

# Métodos Numéricos

## Interpolación y Ajuste de Funciones

Diego Passarella

Universidad Nacional de Quilmes

# Planteo del Problema

Muchas veces es necesario evaluar una función  $f(x)$  en uno o varios puntos de su dominio. Esto puede ser un problema debido a:

- La función es muy costosa de evaluar
- No se conoce la expresión de la función, solamente su valor en algunos puntos

En estos casos es más conveniente aproximar a  $f(x)$  por otra más fácil de evaluar  $f(x) \approx \hat{f}(x)$ .

Si tenemos  $n + 1$  pares de puntos  $\{(x_i, y_i)\}$  con  $i = 0, 1, \dots, n$ , se requiere que la función interpolante cumpla:

$$\hat{f}(x_i) = f(x_i) = y_i$$

# Polinomios de Interpolación

La forma más común que toma  $\hat{f}(x)$  es la de un polinomio ( $\hat{f}(x) = P_m(x)$ ). Notar que el máximo orden de  $P_m(x)$  puede ser  $n$  para que el problema esté bien definido. Cuando  $m = n$  el problema posee una única solución.

## Tipos de Polinomios:

Se pueden utilizar distintos tipos de polinomios para realizar ajuste y/o interpolación de series de datos:

- Polinomio de Taylor
- Polinomio de Lagrange
- Polinomio de Hermite
- Splines
- Mínimos Cuadrados

# Polinomio de Taylor

Si se tiene un punto  $x_0$  y  $n + 1$  valores  $w_0, w_1, \dots, w_n$  que representan:

$$f(x_0) = w_0, \quad f^{(i)}(x_0) = w_i, \quad i = 1, \dots, n$$

Recordando que el desarrollo en serie de Taylor de  $f(x)$  hasta orden  $n$  es:

$$P_n(x) = f(x_0) + \sum_{r=1}^n f^{(r)}(x_0) \frac{(x - x_0)^r}{r!}$$

El polinomio interpolante de Taylor de  $f(x)$  en un entorno centrado en  $x_0$  queda:

$$P_n(x) = w_0 + \sum_{r=1}^n w_r \frac{(x - x_0)^r}{r!}$$

# Polinomio de Taylor

De forma general, el polinomio de Taylor se puede expresar como:

$$P_n(x) = w_0 + \sum_{r=1}^n w_r t_r(x)$$

donde los polinomios característicos (o funciones de base) de Taylor son:

$$t_r(x) = \frac{(x - x_0)^r}{r!}$$

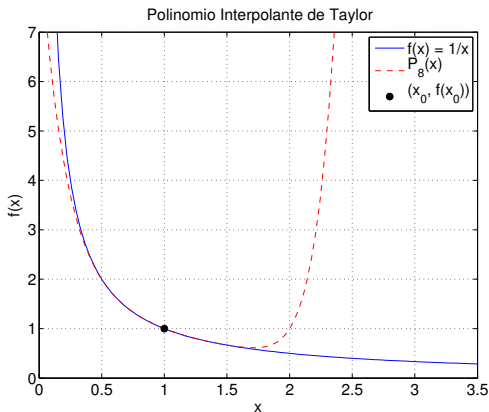
Este polinomio posee un error de la forma:

$$R_n(f) = f^{(n+1)}(\xi) \frac{(x - x_0)^{n+1}}{(n+1)!}, \quad \xi \in (x, x_0)$$

# Polinomio de Taylor

Ventajas: Simple

Desventajas: Es necesario conocer las derivadas hasta orden  $n$ . Generalmente sólo es efectivo para un entorno pequeño de  $x_0$ .



# Problema de Interpolación

Con  $n + 1$  pares de valores  $\{(x_i, y_i)\}$  con  $i = 0, 1, \dots, n$  se puede generar un polinomio de grado  $n$  de forma unívoca que cumpla  $P_n(x_i) = y_i$ . Este polinomio es de la forma:

$$P_n(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$$

Se tienen  $n + 1$  incógnitas y  $n + 1$  pares de datos, por lo que se llega al siguiente sistema lineal:

$$\begin{pmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ 1 & x_2 & x_2^2 & \cdots & x_2^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^n \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$$

# Polinomio de Lagrange - Funciones de Base

El sistema anterior está muy mal condicionado, por lo que la obtención del polinomio de Lagrange se realiza por otros métodos.

