

# Ejercicio interpolación de Newton.

## Curso de Física Computacional

M. en C. Gustavo Contreras Mayén

### 1. Problema

Los datos que se muestran en la siguiente tabla se obtuvieron de la función

$$f(x) = 4.8 \cos\left(\frac{\pi x}{20}\right)$$

Con ese conjunto de datos, interpola mediante el polinomio de Newton en  $x = 0, 0.5, 1.0, \dots, 8.0$  y compara los resultados con el valor "exacto" de los valores  $y_i = f(x_i)$

$x$	0.15	2.30	3.15	4.85	6.25	7.95
$y$	4.79867	4.49013	4.2243	3.47313	2.66674	1.51900

### 2. Código

Ocuparemos **numpy** para facilitar el manejo de los datos en arreglos.

#### 2.1. Cálculo de los coeficientes

```
1 def coeficientes(xDatos, yDatos):
2     m = len(xDatos)
3     a = yDatos.copy()
4
5     for k in range(1,m):
6         a[k:m] = (a[k:m] - a[k-1])/(xDatos[k:m] - xDatos[k-1])
7
8     return a
```

#### 2.2. Evaluación del Polinomio

Una vez obtenido el arreglo de diferencias divididas, corresponde evaluar el polinomio en cada punto.

```

1 def evaluaPoli(a,xDatos,x):
2     n = len(xDatos)-1
3
4     p = a[n]
5
6     for k in range(1,n+1):
7         p = a[n-k] + (x - xDatos[n-k])*p
8
9     return p

```

### 2.3. Datos iniciales

Se necesitan los arreglos con los valores tanto de  $x$ , como de  $y$ , luego se indican los valores en donde se requiere evaluar y se muestra en pantalla, los resultados.

```

1 resultados = open('DatosNewton.dat','w')
2
3 xDatos = array([0.15,2.30,3.15,4.85,6.25,7.95])
4 yDatos = array([4.79867,4.49013,4.2243,3.47313,2.66674,1.51909])
5
6 a = coeficientes(xDatos, yDatos)
7
8 print 'x      yInterpol  yExacta'
9 print '_____'
10
11
12 for x in arange(0.0,8.1,0.5):
13     y = evaluaPoli(a,xDatos,x)
14     yExacta = 4.8 * cos(pi*x/20.0)
15     print '%3.1f %9.5f %9.5f' % (x, y, yExacta)
16     resultados.write('%3.1f %9.5f %9.5f \n' % (x, y, yExacta))
17
18 resultados.close()

```