Máster en Materiales y Sistemas Sensores para Tecnologías Medioambientales (Erasmus Mundus)

PRÁCTICAS DE CÁLCULO NUMÉRICO

Damián Ginestar Peiró

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA
DEL DISEÑO

UNIVERSIDAD POLITÉCNICA DE VALENCIA

Práctica 1

Interpolación y ajuste de los datos de una tabla

El paquete MatLab dispone de distintas funciones para la interpolación y ajuste de los datos de una tabla de simple entrada. Revisaremos estas funciones y estudiaremos algunos ejemplos de su funcionamiento.

1.1. Polinomios en Matlab

Para representar un polinomio Matlab utiliza un vector con los coeficientes del mismo ordenados en potencias de x decrecientes. Así, el vector

```
1 -4 -3 -4 10
```

representa el polinomio

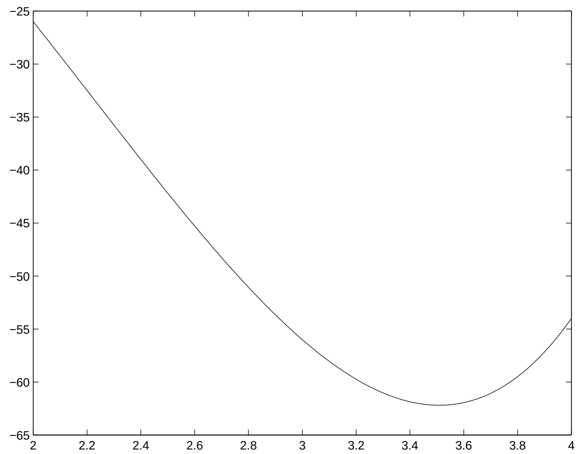
$$P(x) = x^4 - 4x^3 - 3x^2 - 4x + 10 \ .$$

Si p es un vector que contiene los coeficientes de un polinomio en potencias decrecientes, podremos evaluarlo para un valor concreto de la variable s_0 con el comando `polyval(p,s0)`. Esta función nos permite obter una gráfica del polinomio en un cierto intervalo. Por ejemplo, si introducimos en Matlab las instrucciones

```
p=[1   -4  -3  -4  10];
x=2:0.01:4;
```

```
y=polyval(p,x);
plot(x,y)
```

obtenemos la gráfica del polinomio $P(x)$ en el intervalo $[2,4]$, que es de la forma,



Las raíces de un polinomio se obtienen mediante la función `roots( )`. Así la instrucción

```
raices=roots(p)
```

da como resultado las raíces

```
4.7200
-0.8600+1.1743i
-0.8600-1.1743i
1.0000
```

Se puede reconstruir el polinomio a partir de sus raíces haciendo

```
p=poly(raices)
```

Supongamos que se tienen los polinomios

$$
\begin{aligned}
a(x) &= x^2 + 3x + 2 \ , \\
b(x) &= x^2 + 5x + 1 \ ,
\end{aligned}
$$

escribiremos los vectores de coeficientes comenzando por el coeficiente que acompaña al mayor exponente e iremos colocando los coeficientes que acompañen a las potencias de x de forma decreciente, es decir,

```
a = [1 3 2]
b = [1 5 1]
```

el producto de polinomios se obtiene con la función `conv( )`, así,

```
conv(a,b)
```

da como resultado

```
1 8 18 13 2
```

que corresponde al polinomio

$$x^4 + 8x^3 + 18x^2 + 13x + 2 \ .$$

1.2. Interpolación

Dada una tabla de datos de simple entrada

x_0	x_1	$\cdots$	$\cdots$	$\cdots$	x_n
y_0	y_1	$\cdots$	$\cdots$	$\cdots$	y_n

la función de Matlab `interp1( )` permite calcular interpolaciones para valores comprendidos entre los valores extremos recogidos en la tabla.

La sintaxis de la función es

```
yi=interp1(x,y,xi,'metodo')
```

donde `x` es el vector de las xs de la tabla cuyos valores deben estar ordenados de forma creciente o decreciente. `y` es el vector de las ys recogidas en la tabla. `xi` es el valor o valores para los que se quieren obtener las interpolaciones. `'metodo'` es la técnica utilizada para la interpolación. Este argumento es opcional y puede tomar los siguientes valores:

`'linear'` Se realiza una interpolación lineal.

`'spline'` Se realiza una interpolación construyendo un splin cúbico asociado a los datos de la tabla.

`'cubic'` Se realiza una interpolación cúbica entre los datos de la tabla. Para poder utilizar esta opción, los datos del vector `x` deben estar igualmente espaciados.

Veamos un ejemplo. En la siguinte tabla se muestra la población de Estados Unidos, en millones de personas desde 1900 hasta 1990.

año	1900	1910	1920	1930	1940
Pobl.	75.995	91.972	105.711	123.203	131.669
año	1950	1960	1970	1980	1990
Pobl.	150.697	179.323	203.212	226.505	249.633

Podemos escribir las instrucciones

```
tiem=1900:10:1990;
pob=[75.995   91.972 105.711 123.203 131.669 ...
     150.697 179.323 203.212 226.505 249.633];
```

con lo que tenemos dos vectores con los datos de la tabla. Hemos de resaltar que los tres puntos se usan en la definición de `pob` para poder cabiar de línea sin que Matlab interprete que ha finalizado la instrucción.

