R para la investigación cuantitativa

Especialización en Investigación Científica UNEMI

Laboratorio de Investigación para el Desarrollo del Ecuador

Instructor: Daniel Sánchez, MA

Conocimientos previos: Habilidades básicas en computación, Git, y shell.

1 Resumen del módulo

Este módulo introduce los conceptos básicos del lenguaje de programación R para el análisis de datos y la computación estadística. El material está diseñado para quienes buscan utilizar R como una herramienta para métodos de investigación cuantitativa.

Asumo que los estudiantes tienen poca experiencia previa en programación: familiaridad con habilidades básicas de computación, Git y la línea de comandos. Cubriremos muchos conceptos importantes del lenguaje, sin embargo, el enfoque no está en la programación técnica, sino en entender el lenguaje como una herramienta para el análisis estadístico.

2 Contenidos del módulo

El siguiente es un esquema planeado del curso. Esto puede cambiar dependiendo del ritmo de la clase.

2.1 Introducción a R (14, 16 y 17 de agosto)

- **Introducción**
  - Introducción al lenguaje R: breve historia, ventajas y aplicaciones.
  - El entorno de desarrollo: R, RStudio, IDEs y paquetes.
  - La interfaz de RStudio: scripts de R, visor de archivos, consola y entorno.
  - Asegurando la reproducibilidad: rutas de archivos, directorios de trabajo y organización de proyectos; uso de proyectos de R con RStudio.

  - Instalación, carga y actualización de paquetes, mejores prácticas.
  - Sintaxis básica de R: variables, operadores de asignación, tipos de datos, estructuras, operadores básicos y lógicos.
  - Obtener ayuda en R: `?`, `help()`. Uso de la documentación de R, CRAN, StackOverflow y otros recursos.
  - Herramientas de inteligencia artificial para la codificación en R: ChatGPT, GitHub Copilot, Microsoft Copilot.
  - Actualización de R: encontrar la GUI de R, ejecutar la actualización y actualizar RStudio.

- **Manipulación básica de datos**
  - Importación básica de datos, R base y paquetes: `read.csv()`, `read.table()`, `read.csv2()`, haven, readr, readxl.
  - Data frames: definición y manipulación básica con la sintaxis base de R: `head()`, `tail()`, `str()`, `summary()`.
  - Gráficos básicos en R: `plot()`, `hist()`, `boxplot()`, `barplot()`.
  - Bucles: bucles while y for, la familia de funciones `map`.

- **Clase 2**: Introducción al tidyverse
  - R base vs *tidyverse*.
  - Introducción al *tidyverse*: el proceso tidy, datos tidy, el operador pipe (`%>%`).
  - Importación de datos intermedia `readr` y `haven` desde diferentes formatos de archivo.
  - Importación de archivos Excel con manipulación preliminar de `readxl`.
  - Descarga de datos desde paquetes de R, GitHub y la web/URLs.
  - Tibbles: el data frame tidy; uso de `glimpse()` para entender los datos.
  - dplyr: Selección y renombramiento de columnas, flujos de trabajo con pipes, `transmute`.

- **Clase 3**: Más sobre la manipulación de datos
  - Filtrado de filas, ordenamiento, agrupación y resumen de datos.
  - Uso de `mutate()` para crear nuevas columnas: `if_else`, `case_when`, `case_match()`.
  - La función `arrange()` para ordenar data frames.
  - Joins, semijoins y antijoins: aplicaciones para investigadores.
  - La función `bind_rows()` para anexar data frames.
  - **Tareas**: Curso de DataCamp *Manipulación de Datos con dplyr* y *Unión de Datos con dplyr*.

- **Clase 4**: Visualización de datos con ggplot2
  - Introducción al paquete `ggplot2`: la gramática de los gráficos.
  - La función `ggplot()`: estéticas, geoms y capas.
  - Gráficos de dispersión, gráficos de líneas, gráficos de barras, histogramas y gráficos de cajas.

  - Facetado: `facet_wrap()` y `facet_grid()`.
  - Temas y personalización: `theme()`, `labs()`, `scale_x_continuous()`, `scale_fill_manual()`.
  - Guardado de gráficos, cambio de tamaño de gráficos y exportación a diferentes formatos.
  - **Tareas**: Curso de DataCamp *Introducción a la Visualización de Datos con ggplot2*.

- **Clase 5**: Reestructuración de datos en R
  - Comprensión de los formatos de datos largos vs. anchos; reestructuración de datos con tidyr.
  - Los equivalentes de `gather()` y `spread()` a `pivot_longer()` y `pivot_wider()`.
  - Las funciones `separate()` y `unite()` para dividir y combinar columnas.
  - La función `separate_rows()` para dividir filas con múltiples valores.
  - Uso del paquete reshape.
  - Expansión de datos: funciones `expand()` y `complete()`.
  - **Tareas**: Curso de DataCamp *Reestructuración de Datos con tidyr* (capítulos selectos).

- **Clase 6**: Informes y reproducibilidad
  - El concepto de investigación reproducible.
  - Introducción a R Markdown: el documento R Markdown, cabecera YAML, chunks de código.
  - Knit de documentos: HTML, PDF, Word y diapositivas.
  - Personalización de documentos R Markdown: temas, plantillas y CSS.
  - Quarto: la próxima generación de R Markdown.
  - Fórmulas matemáticas: incorporación de LaTeX en R Markdown.
  - Si el tiempo lo permite: archivos `.Rnw` y Knitr/Sweave.
  - **Completar antes de la clase**: Curso de DataCamp *Informes con R Markdown*. Asegúrate de tener instalado LaTeXen tu computadora, o TinyTeX como paquete en R[1].

- **Clase 7**: Temas selectos en la programación con R
  - Estructuras de control, declaraciones condicionales y operadores lógicos; relaciones con funciones preempaquetadas.
  - Funciones: cuándo usar, cómo definir funciones, argumentos, valores de retorno.
  - La familia de funciones `apply`: `apply()`, `lapply()`, `sapply()`, `mapply()`.
  - **Tareas**: Curso de DataCamp *R Intermedio para Finanzas*.

- **Clase 8**: Más temas de limpieza de datos
  - Manejo de datos faltantes: enfoques en R base, `tidyverse` y paquetes.
  - Gestión de tiempo y fechas: `lubridate`.

[1] TinyTeX es una distribución ligera, portátil, multiplataforma y fácil de mantener de LaTeX. Está disponible como un paquete en R y se puede instalar con `tinytex::install_tinytex()`

  - Manipulación de cadenas con `stringr`.
  - Uniones difusas: distancia de cadena y el paquete `fuzzyjoin`.
  - **Tareas**: Capítulos de cursos de DataCamp en *Limpieza de Datos en R* y *Trabajar con Fechas y Tiempos en R*.

- **Clase 9**: Computación de alto rendimiento en R

  - ¿Por qué alto rendimiento? La necesidad de velocidad en el análisis de datos.
  - Carga de conjuntos de datos muy grandes: `data.table` y `fread()`.
  - Lectura de muchos archivos a la vez: `purrr` y `map()` con `fread()`.
  - Agrupación y resumen de grandes conjuntos de datos: comparación entre `dplyr` y `data.table`.
  - Otras alternativas: bigmemoryr, ff y bases de datos SQL.
  - **Tareas**: TBD, consulta el repositorio de GitHub y la plataforma DataCamp para actualizaciones.

- **Clase 10**: Acceso avanzado a datos con R

  - Web scraping: el paquete `rvest`.
  - Acceso a datos de Twitter: el paquete `rtweet`.
  - Acceso a datos financieros: