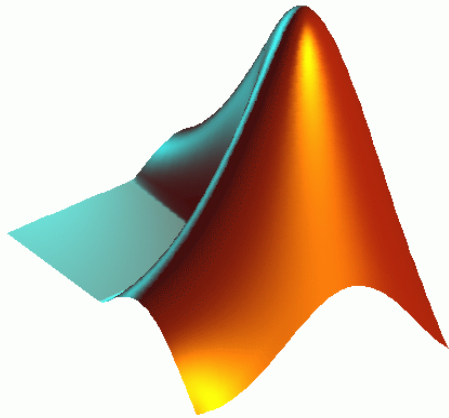




CEU

*Universidad
San Pablo*

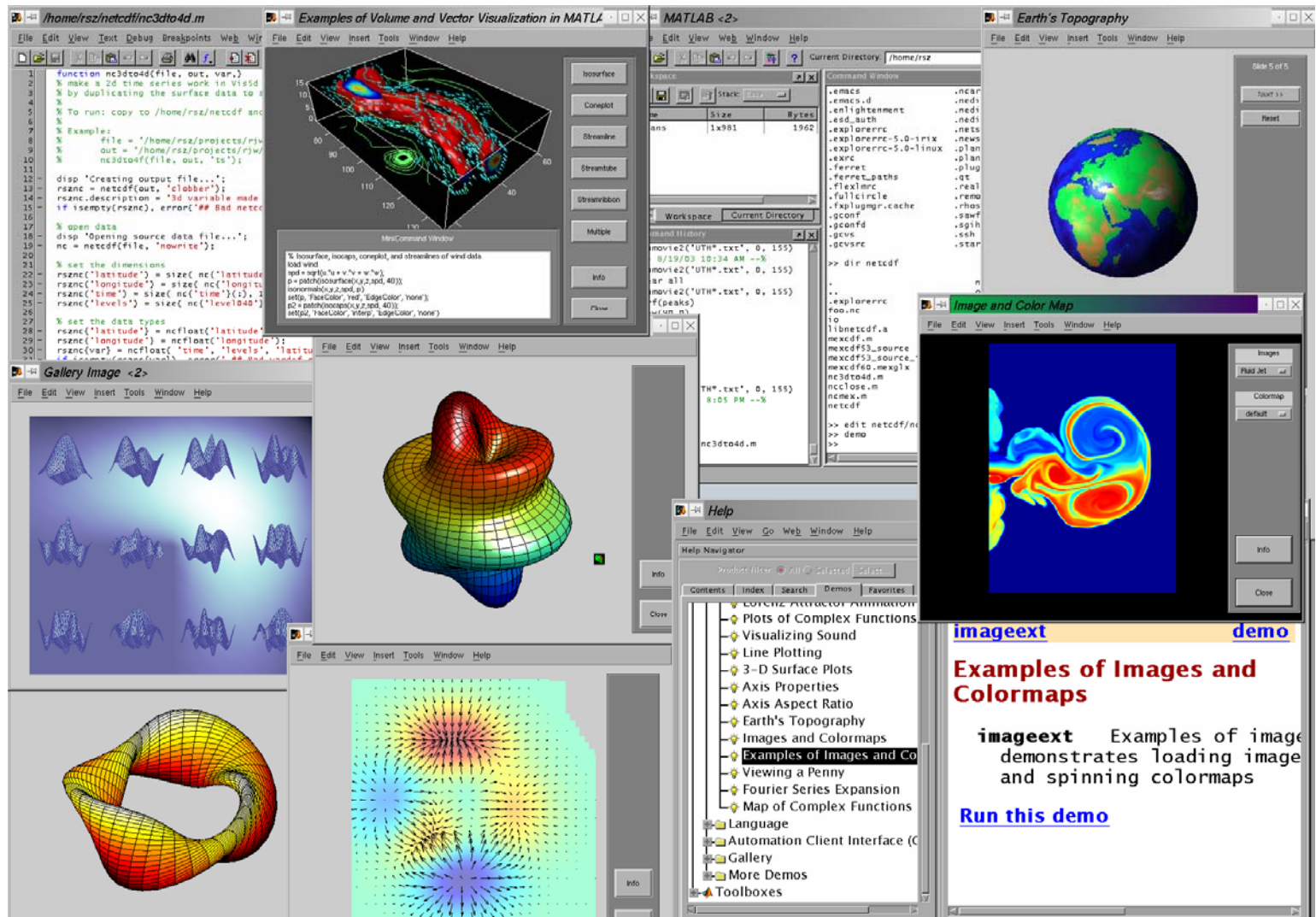


CURSO de UTILIZACIÓN PRÁCTICA de MATLAB

Sesión 0

Carlos Óscar Sánchez Sorzano, Ph.D.
Madrid, July 17th 2006

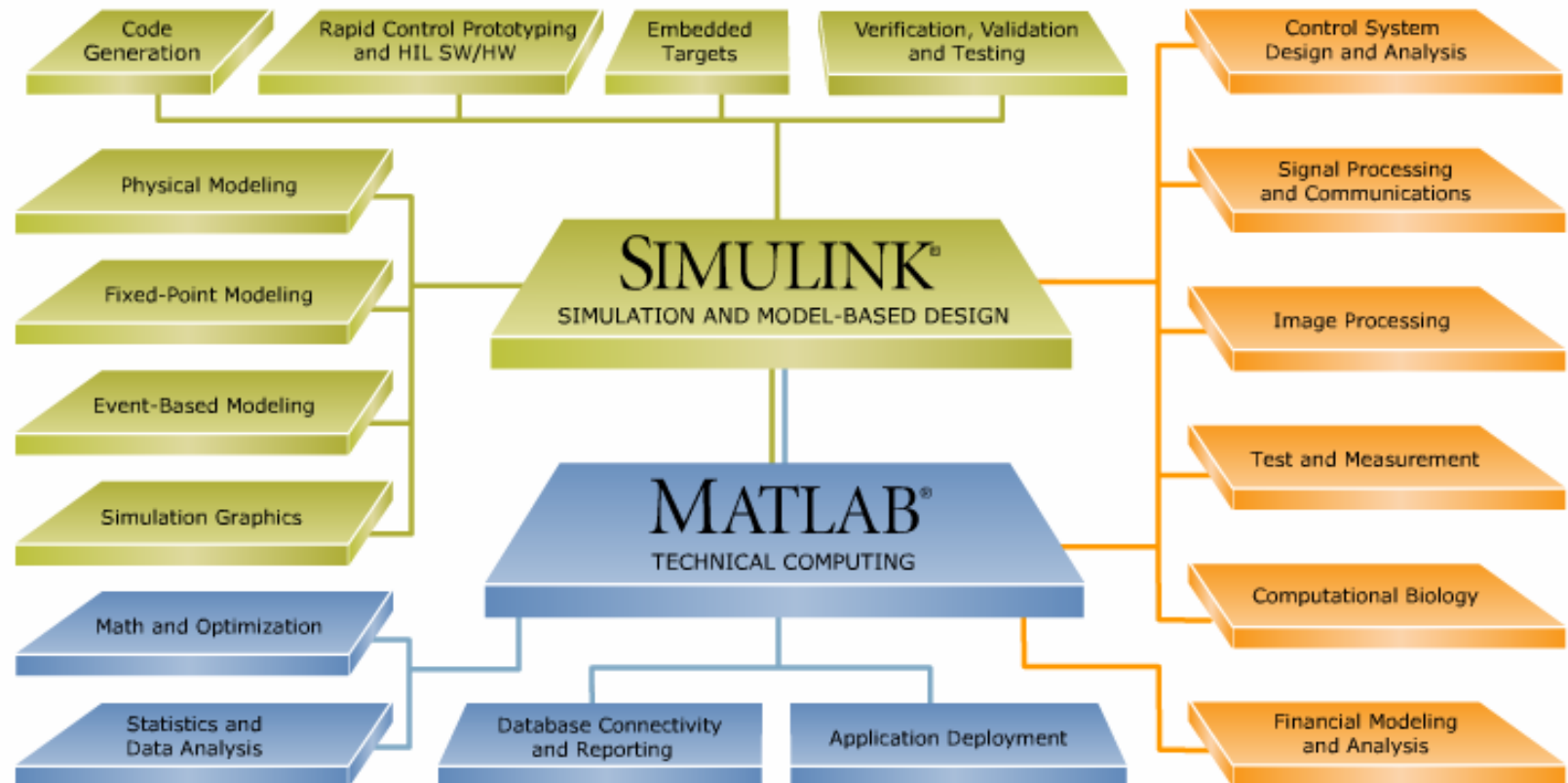
MATLAB como paquete de programas



MATLAB como paquete de programas

Simulink Product Family

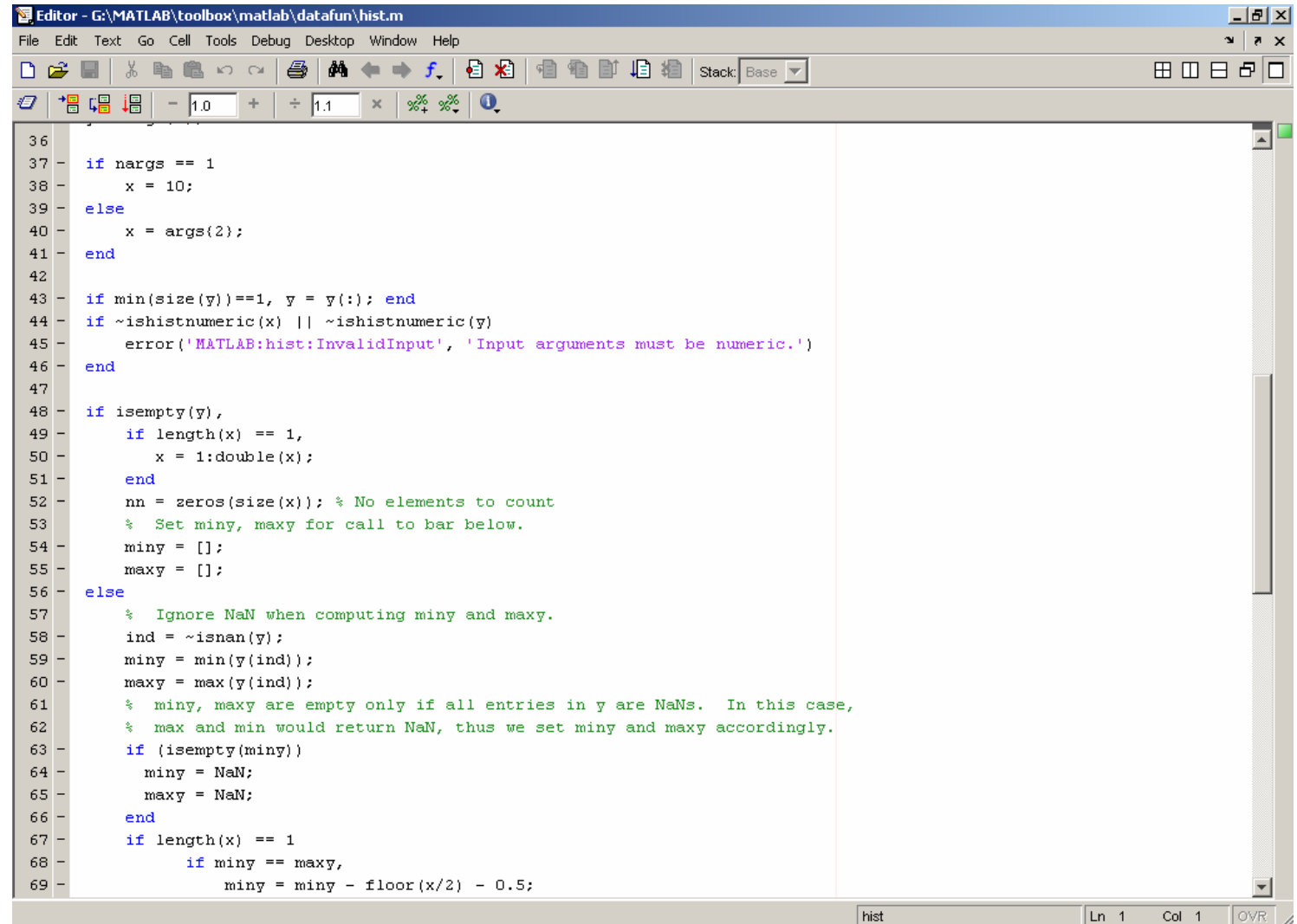
Application-Specific Products



MATLAB Product Family

MATLAB como paquete de programas

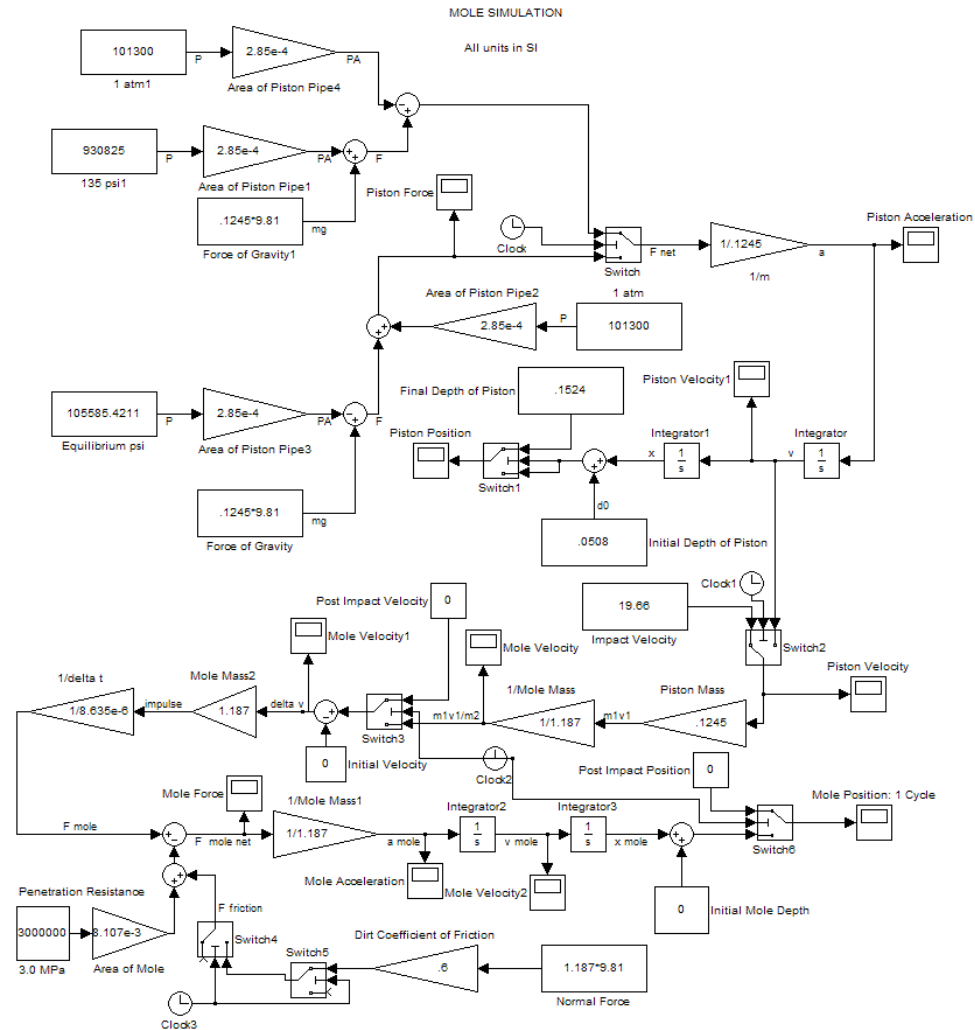
MATLAB



```
36
37 - if nargs == 1
38 -     x = 10;
39 - else
40 -     x = args(2);
41 - end
42
43 - if min(size(y))==1, y = y(:); end
44 - if ~ishistnumeric(x) || ~ishistnumeric(y)
45 -     error('MATLAB:hist:InvalidInput', 'Input arguments must be numeric.')
46 - end
47
48 - if isempty(y),
49 -     if length(x) == 1,
50 -         x = 1:double(x);
51 -     end
52 -     nn = zeros(size(x)); % No elements to count
53 -     % Set miny, maxy for call to bar below.
54 -     miny = [];
55 -     maxy = [];
56 - else
57 -     % Ignore NaN when computing miny and maxy.
58 -     ind = ~isnan(y);
59 -     miny = min(y(ind));
60 -     maxy = max(y(ind));
61 -     % miny, maxy are empty only if all entries in y are NaNs. In this case,
62 -     % max and min would return NaN, thus we set miny and maxy accordingly.
63 -     if (isempty(miny))
64 -         miny = NaN;
65 -         maxy = NaN;
66 -     end
67 -     if length(x) == 1
68 -         if miny == maxy,
69 -             miny = miny - floor(x/2) - 0.5;
```

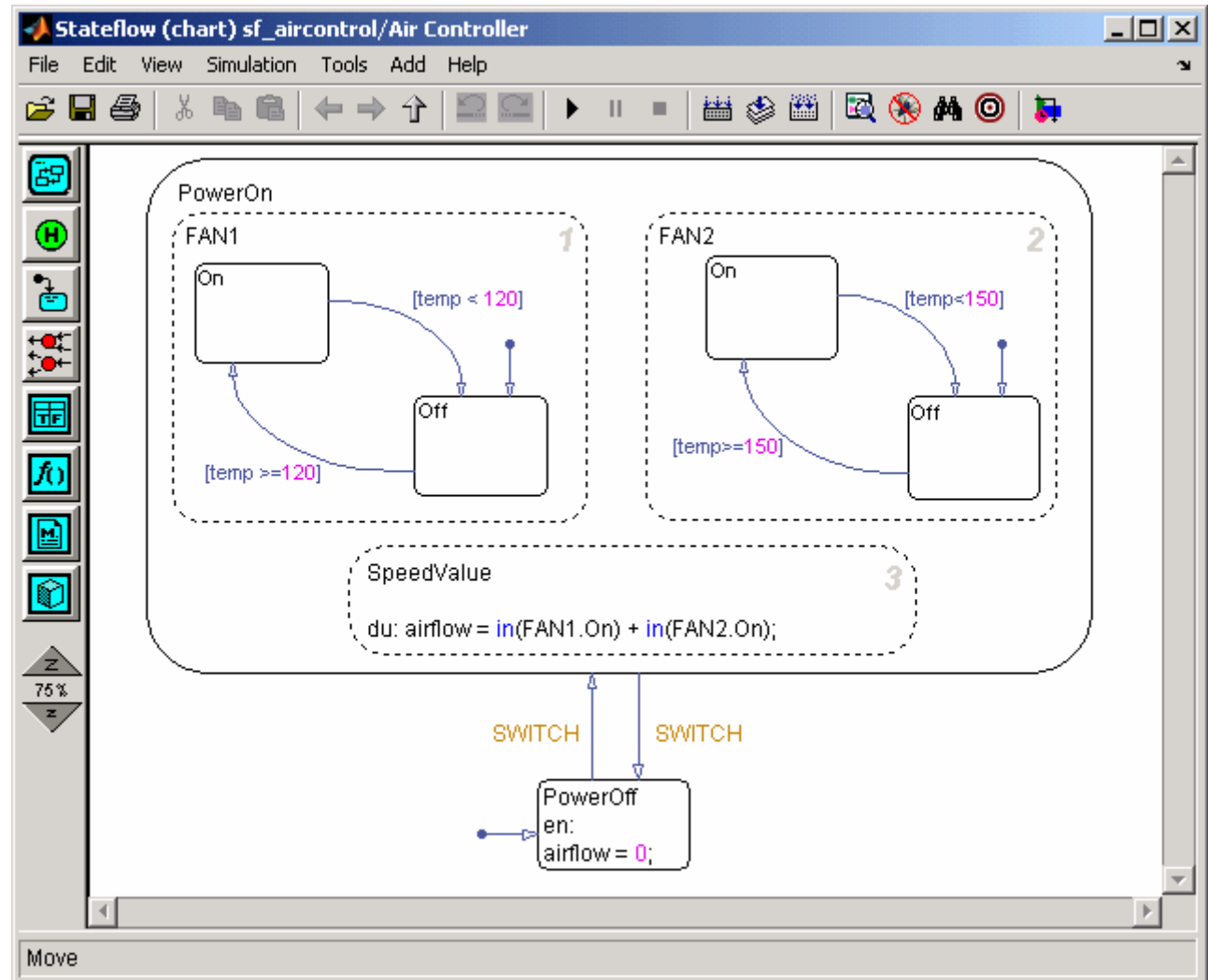

MATLAB como paquete de programas

Simulink



MATLAB como paquete de programas

Stateflow



Un poco de historia

- MATLAB se inventó en los años 70 para poder acceder a las funciones de álgebra lineal de LINPACK y EISPACK sin tener que aprender Fortran.
- Su autor fue Cleve Moler (Univ. New Mexico).
- En 1983, Jack Little (Univ. Stanford) se unió al proyecto y comenzó a comercializarse.



Ventajas y desventajas de MATLAB

- Ventajas
 - Amplio soporte matemático
 - Alta precisión
 - Amplio soporte de funciones ya desarrolladas
 - Rápido prototipado
 - Integración con dispositivos hardware
 - Una comunidad muy extendida
 - Magnífica ayuda
 - Comercial
- Desventajas
 - Gestión “oscura” de la memoria
 - Problemas eventuales de velocidad
 - Comercial
 - Distribución de ejecutables

Motivación para este curso



Septiembre 2006

MATLAB
33.300.000

Michael Jackson
38.500.000

The screenshot shows the MathWorks website with the 'Industries' tab selected. The page lists various industries with brief descriptions and links to learn more or download software. On the right, there is a sidebar with links to contact sales, email the page, and print the page, followed by a section for 'R2006b' software available in September 2006, and a list of application areas including Technical Computing, Control Design, Signal Processing and Communications, Image Processing, Test & Measurement, Computational Biology, and Financial Modeling and Analysis.

The MathWorks
