

PROGRAMA DE CURSO
Programación para Estadística I
XS2210

Características del curso

Horas:	4 práctica
Créditos:	4
Requisitos:	MA1021; XS1130
Correquisitos:	
Ciclo:	I-2024
Clasificación:	propio
Grado virtualidad:	Bajo virtual.
Profesor:	Carlos Solís Fonseca (carlos.solisfonseca@ucr.ac.cr)
Consulta:	L: 17:30 – 19:30 (Virtual)

Descripción

Este curso está dirigido a estudiantes de la carrera del Bachillerato en Estadística para que aprendan a programar en un lenguaje estadístico, así como el uso de control de versiones para el registro de cambios en el código y el trabajo colaborativo. El curso está orientado a adquirir destrezas en conceptos básicos de programación, manipulación de archivos de datos, realizar análisis básicos de estadística descriptiva y visualizarlos adecuadamente. Se trata de un curso que hace un uso intensivo del computador para habituar al estudiantado al ambiente de programación en R realizando trabajos prácticos grupales.

Objetivo general

Desarrollar destrezas en programación para manipular datos, realizar análisis básicos de estadística descriptiva y visualizarlos adecuadamente haciendo uso de control de versiones de código.

Objetivos específicos

Al finalizar el curso el/la estudiante estará en la capacidad de:

1. Utilizar software colaborativo para facilitar la programación en R.
2. Aplicar los conceptos básicos de programación para utilizarlos en el lenguaje R.
3. Aplicar funciones de las principales librerías de R en situaciones prácticas.

Habilidades y conocimientos (perfil de salida)

Habilidades	Conocimientos
HM01 - Manejar entidades matemáticas asociadas a la estadística	CM03 - Conocimientos básicos en lógica matemática
HE04 - Aprender de forma autónoma metodología estadística	CE08 - Conocimiento de estrategias de aprendizaje autodidacta
HT01 - Capturar, visualizar, procesar y analizar datos estructurados y no estructurados	CT04 - Conocimientos avanzados en manejo de herramientas de construcción de gráficos en capas (ej: ggplot2, plotly, matplotlib, seaborn)
HT02 - Crear, interpretar y modificar programas de código escrito en lenguajes de programación (ej: R, Python y SQL)	CT08 - Conocimientos intermedios en lenguajes de programación para análisis estadístico (ej: R, Python) CT09 - Conocimientos básicos en diseño de algoritmos CT10 - Conocimientos básicos en Github como portafolio de proyectos para potenciales empleadores
HT03 - Crear y mantener archivos de datos y usar sistemas de bases de datos	CT14 - Conocimientos intermedios en limpieza y revisión de inconsistencias en archivos de datos

Contenidos

1. Aspectos básicos de la programación en R:
 - a) Conceptos básicos de lenguajes de programación.
 - b) Representación de algoritmos: diagramas de flujo.
 - c) R y RStudio, instalación y usos básicos.
 - d) Configuraciones globales de RStudio.
 - e) Uso básico de archivos de extensión .R y .Rmd y su relación con otros lenguajes de programación.
 - f) Tipos de variables.

- g) Operadores de asignación, aritméticos, lógicos, relacionales.
 - h) Estructuras de datos: vectores, matrices, arreglos, listas, conjuntos de datos.
 - i) Indexación y sustitución.
 - j) Estructuras de control iterativas (*for* y *while*).
 - k) Estructuras de control condicionales (*if*, *else*, *ifelse*).
 - l) Familia de funciones *apply*.
 - m) Diferencias entre mensajes, advertencias y errores.
2. Otros procedimientos para programar con R:
- a) Uso de librerías y funciones de R y el conflicto entre librerías.
 - b) Creación de funciones personalizadas en R.
 - c) Lectura y escritura de datos: *.csv*, *.xlsx*, *.sav*, *.txt*.
 - d) Uso de la librería *tidyverse*.
 - e) Procesamiento y manipulación de un *data.frame* con *dplyr*: selección, filtrado, agrupamiento, cálculo y modificación de variables.
 - f) Procesamiento y manipulación de un *data.frame* con *tidyr*: manipulación de formatos largo y ancho en un conjunto de datos.
 - g) Procesamiento y manipulación de datos temporales con *lubridate*.
 - h) Operadores de uso común: *%>%*, *%in%* y *%nin%*
 - i) Unión de datos con las funciones *left_join()*, *right_join()*, *inner_join()*, *anti_join()*, *full_join()* de la librería *dplyr* y con la función *do.call()*.
3. Visualización de datos:
- a) Construcción de gráficos con funciones base de R.
 - b) Construcción de gráficos por capas con *ggplot2*.
4. Análisis de datos:
- a) Medidas de posición y variabilidad en R.
 - b) Medidas de asociación y correlación en R.
 - c) Distribuciones de probabilidad en R: forma de sus funciones (*p_*, *r_*, *q_*).
 - d) Generación de variables aleatorias.
5. Versiones de control con Git:
- a) Control de cambios con Git.
 - b) Flujos de trabajo con Git.
 - c) Colaboración con Git.
 - d) Diferencias entre Git y Github.
6. Productos de datos: Generación de reportes de extensión *.html*, *.docx* y *.pdf*.

