

1. Información del equipo pedagógico y horarios de atención a estudiantes

Profesor magistral: Lucas Gómez Tobón
Correo electrónico: l.gomezt@uniandes.edu.co
Horario de atención a estudiantes: Agendar por correo.
Lugar de atención a estudiantes: <https://uniandes-edu-co.zoom.us/my/lucasantonio2>

Asistente de docencia: Santiago Neira Hernandez
Correo electrónico: s.neira10@uniandes.edu.co
Horario de atención a estudiantes: Agendar por correo.

Asistente de docencia: Fernando Castrillón
Correo electrónico: f.castrillon@uniandes.edu.co
Horario de atención a estudiantes: miércoles 6 a 7 PM vía Zoom.
Lugar de atención a estudiantes: <https://uniandes-edu-co.zoom.us/j/4186348535?pwd=cDE2L2l5elhQY3FJOW1ETDZDbmUzQT09>

2. Descripción del curso

Este curso busca desarrollar habilidades de programación en el contexto de análisis de datos. Está dirigido a un público no familiarizado con la programación, o al menos no necesariamente en el contexto de análisis de datos. Se cubre desde una introducción rápida al ecosistema Python y a la sintaxis de sus estructuras de datos y control, seguido de tutoriales introductorios sobre manipulación, exploración y análisis de datos con paquetes ampliamente usados, como Numpy, Pandas, Seaborn, entre otros.

Este curso podrá darles herramientas a los estudiantes para trabajar en las clases en donde se requiera manipular datos, así como con el desarrollo de las tesis aplicadas.

3. Resultados de aprendizaje

Al final del curso los estudiantes estarán en capacidad de:

- Realizar operaciones matemáticas vectoriales y matriciales con numpy.
- Limpieza y manipulación de bases de datos.
- Crear visualizaciones de datos con las librerías matplotlib, seaborn y plotly.
- Automatizar tareas.
- Realizar análisis estadísticos descriptivos.

4. Cronograma

Modulo 1. Fundamentos de Python. Este módulo es el Core del Taller donde los estudiantes van a aprender la estructura del lenguaje y su sintaxis. Durante 5 semanas vamos a aprender la instalación de Python y sus librerías, los tipos de clases, las principales estructuras de datos y de control de flujo.

Semana	Semana 1 – enero 27, 2025
Objetivos de la semana	<ul style="list-style-type: none"> • Comprender qué es Python y cuáles son sus características como lenguaje de programación. • Entender las diferencias entre Python y Anaconda. • Entender qué son los Integrated Development Environments (IDEs). • Aprender a instalar Python y sus demás requerimientos funcionales. • Aprender a manipular los objetos básicos de Python: tipos numéricos, secuencias, diccionarios, conjuntos y booleanos. • Instalación y cargue de librerías. • Link para el registro de equipos para la solución de talleres: https://forms.gle/ZJWsfT71MSia8NZX8
Descripción	Se hará una introducción al lenguaje de programación Python y a las demás herramientas tecnológicas que complementarán su uso, como lo son git y los IDEs. Se asistirá a los estudiantes con la instalación de todos los requerimientos computacionales.

Semana	Semana 2 – febrero 3, 2025
Objetivos de la semana	<ul style="list-style-type: none"> • Programar listas. • Programar tuplas. • Entender la sintaxis de los diccionarios y programarlos. • Realizar operaciones matriciales y vectoriales.
Descripción	Familiarizarse con los diferentes tipos de iterables (listas, tuplas y diccionarios). Además se repasarán conceptos de algebra lineal utilizando Python.

Semana	Semana 3 – febrero 10, 2025
Objetivos de la semana	<ul style="list-style-type: none"> • Trabajo en equipo durante clase (A1) • Programar condicionales: if, elif y else.
Descripción	Python es muy útil para automatizar procesos, pero para poder hacerlo es importante dominar los conceptos de Control Flow. A partir de condicionales podrá crear códigos contingentes para diferentes escenarios.

