Práctica 6

INTERPOLACIÓN

6.1. Interpolación Polinómica

Datos de interpolación: $\{(x_k, f_k)\}_{k=0,1,\ldots,n}$
Conocemos los valores de una función, $f_k = f(x_k)$, en $n+1$ puntos distintos, x_k, de un intervalo [a,b]

Funciones interpolantes: Polinomios de grado menor o igual que n

$$p(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_n x^n$$

Problema de interpolación: Determinar los coeficientes $a_0, a_1, \ldots, a_n$ para que se cumplan las condiciones de interpolación:

$$p(x_k) = f_k, \quad k = 0, 1, \ldots, n$$

Método de Lagrange

Consiste en calcular previamente los polinomios $L_i(x)$, $i=0,1,\ldots,n$, llamados polinomio de Lagrange o funciones cardinales de Lagrange, que verifican:

$$L_i(x_i) = 1, \quad L_i(x_j) = 0, \quad i \neq j$$

Estos polinomios vienen dados por la expresión

$$L_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^{n} \frac{x - x_j}{x_i - x_j}$$

El polinomio de interpolación se escribe entonces en la forma

$$p(x) = f_0 L_0(x) + f_1 L_1(x) + \ldots + f_n L_n(x) = \sum_{i=0}^{n} f_i L_i(x)$$

EJEMPLO 1

Calcular el polinomio que interpola al conjunto de datos

{(-1,2),(2,8),(5,-3), (8,10)}

Definimos los puntos

In[1]:=
```
puntos := {{-1, 2}, {2, 8}, {5, -3}, {8, 10}}
```

Definimos los nodos

In[2]:=
```
x0 = -1; x1 = 2; x2 = 5; x3 = 8;
```
$x_0 = -1;\ x_1 = 2;\ x_2 = 5;\ x_3 = 8;$

In[3]:=

$f_0 = 2;\ f_1 = 8;\ f_2 = -3;\ f_3 = 10;$

Calculamos los polinomios de Lagrange

In[4]:=

$$L_0[x\_] = \frac{(x - x_1)\ (x - x_2)\ (x - x_3)}{(x_0 - x_1)\ (x_0 - x_2)\ (x_0 - x_3)}$$

Out[4]=

$$-\frac{1}{162}\ (-8 + x)\ (-5 + x)\ (-2 + x)$$

In[5]:=

$$L_1[x\_] = \frac{(x - x_0)\ (x - x_2)\ (x - x_3)}{(x_1 - x_0)\ (x_1 - x_2)\ (x_1 - x_3)}$$

Out[5]=

$$\frac{1}{54}\ (-8 + x)\ (-5 + x)\ (1 + x)$$

In[6]:=

$$L_2[x\_] = \frac{(x - x_0)\ (x - x_1)\ (x - x_3)}{(x_2 - x_0)\ (x_2 - x_1)\ (x_2 - x_3)}$$

Out[6]=

$$-\frac{1}{54}\ (-8 + x)\ (-2 + x)\ (1 + x)$$

In[7]:=

$$L_3[x\_] = \frac{(x - x_0)\ (x - x_1)\ (x - x_2)}{(x_3 - x_0)\ (x_3 - x_1)\ (x_3 - x_2)}$$