Método de las funciones de base:

Se propone que el polinomio interpolante de Lagrange sea de la forma:

$$P_n(x) = \sum_{i=0}^n y_i L_i(x)$$

Con las funciones de base cumpliendo los siguientes requisitos:

$$L_i(x) \in \mathbb{P}_n \quad y \quad L_i(x_j) = \begin{cases} 1 & \text{si } j = i \\ 0 & \text{si } j \neq i \end{cases}$$



# Polinomio de Lagrange - Funciones de Base

Construcción de las funciones de base:

$$L_i(x) = \prod_{j=0, j \neq i}^n \frac{x - x_j}{x_i - x_j}$$

Ejemplo para  $n = 3$

$$L_0(x) = \frac{x - x_1}{x_0 - x_1} \cdot \frac{x - x_2}{x_0 - x_2} \cdot \frac{x - x_3}{x_0 - x_3}$$

$$L_1(x) = \frac{x - x_0}{x_1 - x_0} \cdot \frac{x - x_2}{x_1 - x_2} \cdot \frac{x - x_3}{x_1 - x_3}$$

$$L_2(x) = \frac{x - x_0}{x_2 - x_0} \cdot \frac{x - x_1}{x_2 - x_1} \cdot \frac{x - x_3}{x_2 - x_3}$$

$$L_3(x) = \frac{x - x_0}{x_3 - x_0} \cdot \frac{x - x_1}{x_3 - x_1} \cdot \frac{x - x_2}{x_3 - x_2}$$

# Polinomio de Lagrange - Fenómeno de Runge

## Problema:

Al aumentar el grado del polinomio interpolante se pueden producir oscilaciones en los extremos de la función. En general no es conveniente utilizar polinomios de alto grado.

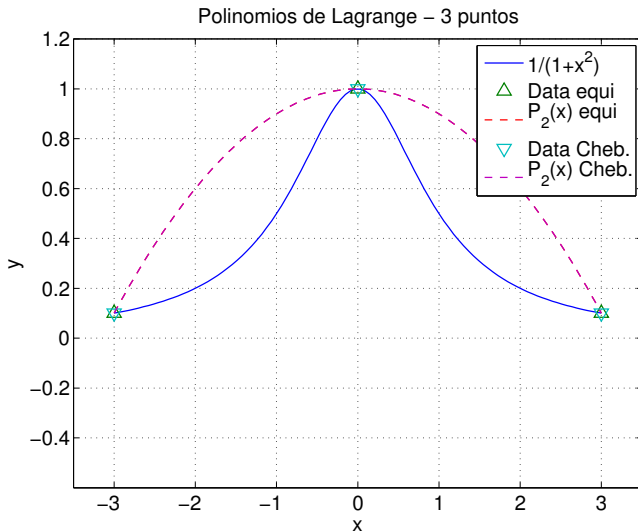
## Solución:

Utilizar nodos de interpolación de Chebyshev. Para  $x \in [a, b]$  se toma:

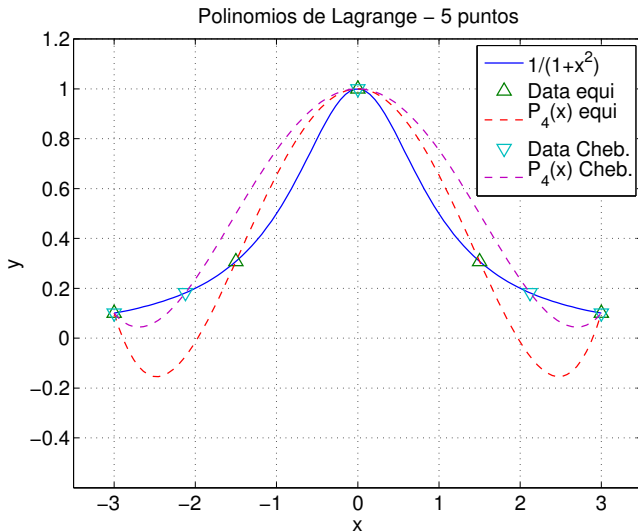
$$x_i = \frac{a+b}{2} + \frac{a-b}{2} \cdot \cos\left(\frac{\pi}{n}i\right) \quad i = 0, 1, \dots, n$$

y en cada  $x_i$  utilizar las funciones de base de Lagrange.

