PEP 2 - Algoritmos Numéricos

Jennifer Andrea Velozo Bruna

2 de agosto de 2021

Resumen

Una función puede ser aproximada utilizando distintos métodos, tales como interpolación lineal, con splines, cúbica, entre otras técnicas. Y la precisión de dicha aproximación va a depender del método seleccionado y la cantidad de puntos que se usen. A diferencia de la aproximación, en la integración numérica, una mayor cantidad de puntos no siempre implica un mejor resultado.

1. Introducción

La interpolación es una técnica matemática que permite describir un conjunto de puntos dados mediante alguna función, es decir, busca una función aproximada que pase por dichos puntos.

Por otro lado, la integración numérica busca calcular el valor de una integral definida.

El presente documento tiene como objetivo mostrar los resultados y un análisis de la PEP N°2 de la asignatura Algoritmos Numéricos.

La experiencia se divide en dos partes:

1. Aplicar tres técnicas de aproximación de funciones en el intervalo [-10, 10] de:

   - $f(x) = sen(\pi x)/(\pi x)$

   Luego comparar y analizar el error usando $N = 10^1, 10^2, 10^3, 10^4$ en los métodos seleccionados anteriormente.

2. Seleccionar dos métodos de interpolación desarrollados en el ítem anterior y calcular la integral de Simpson compuesto, construir un algoritmo que estime el número de N de puntos para obtener un error integral de 10^{-p} al aplicar la integral de Simpson compuesto. Y comparar y concluir sobre los resultados de eficiencia de aplicar el predictor de N y un algoritmo adaptativo de integración.

2. Metodología

Para llevar a cabo la implementación de los algoritmos y realizar el análisis de los resultados obtenidos es necesario comprender algunos conceptos y las técnicas que se utilizan.

2.1. Error

El error utilizado en la experiencia es el Error Cuadrático Medio (RMSE), el cual mide la cantidad de error que hay entre dos conjuntos de datos, es decir, compara el valor predicho y un valor observado o conocido. Su fórmula queda expresada como:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(Predicted_i - Actual_i)^2}{N}}$$

2.2. Tolerancia

Es el valor de error que se desea obtener al utilizar un algoritmo numérico. Para el algoritmo que estima el número de N de puntos y el algoritmo adaptativo de integración, se utiliza una tolerancia de 10^{-p}, donde p es un valor que puede ser modificado.

2.3. Costo temporal

Corresponde al tiempo que tarda en ejecutarse el algoritmo hasta llegar a un resultado final. En la implementación se trabaja con la funciones *tic* y *toc* de Matlab para obtener el tiempo transcurrido, medido en segundos.

2.4. Especificaciones técnicas

No está demás mencionar las características técnicas que posee la máquina donde se implementan y ejecutan los algoritmos, puesto que si se ejecutan en una máquina con diferentes especificaciones, se pueden obtener los resultados en un menor o mayor tiempo que los que se presentan en el documento.

- Sistema operativo: Windows 10 Home 64 bits
- Procesador: Intel(R) Core(TM) i5-9300H CPU @ 2.40GHz (8 CPUs)
- Memoria RAM: 8192 MB

2.5. Técnicas utilizadas

Las tres técnicas de interpolación utilizadas son las siguientes:

- Lineal [1]
- Spline [1]
- Cúbica [1]

La regla de Simpson 1/3 es un método de integración numérica que se puede emplear sólo si el número de segmentos es par [1], y como en esta experiencia se tiene un cantidad par de tramos, se utiliza dicha técnica, la cual queda expresada de la siguiente manera:

$$I = (b-a)\frac{f(x_0)+4\sum_{i=1,3,5}^{n-1} f(x_i)+2\sum_{j=2,4,6}^{n-2} f(x_j)+f(x_n)}{3n}$$

3. Resultados

3.1. Métodos de aproximación

Con el objetivo de comparar la eficacia de cada uno de los métodos, se obtiene el Error Cuadrático Medio entre los puntos de la función original y los puntos que se obtienen de hacer la aproximación con cada uno de los métodos, y con los diferentes valores de N, como se muestra en la Tabla 1.

