

# Python para Economistas

Maestría en Economía, Facultad de Ciencias Económicas, UNLP

## Objetivo

Este curso tiene por objetivo enseñar las bases de la programación en Python para ampliar las herramientas computacionales que los economistas podemos utilizar tanto en el ámbito académico como en el profesional. Basándonos en el conocimiento que los alumnos tienen de Stata, se buscará profundizar en las funciones de Python que pueden servir de complemento. A diferencia de un curso genérico de introducción a Python, este curso presentará ejemplos de aplicaciones en el área de las ciencias económicas para practicar los contenidos aprendidos con temas familiares a los alumnos.

El curso se dictará mediante herramientas virtuales asincrónicas (videos) y sincrónicas (por la plataforma Zoom). En las clases asincrónicas, se utilizarán unos pocos minutos (aprox. 10 a 15) para presentar algunas herramientas para el desarrollo y versionado del código, y luego se explicarán, con breves ejemplos, los temas del día. En las clases sincrónicas, se aclararán dudas de los videos y se conectarán los conceptos teóricos con ejemplos prácticos. Estas clases se apoyarán en Jupyter notebooks en los cuales estará parte del código requerido para completar los trabajos prácticos y se dejarán líneas de código en blanco para que el alumno pueda razonar y completar el programa.

Los Jupyter notebooks pretenden ser la base de códigos replicables que el alumno, a futuro, podrá adaptar y utilizar en sus propios proyectos. Este curso no pretende ser exhaustivo, pero busca generar en el alumno la confianza necesaria para poder leer autónomamente documentación y código en internet y utilizar estas herramientas acorde a las necesidades específicas de sus proyectos de investigación o sus tareas profesionales.

## Docentes

Rodrigo Bonazzola estudió la licenciatura y maestría en física en el Instituto Balseiro. Trabajó en el área de genómica computacional en la universidad de Chicago, desarrollando métodos estadísticos que permiten entender la base genética de enfermedades mediante la integración de distintos tipos de datos. Actualmente es candidato doctoral en computación (Leeds University, Reino Unido) investigando los factores genéticos que afectan la forma del corazón, usando técnicas de aprendizaje profundo para procesamiento de imágenes médicas. Contacto: [rodbonazzola@gmail.com](mailto:rodbonazzola@gmail.com)

Belén Michel Torino es candidata doctoral en economía (UdeSA) y tiene maestrías en Economía (UNLP) y en Análisis Computacional aplicado a Políticas Públicas (University of Chicago). Ha trabajado en evaluación y monitoreo de programas sociales y en áreas de data management/business intelligence en organizaciones como el Banco Mundial, el Banco Inter-Americano de Desarrollo (BID), la organización internacional de las migraciones (NU-OIM) y la administración nacional de la seguridad social (ANSES). Contacto: [belenmichel@gmail.com](mailto:belenmichel@gmail.com)

## Asistentes

Cristian Bonavida: [cristianbonavida@gmail.com](mailto:cristianbonavida@gmail.com)

Julián Borgo: [borgojulian@gmail.com](mailto:borgojulian@gmail.com)

Nicolás Abbate: [abbatenicolas@gmail.com](mailto:abbatenicolas@gmail.com)

### Fechas y horarios de clases

- Clases asincrónicas (videos): se pondrán a disposición la semana previa a la clase por Zoom.
- Clases sincrónicas (zoom): Viernes de 17 a 19hs (23/9, 30/9, 7/10, 14/10, 21/10, 28/10)

### Modalidad y fechas de clases

- Son **6 clases** en total y trataremos los contenidos que se detallan en el temario. Al final del curso habrá **1 clase opcional** donde se cubrirán temas de Aprendizaje Automático y Aprendizaje Profundo.
- Para cada clase dispondrán de una **presentación en pdf, videos** y un **Jupyter notebook** con código para seguir los ejemplos prácticos y completar partes de código.

### Evaluación

- En cada clase sincrónica, se entregará un trabajo práctico que deberá ser resuelto antes de la siguiente clase sincrónica. Se darán 6 días extras para entregas que, por alguna dificultad personal deban entregar algún trabajo con retraso. Estos días pueden ser distribuidos de acuerdo a la conveniencia de los alumnos a lo largo de los seis trabajos prácticos.
- La nota final se basará en la participación en clase y la calificación de los trabajos prácticos, dándole un peso mayor al TP6 dado que este será integrador.

### Temario

Clase	Temas
0	Instalación del intérprete de Python, una IDE o editor, Anaconda, git y Google Colab.
1	Introducción a la programación en Python: sintaxis básica, tipos de datos (caracteres, enteros, punto flotante, lógicos), operaciones básicas, condicionales, ciclos ( <i>for</i> y <i>while</i> ). Listas, tuplas, sets, diccionarios y métodos asociados.
2	Entrada de y salida a archivos. Funciones. Módulos. Clases y objetos. Atributos y métodos de objetos.
3	Numpy y Pandas. Expresiones regulares. Librerías de gráficos (básico).
4	Profundización en Pandas. Visualizaciones avanzadas. Widgets interactivos en Jupyter.
5	Introducción a datos georreferenciados y elaboración de mapas.
6	Conexión con APIs. Bases de datos relacionales (SQL). Scrapear datos de la web o de un pdf.
7	<b>Clase optativa - disparadora de ideas de investigación:</b> Introducción conceptual a Aprendizaje automático (AA) y aprendizaje profundo (AP). Aplicaciones prácticas.

### Bibliografía

Los siguientes libros son una fuente útil de consulta sobre los temas del curso:

- [Think Python, 2e](#) - Es un buen recurso sobre Python y pensamiento computacional.
- [Python the Hard Way](#) - Es otro buen recurso para aprender Python.