Accelerating the pace of engineering and science

Home | Select Country | Contact Us | Store | Search

Create Account | Log In

Products & Services | **Industries** | Academia | Support | User Community | Company

Industries

Select an industry to see how engineers and scientists in organizations like yours accelerate their productivity and enhance their creativity.

- Aerospace and Defense**
Airframes and payloads, avionics and GNC, C⁴ISR and mission systems, propulsion systems...
- Automotive**
Powertrain, chassis, and safety systems, comfort and convenience, driver infotainment...
- Biotech, Pharmaceutical and Medical**
Biosignal processing, bioinformatics, medical research, medical devices...
- Communications**
Wired and wireless devices and services, optical networking...
- Computers and Office Equipment**
Storage devices, communications devices, ...
- Electronics**
Audio, video, and other digital entertainment devices for home, cars, theatres, and portable use...
- Financial Services**
Financial modeling for fixed income, equity, investment management and trading, reinsurance...
- Industrial Automation and Machinery**
Motors, drives, and power electronics; industrial robotics and manufacturing equipment; process automation systems and industrial controls; power generation and distribution equipment...
- Semiconductors**
Analog and mixed-signal systems, digital logic, SoC, fabrication equipment...
- Other Industries**

Contact sales
E-mail this page
Print this page

R2006b
Available Sept. 2006
» [Learn more](#)
» [Download now](#)

Application Areas

- [Technical Computing](#)
- [Control Design](#)
- [Signal Processing and Communications](#)
- [Image Processing](#)
- [Test & Measurement](#)
- [Computational Biology](#)
- [Financial Modeling and Analysis](#)

Planteamiento del curso



Programa

- Operaciones con matrices y vectores
- Funciones de librería
- Otros tipos de datos en MATLAB: cadenas, hipermatrices, estructuras, celdas
- Programación en MATLAB
- Generación de documentación HTML automáticamente
- Gráficos bidimensionales
- Gráficos tridimensionales
- Interfaces de usuario en MATLAB
- Generación de programas autónomos
- Interacción de MATLAB con Office y Visual Basic
- Librerías de interés práctico: optimización, análisis estadístico, redes neuronales, lógica difusa, ajuste de curvas, sistemas de control, control predictivo, análisis financiero, análisis de series temporales financieras, conexión a bases de datos, generación de informes, sistemas eléctricos de potencia, modelos cuantitativos de energía, etc.

Cronograma del curso

- Sesión 1: Operaciones con matrices y vectores. Funciones de librería.
- Sesión 2: Otros tipos de datos en MATLAB. Programación en MATLAB.
- Sesión 3: Gráficos bidimensionales. Gráficos tridimensionales.
- Sesión 4: Interfaces de usuario en MATLAB. Generación de programas autónomos
- Sesión 5: Librerías de interés práctico
- Sesión 6: Interacción de MATLAB con Office
- Sesión 7: Desarrollo de un proyecto

Sesión 1: Operaciones con matrices y vectores.

Funciones de librería.

