

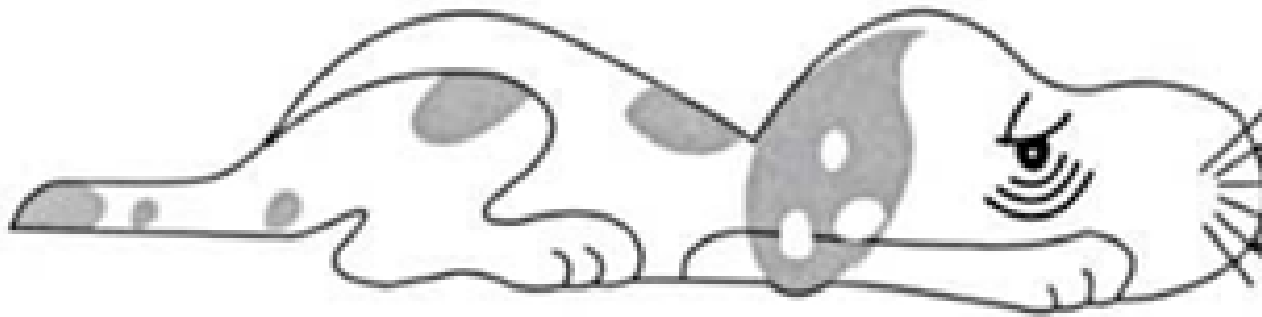
Reto Perrito durmiendo

Analisis Numerico 2019-1

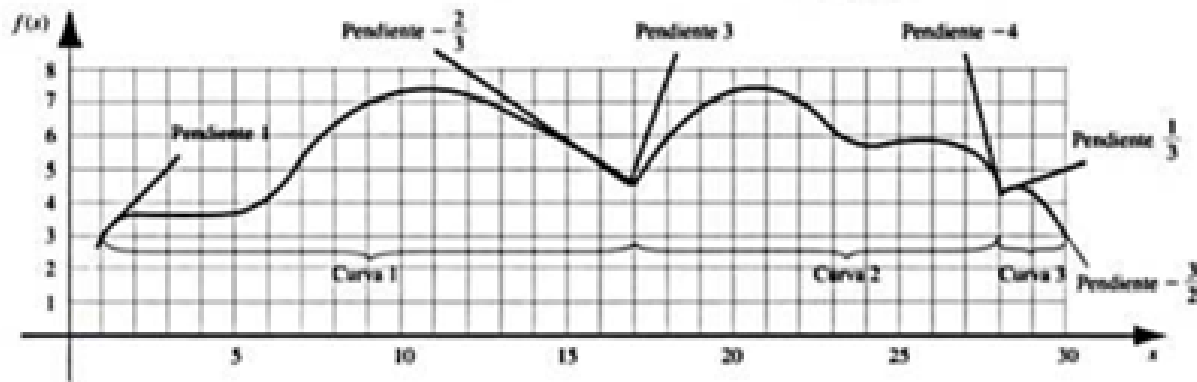
Por Carlos Restrepo

En que consiste el reto:

Construir un Interpolador (no necesariamente en forma polinomica) utilizando la menor cantidad de puntos puntos k (parte superior y/o inferior o en total) y reproducir el dibujo del contorno completo del perrito sin bigotes (mejor exactitud) con la información dada :



A caption



A caption

Criterios:

1. Metodología que explique como se seleccionaron k puntos con $k < n$ con n el total de puntos dados(Selección de más puntos o de los puntos de la parte de abajo)
2. Algoritmo que se aplico(justificación) aplico por ejemplo,interpolación polinomial y como soluciono el sistema
3. Validación del resultado

Productos

1. Algoritmo,requerimientos,codificación
2. Codificación, tabla donde esta la interpolación en los n-k puntos (no seleccionados), el polinomio o la función interpolante. En un plano los puntos originales, los utilizados ,el contorno y el interpolado(utilice el grosor minimo para la curva).
 - Calcular la cota de error de su método con los datos experimentales y comparela con la cota teorica.
3. tabla donde esten los valores interpolados(tenga en cuenta los que no utilizo), los originales y el error relativo , calcule un error relativo total como la suma de los errores relativos.
 - Cree una funcion que cuente el numero aciertos y el numero de diferencias en una cifra entre su función de interpolacion y los originales y implementelo como el índice de Jaccard.

