Universidad Nacional de San Juan

Facultad de Ingeniería

MÉTODOS NUMÉRICOS

Ingeniería CIVIL

Equipo de Cátedra

Profesor Titular	Prof. Beatriz Morales
Profesor Asociado	Ing. Agustina Garcés
Profesor Adjunto	Ing. Marión Castro.
Jefe de Trabajos Prácticos	Ing. Pablo Marcuzzi
Jefe de Trabajos Prácticos	Ing. Germán Rodriguez

Año 2021

AJUSTE

Dado un conjunto de datos experimentales como resultado de observaciones, experimentos, etc. se necesita obtener una ley matemática que modele el fenómeno físico del cual se conocen esas mediciones. Los datos experimentales están sujetos a errores experimentales.

A partir de datos experimentales, se tratará de encontrar la ecuación matemática que permita *predecir resultados sin necesidad de repetir el experimento*, siempre dentro del rango de los valores experimentales.

La estrategia es ajustar una curva directamente mediante los puntos y usar esta curva para predecir valores intermedios.

En Estadística el ajuste lineal se realiza por medio de la Regresión Lineal simple. Para lo cual se hacen supuestos estadísticos que llevan a hacer inferencias o predicciones.

Vamos a suponer que los datos experimentales tienen solamente errores verticales.

La herramienta matemática que resuelve este tipo de problemas es Ajuste o Aproximación

Ejemplo 1

Se han hecho mediciones sobre distintas tensiones aplicadas a una probeta de acero obteniendo distintas elongaciones.

Objetivo calcular la expresión matemática que modela el fenómeno, para predecir la elongación de la probeta si se aplica una tensión de 3.5.

Tensión	Elongación
1	1
2	2.35
3	2.85
4	4.78
5	5.42
6	6.07

De la observación de la Figura 1 se concluye que tienen un comportamiento lineal, o sea la función de ajuste es un polinomio de grado 1 de la forma

$$p(x) = a_1 x + a_2$$

- *De todas las posibles rectas se buscará aquella que se aproxime lo mejor posible a la nube de datos*

- *Se considera que no hay errores horizontales, solamente errores en las mediciones de la variable vertical*

Pero como los datos no están alineados no existe una recta que pase por todos los puntos, lo que realmente se buscará es la recta que pase lo más cerca posible de todos los datos. Esto significa que en cada ordenada habrá un error

El problema consiste en encontrar los valores de a_1 y a_2 que hacen "mínimo el error" o sea que "mejor ajuste" "mejor aproxime" los datos

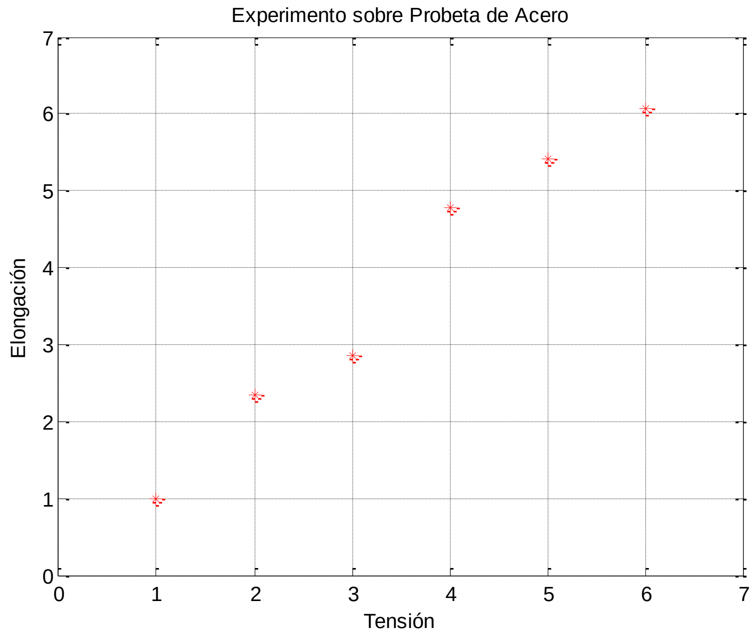


Figura 1

La primera idea que se nos ocurre es reemplazar los datos (x_i, y_i) $i = 1{:}N$ en la ecuación de la recta a determinar.

$$recta\ de\ aproximación \qquad \underline{p(x) = a_1 x + a_2}$$

Luego al reemplazar los datos en la ecuación de la recta queda

$$a_1[x_i] + a_2 = [y_i]$$

$$\begin{cases} 1.\, a_1 + a_2 = 1 \\ 2.\, a_1 + a_2 = 2.35 \\ 3.\, a_1 + a_2 = 2.85 \\ 4.\, a_1 + a_2 = 4.78 \\ 5.\, a_1 + a_2 = 5.42 \\ 6.\, a_1 + a_2 = 6.07 \end{cases}$$

