Podría codificarlas como números enteros (por ejemplo, en una escala de 1 a 4), aunque este enfoque tiene algunos inconvenientes. Implica una relación cuantitativa específica entre valores. Una respuesta numérica también puede calcular estadísticas significativas basadas en las respuestas.

Para evitar estos problemas, use la función `ordered=`, que permite representar la respuesta de esta encuesta. Aquí hay un ejemplo con supuestos registros capturados afuera de dos expendios, un supermercado y un centro comercial:

```
gusto.platillo<-factor(c("muy delicioso", "delicioso",
 "desagradable", "muy desagradable",
 "muy delicioso", "delicioso",
 "desagradable", "muy desagradable"),
 levels = c("muy delicioso", "delicioso",
 "desagradable", "muy desagradable"),
 ordered = TRUE)
gusto.platillo
[1] muy delicioso delicioso
[3] desagradable muy desagradable
[5] muy delicioso delicioso
[7] desagradable muy desagradable
4 Levels: muy delicioso < ... < muy desagradable
```

Es posible crear un objeto de matriz con dimensiones de longitud 1. Sin embargo, al seleccionar subconjuntos, *R* simplifica los objetos devueltos. También es posible reemplazar elementos en un vector o matriz usando la misma notación. Ejemplo de cómo reemplazar valores en el vector *m*. Realice también los ejercicios expuestos a continuación.

```

m[1]<-1000
m
 [,1] [,2] [,3] [,4]
[1,] 1000 104 107 110
[2,] 102 105 108 111
[3,] 103 106 109 112

m[1:2,1:2]<-matrix(c(1001:1004))
m
 [,1] [,2] [,3] [,4]
[1,] 1001 1003 107 110
[2,] 1002 1004 108 111
[3,] 103 106 109 112

```

Asimismo, se puede indexar y ordenar una serie de datos o informaciones de acuerdo con un criterio en común, por nombre, usando la notación de doble corchete al seleccionar un solo elemento. Incluso es posible indexar por nombre parcial usando la opción, *exact* = *FALSE*.

```

despensa<-list(leche="1 galon", mantequilla="1 barra", huevos=12)
despensa$leche
[1] "1 galon"
despensa[["leche"]]
[1] "1 galon"
despensa[["lech"]]
NULL
despensa[["lech", exact=FALSE]]
[1] "1 galon"

```

A veces, un objeto es una lista de listas. También se puede usar la notación de doble corchete para hacer referencia a un elemento en este tipo de estructura de datos. Para hacer esto, usa un vector como argumento. *R* interactúa a través de los elementos en las sublistas de referencia vectorial.

```

frutas<-list(manzanas=8, naranjas=4, platanos=12)
mercado.lista<-list(despensa, frutas)
mercado.lista
[[1]]
[[1]]$leche
[1] "1 galon"
[[1]]$mantequilla
[1] "1 barra"
[[1]]$huevos
[1] 12
[[2]]
[[2]]$manzanas
[1] 8
[[2]]$naranjas
[1] 4
[[2]]$platanos
[1] 12

```

Los estándares para el estilo de código no son lo mismo que la sintaxis, aunque están más o menos relacionados. Por lo general, es aconsejable tener cuidado con el estilo del código para maximizar la legibilidad. Aunque *RStudio* proporcione una consola amigable e inteligente, no olvide lo siguiente, que es muy importante.

### *Sangría*

Sangría con dos espacios. Si el código está dentro de paréntesis, sangría al paréntesis más interno.

### *Espacios*

Use sólo espacios simples. Agregue espacios entre operadores. No agregue espacios entre un nombre de función y la lista de argumentos. Agregue un solo espacio entre los elementos en una lista, después de cada coma.

### *Bloques*

No coloque una abrazadera de apertura sin su corsé de cierre en su propia línea [ } ]. *RStudio* facilita de manera inteligente esta función.

### *Punto y coma*

Omitir punto y coma al final de las líneas cuando son opcionales.



*Nombrar*

Nombra objetos con minúsculas, separados por puntos. Para los nombres de las funciones, escriba el nombre de cada palabra que esté unida, sin espacios. Intente hacer verbos de nombres de función, pero sin nombrar al vector como una función, pues eso crea confusión. No confundir con los nombres de los objetos (tabla 3).

**Tabla 3. Resumen. Tipología de objetos en R**

|            |             |                                                                                                                                                                                                                                                                                          |                                   |
|------------|-------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------|
| Vectores   | entero      | Naturalmente producido a partir de secuencias. Se puede forzar con la función <code>entero</code> .                                                                                                                                                                                      | <code>integer(5)</code>           |
|            | doble       | Se usa para representar números de coma flotante (números con decimales). En la mayoría de las plataformas modernas, esto será de 8 bytes o 64 bits. Por defecto, la mayoría de los valores numéricos se representan como dobles. Se puede forzar con la función <code>double()</code> . | <code>double(5)</code>            |
|            | complejo    | Para usar, se deben incluir tanto las partes reales como imaginarias (incluso si la parte real es 0).                                                                                                                                                                                    | <code>exp(0+1i*pi)</code>         |
|            | carácter    | Una cadena de caracteres (simplemente llamada cadena en muchos otros idiomas).                                                                                                                                                                                                           | <code>"Pink Floyd"</code>         |
|            | lógico      | Representa valores booleanos                                                                                                                                                                                                                                                             | <code>TRUE, FALSE</code>          |
|            | Sin formato | Un vector que contiene bytes sin formato. Útil para codificar objetos desde fuera del entorno R.                                                                                                                                                                                         | <code>raw(8)</code>               |
| Compuestos | lista       | Una colección (posiblemente heterogénea) de otros objetos. Los elementos de una lista pueden ser nombrados. Muchos otros tipos de objetos en R (como marcos de datos) se implementan como listas.                                                                                        | <code>list(5,7,"cachucha")</code> |

**Tabla 3. Resumen. Tipología de objetos en R (continuación)**

|  |                   |                                                                                                                                                                                                                                                                                                                                       |                                                       |
|--|-------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------|
|  | listas aparejadas | Una estructura de datos utilizada para representar un conjunto de pares nombre-valor. <i>pairlist</i> se usa de manera interna, principalmente, pero se puede crear a nivel de usuario. Su función está en desuso en los programas de nivel de usuario, porque los objetos de lista estándar son igual de eficientes y más flexibles. | <code>pairlist(Manzana=1, Piña=2, Zanahoria=3)</code> |
|  | nulo              | Un objeto que significa “no hay objeto”. Devuelto por funciones y expresiones cuyo valor no está definido. El objeto NULL no tiene atributos.                                                                                                                                                                                         | NULL                                                  |

Fuente: elaboración propia con base en Teetor (2011) y Cotton (2013).

*R* tiene sintaxis especializada para acceder a las estructuras de datos. Puede obtener un solo elemento de una estructura o varios elementos (posiblemente como una matriz multidimensional) utilizando la notación de índice de *R*. Buscar elementos por ubicación dentro de una estructura de datos o por nombre (tabla 4).

**Tabla 4. Gestión. Acceso a datos dentro de una estructura de datos**

| Sintaxis            | Objetos          | Descripción                                                                                                                                                                                                                                                                                                          |
|---------------------|------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>X[i]</code>   | Vectores, listas | Devuelve objetos del objeto <i>X</i> descritos por <i>i</i> . Entonces, <i>i</i> puede ser un vector entero. Vector de caracteres (de nombres de objetos) o vector lógico. No permite coincidencias parciales. Cuando se usa con listas, vuelve a incluir una lista. Cuando se usa con vectores, devuelve un vector. |
| <code>X[[i]]</code> | Vectores, listas | Devuelve un único elemento de <i>X</i> , que coincide con <i>i</i> . Puede ser un número entero o un vector de caracteres de longitud 1. Permite coincidencias parciales.                                                                                                                                            |
| <code>X\$n</code>   | Listas           | Devuelve objetos nombrados “ <i>n</i> ” en el objeto <i>X</i> .                                                                                                                                                                                                                                                      |
| <code>x@n</code>    | Objetos          | Devuelve el elemento almacenado en la variable llamada <i>n</i> .                                                                                                                                                                                                                                                    |

Fuente: elaboración propia con base en Teetor (2011).

## Práctica

**1. ¿Cómo imprimir los dos primeros renglones con todas las columnas del vector  $m$ ? Capturado previamente en los ejemplos expuestos.**

Vamos a replicar la siguiente sintaxis. Con el comando `array` generamos un vector,  $a$ , con dos dimensiones de atributo

```
a<-array(c(1,2,3,4,5,6,7,8,9,10,11,12), dim = c(3,4))
```

**2. Se desea localizar el número 7 del vector  $a$ , ¿qué sintaxis producimos?**

**3. Deseamos localizar el número 5 del vector  $a$ , ¿cómo le hacemos?**

**4. Se desea sustituir el valor 5 del vector  $a$  por 500. Representar la sintaxis.**

Una matriz se construye de la siguiente forma:

```
mtx<-matrix(data = c(1,2,3,4,5,6,7,8,9,10,11,12), nrow = 3, ncol = 4)
mtx
```

Asimismo, `array` también puede tener más de dos dimensiones, ejemplo:

```
wf<-array(c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18),dim=c(3,3,2))
wf
```

**5. Si deseamos localizar el número 14 en el vector  $wf$ , ¿cuál es la sintaxis que debemos diseñar?**

**6. Obtener todos los valores del primer renglón en el vector  $a$ .**

**7. Obtener los valores 14, 15, 17 y 18 del vector  $wf$ .**

**8. Obtener la segunda columna del vector  $a$ .**

**9. Obtener el primer y segundo renglones del vector  $a$ .**

**10. Obtener el primer y tercer renglones del vector  $a$ .**

**11. Obtener los valores 4, 8 y 12 del vector  $mtx$ .**

## Bibliografía recomendada: profundizar con los autores

Abedin, J. y Kumar, D. K. (2015). *Data manipulation with R* (2a Ed.). Packt.

- Adler, J. (2012). *R in a nutshell: a desktop quick reference* (2a Ed.). O'Reilly Media.
- Chiu, Y.-W. (2015). *Machine learning with R cookbook*. Packt.
- Crawley, M. (2012). *The R Book* (2a Ed.). Wiley.
- Dennis, B. (2013). *The R Student Companion*. Chapman & Hall/CRC.
- Horton, N. y Kleinman, K. (2015). *Using R and RStudio for data management, statistical analysis and graphics* (2a Ed.). Chapman & Hall/CRC.
- Verzani, J. (2011). *Getting Started with RStudio*. O'Reilly Media.
- Zhao, Y. y Cen, Y. (2013). *Data mining applications with R*. Academic Press.
- Zumel, N. y Mount, J. (2014). *Practical Data Science with R*. Manning.

## **CAPÍTULO 2**

### **ELABORACIÓN Y GESTIÓN DE BANCOS DE DATOS**

**L**os datos son hechos registrados, cifras que se levantan en campo para su análisis, resumen, presentación e interpretación. A todos los datos reunidos para una determinada investigación se les denomina conjunto de datos. Una variable es la característica de algún objeto de interés. Al conjunto de mediciones obtenidas para entender un determinado fenómeno se le llama observaciones.

La recolección de datos requiere alguna de las escalas de medición siguientes: nominal, ordinal, de intervalo o razón. La escala de medición determina la información contenida en el dato, señala la manera más idónea de resumir y de analizar estadísticamente las observaciones.

Cuando el dato de una variable es una característica que se presenta como etiqueta o nombre que identifica o refiere al elemento, se considera que la escala de medición es nominal, por ejemplo, en un levantamiento de información, las variables más comunes son: género, nacionalidad y raza. Cuando la escala de medición es nominal, se usa un código o etiqueta no numérica, en los tres ejemplos anteriores, para género se responde de manera clásica con el sexo al que pertenece, masculino y femenino; en la actualidad, podría incorporarse el género con el que el individuo se identifica, en nacionalidad podría ser mexicana, extranjero y otras, en raza, mestizo, indígena, zambo, por ejemplo.

Una escala de medición para una variable es ordinal si los datos muestran las propiedades de los datos nominales y, además, tiene sentido el orden y jerarquía de los datos, por ejemplo, en evaluaciones de algún objeto, excelente, bueno y malo. Otra escala conocida es la de Likert, que asume la fuerza e

intensidad de la experiencia de forma lineal, por tanto, va desde un totalmente de acuerdo, a un totalmente en desacuerdo, asumiendo que las actitudes pueden ser medidas. Algunos ejemplos pueden ser obtenidos de la revista *Fortune*, donde continuamente se jerarquiza a empresas, empresarios, ambientes culturales y cuestiones económicas en el ámbito mundial: qué empresas petroleras en el mundo son las que mayor ingreso registran, qué universidad tiene mayor incidencia social.

Una escala de medición, para una variable, es de intervalo si los datos tienen las características de los datos ordinales y el intervalo entre valores se expresa en términos de una unidad de medición fija. Los datos de intervalo siempre son numéricos, por ejemplo, las pruebas de inteligencia donde se registra el coeficiente intelectual, las pruebas para medir el nivel de conocimiento de alguna lengua, es por intervalos.

Una variable tiene una escala de razón si los datos poseen las propiedades de intervalo y la proporción entre dos valores adquiere significado. Por ejemplo, la variable velocidad que se describe como distancia sobre tiempo. En algunas investigaciones se utiliza esta escala para expresar la complejidad en el problema. En una investigación que realicé sobre emigración, describí la razón de hombres y mujeres que se desplazan de manera legal a los Estados Unidos de Norteamérica, 6 de cada 10 mujeres emigra con documentos, cifra que difiere con la de los hombres: 4 de cada 10.

Los datos se obtienen de fuentes ya existentes o por medio de encuestas y estudios experimentales con el objeto de recolectar nuevos datos. En algunas ocasiones, los datos indispensables para una investigación no se pueden obtener de las fuentes existentes, en tales casos, los datos suelen conseguirse realizando un estudio estadístico, ya sea experimental u observacional. Es experimental cuando se identifica la variable de interés, posteriormente, se liga con una o más variables que son controladas para explicar de manera causal cómo ésta influye sobre las variables. Por ejemplo, cuando una empresa farmacéutica se interesa por conocer, a través de un experimento, cómo un medicamento influye en la presión arterial: la dosis del medicamento se controla suministrando diferentes dosis a distintos grupos de individuos. Antes y después, se mide la presión sanguínea de cada grupo. En los estudios observacionales no se controlan las variables de interés. En la encuesta, primero se identifican las preguntas de la investigación, después se presenta un cuestionario a los individuos de la muestra. En varios restaurantes acompañan la cuenta con un breve

cuestionario para saber la opinión de los comensales en relación con la calidad en la preparación de los alimentos, el lugar y el servicio.

Al resumen de datos en forma tabular, gráfica o numérica se le conoce como estadística descriptiva. Constantemente, se requiere de información acerca de los grupos sociales, debido al costo, sólo es posible recolectar datos a una parte de estos grupos. Al total de grupos se le llama población y a una parte se le conoce como muestra. Una de las principales contribuciones de la estadística es emplear datos de una muestra para hacer estimaciones y probar hipótesis acerca de las características de una población mediante un proceso al que se le conoce como inferencia estadística.

Este apartado contiene una entrada a la sintaxis de *R*. Los objetos son construcciones límite de un conjunto básico. Cada tipología define cómo es desarrollado. Además, son miembros de una clase. Las clases son definidas con información que contienen y cómo ésta puede ser utilizada.

El *software* dota de algunos mecanismos para programar objetos orientados. Los tipos ideales más básicos son:

- a) **Vectores básicos**, estos contienen un tipo singular de valor: enteros, decimales, números complejos, texto, valores lógicos o un conjunto de datos;
- b) **Objetos compuestos**, contienen varios vectores básicos, listas, pareja de listas y desarrollos, cada uno de estos contiene un número y nombre de objetos;
- c) **Objetos especiales**, tienen una propuesta especial en la programación de *R*, cada uno representa un aspecto importante en un contexto específico, *any*, *null*, ente otros, y
- d) **Lenguaje de *R***, este objeto que representa códigos puede ser revisado para regresar otros.

### **Función *data.frame***

En *R* se pueden construir estructuras más complicadas, estructuras de datos con sólo vectores. La función *data.frame* es una lista con la misma cantidad de

objetos (el mismo tamaño). Es una forma idónea de representar datos observacionales y experimentales.

Los *data.frames* son una forma útil de representar tablas. En contextos científicos, muchos experimentos consisten en observaciones individuales, las cuales implican varias medidas diferentes, múltiples dimensiones y algunas veces son cualitativas. En el contexto empresarial, los datos a menudo se mantienen en las tablas de la base de datos. Cada tabla tiene muchas filas, registros con múltiples “columnas”, atributos que representan diferentes cantidades en diversos formatos. Un marco de datos es una forma natural de representar estos en conjunto. Replique el siguiente ejemplo en su ordenador.

```
data.frame(a=c(1,2,3,4), b=c(1,2,3,4))
```

|   | a | b |
|---|---|---|
| 1 | 1 | 1 |
| 2 | 2 | 2 |
| 3 | 3 | 3 |
| 4 | 4 | 4 |

Un *data.frame* representa una tabla de datos. Cada columna puede ser de un tipo diferente, pero cada fila en el marco de datos debe tener la misma longitud. De lo contrario, el programa no corre la función, emitiendo un mensaje donde realiza los señalamientos que justifican por qué no ejecutó la orden. Ejemplo: en el siguiente vector es posible observar que la variable *a* tiene cinco registros y la variable *b* únicamente cuatro.

```
data.frame(a=c(1,2,3,4,5), b=c(1,2,3,4))
Error in data.frame(a = c(1, 2, 3, 4, 5), b = c(1, 2, 3,
4)) :
arguments imply differing number of rows: 5, 4
```

Por lo general, cada columna se nombra y a veces las filas también se nombran. Las columnas en un marco de datos a menudo se denominan variables. Aquí hay un ejemplo simple de *data.frame*, que muestra con qué frecuencia se realizan manifestaciones sociales en las principales ciudades del mundo. Replique el ejemplo.



```
Revolucion.searching.city<-data.frame(city=c("CDMX",
 "Tijuana", "Toluca",
 "New York", "Chicago",
 "Guadalajara"),
 rank=c(50, 30, 2, 46, 50, 18))
Revolucion.searching.city
```

```
 city rank
1 CDMX 50
2 Tijuana 30
3 Toluca 2
4 New York 46
5 Chicago 50
6 Guadalajara 18
```

Hay que recordar que, en ejercicios previos, conseguimos observar el tipo de información cargada en un vector, a través de la función *typeof* vamos a ejecutar ésta con el vector que contiene información de las manifestaciones en algunas ciudades del mundo. Los *data.frame* se implementan como listas con *typeof*. Ejecutar la sintaxis.

```
typeof(Revolucion.searching.city)
[1] "list"
```

```
class(Revolucion.searching.city)
[1] "data.frame"
```

Vamos a realizar otro ejemplo de base de datos con información imaginada, equipos de futbol de la liga mexicana (MX), información sobre el número de faltas y pases. Dicha información lleva por nombre juego. Elabore por su cuenta el ejercicio.

```
Equipo<-c("CRZ", "AME", "PCA", "ATL", "RMA")
```

```
Fltas<-c(92,89,94,72,59)
```

```

Pses<-c(70,73,77,90,102)

juego<-data.frame(Equipo,Fltas,Pses)

juego

```

```

Equipo Fltas Pses
1 CRZ 92 70
2 AME 89 73
3 PCA 94 77
4 ATL 72 90
5 RMA 59 102

```

Se puede referir un componente de vector lista o *data.frame* por nombre y utilizando el operador (\$) símbolo de pesos. Ejemplo: se pretende saber la cantidad de faltas que cometió cada equipo. Ejecute.

```

juego$Fltas
[1] 92 89 94 72 59

```

Aquí hay una forma de encontrar un valor específico en un marco de datos. Supongamos que quiere encontrar el número de pases que realizó en la temporada el equipo “PCA”. Una forma de seleccionar un número de una matriz es mediante el uso de un vector de valores booleanos para especificar qué elemento devolver de una lista. Puede calcular un vector apropiado como éste. Elabore.

```

juego$Pses[juego$Equipo=="PCA"]
[1] 77

```

Vamos a retomar sintaxis ya expuesta en casos anteriores para aplicarla al conjunto de información agrupada en el vector juego, básicamente recurrirnos a la función *class*.

```

class(Equipo)
[1] "character"

```

```

class(Fltas)
[1] "numeric"

class(Pses)
"numeric"

class(class)
[1] "function"

```

### Subconjunto de un *data.frame*

Una forma de tomar un subconjunto de un conjunto de datos es usar la notación de corchetes. Como recordará, puede seleccionar filas en un marco de datos al proporcionar un vector de valores lógicos. Si puede escribir una expresión simple que describa el conjunto de filas para seleccionar de un marco de datos, puede proporcionar esto como un índice.

Por ejemplo, supongamos como propósito seleccionar de la base de datos elaborada por el INEGI, el Índice de Rezago Social del Estado de México (CNI) para los años 2000, 2005 y 2010, sólo los municipios que pertenecen a la Zona Metropolitana del Valle de Toluca (ZMVT), 14 municipios de los 127 que tiene todo el estado: Toluca, Metepec, Zinacantepec, Almoloya de Juárez, Lerma, Otzolotepec, San Mateo Atenco, Ocoyoacac, Calimaya, Xonacatlán, San Antonio la Isla, Rayón, Mexicaltzingo y Chapultepec (tabla 5). La columna Municipio contiene asociado con cada fila, **CNI\$Municipio**, por lo que podríamos calcular un vector de valores lógicos que describa qué filas mantener con la expresión:

```

CNI$Municipio=="Toluca"|CNI$Municipio=="Metepec"|CNI$Municipio=="Zinacantepec"|CNI$Municipio=="Almoloya de Juárez"|CNI$Municipio=="Lerma"|CNI$Municipio=="Otzolotepec"|CNI$Municipio=="San Mateo Atenco"|CNI$Municipio=="Ocoyoacac"|CNI$Municipio=="Calimaya"|CNI$Municipio=="Xonacatlan"|CNI$Municipio=="San Antonio la Isla"|CNI$Municipio=="Rayon"|CNI$Municipio=="Mexicaltzingo"|CNI$Municipio=="Chapultepec"

```

El reto se encuentra en aplicar lo ya conocido y explicado en la tabla 1, el tipo de conjunción que se expresa en la sintaxis con una vertical (|), asimismo, especificar la información a través de una búsqueda lógica, booleana. Ahora, sólo tenemos que indexar el marco de datos con este vector para seleccionar únicamente filas para los municipios de la Zona Metropolitana del Valle de Toluca.

```
zmtv<-CNI[c(CNI$Municipio=="Toluca"|CNI$Municipio=="Meteppec"|
 CNI$Municipio=="Zinacantepec"|CNI$Municipio=="Almoloya
de Juarez"|
 CNI$Municipio=="Lerma"|CNI$Municipio=="Otzolotepec"|
 CNI$Municipio=="San Mateo Atenco"|CNI$Municipio=="Ocoyoacac"|
 CNI$Municipio=="Calimaya"|CNI$Municipio=="Xonacatlán"|
 CNI$Municipio=="San Antonio la
Isla"|CNI$Municipio=="Rayón"|
 CNI$Municipio=="Mexicaltzingo"|CNI$Municipio=="Chapultepec"),]

zmtv
 Entidad federativa Municipio 2000 2005 2010
6 Mexico Almoloya de Juarez 0.34148 0.25097 0.05331
19 Mexico Calimaya -0.96054 -0.82636 -0.89499
28 Mexico Chapultepec -1.24697 -1.04492 -1.34905
52 Mexico Lerma -1.19225 -1.03212 -1.08321
55 Mexico Meteppec -1.9688 -1.56716 -1.60753
56 Mexico Mexicaltzingo -1.08937 -0.67857 -0.99562
63 Mexico Ocoyoacac -1.22965 -1.07312 -1.06092
68 Mexico Otzolotepec -0.35328 -0.15946 -0.29824
73 Mexico Rayón -1.28257 -1.04200 -1.11098
74 Mexico San Antonio la Isla -1.13384 -0.88654 -1.27703
77 Mexico San Mateo Atenco -1.15123 -0.91667 -1.03127
107 Mexico Toluca -1.39402 -1.14755 -1.19348
116 Mexico Xonacatlán -0.70864 -0.54157 -0.76504
119 Mexico Zinacantepec -0.63567 -0.52447 -0.67416
```

**Tabla 5. Índice de Rezago Social. Estado de México, 2000-2015**

| Municipio                 | 2000     | 2005     | 2010     |
|---------------------------|----------|----------|----------|
| Estatál                   | -0.54346 | -0.36296 | -0.36513 |
| Acambay de Ruíz Castañeda | 0.50596  | 0.38519  | 0.31887  |
| Acolman                   | -1.51153 | -1.23048 | -1.18213 |
| Aculco                    | 0.37643  | 0.19373  | 0.17769  |
| Almoloya de Alquisiras    | 0.27008  | 0.10998  | -0.00909 |
| Almoloya de Juárez        | 0.34148  | 0.25097  | 0.05331  |
| Almoloya del Río          | -1.10418 | -0.79882 | -0.93121 |
| Amanalco                  | 0.8083   | 0.56849  | 0.46295  |
| Amatepec                  | 0.3425   | 0.20519  | 0.28052  |
| Amecameca                 | -1.11936 | -0.85092 | -0.88831 |
| Apaxco                    | -1.1393  | -1.05737 | -0.93097 |
| Atenco                    | -1.24495 | -0.88549 | -0.97896 |
| Atizapán                  | -0.89645 | -0.67374 | -0.77033 |
| Atizapán de Zaragoza      | -1.82932 | -1.50901 | -1.48194 |
| Atlacomulco               | -0.4911  | -0.38416 | -0.48033 |
| Atlautla                  | -0.32054 | -0.24286 | -0.34851 |
| Axapusco                  | -0.67775 | -0.672   | -0.59501 |
| Ayapango                  | -1.01361 | -0.75041 | -0.91678 |
| Calimaya                  | -0.96054 | -0.82636 | -0.89499 |
| Capulhuac                 | -1.20025 | -0.96845 | -0.97132 |
| Coacalco de Berriozábal   | -2.22365 | -1.73468 | -1.75984 |
| Coatepec Harinas          | 0.31696  | 0.19386  | 0.148    |
| Cocotitlán                | -1.46105 | -1.09693 | -1.16585 |
| Coyotepec                 | -1.08712 | -0.84369 | -0.88113 |
| Cuautitlán                | -1.95175 | -1.45087 | -1.68431 |
| Chalco                    | -1.11459 | -0.8217  | -0.95631 |
| Chapa de Mota             | 0.41623  | 0.20174  | 0.05656  |
| Chapultepec               | -1.24697 | -1.04492 | -1.34905 |
| Chiautla                  | -1.21663 | -1.03698 | -1.06704 |

**Tabla 5. Índice de Rezago Social. Estado de México, 2000-2015**  
(continuación)

| Municipio           | 2000     | 2005     | 2010     |
|---------------------|----------|----------|----------|
| Chicoloapan         | -1.40206 | -1.00996 | -1.43285 |
| Chiconcuac          | -1.28732 | -1.07296 | -1.09314 |
| Chimalhuacán        | -1.03632 | -0.78821 | -0.82922 |
| Donato Guerra       | 1.22578  | 0.95631  | 1.10993  |
| Ecatepec de Morelos | -1.67478 | -1.30801 | -1.36873 |
| Ecatzingo           | 0.05285  | 0.18699  | -0.03563 |
| Huehuetoca          | -1.44006 | -1.26617 | -1.4006  |
| Hueypoxtla          | -0.42868 | -0.43862 | -0.52064 |
| Huixquilucan        | -1.53268 | -1.18563 | -1.37281 |
| Isidro Fabela       | -0.39881 | -0.37857 | -0.55844 |
| Ixtapaluca          | -1.53314 | -1.20985 | -1.24488 |
| Ixtapan de la Sal   | -0.34208 | -0.35289 | -0.4199  |
| Ixtapan del Oro     | 0.57652  | 0.47782  | 0.35368  |
| Ixtlahuaca          | 0.29793  | 0.21742  | 0.20427  |
| Xalatlaco           | -0.59402 | -0.44309 | -0.431   |
| Jaltenco            | -1.74556 | -1.53734 | -1.47777 |
| Jilotepec           | -0.16536 | -0.33435 | -0.29937 |
| Jilotzingo          | -0.7936  | -0.77874 | -0.80659 |
| Jiquipilco          | 0.43049  | 0.26653  | 0.17013  |
| Jocotitlán          | -0.4863  | -0.41524 | -0.40119 |
| Joquicingo          | -0.40242 | -0.34533 | -0.38544 |
| Juchitepec          | -0.63235 | -0.52168 | -0.58628 |
| Lerma               | -1.19225 | -1.03212 | -1.08321 |
| Malinalco           | 0.03713  | 0.1189   | -0.09264 |
| Melchor Ocampo      | -1.41327 | -1.14401 | -1.21228 |
| Metepec             | -1.9688  | -1.56716 | -1.60753 |
| Mexicaltzingo       | -1.08937 | -0.67857 | -0.99562 |
| Morelos             | 0.74951  | 0.74744  | 0.58488  |

| Municipio                   | 2000     | 2005     | 2010     |
|-----------------------------|----------|----------|----------|
| Naucalpan de Juárez         | -1.69941 | -1.37183 | -1.34676 |
| Nezahualcóyotl              | -1.70834 | -1.44287 | -1.37588 |
| Nextlalpan                  | -0.95994 | -0.82945 | -0.95069 |
| Nicolás Romero              | -1.35118 | -1.11894 | -1.11785 |
| Nopaltepec                  | -0.81513 | -0.79581 | -0.86337 |
| Ocoyoacac                   | -1.22965 | -1.07312 | -1.06092 |
| Ocuilan                     | 0.09168  | 0.05944  | -0.07308 |
| El Oro                      | 0.03011  | -0.06777 | -0.00731 |
| Otumba                      | -0.70524 | -0.65657 | -0.76807 |
| Otzoloapan                  | 0.31768  | 0.25539  | -0.1102  |
| Otzolotepec                 | -0.35328 | -0.15946 | -0.29824 |
| Ozumba                      | -0.71046 | -0.5883  | -0.59929 |
| Papalotla                   | -1.38219 | -1.07799 | -1.18486 |
| La Paz                      | -1.34737 | -1.03004 | -1.01325 |
| Polotitlán                  | -0.81088 | -0.79529 | -0.82552 |
| Rayón                       | -1.28257 | -1.042   | -1.11098 |
| San Antonio la Isla         | -1.13384 | -0.88654 | -1.27703 |
| San Felipe del Progreso     | 1.25305  | 0.99549  | 1.0062   |
| San Martín de las Pirámides | -1.09821 | -0.9321  | -1.02908 |
| San Mateo Atenco            | -1.15123 | -0.91667 | -1.03127 |
| San Simón de Guerrero       | 0.05342  | -0.08328 | -0.18784 |
| Santo Tomás                 | -0.15476 | -0.21199 | -0.49883 |
| Soyaniquilpan de Juárez     | -0.41807 | -0.60366 | -0.44928 |
| Sultepec                    | 1.13806  | 0.90155  | 0.81364  |
| Tecámac                     | -1.55286 | -1.26356 | -1.52306 |
| Tejupilco                   | 0.09884  | -0.03252 | -0.1233  |
| Temamatla                   | -1.41103 | -0.98499 | -1.17929 |
| Temascalapa                 | -0.77056 | -0.66614 | -0.71366 |
| Temascalcingo               | 0.2833   | 0.1563   | 0.15751  |
| Temascaltepec               | 0.53166  | 0.52021  | 0.21039  |

**Tabla 5. Índice de Rezago Social. Estado de México, 2000-2015**  
(continuación)

| Municipio           | 2000     | 2005     | 2010     |
|---------------------|----------|----------|----------|
| Temoaya             | 0.43314  | 0.33737  | 0.079    |
| Tenancingo          | -0.64407 | -0.5185  | -0.5517  |
| Tenango del Aire    | -1.2053  | -0.98746 | -1.06287 |
| Tenango del Valle   | -0.63665 | -0.53636 | -0.51907 |
| Teoloyucan          | -1.34003 | -1.0771  | -1.11185 |
| Teotihuacán         | -1.33208 | -1.13949 | -1.16037 |
| Tepetlaoxtoc        | -0.96877 | -0.91253 | -0.89654 |
| Tepetlixpa          | -0.50561 | -0.42001 | -0.30026 |
| Tepotzotlán         | -1.50963 | -1.20341 | -1.0981  |
| Tequixquiác         | -0.90323 | -0.9507  | -0.8421  |
| Texcaltitlán        | 0.36761  | 0.33007  | 0.05926  |
| Texcalyacac         | -1.13578 | -1.01837 | -0.98889 |
| Texcoco             | -1.49681 | -1.23316 | -1.16383 |
| Tezoyuca            | -1.40153 | -0.92074 | -0.79707 |
| Tianguistenco       | -0.86635 | -0.6511  | -0.6448  |
| Timilpan            | -0.1169  | -0.25803 | -0.40675 |
| Tlalmanalco         | -1.65404 | -1.24727 | -1.26824 |
| Tlalnepantla de Baz | -1.86556 | -1.50019 | -1.49449 |
| Tlatlaya            | 0.49971  | 0.39293  | 0.21732  |
| Toluca              | -1.39402 | -1.14755 | -1.19348 |
| Tonatico            | -0.64247 | -0.54748 | -0.77455 |
| Tultepec            | -1.64777 | -1.22402 | -1.25688 |
| Tultitlán           | -1.88548 | -1.47706 | -1.51313 |
| Valle de Bravo      | -0.6623  | -0.54851 | -0.65045 |
| Villa de Allende    | 1.07604  | 1.114    | 0.856    |
| Villa del Carbón    | 0.32999  | 0.19394  | 0.07857  |
| Villa Guerrero      | 0.16226  | 0.13839  | 0.08792  |
| Villa Victoria      | 1.30683  | 1.23446  | 1.28379  |



| Municipio                   | 2000     | 2005     | 2010     |
|-----------------------------|----------|----------|----------|
| Xonacatlán                  | -0.70864 | -0.54157 | -0.76504 |
| Zacazonapan                 | -0.47423 | -0.44264 | -0.74723 |
| Zacualpan                   | 0.69828  | 0.58849  | 0.69397  |
| Zinacantepec                | -0.63567 | -0.52447 | -0.67416 |
| Zumpahuacán                 | 0.6774   | 0.62528  | 0.31326  |
| Zumpango                    | -1.24544 | -0.85034 | -1.1087  |
| Cuautitlán Izcalli          | -2.07743 | -1.61801 | -1.6556  |
| Valle de Chalco Solidaridad | -1.16845 | -0.90852 | -0.91349 |
| Luvianos                    | NULL     | 0.35913  | 0.55874  |
| San José del Rincón         | NULL     | 1.45587  | 1.3563   |
| Tonanitla                   | NULL     | -0.94    | -1.1059  |

Fuente: elaboración propia con base en INEGI (2015).

En programación no existe sólo una manera de ejecutar los procedimientos, en el menor tiempo y de la forma más eficiente se pretende obtener la información. Como alternativa de un subconjunto de datos, se puede usar la función de subconjunto para seleccionar información predilecta de filas y columnas dentro de un marco de datos o matriz.

`subset(x, subset, select, drop=FALSE, ...)`

A menudo, es deseable tomar una muestra aleatoria de un conjunto de datos. A veces, puede tener demasiados datos, por razones estadísticas o de rendimiento. Otras veces, simplemente desea dividir sus datos en diferentes partes para modelar, usualmente, en subconjuntos de entrenamiento, pruebas y validación.

Una de las formas más simples de extraer una muestra aleatoria es con la función *sample*. La función devuelve una muestra aleatoria de los elementos de un vector (tabla 6).

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

Donde cada categoría significa:

**Tabla 6. Argumento y descripción para la selección de un conjunto de datos muestra**

|        |                                                                      |       |
|--------|----------------------------------------------------------------------|-------|
| x      | El objeto del cual se tomará la muestra.                             |       |
| size   | Número entero que especifica el tamaño de la muestra.                |       |
| repace | Un valor lógico que indica si se debe muestrear con o sin reemplazo. | FALSE |
| prob   | Un vector de probabilidades para seleccionar cada elemento.          | NULL  |

Fuente: elaboración propia con base en Teetor (2011) y Cotton (2013).

