

CAPÍTULO 18

Interpolación

Con frecuencia se encontrará con que tiene que estimar valores intermedios entre datos definidos por puntos. El método más común que se usa para este propósito es la interpolación polinomial. Recuerde que la fórmula general para un polinomio de n -ésimo grado es

$$f(x) = a_0 + a_1x + a_2x^2 + \cdots + a_nx^n \quad (18.1)$$

Dados $n + 1$ puntos, hay uno y sólo un polinomio de grado* n que pasa a través de todos los puntos. Por ejemplo, hay sólo una línea recta (es decir, un polinomio de primer grado) que une dos puntos (figura 18.1a). De manera similar, únicamente una parábola une un conjunto de tres puntos (figura 18.1b). La *interpolación polinomial* consiste en determinar el polinomio único de n -ésimo grado que se ajuste a $n + 1$ puntos. Este polinomio, entonces, proporciona una fórmula para calcular valores intermedios.

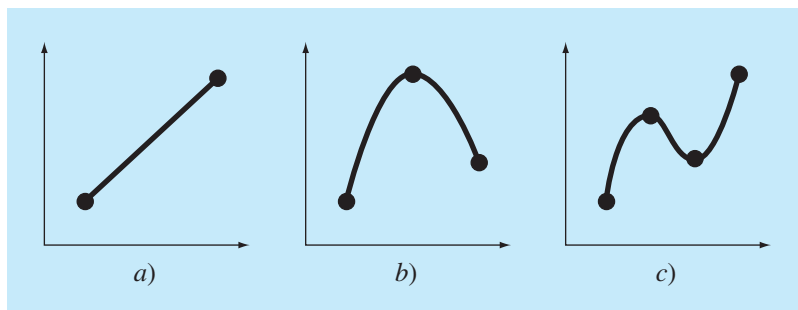
Aunque hay uno y sólo un polinomio de n -ésimo grado que se ajusta a $n + 1$ puntos, existe una gran variedad de formas matemáticas en las cuales puede expresarse este polinomio. En este capítulo describiremos dos alternativas que son muy adecuadas para implementarse en computadora: los polinomios de Newton y de Lagrange.

18.1 INTERPOLACIÓN POLINOMIAL DE NEWTON EN DIFERENCIAS DIVIDIDAS

Como se dijo antes, existe una gran variedad de formas alternativas para expresar una interpolación polinomial. El *polinomio de interpolación de Newton en diferencias di-*

FIGURA 18.1

Ejemplos de interpolación polinomial: a) de primer grado (lineal) que une dos puntos, b) de segundo grado (cuadrática o parabólica) que une tres puntos y c) de tercer grado (cúbica) que une cuatro puntos.



* De hecho se puede probar que dados $n + 1$ puntos, con abscisas distintas entre sí, existe uno y sólo un polinomio de grado a lo más n que pasa por estos puntos.

vididas es una de las formas más populares y útiles. Antes de presentar la ecuación general, estudiaremos las versiones de primero y segundo grados por su sencilla interpretación visual.

18.1.1 Interpolación lineal

La forma más simple de interpolación consiste en unir dos puntos con una línea recta. Dicha técnica, llamada *interpolación lineal*, se ilustra de manera gráfica en la figura 18.2. Utilizando triángulos semejantes,

$$\frac{f_1(x) - f(x_0)}{x - x_0} = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$$

reordenándose se tiene

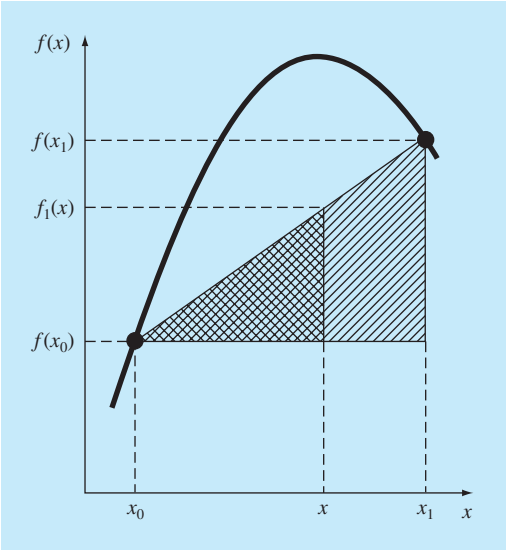
$$f_1(x) = f(x_0) + \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x - x_0)$$

(18.2)

que es una *fórmula de interpolación lineal*. La notación $f_1(x)$ designa que éste es un polinomio de interpolación de primer grado. Observe que además de representar la pendiente de la línea que une los puntos, el término $[f(x_1) - f(x_0)]/(x_1 - x_0)$ es una aproximación en diferencia dividida finita a la primer derivada [ecuación (4.17)]. En general,

FIGURA 18.2

Esquema gráfico de la interpolación lineal. Las áreas sombreadas indican los triángulos semejantes usados para obtener la fórmula de la interpolación lineal [ecuación(18.2)].



cuanto menor sea el intervalo entre los datos, mejor será la aproximación. Esto se debe al hecho de que, conforme el intervalo disminuye, una función continua estará mejor aproximada por una línea recta. Esta característica se demuestra en el siguiente ejemplo.

EJEMPLO 18.1 Interpolación lineal

Planteamiento del problema. Estime el logaritmo natural de 2 mediante interpolación lineal. Primero, realice el cálculo por interpolación entre $\ln 1 = 0$ y $\ln 6 = 1.791759$. Después, repita el procedimiento, pero use un intervalo menor de $\ln 1$ a $\ln 4$ (1.386294). Observe que el valor verdadero de $\ln 2$ es 0.6931472.

Solución. Usamos la ecuación (18.2) y una interpolación lineal para $\ln(2)$ desde $x_0 = 1$ hasta $x_1 = 6$ para obtener

$$f_1(2) = 0 + \frac{1.791759 - 0}{6 - 1}(2 - 1) = 0.3583519$$

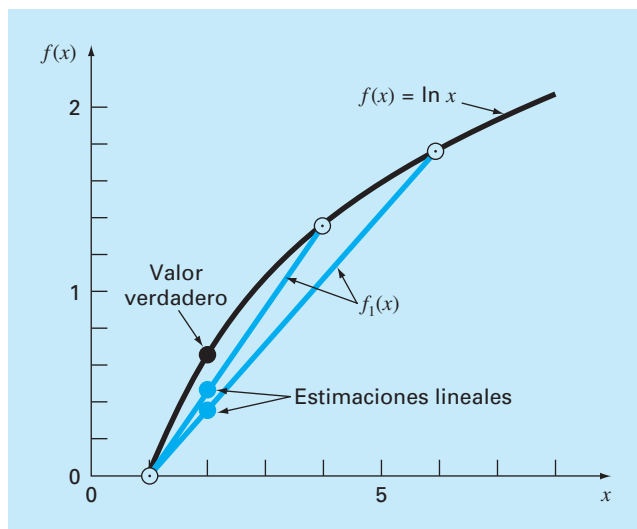
que representa un error: $\varepsilon_t = 48.3\%$. Con el intervalo menor desde $x_0 = 1$ hasta $x_1 = 4$ se obtiene

$$f_1(2) = 0 + \frac{1.386294 - 0}{4 - 1}(2 - 1) = 0.4620981$$

Así, usando el intervalo más corto el error relativo porcentual se reduce a $\varepsilon_t = 33.3\%$. Ambas interpolaciones se muestran en la figura 18.3, junto con la función verdadera.

FIGURA 18.3

Dos interpolaciones lineales para estimar $\ln 2$. Observe cómo el intervalo menor proporciona una mejor estimación.



18.1.2 Interpolación cuadrática

En el ejemplo 18.1 el error resulta de nuestra aproximación a una curva mediante una línea recta. En consecuencia, una estrategia para mejorar la estimación consiste en introducir alguna curvatura a la línea que une los puntos. Si se tienen tres puntos como datos, éstos pueden ajustarse en un polinomio de segundo grado (también conocido como polinomio cuadrático o *parábola*). Una forma particularmente conveniente para ello es

$$f_2(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) \quad (18.3)$$

Observe que aunque la ecuación (18.3) parece diferir del polinomio general [ecuación (18.1)], las dos ecuaciones son equivalentes. Lo anterior se demuestra al multiplicar los términos de la ecuación (18.3):

$$f_2(x) = b_0 + b_1x - b_1x_0 + b_2x^2 + b_2x_0x_1 - b_2xx_0 - b_2xx_1$$

o, agrupando términos,

$$f_2(x) = a_0 + a_1x + a_2x^2$$

donde

$$a_0 = b_0 - b_1x_0 + b_2x_0x_1$$

$$a_1 = b_1 - b_2x_0 - b_2x_1$$

$$a_2 = b_2$$

Así, las ecuaciones (18.1) y (18.3) son formas alternativas, equivalentes del único polinomio de segundo grado que une los tres puntos.

Un procedimiento simple puede usarse para determinar los valores de los coeficientes. Para encontrar b_0 , en la ecuación (18.3) se evalúa con $x = x_0$ para obtener

$$b_0 = f(x_0) \quad (18.4)$$

La ecuación (18.4) se sustituye en la (18.3), después se evalúa en $x = x_1$ para tener

$$b_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0} \quad (18.5)$$

Por último, las ecuaciones (18.4) y (18.5) se sustituyen en la (18.3), después se evalúa en $x = x_2$ y (luego de algunas manipulaciones algebraicas) se resuelve para

$$b_2 = \frac{\frac{f(x_2) - f(x_1)}{x_2 - x_1} - \frac{f(x_1) - f(x_0)}{x_1 - x_0}}{x_2 - x_0} \quad (18.6)$$

Observe que, como en el caso de la interpolación lineal, b_1 todavía representa la pendiente de la línea que une los puntos x_0 y x_1 . Así, los primeros dos términos de la ecuación (18.3) son equivalentes a la interpolación lineal de x_0 a x_1 , como se especificó antes en la ecuación (18.2). El último término, $b_2(x - x_0)(x - x_1)$, determina la curvatura de segundo grado en la fórmula.

Antes de ilustrar cómo utilizar la ecuación (18.3), debemos examinar la forma del coeficiente b_2 . Es muy similar a la aproximación en diferencias divididas finitas de la segunda derivada, que se presentó antes en la ecuación (4.24). Así, la ecuación (18.3) comienza a manifestar una estructura semejante a la expansión de la serie de Taylor. Esta observación será objeto de una mayor exploración cuando relacionemos los polinomios de interpolación de Newton con la serie de Taylor en la sección 18.1.4. Aunque, primero, mostraremos un ejemplo que indique cómo se utiliza la ecuación (18.3) para interpolar entre tres puntos.

EJEMPLO 18.2 Interpolación cuadrática

Planteamiento del problema. Ajuste un polinomio de segundo grado a los tres puntos del ejemplo 18.1:

$$\begin{array}{ll} x_0 = 1 & f(x_0) = 0 \\ x_1 = 4 & f(x_1) = 1.386294 \\ x_2 = 6 & f(x_2) = 1.791759 \end{array}$$

Con el polinomio evalúe $\ln 2$.

Solución. Aplicando la ecuación (18.4) se obtiene

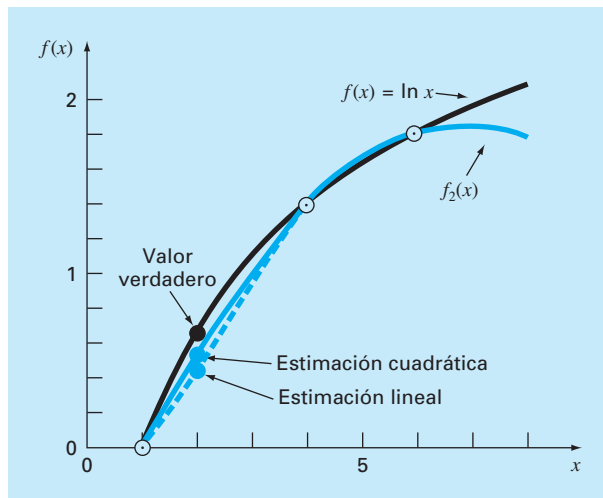
$$b_0 = 0$$

La ecuación (18.5) da

$$b_1 = \frac{1.386294 - 0}{4 - 1} = 0.4620981$$

FIGURA 18.4

El uso de la interpolación cuadrática para estimar $\ln 2$. Para comparación se presenta también la interpolación lineal desde $x = 1$ hasta 4.



y con la ecuación (18.6) se obtiene

$$b_2 = \frac{\frac{1.791759 - 1.386294}{6 - 4} - 0.4620981}{6 - 1} = -0.0518731$$

Sustituyendo estos valores en la ecuación (18.3) se obtiene la fórmula cuadrática

$$f_2(x) = 0 + 0.4620981(x - 1) - 0.0518731(x - 1)(x - 4)$$

que se evalúa en $x = 2$ para

$$f_2(2) = 0.5658444$$

que representa un error relativo de $\varepsilon_t = 18.4\%$. Así, la curvatura determinada por la fórmula cuadrática (figura 18.4) mejora la interpolación comparándola con el resultado obtenido antes al usar las líneas rectas del ejemplo 18.1 y en la figura 18.3.

18.1.3 Forma general de los polinomios de interpolación de Newton

El análisis anterior puede generalizarse para ajustar un polinomio de n -ésimo grado a $n + 1$ datos. El polinomio de n -ésimo grado es

$$f_n(x) = b_0 + b_1(x - x_0) + \cdots + b_n(x - x_0)(x - x_1) \cdots (x - x_{n-1}) \quad (18.7)$$

Como se hizo antes con las interpolaciones lineales y cuadráticas, los puntos asociados con datos se utilizan para evaluar los coeficientes $b_0, b_1, \dots, b_n$. Para un polinomio de n -ésimo grado se requieren $n + 1$ puntos: $[x_0, f(x_0)], [x_1, f(x_1)], \dots, [x_n, f(x_n)]$. Usamos estos datos y las siguientes ecuaciones para evaluar los coeficientes:

$$b_0 = f(x_0) \quad (18.8)$$

$$b_1 = f[x_1, x_0] \quad (18.9)$$

$$b_2 = f[x_2, x_1, x_0] \quad (18.10)$$

⋮
⋮
⋮

$$b_n = f[x_n, x_{n-1}, \dots, x_1, x_0] \quad (18.11)$$

donde las evaluaciones de la función colocadas entre paréntesis son diferencias divididas finitas. Por ejemplo, la primera diferencia dividida finita en forma general se representa como

$$f[x_i, x_j] = \frac{f(x_i) - f(x_j)}{x_i - x_j} \quad (18.12)$$

La *segunda diferencia dividida finita*, que representa la diferencia de las dos primeras diferencias divididas, se expresa en forma general como

$$f[x_i, x_j, x_k] = \frac{f[x_i, x_j] - f[x_j, x_k]}{x_i - x_k} \quad (18.13)$$

i	x_i	$f(x_i)$	Primero	Segundo	Tercero
0	x_0	$f(x_0)$	$f[x_1, x_0]$	$f[x_2, x_1, x_0]$	$f[x_3, x_2, x_1, x_0]$
1	x_1	$f(x_1)$	$f[x_2, x_1]$	$f[x_3, x_2, x_1]$	
2	x_2	$f(x_2)$	$f[x_3, x_2]$		
3	x_3	$f(x_3)$			

FIGURA 18.5

Representación gráfica de la naturaleza recursiva de las diferencias divididas finitas.

