

**Trabajo Practico Sistemas de ecuaciones diferenciales y Estabilidad****Objetivos**

1. Familiarizarse con las herramientas computacionales de cálculo simbólico y numérico mediante la utilización de las librerías de Python `NumPy`, `SymPy`, `SciPy`, y `Matplotlib`, enfocándose en la resolución de ecuaciones diferenciales y la representación gráfica de sus soluciones.
2. Comprender y aplicar los métodos de resolución de sistemas de ecuaciones diferenciales homogéneos y no homogéneos, implementando algoritmos para la solución de estos sistemas mediante las técnicas de variación de parámetros y transformada de Laplace.
3. Analizar la estabilidad de los sistemas de ecuaciones diferenciales, clasificando los puntos de equilibrio y evaluando su estabilidad utilizando métodos como la función de Lyapunov y otros criterios establecidos.
4. Desarrollar habilidades en la programación y análisis matemático, utilizando Python para resolver ejercicios prácticos que incluyan la generación de código, representación gráfica de campos vectoriales y análisis de soluciones.
5. Promover la investigación y comparación entre métodos numéricos y simbólicos, desarrollando una visión crítica sobre cuándo es conveniente utilizar un enfoque sobre el otro en la resolución de problemas matemáticos complejos.
6. Fomentar el trabajo en equipo y la investigación colaborativa, donde cada grupo deberá presentar de manera clara y organizada tanto la investigación previa como la resolución de actividades prácticas, con el fin de consolidar el conocimiento de las herramientas y técnicas estudiadas.

**Presentación del trabajo practico****Consigna de Presentación**

Para la resolución de este trabajo práctico se requiere la utilización de cuadernos de Jupyter. A continuación, se detallan las instrucciones para la correcta presentación de las actividades:

**Formato de Presentación:****1. Investigación Previa:**

- Crear un cuaderno de Jupyter titulado GrupoX\_TPsisistemasInvest, donde X corresponde al número de tu grupo.
- En este cuaderno, se deberá desarrollar la investigación previa con los ejemplos solicitados. Puedes apoyarte en herramientas de Inteligencia Artificial si lo consideras necesario.

**2. Resolución de Actividades:**

- Crear un segundo cuaderno de Jupyter titulado GrupoX\_TPsisistemasActividades, donde se abordará la resolución de los ejercicios solicitados. En esta parte, es importante que no utilices la IA, ya que se espera que puedas comprender y aplicar los conceptos matemáticos y las herramientas de programación por ti mismo.

Cada cuaderno deberá incluir los nombres de los integrantes del grupo al inicio y estar bien organizado con títulos y explicaciones claras.

**Tutoriales para el uso de Jupyter**

Para quienes necesiten familiarizarse con el entorno de Jupyter, se proporcionan los siguientes tutoriales:

Tutorial de Anaconda: [https://youtu.be/rxZYRcd3Mcl?si=P\\_e8uaMP3FndK-Fy](https://youtu.be/rxZYRcd3Mcl?si=P_e8uaMP3FndK-Fy)

Tutorial de Jupyter: <https://youtu.be/uoWDS0JwqE4?si=uKGvixjY1RkudEXs>

## Investigación previa

## Librerías de Python en el Cálculo Numérico y Simbólico

### Parte 1: Introducción a las Librerías

1. Investiga la librería NumPy y explica cómo se realizan las siguientes operaciones con matrices: Multiplicación de matrices, Cálculo de la inversa de una matriz, Cálculo de valores propios y vectores propios de una matriz. Proporciona ejemplos de código en Python utilizando NumPy.
2. ¿Cómo utilizarías NumPy para resolver un sistema de ecuaciones lineales utilizando matrices? Explica el proceso y proporciona un ejemplo en código?
3. ¿Qué es SymPy y en qué áreas del cálculo diferencial se utiliza? Investiga cómo esta librería facilita la resolución simbólica de ecuaciones diferenciales y qué ventajas tiene sobre métodos puramente numéricos.
4. Describe brevemente las funciones principales de la librería SciPy para resolver ecuaciones diferenciales. En particular, investiga las funciones `odeint` y `solve_ivp`. ¿En qué situaciones es mejor utilizar una u otra?
5. ¿Cómo se utiliza Matplotlib para graficar campos vectoriales? Realiza una búsqueda sobre la función `quiver`. Explica qué parámetros necesita y cómo se puede representar un campo vectorial.

### Parte 2: Aplicaciones Numéricas y Simbólicas

1. Toma una ecuación diferencial de la guía de trabajos prácticos y resuélvela usando SymPy y SciPy respectivamente. Compara los enfoques simbólico y numérico. ¿En qué casos preferirías usar uno u otro método?
2. Investiga cómo realizar la Transformada de Laplace utilizando SymPy. Explica el uso de las funciones `laplace_transform` e `inverse_laplace_transform`. ¿En qué aplicaciones prácticas crees que esta herramienta es útil?
3. Explora la librería SymPy y describe cómo se construyen y manipulan matrices simbólicas. Proporciona ejemplos en los que las entradas de una matriz no sean números, sino variables simbólicas.
4. ¿Cómo se calcula el determinante y la inversa de una matriz simbólica en SymPy? Busca los comandos correspondientes y proporciona un ejemplo en el que uses una matriz simbólica.
5. ¿Qué ventajas crees que tiene SymPy sobre NumPy al trabajar con matrices simbólicas? Compara las capacidades de ambas librerías en este contexto.

