

Proyecto Interpolación de un Perro

Sergio Andrés Mejía Tovar
sergio.mejia@javeriana.edu.co

Julián David Parada Galvis
julian_parada@javeriana.edu.co

30 de marzo de 2019



1. Introducción

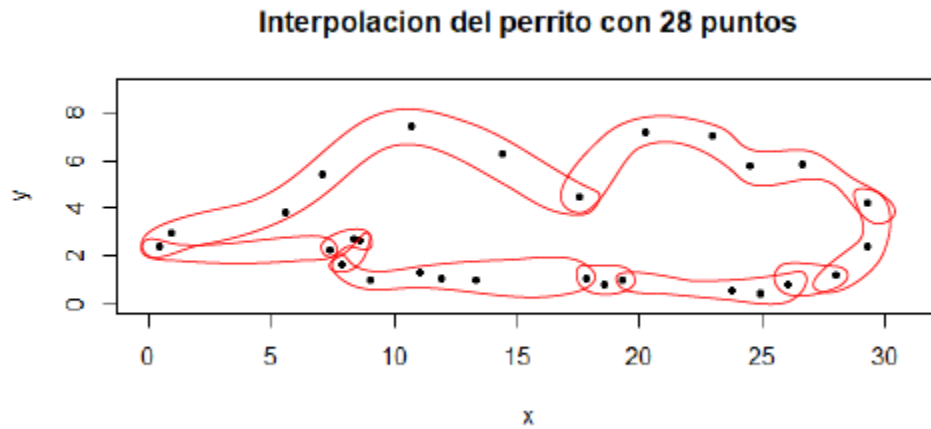
Se desarrolla una metodología con el objetivo de construir un interpolador de forma polinómica a partir de unos puntos seleccionados a través de cierta metodología buscando usar la menor cantidad de puntos que permita recrear el dibujo del contorno de un perro sin bigotes para tener una mayor exactitud, la metodología usada para la selección de puntos inicialmente fue la recreación del contorno por medio de la herramienta Illustrator CC 2019, seguidamente se interpolaron varios conjuntos de puntos con el método de Splines Cúbicos.

2. Algoritmo que se aplicó (justificación) y cómo solucionó el sistema

2.1. Codificación

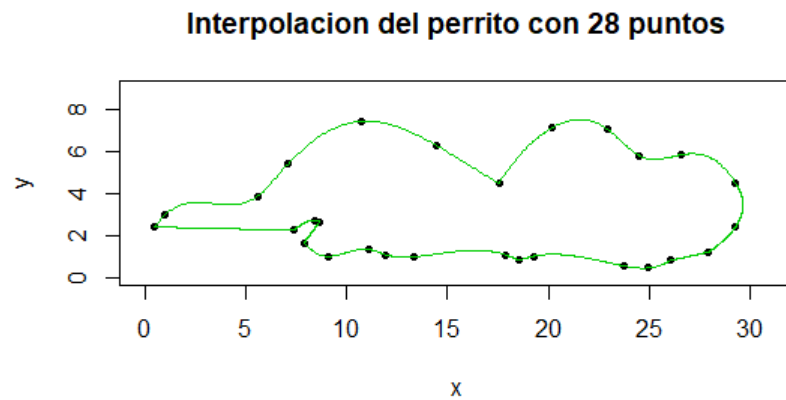
Para hallar los puntos iniciales de trabajo, se importó la imagen del Perrito en Adobe Illustrator CC 2019, colocando el tamaño del área de 8×30 unidades. Con la herramienta pluma se crearon curvas de Bézier (polinomios cúbicos que son creados usando un punto de inicio, uno de final y dos puntos de apoyo) que contornearan la imagen del perro, buscando utilizar una cantidad reducida (aunque inicialmente no es mínima, teniendo en cuenta que una curva de Bézier no es igual a un interpolador Spline). Las coordenadas de estos puntos luego fueron tomadas y pasadas a R. Se crean primeramente los arreglos de las coordenadas (x,y) de los n puntos (en este caso $n = 29$) que se van a interpolar y al final de cada arreglo se agregó el primer punto usado para poder hacer una interpolación de Spline Cúbico completa.

Se crean 10 intervalos o arreglos auxiliares de puntos (extraídos de los arreglos (x,y) creados anteriormente) de diferentes tamaños de acuerdo a las diferentes regiones críticas del contorno del perro para que al momento de aplicar el Spline sea más preciso aún.