En forma similar, la n -ésima diferencia dividida finita es

$$f[x_n, x_{n-1}, \dots, x_1, x_0] = \frac{f[x_n, x_{n-1}, \dots, x_1] - f[x_{n-1}, x_{n-2}, \dots, x_0]}{x_n - x_0} \quad (18.14)$$

Estas diferencias sirven para evaluar los coeficientes en las ecuaciones (18.8) a (18.11), los cuales se sustituirán en la ecuación (18.7) para obtener el polinomio de interpolación

$$f_n(x) = f(x_0) + (x - x_0)f[x_1, x_0] + (x - x_0)(x - x_1)f[x_2, x_1, x_0] + \dots + (x - x_0)(x - x_1) \cdots (x - x_{n-1})f[x_n, x_{n-1}, \dots, x_0] \quad (18.15)$$

que se conoce como *polinomio de interpolación de Newton en diferencias divididas*. Debe observarse que no se requiere que los datos utilizados en la ecuación (18.15) estén igualmente espaciados o que los valores de la abscisa estén en orden ascendente, como se ilustra en el siguiente ejemplo. También, advierta cómo las ecuaciones (18.12) a (18.14) son recursivas (es decir, las diferencias de orden superior se calculan tomando diferencias de orden inferior (figura 18.5). Tal propiedad se aprovechará cuando desarrollemos un programa computacional eficiente en la sección 18.1.5 para implementar el método.

EJEMPLO 18.3 Polinomios de interpolación de Newton en diferencias divididas

Planteamiento del problema. En el ejemplo 18.2, los datos $x_0 = 1$, $x_1 = 4$ y $x_2 = 6$ se utilizaron para estimar $\ln 2$ mediante una parábola. Ahora, agregando un cuarto punto ($x_3 = 5$; $f(x_3) = 1.609438$), estime $\ln 2$ con un polinomio de interpolación de Newton de tercer grado.

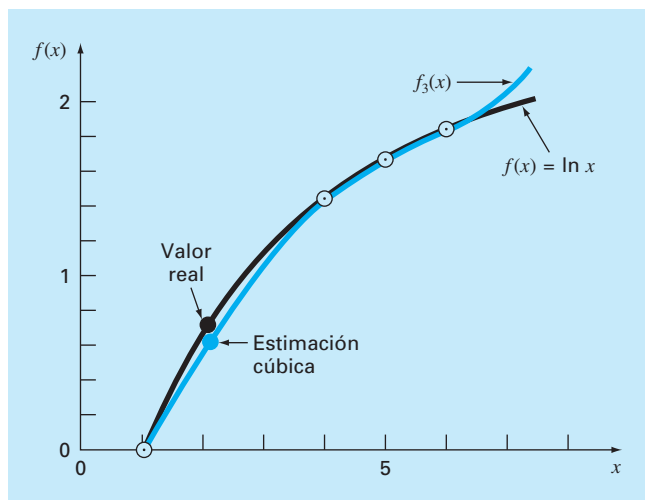
Solución. Utilizando la ecuación (18.7), con $n = 3$, el polinomio de tercer grado es

$$f_3(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) + b_3(x - x_0)(x - x_1)(x - x_2)$$

Las primeras diferencias divididas del problema son [ecuación (18.12)]

$$f[x_1, x_0] = \frac{1.386294 - 0}{4 - 0} = 0.4620981$$

$$f[x_2, x_1] = \frac{1.791759 - 1.386294}{6 - 4} = 0.2027326$$

**FIGURA 18.6**

Uso de la interpolación cúbica para estimar $\ln 2$.

$$f[x_3, x_2] = \frac{1.609438 - 1.791759}{5 - 6} = 0.1823216$$

Las segundas diferencias divididas son [ecuación (18.13)]

$$f[x_2, x_1, x_0] = \frac{0.2027326 - 0.4620981}{6 - 1} = -0.05187311$$

$$f[x_3, x_2, x_1] = \frac{0.1823216 - 0.2027326}{5 - 4} = -0.02041100$$

La tercera diferencia dividida es [ecuación (18.14) con $n = 3$]

$$f[x_3, x_2, x_1, x_0] = \frac{-0.02041100 - (-0.05187311)}{5 - 1} = 0.007865529$$

Los resultados de $f[x_1, x_0]$, $f[x_2, x_1, x_0]$ y $f[x_3, x_2, x_1, x_0]$ representan los coeficientes b_1 , b_2 y b_3 de la ecuación (18.7), respectivamente. Junto con $b_0 = f(x_0) = 0.0$, la ecuación (18.7) es

$$\begin{aligned} f_3(x) = & 0 + 0.4620981(x - 1) - 0.05187311(x - 1)(x - 4) \\ & + 0.007865529(x - 1)(x - 4)(x - 6) \end{aligned}$$

la cual sirve para evaluar $f_3(2) = 0.6287686$, que representa un error relativo: $\varepsilon_t = 9.3\%$. La gráfica del polinomio cúbico se muestra en la figura 18.6.

