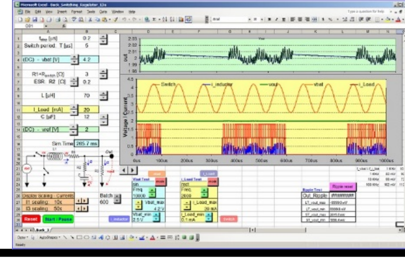
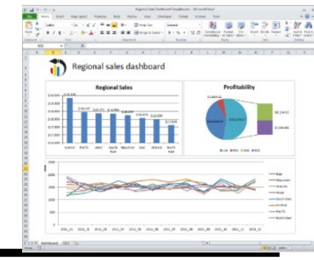




HOJAS DE CÁLCULO

Excel básico

Pedro Corcuera

Dpto. Matemática Aplicada y Ciencias de la Computación

Universidad de Cantabria

corcuerp@unican.es

Índice

- Resolución de ecuaciones
- Sistemas de ecuaciones lineales
- Series
- Interpolación
- Evaluación de derivadas
- Evaluación de integrales
- Optimización

Objetivos

- Aplicación de hojas de cálculo en problemas de ingeniería.

Resolución de ecuaciones

Resolviendo ecuaciones

- En ingeniería es frecuente la tarea de resolver ecuaciones algebraicas complicadas o sistemas de ecuaciones no lineales.
- Hay métodos manuales y computarizados para resolver tal problema, como son el método de Newton y la eliminación gaussiana.
- Excel dispone de funciones y herramientas para ayudar a cumplir esa tarea.
- Ejemplos: Resolviendo_ecuaciones.xls

Resolviendo ecuaciones

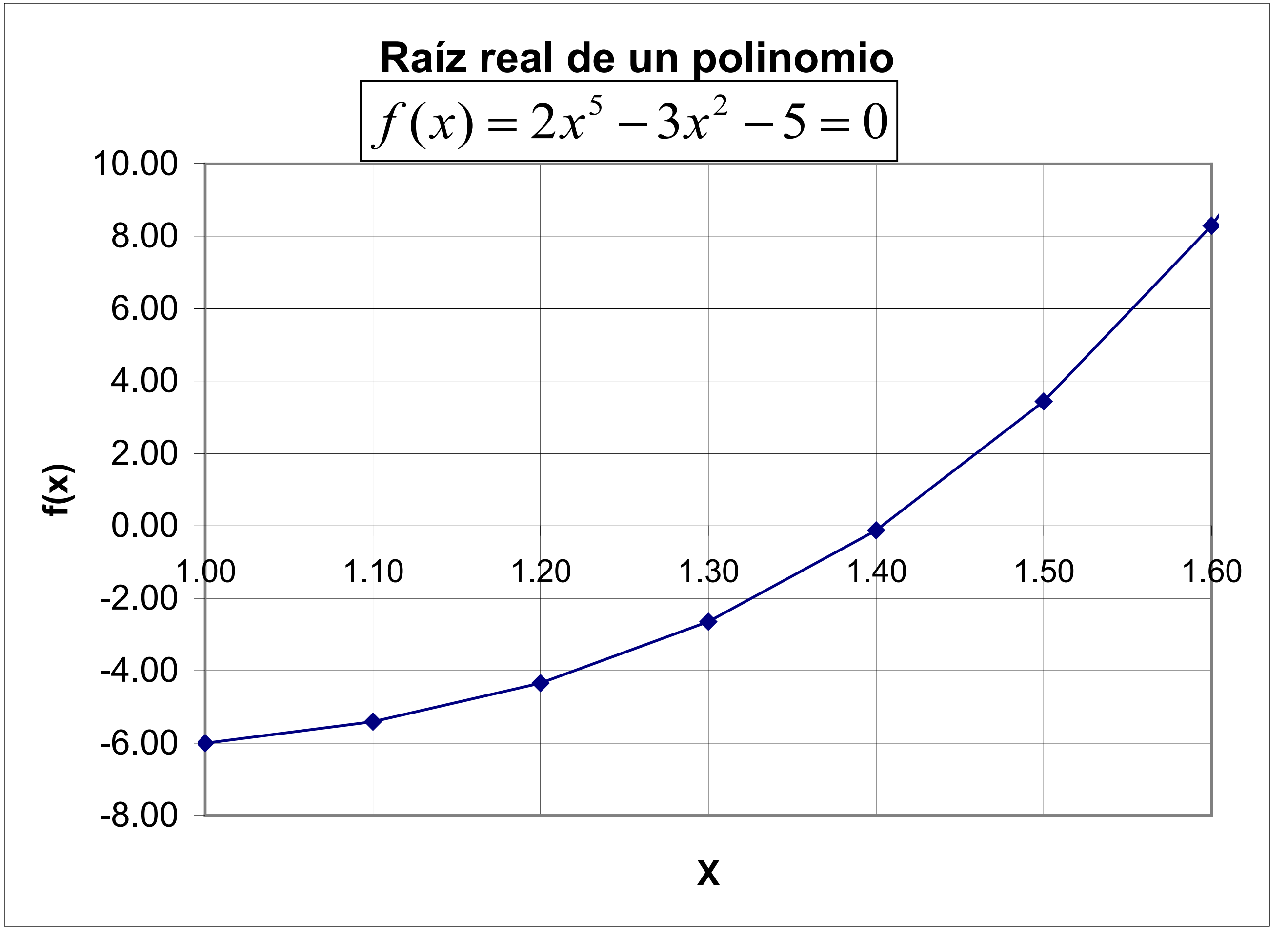
- La raíz de una ecuación algebraica es el valor de la variable independiente que satisface la ecuación.
- Las ecuaciones pueden ser lineales o no lineales.
- Las ecuaciones no lineales se pueden resolver de forma gráfica o numérica y pueden tener múltiples raíces reales o complejas.
- Las ecuaciones polinómicas son un caso especial de ecuaciones no lineales muy frecuentes en ingeniería con las siguientes características:
  - Un polinomio de grado n no puede tener más de n raíces reales.
  - Si el grado de un polinomio es impar, siempre tendrá al menos una raíz real.
  - Las raíces complejas siempre existen en pares de conjugadas complejas.

Resolviendo ecuaciones – método gráfico

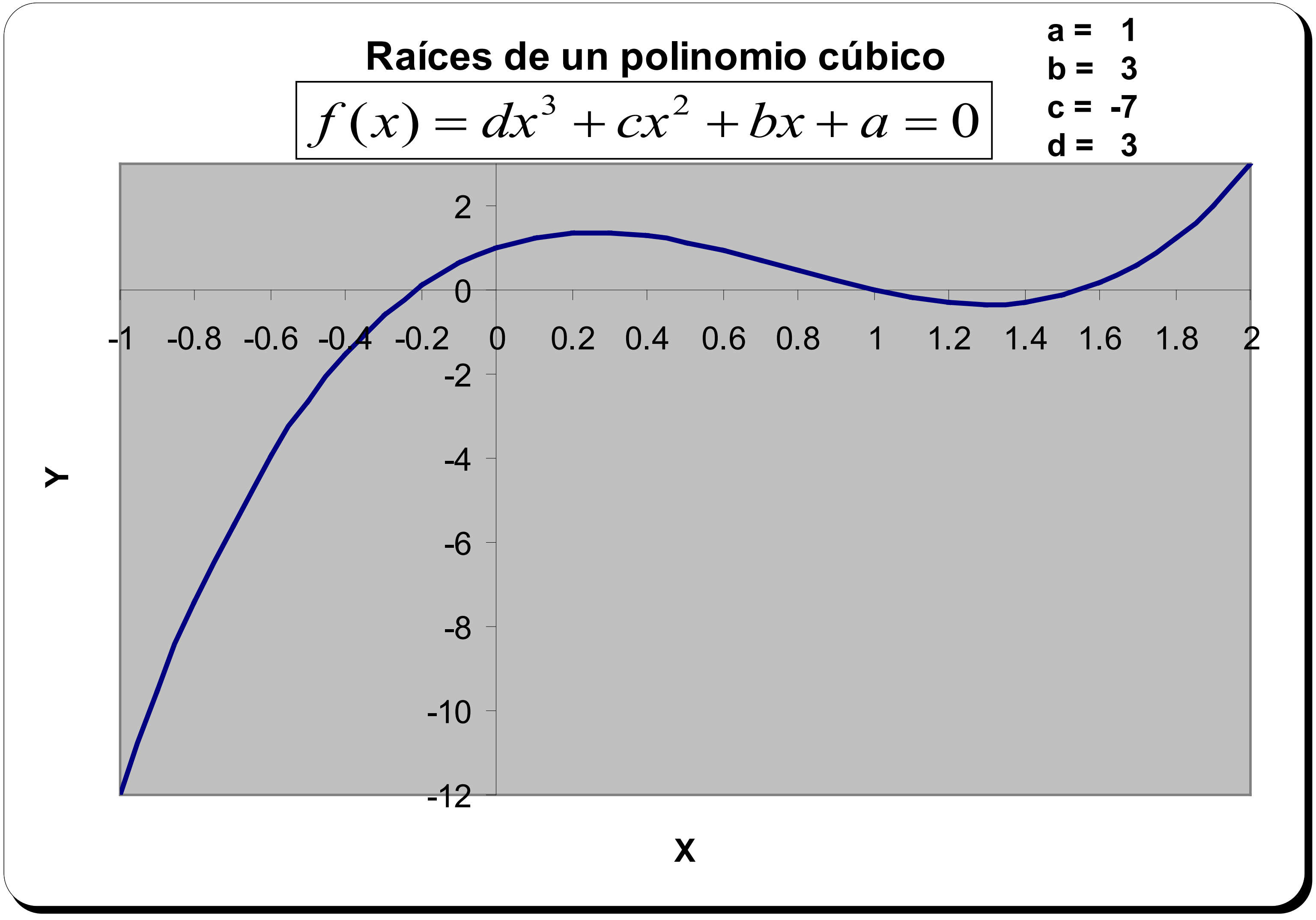
- El procedimiento es escribir la ecuación en la forma $f(x) = 0$ y graficas $f(x)$ vs. x.
- El punto donde $f(x)$ cruza el eje x (valor de x que causa que $f(x)$ sea 0) son las raíces reales de la ecuación.
- La solución se puede leer directamente del gráfico o interpolar entre los valores tabulados para hallar el punto donde $f(x) = 0$.

Resolviendo ecuaciones – método gráfico

x	f(x)
0	-5.00
0.1	-5.03
0.2	-5.12
0.3	-5.27
0.4	-5.46
0.5	-5.69
0.6	-5.92
0.7	-6.13
0.8	-6.26
0.9	-6.25
1	-6.00
1.1	-5.41
1.2	-4.34
1.3	-2.64
1.4	-0.12
1.5	3.44
1.6	8.29
1.7	14.73
1.8	23.07
1.9	33.69
2	47.00



Resolviendo ecuaciones – método gráfico

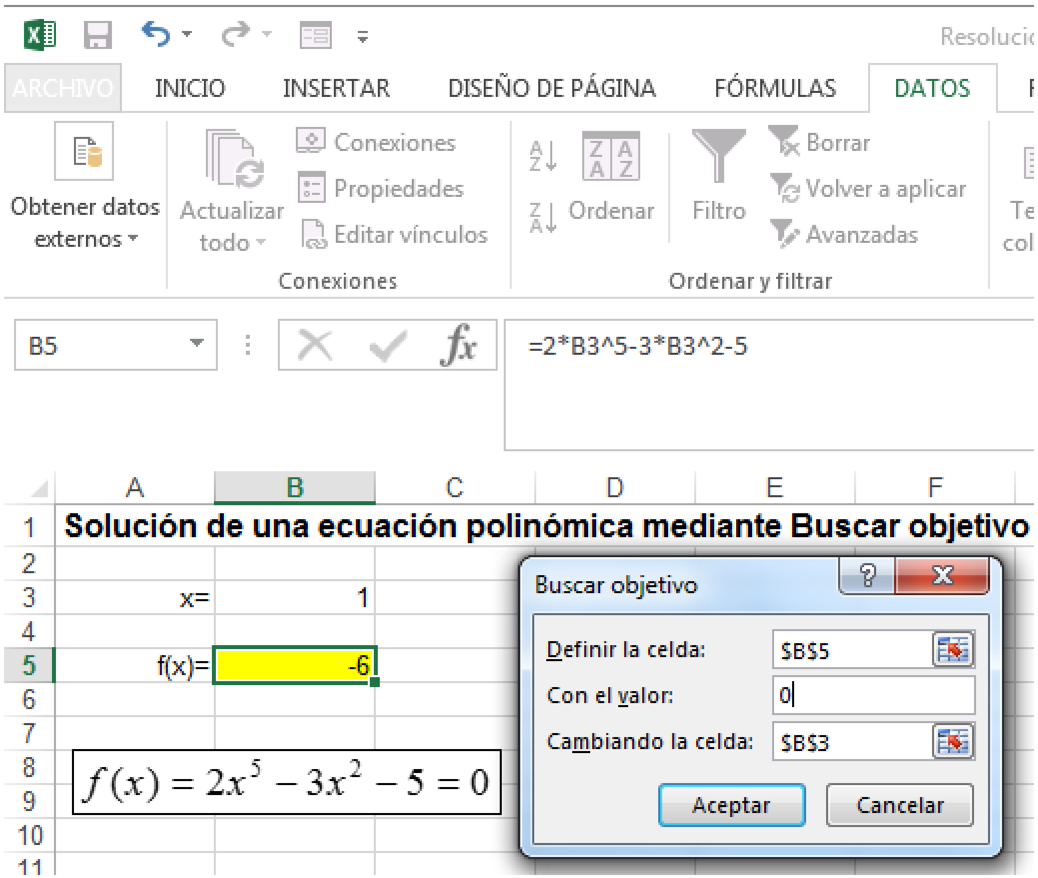


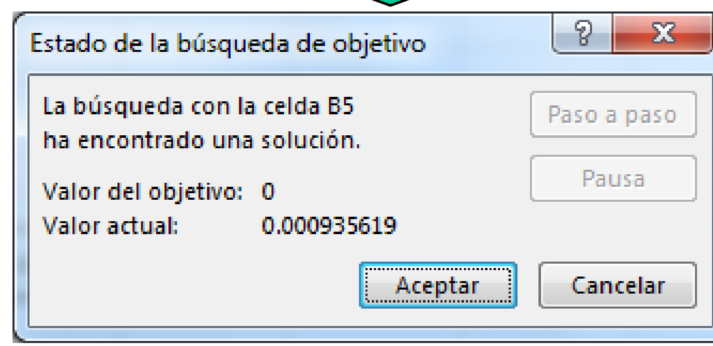
Resolviendo ecuaciones – usando Buscar objetivo

- Se puede obtener una solución rápida de ecuaciones algebraicas simples usando la opción Buscar Objetivo en el menú Datos → Análisis de hipótesis.
- Para ello se sigue:
  - Escribir un valor inicial de x en una celda.
  - Escribir la fórmula de la ecuación en la forma f(x)=0 en otra celda. Escribir la variable x como referencia a la celda que contiene el valor inicial.
  - Seleccionar Buscar Objetivo en el menú Datos → Análisis de hipótesis.
  - En el diálogo escribir la dirección de la celda que contiene la fórmula, el valor 0 en Valor y la dirección de la celda que contiene el valor inicial. Pulsar Aceptar.

Resolviendo ecuaciones – usando Buscar objetivo

- Ejemplo: $f(x) = 2^*x^5 – 3^* x^2 – 5 = 0$





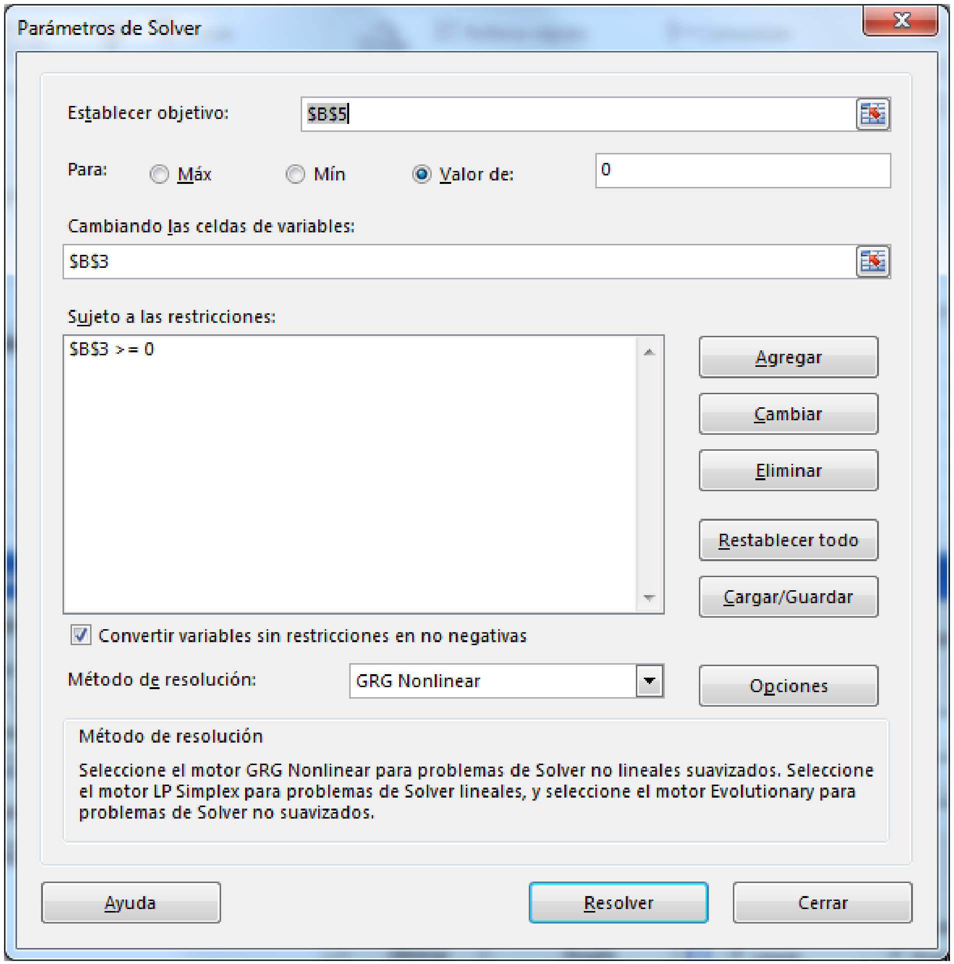
x= 1.40411692

Resolviendo ecuaciones – usando Solver

- Solver se usa para resolver problemas de más complejidad y se puede configurar el método y visualización de solución.
- **Instalar Solver** desde Archivo → Opciones → Complementos.
- Para ello se sigue:
  - Escribir un valor inicial de la variable en una celda.
  - Escribir la fórmula de la ecuación en formato f(x)=0 en otra celda. indicando la variable x como referencia a la celda con el valor inicial.
  - Seleccionar Solver en el menú Datos. En la ventana Solver indicar en Establecer objetivo la dirección de la celda que contiene la fórmula, el valor 0 en Valor de y la dirección de la celda(s) que contiene el valor inicial en Cambiando las celdas de variables.
  - Con Agregar se puede restringir el rango de x. Asegurarse que el método de resolución es GRG Nonlinear.
  - Pulsar Resolver. Se puede configurar con Opciones.

