INTERPOLACIÓN
SPLINE CUBICO

Breayann Ortiz Aldana, Andrés Diaz del Castillo, Brayan Ricardo García
Bogotá, Colombia

Breayanortiz@javeriana.edu.co, a.diazdelcastillo@javeriana.edu.co ,brayan-garcia@javeriana.edu.co
Bogotá D.C.
2019

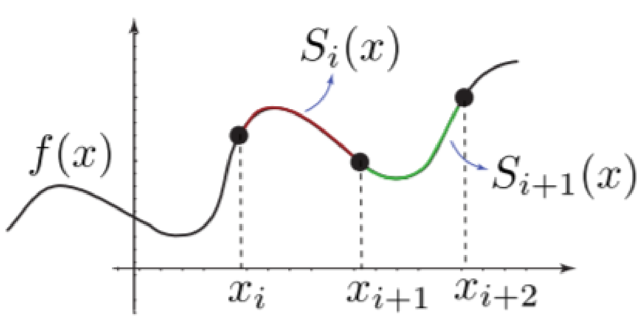
***Resumen*— Dentro de este documento podrá encontrar una solución a un reto propuesto en la asignatura de análisis numérico, específicamente, como interpolar los datos que describen el perímetro de una mano.**

***Palabras claves*—Interpolar, Nimerico.**

I. INTRODUCCIÓN

LA primera parte del ejercicio se puede definir como sigue. Se tienen una serie de puntos que describen el perímetro de una mano, se requiere, con métodos numéricos que involucran interpolación, se proceda a describir dichos puntos. Cuando se habla de describir específicamente se está hablando de un polinomio que describa el comportamiento de estos puntos.

Para el desarrollo se utilizó el método de trazadores cúbicos. Donde se puede interpretar a un trazador cubico como una barra flexible que se ajusta a una serie de datos. A continuación, se presenta una breve descripción de un trazador cubico:



E la anterior imagen se aprecia la finalidad de un spline, el cual describe una curva seccionándola y generando polinomios que describen la curva a trozos. De la imagen anterior se puede deducir que:

$$S(x) = \begin{cases} S_0(x) & \text{si} \quad x \in [x_0, x_1], \\ S_1(x) & \text{si} \quad x \in [x_1, x_2], \\ \vdots & \quad\ \vdots \\ S_{n-1}(x) & \text{si} \quad x \in [x_{n-1}, x_n], \end{cases}$$

Donde S(x) representa cada uno de los polinomios que describen a la función f(x).

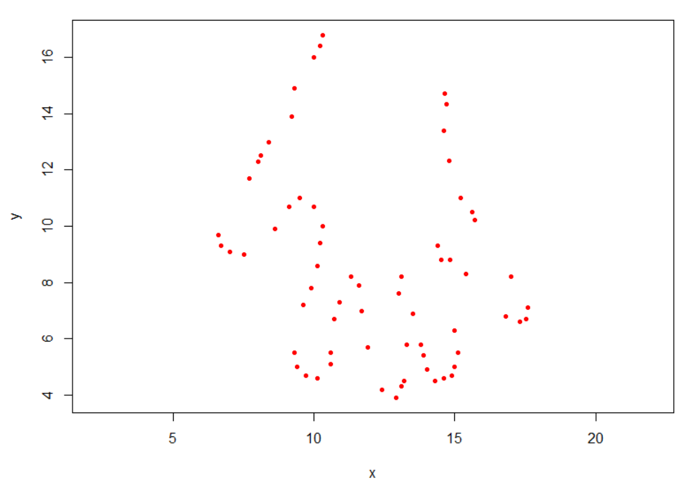
Existen una serie de condiciones para crear un spline cubico, dentro de las cuales se puede establecer dos tipos de fronteras, Frontera libre o natural y frontera sujeta.

$$S''(x_0) = S''(x_n) = 0 \ \textbf{(frontera libre o natural)}$$
$$S'(x_0) = f'(x_0) \ \text{ y } \ S'(x_n) = f'(x_n) \ \textbf{(frontera sujeta)}$$

Teniendo en cuenta lo anterior se procede a buscar la solución para e problema planteado.

II. RETO

La siguiente imagen ilustra la esencia del reto, esta imagen contiene los puntos suministrados graficados en un plano x-y.



Se sabe también que la cantidad de puntos es de 67.

Una descripción acerca de cual es la finalidad del reto es:

Construir un Interpolar (no necesariamente en forma polinómica) utilizando la menor cantidad de puntos k y reproducir el dibujo completo de la mano (mejor exactitud) con la información dada en el script.

Una vez planteada la información necesaria para llevar a cabo una solución se procede a realizar un análisis como sigue.

III. ANÁLISIS DEL PROBLEMA

A continuación, se definen las entradas que se consideran en el problema:

a. Entradas

Se espera recibir una serie de puntos compuestos de coordenadas x-y.

- $X = [x_0, x_1, x_2, \dots, x_n]$ y
- $Y = [y_0, y_1, y_2, \dots, y_n]$, tales que para cada $y_i = s(x_i)$.