Semana	Semana 4 – febrero 17, 2025
Objetivos de la semana	<ul style="list-style-type: none"> • Recapitular los contenidos de la Semana 3. • Programar loops (ciclos): for y while. • Identificar cuándo es necesario utilizar un loop o un condicional. • Entender el enunciado del Taller 1.
Descripción	Los loops son estructuras que permiten ejecutar un bloque de código repetidamente, ya sea recorriendo elementos de un iterable (como listas, cadenas o rangos) o hasta que se cumpla

	<p>una condición específica. Son especialmente útiles para automatizar tareas repetitivas evitando la necesidad de escribir instrucciones repetitivas manualmente.</p> <p>Se presentará el enunciado del primer taller el cual se comenzará a resolver en clase para apoyar a los estudiantes resolviendo dudas.</p>
Evaluación	<p>Taller #1 (22.5%). Simulación para estimar el número de sobres necesarios para llenar el álbum de Panini. <u>Se evaluará el uso de funciones, loops y numpy básico.</u> Se les muestra a los estudiantes durante la clase de febrero 17 para ser entregado antes de la clase de marzo 3.</p>

Semana	Semana 5 – febrero 24, 2025
Objetivos de la semana	<ul style="list-style-type: none"> • Aprender a programar funciones • Identificar cuándo es necesario construir funciones • Aprender a documentar funciones según los estándares de Google • Construir lambda functions • Utilizar el concepto de recursividad para construir funciones más eficientes.
Descripción	Las funciones permiten construir un código modular, más ordenado y fácilmente reproducible. Enseñaremos cómo dividir una tarea en varias funciones y como construir un pipeline de desarrollo adecuado.

Modulo 2. Data Frames y pandas. Introducción a la principal librería para economistas, data analyst y data scientist en Python: pandas. Durante 4 semanas vamos a aprender a importar, limpiar y manipular datos estructurados. También veremos cómo lidiar con missing values, los diferentes tipos de pegues entre bases y la creación de estadísticas para describir nuestros datos.

Semana	Semana 6 – marzo 3, 2025
Objetivos de la semana	<ul style="list-style-type: none"> • Entrega Taller 1 y solución de dudas. • Construir dataframes • Filtrar filas y columnas según condiciones • Crear variables • Renombrar columnas • Calcular principales estadísticos de un dataframe • Importar datos externos: csv, exceles, txt, entre otros.
Descripción	Introducción a la librería pandas: Manipulación de bases de datos para su limpieza

Semana	Semana 7 – marzo 10, 2025
Objetivos de la semana	<ul style="list-style-type: none"> • Pegar filas y columnas. • Pegar data frames: inner, left, right, outer joins • Exportar data frames • Group by • Manipulación de fechas • Entender el enunciado del Taller 2.

Descripción	Continuación de la librería pandas
Evaluación	Taller #2 (22.5%). Funciones recursivas y análisis de datos. Se evaluará la construcción de funciones recursivas y manipulación y limpieza de datos con pandas. Se les muestra a los estudiantes durante la clase de marzo 10 para ser entregado antes de la clase de abril 7.

Semana	Semana 8 – SEMANA DE RECESO – marzo 17, 2025
---------------	--

Semana	Semana 9 – FESTIVO – marzo 24, 2025
---------------	-------------------------------------

Descripción	<ul style="list-style-type: none"> • Plazo para subir las notas parciales a MiBanner: 28 de marzo.
--------------------	--

Modulo 3. Visualizaciones. Esté es un módulo corto para aprender a construir gráficas en Python, personalizarlas y añadir elementos interactivos.

Semana	Semana 10 – marzo 31, 2025
---------------	----------------------------

Objetivos de la semana	<ul style="list-style-type: none"> • Entrega Taller 2 y solución de dudas. • Trabajo en equipo durante clase (A2) • Construcción de scatter plots • Construcción de line plots • Construcción de bar plots • Construcción de gráficos por grupos
-------------------------------	--

Descripción	Introducción a las visualizaciones. Librerías matplotlib y seaborn
--------------------	--

Semana	Semana 11 – abril 7, 2025
---------------	---------------------------

Objetivos de la semana	<ul style="list-style-type: none"> • Introducción a plotly • Construcción de mapas para visualizar variables socioeconómicas • Gráficos de densidad • Identificación de hotspots
-------------------------------	--

Descripción	Introducción a las visualizaciones espaciales e interactivas.
--------------------	---

Modulo 4. Web Scraping. Esté es un módulo para aprender a construir bots que naveguen y recolecten datos de internet de forma automática y estructurada. Aprenderemos a diferenciar los elementos básicos de internet como lo son html, css y javascript para navegar páginas estáticas y dinámicas.

