

---

## Syllabus - Introducción a Métodos Numéricos

### Curso Gratuito

*Meses: Junio/Julio 2020 – Modalidad on-line*

El material pertinente del curso, incluidos este *syllabus* y los recursos relacionados con él, estarán disponibles en el repositorio del curso. Además, se utilizará un [repositorio en Github](#) para almacenar el material.

<b>Curso</b>	Introducción a Métodos Numéricos.
<b>Carga Horaria</b>	3 sesiones semanales de 80 minutos.
<b>Descripción</b>	Los métodos numéricos son herramientas esenciales para probar e implementar modelos matemáticos de diversa índole, además de permitir visualizar los resultados de forma acorde con los propósitos que se persigue (por ejemplo, una investigación). Este curso introducirá a sus participantes en el uso de las herramientas clásicas, basadas en resultados del Cálculo y el Álgebra, con el objetivo de que puedan desarrollar sus propios programas, atendiendo los aspectos prácticos para los usos que quieran darle. Además, se busca pulir sus habilidades y conocimientos en programación, específicamente en el manejo de Python 3 y sus librerías más frecuentadas, lo que involucra desarrollar la capacidad de auto-documentación y búsqueda de información relevante en Internet.
<b>Objetivos</b>	<p>Se espera que al final del curso se pueda:</p> <ul style="list-style-type: none"><li>▪ Operar y manipular eficazmente los objetos que Python permite definir.</li><li>▪ Aplicar reglas del Cálculo y Álgebra en programas computacionales.</li><li>▪ Buscar soluciones ya implementadas en librerías existentes y, en caso de no existir, proponer una solución propia.</li><li>▪ Utilizar las librerías Numpy y Scipy para el tratamiento de datos, y la librería Matplotlib para la visualización de los mismos.</li><li>▪ Definir clases y objetos en el lenguaje Python.</li><li>▪ Entender y aplicar resultados básicos del Álgebra Lineal, Cálculo Diferencial e Integral, Optimización y Probabilidades, con miras en una segunda parte del curso en que se profundizará en conceptos más avanzados.</li><li>▪ Abarcar un problema de manejo de datos, formulando preguntas de interés investigativo, proponiendo métodos de utilidad y reconociendo sus limitaciones.</li></ul>
<b>Audiencia:</b>	El curso es adecuado para cualquier persona que tenga interés en aprender aspectos prácticos de la programación y fundamentos matemáticos básicos. No hay restricciones con el área en el que se desempeñe o estudie el participante, puesto que el campo de aplicación es amplio, aunque en principio se desarrollará la clase entorno a problemas frecuentes en ciencias e ingeniería. Tampoco es requisito que se encuentre estudiando, mientras satisfaga de forma parcial los requisitos que se describen más adelante.
<b>Prerequisitos</b>	Se espera que los participantes tengan conocimientos básicos de algoritmos o entusiasmo por aprenderlos, experiencia de programación básica que le permita abstraer un algoritmo en pseudocódigo y cierta familiaridad con Cálculo y Álgebra (p. ej., sucesiones, funciones reales, nociones de límite, vectores y matrices). Si está interesado en tomar el curso pero no está seguro de tener los conocimientos necesarios, la clase consta de un breve espacio de repaso de los conceptos básicos necesarios y, si el caso lo amerita, se podrá proveer material extra y referencias para su comprensión.
<b>Modalidad</b>	El curso consta de sesiones tres veces a la semana de 80 minutos cada una, en las cuales se desarrollará teoría y práctica de forma simultánea. Se buscará problemas reales a los cuales aplicar los métodos vistos para reportar resultados de interés.

---

## Tópicos del curso

1. Python: Conceptos básicos.  
(0,5 semanas)
  - a) Definición de ambiente e instalación de librerías con Anaconda.
  - b) ¿Qué objetos existen?
  - c) ¿Cómo se opera con estos objetos?
  - d) Definición de funciones y su uso en algoritmos básicos.
2. Fundamentos de Cálculo. (1,5 semanas)
  - a) Sucesiones y funciones reales.
  - b) Curvas en el espacio.
  - c) Visualización de funciones.
  - d) Resolución de ecuaciones de una incógnita.
3. Fundamentos de Cálculo Diferencial e Integral.  
(2 semanas)
  - a) Concepto y métodos básicos de derivación.
  - b) Derivadas de orden superior y aproximación de funciones.
  - c) Optimización de funciones.
  - d) Interpolación de datos.
4. Fundamentos de Álgebra Lineal  
(1,5 semanas)
  - a) Sistemas de ecuaciones.
  - b) Descomposición de matrices.
  - c) Problemas de interés con matrices (regresión lineal, valores propios, autómatas, imágenes).
5. Probabilidades y Estadística. (1 semana)
  - a) Concepto y métodos básicos de integración.
  - b) Conceptos básicos de probabilidades.
  - c) Uso de librería Scipy en muestreos aleatorios.
  - d) Métodos Monte-Carlo.
  - e) Tratamiento estadístico de datos.
  - f) Correlaciones y clusterización.
6. Problemas avanzados del Cálculo. (1,5 semanas)
  - a) Ecuaciones diferenciales ordinarias.
  - b) Sistemas de ecuaciones diferenciales.
  - c) Método de diferencias finitas.
  - d) Método de Runge-Kutta.
  - e) Modelos caóticos.

## Bibliografía

- Kiusalaas, J. (2013). *Numerical Methods in Engineering with Python 3*. [Cambridge]
- Langtangen, H. (2016). *A Primer on Scientific Programming with Python*. [Springer]
- Marsden, J. & Tromba, A. (1996). *Vector Calculus (Sixth Edition)*. [Worth Publ Inc]
- Boyd, S & Vandenberghe, L. (2018). *Introduction to Applied Linear Algebra: Vectors, Matrices, and Least Squares*. [Cambridge]
- DeGroot, M. & Schervish, M. (2011). *Probability and Statistics (4th Edition)*. [Pearson]
- Zill, D. (2005). *First Course in Differential Equations*. [Brooks/Cole]