

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 13:50 y Vi, 15:30 a 16:50
HORARIO (LABORATORIO): Ju 6:30 - 7:50

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: Presentación del curso. Unix. Consola. Comandos básicos. Editores de texto. Intérprete de Python. Variables. Aritmética. Listas. Diccionarios.
- Lecturas preparatorias:
- Talleres: Prueba de 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

Las *habilidades computacionales* se desarrollan trabajando activamente. Por esto en las sesiones magistrales, luego de presentar un resumen de los conceptos teóricos, se hará énfasis en la práctica y experimentación. Para que esto funcione es necesario que los estudiantes estudien el tema correspondiente **antes de cada clase**.

El programa del curso tiene dos componentes diferenciadas. La parte de métodos numéricos y la parte de *carpintería* de software. La parte de métodos numéricos ilustra como pasar de alguna pregunta sobre la realidad a un formalismo matemático general, luego a una descripción numérica y de ahí a su implementación en términos de software. La parte de carpintería busca presentar algunas condiciones mínimas para que los resultados del último paso sea confiables y permitan la producción de resultados reproducibles. En el *Laboratorio de Métodos Computacionales* habrá más tiempo para practicar lo visto en la clase magistral, hacer ejercicios y aclarar dudas.

La Magistral y el Laboratorio cuentan con repositorios en GitHub:

<https://github.com/ComputoCienciasUniandes/MetodosComputacionales>.

<https://github.com/ComputoCienciasUniandes/MetodosComputacionalesLaboratorio>.

VI Criterios de evaluación

La Magistral y el Laboratorio se califican por separado. Habrá exámenes escritos, talleres y ejercicios cortos.

Al comienzo del semestre se hará un examen (corto, sin nota) para diagnosticar el conocimiento general que ya tienen los estudiantes sobre los temas del curso.

En la Magistral habrá cinco talleres para resolver. Cada taller tiene un valor del 20 % de la nota definitiva. Hay un bono de 10 % que consiste en una prueba escrita sobre los temas vistos durante el semestre.

En el Laboratorio habrá un total de cinco ejercicios cortos para desarrollar y entregar en clase. Cada ejercicio corresponde a un 20 % de la nota final del Laboratorio.

De acuerdo a la nota definitiva en el Laboratorio habrá **otro bono** en la nota definitiva de la Magistral. Siendo x la nota de Laboratorio, el bono correspondiente se calcula así: $4,0 < x \leq 4,4 \rightarrow 0,1$, $4,4 < x \leq 4,8 \rightarrow 0,2$, $4,8 < x \leq 5,0 \rightarrow 0,4$.

Todos los exámenes, talleres y ejercicios serán **individuales**. Si en las entregas individuales se detecta que el trabajo no fue individual (esto incluye colaboración con personas no inscritas en el curso), entonces la nota de todos los talleres quedará automáticamente **en cero** y se llevará el caso a comité disciplinario.

Todas las entregas de talleres y ejercicios se harán a través de SICUA. **No se aceptará ninguna tarea por fuera de esa plataforma**, a menos que ocurra una falla en los servidores de SICUA que afecte a **todos** los estudiantes del curso.

VII Bibliografía

Bibliografía principal:

- *Elements of Scientific Computing* Tveito A., Langtangen H.P., Nielsen B.F., Cai X. Springer. 2010.
- *A survey of Computational Physics* . R. H. Landau, M. J. Páez, C. C. Bordeianu. Princeton Univ. Press. 2006
- *Statistical Mechanics: Algorithms and Computations*. W. Krauth, Oxford Univ. Press.
- *Introduction to Computation and Programming Using Python*, Gutttag, J. V. The MIT Press. 2013.
- *The C programming language*. B. Kernighan & D. Ritchie, Second Edition, Prentice Hall.
- <http://software-carpentry.org/>
- <http://xkcd.com/>