Resolviendo ecuaciones – usando Buscar objetivo

- Ejemplo: $f(x) = 2*x^5 – 3* x^2 – 5 = 0$



x= 1.40408619

Resolviendo sistemas de ecuaciones

- Las ecuaciones algebraicas simultáneas se presentan habitualmente en problemas de ingeniería.
- Los sistemas de ecuaciones pueden ser *lineales* o *no lineales*.
- Las técnicas para resolver sistemas de ecuaciones lineales son diferentes que las usadas para las no lineales.
- Para sistemas lineales se usa la notación matricial: [**A**][**x**] = [**b**] donde [**A**] es una matriz *nxn* y **x** y **b** son vectores de *n* elementos.

Operaciones matriciales en Excel

- En Excel las matrices se representan como arrays. Un array es un bloque de celdas que se referencian colectivamente.
- Cualquier operación que se realiza sobre un array producirá lo mismo para todas las celdas dentro del array. Un array se puede especificar como un argumento simple de una función.
- Un array se especifica como un bloque de celdas encerradas entre llaves ({}). Las llaves son añadidas automáticamente por Excel y **NO** deben escribirse.

Operaciones matriciales en Excel

- Para especificar una operación con array:
  - Seleccionar el bloque que forma el array. Mover el cursor hasta la celda superior izquierda dentro del bloque.
  - Escribir una fórmula con el array. Se puede incluir rangos de celdas dentro de la celda.
  - Pulsar **Ctrl-Mayús-Enter** para que la fórmula aparezca entre llaves.
- Operaciones habituales con matrices:
  - Suma, Resta, Transpuesta (TRANSPONER)
  - Multiplicación matricial (MMULT)
  - Determinante, Inversa (MDETER, MINVERSA)

Operaciones matriciales en Excel

$$[A] = \begin{vmatrix} 2 & 3 & -4 \\ 3 & -1 & -2 \\ 4 & -7 & -6 \end{vmatrix}$$

$$Det[A] = 82$$

$$Inverse\ [A] = \begin{vmatrix} -0.09756098 & 0.56097561 & -0.12195122 \\ 0.12195122 & 0.04878049 & -0.09756098 \\ -0.20731707 & 0.31707317 & -0.13414634 \end{vmatrix}$$

$$Transpose\ [A] = \begin{vmatrix} 2 & 3 & 4 \\ 3 & -1 & -7 \\ -4 & -2 & -6 \end{vmatrix}$$

$$[M] = \begin{vmatrix} 1 & 4 & 3 \\ 5 & 2 & 6 \end{vmatrix}$$

$$[N] = \begin{vmatrix} 7 & 12 \\ 11 & 8 \\ 9 & 10 \end{vmatrix}$$

$$[M][N] = \begin{vmatrix} 78 & 74 \\ 111 & 136 \end{vmatrix}$$

$$A + Trans[A] = \begin{vmatrix} 1.90243902 & 3.56097561 & -4.12195122 \\ 3.12195122 & -0.95121951 & -2.09756098 \\ 3.79268293 & -6.68292683 & -6.13414634 \end{vmatrix}$$

$$A - Trans[A] = \begin{vmatrix} 0 & 0 & -8 \\ 0 & 0 & 5 \\ 8 & -5 & 0 \end{vmatrix}$$

Operaciones matriciales en Excel

- Problema: calcular la inversa de una matriz compleja.
- Sea una matriz compleja $\mathbf{X} + j\mathbf{Y}$ cuya inversa es $\mathbf{U} + j\mathbf{V}$. Por definición de la inversa y el producto de una matriz y su inversa: $\mathbf{XU} - \mathbf{YV} = \mathbf{I}$ y $\mathbf{YU} + \mathbf{XV} = \mathbf{0}$ de donde se obtiene: $\mathbf{V} = (-\mathbf{Y}^{-1}\mathbf{X} - \mathbf{X}^{-1}\mathbf{Y})^{-1}\mathbf{X}^{-1}$

$$\mathbf{U} = -\mathbf{Y}^{-1}\mathbf{XV}$$

Operaciones matriciales en Excel

Inversa de una Matriz Compleja

X	Real
2	0.6
0.6	3

Imgy	Y
0.4	0.5
0.15	0.2

X INV	
0.53191489	-0.10638298
-0.10638298	0.35460993

Y INV	
40	-100
-30	80

YINV * X	
20	-276
-12	222

X INV * Y	
0.19680851	0.24468085
0.0106383	0.0177305

(Y INV * X + X INV * Y)	
20.1968085	-275.755319
-11.9893617	222.01773

(Y INV * X + X INV * Y)$^{-1}$	
0.18848297	0.23410374
0.01017842	0.01714617

V = -(Y INV * X + X INV * Y)$^{-1}$ * X INV	
0.07535224	0.06296413
0.00358999	0.00499739

-0.07535224	-0.06296413
-0.00358999	-0.00499739

U = - YINV * X * V	
-0.51620653	0.11999762
0.10724825	-0.35385149

0.51620653	-0.11999762
-0.10724825	0.35385149

Comprobación

X * U	
-0.96806411	0.02768435
0.01202084	-0.9895559

Y * V	
0.03193589	0.02768435
0.01202084	0.0104441

X*U - Y*V	
1	-3.6776E-16
-5.2389E-16	1

Y * U	
-0.15285849	-0.1289267
-0.05598133	-0.05277066

X * V	
0.15285849	0.1289267
0.05598133	0.05277066

Y * U + X * V	
0	0
0	0

Solución de sistemas de ecuaciones lineales mediante matrices

- Un método para resolver un sistema de ecuaciones lineales simultáneas [**A**][**x**] = [**b**] es mediante métodos matriciales [**x**] = [**A**]$^{-1}$ [**b**]
- Para resolver en Excel:
  – Escribir los elementos de la matriz **A**.
  – Escribir los elementos del vector **b**.
  – Seleccionar las celdas para la inversa **A**$^{-1}$. Escribir la fórmula en la celda superior izquierda =MINVERSA() y seleccionar las celdas de **A**. Pulsar Ctrl-Mayus-Enter simultáneamente.
  – Seleccionar las celdas donde se desea aparezca el vector **x**. Escribir la fórmula en la celda superior =MMULT() y seleccionar las celdas de **A**$^{-1}$ y **B** . Pulsar Ctrl-Mayus-Enter simultáneamente.

Solución de sistemas de ecuaciones lineales mediante matrices

Solución de Ecuaciones Simultáneas mediante inversión matricial

[A][x] = [b]

$$[A] = \begin{vmatrix} 9.375 & 3.042 & -2.437 \\ 3.042 & 6.183 & 1.216 \\ -2.437 & 1.216 & 8.443 \end{vmatrix}$$

$$[b] = \begin{vmatrix} 9.233 \\ 8.205 \\ 3.934 \end{vmatrix}$$

$$Inv[A] = \begin{vmatrix} 0.148 & -0.084 & 0.055 \\ -0.084 & 0.214 & -0.055 \\ 0.055 & -0.055 & 0.142 \end{vmatrix}$$

$$[x] = Inv[A]\ [b] = \begin{vmatrix} 0.896 \\ 0.765 \\ 0.614 \end{vmatrix}$$

Comprob. [A][x] = [b]

$$\begin{vmatrix} 9.2333 \\ 8.2049 \\ 3.9339 \end{vmatrix}$$

Solución de sistemas de ecuaciones usando Solver$_1$

- Solver ofrece un enfoque diferente para resolver sistemas ecuaciones simultáneas lineales o no lineales.
- Suponiendo que se tiene un sistema de *n* ecuaciones y *n* incógnitas representados mediante las ecuaciones:

$$
\begin{aligned}
& f_1(x_1, x_2, \ldots, x_n) = 0 \\
& f_2(x_1, x_2, \ldots, x_n) = 0 \\
& \ldots \\
& f_n(x_1, x_2, \ldots, x_n) = 0
\end{aligned}
$$

- Se desea hallar los valores de x_1, x_2 …, x_n que produce que cada ecuación sea cero. Una forma para hacer esto es forzar a que la función (varianza residual):

$y = f_1^2 + f_2^2 + \ldots + f_n^2$ sea cero.

Solución de sistemas de ecuaciones usando Solver$_2$

- Procedimiento:
  - Escribir un valor inicial para cada variable independiente x_1, x_2 …, x_n en celdas diferentes
  - Escribir las ecuaciones f_1 , f_2 , …, f_n e y en celdas diferentes expresadas como fórmulas dependientes de las celdas donde están las variables x_1, x_2 …, x_n
  - Seleccionar Solver de la Barra de herramientas. Dar la dirección de la celda que contiene la fórmula de y para Celda Objetivo. Seleccionar Valores de 0. En Cambiando las celdas dar el rango de las celdas que contienen los valores iniciales de las variables x_1, x_2 …, x_n.
  - Se puede restringir opcionalmente el rango de los valores de las variables independientes pulsando Agregar.
  - Se puede seleccionar la opción de generar Resultados en otra hoja.

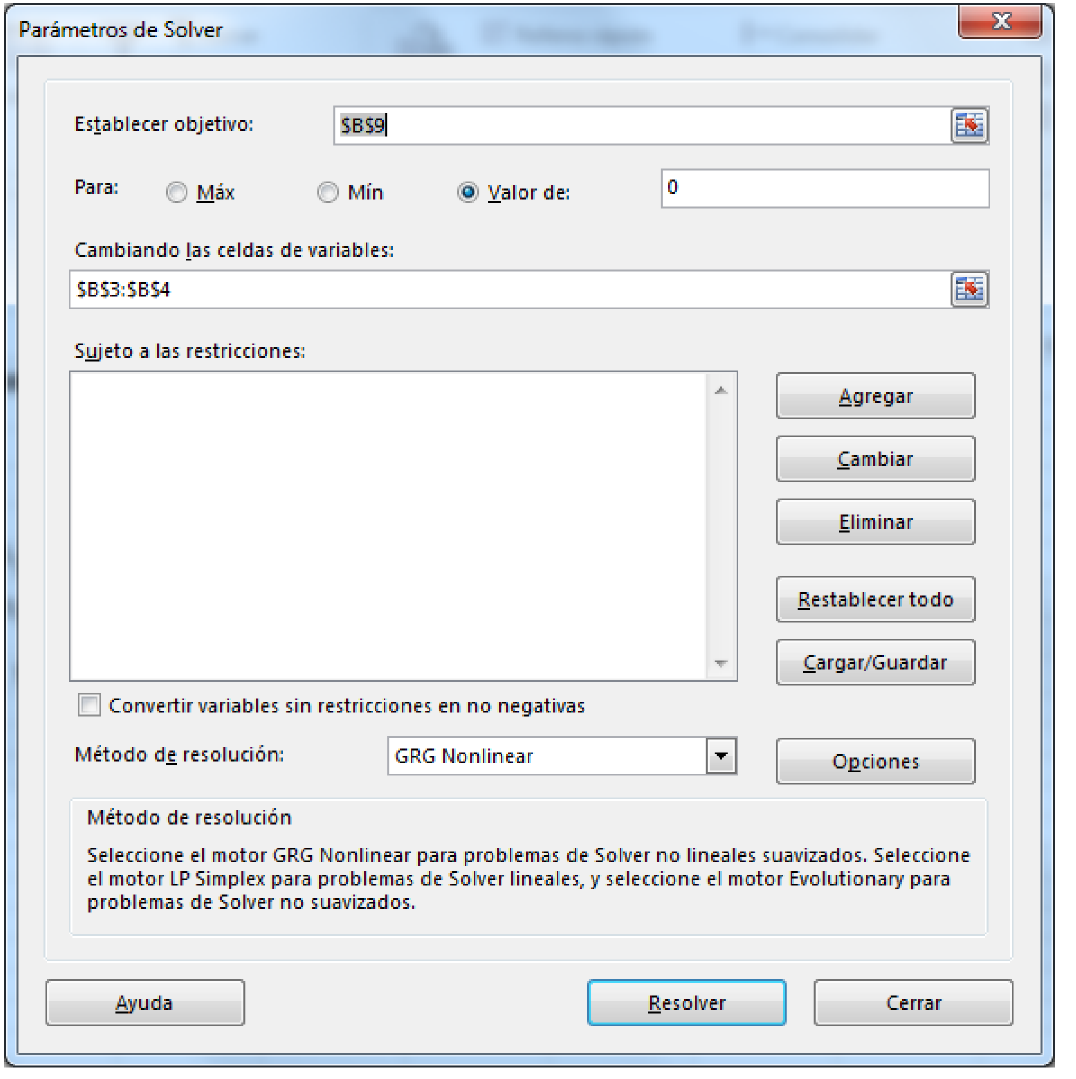
Solución de sistemas de ecuaciones usando Solver$_3$

- Ejemplo 1 (lineal):

$$3x_1 + 2x_2 - 2x_3 = 4$$
$$2x_1 - x_2 + x_3 = 3$$
$$x_1 + x_2 - 2x_3 = -3$$

$$f = 3x_1 + 2x_2 - 2x_3 - 4 = 0$$
$$g = 2x_1 - x_2 + x_3 - 3 = 0$$
$$h = x_1 + x_2 - 2x_3 + 3 = 0$$

$$y = f^2 + g^2 + h^2$$



Solución de sistemas de ecuaciones usando Solver$_3$

- Ejemplo 1 (lineal):

$$3\,x_1 + 2\,x_2 - 2\,x_3 = 4$$
$$2\,x_1 - x_2 + x_3 = 3$$
$$x_1 + x_2 - 2\,x_3 = -3$$

$$f = 3\,x_1 + 2\,x_2 - 2\,x_3 - 4 = 0$$
$$g = 2\,x_1 - x_2 + x_3 - 3 = 0$$
$$h = x_1 + x_2 - 2\,x_3 + 3 = 0$$

$$y = f^2 + g^2 + h^2$$

x1 =	1,428567883
x2 =	4,142852593
x3 =	4,28570703
f(x1,x2,x3) =	-5,2244E-06
g(x1,x2,x3) =	-9,797E-06
h(x1,x2,x3) =	6,41664E-06
y =	1,64449E-10

Solución de sistemas de ecuaciones usando Solver$_4$

- Ejemplo 2 (no lineal):

$$x_1^2 + 2\,x_2^2 - 5\,x_1 + 7\,x_2 = 40$$
$$3\,x_1^2 - x_2^2 + 4\,x_1 + 2\,x_2 = 28$$

$$f\,(x_1, x_2) = x_1^2 + 2\,x_2^2 - 5\,x_1 + 7\,x_2 - 40 = 0$$
$$g\,(x_1, x_2) = 3\,x_1^2 - x_2^2 + 4\,x_1 + 2\,x_2 - 28 = 0$$

$$y\,(x_1, x_2) = f^2 + g^2$$

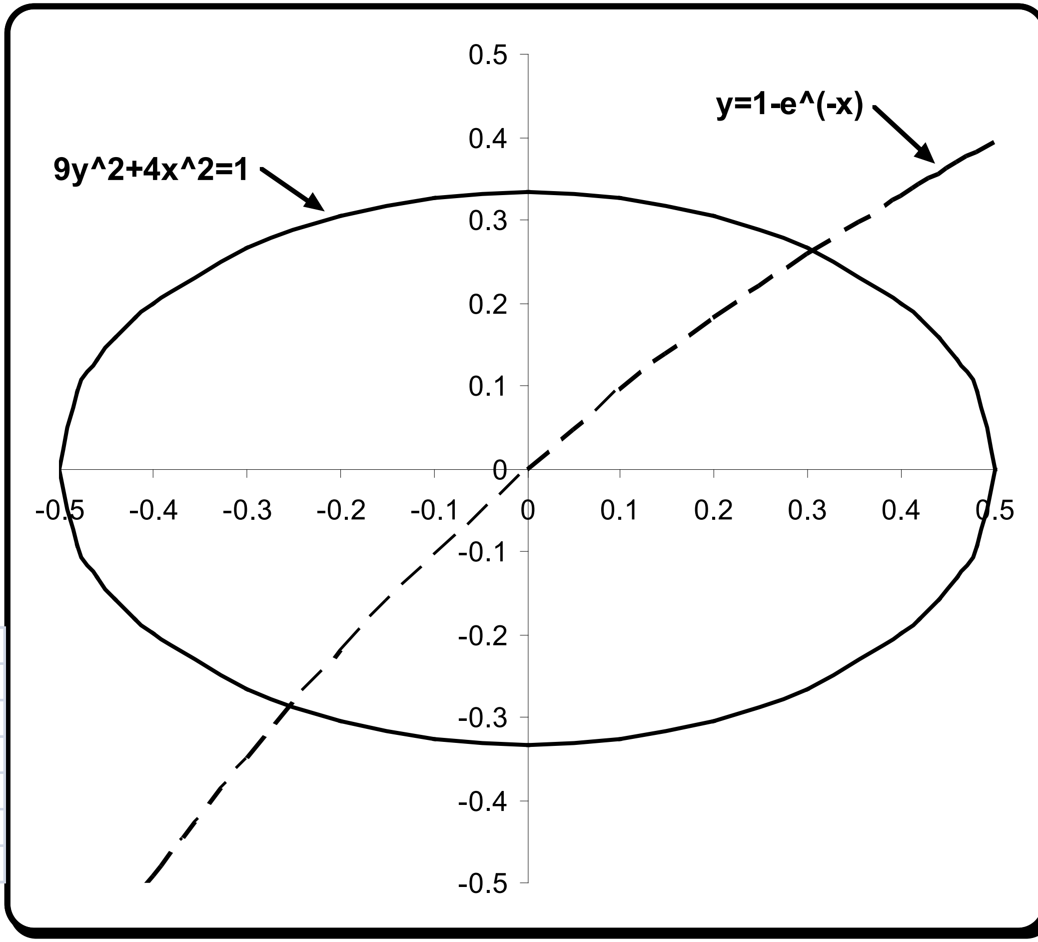
Ecuaciones No Lineales Simultáneas			
x1 =	**2,696285**		
x2 =	**3,365504**		
f(x1, x2) =	**0,000294**		
f(x1, x2) =	**-0,00062**		
y =	**4,66E-07**		

Solución de sistemas de ecuaciones usando Solver$_4$

- Ejemplo 3 (no lineal):

$$y = 1 - e^{-x}$$
$$9y^2 + 4x^2 = 1$$

x	0.30671241		x	-0.25279654
y	0.26329549		y	-0.28767525
f(x) = 0	-0.00084231		f(x) = 0	-5.3977E-05
g(x) = 0	0.00021064		g(x) = 0	0.00043784
	7.5386E-07			1.9462E-07



