

MÉTODOS COMPUTACIONALES

Jaime E. Forero Romero

NOMBRE DEL CURSO: Métodos Computacionales
CÓDIGO DEL CURSO: FISI 2028 (Magistral) FISI 2809 (Laboratorio)
UNIDAD ACADÉMICA: Departamento de Física
PERIODO ACADÉMICO: 201620
HORARIO (MAGISTRAL): Mi 12:30 a 14:00 y Vi, 15:30 a 17:00
HORARIO (LABORATORIO): Ju

NOMBRE PROFESOR(A) PRINCIPAL: Jaime E. Forero Romero
CORREO ELECTRÓNICO: je.forero@uniandes.edu.co
HORARIO Y LUGAR DE ATENCIÓN: Mi 14:00 a 16:00, Oficina Ip208

NOMBRE PROFESOR(A) COMPLEMENTARIO(A):
CORREO ELECTRÓNICO: @uniandes.edu.co

I Introducción

Los métodos computacionales son un aspecto inseparable de cualquier área de trabajo en ciencia e ingeniería. Esto se debe en gran parte a la disminución en costos y complejidad de uso de las computadoras programables, unido al aumento exponencial en su capacidad de procesamiento. Este curso busca guiar a los estudiantes en el desarrollo de sus *habilidades computacionales* en generar y procesar datos para obtener información sobre la realidad que esos datos pretenden describir. Estos datos pueden ser mediciones o simulaciones de sistemas físicos, químicos, geológicos, biológicos, financieros o industriales, entre otros. El programa del curso tiene dos componentes diferenciadas: métodos numéricos y *carpintería* de software. La parte de métodos numéricos busca ilustrar el paso entre la formulación de una pregunta sobre la realidad en términos matemáticos y su descripción numérica/computacional, para mostrar posibles formas de escribir esa pregunta con software. La parte de carpintería busca presentar algunas técnicas y prácticas necesarias (no suficientes) para poder obtener resultados computacionales reproducibles.

II Objetivos

El objetivo principal del curso es presentar algoritmos y técnicas básicas para:

- resolver ecuaciones diferenciales ordinarias, parciales y estocásticas,
- analizar y describir datos con técnicas estadísticas basadas en métodos Monte Carlo,
- desarrollar esquemas reproducibles para el análisis de datos científicos.

III Competencias a desarrollar

Al finalizar el curso, se espera que el estudiante esté en capacidad de:

- manejar dos lenguajes de programación modernos de computación numérica: uno de alto nivel (Python) y otro de bajo nivel (C),
- implementar algoritmos sencillos para la resolución de ecuaciones diferenciales y para el análisis estadístico exploratorio de datos,
- desarrollar un esquema sencillo para obtener resultados computacionales reproducibles.

IV Contenido por semanas

Semana 1

- Temas: Unix. Consola. Editores de texto. Comandos básicos. Intérprete de Python. Variables. Aritmética. Listas. Diccionarios.
- Lecturas preparatorias:
- Talleres: Examen diagnóstico (sin nota).

Semana 2

- Temas: If/while/break/continue. Ciclos. Funciones. Arreglos (numpy). Lectura y escritura de archivos (numpy). Gráficas y visualización (matplotlib).
- Lecturas preparatorias:
- Talleres:

Semana 3

- Temas: Integrales y derivadas.
- Lecturas preparatorias:
- Talleres:

Semana 4

- Temas: Solución de ecuaciones lineales y no lineales.
- Lecturas preparatorias:
- Talleres:

Semana 5

- Temas: Autovalores y autovectores. Principal Component Analysis.
- Lecturas preparatorias:
- Talleres:

Semana 6

- Temas: Transformadas de Fourier
- Lecturas preparatorias:
- Talleres:

Semana 7

- Temas: Ecuaciones diferenciales ordinarias.
- Lecturas preparatorias:
- Talleres:

Semana 8

- Temas: Ecuaciones diferenciales parciales.
- Lecturas preparatorias:
- Talleres:

Semana 9

- Temas: C.
- Lecturas preparatorias:
- Talleres:

Semana 10

- Temas: C. Git. Github.
- Lecturas preparatorias:
- Talleres:

Semana 11

- Temas: Makefile. Unit Tests.
- Lecturas preparatorias:
- Talleres:

Semana 12

- Temas: Conceptos básicos de probabilidad y estadística. Procesos aleatorios.
- Lecturas preparatorias:
- Talleres:

Semana 13

- Temas: Métodos de Monte Carlo.
- Lecturas preparatorias:
- Talleres:

Semana 14

- Temas: Estimación de parámetros en estadística bayesiana.
- Lecturas preparatorias:
- Talleres:

Semana 15

- Temas: Ecuaciones diferenciales estocásticas.
- Lecturas preparatorias:
- Talleres:

V Metodología

Cada clase tendrá una corta presentación teórica (30 minutos aproximadamente) para pasar a practicar todos los conceptos directamente en la computadora/cluster a través de ejercicios de práctica (50 minutos aproximadamente).

VI Criterios de evaluación

El curso tendrá tres entregas de trabajos, cada una con un valor del (100/3) % de la nota definitiva. Los temas de las entregas serán los siguientes:

1. Ecuaciones diferenciales (ordinarias, parciales, estocásticas).
2. Machine Learning.
3. Cómputo en paralelo (OpenMP, MPI).

VII Bibliografía

Bibliografía principal:

- R. L. Burden, J. D. Faires. *Numerical analysis*, 2011. (Biblioteca General - 519.4 B862 2011)
- A. Tveito, H.P. Langtangem, B.F. Nielsen., *Elements of Scientific Computing*, 2010. (Biblioteca General, Recurso Electrónico 510.)
- O. Maimon and L. Rokach, *The Data Mining and Knowledge Discovery Handbook*, 2010. (Biblioteca General, Recurso Electrónico 006.312)
- M. Snir, *MPI : the complete reference*, 1996. (Biblioteca General, 004.35 M637)
- J. Sanders, E. Kandrot. *CUDA by example: an introduction to general-purpose GPU programming*, 2010. (Biblioteca General - 005.275 S152)

Bibliografía complementaria:

- D. Conway and J. M. White. *Machine learning for hackers*, 2012.
- S.Bird. *Natural Language Processing with Python*, 2009.
- Theano Development. *Deep Learning Tutorial* <http://deeplearning.net/tutorial/>