Selección de puntos por Spline

Seguidamente se muestran los puntos y intervalo a intervalo se le aplica el Spline y se va mostrando sobre el conjunto de puntos que le corresponde hasta que con todos los intervalos auxiliares y sus Splines completan el contorno del perro.



Interpolación final del perro

2.2. Calcular la cota de error

El calculo del error se realizo con los puntos dados originalmente en el planteamiento del problema, estos se compararon con el valor que nos daba la imagen de cada punto en la función genera por el Spline correspondiente.

x	y	f(x)	Error
1.00	3.00	2.989	0.355 %
2.00	3.70	3.559	3.807 %
5.00	3.90	3.584	8.090 %
6.00	4.50	4.231	5.968 %
7.50	5.70	5.826	2.227 %
8.10	6.69	6.347	5.116 %
10.00	7.12	7.336	3.036 %
13.00	6.70	6.994	4.391 %
17.60	4.45	4.451	0.037 %
17.60	4.45	4.565	2.590 %
20.00	7.00	7.026	0.372 %
23.50	6.10	6.571	7.726 %
24.50	5.60	5.804	3.652 %
25.00	5.87	5.680	3.229 %
26.50	5.15	5.854	13.589 %
27.50	4.10	5.632	37.369 %

Tabla 1: Tabla de la cota de error de los puntos dados originalmente

También se realizó un análisis del error con los puntos que originalmente nos dio Illustrator CC 2019 comparándolos con los que finalmente usamos.

x	y	f(x)	Error
0.4275	2.24	2.342	4.581 %
1.387	3.43	3.295	3.930 %
5.1395	3.53	3.634	2.955 %
7.049	5.61	5.369	4.291 %
10.906	7.59	7.456	1.757 %
17.442	4.59	4.547	0.919 %
21.413	7.61	7.570	0.524 %
23.5885	6.48	6.490	0.159 %
25.061	5.74	5.677	1.091 %
26.543	5.91	5.850	1.006 %
28.652	5.17	4.947	4.299 %
29.2885	3.34	29.552	0.902 %
27.9775	1.15	1.21	5.217 %
25.935	0.69	0.760	10.178 %
25.5645	0.55	0.612	11.409 %
25.2605	0.59	0.513	13.002 %
24.8995	0.44	0.440	0 %
24.491	0.46	0.431	6.206 %
23.7215	0.52	0.540	3.846 %
19.266	1.01	1.012	0.276 %
18.563	0.8	0.800	0 %
17.86	1.08	1.080	0 %
13.338	1.01	0.980	2.970 %
11.723	1.08	1.143	5.851 %
10.8585	1.33	1.336	0.509 %
9.044	1.01	1.005	0.481 %
7.8375	1.64	1.674	2.099 %
8.512	2.81	2.683	4.487 %
6.5265	2.24	2.260	0.909 %

Tabla 2: Tabla de la cota de error de los puntos dados por Illustrator CC 2019

El análisis de la cota de error se realizó por medio de dos formas, la primera fue de forma experimental graficando los errores que ya se tienen de la comparación de los puntos inicialmente dados del Illustrator CC 2019 con los puntos finalmente interpolados y la segunda fue realizando la comparación de los errores con la distribución Normal que nos permite hacer la función `qqnorm` en R.

Con el primer análisis de la cota de error experimental se tiene que esta es 13 %:

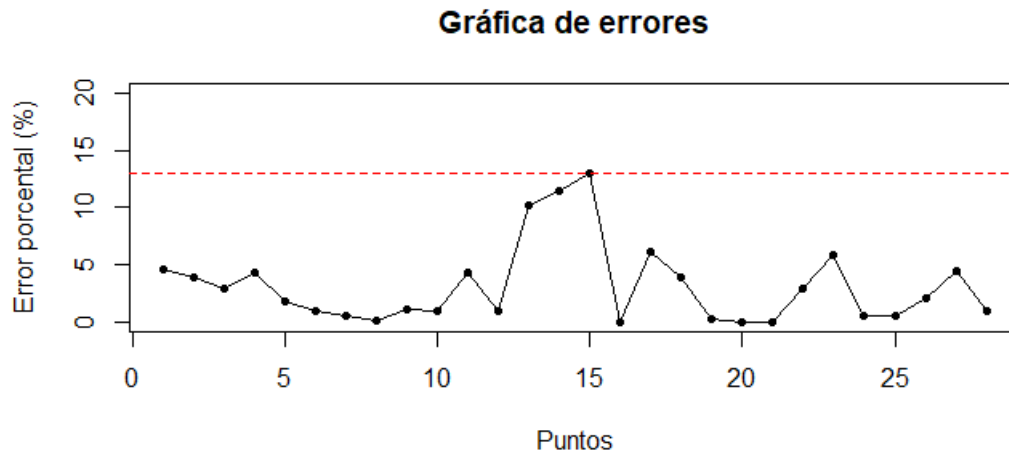


Imagen mostrando el error de cada punto interpolado y la cota de error experimental

El segundo cálculo fue utilizando una revisión por normalización utilizando el comando `qqnorm()` de R. Así, se confrontan los puntos a la distribución normal y a las varianzas del cálculo, con la esperanza que los resultados de los cuantiles sea un crecimiento constante. Cuanto más constante sea el crecimiento, se puede decir que los errores están distribuidos normalmente y son aceptables. Como se puede observar en la siguiente gráfica, se observa una relación aproximadamente constante de crecimiento, por lo que podemos decir que los errores no son muy grandes o erráticos.

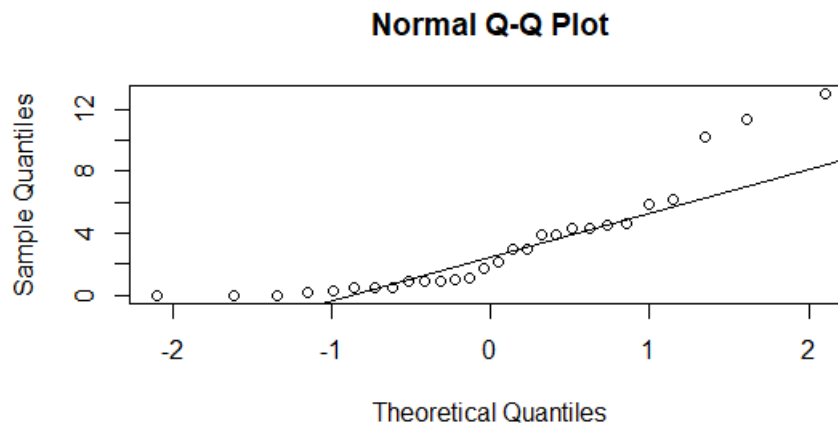


Imagen mostrando la comparación con la distribución normal

2.3. Comparacion con el metodo Lagrange Baricentrico

Se realizó una prueba de interpolar la imagen del perro usando el método de Lagrange Baricentrico obteniendo una imagen del perro un poco extraña, razón por la cual desde el primer momento se decidió

usar el método de Spline Cubico para el desarrollo final del proyecto:

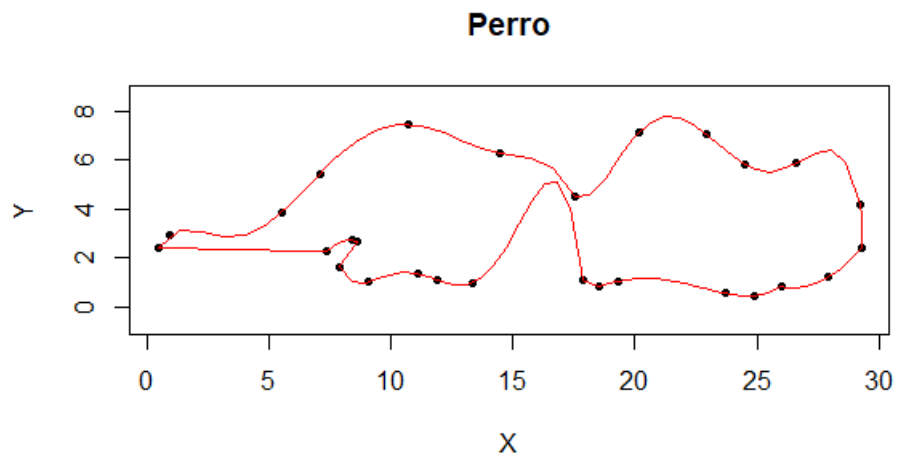


Imagen mostrando el resultado de la interpolación usando el método de Lagrange Baricentrico