The image shows a MATLAB interface with two main windows: the **Workspace** window on the left and the **Command Window** on the right. The **Workspace** window displays a table of variables:

Name	Value	Class
A	[1 2 3; 4 5 6; 7 8 9]	double
B	[1 4 7; 2 5 8; 3 6 9]	double
ans	[1; 4; 7]	double
e1	[1; 0; 0]	double

The **Command Window** shows the following commands and their outputs:

```
>> A=[1 2 3; 4 5 6; 7 8 9]
A =
     1     2     3
     4     5     6
     7     8     9

>> B=A'
B =
     1     4     7
     2     5     8
     3     6     9

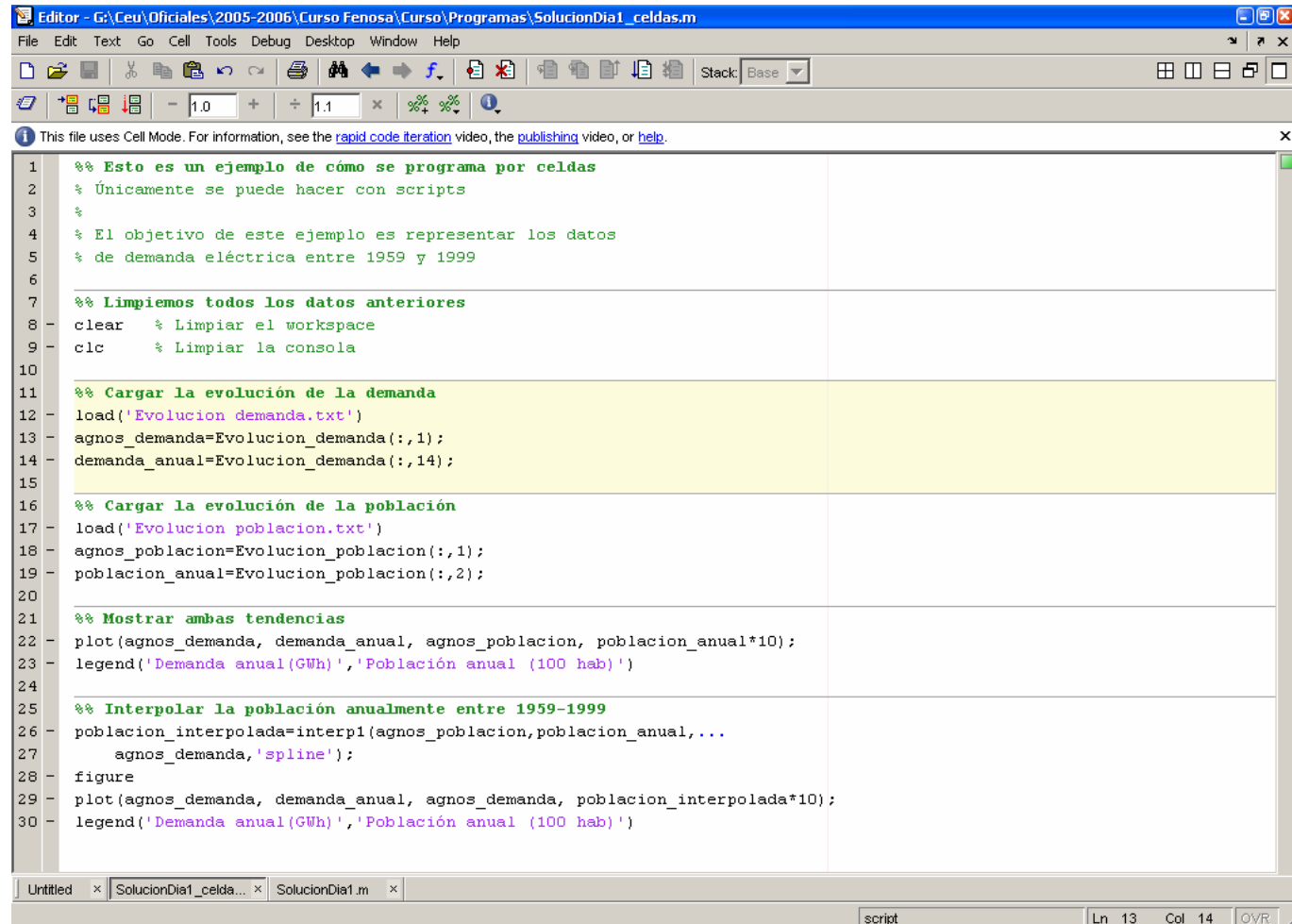
>> e1=[1; 0; 0]
e1 =
     1
     0
     0

>> A*e1
ans =
     1
     4
     7
```

Annotations (green boxes) point to specific parts of the interface:

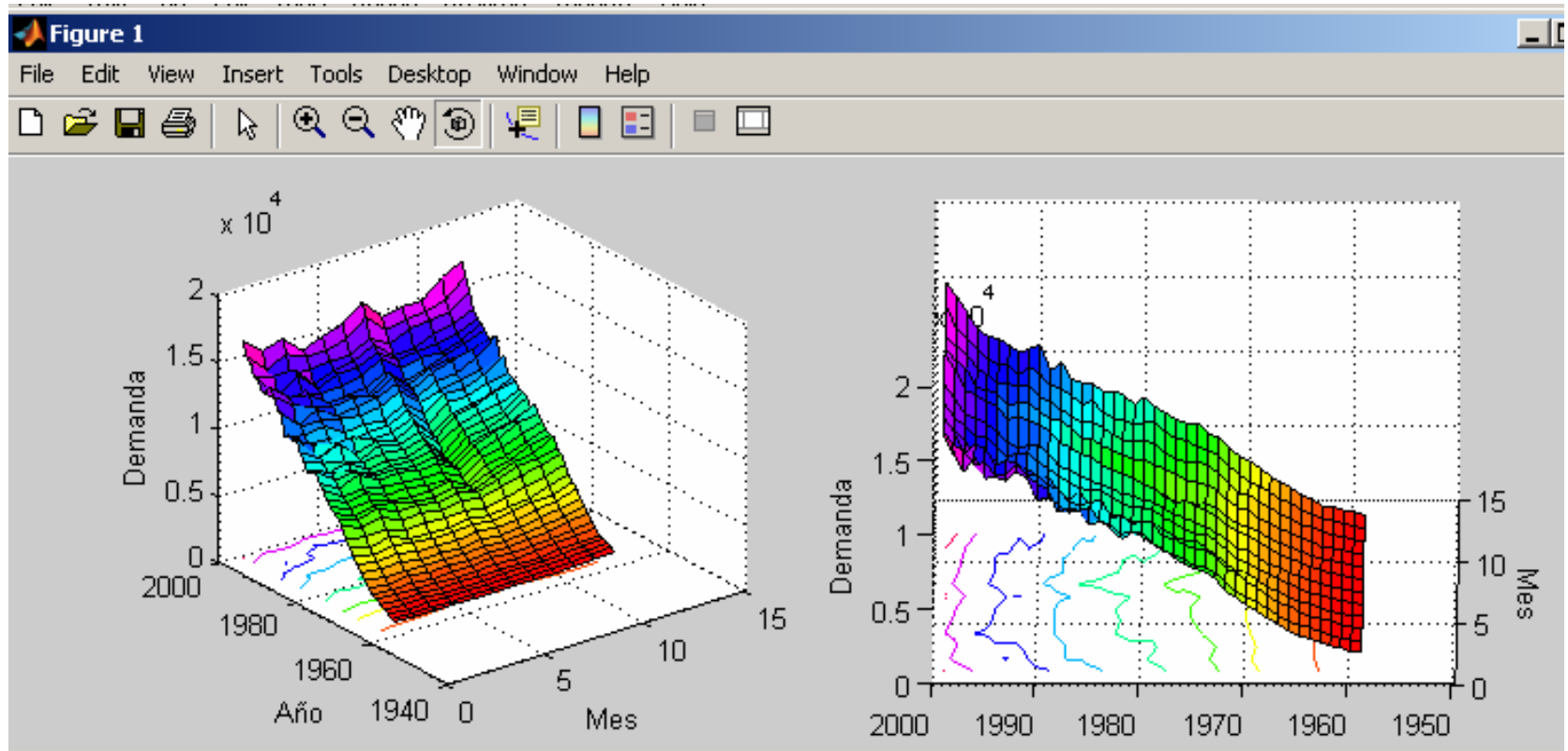
- Definición de la matriz A**: Points to the command `A=[1 2 3; 4 5 6; 7 8 9]` in the Command Window.
- Definición de la matriz B como A transpuesta**: Points to the command `B=A'` in the Command Window.
- Definición de e1 como un vector columna**: Points to the command `e1=[1; 0; 0]` in the Command Window.
- Multiplicación de A por e1 (matriz por vector)**: Points to the command `A*e1` in the Command Window.
- Si no se asigna a nadie, el resultado siempre se asigna a la variable ans**: Points to the `ans` variable in the Workspace window.

Sesión 2: Otros tipos de datos en MATLAB. Programación en MATLAB.

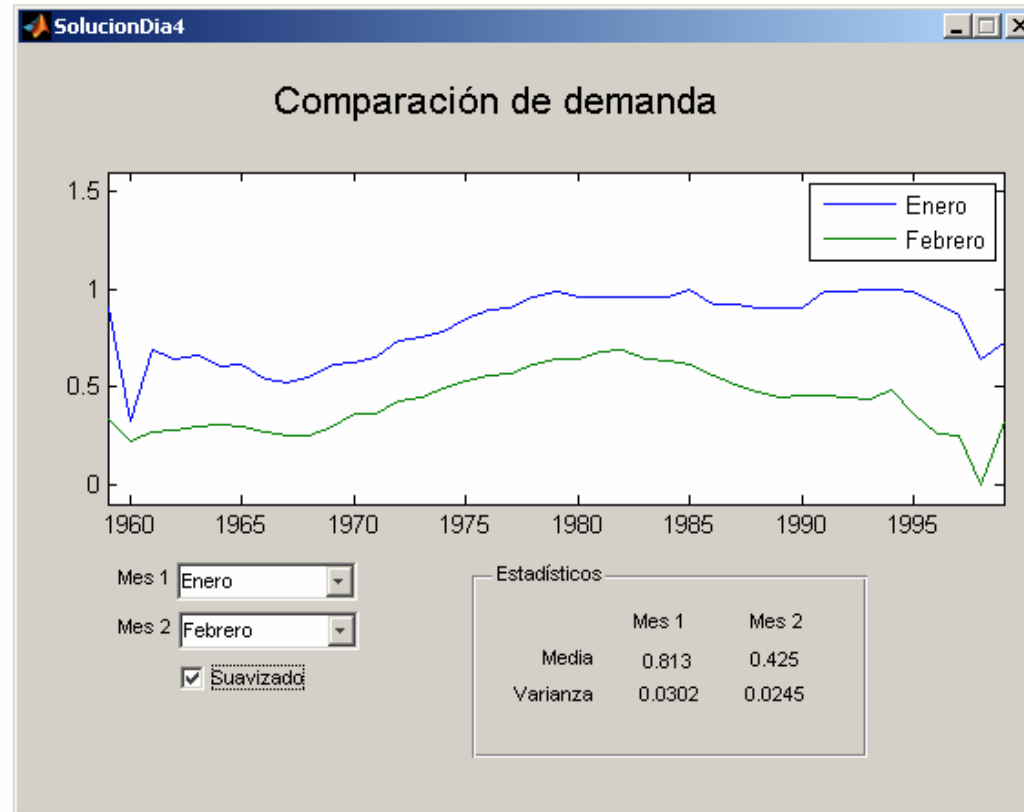


```
1 %% Esto es un ejemplo de cómo se programa por celdas
2 % Únicamente se puede hacer con scripts
3 %
4 % El objetivo de este ejemplo es representar los datos
5 % de demanda eléctrica entre 1959 y 1999
6
7 %% Limpiemos todos los datos anteriores
8 clear % Limpiar el workspace
9 clc % Limpiar la consola
10
11 %% Cargar la evolución de la demanda
12 load('Evolucion_demanda.txt')
13 agnos_demanda=Evolucion_demanda(:,1);
14 demanda_anual=Evolucion_demanda(:,14);
15
16 %% Cargar la evolución de la población
17 load('Evolucion_poblacion.txt')
18 agnos_poblacion=Evolucion_poblacion(:,1);
19 poblacion_anual=Evolucion_poblacion(:,2);
20
21 %% Mostrar ambas tendencias
22 plot(agnos_demanda, demanda_anual, agnos_poblacion, poblacion_anual*10);
23 legend('Demanda anual (GWh)', 'Población anual (100 hab)')
24
25 %% Interpoliar la población anualmente entre 1959-1999
26 poblacion_interpolada=interp1(agnos_poblacion, poblacion_anual, ...
27     agnos_demanda, 'spline');
28 figure
29 plot(agnos_demanda, demanda_anual, agnos_demanda, poblacion_interpolada*10);
30 legend('Demanda anual (GWh)', 'Población anual (100 hab)')
```