Out[7]=

$$\frac{1}{162}\ (-5 + x)\ (-2 + x)\ (1 + x)$$

Cálculo del polinomio de inteprolación

In[8]:=

$$\text{polisolu} = f_0\, L_0[x] + f_1\, L_1[x] + f_2\, L_2[x] + f_3\, L_3[x]$$

Out[8]=

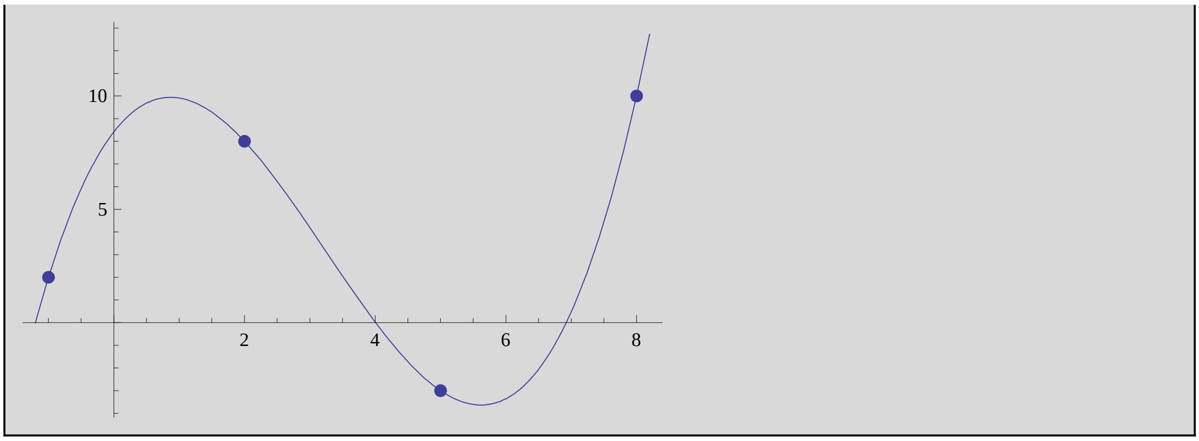
$$-\frac{1}{81}\ (-8 + x)\ (-5 + x)\ (-2 + x) + \frac{4}{27}\ (-8 + x)\ (-5 + x)\ (1 + x) + \frac{1}{18}\ (-8 + x)\ (-2 + x)\ (1 + x) + \frac{5}{81}\ (-5 + x)\ (-2 + x)\ (1 + x)$$

In[9]:=
```
Expand[polisolu]
```

Out[9]=
$$\frac{682}{81}+\frac{100\,x}{27}-\frac{133\,x^2}{54}+\frac{41\,x^3}{162}$$

Visualización de resultados

In[10]:=
```
g1 = ListPlot[puntos, PlotStyle → PointSize[0.02], DisplayFunction → Identity];
g2 = Plot[polisolu, {x, -1.2, 8.2}, DisplayFunction → Identity];
Show[g1, g2, DisplayFunction → $DisplayFunction]
```

Out[12]=


Método de Newton (Diferencias divididas)

Consiste en escribir el polinomio de interpolación en la forma:

$$p\,(x) = A_0 + A_1\,(x - x_0) + A_2\,(x - x_0\,(x - x_1) + ... + A_{n-1}\,(x - x_0\,(x - x_1)\,...\,(x - x_{n-1})$$

Los coeficientes A_k , que se denominan diferencias divididas de la función f en los puntos x_0, x_1, ... x_k, se denotan por A_k= f $[x_0, x_1, \ldots x_k]$ y se generan de forma recursiva mediante la fórmula

$$f[x_0,\,x_1,\,\ldots x_k] = \frac{f[x_1,\,x_2,\,\ldots x_k] - f[x_0,\,x_1,\,\ldots x_{k-1}]}{(x_k - x_0)},\quad k = 1,\,2,\,\ldots,\,n$$

partiendo de f[x_0]=f(x_0).

Con esta notación, el polinomio de interpolación puede escribirse como

$$p\,(x) = f[x_0] + f[x_0,\,x_1]\,(x - x_0) + f[x_0,\,x_1,\,x_2]\,(x - x_0)\,(x - x_1) + ... + \\ f[x_0,\,x_1,\,\ldots,\,x_n]\,(x - x_0)\,(x - x_1)\,...\,(x - x_{n-1}).$$

EJEMPLO 2

Calcular el polinomio que interpola al conjunto de datos

$$\{(-5,\,1),\,(-3,\,2),\,(2,\,10),\,(3,\,2),\,(6,\,0),\,(8,\,-3)\}$$

Entrada de datos

In[13]:=
```
puntos = {{-5, 1}, {-3, 2}, {2, 10}, {3, 2}, {6, 0}, {8, -3}};
```

Determinamos los nodos y los valores de la función

In[14]:=
```
n = Length[puntos] - 1;
For[i = 0, i ≤ n, i++, x_i = puntos[[i + 1, 1]]; f_i = puntos[[i + 1, 2]]]
```