Sistemas de ecuaciones lineales

Matrices

- A *matrix* consists of a rectangular array of elements represented by a single symbol (example: [*A*]).
- An individual entry of a matrix is an *element* (example: a_{23})

$$
[A] = \begin{array}{cc}
\begin{array}{ccccc} \phantom{a_{11}} & \phantom{a_{12}} & \text{Column 3} & & \phantom{a_{1n}} \end{array} & \\
\begin{bmatrix}
a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\
a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\
\cdot & \cdot & \cdot & & \cdot \\
\cdot & \cdot & \cdot & & \cdot \\
\cdot & \cdot & \cdot & & \cdot \\
a_{m1} & a_{m2} & a_{m3} & \cdots & a_{mn}
\end{bmatrix} &
\begin{array}{l} \\ \text{Row 2} \\ \\ \\ \\ \\ \end{array}
\end{array}
$$

Matrices especiales

- Matrices where *m=n* are called *square matrices*.
- There are a number of special forms of square matrices:

Symmetric

$$[A]=\begin{bmatrix} 5 & 1 & 2 \\ 1 & 3 & 7 \\ 2 & 7 & 8 \end{bmatrix}$$

Diagonal

$$[A]=\begin{bmatrix} a_{11} & & \\ & a_{22} & \\ & & a_{33} \end{bmatrix}$$

Identity

$$[A]=\begin{bmatrix} 1 & & \\ & 1 & \\ & & 1 \end{bmatrix}$$

Upper Triangular

$$[A]=\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ & a_{22} & a_{23} \\ & & a_{33} \end{bmatrix}$$

Lower Triangular

$$[A]=\begin{bmatrix} a_{11} & & \\ a_{21} & a_{22} & \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

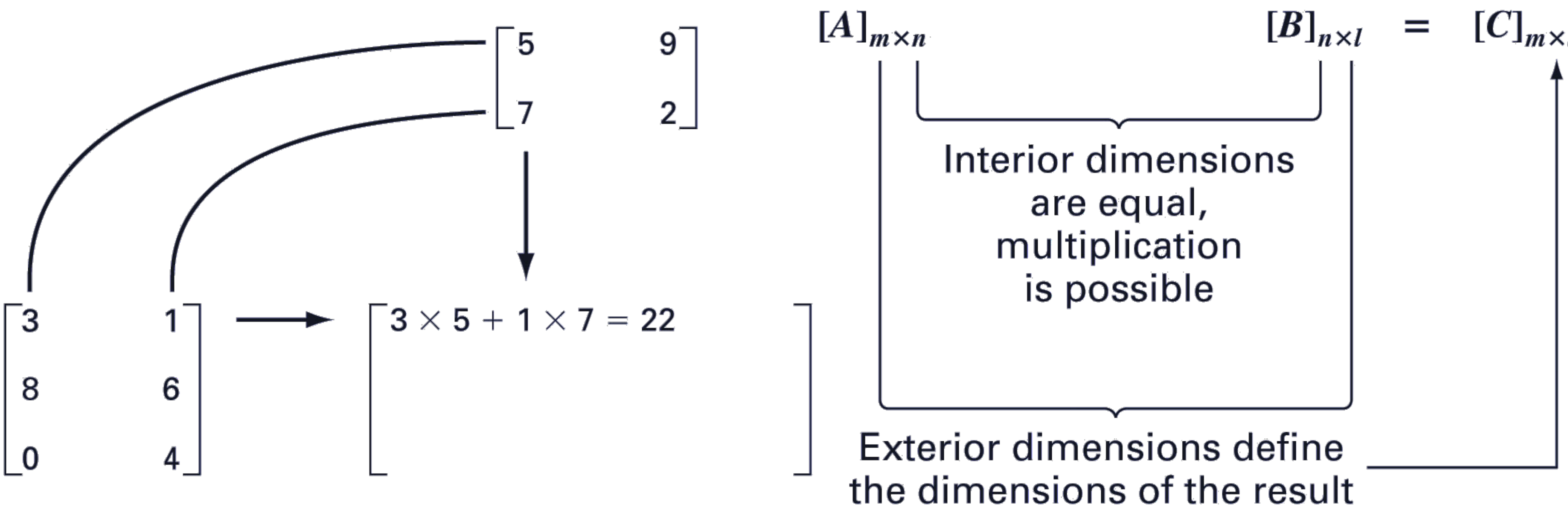
Banded

$$[A]=\begin{bmatrix} a_{11} & a_{12} & & \\ a_{21} & a_{22} & a_{23} & \\ & a_{32} & a_{33} & a_{34} \\ & & a_{43} & a_{44} \end{bmatrix}$$

Multiplicación matricial

- The elements in the matrix [C] that results from multiplying matrices [A] and [B] are calculated using:

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$$



Representación de Algebra Lineal

- Matrices provide a concise notation for representing and solving simultaneous linear equations:

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 &= b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 &= b_2 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 &= b_3 \end{aligned}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix} = \begin{Bmatrix} b_1 \\ b_2 \\ b_3 \end{Bmatrix}$$

$$[A]\{x\} = \{b\}$$

Matriz inversa

- Recall that if a matrix $[A]$ is square, there is another matrix $[A]^{-1}$, called the inverse of $[A]$, for which $[A][A]^{-1}=[A]^{-1}[A]=[I]$
- The inverse can be computed in a column by column fashion by generating solutions with unit vectors as the right-hand-side constants:

$$[A]\{x_1\}=\begin{Bmatrix}1\\0\\0\end{Bmatrix} \quad [A]\{x_2\}=\begin{Bmatrix}0\\1\\0\end{Bmatrix} \quad [A]\{x_3\}=\begin{Bmatrix}0\\0\\1\end{Bmatrix}$$

$$[A]^{-1}=\begin{bmatrix}x_1 & x_2 & x_3\end{bmatrix}$$

Matriz inversa y sistemas estímulo - respuesta

- Recall that *LU* factorization can be used to efficiently evaluate a system for multiple right-hand-side vectors - thus, it is ideal for evaluating the multiple unit vectors needed to compute the inverse.
- Many systems can be modeled as a linear combination of equations, and thus written as a matrix equation:

$$[\text{Interactions}]\{\text{response}\} = \{\text{stimuli}\}$$

- The system response can thus be found using the matrix inverse.

Normas vectoriales y matriciales

- A *norm* is a real-valued function that provides a measure of the size or “length” of multi-component mathematical entities such as vectors and matrices.
- Vector norms and matrix norms may be computed differently.

Normas vectoriales

- For a vector $\{X\}$ of size n, the p-norm is:

$$\|X\|_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p}$$

- Important examples of vector p-norms include:

$$p = 1 : \text{sum of the absolute values} \qquad \|X\|_1 = \sum_{i=1}^{n} |x_i|$$

$$p = 2 : \text{Euclidian norm (length)} \qquad \|X\|_2 = \|X\|_e = \sqrt{\sum_{i=1}^{n} x_i^2}$$

$$p = \infty : \text{maximum} - \text{magnitude} \qquad \|X\|_\infty = \max_{1 \le i \le n} |x_i|$$

Normas matriciales

- Common matrix norms for a matrix [*A*] include:

column - sum norm $\|A\|_1 = \max\limits_{1\le j\le n} \sum_{i=1}^{n} |a_{ij}|$

Frobenius norm $\|A\|_f = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}^2}$

row - sum norm $\|A\|_\infty = \max\limits_{1\le i\le n} \sum_{j=1}^{n} |a_{ij}|$

spectral norm (2 norm) $\|A\|_2 = (\mu_{max})^{1/2}$

- Note - μ_{max} is the largest eigenvalue of $[A]^T[A]$.

Número de condición de una matriz

- The *matrix condition number* Cond[A] is obtained by calculating Cond[A]=$\|A\|\cdot\|A^{-1}\|$
- In can be shown that: $$\frac{\|\Delta X\|}{\|X\|} \leq \mathrm{Cond}[A]\frac{\|\Delta A\|}{\|A\|}$$
- The relative error of the norm of the computed solution can be as large as the relative error of the norm of the coefficients of [A] multiplied by the condition number.
- If the coefficients of [A] are known to t digit precision, the solution [X] may be valid to only t-$\log_{10}$(Cond[A]) digits.

Método iterativo: Gauss - Seidel

- The *Gauss-Seidel method* is the most commonly used iterative method for solving linear algebraic equations [*A*]{*x*}={*b*}.
- The method solves each equation in a system for a particular variable, and then uses that value in later equations to solve later variables. For a 3x3 system with nonzero elements along the diagonal, for example, the j^{th} iteration values are found from the j-1^{th} iteration using:

$$x_1^j = \frac{b_1 - a_{12}x_2^{j-1} - a_{13}x_3^{j-1}}{a_{11}}$$

$$x_2^j = \frac{b_2 - a_{21}x_1^{j} - a_{23}x_3^{j-1}}{a_{22}}$$

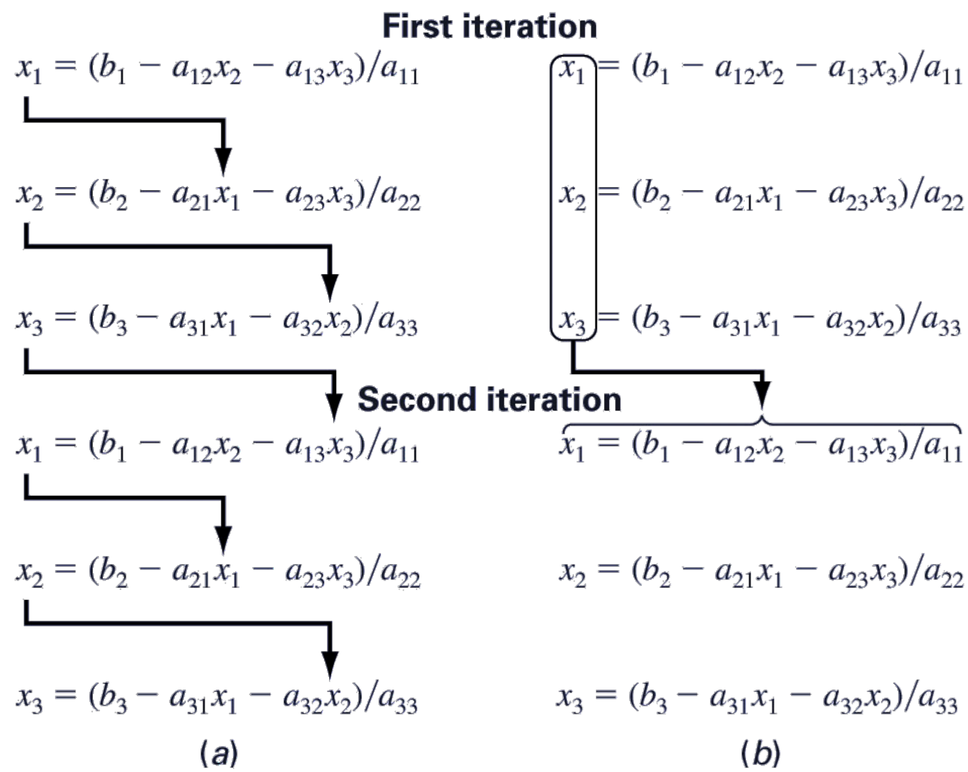
$$x_3^j = \frac{b_3 - a_{31}x_1^{j} - a_{32}x_2^{j}}{a_{33}}$$

Iteración de Jacobi

- The *Jacobi iteration* is similar to the Gauss-Seidel method, except the j-1th information is used to update all variables in the jth iteration:

  a) Gauss-Seidel

  b) Jacobi



Convergencia

- The convergence of an iterative method can be calculated by determining the relative percent change of each element in {*x*}. For example, for the *i*th element in the *j*th iteration,

$$\varepsilon_{a,i} = \left| \frac{x_i^j - x_i^{j-1}}{x_i^j} \right| \times 100\%$$

- The method is ended when all elements have converged to a set tolerance.

Dominancia diagonal

- The Gauss-Seidel method may diverge, but if the system is *diagonally dominant*, it will definitely converge.
- Diagonal dominance means:

$$\left|a_{ii}\right| > \sum_{\substack{j=1 \\ j \neq i}}^{n} \left|a_{ij}\right|$$

Relajación

- To enhance convergence, an iterative program can introduce *relaxation* where the value at a particular iteration is made up of a combination of the old value and the newly calculated value:

$$x_i^{\text{new}} = \lambda x_i^{\text{new}} + (1-\lambda) x_i^{\text{old}}$$

  where λ is a weighting factor that is assigned a value between 0 and 2.
  - $0<\lambda<1$: underrelaxation
  - $\lambda=1$: no relaxation
  - $1<\lambda\leq 2$: overrelaxation

Series

Series de números

- Las series de números son importantes en las matemáticas porque permiten, por ejemplo, la evaluación de funciones trascendentales, integrales o ecuaciones diferenciales.
- Habitualmente la suma de una serie de números se usa como una aproximación a una función que no se puede evaluar directamente.
- La aproximación es más precisa si se añaden más términos a la suma. Si la suma alcanza un valor finito la serie es *convergente*, caso contrario es divergente.

Series de números

- Series y métodos iterativos: cualquier serie $\sum x_n$ se puede convertir en un método iterativo considerando la secuencia de sumas parciales s_n.

$$s_n = \sum_{k=0}^{n} x_k = x_0 + x_1 + x_2 + \cdots + x_n$$

$$s_n = s_{n-1} + x_n$$

- Evaluación de funciones y expansión de Taylor:
  - Del Cálculo se sabe que cualquier función que tiene $n+1$ derivadas en un punto a tiene una expansion polinómica n^{th} de Taylor Polynomial centrada en a y un error.

Series de números – Constantes Array

- Ejemplo: Series.xlsx
- Se puede usar en Excel **constantes Array** para crear fórmulas de series.
  - Una constante array es un array de valores separados por comas y encerrados entre llaves, usado como argumento de una función. Ejemplo de array literal: {40,21,300,10}
  - Se puede usar una constante array para hacer la evaluación de una fórmula de serie más compacta y precisa. Por ejemplo para evaluar:

$$e = 1 + \sum_{k=1}^{\infty} \frac{1}{k!}$$ = 1 +SUMA(1 /FACT({1,2,3,4,5,6,7,8,9,10}))

Series de números – Función FILA

- Ejemplo: Series.xlsx
- Se puede usar la función Excel FILA para generar series de números.
  - Si se introduce en una celda =FILA(1:100), se selecciona y se pulsa la tecla F9 se obtiene:

    ={1;2;3;4;5;6;7;8;9;10;11;12;13;14;15;16;17;18;19;20;21;22;23;24;25;26;27;28;29;30;31;32;33;34;35;36;37;38;39;40;41;42;43;44;45;46;47;48;49;50;51;52;53;54;55;56;57;58;59;60;61;62;63;64;65;66;67;68;69;70;71;72;73;74;75;76;77;78;79;80;81;82;83;84;85;86;87;88;89;90;91;92;93;94;95;96;97;98;99;100}
  - Usando este método se puede evaluar fórmulas de series. Por ejemplo para evaluar *e* se puede usar:

    {=1+SUMA(1/FACT(FILA(1:100)))}

Series de números – Función INDIRECTO

- Ejemplo: Series.xlsx
- Se puede usar la función INDIRECTO para crear una referencia especificada por una cadena de texto.
  - Si se introduce en una celda =INDIRECTO(“A1”) crea una referencia a la celda A1 y devuelve el valor contenida en esa celda.
  - Se puede usar este método junto con FILA para evaluar fórmulas de series. Por ejemplo para calcular e:

    {=1+SUMA(1/FACT(FILA(INDIRECTO("1:20"))))}   o

    {=1+SUMA(1/FACT(FILA(INDIRECTO("1:"&A1))))} donde el valor en A1 especifica el número de términos a evaluar.

Series de Taylor

- Las series de Taylor se usan para la evaluación de funciones por métodos numéricos:

$$f(x) = \sum_{k=0}^{n} \frac{f^{(k)}(a)}{k!}(x-a)^k + error$$

$$f(x) = f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!} + (x-a)^2 + \cdots + \frac{f^{(n)}(a)}{n!}(x-a)^n + R_n(x)$$

$$R_n(x) = \frac{f^{(n+1)}(c)}{(n+1)!}(x-a)^{n+1}$$

  - El valor $f^{(k)}(a)$ es la k^{th} derivada evaluada en a. La función $R_n(x)$ representa el error donde c es un valor entre x y a.