4. Cree una función que muestre la eficiencia de su metodo

- Preguntas?

1. El origen se puede modificar?

- Se podria.

2. Si tenemos nueva información osea nodos como podemos implementar esa información en el algoritmo de interpolación?

- Modificar he ingrisar

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

- No lo es

4. Suponga que tiene más puntos con más cifras significativas como se comporta su algoritmo ? la exactitud decae?

- Alteraciones en la grafica

```
require(pracma)
```

```
## Loading required package: pracma
```

```

xi = c(1,2,5,6,7.5,8.1,10,13,17.6,20,23.5,24.5,25,26.5,27.5,28,29,30)
yi = c(3,3.7,3.9,4.5,5.7,6.69,7.12,6.7,4.45,7,6.1,5.6,5.87,5.15,4.1,4.3,4.1,3)

xd = c(1,2,5,6,7,8.1,10,13,17.6,20,23,24,25,27,27.7,28,29,30,30,1,1)
yd = c(3.0,3.7,3.9,4.2,5.7,6.6,7.1,6.7,4.5,7.0,6.1,5.6,5.8,5.2,4.1,4.3,4.1,3.0,2,2,3)

```

Selección de Puntos para Interpolación

Gráfica de los puntos sin Interpolación

X Datos Prof	Y Datos Prof	X Datos Libro	Y Datos Libro
1	3	1	3
2	3.7	2	3.7
5	3.9	5	3.9
6	4.5	6	4.2
7.5	5.7	7	5.7
8.1	6.69	8.1	6.6
10	7.12	10	7.1
13	6.7	13	6.7
17.6	4.45	17.6	4.5
20	7	20	7
23.5	6.1	23	6.1
24.5	5.6	24	5.6
25	5.87	25	5.8
26.5	5.15	27	5.2
27.5	4.1	27.7	4.1

XDatos Prof	Y Datos Prof	XDatos Libro	Y Datos Libro
28	4.3	28	4.3
29	4.1	29	4.1
30	3	30	3

Medias

Media #datos	Media X Prof	Media Y Prof	Media X Libro	Media Y Libro
9.5	16.9	5.05	16.8	5.03

Regiones a trabajar

Regiones a trabajar	Puntos Cola	Cabeza	Nariz-Pata
3	1-8	8-13	13-18

Algoritmo Usado

- Metodo de la forma baricéntrica de Lagrange:
- Recibe el intervalo de nodos para realizar la interpolación
- La forma de Lagrange del polinomio interpolante es atractiva para propósitos teóricos. Sin embargo se puede re-escribir en una forma que se vuelva eficiente para el cálculo computacional además de ser numéricamente mucho más estable. La forma modificada y la forma baricéntrica de Lagrange son útiles cuando queremos interpolar una función en todo un intervalo con un con un polinomio interpolante.
- **1. Paquete Pracma**
 - Este paquete proporciona implementaciones en R de funciones más avanzadas en análisis numérico, con una vista especial sobre rutinas de optimización y series de tiempo.
- **2. Seq**
 - La función seq () en R genera una secuencia de números.
 - seq(from, to, by, length.out)
 - desde,hasta, rango, tamaño de secuencia

• 3. Barylag

- - Barylag interpola los datos dados utilizando la fórmula de interpolación de Lagrange baricéntrica (vectorizada para eliminar todos los bucles).
- - `barylag(xi, yi, x)`
- - `xi` y `yi` Cordenadas para los nodos
- - `x`. Puntos de Interpolacion