Escrito en forma matricial es :

$$\underbrace{\begin{pmatrix} 1 & 1 \\ 2 & 1 \\ 3 & 1 \\ 4 & 1 \\ 5 & 1 \\ 6 & 1 \end{pmatrix}}_{A} \underbrace{\begin{pmatrix} a_1 \\ a_2 \end{pmatrix}}_{p} = \underbrace{\begin{pmatrix} 1 \\ 2.35 \\ 2.85 \\ 4.78 \\ 5.42 \\ 6.07 \end{pmatrix}}_{b} \qquad (*)$$

$$Ap = b$$

Obtenemos un sistema lineal de 6 ecuaciones con 2 incógnitas. Sistema sobredeterminado (más ecuaciones que incógnitas).

Como ya se analizó en la unidad de Sistemas Lineales, este sistema no tiene solución, tendría solución solamente si todos los puntos (datos) estuviesen alineados, lo cual no sucede en este ejemplo.

Lo que se busca es encontrar los valores de a_1, a_2 que hacen mínimo el error, definido como

$$E = Ap - b$$

Hay diferentes métodos para resolver este problema nosotros estudiaremos el método de Mínimos Cuadrados.

Como ya lo estudiamos en Sistemas Lineales, lo analizamos desde el punto de vista algebraico, pero también se puede analizar desde el punto de vista del Análisis

Idea Principal

La idea principal es aproximar la función con un polinomio según el “criterio de mínimos cuadrados” el cual consiste en hacer mínimo la norma 2 del error.

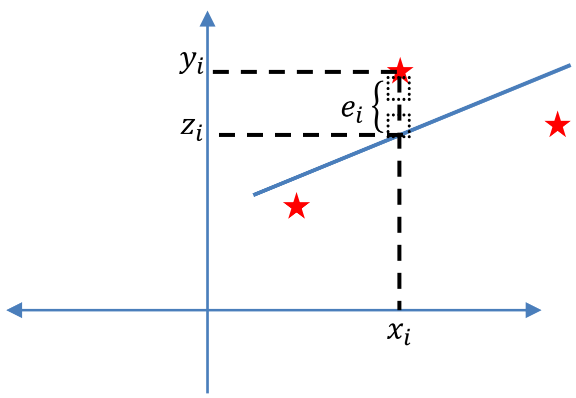


Figura 3

La medida que nos da cuando el vector error es muy pequeño es la norma del error. De todas las normas vectoriales se trabaja con la norma dos, que es una función continua y analíticamente es la más sencilla de trabajar (derivar , integrar).

Si consideramos el vector Error, cuyas componentes son los errores en cada punto al aproximar cada dato con la ordenada de la recta de ajuste

$$\vec{E} = Ap - b = [e_1, e_2, \dots, e_N]$$

Donde N es la cantidad de datos, (en nuestro ejemplo es 6). Entonces lo que yo busco es que la ***norma del error*** sea ***mínima***.

La norma del vector error está dada por

$$\|\vec{E}\| = \left(\sum_{i=1}^{\boldsymbol{N}} {e_i}^2\right)^{1/2}$$

Donde cada componente es:

$$e_i = \underbrace{y_i}_{dato} - \underbrace{z_i}_{\substack{Valor \\ aprox.}}$$

En este ejemplo

$$z_i = a_1 x_i + a_2$$

Entonces

$$e_i = y_i - a_1 x_i - a_2 \qquad (1)$$

Minimizar la norma es lo mismo que minimizar la norma al cuadrado , luego

$$\|\vec{E}\|^2 = \sum_{i=1}^{\boldsymbol{N}} {e_i}^2 \qquad (2)$$

Reemplazando $(1)\ en\ (2)$ se tiene:

$$\|\vec{E}\|^2 = \sum_{i=1}^{\boldsymbol{N=6}} {e_i}^2 = \sum_{i=1}^{6} (y_i - a_1 x_i - a_2)^2 = EM(a_1, a_2)$$

Se puede observar que $\|\vec{E}\|^2$ es una función de dos variable $EM(a_1, a_2)$ y como se estudió en Cálculo II, para hallar el mínimo de una función se deben calcular las derivadas parciales respecto a cada una de las variables e igualarlas a cero. La solución de ese sistema es un punto crítico. Luego

$$\begin{cases} \dfrac{\partial EM}{\partial a_1} = 2\sum_{i=1}^{6} (y_i - a_1 x_i - a_2)(-x_i) = 0 \\ \dfrac{\partial EM}{\partial a_2} = 2\sum_{i=1}^{6} (y_i - a_1 x_i - a_2)(-1) = 0 \end{cases}$$