N	**Lineal**	**Spline**	**Cúbico**
10^1	0,16087	0,19342	0,17668
10^2	$0{,}3490 \times 10^{-2}$	$0{,}2151 \times 10^{-4}$	$0{,}2888 \times 10^{-3}$
10^3	$0{,}3511 \times 10^{-4}$	$0{,}1840 \times 10^{-8}$	$0{,}2335 \times 10^{-6}$
10^4	$0{,}3511 \times 10^{-6}$	$0{,}1838 \times 10^{-12}$	$0{,}2323 \times 10^{-9}$

Tabla 1: RMSE de cada método

Como se puede apreciar en la Tabla 1, la técnica de interpolación con Splines es la que posee menor error, mientras que la de interpolación Lineal resulta tener un error muy alto, por lo tanto, éstos son los dos métodos con los que se trabaja para la Parte 2 de la experiencia.

La Figura 1 muestra la interpolación Lineal con un $N = 10$, donde la línea roja corresponde a la aproximación, y la azul representa la función original.

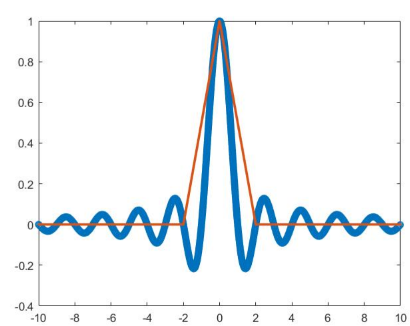

Figura 1: Interpolación Lineal con $N = 10$

La Figura 2 muestra la interpolación con Splines para un $N = 10$, donde la línea roja representa la aproximación.

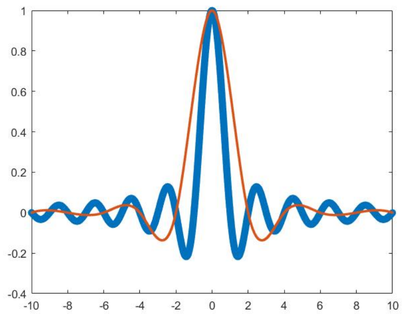

Figura 2: Interpolación con Splines con $N = 10$

La Figura 3 muestra la interpolación con Splines para un $N = 10^4$.

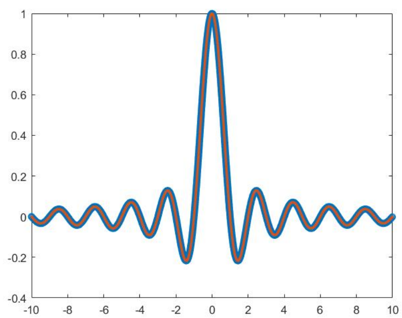

Figura 3: Interpolación con Splines con $N = 10^4$

3.2. Cálculo de integral

Para calcular la integral de Simpson compuesto, se aplica la regla de Simspson 1/3 dado que se tiene una cantidad de tramos par. Luego, se calcula la integral de la función aproximada obtenida por los métodos Lineal y Spline, para $N = 10^1, 10^2, 10^3, 10^4$ como se muestra en la Tabla 2. Cabe mencionar que el valor de la integral de la función original $f(x)$ es 0,979766544642306.

N	**Lineal**	**Spline**
10^1	1,99998	2,02059
10^2	0,98044	0,97976
10^3	0,97977	0,97977
10^4	0,97976	0,97977

Tabla 2: Valor integral

Una vez obtenido el valor de la integral con cada método, se procede a comparar con la integral de la función original, con el fin de evaluar la eficacia de cada método, tal como se muestra en la Tabla 3.