18.1.4 Errores de la interpolación polinomial de Newton

Observe que la estructura de la ecuación (18.15) es similar a la expansión de la serie de Taylor en el sentido de que se van agregando términos en forma secuencial, para mostrar el comportamiento de orden superior de la función. Estos términos son diferencias divididas finitas y, así, representan aproximaciones de las derivadas de orden superior. En consecuencia, como ocurrió con la serie de Taylor, si la función verdadera es un polinomio de n -ésimo grado, entonces el polinomio de interpolación de n -ésimo grado basado en $n + 1$ puntos dará resultados exactos.

También, como en el caso de la serie de Taylor, es posible obtener una formulación para el error de truncamiento. De la ecuación (4.6) recuerde que el error de truncamiento en la serie de Taylor se expresa en forma general como

$$R_n = \frac{f^{(n+1)}(\xi)}{(n+1)!} (x_{i+1} - x_i)^{n+1} \quad (4.6)$$

donde ξ está en alguna parte del intervalo de x_i a x_{i+1} . Para un polinomio de interpolación de n -ésimo grado, una expresión análoga para el error es

$$R_n = \frac{f^{(n+1)}(\xi)}{(n+1)!} (x - x_0)(x - x_1) \cdots (x - x_n) \quad (18.16)$$

donde ξ está en alguna parte del intervalo que contiene la incógnita y los datos. Para que esta fórmula sea útil, la función en turno debe ser conocida y diferenciable. Por lo común éste no es el caso. Por fortuna, hay una formulación alternativa que no requiere del conocimiento previo de la función. Utilizándose una diferencia dividida finita para aproximar la $(n + 1)$ -ésima derivada,

$$R_n = f[x, x_n, x_{n-1}, \dots, x_0](x - x_0)(x - x_1) \cdots (x - x_n) \quad (18.17)$$

donde $f[x, x_n, x_{n-1}, \dots, x_0]$ es la $(n + 1)$ -ésima diferencia dividida finita. Debido a que la ecuación (18.17) contiene la incógnita $f(x)$, no permite obtener el error. Sin embargo, si se tiene un dato más, $f(x_{n+1})$, la ecuación (18.17) puede usarse para estimar el error como sigue:

$$R_n \cong f[x_{n+1}, x_n, x_{n-1}, \dots, x_0](x - x_0)(x - x_1) \cdots (x - x_n) \quad (18.18)$$

EJEMPLO 18.4 Estimación del error por el polinomio de Newton

Planteamiento del problema. Con la ecuación (18.18) estime el error en la interpolación polinomial de segundo grado del ejemplo 18.2. Use el dato adicional $f(x_3) = f(5) = 1.609438$ para obtener sus resultados.

Solución. Recuerde que en el ejemplo 18.2 el polinomio de interpolación de segundo grado proporcionó una estimación, $f_2(2) = 0.5658444$, que representa un error de $0.6931472 - 0.5658444 = 0.1273028$. Si no se hubiera conocido el valor verdadero, como usualmente sucede, la ecuación (18.18), junto con el valor adicional en x_3 , pudo haberse utilizado para estimar el error,

$$R_2 = f[x_3, x_2, x_1, x_0](x - x_0)(x - x_1)(x - x_2)$$

o

$$R_2 = 0.007865529(x-1)(x-4)(x-6)$$

donde el valor de la diferencia dividida finita de tercer orden es como se calculó antes en el ejemplo 18.3. Esta expresión se evalúa en $x = 2$ para obtener

$$R_2 = 0.007865529(2-1)(2-4)(2-6) = 0.0629242$$

que es del mismo orden de magnitud que el error verdadero.

Con el ejemplo anterior y la ecuación (18.18), debe resultar claro que el error estimado para el polinomio de n -ésimo grado es equivalente a la diferencia entre las predicciones de orden $(n+1)$ y de orden n . Es decir,

$$R_n = f_{n+1}(x) - f_n(x) \quad (18.19)$$

En otras palabras, el incremento que se agrega al caso de orden n para crear el caso de orden $(n+1)$ [es decir, la ecuación (18.18)] se interpreta como un estimado del error de orden n . Esto se percibe con claridad al reordenar la ecuación (18.19):