Semana	Semana 12 – abril 14, 2025
---------------	----------------------------

Objetivos de la semana	<ul style="list-style-type: none"> • Entender los conceptos básicos de html y css. • Aprender a hacer un get request • Identificar páginas estáticas y dinámicas • Hacer consultas específicas para construir tablas
-------------------------------	--

Descripción	Introducción al webscraping en Python
--------------------	---------------------------------------

Semana	Semana 13 – abril 21, 2025
Objetivos de la semana	<ul style="list-style-type: none"> • Continuación de web scraping en páginas dinámicas. • Introducción a selenium. • Entender el enunciado del Taller 3.
Descripción	Introducción al webscraping en Python
Evaluación	Taller 3 (22.5%). Webscraping Win Sports y Metro Cuadrado. <u>Se evaluará el uso de las librerías requests, bs4 y selenium para hacer scrapings de páginas dinámicas.</u> Se les muestra a los estudiantes durante la clase del 21 de abril para ser entregado antes de la clase del 5 de mayo.

Semana	Semana 14 – abril 28, 2025
Objetivos de la semana	<ul style="list-style-type: none"> • Trabajo en equipo durante clase (A3).
Descripción	Ejercicios de web scraping.

Modulo 5. Miscelánea. Se presentará a los estudiantes diferentes aplicaciones prácticas de Python con el propósito de motivar a los estudiantes a explorar diferentes librerías y ejercitar su trabajo autónomo.

Semana	Semana 15 – mayo 5, 2025
Objetivos de la semana	<ul style="list-style-type: none"> • Entrega Taller 3 y solución de dudas. • Trabajo en equipo durante clase (A4). • Estimación de regresión por MCO de forma manual. • Multiplicación de matrices. • Cálculo de matriz var-cov. • Aprender a especificar modelos • Aprender a exportar resultados • Aprender a crear gráficas de ajustes
Descripción	Introducción a la econometría

Semana	Semana 16 – mayo 12, 2025
Objetivos de la semana	<ul style="list-style-type: none"> • Trabajo en equipo durante clase (A5). • Aprender como los computadores entienden el lenguaje • Aprender pipeline de limpieza: eliminar stopwords, lematización, stemming, homogenización de formato. • Construir nubes de palabras • Expresiones regulares
Descripción	Introducción al NLP
Evaluación	Taller #4 (22.5%). Tópico libre. Este taller tiene como objetivo fomentar la exploración autónoma y la presentación didáctica de paquetes de Python no cubiertos en clase. Cada equipo seleccionará un paquete de su interés, lo explorará en profundidad y creará un Jupyter Notebook que sirva como un tutorial detallado para otros usuarios interesados en aprender sobre este paquete. El plazo final para la entrega es el lunes 26 de mayo a las 7:30pm.

Semana	Semana 17 – mayo 19, 2025
---------------	---------------------------

Objetivos de la semana	<ul style="list-style-type: none"> • Trabajo en equipo durante clase. • Entendimiento de los datos espaciales y sus diferentes formatos • Introducción a la librería geopandas • Visualización
Descripción	Introducción a los datos espaciales

5. Referencias (Todas gratis!!!)

- Todos los códigos y jupyternotebooks de la clase pueden ser encontrados en el repositorio de GitHub de la clase: https://github.com/lgoomez/Intro_Python
- Galería de visualizaciones de Python con código para replicabilidad. <https://python-graph-gallery.com>
- Markdown CheatSheet. <https://github.com/adam-p/markdown-here/wiki/Markdown-Cheatsheet>
- Intermediate Quantitative Economics with Python. Por Thomas J. Sargent, sí, el Nobel de Economía, y John Stachurski) <https://python.quantecon.org/intro.html>
- Causal Inference for The Brave and True. Basado en las clases de Joshua Angrist, Alberto Abadie and Christopher Walters en econometría pero en Python y con memes! <https://matheusfacure.github.io/python-causality-handbook/landing-page.html>
- Python Data Science Handbook. Essential Tools for Working with Data. <https://jakevdp.github.io/PythonDataScienceHandbook/>