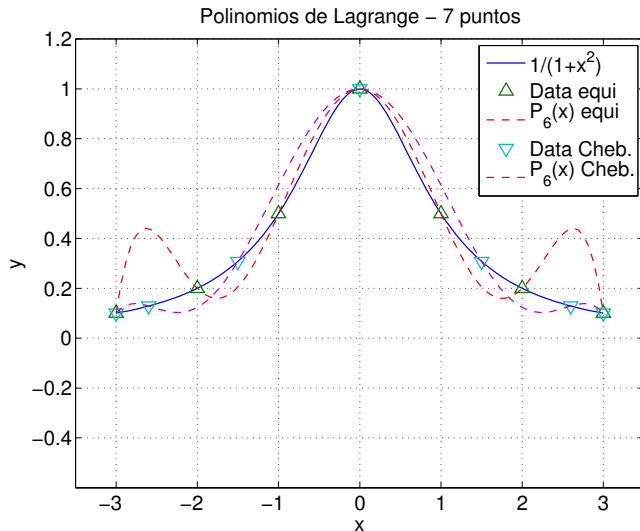
# Polinomio de Lagrange - Ejemplo



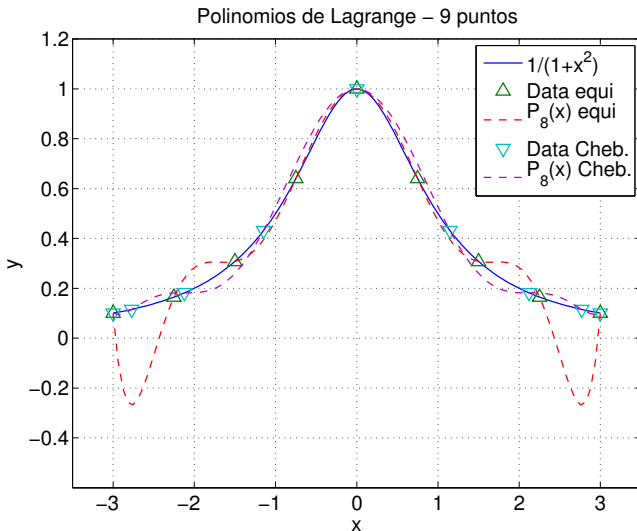
# Polinomio de Lagrange - Ejemplo



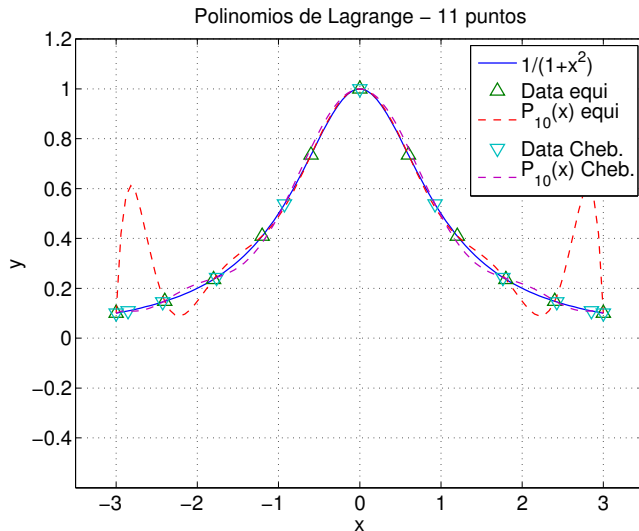
# Polinomio de Lagrange - Ejemplo



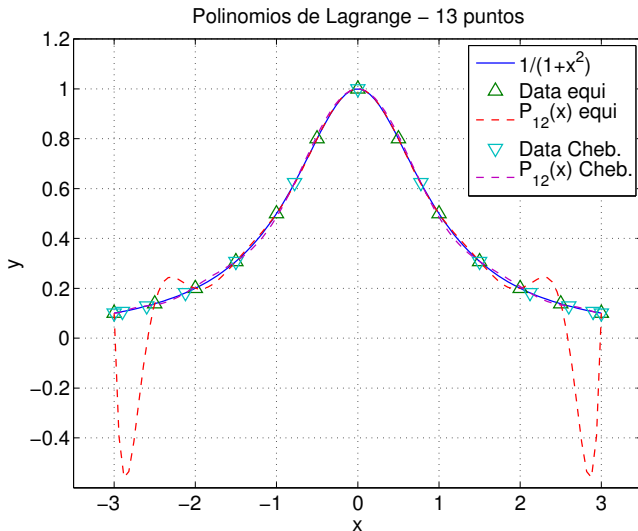
# Polinomio de Lagrange - Ejemplo



# Polinomio de Lagrange - Ejemplo



# Polinomio de