Universidad de Navarra

Programación para ciencia de datos (MCD)

Guía docente 2025-26

PRESENTACIÓN

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 que se trata de aprender. Hay lenguajes especialmente valiosos para el uso científico; en esta asignatura se aprenderá Python , muy adecuado para la ciencia de datos. 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 en Ciencia de Datos para Ciencias Experimentales
- Carácter: Obligatorio. *Módulo 2* (General). *Materia 2.2:* Ciencia de datos y modelos en ciencias experimentales.
- ECTS *: 5.*
- Profesorado: Profesor responsable: Angel Garcimartín Montero
- Idioma: *Se imparte en español (bibliografía en inglés).*
- Aula 1 (Edificio Los Castaños).
- Horario: Primer semestre. Consultar el Calendario del Máster.

Es necesario acudir a las clases con ordenador portátil

PROGRAMA

PROGRAMA TEÓRICO

1. Hardware y software. Sistema operativo. Lenguajes y entornos de programación.
2. IDEs: Jupyter Notebook, Spyder. Ayuda y documentación. Recursos.
3. Sintaxis de Python
    1. Tipos de variable. Operadores. Expresiones. Comandos básicos.
    2. Colecciones: listas, tuplas, diccionarios y sets.
    3. Programación orientada a objetos. Atributos y métodos.
    4. Entrada y salida.
    5. Formateo de texto.
4. Programación estructurada
    1. Pseudocódigo
    2. Control de flujo: condiciones, ramificaciones y bucles.
    3. Funciones
5. Ecosistema Python para tratamiento de datos
    1. Análisis numérico: módulo *numpy*
    2. Gráficos: módulo *matplotlib*
    3. Análisis de datos: módulo *pandas*
6. Algoritmos para ciencia de datos
    1. Módulo *scipy*

PROGRAMA PRÁCTICO

Universidad de Navarra

Aparte de ejercicios sobre la aplicación de los conceptos teóricos, se estudiarán casos prácticos de interés científico. Para el curso 2025/26, se propondrán -entre otros- los siguientes:

- Filtro de suavizado: la media móvil.
- Procesos estocásticos: simulación de una cola.
- Grafos (representación gráfica de relaciones).
- La mapa logísica.
- Análisis de supervivencia.
- *Clustering*: *k-means.*
- Análisis del sonido. Transformada de Fourier.
- *Data wrangling*.

El programa práctico contiene casos adicionales que se pueden realizar de manera opcional.

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 serán de carácter práctico y consistirá en la elaboración de un programa.
2. Notas de clase: 15%. En este apartado entran los siguientes ítems:
    1. exposición pública de casos prácticos resueltos
    2. trabajos y tareas para realizar en casa
    3. intervenciones orales y respuestas a preguntas cortas a petición del profesor
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.

Universidad de Navarra

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 el mismo porcentaje que en 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.

HORARIOS DE ATENCIÓN

Ángel Garcimartín Montero (angel@unav.es)

- Despacho 0-240 (Dpto. de Física y Matemática Aplicada, Edificio Los Castaños)
- Horario de tutoria: Se establecerá un horario de atención de consultas al comienzo de curso de acuerdo con los estudiantes. También se atenderán consultas previa petición de hora por correo electrónico en otros momentos en los que el profesor esté disponible.

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:

- J. VANDERPLAS, "**A whirlwind tour of Python**", (con una versión pdf gratuita en este enlace)
- C. H. SWAROOP, "**A byte of Python**". Libro gratuito, disponible online - Localízalo en la Biblioteca y en epub, pdf. El archivo 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*). Bastante completo sobre Python, pero solo sobre el lenguaje; no trata apenas de los paquetes científicos.

***Bibliografía recomendada, Python*:**

- 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). Localízalo en la Biblioteca

- 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, Python:

- 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**". (El libro electrónico está accesible para estudiantes de la Universidad de Navarra; el archivo está en la carpeta *libros en formato pdf*). 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)

RESULTADOS DE APRENDIZAJE (Competencias)

- 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