



e.Rrados

RESOLUCIÓN Y EXPLICACIONES DEL EJERCICIO PROPUESTO EN R – INTERPOLACIÓN DE LAGRANGE

¡Atención!: En esta práctica que a continuación se detalla se va a emplear la interpolación de Lagrange, por lo que es importante tener nociones previas.

En primer lugar, es menester ponerse en situación: se tienen, como datos, la concentración (expresada en gramos por litro) de bioetanol en función del tiempo (en horas). Esos datos se recogen en la siguiente tabla.

Tiempo (horas)	Concentración de bioetanol (g/l)
1	10
3.5	20
5	35.33
10	35.5

Tabla 1

Vamos a almacenar esos datos en dos vectores: s y C (para tiempo y concentración, respectivamente)

```
s<-c(1, 3.5, 5, 10)
C<-c(10, 20, 35.33, 35.5)
```

Figura 1

Lo que nos pide el ejercicio es que mediante un script, y haciendo uso de la interpolación de Lagrange, estimemos la concentración de bioetanol en los instantes $t=2$ y $t=6.5$, que son datos no conocidos a juzgar por los datos que tenemos en la tabla. Se usa el polinomio interpolador de Lagrange, que se proporciona en el enunciado:

$$p = \sum_{i=1}^n C_i L_i$$

donde:

- L_i denota la función de base de la interpolación de Lagrange.
- C_i denota la concentración de bioetanol para cada uno de los instantes de tiempo.

Es preciso recordar ahora **qué es el polinomio interpolador de Lagrange y cuándo se utiliza.**

Las funciones de base están definidas por la siguiente expresión:

$$L_i = \prod_{\substack{j=1 \\ j \neq i}}^n \frac{t - s_j}{s_i - s_j} \quad (i = 1, 2, \dots, n)$$



e.Rrados

RESOLUCIÓN Y EXPLICACIONES DEL EJERCICIO PROPUESTO EN R – INTERPOLACIÓN DE LAGRANGE

Vamos a construir la función *Polbase*. Utilizamos el comando function (**¡Atención!**)

```
Polbase<-function(t,s){
  L=c(0)
  for(i in 1:length(C)){
    L[i]=1
    for(j in 1:length(C)){
      if(i!=j){
        L[i]=L[i]*(t-s[j])/(s[i]-s[j])
      }
    }
  }
  return(L)
}
```

Figura 2

Recordemos un poco ahora algunos asuntos en relación a las funciones en R. En primer lugar, es necesario recordar que en una función existen **tres elementos**: los argumentos, el cuerpo de operaciones y el resultado. El argumento puede observarse en los valores de entrada que se colocan entre paréntesis y se separan con una coma. Son aquellos que se van a utilizar pero que no se calculan. Por su parte, el cuerpo son todas las operaciones que se realizan dentro de la función. **¡Atención!** Estas operaciones deben localizarse entre corchetes `{}`. Por último, el resultado es la última expresión en ejecutarse. Es muy importante recordar también que para que las funciones *actúen* hay que llamarlas.

En nuestro caso, los argumentos son t y s . Puesto que L es un vector cuyas componentes son los valores de cada polinomio de base en el punto t , inicializamos en primer lugar el vector a 0. A continuación, abrimos un bucle *for* de manera que cuando i se desplace desde 1 hasta la longitud del vector C (donde se encontraba la concentración de bioetanol en g/L para determinados instantes t), el valor que tome $L[i]$ sea 1, dado que nos encontramos ante un productorio. Volvemos a abrir un nuevo bucle *for*, pero esta vez j se desplaza desde 1 hasta la longitud del vector C . La condición que le ponemos es que j sea distinto que i , pues si j e i se igualasen, el denominador daría 0. **¡Atención!**: **recuérdese que el *distinto* que se denota en R de la manera**. Si la condición se cumple, solamente hay que desarrollar el productorio con las operaciones que se muestran en la expresión que aparece más arriba.

A continuación, hay que programar la función *PolInte* para obtener el polinomio interpolador en el punto t . Según los datos que proporciona el enunciado, los argumentos (que irán entre paréntesis) son: el vector C (los valores conocidos de la función que se interpola) y el vector L , calculado previamente y que contiene los polinomios de base de Lagrange evaluados en el punto t .