Ejemplo.

```
sample(CNI$Municipio, size = 5, replace = FALSE, prob = NULL)
```

```
[1] "Zumpango" "Polotitlan" "Nezahualcoyotl" "Temascalcingo"
"Jilotepec"
```

De forma no definitiva, cuando se aplica a un *data.frame*, la muestra devolverá una muestra aleatoria de las columnas, recuerde que un marco de datos se implementa como una lista de vector, por lo que un ejemplo es simplemente tomar una muestra aleatoria de los elementos de la lista. Por tanto, debe ser un poco más inteligente cuando utiliza la muestra con un *data.frame*.

Use la función *sample* para crear una muestra aleatoria de números de fila y luego seleccionar estos números de fila usando un operador de índice. Por ejemplo, tomemos una muestra aleatoria de siete registros de CNI.

```
CNI[sample(1:nrow(CNI),7),]
 Entidad federativa Municipio 2000 2005 2010
72 Mexico Polotitlan -0.81088 -0.79529 -0.82552
60 Mexico Nextlalpan -0.95994 -0.82945 -0.95069
86 Mexico Temascalcingo 0.2833 0.15630 0.15751
47 Mexico Jilotzingo -0.7936 -0.77874 -0.80659
84 Mexico Temamatla -1.41103 -0.98499 -1.17929
46 Mexico Jilotepec -0.16536 -0.33435 -0.29937
74 Mexico San Antonio la Isla -1.13384 -0.88654 -1.27703
```

La función *tapply* es muy flexible para resumir un vector *x*. Puede especificar qué subconjuntos de *x* resumir, así como la función utilizada para sumari- zación (tabla 7).

`tapply(x, INDEX, FUN= , ..., simplify=)`

**Tabla 7. Argumento y descripción para la selección de *tapply***

| Argumento | Descripción                                                                                                                                                                                                            | Default |
|-----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------|
| x         | El objeto sobre el cual aplicar la función, gene-<br>ralmente un vector.                                                                                                                                               |         |
| INDEX     | Una lista de factores que especifican diferen-<br>tes conjuntos de valores de X sobre los cuales<br>calcular FUN, cada uno con la misma longi-<br>tud que X.                                                           |         |
| FUN       | El factor aplicado para elementos de X.                                                                                                                                                                                | NULL    |
| ...       | Los argumentos opcionales se pasan a FUN.                                                                                                                                                                              |         |
| simplify  | Si <code>simplify = TRUE</code> , entonces FUN devuelve<br>un escalar, entonces <i>tapply</i> regresa una matriz<br>con el modo del escalar. Si <code>simplify = FALSE</code> , a<br>continuación, devuelve una lista. | TRUE    |

Fuente: elaboración propia con base en Teetor (2011) y Cotton (2013).

Comúnmente, queremos agregar información en una región, obtener en bloque un resumen de datos de un municipio, teniendo información de todas sus colonias, en otros casos, poseemos información de los municipios y agre- gamos una región o todo el estado y así sucesivamente podemos ir agregando escalas. Vamos a usar *tapply* para promediar el Índice de Rezago Social del año 2005, en todo el Estado de México.

```
tapply(X=CNI$`2005`, INDEX = list(CNI$`Entidad federativa`),
FUN = mean)
México
-0.5260945
```

Otro ejemplo que vamos a mostrar es con los estados registrados en la EOD, sumaremos los viajes de cada entidad federativa. Cabe recordar que el INEGI aplica una clave a cada estado que va del 01 al 32, en orden alfabético. Como se puede observar, la Ciudad de México (09) acumula la mayor cantidad, le sigue el Estado de México (15) y, finalmente, se ubica Hidalgo (13).

```
tapply(X=tr_viajes$NVIAJE, INDEX=list(tr_viajes$CENTIDADDE),
FUN = sum)
05 09 11 12 13 14 15 16 17 19 20
2 270495 9 9 221 13 166079 9 90 1 3
21 22 23 24 29 30 32 99
45 30 4 8 9 13 2 288
```

A menudo, puede ser útil contar el número de observaciones que toman cada valor posible de una variable, *R* proporciona varias funciones para hacer esto.

La función más simple para contar el número de observaciones que toman un valor es la función de tabulación, *tabulate*. Esta función cuenta el número de elementos en un vector que hacen en cada valor entero y devuelve un vector con los recuentos.

```
distritos_mun<-tabulate(tr_viajes$CMUNICIPIO)
distritos_mun
[1] 9147 12010 3322 16957 5878 18035 2609
[8] 1518 12046 3942 7805 7935 11114 19645
[15] 11227 8195 2556 196 1134 2285 258
[22] 1833 220 2708 12787 566 1484 2694
[29] 90 182 9824 11870 122 1842 59
[36] 1402 563 2361 506 1114 212 555
[43] 2181 270 380 8876 353 2151 689
[50] 4155 1428 46 409 571
```

Una función relacionada para valores categóricos es *tabla*, *table*. Supongamos que se le presentan algunos datos que incluyen valores categóricos,

codificados como factor en *R*, y se desea contar cuántas observaciones tiene cada valor categórico. Para hacer esto, puede usar la función de tabla.

```
table(tr_viajes$CMUNICIPIO)

002 003 004 005 006 007 008 009
9147 12010 3322 16957 5878 18035 2609 1518
010 011 012 013 014 015 016 017
12046 3942 7805 7935 11114 19645 11227 8195
020 023 024 025 028 029 030 031
2556 196 1134 2285 258 1833 220 2708
033 035 037 039 044 053 057 058
12787 566 1484 2694 90 182 9824 11870
059 060 069 070 075 081 091 092
122 1842 59 1402 563 2361 506 1114
093 095 099 100 103 104 108 109
212 555 2181 270 380 8876 353 2151
120 121 122 125 998 999
689 4155 1428 46 409 571
```

Para que esto sea un poco más interesante, podríamos hacer una tabla bidimensional que muestre el número de viajes en la Alcaldía Cuauhtémoc, 015, y el municipio de Ecatepec, 033.

```
table(tr_viajes$CMUNICIPIO[c(tr_viajes$CMUNICIPIO=="033"|
tr_viajes$CMUNICIPIO=="015")])

002 003 004 005 006 007 008 009 010 011 012 013 014 015 016
0 0 0 0 0 0 0 0 0 0 0 0 0 0 19645 0
017 020 023 024 025 028 029 030 031 033 035 037 039 044 053
0 0 0 0 0 0 0 0 0 12787 0 0 0 0 0
057 058 059 060 069 070 075 081 091 092 093 095 099 100 103
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
104 108 109 120 121 122 125 998 999
0 0 0 0 0 0 0 0 0
```

## Diseño de fórmulas

Muy a menudo, se necesita expresar una relación entre variables; a veces, trazar un cuadro que muestre la relación entre dos; otras veces, desarrollar un algoritmo matemático. *R* proporciona una función de fórmula que permite describir la relación para ambos propósitos.

Vamos a generar una representación con base en una fórmula. Precisamos develar el desconocimiento geográfico, teoría desarrollada por Lacoste (1977) sin llegar al algoritmo diseñado en el presente libro con las siguientes variables: aumento de los intercambios, más el crecimiento de las ciudades, más representación del espacio por despojo. Es decir, la suma de cada una de las variables independientes permitirá explicar, de forma lineal, la variable dependiente que necesitamos evidenciar.

Sustitución:

$y$  = Desconocimiento geográfico  
 $x_1$  = Aumento de los intercambios  
 $x_2$  = Crecimiento de las ciudades  
 $x_3$  = Representación del espacio por despojo

Sintaxis

```
Desconocimiento.geografico<-as.formula(y~x1+x2+x3)
Desconocimiento.geografico
y ~ x1 + x2 + x3
```

Como hemos realizado en ocasiones anteriores, es conveniente validar la clase de objeto que tiene el vector `Desconocimiento.geografico`, sobra decir que la función es *class*. Asimismo, qué tipo de información, *typeof*. Los objetos lenguaje a menudo son procesos propios del *software R*.

```
class(Desconocimiento.geografico)
[1] "formula"

typeof(Desconocimiento.geografico)
[1] "language"
```

Los diferentes símbolos utilizados en fórmulas se exponen a continuación.

*Tilde* (~)

Se usa para mostrar la relación entre las variables de respuesta, a la izquierda, y la variable de estímulo, a la derecha.

*Asignar* (+)

Símbolo para expresar una relación lineal entre variables.

*Cero* (0)

Cuando se agrega a una fórmula, indica que se debe incluir el término para interceptar.

*Barra vertical* (|)

Usado para especificar variables de condicionamiento.

*Asterisco* (\*)

Usado para indicar interacciones entre variables. Por ejemplo:

$$y \sim (u+v)^*w$$

*Capuchón* (^)

Usado para indicar el cruce con un grado específico. Por ejemplo:

$$y \sim (u+w)^2$$

Que equivale a:

$$y \sim (u+w)^*(u+w)$$

## Atributos

Un objeto en *R* puede tener muchas propiedades asociadas, llamadas atributos. Estas propiedades explican qué representa un objeto y cómo debería ser interpretado. Muy a menudo, la única diferencia entre dos objetos similares son los diferentes atributos. Algunos atributos importantes se muestran en la tabla siguiente. Muchos objetos en *R* se utilizan para representar datos numéricos, en particular, matrices, *array*, y *data.frames*. Por tanto, muchos atributos comunes se refieren a las propiedades de estos objetos (tabla 8).

**Tabla 8. Tipología de objetos en R**

|           |                                                                                                                                                                                                                  |
|-----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| class     | La clase del objeto, también puede utilizar la función <i>mode</i> .                                                                                                                                             |
| comment   | Un comentario sobre el objeto; a menudo, una descripción de lo que significa el objeto.                                                                                                                          |
| dim       | Dimensiones del objeto.                                                                                                                                                                                          |
| dimnames  | Nombres asociados con cada dimensión del objeto.                                                                                                                                                                 |
| names     | Regresa el nombre de los atributos del objeto. Los resultados dependen del tipo de objeto; por ejemplo, devuelve el nombre de cada columna de datos en <i>data.frame</i> o cada objeto con nombre en una matriz. |
| row.names | El nombre de cada renglón en un objeto.                                                                                                                                                                          |
| levels    | Niveles de un factor.                                                                                                                                                                                            |

Fuente: elaboración propia con base en Teetor (2011).

Hay una forma estándar de consultar atributos de objetos en R, con la función *attributes()*. Para un objeto  $x$  y un supuesto vector  $a$ , se refiere como “ $a$ ” través de,  $a(x)$ . En la mayoría de los casos, hay un método para obtener el valor actual, consiste en alterarlos en el entorno actual, pero no los afectará en el medio circundante. Ejemplo:

```
m<-matrix(data=1:12, nrow=4, ncol=3,
 dimnames=list(c("r1", "r2", "r3", "r4"),
 c("c1", "c2", "c3")))
attributes(m)
$dim
[1] 3 4

dimnames(m)
[[1]]
[1] "r1" "r2" "r3" "r4"

[[2]]
[1] "c1" "c2" "c3"
```



```
colnames(m)
[1] "c1" "c2" "c3"
rownames(m)
[1] "r1" "r2" "r3" "r4"
```

Cada símbolo en  $R$  se define dentro de un entorno específico. Un entorno es un objeto  $R$  que contiene el conjunto de símbolos disponibles en un contexto dado, los objetos asociados con esos símbolos y un puntero a un entorno principal. Los símbolos y los objetos asociados se llaman *frame*.

Cada contexto de evaluación en  $R$  está asociado a un entorno. Cuando  $R$  intenta resolver un símbolo, comienza mirando a través del entorno actual. Si no hay coincidencia en el entorno local,  $R$  buscará de forma iterativa a través de parentales buscando una coincidencia.

Cuando se define una variable en  $R$ , en realidad, está asignando un símbolo a un valor en un entorno. Por ejemplo, cuando se ingresa la siguiente formulación:

```
Maradona<-10
```

En la consola,  $R$  asigna el símbolo “ $x$ ” a un objeto vectorial de longitud 1 con el valor constante 10 (doble) en el entorno global. Cuando el intérprete de  $R$  evalúa una expresión, evalúa todos los símbolos. Si compone un objeto a partir de un conjunto de símbolos, el *software* resolverá los símbolos en el momento en que se construye el objeto.

```
Maradona<-10
Messi<-10.2
Ronaldinho<-9
```

```
Trinidad<-c(Maradona, Messi, Ronaldinho)
Trinidad
[1] 10.0 10.2 9.0
```

## Funciones

En *R*, los objetos de función se definen con esta sintaxis:

*función(argumentos) cuerpo.*

Donde los argumentos son un conjunto de nombres, símbolos (y, opcionalmente, valores por defecto) que se definirán dentro del cuerpo de la función, y el cuerpo es una expresión *R*. Típicamente, el cuerpo está envuelto en llaves, pero no tiene que serlo si el cuerpo es una sola expresión. Por ejemplo, las siguientes dos definiciones son equivalentes (Teetor, 2011).

```
F<-function(x,y) x+y
F<-function(x,y) {x+y}
```

Una función son los objetos *R* que evalúan el conjunto de argumentos de entrada y devuelven un valor de salida. Este apartado explica cómo crear y usar funciones en *R*.

Una definición de función en *R* incluye los nombres de los argumentos. Opcionalmente, puede incluir valores predeterminados. Si se especifica un valor predeterminado para un argumento, entonces el argumento se considera opcional.

```
fg<-function(x,y=10){x+y}
>fg(1)
[1] 11
```

Como ejemplo, presentamos una función que imprime el primer argumento y luego pasa todos los demás argumentos a la función de resumen. Para hacer esto, crearemos una función que tiene un argumento “*x*”. La especificación de argumentos también incluye una elipsis para indicar que la función toma otros argumentos. Entonces, podemos llamar a la función de resumen con los puntos suspensivos como su argumento.

```

> vb<-c(sqrt(1:100))
> full<-function(x,...){print(x); summary(...)}
> full("Here is the summary for vb.", vb, digits=2)
[1] "Here is the summary for vb."
 Min. 1st Qu. Median Mean 3rd Qu.
 1.0 5.1 7.1 6.7 8.7
 Max.
 10.0

```

Observe que todos los argumentos después de  $x$  pasaron al resumen. Algunas funciones están asociadas a una clase específica. Estos se llaman métodos. (No todas las funciones están ligadas estrechamente a una clase en particular, el sistema de clases en *R* es mucho más formal que el de un lenguaje como Java.)

Todas las funciones en *R* devuelven un valor. Algunas funciones también hacen otras cosas, cambiar variables en el entorno actual (o en otro entorno), trazar gráficos, cartografías, cargar o guardar archivos o acceder a la red. Estas operaciones se llaman efectos secundarios.

En *R*, los métodos para diferentes clases pueden compartir el mismo nombre. Éstas se llaman funciones genéricas. Las funciones genéricas tienen dos propósitos: primero, hacen que sea fácil adivinar el nombre correcto de la función para una clase desconocida. En segundo lugar, las funciones genéricas hacen posible usar el mismo código para objetos de diferentes tipos. Ejemplo de ello es el operador más (+).

## AUTOEVALUACIÓN

1. ¿Cuál es la diferencia entre dato, valor numérico y estadística?
2. La cadena City Express evalúa la calidad de sus hoteles entre sus huéspedes con el fin de determinar los mejores alojamientos que se encuentran en el país en una muestra de 64 hoteles distribuidos en el territorio nacional. Los precios de una habitación doble estándar van de \$ (precio más bajo) a \$\$\$\$ (precio más alto). La calificación general corresponde a la evaluación de habitaciones, servicio, desayuno buffet, ubicación y

áreas públicas; cuanto más alta sea la calificación general, mayor es el nivel de satisfacción.

- a) ¿Cuántas observaciones hay en este conjunto de datos?
  - b) ¿Cuántas variables hay en este conjunto de datos?
  - c) ¿Cuántas variables son cualitativas?
  - d) ¿Cuáles son cuantitativas?
3. El periódico *Reforma* realiza una encuesta a sus suscriptores, con cerca de 50 preguntas, acerca de sus características e intereses. De las siguientes preguntas, señale para cada una: ¿cuál de ellas proporciona datos cualitativos o cuantitativos?, además, ¿alguna indica la escala de medición apropiada?
- a) ¿Qué edad tiene?
  - b) ¿Género?      M      F
  - c) ¿Cuándo comenzó a leer el periódico *Reforma*?  
Preparatoria, universidad, comienzo de la carrera, mitad de la carrera, final de la carrera, ya retirado.
  - d) ¿Antigüedad en el trabajo?
  - e) ¿Automóvil que compraría en el futuro próximo?  
Sedan, compacto, deportivo, miniván, camioneta.

La encuestadora Gallup realizó una encuesta telefónica empleando una muestra aleatoria compuesta por 1,085 adultos de 18 años y más. Se les preguntó a los participantes: ¿Cómo considera que es su salud física en este momento? Las respuestas podían ser: Excelente, Buena, Regular o Ninguna opinión.

- a) ¿Cuál es el tamaño de la muestra en esta investigación?
- b) ¿Son datos cualitativos o cuantitativos?
- c) ¿Sería conveniente usar promedios o porcentajes para resumir los datos de esas preguntas?

- d) De las personas que respondieron, 41% dijo que su salud era excelente, ¿cuántos individuos dieron esta respuesta?

**Bibliografía recomendada:  
profundizar con los autores**

- Abedin, J. y Kumar, D. K. (2015). *Data manipulation with R* (2a Ed.). Packt.
- Aho, K. (2013). *Foundational and applied statistics for biologists using R*. Chapman & Hall/CRC.
- Aho, K. (2014). *Foundational and applied statistics for biologists using R: introduction to R*. CRC.
- Albert, J. y Rizzo, M. (2012). *R by example: concepts to code*. Springer.
- Allerhand, M. (2011). *A tiny handbook of R*. Springer.
- Boogaart, K. y Tolosana-Delgado, R. (2013). *Analyzing compositional data with R*. Springer.
- Cotton, R. (2013). *Learning R*. O'Reilly Media.
- Gohil, A. (2015). *R data visualization cookbook: over 80 recipes to analyze data and create stunning visualizations with R*. Packt.
- Nolan, D. y Lang, D. (2015). *Data Science in R: a case studies approach to computational reasoning and problem solving*. Chapman & Hall/CRC.
- Stanton, J. (2013). *Introduction to Data Science*. Sage.
- Thiem, A. y Duşa, A. (2013). *Qualitative comparative analysis with R: a user S Guide*. Springer.
- Tollefson, M. (2014). *R quick syntax reference*. Apress.
- Voss, J. (2013). *An introduction to statistical computing: a simulation-based approach*. Wiley.
- Wickham, H. (2014). *Advanced R*. Chapman & Hall/CRC.
- Zhao, Y. y Cen, Y. (2013). *Data mining applications with R*. Academic Press.
- Zumel, N. y Mount, J. (2014). *Practical data science with R*. Manning.



## CAPÍTULO 3

### DISEÑO DE GRÁFICOS

**R** incluye diferentes complementos para bosquejar datos y gráficos. El paquete de gráficos modela una amplia variedad de funciones con los datos. Es fácil personalizar o modificar gráficos con el paquete de gráficos o interactuar con gráficos en la pantalla.

El paquete incluye herramientas para dibujar diferentes tipos de gráficos, entre los más comunes, gráficos de barras, gráficos circulares, gráficos de líneas y diagramas de dispersión. Además, también se puede dibujar algunos gráficos menos familiares, como parcelas de quantil-quantil, Q-Q, diagramas de mosaico y diagramas de contorno. La tabla 9 muestra la genealogía de gráficos incluidos en el paquete de gráficos.

**Tabla 9. Tipos de gráficos**

| Gráficos   | Descripción                                                                                                                                                                                                                                        |
|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Barras     | Representa los datos cualitativos de una distribución de frecuencia relativa o porcentual. En el eje horizontal ( $x$ ) se especifican las etiquetas para las clases.                                                                              |
| Puntos     | Uno de los más sencillos resúmenes gráficos de datos. En el eje horizontal ( $x$ ) se presenta el intervalo de los datos.                                                                                                                          |
| Histograma | Usual para datos cualitativos. Se realiza con datos previamente resumidos mediante una distribución de frecuencia relativa o porcentual. La variable de interés se coloca en el eje horizontal ( $x$ ) y la frecuencia en el eje vertical ( $y$ ). |

**Tabla 9. Tipos de gráficos (continuación)**

| Gráficos         | Descripción                                                                                                 |
|------------------|-------------------------------------------------------------------------------------------------------------|
| Densidad         | El área de las barras del histograma suma 1, valor que representa la probabilidad total de la distribución. |
| Qqnorm           | Gráfico quantil-quantil.                                                                                    |
| Xplot            | Scatter plot, nube de puntos.                                                                               |
| SmoothScatter    | Parcela de dispersión suave.                                                                                |
| Pairs            | Matrices scatter plot.                                                                                      |
| Image            | Gráfico de imagen.                                                                                          |
| Contour          | Presenta distribuciones de frecuencia relativa y porcentual de datos cualitativos.                          |
| Perp             | Perspectiva de los gráficos en tercera dimensión.                                                           |
| Interaction.plot | Resumen de la respuesta para combinaciones de factores bidireccionales.                                     |
| Sunflowerplot    | Parcela de girasol.                                                                                         |

Fuente: elaboración propia con base en Cotton (2013) y Chang (2013).

R puede mostrar gráficos en la pantalla o guardarlos en muchos formatos diferentes. Hay una enorme cantidad de control sobre los gráficos, sin embargo, no se pueden controlar todos los aspectos de un gráfico.

### Utilidad y uso de los gráficos

Una gráfica de barras se utiliza para representar los datos cualitativos de una distribución de frecuencia, relativa o porcentual, asimismo, con base en unos procedimientos técnicos podemos observar la estructura poblacional en una región. La gráfica de pastel presenta distribuciones de frecuencia relativa y de frecuencia porcentual de datos cualitativos.

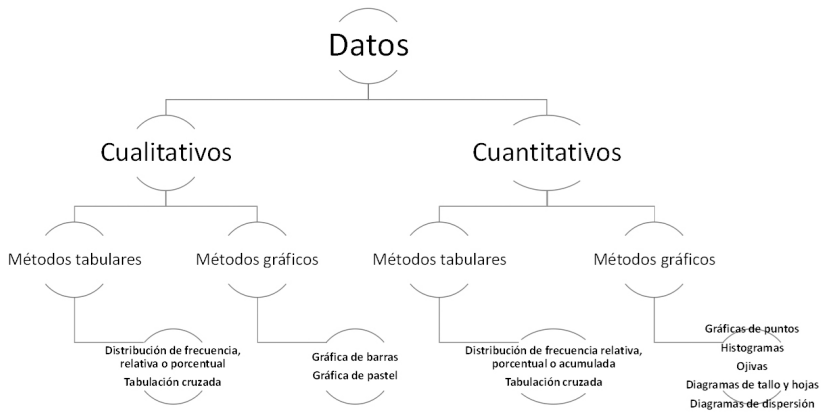
Una gráfica común para los datos cuantitativos es el histograma. Se realiza con los datos previamente resumidos mediante una distribución de frecuencia, relativa o porcentual. Uno de los usos más importantes de un histograma es proveer información acerca de la forma de la distribución. La gráfica de barras y el histograma son en esencia lo mismo, ambas representan la distribución de



frecuencias. Un histograma es sólo una gráfica de barras sin separación entre las barras (Anderson et al., 2008).

Un diagrama de dispersión es una representación gráfica de la relación entre dos variables cuantitativas, la línea de tendencia es una línea que da una aproximación de la relación. La relación no es perfecta, ya que los puntos no trazan una línea recta. Sin embargo, el patrón que siguen los puntos y la línea de tendencia indica que la relación es positiva (figura 5).

**Figura 5. Representación de los datos**



Fuente: Anderson et al. (2008).

Es importante entender la distinción entre la función de gráficos de alto y bajo nivel. Una función de gráficos de alto nivel comienza un nuevo gráfico, establece la escala, dibuja algunos adornos, título y etiquetas. A continuación, presentamos algunas funciones de alto nivel (tabla 10).

**Tabla 10. Funciones gráficas de alto nivel**

|                |                                                                      |
|----------------|----------------------------------------------------------------------|
| <i>plot</i>    | Función de trazado genérico                                          |
| <i>boxplot</i> | Crea un diagrama de caja con tres cuartiles y la desviación estándar |
| <i>hist</i>    | Crea un histograma                                                   |
| <i>qqnorm</i>  | Crear un diagrama cuantil-cuantil (Q-Q)                              |
| <i>curva</i>   | Graficar una función                                                 |

Una función de gráficos de bajo nivel no puede comenzar un nuevo gráfico, más bien, agrega algo a uno existente: puntos, líneas, texto y adornos (tabla 11).

**Tabla 11. Funciones gráficas de bajo nivel**

|                 |                           |
|-----------------|---------------------------|
| <i>points</i>   | Añade puntos              |
| <i>lines</i>    | Agrega líneas             |
| <i>abline</i>   | Agrega una línea recta    |
| <i>segments</i> | Agrega segmentos de línea |
| <i>polygon</i>  | Añade un polígono cerrado |
| <i>text</i>     | Adhiere un texto          |

En este capítulo descubriremos el polimorfismo en *R*. Una función polimórfica o la función genérica es aquella cuyo comportamiento cambia según el tipo de argumento. La función de trazado es polimórfica, por lo que la gráfica (*x*) produce resultados diferentes según la función que se ingrese en *x*, vector, factor, *data.frame*, modelo de regresión podría ser lineal, tabla o lo que sea. Es imprescindible tener en cuenta el tipo de argumentos, ya que ayudará a comprender y recordar por qué la función se comportó como lo hizo.

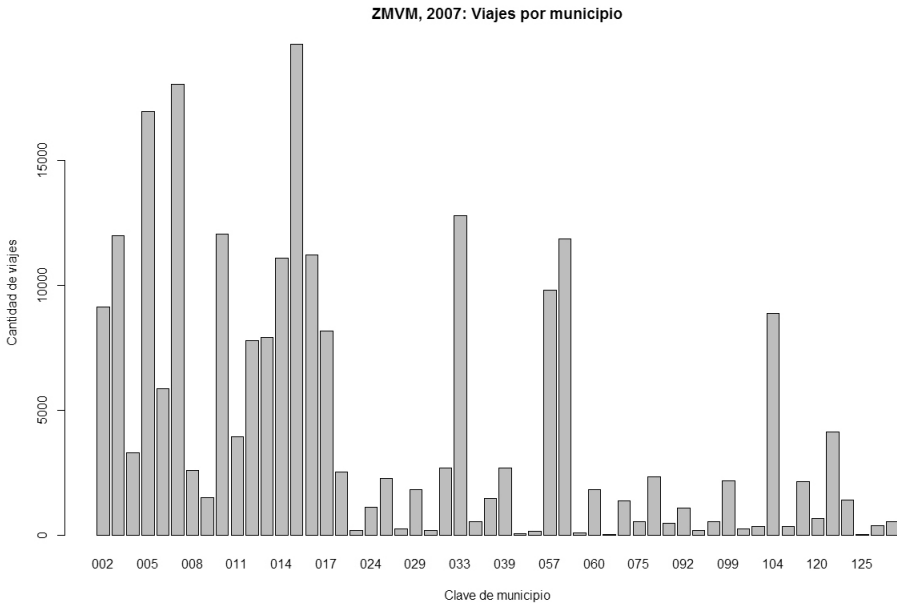
Al convocar un plot:

- Use *main* para titular el gráfico.
- Use el argumento *xlab* para una etiqueta del eje *x*.
- Use el argumento *ylab* para una etiqueta del eje *y*, de tal suerte que el resultado se ejemplifica de la siguiente forma:

```
plot(x, main = "Titulo del grafico", xlab = "Etiqueta del eje X", ylab
= "Etiqueta del eje Y")
```

## Ejemplo de gráficos

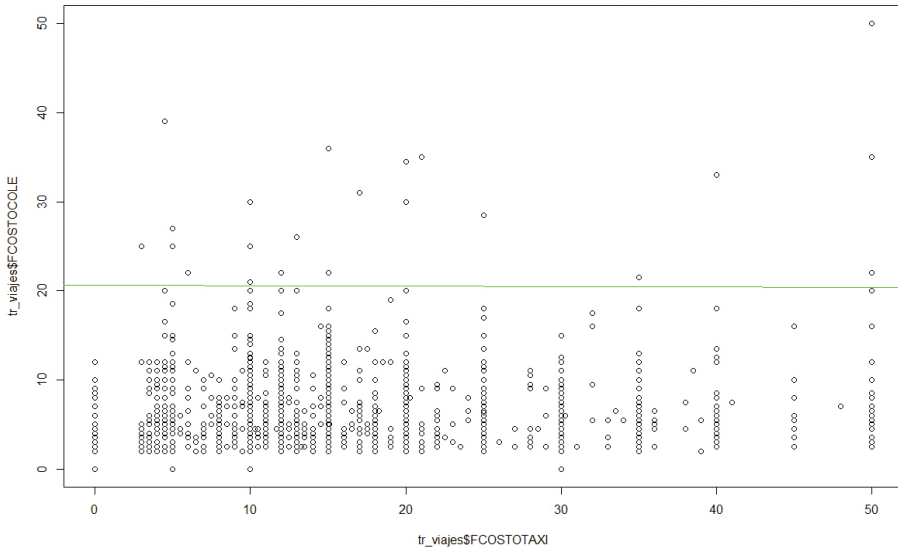
```
plot(tr_viajes$CMUNICIPIO, main = "ZMVM, 2007: Viajes por municipio",
xlab = "Clave de municipio", ylab = "Cantidad de viajes")
```



Para usar diagramas de dispersión, *plot*, es la función de trazado. *Plot*, una función genérica, puede trazar muchos tipos diferentes de objetos, incluidos vectores, *data.frame*, tablas y series de tiempo. Para gráficos de dispersión simples con dos vectores, se recurre a *plot.default*. Veamos un ejemplo graficando la dispersión en el costo de viaje en taxi y el costo de viaje en colectivo de la EOD-2007, la gráfica viene acompañada de su línea de tendencia derivada de un modelo lineal que busca establecer una relación entre ambas variables (figura 6).

```
plot(tr_viajes$FCOSTOTAXI, tr_viajes$FCOSTOCOLE,
 xlim=c(0,50), ylim=c(0,50))
rel<-lm(tr_viajes$FCOSTOTAXI ~ tr_viajes$FCOSTOCOLE)
abline(rel, col="green")
```

**Figura 6. Diagrama de dispersión entre el costo del colectivo y costo de taxi registrada en la EOD, 2007**



Puede utilizar la función *locator* para indicar las coordenadas del punto específico o de los puntos. Trace el punto, a continuación, tipo localizador (1). Haga clic en un punto en las ventanas abiertas. En *RStudio* debe oprimir el botón de *Finish* que se localiza en la ventana de *plots* para imprimir los resultados, o bien, presionar el botón ESC que se localiza en su teclado.

```
locator()
$`x`
[1] 17.02225

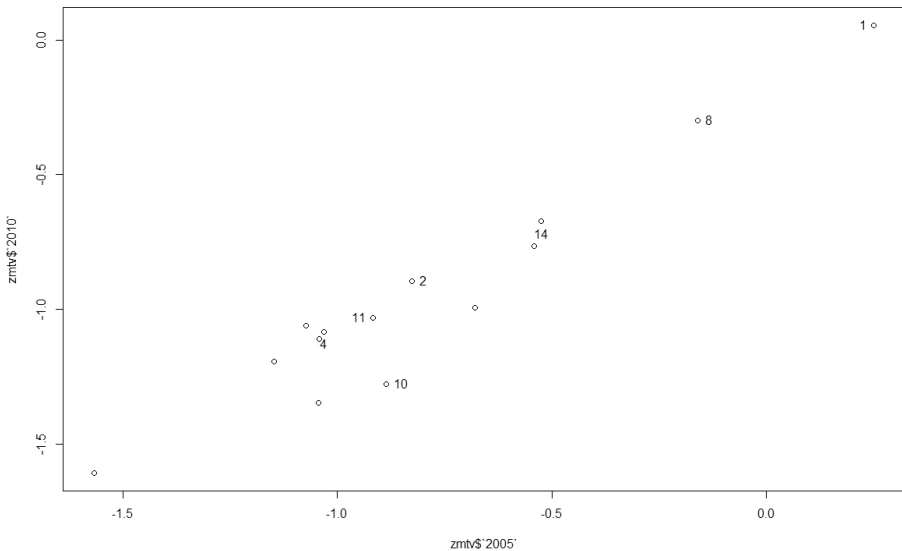
$y
[1] 993.2731
```

Otra característica importante para los puntos de identificación es *identify*. Esta función se puede utilizar para el marcador de etiqueta en punto. Para

utilizar el identificador, *identify*, con los datos del Índice de Rezago Social en la ZMTV, se hace de la siguiente manera (figura 7).

```
identify(zmtv$`2005`, zmtv$`2010`)
[1] 1 2 4 8 10 11 14
```

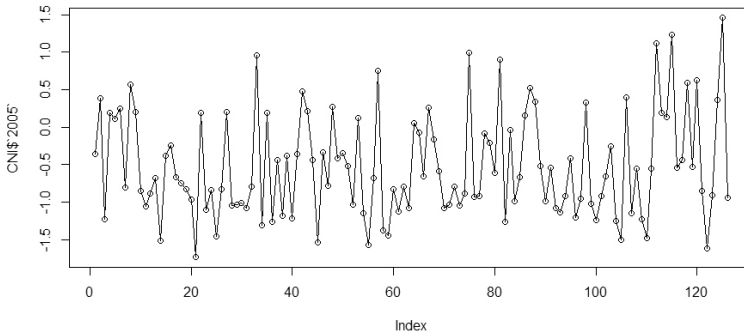
**Figura 7. Diagrama de dispersión  
del Índice de Rezago Social en la ZMTV, 2005-2010**



Cada punto de los señalados en la figura 7 se refiere a uno de los 14 municipios que comprende la región. Ahora, vamos a graficar el Índice de Rezago Social en todos los municipios (125) del Estado de México para el año 2005 (figura 8).

Puede que los argumentos de *matplot* tengan los mismos nombres de las declaraciones estándar que *par*, sin embargo, debido a que *matplot* genera múltiples gráficos al mismo tiempo, estos argumentos se pueden especificar como vectores de varios tipos de bits.

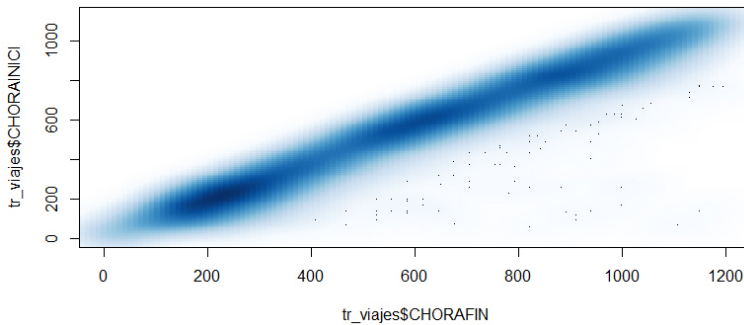
**Figura 8. Diagrama de dispersión del Índice de Rezago Social en el Estado de México, 2005**



Si va a dibujar un conjunto amplio de puntos, a continuación, puede preferir la función *smoothScatter*, que multiplica la densidad de los puntos por recuadro, diferentes regiones del cuadro de diferentes sombras, dependiendo de la densidad de puntos en cada región. Vamos a utilizar los datos de la EOD-2007 para viajes, la cual tiene 232,317 registros y 29 variables. Tal como se muestra en la figura 9, podemos observar que en la ZMVM los viajes inician antes de las 2:00 de la mañana y alcanzan su cumbre al medio día.

```
smoothScatter(tr_viajes$CHORAFIN, tr_viajes$CHORAINICI)
```

**Figura 9. Hora final y de inicio en viajes, EOD 2007**

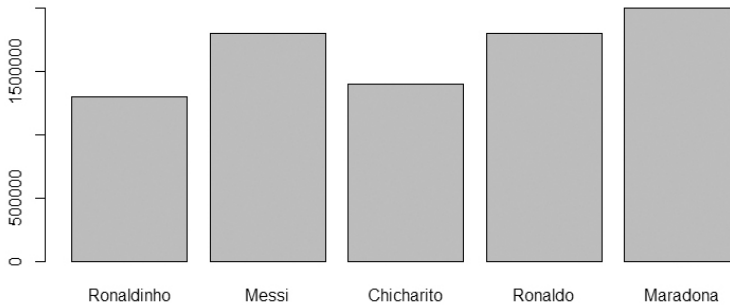


Con un *data.frame* que tiene  $n$  variables, se puede crear un *scatterplot* para cada par de valores en el marco de datos, pruebe la función *pairs*.