Se compararon los errores entre los metodos de Lagrange Baricentrico y Spline Cubicos encontrando que la cota de error experimental aumento hasta 57 cuando se usaba el método de Lagrange Baricentrico:

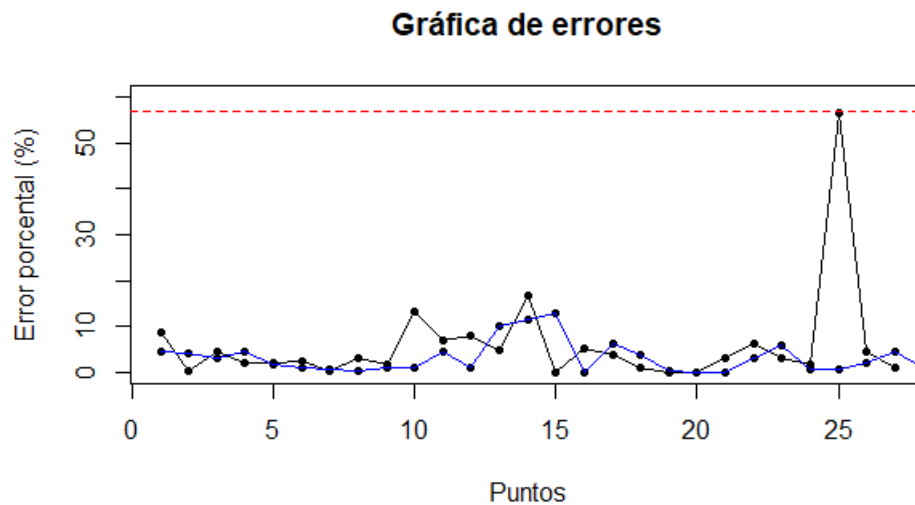


Imagen mostrando la comparación de los errores entre el método de Splines Cubicos (azul) y el de Lagrange Baricentrico (negro)

3. Validación del resultado

3.1. Tabla de los valores interpolados

i	x	y	i	x	y	i	x	y
1	0.48	2.40	12	29.27	4.50	23	11.10	1.33
2	0.96	2.95	13	29.27	2.40	24	9.07	1.00
3	5.56	3.86	14	27.93	1.20	25	7.89	1.64
4	7.09	5.41	15	26.03	0.80	26	8.65	2.65
5	10.71	7.45	16	24.90	0.44	27	8.41	2.70
6	14.44	6.30	17	23.72	0.54	28	7.41	2.24
7	17.54	4.49	18	19.30	1.01			
8	20.19	7.15	19	18.56	0.80			
9	22.94	7.05	20	17.86	1.08			
10	24.51	5.80	21	13.34	0.98			
11	26.6	5.85	22	11.91	1.08			

Tabla 3: Tabla de los valores a interpolar

3.2. Función del numero de aciertos y de diferencias (Índice de Jaccard)

El cálculo de número de aciertos relacionando los puntos elegidos con los puntos teóricos iniciales dados por la profesora es en este caso poco útil puesto que los puntos elegidos fueron hallados con otro métodos completamente diferente, por lo que no tiene mucho sentido realizar esta comparación. Sin embargo, por el bien de realizar la mayor cantidad de operaciones, se muestra a continuación.

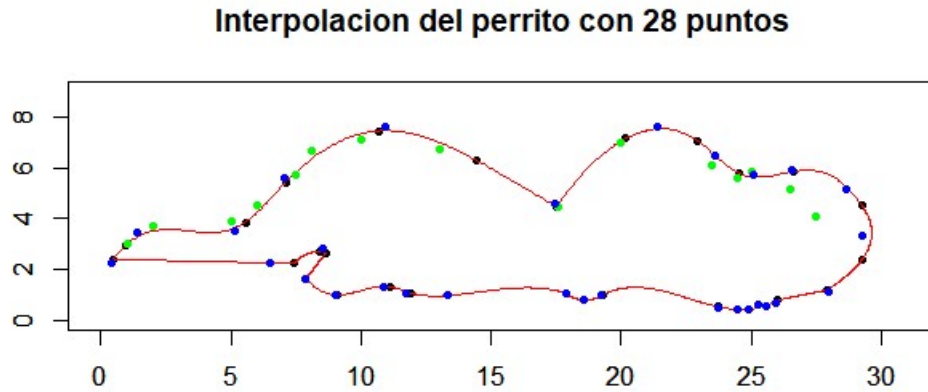


Imagen mostrando los puntos teóricos (verdes) y el perro interpolado

Así, se hizo un índice de Jaccard a nivel de un decimal (es decir, que el punto es válido si $x_e - 0,1 \leq x_t \leq x_e + 0,1$ y $y_e - 0,1 \leq y_t \leq y_e + 0,1$) observando el número de puntos elegidos por el otro método que se asemejaron a los teóricos. Así, se encontró el número de aciertos y de fallos:

Puntos totales	15
Aciertos	2
Fallos	13
Índice (Aciertos/Totales)	13.33 %

Tabla 4: Cálculo del Índice de Jaccard para los puntos dados originalmente

Del mismo modo que al momento del cálculo del error se tomó los puntos dados por la profesora y los puntos iniciales generados por Illustrator, para este Índice se realizará lo mismo. En la siguiente se muestra el índice de Jaccard para los puntos de Illustrator:

Puntos totales	28
Aciertos	11
Fallos	18
Índice (Aciertos/Totales)	39.28 %

Tabla 5: Cálculo del Índice de Jaccard para los puntos de Illustrator

El valor que se encuentra es bajo, pero esto no indica la confiabilidad de los puntos, sino el grado de similitud entre los dos conjuntos. Así, se observa que es bajo ya que los puntos difieren bastante de los originales, aunque resultando en la interpolación en una precisión mayor a los originales. En el caso del Índice relacionado a los puntos de Illustrator, es relativamente bajo (aunque más alto que con los originales de la profesora) ya que se reubicaron múltiples puntos para aumentar la precisión.

3.3. Función que muestra la eficiencia del método

Para el análisis de la eficiencia del método implementado se realizó una comparación entre el número de operaciones que realiza nuestro método (Splines cúbicos) comparado con el método de Lagrange Baricéntrico. Esto probándose en diferentes escenarios con diferentes particiones sobre los puntos tomados.

Se sabe de antemano que según la teoría de Lagrange Baricentrico el número de operaciones que realiza este método en su algoritmo es de $8n - 2$ y que para el método de Splines Cubicos realiza $26n - 32$, donde n para ambos casos es el numero de puntos ingresados.

Escenario	Entrada (particiones)	Número de operaciones
Barylag arriba y abajo	2	236
Barylag particionado	10	284
Spline 1	1	722
Spline arriba y abajo	2	716
Spline particionado	10	688

Tabla 6: Comparación del número de operaciones entre métodos y escenarios.

4. Preguntas

4.1. ¿El origen se puede modificar?

Sí se puede modificar el origen desde donde se comenzó a hacer la interpolación de Splines Cúbicos. Se puede cambiar el origen con la condición de que el origen sea cambiado únicamente por otro nodo en el que comience la elaboración de otro Spline. Esto con el fin de no disminuir la precisión o de tener la necesidad de crear nuevos splines.

4.2. Si se tiene nueva información, ¿cómo se puede implementar esa información en el algoritmo de interpolación?

Se puede integrar a uno de los splines existentes en el modelo actual, teniendo cuidado de cambiar los límites del intervalo de cada spline, así como de observar posibles disminuciones de precisión debido a que el contorno del perro no es una función sino una relación. Partes como el hocico y la curvatura que genera la pata trasera son los mayores lugares a tener cuidado.

4.3. ¿Su método es robusto? En el sentido que: ¿Si se tienen más puntos la exactitud no disminuye?

Como se mencionó en la pregunta anterior, en general el método sí es robusto, puesto que la interpolación por Splines Cúbicos facilita esto. Sin embargo, no se debe olvidar que el contorno es una relación, por lo que se tiene que tener especial cuidado en las zonas donde esto puede notarse con mayor magnitud.

4.4. Suponga que se tienen más puntos con más cifras significativas. ¿Cómo se comporta su algoritmo? ¿La exactitud decae?

Si se tienen aún más puntos puede mejorar la exactitud de la interpolación si se sabe bien dónde ubicar los puntos de forma estratégica en el Spline que le corresponde, y si además los puntos tienen más cifras significativas podría mejorarse un poco más la exactitud pero no se vería tan reflejada en la gráfica por la escala que se maneja. Sin embargo, se debe tener en cuenta que si los puntos son muy cercanos o muy lejanos el Spline va a tender a tener un comportamiento errático y el error aumentará en gran magnitud.

5. Resultado final