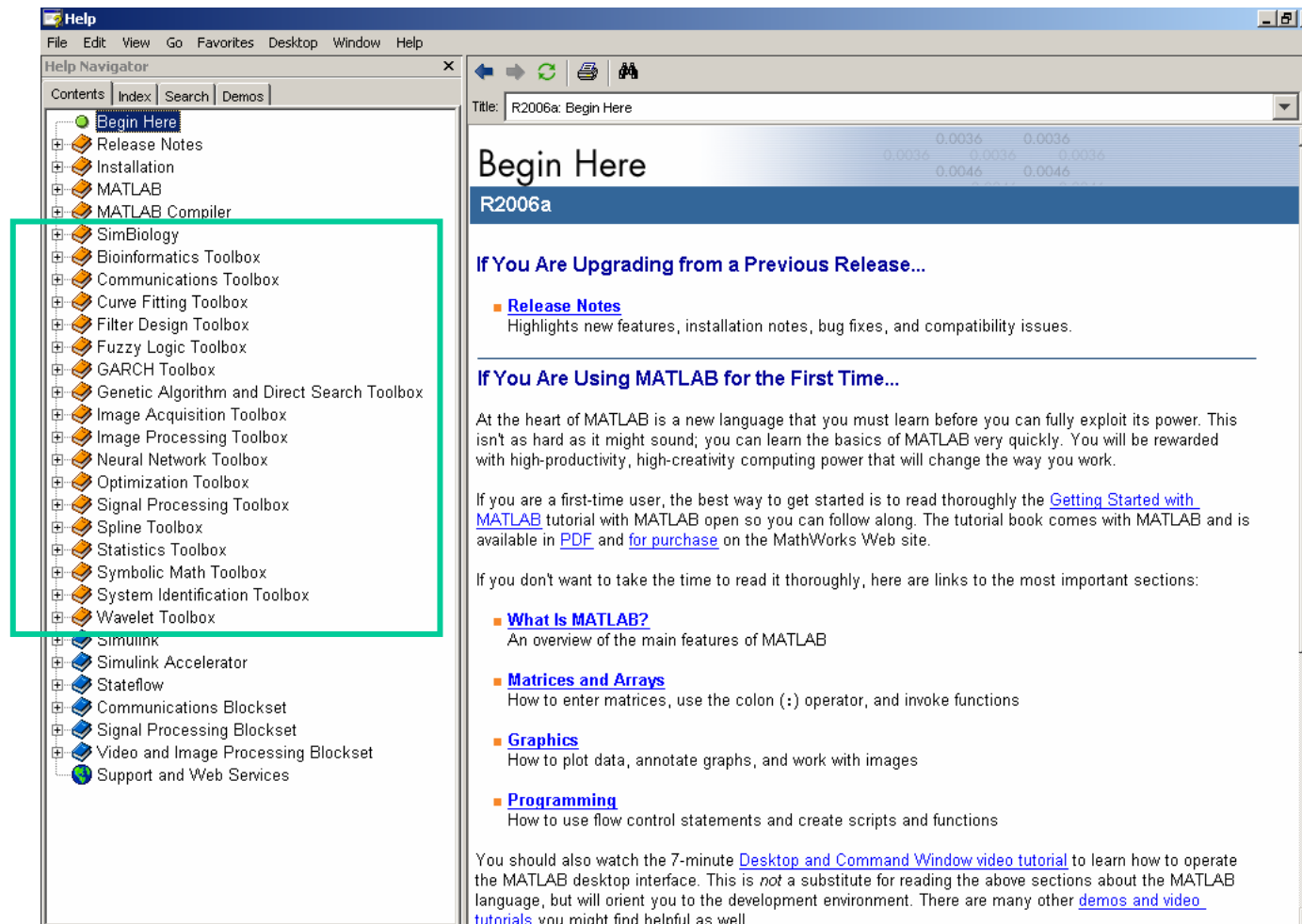
Sesión 3: Gráficos bidimensionales. Gráficos tridimensionales.



Sesión 4: Interfaces de usuario en MATLAB. Generación de programas autónomos



Sesión 5: Librerías de interés práctico



The screenshot shows the MATLAB Help Navigator window. The left pane, titled 'Help Navigator', contains a tree view of the help content. The 'Begin Here' section is highlighted with a green box. The right pane, titled 'Begin Here', displays the content for the 'Begin Here' section, including a table of contents and a list of links to various sections.

Begin Here

R2006a

If You Are Upgrading from a Previous Release...

- [Release Notes](#)
Highlights new features, installation notes, bug fixes, and compatibility issues.

If You Are Using MATLAB for the First Time...

At the heart of MATLAB is a new language that you must learn before you can fully exploit its power. This isn't as hard as it might sound; you can learn the basics of MATLAB very quickly. You will be rewarded with high-productivity, high-creativity computing power that will change the way you work.

If you are a first-time user, the best way to get started is to read thoroughly the [Getting Started with MATLAB](#) tutorial with MATLAB open so you can follow along. The tutorial book comes with MATLAB and is available in [PDF](#) and [for purchase](#) on the MathWorks Web site.

If you don't want to take the time to read it thoroughly, here are links to the most important sections:

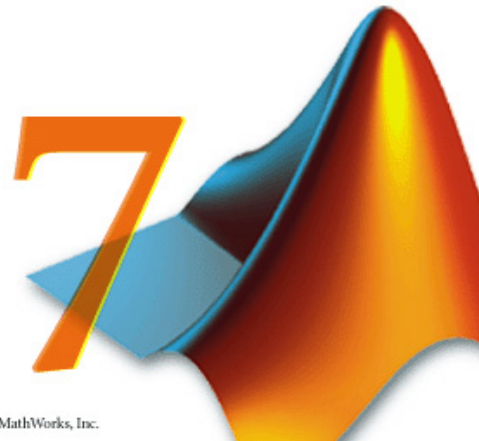
- [What Is MATLAB?](#)
An overview of the main features of MATLAB
- [Matrices and Arrays](#)
How to enter matrices, use the colon (:) operator, and invoke functions
- [Graphics](#)
How to plot data, annotate graphs, and work with images
- [Programming](#)
How to use flow control statements and create scripts and functions

You should also watch the 7-minute [Desktop and Command Window video tutorial](#) to learn how to operate the MATLAB desktop interface. This is not a substitute for reading the above sections about the MATLAB language, but will orient you to the development environment. There are many other [demos and video tutorials](#) you might find helpful as well.

Sesión 6: Interacción de MATLAB con Office y Visual Basic



MATLAB[®]
The Language of Technical Computing



Copyright 1984–2004, The MathWorks, Inc.



Sesión 7: Desarrollo de un proyecto



Organización temporal

	Martes	Miércoles	Jueves	
9'00	Sesión 0	Sesión 3	Sesión 7	
11'00				
11'30				
	Sesión 1	Sesión 4	Sesión 7	
14'00				
16'00				
	Sesión 2	Sesión 5		
14'00				
16'00				
	Sesión 2	Sesión 6		
14'00				

Lecturas recomendadas

- Mathworks, Getting Started (187 págs)
- Mathworks, Programming tips (66 págs)
- Mathworks, Quick reference (12 págs)

Recursos

The screenshot shows the MathWorks Support page in Microsoft Internet Explorer. The browser's address bar displays the URL <http://www.mathworks.com/support/>. The page features a navigation bar with links for Home, Select Country, Contact Us, Store, and Search. Below this, a menu highlights the 'Support' section among other options like Products & Services, Industries, Academia, User Community, and Company.

The main content area is titled 'Support' and is organized into several sections:

- Browse by Product:** Includes a dropdown for 'View All Product Support Resources' (currently set to 'Full Product List') and links for 'Installation and Licensing' (Installation Troubleshooting, Get My Passcode(PLP)/License File, System Requirements) and 'License Manager Errors' (License Manager Errors).
- Search Support:** A search bar with a dropdown menu (currently set to 'Entire Support Site') and a 'for:' field.
- Manage Account:** Links for 'My Account - Licenses, Orders...', 'Create an Account', 'Account Benefits', and 'Customer Service Options'.
- Browse by Resource:** A grid of links for 'Documentation' (Product Documentation, Index of MATLAB Examples, Function List - All Products, Tech Notes / How-To Guides), 'User Community' (File Exchange, MATLAB Newsgroup, Link Exchange), 'Bug Reports', and 'Additional Resources' (MATLAB and Simulink Based Books).
- New Support Resources:** A section titled 'New Support Resources' with the text 'Learn how to effectively use MathWorks products and quickly resolve technical problems.' and a link to 'Learn more'.
- R2006b:** A section for 'R2006b' with the text 'Available Sept. 2006' and links to 'Learn more' and 'Download now'.
- Track Your Support Requests Online:** A section titled 'Track Your Support Requests Online' with the text 'Create and monitor customer service and technical support requests.' and a link to 'Learn more'.
- Provide Feedback:** A section titled 'Provide Feedback'.

The status bar at the bottom indicates '(Quedan 1 elemento) Abriendo página http://www.mathworks.com/support/...' and shows the Internet Explorer logo.

Recursos

The screenshot shows the MathWorks website in a Microsoft Internet Explorer browser window. The address bar displays the URL: <http://www.mathworks.com/access/helpdesk/help/helpdesk.html>. The page title is "The MathWorks - Online Documentation, R2006b". The website features a navigation bar with links for Home, Select Country, Contact Us, Store, and Search. Below this is a sidebar with links for Products & Services, Industries, Academia, Support, User Community, and Company. The main content area is titled "Documentation" and "Documentation for MathWorks Products, R2006b". It includes a search box with a dropdown menu set to "R2006b Documentation" and a search button. Below the search box, there are two columns of links. The left column lists links for MATLAB, including Release Notes, Installation Guides, License Administration Guide, and Function List. The right column lists links for Simulink, including Embedded Target for Infineon C166, Embedded Target for Motorola HC12, and Embedded Target for TI C6000 DSP. At the bottom of the page, there is a footer with the text "Internet".

The MathWorks
Accelerating the pace of engineering and science

Home | Select Country | Contact Us | Store | Search

Create Account | Log In

Products & Services | Industries | Academia | Support | User Community | Company

Documentation

Documentation for MathWorks Products, R2006b

Alphabetical list by product Search

[Release Notes for R2006b](#)
[Installation Guides](#)
[License Administration Guide](#)
[Function List - All Products](#)

MATLAB
[MATLAB®](#) (including *External Interfaces/API, GUIDE, Handle Graphics, File I/O, Notebook*)
[MATLAB® Builder for .NET](#)
[MATLAB® Builder for Excel](#)
[MATLAB® Builder for Java™](#)
[MATLAB® Compiler](#)
[Distributed Computing Toolbox](#)
[MATLAB® Distributed Computing Engine](#)
[MATLAB® Report Generator](#)
[SimBiology™](#)
[SystemTest](#)

MATLAB Toolboxes
[Aerospace Toolbox](#)

Simulink
[Simulink®](#)
[Embedded Target for Infineon C166® Microcontrollers](#)
[Embedded Target for Motorola® HC12](#)
[Embedded Target for Motorola® MPC555](#)
[Embedded Target for TI C2000™ DSP](#)
[Embedded Target for TI C6000™ DSP](#)
[Link for Code Composer Studio™ Development Tools](#)
[Link for Incisive®](#)
[Link for ModelSim®](#)
[Link for TASKING®](#)
[Real-Time Windows Target](#)
[Real-Time Workshop®](#)

R2006b
Available Sept. 2006
[» Learn more](#)
[» Download now](#)

Provide Feedback
If you are a MATLAB user in the Boston area, provide Web site feedback and earn \$100.
[» Sign up today](#)

E-mail this page
Print this page

Internet

Recursos

The screenshot shows the MATLAB Central website as it appeared in 2006. The browser window is titled "MATLAB Central - Home - Microsoft Internet Explorer". The address bar shows the URL "http://www.mathworks.com/matlabcentral/". The page features a navigation bar with links to "File Exchange", "MATLAB Newsgroup", "Link Exchange", "Blogs", "Contest", and "MathWorks.com". The main content area is divided into several sections: "File Exchange" with a "Recently Added Files..." list; "MATLAB Newsgroup" with a "Most Active Threads..." list; "Link Exchange" with a "Recently Added Links..." list; "MATLAB Central Blogs" with a "Doug's Pick of the Week" and "Loren on the Art of MATLAB"; "Now Available R2006b" with a "Download now" link; "Job Opening" for a "Web Community Business Analyst"; "Free Webinars" for "Open Architecture Solutions for Rapid Prototyping and HIL Using Simulink"; "Link to Us" with a "Link to MATLAB Central" link; and "MATLAB Screensaver" with a "Download Now" link. A "Public Submission Policy" notice is at the bottom left. The status bar at the bottom indicates "(Quedan 1 elemento) Abriendo página http://www.mathworks.com/matlabcentral/..." and shows the Internet Explorer logo.