Podemos estimar la población de 1975 escribiendo

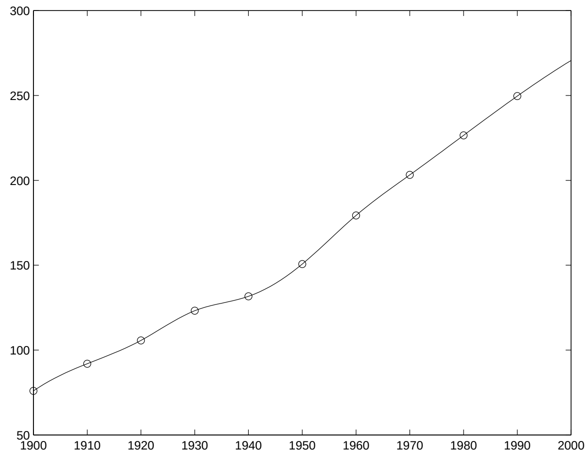
```
interp1(tiem,pob,1975)
```

obteniendo como resultado 214.8585 millones.

Por otra parte, podemos dibujar la evolución de la población mediante las instrucciones siguintes:

```
xi=1900:2000;
yi=interp1(tiem,pob,xi,'spline')
plot(tiem,pob,'o',xi,yi)
```

obteniendo la gráfica



1.3. Ajuste

La función `polyfit( )` nos proporciona un ajuste polinómico de los datos de una tabla. La sintaxis de la función es la siguiente:

```
p=polyfit(x,y,n)
```

donde `x` e `y` son vectores que contienen los datos de la tabla, `n` es el grado del polinomio que se quiere ajustar, y `p` es el vector de coeficientes del polinomio obtenido para el ajuste.

Veamos un ejemplo donde se puede utilizar esta función.

Se sabe que los datos de la siguiente tabla

x	1	2	3	4	5
y	4.52	4.09	3.70	3.35	3.03
x	6	7	8	9	10
y	2.74	2.48	2.25	2.03	1.84

corresponden a un decaimiento exponencial de la forma

$$y = a\exp(bx) \ .$$

Para estimar los valores de a y b a partir de los datos de la tabla procedemos del siguiente modo.

```
x=1:10;
y=[ 4.52,4.09,3.70,3.35,...
3.03,2.74,2.48,2.25,2.03,1.84];
y1=log(y);
p=polyfit(x,y1,1)
```

y obtenemos

```
-0.0999 1.6082
```

que son los parámetros de la recta

$$\ln(y) = bx + \ln(a) \ ,$$

con lo que tenemos $b = -0{,}0999$ y $a = 4{,}9938$.

1.4. Ejercicios

1. Calcular:

   a) Las raíces quintas de -1.

   b) Calcula $(x^4 + 3x^3 + 2x + 0{,}5)(x^6 + 3x + 0{,}3)$ en el punto $x = 3$.

   c) Las raíces reales de $x^6 - 3x^5 - x^4 + 7x^3 - 14x^2 + 10x - 12$.

2. Dada la tabla en donde se muestra la evolución del nitrógeno de mineralización

Tiempo (días)	Nmin (mg/kg)
7	9.466
14	8.211
27	15.590
41	17.615
55	20.215
83	21.734

   Obtener el tiempo necesario para que el nitrógeno de mineralización sea de 18.5 mg/Kg. Utilizad para ello un interpolante lineal y un spline, y comparar los resultados.

3. Se conoce que la relación existente entre el peso vivo de las larvas de la mariposa nocturna, W (g), y el oxígeno consumido por la larva, R en ml/h, es aproximadamente de la forma

$$R = bW^a \ .$$

Obtened los valores de a y b a partir de los datos de la siguiente tabla

W	R
0.017	0.154
0.087	0.296
0.174	0.363
1.11	0.531
1.74	2.23
4.09	3.58
5.45	3.52
5.96	2.40

4. Bajo ciertas condiciones se conoce que la evolución de una población con el tiempo se puede modelizar mediante una ecuación logística de la forma

$$P(t) = \frac{1000}{1 + Ce^{At}} \ .$$

Obtened C y A para la siguiente tabla de datos:

$P(t)$	200	400	650	850	950
t	0	1	2	3	4

5. Del calibrado de un motor se han obtenido los siguintes datos

Tiempo (milliseconds)	Distancia (cm)
0.00	0.0
1.40	1.0
2.37	2.0
3.30	3.0
3.37	4.0
4.11	5.0
5.42	6.0
5.71	7.0
6.39	8.0
7.26	9.0
7.82	10.0
8.67	11.0
9.12	12.0
9.66	13.0
10.70	14.0
11.23	15.0
11.25	16.0
12.47	17.0
12.79	18.0
13.20	19.0
14.12	20.0
14.83	21.0
15.83	22.0
16.04	23.0

Ajustad un polinomio de grado 4 que modelice la distancia en función del tiempo. Haced una gráfica con los datos de la tabla y el polinomio ajustado.

6. Es ampliamente aceptado que la tasa de crecimiento de bacterias respecto de tiempo es proporcional al número de bacterias presente en el cultivo. El factor de proporcionalidad, C, no es constante y está relacionado con unas constantes M y k, que dependen del tipo de bacterias, y de q, la cantidad de comida, del modo siguiente

$$\frac{1}{C} = \frac{k}{M}\frac{1}{q} + \frac{1}{M} \; .$$

Haz una estimación de los valores de estas constantes M y k a partir de los datos de la tabla siguiente

q	7	9	15	25	49	75	150
C	0.29	0.37	0.48	0.65	0.80	0.97	1.07

7. Interpolad la función

$$f(x) = \frac{1}{1+x^2} \ ,$$

de dos modos, primero un conjunto de puntos $x_n = -5 + n$, $y_n = f(x_n)$ con $n = 0, 1, 2, \ldots, 10$, y luego utilizando otro conjunto de puntos, $x_n = 5\cos(\pi n/10)$, $y_n = f(x_n)$ con $n = 0, 1, 2, \ldots, 10$. Utilizad un polinomio de grado 9 para la interpolación de la funcón. Realizad una gráfica con las dos interpolaciones y los valores de la función.