

Dibujar el polinomio de Newton frente a los puntos del soporte en R

Objetivos

El documento consiste en una explicación detallada(basada en la presentación de Virginia) sobre cómo funciona el código para dibujar el polinomio de Newton frente a los puntos de soporte.

Además contiene todos los códigos necesarios y un ejercicio para comprobar lo aprendido. Todos los programas vienen linkeados en los títulos.

dibujar_canonicos

Dibuja una gráfica con el polinomio interpolador de Newton

[dibujar_canonicos](#)

Esta función incluye dentro otra función para hallar el polinomio interpolador de forma similar a como lo hace la función *polinomio_newton_canonico*.

Parámetros de entrada: serían los puntos del soporte y los coeficientes del polinomio de Newton que se hallaron con *polinomio_newton_canonico*.

Análisis de código: En primer lugar “marca” cual es el valor máximo y mínimo de los puntos de soporte con las funciones `max` y `min`, esto le servirá para crear la secuencia de puntos a dibujar. Utilizando `seq` crea un vector con valores de x para una curva suave, partiendo del valor mínimo al máximo y creando mil puntos equiespaciados dentro de ese intervalo, es importante dibujar muchos valores para que al unir las coordenadas la gráfica quede lo mejor posible. Además, halla el rango (`x_max - x_min`) para hacer la operación (`x_min - 0.1*rango, x_max + 0.1*rango`), así consigue separar el polinomio de los ejes. Finalmente, las coordenadas x quedan como:

```
x <- seq(x_min - 0.1*rango, x_max + 0.1*rango, length.out = 1000)
```

Al incluir una función para hallar el polinomio, el siguiente paso sería hallar los valores del polinomio para cada punto del soporte, en otras palabras, evaluar los puntos.

```

# Crear una función para el polinomio
polinomio <- function(x) {
  suma <- 0
  for (i in 1:length(coeficientes)) {
    suma <- suma + coeficientes[i] * x^(i-1)
  }
  return(suma)
}

```

Con `y <- sapply(x, polinomio)` llama a la función de la imagen y evalúa los puntos. Un ejercicio que podría ser interesante es hallar las coordenada llamando a anteriores funciones como *polinomio_newton_canonico*.

El código para dibujar la tabla nos sirve de modelo para cualquier otra gráfica que tengamos que hacer. Para representar los puntos es fundamental la función `plot` a la cual hay que darle cuatro parámetros, los puntos del eje de abscisas y de ordenadas y luego otros dos parámetros para cuestiones estéticas que no nos importan mucho.

```

# Crear la gráfica
plot(x, y, type = "l", col = "blue",
     main = "Gráfica del Polinomio Canónico",
     xlab = "x", ylab = "y",
     xlim = c(min(x), max(x)), ylim = range(y))

# Añadir puntos para el soporte original
points(x_orig, sapply(x_orig, polinomio), col = "red", pch = 19)
}

```

Ejercicio

Dibuja una gráfica que represente el polinomio interpolador de Newton para los puntos de soporte (2,4,6) y la función $y = e^x + 5$.

Solución
