



UNIVERSIDAD DE  
GUANAJUATO

Programación

**Curvas paramétricas: curvas de Bézier**

Jorge H. Martínez González

Noviembre, 2022

# Índice

<b>1. Introducción</b>	<b>2</b>
<b>2. Curvas de paramétricas</b>	<b>3</b>
2.1. Curvas de Bézier . . . . .	3
2.2. Algoritmo . . . . .	6
2.3. Aplicación y desventaja: Reconstrucción geométrica basada en tomografías computacionales.	7

## 1. Introducción

Consideremos el siguiente ejemplo.

Se realiza un censo de la población de Estados Unidos cada 10 años. La siguiente a continuación, se muestra la población, en miles de personas, desde 1960 hasta 2010.

Año	1960	1970	1980	1990	2000	2010
Población	179323	203 302	226542	249 633	281 422	308 746

Cuadro 1: Censo de la población de Estados Unidos cada 10 años.

Al revisar los datos, surge el problema de investigar si se podrían usar para realizar un cálculo razonable, por ejemplo, de la población en 1950 o 2020. Este tipo de aproximaciones pueden obtenerse por medio de una función que se ajuste a los datos proporcionados. Este proceso se conoce como *interpolación*.

Existen distintos métodos que abordan este problema. Entre algunos esta, por ejemplo, la interpolación mediante polinomios.

**Proposición 1.1.** *Dados  $n + 1$  puntos  $\{(x_i, y_i)\}_{i=1}^{n+1}$  que cumplen que  $x_i \neq x_j$  si  $i \neq j$ , existe un único polinomio  $p_n(x) = \sum_{n=0}^n a_n x^n \in \mathbb{P}_n$  tal que*

$$p_n(x_i) = y_i \quad \forall i = 0, \dots, n$$

Algunos de métodos conocidos son, por ejemplo.

- **Polinomios de Lagrange.** Para  $i = 1, \dots, n$  definimos los polinomios de Lagrange

$$L_{n,i}(x) \in \mathbb{P}_n, \quad L_{n,i} = \prod_{j=0, j \neq i}^n \frac{x - x_j}{x_i - x_j}.$$

Sea que  $L_{n,i}(x_i) = 1$  y  $L_{n,i}(x_j) = 0$  si  $i \neq j$ . Entonces el polinomio

$$p_n(x) = \sum_{j=0}^n y_j L_{n,j}(x)$$

interpola al conjunto de datos.

- **Polinomios de Newton.**

Así como la opción de la **interpolación mediante splines**.

Cada una de estos métodos tiene distintas ventajas y desventajas, tanto en la cuestión computacional, como en el error de interpolación. En el presente trabajo, presentamos una descripción de un otro método que responde a algunos de los problemas de los anteriores métodos, **las curvas de Bézier**.

Pierre Etienne Bézier (1910 - 1999) fue un director de diseño y producción de los automóviles Renault durante la mayor parte de su vida profesional. Comenzó su investigación en diseño y fabricación asistidos por computadora en 1960, desarrollando herramientas interactivas para el diseño de curvas y superficies, e inició el bobinado generado por computadora para modelado de automóviles. Las curvas que llevan su nombre tiene la ventaja de estar basadas en una teoría matemática rigurosa que no necesita ser reconocida por el practicante.

## 2. Curvas de paramétricas

Ninguno de los métodos mencionados anteriormente se puede usarse para generar curvas como la presentada en la figura(1), pues esta curva no se puede expresar como una función de una variable en términos de la otra.

