

**Secretaría de Investigación y Posgrado
Facultad de Ciencias Agrarias - UNNE**

Introducción a la Programación en R

- a) Unidad Académica Responsable: Facultad de Ciencias Agrarias.
- b) Destinatarios: Ingenieros, Licenciados, Veterinarios, Investigadores en el área de las ciencias: Biológicas, Exactas, Sociales y Económicas. Docentes.
- c) **Fecha de inicio: 28/07/2023 y finalización: 12/08/2023.**
- d) Modalidad del cursado: Presencial con Estrategia Híbrida: las clases se dictan con actores/participantes en sede o de manera remota, en el espacio aula de video comunicación, bajo protocolos y disposiciones específicas que aseguren las interacciones sincrónicas de los participantes.
- e) Carga horaria total: 30 h (24 horas con actividades guiadas por los docentes y 6 horas de estudio independiente, participación en el aula virtual y elaboración del trabajo final)
- f) Créditos propuestos: 2
- g) Certificaciones a otorgar: Aprobación.
- h) Condiciones a cumplir para la emisión del certificado: título de grado, cumplir con un mínimo de 80% de asistencias presenciales en sede/remota, aprobar la instancia de evaluación.
- i) Coordinación y Docentes a cargo.
 - Coordinadora: Dra. Laura I. Giménez
 - Profesoras Dictantes:
 - Lic. en sistemas (Diplomada) Patricia Loto (25 hs)
 - Bioq. (Dra.) Melisa J. Hidalgo (15 hs).

1.1.1. Programación didáctica del curso

a) Fundamentación

El lenguaje de programación R se ha vuelto popular en varias disciplinas. Ya que es útil para la visualización y limpieza de los datos. Muchos de los científicos que lo necesitan y emplean carecen de entrenamiento en programación.

¿Cuáles son las ventajas de adoptar R en nuestro flujo de trabajo? ¿Es R meramente una herramienta estadística? El objetivo de este curso es brindar una iniciación a este versátil lenguaje de programación a través de sus potencialidades, que incluyen tanto generar gráficos interactivos, reportes, así como aplicaciones.

Siendo un problema en la actualidad la falta de la reproducibilidad en las publicaciones científicas, presentaremos algunos conceptos básicos en relación al análisis de datos reproducible. Además, se proveerá de materiales, sugerencias de foros y comunidades de usuarios para que los asistentes puedan acceder a fuentes de soporte continuo y facilitar así la inclusión de la programación como práctica activa en los flujos de trabajo cotidianos de sus investigaciones.

b) Objetivo General

Comprender generalidades de programación en el entorno del software R y aplicarlas a una necesidad concreta de análisis de datos.

Objetivos Específicos

- Comprender la lógica del pensamiento computacional.
- Comprender la necesidad de documentar el código que se desarrolla, y la importancia del trabajo con control de versiones.
- Desarrollar habilidades en el manejo de datos.
- Aprender los fundamentos de los datos ordenados.
- Aprender a realizar visualizaciones y análisis básicos, que sean efectivas y aptas para ser publicadas.

c) Contenidos

Tema 1: Generalidades y nociones de programación. R e interfaz de RStudio. Reproducibilidad en R. Paquetes. Utilización de la ayuda. Comunidades de usuarios. ¿Qué es tidyverse? Diferencia de sintaxis entre Rbase y Tidyverse.

Tema 2: Manipulación de datos. Ingreso de datos. Datos ordenados (tidy data). Paquetes *tidyr* y *dplyr*. Funciones `select()`, `filter()`, `mutate()`, `summarise()`, `arrange()`, `group_by()`. Otras funciones de *dplyr*. Ejercicios de aplicación

Tema 3: Visualización de datos. Visualización efectiva. Gramática de gráficos. Paquete *ggplot2*. Capas de un gráfico: datos, parámetros estéticos, parámetros geométricos, uso de facets, estadísticas, coordenadas, temas. Ejercicios de aplicación. Breve introducción a la visualización interactiva con el paquete *Plotly*.

Tema 4: Modelización. Generalidades de la modelización de datos. Nociones de regresión y clasificación. Ejercicios de aplicación. Uso de paquetes para modelización de datos.

Tema 5: Documentación y ciencia reproducible. Buenas prácticas de reproducibilidad. Introducción a git y repositorios remotos (GitHub/GitLab). Sintaxis de markdown. Generación de reportes, ¿con Rmarkdown o Quarto?, ventajas y desventajas.

d) Metodología de enseñanza:

Desarrollo de la parte teórica de parte de las docentes a cargo, que a su vez tendrán ejemplos de aplicación. Luego del desarrollo de la parte teórica de cada tema se realizarán actividades prácticas utilizando R con la interface R Studio. Cada participante realizará los ejercicios los que serán agrupados por semana (1,2,3) y ayudando al estudiante a relacionar y aplicar a su propio trabajo. Todos los temas dados se mostrarán Aplicaciones y ejemplos.

- e) Materiales didácticos a utilizar:

Guía Semanal de actividades, aula virtual, Lenguaje R Project y R Studio disponibles en internet.