Cotas:

Cota teoricas:

X	YProf	YLibro
5	3.9	3.9
10	7.12	7.1
20	7	7
26	5.15	5.2
28	4.3	4.3

Resultados graficos para datos profesora

You can also embed plots, for example:

```
#Gráfica de los puntos
plot(xi,yi, pch=21, cex=1, col = "red", asp=1,xlab="X", ylab="Y", main="Perro Durmiendo Datos profesora")

LineaInterpolacion<-function(inicio, final){
  xi0 = xi[inicio:final]
  yi0 = yi[inicio:final]
  x0 <- seq(xi[inicio], xi[final], len=20)
  y0 <- barylag(xi0, yi0, x0)
  print(x0)
  lines(x0, y0, col="blue")
}

LineaInterpolacion (1,4)
```

```
## [1] 1.000000 1.263158 1.526316 1.789474 2.052632 2.315789 2.578947
## [8] 2.842105 3.105263 3.368421 3.631579 3.894737 4.157895 4.421053
## [15] 4.684211 4.947368 5.210526 5.473684 5.736842 6.000000
```

```
LineaInterpolacion (4,6)
```

```
## [1] 6.000000 6.110526 6.221053 6.331579 6.442105 6.552632 6.663158
## [8] 6.773684 6.884211 6.994737 7.105263 7.215789 7.326316 7.436842
## [15] 7.547368 7.657895 7.768421 7.878947 7.989474 8.100000
```

```
LineaInterpolacion (6,8)
```

```
## [1] 8.100000 8.357895 8.615789 8.873684 9.131579 9.389474 9.647368
## [8] 9.905263 10.163158 10.421053 10.678947 10.936842 11.194737 11.452632
## [15] 11.710526 11.968421 12.226316 12.484211 12.742105 13.000000
```

```
LineaInterpolacion (8,9)
```

```
## [1] 13.00000 13.24211 13.48421 13.72632 13.96842 14.21053 14.45263
## [8] 14.69474 14.93684 15.17895 15.42105 15.66316 15.90526 16.14737
## [15] 16.38947 16.63158 16.87368 17.11579 17.35789 17.60000
```

LineaInterpolacion (9,11)

```
## [1] 17.60000 17.91053 18.22105 18.53158 18.84211 19.15263 19.46316
## [8] 19.77368 20.08421 20.39474 20.70526 21.01579 21.32632 21.63684
## [15] 21.94737 22.25789 22.56842 22.87895 23.18947 23.50000
```

LineaInterpolacion (11,13)

```
## [1] 23.50000 23.57895 23.65789 23.73684 23.81579 23.89474 23.97368
## [8] 24.05263 24.13158 24.21053 24.28947 24.36842 24.44737 24.52632
## [15] 24.60526 24.68421 24.76316 24.84211 24.92105 25.00000
```

LineaInterpolacion (13,15)

