

ANÁLISIS NUMÉRICO I/ANÁLISIS NUMÉRICO – 2023

Trabajo de Laboratorio N^o 3

1. Programar una función en **python** que evalúe el polinomio interpolante p usando la forma de Lagrange. La función debe llamarse “**ilagrange**” y tener como entrada (x, y, z) donde $x, y \in \mathbb{R}^n$ son las coordenadas de los pares a interpolar (o sea $p(x_i) = y_i, i = 1, \dots, n$) y $z \in \mathbb{R}^m$ son valores para evaluar p . La salida debe ser $w \in \mathbb{R}^m$ tal que $w_j = p(z_j), j = 1, \dots, m$. La sintaxis a utilizar debe ser:

```
>>> w = ilagrange(x, y, z)
```

2. Realizar una función en **python** análoga a la del ejercicio 1 pero utilizando la forma de Newton del polinomio interpolante, calculando los coeficientes mediante *diferencias divididas*. La función debe llamarse “**inewton**”.
3. Considerar la función f tal que $f(x) = 1/x$. Utilizando el ejercicio anterior, graficar en una misma figura f y p que interpole $\{(i, f(i))\}_{i=1}^5$, usando para ambas los puntos equiespaciados $z_j = 24/25 + j/25, j = 1, \dots, 101$.
4. El polinomio interpolante se puede ver afectado por el conjunto de puntos elegidos. Considerar la función f tal que $f(x) = 1/(1 + 25x^2)$.

Graficar f, p_n y q_n en una misma figura usando 200 puntos igualmente espaciados en el intervalo $[-1, 1]$, donde:

- (a) p_n es el polinomio que interpola los pares $\{(x_i, f(x_i))\}_{i=1}^{n+1}$ con

$$x_i = \frac{2(i-1)}{n} - 1,$$

para $i = 1, \dots, n+1$.

- (b) q_n es el polinomio que interpola los pares $\{(x_i, f(x_i))\}_{i=0}^n$ con

$$x_i = \cos\left(\frac{2i+1}{2n+2}\pi\right),$$

para $i = 0, \dots, n$. Estos puntos son conocidos como nodos de Tchebychev.

Varíe n entre 1 y 15. Implementar la resolución de este ejercicio en el **script** “lab3ej4”. Al ejecutarlo debe mostrar 15 gráficos.

5. Leer cómo utilizar la función “**interp1d**” del paquete “**scipy.interpolate**” de **python** en la página <https://docs.scipy.org/doc/scipy/reference/generated/scipy.interpolate.interp1d.html>

El archivo **datos_aeroCBA.dat** contiene una matriz con los datos de la página <https://www.tutiempo.net/clima/ws-873440.html>

Cargar la matriz de datos en **python** usando **np.loadtxt** y extraer los datos existentes de temperatura media anual registrados en el Aeropuerto de Córdoba. Mediante un spline

cúbico estimar los valores faltantes y graficar. Observación: en algunos casos será necesario extrapolar.

Implementar la resolución de este ejercicio en el **script** “lab3ej5”, que realice el gráfico y devuelva los valores de temperatura media para TODOS los años entre 1957 y 2017.

6. Consideremos la siguiente tabla de datos:

x	-3	-2	-1	0	1	2	3
f	1	2	5	10	5	2	1

Interpolar la tabla utilizando los métodos de Lagrange, Newton y el de la función `interp1d`, luego graficar los 3 polinomios juntos. ¿Cuál polinomio parece más suave?

7. Escriba una función `rinterp(fun,x0,x1,x2,err,mit)` que encuentre un cero de la función `fun` de la siguiente forma. En cada paso, sea q_2 el polinomio interpolante cuadrático de los puntos $(x_{n-2}, f(x_{n-2}))$, $(x_{n-1}, f(x_{n-1}))$ y $(x_n, f(x_n))$. Elegir como x_{n+1} al cero de q_2 que esté más cerca de x_n . Comparar su performance con los métodos para encontrar raíces del laboratorio 2.