Trabajando algebraicamente estas ecuaciones, considerando que $(x_i z_i)$ son datos, (constantes conocidas) se obtiene el siguiente sistema

$$\begin{cases} a_1 \sum_{i=1}^{N} x_i + \sum_{i=1}^{N} a_2 = \sum_{i=1}^{N} z_i \\ a_1 \sum_{i=1}^{N} {x_i}^2 + a_2 \sum_{i=1}^{N} x_i = \sum_{i=1}^{N} z_i x_i \end{cases}$$

Escrito en forma matricial y considerando que $\sum_{i=1}^{N} a_2 = N\, a_2$

$$\underbrace{\begin{pmatrix} \sum_{i=1}^{N} x_i & N \\ \sum_{i=1}^{N} {x_i}^2 & \sum_{i=1}^{N} x_i \end{pmatrix}}_{AEN} \underbrace{\begin{pmatrix} a_1 \\ a_2 \end{pmatrix}}_{w} = \underbrace{\begin{pmatrix} \sum_{i=1}^{N} z_i \\ \sum_{i=1}^{N} z_i x_i \end{pmatrix}}_{bEN}$$

$$AEN\, w = bEN$$

$$w = [a_1, a_2]^T$$

Sistema conocido con el nombre de *ECUACIONES NORMALES* QUE TIENE SOLUCIÓN , y es única. Este sistema es mal condicionado, de manera que la solución es inestable numéricamente. Esta es la forma que trabajan es estadística en Regresión Lineal

La solución de este sistema son los coeficientes de la recta de ajuste, para los cuales el error de aproximar los datos con la recta es mínimo.

Esta solución que se obtiene de las Ecuaciones Normales es la solución en el sentido de los mínimos cuadrados.

Para este ejemplo la solución de la Ecuaciones Normales es

$$a_1 = 1.0426$$

$$a_2 = 0.0960$$

Otra forma de obtener las Ecuaciones Normales

Como no siempre los datos tienen comportamiento lineal, el procedimiento anterior para obtener las Ecuaciones Normales se torna muy complicado, de allí la necesidad de recurrir, tal como se estudió en Sistemas Lineales, a las propiedades del Álgebra para obtener las Ecuaciones Normales

Repaso

Sea un sistema sobredeterminado obtenido al reemplazar los datos en la ecuación de la recta

$$Ap = b$$

Si pre multiplicamos por la transpuesta de A a ambos miembros se obtienen las Ecuaciones Normales

$$A^T A p = A^T b$$

Donde A^T es la matriz de coeficientes del sistema (*) y $p = [a_1, a_2]$

Recordando cómo son las matrices

$$A = \begin{pmatrix} 1 & 1 \\ 2 & 1 \\ 3 & 1 \\ 4 & 1 \\ 5 & 1 \\ 6 & 1 \end{pmatrix} \qquad p = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} \qquad b = \begin{pmatrix} 1 \\ 2.35 \\ 2.85 \\ 4.78 \\ 5.42 \\ 6.07 \end{pmatrix}$$

Recuerden que este sistema se obtiene de reemplazar los datos en la ecuación de la recta de ajuste y que las incógnitas son los coeficientes de la recta

$$[x_i]a_1 + a_2 = [y_i]$$

Podemos observar que la primera columna de los coeficientes son los valores de x y la segunda es una columna de unos

Primero se cargan los datos:

>>x=$[x_i]$;   y=$[y_i]$

Si los datos fueron introducidos como vectores filas , el comando (:) hace **siempre** vertical el vector, entonces para asegurar que las operaciones de las matrices se puedan realizar, es que se hacen verticales los vectores x e y