Series de Taylor

- Versión de los polinomios de Taylor apropiada para computación:
  - La serie de Taylor para evaluar una función f en el punto $x+h$, dados el valor de la función y sus derivadas en el punto x es:

$$f(x+h)=\sum_{k=0}^{n}\frac{f^{(k)}(x)}{k!}h^k = f(x)+\frac{f'(x)}{1!}h+\frac{f''(x)}{2!}h^2+\cdots+\frac{f^{(k)}(x)}{k!}h^k+\frac{f^{(k+1)}(x+c)}{(k+1)!}h^{k+1}$$

  - Siendo las k^{th} derivadas

$$\left.\frac{d^k f(x+h)}{dh^k}\right|_{h=0} = \left.f^{(k)}(x+h)\right|_{h=0} = f^{(k)}(x)$$

- Ejemplo: Series.xlsx

Series de Taylor - Ejemplos

– Las derivadas de mayor orden de algunas funciones se repiten según un patrón.

$$\frac{1}{1-x}=\sum_{k=0}^{n}x^{k}=1+x+x^{2}+x^{3}+\cdots+x^{n}+\frac{1}{(1-c)^{n+2}}x^{n+1}$$

$$e^{x}=\sum_{k=0}^{\infty}\frac{x^{k}}{k!}=1+\frac{x}{1!}+\frac{x^{2}}{2!}+\frac{x^{3}}{3!}+\cdots+\frac{x^{n}}{n!}+\frac{e^{c}}{(n+1)!}x^{n+1}$$

$$\sin x=\sum_{k=0}^{n}\frac{(-1)^{k}x^{2k+1}}{(2k+1)!}=x-\frac{x^{3}}{3!}+\frac{x^{5}}{5!}\pm\cdots\pm\frac{x^{2n+1}}{(2n+1)!}+\frac{\pm\sin c}{(2n+2)!}x^{2n+2}$$

$$\cos x=\sum_{k=0}^{n}\frac{(-1)^{k}x^{2k}}{(2k)!}=1-\frac{x^{2}}{2!}+\frac{x^{4}}{4!}\pm\cdots\pm\frac{x^{2n}}{(2n)!}+\frac{\pm\cos c}{(2n+1)!}x^{2n+2}$$

Series de Taylor - Ejemplos

– Serie de Taylor para la función $f(x)$ = $\arctan(x)$.

$$\frac{1}{1-x}=\sum_{k=0}^{\infty}x^k=1+x+x^2+x^3+\cdots$$

$$\frac{1}{1+x}=\sum_{k=0}^{\infty}(-x)^k=1-x+x^2-x^3\pm\cdots$$

$$\frac{1}{1+x^2}=\sum_{k=0}^{\infty}(-1)^k x^{2k}=1-x^2+x^4-x^6\pm\cdots$$

$$\arctan x=\sum_{k=0}^{\infty}\frac{(-1)^k x^{2k+1}}{2k+1}=x-\frac{x^3}{3}+\frac{x^5}{5}-\frac{x^7}{7}\pm\cdots$$

– Serie de Taylor para la función $g(x)=x^3\cosh\left(\sqrt{x}\right)$

$$\cosh x=\sum_{k=0}^{\infty}\frac{x^{2k}}{(2k)!}=1+\frac{x^2}{2!}+\frac{x^4}{4!}+\frac{x^6}{6!}+\cdots \qquad \cosh\left(\sqrt{x}\right)=\sum_{k=0}^{\infty}\frac{x^{k}}{(2k)!}=1+\frac{x}{2!}+\frac{x^2}{4!}+\frac{x^3}{6!}+\cdots$$

$$x^3\cosh\left(\sqrt{x}\right)=\sum_{k=0}^{\infty}\frac{x^{k+3}}{(2k)!}=x^3+\frac{x^4}{2!}+\frac{x^5}{4!}+\frac{x^6}{6!}+\cdots$$

Series de Taylor - Observaciones

- Versión de la serie de Taylor para computación:
  - La serie de Taylor para evaluar una función f en el punto x+h, dados el valor de la función y sus derivadas en el punto x es:

$$f(x+h)=\sum_{k=0}^{n}\frac{f^{(k)}(x)}{k!}h^{k}=f(x)+\frac{f'(x)}{1!}h+\frac{f''(x)}{2!}h^{2}+\cdots+\frac{f^{(k)}(x)}{k!}h^{k}+\frac{f^{(k+1)}(x+c)}{(k+1)!}h^{k+1}$$

  - Siendo las k^{th} derivadas

$$\left.\frac{d^{k}f(x+h)}{dh^{k}}\right|_{h=0}=\left.f^{(k)}(x+h)\right|_{h=0}=f^{(k)}(x)$$

  - La precisión de la aproximación aumenta si: se incrementa el número de términos y h se hace pequeño.
- Ejemplo: Series.xlsx

Interpolación

Interpolación

- Dada una tabla de puntos x, y es frecuente determinar el valor de y para una valor de x que se encuentra entre los valores tabulados.
- El proceso de interpolación involucra la determinación de una función matemática que pasa por los puntos dados.
- Existe gran cantidad de métodos para resolver la interpolación. Los métodos numéricos generalmente tratan de usar polinomios como función interpolante y resolver un número de ecuaciones y coeficientes igual al número de puntos datos.
- Ejemplos: Interpolacion.xlsx

Interpolación mediante búsqueda

- Excel dispone de las funciones:
  - BUSCARV para búsquedas verticales en una tabla
  - BUSCARH para búsquedas horizontales en una tabla
  - BUSCAR para búsquedas en general
- También se tiene la opción de construir una fórmula de búsqueda con las funciones básicas:
  - COINCIDIR para buscar un elemento especificado en un intervalo de celdas y obtener la posición relativa de ese elemento en el rango
  - INDICE que devuelve un valor o la referencia a un valor de una tabla o rango
- Se puede aplicar a tablas de dos variables

Interpolación lineal mediante búsqueda

- Se puede realizar una interpolación lineal sobre los resultados de una búsqueda con COINCIDIR e INDEX:

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

  - Donde posicion =COINCIDIR(Valor buscado,rango x,1)

    x_0 =INDICE(rango x,posicion)

    x_1 =INDICE(rango x,posicion + 1)

    y_0 =INDICE(rango y,posicion)

    y_1 =INDICE(rango y,posicion + 1)

- Otra alternativa es usar la función TENDENCIA

Interpolación lineal mediante VBA

- También se puede programar una función propia para la interpolación lineal:

```
Function InterpolateL(lookup_value, known_x´s, known_y´s)
    Dim pointer As Integer
    Dim X0 As Double
    Dim Y0 As Double
    Dim X1 As Double
    Dim Y1 As Double
    ‘Para evitar extrapolacion
    If lookup_value < Application.Min(known_x´s) Or lookup_value >
    Application.Max(known_x´s) Then
        InterpolateL = CVErr(xlErrRef): Exit Function
    End If
    pointer = Application.Match(lookup_value, known_x´s, 1)
    X0 = known_x´s(pointer)
    Y0 = known_y´s(pointer)
    X1 = known_x´s(pointer + 1)
    Y1 = known_y´s(pointer + 1)
    InterpolateL = Y0 + (lookup_value - X0) * (Y1 - Y0) / (X1 - X0)
End Function
```

Interpolación cúbica

- Los valores de una tabla pueden ser tales que la interpolación cúbica es más adecuada que la lineal.
- La interpolación cúbica usa cuatro valores adyacentes de la tabla, x_0 , x_1 , x_2 , x_3 para obtener los coeficientes de la ecuación cúbica

$$y = a + bx + cx^2 + dx^3$$

para usarse como función interpolante entre x_1 y x_2.
- Para realizar la interpolación se utiliza un polinomio de Lagrange de orden 4 programado en VBA.

$$y_x = \frac{(x-x_2)(x-x_3)(x-x_4)}{(x_1-x_2)(x_1-x_3)(x_1-x_4)}y_1 + \frac{(x-x_1)(x-x_3)(x-x_4)}{(x_2-x_1)(x_2-x_3)(x_2-x_4)}y_2$$

$$+ \frac{(x-x_1)(x-x_2)(x-x_4)}{(x_3-x_1)(x_3-x_2)(x_3-x_4)}y_3 + \frac{(x-x_1)(x-x_2)(x-x_3)}{(x_4-x_1)(x_4-x_2)(x_4-x_3)}y_4$$

Interpolación cúbica mediante VBA

- Requiere pasar los arrays de valores x (en orden ascendente) e y

```
Function InterpolateC(lookup_value, known_x´s, known_y´s)
'    Performs cubic interpolation, the known_x´s must be in ascending order.
Dim row As Integer
Dim i As Integer, j As Integer
Dim Q As Double, Y As Double
row = Application.Match(lookup_value, known_x´s, 1)
    If row < 2 Then row = 2
    If row > known_x´s.Count - 2 Then row = known_x´s.Count - 2
For i = row - 1 To row + 2
    Q = 1
For j = row - 1 To row + 2
    If i <> j Then Q = Q * (lookup_value - known_x´s(j)) / (known_x´s(i) -
known_x´s(j))
Next j
    Y = Y + Q * known_y´s(i)
Next i
InterpolateC = Y
End Function
```

Interpolación cúbica sobre tablas de dos parámetros

- Se puede aplicar la interpolación cúbica a tablas de dos parámetros
- En ese caso hay que seleccionar la matriz de valores para realizar la interpolación.
- Ejemplos: Interpolacion2.xlsx

Interpolación numérica

- Existe varios métodos para realizar una interpolación mediante polinomios.
- Algunos de los métodos son:
  - Polinomios de diferencias divididas de Newton
  - Interpolación cuadrática
  - Polinomios de diferencias divididas de Newton de orden n
  - Interpolación de Lagrange
- Ejemplos: Interpolacionnum.xlsx

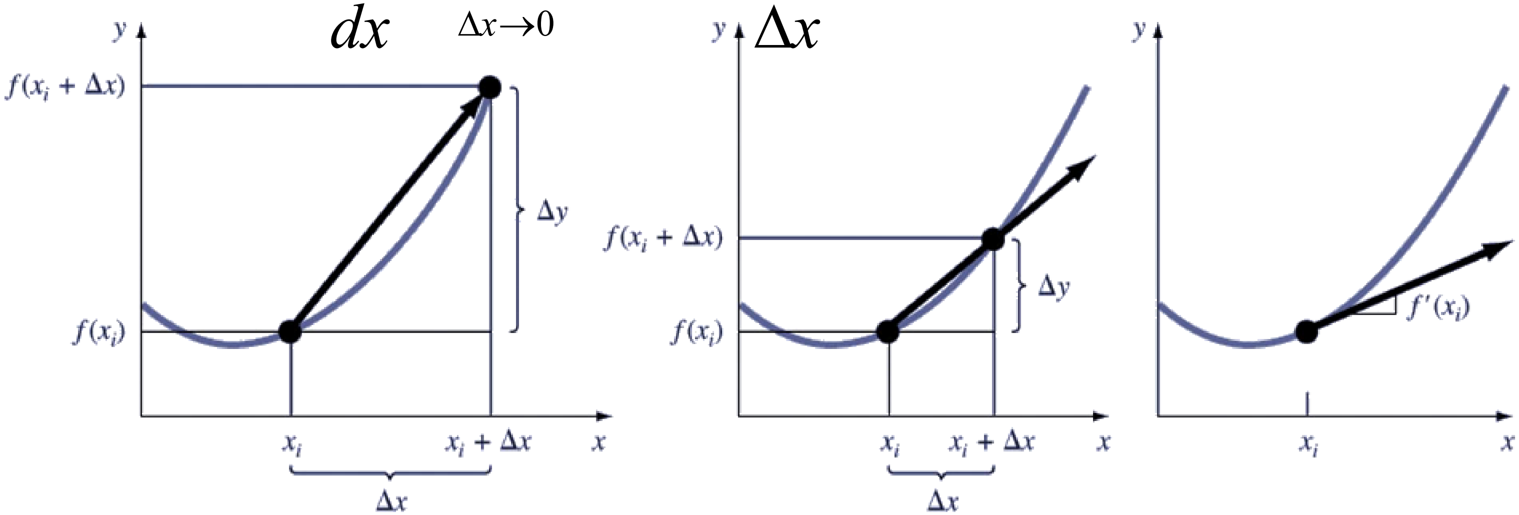
Evaluación de derivadas

Diferenciación de funciones continuas

- La definición matemática de la derivada empieza con una aproximación de la diferencia finita:

$$\frac{\Delta y}{\Delta x} = \frac{f(x_i + \Delta x) - f(x_i)}{\Delta x}$$

cuando Δx tiende a cero, la diferencia se convierte en derivada: $\dfrac{dy}{dx} = \lim\limits_{\Delta x \to 0} \dfrac{f(x_i + \Delta x) - f(x_i)}{\Delta x}$



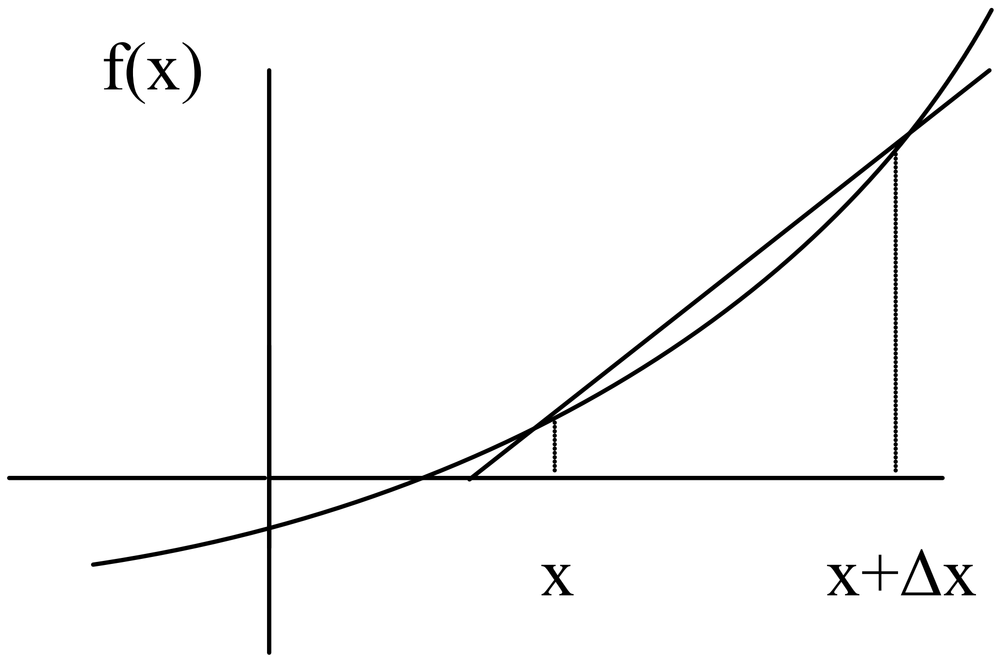
Derivadas a partir de datos

- Cuando se tiene un conjunto de datos de la forma (x_i, y_i) se puede aproximar la derivada en un punto i mediante varias diferencias:
  - *Diferencia finita en adelanto (forward finite-difference)*
  - *Diferencia finita en atraso (backward finite-difference)*
  - *Diferencia finita centrada (centered finite-difference)*
- La expansión de la serie de Taylor se puede usar para generar fórmulas de gran precisión para las derivadas aplicando algebra lineal para combinar la expansión alrededor de varios puntos.

Aproximación por diferencia en adelante

$$f'(x) = \lim_{\Delta x \to 0} \frac{f(x+\Delta x) - f(x)}{\Delta x}$$

Para un finito $'\Delta x'$ $\quad f'(x) \approx \dfrac{f(x+\Delta x) - f(x)}{\Delta x}$



Graphical Representation of forward difference approximation of first derivative.

Aproximación por diferencia en atraso

- Sabemos que $f'(x) = \lim_{\Delta x \to 0} \frac{f(x+\Delta x) - f(x)}{\Delta x}$
- Para un finite 'Δx', $f'(x) \approx \frac{f(x+\Delta x) - f(x)}{\Delta x}$
- Si 'Δx' se toma como número negativo,

$$f'(x) \approx \frac{f(x-\Delta x) - f(x)}{-\Delta x} = \frac{f(x) - f(x-\Delta x)}{\Delta x}$$

Aproximación por diferencia en atraso

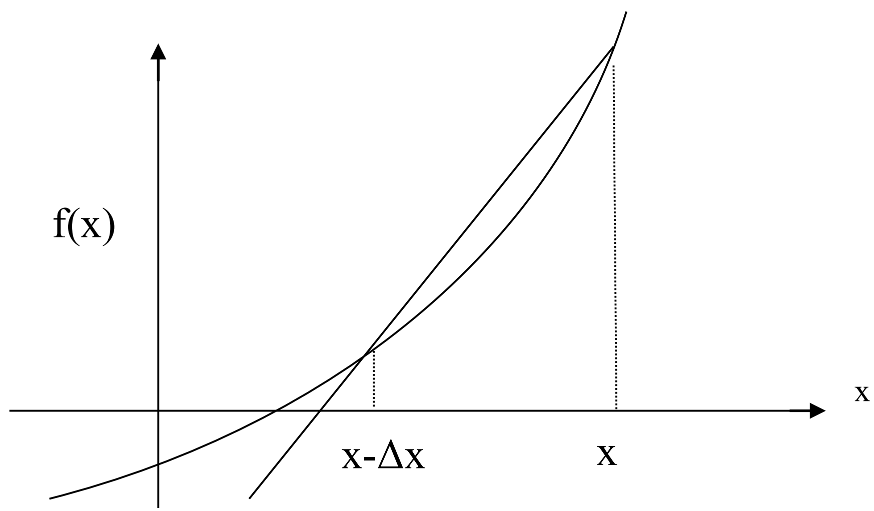
- This is a backward difference approximation as you are taking a point backward from x. To find the value of $f'(x)$ at $x = x_i$, we may choose another point 'Δx' behind as $x = x_{i-1}$. This gives

$$f'(x_i) \approx \frac{f(x_i) - f(x_{i-1})}{\Delta x}$$

$$= \frac{f(x_i) - f(x_{i-1})}{x_i - x_{i-1}}$$

where

$$\Delta x = x_i - x_{i-1}$$



Graphical Representation of backward difference approximation of first derivative.

Obtención de la adad a partir de las series de Taylor

- Taylor's theorem says that if you know the value of a function f at a point x_i and all its derivatives at that point, provided the derivatives are continuous between x_i and x_{i+1}, then

$$f(x_{i+1}) = f(x_i) + f'(x_i)(x_{i+1} - x_i) + \frac{f''(x_i)}{2!}(x_{i+1} - x_i)^2 + \ldots$$

Substituting for convenience $\Delta x = x_{i+1} - x_i$

$$f(x_{i+1}) = f(x_i) + f'(x_i)\Delta x + \frac{f''(x_i)}{2!}(\Delta x)^2 + \ldots$$

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_i)}{\Delta x} - \frac{f''(x_i)}{2!}(\Delta x) + \ldots$$

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_i)}{\Delta x} + O(\Delta x)$$

Obtención de la adad a partir de las series de Taylor

- The $O(\Delta x)$ term shows that the error in the approximation is of the order of Δx. It is easy to derive from Taylor series the formula for backward divided difference approximation of the first derivative.
- As shown above, both forward and backward divided difference approximation of the first derivative are accurate on the order of $O(\Delta x)$.
- Can we get better approximations? Yes, another method is called the **Central difference approximation** of the first derivative.