La gráfica de barras o por columnas en *R* se genera a través de la función *barplot*, tal como veremos en el siguiente ejemplo (figura 10), donde se representa con valores imaginarios el salario de los principales jugadores de fútbol contemporáneo.

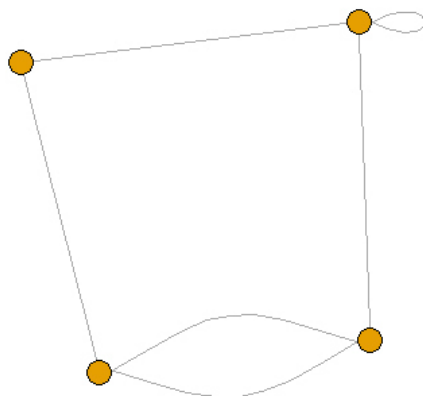
```
barplot(Dolares$Salario, names.arg = Dolares$Nombres)
```

**Figura 10. Gráfico de barras  
con el salario de los jugadores**



Use el paquete *igraph* para crear un gráfico de red. Elabore un vector que contiene elementos pares, luego grafique el resultante, tal como se muestra a continuación y en la figura 11.

```
install.packages("igraph")
library("igraph")
NeTs <- graph(c(1,1, 2,3, 2,4, 1,4, 3,2, 3,1),
 directed=FALSE)
plot(NeTs, vertex.label=NA)
```

**Figura 11. Red**

Fuente: elaboración propia con base en Chang (2013).

Como hemos podido observar, conseguimos graficar de forma sencilla algunas variables contenidas en un *data.frame* o en un vector, la riqueza del *software* radica en los complementos, entre los cuales se encuentran aquellos especializados en la representación gráfica de los datos, con ello se agudiza la manera de ilustrar un estadístico, o bien, se afinan elementos contenidos en el gráfico para ser más explícito.

## Pirámide poblacional

Una pirámide poblacional representa la estructura y composición en un determinado espacio, tres características dotan a la persona al existir: haber nacido en un lugar determinado, tener un sexo y edad, son esas dimensiones las que demográficamente componen las características para el desenvolvimiento espacial. A continuación, realizaremos la estructura de cada uno de los distritos de la Encuesta Origen y Destino (INEGI, 2017) con base en los datos del último Censo de Población y Vivienda del año 2020.

Existen dos librerías importantísimas para transformar los datos y acoplarlos en un lugar diferente de las dimensiones de un *data.frame*: *dplyr* y *tidyverse*,



los cuales, al saber manejarlos, permitirán mantener la base de datos intacta y generar otras observaciones apoyados del pivote “%>%”, que continúa con un comando para seleccionar algún fragmento de la base de datos, *select()*, renombrar las observaciones y variables por medio de *rename()* (Xie, 2016; 2017; Xie et al., 2018).

En el sentido de una pirámide poblacional, el Censo ordena la información etaria como una variable y las observaciones corresponden a una respectiva escala geográfica, no obstante, para esta práctica se necesita organizar los datos de tal forma que los registros se ordenen por la edad y género y la cantidad resultante será la variable, dicha actividad es posible de realizar a través de la función *gather()*, tal como se muestra en el ejercicio que continúa, en sentido contrario, si tenemos como observación el género de un conjunto de personas, la función *spread* nos permite formar las dos variables formales del género masculino y femenino.

Continuando en la administración de datos, en otras ocasiones es necesario combinar los datos de dos variables, la función *unite()* conviene para pegar los valores en una sola. La acción contraria a *unite* es *separate()*, que permite disgregar información de una variable. Los valores también pueden ser reconfigurados, resignificados a través de la función *mutate()* (Xie, 2016; 2017; Xie et al., 2018).

Para este momento es indispensable la descarga de la librería *Rmarkdown*, que permite realizar presentaciones, documentos, respuestas por *e-mail*, realizar gráficos creativos con los principales conceptos de un libro, correlaciones gráficas entre diversos fenómenos y elaboración de diagramas.

Acudir al Censo de Población y Vivienda del año 2020 y descargar la información poblacional de algún estado, posteriormente, elegir cinco municipios que permitan construir un gráfico comparativo de sus respectivas estructuras poblacionales, tal como se muestra a continuación y se plasma en la figura 12. Cabe señalar que los valores poblacionales están en razón de mil personas, así que cuando en el gráfico se observen los valores cinco o diez, significa cinco mil y diez mil, respectivamente.

Sintaxis para elaborar una pirámide poblacional. En este caso se realizaron, aproximadamente, 195 que corresponden a los distritos de la EOD 2017

(INEGI, 2017), los cuales se localizan dentro de 76 municipios que corresponden a la Zona Metropolitana del Valle de México, asimismo, se conformó la información a través de la adición de las 196,129 manzanas que componen las estructuras espaciales de mayor rango.

Es importante señalar que para cada acción que requiera de algún gráfico o cálculo, es necesario abrir un segmento *Chuck*, indicando la apertura y cierre por medio de tres apóstrofes, como los que se muestran al principio y fin de la sintaxis.

```

```{r fig.height=20, fig.width=35}
library(readxl)
Estructuradistritos <- read_excel("C:/Recobrar/EOD2017/EOD2017/
Mapa/Piramide_distritos/Distritos_población.xlsx")
library(dplyr)
library(tidyverse)
library(ggplot2)
pdf("C:/Recobrar/EOD2017/EOD2017/Mapa/Piramide_distritos/Dis-
tritos_movilidad.pdf", width=20, height=35)
DistritosEstructuraM<-Estructuradistritos%>%select(Desc,      M_0a4,
M_5a9, M_10a14, M_15a19, M_20a24, M_25a29, M_30a34, M_35a39,
M_40a44, M_45a49, M_50a54, M_55a59, M_60a64, M_65a69, M_70a74,
M_75a79, M_80a84, `M_85y+`) %>% rename(
  "00-04"=M_0a4, "05-09"=M_5a9, "10-14"=M_10a14, "15-19"=M_15a19,
  "20-24"=M_20a24, "25-29"=M_25a29, "30-34"=M_30a34, "35-39"=M_35a39,
  "40-44"=M_40a44, "45-49"=M_45a49, "50-54"=M_50a54, "55-59"=M_55a59,
  "60-64"=M_60a64, "65-69"=M_65a69, "70-74"=M_70a74, "75-79"=M_75a79,
  "80-84"=M_80a84,
  "85-+"=`M_85y+`) %>% gather(Edad, N, 2:18) %>%
mutate(Sexo="Mujeres")

```

```

DistritosEstructuraH<-Estructuradistritos%>%select(Desc,      H_0a4,
H_5a9, H_10a14, H_15a19, H_20a24, H_25a29, H_30a34, H_35a39,
H_40a44, H_45a49, H_50a54, H_55a59, H_60a64, H_65a69, H_70a74,
H_75a79, H_80a84, `H_85y+`) %>% rename(
  "00-04"=H_0a4, "05-09"=H_5a9, "10-14"=H_10a14, "15-19"=H_15a19,
  "20-24"=H_20a24, "25-29"=H_25a29, "30-34"=H_30a34, "35-39"=H_35a39,
  "40-44"=H_40a44, "45-49"=H_45a49, "50-54"=H_50a54, "55-59"=H_55a59,
  "60-64"=H_60a64, "65-69"=H_65a69, "70-74"=H_70a74, "75-79"=H_75a79,
  "80-84"=H_80a84,
  "85-+"=`H_85y+`) %>% gather(Edad, N, 2:18) %>%
  mutate(Sexo="Hombres")
#Base marco para la estructura poblacional de los municipios de la ZMVM
DistritoESTRUCTURA2020<-rbind(DistritosEstructuraM,
DistritosEstructuraH)%>%mutate(PMIL=N/1000)
#Piramide
ggplot(DistritoESTRUCTURA2020, aes(x= Edad, fill=Sexo,
y=ifelse(test = Sexo=="Hombres",
yes = -PMIL, no=PMIL)))+
  geom_bar(stat = "identity")+
  scale_y_continuous(labels = abs, limits =
max(DistritoESTRUCTURA2020$PMIL)*c(-1,1))+
  labs(y="Población")+
  coord_flip()+
  facet_wrap(~Desc)
dev.off`
`

```

Violin plots comparing the distributions of 20 variables for two groups: Normal (red) and Abnormal (teal). The variables are listed on the left and top of the grid. The plots show the density of each variable for both groups, with the 'Normal' group generally showing higher density in the center and the 'Abnormal' group showing more spread or skewness. A legend on the right indicates 'Normal' in red and 'Abnormal' in teal.

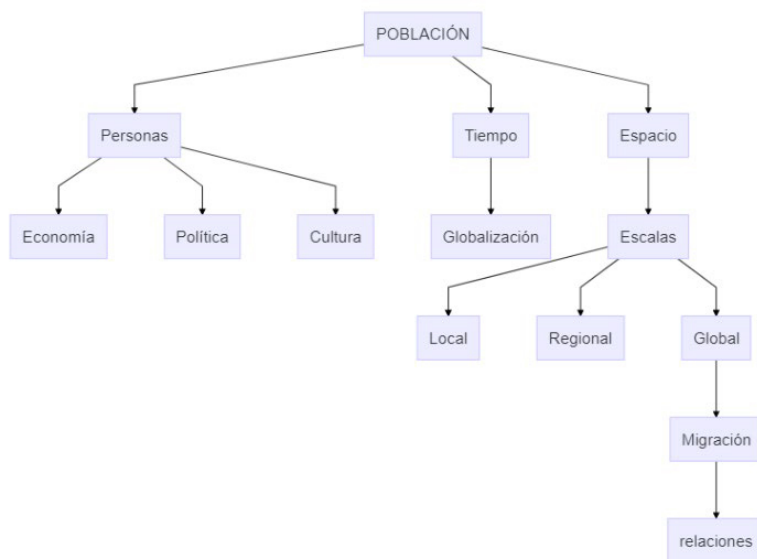
Fuente: elaboración propia con base en INEGI (2017; 2021) y Preston et al. (2000).

Elaboración de un diagrama

En ciencias sociales, es hiper importante consolidar el conocimiento a través de la exposición temática, presentar datos a los compañeros apoyados con gráficos e imágenes que conecten con los receptores, para ello, la librería *Diagrammer* es fundamental. La relación es un poco similar a la estructura de una red, sin embargo, un diagrama resume la información con la finalidad de desarrollar visual y verbalmente las conexiones y vínculos entre ésta.

A continuación, replique en un documento *Rmarkdown* la siguiente sintaxis para obtener el mismo resultado.

```
```{r}
library(DiagrammeR)
DiagrammeR::mermaid(“
graph TB
POBLACION-->Personas
POBLACION-->Tiempo
POBLACION-->Espacio
subgraph one
Personas-->Economia
Personas-->Politica
Personas-->Cultura
Tiempo-->Globalizacion
Espacio-->Escalas
subgraph two
Escalas-->Local
Escalas-->Regional
Escalas-->Global
subgraph three
Global-->Migracion
Migracion-->relaciones
”)
```
```

Figura 13. Mapa conceptual**Bibliografía recomendada: profundizar con los autores**

- Abedin, J. y Kumar, D. K. (2015). *Data Manipulation with R* (2a Ed.). Packt.
- Abedin, J. y Mittal, H. (2014). *R Graphs Cookbook* (2a Ed.). Packt.
- Adler, J. (2012). *R in a nutshell: a desktop quick reference* (2a Ed.). O'Reilly Media.
- Altham, P. (2012). *Introduction to Statistical Modelling in R*. University of Cambridge.
- Arnold, T. y Tilton, L. (2015). *Humanities data in R: exploring networks, geospatial data, images, and text*. Springer.
- Auerbach, C. y Zeitlin, W. (2015). *Making your case: using R for program evaluation*. Oxford University Press.
- Baayen, R. (2008). *Analyzing linguistic data: a practical introduction to statistics using R*. Cambridge University Press.
- Barker, T. (2013). *Pro Data Visualization using R and JavaScript*. Apress.
- Burns, P. (2012). *The R Inferno*. Lulu.com

- Chang, W. (2013). *R Graphics Cookbook*. O'Reilly Media.
- Chiu, Y.-W. (2015). *Machine learning with R cookbook*. Packt.
- Frery, A. y Perciano, T. (2013). *Introduction to image processing using R: learning by examples*. Springer.
- Gohil, A. (2015). *R data visualization cookbook: over 80 recipes to analyze data and create stunning visualizations with R*. Packt.
- Grolemund, G. (2014). *Hands-on programming with R: write your own functions and simulations*. O'Reilly Media.
- Horton, N. y Kleinman, K. (2015). *Using R and RStudio for data management, statistical analysis and graphics* (2a Ed.). CRC.
- Kabacoff, R. (2015). *R in action: data analysis and graphics with R* (2a Ed.). Manning.
- Loo M. van der y Jonge, E. (2012). *Learning RStudio for R Statistical Computing*. Packt.
- Micheaux, P., Drouilhet, R. y Lique, B. (2013). *The R software: fundamentals of programming and statistical analysis*. Springer.
- Xie, Y. (2016). *Bookdown: authoring books and technical documents with R markdown*. CRC.
- Xie, Y. (2017). *Dynamic Documents with R and knitr*. Chapman & Hall/CRC.
- Xie, Y., Allaire, J. J. y Grolemund, G. (2018). *R Markdown: the definitive guide*. CRC Press.
- Yu-Wei, C. (2015). *Machine learning with R cookbook*. Packt.

CAPÍTULO 4

ANÁLISIS DE DATOS

Gran parte del análisis estadístico se basa en describir la relación entre dos variables. Por ejemplo:

- ¿Cómo el volumen de siembra (alta, media o baja) altera el rendimiento del cultivo?
- ¿Cómo se relaciona el precio de la vivienda con el tamaño del lote?
- ¿Cómo se relacionan la altura y el tamaño del pie?
- ¿Es diferente la incidencia de enfermedades del corazón para hombres y mujeres?
- ¿Cómo determinar la incidencia por día y hora de la semana en que se presentan más casos de síntomas cardíacos?

Test por tipo de variable

En el mundo cotidiano, en general, nos vamos a enfrentar con dos variables cualitativas; una cualitativa y una variable cuantitativa, y dos variables cuantitativas. Vamos a presentar qué tipos de pruebas estadísticas se pueden realizar según la clase de variables (tabla 12).

Tabla 12. Aplicación de test por tipo de variable

| Cualitativas | Cuantitativas | Cualitativas-Cuantitativas |
|------------------------|---------------|----------------------------|
| Chi Cuadrada (X^2) | Correlación | t-test |
| | Regresión | ANOVA |

Fuente: elaboración propia con base en Anderson et al. (2008).

Cabe señalar que cuando mixturamos variables, el orden es importante, en un test se sustituye la variable nominal en el lugar de la x (Kabacoff, 2015; James et al., 2013; Beaujean, 2014; Pathak, 2014). Las funciones para cada uno de los antes mencionados se presentan a continuación.

- Chi Cuadrada *chisq.test()*
- Correlación *cor()*
- Regresión *lm()*
- t.test *t-test()*
- ANOVA *lm()* o *oneway.test()*

Para comprender y analizar las relaciones que se establecen en las diferentes pruebas que se han expuesto, es necesario remitirnos a las medidas de tendencia central, posteriormente, retomaremos el tema.

Medidas de tendencia central y distribución

La medida de localización más importante es la media, promedio de una variable, que proporciona una medida de localización central de todos los datos, se obtiene sumando el valor de todas las observaciones y dividiendo entre el número de observaciones (Mailund, 2017; Horton y Kleinman, 2015; Cano et al., 2012).

La mediana es otra medida de localización central. Es el valor que se ubica en medio de los datos ordenados de menor a mayor. Cuando la cantidad de observaciones es impar, la mediana es el valor de en medio. Cuando la cantidad de observaciones es par, no hay número de en medio, por lo que es definida como el promedio de las dos observaciones de en medio. Existen variables que pueden inflarse por algunos valores, en tales casos, la mediana es la medida de

localización idónea, ejemplo de ello son los ingresos o el valor de los inmuebles (Anderson et al., 2008; Agresti, 2002; 2015; Crawley, 2015; 2012).

La moda es el valor que se presenta con mayor frecuencia. Hay ocasiones que la frecuencia mayor se presenta con dos o más valores distintos. En este caso, ocurre que hay más de una moda. Si contiene dos modas, se dice que son bimodales, más de dos, multimodales (Pathak, 2014; James et al., 2013).

Un percentil aporta información acerca de la dispersión de los datos en el intervalo que va del menor al mayor. Con frecuencia, es conveniente dividir los datos en cuatro partes. Los cuartiles han sido definidos como el percentil 25, el percentil 50 y el percentil 75 (Pathak, 2014; Anderson et al., 2008). ¿El segundo cuartil y la mediana arrojan el mismo resultado de cualquier conjunto de observaciones?

R incluye una variedad de funciones para calcular estadísticas de resumen. Para calcular la media de un vector, use la función *mean*. Puede calcular mínimos con la función *min*, los máximos con la función *max*. Para devolver el rango intercuartílico, la diferencia entre los valores del percentil 25 y 75, use la función *IQR*. Ejemplos:

```
> mean(zmtv$`2010`)
[1] -0.9491579

> max(zmtv$`2010`)
[1] 0.05331

> min(zmtv$`2010`)
[1] -1.60753

> range(zmtv$`2010`)
[1] -1.60753 0.05331

> quantile(zmtv$`2010`, probs = c(0, 0.25, 0.50, 0.75, 1))
0% 25% 50%
-1.6075300 -1.1728550 -1.0460950
75% 100%
-0.7975275 0.0533100
```

```
> fivenum(zmtv$`2010`)
[1] -1.607530 -1.193480 -1.046095
[4] -0.765040 0.053310

> IQR(zmtv$`2010`)
[1] 0.3753275
```

La función más conveniente para ver información resumida es *summary*. Ésta arroja información sobre el mínimo, primer cuartil, mediana, promedio (segundo cuartil), tercer cuartil y, finalmente, el máximo valor, es decir, la distribución de datos de nuestras observaciones (Pathak, 2014). Es una función genérica que se aplica en *data.frames*, matrices, tablas, factores y objetos originales (James et al., 2013). Como ejemplo, echemos un vistazo a la salida del resumen del Índice de Rezago Social del año 2010 (*zmtv\$2010*).

```
summary(zmtv$`2010`)
  Min. 1st Qu. Median
-1.60753 -1.17285 -1.04609
  Mean 3rd Qu. Max.
-0.94916 -0.79753 0.05331
```

Como se puede ver, *summary* presenta información sobre cada variable en el *data.frame*. En el ejemplo se constatan, para valores numéricos, las siguientes medidas: mínimo, primer cuartil, mediana, media, tercer cuartil y máximo. Para los factores, el resumen muestra el recuento de los valores más recurrentes.

La varianza es una medida de variabilidad que utiliza todos los datos. Está basada en la diferencia entre el valor de cada observación y la media. La desviación estándar se define como la raíz cuadrada positiva de la varianza. La desviación estándar es más fácil de interpretar que la varianza debido a que la desviación estándar se mide en las mismas unidades que los datos. La covarianza es una medida de asociación lineal entre dos variables (Horton y Kleinman, 2015; Cano et al., 2012; Pathak, 2014; Kabacoff, 2015; Anderson et al., 2008). Otras funciones se pueden observar en la tabla 13.

En consecuencia, se presenta un ejemplo donde se aplicó la función *var* para conocer la varianza entre el índice de rezago social para los municipios del

Estado de México entre los años 2000 y 2010. Asimismo, se expone el modelo para calcular la covarianza de dos variables (Pathak, 2014).

```
var(zmtv$`2000`, zmtv$`2010`)
[1] 0.2240154
```

```
cov(x,y=NULL, use="everything", method = c("pearson", "kendall",
"spearman"))
```

Tabla 13. Funciones para observar la tendencia de las observaciones

| Función | Descripción |
|----------|--|
| sd() | Desviación estándar |
| var() | Varianza |
| median() | Mediana |
| max() | Valor máximo |
| min() | Valor mínimo |
| range | Valor máximo y mínimo |
| length() | Número de elementos de un vector |
| cummin() | Mínimo acumulado |
| cummax() | Máximo acumulado |
| diff | Diferencias entre elementos sucesivos de un vector |

Fuente: elaboración propia con base en Beaujean (2014) y Pathak (2014).

Distribución de probabilidad

Muchos trabajos con pruebas estadísticas calculan una medida asociada y luego comparan los resultados del test con el valor de una distribución acústica. *R* proporciona un conjunto de funciones para calcular las densidades, distribuciones y cuantiles (Crawley, 2015; 2012). También puede generar valores aleatorios a partir de estas distribuciones.

Aquí se describe cómo usar la sintaxis, empleando la distribución normal como ejemplo, y luego enumera la mayoría de las operaciones incluidas en la

biblioteca *R stast* (Kabacoff, 2015; Beaujean, 2014; Pathak, 2014; Mailund, 2017).

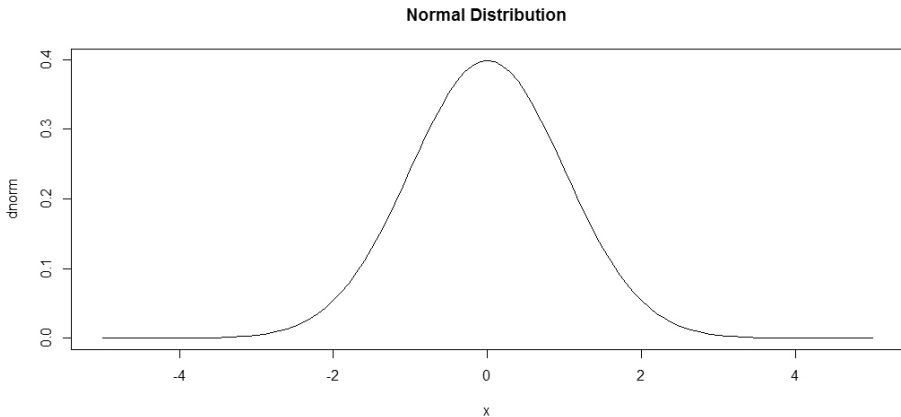
Para encontrar la densidad de probabilidad en un valor dado, la función *dnorm* es la indicada:

```
dnorm(x, mean=0, sd=1, log=FALSE)
```

Los argumentos para esta función son bastante intuitivos: *x* especifica el valor para evaluar la densidad; *mean* la media de la distribución; *sd* la desviación estándar, y *log* se define en caso de devolver la densidad bruta (*log* = *FALSE*) o el algoritmo de la densidad (*log* = *TRUE*). Como ejemplo, puede trazar la distribución normal que se presenta en la siguiente línea (figura 14).

```
plot(dnorm, -5, 5, main="Normal Distribution")
```

Figura 14. Simulación de la distribución normal



Fuente: elaboración propia con base en Pathak (2014) y Beaujean (2014).

La función para la distribución normal es *pnorm*:

```
pnorm(q, mean=0, sd=1, lower.tail=TRUE, log.p=FALSE)
```

Se recurre a la función de distribución para determinar la probabilidad de que un valor seleccionado aleatoriamente de la distribución sea menor o igual a q . Específicamente, devuelve $p = Pr(x \leq q)$. El valor q se especifica por el argumento, la media es *mean*, y la desviación estándar por *sd*. Si desea registrar (p), especifique *log.p = TRUE*. Por defecto, *lower.tail = TRUE*, si prefiere $Pr(x > q)$, especifique *lower.tail = FALSE*. Aquí se muestran algunos ejemplos de *pnorm*:

La media es cero, la distribución normal es simétrica, entonces, la probabilidad ($q \leq 0$) es (.5).

```
pnorm(0)
```

```
[1] 0.5
```

¿Cuál es la probabilidad de que el valor sea inferior a 2 desviaciones estándar por debajo de la media? La respuesta es 0.02275013.

```
pnorm(-2)
```

```
[1] 0.02275013
```

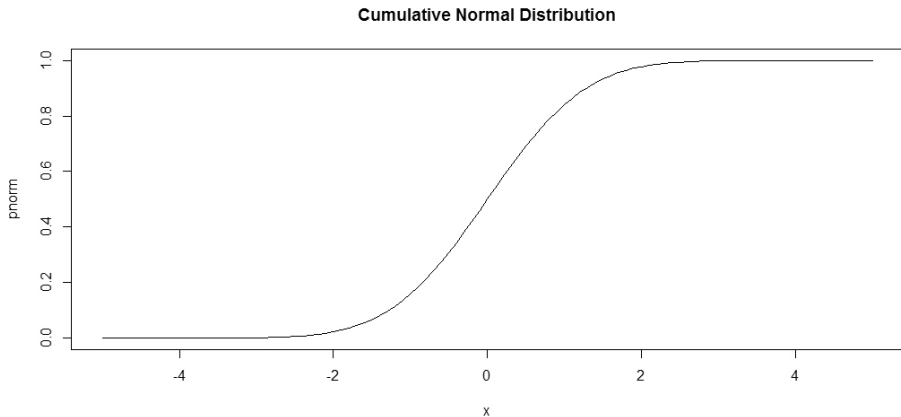
¿Cuál es la probabilidad de que el valor esté dentro de 1.96 desviaciones estándar de la media? Es 0.04999579.

```
pnorm(-1.96, lower.tail = TRUE) + pnorm(1.96, lower.tail  
= FALSE)
```

```
[1] 0.04999579
```

Es posible trazar la distribución normal acumulada con el siguiente comando (figura 15).

```
plot(pnorm, -5, 5, main="Cumulative Normal Distribution")
```

Figura 15. Probabilidad acumulada bajo la normal

Como se indicó anteriormente, p es la probabilidad, donde $p = Pr(x < q)$; *mean*, la media de la distribución; *sd* la desviación estándar, y *lower.tail* si $p = Pr(x \leq q)$, *lower.tail*=*TRUE*; o $p = Pr(x > q)$, *lower.tail* = *FALSE*. El argumento *log.p* especifica si el valor de entrada es el algoritmo de p , *log.p* = *TRUE*, o simplemente p , *log.p* = *FALSE* (Beaujean, 2014; Kabacoff, 2015). Finalmente, vamos a encontrar los lados izquierdo y derecho de un intervalo de confianza de 95 por ciento.

```
qnorm(0.02275013)
```

```
[1] -2
```

```
qnorm(log(0.02275013), log.p=TRUE)
```

```
[1] -2
```

```
qnorm(pnorm(-2))
```

```
[1] -2
```

```
c(qnorm(.37), qnorm(.63))
```

```
[1] -0.3318533 0.3318533
```

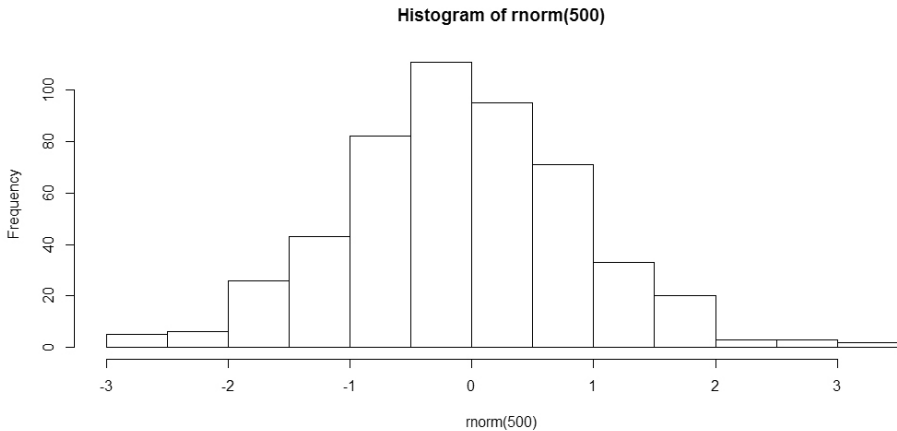

Es posible generar números aleatorios tomados de la distribución normal. Seleccionarlos de una distribución específica puede ser útil para probar funciones estadísticas, ejecutar simulaciones, en métodos de muestreo y en muchos otros contextos (Nord, 2019; Burns, 2012; Grolemond, 2014). Para hacer esto en *R*, use la función *rmnorm*.

```
rmnorm(n, mean = 0, sd=1)
```

Por ejemplo, puede generar 500 valores seleccionados al azar de una distribución normal con el comando *rmnorm(500)*. Podría trazar estos con una expresión como la siguiente (figura 16).

```
hist(rmnorm(500), breaks = 20)
```

Figura 16. Histograma de 500 valores al azar de una distribución normal



Fuente: elaboración propia con base en Teetor (2011).

Test estadísticos

Muy a menudo, al analizar datos, se desea saber si dos variables están correlacionadas. Informalmente, la correlación responde a la cuestión: cuando

aumentamos o disminuimos x , aumenta o disminuye y , ¿por cuánto? Formalmente, el coeficiente de correlación mide la dependencia lineal entre dos variables aleatorias. Las medidas de correlación oscilan entre -1 y 1, donde 1 significa que una variable es una función lineal positiva de la otra, 0 significa que las dos variables no están correlacionadas en absoluto, y -1 significa que una variable es una función lineal negativa de la otra, las dos en direcciones completamente opuestas (Beaujean, 2014; Mailund, 2017; Agresti, 2015; 2002; James et al., 2013; Kabacoff, 2015).

La correlación proporciona una medida de asociación y no necesariamente de causalidad. Que el coeficiente entre dos variables sea alto no significa que los cambios en una de las variables ocasionen modificaciones en la otra, ejemplo de ello es una evaluación de calidad en un servicio automotriz cuyos precios tengan una relación positiva, no obstante, aumentar los precios no provocará que las evaluaciones mejoren (Anderson et al., 2008).

Se puede calcular las correlaciones en dos vectores asignados a los argumentos x y, un *data.frame* asignado a " x " con " y " = NULL, o una matriz asignada a x con y = NULL. Si especifica una matriz o *data.frame*, entonces *cor* calculará la correlación entre cada par de variables y devolverá una matriz de resultados (Crawley, 2015; 2012). Veamos un ejemplo con la tabla de hogares de la EOD-2007, la cual tiene 49,488 registros y 44 variables.

```
cor(moviv$NTOTHOGVIV, moviv$NPERSONAST, method = "spearman")
[1] 0.2429822
```

En el ejemplo de correlación, la medida indica que existe una correlación modesta entre las dos variables, total de hogares en vivienda (NTOTHOGVIV) y número de personas totales en vivienda (NPERSONAST) registradas en la EOD-2007.

Una idea estrechamente relacionada es la covarianza. La covarianza se define como medida descriptiva de la asociación entre dos variables:

$$\text{Cov}(x, y) = E[(x - E[x])(y - E[y])]$$

```
cov(moviv$NTOTHOGVIV, moviv$NPERSONAST)
[1] 0.1863358.
```

Muchos problemas de datos se reducen a una prueba estadística. Por ejemplo, se puede querer responder un planteamiento, como: En Veracruz, durante los gobiernos de derecha, ¿el suministro de suero para el cáncer funciona mejor que un placebo? ¿El diseño de la nueva página web para el Sistema de Administración Tributaria promueve que las grandes compañías no eludan impuestos? En México, ¿la producción inmobiliaria es más especulativa que un instrumento financiero?

Muchas de las pruebas estadísticas en este capítulo usan un paradigma de estadística inferencial probado por el tiempo. En el paradigma, tenemos una o dos muestras de datos. También tenemos dos competidores hipótesis, cualquiera de las cuales podría ser razonablemente cierta. Una hipótesis, llamada nula (h_0), significa que no pasó nada: la media se queda sin alterar; el tratamiento no tuvo efecto; la respuesta es la esperada; el modelo no hizo una mejora. La otra, llamada alternativa (h_1), representa que algo sucedió: el tratamiento mejoró la salud de los pacientes; hay una respuesta inesperada; el modelo se ajusta mejor. Queremos determinar cuál de las hipótesis es más probable a la luz de los datos.

1. Para comenzar, suponemos que la hipótesis nula, h_0 , es verdadera.
2. Calculamos una estadística de prueba. Podría ser algo simple, como la media muestral, o podría ser bastante complejo. El requisito crítico es que debemos conocer la distribución de los datos.
3. A partir de la estadística y su distribución podemos calcular un valor p , la probabilidad de un valor estadístico de prueba tan extremo o más extremo que el que observamos, asumiendo que la hipótesis nula es verdadera.
4. Si el valor de p es demasiado pequeño, tenemos pruebas sólidas contra la hipótesis nula. Esto se llama aceptar la hipótesis alterna.
5. Si el valor de p no es pequeño, entonces no tenemos tal evidencia. Esto se llama no descartar la hipótesis nula.

Aquí, aceptamos la hipótesis alterna cuando $p < 0.05$ y no la descartamos cuando $p > 0.05$. Se eligió un nivel de significancia de $\alpha = 0.05$ para definir el borde entre pruebas sólidas e insuficiente evidencia contra la hipótesis nula.

La verdadera respuesta “depende”. El nivel de significancia elegido estriba en el dominio del problema. El límite convencional de $p < 0.05$ funciona para

muchos problemas. En muchas investigaciones, los datos son especialmente ruidosos, por lo que a menudo se satisface con $p < 0.10$. Para alguien trabajando en áreas de alto riesgo, puede ser necesario conseguir $p < 0.01$ o $p < 0.001$.

Infelizmente, las conclusiones correctas no siempre son posibles. Como la prueba de hipótesis se basa en una información muestral, debe tenerse en cuenta que existe la posibilidad de error. Existen dos tipos de errores: cuando se acepta h_0 , si ésta es verdadera, la conclusión es correcta; pero si h_1 es verdadera, se comete un error tipo II, es decir, se acepta h_0 cuando es falsa. Si la conclusión es rechazar h_0 , cuando h_0 es verdadera, se comete un error tipo I; es decir, se rechaza h_0 cuando es verdadera. Pero si h_1 es verdadera, es correcto rechazar h_0 (tabla 14).

Tabla 14. Errores y conclusiones correctas en las pruebas de hipótesis

| | h_0 verdadera | h_1 verdadera |
|------------------|---------------------|---------------------|
| Se acepta h_0 | Conclusión correcta | Error tipo II |
| Se rechaza h_0 | Error tipo I | Conclusión correcta |

Fuente: elaboración propia con base en Anderson et al. (2008).

Al elegir p se controla la probabilidad de cometer un error tipo I. Si el costo de cometer un error tipo I es elevado, los valores pequeños de p son preferibles. Si el costo de cometer un error tipo I no es demasiado elevado, entonces se usan valores mayores para p . A las aplicaciones de la prueba de hipótesis en que sólo se controla el error tipo I, se les llama pruebas de significancia (Anderson et al., 2008).

Generalmente, en las aplicaciones de las pruebas de hipótesis se controla la probabilidad de cometer un error tipo I, no siempre sucede lo mismo con un error tipo II. Por tanto, si se decide aceptar h_0 , no es posible establecer la confianza en esa decisión. Debido a la incertidumbre de cometer el error tipo II al realizar una prueba de significancia, los dedicados a la estadística suelen recomendar que se diga “no se rechaza h_0 ” en lugar de se acepta h_0 . A saber: si no se acepta directamente h_0 , se evita el riesgo de cometer un error tipo II, siempre que no se determine y controle la probabilidad de cometer un error tipo II, no se dirá “se acepta h_0 ”.

Pruebas para datos continuos. T test. Supongamos que se diseñó un experimento para demostrar que algún efecto es cierto. Por ejemplo, que el uso de transporte público reduce los tiempos de viaje. Se han recopilado algunos

datos y ahora es preciso saber si los datos prueban la hipótesis. Una pregunta común es si la media de los datos experimentales es diferente de lo que sería con la hipótesis no verdadera, hipótesis nula o la hipótesis alternativa. Específicamente, suponga que tiene un conjunto de observaciones $x_1, x_2, \dots, x_n$ con una media experimental y quiere saber si la media experimental es diferente de la hipótesis nula, h_0 . Además, suponga que las observaciones se distribuyen normalmente. Para probar la validez de la hipótesis, puede usar una prueba t . En R , se usa la función `t.test` (tabla 15).