Donde se puede establecer las siguientes condiciones:

- $s(x_i)$ es un polinomio de grado ≤ K en cada subintervalo.
- $s(x_i)$ tiene derivada continua hasta de orden K-1 en = $[x_0, x_n]$.

Una vez definidas las entradas se procede a definir las salidas.

b. Salidas

Ya que se está utilizando interpolación con spline cubico se tendrá un sistema lineal descrito por una ecuación vectorial de la forma A**s**=**b**. La solución del sistema de ecuaciones descrito por la ecuación vectorial proporciona los valores de los coeficientes de cada spline cubico que describirá la totalidad de los puntos.

Analizando la forma de los spline cúbicos se tiene que:

- $f_k = a_{0,k} + a_{1,k}x + a_{2,k}x^2 + a_{3,k}x^3$ k=1,…,n
- $f_k(x_{k-1}) = y_{k-1}$
- $f'_k(x_k) = f'_{k+1}(x_k)$
- $f''_k(x_k) = f''_{k+1}(x_k)$

Donde es posible definir lo siguiente:

$$h(i) = x(i+i) - x(i)$$

Luego se calcula A, con lo cual se tiene que:

- cuando (i,j) i = j = 2(h_i+h_{i+1})
- cuando (i,j-1) = hi
- cuando (i,j+1) = h_{i+1}.

Después de esto, se procede a calcular B, con lo cual se tiene que:

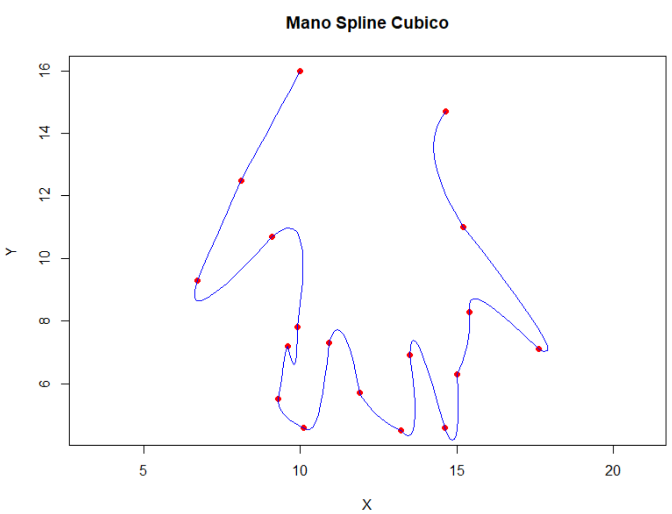
- cuando (i,1) = 6((y_i+2-y_{i+1})/ h_i+1-(y_i+1-y_i)/ h_i).

Con todos los parámetros encontrados podemos sacar los coeficientes de As=B.

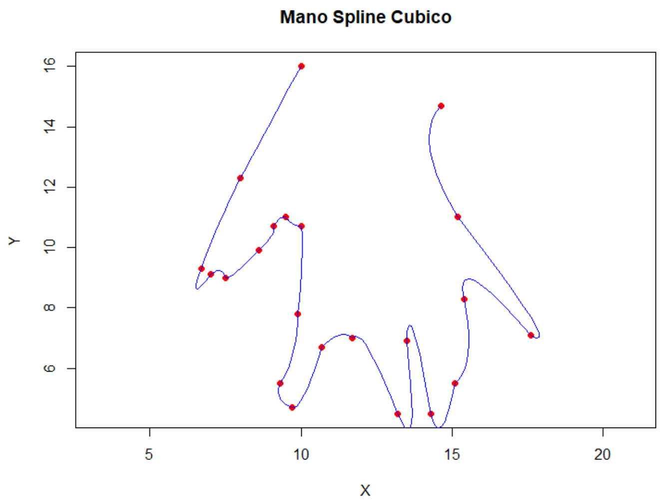
c. Selección de puntos

La selección de puntos se realizo de una manera poco práctica, ya que se tomaron de manera cualitativa respecto al comportamiento de los splines cúbicos. Se graficaron los puntos en Excel, y observando el comportamiento del spline al ingresar ciertos puntos se pudo ver que para describir tramos de la mano poco suaves era optimo elegir puntos cercanos, pero no tanto.

A continuación, se presentan 2 imágenes que se obtuvieron como resultado durante la etapa de solución.



Prueba, primera selección de puntos.



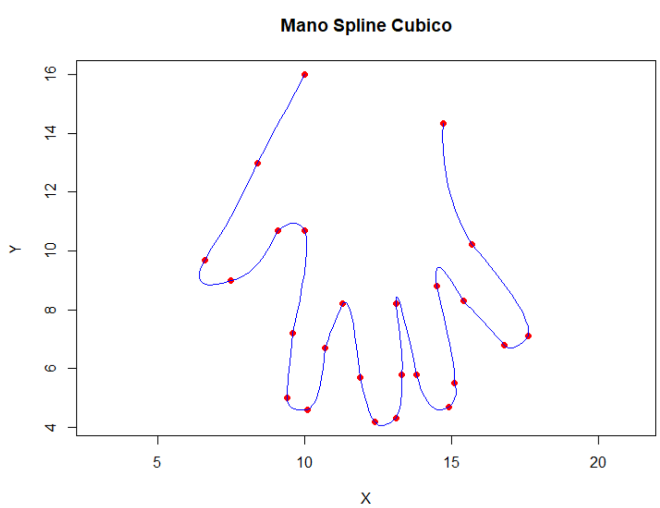
Prueba, segunda selección de puntos.