Obtención de la adc a partir de las series de Taylor

- From Taylor series

$$f(x_{i+1}) = f(x_i) + f'(x_i)\Delta x + \frac{f''(x_i)}{2!}(\Delta x)^2 + \frac{f'''(x_i)}{3!}(\Delta x)^3 + \ldots \qquad (1)$$

$$f(x_{i-1}) = f(x_i) - f'(x_i)\Delta x + \frac{f''(x_i)}{2!}(\Delta x)^2 - \frac{f'''(x_i)}{3!}(\Delta x)^3 + \ldots \qquad (2)$$

Subtracting equation (2) from equation (1)

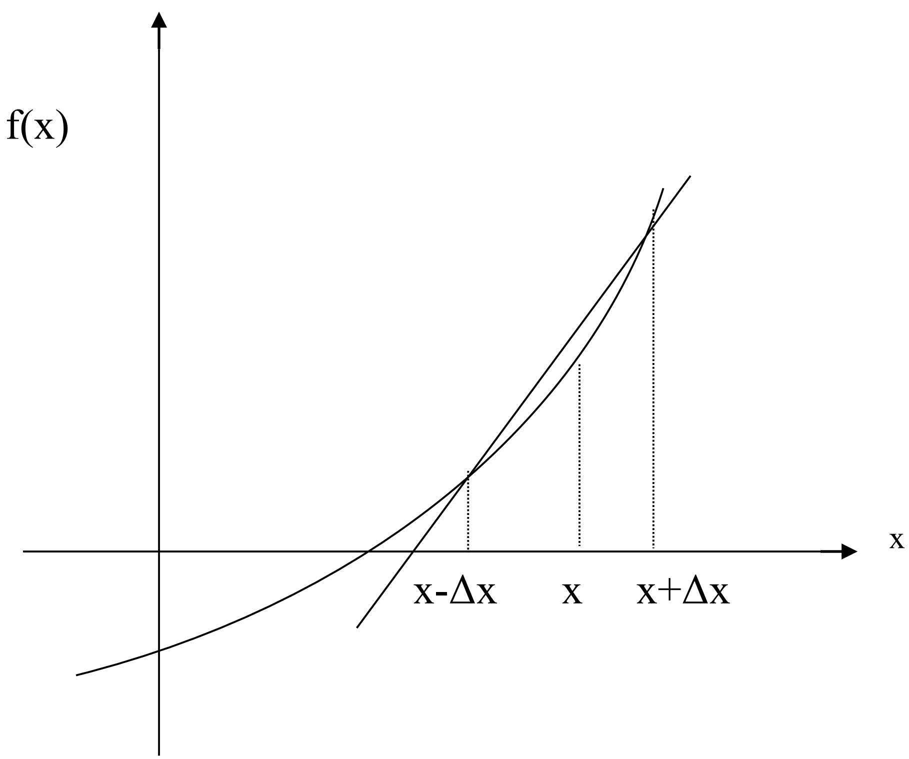
$$f(x_{i+1}) - f(x_{i-1}) = f'(x_i)(2\Delta x) + \frac{2f'''(x_i)}{3!}(\Delta x)^3 + \ldots$$

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_{i-1})}{2\Delta x} - \frac{f'''(x_i)}{3!}(\Delta x)^2 + \ldots$$

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_{i-1})}{2\Delta x} + O(\Delta x)^2$$

Obtención de la adc a partir de las series de Taylor

- Hence showing that we have obtained a more accurate formula as the error is of the order of $O(\Delta x)^2$



Graphical Representation of central difference approximation of first derivative

Fórmula de 5 puntos

- La fórmula de 5 puntos corresponde a la diferencia finita centrada con error de orden cuartico

$$f(x_i)=\frac{1}{12\Delta x}\left(f(x_{i-2})-8f(x_{i-1})+8f(x_{i+1})-f(x_{i+2})\right)$$

- Ejemplo: derivadas.xlsx

Fórmulas de diferencia finita en adelanto

First Derivative	Error
$f'(x_i) = \dfrac{f(x_{i+1}) - f(x_i)}{h}$	$O(h)$
$f'(x_i) = \dfrac{-f(x_{i+2}) + 4f(x_{i+1}) - 3f(x_i)}{2h}$	$O(h^2)$
Second Derivative	
$f''(x_i) = \dfrac{f(x_{i+2}) - 2f(x_{i+1}) + f(x_i)}{h^2}$	$O(h)$
$f''(x_i) = \dfrac{-f(x_{i+3}) + 4f(x_{i+2}) - 5f(x_{i+1}) + 2f(x_i)}{h^2}$	$O(h^2)$
Third Derivative	
$f'''(x_i) = \dfrac{f(x_{i+3}) - 3f(x_{i+2}) + 3f(x_{i+1}) - f(x_i)}{h^3}$	$O(h)$
$f'''(x_i) = \dfrac{-3f(x_{i+4}) + 14f(x_{i+3}) - 24f(x_{i+2}) + 18f(x_{i+1}) - 5f(x_i)}{2h^3}$	$O(h^2)$
Fourth Derivative	
$f''''(x_i) = \dfrac{f(x_{i+4}) - 4f(x_{i+3}) + 6f(x_{i+2}) - 4f(x_{i+1}) + f(x_i)}{h^4}$	$O(h)$
$f''''(x_i) = \dfrac{-2f(x_{i+5}) + 11f(x_{i+4}) - 24f(x_{i+3}) + 26f(x_{i+2}) - 14f(x_{i+1}) + 3f(x_i)}{h^4}$	$O(h^2)$

Fórmulas de diferencia finita en atraso

First Derivative	Error
$f'(x_i) = \dfrac{f(x_i) - f(x_{i-1})}{h}$	$O(h)$
$f'(x_i) = \dfrac{3f(x_i) - 4f(x_{i-1}) + f(x_{i-2})}{2h}$	$O(h^2)$
Second Derivative	
$f''(x_i) = \dfrac{f(x_i) - 2f(x_{i-1}) + f(x_{i-2})}{h^2}$	$O(h)$
$f''(x_i) = \dfrac{2f(x_i) - 5f(x_{i-1}) + 4f(x_{i-2}) - f(x_{i-3})}{h^2}$	$O(h^2)$
Third Derivative	
$f'''(x_i) = \dfrac{f(x_i) - 3f(x_{i-1}) + 3f(x_{i-2}) - f(x_{i-3})}{h^3}$	$O(h)$
$f'''(x_i) = \dfrac{5f(x_i) - 18f(x_{i-1}) + 24f(x_{i-2}) - 14f(x_{i-3}) + 3f(x_{i-4})}{2h^3}$	$O(h^2)$
Fourth Derivative	
$f''''(x_i) = \dfrac{f(x_i) - 4f(x_{i-1}) + 6f(x_{i-2}) - 4f(x_{i-3}) + f(x_{i-4})}{h^4}$	$O(h)$
$f''''(x_i) = \dfrac{3f(x_i) - 14f(x_{i-1}) + 26f(x_{i-2}) - 24f(x_{i-3}) + 11f(x_{i-4}) - 2f(x_{i-5})}{h^4}$	$O(h^2)$

Fórmulas de diferencia finita centrada

	Error
First Derivative	
$f'(x_i) = \dfrac{f(x_{i+1}) - f(x_{i-1})}{2h}$	$O(h^2)$
$f'(x_i) = \dfrac{-f(x_{i+2}) + 8f(x_{i+1}) - 8f(x_{i-1}) + f(x_{i-2})}{12h}$	$O(h^4)$
Second Derivative	
$f''(x_i) = \dfrac{f(x_{i+1}) - 2f(x_i) + f(x_{i-1})}{h^2}$	$O(h^2)$
$f''(x_i) = \dfrac{-f(x_{i+2}) + 16f(x_{i+1}) - 30f(x_i) + 16f(x_{i-1}) - f(x_{i-2})}{12h^2}$	$O(h^4)$
Third Derivative	
$f'''(x_i) = \dfrac{f(x_{i+2}) - 2f(x_{i+1}) + 2f(x_{i-1}) - f(x_{i-2})}{2h^3}$	$O(h^2)$
$f'''(x_i) = \dfrac{-f(x_{i+3}) + 8f(x_{i+2}) - 13f(x_{i+1}) + 13f(x_{i-1}) - 8f(x_{i-2}) + f(x_{i-3})}{8h^3}$	$O(h^4)$
Fourth Derivative	
$f''''(x_i) = \dfrac{f(x_{i+2}) - 4f(x_{i+1}) + 6f(x_i) - 4f(x_{i-1}) + f(x_{i-2})}{h^4}$	$O(h^2)$
$f''''(x_i) = \dfrac{-f(x_{i+3}) + 12f(x_{i+2}) + 39f(x_{i+1}) + 56f(x_i) - 39f(x_{i-1}) + 12f(x_{i-2}) + f(x_{i-3})}{6h^4}$	$O(h^4)$

Evaluación de Integrales

Evaluación de Integrales

- Muchos problemas de ingeniería requieren la evaluación de integrales. Para ello se usan métodos analíticos y numéricos.
- Suponiendo que se da una función continua y=f(x) definida en el intervalo a $\leq$ x $\leq$ b, entonces la integral

$$I = \int_a^b f(x)\,dx = \int_a^b y\,dx$$

  se interpreta como el área bajo la curva.
- La integración numérica trata de aproximar el resultado de una integral aplicando las fórmulas de integración Newton-Cotes, cuadratura gaussiana o Monte Carlo.
- Ejemplo: Integrales1.xls

Evaluación de Integrales – método trapezoidal

- Regla trapezoidal (datos espaciados uniformes):
  - Datos: n pares de puntos equiespaciados (x_1,y_1), (x_2,y_2),… (x_n,y_n) donde x_1 = a y x_2 = b
  - Los puntos definen n-2 intervalos rectangulares con un ancho igual a Δx
  - La altura de cada intervalo se expresa como: $\bar{y}_i = \dfrac{y_i + y_{i+1}}{2}$
  - Por tanto la integral se aproxima como:

$$I = \int_a^b y\, dx = \left(\frac{(y_1 + y_n)}{2} + \sum_{i=2}^{n-1} y_i \right) \Delta x$$

Evaluación de Integrales – método trapezoidal

- Ejemplo:
  - La presión media de una gas cuando la temperatura del gas varía en el tiempo se calcula por:

$$\overline{P} = 8.571x10^{-4} \int_{0}^{100} (300 + 12t)dt - 3.592$$

Para hallar la integral, se obtiene 21 puntos igualmente espaciados y se aplica la trapezoidal, de donde se obtiene que la presión es:

$$\overline{P} = (8.571x10^{-4})(9x10^{4}) - 3.592 = 73.547$$

Evaluación de Integrales – método trapezoidal

Integración numérica usando la regla trapezoidal con datos espaciados uniformemente

tiempo	Temperatura						
0	300						
5	360		360				
10	420		420				
15	480		480				
20	540		540				
25	600		600				
30	660		660				
35	720		720				
40	780		780				
45	840		840				
50	900		900				
55	960		960				
60	1020		1020				
65	1080		1080				
70	1140		1140				
75	1200		1200				
80	1260		1260				
85	1320		1320				
90	1380		1380				
95	1440		1440				
100	1500						
		Suma =	17100	**Integral =**	90000	**Presión =**	73.547

Evaluación de Integrales – método trapezoidal

- Regla trapezoidal (datos espaciados **no** uniformes):
  - Datos: n pares de puntos (x_1,y_1), (x_2,y_2),… (x_n,y_n) donde x_1 = a y x_2 = b
  - Estos puntos definen n-1 intervalos rectangulares con un ancho para el i-ésimo intervalo $\Delta x_i = x_{i+1} - x_i$
  - La altura de cada intervalo se expresa como: $\bar{y}_i = \frac{y_i + y_{i+1}}{2}$
  - Por tanto la integral se aproxima como:

$$I = \int_a^b y\,dx = \frac{1}{2}\sum_{i=1}^{n-1}(y_{i+1} + y_i)(x_{i+1} - x_i)$$

Evaluación de Integrales – método trapezoidal

- Ejemplo:
  - La corriente por una inductancia se puede obtener con la fórmula

$$i = \frac{1}{L}\int_0^t v\,dt$$

donde: i = corriente (amperios), L = inductancia (henrios),

v = voltaje (voltios) y t=tiempo (seg).

Se induce una corriente de 2.15 amperios por un periodo de 500 milisegundos. La variación del voltaje con el tiempo en este periodo se muestra en la siguiente tabla:

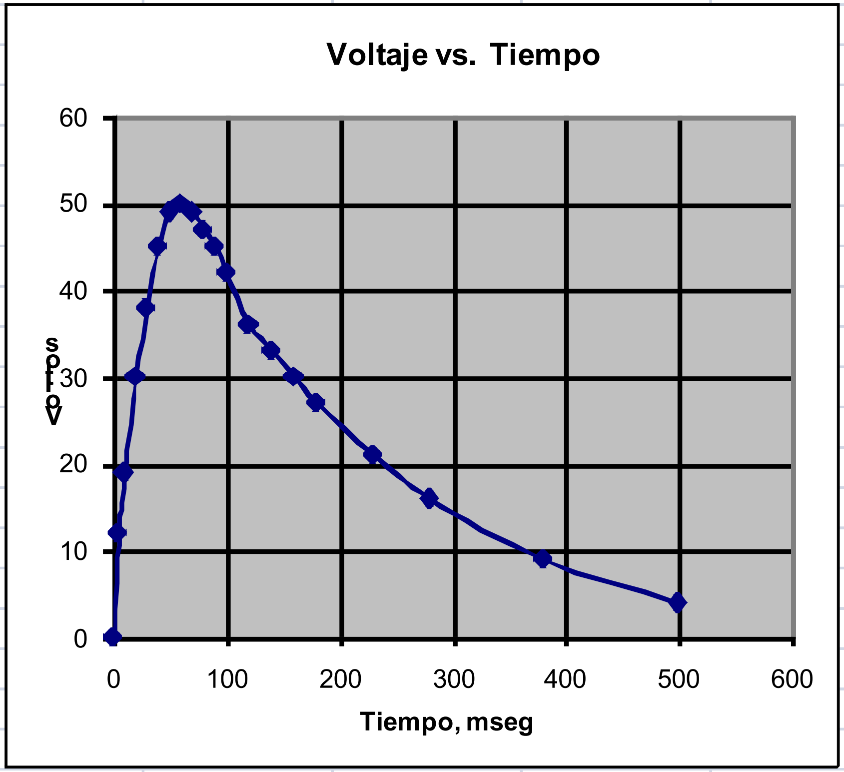
t (msec)	v (volts)	t	v	t	v	t	v
0	0	40	45	90	45	180	27
5	12	50	49	100	42	230	21
10	19	60	50	120	36	280	16
20	30	70	49	140	33	380	9
30	38	80	47	160	30	500	4

Hallar la inductancia evaluando la integral por medio de la regla trapezoidal

Evaluación de Integrales – método trapezoidal

Integración numérica usando la regla trapezoidal con datos espaciados no uniformes

t (mseg)	v (volts)	deltat (seg)	vbar	area
0	0	0.005	6	0.03
5	12	0.005	15.5	0.0775
10	19	0.01	24.5	0.245
20	30	0.01	34	0.34
30	38	0.01	41.5	0.415
40	45	0.01	47	0.47
50	49	0.01	49.5	0.495
60	50	0.01	49.5	0.495
70	49	0.01	48	0.48
80	47	0.01	46	0.46
90	45	0.01	43.5	0.435
100	42	0.02	39	0.78
120	36	0.02	34.5	0.69
140	33	0.02	31.5	0.63
160	30	0.02	28.5	0.57
180	27	0.05	24	1.2
230	21	0.05	18.5	0.925
280	16	0.1	12.5	1.25
380	9	0.12	6.5	0.78
500	4			
			Total =	10.7675



Evaluación de Integrales – método Simpson

- Regla de Simpson (número de datos impar – número de subintervalos par):
  - En lugar de considerar rectángulos entre los puntos, se pasa un polinomio de segundo orden (parábola) a través de tres puntos adyacentes igualmente espaciados.
  - Por tanto la integral se aproxima como:

$$I = \int_a^b y\,dx = \frac{1}{3}(y_1 + 4y_2 + 2y_3 + 4y_4 + 2y_5 + \cdots + 2y_{n-2} + 4y_{n-1} + y_n)\,\Delta x$$

- Ejemplo: Evaluar la integral $I = \int_0^1 e^{-x^2}\,dx$

en el rango de 0 a 1 con un Δ=0.1 entre puntos.

Evaluación de Integrales – método Simpson

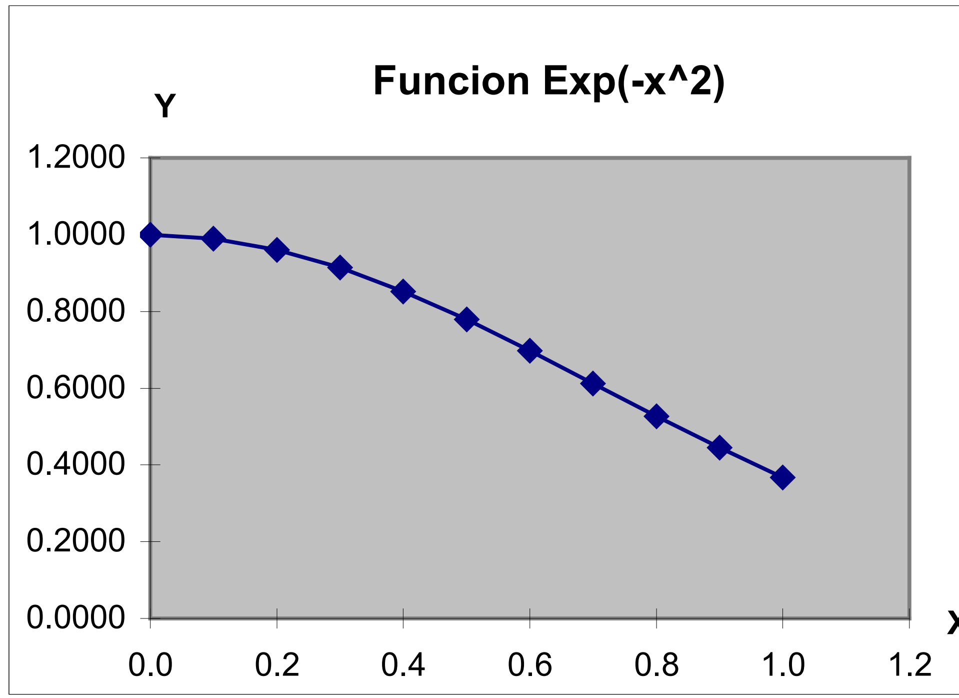
Integración numérica usando la regla de Simpson

$$I = \int_0^1 e^{-x^2}\,dx$$

x	y	
0.0	1.0000	1.0000
0.1	0.9900	3.9602
0.2	0.9608	1.9216
0.3	0.9139	3.6557
0.4	0.8521	1.7043
0.5	0.7788	3.1152
0.6	0.6977	1.3954
0.7	0.6126	2.4505
0.8	0.5273	1.0546
0.9	0.4449	1.7794
1.0	0.3679	0.3679

SUMA = 22.4047

INTEGRAL = 0.7468



Cálculo de la superficie y centroide mediante Integrales

- Problema: para una función dada en forma tabular, calcular el área encerrada por la función y el centro del área.
- Se aplica una técnica de integración numérica para el cálculo del área y el primer momento del área para el centroide.
- El primer momento del área respecto a los ejes x e y se calculan como:

$$M_x = \int y dA = \int \frac{1}{2} y^2 dx \qquad M_y = \int x dA = \int x y dx$$

- De donde se obtiene el centroide: $x_c = \dfrac{M_y}{A} \quad y_c = \dfrac{M_x}{A}$

Cálculo del segundo momento de una superficie

- Problema: calcular el segundo momento de un área (momento de inercia).
- Se usa la misma técnica que la sección anterior para el primer momento, pero usando x^2 e y en lugar de x e y. Es decir para el eje y: $I_y = \int x^2 dA = \int x^2 y dx$
- El momento de inercia para un eje que pasa por el centro de la superficie se calcula aplicando el teorema del eje paralelo,

$$I_{na} = I_y - Ad^2$$

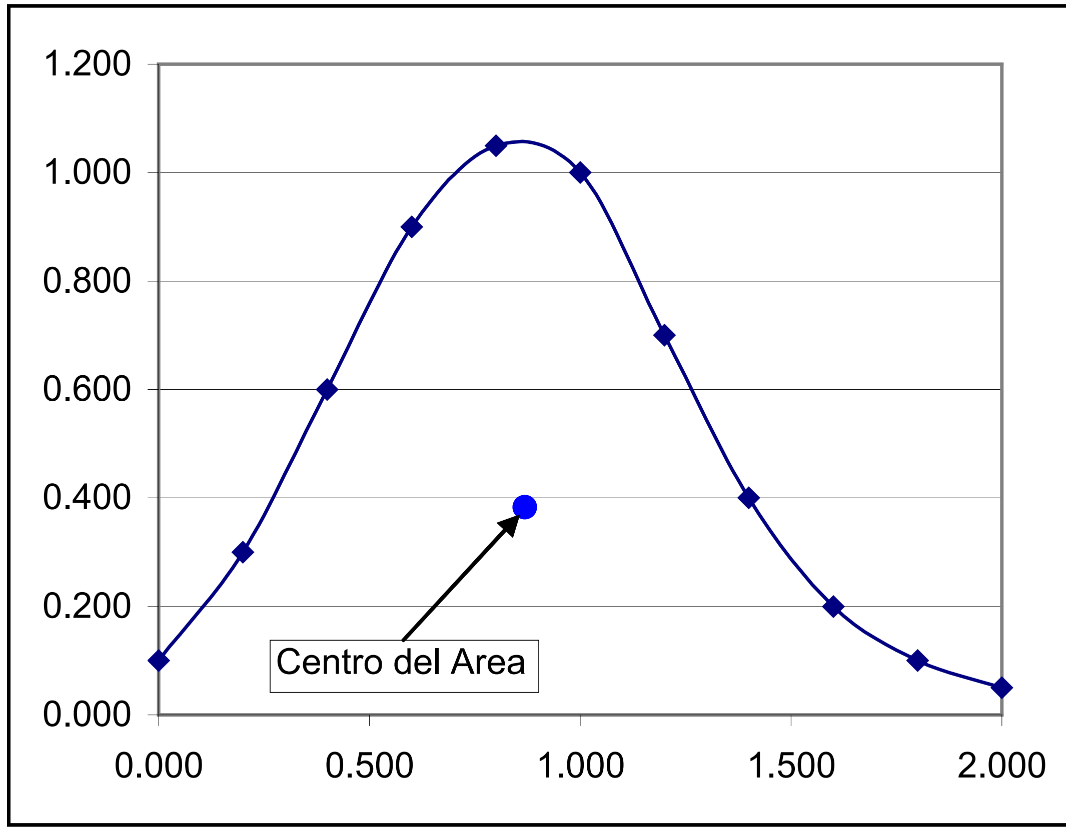
donde I_{na} es el momento de inercia del área sobre el eje paralelo a y que pasa por el centroide, A es el área y d es la distancia al eje y.

