

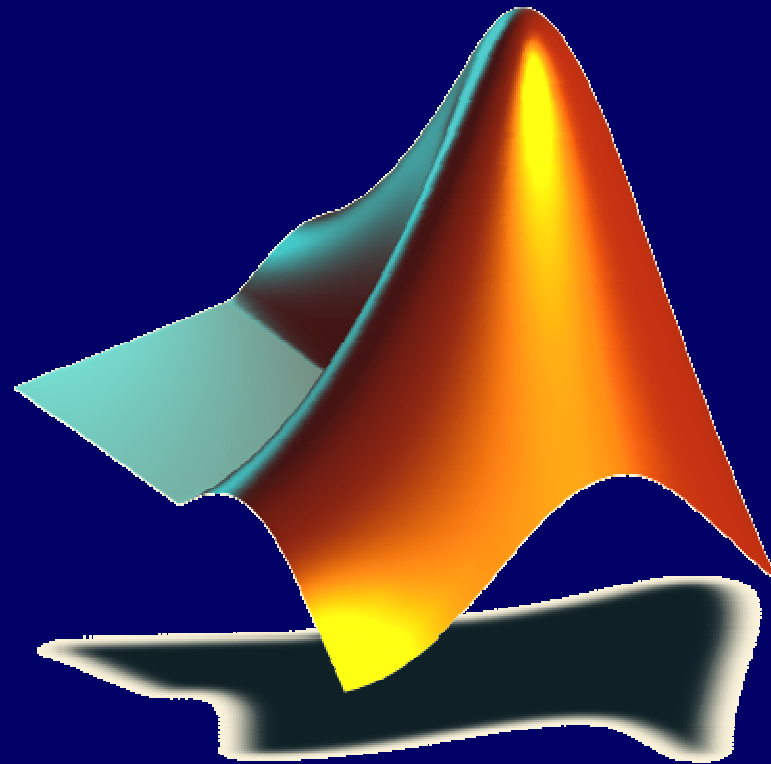
UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO
FACULTAD DE CIENCIAS QUÍMICAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO ACADÉMICO DE MATEMÁTICAS Y ESTADÍSTICA



UNSAAC

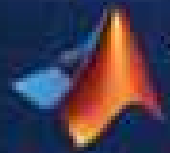
Lic. Guillermo Mario Chuquipoma Pacheco
mariochuqui@hotmail.com
www.mariochuqui.jimdo.com

Introducción al MATLAB



Temario General

1. **Introducción.** ←
2. Interfaz de usuario de MATLAB.
3. Variables del MATLAB.
4. Ploteo y visualización de datos.
5. m-Files.
6. Estadística básica y análisis de datos.
7. Tipos de datos.
8. Entrada y salida de datos.
9. Programación.
10. Construcción de interfaces de usuarios gráficas.



The MathWorks™

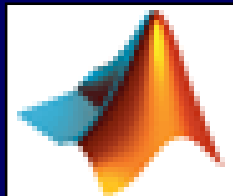
Accelerating the pace of engineering and science



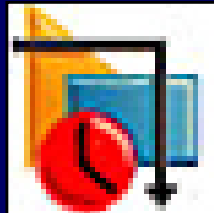
The MathWorks es una compañía que va a la cabeza del mercado para software técnico de computación. Sus productos, como MATLAB y SIMULINK, son usados extensamente en centros de investigación e industria.

Mathworks

- Creado en 1984.
- Es una empresa global con sede en 60 países:
- Ubicada en 3 Apple Hill Drive. Natick, Boston, Massachusetts
- Los principales productos que ofrece son:



MATLAB®



Simulink®

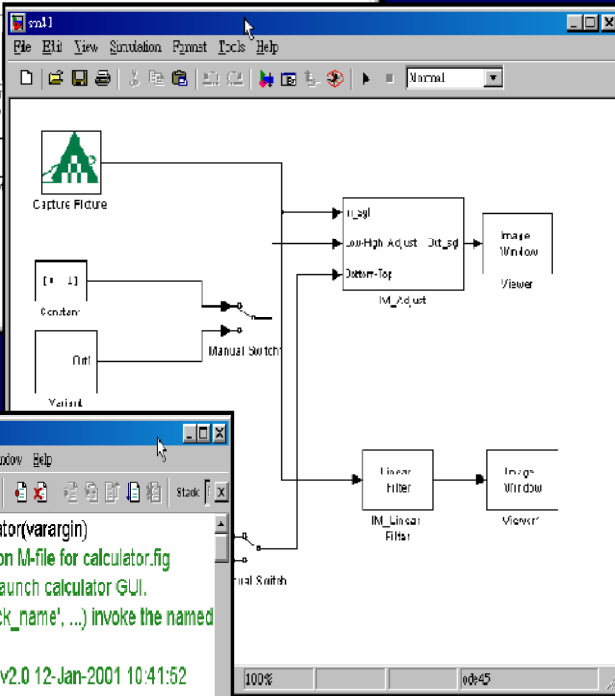
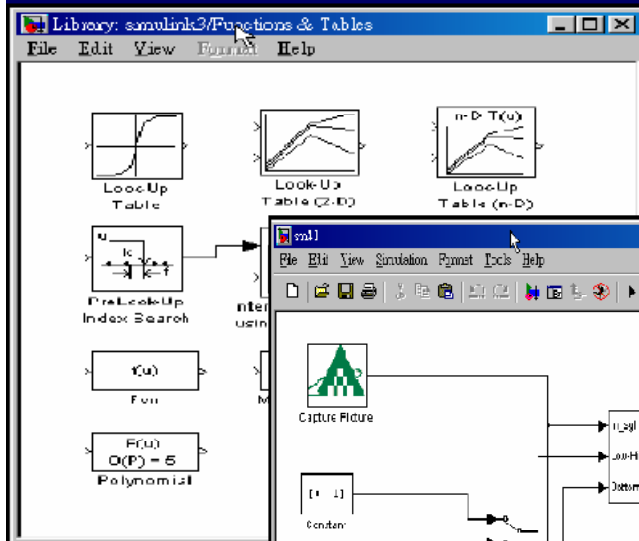


Stateflow®

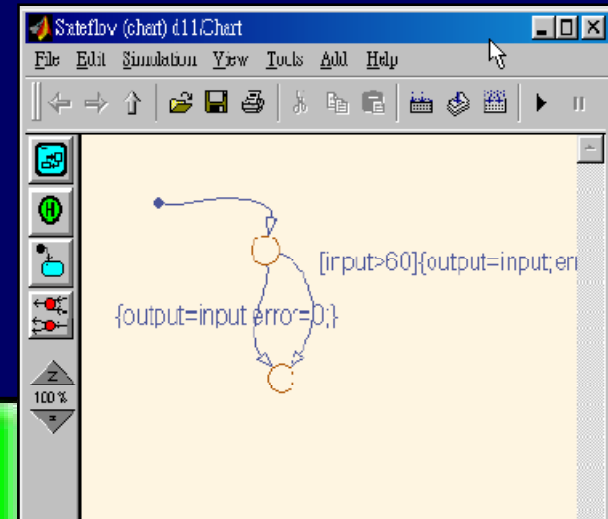
Matlab

- MatLab significa **MAT**rix **LAB**oratory.
- Es un programa para hacer computación numérica.
- Fue diseñado para manipular matrices y ploteo de datos.
- Ahora incluye funciones para: analizar datos, procesar señales, optimizar funciones.
- Contiene funciones para los gráficos 2-D y 3-D.
- Tiene interfaces con otros lenguajes.
- Permite la computación simbólica con el Maple.

