



PROGRAMACIÓN PARA CIENCIA DE DATOS

PRESENTACIÓN

Breve descripción

Programar consiste en comunicarse con el ordenador para lograr que lleve a cabo una serie de tareas. En esta comunicación es fundamental dominar un **lenguaje** que pueda entender tanto una persona como el ordenador. El vocabulario y la gramática requieren de una **sintaxis** especial. En esta asignatura se aprenderá **Python**, especialmente indicado para la ciencia de datos. En realidad, muchos conceptos de programación son comunes a todos los lenguajes; programar es un arte que se puede adquirir y cultivar. El científico que se libera de la condición de usuario para adquirir la de programador (aun a nivel elemental) multiplica su eficiencia a la hora de extraer información de los datos: tal es el objetivo de esta asignatura.

- **Titulación:** MÁSTER UNIVERSITARIO EN CIENCIA DE DATOS PARA CIENCIAS EXPERIMENTALES
- **Módulo/Materia:** Módulo 2 General/Materia 2.2. Ciencia de datos y modelos en ciencias experimentales
- **ECTS:** 5
- **Curso, semestre:** 1º semestre
- **Carácter:** Obligatoria
- **Profesorado:**
Dr. Ángel Garcimartín Montero (Responsable)
- **Idioma:** Se imparte en español (bibliografía en inglés)
- **Aula, Horario:** Consultar el calendario del máster

RESULTADOS DEL APRENDIZAJE

RA1 Aplicar soluciones computacionales para problemas científicos, usando una amplia gama de plataformas de análisis de datos

RA7 Implementar algoritmos eficientes para la resolución de problemas científicos utilizando programación estructurada y pseudocódigo

RA8 Evaluar la aplicabilidad de diferentes lenguajes de programación (Python y Matlab) en el contexto de las ciencias experimentales.

RA9 Emplear con soltura la sintaxis de Python y Matlab para la manipulación de datos y la implementación de algoritmos científicos.

RA10 Seleccionar y aplicar técnicas de tratamiento de datos (ajustes, filtros y procesamiento de señales) para el análisis de datos experimentales.

RA14 Manipular y procesar datos científicos de diversas fuentes (archivos de texto, bases de datos, etc.) utilizando técnicas de data wrangling en diferentes entornos de programación

RA15 Generar e interpretar representaciones gráficas adecuadas para comunicar resultados científicos de manera clara y efectiva en diferentes entornos de programación

PROGRAMA

A. Programa teórico

1. Hardware y software. Sistema operativo. Lenguajes y entornos de programación.



Universidad
de Navarra

Facultad de Ciencias

2. Sintaxis: operadores y expresiones, comandos básicos de Python. Programación orientada a objetos.
3. Programación estructurada. Pseudocódigo. Condiciones, ramificaciones y bucles.
4. Ecosistema Python de tratamiento de datos: módulos numpy, matplotlib, pandas y scipy.
5. Entrada y salida. Escritura y lectura de ficheros. Gráficos.
6. Algoritmos de tratamiento de datos y su implementación en Python.

B. Programa práctico

Se estudiarán casos prácticos de interés científico. Entre ellos, para el curso 2025/26 se propondrán - entre otros- los siguientes:

- Suavizado de datos: la media móvil.
- Simulación de una cola.
- La mapa logística.
- Análisis de supervivencia.
- Data wrangling.
- *Clustering: k-means.*
- *Data wrangling.*

Es necesario acudir a las clases con ordenador portátil.

ACTIVIDADES FORMATIVAS

- **Clases teóricas:** 12 horas. Cubrirán el programa teórico.
- **Clases prácticas:** 18 horas. Se programarán detenidamente algunos casos prácticos.
- **Seminarios:** 17 horas: explicaciones sobre problemas y casos prácticos y presentaciones en público de trabajos personales; implementación de otros métodos y algoritmos de interés.
- **Tutorías:** 3 horas, *ad lib.*

Todas estas actividades son presenciales. La ausencia por motivos razonables debe justificarse.

Además, se requieren unas 50 horas de **trabajo personal** (el tiempo necesario para asimilar la materia puede variar de una persona a otra).