```
t.test(x,y=NULL, alternative = c("two.sided", "less", "greater"),
mu=0, paired = FALSE, var.equal = FALSE,
conf.level = 0.95,...)
```

Tabla 15. Argumentos de la función `t.test`

| Argumento | Descripción | Default |
|--------------------------|---|--|
| <code>x</code> | Un vector numérico con observaciones. | |
| <code>y</code> | Un vector numérico de valores de datos, use <code>y = NULL</code> para comparar un único vector de valores con una media de hipótesis nula, <code>mu</code> , o un vector para comparar el vector <code>x</code> con el vector <code>y</code> . | NULL |
| <code>alternative</code> | Un valor de carácter que especifica la hipótesis alternativa. Use <code>alternative = "two.sided"</code> para una distribución de dos colas, <code>alternative = "les"</code> para mínimo, y <code>alternative = "greater"</code> para máximo. | <code>c("two.sided", "less", "greater")</code> |
| <code>mu</code> | Un valor numérico que especifica el valor de la media bajo la hipótesis nula, si se prueba un solo vector de valores, o la diferencia entre los medios, si se comparan los promedios de dos vectores. | 0 |
| <code>paired</code> | Un valor lógico que indica si los vectores están emparejados. | FALSE |
| <code>var.equal</code> | Un valor lógico que indica si se supone que la varianza de los dos vectores es la misma. Si <code>var.equal = TRUE</code> , se usa la varianza combinada. Si <code>var.equal = FALSE</code> , entonces se usa el método Welch. | FALSE |
| <code>conf.level</code> | Intervalo de confianza | 0.95 |

Fuente: elaboración propia con base en Teetor (2011) y Cotton (2013).

Vamos a testear una prueba de hipótesis a través de *t.test*, donde:

h_0 : el Índice de Rezago Social en la Zona Metropolitana del Valle de Toluca permaneció constante en los años 2005 al 2010.

h_1 : el Índice de Rezago Social en la Zona Metropolitana del Valle de Toluca disminuyó de los años 2005 al 2010.

```
t.test(zmtv$`2005`, zmtv$`2010`)
```

```
welch Two Sample t-test
```

```
data: zmtv$`2005` and zmtv$`2010`
```

```
t = 0.90282, df = 25.917, p-value = 0.3749
```

```
alternative hypothesis: true difference in means is not  
equal to 0
```

```
95 percent confidence interval:
```

```
-0.1914483 0.4912583
```

```
sample estimates:
```

```
mean of x mean of y
```

```
-0.7992529 -0.9491579
```

Aquí hay una explicación de la salida de la función *t.test*. Primero, la función nos muestra el estadístico de prueba, $t = 0.90282$, los grados de libertad, $df = 25.917$, y el valor de p calculado para la prueba, $p\text{-value} = 0.3749$. El $p\text{-value}$ significa la probabilidad de que el valor promedio de la muestra fue mayor que -0.1914483 y menor que 0.4912583 . Esto significa que es muy probable que el Índice de Rezago Social en la Zona Metropolitana del Valle de Toluca (ZMTV) se esté achicando en el tiempo, 2005-2010, la media de un año con respecto al siguiente muestra que el rezago se reduce. Por tanto, se acepta la hipótesis alterna. La siguiente línea indica la hipótesis alterna: la verdadera media no es igual a 0, que rechazaríamos en función del resultado de esta prueba. A continuación, la función *t.test* muestra intervalos de confianza de 95% para esta prueba y, finalmente, proporciona la media real.

Para la prueba t de dos muestras también se puede usar una fórmula para especificar una prueba t si los datos están incluidos en un banco de datos y los dos grupos de observaciones son diferenciales por un factor.

Comparando datos emparejados. A veces, se presentan datos emparejados. Por ejemplo, puede tener dos observaciones por tema: una antes de un experimento y una después del experimento. En este caso, usaría una prueba t de comparación. Puede usar la t función de prueba, especificando *paired* = *VERDADERO* para realizar esta prueba.

```
t.test(zmtv$`2005`, zmtv$`2010`, paired = TRUE)
```

Paired t-test

```
data: zmtv$`2005` and zmtv$`2010`
t = 4.6314, df = 13, p-value = 0.0004701
alternative hypothesis: true difference in means is not
equal to 0
95 percent confidence interval:
 0.07997946 0.21983054
sample estimates:
mean of the differences
 0.149905
```

En este caso, podemos ver claramente que los resultados fueron significativos en la inversión de confianza de 95%. Esto no es una gran sorpresa. Es bien sabido que la optimización de la configuración del compilador y los parámetros del sistema pueden marcar una gran diferencia en el rendimiento del sistema. Además, la presentación de resultados optimizados es opcional.

Comparar las varianzas de dos poblaciones. Para ejecutar este procedimiento, *R* incluye la función *var.test* que realiza una prueba F .

```
var.test(zmtv$`2005`, zmtv$`2010`)
```

F test to compare two variances

```
data: zmtv$`2005` and zmtv$`2010`
F = 1.1196, num df = 13, denom df = 13, p-value = 0.8416
alternative hypothesis: true ratio of variances is not
equal to 1
```

```

95 percent confidence interval:
 0.3594309 3.4877191
sample estimates:
ratio of variances
 1.11964

```

Este test reafirma el resultado anterior y señala que no existe una identidad entre las varianzas, por lo que la modificación del Índice de Rezago Social en la Zona Metropolitana del Valle de México para los años 2005 y 2010 se confirma. Para probar que las varianzas en cada uno de los grupos (muestras) son las mismas, puede usar la prueba de Bartlett. En R, esto está disponible a través de la función *bartlett.test*.

Para comparar los promedios en más de dos grupos, puede usar un método llamado análisis de varianza, ANOVA. Los métodos ANOVA son muy importantes para las estadísticas (Kabacoff, 2015; James et al., 2013; Beaujean, 2014; Pathak, 2014; Mailund, 2017; Horton y Kleinman, 2015; Agresti, 2002; Cano et al., 2012). Hagamos un modelo ANOVA de los datos y mostremos un resumen del modelo. Para hacer esto en R, simplemente necesitamos usar la función *aov*.

```

aov(cuadro_31$REG_CASA~cuadro_31$TRABAJO)
Call:
aov(formula = cuadro_31$REG_CASA ~ cuadro_31$TRABAJO)

Terms:
cuadro_31$TRABAJO Residuals
Sum of Squares 27451581966 359427422467
Deg. of Freedom 1 156

Residual standard error: 48000.23
Estimated effects may be unbalanced

```

Con el fin de obtener más información sobre los resultados ANOVA, puede utilizar el *modelo.tables* para imprimir información sobre un objeto *aov*. A menudo, es mejor usar *lm* para ajustarse a un modelo lineal y luego usar la función ANOVA para extraer información sobre el análisis de varianza. Para los

modelos grandes, con frecuencia es más eficiente usar la función de *update*, cambiar un modelo existente en lugar de crear un nuevo modelo de formulario cero (Beaujean, 2014).

En el modelo, lo que se desea observar es la asociación entre los viajes motivados por trabajo y de regreso a casa en la Zona Metropolitana del Valle de México, reportados durante el año 2007. La variable dependiente es viaje con motivo de regreso a casa, y la variable independiente es viajes cuyo motivo es el trabajo.

```
Regres.trabajo<-lm(cuadro_31$REG_CASA~cuadro_31$TRABAJO)
> anova(Regres.trabajo)
Analysis of Variance Table

Response: cuadro_31$REG_CASA
 Df Sum Sq Mean Sq F value
cuadro_31$TRABAJO 1 2.7452e+10 2.7452e+10 11.915
Residuals 156 3.5943e+11 2.3040e+09
Pr(>F)
cuadro_31$TRABAJO 0.0007169 ***
Residuals
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Los cálculos de ANOVA suponen que la varianza no es igual entre los grupos. Cuando desee comprobar el resultado, puede usar la función *onewar.test()* para calcular si dos o más muestras tienen el mismo significado.

Probando la normalidad. Para probar si una distribución se modifica normalmente en *R*, use la prueba de Shapiro-Wilk para la normalidad a través de la función *shapiro.test* (Teetor, 2011) (figura 17).

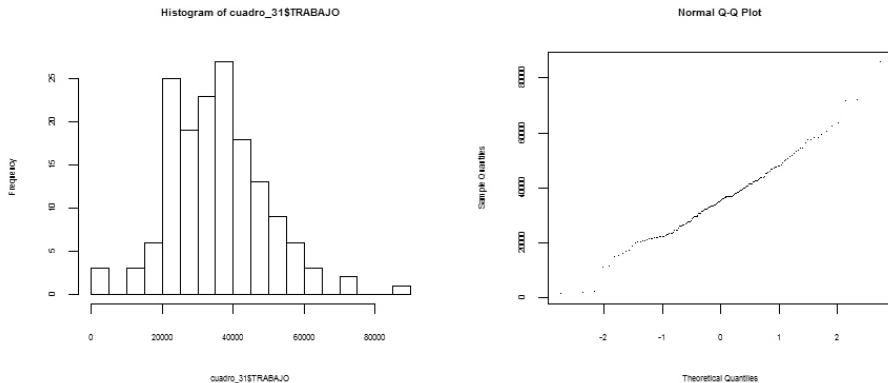
```
shapiro.test(cuadro_31$TRABAJO)

shapiro-wilk normality test

data: cuadro_31$TRABAJO
w = 0.98258, p-value = 0.04389
```

```
qqnorm(cuadro_31$TRABAJO, pch=".")
par(mfcol=c(1,2), ps=6.5)
hist(cuadro_31$TRABAJO, breaks = 25)
qqnorm(cuadro_31$TRABAJO, pch=".")
```

Figura 17. Distribución de viajes por motivo de trabajo en la ZMVM, 2007



Fuente: elaboración propia con base en Beaujean (2014) y Munzert et al. (2015).

La función de correlación y covarianza simplemente calcula el grado de correlación entre pares de vectores, pero no dicen si la correlación es significativa. Si desea verificar si existe una correlación estadísticamente significativa entre dos vectores, puede usar la función *cor.test* (figura 18).

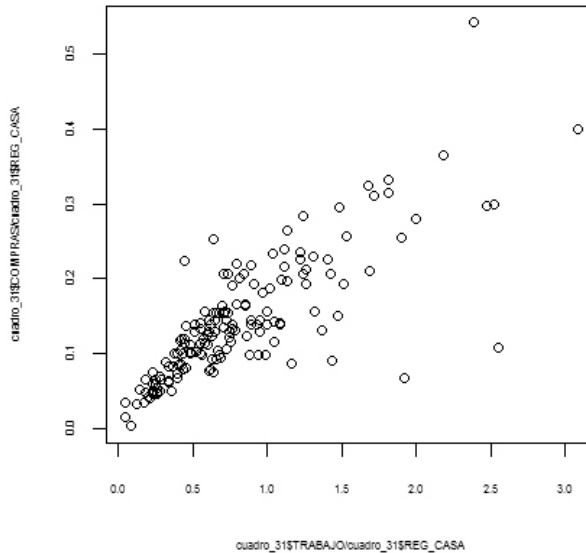
```
cor.test(cuadro_31$TRABAJO/cuadro_31$REG_CASA,
+ cuadro_31$COMPRAS/cuadro_31$REG_CASA)
```

Pearson's product-moment correlation

```
data: cuadro_31$TRABAJO/cuadro_31$REG_CASA and
cuadro_31$COMPRAS/cuadro_31$REG_CASA
t = 16.302, df = 156, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.7278747 0.8451840
```

```
sample estimates:
cor
0.7937992
```

Figura 18. Nube de puntos de la correlación anterior (0.7937)



Fuente: elaboración propia con base en Agresti (2015; 2002).

La prueba muestra que hay una correlación positiva estadísticamente significativa (0.7937992) entre estas dos variables compuestas. Sin embargo, no se debe inferir que existe una relación casual entre las tasas de viajes de regreso a casa por motivo de trabajo y las tasas de viajes de regreso a casa por motivos de compras.

Curiosamente, en la Zona Metropolitana del Valle de Toluca, el Índice de Rezago Social disminuye en todos los municipios, excepto en Almoloya, donde es ligeramente superior la cantidad del índice con respecto a los demás municipios. Podemos usar una prueba de chi-cuadrado para determinar la probabilidad de que esta distribución surja por casualidad. Como se indicó anteriormente, de manera predeterminada, realizamos una prueba de chi cuadrado bajo el supuesto de que la probabilidad real de que un municipio disminuya

su índice viene dada por el vector $p = rep(1 / longitud(x), longitud(x))$, que en este caso es 1/14 para cada municipio. Entonces, nos preguntamos: ¿cuál es la probabilidad de que una distribución al menos tan desequilibrada como la anterior haya surgido por casualidad? Adicionalmente, se muestra la tabla de contingencia con cuatro rangos del rezago.

```
chisq.test(zmtv$Municipio, cut(zmtv$`2010`, breaks = 4))
```

Pearson's Chi-squared test

```
data: zmtv$Municipio and cut(zmtv$`2010`, breaks = 4)
X-squared = 42, df = 39, p-value = 0.3422
```

```
table(zmtv$Municipio, cut(zmtv$`2010`, breaks = 4))
```

```
(-1.61,-1.19] (-1.19,-0.777] (-0.777,-0.362]
(-0.362,0.055]
Almoloya de Juárez 0 0 0 1
Calimaya 0 1 0 0
Chapultepec 1 0 0 0
Lerma 0 1 0 0
Metepec 1 0 0 0
Mexicaltzingo 0 1 0 0
Ocoyoacac 0 1 0 0
Otzolotepec 0 0 0 1
Rayon 0 1 0 0
San Antonio la Isla 1 0 0 0
San Mateo Atenco 0 1 0 0
Toluca 1 0 0 0
Xonacatlan 0 0 1 0
Zinacantepec 0 0 1 0
```

Lo que determina el modelo chi cuadrado para el municipio de Almoloya es que no es casualidad que el Índice de Rezago Social para ese territorio no haya mejorado durante el periodo de comparación. El valor $p = 0.3422$ es

muy superior al esperado (0.05). Sólo dos municipios se localizan en el último rango, Almoloya de Juárez y Otzolotepec.

Un modelo de regresión muestra cómo un valor continuo, llamado variable de respuesta, o la variable dependiente, está relacionada con un conjunto de otros valores, llamados predictores, variables de estímulo o variables independientes. A menudo, se usa un modelo de regresión para predecir valores donde son desconocidos.

A veces, un modelo de regresión simplemente se usa para explicar un fenómeno, pero no para predecir valores. Por ejemplo, un científico podría sospechar que el peso se correlaciona con el consumo de ciertos tipos de alimentos, pero desea ajustarlo por una variedad de factores, como la edad, el ejercicio y la genética. El científico podría usar un modelo de regresión para ayudar a mostrar la relación entre el peso y los alimentos consumidos al incluir otras variables en la regresión.

Para los científicos sociales, un modelo es una forma concisa de describir un conjunto de datos, generalmente, con una fórmula matemática. A veces, el objetivo es construir un modelo predictivo con datos de entrenamiento para predecir valores basados en otros datos. Otras veces, el objetivo es construir un modelo descriptivo que lo ayude a comprender mejor los datos.

R tiene una notación especial para describir las relaciones entre variables. Supongamos que está asumiendo un modelo lineal para una variable y pronosticado a partir de las siguientes observaciones $x_1, x_2, \dots, x_n$ (los estadísticos, comúnmente, se refieren a y como la variable dependiente, entonces, $x_1, x_2, \dots, x_n$ como las variables independientes). En la ecuación, esto implica una relación como la siguiente:

$$y = C_0 + C_1X_1 + C_2X_2 + \dots + C_nX_n$$

En R , se escribiría la relación como $y \sim x_1 + x_2 + \dots + x_n$, que es un objeto de fórmula.

Entonces, intentemos utilizar una regresión lineal para estimar la relación. Usaremos la función `lm` para estimar los parámetros de un modelo lineal. La función `lm` devuelve un objeto de la clase `lm`, que asignaremos a las variables grado de rezago social en la Zona Metropolitana del Valle de Toluca en los años 2000, 2005 y 2010.

```
compara<-lm(zmtv$`2000`~zmtv$`2005`+zmtv$`2010`)
```

```
compara
```

```
Call:
```

```
lm(formula = zmtv$`2000` ~ zmtv$`2005` + zmtv$`2010`)
```

```
Coefficients:
```

```
(Intercept) zmtv$`2005` zmtv$`2010`
```

```
0.02138 0.86623 0.34706
```

```
summary(compara)
```

```
Call:
```

```
lm(formula = zmtv$`2000` ~ zmtv$`2005` + zmtv$`2010`)
```

```
Residuals:
```

```
Min 1Q Median 3Q Max
```

```
-0.17740 -0.01942 0.01794 0.05365 0.10501
```

```
Coefficients:
```

```
Estimate Std. Error t value
```

```
(Intercept) 0.02138 0.06820 0.313
```

```
zmtv$`2005` 0.86623 0.20187 4.291
```

```
zmtv$`2010` 0.34706 0.21360 1.625
```

```
Pr(>|t|)
```

```
(Intercept) 0.75984
```

```
zmtv$`2005` 0.00127 **
```

```
zmtv$`2010` 0.13249
```

```
---
```

```
Signif. codes:
```

```
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 0.08794 on 11 degrees of freedom
```

```
Multiple R-squared: 0.9777, Adjusted R-squared: 0.9736
```

```
F-statistic: 240.8 on 2 and 11 DF, p-value: 8.288e-10
```

El modelo en su conjunto nos señala que las dos variables independientes no tienen una función explicativa o predictiva de la variable dependiente. Por separado, nos señala que la variable del año 2005 tiene más relación con la variable dependiente. Finalmente, el valor de R -al cuadrado (0.9736) señala que 97% del conjunto de los datos predice o explica la modelación.

Como se puede ver, la opción de resumen muestra la apelación a la función, la distribución del residuo del ajuste, los coeficientes y la información sobre el ajuste. Por cierto, es posible simplemente llamar la función *lm* o llamar al resumen (*lm (...)*) y no asignar un nombre al objeto modelo.

En algunos casos, esto puede ser más conveniente. Sin embargo, a menudo se desea realizar análisis validatorios, como trazar residuos, calcular estadísticas adicionales o actualizar un modelo para agregar o quitar variables. Al asignarle un nombre al modelo, puede hacer que su código sea más fácil de comprender y modificar. Además, reinstalar un modelo puede consumir mucho tiempo para modelos complejos y grandes conjuntos de datos. Al asignar el modelo a un nombre de variable, puede evitar estos problemas.

Finalmente, las redes neuronales son un tipo muy popular de modelo estadístico. Las redes neuronales fueron diseñadas originalmente para aproximarse a cómo funcionan las neuronas en el cerebro humano; gran parte de la investigación original sobre redes neuronales provino de investigadores de Inteligencia Artificial. Las redes neuronales son muy flexibles y se pueden usar para modelar una gran cantidad de problemas diferentes. Al cambiar la estructura de las redes neuronales, es posible modelar algunas relaciones no lineales muy complicadas. Las redes neuronales son tan populares que hay revistas académicas enteras dedicadas a ellas.

Los modelos de redes neuronales usan la función *nnet* en el paquete *nnet* (tabla 16). Fórmula:

```
nnet(x,y, weights, size, Wts, mask,
linout=FALSE, entropy=FALSE, softmax=FALSE,
censored=FALSE, skip=FALSE, rang=0.7, decay=0,
maxit=100, Hess=FALSE, trace=TRUE, MaxNWts=1000,
abstol=1.0e-4, reltol=1.0e-8,...)
```

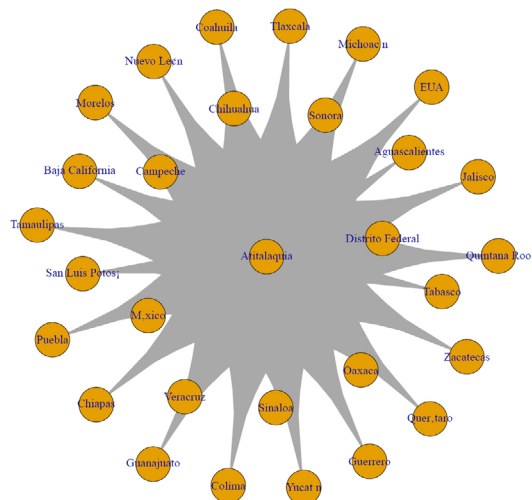
Tabla 16. Argumentos de la función *nnet*

| Argumento | Descripción | Default |
|-----------|---|------------------------------------|
| formula | La fórmula describe la relación entre la respuesta y la predicción de variables. | |
| data | El <i>data.frame</i> que contiene los datos. | |
| weights | Un vector opcional de pesos para los datos de entrenamiento. | |
| ... | Argumentos adicionales pasados a otras funciones. | |
| subset | Un vector opcional que especifica el subconjunto de observaciones para usar en el ajuste del modelo. | |
| size | Número de unidades en la capa oculta. | NULL |
| wts | Parámetros iniciales del vector. | Elige al azar, si no se especifica |
| linot | Use <i>linout</i> = <i>FALSE</i> para unidades de salida logísticas, <i>linout</i> = <i>TRUE</i> para unidades lineales | FALSE |
| decay | Un parámetro numérico para la disminución de peso. | 0 |
| skip | Un valor lógico que especifica si se deben agregar conexiones de capa de salto de entrada a salida. | FALSE |
| softmax | Un valor lógico que especifica si se debe usar un modelo <i>softmax</i> / <i>log-linear</i> y un ajuste de probabilidad condicional máximo. | FALSE |

Fuente: elaboración propia con base en Teetor (2011).

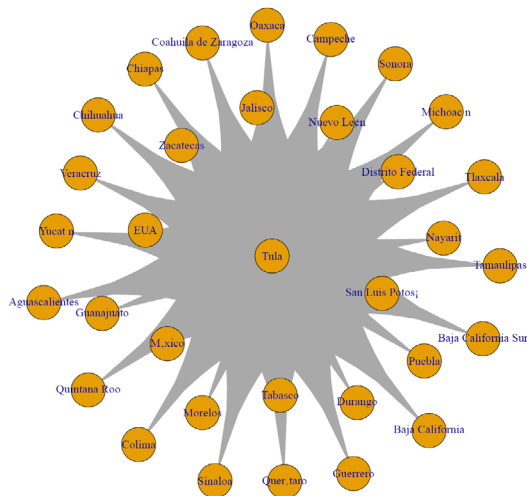
En seguida, se presentará el resultado de un algoritmo aplicado para representar la red de emigración interna concéntrica basada en el origen de dos municipios del estado de Hidalgo, Tula y Atitalaquia, población que se dirige con intensidad diferenciada a los demás estados que comprende la República mexicana. A medida que el nodo se encuentra más próximo al centro, se entiende que hay mayor intensidad migratoria en relación con los nodos que se alejan del centro; ésta es otra forma de abordar el espacio desde las ciencias sociales sin necesidad de recurrir a la cartografía, que en algunas ocasiones se sobresatura en un análisis o no se utiliza de forma adecuada. Ambas redes se realizaron con información del Instituto Nacional de Estadística y Geografía y operaciones con el programa *R*, damos un adelanto de lo que potencialmente se puede realizar y analizaremos en un libro futuro. Cabe señalar que en las etiquetas los acentos no se reconocen, por lo que estos se sustituyen por otro símbolo (figuras 19 y 20; tabla 17).

Figura 19. Atitalaquia. Red de intensidad de población emigrante a los estados y EUA, 2010



Fuente: elaboración propia con base en INEGI (2010a).

Figura 20. Tula. Red de intensidad de población emigrante a los estados y EUA, 2010



Fuente: elaboración propia con base en INEGI (2010a).

**Tabla 17. Atitalaquia-Tula:
población emigrante a los estados y EUA, 2010**

| Origen | Destino | Total | Hombres | Mujeres | Origen | Destino | Total | Hombres | Mujeres |
|-------------|------------------|-------|---------|---------|--------|---------------------|-------|---------|---------|
| Atitalaquia | Aguascalientes | 14 | 4 | 10 | Tula | Aguascalientes | 57 | 31 | 26 |
| Atitalaquia | Baja California | 22 | 13 | 9 | Tula | Baja California | 68 | 41 | 27 |
| | | | | | Tula | Baja California Sur | 8 | 3 | 5 |
| Atitalaquia | Campeche | 4 | 2 | 2 | Tula | Campeche | 18 | 8 | 10 |
| Atitalaquia | Coahuila | 1 | 0 | 1 | Tula | Coahuila | 51 | 20 | 31 |
| Atitalaquia | Colima | 4 | 2 | 2 | Tula | Colima | 20 | 8 | 12 |
| Atitalaquia | Chiapas | 5 | 4 | 1 | Tula | Chiapas | 46 | 16 | 30 |
| Atitalaquia | Chihuahua | 14 | 5 | 9 | Tula | Chihuahua | 82 | 38 | 44 |
| Atitalaquia | Distrito Federal | 391 | 196 | 195 | Tula | Distrito Federal | 1,201 | 635 | 566 |
| | | | | | Tula | Durango | 22 | 10 | 12 |
| Atitalaquia | Guanaajuato | 48 | 21 | 27 | Tula | Guanaajuato | 156 | 72 | 84 |
| Atitalaquia | Guerrero | 5 | 3 | 2 | Tula | Guerrero | 92 | 41 | 51 |
| Atitalaquia | Jalisco | 20 | 8 | 12 | Tula | Jalisco | 67 | 36 | 31 |
| Atitalaquia | México | 399 | 195 | 204 | Tula | México | 996 | 466 | 530 |
| Atitalaquia | Michoacán | 15 | 9 | 6 | Tula | Michoacán | 117 | 54 | 63 |
| Atitalaquia | Morelos | 7 | 6 | 1 | Tula | Morelos | 68 | 30 | 38 |
| | | | | | Tula | Nayarit | 8 | 3 | 5 |
| Atitalaquia | Nuevo León | 35 | 18 | 17 | Tula | Nuevo León | 82 | 43 | 39 |
| Atitalaquia | Oaxaca | 27 | 12 | 15 | Tula | Oaxaca | 114 | 52 | 62 |
| Atitalaquia | Puebla | 58 | 27 | 31 | Tula | Puebla | 183 | 99 | 84 |
| Atitalaquia | Querétaro | 50 | 28 | 22 | Tula | Querétaro | 194 | 101 | 93 |
| Atitalaquia | Quintana Roo | 8 | 7 | 1 | Tula | Quintana Roo | 17 | 11 | 6 |

| Origen | Destino | Total | Hombres | Mujeres | Origen | Destino | Total | Hombres | Mujeres |
|-------------|-----------------|-------|---------|---------|--------|-----------------|-------|---------|---------|
| Atitalaquia | San Luis Potosí | 15 | 5 | 10 | Tula | San Luis Potosí | 58 | 28 | 30 |
| Atitalaquia | Sinaloa | 18 | 8 | 10 | Tula | Sinaloa | 29 | 13 | 16 |
| Atitalaquia | Sonora | 2 | 1 | 1 | Tula | Sonora | 39 | 22 | 17 |
| Atitalaquia | Tabasco | 11 | 5 | 6 | Tula | Tabasco | 26 | 14 | 12 |
| Atitalaquia | Tamaulipas | 71 | 36 | 35 | Tula | Tamaulipas | 187 | 89 | 98 |
| Atitalaquia | Tlaxcala | 4 | 2 | 2 | Tula | Tlaxcala | 42 | 19 | 23 |
| Atitalaquia | Veracruz | 68 | 34 | 34 | Tula | Veracruz | 247 | 124 | 123 |
| Atitalaquia | Yucatán | 1 | 1 | 0 | Tula | Yucatán | 9 | 5 | 4 |
| Atitalaquia | Zacatecas | 2 | 1 | 1 | Tula | Zacatecas | 9 | 5 | 4 |
| Atitalaquia | EUA | 85 | 63 | 22 | Tula | EUA | 621 | 431 | 190 |

Fuente: elaboración propia con base en INEGI (2010b).

Bibliografía recomendada: profundizar con los autores

- Agresti, A. (2015). *Foundations of linear and generalized linear models*. Wiley.
- Aho, K. (2013). *Foundational and applied statistics for biologists using R*. Chapman & Hall/CRC.
- Aho, K. (2014). *Foundational and applied statistics for biologists using R: introduction to R*. Taylor & Francis.
- Ang, C. (2015). *Analyzing financial data and implementing financial models using R*. Springer.
- Aragón, T. (2013). *Applied epidemiology using R*. University of California.
- Armstrong II, D., Bakker, R., Carroll, R., Hare, C., Poole, K. T. y Rosenthal, H. (2014). *Analyzing spatial models of choice and judgment with R*. Chapman & Hall/CRC.
- Arnold, T. y Tilton, L. (2015). *Humanities data in R: exploring networks, geospatial data, images, and text*. Springer.

- Bartolucci, F., Bacci, S. y Gnaldi, M. (2015). *Statistical analysis of questionnaires: a unified approach based on R and Stata*. Chapman & Hall/CRC.
- Beaujean, A. (2014). *Latent variable modeling using R: a step-by-step guide*. Routledge.
- Berridge, D. y Crouchley, R. (2011). *Multivariate generalized linear mixed models using R*. CRC.
- Beyersmann, J., Allignol, A. y Schumacher, M. (2012). *Competing risks and multistate models with R*. Springer.
- Bilder, C. (2013). *Analysis of categorical data with R*. Chapman & Hall/CRC.
- Black, K. (2014). *R object-oriented programming*. Packt.
- Boos, D. y Stefanski, L. (2013). *Essential statistical inference: theory and methods*. Springer.
- Braun, W. y Murdoch, D. (2007). *A first course in statistical programming with R*. Cambridge University Press.
- Bretz, F. y Hothorn, T. (2011). *Westfall P. Multiple comparisons using R*. Chapman & Hall/CRC.
- Chatterjee, S. y Simonoff, J. S. (2013). *Handbook of Regression Analysis*. Wiley.
- Cichosz, P. (2015). *Data mining algorithms: explained using R*. Wiley.
- Crawley, M. (2014). *Statistics: an introduction using R* (2a Ed.). Wiley.
- Crowder, M. (2016). *Multivariate survival analysis and competing risks*. Chapman & Hall/CRC.
- Danneman, N. y Heimann, R. (2014). *Social media mining with R: deploy cutting-edge sentiment analysis techniques to real-world social media data using R*. Packt.
- Daróczi, G. (2015). *Mastering data analysis with R*. Packt.
- Demidenko, E. (2013). *Mixed models: theory and applications with R* (2a Ed.). Wiley.
- Georgakopoulos, H. (2015). *Quantitative trading with R: understanding mathematical and computational tools from a quant's perspective*. Palgrave Macmillan.
- James, G., Witten, D., Hastie, T. y Tibshirani, R. (2013). *An introduction to statistical learning: with applications in R*. Springer.
- Santos-Fernández, E. (2012). *Multivariate statistical quality control using R*. Springer.
- Takezawa, K. (2014). *Learning regression analysis by simulation*. Springer.

- Thiem, A. y Duşa, A. (2013). *Qualitative comparative analysis with R: a user S guide*. Springer.
- Verzani, J. (2014). *Using R for introductory statistics* (2a Ed.). Chapman & Hall/CRC.
- Voss, J. (2013). *An introduction to statistical computing: a simulation-based approach*. Wiley.
- Wickham, H. (2014). *Advanced R*. Chapman & Hall/CRC.
- Xie, Y. (2015). *Dynamic documents with R and knitr* (2a Ed.). Chapman & Hall/CRC.
- Zieffer, A., Harring, J. y Long, J. (2011). *Comparing groups: randomization and bootstrap methods using R*. Wiley.

CAPÍTULO 5

REPASO Y REFUERZO PARA ORDENAR LOS DATOS

#Sumar un conjunto de números:

1+1

[1] 2

#Imprimir un enunciado con signos de admiración, formato carácter, identificado por las comillas que lo envuelven (figura 21).

“¡Arre, Mateo!”

[1] “¡Arre, Mateo!”

#Una multiplicación con dos números (figura 21).

7*8

[1] 56

#Realizar una serie de operaciones lógicas en R (figura 21).

10<8

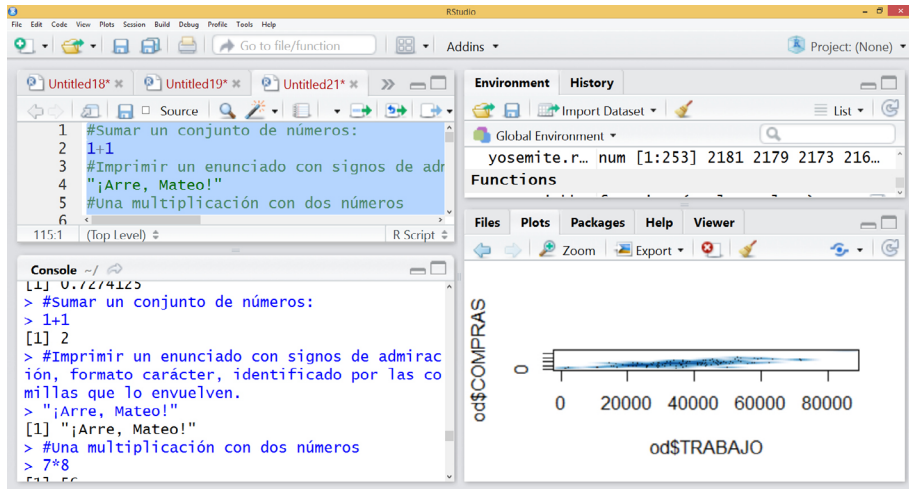
[1] FALSE

4+10==8

[1] FALSE

T==TRUE

[1] TRUE

Figura 21. Flujos computacionales

#Generar un vector y posteriormente realizar operaciones con él (figura 22).

```

X<-42
X
[1] 42

```

```

X/2
[1] 21

```

```

X<- "¡Arre, Mateo!"
X
[1] "¡Arre, Mateo!"