La Familia del MatLab



```
function varargout = calculator(varargin)
% CALCULATOR Application M-file for calculator.fig
% FIG = CALCULATOR launch calculator GUI.
% CALCULATOR('callback_name', ...) invoke the named
%
% Last Modified by GUIDE v2.0 12-Jan-2001 10:41:52
%
if nargin == 0 % LAUNCH GUI
    fig = openfig(mfilename,'reuse');
    % Use system color scheme for figure:
    set(fig,'Color',get(0,'defaultUicontrolBackgroundColor'));
    % Generate a structure of handles to pass to callbacks.
    handles = guihandles(fig);
end
```



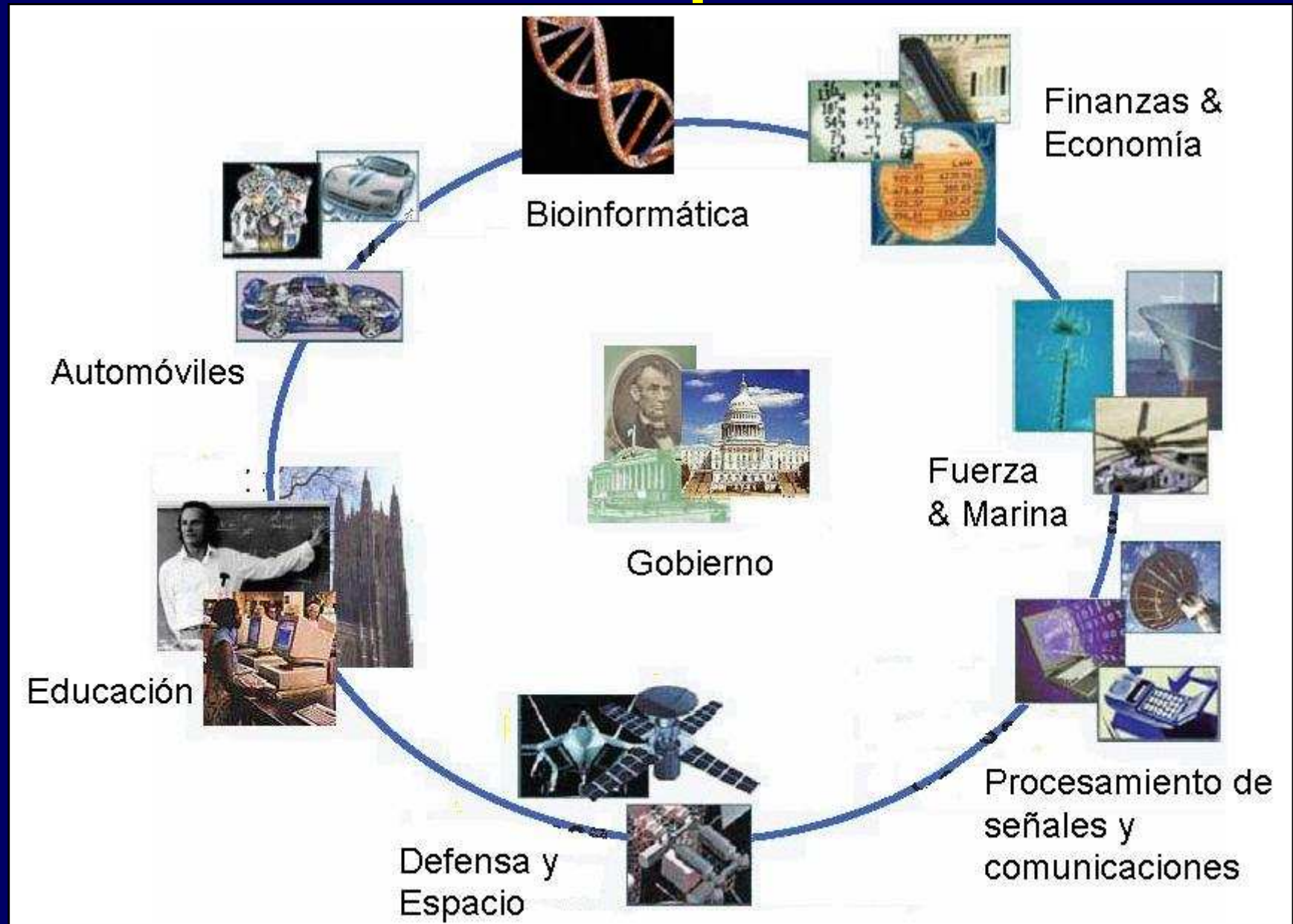
MATLAB

Workspace, Command, Current Directory, All files

C/C++
VHDL
COM+
Java Classes

pathwin.m
readmewin.fig
readmewin.m
rpc_setup1.m
rpc_setup2.m
setup2.m

Áreas de Aplicación



Familia de Productos MatLab®

MATLAB®

MATLAB® Distributed Computing Toolbox™
MATLAB® Distributed Computing Engine™

Math and Optimization

Optimization Toolbox™ Symbolic Math Toolbox™
Extended Symbolic Math Toolbox™
Partial Differential Equation Toolbox™ Genetic Algorithm and Direct Search Toolbox™

Statistics and Data Analysis

Statistics Toolbox™ Neural Network Toolbox™
Curve Fitting Toolbox™ Spline Toolbox™
Model-Based Calibration Toolbox™

Control System Design and Analysis

Control System Toolbox™ System Identification Toolbox™
Fuzzy Logic Toolbox™
Robust Control Toolbox™ Model Predictive Control Toolbox™
Aerospace Toolbox

Signal Processing and Communications

Signal Processing Toolbox™ Communications Toolbox™
Filter Design Toolbox™ Filter Design HDL Coder™
Wavelet Toolbox™ Fixed-Point Toolbox™
RF Toolbox™

Image Processing

Image Processing Toolbox™ Image Acquisition Toolbox™
Mapping Toolbox™

Test & Measurement

Data Acquisition Toolbox™ Instrument Control Toolbox™
Image Acquisition Toolbox™
SystemTest™ OPC Toolbox™

Computational Biology

Bioinformatics Toolbox™ SimBiology®

Financial Modeling and Analysis

Financial Toolbox™ Financial Derivatives Toolbox™
GARCH Toolbox™ Datafeed Toolbox™
Fixed-Income Toolbox™

Application Deployment

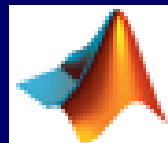
MATLAB® Compiler™ Excel® Link™

Application Deployment Targets

MATLAB® Builder for Excel® MATLAB® Builder for .NET
MATLAB® Builder for Java™

Database Connectivity and Reporting

Database Toolbox™ MATLAB® Report Generator™



Familia de Productos Simulink®

Simulink®

[Simulink® Simulink® Report Generator™](#)

Fixed-Point Modeling

[Simulink® Fixed Point™](#)

Event-Based Modeling

[Stateflow® SimEvents®](#)

Physical Modeling

[Simscape™ SimMechanics™](#)

[SimPowerSystems™ SimDriveline™](#)

[SimHydraulics®](#)

Simulation Graphics

[Virtual Reality Toolbox™ Gauges](#)

[Blockset™](#)

Control System Design and Analysis

[Simulink® Control Design™ Simulink®](#)

[Response Optimization™ Simulink®](#)

[Parameter Estimation™ Aerospace](#)

[Blockset™](#)

Signal Processing and Communications

[Signal Processing Blockset™ Communications Blockset™](#)
[RF Blockset™ Video and Image Processing Blockset™](#)

Code Generation

[Real-Time Workshop® Real-Time Workshop® Embedded](#)
[Coder™ Stateflow® Coder™ Simulink® HDL Coder™](#)

Rapid Control Prototyping and HIL SW/HW

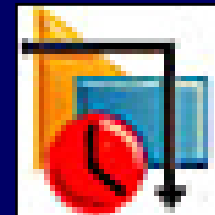
[xPC Target™ xPC Target Embedded Option™ Real-Time](#)
[Windows Target™](#)