Cálculo del centroide y segundo momento de una superficie

Cálculo del Centro y Momento de Inercia de un Area usando integración numérica

x	y	Coef.Simpson
0.000	0.100	1
0.200	0.300	4
0.400	0.600	2
0.600	0.900	4
0.800	1.050	2
1.000	1.000	4
1.200	0.700	2
1.400	0.400	4
1.600	0.200	2
1.800	0.100	4
2.000	0.050	1

s =	0.2
Area =	1.070
xc =	0.869
yc =	0.382
Iy =	0.97013333
Iyc =	0.16297404



Cálculo de Integrales dobles

- Problema: se requiere integrar numéricamente una integral doble para calcular, por ejemplo, un volumen bajo una superficie.
- La técnica propuesta es dividir una integral múltiple en sucesivas integrales sencillas y aplicar las técnicas de integración numérica mostradas antes.
- Así para calcular el volumen bajo una superficie se calculan las áreas de las secciones transversales al eje y y después integrarlas para el eje x.

Cálculo del volumen bajo una superficie

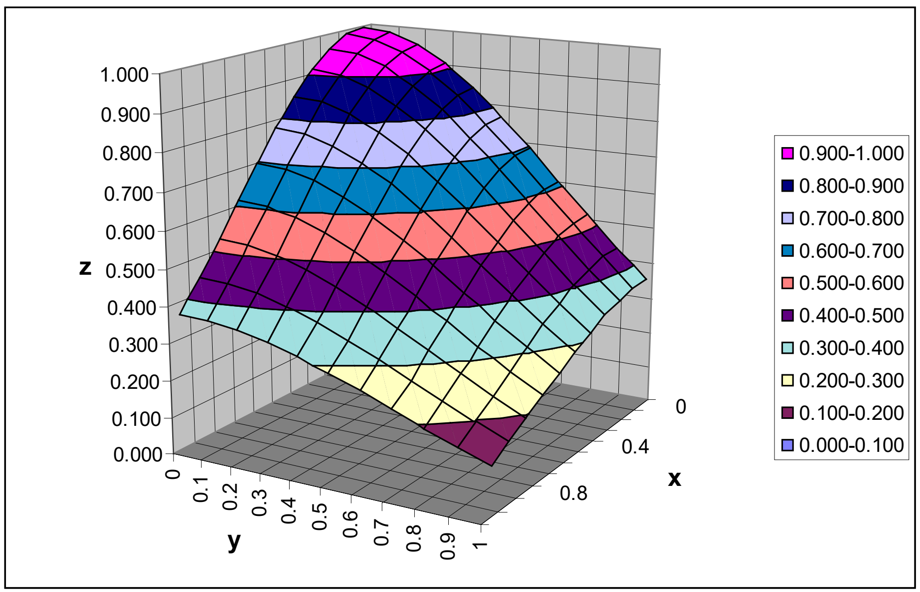
Cálculo de Volumen mediante Integrales dobles

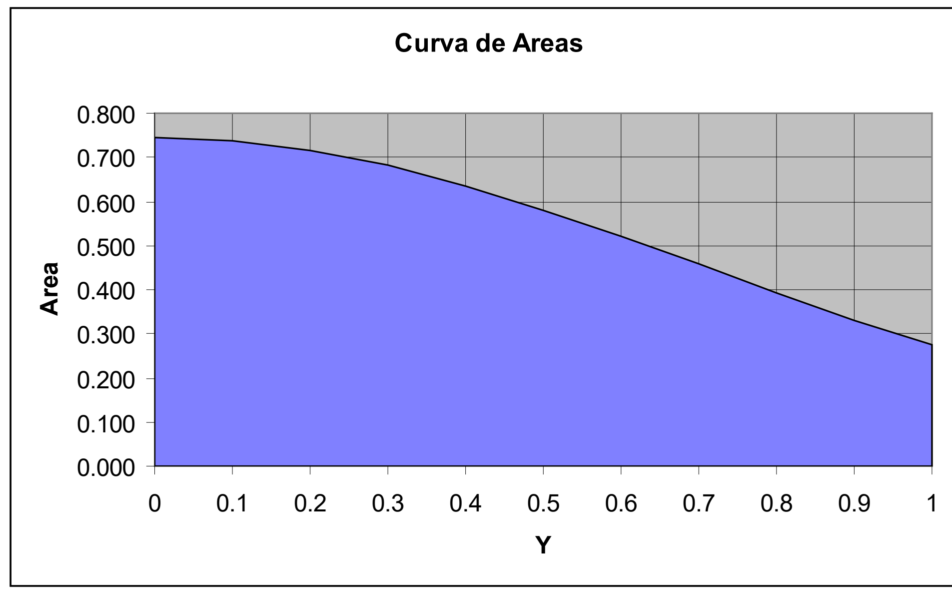
sx = 0.1

sy = 0.1

Coeficientes	x \ y	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
0.5	**0**	1.000	0.990	0.961	0.914	0.852	0.779	0.698	0.613	0.527	0.445	0.368
1	**0.1**	0.990	0.980	0.951	0.905	0.844	0.771	0.691	0.607	0.522	0.440	0.364
1	**0.2**	0.961	0.951	0.923	0.878	0.819	0.748	0.670	0.589	0.507	0.427	0.353
1	**0.3**	0.914	0.905	0.878	0.835	0.779	0.712	0.638	0.560	0.482	0.407	0.336
1	**0.4**	0.852	0.844	0.819	0.779	0.726	0.664	0.595	0.522	0.449	0.379	0.313
1	**0.5**	0.779	0.771	0.748	0.712	0.664	0.607	0.543	0.477	0.411	0.346	0.287
1	**0.6**	0.698	0.691	0.670	0.638	0.595	0.543	0.487	0.427	0.368	0.310	0.257
1	**0.7**	0.613	0.607	0.589	0.560	0.522	0.477	0.427	0.375	0.323	0.273	0.225
1	**0.8**	0.527	0.522	0.507	0.482	0.449	0.411	0.368	0.323	0.278	0.235	0.194
1	**0.9**	0.445	0.440	0.427	0.407	0.379	0.346	0.310	0.273	0.235	0.198	0.164
0.5	**1**	0.368	0.364	0.353	0.336	0.313	0.287	0.257	0.225	0.194	0.164	0.135
	Areas:	0.746	0.739	0.717	0.682	0.636	0.581	0.521	0.457	0.393	0.332	0.275
	Coeficientes:	0.5	1	1	1	1	1	1	1	1	1	0.5

Volumen = 0.55683





Método de Monte Carlo en Integración

- El método de Monte Carlo es un método estadístico numérico basado en la aplicación repetida de muestras aleatorias.
- Se puede aplicar en integración, considerando el área delimitada por los límites de integración y un valor superior al máximo de la función a integrar (S). Se generan una serie de pares de números aleatorios (N dardos) dentro del área, contando los que quedan bajo la curva que describe la función (n).
- La integral se calcula como S*n/N

Optimización

Optimización

- Los problemas de ingeniería se modelan mediante sistemas de ecuaciones para su análisis.
- Muchos problemas de ingeniería requieren la optimización de un criterio, como el costo, ganancia, peso, etc, al que se llama **función objetivo**.
- Adicionalmente hay una serie de condiciones, tales como leyes de conservación, restricciones de capacidad u otra restricción técnica, que deben ser satisfechas. Estas condiciones se llaman **restricciones**.

Optimización

- El objetivo de una solución óptima es determinar una solución que produce que la función objetivo sea *maximizada* o *minimizada* cumpliendo todas las restricciones.
- Los problemas de este tipo se conocen como problemas de **optimización**.

Optimización - Formalización

- Un problema de optimización se puede escribir como: Determinar los valores de la variables $x_1, x_2, \ldots, x_n$ que maximice o minimice la **función objetivo**:

$$y = f(x_1, x_2, \cdots, x_n)$$

Sujeto a las siguientes restricciones $j = 1, 2, \ldots, m$:

$$g_j(x_1, x_2, \cdots, x_n) = 0$$

o $$g_j(x_1, x_2, \cdots, x_n) \leq 0$$

o $$g_j(x_1, x_2, \cdots, x_n) \geq 0$$

además, es común restringir $x_i \geq 0$ $i = 1, 2, \ldots, n$

- Ejemplos: Optimizacion.xlsx

Optimización – Métodos y técnicas de solución

- Los problemas de optimización se pueden clasificar en lineales y no lineales.
- Las técnicas para solucionar los problemas de optimización se basan en cálculo, enumerativas y aleatorias.
- Dentro de las técnicas basadas en cálculo están el Simplex para programación lineal y GRG para problemas no lineales continuos. Dentro de las técnicas aleatorias están las Evolutivas (algoritmos genéticos) para problemas no continuos.
- Referencias: Simplex, GRG, Evolutivos

Optimización usando Solver

- Procedimiento:
  - Escribir un *valor inicial* para cada variable independiente $x_1, x_2 \ldots, x_n$ en celdas diferentes.
  - Escribir la *función objetivo* como fórmula Excel en una celda.
  - Escribir las ecuaciones de *cada restricción* como fórmulas Excel.
  - Seleccionar Herramientas→Solver. Dar la dirección de la celda que contiene la función objetivo para *Celda objetivo*. Seleccionar Máximo o Mínimo en *Valor de la celda objetivo*. Dar el rango de las celdas que contienen los valores iniciales de las variables $x_1, x_2 \ldots, x_n$ en *Cambiando las celdas*.
  - Escribir las celdas que contienen cada restricción, el tipo de restricción y el valor del lado derecho usando Agregar.
  - Si la función objetivo y las restricciones son lineales, pulsar en el botón Opciones y seleccionar *Adoptar Modelo Lineal*.
  - Pulsar Aceptar y después Resolver. Se puede seleccionar la opción de generar Resultados en otra hoja.

Optimización - Programación lineal clásica

- Maximizar $f(x_1, x_2) = 29\, x_1 + 45\, x_2$

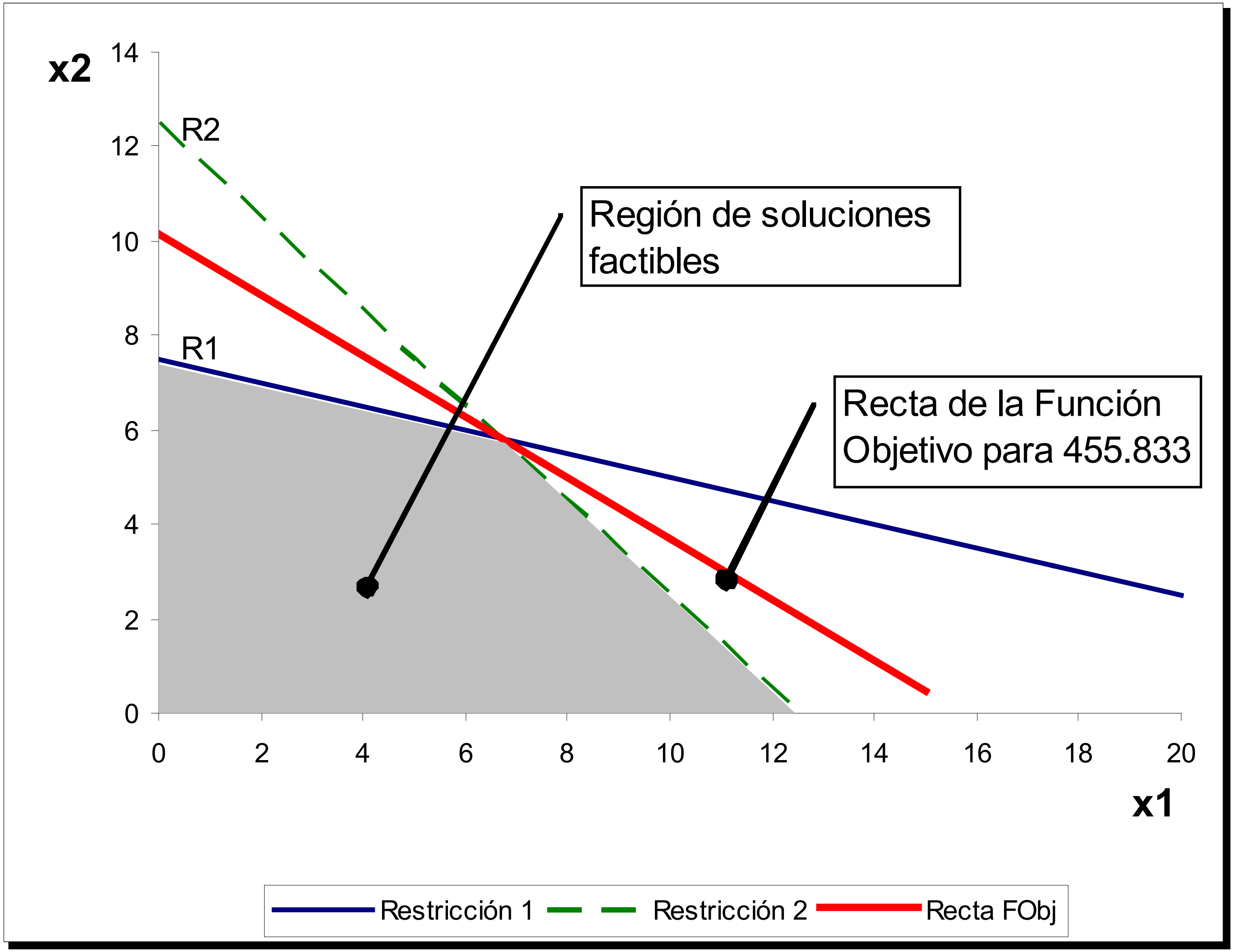
  Sujeto a las restricciones:

  $2x_1 + 8x_2 \leq 60$

  $4x_1 + 4x_2 \leq 50$

  $x_1, x_2 \geq 0$

Optimización - Programación lineal clásica

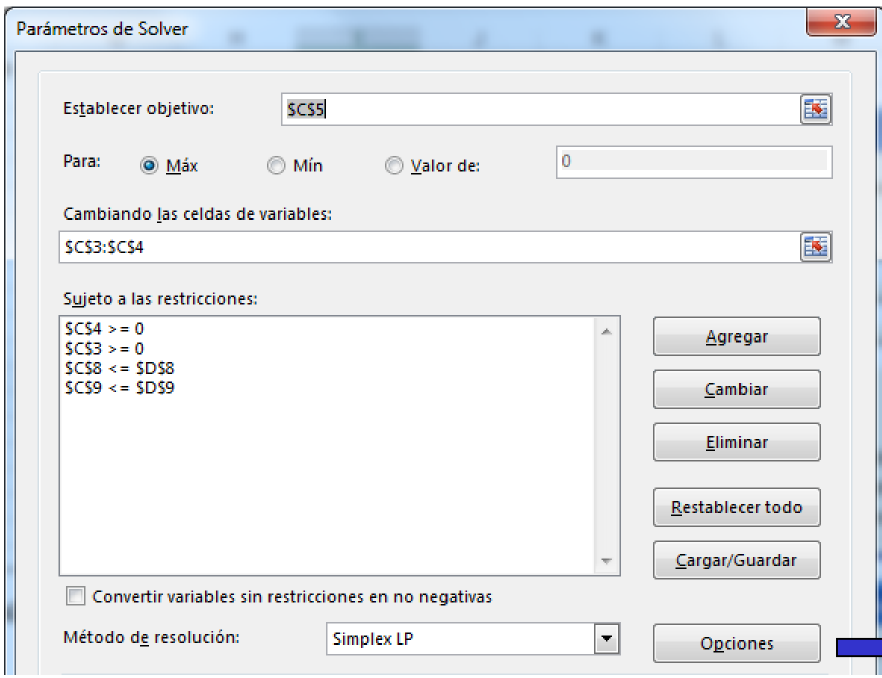


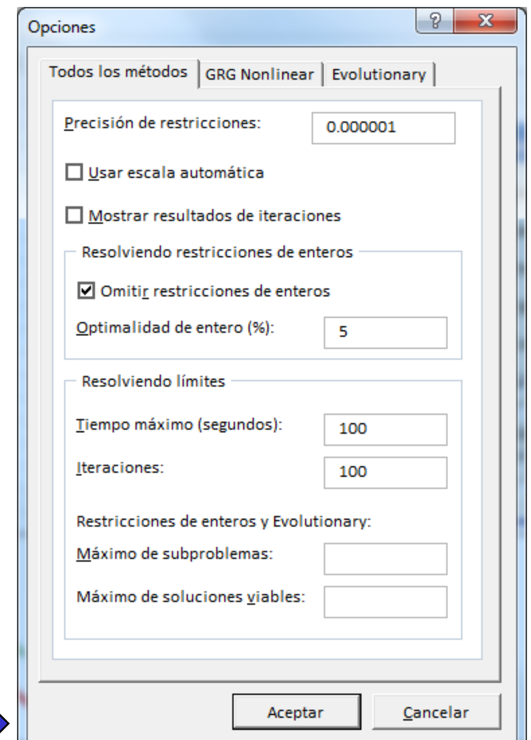
Optimización – Simplex LP Solver

Solución (Solver) de problema de programación lineal clá

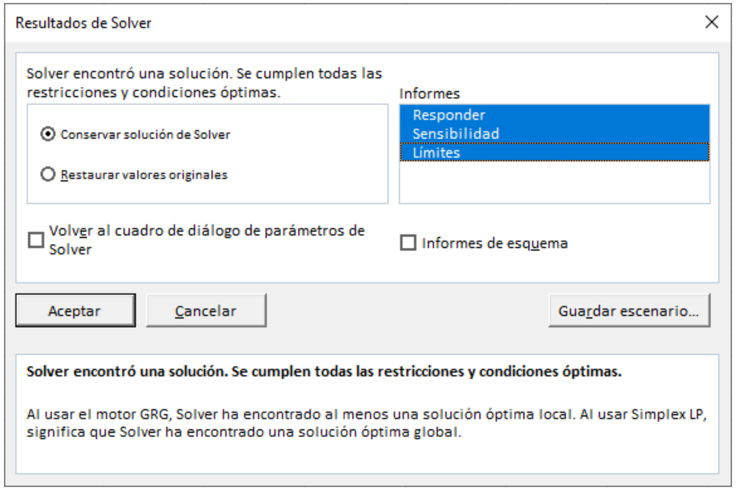
x1 =	6.6667
x2 =	5.8333
f(x1, x2) =	**455.8333 ? Función Objectivo**

	Valor	Límite
Restricción 1	60	60
Restricción 2	50	50





Optimización - Simplex LP Solver



Informe Responder

Resultado: Solver encontró una solución. Se cumplen todas las restricciones y condiciones óptimas.