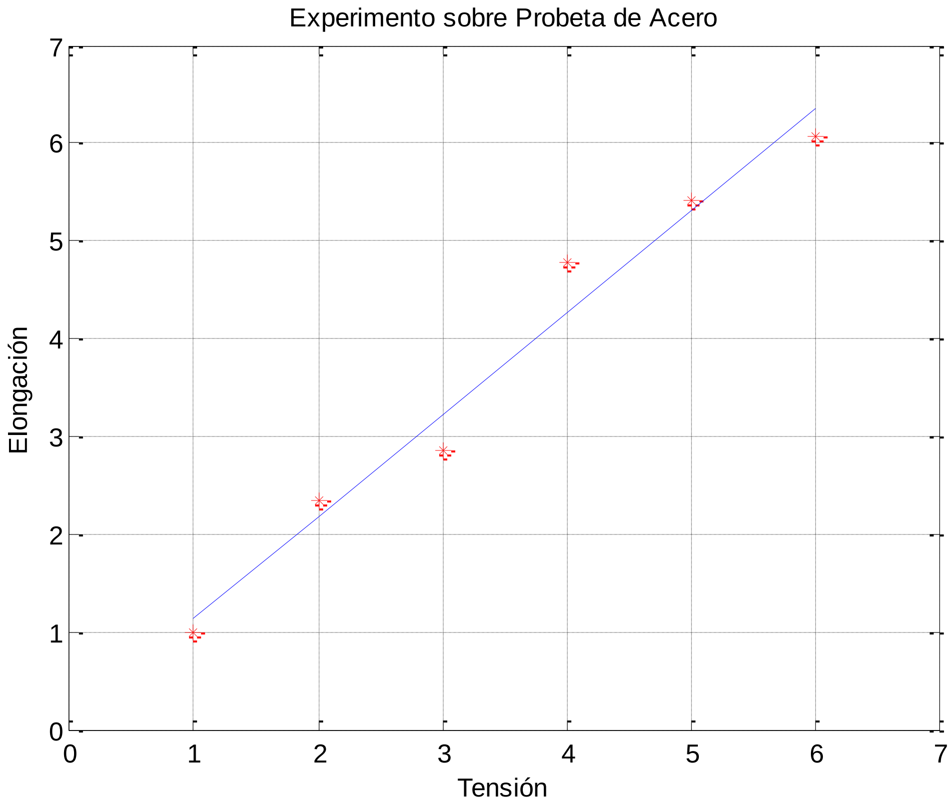
```
>>x=x(:);   y=y(:);          %  este comando los hace siempre verticales

>>A=[x  ones(N,1)];          % se genera la matriz de coeficientes

>>b=[y_i];                   % vector de términos independientes

>>p=inv(A'*A)*A'*b

>>p

     1.0426    0.0960
```

Para cumplir el objetivo propuesto se evalúa el polinomio en 3.5

```
>>s=polyval(p,3.5)

   s=

          3.7450
```

Como es una recta puedo evaluar la función en los datos y graficarlos

```
>> z=polyval(p,x);

>>hold on

>> plot(x,z,'b')
```



Esta es la gráfica de los puntos junto con el ajuste lineal

GENERALIZANDO

Supongamos que se tiene N datos, o sea N puntos de coordenadas (x_i, y_i) $\quad i = 1{:}N$ y se quiere realizar un ajuste de grado mayor a 1. El máximo grado que se puede elegir para hacer un ajuste es $m \leq N - 2$

Luego el polinomio de ajuste será

$$P_m = a_1 x^m + a_2 x^{m-1} + \cdots + a_m x + a_{m+1}$$

Luego al reemplazar los datos en la ecuación del polinomio se genera un sistema de N ecuaciones con $m + 1$ incógnitas.

$$a_1 {x_i}^m + a_2 {x_i}^{m-1} + \cdots + a_m x_i + a_{m+1} = y_i$$

Escrito en forma matricial será

$$Ap = b \qquad (1)$$

$$\begin{pmatrix} {x_1}^m & {x_1}^{m-1} & \dots & x_1 & 1 \\ {x_2}^m & {x_2}^{m-1} & \dots & x_2 & 1 \\ {x_3}^m & {x_3}^{m-1} & \dots & x_3 & 1 \\ \dots & \dots & \dots & \dots & \dots \\ {x_N}^m & {x_N}^{m-1} & \dots & x_N & 1 \end{pmatrix}_{N,m+1} \cdot \begin{pmatrix} a_1 \\ a_2 \\ \dots \\ a_{m+1} \end{pmatrix}_{m+1,1} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \dots \\ y_N \end{pmatrix}_{N,1}$$

Las Ecuaciones Normales son:

$$(A^T A)p = A^T b \quad (2)$$

Sistema (2) de $m+1$ ecuaciones con $(m+1)$ incógnitas, donde p es el vector que contiene los coeficientes del polinomio de grado m. Recordemos que la solución del sistema (2) es la solución del sistema original (1) en el sentido de los mínimos cuadrados, o sea la que hace mínimo el error.

Dados N datos

¿Cuál es el grado del polinomio de ajuste más adecuado para aproximar los datos?.