Embedded Targets

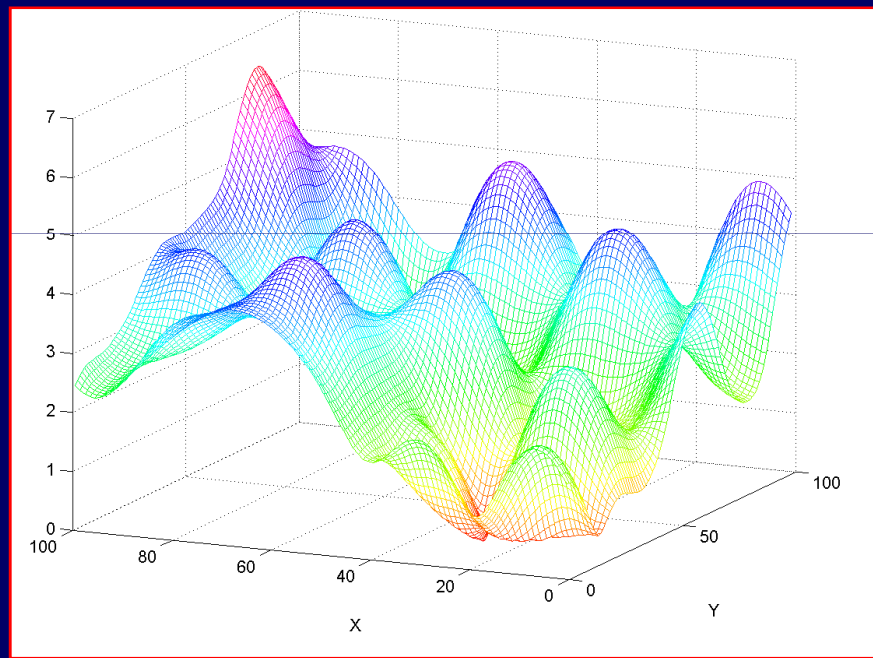
[Target for Infineon C166® Target for Freescale™ MPC5xx](#)
[Target for TI C2000™ Target for TI C6000™](#)

Verification, Validation, and Testing

[Simulink® Verification and Validation™ Simulink® Design](#)
[Verifier™ SystemTest™](#)



Características del MATLAB



Entorno del sistema

- Windows XP/2000
 - MATLAB se instala en c: \MATLAB7
 - El código de usuario ...en algún lugar (por ejemplo h: \matlab)
- Linux (entorno de red)
 - MATLAB se instala en /apps/matlab
 - El código de usuario en /home/username/matlab
 - La configuración del entorno en ~/.matlab

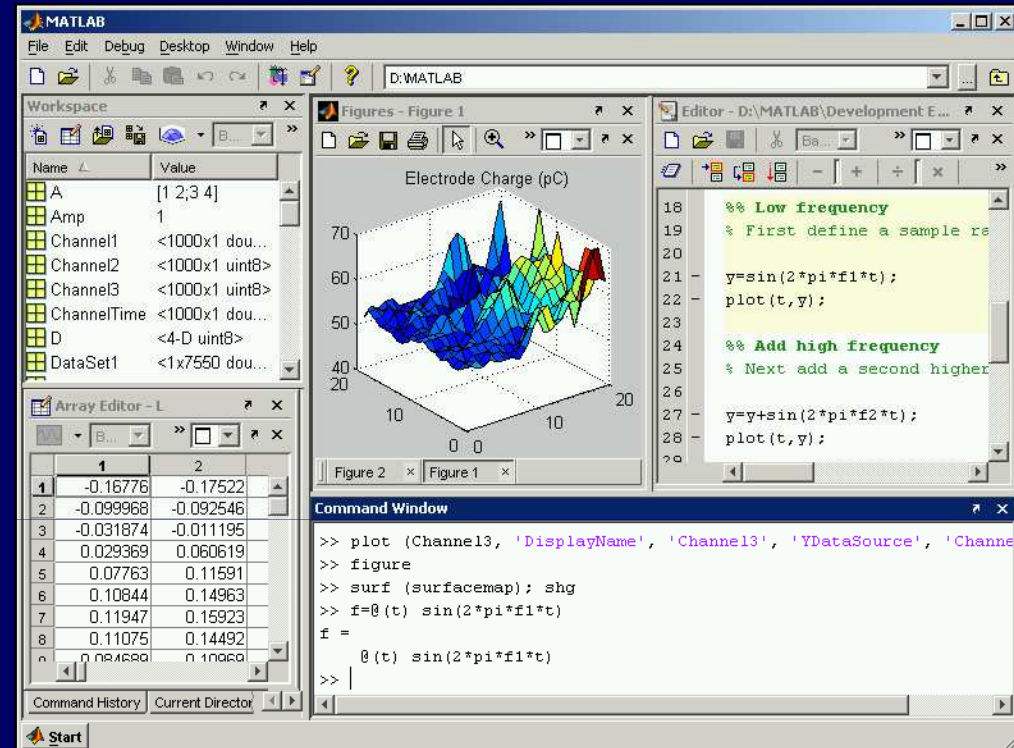
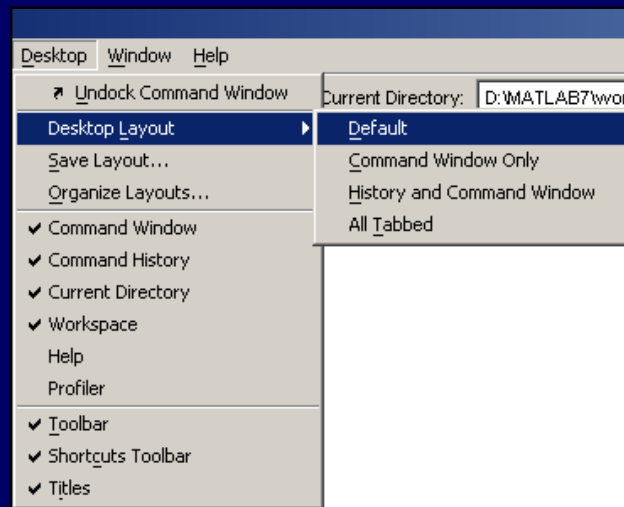
¿Qué es a MATLAB?

- Trabaja números escalares (reales y complejos), con caracteres y otras estructuras de datos.
- Tiene un lenguaje de programación propio.
- Permite un rápido prototipo de aplicaciones científicas.
- Pero puede ser más lento que C/C++ o Fortran.
- Dispone de código básico y toolboxes

1. Entorno de desarrollo a nivel alto
2. Lenguaje de programación
3. Gráficos
4. Cajas de herramientas (toolbox)
5. Línea de comandos.

Es un entorno integrado de software

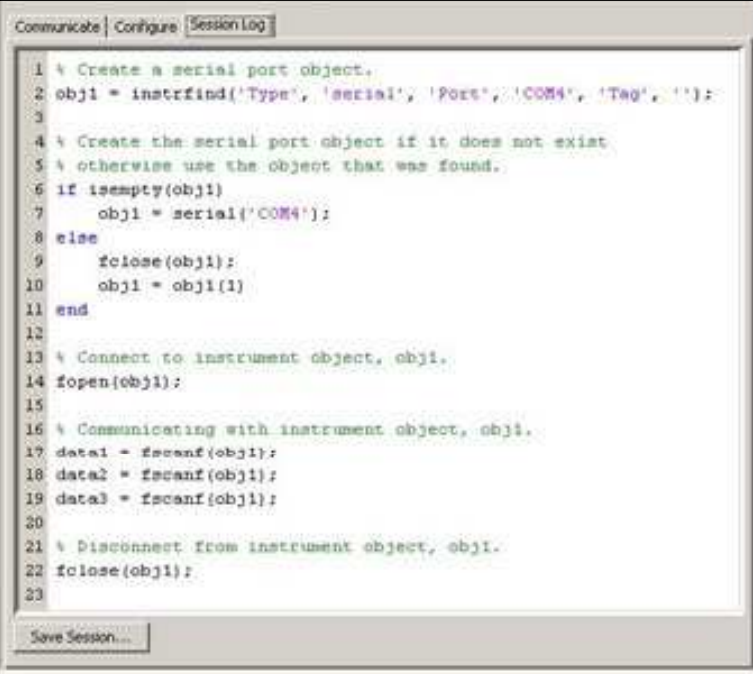
- Command Window
- Command History
- Help Browser
- Workspace Browser
- Editor/Debugger