Motor de Solver

- Motor: Simplex LP
- Tiempo de la solución: 0,016 segundos.
- Iteraciones: 2 Subproblemas: 0

Opciones de Solver

- Tiempo máximo Ilimitado, Iteraciones Ilimitado, Precision 0,000001, Usar escala automática
- Máximo de subproblemas Ilimitado, Máximo de soluciones de enteros Ilimitado, Tolerancia de enteros 1%, Asumir no negativo

Celda objetivo (Máx)

Celda	Nombre	Valor original	Valor final
$B$4	fo	74	455.8333333

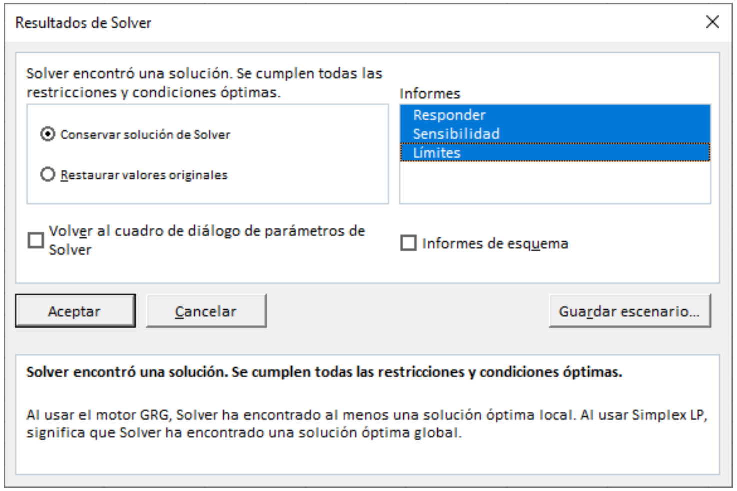
Celdas de variables

Celda	Nombre	Valor original	Valor final	Entero
$B$1	x1	1	6.666666667	Continuar
$B$2	x2	1	5.833333333	Continuar

Restricciones

Celda	Nombre	Valor de la celda	Fórmula	Estado	Demora
$B$6	r1	60	$B$6<=$C$6	Vinculante	0
$B$7	r2	50	$B$7<=$C$7	Vinculante	0

Optimización - Simplex LP Solver



Informe Sensibilidad

Celdas de variables

Celda	Nombre	Final Valor	Reducido Coste	Objetivo Coeficiente	Permisible Aumentar	Permisible Reducir
$B$1	x1	6.666666667	0	29	16	17.75
$B$2	x2	5.833333333	0	45	71	16

Restricciones

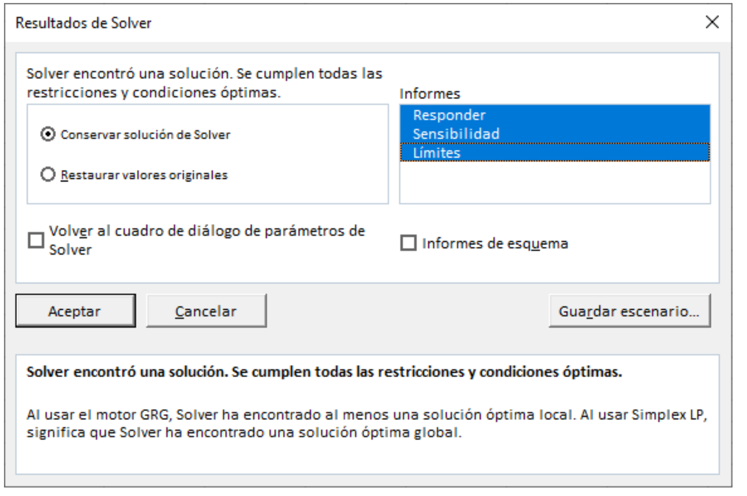
Celda	Nombre	Final Valor	Sombra Precio	Restricción Lado derecho	Permisible Aumentar	Permisible Reducir
$B$6	r1	60	2.666666667	60	40	35
$B$7	r2	50	5.916666667	50	70	20

Informe Límites

Celda	Objetivo Nombre	Valor
$B$4	fo	455.8

Celda	Variable Nombre	Valor	Inferior Límite	Objetivo Resultado	Superior Límite	Objetivo Resultado
$B$1	x1	6.667	0	262.5	6.666667	455.83333
$B$2	x2	5.833	0	193.33333	5.833333	455.83333

Optimización - Simplex LP Solver Informes



- Conviene seleccionar los informes:
  - Responder o respuestas:
    - Muestra los valores finales de la función objetivo, de las variables y de las restricciones. En las restricciones muestra si se han cumplido las igualdades (Vinculante) o no (No vinculante).
  - Sensibilidad:
    - Celdas Variables:
      - Final Valor: indica los valores de la solución óptima para cada variable.
      - Reducido Coste: se interpreta como la variación que tendrá el valor final de la función objetivo por cada unidad que variamos en una determinada variable.
      - Objetivo Coeficiente: Representan los coeficientes que tiene la función objetivo.

Optimización - Simplex LP Solver Informes

  - Permisible Aumentar: Evalúa el nivel de incremento permitido en los coeficientes de la función objetivo, sin cambiar la solución óptima.
  - Permisible Reducir: Evalúa el nivel de reducción permitido en los coeficientes de la función objetivo, sin cambiar la solución óptima.
- Restricciones:
  - Final Valor: Representa el valor que toma la restricción en la solución óptima.
  - Precio Sombra: También conocido como precio dual; indican en cuanto mejora o disminuye la función objetivo, si se aumenta/disminuye el límite que la restringe.
  - Restricción Lado Derecho: Indica el valor que tiene el lado derecho de las desigualdades. En algunos casos son los mismos valores que se indican en la columna “Disponible”.
  - Permisible Aumentar: Hacen referencia hasta que punto puede incrementarse la desigualdad de la restricción sin que varíe el precio sombra.
  - Permisible Reducir: Representa hasta que punto puede disminuirse la desigualdad de la restricción sin afectar el precio sombra.

– Límites:
- Valor: valor óptimo de cada una de las variables.
- Límite Inferior/Superior: Es el menor/mayor valor que puede tomar la variable y cumplir todas las restricciones, cuando las demás variables mantienen su valor óptimo.
- Resultado Objetivo: Es el valor de la función objetivo si la variable toma el valor del límite inferior/superior y las otras variables mantienen su valor óptimo.

Optimización - Simplex LP Solver
Informes

- Problema de optimización de producción:

Maximizar $y = 60\,x_1 + 44\,x_2$

Sujeto a las restricciones:

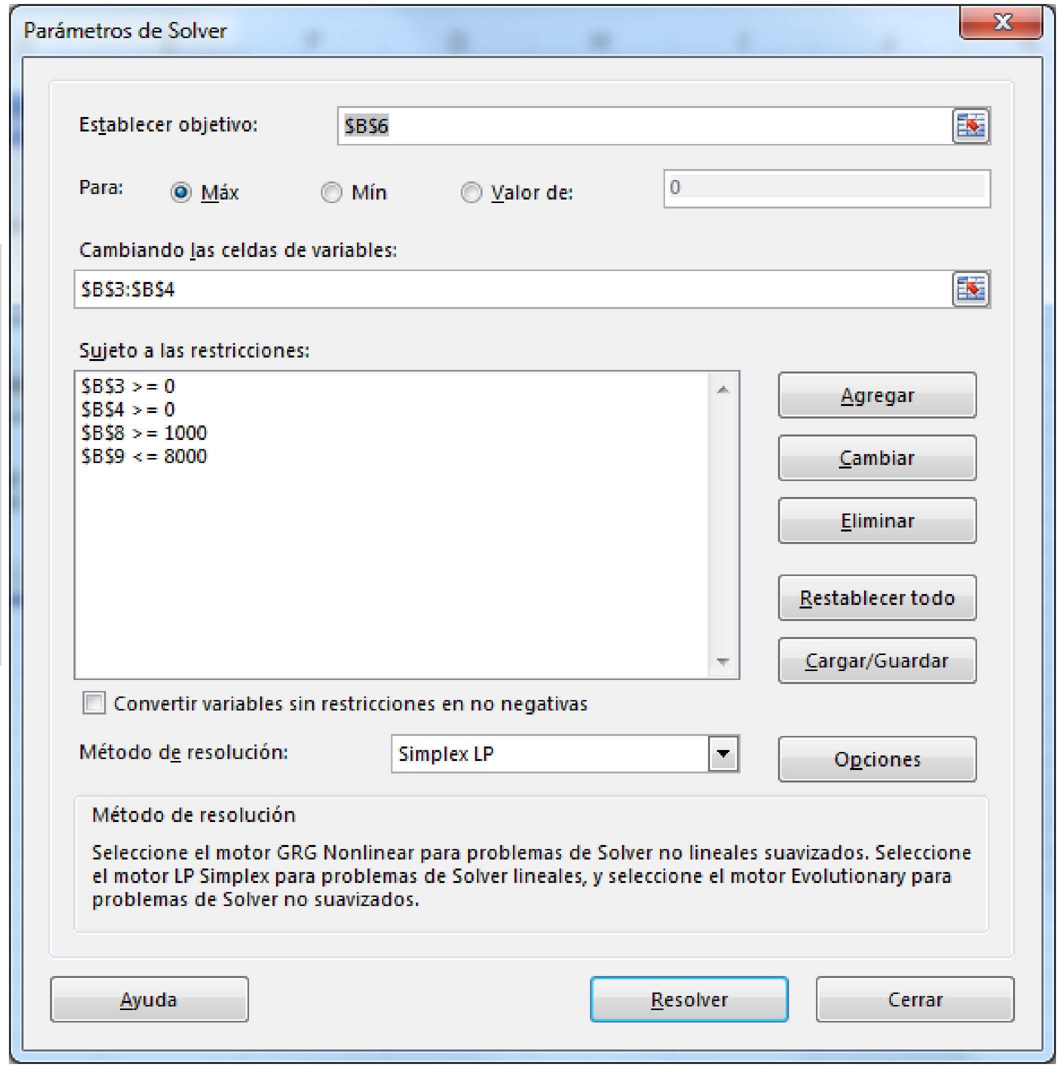
$$x_1 + x_2 \geq 1000$$

$$5x_1 + 3\,x_2 \leq 8000$$

$$x_1, x_2 \geq 0$$

Optimización - Programación lineal clásica

Optimización de la producción	
Unidades de A /mes:	0
Unidades de B /mes:	2666.66667
Ganacia (y) =	117333.333
Producción mínima requerida (g1):	2666.66667
Disponibilidad mano de obra (g2):	8000



Optimización - Simplex LP Solver informes

Informe Responder

Celda objetivo (Máx)

Celda	Nombre	Valor original	Valor final
$B$6	Ganacia (y) =	117333.3333	117333.3333

Celdas de variables

Celda	Nombre	Valor original	Valor final	Entero
$B$3	Unidades de A /mes:	0	0	Continuar
$B$4	Unidades de B /mes:	2666.666667	2666.666667	Continuar

Restricciones

Celda	Nombre	Valor de la celda	Fórmula	Estado	Demora
$B$8	Producción mínima requerida (g1):	2666.666667	$B$8>=1000	No vinculante	1666.666667
$B$9	Disponibilidad mano de obra (g2):	8000	$B$9<=8000	Vinculante	0
$B$3	Unidades de A /mes:	0	$B$3>=0	Vinculante	0
$B$4	Unidades de B /mes:	2666.666667	$B$4>=0	No vinculante	2666.666667

Informe Límites

	Objetivo	
Celda	Nombre	Valor
$B$6	Ganacia (y) =	117333.33

	Variable		Inferior	Objetivo	Superior	Objetivo
Celda	Nombre	Valor	Límite	Resultado	Límite	Resultado
$B$3	Unidades de A /mes:	0	0	117333.33	-1.8E-13	117333.33
$B$4	Unidades de B /mes:	2666.6667	1000	44000	2666.667	117333.33

Informe Sensibilidad

Celdas de variables

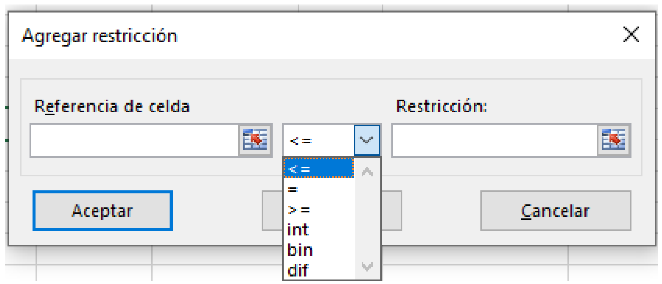
		Final	Reducido	Objetivo	Permisible	Permisible
Celda	Nombre	Valor	Coste	Coeficiente	Aumentar	Reducir
$B$3	Unidades de A /mes:	0	-13.33333333	60	13.33333333	1E+30
$B$4	Unidades de B /mes:	2666.666667	0	44	1E+30	8

Restricciones

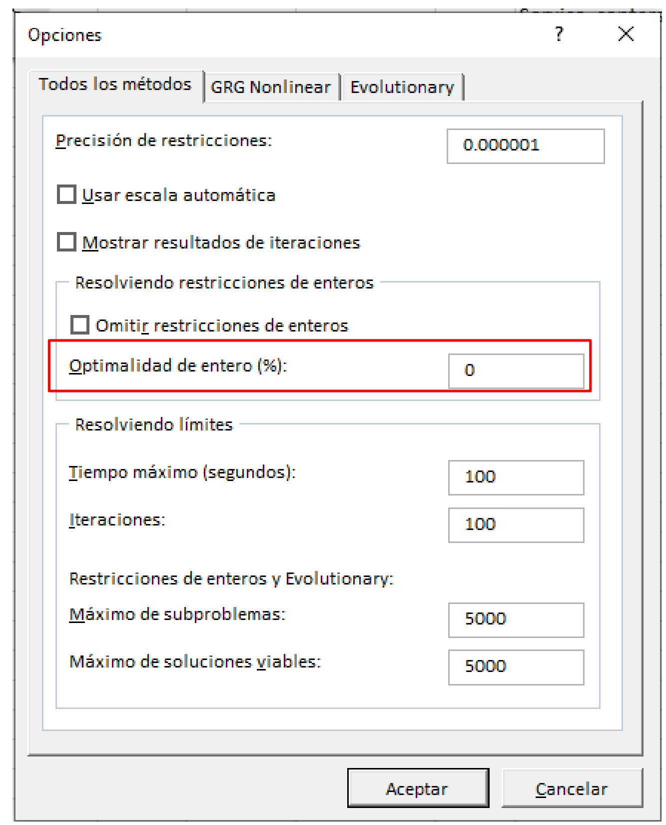
		Final	Sombra	Restricción	Permisible	Permisible
Celda	Nombre	Valor	Precio	Lado derecho	Aumentar	Reducir
$B$8	Producción mínima requerida (g1):	2666.666667	0	1000	1666.666667	1E+30
$B$9	Disponibilidad mano de obra (g2):	8000	14.66666667	8000	1E+30	5000

Optimización - Simplex LP Restricciones de tipo entero o binario

- Se puede resolver problemas de optimización utilizando otros tipos de restricciones dependiendo del planteamiento del problema:



- En caso de usar int o bin es necesario configurar la Optimalidad de entero (%), que indica si Solver parará en una solución cerca del óptimo o continuará con la búsqueda. El valor de 1% o 5% no garantiza optimalidad; con 0% si lo hace a costa del tiempo.