```

```

X<-TRUE
X
[1] TRUE

```

```

#Imprimir algo (figura 22).
print(X)
[1] TRUE

```

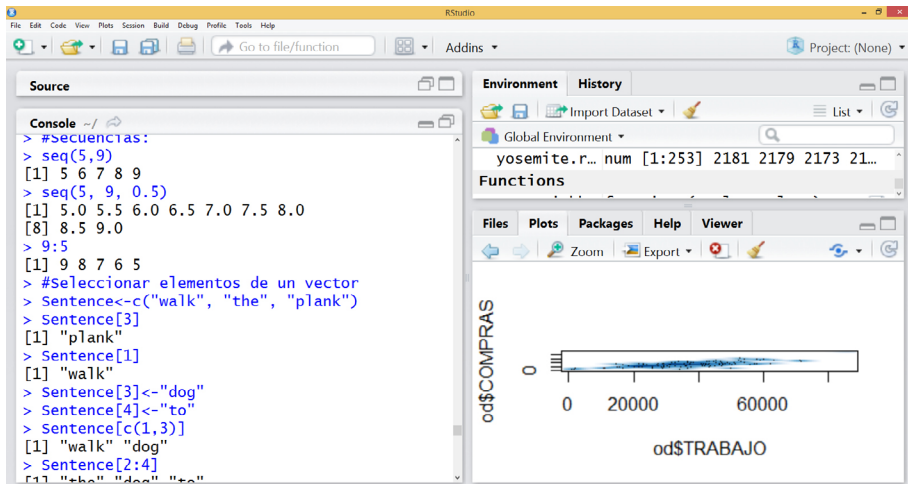

#Generar un vector con el conjunto de los primeros 10 números de Fibonacci (figura 22).

```
Fibonacci<-c(1,1,2,3,5,8,13,21,33,54)
sum(Fibonacci)
[1] 141
```

#Realizar repeticiones con la función indicada (figura 22).

```
rep("¡Arre, Mateo!", times=10)
[1] "¡Arre, Mateo!"
[2] "¡Arre, Mateo!"
[3] "¡Arre, Mateo!"
[4] "¡Arre, Mateo!"
[5] "¡Arre, Mateo!"
[6] "¡Arre, Mateo!"
[7] "¡Arre, Mateo!"
[8] "¡Arre, Mateo!"
[9] "¡Arre, Mateo!"
[10] "¡Arre, Mateo!"
```

Figura 22. Ejecución de funciones



```
#Una raíz cuadrada
sqrt(25)
[1] 5
```

```
#Pedir auxilio con respecto a las características de una función.
help(sum)
help(mean)
help(sqrt)
?summary
#Función de conjunto:
c(4, 7, 9)
[1] 4 7 9
```

```
c("a", "b", "c")
[1] "a" "b" "c"
```

```
c(1, "TRUE", "three")
[1] "1" "TRUE" "three"
```

```
#Secuencias (figura 23).
seq(5,9)
[1] 5 6 7 8 9
```

```
seq(5, 9, 0.5)
[1] 5.0 5.5 6.0 6.5 7.0 7.5
[7] 8.0 8.5 9.0
```

```
9:5
[1] 9 8 7 6 5
```

```
#Seleccionar elementos de un vector (figura 23).
Sentence<-c("walk", "the", "plank")
Sentence[3]
[1] "plank"
```

```

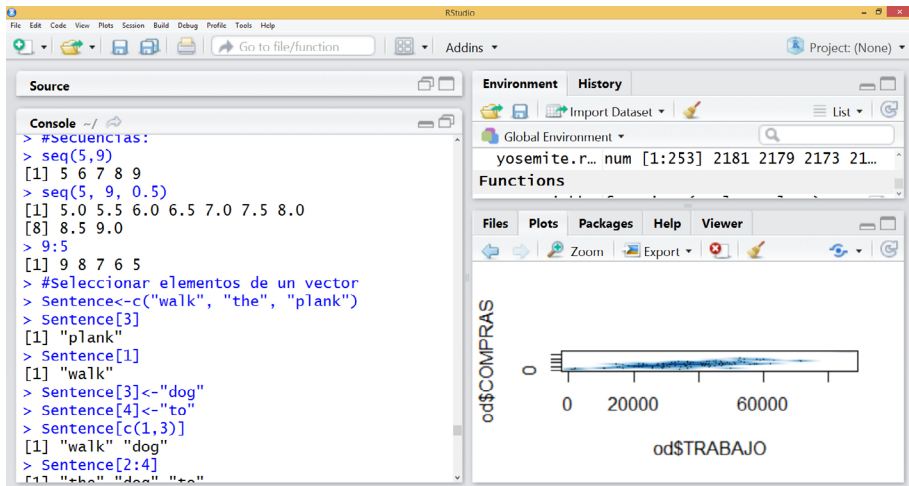
Sentence[1]
[1] "walk"

Sentence[3]<-"dog"
Sentence[4]<-"to"
Sentence[c(1,3)]
[[1] "walk" "dog"

Sentence[2:4]
[1] "the" "dog" "to"

```

Figura 23. Esquema de trabajo



```

#Incrementar objetos del vector
Sentence[5:7]<-c("the", "poop", "deck")
Sentence
[1] "walk" "the" "dog"
[4] "to" "the" "poop"
[7] "deck"

Sentence[c(6)]
[1] "poop"

```

```
#Otro vector
Ranks<-1:3
names(Ranks)<-c("first", "second", "third")
Ranks["third"]<-4
Ranks
first second third
 1  2  4

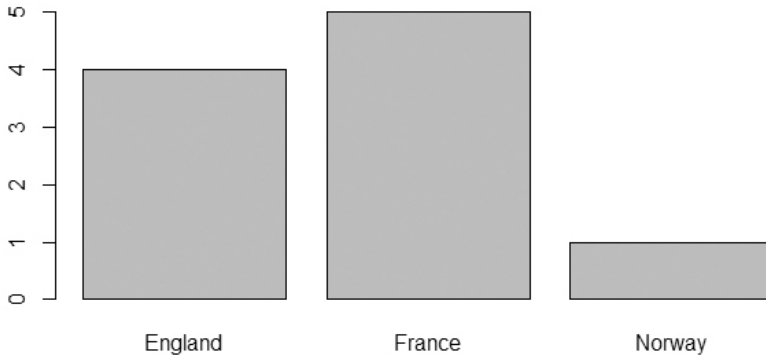
vesselsSunk<-c(4,5,1)
vesselsSunk
[1] 4 5 1

#Generar gráficos de barras (figura 24).
barplot(vesselsSunk)
```

Figura 24. Gráfica de barras sin etiquetas en el eje x

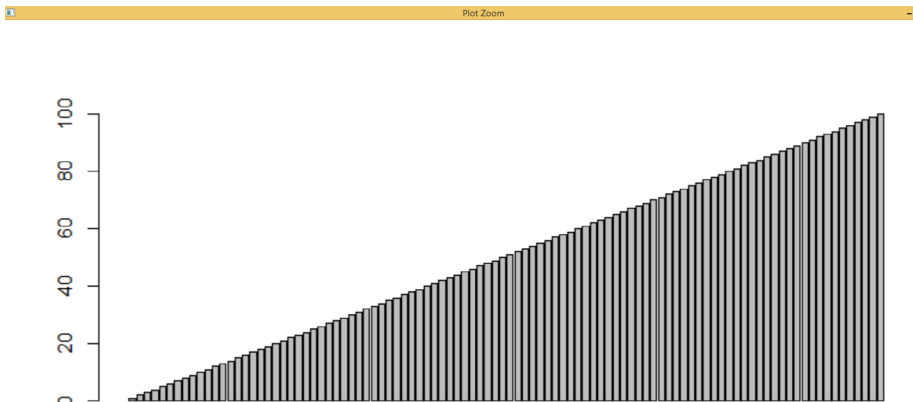
```
names(vesselsSunk)<-c("England", "France", "Norway")
#De nueva cuenta, graficamos el vector vesselsSunk, pero ahora con su eti-
queta (figura 25).
```

Figura 25. Gráfica de barras con sus etiquetas en el eje x



#Graficar de nuevo un vector (figura 26).
`barplot(1:100)`

Figura 26. Modelo secuencial de gráfica de barras



#Otra vez, operaciones con un vector (figura 27) `a<-c(1,2,3)`

```
a+1
```

```
[1] 2 3 4
```

```
a/2
```

```
[1] 0.5 1.0 1.5
```

```
a*2
```

```
[1] 2 4 6
```

```
b<-c(4,5,6)
```

```
a+b
```

```
[1] 5 7 9
```

```
a-b
```

```
[1] -3 -3 -3
```

```
a==c(1,99,3)
```

```
[1] TRUE FALSE TRUE
```

```
a<b
```

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

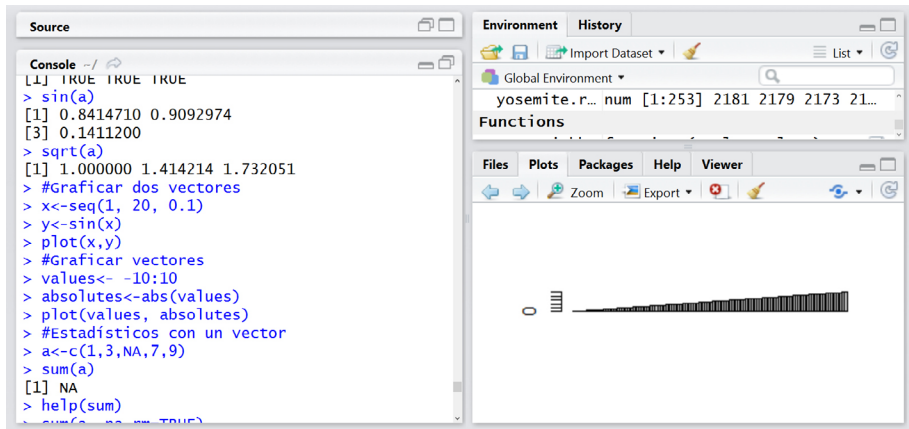
```
sin(a)
```

```
[3] 0.1411200
```

```
sqrt(a)
```

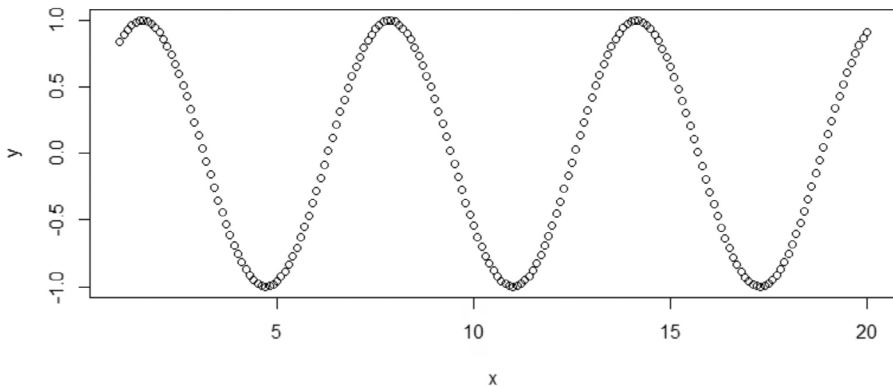
```
[1] 1.000000 1.414214 1.732051
```

Figura 27. Visualización del vector

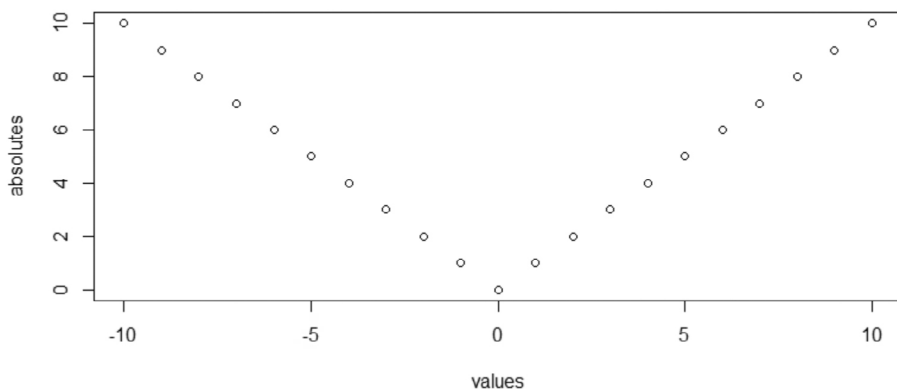


#Graficar dos vectores (figura 28)x<-seq(1,20,0.1)
 y<-sin(x)
 plot(x,y)
 #Ver (figura 28).

Figura 28. Plot xy



```
values<- -10:10
absolutes<-abs(values)
plot(values, absolutes)
#Ver figura 29.
```

Figura 29. Distribución de vector

```
#Estadísticos con un vector
a<-c(1,3,NA,7,9)
sum(a)
[1] NA
```

```
help(sum)
sum(a, na.rm=TRUE)
[1] 20
```

```
matrix(0, 3, 4)
  [,1] [,2] [,3] [,4]
[1,] 0 0 0 0
[2,] 0 0 0 0
[3,] 0 0 0 0
```



```
###Segunda parte
```

```
a<-1:12
```

```
print(a)
```

```
[1] 1 2 3 4 5 6 7 8
```

```
[9] 9 10 11 12
```

```
#Elaborar una matriz con el vector a
```

```
matrix(a, 3, 4)
```

```
[,1] [,2] [,3] [,4]
```

```
[1,] 1 4 7 10
```

```
[2,] 2 5 8 11
```

```
[3,] 3 6 9 12
```

```
plank<-1:8
```

```
dim(plank)<-c(2,4)
```

```
print(plank)
```

```
[,1] [,2] [,3] [,4]
```

```
[1,] 1 3 5 7
```

```
[2,] 2 4 6 8
```

```
matrix(1, 5, 5)
```

```
[,1] [,2] [,3] [,4] [,5]
```

```
[1,] 1 1 1 1 1
```

```
[2,] 1 1 1 1 1
```

```
[3,] 1 1 1 1 1
```

```
[4,] 1 1 1 1 1
```

```
[5,] 1 1 1 1 1
```

```
plank[2,3]
```

```
[1] 6
```

```
plank[1,4]
```

```
[1] 7
```

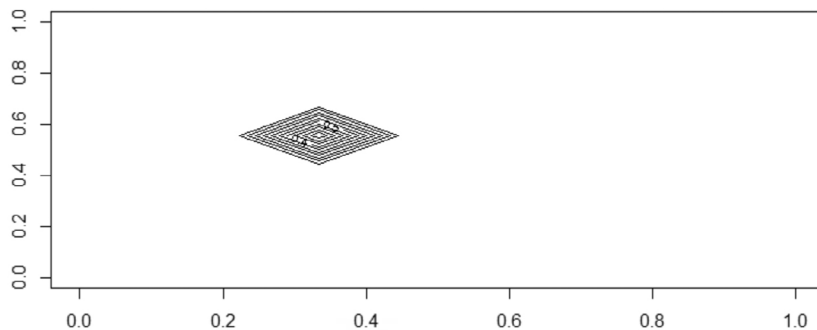
```
plank[1,4]<-0
plank[2,]
[1] 2 4 6 8
```

```
plank[,4]
[1] 0 8
```

```
plank[, 2:4]
  [,1] [,2] [,3]
[1,] 3 5 0
[2,] 4 6 8
```

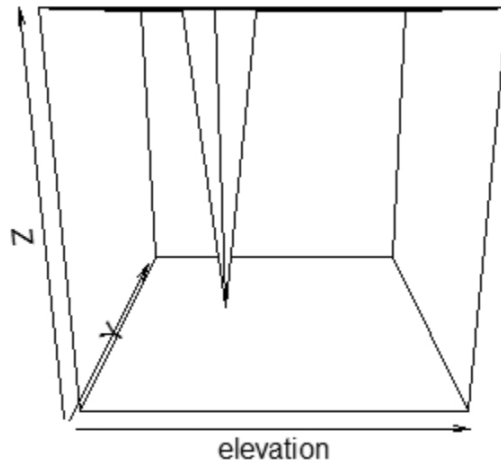
```
#Elaborar un vector y una imagen de éste (figuras 30, 31 y 32).
elevation<-matrix(1,10,10)
elevation[4,6]<-0
contour(elevation)
persp(elevation)
```

Figura 30. Perspectiva de elevación



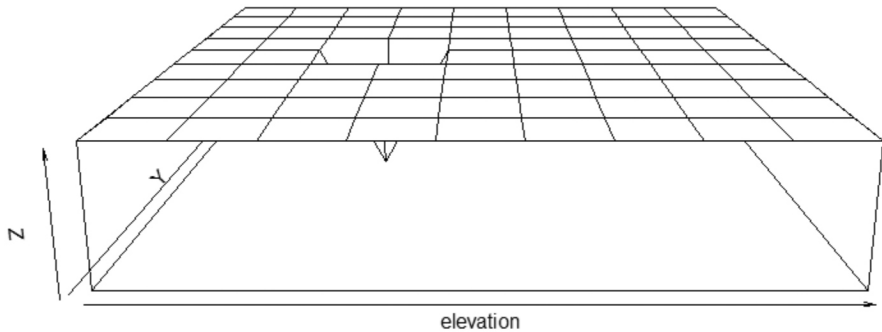
```
persp(elevation, expand=0.2)
contour(elevation)
```

Figura 31. Contorno de elevación



`image(elevation)` → ctrl+enter (al mismo tiempo)

Figura 32. Imagen en tres dimensiones de elevación

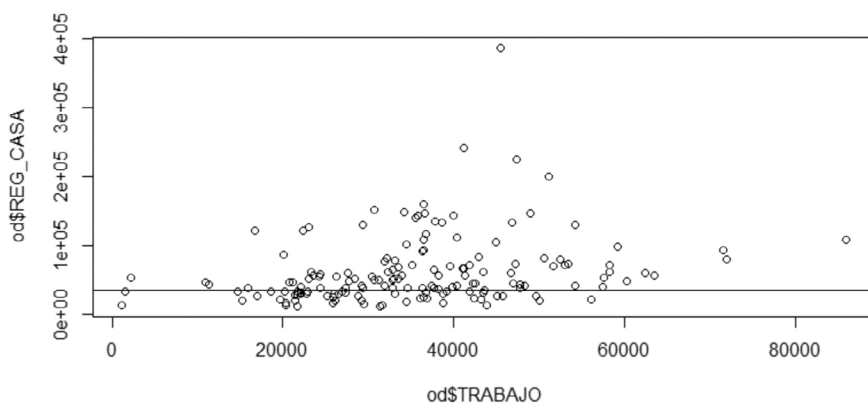


#Montar una base de datos en la plataforma y realizar unos estadísticos y gráficos con las variables de ésta, datos que se descargan de la EOD 2017 (figura 33).

```
od<-read.csv("C:/R/Bases_alumnos/cuadro_31.csv", header=TRUE,
sep=";", dec=".", fill=TRUE)
plot(od$TRABAJO, od$REG_CASA)
```

#Ver figura 33.

Figura 33. Correlación de viajes por motivo de trabajo y regreso a casa en la EOD, 2007

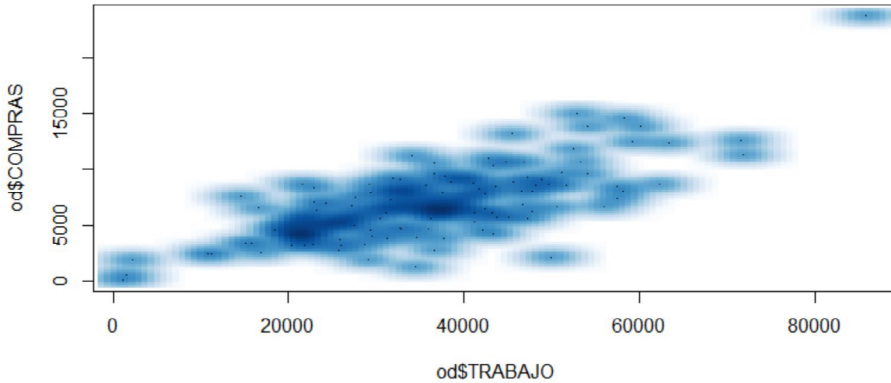


```
abline(h=median(od$TRABAJO))
cor(od$TRABAJO, od$REG_CASA)
smoothScatter(od$TRABAJO, od$COMPRAS)
```

#Ver figura 34.

```
cor(od$TRABAJO, od$COMPRAS)
```

Figura 34. Correlación entre viajes por motivo de trabajo y viajes por motivo de compras, EOD-2007



R incluye varios paquetes para visualizar datos. Usualmente, encontrará que las funciones dentro de los gráficos y los paquetes de *R* son los más útiles. Si el usuario está familiarizado con Microsoft Excel, encontrará que *R* puede generar todos los gráficos conocidos: de columnas, de barras, de líneas, circulares y diagramas de dispersión. Esto hace que sea mucho más fácil automatizar la creación de gráficos y personalizarlos que en Excel. Sin embargo, hay muchos más tipos de gráficos disponibles en *R*, muchos de ellos bastante intuitivos y elegantes (tabla 10).

Instalar, desinstalar y crear complementos

En el programa *R* es posible instalar complementos con la función *install.packages*, por ejemplo, vamos a instalar el programa para análisis de mapas, *maptree*, acción que le solicitamos al lector replicar desde su ordenador.

```
install.packages("maptree")
```

Para desinstalar un programa debemos utilizar la función *remove.package*, además, se necesita saber el nombre de la librería que es preciso remover, por ejemplo, *maptree*. En la tabla 18 se pueden observar algunas funciones para instalar, desinstalar, actualizar, conocer los paquetes ya instalados, los incorporados recientemente y los más antiguos.

```
remove.package("maptree")
```

Tabla 18. Comandos más comunes para instalación de complementos

| | |
|-----------------------------------|--|
| <code>installed.packages()</code> | Devuelve una matriz con información sobre todos los paquetes actualmente instalados. |
| <code>available.packages()</code> | Devuelve una matriz de todos los paquetes disponibles en el repositorio. |
| <code>old.packages()</code> | Devuelve una matriz de todos los paquetes actualmente instalados para los cuales hay versiones más nuevas disponibles. |
| <code>new.packages()</code> | Devuelve una matriz que muestra todos los paquetes actualmente no disponibles en los repositorios de paquetes. |
| <code>download.packages()</code> | Descarga un conjunto de paquetes en un directorio local. |
| <code>install.packages()</code> | Instala un conjunto de paquetes desde el repositorio. |
| <code>remove.package()</code> | Elimina un conjunto de paquetes instalados. |
| <code>update.packages()</code> | Actualiza los paquetes instalados a las últimas versiones. |
| <code>setRepositories()</code> | Establece la lista actual de repositorios de paquetes. |

Fuente: elaboración propia con base en Teetor (2011).

Crear paquetes propios es una buena idea si se desea compartir el código o los datos con otras personas, o si sólo se requiere empaquetarlos en una forma que se pueda reutilizar; es necesario considerar la posibilidad de crear un paquete propio. Esta sección explica la manera fácil de hacerlo (tabla 19).

Para compilar un paquete, se deben colocar todos los archivos de éste (código, datos, documentación, entre otros) dentro de un único directorio. Es posible crear una estructura de directorio apropiada usando la función *R.package.skeleton*. Los argumentos se describen en la tabla 19.

Tabla 19. Elementos para compilar un paquete

| Argumento | Descripción | Default |
|-------------|---|--|
| name | Un valor tipo carácter especifica el nombre del nuevo paquete. | <i>"anRpackage"</i> . Como nota al margen, éste puede ser el valor predeterminado menos útil para cualquier función <i>R</i> . |
| list | Un vector carácter que contiene nombres de los objetos <i>R</i> para adjuntar al paquete. | |
| environment | El entorno en el que evaluar la lista. | .GlobalEnv |
| path | Un vector carácter especificando la ruta de archivo del sistema. | FALSE |
| namespace | Un valor booleano que especifica si se quiere sobrescribir archivos, en caso de que ya exista un nombre de directorio en la ruta. | FALSE |
| code_files | Un vector de caracteres que especifica las rutas de los archivos que contienen el código <i>R</i> . | character() |

Fuente: elaboración propia con base en Teetor (2011).

Administrar datos

A menudo, es conveniente poner estos vectores en un *data.frame*. Para crear un *data.frame*, use la función de *data.frame* para compilar el vector.

```

Nombres<-c("Ronaldo", "Messi", "Chicharito", "Ronaldo", "Maradona")
Equipos<-c("Santos", "Real Madrid", "Guadalajara", "Portugal",
"Argentina")
Posicion<-c("Medio", "Delantero", "Delantero", "Delantero", "Medio")
Salario<-c(1300000, 1800000, 1400000, 1800000, 2000000)
Teams<-data.frame(Nombres,Equipos,Posicion,Salario)
Teams

```

```

Nombres Equipos Posicion Salario
1 Ronaldinho Santos Medio 1300000
2 Messi Real Madrid Delantero 1800000
3 Chicharito Guadalajara Delantero 1400000
4 Ronaldo Portugal Delantero 1800000
5 Maradona Argentina Medio 2000000

```

Ingresa datos usando declaraciones individuales puede ser incómodo para más de un puñado de observaciones. Por eso, el ejemplo anterior sólo incluye cinco observaciones.

Para editar un objeto con el editor de datos, úsese la función de *edit*. La función de edición abrirá el editor de datos y devolverá el objeto editado. Por ejemplo, para editar el *data.frame* “*Teams*”, se debe utilizar el siguiente comando.

```
edit(Teams)
```

Figura 35. Variables contenidas en un *data.frame*

| Nombre | Equipos | Posicion | Salario | var5 | var6 | var7 | var8 | var9 | var10 | var11 | var12 | var13 | var14 | var15 | var16 | var17 |
|--------|------------|-------------|-----------|---------|------|------|------|------|-------|-------|-------|-------|-------|-------|-------|-------|
| 1 | Ronaldinho | Santos | Medio | 1300000 | | | | | | | | | | | | |
| 2 | Messi | Real Madrid | Delantero | 1800000 | | | | | | | | | | | | |
| 3 | Chicharito | Guadalajara | Delantero | 1400000 | | | | | | | | | | | | |
| 4 | Ronaldo | Portugal | Delantero | 1800000 | | | | | | | | | | | | |
| 5 | Maradona | Argentina | Medio | 2000000 | | | | | | | | | | | | |
| 6 | | | | | | | | | | | | | | | | |
| 7 | | | | | | | | | | | | | | | | |
| 8 | | | | | | | | | | | | | | | | |
| 9 | | | | | | | | | | | | | | | | |
| 10 | | | | | | | | | | | | | | | | |
| 11 | | | | | | | | | | | | | | | | |
| 12 | | | | | | | | | | | | | | | | |
| 13 | | | | | | | | | | | | | | | | |
| 14 | | | | | | | | | | | | | | | | |
| 15 | | | | | | | | | | | | | | | | |
| 16 | | | | | | | | | | | | | | | | |
| 17 | | | | | | | | | | | | | | | | |
| 18 | | | | | | | | | | | | | | | | |
| 19 | | | | | | | | | | | | | | | | |
| 20 | | | | | | | | | | | | | | | | |
| 21 | | | | | | | | | | | | | | | | |
| 22 | | | | | | | | | | | | | | | | |
| 23 | | | | | | | | | | | | | | | | |
| 24 | | | | | | | | | | | | | | | | |
| 25 | | | | | | | | | | | | | | | | |
| 26 | | | | | | | | | | | | | | | | |
| 27 | | | | | | | | | | | | | | | | |
| 28 | | | | | | | | | | | | | | | | |
| 29 | | | | | | | | | | | | | | | | |
| 30 | | | | | | | | | | | | | | | | |
| 31 | | | | | | | | | | | | | | | | |
| 32 | | | | | | | | | | | | | | | | |
| 33 | | | | | | | | | | | | | | | | |
| 34 | | | | | | | | | | | | | | | | |
| 35 | | | | | | | | | | | | | | | | |
| 36 | | | | | | | | | | | | | | | | |
| 37 | | | | | | | | | | | | | | | | |
| 38 | | | | | | | | | | | | | | | | |
| 39 | | | | | | | | | | | | | | | | |

El editor de datos está diseñado para editar objetos de datos tabulares, específicamente, marcos de datos y matrices. La función de edición se puede

usar con otros tipos de objetos, como vectores, funciones y listas, pero abrirá un editor de texto.

Alternativamente, puede usar la función de reparación *fix*. La función de corrección, *fix*, llama a *edit* su argumento y luego asigna el resultado al mismo símbolo en el entorno de llamada. Para el ejemplo anterior, aquí mostramos cómo usar el arreglo.

```
fix(Teams)
```

La forma más sencilla de guardar un objeto es la función guardar, *save*. Por ejemplo, podemos usar el siguiente comando para guardar el objeto anteriormente editado.

```
save(Teams, file = "~/Dolares.RData")
```

En *R*, las rutas de archivos siempre se especifican con barras diagonales (“/”), incluso en Microsoft Windows. Entonces, para guardar este archivo en el directorio, se usaría el comando anterior.

Se debe tener en cuenta que el argumento del archivo llevará un nombre explícito. Nueve de cada diez veces, alguien se olvida de hacerlo (Teetor, 2011). Ahora, es posible cargar fácilmente un objeto en *R* con la función de cargar, *load*.

Si el usuario es novato en el análisis de datos, probablemente se pregunte qué importancia tiene la preparación de datos. Supongamos que se están obteniendo algunos datos del servidor web, de alguna empresa o de una base de datos financiera, o de registros electrónicos de pacientes, todo procedente de las computadoras, entonces, ¿es perfecto, correcto?

En la práctica, los datos casi nunca se almacenan directamente para el análisis. Incluso, cuando los datos están albergados en su totalidad, a menudo hay sorpresas en ellos. Reunir un conjunto de datos utilizables conlleva mucho trabajo.

En la combinación de conjuntos de datos existe un obstáculo más común para el análisis: trabajar con información almacenada en dos lugares diferentes. Por ejemplo, suponga que desea ver las estadísticas de tiros a gol de

los jugadores de la liga mexicana por edad. En la mayoría de las fuentes de datos de fútbol, la información del jugador se guarda en diferentes archivos de datos de rendimiento. Entonces, se necesitaría combinar dos archivos para hacer dicho análisis. Esta sección aborda varias herramientas en *R* utilizadas para combinar el conjunto de datos.

Paste. La más simple de estas funciones es *paste*. La función de pegado permite concatenar vectores de caracteres múltiples en un solo vector. Si concatena un vector de otro tipo, primero se asociará con un vector de caracteres.

```
VAR1<-c("a", "b", "c", "d", "e")
VAR2<-c("A", "B", "C", "D", "E")
paste(VAR1,VAR2)
[1] "a A" "b B" "c C" "d D" "e E"
```

Por defecto, los valores están separados por un espacio; es posible especificar otro separador o ninguno con el argumento *sep*.

```
Horas<-c(09, 10, 11, 12, 13)
Minutos<-c(53, 10, 43, 01, 09)
paste(Horas, Minutos, sep=":")
[1] "9:53" "10:10" "11:43" "12:1" "13:9"
```

Si desea que todos los valores del vector devuelto se concatenen entre sí, para devolver un solo valor, especifique un valor para el argumento de contracción. El valor de colapso se usará como el separador en este valor.

```
paste(Horas, Minutos, sep = ":", collapse = "@")
[1] "9:53@10:10@11:43@12:1@13:9"
```

A veces, se desea vincular varios marcos de datos o matrices. Esto se puede hacer con las funciones *rbind* y *cbind*. La función *cbind* combinará objetos mediante la adición de columnas, esto es como combinar dos tablas horizontalmente. Como ejemplo, comencemos con el marco de datos para los cinco salarios principales en el fútbol.

Teams

```
Nombres Equipos Posicion Salario
1 Ronaldinho Santos Medio 1300000
2 Messi Real Madrid Delantero 1800000
3 Chicharito Guadalajara Delantero 1400000
4 Ronaldo Portugal Delantero 1800000
5 Maradona Argentina Medio 2000000
```

```
año<-c(2015, 2015, 2015, 2015, 2015)
posición<-c(5,3,4,2,1)
parche<-data.frame(año,posición)
cbind(Teams,parche)
```

```
Nombres Equipos Posicion Salario año posicion
1 Ronaldinho Santos Medio 1300000 2015 5
2 Messi Real Madrid Delantero 1800000 2015 3
3 Chicharito Guadalajara Delantero 1400000 2015 4
4 Ronaldo Portugal Delantero 1800000 2015 2
5 Maradona Argentina Medio 2000000 2015 1
```

La función *rbind* combinará objetos mediante la adición de filas, esto es como combinar dos tablas verticalmente. Como ejemplo, supongamos que se tiene un marco de datos con los cinco salarios de los principales futbolistas (como se muestra arriba) y un segundo marco de datos con los siguientes tres salarios.

```
mas.tres
Nombres Equipos Posicion Salario
1 Chelito Azul Delantero 100000
2 Conejo Azul Portero 1100000
3 Temo América Delantero 1200000
```

#Formula agregando a Teams, *mas.tres*

```

rbind(Teams,más.tres)
  Nombres Equipos Posicion Salario
1 Ronaldinho Santos Medio 1300000
2 Messi Real Madrid Delantero 1800000
3 Chicharito Guadalajara Delantero 1400000
4 Ronaldo Portugal Delantero 1800000
5 Maradona Argentina Medio 2000000
6 Chelito Azul Delantero 100000
7 Conejo Azul Portero 1100000
8 Temo America Delantero 1200000

```

Cuando se desea transformar datos, un operador utilizado comúnmente es *apply*, función que se utiliza para un conjunto de objetos; éste acepta tres argumentos: *X* es la matriz a la que se aplica una función, *FUN* es la función y *MARGIN* especifica las dimensiones para aplicar la función. Opcionalmente, se pueden especificar argumentos para *FUN* como suma para aplicar argumentos. Para mostrar cómo funciona esto, aquí hay un ejemplo simple. Vamos a crear una matriz con cuatro filas de cinco elementos, correspondientes a los números entre 1 y 20.

```

x<-1:20
dim(x)<-c(5,4)
x
  [,1] [,2] [,3] [,4]
[1,]  1  6 11 16
[2,]  2  7 12 17
[3,]  3  8 13 18
[4,]  4  9 14 19
[5,]  5 10 15 20

```

Ahora, vamos a mostrar cómo funciona *apply*. Usaremos la función mínimo, porque es fácil ver la matriz de arriba, de donde provienen los resultados. El vector *x* se desglosa en renglones y columnas, estos son los márgenes. Si le solicito los valores mínimos del margen uno, arrojará las observaciones con los valores mínimos que se encuentran en los renglones (1, 2, 3, 4, 5), porque ése es el primer marco que se especifica en las dimensiones, por lo contrario, si

le indico los valores mínimos del margen número dos, expondrá los datos de las columnas (1, 6, 11, 16). Recuerdese que pude haber indicado tres dimensiones, la tercera señala la cantidad de tablas en que quiero expresar los datos. Para hacer esto, especificaremos $X = x$, $MARGEN = 1$, las filas son la primera dimensión, y $FUN = min$.

```
apply(X=x, MARGIN = 1, FUN = min)
[1] 1 2 3 4 5
```

Como ya se señaló, para hacer lo mismo con las columnas, simplemente tenemos que cambiar el valor de *MARGIN*.

```
apply(X=x, MARGIN = 2, FUN = min)
[1] 1 6 11 16
```

Ahora presentaremos un ejemplo un poco más complejo, también podemos usar *MARGIN* para aplicar una función de dimensiones múltiples. Pasaremos a la función *pegar*, *paste*, para mostrar qué elementos se incluyeron. Considere la siguiente matriz tridimensional.

```
x<-37:63
dim(x)<-c(3,3,3)
x
, , 1

[,1] [,2] [,3]
[1,] 37 40 43
[2,] 38 41 44
[3,] 39 42 45

, , 2

[,1] [,2] [,3]
[1,] 46 49 52
[2,] 47 50 53
[3,] 48 51 54
```

```
, , 3
```

```
  [,1] [,2] [,3]
[1,] 55 58 61
[2,] 56 59 62
[3,] 57 60 63
```

Comencemos por ver qué valores se acumulan para cada valor de *MARGIN*.

```
apply(X=x, MARGIN = 1, FUN = paste, collapse="")
[1] "37,40,43,46,49,52,55,58,61"
[2] "38,41,44,47,50,53,56,59,62"
[3] "39,42,45,48,51,54,57,60,63"
> apply(X=x, MARGIN = 2, FUN = paste, collapse="")
[1] "37,38,39,46,47,48,55,56,57"
[2] "40,41,42,49,50,51,58,59,60"
[3] "43,44,45,52,53,54,61,62,63"
> apply(X=x, MARGIN = 3, FUN = paste, collapse="")
[1] "37,38,39,40,41,42,43,44,45"
[2] "46,47,48,49,50,51,52,53,54"
[3] "55,56,57,58,59,60,61,62,63"
```

Haremos algo más complicado. Seleccionaremos $MARGIN=c(1,2)$. Esto es el equivalente de hacer lo siguiente: para cada valor de i entre 1 y 3 y cada valor de j entre 1 y 3, calcula FUN de $x[i][j][1]$, $x[i][j][2]$, $x[i][j][3]$.

```
apply(X=x, MARGIN = c(1,2), FUN = paste, collapse="")
  [,1] [,2] [,3]
[1,] "37,46,55" "40,49,58" "43,52,61"
[2,] "38,47,56" "41,50,59" "44,53,62"
[3,] "39,48,57" "42,51,60" "45,54,63"
```

Aplicar una función a una lista o vector. Usar la función *lapply* requiere dos argumentos: un objeto x y ejecutar el comando. Asimismo, se pueden especificar argumentos adicionales. Veamos un ejemplo simple de cómo usar *lapply*.

```

x<-as.list(1:5)
> lapply(x,function(x) 2^x)
[[1]]
[1] 2

[[2]]
[1] 4

[[3]]
[1] 8

[[4]]
[1] 16

[[5]]
[1] 32

```

Al aplicar una función a un *data frame*, se aplicará a cada vector. Por ejemplo:

```

Deuda.fantasma<-data.frame(x=1:5, y=6:10)
Deuda.fantasma
  x y
1 1 6
2 2 7
3 3 8
4 4 9
5 5 10
lapply(Deuda.fantasma, function(x) 2^x)
$x
[1] 2 4 8 16 32

$y
[1] 64 128 256 512 1024

```

```
lapply(Deuda.fantasma, FUN = max)
$х
[1] 5

$у
[1] 10
```

En ocasiones, es posible que prefiera obtener un vector, matriz o *array* en lugar de una lista. Para hacer esto, se recurre a la función *sapply*. Esta función activa exactamente de la misma manera que *apply*, excepto que devuelve un vector o matriz cuando corresponda.

```
sapply(Deuda.fantasma, FUN = function(x) 2^x)
  x y
[1,] 2 64
[2,] 4 128
[3,] 8 256
[4,] 16 512
[5,] 32 1024
```

Otra función relacionada es *mapply*, la versión multivariante de *sapply*.

```
mapply(paste,
+ c(1:5),
+ c("a", "b", "c", "d", "e"),
+ c("A", "B", "C", "D", "E"),
+ MoreArgs = list(sep="-"))
[1] "1-a-A" "2-b-B" "3-c-C" "4-d-D" "5-e-E"
```

Vamos a presentar un ejemplo de Teetor (2011) con múltiples funciones, un ejercicio complejo. El corte de función es útil para tomar una variable continua y dividirla en partes discretas. Aquí está la forma predeterminada de corte para usar con el vector numérico. La función de corte, *cut*, toma un vector numérico como entrada y devuelve un factor. Cada nivel en el factor corresponde a una inversión de valores en el vector de entrada. Aquí hay una descripción del corte.

