Universidad de Chile
Facultad de Ciencias Físicas y Matemáticas
Departamento de Ingeniería Química y Biotecnología
IQ46B – Operaciones de Transferencias I

MATLAB: MATRIX LABORATORY

Igor Guzmán Olivares

0. ESTRUCTURA DE LA PRESENTACIÓN

1.- Método Numérico para resolución de EDOs (Runge – Kutta).

2.- Reseña de MATLAB y ejemplo de resolución de cinética de Michaelis – Menten.

3.- Método Numérico para resolución de EDP (Diferencias Finitas).

4.- Ejemplo de transmisión de calor transiente.

1. Resolución de ecuaciones diferenciales

Una **ecuación diferencial** es una expresión matemática que relaciona de manera *no trivial* a una función desconocida con una o más de sus derivadas con respecto a una o más variables independientes.

- Si la función desconocida depende de una sola variable la ecuación diferencial se denomina *ordinaria*.
- Si la función desconocida depende de más de una variable la ecuación diferencial se denomina *parcial*.

EDO: ECUACIONES DIFERENCIALES ORDINARIAS

$$\boxed{F(x,\ y,\ y^{(1)},\ y^{(2)},\ \ldots,\ y^{(n)}) = 0}$$

y: función dependiente de x

x: variable independiente

n: orden de la derivada (número natural)

1. Resolución de ecuaciones diferenciales

Unicidad de solución:

Un *problema de valor inicial* o *de Cauchy*

$$\frac{d^n y}{dx^n} = f(x, \ y, \ y^{(1)}, \ .., \ y^{(n-1)})$$
$$y(x_0) = y_0$$
$$y^{(1)}(x_0) = y_1$$
$$y^{(2)}(x_0) = y_2$$
$$.$$
$$.$$
$$.$$
$$y^{(n-1)}(x_0) = y_{n-1}$$

Un *problema de valores en la frontera* o *de Dirichlet*

$$\frac{d^n y}{dx^n} = f(x, \ y, \ y^{(1)}, \ .., \ y^{(n-1)})$$
$$y(x_0) = y_0$$
$$y(x_1) = y_1$$
$$y(x_2) = y_2$$
$$.$$
$$.$$
$$.$$
$$y(x_{n-1}) = y_{n-1}$$

1. RESOLUCIÓN DE ECUACIONES DIFERENCIALES

¡IMPORTANTE!

Una EDO de orden mayor que 1 siempre se puede convertir en un sistema EDO de primer orden:

$$
\begin{aligned}
\frac{d^n y}{dx^n} &= f(x,\ y,\ y^{(1)},\ ..,\ y^{(n-1)}) \\
y(x_0) &= y_0 \\
y^{(1)}(x_0) &= y_1 \\
y^{(2)}(x_0) &= y_2 \\
&. \\
&. \\
&. \\
y^{(n-1)}(x_0) &= y_{n-1}
\end{aligned}
$$

cambio de variables

$$
Y = \begin{bmatrix} \eta_1 = y \\ \eta_2 = \dfrac{dy}{dx} \\ \eta_3 = \dfrac{d^2 y}{dx^2} \\ . \\ . \\ . \\ \eta_n = \dfrac{d^{n-1} y}{dx^{n-1}} \end{bmatrix}, \quad F(x,\ Y) = \begin{bmatrix} \eta_2 \\ \eta_3 \\ \eta_4 \\ . \\ . \\ . \\ f(x,\ y,\ y^{(1)},\ ...,\ y^{(n-1)}) \end{bmatrix}
$$

$$
\boxed{\begin{aligned} \frac{dY}{dx} &= F(x,\ Y) \\ Y(x_0) &= Y_0 \end{aligned}}
$$

1.1 MÉTODOS NUMÉRICOS

Nos avocaremos a encontrar la función y(x) definida en el intervalo [a, b] que cumpla:

$$\begin{cases} \dfrac{dy}{dx} = f(x, y), \; \forall x \in [a, b] \\ y(x_0) = y_0 \end{cases}$$

- Método de Euler
- Método de Trapecio
- Método de punto medio
- Métodos Runge - Kutta

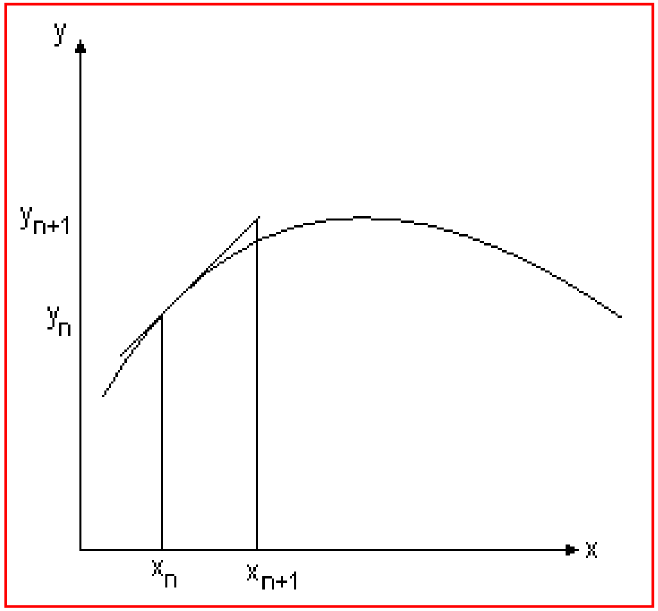
1.1 MÉTODOS NUMÉRICOS

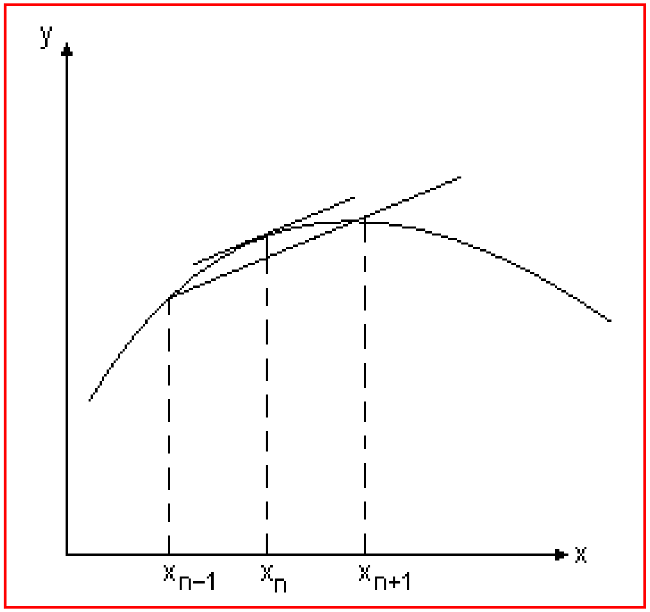
Método de Euler:

$$s'(x_n, y_n) \approx \frac{y_{n+1} - y_n}{h} \Rightarrow \boxed{y_{n+1} = y_n + h \cdot s'(x_n, y_n)}$$