Lagrange - Ejemplo





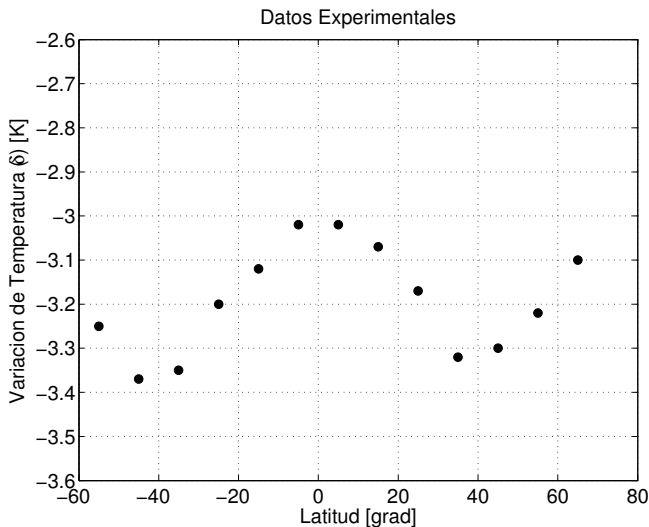
# Polinomio de Lagrange - Lineal a trozos

Cuando no se conoce nada acerca de la regularidad de la función a aproximar, una de las interpolaciones más conviene es la lineal a trozos.

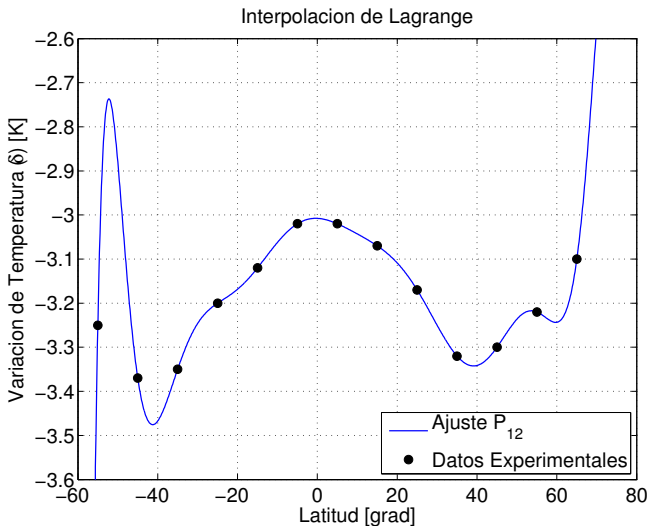
Para un intervalo compuesto por  $\{(x_0, y_0), (x_1, y_1)\}$  se tiene:

$$p_1(x) = y_0 + \frac{y_1 - y_0}{x_1 - x_0} \cdot (x - x_0)$$

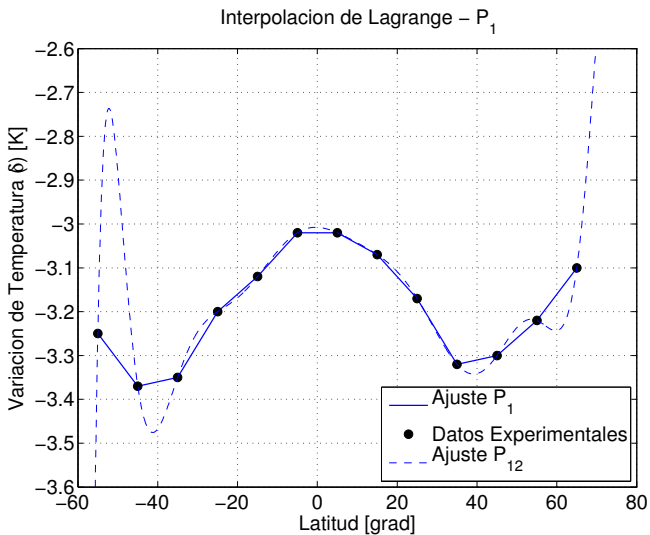
# Polinomio de Lagrange - Fenómeno de Runge



# Polinomio de Lagrange - Fenómeno de Runge



# Polinomio de Lagrange - Lineal a trozos



# Definición

Si se dispone de los valores de la función y de su derivada en los puntos de interpolación:

$x_i$	$x_0$	$x_1$	$\cdots$	$x_n$
$y_i$	$y_0$	$y_1$	$\cdots$	$y_n$
$y'_i$	$y'_0$	$y'_1$	$\cdots$	$y'_n$

Se dispone de  $2(n+1)$  pares de datos para realizar un ajuste polinómico de orden  $2n+1$  unívoco

El polinomio de Hermite se define como:

$$P_{2n+1}(x) = \sum_{j=0}^n h_j(x) y_j + \sum_{j=0}^n g_j(x) y'_j$$

# Funciones de Base

$$P_{2n+1}(x) = \sum_{j=0}^n h_j(x) y_j + \sum_{j=0}^n g_j(x) y'_j$$

con,

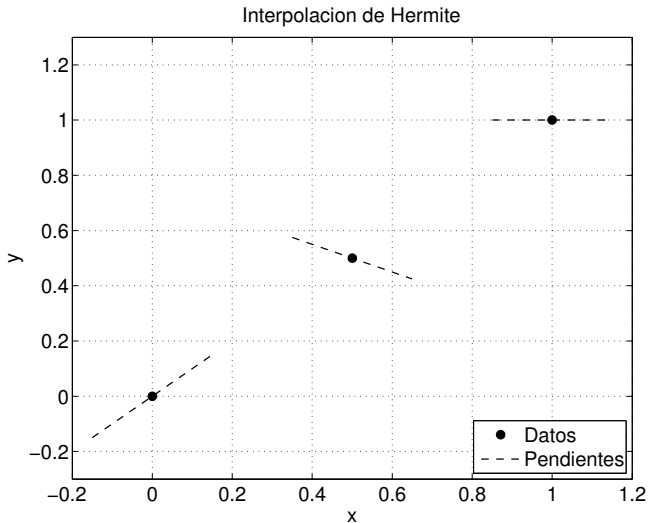
$$h_j(x) = [1 - 2(x - x_j) L'_j(x_j)] L_j^2(x), \quad g_j(x) = (x - x_j) L_j^2(x)$$