MATLAB CENTRAL
An open exchange for the MATLAB and Simulink user community

Hosted by The MathWorks

Search:

[File Exchange](#) [MATLAB Newsgroup](#) [Link Exchange](#) [Blogs](#) [Contest](#) [MathWorks.com](#)

File Exchange
[User-contributed code library](#)
Recently Added Files...

- Whats New - MATLAB - R2006b - Ke...
- Knight_Tour - Krishna Lalit...
- structure tensor - introduction ...

MATLAB Newsgroup
[MATLAB Usenet newsgroup access](#)
Most Active Threads...

- Reassigning Commands Should NOT Be...
- Displaying final numerical results...
- cellfun mean w subset indexes

Now Available
R2006b
Learn more
[Download now](#)

Job Opening
Web Community Business Analyst

Free Webinars
Open Architecture Solutions for Rapid Prototyping and HIL Using Simulink
October 5, 2006

Link to Us
[Link to MATLAB Central](#)

MATLAB Screensaver
[Download Now](#)

Link Exchange
[MATLAB related web sites](#)
Recently Added Links...

- MAT-file Viewer plugin for Total Command...
- Hplot (Claudio Gambelli)
- Video seminar : Direct methods for spars...

MATLAB Central Blogs
Doug's Pick of the Week
Latest pick: [Simulink pick](#)
Loren on the Art of MATLAB
Latest entry: [Working with Low Level File I/O and Encodings](#)
Steve on Image Processing
Latest entry: [Unusual red-eye reduction technique](#)

Public Submission Policy
NOTICE: Any content you submit to MATLAB Central, including personal information, is not subject to the protections which may be afforded information collected under the MathWorks, Inc. Web site. You are entirely responsible for all content that you upload, post, e-mail, transmit or otherwise make available via MATLAB Central. The MathWorks does not control the content posted by visitors to MATLAB Central and, does not guarantee the accuracy, integrity, or quality of such content. Under no circumstances will The MathWorks be liable in any way for any content not authored by The MathWorks, or any loss or damage of any kind incurred as a result of the use of any content posted, e-mailed, transmitted or otherwise

(Quedan 1 elemento) Abriendo página http://www.mathworks.com/matlabcentral/...

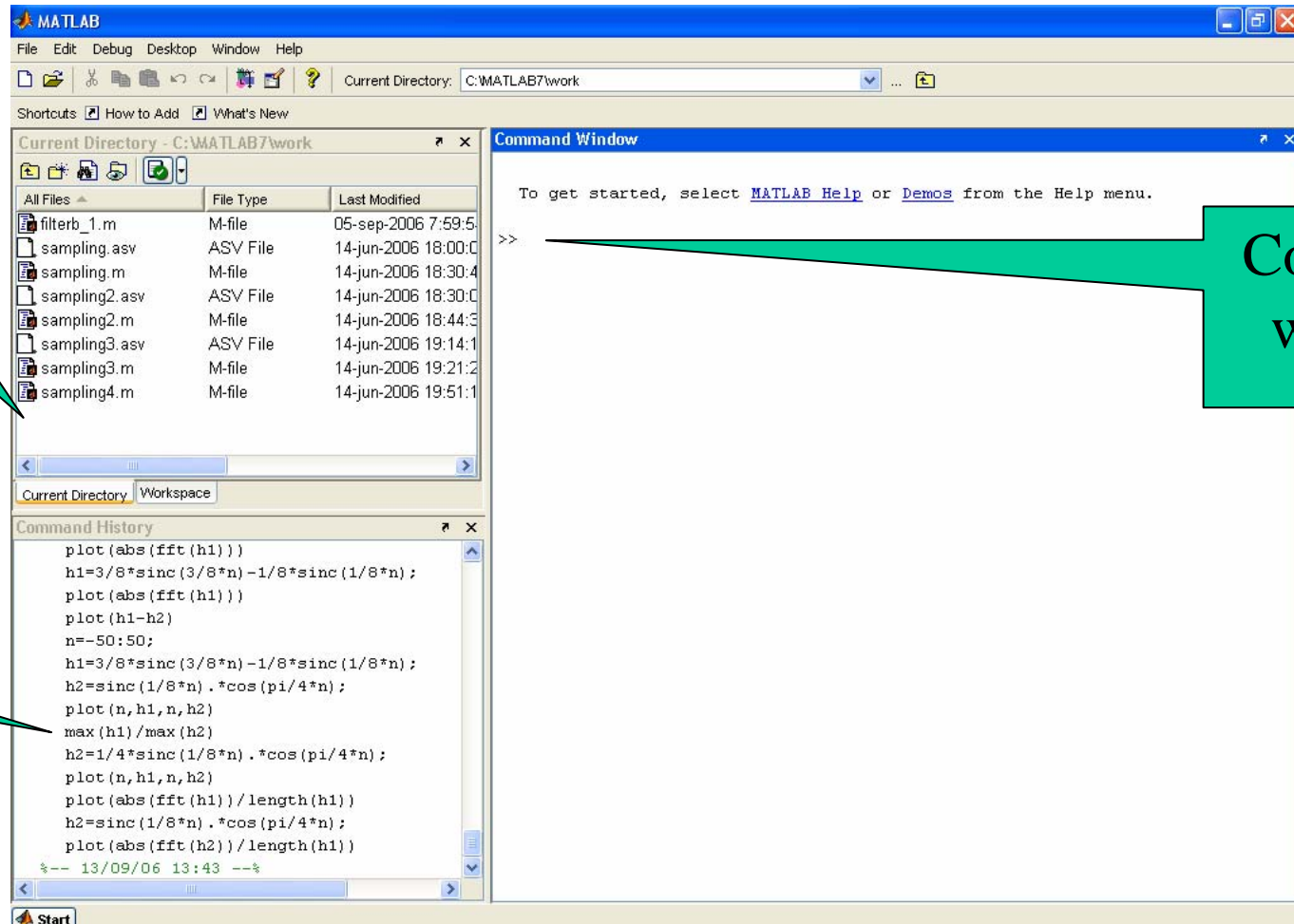
Bibliografía

- Mathworks. *MATLAB online help*
- J. Atencia, R. Néstar. *Aprenda MATLAB 6.0 como si estuviera en primero*. Univ. Navarra, 2001.
- C. Pérez. *MATLAB y sus aplicaciones en las ciencias y la ingeniería*. Prentice Hall, 2002
- G. Amos. *MATLAB: una introducción con ejemplos prácticos*. Reverte, 2006.

Entorno de MATLAB

Current
directory

Command
history



Command
window

Entorno de MATLAB

The image shows the MATLAB desktop environment with three callout boxes pointing to specific components:

- Workspace:** A callout box points to the Workspace window, which displays a table of variables.
- Command window:** A callout box points to the Command Window, which shows the command prompt and the execution of code.
- Command history:** A callout box points to the Command History window, which lists previous commands and their execution times.

Workspace Window:

Name	Value	Class
a	3	double
b	2	double

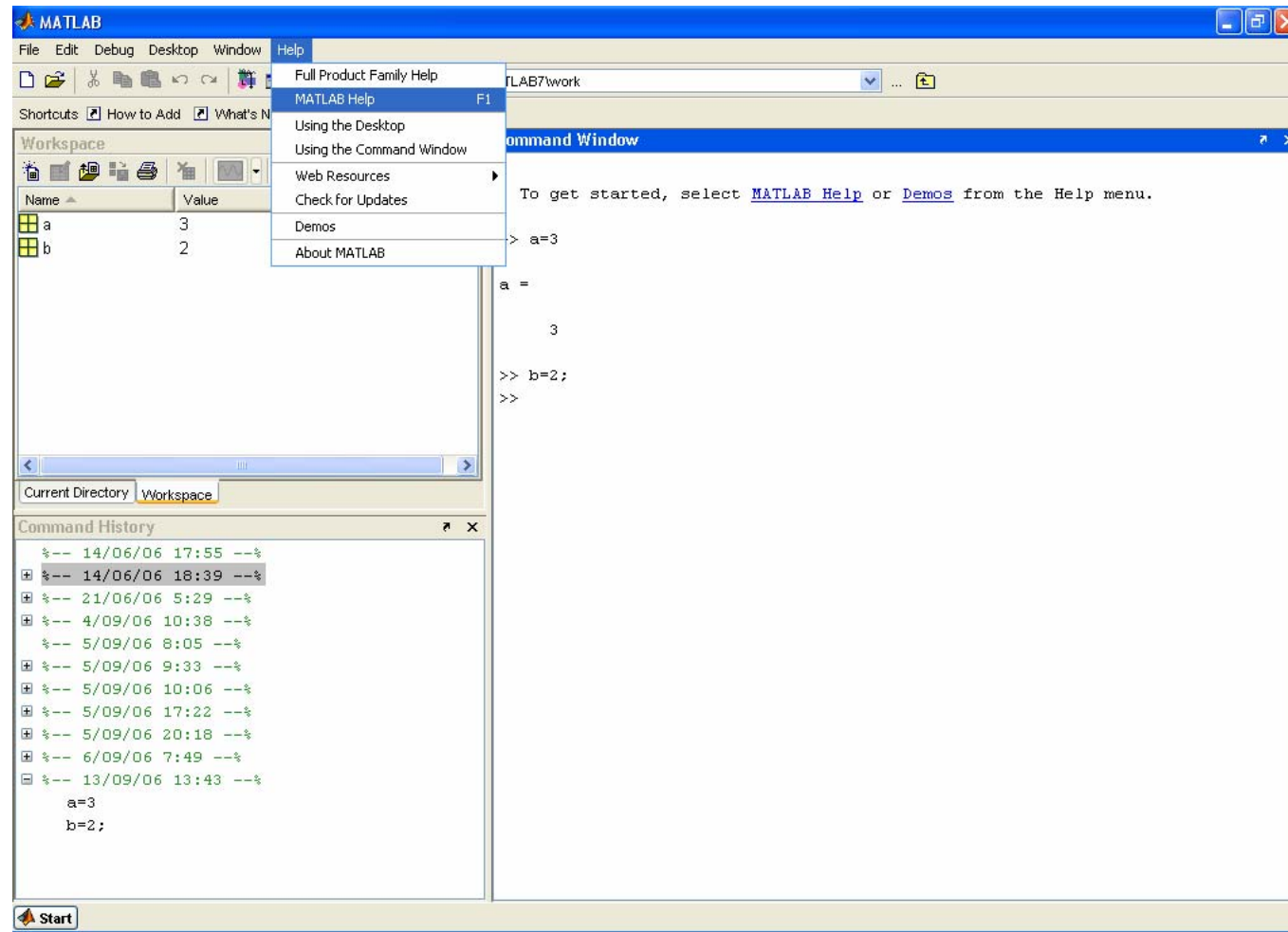
Command Window:

```
>> a=3  
  
a =  
  
    3  
  
>> b=2;  
>>
```