- Cálculo
- Visualización
- Entorno fácil de usar

Lenguaje de Alto nivel

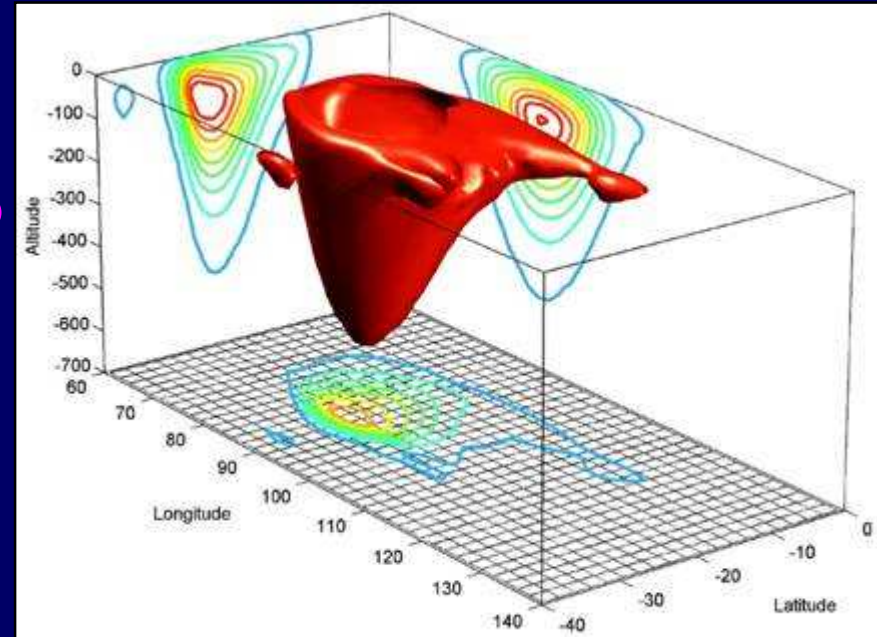
- Tipos de datos
- Funciones
- Declaraciones de flujo de control
- Entrada / Salida
- Gráficos
- Programación orientada a objetos.



```
Communicate | Configure | Session Log
1 % Create a serial port object.
2 obj1 = instrfind('Type', 'serial', 'Port', 'COM4', 'Tag', '1');
3
4 % Create the serial port object if it does not exist
5 % otherwise use the object that was found.
6 if isempty(obj1)
7     obj1 = serial('COM4');
8 else
9     fclose(obj1);
10    obj1 = obj1(1)
11 end
12
13 % Connect to instrument object, obj1.
14 fopen(obj1);
15
16 % Communicating with instrument object, obj1.
17 data1 = fscanf(obj1);
18 data2 = fscanf(obj1);
19 data3 = fscanf(obj1);
20
21 % Disconnect from instrument object, obj1.
22 fclose(obj1);
23
Save Session...
```

Gráficos.

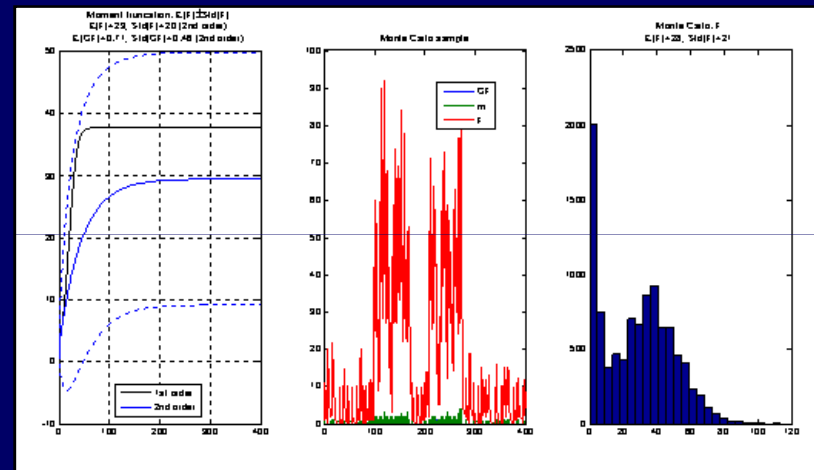
- Gráficos 2D y 3D.
- Gráficos a colores.
- Presentación de datos como imágenes.
- Técnicas volumétricas 3D.
- Iso-superficies (niveles)
- Presentación de volúmenes.



Caja de herramientas.

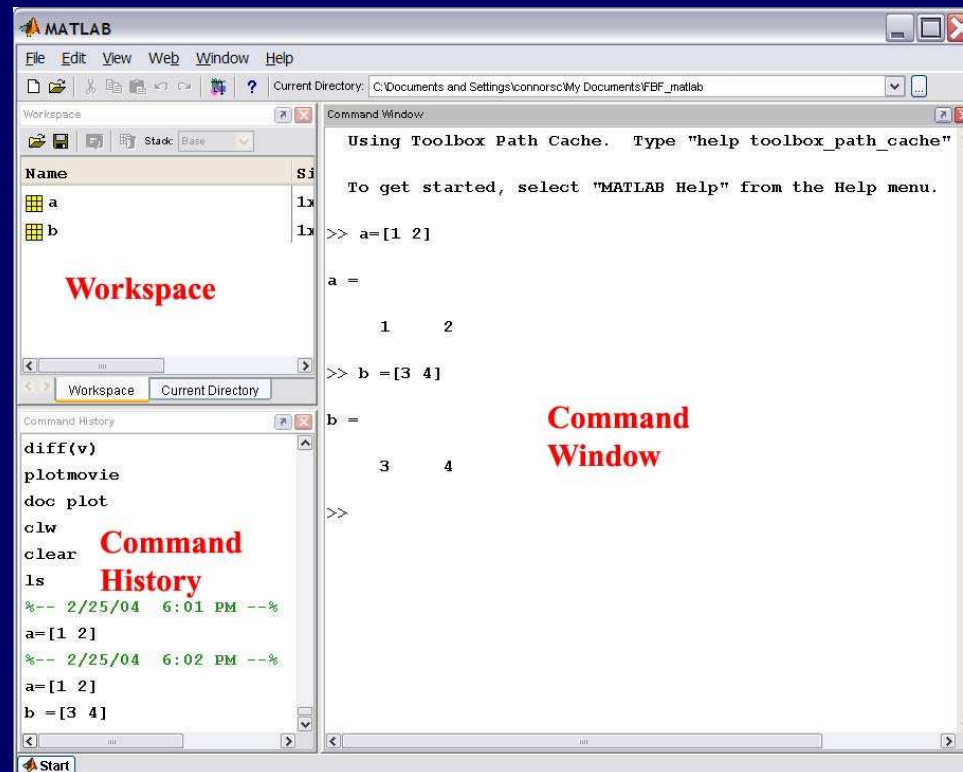
Conjunto de funciones para resolver problemas.

- DSP (Digital Signal Processing) Toolbox
- Image Toolbox
- Wavelet Toolbox
- Neural Network Toolbox
- Fuzzy Logic Toolbox
- Control Toolbox
- Multibody Simulation Toolbox
- Otros



Línea de comandos

- Permite usar el MatLab como una calculadora.
- Permite ingresar órdenes directamente al MatLab.
- Los resultados se presentan en la misma pantalla.
- Permite ejecutar los m-files.



Interfaz de Usuario del MatLab

The screenshot displays the MATLAB user interface with several key components:

- Workspace:** Lists variables and their values. The 'Value' column contains truncated data for several variables.
- Figures - Figure 1:** A 3D surface plot titled 'Electrode Charge (pC)'. The plot shows a complex, multi-peaked surface with a color gradient from blue to red. The axes are labeled with values 20, 40, 50, 60, and 70.
- Editor:** Contains MATLAB code for plotting a signal. The code includes comments for 'Low frequency' and 'Add high frequency', and uses `sin` and `plot` functions.
- Array Editor - L:** A table editor showing a 2x8 grid of numerical data.
- Command Window:** Shows the execution of MATLAB commands, including `plot`, `figure`, `surf`, and `shg`, along with the resulting function definition for `f`.