que verifican,

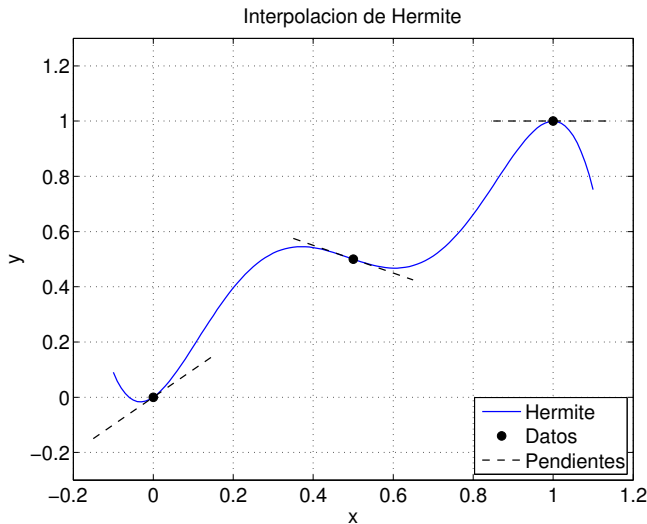
$$h_j(x_i) = \begin{cases} 1 & \text{si } j = i \\ 0 & \text{si } j \neq i \end{cases} \quad h'_j(x_i) = \begin{cases} 0 & \text{si } j = i \\ 0 & \text{si } j \neq i \end{cases}$$

$$g_j(x_i) = \begin{cases} 0 & \text{si } j = i \\ 0 & \text{si } j \neq i \end{cases} \quad g'_j(x_i) = \begin{cases} 1 & \text{si } j = i \\ 0 & \text{si } j \neq i \end{cases}$$

# Interpolación de Hermite - Datos



# Interpolación de Hermite - Ajuste





# Definición

La aproximación lineal a trozos genera una función continua pero no suave. Una opción para obtener interpoladores más suaves es la utilización de *splines*.

Condiciones y características de un spline cúbico:

- En cada sub-intervalo  $I_i = [x_i, x_{i+1}]$  para  $i = 0, 1, \dots, n-1$ ,  $s_3$  es un polinomio de grado 3 que interpola a los pares de puntos  $(x_i, y_i)$ ,  $(x_{i+1}, y_{i+1})$ .
- Cada  $s_3$  es continua y posee derivadas primera y segunda continuas en los nodos  $x_i$ ,  $i = 1, 2, \dots, n-1$ .

Un spline cúbico es interesante para describir trayectorias suaves  
¿Por qué?

# Splines - Obtención

Para definir las  $n$  funciones  $s_3$  de cada sub-intervalo, es necesario determinar  $4n$  constantes. Las condiciones para determinarlas surgen de:

- $n + 1$  de la condición de interpolación  $(x_i, y_i)$  en cada nodo.
- $3(n - 1)$  de la condición de continuidad de la función y de las derivadas primera y segunda.
- Se necesitan 2 condiciones extras.
  - Spline natural:  $s_3^{(2)}(x_0) = s_3^{(2)}(x_n) = 0$  (la opción más común).
  - Periodicidad:  $s_3^{(1)}(x_0) = s_3^{(1)}(x_n)$ ,  $s_3^{(2)}(x_0) = s_3^{(2)}(x_n)$  (para curvas periódicas).

# Splines - Obtención

Definiciones:  $f_i = s_3(x_i)$ ,  $m_i = s_3^{(1)}(x_i)$ ,  $M_i = s_3^{(2)}(x_i)$

Como  $s_3(x) \in \mathbb{P}_3 \Rightarrow s_3^{(2)}(x)$  es lineal. Entonces, en el  $I_{i-1}$  se tiene:

$$s_3^{(2)}(x) = M_{i-1} \frac{x_i - x}{x_i - x_{i-1}} + M_i \frac{x - x_{i-1}}{x_i - x_{i-1}} \quad \text{y } h_i = x_i - x_{i-1}$$

Integrando dos veces esta expresión se llega a:

$$s_3(x) = M_{i-1} \frac{(x_i - x)^3}{6h_i} + M_i \frac{(x - x_{i-1})^3}{6h_i} + A_{i-1}(x - x_{i-1}) + B_{i-1}$$

En los extremos de  $I_{i-1}$ :  $s_3(x_{i-1}) = f_{i-1}$  y  $s_3(x_i) = f_i$ , por lo tanto:

$$B_{i-1} = f_{i-1} - \frac{M_{i-1}}{6} h_i^2 \quad \text{y} \quad A_{i-1} = (M_{i-1} - M_i) \frac{h_i}{6} + \frac{f_i - f_{i-1}}{h_i}$$