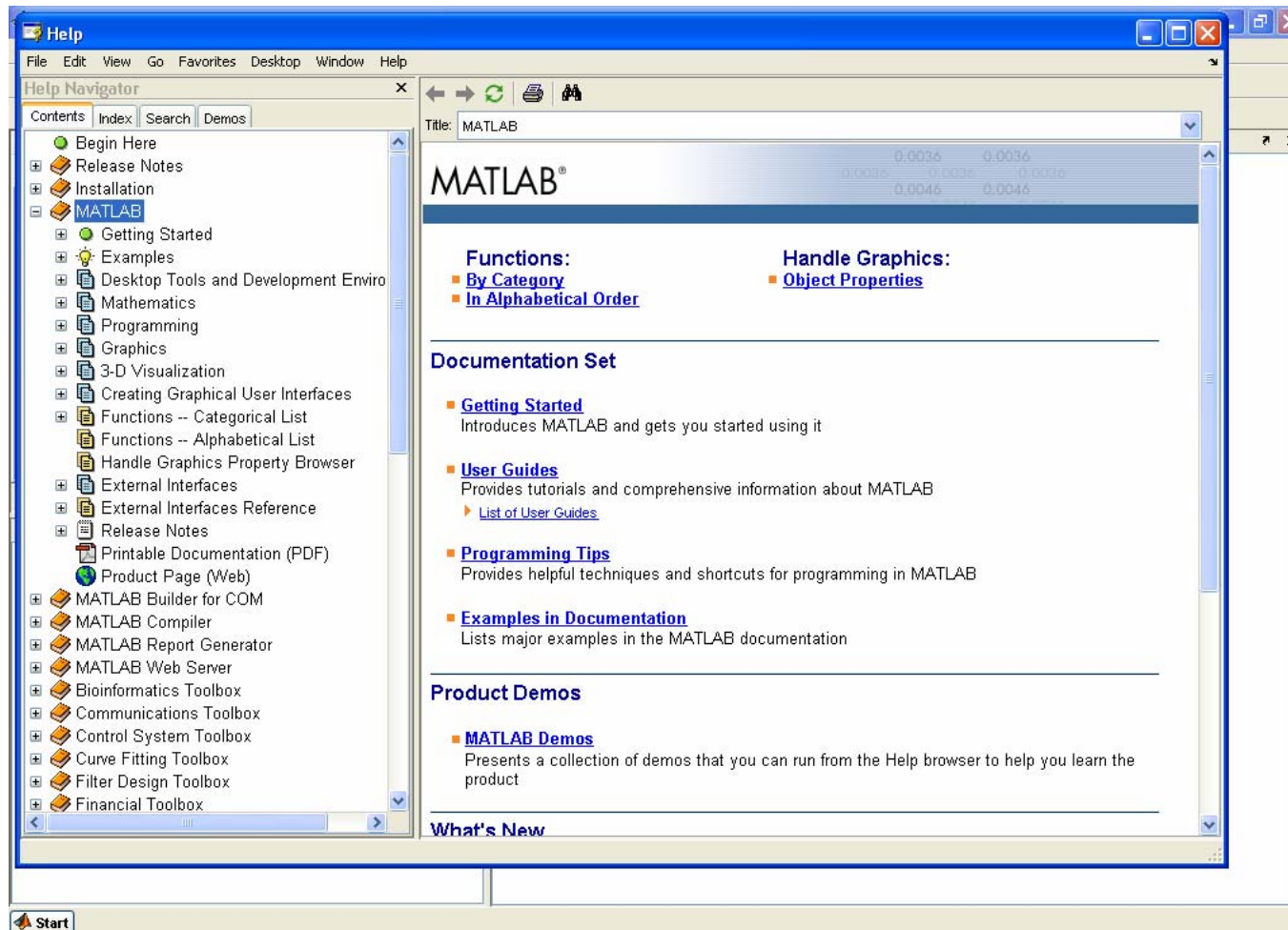
Command History Window:

```
%-- 14/06/06 17:55 --%  
+ %-- 14/06/06 18:39 --%  
+ %-- 21/06/06 5:29 --%  
+ %-- 4/09/06 10:38 --%  
  %-- 5/09/06 8:05 --%  
+ %-- 5/09/06 9:33 --%  
+ %-- 5/09/06 10:06 --%  
  %-- 5/09/06 17:22 --%  
+ %-- 5/09/06 20:18 --%  
+ %-- 6/09/06 7:49 --%  
+ %-- 13/09/06 13:43 --%  
  a=3  
  b=2;
```

Entorno de MATLAB: Ayuda



Entorno de MATLAB: Ayuda



Entorno de MATLAB: Ayuda

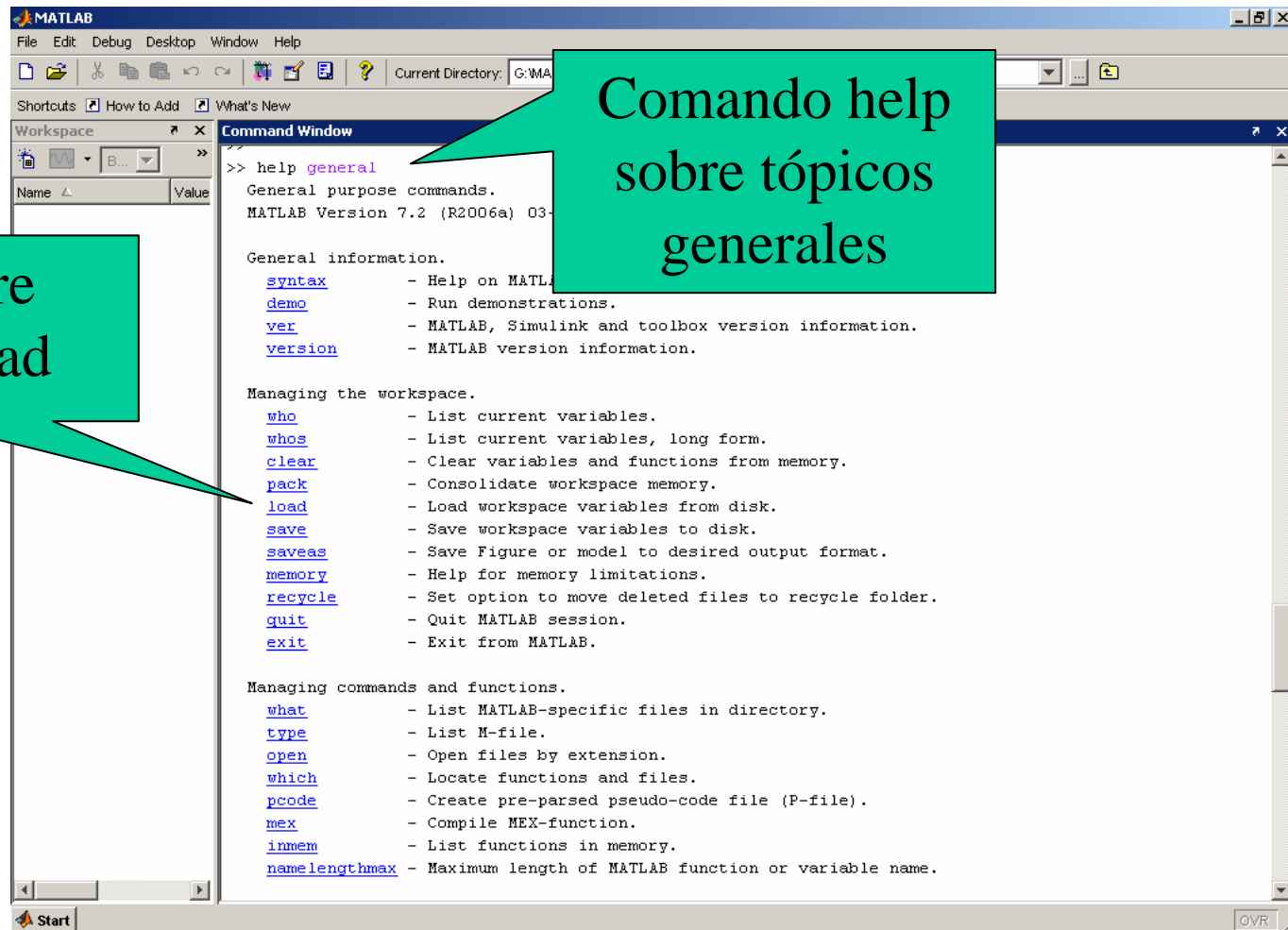
Comando help

Ayuda sobre tópicos generales

```
>> help
HELP topics

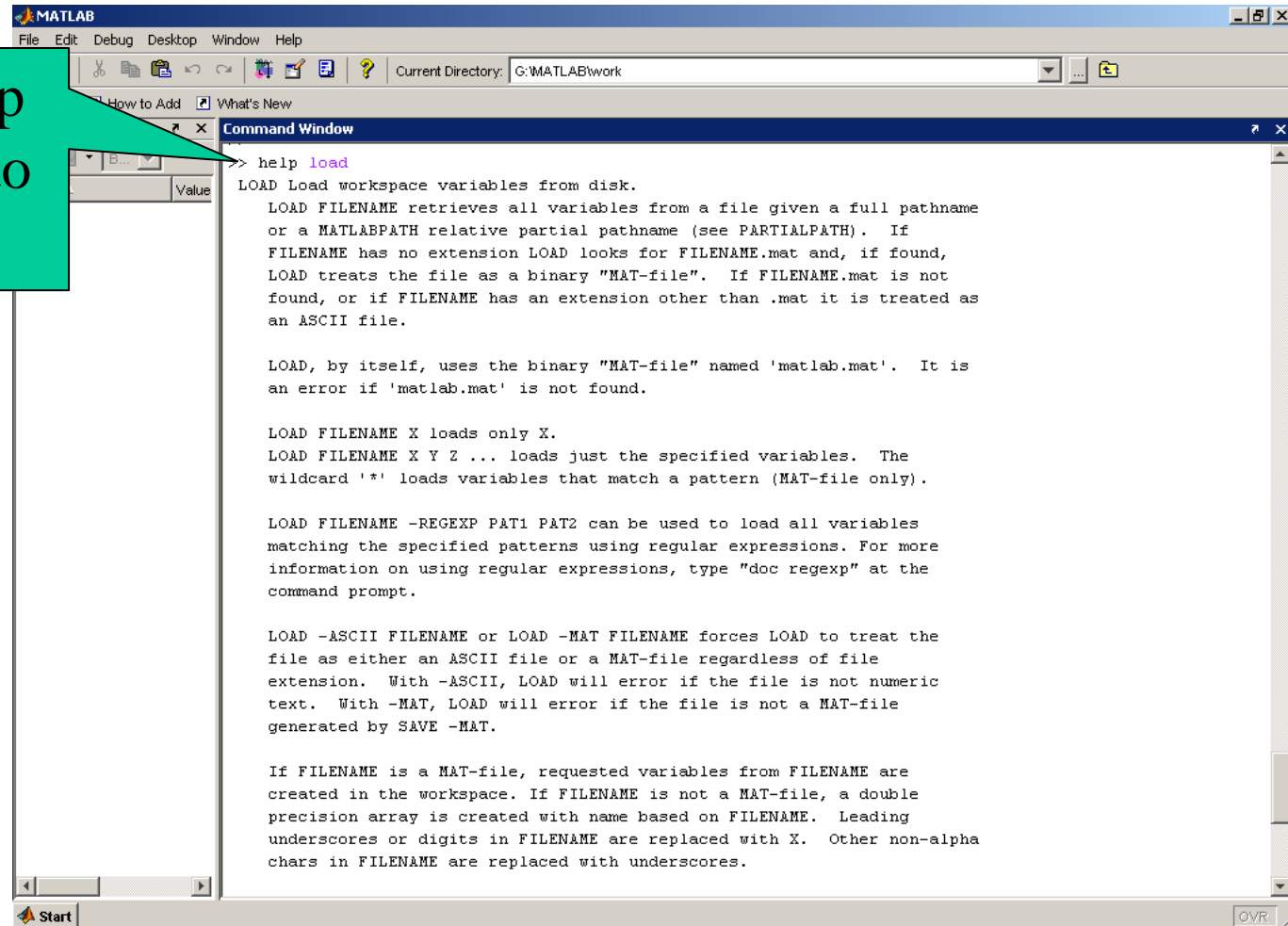
Ceul\MyMatlabStuff - (No table of contents file)
matlab\general - General purpose commands.
matlab\ops - Operators and special characters.
matlab\lang - Programming language constructs.
matlab\elmat - Elementary matrices and matrix manipulation.
matlab\elfun - Elementary math functions.
matlab\specfun - Specialized math functions.
matlab\matfun - Matrix functions - numerical linear algebra.
matlab\datafun - Data analysis and Fourier transforms.
matlab\polyfun - Interpolation and polynomials.
matlab\funfun - Function functions and ODE solvers.
matlab\sparfun - Sparse matrices.
matlab\scribe - Annotation and Plot Editing.
matlab\graph2d - Two dimensional graphs.
matlab\graph3d - Three dimensional graphs.
matlab\specgraph - Specialized graphs.
matlab\graphics - Handle Graphics.
matlab\uitools - Graphical user interface tools.
matlab\strfun - Character strings.
matlab\imagesci - Image and scientific data input/output.
matlab\iofun - File input and output.
matlab\audiovideo - Audio and Video support.
matlab\timefun - Time and dates.
matlab\datatypes - Data types and structures.
matlab\verctrl - Version control.
matlab\codetools - Commands for creating and debugging code.
matlab\helptools - Help commands.
matlab\winfun - Windows Operating System Interface Files (COM/DDE)
matlab\demos - Examples and demonstrations.
matlab\timeseries - Time series data visualization and exploration.
```

