



Universidad de Sonora  
División de Ciencias Exactas y Naturales  
Departamento de Física  
Física computacional  
**Álgebra Lineal con Python**  
**Actividad 7**  
Iveth R. Navarro

05 de marzo de 2021  
Puerto Peñasco, Sonora, México

## Resumen

La presente es una práctica de Física Computacional cuyo objetivo es la revisión en Python de los elementos básicos del Álgebra Lineal: vectores, matrices, y las operaciones básicas entre ellos.

## 1. Introducción

Como es de esperarse, en la física computacional (así como en muchos otros campos) el dominio básico de Python es una herramienta imprescindible. Esto implica, naturalmente, el conocimiento del manejo de los elementos básicos del álgebra lineal, como lo son los vectores, las matrices y las operaciones básicas entre ellos. Esta actividad tiene el objetivo de poner en práctica este conocimiento.

La biblioteca de Python [SciPy](#), contiene una serie de bibliotecas especiales para el trabajo, mencionamos algunas:

- Integración: [scipy.integrate](#)
- Interpolación: [scipy.interpolate](#)
- Optimización: [scipy.optimize](#)
- Estructuras espaciales: [scipy.spatial](#)
- Álgebra Lineal: [scipy.linalg](#)
- Otras bibliotecas especializadas.

A nosotros nos interesa la biblioteca para trabajar con vectores y matrices, por lo que utilizaremos la biblioteca [scipy.linalg](#).

Cuando [SciPy](#) se construye utilizando las bibliotecas optimizadas **ATLAS**, **LAPACK** y **BLAS**, tiene capacidades de álgebra lineal muy rápidas. Si excava lo suficientemente profundo, todas las bibliotecas **LAPACK** y **BLAS** sin procesar están disponibles para su uso para una velocidad aún mayor. Algunos ejemplos de rutinas que podemos emplear son: Encontrar el inverso, resolución de

un sistema de ecuaciones lineales, encontrar el determinante, normas informáticas, resolver problemas lineales de mínimos cuadrados y pseudo-inversos, inversa generalizada... Todas estas rutinas

de álgebra lineal esperan un objeto que se puede convertir en una matriz 2D. La salida de estas rutinas también es una matriz 2D.

## 2. Desarrollo

Esta actividad está dividida en 6 ejercicios:

### 1. Actividad 7.1

Esta actividad consistió en definir un par de matrices de  $2 \times 2$  y la matriz identidad, así como realizar una operación básica con ellas.

### 2. Actividad 7.2

Esta actividad consistió en una demostración relacionada con los polinomios característicos y el Teorema de Cayley-Hamilton.

### 3. Actividad 7.3

Este tercer ejercicio consistió en la resolución de un sistema de ecuaciones lineales utilizando el Método de Eliminación Gaussiana y la función [scipy.linalg.solve\(\)](#).

### 4. Actividad 7.4

Este ejercicio consistió en encontrar los eigenvalores y eigenvectores de cada matriz utilizando la función de [scipy.linalg.eig\(\)](#)

### 5. Actividad 7.5

Este ejercicio consistió en encontrar un polinomio interpolante  $p(x)$  que pase por 8 puntos dados.

También se solicitó sobreponer la función  $y = \sin(x)$  y contrastarla con el polinomio  $p(x)$ .