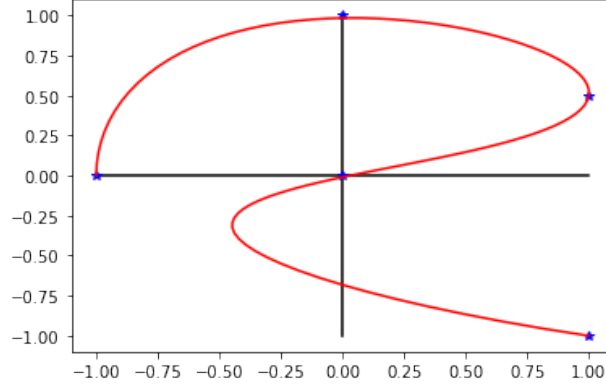


Figura 1: curva en forma paramétrica

Una forma de abordar este problema es considerar las curvas paramétricas, en las que cada componente de la curva, escrita en forma paramétrica es aproximado por un polinomio. Consideramos una curva en el plano en forma paramétrica  $P(t) = ((x(t), y(t)))$ , con  $t \in [0, T]$ , luego, se toma el conjunto de puntos en el plano de coordenadas  $P_i = (x_i, y_i)$  para cada  $i = 1, \dots, n$ , y se introduce una partición en el intervalo  $[0, T] = 0 = t_0 < t_1 < \dots, < t_n = T$ .

Usando ahora el los dos conjuntos de valores  $\{t_i, x_i\}_{i=0}^n$  y  $\{t_i, y_i\}_{i=0}^n$  y construir funciones de aproximación ellos.

### 2.1. Curvas de Bézier

Sean  $P_0, P_1, \dots, P_n$  un conjunto de  $n+1$  puntos plano. El polígono orientado formado por estos conjuntos es llamado el *polígono de característico* o el *polígono de Bézier*. En este conjunto de puntos, el primero y el último son siempre los extremos de la curva de Bézier, sin embargo, en general los puntos intermedios no caen en la curva.

Definamos los polinomios de Bernstein.

**Definición 2.1** (Polinomios de Bernstein).  $P_0, P_1, \dots, P_n$  un conjunto de  $n+1$ . El  $n$ -ésimo **polinomio de Bernstein** sobre el intervalo  $[0, 1]$  de  $f$  es el polinomio

$$b_{n,k}(t) = \binom{n}{k} t^k (1-t)^{n-k} = \frac{n!}{k!(n-k)!} t^k (1-t)^{n-k},$$

para  $t \in [0, 1]$ ,  $n \in \mathbb{N} \cup \{0\}$  y  $k = 0, 1, \dots, n$ .

Notemos que ellos pueden ser obtenidos de la siguiente manera recursiva

$$\begin{cases} b_{n,0} &= (1-t)^n \\ b_{n,k} &= (1-t)b_{n-1,k}(t) + tb_{n-1,k-1}(t), \quad k = 1, 2, \dots, n; t \in [0, 1]. \end{cases}$$

Se cumple que  $b_{k,n} \in \mathbb{P}_n$  para  $k = 0, 1, \dots, n$ , de hecho,  $\{b_{n,k}\}_{k=0}^n$  forma una base para  $\mathbb{P}_n$ .

**Definición 2.2** (Curvas de Bézier). La *curva de Bézier* se define como

$$B_n(P_0, \dots, P_n, t) := \sum_{k=0}^n P_k b_{n,k}(t), \quad t \in [0, 1]. \quad (2.1)$$

La expresión anterior se puede pensar como un promedio con peso de los puntos  $P_k$ , con pesos  $b_{n,k}$ . Veamos a continuación algunos ejemplos.

■ Curvas de Bézier Lineal.

Dados dos puntos distintos  $P_0, P_1$  la curva de Bézier entre ellos es la línea recta entre ellas expresada de forma paramétrica, es decir

$$B(t) = P_0 - t(P_1 - P_0) = (1 - t)P_0 + tP_1$$

con  $t \in [0, 1]$ .

■ Curvas de Bézier cuadráticas.

Dado los puntos  $P_0, P_1, P_2$ , las curvas resultante es dada por la expresión

$$B(t) = (1 - t)[(1 - t)P_0 + tP_1] + t[(1 - t)P_1 + tP_2], \quad t \in [0, 1].$$

qué puede ser pensado como la interpolación lineal de los correspondientes puntos de las curvas de Bézier de  $P_0$  a  $P_1$  y de  $P_1$  a  $P_2$  respectivamente. Simplificando la ecuación anterior se tiene