Entorno de MATLAB: Ayuda



Entorno de MATLAB: Ayuda

Comando help
sobre comando
load



The image shows a screenshot of the MATLAB Command Window. The window title is 'MATLAB' and it has a menu bar with 'File', 'Edit', 'Debug', 'Desktop', 'Window', and 'Help'. Below the menu bar is a toolbar with various icons. The 'Current Directory' is set to 'G:\MATLAB\work'. The Command Window shows the command 'help load' entered at the prompt. The output text describes the 'LOAD' command, its syntax, and its behavior. A green callout box on the left points to the 'help load' command in the Command Window.

```
>> help load
LOAD Load workspace variables from disk.

LOAD FILENAME retrieves all variables from a file given a full pathname
or a MATLABPATH relative partial pathname (see PARTIALPATH). If
FILENAME has no extension LOAD looks for FILENAME.mat and, if found,
LOAD treats the file as a binary "MAT-file". If FILENAME.mat is not
found, or if FILENAME has an extension other than .mat it is treated as
an ASCII file.

LOAD, by itself, uses the binary "MAT-file" named 'matlab.mat'. It is
an error if 'matlab.mat' is not found.

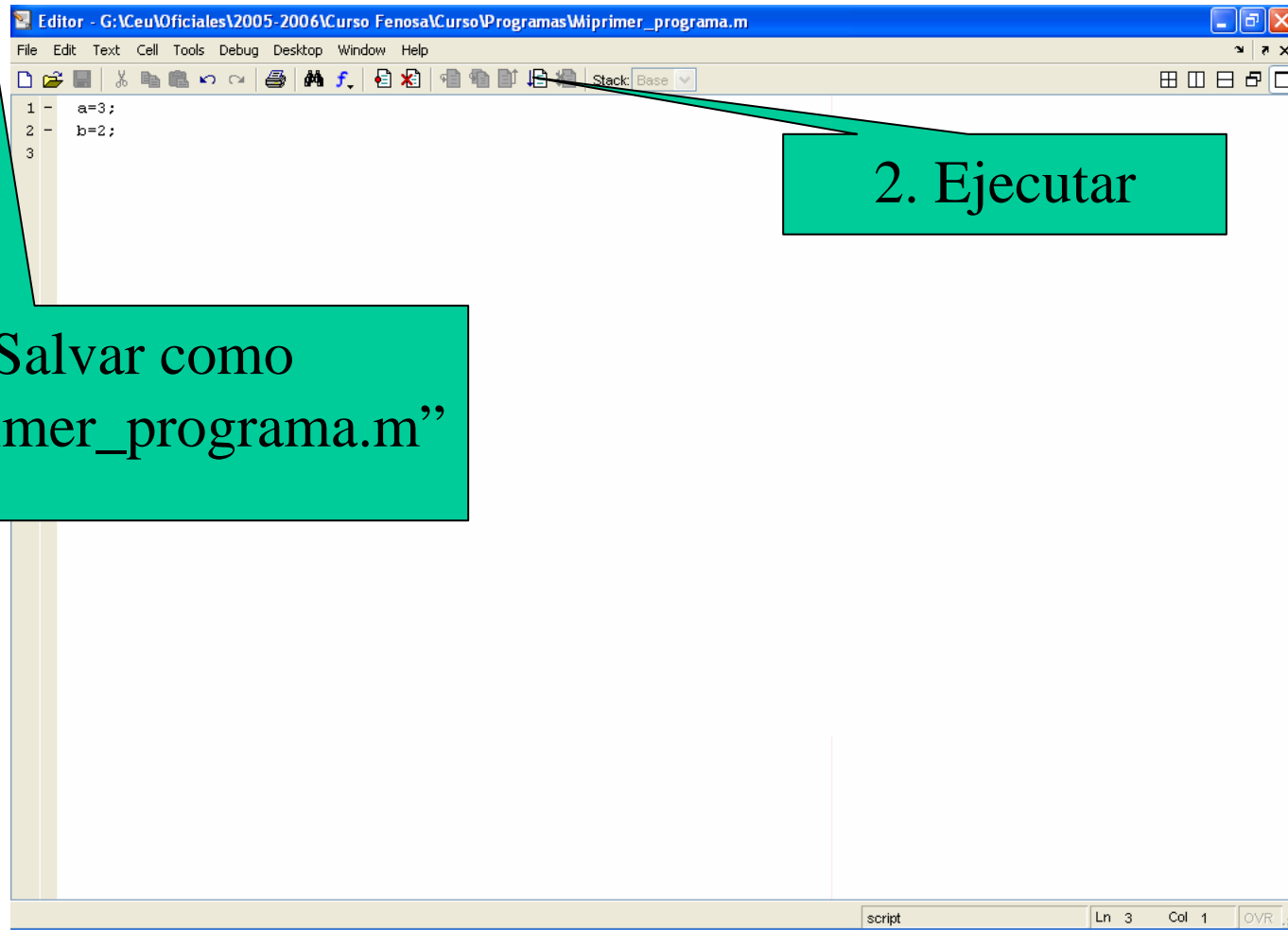
LOAD FILENAME X loads only X.
LOAD FILENAME X Y Z ... loads just the specified variables. The
wildcard '*' loads variables that match a pattern (MAT-file only).

LOAD FILENAME -REGEXP PAT1 PAT2 can be used to load all variables
matching the specified patterns using regular expressions. For more
information on using regular expressions, type "doc regexp" at the
command prompt.

LOAD -ASCII FILENAME or LOAD -MAT FILENAME forces LOAD to treat the
file as either an ASCII file or a MAT-file regardless of file
extension. With -ASCII, LOAD will error if the file is not numeric
text. With -MAT, LOAD will error if the file is not a MAT-file
generated by SAVE -MAT.

If FILENAME is a MAT-file, requested variables from FILENAME are
created in the workspace. If FILENAME is not a MAT-file, a double
precision array is created with name based on FILENAME. Leading
underscores or digits in FILENAME are replaced with X. Other non-alpha
chars in FILENAME are replaced with underscores.
```

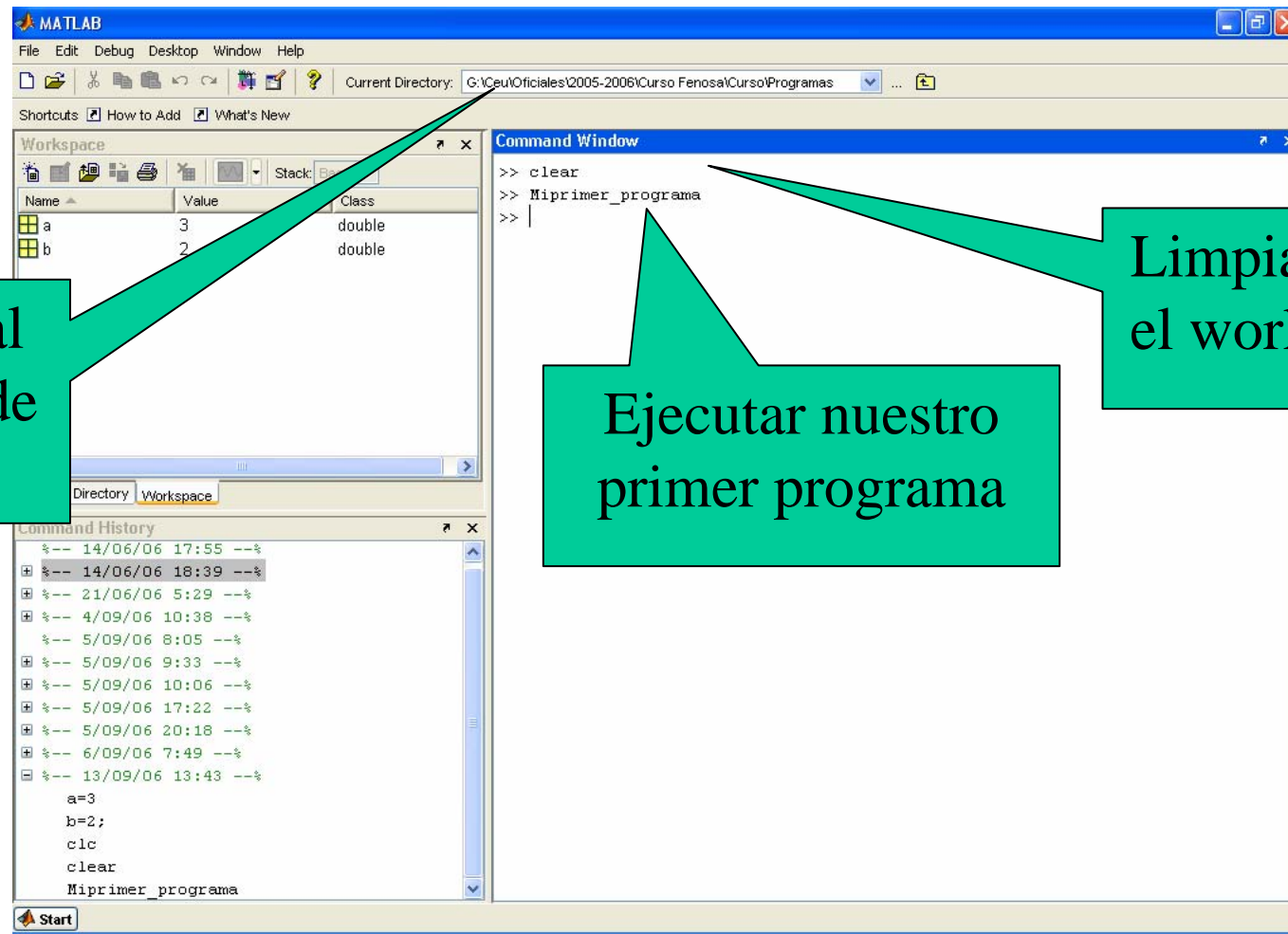

Entorno de MATLAB: El editor



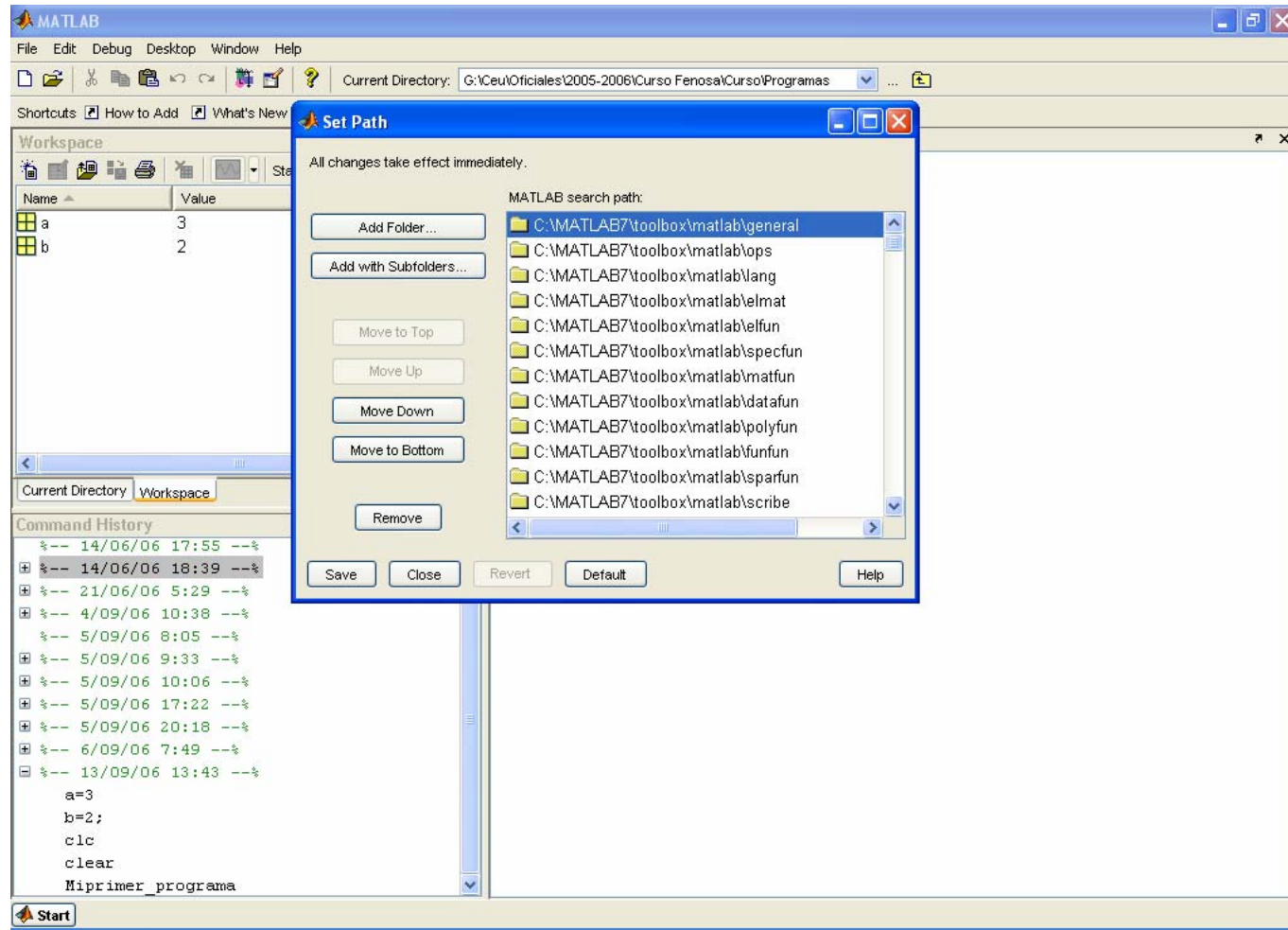
1. Salvar como
“Miprimer_programa.m”