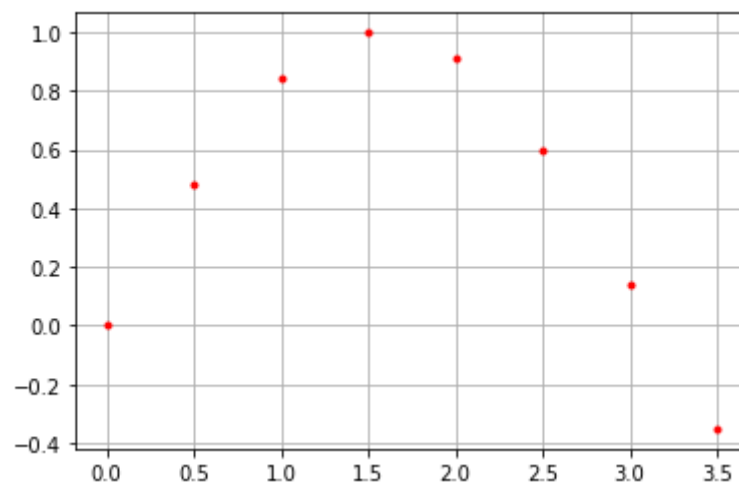
### 6. Actividad 7.6

Finalmente, este ejercicio necesitó un regreso al modelo de análisis de series de tiempo previamente reportado en la clase de Física Computacional.

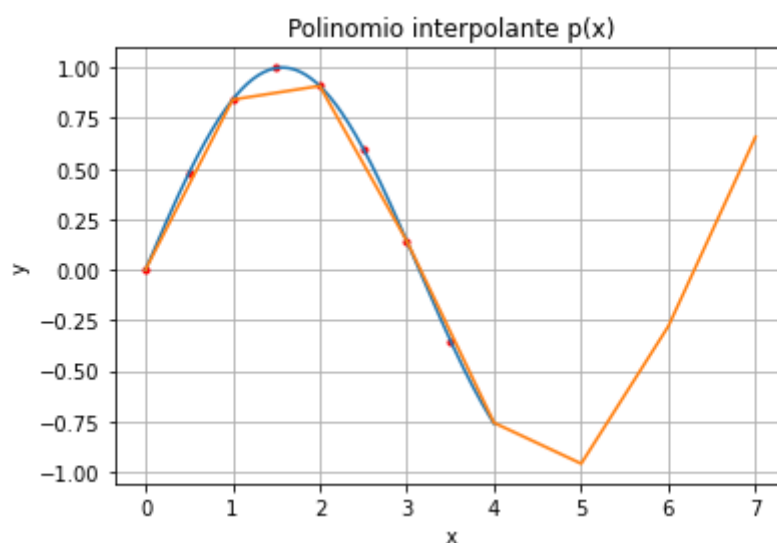
En dicho modelo, se analizó una descomposición de la serie de tiempo de las temperaturas ( $T_{\max}$  y  $T_{\min}$ ), y se realizó una regresión lineal sobre la serie de Tendencia de Temp ( $T = a_0 + a_1 t$ ). Luego, se encontró si la pendiente era positiva o negativa de la tendencia (signo de  $a_1$ ), es decir si las Temperaturas ( $T_{\max}$ ,  $T_{\min}$ ) estaban subiendo o bajando en el periodo analizado.

## 3. Resultados

Los resultados completos se encuentran en el documento .pynb anexo en este repositorio.



**Figura 1. Matriz de coordenadas**



**Figura 2. Polinomio de interpolación**

## 4. Conclusión

Esta actividad me parece de gran utilidad y me resultó muy interesante. Las actividades de la 7.1 a la 7.5 me resultaron sencillas, lo que me hizo sentir alivio, pues a veces en la programación con otros lenguajes da pie a múltiples errores dada su complejidad de expresión. La actividad 7.6, en lo particular, me resultó compleja, y tengo motivos para pensar que la desarrollé de manera errónea, pero el planteamiento me parece muy interesante, pues pasamos de practicar álgebra en Python a aplicarlo a otro proyecto, y la actividad logró ser atractiva.

Personalmente, puedo decir que la actividad me pareció entretenida, pues aunque la complejidad parecía baja en un principio, finalizó con un aumento de la misma; por ello, no se me ocurre una manera en que se pueda mejorar el planteamiento de esta actividad.

## 5. Bibliografía

- Comunidad SciPy. (2008). Linear Algebra (scipy.linalg) — SciPy v1.6.3 Reference Guide. SciPy.org. <https://docs.scipy.org/doc/scipy/reference/tutorial/linalg.html>