$$f_{n+1}(x) = f_n(x) + R_n$$

La validez de tal procedimiento se refuerza por el hecho de que la serie es altamente convergente. En tal situación, la predicción del orden $(n+1)$ debería ser mucho más cercana al valor verdadero que la predicción de orden n . En consecuencia, la ecuación (18.19) concuerda con nuestra definición estándar de error, al representar la diferencia entre la verdad y una aproximación. No obstante, observe que mientras todos los otros errores estimados para los procedimientos iterativos presentados hasta ahora se encontraron como una predicción presente menos una previa, la ecuación (18.19) constituye una predicción futura menos una presente. Lo anterior significa que para una serie que es de convergencia rápida, el error estimado de la ecuación (18.19) podría ser menor que el error verdadero. Esto representaría una calidad muy poco atractiva si el error estimado fuera a emplearse como un criterio de terminación. Sin embargo, como se expondrá en la siguiente sección, los polinomios de interpolación de grado superior son muy sensibles a errores en los datos (es decir, están mal condicionados). Cuando se emplean para interpolación, a menudo dan predicciones que divergen en forma significativa del valor verdadero. Si se trata de detectar errores, la ecuación (18.19) es más sensible a tal divergencia. De esta manera, es más valiosa con la clase de análisis de datos exploratorios para los que el polinomio de Newton es el más adecuado.

18.1.5 Algoritmo computacional para el polinomio de interpolación de Newton

Tres propiedades hacen a los polinomios de interpolación de Newton muy atractivos para aplicaciones en computadora:

1. Como en la ecuación (18.7), es posible desarrollar de manera secuencial versiones de grado superior con la adición de un solo término a la ecuación de grado inferior.

Esto facilita la evaluación de algunas versiones de diferente grado en el mismo programa. En especial tal capacidad es valiosa cuando el orden del polinomio no se conoce *a priori*. Al agregar nuevos términos en forma secuencial, podemos determinar cuándo se alcanza un punto de regreso disminuido (es decir, cuando la adición de términos de grado superior ya no mejora de manera significativa la estimación, o en ciertas situaciones incluso la aleja). Las ecuaciones para estimar el error, que se analizan en el punto 3, resultan útiles para visualizar un criterio objetivo para identificar este punto de términos disminuidos.

2. Las diferencias divididas finitas que constituyen los coeficientes del polinomio [ecuaciones (18.8) hasta (18.11)] se pueden calcular eficientemente. Es decir, como en la ecuación (18.14) y la figura 18.5, las diferencias de orden inferior sirven para calcular las diferencias de orden mayor. Utilizando esta información previamente determinada, los coeficientes se calculan de manera eficiente. El algoritmo en la figura 18.7 incluye un esquema así.
3. El error estimado [ecuación (18.18)] se incorpora con facilidad en un algoritmo computacional debido a la manera secuencial en la cual se construye la predicción.

Todas las características anteriores pueden aprovecharse e incorporarse en un algoritmo general para implementar el polinomio de Newton (figura 18.7). Observe que el algoritmo consiste de dos partes: la primera determina los coeficientes a partir de la ecuación (18.7); la segunda establece las predicciones y sus errores correspondientes. La utilidad de dicho algoritmo se demuestra en el siguiente ejemplo.

FIGURA 18.7

Un algoritmo para el polinomio de interpolación de Newton escrito en pseudocódigo.

```

SUBROUTINE NewtInt (x, y, n, xi, yint, ea)
  LOCAL fddn,n
  DOFOR i = 0, n
    fddi,0 = yi
  END DO
  DOFOR j = 1, n
    DOFOR i = 0, n - j
      fddi,j = (fddi+1,j-1 - fddi,j-1) / (xi+j - xi)
    END DO
  END DO
  xterm = 1
  yint0 = fdd0,0
  DOFOR order = 1, n
    xterm = xterm * (xi - xorder-1)
    yint2 = yintorder-1 + fdd0,order * xterm
    Eaorder-1 = yint2 - yintorder-1
    yintorder = yint2
  END DO
END NewtInt

```

EJEMPLO 18.5 Estimaciones del error para determinar el grado de interpolación adecuado

Planteamiento del problema. Después de incorporar el error [ecuación (18.18)], utilice el algoritmo computacional que se muestra en la figura 18.7 y la información siguiente para evaluar $f(x) = \ln x$ en $x = 2$:

x	$f(x) = \ln x$
0	1
4	1.3862944
6	1.7917595
5	1.6094379
3	1.0986123
1.5	0.4054641
2.5	0.9162907
3.5	1.2527630

Solución. Los resultados de emplear el algoritmo de la figura 18.7 para obtener una solución se muestran en la figura 18.8. El error estimado, junto con el error verdadero (basándose en el hecho de que $\ln 2 = 0.6931472$), se ilustran en la figura 18.9. Observe que el error estimado y el error verdadero son similares y que su concordancia mejora conforme aumenta el grado. A partir de estos resultados se concluye que la versión de quinto grado da una buena estimación y que los términos de grado superior no mejoran significativamente la predicción.