Supongamos que tenemos 100 datos, podemos elegir como polinomio de ajuste según la teoría un polinomio de grado 1, 2,…, hasta 98. Sin embargo un polinomio de grado 98 es muy difícil de trabajar sobre todo calcular en valores intermedios, derivarlo etc.

A veces no porque el grado sea mayor será mejor el ajuste (o sea menor el error)

Para resolver este problema tenemos una medida estadística que permite medir las desviaciones con respecto al valor central, de esa manera, puedo medir cuan dispersos están los errores, con respecto a un error promedio, esto es medido por la varianza

$$\sigma^2 = \frac{\sum_{i=1}^{N} {e_i}^2}{N-m-1}$$

Trabajamos en Octave o Matlab

Lo primero que debe realizarse es graficar los puntos y de esa manera observando la gráfica se puede elegir que polinomio de ajuste es el más adecuado. Los puntos no deben unirse, sino que deben trabajarse como puntos. De allí que su gráfica debe ser así

```
>>x=[x1; x2; … xN];

>>y=[y1; y2;…; yN];

>> plot(x,y,'*')
```

Se puede trabajar de dos formas diferentes:

- Como el sistema planteado en la teoría
  Supongamos que tenemos cargado lo datos en forma vertical , entonces se construye la matriz A y b

```
>>A=[x.^m  x.^(m-1)   …   x  ones(N,1)];
>>b=y
```

a) Resolviendo con el comando de Octave que calcula la solución de un sistema

```
>>p=A\b                           %(directamente da la solución de mínimos cuadrados)
```

b) Construyendo las Ecuaciones Normales

```
>>AEN=A'*A;
>>bEN=A'*b;
>>p= AEN\bEN;
```

- Usando el comando **polyfit** que Octave y Matlab tienen programado para hacer un ajuste, se debe ingresar como argumento de entrada la nube de datos y el grado de polinomio. Se obtiene el mismo polinomio que por las Ecuaciones Normales

```
>>pp=polyfit(x,z,m)            %  m es el grado del polinomio de ajuste
```

Para graficar el ajuste se evalúa el polinomio obtenido en un vector que tenga más puntos que los datos, pero debe estar comprendido en el rango de los valores de x tabulados. Fuera de ese rango no se puede decir nada. Por si los datos no están ordenados, se construye un vector que comience el menor valor de x y termine en el máximo valor de x

```
>> xx=min(x):.1:max(x);            % el paso se elige según los datos

>> zz=polyval(p,xx);

>>plot(x,y,'*',xx zz)
```

Para calcular la varianza al cuadrado y así comparar con el ajuste hecho con otros polinomios se procede de la siguiente manera:

1) se debe calcular el polinomio en los valores de x(datos)

2) se calculan los errores

3) si se observa el numerador de la varianza no es otra cosa que la norma del error al cuadrado

$$\sum_{i=1}^{N} {e_i}^2 = \left\|\vec{E}\right\|^2$$

La secuencia de comandos es

```
>>z=polyval(p,x);
```

```
>>E=y-z;

>>sigma2=norm(E)^2/(N-m-1)
```

INTERPOLACION

Anteriormente vimos como dado un conjunto de datos, se podía encontrar la función polinómica que pase los más cerca posible de los datos, (o sea que el error sea mínimo) para estimar el valor de la función en datos intermedios

Otra forma de resolver este problema es mediante el Polinomio de Interpolación

En Álgebra se vió que por N puntos pasa uno y solo un polinomio de grado N-1

Por ejemplo:

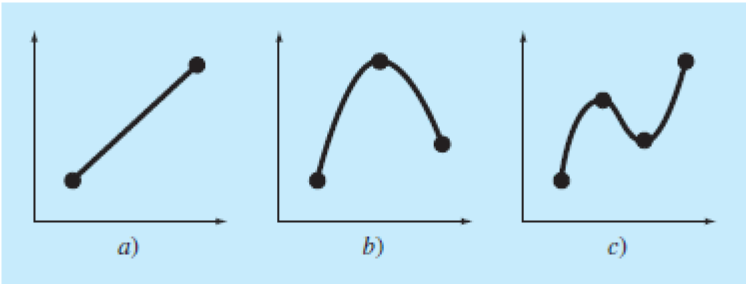


Figura 3

- En la Figura 3 a) Por dos puntos pasa una y sola una recta (polinomio de grado uno)
- En la Figura 3 b) Por tres puntos pasa una y solo una parábola
- En la Figura 3 c) Por cuatro puntos pasa una y sola una parábola cúbica

Interpolación Polinómica

Consiste en determinar el único polinomio de grado N-1 que pasa por N puntos. Este polinomio es único , y permite aproximar el valor de la función real en valores intermedios.

