



PROGRAMA DE CURSO
Programación para Estadística II
XS0130

Características del curso

Horas:	4 práctica
Créditos:	4
Requisitos:	XS0129 ó XS2210 XS3150
Correquisitos:	Ninguno
Ciclo:	II-2025
Clasificación:	propio
Grado virtualidad:	Bimodal
Profesor:	Carlos Solís Fonseca (carlos.solisfonseca@ucr.ac.cr)
Horario:	L: 17:00-18:50 (Presencial) / M: 19:00-20:50 (Virtual)
Consulta:	M: 17:00 – 19:00 (Virtual)

Descripción

Este curso está dirigido a estudiantes de segundo año de la carrera Bachillerato en Estadística y busca profundizar en los conceptos de presentación de la información y en fundamentos de programación utilizando los lenguajes R y Python. Se trata de un curso que hace un uso intensivo del computador para habituar al estudiantado al ambiente de programación en Python para la manipulación de información, realización de estadística descriptiva, ajuste de modelos y visualización de datos.

Objetivo general

Reforzar las destrezas en programación para manipular datos, realizar análisis estadísticos y ajuste de modelos, así como visualización mediante el uso de los lenguajes programación (R y Python).





Objetivos específicos

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

1. Crear aplicaciones de visualización de datos utilizando la librería Shiny.
2. Aplicar los conceptos básicos de programación en Python.
3. Aprender a manipular datos utilizando las librerías NumPy y Pandas.
4. Realizar análisis estadísticos descriptivos y modelos estadísticos utilizando Python.

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) CT05 - Conocimientos intermedios en herramientas informáticas de visualización (ej: Shiny, Powe BI y Tableau)
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 CT11 - Conocimientos básicos en el uso de herramientas en la nube (ej: Rcloud, Google colab, Jupyter)
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. Introducción a Python
 - a) Uso de IDEs y plataformas como: Google colab y jupyter notebooks
 - b) Introducción a la programación orientada a objetos.
 - c) Tipos de datos y variables.
 - d) Operadores de asignación, aritméticos, lógicos, relacionales.
 - e) Estructuras de datos: listas, diccionarios, conjuntos de datos.
 - f) Indexación y sustitución.
 - g) Estructuras de control iterativas (*for*, *while*).
 - h) Estructuras de control condicionales: *if*, *else*, *elif*.
2. Otros procedimientos para programar con Python
 - a) Uso de librerías y funciones de Python.
 - b) Creación de funciones personalizadas en Python.
 - c) Lectura y escritura de datos: *.csv*, *.xlsx*, *.sav*, *.json*, *.xml*.
 - d) Uso de la librería NumPy: Arreglos n-dimensionales, indexación, filtrado de arreglos.
 - e) Uso de la librería Pandas: Ordenamiento, selección, filtrado, cálculos agrupados sobre un DataFrame.
 - f) Procesamiento y manipulación de un DataFrame: Uso de funciones *Stack* y *unstack*.
 - g) Unión de distintos DataFrame: función *merge*
 - h) Procesamiento de datos temporales: *datetime* y *date*
3. Visualización de datos
 - a) Usos de las librerías para gráficos: *Matplotlib*, *Seaborn* y *Plotly*.
 - b) Mapas: Uso de las librerías *geopandas* y *leaflet*.
4. Desarrollo de visualizaciones de datos en Python.
 - a) Ejemplos de aplicaciones *Shiny*.
 - b) Estructura básica de una aplicación con *Shiny*.
5. Estadística descriptiva con Python.
 - a) Cálculo de medidas de posición y variabilidad.



- b) Cálculo de medidas de asociación y correlación.
 - c) Generación de variables aleatorias.
6. Ajuste de modelos en Python.
- a) Familiarización con las funciones para modelado, preprocesamiento y medidas de desempeño de la librería sklearn
 - b) Desarrollo de modelos de regresión lineal con sklearn (regresión lineal y LASSO)
 - c) Revisión de supuestos de modelos en Python (normalidad, independencia, homoscedasticidad)



Metodología

Los contenidos del curso son abordados en clase (laboratorio) y el/la docente desarrolla actividades formativas para apoyar el proceso de enseñanza-aprendizaje. Para abordar los contenidos del curso el personal docente facilita espacios para la puesta en práctica de los conocimientos adquiridos, por lo que se aplica una metodología de un aprendizaje activo, donde cada estudiante debe realizar las actividades propuestas por su profesor(a), así como asistir y participar en los espacios de trabajo facilitados.