NUMERO DE PUNTOS? 8		
X(0),	Y(0) = ?	1,0
X(1),	Y(1) = ?	4,1.3862944
X(2),	Y(2) = ?	6,1.7917595
X(3),	Y(3) = ?	5,1.6094379
X(4),	Y(4) = ?	3,1.0986123
X(5),	Y(5) = ?	1.5,0.40546411
X(6),	Y(6) = ?	2.5,0.91629073
X(7),	Y(7) = ?	3.5,1.2527630
INTERPOLACION EN X = 2		
GRADO	F(X)	ERROR
0	0.000000	0.462098
1	0.462098	0.103746
2	0.565844	0.062924
3	0.628769	0.046953
4	0.675722	0.021792
5	0.697514	-0.003616
6	0.693898	-0.000459
7	0.693439	

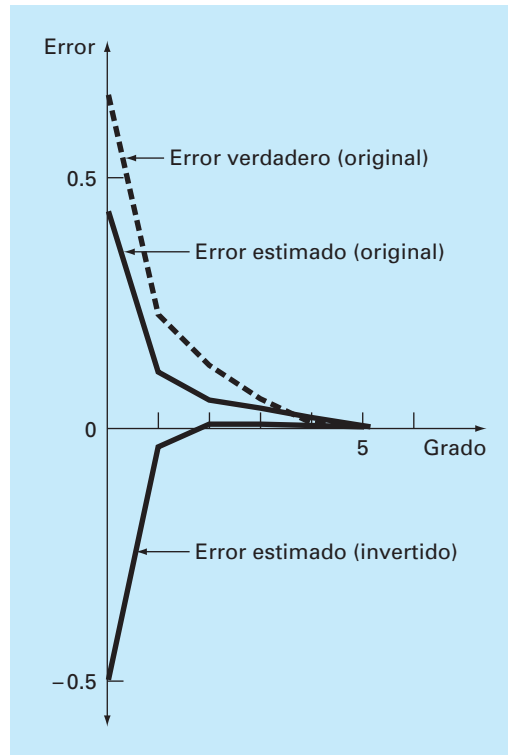
FIGURA 18.8 Resultados de un programa, basado en el algoritmo de la figura 18.7, para evaluar $\ln 2$.

Este ejercicio también ilustra la importancia de la posición y el orden de los puntos. Por ejemplo, hasta la estimación de tercer grado, la mejoría es lenta debido a que los puntos que se agregaron (en $x = 4, 6$ y 5) están distantes y a un lado del punto de análisis en $x = 2$. La estimación de cuarto grado muestra una mejoría un poco mayor, ya que el nuevo punto en $x = 3$ está más cerca de la incógnita. Aunque, la disminución más dramática en el error corresponde a la inclusión del término de quinto grado usando el dato en $x = 1.5$. Dicho punto está cerca de la incógnita y también se halla al lado opuesto de la mayoría de los otros puntos. En consecuencia, el error se reduce a casi un orden de magnitud.

La importancia de la posición y el orden de los datos también se demuestra al usar los mismos datos para obtener una estimación para $\ln 2$, pero considerando los puntos en un orden diferente. La figura 18.9 muestra los resultados en el caso de invertir el orden de los datos originales; es decir, $x_0 = 3.5$, $x_1 = 2.5$, $x_3 = 1.5$, y así sucesivamente. Como los puntos iniciales en este caso se hallan más cercanos y espaciados a ambos lados de $\ln 2$, el error disminuye mucho más rápidamente que en la situación original. En el término de segundo grado, el error se redujo a menos de $\varepsilon_t = 2\%$. Se podrían emplear otras combinaciones para obtener diferentes velocidades de convergencia.

FIGURA 18.9

Errores relativos porcentuales para la predicción de $\ln 2$ como función del orden del polinomio de interpolación.



El ejemplo anterior ilustra la importancia de la selección de los puntos. Como es intuitivamente lógico, los puntos deberían estar centrados alrededor, y tan cerca como sea posible, de las incógnitas. Esta observación también se sustenta por un análisis directo de la ecuación para estimar el error [ecuación (18.17)]. Si suponemos que la diferencia dividida finita no varía mucho a través de los datos, el error es proporcional al producto: $(x - x_0)(x - x_1) \cdots (x - x_n)$. Obviamente, cuanto más cercanos a x estén los puntos, menor será la magnitud de este producto.

18.2 POLINOMIOS DE INTERPOLACIÓN DE LAGRANGE

El *polinomio de interpolación de Lagrange* es simplemente una reformulación del polinomio de Newton que evita el cálculo de las diferencias divididas, y se representa de manera concisa como

$$f_n(x) = \sum_{i=0}^n L_i(x) f(x_i) \quad (18.20)$$

donde

$$L_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^n \frac{x - x_j}{x_i - x_j} \quad (18.21)$$

donde Π designa el “producto de”. Por ejemplo, la versión lineal ($n = 1$) es

$$f_1(x) = \frac{x - x_1}{x_0 - x_1} f(x_0) + \frac{x - x_0}{x_1 - x_0} f(x_1) \quad (18.22)$$

y la versión de segundo grado es

$$\begin{aligned} f_2(x) = & \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} f(x_0) + \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} f(x_1) \\ & + \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)} f(x_2) \end{aligned} \quad (18.23)$$

La ecuación (18.20) se obtiene de manera directa del polinomio de Newton (cuadro 18.1). Sin embargo, el razonamiento detrás de la formulación de Lagrange se comprende directamente al darse cuenta de que cada término $L_i(x)$ será 1 en $x = x_i$ y 0 en todos los otros puntos (figura 18.10). De esta forma, cada producto $L_i(x) f(x_i)$ toma el valor de $f(x_i)$ en el punto x_i . En consecuencia, la sumatoria de todos los productos en la ecuación (18.20) es el único polinomio de n -ésimo grado que pasa exactamente a través de todos los $n + 1$ puntos, que se tienen como datos.