Existen diversos métodos para hallar dicho polinomio.

Interpolación Lineal

Es el polinomio que conecta dos puntos mediante una línea recta.

Consiste en hallar la recta que pasa por los puntos $(x_0, f(x_0))$, $(x_1, f(x_1))$. Utilizando semejanza de triángulos (Figura 4) se tiene:

$$\frac{f_1(x) - f(x_0)}{x - x_0} = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$$

Reordenando la expresión se tiene

$$\boxed{f_1(x) = f(x_0) + \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x - x_0)}$$

Fórmula de Interpolación Lineal. La notación f_1 es para designar que es un polinomio de interpolación de grado uno.

Se puede observar en la Figura4 que mientras más próximo están los puntos, la recta secante será más próxima a la curva, luego la aproximación de la función en los puntos intermedios será mejor

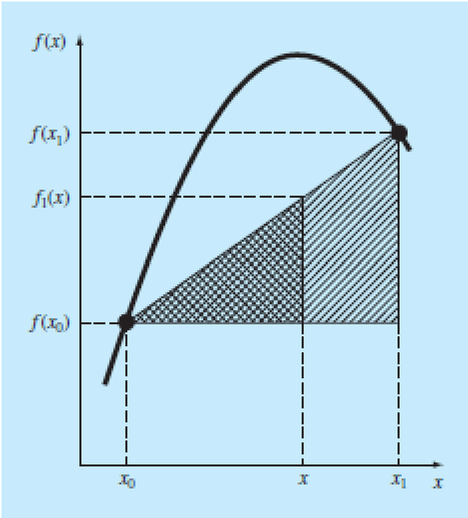


Figura 4

Ejemplo2

Hallar $\ln(2)$ mediante interpolación lineal, tomando los puntos $x_0 = 1 \quad x_1 = 6$.

Luego repetir el proceso en un intervalo más pequeño por ejemplo $x_0 = 1 \quad x_1 = 4$

Comparar con el valor sabiendo que $\ln(2) = 0.693$

Para el primer intervalo

$$f_{11}(x) = \ln(1) + \frac{\ln(6) - \ln(1)}{6 - 1}(x - 1)$$

$$f_1(2) = \frac{\ln(6)}{5} = 0.3584$$

Para el segundo intervalo

$$f_{12}(x) = \ln(1) + \frac{\ln(4) - \ln(1)}{4 - 1}(x - 1)$$

$$f_{12}(2) = \frac{\ln(4)}{3} = 0.4621$$

Si comparamos con el valor exacto

$$|f_{11}(2) - \ln(2)| = |0.3584 - 0.6931| = 0.3347$$

$$|f_{12}(2) - \ln(2)| = |0.4621 - 0.6931| = 0.2310$$

Es evidente que el error es menor cuando los puntos están más próximos al valor buscado.(Ver Figura 5)

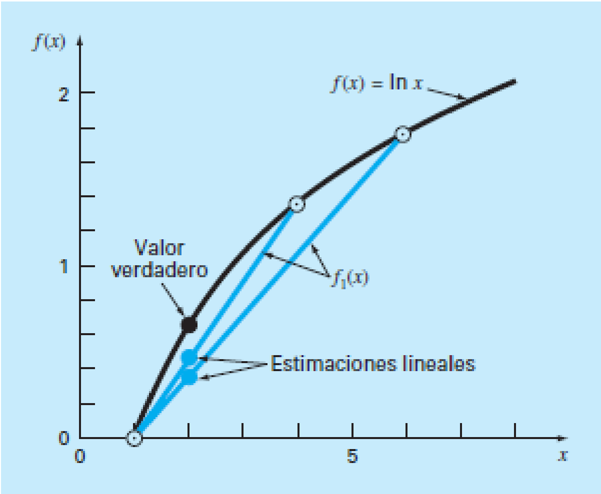


Figura 5

Método Directo

Para polinomios de mayor grado es más conveniente usar el método directo.