Evaluación

La evaluación del curso se llevará a cabo mediante la aplicación de cuatro tipos de evaluación: laboratorios, un anteproyecto, un proyecto y exámenes. El proyecto y anteproyecto se evaluarán de forma grupal. Los laboratorios y exámenes se evaluarán de forma individual.

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

Evaluación	Porcentaje en la nota
Laboratorios	20% (4 en total 5% c/u)
Proyectos	40% (2 en total 20% c/u)
Exámenes parciales Parcial 1: Temas 1 y 2 (a, b, c y d). Parcial 2: Temas 2 (e, f, g y h), 3, 5 y 6.	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 proyectos tendrán como objetivo brindar a los y las estudiantes una oportunidad práctica de poner en práctica los temas vistos en el curso. En total se realizarán 2 proyectos. El primero de ellos se enfocará en evaluar las capacidades en la creación y uso de clases usando Python. En el segundo se evaluará el dominio usando Git y Shiny en Python. Estos se realizarán en grupos (definir cantidad de estudiantes) y deberá presentarse ante el profesor.

Los laboratorios se evaluarán en el período de una clase y buscará que el estudiante comprenda de forma práctica, los temas expuestos y explicados por el profesor. En los laboratorios se calificará la resolución de forma correcta de las preguntas planteadas.

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.



Cronograma

SEMANA / FECHA	CONTENIDO	DETALLE
1 / 11-08-2025	Introducción a Python	1.a – 1.d
1 / 13-08-2025		1.a – 1.d
2 / 18-08-2025		1.e – 1.f
2 / 20-08-2025		Práctica
3 / 25-08-2025		Laboratorio 1
3 / 27-08-2025		1.fg – 1.h
4 / 01-09-2025	Otros procedimientos para programar con Python	2.a – 2.b
4 / 03-09-2025		Práctica
5 / 08-09-2025		Laboratorio 2 (Asig. Proyecto 1)
5 / 10-09-2025		2.c – 2.d
6 / 15-09-2025		Feriado
6 / 17-09-2025		2.e – 2.f
7 / 22-09-2025		Examen 1
7 / 24-09-2025	2.g – 2.h	
8 / 29-09-2025	Visualización en Python	3.a
8 / 01-10-2025		Presentación Proyecto 1
9 / 06-10-2025		3.a – 3.b
9 / 08-10-2025		Práctica
10 / 13-10-2025		3.b
10 / 15-10-2025		Práctica
11 / 20-10-2025	Visualización con Shiny	Laboratorio 3
11 / 22-10-2025		4.a – 4.b
12 / 27-10-2025	Estadística descriptiva	5.a – 5.b (Asig. Proyecto 2)
12 / 29-10-2025		Práctica
13 / 03-11-2025		5.c – 6.a
13 / 05-11-2025		Práctica
14 / 10-11-2025	Ajuste de modelo	Laboratorio 4
14 / 12-11-2025		6.b – 6.c
15 / 17-11-2025	Cierre del curso	Atención dudas
15 / 19-11-2025		Práctica Examen





16 / 24-11-2025		Examen 2
16 / 26-11-2025		Presentación Proyecto 2
08-12-2025		Ampliación

Sombreado en gris: Clase presencial.

Fondo blanco: Clase virtual.



Bibliografía

- Haslwanter, T. (2016). An introduction to statistics with python: With applications in the life sciences.
- Miller, C. (2018). Hands-On Data Analysis with NumPy and Pandas. Birmingham: Packt Publishing, Limited.
- McMullen, K. (2023). Readings from programming with Python. Cengage Learning.
<https://ebooks724.proxyucr.elogim.com:443/?il=34423>
- Johansson, R. (2024). Numerical Python: Scientific Computing and Data Science Applications with Numpy, SciPy and Matplotlib. Apress Berkeley.
<https://springerlink.proxyucr.elogim.com/book/10.1007/979-8-8688-0413-7#bibliographic-information>
- Nelli, F. (2023). Python Data Analytics: With Pandas, NumPy, and Matplotlib. Apress Berkeley.
<https://springerlink.proxyucr.elogim.com/book/10.1007/978-1-4842-9532-8#bibliographic-information>
- Muddana, A. (2024). Python for Data Science. Springer Cham.
<https://springerlink.proxyucr.elogim.com/book/10.1007/978-3-031-52473-8>