RESOLUCIÓN Y EXPLICACIONES DEL EJERCICIO PROPUESTO EN R – INTERPOLACIÓN DE LAGRANGE

```
Polinte<-function(C, L){  
  p=0  
  for(i in 1:length(C)){  
    p=p+(C[i]*L[i])  
  }  
  return(p)  
}
```

Figura 3

En un inicio, inicializamos p (que es nuestro polinomio interpolador) a 0 (estamos haciendo un sumatorio). A continuación, abrimos un bucle donde i se desplaza desde 1 hasta la longitud del vector C (como previamente habíamos hecho con la función del polinomio de base). Desarrollamos las operaciones necesarias del sumatorio y con *return* obtenemos el resultado.

```
s[1]  
n=length(C)  
s[n]  
x=seq(s[1],s[n],length=1001)  
F=0  
for (k in 1:1001){  
  t=x[k]  
  LH= Polbase(t,s)  
  PA= Polinte(C,LH)  
  F[k]=PA  
}
```

Figura 4

Luego se pide obtener 1001 puntos de abscisas, equidistantes, y que van desde el primer punto del vector s hasta el último: es decir, desde $s[1]$ hasta $s[n]$. Es importante definir previamente n como la longitud del vector C . Para generar los puntos deberemos utilizar el comando *seq*. En nuestro caso, definimos *seq* con nuestro parámetro inicial, nuestro parámetro final y la longitud, que en este caso es 1001. Los puntos de abscisas los almacenamos en un vector llamado x . El objetivo es, una vez hemos generado esos puntos de abscisas, calcular el valor de la función mediante el polinomio interpolador.

Para ello, abrimos un bucle donde k se mueve desde 1 hasta 1001 (abarcando así todos los puntos de abscisas generados). El argumento t (tiempo en horas, del que depende la función *Polbase*) va a adoptar el valor de cada uno de los elementos del vector x que se había definido previamente. A medida que avanza el bucle, se calcula el polinomio de base y con ello el polinomio interpolador, cuya salida se guardará en un vector $F[k]$, como puede observarse en la *figura 4* de arriba. Es el valor de la función en cada uno de los puntos de abscisa, calculados a partir del polinomio interpolador.



e.Rrados

RESOLUCIÓN Y EXPLICACIONES DEL EJERCICIO PROPUESTO EN

R – INTERPOLACIÓN DE LAGRANGE

Por fin, llega la hora de graficar. Para ello, utilizaremos el ya conocido comando *plot*. El código es el siguiente, del que se dará una explicación más detallada a continuación.

```
plot(s, C, xlim=c(s[1], s[n]), ylim=c(0,60))
par(new='TRUE')
plot(x, F, type='l', col='blue', xlim=c(s[1], s[n]), ylim=c(0,60),xlab="", ylab="")
.
```

Figura 5

La función *plot()* es una de las más importantes y usadas, amén de ser muy variado su manejo. Existen multitud de argumentos opcionales que nos dará un tipo de gráfica u otra, pero siempre será necesario poner un argumento *x*, que corresponde al eje X de la gráfica y será un vector que debe estar especificado.

En el primer *plot*, el argumento *x* lo da el vector *s* con los instantes de tiempo. El argumento *y* está representado por el vector *C* que representa las concentraciones de bioetanol. Se procede también a modificar los límites de los ejes, con *xlim* y *ylim*. Con *xlim* limitamos el eje X desde el primer valor del vector *s* hasta el último (*n* lo habíamos definido ya antes como la longitud del vector *C*).

La función *par()* sirve para poner múltiples gráficos en único *plot*. Este es nuestro caso. La función *par()* tiene muchos argumentos. Nosotros solo vamos a hacer uso de *new*.

En el siguiente *plot* que definimos, *x* son nuestros 1001 valores de abscisas equidistantes en el intervalo que va de *s[1]* a *s[n]* y que habíamos obtenido con la función *seq*. El argumento *y* es el valor de la función en cada uno de esos puntos de abscisas, haciendo uso del polinomio interpolador y que habíamos calculado ya previamente. El tipo de *plot* 'l' que escogemos nos da un gráfico de líneas. Como anteriormente, delimitamos los valores del eje X y del eje Y con los argumentos *xlim* e *ylim*. *xlab* e *ylab* son las etiquetas de los ejes X e Y. En nuestro caso no pondremos ninguna, por eso está vacío.