Generamos la tabla de diferencias divididas (creamos una matriz rectangular de orden **(2n+2)×(n+2)** que llamanos **dif** y cuyos elementos son de la forma **d[i,j],** con **i=-1,0,…,2n** y **j=-1,0,…, n.**

In[16]:=
```
Clear[d];
dif = Array[d, {2 n + 2, n + 2}, -1];
```

A continuación rellenamos la matriz con espacios

In[18]:=
```
For[i = -1, i ≤ 2 n, i++, For[j = -1, j ≤ n, j++,
  d[i, j] = " "]]
```

Ahora rellenamos la primera y segunda filas (fila -1 y fila 0, respectivamente) con los datos de interpolación y ponemos de cabecera el texto “x_k” y “f_k” en la posición (-1,-1) y (-1,0) , respectivamente.

In[19]:=
```
d[-1, -1] = "x_k"; d[-1, 0] = "f_k"; For[i = 0, i ≤ n + 1, i++, d[2 i, -1] = x_i; d[2 i, 0] = f_i];
```

Visualizamos el resultado

In[20]:=
```
dif // MatrixForm
```

Out[20]//MatrixForm=

$$\begin{pmatrix} x_k & f_k \\ -5 & 1 \\ & \\ -3 & 2 \\ & \\ 2 & 10 \\ & \\ 3 & 2 \\ & \\ 6 & 0 \\ & \\ 8 & -3 \end{pmatrix}$$

Finalmente calculamos los restantes elementos de la tabla aplicando la ley de recurrencia

In[21]:=
```
For[j = 1, j ≤ n, j++, For[i = j, i ≤ 2 n - j, i = i + 2,
  d[i, j] = (d[i + 1, j - 1] - d[i - 1, j - 1]) / (d[i + j, -1] - d[i - j, -1])]]
```

Visualizamos el resultado final

In[22]:=
```
dif // MatrixForm
```

Out[22]//MatrixForm=

$$\begin{pmatrix}
x_k & f_k & & & & & \\
-5 & 1 & & & & & \\
 & & \frac{1}{2} & & & & \\
-3 & 2 & & \frac{11}{70} & & & \\
 & & \frac{8}{5} & & -\frac{123}{560} & & \\
2 & 10 & & -\frac{8}{5} & & \frac{9089}{166\,320} & \\
 & & -8 & & \frac{103}{270} & & -\frac{19\,897}{2\,162\,160} \\
3 & 2 & & \frac{11}{6} & & -\frac{193}{2970} & \\
 & & -\frac{2}{3} & & -\frac{1}{3} & & \\
6 & 0 & & -\frac{1}{6} & & & \\
 & & -\frac{3}{2} & & & & \\
8 & -3 & & & & &
\end{pmatrix}$$

Calculamos el polinomio de interpolación

In[23]:=

$$\texttt{polisolu} = \sum_{k=0}^{n} d[k,k] \left(\prod_{j=0}^{k-1} (x - x_j)\right)$$

Out[23]=

$$1 + \frac{5+x}{2} + \frac{11}{70}(3+x)(5+x) - \frac{123}{560}(-2+x)(3+x)(5+x) + \frac{9089(-3+x)(-2+x)(3+x)(5+x)}{166\,320} - \frac{19\,897(-6+x)(-3+x)(-2+x)(3+x)(5+x)}{2\,162\,160}$$