$$B(t) = (1 - t)^2 P_0 + 2(1 - t)t P_1 + t^2 P_2,$$

o escrita de manera que se pueda ver cierta simetría con respecto a  $P_1$

$$B(t) = P_1 + (1 - t)^2(P_0 - P_1) + t^2(P_2 - P_1).$$

De donde podemos ver la derivada de la curva con respecto a  $t$

$$B'(t) = 2(1 - t)(P_1 - P_0) + 2t(P_2 - P_1)$$

de donde podemos ver que la tangente a la curva en los puntos  $P_0$  y  $P_2$  intersectan a  $P_1$ . Además, vemos que, mientras  $t$  aumenta de 0 a 1, la curva surge de  $P_0$  con dirección a  $P_1$ , luego, cambia de dirección y llega a  $P_2$  de la dirección de  $P_1$ .

■ Curvas cúbicas de Bézier.

Dados los puntos  $P_0, P_1, P_2, P_3$  en el plano, ellos determinan una curva cúbica. La curva comienza en  $P_0$  en dirección  $P_1$  y llega a  $P_3$  de la dirección de  $P_2$ . En general, no se da que la curva pasa por  $P_1$  y  $P_2$ , estos puntos proporcionan información direccional. La distancia entre  $P_1$  y  $P_2$  determina "que tanto" va la curva hacia  $P_1$  antes de cambiar su dirección hacia  $P_2$ .

Entonces, la curva de Bézier entre estos puntos, se puede escribir como la curva lineal de dos curvas de Bézier cuadráticas, como a continuación

$$B(t) = (1 - t)B(P_0, P_1, P_2, t) + tB(P_1, P_2, P_3), \quad t \in [0, 1].$$

La forma explícita es

$$B(t) = (1 - t)^3 P_0 + 3(1 - t)^2 t P_1 + 3(1 - t)t^2 P_2 + t^3 P_3.$$

Según la selección de puntos, la curva se puede intersectar a sí misma o contener un punto singular. Dados cuatro puntos se puede construir una curva de Bézier que pase por los cuatro puntos, dando los puntos el punto inicial y final de la curva, y los puntos correspondientes a  $t = 1/3$  y  $t = 2/3$ , se puede recuperar los puntos de control originales de la curva.

Veamos a continuación un ejemplo sencillo de la construcción de una curva.

**Ejemplo 2.1.** Las curvas de Bézier se pueden pensar de manera totalmente geométrica comenzando por el polígono característico. Para cada  $t \in [0, 1]$  fijo, definimos  $P_{i,1}(t) = (1-t)P_i + tP_{i+1}$  para  $i = 0, \dots, n-1$ . Este nuevo conjunto de  $n-1$  vértices define un nuevo polígono característico. Se puede repetir entonces el proceso en para generar nuevos puntos  $P_{i,2}(t)$   $i = 0, \dots, n-2$ , y así hasta obtener únicamente los vértices  $P_{0,n-1}(t)$  y  $P_{i,n-1}(t)$ . Así, se tiene que

$$P_{0,n}(t) = (1-t)P_{0,n-1} + tP_{1,n} = B(P_0, \dots, P_n, t).$$

La figura (2) muestra el proceso descrito con el polígono de dado por los puntos, en orden  $\{(0, 1), (0, 0), (1, 0)\}$  para  $t = 0,5$ .