6. Metodología

Todas las sesiones comenzarán con 10 minutos de respuesta a preguntas y de socialización de respuesta de las tareas no evaluativas. Luego les seguirán 50 minutos de explicación del tema acompañados de ejercicios sincrónicos y finalmente la clase tendrá de 20 a 30 minutos de ejercicios para que los estudiantes resuelvan guiados en menor medida por el profesor. Las sesiones se realizarán por Zoom.

Nota: En caso de que alguna sesión se cruce con un festivo, se el contenido se cubrirá de forma asincrónica. Para la solución de las dudas habrá un foro y también horarios de atención.

7. Evaluaciones

La clase cuenta con dos tipos de actividades evaluativas: actividades de clase y talleres grupales.

En total se enviarán cuatro talleres que equivaldrán al 80% de la nota del curso (cada uno tendrá un valor de 20%). Los talleres se deberán realizar en equipos entre 2 a 3 personas como máximo. No se aceptarán talleres entregados de forma individual. Los equipos para la realización de los talleres se deberán registrar en el siguiente link antes de la segunda sesión de clases <https://forms.gle/iwzfPqjTw1AF2g238>. Solo un integrante por equipo debe llenar el formulario.

- Taller #1 (20%). **Simulación para estimar el número de sobres necesarios para llenar el álbum de Panini.** Se evaluará el uso de funciones, loops y numpy básico.

- Taller #2 (20%). **Funciones recursivas y análisis de datos.** Se evaluará la construcción de funciones recursivas y manipulación y limpieza de datos con pandas.
- Taller #3 (20%). **Webscraping Win Sports y Metro Cuadrado.** Se evaluará el uso de las librerías requests, bs4 y selenium para hacer scrapings de páginas dinámicas.
- Taller #4 (20%). **Tópico libre.** Este taller tiene como objetivo fomentar la exploración autónoma y la presentación didáctica de paquetes de Python no cubiertos en clase. Cada equipo seleccionará un paquete de su interés, lo explorará en profundidad y creará un Jupyter Notebook que sirva como un tutorial detallado para otros usuarios interesados en aprender sobre este paquete.

Los talleres en clase se evaluarán teniendo en cuenta: 1) Organización del código/script, 2) Que el código genere un resultado correcto y 3) Número de errores en el código.

Los talleres deben ser presentados a través de Bloque Neón. Para aprobar el curso cada estudiante debe alcanzar una nota igual o superior a 3.00.

El 20% de la nota final hará parte de 6 sesiones de trabajo en clase. En estas se conformarán grupos de alrededor de 3 estudiantes de forma aleatoria para resolver actividades relacionados con los tópicos del módulo que se esté desarrollando. **La evaluación de estos talleres solo tendrá en cuenta que el estudiante esté presente durante toda la clase, tenga su cámara prendida y colabore con sus compañeros.**

Las calificaciones definitivas de las materias serán numéricas de uno punto cinco (1,50) a cinco punto cero (5,00), en unidades, décimas y centésimas.

Según los artículos 62 y 63 del Reglamento general de estudiantes, el estudiante tendrá ocho días hábiles después de la entrega de la evaluación calificada para presentar un reclamo. El profesor responderá al reclamo en los diez días hábiles siguientes. Si el estudiante considera que la respuesta no concuerda con los criterios de evaluación podrá solicitar un segundo calificador al Consejo de la Facultad en los ochos días hábiles posteriores a la recepción de la decisión del profesor.

8. Políticas generales de los cursos de Economía y fechas importantes

Los estudiantes deben consultar [este enlace](#), donde se encuentran las reglas sobre asistencia a clase, excusas válidas, fraude académico y faltas disciplinarias, reclamos, políticas de bienestar y fechas importantes del semestre.