In[24]:=
```
Expand[polisolu]
```

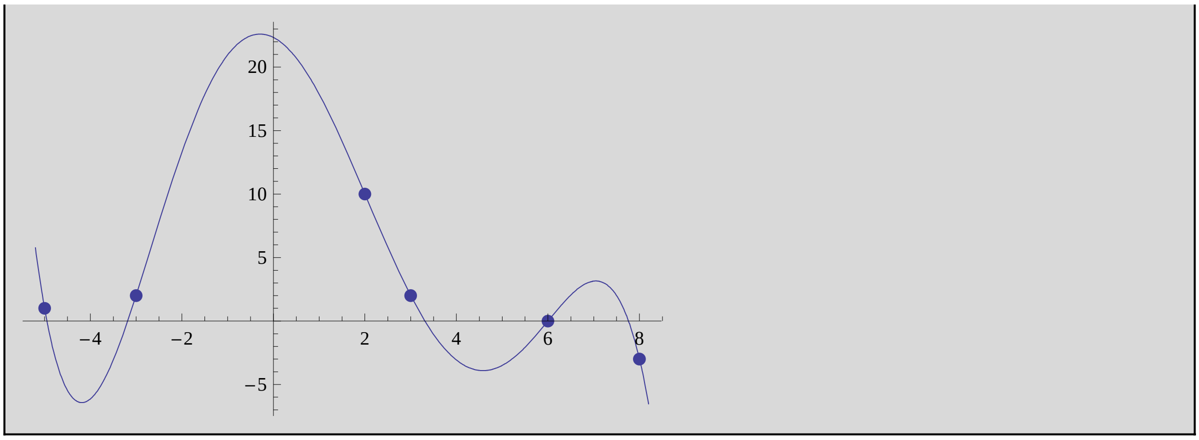
Out[24]=

$$\frac{67\,069}{3003} - \frac{109\,171\,x}{60\,060} - \frac{810\,709\,x^2}{270\,270} + \frac{615\,757\,x^3}{2\,162\,160} + \frac{2021\,x^4}{24\,570} - \frac{19\,897\,x^5}{2\,162\,160}$$

Visualizamos el resultado (en las variables xmin y xmax, guardamos el valor mínimo y máximo de los nodos)

In[25]:=
```
xmin = Min[Table[x_i, {i, 0, n}]];
xmax = Max[Table[x_i, {i, 0, n}]];
g1 = ListPlot[puntos, PlotStyle → PointSize[0.02`], DisplayFunction → Identity];
g2 = Plot[polisolu, {x, xmin - 0.2`, xmax + 0.2`}, DisplayFunction → Identity];
Show[g1, g2, DisplayFunction → $DisplayFunction]
```

Out[29]=



6.2. Interpolación polinómica con *Mathematica*

El programa Mathematica puede calcular directamente el polinomio de interpolación mediante la instrucción:

InterpolatingPolynomial[*puntos*, *variable*]

Calcula el polinomio de interpolación en la *variable* especificada para el conjunto de *puntos* dados en la forma:

$$\text{puntos}=\{\{x_0,\ f_0\},\ \{x_1,\ f_1\},\ \ldots,\ \{x_n,\ f_n\}\}.$$

EJEMPLO 3

Determinar el polinomio que interpola al conjunto de puntos

{ (0,0), (3,2), (5,2), (7,3), (9,5), (11,3) }

Introducimos los datos

In[30]:=
```
puntos := {{0, 0}, {3, 2}, {5, 2}, {7, 3}, {9, 5}, {11, 3}}
```

Calculamos el polinomio

In[31]:=
```
p[x_] = InterpolatingPolynomial[puntos, x]
```

Out[31]=

$$\left(\frac{2}{3}+\left(-\frac{2}{15}+\left(\frac{31}{840}+\left(-\frac{31}{7560}-\frac{1079\,(-9+x)}{1\,330\,560}\right)(-7+x)\right)(-5+x)\right)(-3+x)\right)x$$

In[32]:=
```
Expand[%]
```

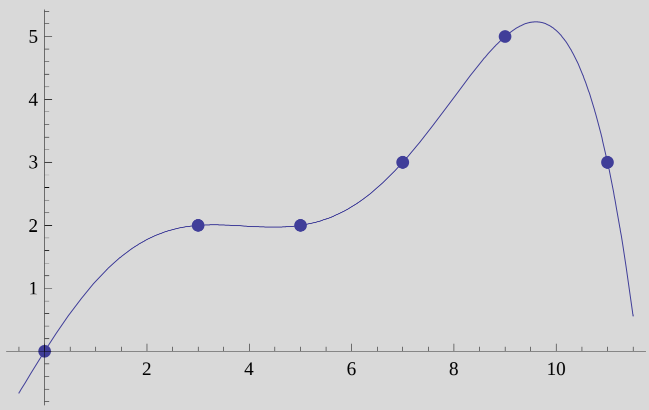
Out[32]=

$$\frac{569683\,x}{443520}-\frac{553\,x^2}{4752}-\frac{9133\,x^3}{133056}+\frac{73\,x^4}{4752}-\frac{1079\,x^5}{1330560}$$

