UNIVERSIDAD DE CONCEPCION
FACULTAD DE CIENCIAS
FISICAS Y MATEMATICAS
DEPARTAMENTO DE INGENIERIA MATEMATICA

Laboratorio 4: Interpolación.

Cálculo Numérico 521230/525240

Recordemos que, dados los $n+1$ puntos $(x_0, y_0), (x_1, y_1), \ldots, (x_n, y_n)$, donde $x_0 < x_1 < \ldots < n_n$ y $x_i \neq x_j$ si $i \neq j$, el problema de **interpolación polinomial** consiste en determinar el **único** polinomio p, de grado menor o igual que n, que satisface $p(x_i) = y_i$ para cada i entre 0 y n.
Por otro lado una **spline cúbica** que interpola los puntos dados es una función $s \in C^2([x_0, x_n])$ tal que

$$s(x) = \begin{cases} q_0(x), & \text{si } x \in [x_0, x_1], \\ q_1(x), & \text{si } x \in [x_1, x_2], \\ & \vdots \\ q_{n-1}(x), & \text{si } x \in [x_{n-1}, x_n] \end{cases}$$

siendo $q_0, \ldots, q_{n-1}$, polinomios de grado menor o igual que 3 que satisfacen $q_0(x_0) = y_0$, $q_{r-1}(x_r) = q_r(x_r) = y_r$ para todo $r = 1, \ldots, n-1$, y $q_{n-1}(x_n) = y_n$.
En MATLAB podemos usar comandos para determinar p y s, y también para luego evaluarlos en valores de interés. Denotemos por `x` al vector cuyas componentes son $x_0, x_1, \ldots, x_n$, por `y` al vector con componentes $y_0, y_1, \ldots, y_n$ y por `z` a algún vector de componentes $z_0, z_1, \ldots, z_m$, en el que deseamos evaluar a p y a s. Entonces:

- El comando `polyfit` permite determinar p. Recibe como entradas los vectores `x, y` y el número entero no negativo `n` y retorna un vector con los coeficientes de p, ordenados de manera decreciente según la potencia de x que cada uno de ellos multiplica.

- El comando `polyval` permite evaluar p. Recibe como entrada un vector tal como el que devuelve `polyval` y el vector `z`, y retorna un vector con los valores de p evaluado en las componentes de `z`.

- El comando `spline` permite determinar una spline cúbica s a través de una estructura, la cual depende de qué valores se den como entrada.

  - Si simplemente se digita la instrucción `s=spline(x,y)`, la spline cúbica s que se calcula es la llamada *spline cúbica no-nodo* (del inglés *not-a-knot*), y ella cumple

    $$q_0'''(x_1) = q_1'''(x_1) \qquad \text{y} \qquad q_{n-2}'''(x_{n-1}) = q_{n-1}'''(x_{n-1}),$$

    es decir, que la tercera derivada es continua en el segundo y en el penúltimo punto.

  - Si se modifica la segunda entrada y, en cambio, se digita la instrucción `s=spline(x,[a y b])`, entonces la spline cúbica s que se calcula es la llamada *spline cúbica completa*, la cual cumple que

    $$s'(x_0) = a \qquad \text{y} \qquad s'(x_n) = b,$$

    es decir, su primera derivada tiene valores dados en ambos extremos.

- El comando `ppval` permite evaluar s. Recibe como entrada la estructura devuelta por `spline` y el vector `z`, y retorna un vector con los valores de s evaluada en las componentes del vector `z`.

- El comando `csape` también permite determinar un spline cúbica s que cumple ciertas condiciones. Por ejemplo, la instrucción `s=csape(x,[a y b],[2 2])` calcula una spline cúbica que cumple

$$s''(x_0) = a \qquad \text{y} \qquad s''(x_n) = b,$$

  es decir, su segunda derivada tiene valores dados en ambos extremos. Luego, en particular, si `a` y `b` son iguales a cero, la spline cúbica resultante es la *natural*.