Name	Value
A	[1 2;3 4]
Amp	1
Channel1	<1000x1 dou...>
Channel2	<1000x1 uint8>
Channel3	<1000x1 uint8>
ChannelTime	<1000x1 dou...>
D	<4-D uint8>
DataSet1	<1x7550 dou...>

```
18 %% Low frequency
19 % First define a sample re
20
21 y=sin(2*pi*f1*t);
22 plot(t,y);
23
24 %% Add high frequency
25 % Next add a second higher
26
27 y=y+sin(2*pi*f2*t);
28 plot(t,y);
29
```

	1	2
1	-0.16776	-0.17522
2	-0.099968	-0.092546
3	-0.031874	-0.011195
4	0.029369	0.060619
5	0.07763	0.11591
6	0.10844	0.14963
7	0.11947	0.15923
8	0.11075	0.14492

```
>> plot (Channel13, 'DisplayName', 'Channel13', 'YDataSource', 'Channe
>> figure
>> surf (surfacemap); shg
>> f=@(t) sin(2*pi*f1*t)
f =
    @(t) sin(2*pi*f1*t)
>>
```

Temario General

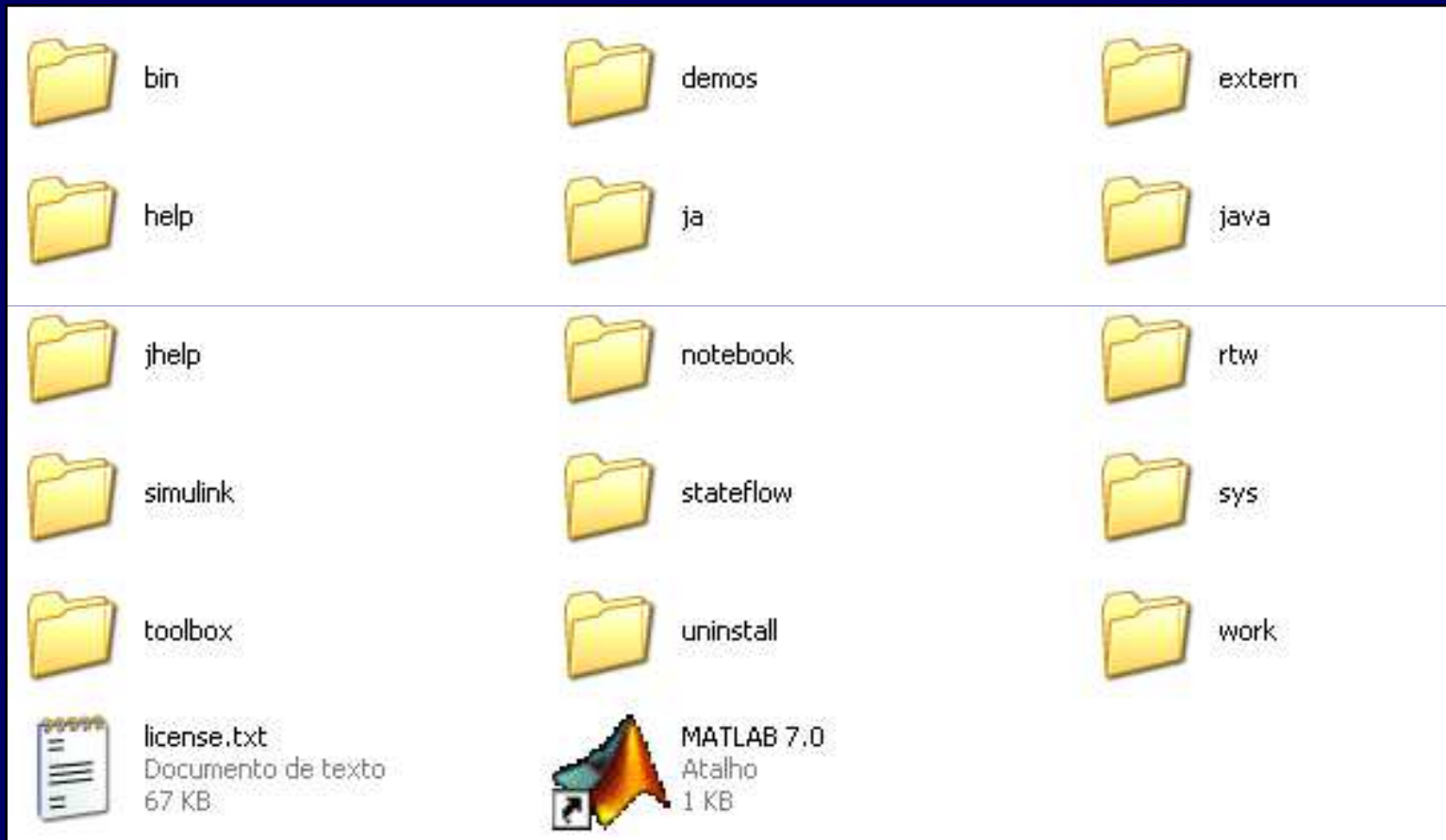
1. Introducción.
- 2. Interfaz de usuario de MATLAB.** ←
3. Variables del MATLAB.
4. Ploteo y visualización de datos.
5. m-Files.
6. Estadística básica y análisis de datos.
7. Tipos de datos.
8. Entrada y salida de datos.
9. Programación.
10. Construcción de interfaces de usuarios gráficas.

Objetivo

- Esta sección introduce las características principales del MATLAB integrado el entorno de diseño e interfaces de usuarios.
- Muchos temas del curso son establecidos en esta sección, para exponerse en detalle en las siguientes secciones.

Ejercicio 1

- Identifique el **path** donde se ubica el matlab en su instalación.



Entorno de usuario MATLAB

The screenshot displays the MATLAB environment with three main components highlighted by white boxes:

- Workspace Browser:** A table showing the current workspace variables.
- Command Window:** A text area for entering and executing MATLAB commands, showing the output of the current session.
- Command History:** A list of previously executed commands for review.

Name	Size	Bytes	Class
x	1x10	80	double array
y	1x10	80	double array

```
< M A T L A B >
Copyright 1984-2002 The MathWorks, Inc.
Version 6.5.0.180913a Release 13
Jun 18 2002

Using Toolbox Path Cache. Type "help toolbox_path_cache" for more info.

To get started, select "MATLAB Help" from the Help menu.

>> x=[1:10]

x =

     1     2     3     4     5     6     7     8     9    10

>> y=x.^2

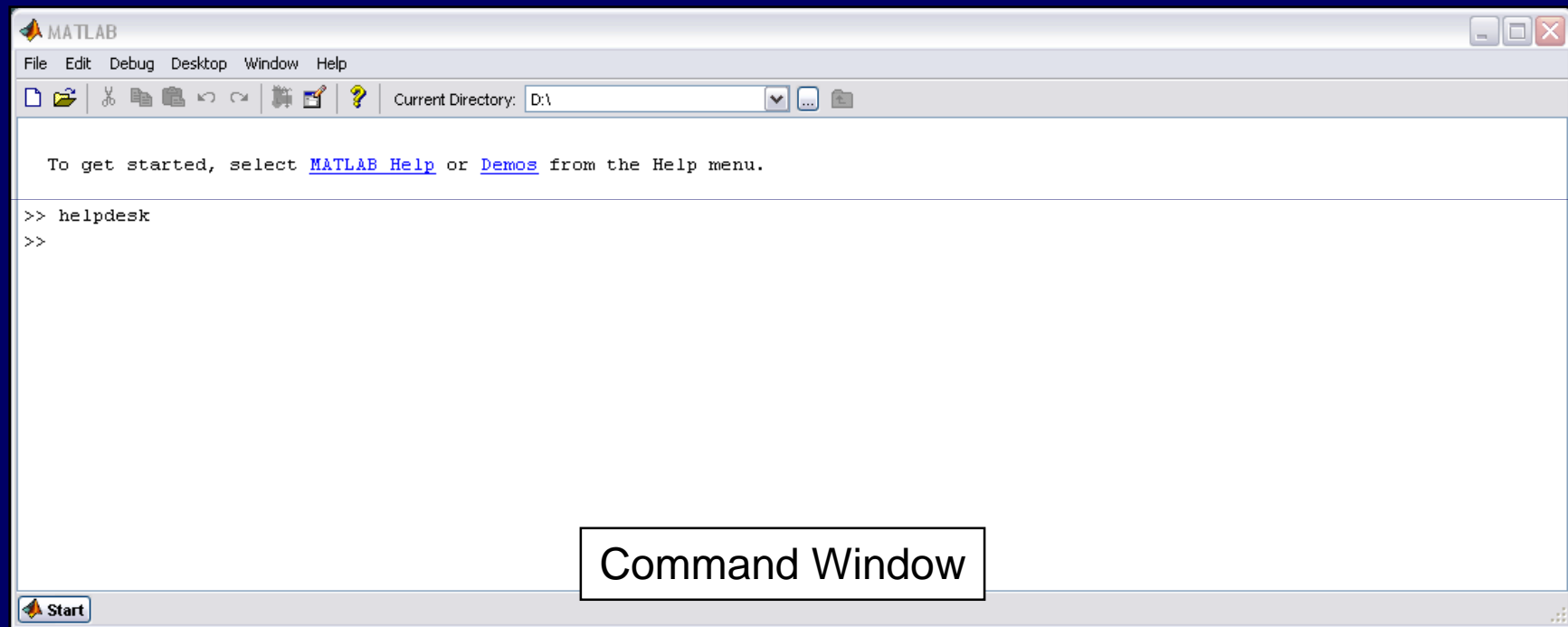
y =

     1     4     9    16    25    36    49    64    81   100

>> |
```