Visualizamos resultados

In[33]:=
```
grafpuntos = ListPlot[puntos, PlotStyle → PointSize[0.02], DisplayFunction → Identity];
grafpoli = Plot[p[x], {x, -0.5, 11.5}, DisplayFunction → Identity];
Show[grafpuntos, grafpoli, DisplayFunction → $DisplayFunction]
```

Out[35]=



6.3. Interpolación con funciones splines

Sea Δ una partición del intervalo [a,b],

$$\Delta : a = x_0 < x_1 < ... < x_n = b.$$

Un spline es una función polinómica a trozos en cada uno de los intervalos $[x_i, x_{i+1}]$ de la partición.

Notaremos por $S_m^k(\Delta)$ al conjunto de funciones de clase k que son polinomios a trozos de grado m en cada uno de los intervalos de la partición:

$$S_n^k(\Delta) = \{S \in C_m^k([a, b]) : s|_{[x_i, x_{i+1}]} \in \mathbb{P}_m\},$$

donde $\mathbb{P}_m$ denota el conjunto de polinomios de grado a lo sumo m.

En lo sucesivo notaremos:
$\Delta=\{x_0, x_1, \ldots, x_n\}$ la partición dada por los nodos.
$h_i = x_{i+1} - x_i$, a la amplitud del intervalo $[x_i, x_{i+1}]$.
f_i a los valores de la función que queremos interpolar.
$m_i = (f_{i+1} - f_i)/h_i$, a la pendiente de la curva en el intervalo $[x_i, x_{i+1}]$.

Interpolación con splines cúbicos: $S_3^2(\Delta)$

Funciones interpolantes: Splines en $S_3^2(\Delta)$. Se trata de funciones de clase 2 definidas a trozos mediante polinomios de grado 3 en cada intervalo de la partición.

Problema de interpolación: Determinar un spline s $\in S_3^2(\Delta)$ tal que:

$$s(x_i) = f_i\,,\quad i = 0,\ 1,\ \ldots,\ n,$$

y necesitamos imponer además 2 condiciones adicionales en los puntos frontera x_0 y x_n. Por ejemplo si el espline satisface $s''(x_0) = 0$ y $s''(x_n) = 0$ se dice que es un spline cúbico natural.

EJEMPLO 4

Calcular el spline cúbico natural que interpola al conjunto de datos :

$$\{(-1,0), (3,1), (5,-3), (7,2)\}\ ,$$

En este caso el spline será una función de clase 2 definida a trozos, mediante 3 polinomios de tercer grado, que escribiremos en la forma

$$s(x) = \begin{cases} a_0 + b_0(x+1) + c_0(x+1)^2 + d_0(x+1)^3, & x \in [-1, 3) \\ a_1 + b_1(x-3) + c_1(x-3)^2 + d_1(x-3)^3, & x \in [3, 5) \\ a_2 + b_2(x-5) + c_2(x-5)^2 + d_2(x+5)^3, & x \in [5, 7] \end{cases}$$

con las condiciones adicionales: $s''(x_0) = s''(x_n) = 0$

Definimos los puntos

In[36]:=
```
puntos = {{-1, 0}, {3, 1}, {5, -3}, {7, 2}};
```

Definimos cada uno de los trozos que definen el spline

In[37]:=
```
s0[x_] := a0 + b0 (x + 1) + c0 (x + 1)^2 + d0 (x + 1)^3;
s1[x_] := a1 + b1 (x - 3) + c1 (x - 3)^2 + d1 (x - 3)^3;
s2[x_] := a2 + b2 (x - 5) + c2 (x - 5)^2 + d2 (x - 5)^3
```