### Parte 3: Cálculo de Derivadas, límites e integrales

1. Investiga cómo se calculan derivadas simbólicas utilizando SymPy. Utiliza la función `diff` para calcular derivadas simples y parciales. Proporciona un ejemplo para cada caso.
2. Explora cómo se calcula un límite simbólico usando SymPy. Investiga la función `limit` y proporciona un ejemplo de cómo se calcularía el límite de una función cuando una variable tiende a un valor específico.

- ¿Qué ventajas ofrece el cálculo simbólico de límites con SymPy en comparación con hacerlo manualmente? Reflexiona sobre cómo SymPy maneja el cálculo simbólico en límites complicados.
- Investiga algún caso en el que calcular el límite de una función simbólicamente sea más útil que evaluarlo numéricamente. Explica por qué el enfoque simbólico es necesario en ese caso.
- Investiga la función integrate de SymPy y explica cómo se utilizan para calcular integrales indefinidas y definidas. Proporciona un ejemplo de cada tipo de integral (indefinida y definida) resuelto simbólicamente.

#### Parte 4: Exploración Avanzada

- Investiga cómo SciPy puede utilizarse para resolver sistemas de ecuaciones diferenciales. Busca ejemplos en los que se resuelvan sistemas acoplados de ecuaciones utilizando solve\_ivp. Explica brevemente los pasos clave.
- Explora una visualización interactiva de funciones o campos vectoriales usando Plotly o Mayavi. Busca ejemplos de gráficos 3D y describe las herramientas necesarias para realizar estos gráficos en Python.

#### Parte 4: Reflexión Final

- Después de investigar y realizar ejemplos con NumPy, SciPy y SymPy, ¿cuál consideras que es la librería más poderosa para el trabajo matemático simbólico y numérico? Argumenta tu respuesta con ejemplos prácticos.

### Actividades de aplicación

#### Parte 1

- Realice un diagrama del algoritmo de resolución de los sistemas de ecuaciones:
  - Homogéneos
  - No Homogéneos aplicando Laplace.
  - No Homogéneos aplicando variación de parámetros.
- Realizar un esquema de la clasificación de los puntos de equilibrio y estabilidad asociada.

#### Parte 2

Utilizando los ejercicios de la guía de trabajos practico

- Desarrollar los ejercicios indicados en la siguiente tabla.
  - Cada ejercicio deberá estar resuelto aplicando el código correspondiente en Python.
  - Realizar la grafica de los campos vectoriales asociados y de las funciones solución.
  - Analizar la estabilidad.

	Grupos																			
Ejercicios	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
3	a	b	c	d	a	b	c	d	a	b	c	d	a	b	c	d	a	b	c	d
4	e	f	g	h	e	f	g	h	e	f	g	h	e	f	g	h	e	f	g	h
5	i	j	k	L	i	j	k	L	i	j	k	L	i	j	k	L	i	j	k	L
6	a	b	c	a	b	c	a	b	c	a	b	c	a	b	c	a	b	c	a	b

- 2) Sistemas no homogéneos con variación de parámetros (punto 7 de la guía). Debe generar el código necesario para resolver el sistema asignado de manera tal que se visualicen los pasos de resolución con este método, su solución y su grafica.

	Grupos																			
Ejercicios	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
7	a	b	c	d	e	f	g	h	a	b	c	d	e	f	g	h	a	b	c	d

- 3) Sistemas no homogéneos con Laplace (punto 8). Debe generar el código necesario para resolver el sistema asignado de manera tal que se visualicen los pasos de resolución con este método, su solución y su grafica.

	Grupos																			
Ejercicios	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
8	a	b	c	d	a	b	c	d	a	b	c	d	a	b	c	d	a	b	c	d

- 4) Dado el siguiente sistema de ecuaciones

$$\begin{pmatrix} x_1'(t) \\ x_2'(t) \end{pmatrix} = \underbrace{\begin{pmatrix} 0 & 1 \\ -1 & -k \end{pmatrix}}_A \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix}$$

Desarrolle el código necesario para analizar la estabilidad para los siguientes valores de k

- |                |                 |
|----------------|-----------------|
| a. $k > 2$     | e. $-2 < k < 0$ |
| b. $k = 2$     | f. $k = -2$     |
| c. $0 < k < 2$ | g. $k < -2$     |
| d. $k = 0$     |                 |

- 5) Desarrolle el código necesario para evaluar la estabilidad a partir de la función de Lyapunov, visualizando los cálculos necesarios, del punto 2 de Estabilidad. Realice la gráfica correspondiente.

	Grupos																			
Ejercicios	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
2	a	b	c	a	b	c	a	b	c	a	b	c	a	b	c	a	b	c	a	b