Método del Punto Medio:

$$s'(x_n, y_n) \approx \frac{y_{n+1} - y_{n-1}}{2 \cdot h} \Rightarrow \boxed{y_{n+1} = y_{n-1} + 2 \cdot h \cdot f(x_n, y_n)}$$





1.1 Métodos numéricos: Runge - Kutta

Los métodos Runge – Kutta (RK) se pueden expresar como:

$$y_{n+1} = y_n + \psi(x_n, y_n, h)h$$

$\psi(x_n, y_n, h)$: Función de incrementos

La idea básica detrás de los métodos RK es hallar la pendiente en x_i y estimarla en otros puntos intermedios, se combinan linealmente, se multiplica este valor por h y se suma a y_i

$$\psi(x_n, y_n, h) = \sum_{i=1}^{v} w_i \cdot k_i \cdot h$$

$$k_i = \left(x_n + c_i h,\ y_n + \sum_{j=1}^{i-1} a_{ij} k_j h \right),\ i = 1,2,\ldots\ldots,v;\ c_1 = 0$$

1.1 MÉTODOS NUMÉRICOS: RUNGE - KUTTA

$$y_{n+1} = y_n + \psi(x_n, y_n, h)h$$

Visto de una manera mas amable:

$$\psi(x_n, y_n, h) = \sum_{i=1}^{v} w_i \cdot k_i \cdot h$$

$$k_1 = f(x_n, y_n)$$
$$k_2 = f(x_n + c_2 h, y_n + a_{21} k_1 h)$$
$$k_3 = f(x_n + c_3 h, y_n + a_{31} k_1 h + a_{32} k_2 h)$$

$$Donde\; c_2, c_3, \ldots, c_v, c_{21}, c_{v(v-1)} \; y \; w_i \; son\; en\; principio\; arbitrarios$$

1.1 Métodos numéricos: Runge - Kutta

Método de RK de cuarto orden:

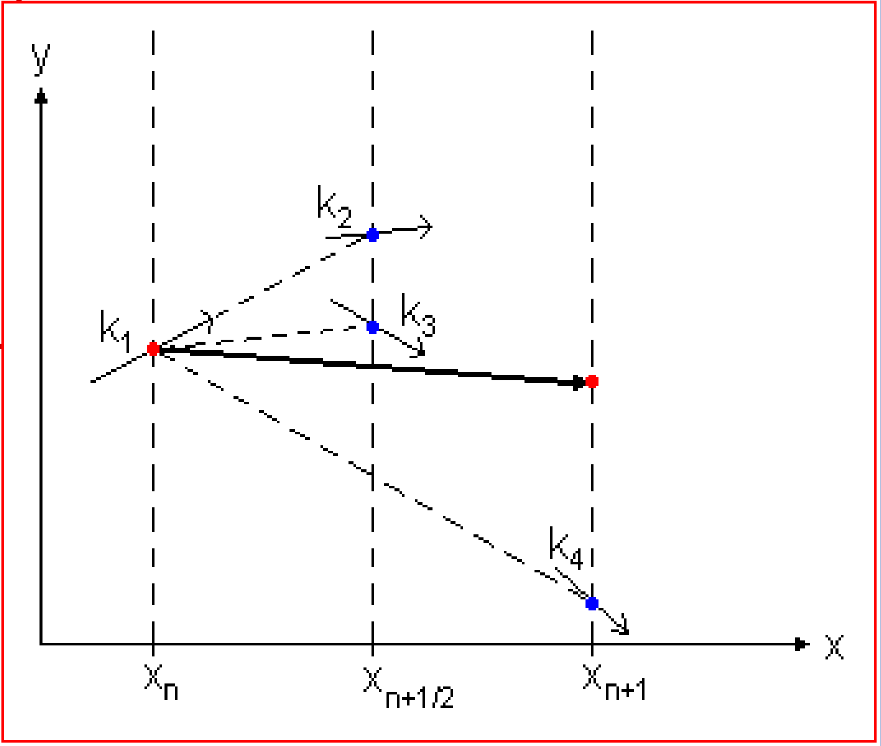
$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)h$$

$$k_1 = f(x_n, y_n)$$

$$k_2 = f(x_n + \frac{1}{2}h,\ y_n + \frac{1}{2}k_1 h)$$

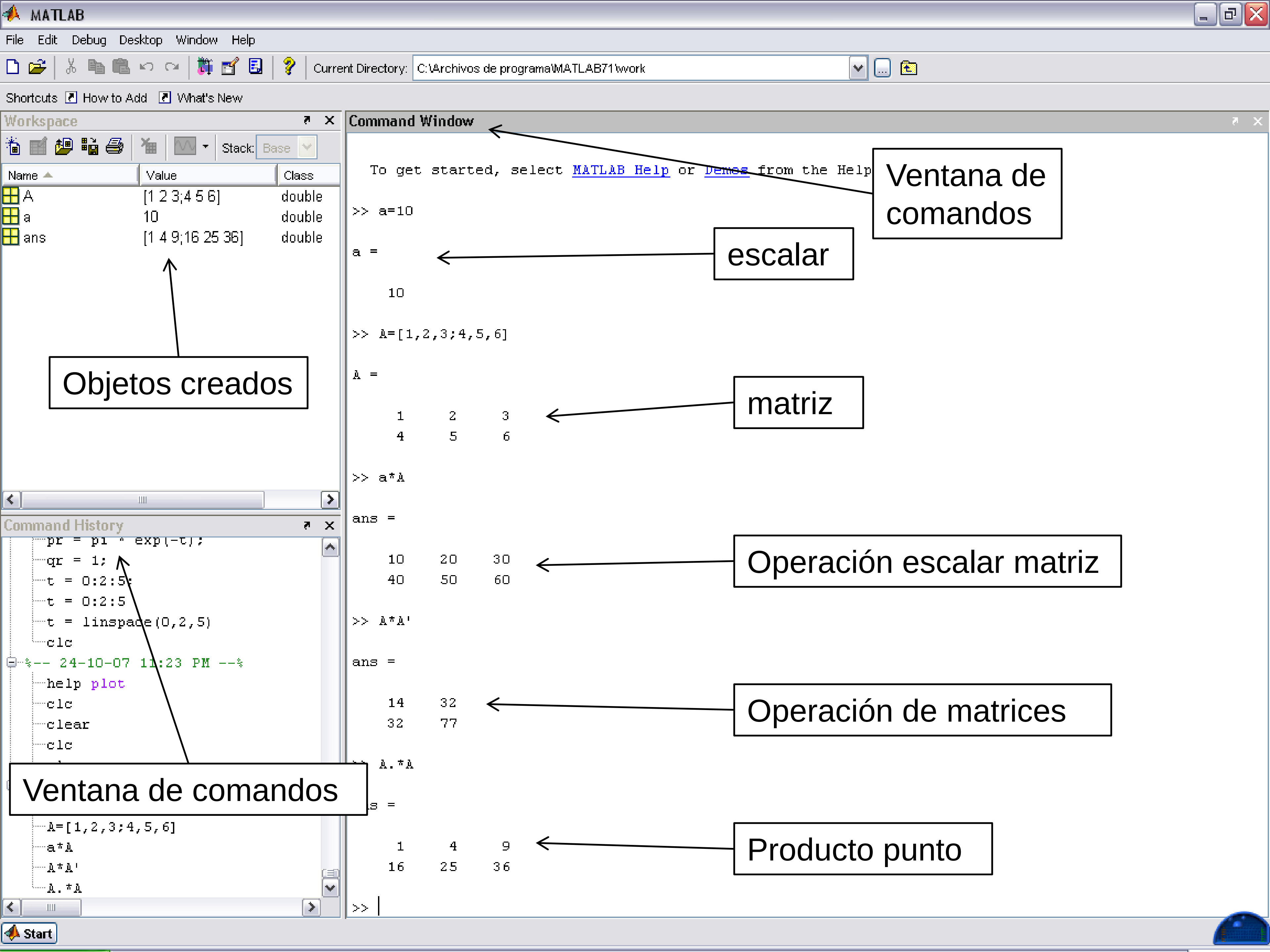
$$k_3 = f(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_2 h)$$