Imponemos las condiciones de interpolación

In[40]:=
```
ecu1 = s0[-1] == 0
```

Out[40]= $a_0 == 0$

In[41]:=
```
ecu2 = s1[3] == 1
```

Out[41]= $a_1 == 1$

In[42]:=
```
ecu3 = s2[5] == -3
```
Out[42]= $a_2 == -3$

In[43]:=
```
ecu4 = s2[7] == 2
```
Out[43]= $a_2 + 2 b_2 + 4 c_2 + 8 d_2 == 2$

Imponemos las condiciones de continuidad (en nodos interiores)

In[44]:=
```
ecu5 = s0[3] == s1[3]
```
Out[44]= $a_0 + 4 b_0 + 16 c_0 + 64 d_0 == a_1$

In[45]:=
```
ecu6 = s1[5] == s2[5]
```
Out[45]= $a_1 + 2 b_1 + 4 c_1 + 8 d_1 == a_2$

Imponemos las condiciones de derivabilidad: clase 1 (en nodos interiores)

In[46]:=
```
ecu7 = s0'[3] == s1'[3]
```
Out[46]= $b_0 + 8 c_0 + 48 d_0 == b_1$

In[47]:=
```
ecu8 = s1'[5] == s2'[5]
```
Out[47]= $b_1 + 4 c_1 + 12 d_1 == b_2$

Imponemos las condiciones de derivabilidad: clase 2 (en nodos interiores)

In[48]:=
```
ecu9 = s0''[3] == s1''[3]
```
Out[48]= $2 c_0 + 24 d_0 == 2 c_1$

In[49]:=
```
ecu10 = s1''[5] == s2''[5]
```
Out[49]= $2 c_1 + 12 d_1 == 2 c_2$

Imponemos la condición en la frontera s''(-1)=s''(7)=0

In[50]:=
```
ecu11 = s0''[-1] == 0
```

Out[50]= $2 c_0 == 0$

In[51]:=
```
ecu12 = s2''[7] == 0
```

Out[51]= $2 c_2 + 12 d_2 == 0$

Resolvemos el sistema formado por todas las ecuaciones

In[52]:=
```
coeficientes =
 Solve[{ecu1, ecu2, ecu3, ecu4, ecu5, ecu6, ecu7, ecu8, ecu9, ecu10, ecu11, ecu12}]
```

Out[52]= $\left\{\left\{a_0 \to 0,\ a_1 \to 1,\ a_2 \to -3,\ c_0 \to 0,\ c_1 \to -\frac{81}{92},\ d_0 \to -\frac{27}{368},\ b_0 \to \frac{131}{92},\ b_1 \to -\frac{193}{92},\ c_2 \to \frac{351}{184},\ d_1 \to \frac{171}{368},\ b_2 \to -\frac{1}{23},\ d_2 \to -\frac{117}{368}\right\}\right\}$

Sustituimos los coeficientes en las expresiones de $s_0(x)$, $s_1(x)$ y $s_2(x)$

In[53]:=
```
sp0[x_] = s0[x] /. coeficientes[[1]]
```

Out[53]= $\frac{131 (1 + x)}{92} - \frac{27}{368} (1 + x)^3$

In[54]:=
```
sp1[x_] = s1[x] /. coeficientes[[1]]
```

Out[54]= $1 - \frac{193}{92} (-3 + x) - \frac{81}{92} (-3 + x)^2 + \frac{171}{368} (-3 + x)^3$

In[55]:=
```
sp2[x_] = s2[x] /. coeficientes[[1]]
```

Out[55]= $-3 + \frac{5 - x}{23} + \frac{351}{184} (-5 + x)^2 - \frac{117}{368} (-5 + x)^3$