Optimización - Dualidad

Primal

$$\text{Maximize } z = \sum_{j=1}^{n} c_j x_j,$$

subject to:

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i \qquad (i = 1, 2, \ldots, m),$$

$$x_j \geq 0 \qquad (j = 1, 2, \ldots, n).$$

Associated with this primal problem there is a corresponding dual problem given by:

Dual

$$\text{Minimize } v = \sum_{i=1}^{m} b_i y_i,$$

subject to:

$$\sum_{i=1}^{m} a_{ij} y_i \geq c_j \qquad (j = 1, 2, \ldots, n),$$

$$y_i \geq 0 \qquad (i = 1, 2, \ldots, m).$$

Optimización - no lineal

- Minimizar $y = 10 + (x_1 - 0.5)^2 + (x_2 + 0.5)^2$ (problema no lineal)

Sujeto a las restricciones:

$\pi\,(x_1^2 + x_2^2) \geq 10$

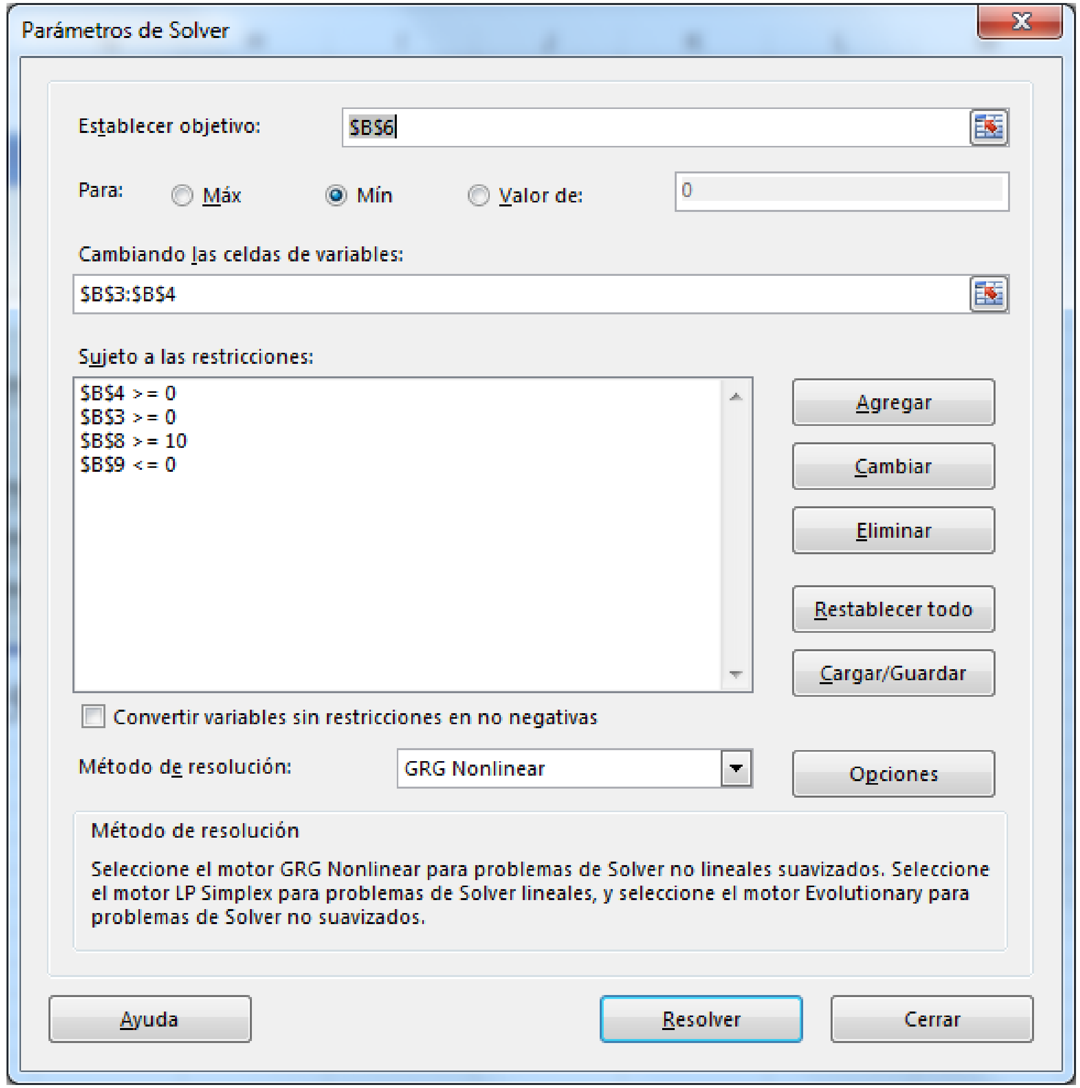
$x_1 \leq 1.25\, x_2$

$x_1, x_2 \geq 0$

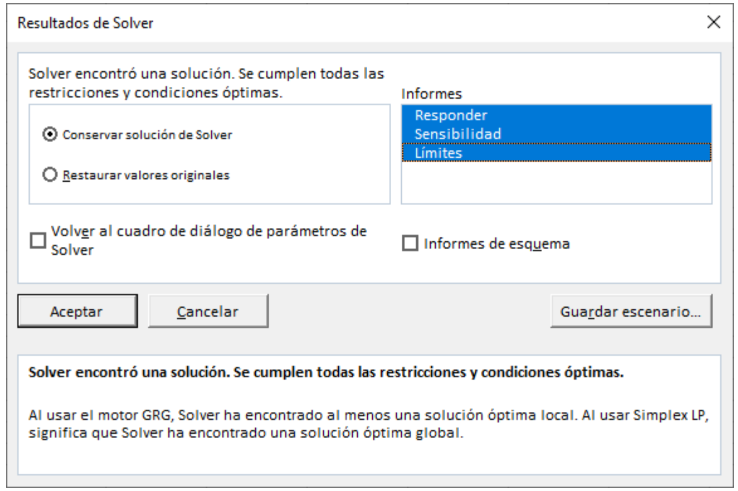
Nota: Excel **No** indica si la solución es una solución local o global.

Optimización - no lineal

Optimización no lineal	
x1 =	1.393166873
x2 =	1.114533498
y =	13.40446548
g1 =	9.999999972
g2 =	0



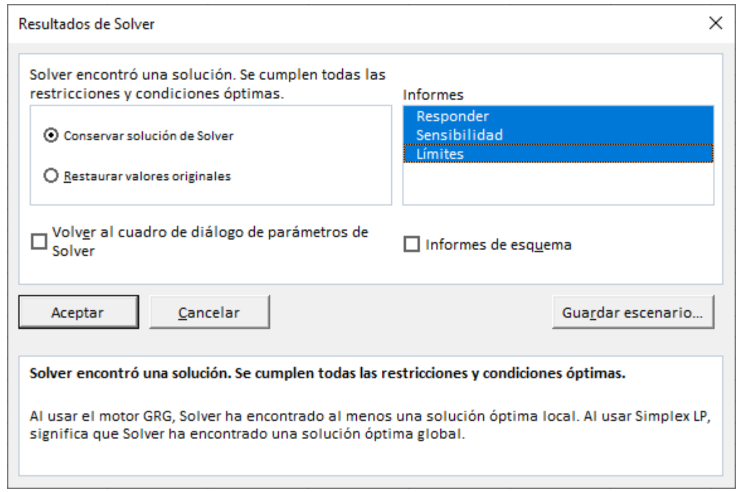
Optimización no lineal - Informes



Similar al de Simplex LP con la diferencia en el informe de Sensibilidad

Celdas de variables			
		Final	Reducido
Celda	Nombre	Valor	Degradado
$B$3	x1 =	1.393166873	0
$B$4	x2 =	1.114533498	0
Restricciones			
		Final	Lagrange
Celda	Nombre	Valor	Multiplicador
$B$8	g1 =	9.999999972	0.304378221
$B$9	g2 =	0	-0.878048733

Optimización no lineal – Informe sensibilidad



- Similar al de Simplex LP con la diferencia en:
  - Sensibilidad:
    - Restricciones:
      - Lagrange Multiplicador: representa el valor marginal, es decir, en cuánto mejoraría (aumenta en caso de máximo, disminuye en caso de mínimo) el valor actual de la función objetivo si se relajara la restricción asociada.

Optimización - no lineal

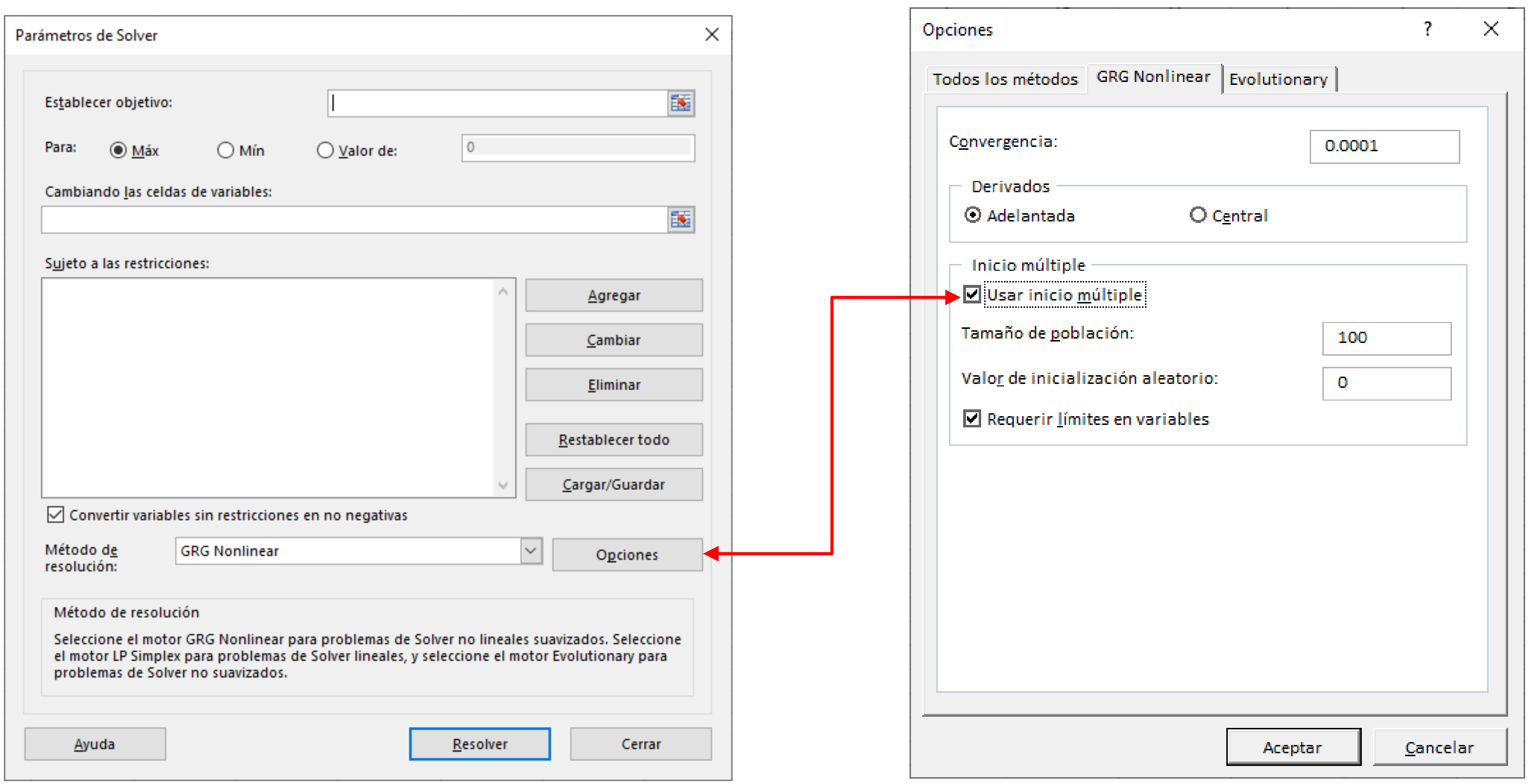
- La solución de problemas de optimización no lineales pueden depender de los valores iniciales supuestos, con lo que se pueden obtener soluciones óptimas locales y no hallar la solución óptima global.
- Una opción es graficar la función objetivo para ver su comportamiento en problemas sencillos, sino se puede usar técnicas de Monte Carlo (aleatorio).
- Ej: Minimizar

$-3 \leq x_1, x_2 \leq 3$

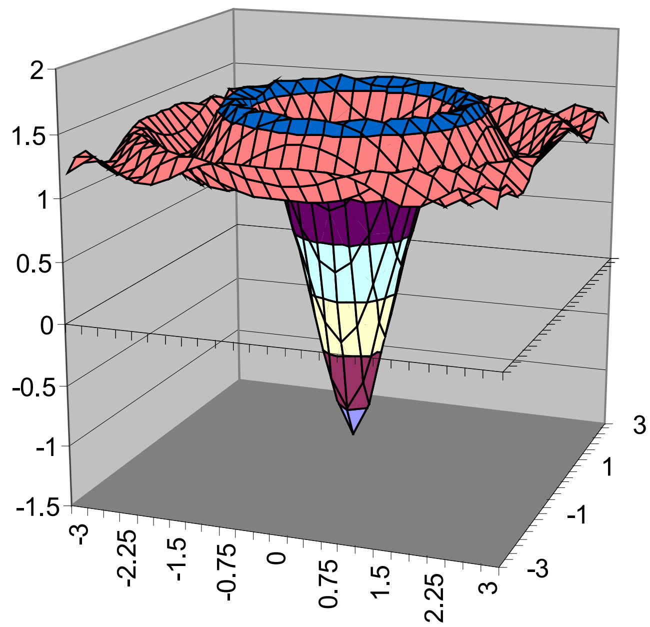
$$z = \left(1 - \frac{\cos(x^2 + y^2)}{(x^2 + y^2 + 0.5)}\right) \cdot 1.25$$

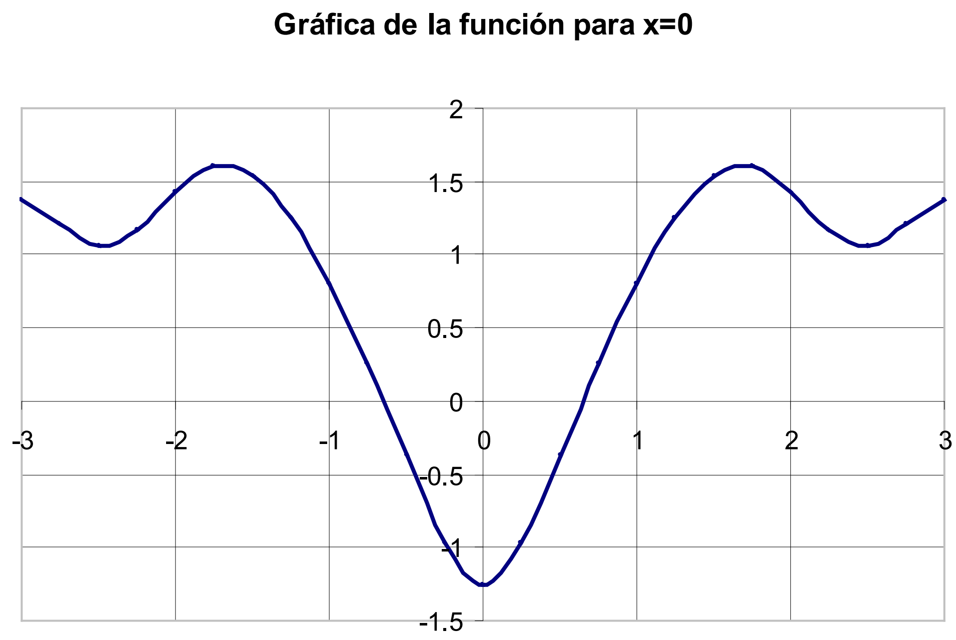
Optimización - no lineal

- Es posible obtener una solución global seleccionando la opción de "Usar inicio múltiple" en la pestaña GRG Nonlinear de Opciones:



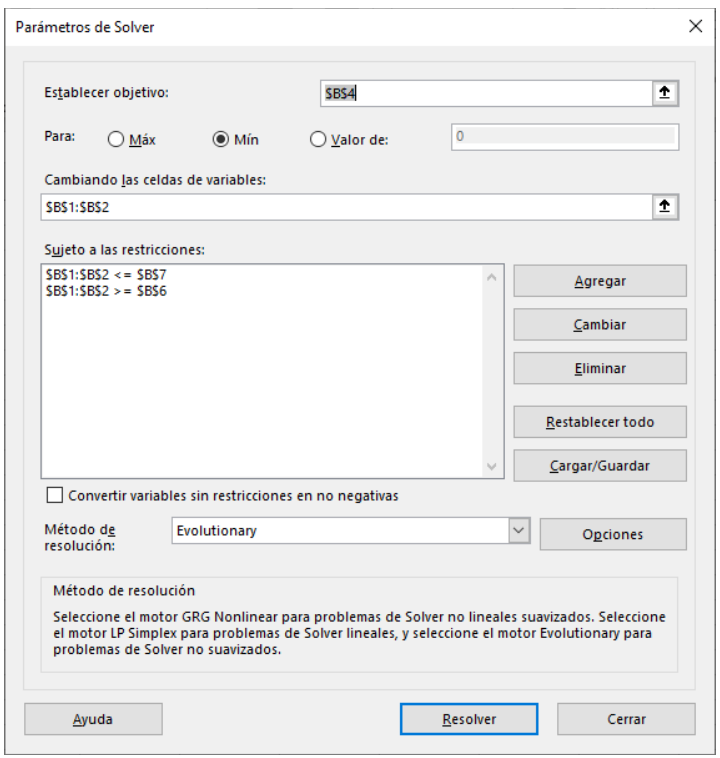
Optimización - no lineal

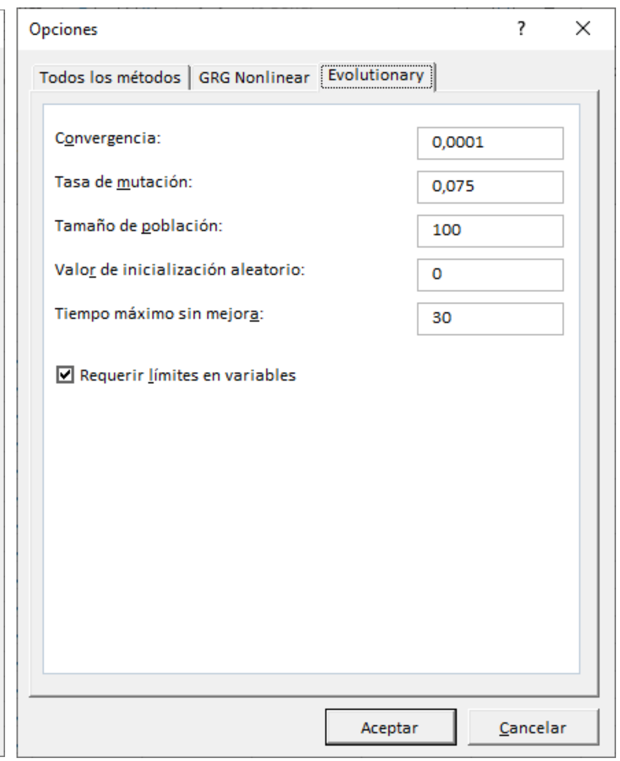




Optimización – método Evolutionary

- Cuando la función a optimizar no es continua o diferenciable y se require obtener el valor óptimo global puede usarse el método de resolución Evolutionary basado en algoritmo genético.





Optimización multiobjetivo

- La *verdadera optimización* es multiobjetivo: los problemas reales en general involucran más de un objetivo a la vez. Ejemplo: en el diseño de un dispositivo electrónico se desea maximizar el desempeño y minimizar el costo de manufacturarlo, así como el tiempo medio entre fallas.
- Matemáticamente se puede formular:

$$\begin{aligned} &\text{minimize} \quad f = [f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x})] \\ &\text{subject to} \quad \mathbf{x} \in \Omega \end{aligned}$$

- Tales problemas se resuelven considerando la optimalidad de Pareto.
- Existen varios métodos para identificar una solución de compromiso.