- El comando `fnval` funciona exactamente como `ppval`, sólo que para splines que han sido calculadas con la función `csape`. Recibe como entrada la estructura calculada con `csape` y el vector `z`, y retorna un vector con los valores de s evaluada en las componentes del vector `z`.

Ejercicio 1 (ejercicio guiado por el/la ayudante): Escriba las siguientes líneas y guárdelas en un archivo llamado `polyinterp.m`:

```
function v = polyinterp(x,y,u)
% polyinterp(x,y,u) retorna los valores en las entradas
% de u del polinomio de interpolacion que pasa por los
% puntos cuyas coordenadas son las entradas de x e y
% El polinomio fue calculado usando los polinomios de Lagrange
n = length(x)-1
v = zeros(size(u));
for k = 0:n
  w = ones(size(u));
  for i = 0:n
        if i ~= k
              % En Matlab los indices comienzan en 1
              w = (u-x(i+1))./(x(k+1)-x(i+1)) .*w;
        end
  end
  v = v + w*y(k+1);
end
end
```

1. Considere la siguiente tabla que contiene los valores de una función f en los puntos dados:

x	-2	-1	0	1	2
$f(x)$	-25	-4	-1	8	47

Tabla 1: Datos función f.

2. Use la función `polyinterp` para evaluar el polinomio de interpolación de f en los puntos del vector `X = -2:4/1000:2`.

3. Grafique la interpolación obtenida en el item anterior.

4. Use el comando `polyfit` para calcular los coeficientes del polinomio de grado 4 que pasa por los puntos de la Tabla 1.

5. Use el comando `polyval` para evaluar el polinomio definido en el punto anterior en los puntos del vector `X = -2:4/1000:2`.

6. Grafique los datos obtenidos en el punto anterior.
7. Compare en el mismo gráfico lo obtenido en los items 3 y 6, junto a los puntos de la Tabla 1. ¿Se obtiene el mismo resultado? ¿A qué se debe esto?

Ejercicio 2 (ejercicio guiado por el/la ayudante): Considere la función $f(x) = \ln(x)$.

1. Determine, mediante el comando `polyfit` de MATLAB, un polinomio que interpole a f en los puntos $(1, \ln(1)), (2, \ln(2))$ y $(3, \ln(3))$. Escoja el grado de este polinomio de modo que su existencia y unicidad estén aseguradas.
2. Grafique, en una misma figura, la función f y el polinomio calculado, evaluados en 200 puntos entre 1 y 3. Recuerde que puede usar el comando `polyval` para evaluar al polinomio. ¿Cómo se reconoce en el gráfica que p interpola a f, así como los puntos de interpolación?
3. Grafique, en una misma figura, los valores de $|f(x_i) - p(x_i)|$ y

$$\frac{1}{3!} \max_{x\in[1,3]} \left|f'''(x)\right| |(x_i - 1)(x_i - 2)(x_i - 3)|,$$

con $x_i = 1 + ih$, $i = 0, 1, 2, \ldots, 100$ y $h = \dfrac{1}{50}$. ¿Se cumple que

$$|f(x_i) - p(x_i)| \leq \frac{1}{3!} \max_{x\in[1,3]} \left|f'''(x)\right| |(x_i - 1)(x_i - 2)(x_i - 3)|, \ i = 0, 1, 2, \ldots, 100?$$

Ejercicio 3 (ejercicio guiado por el/la ayudante): La siguiente tabla muestra los valores de *emitancia térmica* o *emisividad térmica* del tungsteno como función de la temperatura (en grados Kelvin).

Temperatura	300	400	500	600	700	800	900	1000	1100
Emisividad térmica	0.024	0.035	0.046	0.058	0.067	0.083	0.097	0.111	0.125

Nuestro objetivo es, con ayuda de la tabla anterior, determinar valores aproximados para la emisividad térmica del tungsteno a 550, 750 y 1080 grados Kelvin. Escriba el rutero `tungsteno.m` en MATLAB, en el que se:

1. determine el polinomio p que interpola a los pares ordenados en la tabla,
2. evalúe al polinomio en 550, 750 y 1080,
3. realice un gráfico del polinomio, evaluado en 1000 puntos entre 300 y 1100 y en el que también se incluya los pares en la tabla.

