

## MÉTODOS COMPUTACIONALES AVANZADOS

Jaime E. Forero Romero

NOMBRE DEL CURSO: Métodos Computacionales Avanzados

CÓDIGO DEL CURSO: FISI 3028 / FISI 4028

UNIDAD ACADÉMICA: Departamento de Física

PERIODO ACADÉMICO: 201610

HORARIO: Ma y Ju, 8:30 a 9:50

---

NOMBRE PROFESOR(A) PRINCIPAL: Jaime E. Forero Romero

CORREO ELECTRÓNICO: [je.forero@uniandes.edu.co](mailto:je.forero@uniandes.edu.co)

NOMBRE PROFESOR(A) COMPLEMENTARIO(A):

CORREO ELECTRÓNICO: [@uniandes.edu.co](mailto:@uniandes.edu.co)

---

### I Introducción

Los métodos computacionales fundamental el trabajo en todas las áreas técnicas y científicas, ya sean principalmente experimentales o teóricas. Esto se debe en gran parte a que la capacidad de utilizar computadoras de alto rendimiento ha disminuido en costo monetario y en complejidad.

El curso de Métodos Computacionales Avanzados presenta estas posibilidades computacionales a estudiantes de diferentes disciplinas científicas. Para esto se propone profundizar sus conocimientos en tres áreas: resolución de ecuaciones diferenciales, implementación de métodos de machine learning (i.e. algoritmos que aprenden de datos) y utilización de técnicas de cómputo masivamente paralelo.

Se asume que los estudiantes de este curso ya tienen conocimientos básicos en métodos computacionales equivalentes al nivel del curso Métodos Computacionales (FISI-2028).

### II Objetivos

Los objetivos principales del curso son:

- Presentar métodos para la resolución de ecuaciones diferenciales ordinarias, parciales y estocásticas.
- Demostrar el uso de métodos y librerías de Machine Learning.
- Mostrar diferentes arquitecturas, paradigmas, lenguajes y librerías para cómputo masivamente paralelo.
- Estudiar diferentes aplicaciones prácticas a problemas científicos y del contexto industrial de las metodologías computacionales modernas.

### III Competencias a desarrollar

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

- Manejar lenguajes modernos de computación numérica de bajo nivel (i.e. C/C++) y de alto nivel (i.e. Python/R/Julia).
- Tener un esquema para pre-procesar, analizar y generar reportes científicos y técnicos a partir de diversas fuentes de datos utilizando métodos computacionales.
- Desplegar programas en paralelo en un cluster, al igual que en sistemas de cómputo distribuidos tales como Amazon Web Services.
- Generar conocimiento a partir del modelamiento teórico y computacional de los conceptos vistos en clase.

## IV Contenido por semanas

**Semana 1.** Unix. Repositorios. Makefiles. C. Python. Norma IEEE para aritmética de punto flotante.

**Semana 2.** Integración de ecuaciones diferenciales ordinarias. Método de Euler (explícito e implícito), Métodos de Runge-Kutta, paso de tiempo adaptativo, leapfrog, integradores simpléticos.

**Semana 3.** Ecuaciones diferenciales parciales. Diferentes esquemas de solución. Mecánica de fluidos. Leyes de conservación hiperbólicas. Riemann Solvers. Método de Godunov.

**Semana 4.** Generadores de números aleatorios. Integración. Cadenas de Markov.

**Semana 5.** Ecuaciones diferenciales estocásticas. Convergencia. Estabilidad.

**Semana 6.** Termodinámica de sistemas simples. Estimación de parámetros. **Entrega Taller 1.**

**Semana 7.** Clustering: k-means, Jerárquico y Maximización del valor esperado.

**Semana 8.** Aprendizaje Supervisado. Árboles de Decisión, Clasificación, Ranking Regresión, Redes Neuronales.

**Semana 9.** Reducción de Dimensionalidad y Predicción Estructurada: Principal Component Analysis (PCA), Multi-dimensional Scaling (MDS), Linear Discriminant Analysis (LDA); Modelos Gráficos, Redes Complejas y Análisis de datos Topológico.

**Semana 10. Receso.**

**Semana 11.** Procesamiento Natural de Lenguaje (Naïve Bayes) y Aprendizaje Profundo (Deep Learning).

**Semana 12.** Fundamentos de programación en paralelo. Taxonomía de arquitecturas para cómputo en paralelo. Máquinas para cómputo en paralelo en Uniandes. **Entrega Taller 2.**

**Semana 13.** Fundamentos de programación en paralelo: OpenMP

**Semana 14.** Fundamentos de programación en paralelo: MPI

**Semana 15.** Fundamentos de programación en paralelo: MPI

**Semana 16.** Cómo desplegar máquinas de cómputo en La Nube. Ejemplos de aplicación en Amazon Web Services y Docker. **Entrega Taller 3 al final de la primera semana de finales.**

## 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)

Bibliografia 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/>