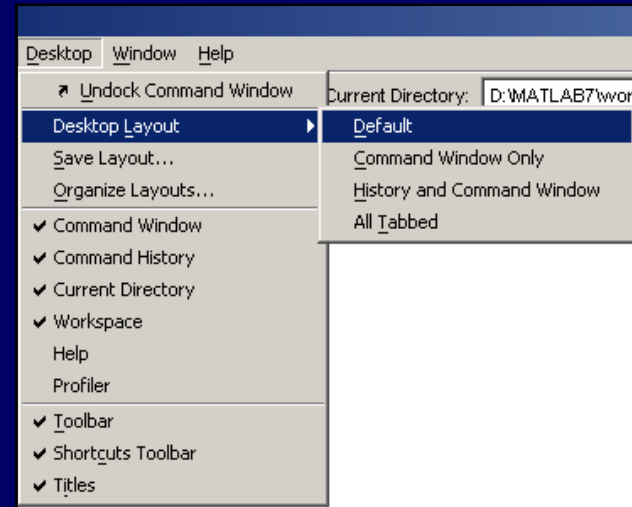
```
gtext('Q_{vap} (g Kg^{-1})')
cprint
get(gcf,'renderer')
get(0,'DefaultFigureRenderer')
close all
clear all
load r006.5.3D.mat THETA
clear
help fopen
help fft
helpdesk
clear
%-- 3/17/05 11:01 AM --%
x=[1:10]
y=x.^2
```

Línea de comandos



Ejercicio 2

- Ingrese al MATLAB
- Identifique el path por defecto.
- Identifique el entorno de trabajo por defecto.
- Identifique los entornos:
 - **Command Window**
 - **Command History**
 - **Workspace Browser**
 - **Editor/Debugger**
 - **Help Browser**



Ejercicio 3

- Ingrese en el command window las siguientes instrucciones, e interprete los resultados:

```
x = [-3:0.1:3];
```

```
y = sin(x);
```

```
plot(x)
```

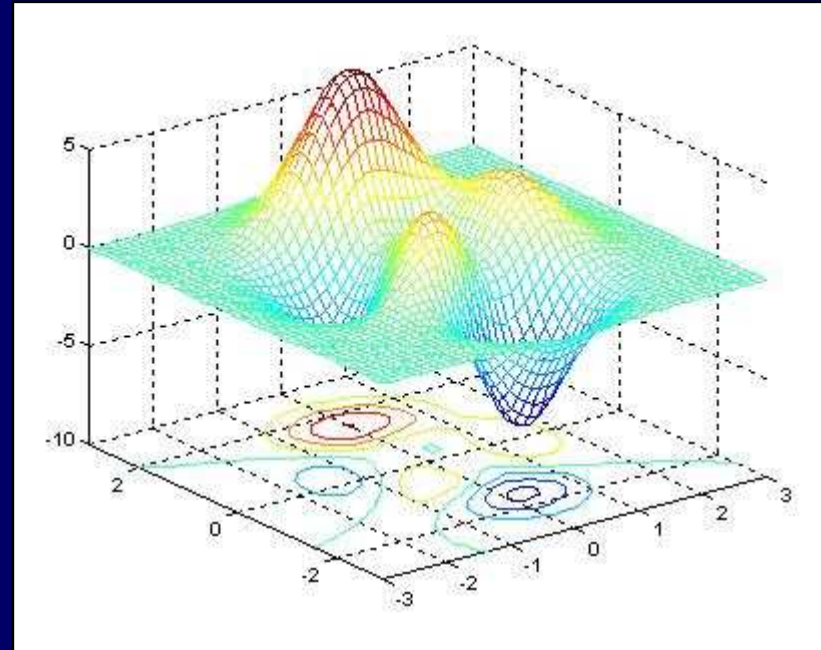
```
plot(y)
```

```
plot(x,y)
```

Ejercicio 4

- Ingrese en el command window las siguientes instrucciones:

```
[X,Y] = meshgrid(-3:.125:3);  
size(X);  
size(Y);  
Z = peaks(X,Y);  
size(Z)  
meshc(X,Y,Z);  
axis([-3 3 -3 3 -10 5]);
```

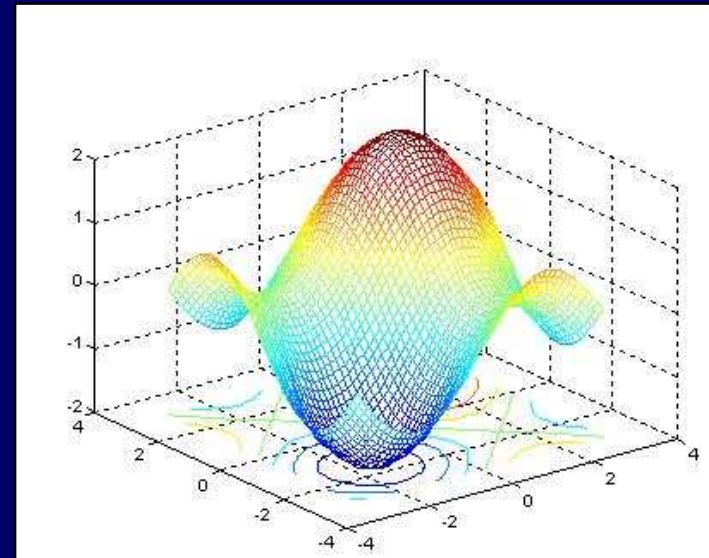


peaks es una función de dos variables, obtenida traducida y a escala de distribuciones Gaussianas.