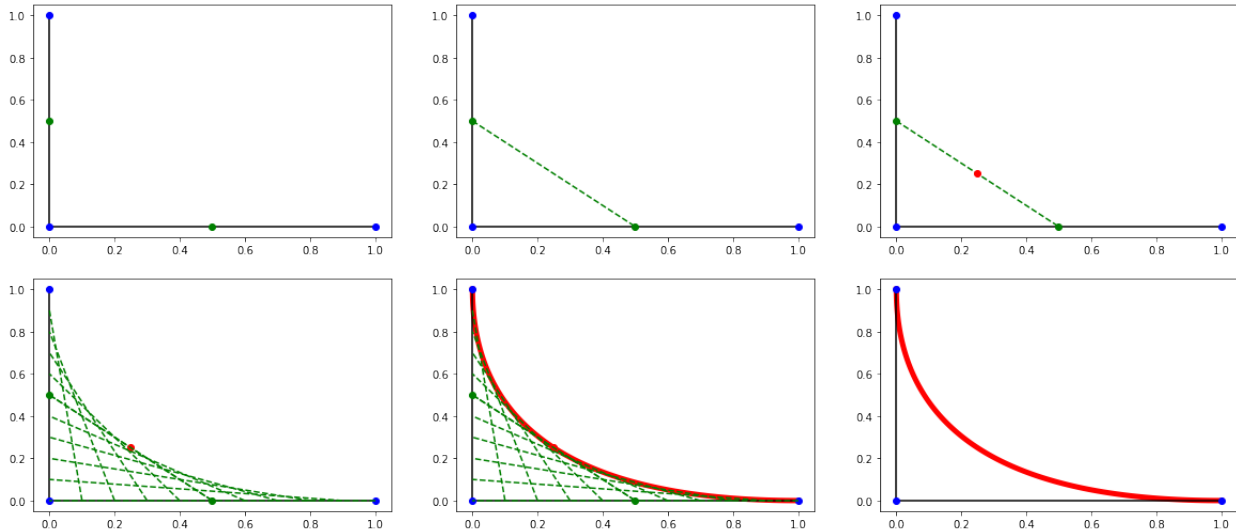
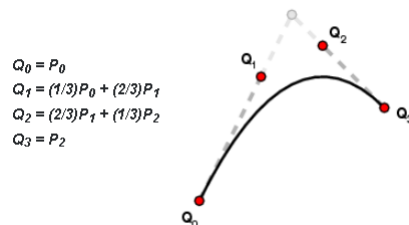


Figura 2: Construcción de la curva de Bézier entre los puntos  $(0, 1), (0, 0), (1, 0)$  para  $t = 0,5$ .

Algunas de las propiedades de las curvas de Bézier son se presentan a continuación

- La curva es una línea recta si y sólo si los puntos dados son colineales.
- El inicio y el final de la curva son tangentes, respectivamente, a la primera y última sección del polinomio característico.
- Cada curva puede ser dividida en dos o mas subcurvas, cada una de las cuales es en sí una curva de Bézier.
- Algunas curvas que parecen simples, como el círculo, no pueden ser exactamente por una curva de Bézier, para ello, por ejemplo, se usan cuatro curvas de Bézier que aproximan cada una un arco de 90 grados de un círculo.
- Una curva de Bézier de algún grado puede ser descrita como una curva de un grado más alto, por ejemplo, la curva cuadrática de Bézier con puntos de control  $P_0, P_1, P_2$  es la misma que la curva de Bézier cúbica con puntos



## 2.2. Algoritmo

En la figura (3) y (4) los algoritmos necesarios para las curvas de Bezier, tomados de (1):

### Program 69 - bernstein : Bernstein polynomials

```
function [bnk]=bernstein (n,k,x)
if k == 0,
    C = 1;
else,
    C = prod ([1:n])/( prod([1:k])*prod([1:n-k]));
end
bnk = C * x^k * (1-x)^(n-k);
```

Figura 3: Algoritmo que evalúa  $b_{n,k}$  en el punto  $x$  para  $x \in [0, 1]$ .

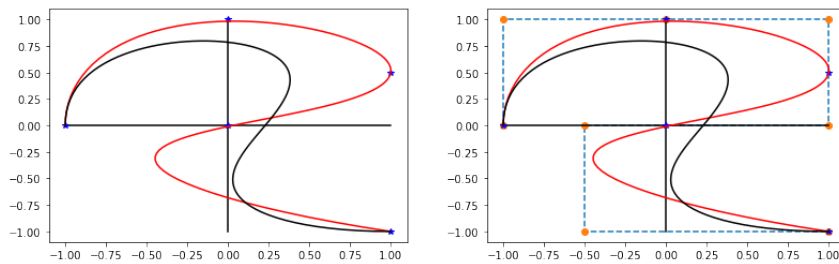
### Program 70 - bezier : Bézier curves

```
function [bezx,bezy] = bezier (x, y, n)
i = 0; k = 0;
for t = 0:0.01:1,
    i = i + 1; bnk = bernstein (n,k,t); ber(i) = bnk;
end
bezx = ber * x (1); bezy = ber * y (1);
for k = 1:n
    i = 0;
    for t = 0:0.01:1
        i = i + 1; bnk = bernstein (n,k,t); ber(i) = bnk;
    end
    bezx = bezx + ber * x (k+1); bezy = bezy + ber * y (k+1);
end
plot(bezx,bezy)
```