N	**Lineal**	**Spline**
10^1	$0{,}1020 \times 10^1$	$0{,}1041 \times 10^1$
10^2	$0{,}6711 \times 10^{-3}$	$0{,}9424 \times 10^{-5}$
10^3	$0{,}6667 \times 10^{-5}$	$0{,}4558 \times 10^{-9}$
10^4	$0{,}6666 \times 10^{-7}$	$0{,}4041 \times 10^{-13}$

Tabla 3: RMSE integral para Lineal y Spline

3.3. Predictor de N

Para construir el algoritmo que estime el número N de puntos equidistantes mínimos requeridos para obtener un error integral de 10^{-p}, se utiliza la fórmula de Error de la Regla de Simpson, la cual se expresa como:

$$E = h^4 \frac{(b-a)}{180} max|f^{(4)}(\theta)|, \text{ con } \theta \epsilon [a, b]$$

Con esto, se puede despejar el valor de N, ya que $h = (b-a)/N$, y como está elevado a la cuarta, se aplica 2 veces la raíz cuadrada, resultando la siguiente expresión para obtener N:

$$N = (b-a)\sqrt{\sqrt{\frac{(b-a)max|f^{(4)}(\theta)|}{E \times 180}}}$$

3.4. Algoritmo adaptativo de integración

Para implementar el algoritmo adaptativo de integración, se utiliza la fórmula de Simpson 1/3 para calcular la integral definida entre a y b, como lo expresa la siguiente ecuación:

$$S(a, b) = \frac{b-a}{6}[f(a) + f(b) + 4f(\frac{a+b}{2})]$$

El algoritmo es recursivo, ya que el intervalo $[a, b]$ se va dividiendo a la mitad en un punto m, con lo que se puede calcular la integral Simpson 1/3 entre a y m, y otra entre m y b, es decir, $S(a, m)$ y $S(m, b)$, cuya suma es una mejor aproximación de la integral definida que se desea encontrar. Con esto, el resultado de la integral queda definido como:

$$I = S(a,m) + S(m,b) + \frac{S(a,m)+S(m,b)-S(a,b)}{15}$$

Donde la fracción corresponde al término de ajuste, y a medida que la aproximación se acerque al valor real, ésta tiende a 0. Por lo tanto, esa expresión se puede utilizar como una medida de error, verificando lo siguiente para una tolerancia ϵ:

$$\left|\frac{S(a,m)+S(m,b)-S(a,b)}{15}\right| < \epsilon$$

En la Tabla 4, se presenta el número de ejecuciones, el tiempo que tarda en ejecutarse y la cantidad de puntos que entrega el algoritmo predictor y el adaptativo para una tolerancia de 10^{-9}.

Algoritmo	**Ejec.**	**Tiempo**	**Puntos**
Predictor	1	$0{,}4174 \times 10^{-8}$	4448
Adaptativo	523	0,0074	263

Tabla 4: Comparación entre algoritmo predictor y adaptativo

La Figura 4 muestra la gráfica de la función con la cantidad de puntos que obtiene el algoritmo predictor.

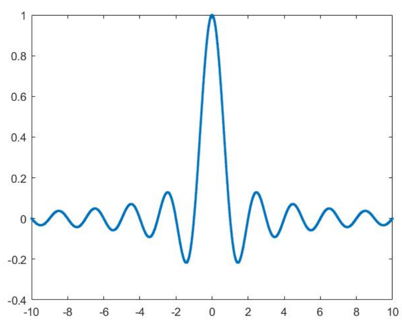

Figura 4: Gráfica de la función utilizando la cantidad de puntos del predictor

La Figura 5 muestra la gráfica de la función utilizando los puntos que obtiene el algoritmo adaptativo.