Ejercicio 5

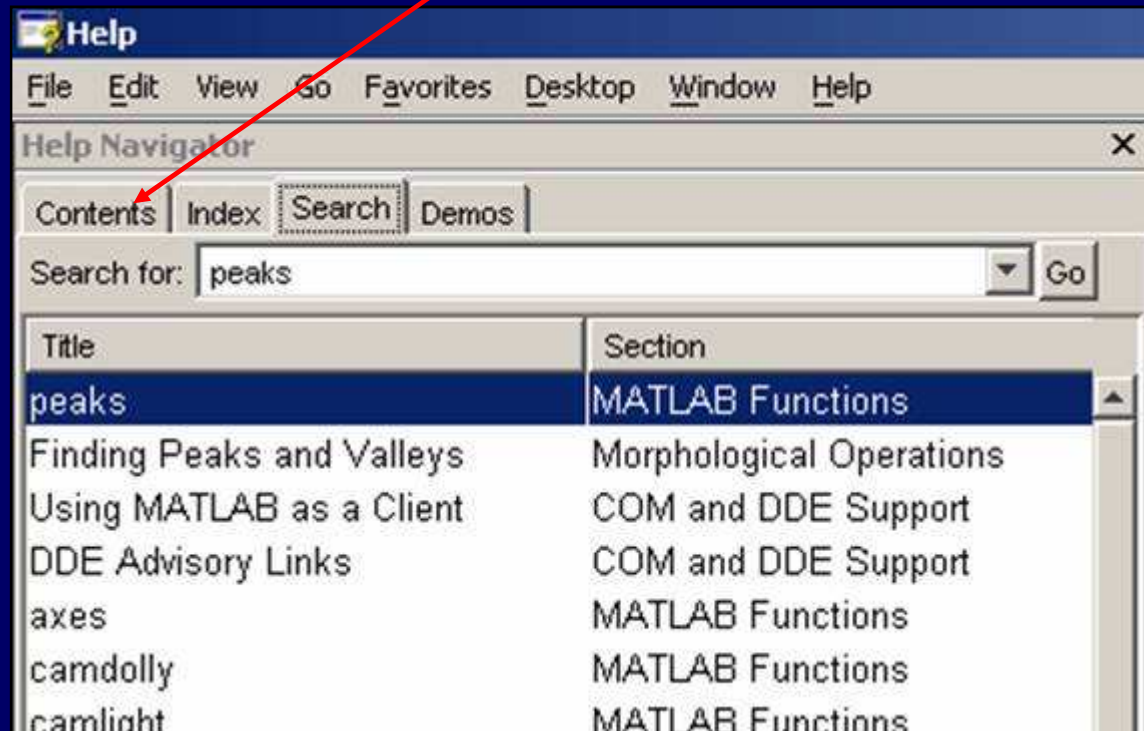
- Ingrese en el command window las siguientes instrucciones:

```
[X,Y] = meshgrid(-3:.125:3);  
[Rows, Cols] = size(X);  
for r = 1:Rows  
    for c = 1:Cols  
        Z(r,c) = sin(X(r,c)) + sin(Y(r,c));  
    end  
end  
meshc(X,Y,Z);
```



Ejercicio 6

- Ingrese a la ayuda del MatLab
helpdesk
- Identifique cada uno de los toolbox disponibles.



Espacio de trabajo

- MATLAB mantiene un espacio de trabajo activa.
- Las variables (datos) cargados o definidos están siempre disponibles.

who: lista las variables del espacio de trabajo

```
>> who
```

```
Your variables are:
```

```
x y
```

whos: listas nombres y propiedades básicas de las variables

```
>> whos
Name      Size      Bytes Class

x         3x1       24 double array
y         3x2       48 double array

Grand total is 9 elements using 72 bytes
```

pwd, cd, dir, ls: similar a los comandos del sistema operativo

```
>> pwd
ans =

D:\

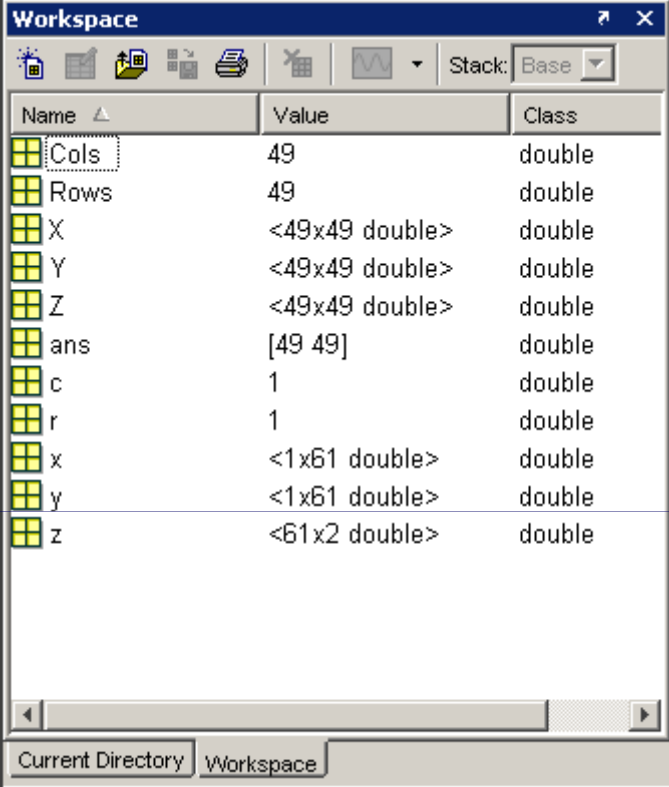
>> cd cw96\jun02
>> dir
.   30m_wtv.mat   edson2km.mat   jun02_30m_runs.mat
..  960602_sst.mat edson_2km_bulk.mat
```

Funciones Generales

whos	Lista las variables actuales y su tamaño
clear	Limpia las variables y funciones de la memoria
cd	Cambie el directorio actual de trabajo
dir	Presenta la lista de archivos en el directorio
pwd	Indica el directorio de trabajo actual
clc	Limpia el contenido de la línea de comandos
echo	eco de los comandos en m-files
format	Asigna el formato de salida (long, short, etc.)
diary(filename)	Graba todos los comandos ingresados en un archivo en el directivo actual, el nombre es filename.

Ejercicio 7

- Revise las variables creadas desde que ingreso al MatLab.
- Defina una carpeta de trabajo.
`cd path_de_trabajo`
- Grabe todas las variables creadas hasta en el momento.
`save primeraclase`
- Elimine todas la variables
`clear`
- Recupere las variables
`load primeraclase`



The screenshot shows the MATLAB Workspace window with a table of variables. The table has three columns: Name, Value, and Class. The variables listed are Cols, Rows, X, Y, Z, ans, c, r, x, y, and z. The 'ans' variable has a value of [49 49]. The other variables have values represented by their dimensions and data type (double).

Name	Value	Class
Cols	49	double
Rows	49	double
X	<49x49 double>	double
Y	<49x49 double>	double
Z	<49x49 double>	double
ans	[49 49]	double
c	1	double
r	1	double
x	<1x61 double>	double
y	<1x61 double>	double
z	<61x2 double>	double

Cálculos en la línea de comandos

- Calculadora

```
» -5/(4.8+5.32)^2
ans =
    -0.0488
» (3+4i)*(3-4i)
ans =
    25
» cos(pi/2)
ans =
    6.1230e-017
» exp(acos(0.3))
ans =
    3.5470
```

- Asignación de Variables

```
» a = 2;
» b = 5;
» a^b
ans =
    32
» x = 5/2*pi;
» y = sin(x)
y =
    1
» z = asin(y)
z =
    1.5708
```

Punto y coma
permite que no
presente el
resultado

El resultado tiene el
nombre "ans" si el
nombre no se
especifica

() paréntesis para el
ingreso de
funciones

En el Workspace los números son almacenados con formato de punto flotante de doble precisión