Como se puede apreciar en las anteriores imágenes es claro ver una mayor cantidad de puntos, no implica mayor exactitud. Por esta razón se concluye que lo que realmente importa son los puntos y no la cantidad.

IV. SOLUCIÓN

Para dar solución al problema se desarrolló un código en r, el cual se encuentra en el apartado de anexos. Este código recibe los puntos calcula el error en la interpolación, y encuentra los respectivos polinomios que describen los puntos ingresados.

A continuación, se presentan los resultados utilizando el código desarrollado.



Resultado final.

La anterior imagen se obtuvo de los siguientes puntos, registrados en la tabla 1.

Cantidad	x	y
1	10	16
2	8,4	13
3	6,6	9,7
4	7,5	9
5	9,1	10,7
6	10	10,7
7	9,6	7,2
8	9,4	5
9	10,1	4,6
10	10,7	6,7
11	11,3	8,2
12	11,9	5,7
13	12,4	4,2
14	13,1	4,3
15	13,3	5,8
16	13,1	8,2
17	13,8	5,8
18	14,9	4,7
19	15,1	5,5
20	14,5	8,8
21	15,4	8,3
22	16,8	6,8
23	17,6	7,1
24	15,7	10,22
25	14,71	14,33

Tabla #1

La siguiente imagen ilustra los puntos que se seleccionaron sobre los puntos que se suministraron.

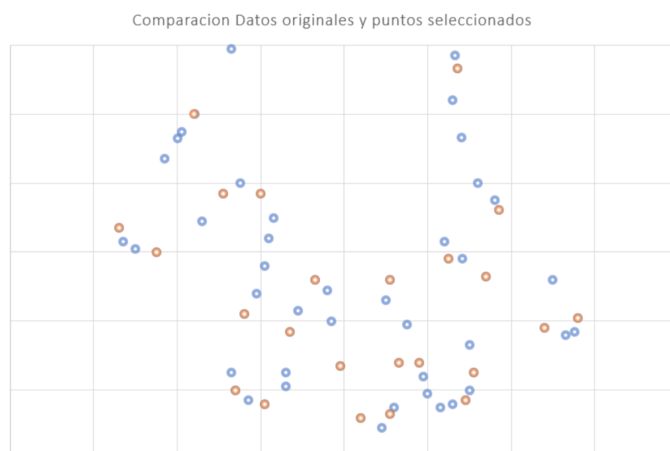


Imagen con los datos seleccionados y los datos originales

Respecto al error que posee la aproximación, dentro del código se imprime los resultados del error.

En resumen de la solución, se puede decir que se llegó a una solución con 25 puntos y una muy buena aproximación, pero el ya que la solución de este problema no involucra la escogencia de los puntos dentro del código, hace que la solución no sea universal, sino que mas bien particular, esto quiere decir para puntos con distribución similar a los puntos suministrados.

V. PREGUNTAS

- ¿Se puede cambiar el origen?

Basados en las pruebas realizadas se puede decir que, se puede cambiar el origen, pero no debe estar muy lejos este valor de los puntos en la imagen, esto a la dependencia del spline cubico a el punto especifico y no a la cantidad.

- ¿Si tenemos nueva información ósea nodos como podemos implementar esa información en el algoritmo de interpolación?

Los puntos que se ingresen deben estar en orden, y de igual forma que ocurrió antes, deben estar cerca a los puntos originales.

- ¿Su método es robusto, en el sentido que si se tienen más puntos la exactitud no disminuye?

Esta pregunta se responde directamente del análisis del comportamiento del spline cubico, ya que mas puntos no implica mas exactitud, lo que afecta directamente a la exactitud de la interpolación es la escogencia de los puntos, y debido los puntos fueron seleccionados observando el comportamiento de cada uno al ingresarlos en el spline se concluye que el método no es robusto.

VI. CONCLUSIONES

El reto propuesto fue solucionado, pero no de forma óptima, ya que si se hubiese considerado la escogencia de los puntos dentro del software la solución se convertiría en universal, y funcionaria para cualesquiera distribuciones de datos.

Reconocer el funcionamiento de los splines nos ayudó a comprender, cuales puntos eran mas significativos a la hora de definir una serie de datos para realizar una interpolación, Hubiese sido interesante plantear un algoritmo que realizara de forma óptima la elección de los puntos con mayor importancia en la interpolación.

VII. ANEXOS

En el siguiente link se puede acceder al código que se implementó para la solución del reto.

- https://github.com/BreayannOrtiz1/Analisis-Numerico/blob/master/Mano_SplineCubico.R