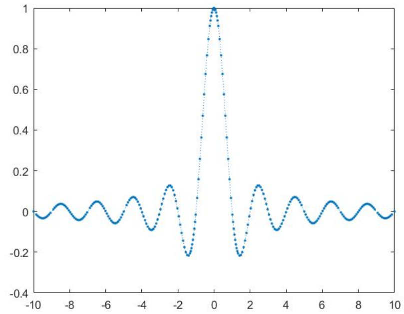

Figura 5: Gráfica de la función utilizando los puntos del adaptativo

4. Discusión

Como se pudo apreciar en la Tabla 1, para un $N = 10^1$ el mejor método de aproximación es el de interpolación Lineal, y el peor es el Spline. Pero a medida que aumenta el valor de N, Spline comienza a ser el más eficaz, mientras que Lineal comienza a tener un error mayor que Spline, con lo cual se deduce que el método de interpolación con splines posee una mejor precisión. Esto se debe a que Spline trata de ajustar de mejor manera las curvas, construyendo diferentes sistemas de ecuaciones, donde cada ecuación corresponde a un polinomio diferente, que por lo general son de grado 2 o 3. Por otro lado, la interpolación lineal, utiliza un ajuste lineal, donde las funciones no tienen potencia.

Al analizar los valores de la Tabla 2, donde se muestra el resultado de la integral de Simpson, utilizando los métodos de interpolación Lineal y con Splines, se puede observar que desde un $N = 10^3$ el resultado comienza a acercarse más al valor real. De hecho, al observar los RMSE de la Tabla 3, con el método de interpolación con Splines, el error integral es el menor, sobre todo para un $N = 10^4$.

Con todo lo anterior dicho, se puede deducir que para estos métodos, a medida que aumenta la cantidad de puntos, la aproximación es mejor sobre todo para la interpolación con Splines.

Por otro lado, comparando el algoritmo predictor de N y el algoritmo adaptativo, se puede concluir que en términos de eficiencia, es mejor el predictor, ya que realiza el cálculo sólo una vez para estimar la cantidad de puntos, mientras que el adaptativo debe realizar 523 llamadas recursivas para alcanzar una tolerancia de 10^{-9}. Además el predictor posee un costo temporal mucho menor que el adaptativo.

Sin embargo, la idea del adaptativo es obtener una cantidad de puntos mucho menor, lo cual resulta ser así, ya que para la tolerancia mencionada anteriormente, el adaptativo entrega 263 puntos, mientras que el predictor obtiene 4448, lo cual equivale a aproximadamente 17 veces el valor que obtiene el adaptativo.

El hecho de que el predictor entregue muchos puntos, se debe a que en este algoritmo, se solicita que los puntos sean equidistantes, mientras que en el adaptativo, los puntos no necesariamente tienen la misma distancia entre ellos, ganando ventaja sobre el predictor al tener mayor libertad para determinar los puntos.

5. Conclusiones

A modo de conclusión, se puede decir que los objetivos fueron cumplidos de manera satisfactoria, pudiendo realizar todo lo solicitado en el enunciado de la experiencia.

Al dar por finalizada la experiencia, se puede concluir que al momento de aproximar una función, la precisión va a depender netamente del método que se utilice y la cantidad de puntos. En este caso, como la función se utilizó con un $h = (a - b)/10^5$, al aproximar con un $N = 10$ o un $N = 10^2$ el error fue muy grande, mientras que para un $N = 10^3$ o mayor, fue disminyendo.

Por otro lado, respecto a la integración, se puede decir que estos cálculos sirvieron para confirmar aún más si las aproximaciones obtenidas en la Parte 1 fueron buenas o malas, pudiendo comparar los valores de las integrales de estas aproximaciones con el valor real de la integral de la función.

Respecto a los algoritmos predictor y adaptativo, se puede decir que no siempre se necesita una elevada cantidad de puntos, ya que como fue en el caso del algoritmo adaptativo, éste no necesitó de muchos puntos para alcanzar una tolerancia de 10^{-9}. Por lo tanto, no siempre una mayor cantidad de puntos implica un valor más preciso de integración.

Referencias

[1] Plaza, Sergio. (2017) Métodos Numéricos. Depto. de Matemática, Facultad de Ciencias, Universidad de Santiago de Chile.