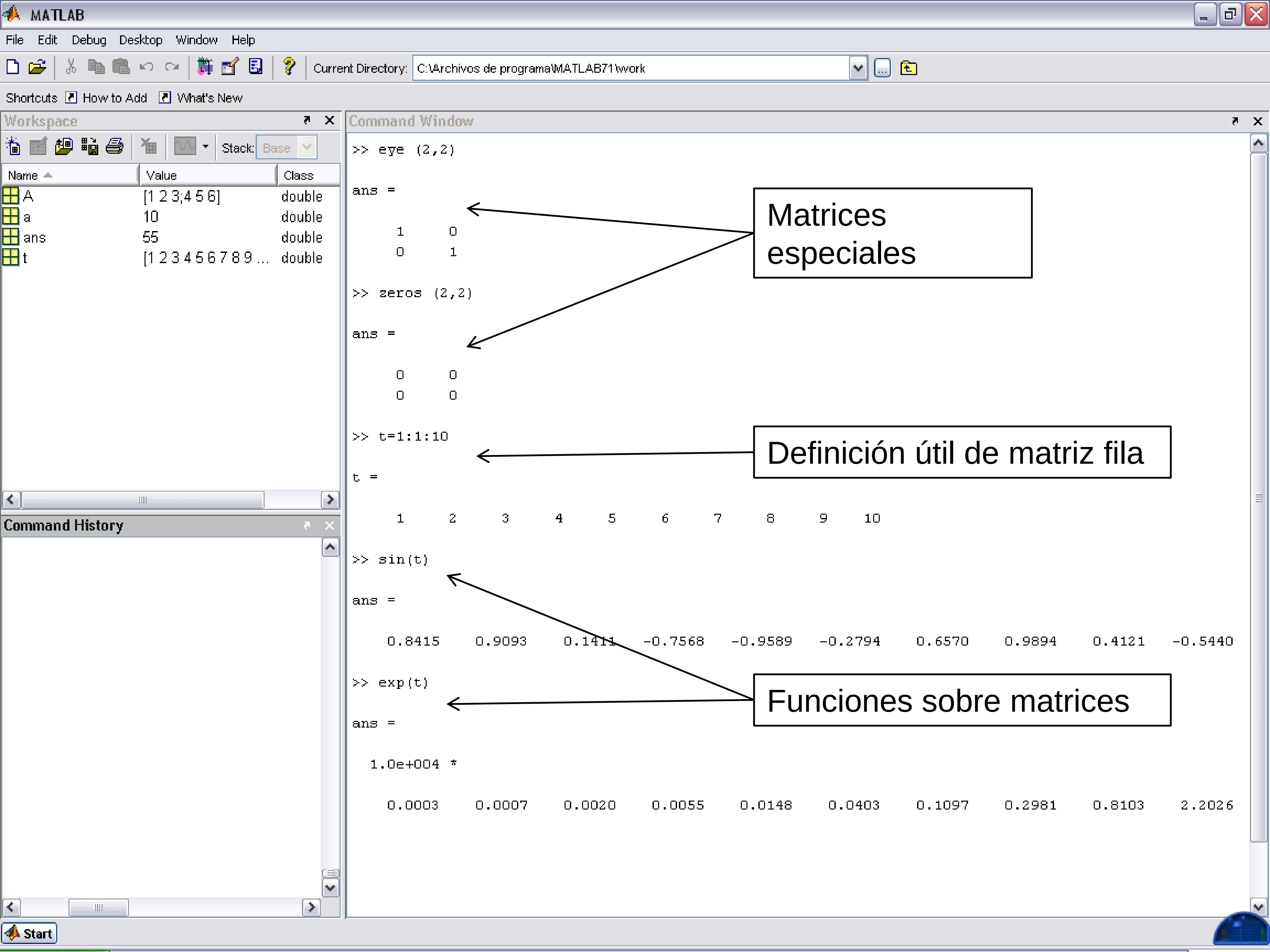
$$k_4 = f(x_n + h, y_n + k_3 h)$$

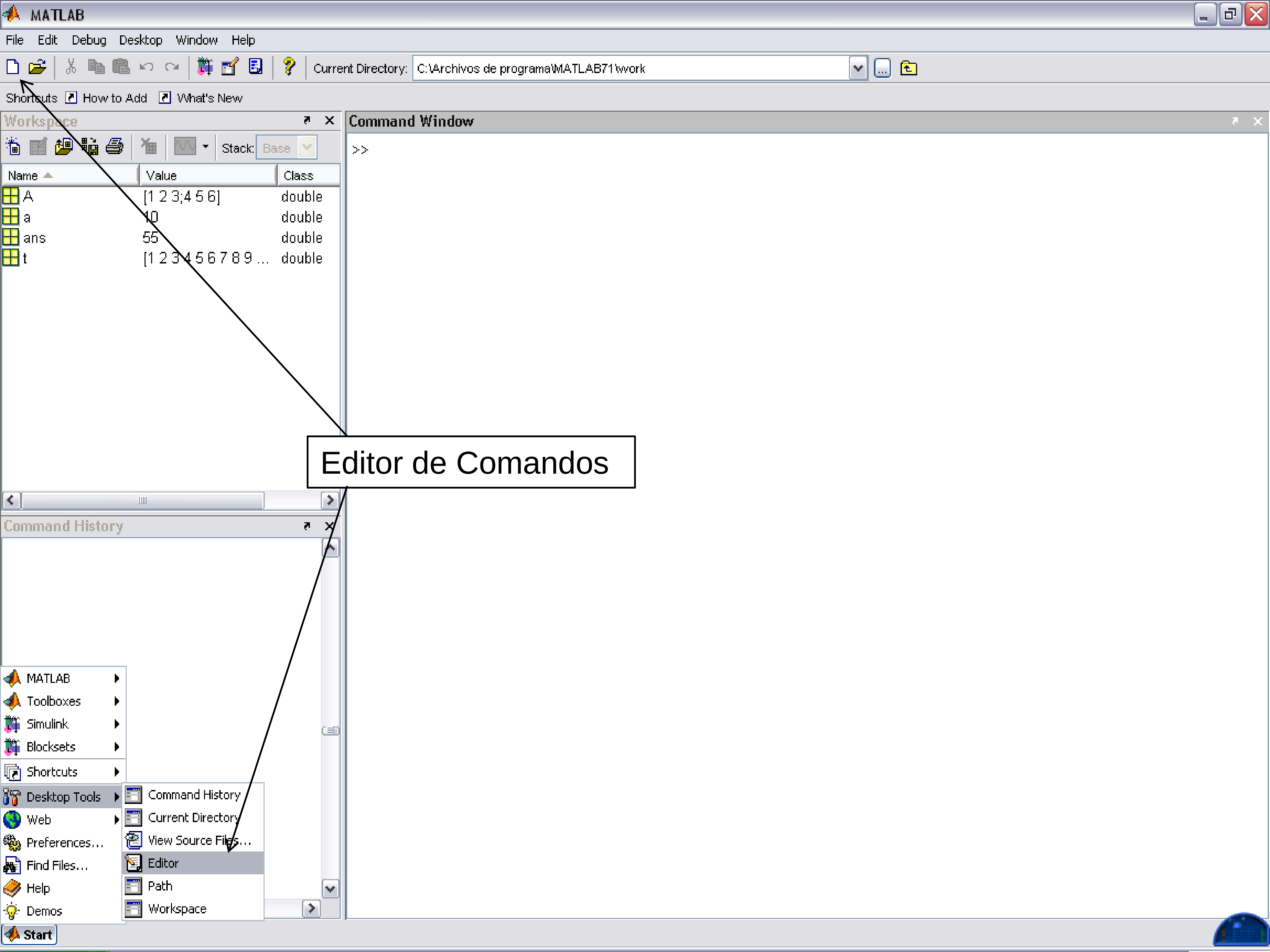


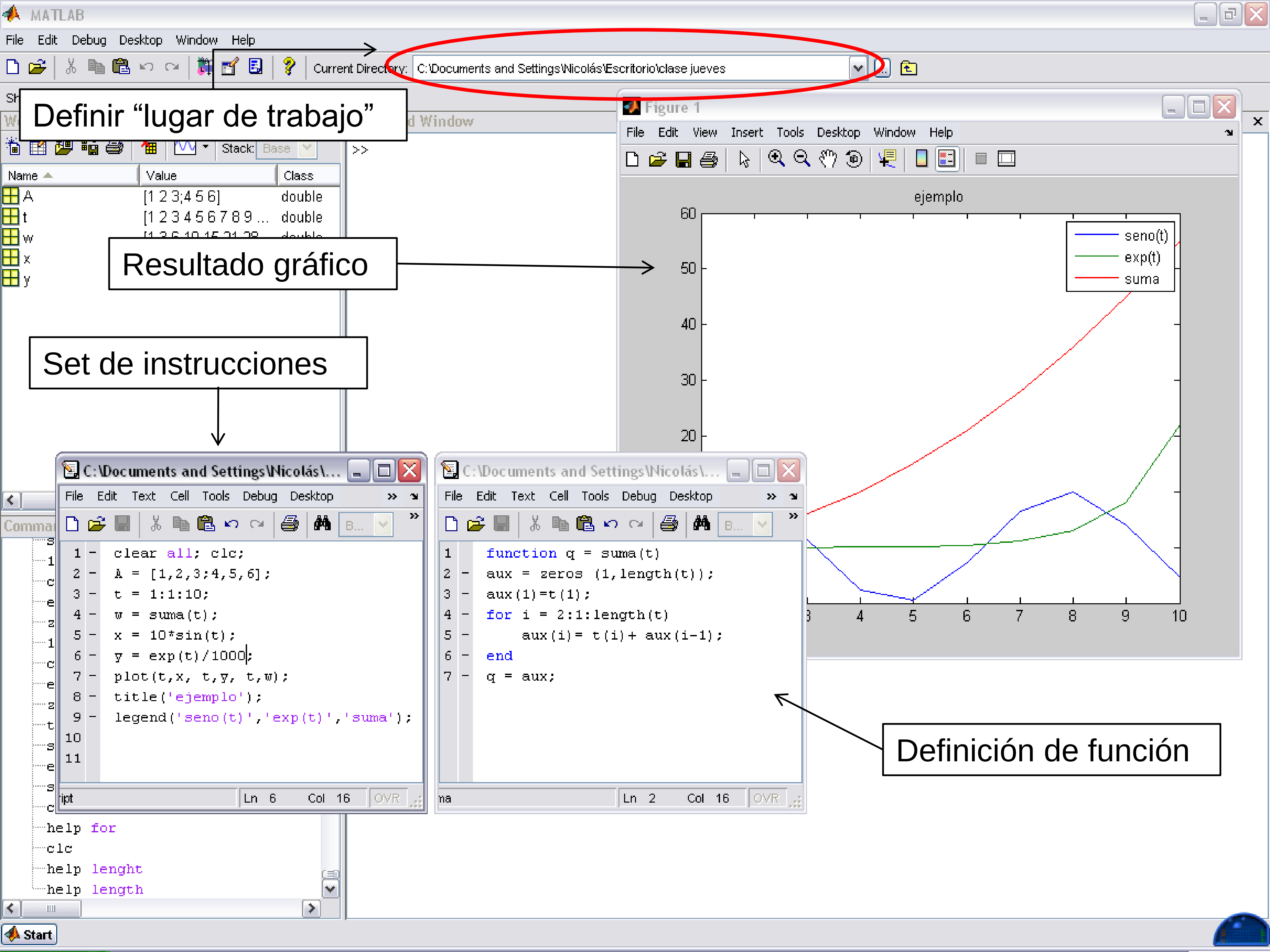
Ahora…

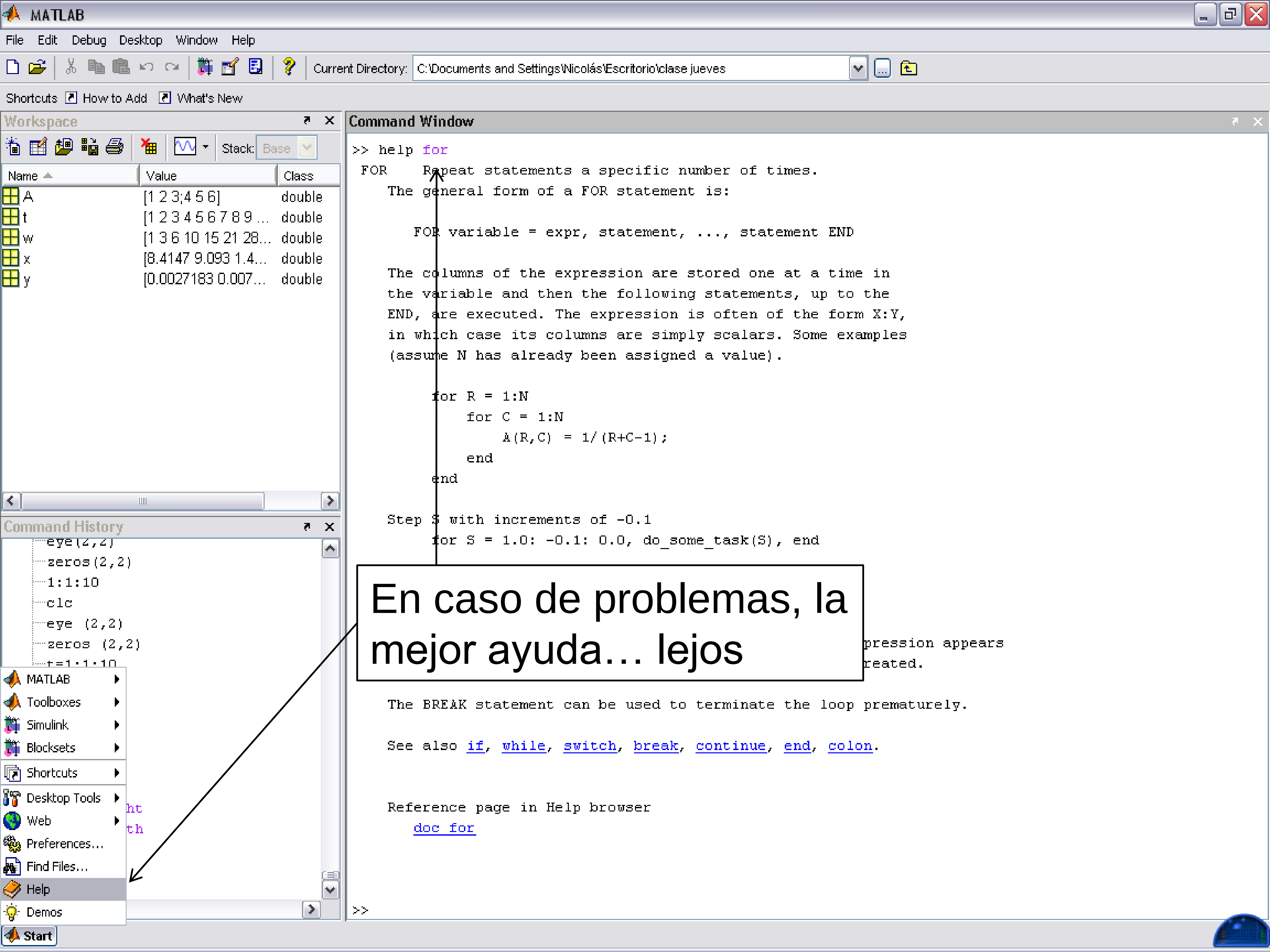
MATLAB


MATLAB
File Edit Debug Desktop Window Help
Current Directory: C:\Archivos de programa\MATLAB71\work
Shortcuts How to Add What's New
Workspace
Stack: Base
Name Value Class
A [1 2 3;4 5 6] double
a 10 double
ans [1 4 9;16 25 36] double
Objetos creados
Command History
qr = 1;
t = 0:2:5;
t = 0:2:5
t = linspace(0,2,5)
clc
%-- 24-10-07 11:23 PM --%
help plot
clc
clear
clc
Ventana de comandos
A=[1,2,3;4,5,6]
a*A
A*A'
A.*A
Command Window
Ventana de comandos
To get started, select MATLAB Help or Demos from the Help
>> a=10
a =
10
escalar
>> A=[1,2,3;4,5,6]
A =
1 2 3
4 5 6
matriz
>> a*A
ans =
10 20 30
40 50 60
Operación escalar matriz
>> A*A'
ans =
14 32
32 77
Operación de matrices
A.*A
1 4 9
16 25 36
Producto punto
>>
Start


MATLAB
File Edit Debug Desktop Window Help