# Splines - Obtención

Dado que la derivada primera en  $x_i$  debe ser continua:

$$s_3^{(1)}(x_i^-) = s_3^{(1)}(x_i^+)$$

$$M_{i-1} \frac{h_i}{6} + M_i \frac{h_i}{3} + \frac{f_i - f_{i-1}}{h_i} = -M_i \frac{h_{i+1}}{3} + M_{i+1} \frac{h_{i+1}}{6} + \frac{f_{i+1} - f_i}{h_{i+1}}$$

Reordenando queda:

$$M_{i-1} \frac{h_i}{6} + M_i \left( \frac{h_i}{3} + \frac{h_{i+1}}{3} \right) + M_{i+1} \frac{h_{i+1}}{6} = \frac{f_{i+1} - f_i}{h_{i+1}} - \frac{f_i - f_{i-1}}{h_i}$$

# Splines - Obtención

Multiplicando toda la ecuación anterior por  $6/(h_i + h_{i+1})$  y redefiniendo:

$$\mu_i = \frac{h_i}{h_i + h_{i+1}} \quad \lambda_i = \frac{h_{i+1}}{h_i + h_{i+1}}$$

y al miembro de la derecha que depende de  $f_{i-1}$ ,  $f_i$  y  $f_{i+1}$  como  $d_i$ , se llega a:

$$\mu_i M_{i-1} + 2M_i + \lambda_i M_{i+1} = d_i, \quad i = 1, 2, \dots, n-1$$

por lo que se puede armar un sistema tridiagonal de ecuaciones lineales para determinar los valores de las derivadas segundas  $M_k$  en los nodos interiores.

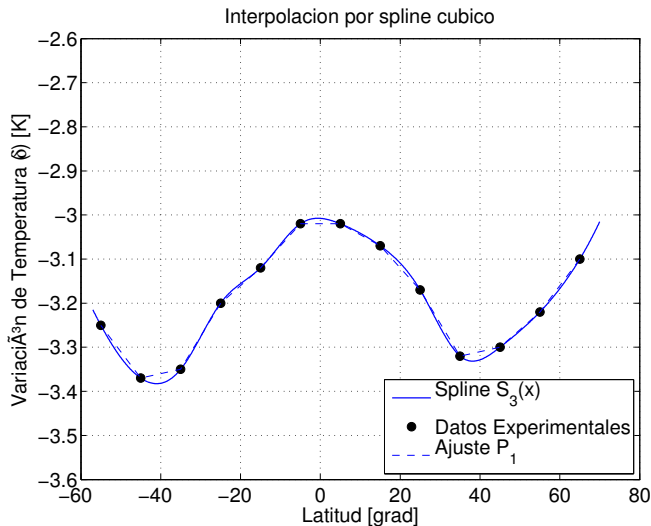
# Splines - Obtención

El sistema queda:

$$\begin{pmatrix} 2 & \lambda_1 & 0 & 0 & \cdots & 0 \\ \mu_2 & 2 & \lambda_2 & 0 & \cdots & 0 \\ 0 & \mu_3 & 2 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mu_{n-2} & 2 & \lambda_{n-2} \\ 0 & 0 & 0 & \cdots & \mu_{n-1} & 2 \end{pmatrix} \begin{pmatrix} M_1 \\ M_2 \\ M_3 \\ \vdots \\ M_{n-2} \\ M_{n-1} \end{pmatrix} = \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ \vdots \\ d_{n-2} \\ d_{n-1} \end{pmatrix}$$

Una vez obtenidos los valores de las derivadas segundas en cada uno de los nodos interiores, se puede reconstruir el  $s_3(x)$  en cada sub-intervalo.