Supongamos que tenemos N puntos y queremos encontrar el polinomio de interpolación que pasa por dichos puntos. Este polinomio será de grado $N - 1$, cuya expresión está dad por:

$$P_{N-1} = a_1 x^{N-1} + a_2 x^{N-2} + \cdots + a_{N-1} x + a_N$$

El método directo para calcular los coeficientes de este polinomio, consiste en que se necesitan N puntos para determinar los N coeficientes. Así se utiliza un sistema de ecuaciones algebraicas lineales generado al reemplazar los puntos en la ecuación del polinomio, y como sabemos que por N puntos pasa un polinomio de grado N-1 este sistema tiene siempre solución

Ejemplo 3

Hallar el polinomio de interpolación que pasa por los puntos $(x_1, y_1), (x_2, y_2), (x_3, y_3)$. Como debe pasar por los tres puntos será un polinomio de segundo grado de la forma

$$P_2 = a_1 x^2 + a_2 x + a_3$$

Al reemplazar los puntos en el polinomio se genera el siguiente sistema

$$a_1 {x_i}^2 + a_2 x_i + a_3 = y_i$$

$$\begin{pmatrix} {x_1}^2 & x_1 & 1 \\ {x_2}^2 & x_2 & 1 \\ {x_3}^2 & x_2 & 1 \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix}$$

Sistema de tres ecuaciones con tres incógnitas que tiene solución única

En Matlab se procede así

```
>>x=[x1 x2 x3]
>>y=[y1 y2 y3]
>>A=[x.^2  x  ones(3,1)];
>>b=y
>>p2=A\b
```

También puede usarse el comando polyfit

```
>>p2=polyfit(x,y,2)
```

Para graficar el polinomio de interpolación deben tomarse más puntos así la gráfica queda más suave, pero siempre valores intermedios entre el mínimo y el máximo valor de los x_i

Diferencia entre Ajuste e Interpolación

La diferencia entre polinomio de ajuste y polinomio de interpolación es :

- El polinomio de ajuste no pasan por los puntos y se busca de todos los posibles polinomios de grado m, el que hace mínimo el error cuadrático.
- El polinomio de Interpolación pasa por cada uno de los puntos. El error en dichos puntos es cero.
- Existe más de un polinomio de ajuste, para N puntos
- El polinomio de Interpolación es único.
- El grado del polinomio de ajuste varía hasta $m \leq N - 2$
- Para N puntos el grado del Polinomio de Interpolación es $N - 1$

Ejemplo 4

1) Dada la función $(x) = sen(x) + 8\ cos(x)$; genere una nube de datos (xi, yi) tomando

14 puntos, en el intervalo $[0\ \pi]$, siendo $yi = (xi)$. Haga el gráfico de dispersión. Luego:

a) Ajuste con Polinomios de Primer, Segundo y Tercer Grado aplicando mínimos cuadrado

b) Grafique la función y sus ajustes.

c) Halle el valor aproximado de la función en $x = {}^{\pi}/_{2}$ con cada uno de los polinomios de ajuste. Compare los resultados con el valor exacto

d) Halle la interpolación lineal para calcular en forma aproximada en $x1 = {}^{\pi}/_{2}$

Solución

En Matlab construyo la función dada y el vector x con las condiciones dadas

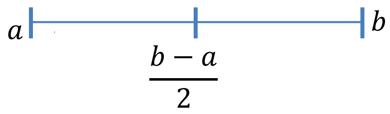
>> f=@(x) sin(x)+8*cos(x);

Para construir los 14 puntos en el intervalo $[0\ \pi]$ debo calcular el paso de tal manera que obtenga 14 puntos. Para ello dado un intervalo (a, b) para obtener n-puntos debo elegir

$$h = \frac{b-a}{n-1}$$

Veamos si esto se cumple:

– Supongamos que quiero 3 puntos debo dividir el intervalo en dos partes iguales y así obtengo tres puntos , luego $h = \frac{b-a}{2}$



– Si divido la longitud del intervalo en cuatro partes obtengo cinco puntos

Luego debo construir el vector x de la siguiente manera:

>> h=(pi-0)/13

h =

0.2417

>> x=0:h:pi;

>> y=f(x);

a)

- Ajuste con polinomio de primer grado

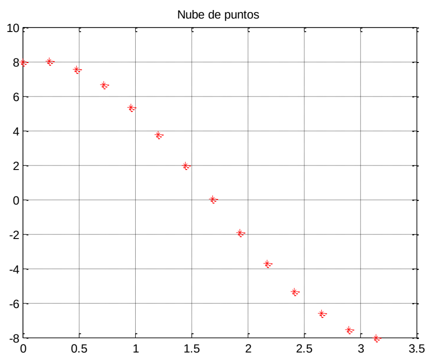
  >> plot(x,y,'r*')

  >> p1=polyfit(x,y,1)

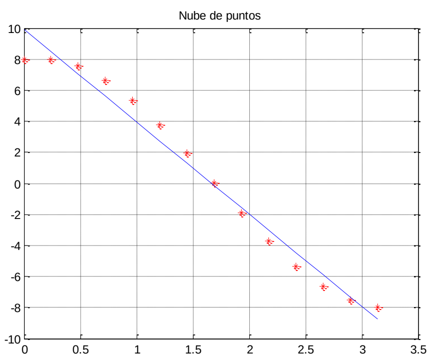
  p1 =

  -5.9535   9.9400

  Obteniendo la siguiente gráfica



Nube de puntos



Ajuste lineal

La gráfica del ajuste se realiza de la siguiente forma:

```
>> z=polyval(p1,x);
>> hold on
>> plot(x,z,'b')
```