Current Directory: C:\Archivos de programa\MATLAB71\work
Shortcuts How to Add What's New
Workspace
Stack: Base
Name Value Class
A [1 2 3;4 5 6] double
a 10 double
ans 55 double
t [1 2 3 4 5 6 7 8 9 ... double
Command History
Command Window
>> eye (2,2)
ans =
1 0
0 1
>> zeros (2,2)
ans =
0 0
0 0
>> t=1:1:10
t =
1 2 3 4 5 6 7 8 9 10
>> sin(t)
ans =
0.8415 0.9093 0.1411 -0.7568 -0.9589 -0.2794 0.6570 0.9894 0.4121 -0.5440
>> exp(t)
ans =
1.0e+004 *
0.0003 0.0007 0.0020 0.0055 0.0148 0.0403 0.1097 0.2981 0.8103 2.2026
Matrices especiales
Definición útil de matriz fila
Funciones sobre matrices
Start


MATLAB
File Edit Debug Desktop Window Help
Current Directory: C:\Archivos de programa\MATLAB71\work
Shortcuts How to Add What's New
Workspace
Stack: Base
Name Value Class
A [1 2 3;4 5 6] double
a 10 double
ans 55 double
t [1 2 3 4 5 6 7 8 9 ... double
Command Window
>>
Editor de Comandos
Command History
MATLAB
Toolboxes
Simulink
Blocksets
Shortcuts
Desktop Tools
Web
Preferences...
Find Files...
Help
Demos
Command History