Observe que en los subintervalos $[300, 400]$ y $[1000, 1100]$ p tiene un comportamiento distinto al que tiene en $[400, 1000]$. Ya sabemos que si los puntos a interpolar son muchos y equidistantes, el polinomio de interpolación puede alejarse mucho de la función a interpolar, sobre todo en puntos cercanos a los extremos del intervalo. Esto se conoce como Fenómeno de Runge. Los valores 550 y 750 no son cercanos a los extremos del intervalo, pero 1080 sí lo es. En el rutero `tungsteno.m` realice ahora lo siguiente:

1. determine la spline cúbica natural s que interpola los datos en la tabla y evalúela en 550, 750 y 1080,
2. calcule $|p(550) - s(550)|$, $|p(750) - s(750)|$ y $|p(1080) - s(1080)|$.

Observe que la diferencia en 1080 es casi 8 veces mayor que en 550 y casi 25 veces mayor que en 750.

Ejercicio 4 (ejercicio para trabajo autónomo): Dado $x > 0$, se define la función **Gamma** por

$$\Gamma(x) = \int_0^{\infty} t^{x-1} e^{-t} \, \mathrm{d}t,$$

la cual tiene aplicaciones en estadística, probabilidad y resolución de ecuaciones diferenciales, entre otras. Se sabe que para todo $n \in \mathbb{N}$ se cumple

$$\Gamma(n) = (n-1)!,$$

y en consecuencia, un conjunto de puntos de esta función son

x	1	2	3	4	5
y	1	1	2	6	24

1. Calcule el polinomio de grado cuatro que interpola estos puntos. Dibuje este polinomio y también la función, usando la función `gamma` de Matlab.
2. Realice ahora la interpolación de estos puntos con la spline cúbica completa con derivadas nulas en los extremos, y dibújela junto con la función `gamma`.
3. ¿Cuál de los dos interpolantes es mejor?
4. ¿Cuál de los dos interpolantes es mejor en el intervalo $[1, 2]$?

Ejercicio 5 (ejercicio para trabajo autónomo): Con el objetivo de aproximar el contorno superior del pato que se muestra en la Figura 1 se midieron las coordenadas de 21 puntos $(x_i,\, y_i),\ i = 1, 2, \ldots, 21$ sobre él, los cuales se muestran en la Tabla 2

x	0.9	1.3	1.9	2.1	2.6	3	3.9	4.4	4.7	5
y	1.3	1.5	1.85	2.1	2.6	2.7	2.4	2.15	2.05	2.1
6	7	8	9.2	10.5	11.3	11.6	12	12.6	13	13.3
2.25	2.3	2.25	1.95	1.4	0.9	0.7	0.6	0.5	0.4	0.25

Tabla 2: Pares en contorno superior del pato

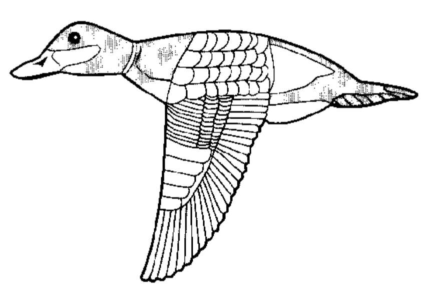

Figura 1: Se quiere interpolar borde superior de figura mostrada

1. Encuentre el polinomio que interpola a los datos en la tabla anterior, escoja el grado del mismo de forma que su existencia y unicidad estén garantizadas.

2. Grafique, en un mismo gráfico, los puntos de interpolación y el polinomio antes determinado, evaluado en 100 puntos entre 0.9 y 13.3. ¿Cree usted que este polinomio proporcione una buena aproximación a la curva que se quiere aproximar? Justifique.

3. Calcule la interpolante spline cúbica no-nodo que interpola los pares de valores dados. Grafique nuevamente los valores en la tabla y la spline cúbica obtenida, evaluada en 100 puntos entre 0.9 y 13.3. ¿Se obtiene con éste una mejor aproximación al contorno superior del pato en la Figura 1?