- Ajuste con polinomio de Segundo grado

```
>> p2=polyfit(x,y,2)
p2 =
   -0.4075   -4.6734    9.3213
>> xx=min(x):.1:max(x);
>> zz=polyval(p2,xx);
>> plot(xx,zz,'c')
```

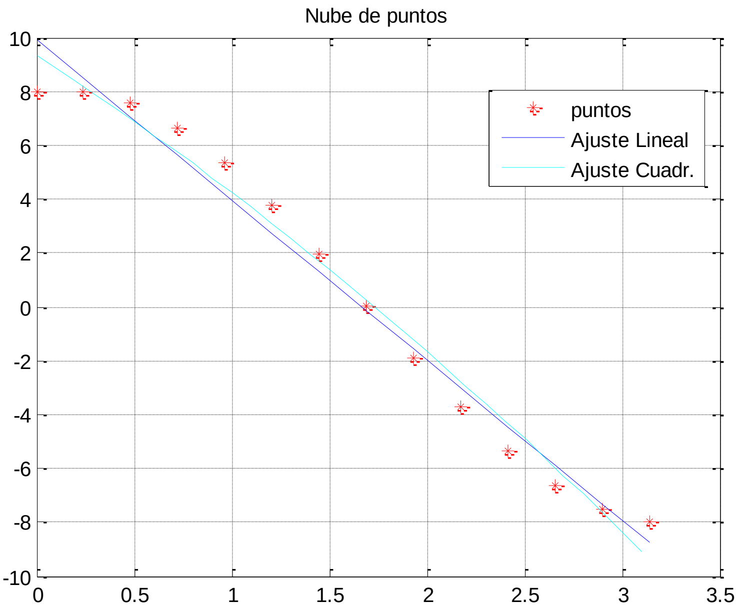


Figura : Ajuste Lineal y Cuadrático

- El ajuste de tercer grado se deja como ejercicio

c) Se debe calcular el valor del polinomio de ajuste en el punto indicado.

% Evaluación con el Polinomio de ajuste de grado 1

```
>> z1=polyval(p1,pi/2)
z1 =
   0.5883
```

% Evaluación con el Polinomio de ajuste de grado 2

```
>> z2=polyval(p2,pi/2)
z2 =
```

```
    0.9749
```

%Cálculo del valor exacto

```
>> yexac=f(pi/2)

yexac =

    1.0000
```

Se puede observar que es mejor el ajuste de segundo grado que el de primer grado.

e) Para calcular la interpolación lineal se necesitan dos puntos, y como el punto en cuestión es $x1 = {}^{\pi}/_{2}$, se deben tomar los puntos de la nube que están más próximos a x1 uno a la derecha y el otro a la izquierda. Entonces debemos hallar los valores del intervalo de x, que están próximos a x1. Esto se realiza con el comando **find**

```
>> j=find(x<pi/2)

j =

     1     2     3     4     5     6     7
```

Estas son las ubicaciones de los elementos de x menores a x1. El que interesa es el que está más cerca, o sea el que está en la posición 7. O lo que es lo mismo decir el que está en la posición max(i). Luego construimos un intervalo de dos puntos, que son los valores de x que se encuentran antes y después de ${}^{\pi}/_{2}$, de la siguiente manera:

```
>> i=max(j)

i =

     7

>> xl=[x(i) x(i+1)]

xl =

    1.4500    1.6916

>> pi/2=  1.5708
```

Como ven los valores que forman el vector **xl** están a la derecha y a la izquierda de ${}^{\pi}/_{2}$

Para formar la nube de datos necesito los valores del vector **y** correspondientes a los valores del vector **x** que forman **xl**

```
>> yl=[y(i) y(i+1)]

yl =

    1.9570    0.0284
```

De esta forma tengo los dos pares de puntos por lo que pasará la recta de interpolación

```
>> pl=polyfit(xl,yl,1)
```

pl =

-7.9805 13.5285

Luego si evaluamos esta recta en $\pi/2$ será

y_intl=polyval(pl,pi/2)

y_intl =

0.9927

Como pueden observar es mucho mejor que el valor obtenido con el ajuste, y muy cercano al ajuste cuadrático, y el cálculo es más sencillo