

Tarea 10

fl.gomez10 at uniandes.edu.co

11 de abril de 2019

Horario de atención: Principalmente de 2:00pm a 5:00pm en la oficina i-109. También se pueden enviar dudas al correo electrónico. Entregar antes de finalizar la clase.

Trabaje iniciando sesión en la máquina virtual en línea mybinder.org/¹.

1. Ejercicio 1 (50 puntos) Trabajo en Casa - Ajuste función polinómica

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3 from scipy.optimize import curve_fit
4
5 ### Cargar y visualizar datos
6
7 data = np.loadtxt("dos_picos_2.dat")
8 x = data[:,0]
9 y = data[:,1]
10 plt.scatter(x,y)
11
12
13 ### Ajuste de curvas
14
15 def f(x, a, b):
16     y = a + b * x
17     return y
18
19 curve_fit(f,x,y)
```

El fragmento anterior de código hace un ajuste de una recta al conjunto de datos `dos_picos_2.dat`², retornando el valor de los parámetros a y b .

Modifique el código para que realice el ajuste a un polinomio de grado no mayor que 10.(20 pts).

Muestre los parámetros óptimos (10pts).

¹<https://mybinder.org/v2/gh/ComputoCienciasUniandes/FISI2026-201910/master?urlpath=lab>

²https://github.com/ComputoCienciasUniandes/FISI2026-201910/raw/master/Talleres/Grupo_2/dos_picos_2.dat

Grafique la curva de mejor ajuste junto con los datos originales (20pts).

2. Ejercicio 2: (50 pts) Trabajo en Clase

20 pts

- Cargar en un array el archivo `munich_temperatures_average_with_bad_data.txt`
- Graficar temperatura vs tiempo
- usar `np.where()` para eliminar los datos que están locos.
- Volver a graficar
- Tomar una ventana de tiempo desde 2010 hasta el último dato
- graficar
- Hacer un ajuste a la función

$$f(x) = a \cos(bx - c) + d \quad (1)$$

30 pts