Por ejemplo, supongamos que quiere contar el número de jugadores con promedios de bateo en ciertos rangos. Para hacer esto, podría usar la función *cut* y la función *table*. Abrimos la librería *nutshell*, después de haber instalado el complemento.

```
library(nutshell)
> data("batting.2008")
> batting.2008.AB<-transform(batting.2008, AVG=H/AB)
> batting.2008.over100AB<-subset(batting.2008.AB, subset
= (AB>100))
> batting.2008.bins<-cut(batting.2008.over100AB$AVG,
breaks = 10)
> table(batting.2008.bins)
batting.2008.bins
(0.137,0.163] (0.163,0.189] (0.189,0.215] (0.215,0.24]
 4  6 24 67
(0.24,0.266] (0.266,0.292] (0.292,0.318] (0.318,0.344]
121 132 70 11
(0.344,0.37] (0.37,0.396]
 5  2
```

Como se puede observar en el primer vector, *batting.2008.AB*, se generó una nueva variable con la operación división de dos ya existentes, *H* y *AB*, en la base de datos, *batting.2008*; subsecuentemente, se seleccionaron del nuevo vector todos los registros *AB* cuyo valor superaba los 100, posteriormente, se eligió de este último vector la variable *AVG* y se generaron 10 rangos para conocer la frecuencia, tal como se muestra al final.

Bibliografía recomendada: profundizar con los autores

- Abedin, J. y Kumar, D. K. (2015). *Data manipulation with R* (2a Ed.). Packt.
- Horton, N. y Kleinman, K. (2015). *Using R and RStudio for data management, statistical analysis and graphics* (2a Ed.). Chapman & Hall/CRC.
- Loo M. van der y Jonge, E. (2012). *Learning RStudio for R Statistical Computing*. Packt.

- Micheaux, P., Drouilhet, R. y Liquet, B. (2013). *The R software: fundamentals of programming and statistical analysis*. Springer.
- Nolan, D. y Lang, D. (2015). *Data science in R: a case studies approach to computational reasoning and problem solving*. Chapman & Hall/CRC.
- Sánchez, G. (2013). *Handling and processing strings in R*. Trowchez.
- Verzani, J. (2014). *Using R for introductory statistics* (2a Ed.). Chapman & Hall/CRC.

CAPÍTULO 6

EJERCICIOS DE AUTOEVALUACIÓN Y CONSOLIDACIÓN

Este capítulo es un reto a la creatividad, pone a prueba la capacidad de integrar los conocimientos del libro y, con base en problemas, aplicar las funciones y sintaxis del programa, asimismo, es una base para que los alumnos de ciencias sociales apliquen los modelos para contrastar hipótesis.

Autoevaluación 1

Problema 1. Un ayudante de posgrado necesita auxiliar a su coordinador para obtener una base de datos con información de egresados del doctorado según el área, del año 2001 al 2006. Para Ingeniería se tienen los siguientes datos: 5323, 5511, 5079, 5280, 5777, 6425; Ciencias: 20643, 20017, 19529, 20001, 20498, 21564; Educación: 6436, 6349, 6503, 6643, 6635, 6226; Salud: 1591, 1541, 1654, 1633, 1720, 1785; Humanidades: 5213, 5178, 5051, 5020, 5013, 4949; Sociales: 2159, 2141, 2209, 2180, 2480, 2436. Elabore la base de datos.

- ¿Cuántas variables y registros tiene la base de datos elaborada en el problema 1?
- ¿Qué tipo de dato contiene cada una de las variables?
- La variable año, ¿qué tipo de dato es?

Problema 2. Elaborar una gráfica que en el libro se definió como el modelo que permite representar la distribución de muestras o casos en determinadas categorías, sólo para el año 2001, por cada una de las áreas del problema 1.

- ¿Qué se muestra en el gráfico del problema 2?
- El problema 2 se puede graficar como un histograma: ¿sí o no? ¿Por qué?
- ¿Se puede obtener un análisis chi-cuadrada con todas las observaciones de ingeniería y ciencia del problema 1?
- ¿Qué tipo de medida tiene la variable sociales del problema 1?

Autoevaluación 2

Lea detenidamente cada una de las preguntas y conteste. Consulte la página de internet del INEGI para descargar los datos del Índice de Rezago Social para los años 2000, 2005 y 2010, sólo para el Estado de México (https://www.snieg.mx/cni/serieestadisticas.aspx?idOrden=1.2&IndBase=6207051588*6207051589*6207051590*6207051591*6207051592*6207051593*6207051594*6207051595*6207051596*6207051597*6207051598*6207123199&indCve=6200009453&gen=517&d=s).

Genere una base de datos con el Índice de Rezago Social para los años 2000, 2005 y 2010 con todos los municipios del Estado de México. Realice los siguientes análisis:

- Generar una distribución del índice por año y municipio para la Región Metropolitana de Toluca. Los municipios de la zona se especifican en la nota siguiente.
- Generar una comparación de medias para la Región Metropolitana de Toluca, contrastando 2010 y 2005 con el año 2000.

#¡Suerte! ;)

Nota: considerar los siguientes municipios para la Región Metropolitana de Toluca:

Toluca, Metepec, Zinacantepec, Almoloya de Juárez, Lerma, Otzolotepec, San Mateo Atenco, Ocoyoacac, Calimaya, Xonacatlán, San Antonio la Isla, Rayón, Mexicaltzingo y Chapultepec.

CAPÍTULO 7

CREACIÓN DE MAPAS

La visualización de datos para demostrar la relación existente entre variables es muy importante en ciencias sociales, sin embargo, un gran número de estudiantes e investigadores han dejado de lado esta acción. La falta de representación en los datos dificulta el entendimiento de los fenómenos subyacentes, evidenciar patologías sociales y aportar conocimiento valioso. El *software* contiene suficientes complementos para crear elegantes y poderosos gráficos, el más notable es el paquete *ggplot2*. En la actualidad, es imperativo desarrollar visualizaciones geoespaciales, por tanto, en el presente apartado se presenta el paquete *tmap*, que como lo sustenta Tennekes (2018) es una manera fácil de usar recurriendo a capas, *shapes*, para elaborar mapas temáticos. También se encuentran disponibles los siguientes paquetes: *tmap*, *rgdal*, *Spatial*, *RQGIS sf*, *raster*, *spData*, *maps*, *mapproj*, *osm*, *RgoogleMaps*, *maptools*, *shapefiles*, *ggmap* (Bivand et al., 2013; Lovelace et al., 2019; Barker, 2013; Abedin y Mittal, 2014; Walker, 2016; Wickham, 2016).

Los mapas temáticos son mapas geográficos en los que se muestran las distribuciones de datos espaciales. Son ampliamente utilizados en publicaciones científicas y notas periodísticas. Uno de los ejemplos más recurrentes es el mapa de puntos con el que el médico John Snow solía localizar la fuente del brote de cólera en Londres durante la segunda mitad del siglo XIX (Snow, 1849).

Los laboratorios de los diferentes departamentos de Ciencias Sociales de la Universidad Autónoma Metropolitana han recurrido a dichos instrumentos para analizar temas como el cambio climático, resiliencia ambiental, participación política en las elecciones para la presidencia de la República mexicana,

elecciones intermedias para la conformación de la Cámara de Diputados, el cáncer y la especialización económica. Dichas exploraciones, sin lugar a duda, han permitido observar las variables espaciales, cada vez menos subyacentes, que contribuyen a explicar los fenómenos sociales. Entre los diferentes mapas que podemos utilizar se encuentran los enunciados a continuación.

Actualmente, el Instituto Nacional de Estadística y Geografía (INEGI) utilizó el *software R* en el Sistema para la Consulta de Información Censal 2020 (SCINCE) señalado en sus metadatos como un producto de R Foundation for Statistical Computing. Asimismo, menciona para el análisis multivariable por el método de Componentes Principales y Dalenius-Hodges, estratificación multivariada, dado que se incorporó al programa una implementación para que los cálculos utilicen la sintaxis. Dedica unos cuantos párrafos a la descripción del programa, su instalación, la dirección de descarga. El documento describe de manera detallada cada paso para la instalación correcta, hasta enunciar las librerías principales para llevar a cabo análisis estadísticos: *sp*, *spdep*, *stratification* y *Matrix*. Para ejecutarlo, primero se debe acudir a la opción de *Datos relativos*, continúe en la sección *Estratificación Multivariada*. Los resultados del modelo se pueden visualizar a través de un mapa temático (INEGI, 2021).

Las coropletas permiten visualizar regiones, países, localidades y municipios. A través de los colores se representan concentraciones y dispersiones, la densidad de una variable. Los intervalos de clase asientan la legibilidad de los valores (Lovelace et al., 2019; De Vries y Meys, 2015; Wickham, 2016; INEGI, 2021). El número idóneo de clases es impar, puede ser tres, cinco o siete, asimismo, se obtiene con los cuartiles (MacEachren, 1994; Brewer y Pickle, 2002; Wickham, 2016).

En el mapa categórico las regiones reales están llenas de colores que representan una variable. Los mapas categóricos a menudo se confunden con coropletas (Lovelace et al., 2019; Wickham, 2016; Barker, 2013).

Otro tipo es el mapa de burbujas, también llamado de símbolos proporcionales. Los puntos de datos están codificados por símbolos escalonados en proporción a una variable. Las burbujas son los símbolos más utilizados (Tennekes, 2018; De Vries y Meys, 2015).

En el mapa de contorno, las líneas se dibujan a través de puntos de datos igualmente valorados. Dado que los puntos de datos a menudo se muestrean, estas líneas deben ser aproximadas de acuerdo con un estimador bidimensional de densidad de Kernel, método útil cuando la distribución de datos no es

uniforme, lo que puede dar lugar a un atasco de líneas de contorno. La suavización de los datos espaciales con este estimador puede dar como resultado líneas delgadas que proporcionan una mejor comprensión en los patrones generales de la distribución de datos (Munzert et al., 2015; Wickham, 2016). Las isopletras se usan comúnmente en mapas de elevación y clima (Wickham, 2016; Barker, 2013).

El mapa de desimetría es similar a la coropleta, excepto que los polígonos no están predefinidos. Áreas definidas por niveles de la variable, codificada por el color. La ventaja de este mapa es que los polígonos no están predefinidos (Wickham, 2016).

En el mapa de puntos, los puntos son de menor tamaño que el mapa de burbujas, el mapa de puntos se basa en ubicaciones especiales en lugar de áreas en general, los mapas de puntos contienen gran cantidad de estos, que lo diferencian del mapa ya señalado (De Vries y Meys, 2015; Slocum et al., 2009; Tyner, 2010).

El mapa de trama son celdas, cuadrícula por cuadrícula, se colorean de acuerdo con los valores de una variable de datos. Los mapas ráster son especialmente populares en geología y agroecología, donde la cobertura terrestre es representada por el valor de uso del suelo (Lovelace et al., 2019).

Finalmente, el cartograma muestra áreas distorsionadas, de tal forma que los tamaños de área obtenidos son proporcionales a una variable cuantitativa (Gastner y Newman, 2004; Nusrat y Kobourov, 2016). En un cartograma de área contigua, las formas de los polígonos están distorsionadas, donde las relaciones de vecindad entre ellos se conservan tanto como sea posible. Cuando no son contiguos, los polígonos sólo se redimensionan. El proyecto World Mapper (<http://www.worldmapper.org>) contiene un par de ejemplos ilustrativos, un mapa mundial distorsionado en el que las áreas de los países son proporcionales al tamaño de su población (Barford y Dorling, 2008).

La visualización es una herramienta para describir un estadístico de forma gráfica. Las variables se asignan como posición, tamaño, color y forma. Un mapa tiene una o más capas, cada capa está definida por una geometría. El sistema de coordenadas determina la estética de la posición. Mientras que las coordenadas están determinadas por datos mapeados, las coordenadas de una burbuja están determinadas por ubicaciones geográficas. El sistema de coordenadas se llama sistema de referencia, que generalmente se conoce como la proyección del mapa. Determina cómo se proyectan las coordenadas de

longitud-latitud desde la superficie terrestre a un plano bidimensional (Slocum et al., 2009; Tyner, 2010; Barker, 2013; Wickham, 2016; Abedin y Mittal, 2014; Walker, 2016). El *shape* se refiere a un objeto que contiene los siguientes componentes.

Las coordenadas y topología son requeridas para definir las unidades espaciales. Cuatro tipos básicos pueden ser distinguidos: polígonos, líneas, puntos y ráster. Una unidad espacial puede contener múltiples líneas y polígonos (Abedin y Mittal, 2014; Barker, 2013; Walker, 2016; Tyner, 2010; Slocum et al., 2009). La ciudad es un buen ejemplo de los anteriores geo-objetos mencionados (polígonos, líneas y puntos).

El formato de los datos es tabular, las filas corresponden a los registros espaciales y las columnas a las variables de datos. La proyección del mapa define cómo son las coordenadas de longitud y latitud (Abedin y Mittal, 2014; Walker, 2016; Barker, 2013). El cuadro delimitador determina qué parte del mapa se muestra.

La transformación espacial determina la forma y composición, puede estar relacionada con las coordenadas, topología y los datos. Los tres ejemplos comunes son: 1) un cartograma donde los polígonos se muestran de acuerdo con los valores de una variable de datos numéricos; 2) un mapa de densidad Kernel bidimensional, donde se aplica el estimador a una forma que consiste en puntos, y 3) mapa de trama, donde los polígonos o puntos espaciales se convierten a un espacio ráster, junto con los datos correspondientes (Munzert et al., 2015; Abedin y Mittal, 2014; Walker, 2016; Lovelace et al., 2019).

La geometría determina qué se muestra, por ejemplo, polígonos, símbolos o etiquetas de texto. La disponibilidad de las geometrías depende del tipo de unidades espaciales en el geo-objeto de forma. Obviamente, las unidades de polígono pueden ser visualizadas con una geometría de polígono, pero también con una geografía de símbolos mediante el uso de centroides del polígono como coordenadas. Lo contrario no es posible, es decir, unidades de puntos no se pueden visualizar con una geometría de polígono (Lovelace et al., 2019; Abedin y Mittal, 2014; Walker, 2016).

Un mapa temático se construye como un bloque cuya función, después de activar la librería *tmap*, inicia siempre con el prefijo *tm_*, y se apila con el operador (+). Ejemplo: para obtener la forma y proyección deseadas, se utiliza la función *tm_map* (Tennekes, 2018). La proyección deseada puede tomar características específicas de acuerdo con la sintaxis *PROJ.4; EPSG* números de

proyección o caracteres de acceso directo para proyecciones de mapas comúnmente utilizados como *longlat* y *wintri*, para las coordenadas de longitud-latitud, y la proyección Triple de Winkel, respectivamente (Barker, 2013; Abedin y Mittal, 2014; Lovelace et al., 2019) (tabla 20).

Tabla 20. Funciones para diseñar un mapa temático

| Función | Geometría | Unidad espacial | Aesthetics |
|-------------|----------------------------------|----------------------------|------------------|
| tm_polygons | polígonos | polígonos | col, fill color |
| tm_symbols | símbolos | puntos, polígonos y líneas | size symbol size |
| tm_lines | líneas | líneas | col |
| tm_ráster | celdas | celdas | col |
| tm_text | texto | punto, polígonos y líneas | text text labels |
| tm_fill | polígonos | polígonos | |
| tm_borders | polígonos | polígonos | |
| tm_bubbles | burbujas | puntos, polígonos y líneas | |
| tm_squares | cuadrados | puntos, polígonos y líneas | |
| tm_dots | puntos | puntos, polígonos y líneas | |
| tm_rgb | imagen rgb | celdas | |
| tm_markers | símbolos con etiquetas | puntos, polígonos y líneas | |
| tm_iso | líneas de contorno con etiquetas | líneas | |

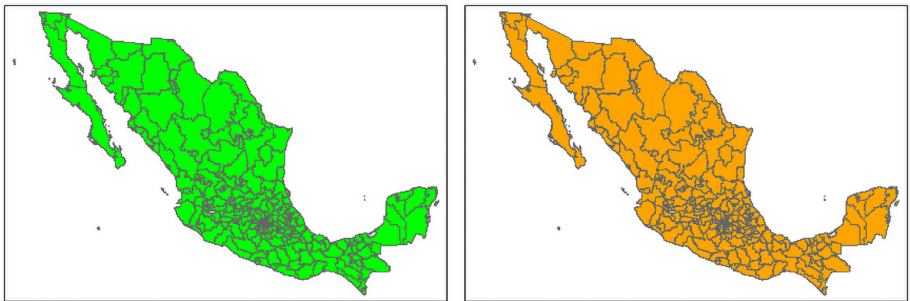
Fuente: elaboración propia con base en Tennekes (2018).

Se presentará el primer mapa con base en la cartografía electoral nacional a escala de distritos, 300 polígonos, para el año 2005. Éste se puede descargar desde la página del INEGI o a través del siguiente vínculo electrónico: <https://gaia.inegi.org.mx/mdm6/?v=bGF0OjIzLjMyMDA4LGxvbjotMTAxLjUwM-DaWlHo6MSxsOmNnZW9lbGVjdG9yYWxlcw==&theme=goelectorales>

No se analizará variable alguna, sólo se reproducirá el mismo mapa en dos colores, verde y naranja (figura 36).

```
tm_shape(Fede) + tm_polygons(col = c("green", "orange"))
```

Figura 36. Distritos electorales nacionales, 2005



Fuente: elaboración propia con base en IFE (2005; 2015).

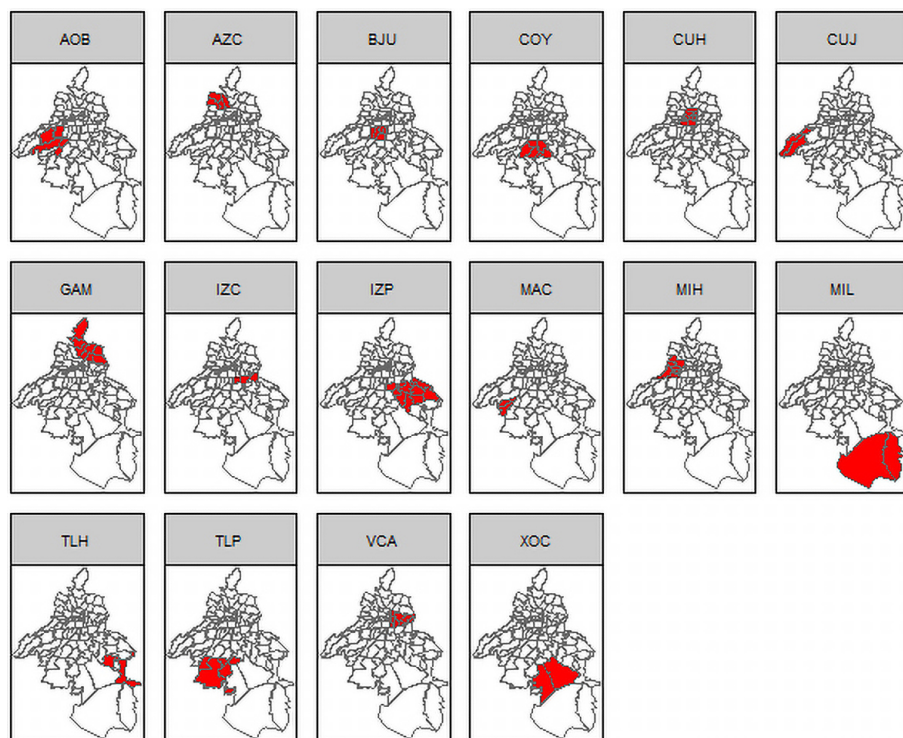
El segundo método para diseñar múltiplos es mediante la asignación de una variable, comúnmente referida a un segmento del *shape*, el argumento *tm_facets* divide el territorio según la variable, observaremos los sectores policiacos por Alcaldía para la Ciudad de México (CDMX) (tabla 21, figura 37).

Tabla 21. Función para comparar mapas temáticos con facetas

| Función | Descripción | Argumentos |
|-----------|--------------------|--------------------|
| tm_facets | Pequeños múltiplos | Coordenadas libres |
| | | Escala libre |

Fuente: elaboración propia con base en Tennekes (2018).

```
tm_shape(sectorespoliciacos) + tm_polygons(col = "red") +  
  tm_facets(by = "CVEDEL", free.coords = FALSE)
```

Figura 37. Sectores policiacos por Alcaldía en la CDMX, 2008

Fuente: elaboración propia con base en el OCCM (2009).

Otro método que permite compilar mapas en una sola presentación es la función *tmap_arrange*. A continuación, se muestran dos mapas de la Ciudad de México, el de la izquierda señala, a través de puntos, los principales hospitales públicos y privados, y el de la derecha expone las líneas del Metro que hasta el año 1997 se habían construido (figura 38). La base de datos de los hospitales públicos y privados en la Ciudad de México se encuentra en los anexos 1.1 y 1.2.

```
modelo03<-tm_shape(redmetrodf)+tm_lines()
modelo04<-tm_shape(ubicaestmetrodf)+tm_dots()
tmap_arrange(modelo04, modelo03)
```

**Figura 38. Ciudad de México:
hospitales y red del metro**



Fuente: elaboración propia con base en la cartografía del OCCM (2009).

Los argumentos relacionados con un mapa, tales como márgenes, escala, título, leyenda y atributos, muchas veces dependen de la geometría de la imagen, se presentan con la función *tm_layout*. La función *style_catalogue* genera un catálogo de todos los estilos disponibles. La función *tm_view* contiene los argumentos que son específicos para el modo de visualización.

La función *qtm* es adecuada para un trazado rápido de mapa temático. Lo que vamos a ejecutar es un mapa fabuloso, donde por cada estación del metro de la Ciudad de México podamos visualizar la densidad de población que asiste a tal ubicación, los datos son del año 1997 (consúltase anexo 1.2 y figura 39), éste se construyó con la sintaxis que se muestra a continuación. Se puede observar que primero convocamos la figura, *shape*, posteriormente, se define la variable que se va a representar por los diferentes tamaños que muestran la medida de los datos, finalmente, se señala el formato y estilo del mapa.

En dicho mapa se puede observar que las líneas del metro que concentran mayor cantidad de usuarios tienen una construcción histórica, son las tres primeras diseñadas para planificar la movilidad en la Ciudad de México.

```
qtm(redmetrodf, symbols.size = "DENSIDAD_P", format =  
"world", style = "cobalt")
```

```
+ tm_text("LINEA", size = 1, col = "gray30", root = 3,
remove.overlap = TRUE)
```

**Figura 39. Ciudad de México:
densidad poblacional en cada estación del metro, 1997**



Fuente: elaboración propia con base en la cartografía del OCCM (2009).

En el modo interactivo, la función *qtm* puede llamar figuras sin argumentos sólo con una búsqueda. Cuando se proporciona una consulta de búsqueda, un marcador de lugar se agrega al mapa en las coordenadas encontradas por el *OpenStreetMap*, motor de búsqueda Nominatim.

Los datos *OpenStreetMap* (OSM) existen en dos formatos: vectores y ráster. Regularmente, los vectores se usan cuando se analizan datos OSM y ráster cuando se grafican datos OSM como una capa de fondo. Ambos se pueden leer con *read_osm*, requiere de un cuadro delimitador o un objeto de forma desde el cual el cuadro delimitador se extrae. La función *read_osm* requiere el paquete sugerido, *R OpenStreetMap*, el cual se basa en Java para estar instalado (Barker, 2013; Abedin y Mittal, 2014; Wickham, 2016; Walker, 2016).

La función *bb* es una opción súper ágil con respecto a los cuadros delimitadores (Barker, 2013; Abedin y Mittal, 2014). Se pueden recuperar y modificar cuadros de delimitación existentes con el argumento del factor de extensión *ext* y re proyectarlos. Además, permite crear nuevos cuadros delimitadores desde cero, ya sea usando coordenadas o consultando *OpenStreetMap Nominatim*. Para este último, se recurre a la función *geocode_OSM*.

A menudo, los datos espaciales aún no están contenidos en un objeto de forma. Los datos pueden ser anexados a un objeto de forma con la función *append_data*. En la terminología de la base de datos, la unión, *join*, se ejecuta, es decir, los datos coincidentes se adherirán a las unidades espaciales correspondientes donde los valores faltantes se usarán para unidades espaciales sin coincidencia.

El complemento *tmaptools* contiene un par de funciones de transformación espacial. Las funciones *set_projection* y *crop_shape* se pueden usar para establecer la proyección del mapa y recortar la forma por un cuadro delimitador, respectivamente. Tanto la proyección del mapa como el cuadro delimitador de igual forma se pueden especificar directamente en el código de la trama como argumentos de *tm_shape*. Además, con *crop_shape*, también es posible recortar una forma por un polígono en lugar de un rectángulo. La función *simplify_shape* reduce el número de coordenadas utilizadas para describir polígonos y líneas. La función *smooth_map* crea un mapa mediante la estimación de una densidad de Kernel bidimensional de la distribución de datos espaciales de la forma de entrada (Barker, 2013; Lovelace et al., 2019; Abedin y Mittal, 2014).

Las densidades de Kernel bidimensionales, según Wand (1995), se calculan con el paquete KernSooth (tabla 22).

Tabla 22. Sintaxis para elaborar un mapa

| Función | Descripción | Argumento | |
|-----------------|--------------------------------|-----------|-------------------|
| read_shape | Montar <i>ESRI shape files</i> | File | |
| read_osm | Capa de <i>OpenStreetMap</i> | X | |
| geocode_OSM | Búsqueda de localización | Q | Búsqueda |
| rev_geocode_OSM | Dirección | x-y | Coordenadas x-y |
| Bb | Caja delimitadora | X | shape |
| append_data | Anexar datos al <i>shape</i> | Shp | shape |
| approx_areas | Áreas próximas | Shp | shape |
| set_projection | Proyección-reproyección | Shp | código-proyección |
| simplify_shape | Simplificación de <i>shape</i> | Shp | shape |
| crop_shape | Recortar <i>shape</i> | x-y | shape |
| aggregate_map | Agregar un mapa | Shp | shape |
| smoth_map | Un mapa sin problemas | Shp | shape |
| cartogram | Cartograma | Shp | shape |

Fuente: elaboración propia con base en Lovelace et al. (2019), De Vries y Meys (2015), Barker (2013).

Ahora vamos a ejecutar algunos de los comandos expuestos en la tabla anterior para obtener las edificaciones en la Zona Metropolitana de la Ciudad de México, las cuales se presentan casi en tiempo real, puesto que los datos son una extracción de la aplicación *StreepMap*, de la cual se pueden realizar consultas sobre las líneas del metro, oficinas de gobierno, unidades económicas, lugares de recreación, es decir, tiene un menú amplísimo para relacionar geo-objetos con diversos fenómenos sociales. Revise la siguiente sintaxis y aplíquela a su tema de estudio.

```

```{r}
###extraer la ciudad de mexico
library(OpenStreetMap)
library(osmdata)
library(osmplotr)
#mapa base zmcm
zmcm<- get_bbox(c(-99.37546,19.58963,-98.56522,19.3151))
#construcciones
zmcm1<-extract_osm_objects(key = 'building', bbox = zmcm)
contornozmcm <- osm_basemap(zmcm, bg = 'gray20')
contornozmcm<- add_osm_objects(contornozmcm,zmcm1,col = 'gray40')
print_osm_map (contornozmcm)
```

```

Figura 40. ZMVM: Construcciones, 2021



Fuente: elaboración propia con base en la librería *OpenStreetMap*; *osmdata*; *osmplotr*.

Comparativamente, podemos observar entre las figuras 40 y 41 construcciones y manzanas, respectivamente, se observa la densificación, pero las manzanas contienen información sobre áreas que tienen diferente uso de suelo, lo que se descarta en la búsqueda de *StreetMap* en torno a los inmuebles de tipo habitacional.

Figura 41. ZMVM: Manzanas, 2021



Fuente: elaboración propia con base en INEGI (2021).

Dentro del mapa se puede observar en la parte superior derecha, encerrada en un círculo, una forma humana de lo que pudo haber sido el Nuevo Aeropuerto de Texcoco, sin embargo, se reubicó el plan.

Vamos a contrastar la información con la forma de los distritos que integran la movilidad del área con base en la información de manzanas del Censo 2020 (INEGI, 2021).

Ejemplo

Vamos a realizar un mapa para representar el Índice de Desarrollo Humano en la Zona Metropolitana del Valle de México (ZMVM) para el año 2000, cuando el número de municipios que comprendían dicha región era de 54, según el Instituto Nacional de Estadística y Geografía (anexo 1.3). La variable IDH se va a distribuir en cinco rangos que están representados por el color azul, cuando es más fuerte el color, señala que el desarrollo es mayor que en los polígonos donde el color azul es más tenue. Asimismo, sobrepuesta se encuentra, con colores muy suaves, la capa de la red del metro, la cual modifica su color dependiendo de la densidad de población que la asiste para realizar un desplazamiento por la zona (figura 42).

```
tm_shape(zmvm2000)+tm_polygons("IDH", palette = "Blues",
  title = "IDH", contrast = 0.7, border.col = "gray30")+
  tm_text("CLVMUN", size = "AREA", col = "gray30", root = 3)+
  tm_shape(redmetrodf) + tm_dots("DENSIDAD_P",
  border.col = "black", border.alpha = 0.5,
  palette = "-RdYlGn",
  title.size = "Densidad de metro (1997)",
  popup.vars = c("IDH", "DENSIDAD_P"))
```

Las etiquetas de texto en mapas temáticos a menudo no se pueden ver. Formas comunes de prevenir esto es usar diferentes tamaños de fuente, ocultar etiquetas de texto menos importantes y cambiar la ubicación de las etiquetas. Dichas opciones se pueden habilitar utilizando la especificación de *tm_text*. Los argumentos *auto.placement* y *remove.overlap* se pueden usar para cambiar la ubicación de las etiquetas de texto (Tennekes, 2018).

representado por el área. Estos nombres se incluyen en el texto emergente como títulos (Tennekes, 2018).

Output

La función de impresión dibuja el mapa desde un objeto *tmap*, estático o interactivo, basado en el modo elegido. El diseño del mapa está configurado para el tamaño original y la relación de aspecto de la ventana del dispositivo gráfico. La función *last_map* se utiliza para volver a dibujar el último mapa en un tamaño redimensionando (Tennekes, 2018).

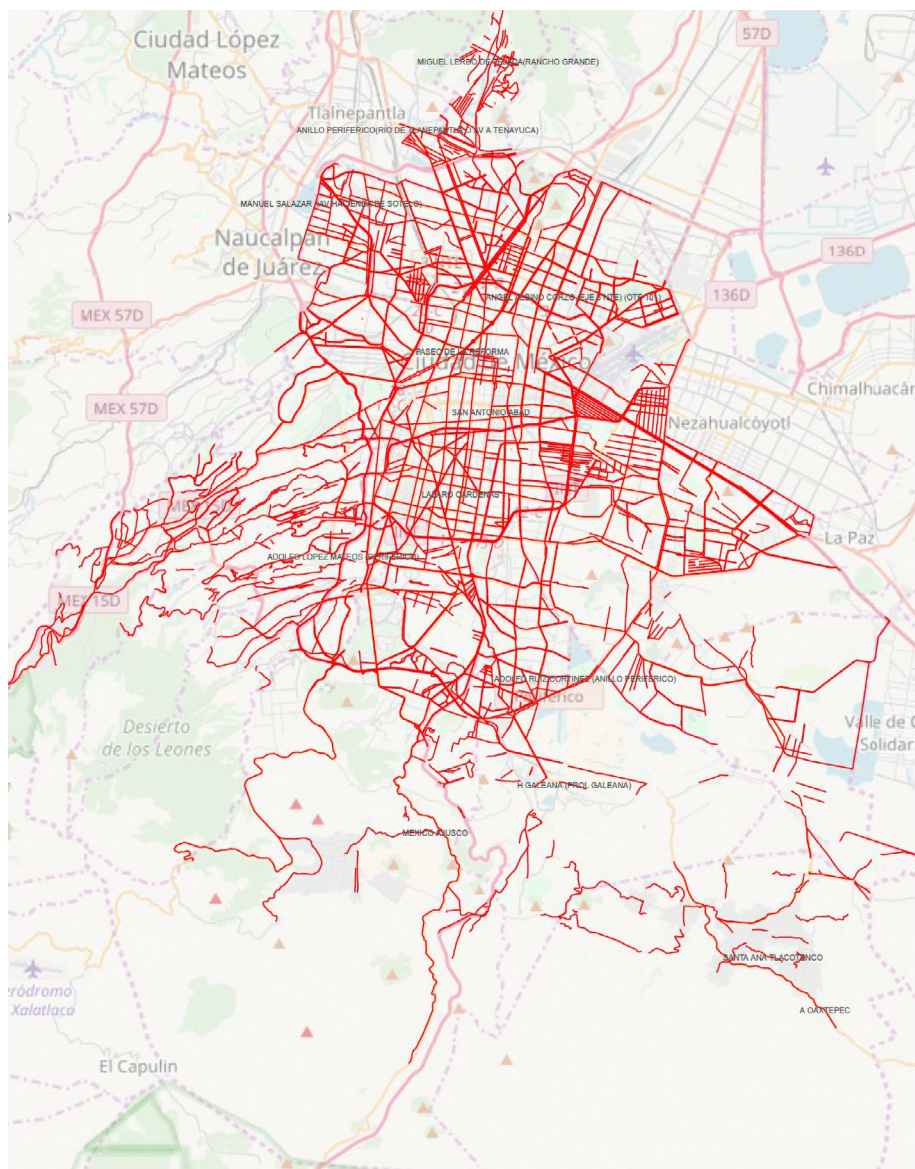
Los mapas realizados con *tmap* se pueden guardar fácilmente en los formatos habituales, jpg, png y pdf, con *save_tmap*. Si sólo se especifica una de las dimensiones, ancho y alto, la otra se establecerá automáticamente en función de la relación de aspecto de la trama, que depende de la relación de aspecto de la forma. La función *save_tmap* puede exportar mapas interactivos a páginas HTML independientes (Tennekes, 2018).

Cuando se instala la herramienta de *software* libre *ImageMagick*, se puede hacer con *animation_tmap*, especialmente, para visualizar una serie de datos espaciales. El argumento principal de *animation_tmap* es un objeto *tmap* que resulta en múltiples parcelas. Se logra mediante la producción de pequeños múltiplos trazados en una página separada, que se consigue utilizando el argumento *tmap_facets* (Tennekes, 2018).

Vamos a continuar con un mapa que representa las calles de la Ciudad de México, para ello, con la librería *OpenStreetMap* conseguiremos georreferenciar el *shape* de calles en el mapa que nos ofrece la plataforma de la librería mencionada (Abedin y Mittal, 2014; Wickham, 2016; Walker, 2016). Se realizó la selección de las avenidas principales (figura 43). El *shape* de calles se localiza en los datos abiertos del Gobierno del Distrito Federal, o bien, en la página del Instituto Nacional de Estadística y Geografía.

```
qtm(callescdos, raster.alpha = 0.5) +
  qtm(callescd[callescd$CATEGORIA==1, ])+
  tm_text("NOMBRE", size = 0.3, auto.placement=F, col =
    "gray30", root = 3, remove.overlap = TRUE)
```

Figura 43. Ciudad de México, avenidas principales



Fuente: elaboración propia con base en INEGI (2010a); Abedin y Mittal, 2014; Wickham, 2016; Walker, 2016).