- f) **Instancias de evaluación y aprobación**

La evaluación consistirá en la presentación de un trabajo integrador individual, que consistirá en el análisis de un set de datos. El trabajo deberá incluir, la importación, el tratamiento y transformación de datos, como también la visualización de los mismos con el paquete ggplot2. La entrega deberá realizarse mediante un archivo comprimido (.zip o .rar) que contenga el proyecto de trabajo o en formato de reporte con quarto (opcional). Se prevé una instancia de recuperación.

- g) Modalidad de la evaluación final: individual.

- h) Requisitos de aprobación del curso

Ochenta por ciento de asistencia y aprobación del trabajo final escrito, con una nota mínima de seis.

- i) Cronograma estimativo:

Semana 1

28/7 Tema 1: (teoría y práctica) 4 horas

29/7 Tema 2: (teoría y práctica) 4 horas

Estudio Guiado (consignas y cuestionarios a desarrollar en el aula virtual) 1 h

Estudio Independiente 1 h

Semana 2

4/8 Tema 3 (teoría y práctica) 4 hs

5/8 Tema 4 (teoría y práctica): 4 hs.

Estudio Guiado (consignas y cuestionarios a desarrollar en el aula virtual) 1 h

Estudio Independiente 1 h

Semana 3

11/8 Tema 5 (teoría y práctica): 5 hs.

Estudio Guiado (consignas y cuestionarios a desarrollar en el aula virtual) 1 h

Estudio Independiente 1 h

12/8 Evaluación 3 hs.

Todas las clases serán dictadas por la Lic. Patricia Loto y la Dra. Melisa J. Hidalgo

- j) Discriminar la cantidad de horas que dictará cada uno de los docentes. *Cantidad de Horas de trabajo de cada docente según actividad y tema.*

Semana 1

Tema 1: (teoría y práctica) 3 horas Lic. Patricia Loto 2 horas Dra. Melisa J. Hidalgo

Tema 2: (teoría y práctica) 3 horas Lic. Patricia Loto 2 hora Dra. Melisa J. Hidalgo

Estudio Guiado (consignas y cuestionarios a desarrollar en el aula virtual) 2 hs. P. Loto y 2 hs Melisa Hidalgo

Semana 2

Tema 3: (teoría y práctica) 3 horas Lic. Patricia Loto 3 horas Dra. Melisa J. Hidalgo
Tema 4: (teoría y práctica) 3 horas Lic. Patricia Loto 1 hora Dra. Melisa J. Hidalgo
Estudio Guiado (consignas y cuestionarios a desarrollar en el aula virtual) 2 horas Lic. Patricia Loto y 2 hs. Melisa Hidalgo

Semana 3

Tema 5 (teoría y práctica): 3 horas Lic. Patricia Loto 3 horas Dra. Melisa J. Hidalgo
Estudio Guiado (consignas y cuestionarios a desarrollar en el aula virtual) 2 horas Lic. Patricia Loto y 2 hs. Melisa Hidalgo

Evaluación: 4 hs. Lic. Patricia Loto y 2 hs. a Dra. Melisa J. Hidalgo

Total Lic. Patricia Loto 25 hs y Dra. Melisa Hidalgo 15 hs.

k) Bibliografía básica

1. Bryan, J. (2015) How to name files. Disponible en <https://speakerdeck.com/jennybc/how-to-name-files> Visitado el 22/07/2020
2. Bryan, J. (2017) Happy Git and GitHub for useR. Disponible en <https://happygitwithr.com/> Visitado el 22/07/2020.
3. Fay, C; Rochette, S; Guyader, V; Girard C. (2020). Engineering Production-Grade Shiny Apps. Disponible en <https://engineering-shiny.org/> Visitado el 22/07/2020
4. Fay, C. (2020) Testing Shiny: why, what and how. Disponible en <https://speakerdeck.com/colinfay/erum-2020-testing-shiny-why-what-and-how> Visitado el 24/07/2020
5. Fay, C. (2019) Engineering Production-Grade Shiny Apps with {golem}
6. <https://speakerdeck.com/colinfay/engineering-production-grade-shiny-apps-with-golem> Visitado el 24/07/2020
7. Gillespie, C & Lovelace, R. (2017) Efficient R Programming. A Practical Guide to Smart Programming. O'Reilly.
8. Grolemund, G. (2014) Hands on programming with R. Write your own functions and simulations. O'Reilly. Disponible en <https://rstudio-education.github.io/hopr/>
9. Grolemund, G; Wickham, H. (2017) R for data science. O'Reilly. Disponible en <https://r4ds.had.co.nz/>
10. Matloff, N. (2011). The art of R Programming. A tour of statistical software design. No Starch Press.
11. ROpenSci Reproducibility in Science. A Guide to enhancing reproducibility in scientific results and writing. Disponible en <http://ropensci.github.io/reproducibility-guide/> Visitado el 22/07/2020
12. Santana, J.S; Farfán, E.M. (2014) El arte de programar en R. Un lenguaje para la estadística. Disponible en https://cran.r-project.org/doc/contrib/Santana_El_arte_de_programar_en_R.pdf
13. Wilke, C. O. (2019) Fundamentals of Data Visualization: A primer on making informative and compelling figures. O'Reilly. Disponible en <https://serialmentor.com/dataviz/>
14. Wickham, H. (2019) Advanced R. Second Edition. CRC Press. Disponible en <https://adv-r.hadley.nz/>