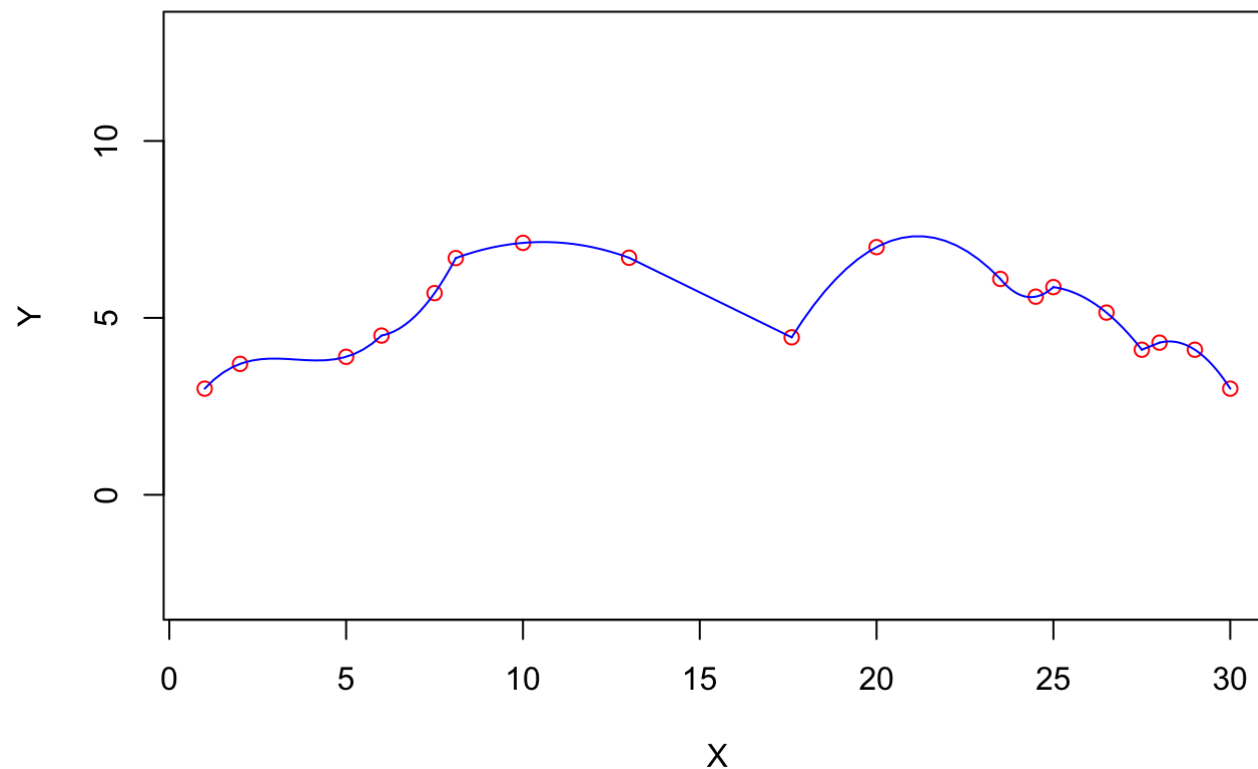
```
## [1] 25.00000 25.13158 25.26316 25.39474 25.52632 25.65789 25.78947
## [8] 25.92105 26.05263 26.18421 26.31579 26.44737 26.57895 26.71053
## [15] 26.84211 26.97368 27.10526 27.23684 27.36842 27.50000
```

LineaInterpolacion (15,16)

```
## [1] 27.50000 27.52632 27.55263 27.57895 27.60526 27.63158 27.65789
## [8] 27.68421 27.71053 27.73684 27.76316 27.78947 27.81579 27.84211
## [15] 27.86842 27.89474 27.92105 27.94737 27.97368 28.00000
```

LineaInterpolacion (16,18)

Perro Durmiendo Datos profesora



```
## [1] 28.00000 28.10526 28.21053 28.31579 28.42105 28.52632 28.63158  
## [8] 28.73684 28.84211 28.94737 29.05263 29.15789 29.26316 29.36842  
## [15] 29.47368 29.57895 29.68421 29.78947 29.89474 30.00000
```