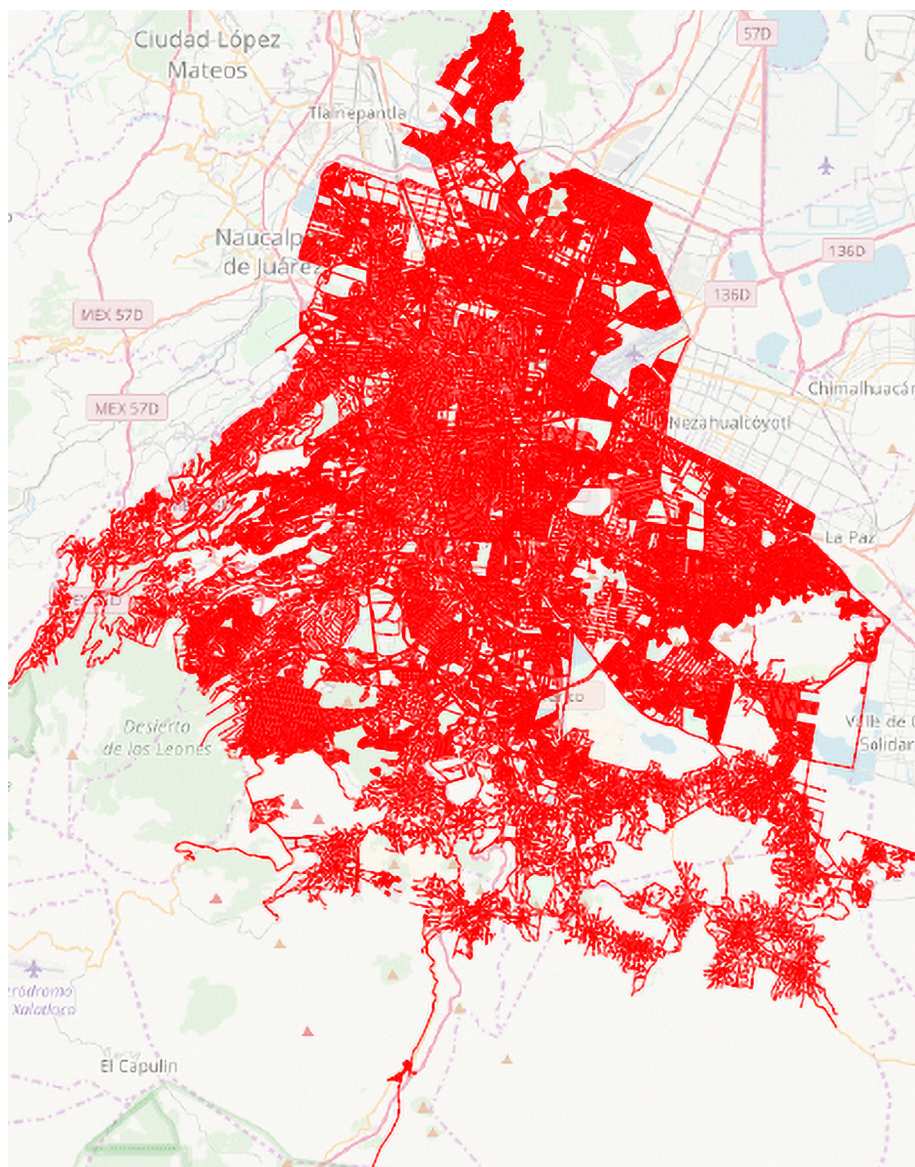
Dado que dicho mapa se produjo con una capa que contiene las avenidas primarias, secundarias y terciarias de la Ciudad de México, un gran aporte es la selección de avenidas y colocar el nombre de éstas sin afectar la visibilidad para generar análisis (figura 44).

```
library(tmaptools)
library(OpenStreetMap)
qtm(callescdos, ráster.alpha = 0.5) +
qtm(callescd)
```

Bibliografía recomendada: profundizar con los autores

- Abedin, J. y Kishor Kumar, D. (2015). *Data manipulation with R*. Packt.
- Barker, T. (2013). *Pro data visualization using R and JavaScript*. Apress.
- Beeley, C. (2013). *Web application development with R using Shiny*. Packt.
- Bilder, C. (2013). *Analysis of categorical data with R*. Chapman & Hall/CRC.
- Broman, K. W. y Sen, S. (2009). *A guide to QTL mapping with R/qtl*. Springer.
- Chang, W. (2013). *R Graphics Cookbook*. O'Reilly Media.
- De Vries, A. y Meys, J. (2015). *R for dummies*. John Wiley & Sons.
- Tennekes, M. (2018). Tmap: thematic maps in R. *Journal of Statistical Software*, 84(6), 1-39.
- Urbano, F. y Cagnacci, F. (2014). *Spatial database for GPS wildlife tracking data: a practical guide to creating a data management system with PostgreSQL/PostGIS and R*. Springer.
- Verzani, J. (2011). *Getting started with RStudio*. O'Reilly Media.
- Walker, K. E. (2016). Tools for interactive visualization of global demographic concepts in R. *Spatial Demography*, 4(3), 207-220.
- Wickham, H. (2016). *Ggplot2. Elegant graphics for data analysis*. Springer.

Figura 44. Ciudad de México, avenidas primarias, secundarias y terciarias



Fuente: elaboración propia con base en INEGI (2010a; Abedin y Mittal, 2014; Wickham, 2016; Walker, 2016).

CAPÍTULO 8

AUTOEVALUACIÓN PARA GENERAR UN MAPA

Vamos a realizar la última evaluación, donde pondremos a prueba la habilidad para modelar un mapa en el *software R*.

Como primer ejercicio, hay que acudir a la página electrónica del Grupo de Investigación en Ingeniería de Transporte y Logística (GiiTral), donde se va a descargar el archivo *shape*, cuya extensión es *.shp*, de los distritos de la Encuesta Origen Destino 2017. La liga para acceder al portal es: <https://giitral.ingen.unam.mx/Estudios/EOD-Hogares-01.html#distritos>

En la sección de encuestas en hogares, debajo de la imagen que contiene el mapa 1, se encuentra el ícono *Descargar Archivo shp*, el cual contiene cuatro archivos con el mismo nombre, pero diferente extensión, puesto que:

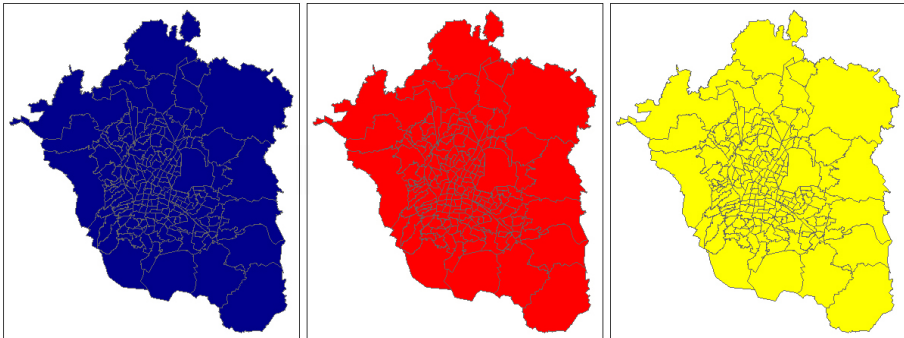
- a) Tienen información sobre el banco de datos. Información de atributos (.dbf);
- b) Sistema de coordenadas (.prj);
- c) La geometría (.shp);
- d) El índice de la geometría (.shx). DistritosEODHogaresZMVM2017.

Problema 1. Montar la cartografía a la plataforma *R* a través de un vector que se denomine EOD2017. Cabe señalar que la función se describió en el capítulo 7, se debe elegir la ruta donde se encuentra el archivo ("C:/.."), en caso de copiarla, Microsoft presenta las inclinadas de la siguiente manera (\), cambiarlas por el lado contrario (/); toda la sintaxis se abraza por comillas.

- ¿Cuántas observaciones y variables integran el *shape* DistritosEODHogaresZMVM2017?
- Las observaciones del *shape* poseen una escala geográfica, ¿cuál es?
- ¿Qué número de distrito tiene San Cristóbal Ecatepec?

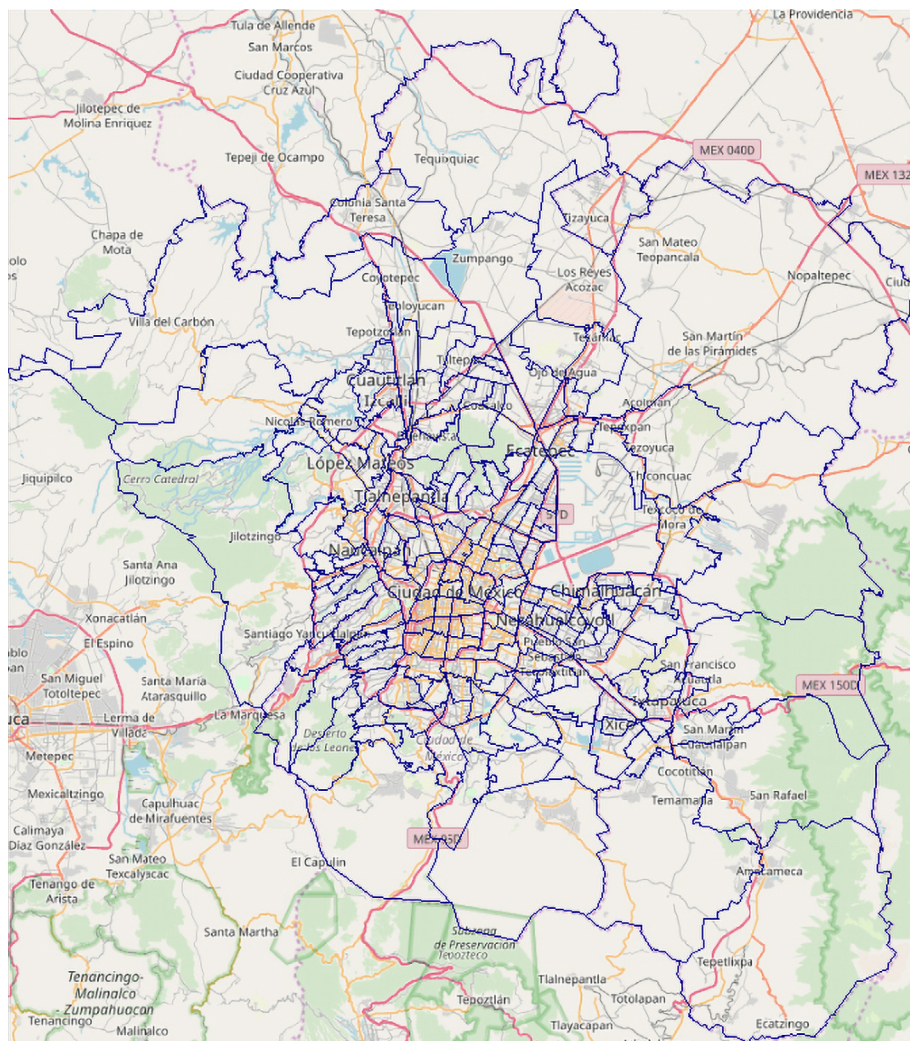
Problema 2. Presente el mapa de EOD2017 con tres colores diferentes, azul, rojo y amarillo, tal como se muestra a continuación.

- ¿Dónde se concentran los distritos más extensos? Realice una interpretación del porqué se encuentran de la siguiente forma.
- ¿Se puede hablar de una región de distritos compactos y distritos extensos para entender la movilidad en la ZMVM?



Problema 3. Yuxtaponer dos capas que permitan visualizar la región a través de una imagen de satélite y los polígonos de los distritos de la EOD 2017, tal como se muestra a continuación. Cabe señalar que cuando se asocia información a un *shape*, se conjunta con el símbolo de más (+).

- ¿Cuáles son los municipios colindantes al distrito de Tizayuca?
- Analice las vías de comunicación del distrito Tizayuca. ¿Se puede observar conexión desde este distrito hacia Tula? ¿En el distrito mencionado se encuentran mayores vías de accesibilidad a Tula que al noreste del Estado de México o al norte y centro de la CDMX?



CAPÍTULO 9

BIBLIOGRAFÍA “R”ECOMENDADA

A nivel internacional, se encuentra disponible gran cantidad de bibliografía para el uso, programación y análisis con el *software R*. Dichas obras detallan las funciones básicas, exponen algunos ejemplos y desarrollan inferencias a través de la estadística descriptiva, subsecuentemente, realizan algunas modelaciones para inferir asociación entre dos o más variables; en otros casos, se especializan en algoritmos financieros, de marketing, o bien, en la representación gráfica. Para que los alumnos puedan ahondar en algún tema sobre los expuestos en el libro y también quieran conocer y se apasionen del *software*, este capítulo se aboca a mostrar los libros que en la actualidad y a mi parecer son de gran relevancia y se expresan de la manera más ordenada y creativa.

- Abedin, J. y Kumar, D. K. (2015). *Data manipulation with R* (2a Ed.). Packt.
- Abedin, J. y Mittal, H. (2014). *R graphs cookbook* (2a Ed.). Packt.
- Adler, J. (2012). *R in a nutshell: a desktop quick reference* (2a Ed.). O'Reilly Media.
- Agresti, A. (2002). *Categorical data analysis* (Vol. 482). John Wiley & Sons.
- Agresti, A. (2015). *Foundations of linear and generalized linear models*. Wiley.
- Aho, K. (2013). *Foundational and applied statistics for biologists using R*. Chapman & Hall/CRC.
- Albert, J. y Rizzo, M. (2012). *R by example: concepts to code*. Springer.
- Allerhand, M. (2011). *A tiny handbook of R*. Springer.
- Altham, P. (2012). *Introduction to statistical modelling in R*. University of Cambridge.

- Ang, C. (2015). *Analyzing financial data and implementing financial models using R*. Springer.
- Aragón, T. (2013). *Applied epidemiology using R*. University of California.
- Armstrong II, D., Bakker, R., Carroll, R., Hare, C., Poole, K. T. y Rosenthal, H. (2014). *Analyzing spatial models of choice and judgment with R*. Chapman & Hall/CRC.
- Arnold, T. y Tilton, L. (2015). *Humanities data in R: exploring networks, geospatial data, images, and text*. Springer.
- Arratia, A. (2014). *Computational finance: an introductory course with R*. Atlantis.
- Auerbach, C. y Zeitlin, W. (2015). *Making your case: using R for program evaluation*. Oxford University Press.
- Baayen, R. (2008). *Analyzing linguistic data: a practical introduction to statistics using R*. Cambridge University Press.
- Barford, A. y Dorling, D. (2008). Telling an old story with new maps. En M. Dodge, M. McDerby y M. Turner (Eds.), *Geographic visualization: concepts, tools and applications* (pp. 67-107). John Wiley & Sons.
- Barker, T. (2013). *Pro data visualization using R and JavaScript*. Apress.
- Bartolucci, F., Bacci, S. y Gnaldi, M. (2015). *Statistical analysis of questionnaires: a unified approach based on R and Stata*. Chapman & Hall/CRC.
- Beaujean, A. (2014). *Latent variable modeling using R: a step-by-step guide*. Routledge.
- Beeley, C. (2013). *Web application development with R using Shiny*. Packt.
- Berridge, D. y Crouchley, R. (2011). *Multivariate generalized linear mixed models using R*. CRC Press.
- Beyersmann, J., Allignol, A. y Schumacher, M. (2012). *Competing risks and multistate models with R*. Springer.
- Bilder, C. y Loughin, T. (2015). *Analysis of categorical data with R*. Taylor & Francis.
- Bivand, R., Pebesma, E. y Gómez-Rubio, V. (2013). *Applied spatial data analysis with R*. Springer.
- Black, K. (2014). *R object-oriented programming*. Packt.
- Bogetoft, P. y Otto, L. (2011). *Benchmarking with DEA, SFA, and R*. Springer.
- Boogaart, K. y Tolosana-Delgado, R. (2013). *Analyzing compositional data with R*. Springer.
- Boos, D. y Stefanski, L. (2013). *Essential statistical inference: theory and methods*. Springer.

- Braun, W. y Murdoch, D. (2007). *A first course in statistical programming with R*. Cambridge University Press.
- Bretz, F. y Hothorn, T. (2011). *Westfall P. Multiple comparisons using R*. Chapman & Hall/CRC.
- Brewer, C. y Pickle, L. (2002). Evaluation of methods for classifying epidemiological data on choropleth maps in series. *The Annals of the Association of American Geographers*, 92(4), 662-681.
- Broman, K. W. y Sen, S. (2009). *A guide to QTL mapping with R/qtl*. Springer.
- Broström, G. (2012). *Event history analysis with R*. CRC Press.
- Burns, P. (2012). *The R inferno*. Lulu.com.
- Cano, E. L., Martínez Moguerza, J. y Redchuk, A. (2012). *Six sigma with R: statistical engineering for process improvement* (Vol. 36). Springer Science & Business Media.
- Chang, W. (2013). *R graphics cookbook*. O'Reilly.
- Chatterjee, S. y Hadi, A. (2012). *Regression analysis by example* (5a Ed.). Wiley.
- Chatterjee, S. y Simonoff, J. S. (2013). *Handbook of regression analysis*. Wiley.
- Chernick, M. y LaBudde, R. (2011). *An introduction to Bootstrap Methods with applications to R*. Wiley.
- Chiu, Y.-W. (2015). *Machine learning with R cookbook*. Packt.
- Cichosz, P. (2015). *Data mining algorithms: explained using R*. Wiley.
- Claude, J. (2008). *Morphometrics with R*. Springer.
- Cotton, R. (2013). *Learning R*. O'Reilly.
- Crawley, M. J. (2012). *The R book*. John Wiley & Sons.
- Crawley, M. J. (2015). *Statistics: an introduction using R*. Wiley.
- Crowder, M. (2016). *Multivariate survival analysis and competing risks*. Chapman & Hall/CRC.
- Danneman, N. y Heimann, R. (2014). *Social media mining with R: deploy cutting-edge sentiment analysis techniques to real-world social media data using R*. Packt.
- Daróczi, G. (2015). *Mastering data analysis with R*. Packt.
- De Vries, A. y Meys, J. (2015). *R for dummies*. John Wiley & Sons.
- Demidenko, E. (2013). *Mixed models: theory and applications with R* (2a Ed.). Wiley.
- Dennis, B. (2013). *The R student companion*. Chapman & Hall/CRC.
- Dorman, M. (2014). *Learning R for geospatial analysis*. Packt.

- Frery, A. y Perciano, T. (2013). *Introduction to image processing using R: learning by examples*. Springer.
- Gandrud, O. (2013). *Reproducible research with R and R studio*. Chapman & Hall/CRC.
- Gardener, M. (2012). *Beginning R: the statistical programming language*. Wrox.
- Gardener, M. (2013). *The essential R reference*. Wiley.
- Gastner, M. y Newman, M. (2004). Diffusion-based method for producing density-equalizing maps. *Proceedings of the National Academy of Sciences of the United States of America*, 101(20), 7499-7504.
- Georgakopoulos, H. (2015). *Quantitative trading with R: understanding mathematical and computational tools from a quant's perspective*. Palgrave Macmillan.
- Gohil, A. (2015). *R data visualization cookbook: over 80 recipes to analyze data and create stunning visualizations with R*. Packt.
- Grolemund, G. (2014). *Hands-on programming with R: write your own functions and simulations*. O'Reilly Media.
- Horton, N. y Kleinman, K. (2015). *Using R and RStudio for data management, statistical analysis and graphics* (2a Ed.). Chapman & Hall/CRC.
- James, G., Witten, D., Hastie, T., Tibshirani, R. y Taylor, J. (2013). *An introduction to statistical learning with applications in R*. Springer.
- Kabacoff, R. (2015). *R in action: data analysis and graphics with R* (2a Ed.). Manning.
- Krajewski, G. y Matthews, D. (2008). *RH Baayen, analyzing linguistic data: a practical introduction to statistics using R*. Cambridge University Press.
- Loo, M. van der y Jonge, E. (2012). *Learning RStudio for R Statistical Computing*. Packt.
- Lovelace, R., Nowosad, J. y Muenchow, J. (2019). *Geocomputation with R*. CRC Press.
- Mailund, T. (2017). *Beginning data science in R*. Apress.
- Micheaux, P., Drouilhet, R. y Lique, B. *The R software: fundamentals of programming and statistical analysis*. Springer.
- Munzert, S., Rubba, C., Meißner, P. y Nyhuis, D. (2015). *Automated data collection with R: a practical guide to web scraping and text mining*. Wiley.
- Nolan, D. y Lang, D. (2015). *Data Science in R: a case studies approach to computational reasoning and problem solving*. Chapman & Hall/CRC.

- Nord, E. (2019). *Essential R: What you need to get started*. R Foundation for Statistical Computing.
- Nusrat, S. y Kobourov, S. (2016). The state of the art in cartograms. *Computer Graphics Forum*, 35(3), 619-642.
- Pathak, M. (2014). *Beginning data science with R*. Springer.
- Pruscha, H. (2013). *Statistical analysis of climate series: analyzing, plotting, modeling, and predicting with R*. Springer.
- Sánchez, G. (2013). *Handling and processing strings in R*. R Foundation for Statistical Computing.
- Santos-Fernández, E. (2012). *Multivariate statistical quality control using R*. Springer.
- Slocum, T., McMaster, R., Kessler, F. y Hugh, H. (2009). *Thematic cartography and geovisualization* (3a Ed.). Pearson Prentice Hall.
- Snow, J. (1849). *On the mode of communication of cholera*. John Churchill-Princes Street-Soho.
- Stanton, J. (2013). *Introduction to data science*. Sage.
- Takezawa, K. (2013). *Guidebook to R Graphics Using Microsoft Windows*. Wiley.
- Takezawa, K. (2014). *Learning regression analysis by simulation*. Springer.
- Teetor, P. (2011). *R Cookbook*. O'Reilly.
- Tennekes, M. (2018). Tmap: thematic maps in R. *Journal of Statistical Software*, 84(6), 1-39.
- Thiem, A. y Duşa, A. (2013). *Qualitative comparative analysis with R: a user S guide*. Springer.
- Tollefson, M. (2014). *R quick syntax reference*. Apress.
- Toomey, D. (2014). *R for data science: learn and explore the fundamentals of data science with R*. Packt.
- Tyner, J. (2010). *Principles of map design*. Guilford Press.
- Urbano, F. y Cagnacci, F. (2014). *Spatial database for GPS wildlife tracking data: a practical guide to creating a data management system with PostgreSQL/PostGIS and R*. Springer.
- Van Den Boogaart, K. G. y Tolosana-Delgado, R. (2013). *Analyzing compositional data with R* (Vol. 122). Springer.
- Verzani, J. (2011). *Getting started with RStudio*. O'Reilly Media.
- Verzani, J. (2014). *Using R for introductory statistics* (2a Ed.). Chapman & Hall/CRC.

- Voss, J. (2013). *An introduction to statistical computing: a simulation-based approach*. Wiley.
- Wade, R. (2020). *Advanced analytics in Power BI with R and Python*. Apress.
- Walker, K. E. (2016). Tools for interactive visualization of global demographic concepts in *R. Spatial Demography*, 4(3), 207-220.
- Wand, M. (2014). *KernSmooth: Functions for Kernel Smoothing Supporting Wand & Jones (1995)*. R package version 2.23-15. H. *Advanced R*. Chapman & Hall/CRC.
- Wickham, H. (2016). *Ggplot2. Elegant graphics for data analysis*. Springer.
- Xie, Y. (2015). *Dynamic documents with R and knitr* (2a Ed.). Chapman & Hall/CRC.
- Xie, Y. (2016). *Bookdown: authoring books and technical documents with R markdown*. CRC Press.
- Xie, Y. (2017). *Dynamic Documents with R and knitr*. Chapman & Hall/CRC.
- Xie, Y., Allaire, J. J. y Golemund, G. (2018). *R Markdown: the definitive guide*. Hall/CRC.
- Yu-Wei, C. (2015). *Machine learning with R cookbook*. Packt.
- Zacks, S. (2019). *Stage-wise adaptive designs*. Wiley.
- Zhao, Y. y Cen, Y. (2013). *Data mining applications with R*. Academic Press.
- Zieffer, A., Haring, J. y Long, J. (2011). *Comparing groups: randomization and bootstrap methods using R*. Wiley.
- Zumel, N. y Mount, J. (2014). *Practical data science with R*. Manning.

REFERENCIAS

- Abedin, J. y Mittal, H. V. (2014). *R Graphs Cookbook* (2ª Ed.). Packt.
- Adler, J. (2010). *R in a nutshell. A desktop quick reference*. O'Reilly.
- Agresti, A. (2002). *Categorical data analysis* (Vol. 482). John Wiley & Sons.
- Agresti, A. (2015). *Foundations of linear and generalized linear models*. Wiley.
- Aho, K. (2014). *Foundational and applied statistics for biologists using R. Electronic Appendix (Introduction to R)*. Taylor & Francis.
- Anderson, D., Sweeney, D. y William, T. (2008). *Estadística para administración y economía*. Cengage Learning.
- Arnold, T. y Tilton, L. (2015). *Humanities data in R. Exploring networks, geospatial data images, and text*. Springer.
- Barford, A. y Dorling, D. (2008). Telling an old story with new maps. En M. Dodge, M. McDerby y M. Turner (Eds.), *Geographic visualization: concepts, tools and applications* (pp. 67-107). John Wiley & Sons.
- Barker, T. (2013). *Pro data visualization using R and JavaScript*. Apress.
- Beaujean, A. (2014). *Latent variable modeling using R. A step-by-step guide*. Routledge.
- Bilder, C. y Loughin, T. (2015). *Analysis of categorical data with R*. Taylor & Francis.
- Bivand, R., Pebesma, E. y Gómez-Rubio, V. (2013). *Applied spatial data analysis with R*. Springer.
- Bourdieu, P. (2004). *El baile de los solteros*. Anagrama.
- Brewer, C. y Pickle, L. (2002). Evaluation of methods for classifying epidemiological data on choropleth maps in series. *The Annals of the Association of American Geographers*, 92(4), 662-681.
- Burns, P. (2012). *The R inferno*. Lulu.com.

- Cano, E. L., Martínez Moguerza, J. y Redchuk, A. (2012). *Six sigma with R: statistical engineering for process improvement* (Vol. 36). Springer Science & Business Media.
- Chang, W. (2013). *R Graphics Cookbook*. O'Reilly.
- Comisión Nacional de los Derechos Humanos. (2012). *Informe especial sobre el derecho a la movilidad en el Distrito Federal, 2011-2012*. CNDH.
- Consejo Nacional de Población. (2000). *Índices de desarrollo humano, 2000*. CONAPO.
- Cotton, R. (2013). *Learning R*. O'Reilly.
- Crawley, M. J. (2012). *The R book*. John Wiley & Sons.
- Crawley, M. J. (2015). *Statistics: an introduction using R*. Wiley.
- De Vries, A. y Meys, J. (2015). *R for dummies*. John Wiley & Sons.
- Díaz-Guerrero, R. (2014). *Psicología del mexicano. Descubrimiento de la etnopsicología*. Trillas.
- Fierro, C. (2024). ¿Lo sabía? Esto es lo que significa el código oculto de la película 'Matrix'. *PubliMetro*, 12 de marzo. <https://www.publimetro.co/entretenimiento/2024/03/12/lo-sabia-esto-es-lo-que-significa-el-codigo-oculto-de-la-pelicula-matrix/>
- Fromm, E. y Maccoby, M. (2007). *Socio-psicoanálisis del campesino mexicano*. Fondo de Cultura Económica.
- Gaetan, C. y Guyon, X. (2010). *Spatial statistics and modeling*. Springer.
- Gallardo, K.E., Gil, M.E., Contreras, B., García, E., Lázaro, R.A., Ocaña, L. (2012). Toma de decisiones para la evaluación formativa: el proceso de planeación y determinación de sus mecanismos. *Sinéctica*, 39. https://www.scielo.org.mx/scielo.php?pid=S1665-109X2012000200011&script=sci_abstract
- Gastner, M. y Newman, M. (2004). Diffusion-based method for producing density-equalizing maps. *Proceedings of the National Academy of Sciences of the United States of America*, 101(20), 7499-7504.
- Grolemund, G. (2014). *Hands-on programming with R: write your own functions and simulations*. O'Reilly Media.
- Hardt, M. y Negri, A. (2014). *Imperio*. Paidós.
- Horton, N. J. y Kleinman, K. (2015). *Using R and RStudio for data management, statistical analysis, and graphics*. CRC Press.
- Instituto Federal Electoral. (2005). *Cartografía digital a escala distrital 1996*. IFE.
- Instituto Federal Electoral. (2010). *Concentrado General Nacional de los Distritos*. IFE.

- Instituto Federal Electoral. (2015). *Tipología de los distritos electorales*. IFE.
- Instituto Nacional de Estadística y Geografía. (2010a). *Censo de Población y Vivienda 2010*. INEGI.
- Instituto Nacional de Estadística y Geografía. (2010b). *Estadísticas de Hidalgo*. INEGI.
- Instituto Nacional de Estadística y Geografía. (2015). *Índice de Rezago Social*. INEGI.
- Instituto Nacional de Estadística y Geografía. (2017). *Encuesta Origen-Destino en Hogares de la Zona Metropolitana del Valle de México (EOD 2017)*. INEGI.
- Instituto Nacional de Estadística y Geografía. (2021). *Censo de Población y Vivienda 2020*. INEGI.
- James, G., Witten, D., Hastie, T., Tibshirani, R. y Taylor, J. (2013). *An introduction to statistical learning with applications in R*. Springer.
- Kabacoff, R. I. (2015). *R in action: data analysis and graphics with R*. Simon and Schuster.
- Lacoste, Y. (1977). *La geografía: un arma para la guerra*. Anagrama.
- Lovelace, R., Nowosad, J. y Muenchow, J. (2019). *Geocomputation with R*. CRC Press.
- MacEachren, A. (1994). *Some truth with maps: a primer on symbolization and design*. Association of American Geographers.
- Mailund, T. (2017). *Beginning data science in R*. Apress.
- Marx, K. (1999). *El Capital. Crítica de la economía política*. Fondo de Cultura Económica.
- Munzert, S., Rubba, C., Meißner, P. y Nyhuis, D. (2015). *Automated data collection with R: a practical guide to web scraping and text mining*. Wiley.
- Nord, E. (2019). *Essential R: What you need to get started*. R Foundation for Statistical Computing.
- Nusrat, S. y Kobourov, S. (2016). The state of the art in cartograms. *Computer Graphics Forum*, 35(3), 619-642.
- Observatorio Ciudadano de la Ciudad de México. (2009). *Evaluación Ciudadana de Servicios Públicos, 2009*. Universidad Autónoma Metropolitana, Unidad Cuajimalpa.
- Pathak, M. (2014). *Beginning Data Science with R*. Springer.
- Preston, S. H., Heuveline, P. y Guillot, M. (2000). *Demography measuring Modeling Population Processes*. Blackwell.

- Slocum, T., Macmaster, R., Kessler, F. y Hugh, H. (2009). *Thematic cartography and geovisualization* (3a Ed.). Pearson Prentice Hall.
- Snow, J. (1849). *On the mode of communication of cholera*. John Churchill-Princes Street-Soho.
- Teetor, P. (2011). *R Cookbook*. O'Reilly.
- Tennekes, M. (2018). Tmap: thematic maps in R. *Journal of Statistical Software*, 84(6), 1-39.
- Tyner, J. (2010). *Principles of map design*. Guilford Press.
- Walker, K. E. (2016). Tools for interactive visualization of global demographic concepts in R. *Spatial Demography*, 4(3), 207-220.
- Wand, M. (1995). *KernSmooth: Functions for Kernel Smoothing Supporting Wand & Jones*. R package version 2.23-15.2015.
- Wickham, H. (2016). *Ggplot2. Elegant graphics for data analysis*. Springer.
- Xie, Y. (2016). *Bookdown: authoring books and technical documents with R markdown*. CRC Press.
- Xie, Y. (2017). *Dynamic documents with R and knitr*. Chapman & Hall/CRC.
- Xie, Y., Allaire, J. J. y Golemund, G. (2018). *R markdown: the definitive guide*. CRC Press.