Ejercicio 8

- Calcule las siguientes expresiones en MatLab

$$E = 2.3026$$

$$x = e^2 - 2 / \text{sen}(3)$$

$$y = 2x + 3\cos(x) - 3^x$$

$$a = 3 + 2i$$

$$b = 4 + 5i$$

$$c = a + b$$

$$d = a - b$$

$$e = c * d$$

$$s = \text{sen}(e)$$

← debe de pasar a matlab
←

$$\text{sen}(0)$$

$$0/0$$

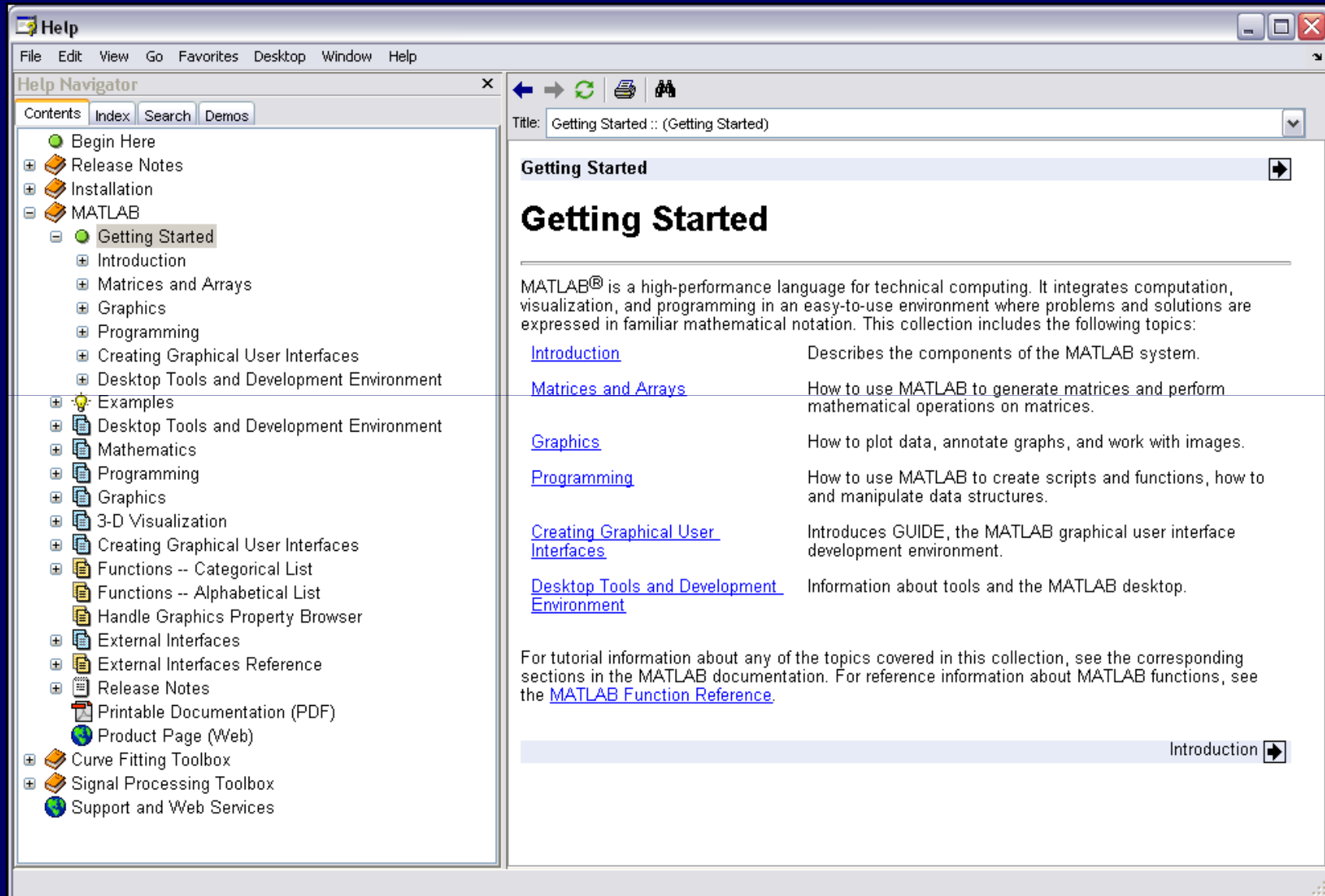
$$1/0$$

Consiguiendo Ayuda

- Comando de ayuda (**help**)
- Comando lookfor (**lookfor**)
- Visor de ayuda (**doc**)
- Comando helpwin (**helpwin**)
- Comando heldesk (**helpdesk**)
- Motor de búsqueda

- Documentos imprimibles “**MATLAB_ROOT\help\pdf_doc**”
- Enlace a la página del MathWorks: **www.mathworks.com**

Conseguir ayuda



startup.m y pathdef.m

`$matlab_root\toolbox\local\startup.m`

- Siempre se ejecuta al inicio.
- Puede definir path y forzar configuraciones predefinidas.

`$matlab_root\toolbox\local\pathdef.m`

- Lee las variables de entorno de sistema e inicializa las variables dependientes de la plataforma.
- Almacena nuevos paths para la búsqueda de funciones.

startup.m y pathdef.m

```
% pathdef.m
function p = pathdef
% PATHDEF Search path defaults.
% PATHDEF returns string that can be used as input to
MATLABPATH
% in order to set the path.
```

```
% Copyright 1984-2002 The MathWorks, Inc.
% $Revision: 1.4.2.1 $ $Date: 2003/01/16 12:51:34 $
```

```
% DO NOT MODIFY THIS FILE. IT IS AN AUTOGENERATED FILE.
% EDITING MAY CAUSE THE FILE TO BECOME UNREADABLE TO
% THE PATHTOOL AND THE INSTALLER.
```

```
p = [...
%%% BEGIN ENTRIES %%%
matlabroot, '\toolbox\matlab\general;',...
matlabroot, '\toolbox\matlab\ops;',...
matlabroot, '\toolbox\matlab\lang;',...
matlabroot, '\toolbox\matlab\elmat;',...
matlabroot, '\toolbox\simulink\slvnx;',...
matlabroot, '\work;',...
%%% END ENTRIES %%%
...
];
```

```
p = [userpath,p];
```

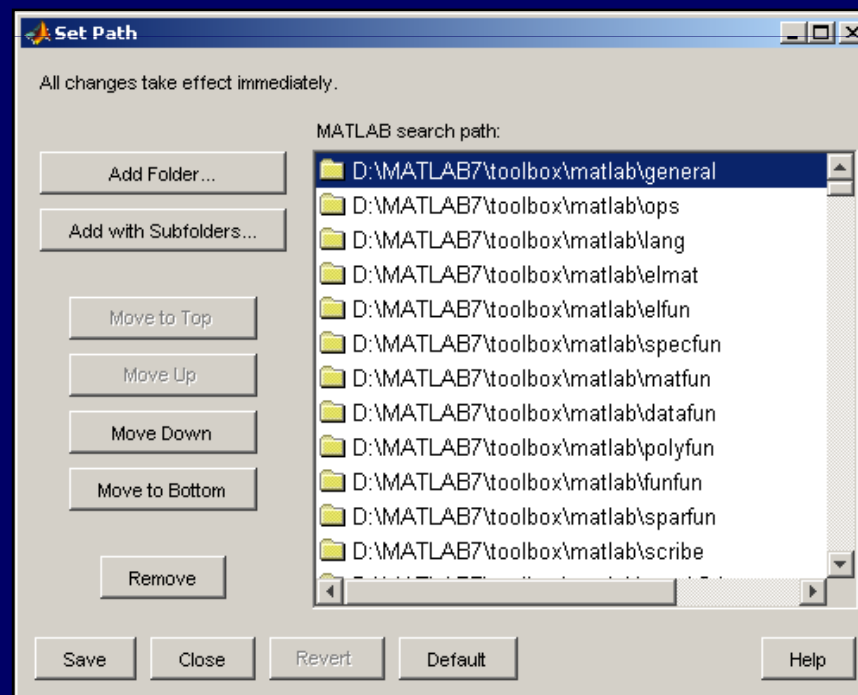
```
% startup.m Startup file
% Change the name of this file to STARTUP.M. The file
% is executed when MATLAB starts up, if it exists
% anywhere on the path. In this example, the
% MAT-file generated during quitting using FINISHSAV
% is loaded into MATLAB during startup.
```

```
% Copyright 1984-2000 The MathWorks, Inc.
% $Revision: 1.4 $ $Date: 2000/06/01 16:19:26 $
```

```
cd c:
```

addpath

- Agrega directorios al **path** de búsqueda del MatLab.
- El MatLab mirará todos los directorios en el **path** para:
 - Funciones y scripts (m-files)
 - Los archivos de datos (mat-files)
- También define el directorio de trabajo actual.



Ejercicio 9

- Identifica los archivos `startup.m` y `pathdef.m` en tu instalación.
- Verifica la configuración disponible.
- Modifica (crea) el archivo `startup.m` para que por defecto:
 - Se inicie en tu carpeta de trabajo
 - Tenga definida las variables `x = 2, y = 3.`

Conclusiones

- Matlab es un lenguaje científico.
- Matlab, un lenguaje de alto nivel.
- El tiempo de prototipeo de una aplicación científica se puede reducir haciendo uso de los toolbox disponibles.

GRACIAS POR SU ATENCIÓN

UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO
FACULTAD DE CIENCIAS QUÍMICAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO ACADÉMICO DE MATEMÁTICAS Y ESTADÍSTICA



UNSAAC

Lic. Guillermo Mario, Chuquipoma Pacheco
mariochuqui@hotmail.com
www.mariochuqui.jimdo.com