Resultados graficos para datos del libro


```
#Gráfica de los puntos
plot(xd,yd, pch=21, cex=1, col = "red", asp=1,xlab="X", ylab="Y", main="Perro Durmiendo Datos Libro")

LineaInterpolacion2<-function(inicio, final){
  xi0 = xd[inicio:final]
  yi0 = yd[inicio:final]
  x0 <- seq(xd[inicio], xd[final], len=20)
  print(x0)
  y0 <- barylag(xi0, yi0, x0)
  lines(x0, y0, col="blue")
}

LineaInterpolacion2 (1,4)
```

```
## [1] 1.000000 1.263158 1.526316 1.789474 2.052632 2.315789 2.578947
## [8] 2.842105 3.105263 3.368421 3.631579 3.894737 4.157895 4.421053
## [15] 4.684211 4.947368 5.210526 5.473684 5.736842 6.000000
```

```
LineaInterpolacion2 (4,6)
```

```
## [1] 6.000000 6.110526 6.221053 6.331579 6.442105 6.552632 6.663158
## [8] 6.773684 6.884211 6.994737 7.105263 7.215789 7.326316 7.436842
## [15] 7.547368 7.657895 7.768421 7.878947 7.989474 8.100000
```

```
LineaInterpolacion2 (6,8)
```

```
## [1] 8.100000 8.357895 8.615789 8.873684 9.131579 9.389474 9.647368
## [8] 9.905263 10.163158 10.421053 10.678947 10.936842 11.194737 11.452632
## [15] 11.710526 11.968421 12.226316 12.484211 12.742105 13.000000
```

```
LineaInterpolacion2 (8,9)
```

```
## [1] 13.00000 13.24211 13.48421 13.72632 13.96842 14.21053 14.45263
## [8] 14.69474 14.93684 15.17895 15.42105 15.66316 15.90526 16.14737
## [15] 16.38947 16.63158 16.87368 17.11579 17.35789 17.60000
```

LineaInterpolacion2 (9,11)

```
## [1] 17.60000 17.88421 18.16842 18.45263 18.73684 19.02105 19.30526
## [8] 19.58947 19.87368 20.15789 20.44211 20.72632 21.01053 21.29474
## [15] 21.57895 21.86316 22.14737 22.43158 22.71579 23.00000
```

LineaInterpolacion2 (11,13)

```
## [1] 23.00000 23.10526 23.21053 23.31579 23.42105 23.52632 23.63158
## [8] 23.73684 23.84211 23.94737 24.05263 24.15789 24.26316 24.36842
## [15] 24.47368 24.57895 24.68421 24.78947 24.89474 25.00000
```

LineaInterpolacion2 (13,15)

```
## [1] 25.00000 25.14211 25.28421 25.42632 25.56842 25.71053 25.85263
## [8] 25.99474 26.13684 26.27895 26.42105 26.56316 26.70526 26.84737
## [15] 26.98947 27.13158 27.27368 27.41579 27.55789 27.70000
```

LineaInterpolacion2 (15,16)

```
## [1] 27.70000 27.71579 27.73158 27.74737 27.76316 27.77895 27.79474
## [8] 27.81053 27.82632 27.84211 27.85789 27.87368 27.88947 27.90526
## [15] 27.92105 27.93684 27.95263 27.96842 27.98421 28.00000
```

LineaInterpolacion2 (16,18)

```
## [1] 28.00000 28.10526 28.21053 28.31579 28.42105 28.52632 28.63158
## [8] 28.73684 28.84211 28.94737 29.05263 29.15789 29.26316 29.36842
## [15] 29.47368 29.57895 29.68421 29.78947 29.89474 30.00000
```

LineaInterpolacion2 (18,19)

