

ELEMENTOS DE CÁLCULO NUMÉRICO / CÁLCULO NUMÉRICO

Primer Cuatrimestre 2020

Séptimo ejercicio computacional

Martes 16/06/20 al Martes 23/06/20

Recuerde subir el archivo en formato ejercicioX_NOMBREPELLIDO.py

Recuerde enviar su código al hacer consultas

En este ejercicio el objetivo es implementar el método de diferencias divididas. Este método nos permitirá interpolrar un conjunto de N puntos (x, y) . La idea es encontrar coeficientes c tales que $Ac = y$, donde A es una matriz que depende de los valores de x , y c es un vector con coeficientes c_i de los polinomios $p_i(x)$, de forma que nuestra función interpoladora será $P(x) = \sum_{i=1}^N c_i p_i(x)$, $p_{i>0}(x) = \prod_{q=1}^i (x - x_{q-1})$, $p_0 = 1$ Para esto:

A Construya una función `matriz_dif(x)` que construya la matriz A (revisar la teórica!)

```
def matriz_dif(x):
    A = np.zeros((len(x),len(x)))
    A[:,0] = 1
    for q in range(len(x)):
        for r in range(1,q+1):
            A[q,r] = ## COMPLETAR ##
    return(A)
```

B Una función `newton_dif(x,y,max_iter,tol)` que reciba x , y , y parámetros para la función `jacobi` del ejercicio 6, y calcule los coeficientes del polinomio interpolador:

```
def newton_dif(x,y,max_iter,tol):
    A = matriz_dif(x)
    b = ## COMPLETAR ##
    x0 = np.zeros(len(y))
    coef = jacobi(A,b,x0,max_iter,tol)
    return(coef)
```

C Una función `funcion_interpol(x,y,max_iter,tol)` que calcule los coeficientes y retorne una función que calcule los valores del polinomio interpolador:

```
def funcion_interpol(x,y,max_iter,tol):
    coef = newton_dif(x,y,max_iter,tol)
    def function(m):
        X = np.ones(len(coef))
        for i in range(1,len(coef)):
            X[i] = ## COMPLETAR ##
        return( ## COMPLETAR ##)
    return(function)
```

D Emplee la función construida para interpolar la función $\sin(x)$ entre 0 y π . Para esto:

- i Considere los valores `tol = 10**-3`, `max_iter = 10**3`.
- ii Genere un vector x de 10 valores equiespaciados entre 0 y π , y un vector y igual al $\sin(x)$.
- iii Compare el resultado de usar el polinomio interpolador con el valor correcto, graficando ambas curvas a la vez, junto a los puntos empleados

D Empleando el paquete `pandas`, importe los datos provistos.

```
import pandas as pd
datos = pd.read_csv('data_prob7.csv')
x = np.log2(np.array(datos['area_acres']))
y = np.array(datos['pop_log2'])
```

Estos datos representan el área de los radios censales de Boston, junto a su población estimada (en verdad, el logaritmo en base 2). Interpole estos datos usando las funciones construidas. ¿Es razonable el resultado en todo el rango de los datos?