A la hora de ejecutar el programa y graficar obtenemos lo siguiente:



e.Rrados

RESOLUCIÓN Y EXPLICACIONES DEL EJERCICIO PROPUESTO EN
R – INTERPOLACIÓN DE LAGRANGE

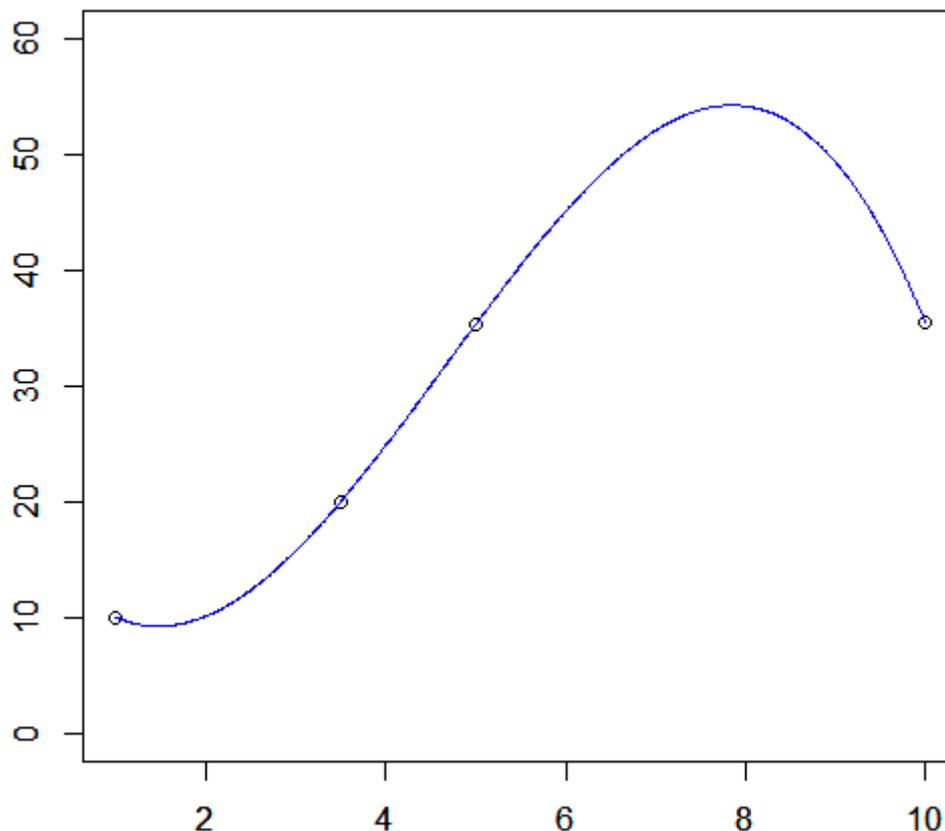


Figura 6

Una vez llegados a este paso, queda por responder a la cuestión que se nos planteaba en el ejercicio, que es la de calcular la concentración de bioetanol en los instantes $t=2$ y $t=6.5$.

Esta parte se ha hecho con Rstudio. Utilizamos la función *locator* (). En la consola, tenemos que escribir dicha función, y entonces pinchamos en el gráfico los valores de los que queremos obtener sus coordenadas. Una vez hayamos terminado de seleccionar todos los puntos, presionamos la tecla *esc*. Puesto que en la gráfica obtenida no hay mucha precisión, y la elección de los puntos los hacemos *a ojo*, obtenemos valores aproximados.



e.Rrados

RESOLUCIÓN Y EXPLICACIONES DEL EJERCICIO PROPUESTO EN R – INTERPOLACIÓN DE LAGRANGE

```
> locator()
$x
[1] 1.992114 1.992114 6.678576

$y
[1] 10.23917 10.23917 50.34298
```

Figura 7. Se observan que los puntos seleccionados se acercan mucho a los pedidos por el enunciado, y aunque no sean del todo preciso, sirven para dar una idea. Cuando hayamos terminado de seleccionar los puntos en la gráfica, y después de darle a esc, nos aparecen en la consola las coordenadas de cada uno de los puntos seleccionados.