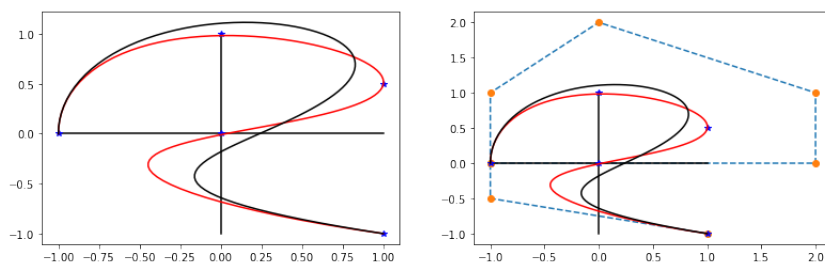
Figura 4: Algoritmo que imprime la curva de Bézier relativa al conjunto de puntos  $(x, y)$ .

Usando este código veamos el siguiente ejemplo.

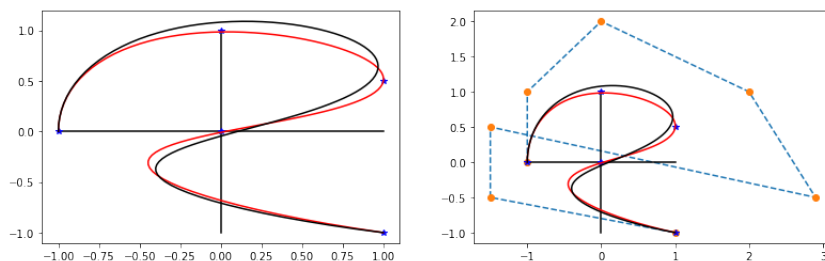
**Ejemplo 2.2.** Consideremos la figura (1) presentada al comienzo, la figura (5) muestran un intento de aproximación usando el código anterior.



(a) Primer intento de aproximar la curva de la figura (1).



(b) Segundo intento de aproximar la curva de la figura (1).



(c) Tercer intento de aproximar la curva de la figura (1).

Figura 5: Uso de las curvas de Bézier. A la derecha se incluyen los puntos de control y el polinomio de Bézier usados.

### 2.3. Aplicación y desventaja: Reconstrucción geométrica basada en tomografías computacionales.

Tratemos el siguiente ejemplo tomado de (1) Una aplicación típica, que además muestra una de las desventajas de las curvas de Bézier en la interpolación, trata sobre la reconstrucción de estructuras tridimensionales de órganos internos del cuerpo humano basado en *Tomografías Computacionales [TC]*.

Las *TC* usualmente proporciona una secuencia de imágenes que representan las secciones de un órgano en varios planes horizontales. Como convención, se dice que las *CT* produce secciones del plano  $x, y$  en correspondencia a varios valores de  $z$ . El resultado es análogo al que obtendríamos al seccionar el órgano en distintos valores de  $z$  y tomásemos una fotografía de la sección correspondiente. Uno de las ventajas de las *CT* es que el órgano en investigación puede ser observada sin estar oculta por órganos vecinos, como ocurre en otras imágenes médicas, como pasa por ejemplo en las angiografías.