EVALUACIÓN

CONVOCATORIA ORDINARIA

La nota incluirá los siguientes apartados:

1. **Examen final** (50 %). Esta prueba será de carácter práctico y consistirá en la elaboración de un programa.
2. **Notas de clase:** 15%. En este apartado entran los siguientes ítems: - exposición pública de casos prácticos resueltos
 - trabajos y tareas para realizar en casa
 - intervenciones orales y respuestas a preguntas cortas a petición del profesor



Universidad
de Navarra

Facultad de Ciencias

3. Resolución de **casos prácticos**: 15%. Se recogerán durante el curso, y se calificará tanto el cuaderno (diagrama de flujo, etc.) como el programa elaborado.

Nótese que los porcentajes suman el 80%: si solo se resuelven los casos prácticos obligatorios, la nota máxima que se puede obtener es Notable (8). Para sacar más de Notable, es necesario presentar también los casos prácticos voluntarios, de acuerdo con el profesor; en ese caso los porcentajes de los apartados (2) y (3) suben hasta el 20% cada uno de ellos.

Las calificaciones se otorgarán con el siguiente baremo (se indica la correspondencia numérica en una escala de 0 a 10):

A (excelente) = 9; **B** (muy bien) = 7.5; **C** (bien) = 6; **D** (aceptable) = 5; **E** (deficiente) = 3.

Las calificaciones se pueden modular con + y -

Durante el curso se asignarán trabajos, problemas o casos prácticos para entregar en los días fijados por el profesor. Salvo causa justificada **la asistencia a clase es obligatoria**, pues se puede solicitar en cualquier momento al alumno, sin aviso previo, entregar y exponer públicamente el trabajo realizado. La falta de asistencia no justificada puede conllevar una penalización en la calificación de la asignatura.

CONVOCATORIA EXTRAORDINARIA Se repetirá el examen final, que contará con los mismos porcentajes de la convocatoria ordinaria. Las notas de los casos prácticos, notas de clase, y trabajos prácticos (apartados 2 y 3 de la convocatoria ordinaria) se guardan para la convocatoria extraordinaria.

BIBLIOGRAFÍA Y RECURSOS

Muchos recursos no están en forma de libro, sino en herramientas y contenidos digitales que serán presentados durante el curso: por ejemplo, la ayuda en línea de las interfaces de programación.

Las obras disponibles en versión electrónica se pueden encontrar en la sección **Material docente** de esta asignatura, en la carpeta *Colección de libros (en formato PDF)*.

Bibliografía para iniciarse en Python:

- S. R. DOTY, "**Python basics**". Documento gratuito (el pdf está en la carpeta *libros en formato pdf*) . Una introducción rápida, en 20 páginas.
- C. H. SWAROOP, "**A byte of Python**". Libro gratuito, disponible [online - Localízalo en la Biblioteca](#) y en [epub](#), [pdf](#). También se puede comprar una [edición impresa](#) (el pdf está en la carpeta *libros en formato pdf*).
- R. GONZÁLEZ DUQUE, "**Python para todos**". Libro gratuito (el pdf está en la carpeta *libros en formato pdf*). Libro bastante completo sobre Python, pero solo sobre el lenguaje; no trata apenas de los paquetes científicos.

Bibliografía recomendada:

- J. VANDERPLAS, "**Python Data Science Handbook**", excelente manual que explica en profundidad los paquetes Numpy, Pandas, Matplotlib y una introducción Machine Learning. [Versión online](#) (en el repositorio del autor).



Universidad
de Navarra

Facultad de Ciencias

- VARIOS AUTORES, "**NumPy User Guide**". Libro gratuito (el pdf está en la carpeta *libros en formato pdf*) . Contiene dos capítulos de introducción, uno básico y otro intermedio, muy recomendables.

Bibliografía complementaria:

- E. BRESSERT, "**SciPy and NumPy**". [Agotado]. [Hay ejemplares electrónicos]. Muy cortito, es una introducción a los módulos científicos.
- VARIOS AUTORES, "**Scipy Lecture Notes**". Libro gratuito (el pdf está en la carpeta *libros en formato pdf*) . Trata sobre las librerías científicas en Python. Interesante para consultar procedimientos específicos.
- C. HILL, " **Learning scientific programming with Python**". Disponible en versión electrónica en la biblioteca: [Localízalo en la Biblioteca](#)

Sitios web:

- [Centro de documentación de Python](#)
- [NumPy \(computación científica en Python\)](#)
- [matplotlib \(visualización gráfica en Python\)](#)
- [SciPy \(librerías científicas de Python\)](#)