ANEXOS

1.1. Banco de datos de hospitales en México

| HOSPITAL | CALLE | COLONIA |
|---|--------------------------------|-----------------------|
| CENTRO DE URGENCIAS # 7 CRUZ ROJA | PERIFERICO SUR Y HDA VISTA HER | U.H. HUESO PERIFERICO |
| CENTRO GINECO OBSTETRICO SANTA MONICA S. A. | TEMISTOCLES NO 210 | POLANCO REFORMA |
| CENTRO MEDICO DALINDE | TUXPAN NO 25 | ROMA SUR |
| CENTRO MEDICO NACIONAL 20 DE NOVIEMBRE ISSSTE | AV COYOACAN NO 1344 Y FELIX C | DEL VALLE |
| CENTRO MEDICO NACIONAL SIGLO XXI | AV CUAUHTEMOC | DOCTORES |
| CENTRO MEDICO QUIRURGICO DR. FCO. MONTES DE OCA | AV CHAPULTEPEC NO 5 | CUAUHT_MOC |
| CIRUGIA DEL VALLE | AMORES NO 942 | DEL VALLE |
| CLINA # 19 IMSS | PDTE VENUSTIANO CARRANZA NO100 | VILLA COYOACAN |
| CLINACA # 23 | SAN JUAN DE ARAGON NO 285 | SAN PEDRO EL CHICO |

| HOSPITAL | CALLE | COLONIA |
|-----------------------|-----------------------------------|--------------------------|
| CLINICA # 1 IMSS | ORIZABA # 15 ESQ AV
CHAPULTEPE | ROMA |
| CLINICA # 11 DEL IMSS | CARUSO Y
LEONCAVALLO | VALLEJO |
| CLINICA # 12 IMSS | PLAZA DE LOS HEROES | U.H. SANTA FE |
| CLINICA # 14 IMSS | BLVR PUERTO AEREO
NO 88 | MOCTEZUMA |
| CLINICA # 15 IMSS | CALZ ERMITA
IZTAPALAPA NO 411 | PRADO CHURUBUSCO |
| CLINICA # 16 IMSS | FCO GONZALEZ
BOCANEGRA NO 10 | PERALVILLO |
| CLINICA # 160 IMSS | CALZ TLAHUAC ESQ
AGUJAS | VERGEL |
| CLINICA # 17 IMSS | CALZ LEGARIA NO 354 | PENSIL |
| CLINICA # 18 IMSS | AV MEXICO NO 98 | SANTA TERESA |
| CLINICA # 2 IMSS | SORJUANA INES DE LA
CRUZ NO 61 | SANTA MARIA LA
RIBERA |
| CLINICA # 21 IMSS | FCO DEL PASO Y
TRONCOSO NO 91 | JARDIN BALBUENA |
| CLINICA # 22 IMSS | RIO CHICO ESQ SAN
RAMON | SAN JOSE EL BATAN |
| CLINICA # 3 IMSS | NORTE NO 70 ESQ
ORIENTE NO 91 | LA JOYA |
| CLINICA # 31 IMSS | CALZ ERMITA
IZTAPALAPA NO 1771 | EL MANTO |
| CLINICA # 34 IMSS | AV MEXICO ESQ CALLE 2 | PANTITLAN |
| CLINICA # 35 IMSS | CALLE 7 NO 200 | JUAN ESCUTIA |
| CLINICA # 36 IMSS | CDA 5 DE MAYO ESQ 5
DE MAYO | ATZACOALCO |
| CLINICA # 37 IMSS | VALERIO TRUJANO NO
17 | GUERRERO |
| CLINICA # 38 IMSS | PARQUE ESPAÑA NO 9 | CONDESA |
| CLINICA # 4 IMSS | AV NIÑOS HEROES NO
165 | DOCTORES |
| CLINICA # 40 IMSS | AV HIDALGO NO 24 | SANTA BARBARA |

| HOSPITAL | CALLE | COLONIA |
|--|-----------------------------------|-----------------------------|
| CLINICA # 42 IMSS | AV JESUS DEL MONTE
ESQ MONTE A | CUAJIMALPA |
| CLINICA # 43 IMSS | CALZ ERMITA
IZTAPALAPA NO 1515 | HIDALGO Y MINA |
| CLINICA # 44 IMSS | PLAN DE SAN LUIS ESQ
LA SALLE | LA ESCALERA |
| CLINICA # 45 IMSS | CANAL DE TEZONTLE
ESQ SUR 159 | AMP GABRIEL RAMOS
MILLAN |
| CLINICA # 46 IMSS | CALZ DE TLALPAN NO
2655 | XOTEPINGO |
| CLINICA # 5 IMSS | AV MARINA NAL ESQ M
ESCOBEDO | ANAHUAC |
| CLINICA # 6 IMSS | CIRCUNVALACION ESQ
CORREGIDORA | |
| CLINICA # 7 IMSS | CALZ DE TLALPAN NO
4220 | TLALPAN |
| CLINICA # 9 IMSS | AV REVOLUCION ESQ
CALLE 7 | SAN PEDRO DE LOS
PINOS |
| CLINICA # 94 IMSS | CAM A SAN JUAN DE
ARAGON 235 | SAN JUAN DE ARAGON |
| CLINICA 5 DE
FEBRERO ISSSTE | 5 DE FEBRERO NO 111 | CENTRO |
| CLINICA
AZCAPOTZALCO
ISSSTE | SAN MATEO NO 31 | VILLA
AZCAPOTZALCO |
| CLINICA BALBUENA
ISSSTE | FDO IGLESIAS ESQ AV
DEL TALLER | JARDIN BALBUENA |
| CLINICA
CHAPULTEPEC ISSSTE | AV JOSE VASCONCELOS
NO 10 | CONDESA |
| CLINICA COYOACAN
ISSSTE | FERNANDEZ LEAL NO 11 | EL CARMEN |
| CLINICA CUITLAHUAC
ISSSTE | AV CUITLAHUAC NO 548 | NUEVA SANTA MARIA |
| CLINICA DE
DETECCION Y
DIAGNOSTICO
AUTOMATIZADO | AV UNIVERSIDAD NO
1321 | FLORIDA |

| HOSPITAL | CALLE | COLONIA |
|---|-------------------------------|--------------------------------|
| CLINICA DE ESPECIALIDADES ALBERTO PIZANTY | DOCTORA NO 28 | TACUBAYA |
| CLINICA DE ESPECIALIDADES CHURUBUSCO ISSSTE | CALZ DE TLALPAN NO 1983 | PARQUE SAN ANDRES |
| CLINICA DEL VALLE ISSSTE | PATRICIO SANZ NO 51 ESQ XOLA | DEL VALLE |
| CLINICA DIVISION DEL NORTE ISSSTE | AV DIVISION DEL NORTE NO 3755 | SAN PABLO TEPETLAPA |
| CLINICA ERMITA ISSSTE | CALZ ERMITA IZTAPALAPA NO 67 | PRADO |
| CLINICA GUADALUPE ISSSTE | CALZ DE GUADALUPE NO 162 | EX HIPODROMO DE PERALVILLO |
| CLINICA GUERRERO ISSSTE | AV GUERRERO NO 65 | GUERRERO |
| CLINICA GUSTAVO A MADERO ISSSTE | CALZ DE GUADALUPE NO 712 | GUSTAVO A MADERO |
| CLINICA IGNACIO CHAVEZ ISSSTE | ORIENTAL NO 10 | ALIANZA POPULAR REVOLUCIONARIA |
| CLINICA INDIANILLA ISSSTE | DR CLAUDIO BERNARD NO 85 | DOCTORES |
| CLINICA IZTAPALAPA ISSSTE | CALZ ERMITA IZTAPALAPA NO 768 | GRANJAS SAN ANTONIO |
| CLINICA JUAREZ | JALAPA NO 252 | ROMA |
| CLINICA LEGARIA ISSSTE | CALZ LEGARIA NO 361 | DEPORTIVA PENSIL |
| CLINICA LONDRES | DURANGO NO 50 | ROMA |
| CLINICA MARINA NACIONAL ISSSTE | MARINA NACIONAL NO 261 | ANAHUAC |
| CLINICA MILPA ALTA ISSSTE | NIDOS HEROES ESQ ORIZABA | BARRIO SAN AGUSTIN |
| CLINICA MOCTEZUMA ISSSTE | ORIENTE 144 NO 206 ESQ NTE 21 | MOCTEZUMA |
| CLINICA NARVARTE ISSSTE | AV EUGENIA ESQ LA QUEMADA | NARVARTE |

| HOSPITAL | CALLE | COLONIA |
|-------------------------------------|--------------------------------|--------------------------------|
| CLINICA NEZAHUALCOYOTL ISSSTE | CALZ IGNACIO ZARAGOZA NO 1189 | AGRICOLA ORIENTAL |
| CLINICA OBSERVATORIO ISSSTE | AV OBSERVATORIO NO 222 | DANIEL GARZA |
| CLINICA ORIENTE ISSSTE | AV TELECOMUNICACIONES | UH EJERCITO CONSTITUCIONALISTA |
| CLINICA PERALVILLO ISSSTE | CONSTANCIA NO 18 Y PZA DE LA C | MORELOS |
| CLINICA PERU ISSSTE | REPUBLICA DE PERU NO 25 | CENTRO |
| CLINICA REVOLUCION ISSSTE | AV REVOLUCION NO 537 | SAN PEDRO DE LOS PINOS |
| CLINICA SAN ANTONI ABAD ISSSTE | MANUEL JOSE OTHON NO 169 | TRANSITO |
| CLINICA SAN RAFAEL | INSURGENTES SUR NO 4177 | SANTA URSULA XITLA |
| CLINICA SAN RAFAEL ISSSTE | IGNACIO ALTAMIRANO NO 115 | SAN RAFAEL |
| CLINICA SANTA MARIA ISSSTE | JOSE ANTONIO ALZATE NO 168 | SANTA MARIA LA RIBERA |
| CLINICA TLALPAN ISSSTE | GENERAL GUADALUPE VICTORIA NO2 | TLALPAN |
| CLINICA VILLA ALVARO OBREGON ISSSTE | AV REVOLUCION NO 1556 | SAN ANGEL |
| CLINICA VOCEADORES | NARANJO NO 121 | SANTA MARIA LA RIBERA |
| CLINICA XOCHIMILCO ISSSTE | GLADIOLAS NO 158 | BARRIO SAN PEDRO |
| HOSPITAL ANGELES DEL PEDREGAL | CAMINO A SANTA TERESA NO 1055 | H_ERODES DE PADIERNA |
| HOSPITAL CAMI | CALZ IGNACIO ZARAGOZA NO 189 | JARD_N BALBUENA |
| HOSPITAL CENTRAL CRUZ ROJA | EJERCITO NAL Y B PEREZ GALDOZ | POLANCO REFORMA |

| HOSPITAL | CALLE | COLONIA |
|---|-------------------------------|---------------------------|
| HOSPITAL CENTRAL DE SERVICIOS | FORTUNA | MAGDALENA DE LAS SALINAS |
| HOSPITAL CLINICA GUADALUPE TEPEYAC | AV TICOMAN NO 128 | LINDAVISTA |
| HOSPITAL DE GINECO OBSTETRICIA # 13 | REFORMA NO Y ESQ HIDALGO | VILLA AZCAPOTZALCO |
| HOSPITAL DE JESUS | 20 DE NOVIEMBRE NO 82 | CENTRO |
| HOSPITAL DE LA MUJER | PROL DIAZ MIRON NO 374 | SANTO TOMAS |
| HOSPITAL DE MEXICO | AGRARISMO NO 208 | ESCAND_N |
| HOSPITAL DE URGENCIAS TRAUMATOLOGICAS DEL SUR | TORRES ADALID NO 1305 | DEL VALLE |
| HOSPITAL DEL DDF | CAM VIEJO A CUERNAVACA | SAN MIGUEL TOPILEJO |
| HOSPITAL ESPAÑOL | EJERCITO NACIONAL NO 613 | GRANADA |
| HOSPITAL GENERAL | AV CUAUHTEMOC | DOCTORES |
| HOSPITAL GENERAL BALBUENA | CECILIO ROBELO Y ESQ SUR 103 | JARDIN BALBUENA |
| HOSPITAL GENERAL DE MEXICO | DR BALMIS NO 148 | DOCTORES |
| HOSPITAL GENERAL DE TICOMAN SS | PLAN DE SAN LUIS ESQ BANDERA | TICOMAN |
| HOSPITAL GENERAL DE ZONA # 1 | GABRIEL MANCERA NO 222 | DEL VALLE |
| HOSPITAL GENERAL DE ZONA # 1A | MUNICIPIO LIBRE NO 270 | PORTALES |
| HOSPITAL GENERAL DE ZONA # 24 | INSURGENTES NORTE NO 1322 | MAGDALENA DE LAS SALINAS |
| HOSPITAL GENERAL DE ZONA # 26 | CHILPANCINGO NO 56 | HIPODROMO |
| HOSPITAL GENERAL DE ZONA # 27 | EJE CENTRAL L CARDENAS NO 445 | U. H. NONOALCO TLATELOLCO |

| HOSPITAL | CALLE | COLONIA |
|---|--------------------------------|---------------------------|
| HOSPITAL GENERAL DE ZONA # 32 | CALZ DEL HUESO | EX HACIENDA COAPA |
| HOSPITAL GENERAL DE ZONA # 47 | CAMPAÑAS DEL EBANO | UNIDAD VICENTE GUERRERO |
| HOSPITAL GENERAL DE ZONA # 8 | RIO MAGDALENA NO 289 | TIZAPAN SAN ANGEL |
| HOSPITAL GENERAL DE ZONA #30 | AV. PLUTARCO ELIAS CALLES 473 | SANTA ANITA |
| HOSPITAL GENERAL DE ZONA 2A | AÑIL NO 144 | GRANJAS MEXICO |
| HOSPITAL GENERAL DR GONZALO CASTAÑEDA | PROL LERDO ESQ MANUEL GONZALEZ | U. H. NONOALCO TLATELOLCO |
| HOSPITAL GENERAL DR MANUEL GEA GONZALEZ | CALZ TLALPAN NO 4800 | TLALPAN |
| HOSPITAL GENERAL FERNANDO QUIROZ | FELIPE ANGELES ESQ CANARIO | BELLAVISTA |
| HOSPITAL GENERAL GREGORIO SALAS | DEL CARMEN NO 42 | CENTRO |
| HOSPITAL GENERAL IZTAPALAPA | CALZ ERMITA IZTAPALAPA NO 3318 | SIERRA DEL VALLE |
| HOSPITAL GENERAL JOSE MARIA MORELOS Y PAVON | CHILPANCINGO SUR ESQ CHIL NTE | UNIDAD ERMITA ZARAGOZA |
| HOSPITAL GENERAL LA RAZA | CIR. INTERIOR ESQ CLAZ VALLEJO | LA RAZA |
| HOSPITAL GENERAL MILPA ALTA | AV JALISCO ESQ YUCATAN | VILLA MILPA ALTA |
| HOSPITAL GENERAL REGIONAL # 25 | IGNACIO ZARAGOZA NO 1840 | JUAN ESCUTIA |
| HOSPITAL GENERAL TACUBA | LAGO ONTARIO NO 119 | LEGARIA |
| HOSPITAL GENERAL XOCO | AV. MEXICO COYOACAN | XOCO |
| HOSPITAL GINECO PEDIATRIA # 3A | INST POLITECNICO ESQ FORTUNA | MAGDALENA DE LAS SALINAS |

| HOSPITAL | CALLE | COLONIA |
|--|--------------------------------|------------------------|
| HOSPITAL INFANTIL DE MEXICO FEDERICO GOMEZ | DR MARQUEZ NO 162 | DOCTORES |
| HOSPITAL INFANTIL PRIVADO | VIADUCTO RIO BECERRA NO 97 | NÁPOLES |
| HOSPITAL JUAREZ DEL CENTRO SS | SAN PABLO NO 13 | CENTRO |
| HOSPITAL MATERNO INFANTIL | ESTADO ESQ CIEGUILLAS | FELIPE ANGELES |
| HOSPITAL MATERNO INFANTIL CUAUTEPEC | EMILIANO ZAPATA NO 17 | CUAUTEOEC BAJO |
| HOSPITAL MATERNO INFANTIL M. C. | AV LUIS CABRERA | LOMAS QUEBRADAS |
| HOSPITAL MATERNO INFANTIL TLAHUAC | HIDALGO ESQ CUAUHTEMOC | BARRIO SAN MIGUEL |
| HOSPITAL MATERNO INFANTIL TOPILEJO | SANTA CRUZ NO 1 | SAN MIGUEL TOPILEJO |
| HOSPITAL METROPOLITANO | TLACOTALPAN NO. 51 | ROMA |
| HOSPITAL MILITAR | EJERCITO NACIONAL Y PERIFERICO | RESIDENCIAL MILITAR |
| HOSPITAL MOCEL | GELATI NO 29 | SAN MIGUEL CHAPULTEPEC |
| HOSPITAL NACIONAL DE NEUROLOGIA Y NEUROCIRUGIA | INSURGENTE SUR NO 3877 | FUENTES BROTTANTES |
| HOSPITAL OBREGON MEDICO QUIRURGICO | ALVARO OBREGON NO 23 | ROMA |
| HOSPITAL ORTOPEDICO | OTHON DE MENDIZABAL OTE | TORRES DE LINDAVISTA |
| HOSPITAL PEDIATRICO AZCAPOTZALCO | AZCAPOTZALCO NO 731 | VILLA AZCAPOTZALCO |
| HOSPITAL PEDIATRICO COYOACAN | M. ANGEL DE QUEVEDO ESQ AURORA | VILLA COYOACAN |

| HOSPITAL | CALLE | COLONIA |
|--|--------------------------------|-----------------------------|
| HOSPITAL PEDIATRICO IZTACALCO | AV COYUYA ESQ CORREGIDORA | SANTA ANITA |
| HOSPITAL PEDIATRICO IZTAPALAPA | CALZ ERMITA IZTAPALAPA NO 780 | GRANJAS SAN ANTONIO |
| HOSPITAL PEDIATRICO LEGARIA | CALZ LEGARIA NO 347 | PENSIL NORTE |
| HOSPITAL PEDIATRICO MOCTEZUMA | AV ORIENTE 158 NO 59 | MOCTEZUMA 2D SECCION |
| HOSPITAL PEDIATRICO PERALVILLO | TOLNAHUAC ESQ PROL ZARCO NO 14 | SAN SIMON TOLNAHUAC |
| HOSPITAL PEDIATRICO SAN JUAN DE ARAGON | AV 506 Y 521 | U. H. SAN JUAN DE ARAGON |
| HOSPITAL PEDIATRICO TACUBAYA | ARQ LAZO Y GAVIOTA | OBSERVATORIO |
| HOSPITAL PEDIATRICO VILLA | AV HIDALGO ESQ AV CANTERA | GUSTAVO A MADERO |
| HOSPITAL PEDIATRICO XOCHIMILCO | CAM A NATIVITAS ESQ 16 DE SEP | BARRIO XALTOCAN |
| HOSPITAL PSIQUIATRICO | AV SAN FERNANDO NO 201 | BARRIO NIDO JESUS |
| HOSPITAL PSIQUIATRICO | CALZ SAN JUAN DE ARAGON NO 311 | SAN PEDRO EL CHICO |
| HOSPITAL PSIQUIATRICO INFANTIL | VIADUCTO TLALPAN | SECCION XVI |
| HOSPITAL PSIQUIATRICO UMF 10 IMSS | CALZ. DE TLALPAN NO. 931 | NIDOS HEROES DE CHAPULTEPEC |
| HOSPITAL REGIONAL 1R DE OCTUBRE | INSTITUTO POLITECNICO NO 1690 | MAGDALENA DE LAS SALINAS |

| HOSPITAL | CALLE | COLONIA |
|---|----------------------------------|-------------------------------|
| HOSPITAL REGIONAL
DR. DARIO
FERNANDEZ FIERRO | BARRANCA DEL
MUERTO NO. 1182 | SAN JOSE
INSURGENTES |
| HOSPITAL REGIONAL
IGNACIO ZARAGOZA | CALZ IGNACIO
ZARAGOZA NO 1711 | REFORMA EDUCATIVA |
| HOSPITAL REGIONAL
LIC ADOLFO LOPEZ
MATEOS | AV UNIVERSIDAD NO
1321 | FLORIDA |
| HOSPITAL RUBEN
LEDERO | PLAN DE SAN LUIS | SATO TOMAS |
| HOSPITAL SAN ANGEL
INN | AV MEXICO NO. 2 | PROGRESO TIZAPÁN
SAN ANGEL |
| HOSPITAL SANTA FE | SAN LUIS POTOSI NO 143 | ROMA |
| HOSPITAL SANTA
TERESA | ACUEDUCTO RIO
HONDO NO 20 | LOMAS VIRREYES |
| HOSPITAL
SANTELENA | QUERETARO NO 62 | ROMA SUR |
| HOSPITAL SIND DE
TRABAJADORES DE LA
IND AZUCARERA | TEMISTOCLES NO 349 | CHAPULTEPEC
MORALES |
| HOSPITAL VERTIZ | LAGO GARDA NO 76 | ANÁHUAC |
| INS NAL DE DIAG
Y REFERENCIA
EPIDEMIOLOGICOS SS | PROL DE CARPIO NO 470 | SANTO TOMAS |
| INST NACIONAL
DE MEDICINA DE
REHABILITACION SS | AV MARIANO
ESCOBEDO NO 150 | ANAHUAC |
| INST. NAL. DE
CARDIOLOGIA DR.
IGNACIO CHAVEZ | JUAN BADIANO NO 1 | SECCI_N XVI |
| INST. NAL. DE
ENFERMEDADES
RESPIRATORIAS | CALZ TLALPAN NO 4502 | SECCI_N XVI |
| INSTITUTO
MEXICANO
DE CIRUGIA
AMBULATORIA | CARR PICACHO AJUSCO
NO 1 | JARDINES DE LA
MONTANA |

| HOSPITAL | CALLE | COLONIA |
|--|------------------------------|--------------------------|
| INSTITUTO NACIONAL DE CANCEROLOGIA | AV SAN FERNANDO NO 22 | TLALPAN |
| INSTITUTO NACIONAL DE LA COMUNICACION HUMANA | DR FCO DE P MIRANDA | U. H. LOMAS DE PLATEROS |
| INSTITUTO NACIONAL DE LA NUTRICION | VASCO DE QUIROGA NO 15 | TLALPAN |
| INSTITUTO NACIONAL DE PERINATOLOGIA | MONTES URALES NO 800 | LOMAS DE VIRREYES |
| INSTITUTO NACIONAL DE VIROLOGIA SS | PROL DE CARPIO NO 492 | SANTO TOMAS |
| MEDICA LONDRES | LONDRES NO 34 | JUÁREZ |
| MEDICA SUR | PUENTE DE PIEDRA NO 150 | TORIELLO GUERRA |
| NUEVO HOSPITAL JUAREZ | INST POLITECNICO NAL NO 5160 | MAGDALENA DE LAS SALINAS |
| SANATORIO DURANGO | DURANGO NO 290 | ROMA |
| SANATORIO LOURDES | ALVARO OBREGON NO 63 | ROMA |
| SANATORIO MIG | RIO BAMBA NO 800 | LINDAVISTA |
| SANATORIO NOTRE DAME | AV CHAPULTEPEC NO 48 | JUÁREZ |
| SANATORIO OFTALMOLOGICO MERIDA | CHIHUAHUA NO 71 | ROMA |
| SANATORIO TRINIDAD | TLAXCALA 90-BIS | ROMA SUR |
| THE AMERICAN BRITISH COWDRA Y HOSPITAL (ABC) | SUR 136 NO. 188 | LAS AM_RICAS |
| TORRE MEDICA | JOSE MARIA IGLESIA NO 21 | REVOLUCI_N |
| UNIDAD DE MEDICINA FAMILIAR # 1 | ORIZABA NO 15 | ROMA NORTE |

| HOSPITAL | CALLE | COLONIA |
|--|--------------------------------|--------------------------|
| UNIDAD DE MEDICINA FAMILIAR # 20 IMSS | CALZ VALLEJO NO 675 | MAGDALENA DE LAS SALINAS |
| UNIDAD DE MEDICINA FAMILIAR # 28 IMSS | GABRIEL MANCERA NO 919 | DEL VALLE |
| UNIDAD DE MEDICINA FAMILIAR # 31 | CALZ ERMITA IZTAPALAPA NO 1771 | EL MANTO |
| UNIDAD DE MEDICINA FISICA REGION SUR | CALZ DEL HUESO | EX HACIENDA COAPA |
| UNIDAD DE MEDICINA FISICA REGION NORTE | AV POLITECNICO NACIONAL S/N | MAGDALENA DE LAS SALINAS |
| UNIDAD MEDICA FISICA CENTRO | MANUEL VILLALONGIN NO 117 | CUAUHTEMOC |

1.2. Líneas de Metro en la Ciudad de México, 1997

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

| LINEA | NOMBRE | AFLUENCIA | DENSIDAD | CAPTO97 |
|---------|--|-----------|----------|-------------------------------|
| LINEA 6 | EL ROSARIO
- MARTIN
CARRERA | 1400000 | 12484 | Línea 6 - 142600
pasajeros |
| LINEA 6 | EL ROSARIO
- MARTIN
CARRERA | 1400000 | 12484 | Línea 6 - 142600
pasajeros |
| LINEA 6 | EL ROSARIO
- MARTIN
CARRERA | 1400000 | 12484 | Línea 6 - 142600
pasajeros |
| LINEA 6 | EL ROSARIO
- MARTIN
CARRERA | 1400000 | 12484 | Línea 6 - 142600
pasajeros |
| LINEA 6 | EL ROSARIO
- MARTIN
CARRERA | 1400000 | 12484 | Línea 6 - 142600
pasajeros |
| LINEA 6 | EL ROSARIO
- MARTIN
CARRERA | 1400000 | 12484 | Línea 6 - 142600
pasajeros |
| LINEA 6 | EL ROSARIO
- MARTIN
CARRERA | 1400000 | 12484 | Línea 6 - 142600
pasajeros |
| LINEA 8 | GARIBALDI -
CONSTITUCION
DE 1917 | 1400000 | 12484 | Línea 8 - 312600
pasajeros |
| LINEA 8 | GARIBALDI -
CONSTITUCION
DE 1917 | 1400000 | 12484 | Línea 8 - 312600
pasajeros |
| LINEA 8 | GARIBALDI -
CONSTITUCION
DE 1917 | 1400000 | 12484 | Línea 8 - 312600
pasajeros |
| LINEA 8 | GARIBALDI -
CONSTITUCION
DE 1917 | 1400000 | 12484 | Línea 8 - 312600
pasajeros |
| LINEA 8 | GARIBALDI -
CONSTITUCION
DE 1917 | 1400000 | 12484 | Línea 8 - 312600
pasajeros |

[illegible]

[illegible]

| LINEA | NOMBRE | AFLUENCIA | DENSIDAD | CAPTO97 |
|---------|--|-----------|----------|-------------------------------|
| LINEA 8 | GARIBALDI -
CONSTITUCION
DE 1917 | 1400000 | 12484 | Línea 8 - 312600
pasajeros |
| LINEA 8 | GARIBALDI -
CONSTITUCION
DE 1917 | 1400000 | 12484 | Línea 8 - 312600
pasajeros |
| LINEA 8 | GARIBALDI -
CONSTITUCION
DE 1917 | 1400000 | 12484 | Línea 8 - 312600
pasajeros |
| LINEA 8 | GARIBALDI -
CONSTITUCION
DE 1917 | 1400000 | 12484 | Línea 8 - 312600
pasajeros |
| LINEA 8 | GARIBALDI -
CONSTITUCION
DE 1917 | 1400000 | 12484 | Línea 8 - 312600
pasajeros |
| LINEA 8 | GARIBALDI -
CONSTITUCION
DE 1917 | 1400000 | 12484 | Línea 8 - 312600
pasajeros |
| LINEA 8 | GARIBALDI -
CONSTITUCION
DE 1917 | 1400000 | 12484 | Línea 8 - 312600
pasajeros |
| LINEA 8 | GARIBALDI -
CONSTITUCION
DE 1917 | 1400000 | 12484 | Línea 8 - 312600
pasajeros |
| LINEA 8 | GARIBALDI -
CONSTITUCION
DE 1917 | 1400000 | 12484 | Línea 8 - 312600
pasajeros |
| LINEA 8 | GARIBALDI -
CONSTITUCION
DE 1917 | 1400000 | 12484 | Línea 8 - 312600
pasajeros |
| LINEA 8 | GARIBALDI -
CONSTITUCION
DE 1917 | 1400000 | 12484 | Línea 8 - 312600
pasajeros |
| LINEA A | PANTITLAN - LA
PAZ | 1400000 | 12484 | Línea A - 250300
pasajeros |

| LINEA | NOMBRE | AFLUENCIA | DENSIDAD | CAPTO97 |
|---------|--------------------------|-----------|----------|----------------------------|
| LINEA A | PANTITLAN - LA PAZ | 1400000 | 12484 | Línea A - 250300 pasajeros |
| LINEA A | PANTITLAN - LA PAZ | 1400000 | 12484 | Línea A - 250300 pasajeros |
| LINEA A | PANTITLAN - LA PAZ | 1400000 | 12484 | Línea A - 250300 pasajeros |
| LINEA A | PANTITLAN - LA PAZ | 1400000 | 12484 | Línea A - 250300 pasajeros |
| LINEA A | PANTITLAN - LA PAZ | 1400000 | 12484 | Línea A - 250300 pasajeros |
| LINEA A | PANTITLAN - LA PAZ | 1400000 | 12484 | Línea A - 250300 pasajeros |
| LINEA A | PANTITLAN - LA PAZ | 1400000 | 12484 | Línea A - 250300 pasajeros |
| LINEA A | PANTITLAN - LA PAZ | 1400000 | 12484 | Línea A - 250300 pasajeros |
| LINEA A | PANTITLAN - LA PAZ | 1400000 | 12484 | Línea A - 250300 pasajeros |
| LINEA A | PANTITLAN - LA PAZ | 1400000 | 12484 | Línea A - 250300 pasajeros |
| LINEA A | PANTITLAN - LA PAZ | 1400000 | 12484 | Línea A - 250300 pasajeros |
| LINEA A | PANTITLAN - LA PAZ | 1400000 | 12484 | Línea A - 250300 pasajeros |
| LINEA A | PANTITLAN - LA PAZ | 1400000 | 12484 | Línea A - 250300 pasajeros |
| LINEA A | PANTITLAN - LA PAZ | 1400000 | 12484 | Línea A - 250300 pasajeros |
| LINEA A | PANTITLAN - LA PAZ | 1400000 | 12484 | Línea A - 250300 pasajeros |
| LINEA 1 | OBSERVATORIO - PANTITLAN | 2600000 | 40287 | Línea 1 - 905300 pasajeros |
| LINEA 1 | OBSERVATORIO - PANTITLAN | 2600000 | 40287 | Línea 1 - 905300 pasajeros |

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

| LÍNEA | NOMBRE | AFLUENCIA | DENSIDAD | CAPTO97 |
|-------------|-----------------------------|-----------|----------|----------------------------|
| TREN LIGERO | TAXQUEDA - XOCHIMILCO | 1400000 | 12484 | Línea B - SIN DATOS |
| TREN LIGERO | TAXQUEDA - XOCHIMILCO | 1400000 | 12484 | Línea B - SIN DATOS |
| TREN LIGERO | TAXQUEDA - XOCHIMILCO | 1400000 | 12484 | Línea B - SIN DATOS |
| TREN LIGERO | TAXQUEDA - XOCHIMILCO | 1400000 | 12484 | Línea B - SIN DATOS |
| TREN LIGERO | TAXQUEDA - XOCHIMILCO | 1400000 | 12484 | Línea B - SIN DATOS |
| TREN LIGERO | TAXQUEDA - XOCHIMILCO | 1400000 | 12484 | Línea B - SIN DATOS |
| TREN LIGERO | TAXQUEDA - XOCHIMILCO | 1400000 | 12484 | Línea B - SIN DATOS |
| TREN LIGERO | TAXQUEDA - XOCHIMILCO | 1400000 | 12484 | Línea B - SIN DATOS |
| TREN LIGERO | TAXQUEDA - XOCHIMILCO | 1400000 | 12484 | Línea B - SIN DATOS |
| TREN LIGERO | TAXQUEDA - XOCHIMILCO | 1400000 | 12484 | Línea B - SIN DATOS |
| TREN LIGERO | TAXQUEDA - XOCHIMILCO | 1400000 | 12484 | Línea B - SIN DATOS |
| TREN LIGERO | TAXQUEDA - XOCHIMILCO | 1400000 | 12484 | Línea B - SIN DATOS |
| TREN LIGERO | TAXQUEDA - XOCHIMILCO | 1400000 | 12484 | Línea B - SIN DATOS |
| TREN LIGERO | TAXQUEDA - XOCHIMILCO | 1400000 | 12484 | Línea B - SIN DATOS |
| TREN LIGERO | TAXQUEDA - XOCHIMILCO | 1400000 | 12484 | Línea B - SIN DATOS |
| TREN LIGERO | TAXQUEDA - XOCHIMILCO | 1400000 | 12484 | Línea B - SIN DATOS |
| LÍNEA 3 | UNIVERSIDAD - INDIOS VERDES | 2600000 | 40287 | Línea 3 - 875800 pasajeros |

[illegible]

| LÍNEA | NOMBRE | AFLUENCIA | DENSIDAD | CAPTO97 |
|---------|-----------------------------|-----------|----------|----------------------------|
| LÍNEA 3 | UNIVERSIDAD - INDIOS VERDES | 2600000 | 40287 | Línea 3 - 875800 pasajeros |
| LÍNEA 3 | UNIVERSIDAD - INDIOS VERDES | 2600000 | 40287 | Línea 3 - 875800 pasajeros |
| LÍNEA 3 | UNIVERSIDAD - INDIOS VERDES | 2600000 | 40287 | Línea 3 - 875800 pasajeros |
| LÍNEA 3 | UNIVERSIDAD - INDIOS VERDES | 2600000 | 40287 | Línea 3 - 875800 pasajeros |
| LÍNEA 3 | UNIVERSIDAD - INDIOS VERDES | 2600000 | 40287 | Línea 3 - 875800 pasajeros |

1.3. Índice de Desarrollo Humano en la ZMVM, 2000

| CLVEDO | CLVMUN | EDO | MUN | IDH |
|--------|--------|------------------|------------------------|------|
| 09 | 09002 | DISTRITO FEDERAL | AZCAPOTZALCO | 87.3 |
| 09 | 09003 | DISTRITO FEDERAL | COYOACAN | 90.1 |
| 09 | 09004 | DISTRITO FEDERAL | CUAJIMALPA DE MORELOS | 87.4 |
| 09 | 09005 | DISTRITO FEDERAL | GUSTAVO A. MADERO | 86.0 |
| 09 | 09006 | DISTRITO FEDERAL | IZTACALCO | 86.6 |
| 09 | 09007 | DISTRITO FEDERAL | IZTAPALAPA | 84.6 |
| 09 | 09008 | DISTRITO FEDERAL | LA MAGDALENA CONTRERAS | 87.0 |
| 09 | 09009 | DISTRITO FEDERAL | MILPA ALTA | 81.5 |
| 09 | 09010 | DISTRITO FEDERAL | ALVARO OBREGON | 88.1 |
| 09 | 09011 | DISTRITO FEDERAL | TLAHUAC | 84.4 |

| CLVEDO | CLVMUN | EDO | MUN | IDH |
|--------|--------|------------------|-------------------------|------|
| 09 | 09012 | DISTRITO FEDERAL | TLALPAN | 88.2 |
| 09 | 09013 | DISTRITO FEDERAL | XOCHIMILCO | 86.2 |
| 09 | 09014 | DISTRITO FEDERAL | BENITO JUAREZ | 93.0 |
| 09 | 09015 | DISTRITO FEDERAL | CUAUHTEMOC | 98.8 |
| 09 | 09016 | DISTRITO FEDERAL | MIGUEL HIDALGO | 90.4 |
| 09 | 09017 | DISTRITO FEDERAL | VENUSTIANO CARRANZA | 86.6 |
| 13 | 13069 | HIDALGO | TIZAYUCA | 80.6 |
| 15 | 15002 | MEXICO | ACOLMAN | 78.5 |
| 15 | 15011 | MEXICO | ATENCO | 76.8 |
| 15 | 15013 | MEXICO | ATIZAPAN DE ZARAGOZA | 82.2 |
| 15 | 15020 | MEXICO | COACALCO DE BERRIOZABAL | 83.0 |
| 15 | 15022 | MEXICO | COCOTITLAN | 78.5 |
| 15 | 15023 | MEXICO | COYOTEPEC | 76.2 |
| 15 | 15024 | MEXICO | CUAUTITLAN | 81.6 |
| 15 | 15025 | MEXICO | CHALCO | 76.1 |
| 15 | 15028 | MEXICO | CHIAUTLA | 77.7 |
| 15 | 15029 | MEXICO | CHICOLOAPAN | 77.4 |
| 15 | 15030 | MEXICO | CHICONCUAC | 78.0 |
| 15 | 15031 | MEXICO | CHIMALHUACAN | 77.3 |
| 15 | 15033 | MEXICO | ECATEPEC DE MORELOS | 79.2 |
| 15 | 15035 | MEXICO | HUEHUETOCA | 77.6 |
| 15 | 15037 | MEXICO | HUIXQUILUCAN | 83.0 |
| 15 | 15039 | MEXICO | IXTAPALUCA | 78.7 |
| 15 | 15044 | MEXICO | JALTENCO | 79.7 |
| 15 | 15053 | MEXICO | MELCHOR OCAMPO | 77.9 |
| 15 | 15057 | MEXICO | NAUCALPAN DE JUAREZ | 82.0 |

| CLVEDO | CLVMUN | EDO | MUN | IDH |
|--------|--------|--------|--------------------------------|------|
| 15 | 15058 | MEXICO | NEZAHUALCOYOTL | 81.1 |
| 15 | 15059 | MEXICO | NEXTLALPAN | 79.6 |
| 15 | 15060 | MEXICO | NICOLAS ROMERO | 77.7 |
| 15 | 15070 | MEXICO | LA PAZ | 79.9 |
| 15 | 15075 | MEXICO | SAN MARTIN DE LAS
PIRAMIDES | 77.6 |
| 15 | 15081 | MEXICO | TECAMAC | 79.3 |
| 15 | 15083 | MEXICO | TEMAMATLA | 79.1 |
| 15 | 15091 | MEXICO | TEOLOYUCAN | 77.7 |
| 15 | 15092 | MEXICO | TEOTIHUACAN | 77.5 |
| 15 | 15095 | MEXICO | TEPOTZOTLAN | 79.5 |
| 15 | 15099 | MEXICO | TEXCOCO | 80.0 |
| 15 | 15100 | MEXICO | TEZOYUCA | 78.4 |
| 15 | 15104 | MEXICO | TLALNEPANTLA DE BAZ | 81.8 |
| 15 | 15108 | MEXICO | TULTEPEC | 79.1 |
| 15 | 15109 | MEXICO | TULTITLAN | 80.2 |
| 15 | 15120 | MEXICO | ZUMPANGO | 76.8 |
| 15 | 15121 | MEXICO | CUAUTITLAN IZCALLI | 82.8 |
| 15 | 15122 | MEXICO | VALLE DE CHALCO
SOLIDARIDAD | 77.7 |



R PARA GEOESTADÍSTICA

se terminó de editar en febrero de 2026

por Cromberger Editores e Impresores, S. A. de C. V.

Este libro, orientado a los estudiosos de los fenómenos sociales, es una herramienta útil para resumir gran cantidad de información a través de diversos procedimientos estadísticos, modelaciones cartográficas, tablas y gráficos con la finalidad de predecir el comportamiento de una variable, establecer correlaciones y dependencias para identificar efectos e impactos en el territorio.

R es un *software* libre que permite codificar una rutina con procedimientos y funciones. Su renderización nos ofrece insumos para contrastar nuestras hipótesis de investigación.

Rmarkdown, plataforma anclada a *R*, permite trabajar de manera total una investigación. No es necesario intercambiar en el ordenador diferentes programas. A través de sus librerías se genera un reporte global que incluye: texto, imágenes, cálculos, fórmulas, cartografía, referencias, entre otros elementos. Un conjunto importante para diseminar los resultados de investigación: tesis, libros, presentaciones animadas, pósteres y rutinas se tejen desde aquí.