Metodología

Este es un curso fundamentalmente práctico. Los contenidos del curso serán abordados en clase (laboratorio) y cada docente desarrollará actividades formativas para apoyar el proceso de enseñanza-aprendizaje. Para abordar los contenidos del curso el personal docente facilitará espacios para la puesta en práctica de los conocimientos adquiridos, por lo que se aplicará una metodología de un aprendizaje activo, donde cada estudiante debe realizar las actividades propuestas por su profesor o profesora, así como asistir y participar en los espacios de trabajo facilitados. El proceso para apoyar el aprendizaje del estudiantado incluye pruebas cortas y trabajos prácticos.

Se utilizará Mediación Virtual para ejecutar y evaluar las pruebas cortas, además, el material para la clase y material complementario se estarán subiendo a esta plataforma para disponibilidad del estudiante. También, se recurrirá a esta plataforma para la entrega de las tareas y exámenes que se ejecutaran durante el semestre, además, será el sitio donde podrán consultar el enlace de horas consulta y medio de comunicación activa.

Evaluación

La evaluación del curso se llevará a cabo mediante la aplicación de cuatro tipos de evaluación: quices, tareas programadas, un proyecto y exámenes. Cada una de las evaluaciones se desarrollan y calificaran de forma individual siguiendo los rubros indicados en las instrucciones dadas en cada una de las evaluaciones.

La distribución porcentual de las evaluaciones que se llevaran a cabo es la siguiente:

Evaluación	Porcentaje en la nota
Quices o pruebas cortas	20% (4 en total 5% c/u)
Tareas programadas	30% (2 en total 15% c/u)
Proyecto	10%
Exámenes parciales	40% (2 en total 20% c/u)

La nota obtenida en el proyecto se divide en cinco rubros:

- Dominio del contenido evaluado (se evalúa individualmente mediante preguntas hechas por el(la) docente asignado(a)).
- Documentación del código (se evalúa grupalmente).
- Lógica y validez de los resultados obtenidos (se evalúa grupalmente).
- Fuentes consultadas para resolver el trabajo (se evalúa grupalmente).
- Eficiencia del código generado (se evalúa grupalmente).

Los quices consistirán en evaluaciones cortas prácticas de alguno de los temas vistos en clase, el profesor indicará cuales son los temas a evaluar.

Las tareas programadas tendrán una complejidad un poco mayor ya que requerirán de un poco de investigación para incentivar a los y las estudiantes a su desarrollo en el tema de investigación. Las asignaciones serán meramente prácticas con una defensa ante el profesor. Si él o la estudiante no presenta esta defensa o es incapaz de realizarla, tendrá un 0 en la calificación.

El proyecto se realizará al final del curso y consistirá en la realización de un proyecto usando Github como herramienta de colaboración. Este se realizará en grupos (definir cantidad de estudiantes) y deberá presentarse ante el profesor.

Los exámenes evaluarán de forma individual los conocimientos adquiridos por las y los estudiantes y abarcarán todos los temas vistos en el curso. Se desarrollarán de manera práctica mediante problemas de desarrollo.

Parcial 1: Temas 1.a hasta 2.b inclusive.

Parcial 2: 2.c hasta 4.d inclusive y tema 6 por completo.

Cronograma

SEMANA / FECHA	CONTENIDO	DETALLE
1 / 13 de marzo	Lógica y Mate Discreta	1.a - 1.b
2 / 20 de marzo		1.c - 1.d - 1.e - 1.f - 1.g (Quiz 1)
3 / 27 de marzo	Introducción a R	(Semana Santa)
4 / 3 de abril		1.h - 1.i - 1.j - 1.k (Asignación Tarea 1)
5 / 10 de abril		1.l - 1.m (Quiz 2)
6 / 17 de abril	R Intermedio	2.a - 2.b
7 / 24 de abril		2.c - 2.d - 2.e (Entrega Tarea 1)
8 / 1 de mayo		(Feriado) (Parcial 1 – Sab 4 de mayo)
9 / 8 de mayo		2.f - 2.g - 2.h (Repo. Parcial I 11/05)
10 / 15 de mayo		2.i (Quiz 3) (Asignación Tarea 2)
11 / 22 de mayo	Visualización	3.a – 3.b
12 / 29 de mayo	Git	Tema 5 (Asignación Proyecto)
13 / 5 de junio	Análisis de datos	4.a - 4.b - 4.c – 4.d
14 / 12 de junio	Productos de datos	Tema 6 (Quiz 4) (Entrega Tarea 2)
15 / 19 de junio		Proyecto
16 / 26 de junio		(Parcial 2-29/06) (Repo. Parcial II 03/07)
17	Ampliación	6 de julio

Bibliografía

- Aguilar Fernández, E. & Zamora Araya, A. (2020). Introducción a la estadística descriptiva con R. <https://tinyurl.com/2oo9bmwk>
- Pace, L. (2012). Beginning R: An Introduction to Statistical Programming. <https://tinyurl.com/2qj7tyaz>
- Venables, W.N. & Ripley, D. M. (2002). An Introduction to R. <https://tinyurl.com/y9u5nxsy>
- Wickham, H. & Grolemund, G. (2018). R para Ciencia de Datos. <https://es.r4ds.hadley.nz/>