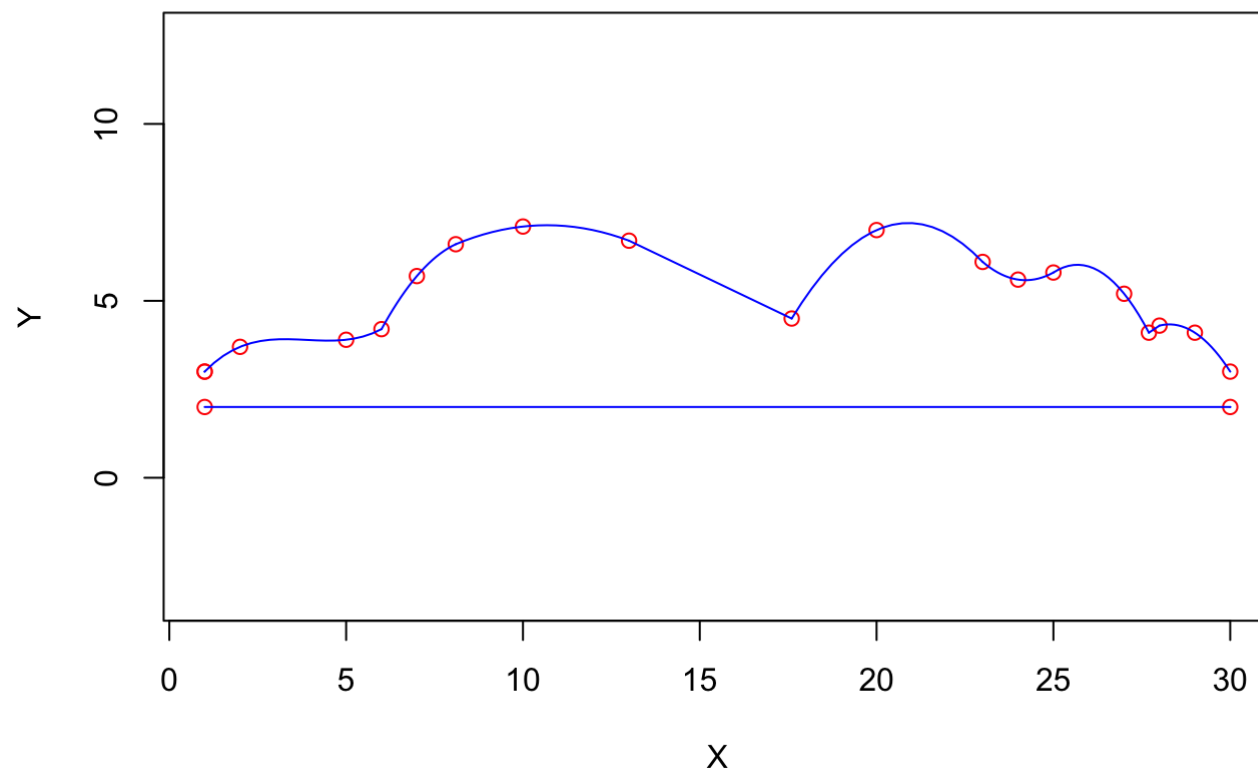
```
## [1] 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30
```

```
LineaInterpolacion2 (19,20)
```

```
## [1] 30.000000 28.473684 26.947368 25.421053 23.894737 22.368421 20.842105  
## [8] 19.315789 17.789474 16.263158 14.736842 13.210526 11.684211 10.157895  
## [15] 8.631579 7.105263 5.578947 4.052632 2.526316 1.000000
```

```
LineaInterpolacion2 (20,21)
```

Perro Durmiendo Datos Libro



```
## [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```

<http://disi.unal.edu.co/~lctorress/MetNum/MeNuCI03.pdf> (<http://disi.unal.edu.co/~lctorress/MetNum/MeNuCI03.pdf>)

https://fac.ksu.edu.sa/sites/default/files/numerical_analysis_9th.pdf
(https://fac.ksu.edu.sa/sites/default/files/numerical_analysis_9th.pdf)

http://numat.net/tutor/error_interp.pdf (http://numat.net/tutor/error_interp.pdf)

<https://w3.ual.es/~aposadas/TeoriaErrores.pdf> (<https://w3.ual.es/~aposadas/TeoriaErrores.pdf>)

<http://www.sc.ehu.es/nmwmigaj/instrum.htm> (<http://www.sc.ehu.es/nmwmigaj/instrum.htm>)

https://tarwi.lamolina.edu.pe/~fmendiburu/index-filer/academic/script_numerico.htm
(https://tarwi.lamolina.edu.pe/~fmendiburu/index-filer/academic/script_numerico.htm)

Carlos Manuel Restrepo Riveros

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.