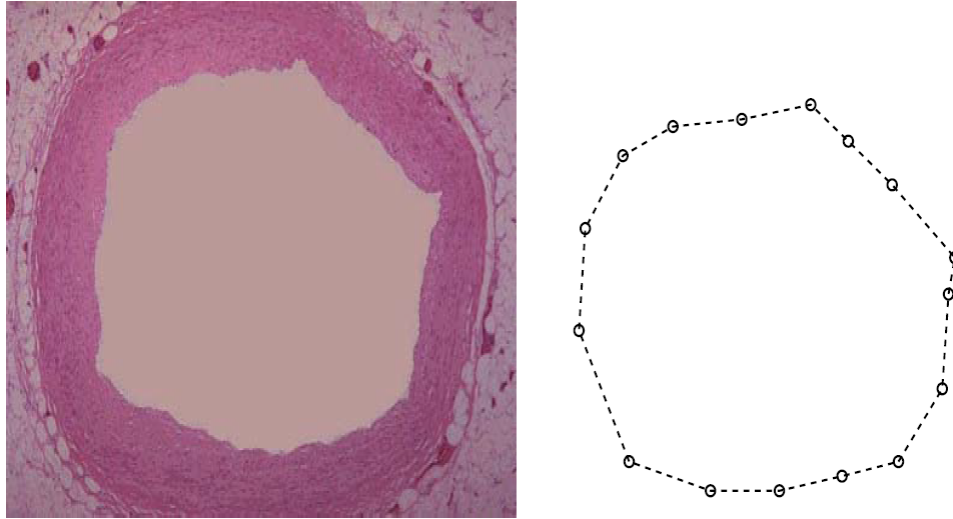


Figura 6: Sección transversal de un vaso sanguíneo (izquierda) y su polígono característico asociado usando 16 puntos  $P_1$  (derecha).

La imagen que se obtiene para cada sección es almacenada en una matriz de *píxeles* en el plano  $x, y$ , un cierto valor esta asociado con cada píxel que expresa el nivel de gris de la imagen en ese punto. Este nivel esta determinado por la densidad de rayos  $X$  que son recogidos por un detector luego de pasar por el cuerpo humano. En la práctica la información de una  $CT$  para algún valor  $z$  es expresado por un conjunto de puntos  $(x_i, y_i)$  que identifican los límites del órgano.

Para mejorar el diagnóstico, usualmente es reconstruir la figura tridimensional del órgano examinado dadas las secciones proporcionadas por las  $CT$ . Para ello, es necesario convertir la información codificada por píxeles a una representación paramétrica que puede ser expresada por funciones adecuadas que interpolan la imagen en algunos de sus puntos límites. La figura (7) muestra este proceso hecho con distintas formas de interpolación, específicamente, este ejemplo muestra que las curvas de Bézier no aproximan al polígono característico. Es por ello que en la práctica, estas curvas no son raramente usadas en interpolación ya que ellas no proporcionan una aproximación suficientemente precisa al polígono característico.

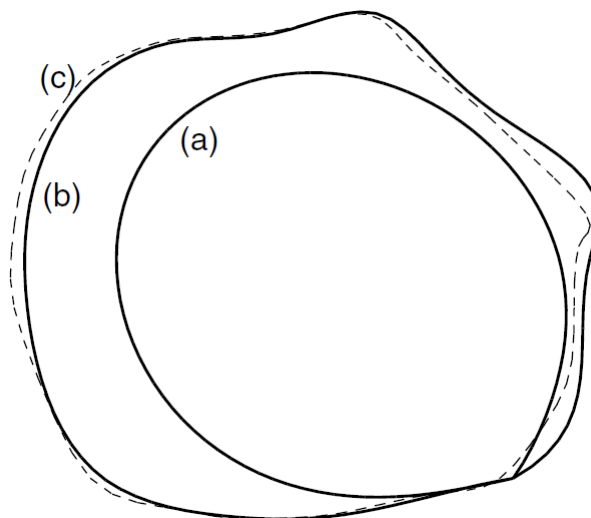


Figura 7: Reconstrucción del vaso interno de la figura (6) usando diferentes métodos de interpolación con el mismo polígono característico: a) Curvas de Bézier, b) splines paramétricos, c) B-splines paramétricos.

## Referencias

- [1] Quarteroni, A., Sacco, R. and Saleri, F., 2010. Numerical mathematics. Berlin: Springer.
- [2] Burden, R. L., Faires, D. J., & Burden, A. M. (2017). Análisis numérico (10a. Ed.). CENGAGE Learning.
- [3] Columna de AMS. American Mathematical Society. (n.d.). Consultado en Noviembre 1, 2022, de <https://www.ams.org/publicoutreach/feature-column/fcarc-bezier#2>