Current Directory
View Source Files...
Editor
Path
Workspace
Start


MATLAB
File Edit Debug Desktop Window Help
Current Directory: C:\Documents and Settings\Nicolás\Escritorio\clase jueves
Definir “lugar de trabajo”
Stack: Base
Name Value Class
A [1 2 3;4 5 6] double
t [1 2 3 4 5 6 7 8 9 ... double
w
x
y
Resultado gráfico
Set de instrucciones
Figure 1
File Edit View Insert Tools Desktop Window Help
ejemplo
seno(t)
exp(t)
suma
60
50
40
30
20
4 5 6 7 8 9 10
C:\Documents and Settings\Nicolás\...
File Edit Text Cell Tools Debug Desktop
clear all; clc;
A = [1,2,3;4,5,6];
t = 1:1:10;
w = suma(t);
x = 10*sin(t);
y = exp(t)/1000;
plot(t,x, t,y, t,w);
title('ejemplo');
legend('seno(t)','exp(t)','suma');
Ln 6 Col 16 OVR
C:\Documents and Settings\Nicolás\...
File Edit Text Cell Tools Debug Desktop
function q = suma(t)
aux = zeros (1,length(t));
aux(1)=t(1);
for i = 2:1:length(t)
aux(i)= t(i)+ aux(i-1);
end
q = aux;
Ln 2 Col 16 OVR
Definición de función
help for
clc
help lenght
help length
Start