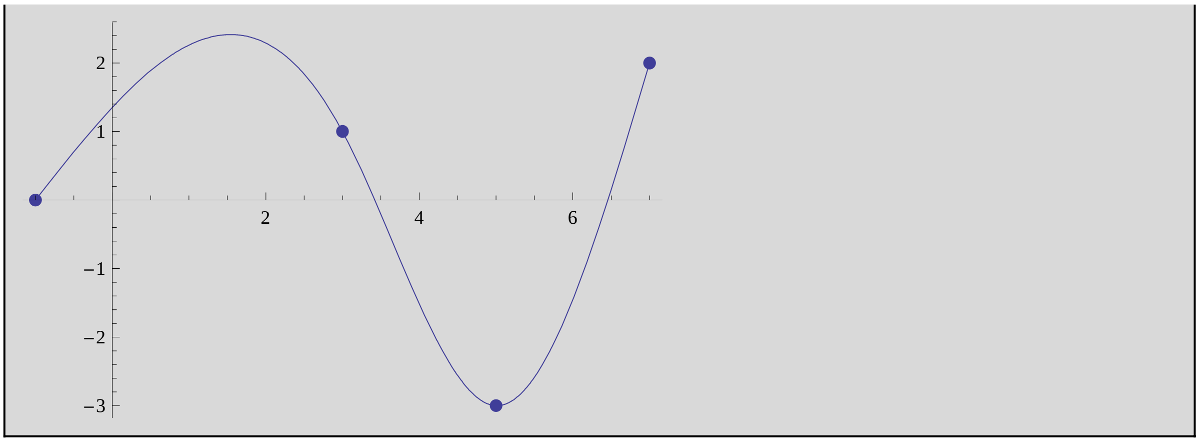
Nuestra función spline vendrá dada por

In[56]:=
```
s[x_] := Which[-1 ≤ x < 3, sp0[x], 3 ≤ x < 5, sp1[x], 5 ≤ x ≤ 7, sp2[x]]
```

Visualizamos los resultados

In[57]:=
```
grafpuntos =
  ListPlot[puntos, PlotStyle → PointSize[0.02`], DisplayFunction → Identity];
grafspline = Plot[s[x], {x, -1, 7}, DisplayFunction → Identity];
Show[grafpuntos, grafspline, DisplayFunction → $DisplayFunction]
```

Out[59]=



Ejercicios propuestos

EJERCICIO 1.- Calcular usando el método de Newton (diferencias divididas) el polinomio p(x) que interpola a la función f(x)=ln(x) en los puntos que resulta de dividir [1,4] en 6 partes iguales.

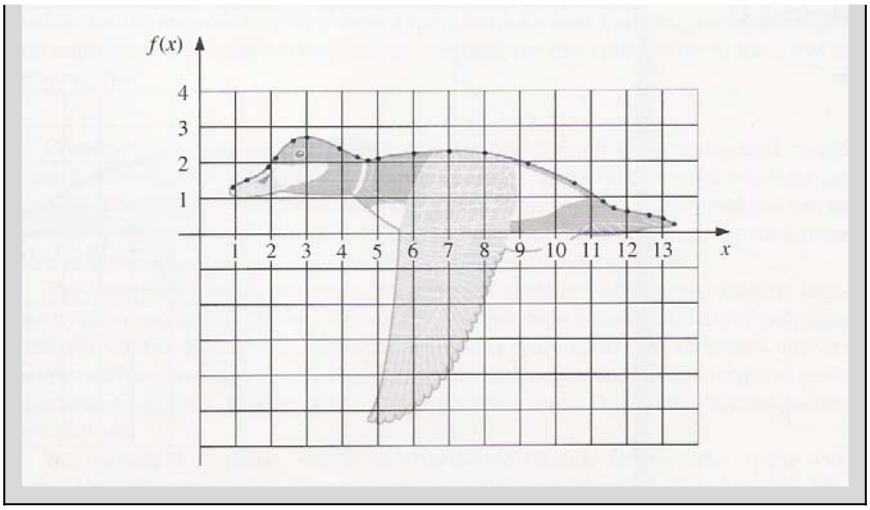
EJERCICIO 2.- Los datos correspondientes al censo de una población (en miles de habitantes) se recogen en la siguiente tabla:

Año	1950	1960	1970	1980	1990	2000
Número habitantes	123.5	131.2	150.7	141.3	203.2	240.5

a) Utilizar interpolación polinómica para estimar el número de habitantes en el año 1965.

b) ¿Cuál será la población estimada para el año 2020?. Comentar el resultado obtenido.

EJERCICIO 3.- Aproximar el perfil superior del pato de la imagen interpolando en los 21 puntos señalados mediante:



a) Un polinomio

b) Un spline cúbico natural

c) ¿Qué función proporciona un resultado más satisfactorio?