EJEMPLO 18.6 Polinomios de interpolación de Lagrange

Planteamiento del problema. Con un polinomio de interpolación de Lagrange de primero y segundo grado evalúe $\ln 2$ basándose en los datos del ejemplo 18.2:

$$\begin{aligned}x_0 &= 1 & f(x_0) &= 0 \\x_1 &= 4 & f(x_1) &= 1.386294 \\x_2 &= 6 & f(x_2) &= 1.791760\end{aligned}$$

Solución. El polinomio de primer grado [ecuación (18.22)] se utiliza para obtener la estimación en $x = 2$,

$$f_1(2) = \frac{2-4}{1-4}0 + \frac{2-1}{4-1}1.386294 = 0.4620981$$

De manera similar, el polinomio de segundo grado se desarrolla así: [ecuación (18.23)]

$$\begin{aligned}f_2(2) &= \frac{(2-4)(2-6)}{(1-4)(1-6)}0 + \frac{(2-1)(2-6)}{(4-1)(4-6)}1.386294 \\&\quad + \frac{(2-1)(2-4)}{(6-1)(6-4)}1.791760 = 0.5658444\end{aligned}$$

Como se esperaba, ambos resultados concuerdan con los que se obtuvieron antes al usar el polinomio de interpolación de Newton.

Cuadro 18.1 Obtención del polinomio de Lagrange directamente a partir del polinomio de interpolación de Newton

El polinomio de interpolación de Lagrange se obtiene de manera directa a partir de la formulación del polinomio de Newton. Haremos esto únicamente en el caso del polinomio de primer grado [ecuación (18.2)]. Para obtener la forma de Lagrange, reformulamos las diferencias divididas. Por ejemplo, la primera diferencia dividida,

$$f[x_1, x_0] = \frac{f(x_1) - f(x_0)}{x_1 - x_0} \quad (\text{B18.1.1})$$

se reformula como

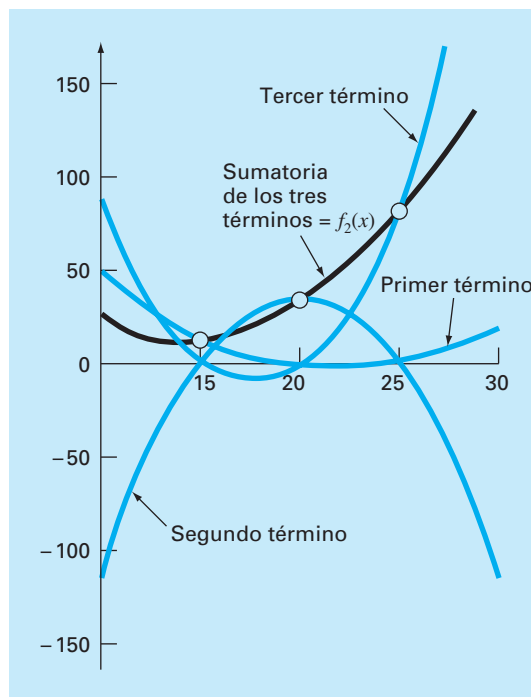
$$f[x_1, x_0] = \frac{f(x_1)}{x_1 - x_0} + \frac{f(x_0)}{x_0 - x_1} \quad (\text{B18.1.2})$$

conocida como la *forma simétrica*. Al sustituir la ecuación (B18.1.2) en la (18.2) se obtiene

$$f_1(x) = f(x_0) + \frac{x - x_0}{x_1 - x_0}f(x_1) + \frac{x - x_1}{x_0 - x_1}f(x_0)$$

Por último, al agrupar términos semejantes y simplificar se obtiene la forma del polinomio de Lagrange,

$$f_1(x) = \frac{x - x_1}{x_0 - x_1}f(x_0) + \frac{x - x_0}{x_1 - x_0}f(x_1)$$

**FIGURA 18.10**

Descripción visual del razonamiento detrás del polinomio de Lagrange. Esta figura muestra un caso de segundo grado. Cada uno de los tres términos en la ecuación (18.23) pasa a través de uno de los puntos que se tienen como datos y es cero en los otros dos. La suma de los tres términos, por lo tanto, debe ser el único polinomio de segundo grado $f_2(x)$ que pasa exactamente a través de los tres puntos.

FIGURA 18.11

Seudocódigo para la interpolación de Lagrange. Este algoritmo se establece para calcular una sola predicción de grado n -ésimo, donde $n + 1$ es el número de datos.

```

FUNCTION Lagrng(x, y, n, x)
  sum = 0
  DOFOR i = 0, n
    product = y_i
    DOFOR j = 0, n
      IF i ≠ j THEN
        product = product*(x - x_j)/(x_i - x_j)
      ENDIF
    END DO
    sum = sum + product
  END DO
  Lagrng = sum
END Lagrng

```