MATLAB
File Edit Debug Desktop Window Help
Current Directory: C:\Documents and Settings\Nicolás\Escritorio\clase jueves
Shortcuts How to Add What's New
Workspace
Stack: Base
Name Value Class
A [1 2 3;4 5 6] double
t [1 2 3 4 5 6 7 8 9 ... double
w [1 3 6 10 15 21 28... double
x [8.4147 9.093 1.4... double
y [0.0027183 0.007... double
Command History
zeros(2,2)
1:1:10
clc
eye (2,2)
zeros (2,2)
MATLAB
Toolboxes
Simulink
Blocksets
Shortcuts
Desktop Tools
Web
Preferences...
Find Files...
Help
Demos
Start
Command Window
>> help for
FOR Repeat statements a specific number of times.
The general form of a FOR statement is:
FOR variable = expr, statement, ..., statement END
The columns of the expression are stored one at a time in
the variable and then the following statements, up to the
END, are executed. The expression is often of the form X:Y,
in which case its columns are simply scalars. Some examples
(assume N has already been assigned a value).
for R = 1:N
for C = 1:N
A(R,C) = 1/(R+C-1);
end
end
Step S with increments of -0.1
for S = 1.0: -0.1: 0.0, do_some_task(S), end
pression appears
reated.
The BREAK statement can be used to terminate the loop prematurely.
See also if, while, switch, break, continue, end, colon.
Reference page in Help browser
doc for
>>
En caso de problemas, la mejor ayuda… lejos

2. Ejemplo de Michaelis - Menten

Modelo Cinético:

$$E + S \xrightarrow{K1} ES$$

$$ES \xrightarrow{K2} E + S$$

$$ES \xrightarrow{K3} E + P$$

$$\frac{d[S]}{dt} = -K_1 \cdot [S][E] + K_2 \cdot [ES]$$

$$\frac{d[ES]}{dt} = K_1 \cdot [S][E] - K_2 \cdot [ES] - K_3 \cdot [ES]$$

$$\frac{d[E]}{dt} = -K_1 \cdot [S][E] + K_2 \cdot [ES] + K_3 \cdot [ES]$$

$$\frac{d[P]}{dt} = K_3 \cdot [ES]$$

Cuasi Estacionalidad:

$$\frac{d[ES]}{dt} = 0$$

$$[Eo] = [E] + [ES]$$

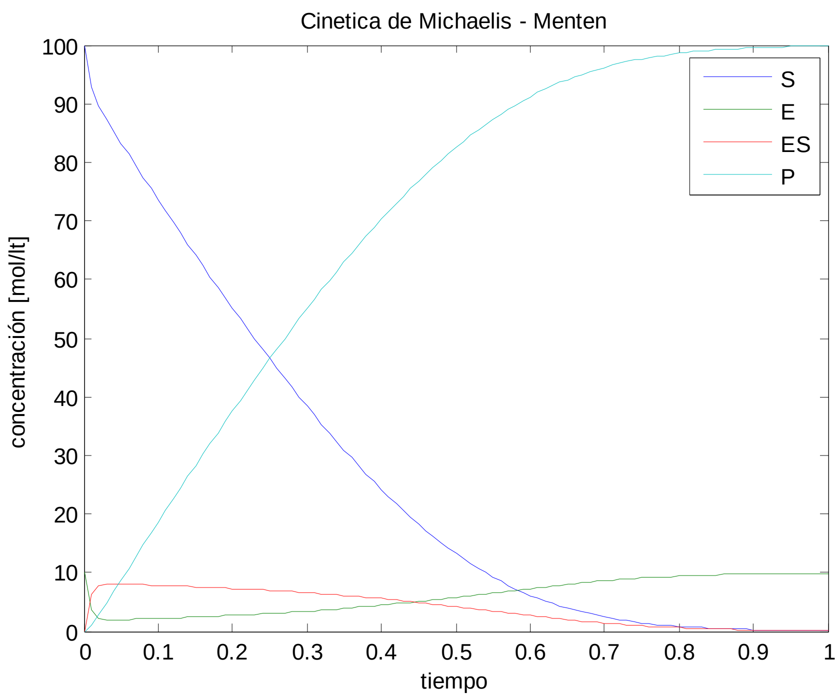
$$\frac{d[P]}{dt} = -\frac{d[S]}{dt} = K_3[E_0] \cdot \frac{[S]}{K_m + [S]}$$

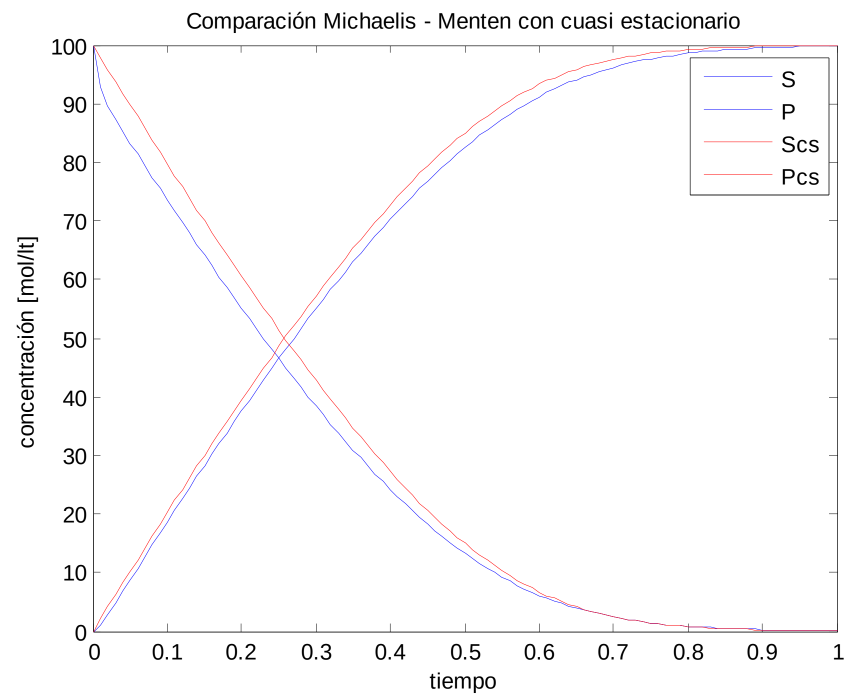
$$K_m = \frac{K_3 + K_2}{K_1}$$

2. Ejemplo de Michaelis - Menten

Resultados:

Modelo Cinético:





K1 = K2 = 0.005; K3 = 0.1

Ahora…

Ecuaciones en Derivadas Parciales

3. Resolución de EDPs

Expresión analítica

Las PDE de segundo orden pueden escribirse de forma general (para dependencia de 2 variables):

$$a\frac{\partial^2 u}{\partial x^2}+b\frac{\partial^2 u}{\partial x\partial y}+c\frac{\partial^2 u}{\partial y^2}+d\frac{\partial u}{\partial x}+e\frac{\partial u}{\partial y}+f\cdot u+g=0$$

En que:
x e *y* son las variables independientes.

u es la variable dependiente.

Si *a*, *b*, *c*, etc. no son funciones de *u*, se dice que la ecuación es lineal.