# Splines - Ejemplo



# Ajuste por Mínimos Cuadrados

Se tienen  $n + 1$  pares de puntos  $(x_i, y_i)$  obtenidos del muestreo de una señal o de la realización de un experimento. Se busca encontrar un  $P_m(x)$  con  $m \ll n$  que cumpla:

$$\sum_{i=0}^n [y_i - P_m(x_i)]^2 \quad \text{sea mínimo}$$

Siendo el  $P_m(x)$  óptimo  $a_0 + a_1x + a_2x^2 + \cdots + a_mx^m$  y un  $P_m(x)$  cualquiera expresado como  $b_0 + b_1x + b_2x^2 + \cdots + b_mx^m$ , el problema de ajuste por mínimos cuadrados se puede reescribir como:

$$\Phi(a_0, a_1, \dots, a_m) = \min_{(b_i)_{i=0, \dots, m}} \Phi(b_0, b_1, \dots, b_m)$$

con  $\Phi = \sum_{i=0}^n [y_i - (b_0 + b_1x_i + \cdots + b_mx_i^m)]^2$  y  $\Phi : \mathbb{R}^{m+1} \rightarrow \mathbb{R}$ .



# Ajuste por Mínimos Cuadrados

El funcional  $\Phi$  de forma desarrollada queda:

$$\Phi(\vec{b}) = \sum_{i=0}^n y_i^2 - 2y_i (b_0 + b_1x_i + \cdots + b_mx_i^m) + (b_0 + b_1x_i + \cdots + b_mx_i^m)^2$$

la solución  $(a_0, a_1, \cdots, a_m)$  que hace mínimo al problema se obtiene de realizar  $\nabla_b \Phi = \vec{0}$ , es decir:

$$\frac{\partial \Phi}{\partial b_j} = - \sum_{i=0}^n y_i x_i^j + b_0 \sum_{i=0}^n x_i^j + b_1 \sum_{i=0}^n x_i^{1+j} + \cdots + b_m \sum_{i=0}^n x_i^{m+j} = 0$$

con  $j = 0, 1, \cdots, m$ . De esta forma, obtengo un sistema lineal de  $m + 1$  ecuaciones para determinar los  $m + 1$  coeficientes óptimos.

# Ajuste por Mínimos Cuadrados

El problema de mínimos cuadrados termina siendo:

$$\begin{pmatrix} n+1 & \sum_{i=0}^n x_i & \sum_{i=0}^n x_i^2 & \cdots & \sum_{i=0}^n x_i^m \\ \sum_{i=0}^n x_i & \sum_{i=0}^n x_i^2 & \sum_{i=0}^n x_i^3 & \cdots & \sum_{i=0}^n x_i^{1+m} \\ \sum_{i=0}^n x_i^2 & \sum_{i=0}^n x_i^3 & \sum_{i=0}^n x_i^4 & \cdots & \sum_{i=0}^n x_i^{2+m} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \sum_{i=0}^n x_i^m & \sum_{i=0}^n x_i^{1+m} & \sum_{i=0}^n x_i^{2+m} & \cdots & \sum_{i=0}^n x_i^{2+m} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_m \end{pmatrix} = \begin{pmatrix} \sum_{i=0}^n y_i \\ \sum_{i=0}^n y_i x_i \\ \sum_{i=0}^n y_i x_i^2 \\ \vdots \\ \sum_{i=0}^n y_i x_i^m \end{pmatrix}$$

# Ejercicio

Se tienen los siguientes valores de la densidad del agua ( $\rho [kg/m^3]$ ) en función de la temperatura ( $T [C]$ ):

T	4	8	12	16	20
$\rho$	1000.7794	1000.6427	1000.2805	999.7165	998.9700

Realice distintos tipos de ajuste (Lagrange  $P_4$ , Lineal a trozos, spline, etc...). Compare los resultados obtenidos con cada ajuste y el error cometido siendo que otros valores de la densidad en función de la temperatura son:

T	6	10	14	18
$\rho$	1000.74088	1000.4882	1000.0224	999.3650

# Comandos de Matlab/Octave

Para realizar estos ejercicios y/o comparar el funcionamiento de scripts propios, puede resultar práctico utilizar los siguientes comandos de Matlab/Octave:

```
p = polyfit(x,y,n)
y = polyval(p,x)
y2 = spline(x1,y1,x2)
```