Complementariamente, las PDE se clasifican según la relación entre sus coeficientes. Si:

$b^2 - 4ac < 0$ → elíptica
$b^2 - 4ac = 0$ → parabólica
$b^2 - 4ac > 0$ → hiperbólica

3. Resolución de EDPs

Para resolver numéricamente este tipo de problemas existen 3 grandes métodos a saber:
- Diferencias Finitas
- Volúmenes Finitos
- Elementos Finitos

Se analiza los métodos numéricos más comunes para resolver PDE mediante el MDF. En particular, se hará mediante la ecuación de transmisión de calor (o de Fick) simplificada.

$$\boxed{\nabla^2 u = \frac{1}{k}\frac{\partial u}{\partial t}}$$

Junto a la ecuación, la interpretación física de una PDE va acompañada con condiciones adicionales. Para la ecuación de calor hay 2 tipos de condiciones:

1.- Condición inicial:

$$u(\vec{x},0) = u_0(\vec{x}) \quad \forall \vec{x} \in \text{Dominio}$$

2.- Condiciones de borde:
2.a.- Condiciones tipo Dirichlet:
Consiste en imponer

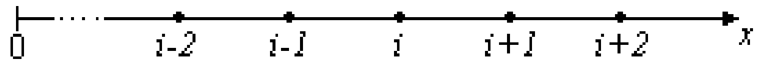
$$u(\vec{x},t) = \varphi_0(\vec{x}) \quad \forall t \in [0,T], \quad \forall \vec{x} \in \text{Borde}$$

2.b.- Condiciones tipo Newmann: “Flujo de calor en el borde del dominio es conocido”

$$\frac{\partial u}{\partial \hat{n}}(\vec{x},t) = \psi_0(\vec{x}) \quad \forall t \in [0,T], \quad \forall \vec{x} \in \text{Borde}$$

3. Resolución de EDPs

MDF se basa en la expansión en series de Taylor aproximando la derivada parcial por cociente donde se realiza una discretización del dominio obteniendo una malla de trabajo:

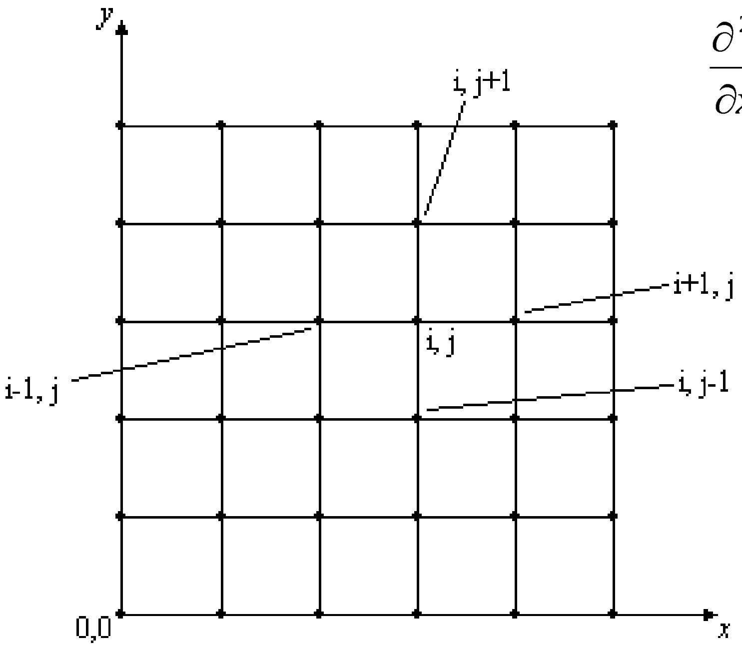


	Primer orden	Error	Segundo orden	Error
Dif. hacia delante	$\frac{\partial u}{\partial x} = \frac{u_{i+1} - u_i}{\Delta x}$	$0(\Delta x)$	$\frac{\partial^2 u}{\partial x^2} = \frac{u_{i+2} - 2u_{i+1} + u_i}{\Delta x^2}$	$0(\Delta x^2)$
Dif. hacia atrás	$\frac{\partial u}{\partial x} = \frac{u_i - u_{i-1}}{\Delta x}$	$0(\Delta x)$	$\frac{\partial^2 u}{\partial x^2} = \frac{u_i - 2u_{i-1} + u_{i-2}}{\Delta x^2}$	$0(\Delta x^2)$
Dif. central	$\frac{\partial u}{\partial x} = \frac{u_{i+1} - u_{i-1}}{2\Delta x}$	$0(\Delta x^2)$	$\frac{\partial^2 u}{\partial x^2} = \frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2}$	$0(\Delta x^2)$

3. Resolución de EDPs: elíptica

El caso estacionario de transmisión de calor para una placa cuadrada (2 dimensiones) es:

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0 \qquad \textit{ecuación de Laplace}$$



$$\frac{\partial^2 T}{\partial x^2} = \frac{T_{i+1,j} - 2T_{i,j} + T_{i-1,j}}{\Delta x^2} \qquad \frac{\partial^2 T}{\partial y^2} = \frac{T_{i,j+1} - 2T_{i,j} + T_{i,j-1}}{\Delta y^2}$$

Reemplazando:

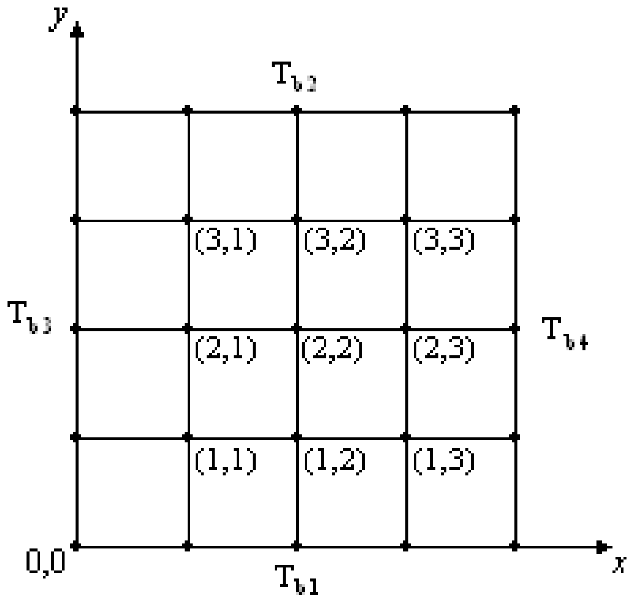
$$\frac{T_{i+1,j} - 2T_{i,j} + T_{i-1,j}}{\Delta x^2} + \frac{T_{i,j+1} - 2T_{i,j} + T_{i,j-1}}{\Delta y^2} = 0$$

Considerando una malla en que $\Delta x = \Delta y$:

$$T_{i+1,j} + T_{i-1,j} + T_{i,j+1} + T_{i,j-1} - 4T_{i,j} = 0$$

3. Resolución de EDPs: elíptica

C.B. tipo Dirichlet:

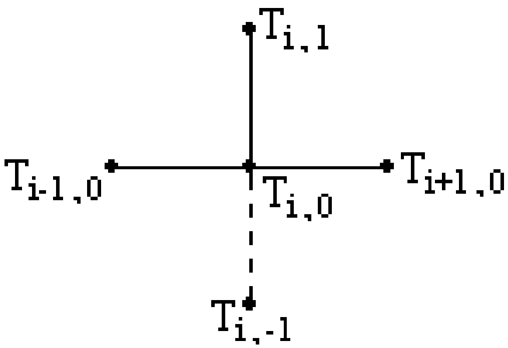


A·T = b, y la solución estará dada por **T = A^{-1}·b**

$$
\underset{A}{\begin{bmatrix}
-4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
1 & -4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 1 & -4 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 0 & -4 & 1 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 1 & -4 & 1 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 1 & -4 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & -4 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 1 & -4 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & -4
\end{bmatrix}}
\quad
\underset{T}{\begin{vmatrix}
T_{11} \\ T_{21} \\ T_{31} \\ T_{12} \\ T_{22} \\ T_{32} \\ T_{13} \\ T_{23} \\ T_{33}
\end{vmatrix}}
\quad
\underset{b}{\begin{vmatrix}
-T_{b1} - T_{b3} \\ -T_{b1} \\ -T_{b1} - T_{b4} \\ -T_{b3} \\ 0 \\ -T_{b4} \\ -T_{b2} - T_{b3} \\ -T_{b2} \\ -T_{b2} - T_{b4}
\end{vmatrix}}
$$

3. Resolución de EDPs: elíptica

C.B. tipo Newmann:



$$T_{i+1,0} + T_{i-1,0} + T_{i,1} + T_{i,-1} - 4T_{i,0} = 0$$

Aproximando la derivada en la dimensión *y* mediante diferencia finita central:

$$\frac{\partial T}{\partial y} \cong \frac{T_{i,1} - T_{i,-1}}{2\Delta y} \quad \Rightarrow \quad T_{i,-1} = T_{i,1} - 2\Delta y \frac{\partial T}{\partial y}$$

Reemplazando:

$$T_{i+1,0} + T_{i-1,0} + 2T_{i,1} - 2\Delta y \frac{\partial T}{\partial y} - 4T_{i,0} = 0$$

3. Resolución de EDPs: parabólica

La ecuación de calor unidimensional (a través de una barra delgada) es:

$$k\,\frac{\partial^2 T}{\partial x^2} = \frac{\partial T}{\partial t}$$

- Método explícito

- Método implícito

3. Resolución de EDPs: parabólica

- Método explícito:

Para éste caso se utiliza una diferencia finita central para estimar la derivada espacial y una diferencia finita hacia adelante para aproximar la derivada en el tiempo.

$$\frac{\partial^2 T}{\partial x^2} = \frac{T_{i+1}^{l} - 2T_{i}^{l} + T_{i-1}^{l}}{\Delta x^2} \qquad \frac{\partial T}{\partial t} = \frac{T_{i}^{l+1} - T_{i}^{l}}{\Delta t}$$

Sustituyendo:

$$k\frac{T_{i+1}^{l} - 2T_{i}^{l} + T_{i-1}^{l}}{\Delta x^2} = \frac{T_{i}^{l+1} - T_{i}^{l}}{\Delta t} \quad \Longrightarrow \quad T_{i}^{l+1} = T_{i}^{l} + \alpha \cdot (T_{i+1}^{l} - 2T_{i}^{l} + T_{i-1}^{l})$$

Con: $\alpha = k\Delta t/(\Delta x)^2$

Criterio de convergencia: $\alpha \le 1/2 \quad \rightarrow \Delta t \le \dfrac{(\Delta x)^2}{2k}$

3. Resolución de EDPs: parabólica

- Método implícito:

Si bien en el método explicito es bastante simple, existen problemas de inestabilidad de la solución. Este problema se soluciona implementando un método implícito, en el que la derivada espacial se aproxima mediante una diferencia finita central, pero en un tiempo *l+1*:

$$\frac{\partial^2 T}{\partial x^2} = \frac{T_{i+1}^{l+1} - 2T_i^{l+1} + T_{i-1}^{l+1}}{\Delta x^2} \qquad \frac{\partial T}{\partial t} = \frac{T_i^{l+1} - T_i^{l}}{\Delta t}$$

Sustituyendo:

$$k\frac{T_{i+1}^{l+1} - 2T_i^{l+1} + T_{i-1}^{l+1}}{\Delta x^2} = \frac{T_i^{l+1} - T_i^{l}}{\Delta t} \Longrightarrow -\alpha T_{i-1}^{l+1} + (1+2\alpha)T_i^{l+1} - \alpha T_{i+1}^{l+1} = T_i^{l}$$

3. Resolución de EDPs: parabólica

Ambos casos pueden escribirse de manera matricial

$$\begin{bmatrix} T_1^{l+1} \\ \vdots \\ T_n^{l+1} \\ \vdots \\ T_{Nx}^{l+1} \end{bmatrix} = \begin{bmatrix} 1-2\alpha & \alpha & & \\ \alpha & 1-2\alpha & \ddots & \\ & \alpha & \ddots & \\ & & \ddots & \\ & & & \end{bmatrix} \cdot \begin{bmatrix} T_1^{l} \\ \vdots \\ T_n^{l} \\ \vdots \\ T_{Nx}^{l} \end{bmatrix} + \begin{bmatrix} \alpha \cdot f_0(t^l) \\ \vdots \\ 0 \\ \vdots \\ \alpha \cdot f_{m+1}(t^l) \end{bmatrix}$$

Método explícito

Método implícito

$$\begin{bmatrix} 1+2\alpha & -\alpha & & & \\ -\alpha & 1+2\alpha & -\alpha & & \\ & -\alpha & \ddots & \ddots & \\ & & \ddots & & \\ & & & & \end{bmatrix} \cdot \begin{bmatrix} T_1^{l+1} \\ \vdots \\ T_n^{l+1} \\ \vdots \\ T_{Nx}^{l+1} \end{bmatrix} = \begin{bmatrix} T_1^{l} \\ \vdots \\ T_n^{l} \\ \vdots \\ T_{Nx}^{l} \end{bmatrix} + \begin{bmatrix} f_0(t^{l+1}) \\ \vdots \\ 0 \\ \vdots \\ f_{m+1}(t^{l+1}) \end{bmatrix}$$

3. Resolución de EDPs: parabólica en 2D

Finalmente se analiza el caso en que se tiene transmisión de calor en estado transiente sobre una placa.

$$k\left(\frac{\partial^2 T}{\partial x^2}+\frac{\partial^2 T}{\partial y^2}\right)=\frac{\partial T}{\partial t}$$

Para la resolución numérica de este problema se utiliza el *método de las líneas*.
En este se aproxima el problema PDE a un sistema de EDOs *lineales*, por eso su nombre.

$$\frac{\partial T_{i,j}}{\partial t}=k\left(\frac{T_{i+1,j}-2T_{i,j}+T_{i-1,j}}{\Delta x^2}+\frac{T_{i,j+1}-2T_{i,j}+T_{i,j-1}}{\Delta y^2}\right)$$

La ecuación anterior se realiza para una malla. Considerando $\Delta x = \Delta y$:

$$\frac{\partial T_{i,j}}{\partial t}=\frac{k}{\Delta x^2}\left(T_{i+1,j}+T_{i-1,j}+T_{i,j+1}+T_{i,j-1}-4T_{i,j}\right)$$

Criterio de convergencia: $\partial t \leq \frac{(\Delta x)^2}{4k}$

Ahora…

MATLAB… otra vez

DUDAS?