2. Ejecutar

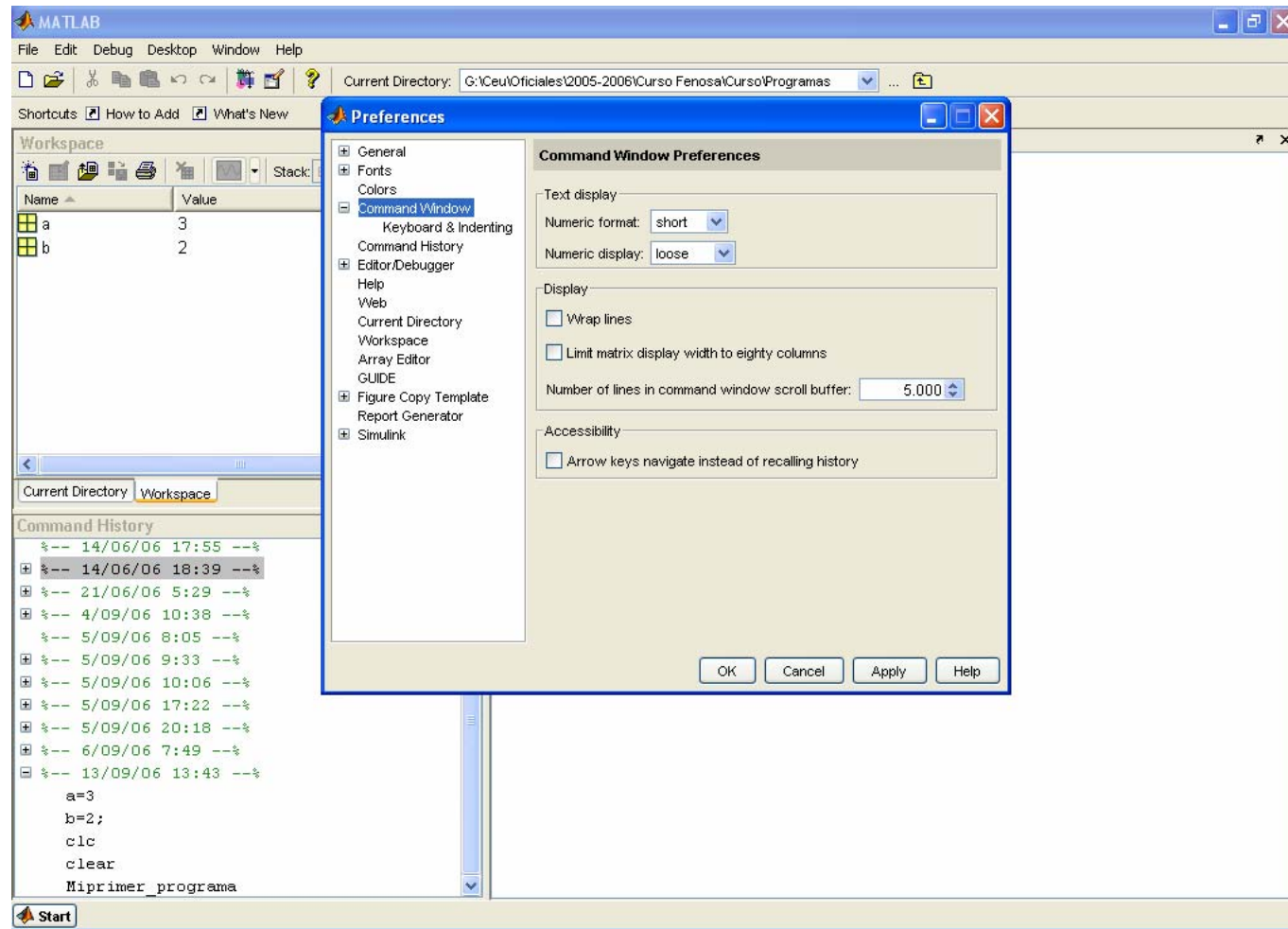
Entorno de MATLAB



Entorno de MATLAB: File -> Set Path



Entorno de MATLAB: File -> Preferences





CEU
*Universidad
San Pablo*

CURSO de UTILIZACIÓN PRÁCTICA de MATLAB

Sesión 1

Carlos Óscar Sánchez Sorzano, Ph.D.
Madrid, July 17th 2006

Cronograma del curso

- **Día 1: Operaciones con matrices y vectores. Funciones de librería.**
- Día 2: Otros tipos de datos en MATLAB. Programación en MATLAB.
- Día 3: Gráficos bidimensionales. Gráficos tridimensionales.
- Día 4: Interfaces de usuario en MATLAB. Generación de programas autónomos
- Día 5: Librerías de interés práctico
- Día 6: Interacción de MATLAB con Office y Visual Basic
- Día 7: Desarrollo de un proyecto

Definición de vectores y matrices

The image shows a MATLAB interface with two main windows: the **Workspace** and the **Command Window**. The **Workspace** window displays a table of variables:

Name	Value	Class
A	[1 2 3; 4 5 6; 7 8 9]	double
B	[1 4 7; 2 5 8; 3 6 9]	double
ans	[1; 4; 7]	double
e1	[1; 0; 0]	double

The **Command Window** shows the following commands and their outputs:

```
>> A=[1 2 3; 4 5 6; 7 8 9]
A =
     1     2     3
     4     5     6
     7     8     9

>> B=A'
B =
     1     4     7
     2     5     8
     3     6     9

>> e1=[1; 0; 0]
e1 =
     1
     0
     0

>> A*e1
ans =
     1
     4
     7
```

Annotations (green boxes) point to specific parts of the interface:

- Definición de la matriz A**: Points to the command `A=[1 2 3; 4 5 6; 7 8 9]`.
- Definición de la matriz B como A transpuesta**: Points to the command `B=A'`.
- Definición de e1 como un vector columna**: Points to the command `e1=[1; 0; 0]`.
- Multiplicación de A por e1 (matriz por vector)**: Points to the command `A*e1`.

A separate green box on the left contains the text: **Si no se asigna a nadie, el resultado siempre se asigna a la variable ans**. This points to the `ans` variable in the workspace and the `ans` output in the command window.

Definición de vectores y matrices

The image shows a MATLAB interface with two main windows: 'Workspace' and 'Command Window'. The 'Workspace' window displays variables 'ans', 'e1', and 'e2' with their values and classes. The 'Command Window' shows the commands entered and the resulting output.

Workspace:

Name	Value	Class
ans	0	double
e1	[1;0;0]	double
e2	[0 1 0]	double

Command Window:

```
>> e1=[1; 0; 0]

e1 =

     1
     0
     0

>> e2=[0 1 0]

e2 =

     0     1     0

>> e1*e2

ans =

     0     1     0
     0     0     0
     0     0     0

>> e2*e1

ans =

     0

>> e1'*e2
??? Error using ==> mtimes
Inner matrix dimensions must agree.
```

Annotations:

- Definición de e1 como vector columna (3x1)**: Points to the command `e1=[1; 0; 0]`.
- Definición de e2 como vector fila (1x3)**: Points to the command `e2=[0 1 0]`.
- (3x1)x(1x3)=3x3**: Points to the command `e1*e2`.
- (1x3)x(3x1)=1x1**: Points to the command `e2*e1`.
- (1x3)x(1x3)=Error**: Points to the command `e1'*e2`.

Operaciones con vectores y matrices

The image shows a MATLAB window with the following components:

- Workspace:** A table listing variables A, B, and ans, all of type 'double'.
- Command Window:** Contains the following commands and output:

```
>> clear
>> A=[1 1 1; 1 1 1; 1 1 1];
>> B=[2 2 2; 2 2 2; 2 2 2];
>> A*B

ans =

     6     6     6
     6     6     6
     6     6     6

>> A.*B

ans =

     2     2     2
     2     2     2
     2     2     2

>>
```

Three green callout boxes with black text point to specific parts of the Command Window:

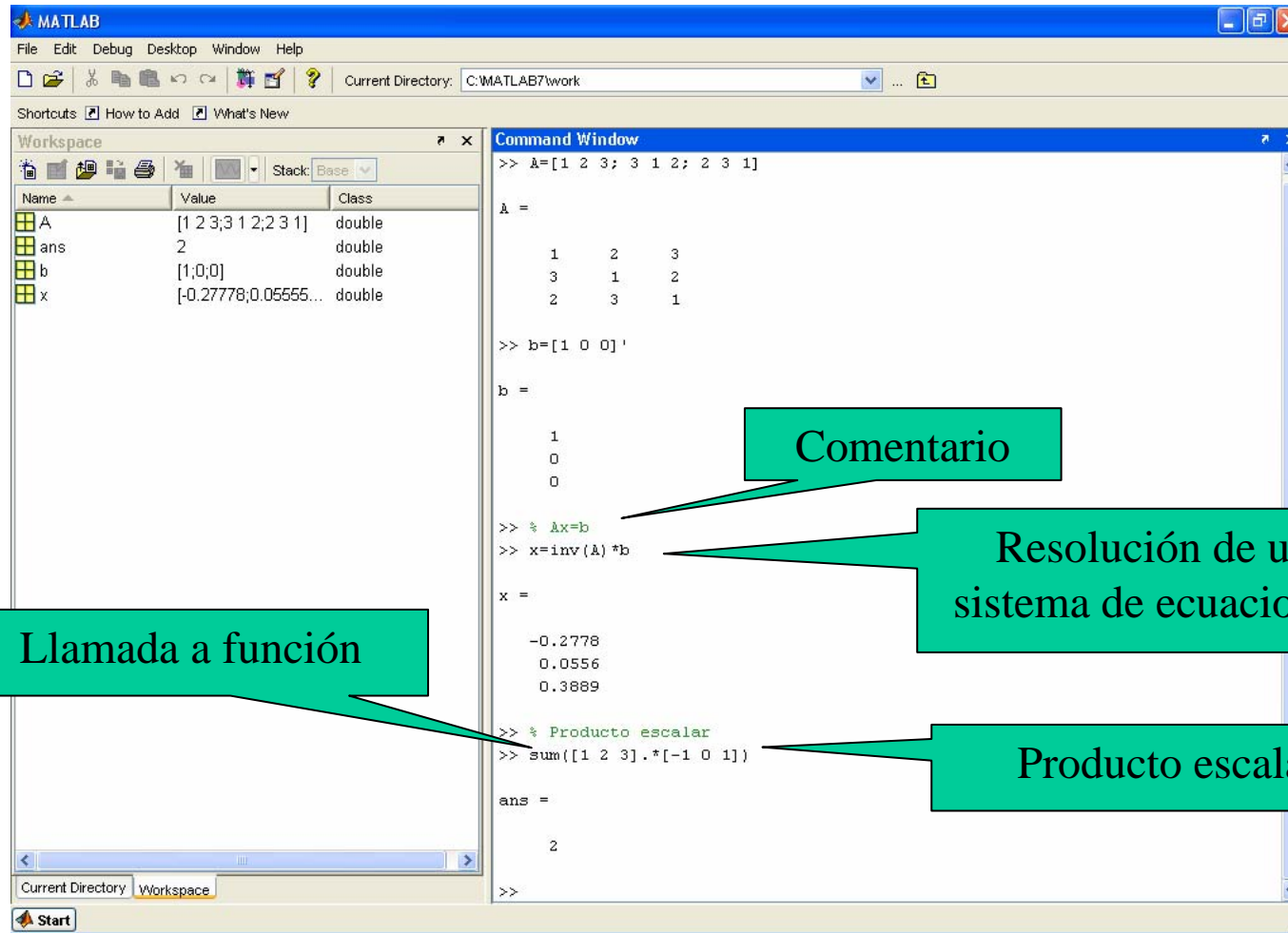
- No muestra el resultado:** Points to the first three lines of code (clear, A=[...], B=[...]).
- Multiplicación matricial:** Points to the output of the `A*B` command.
- Multiplicación elemento a elemento:** Points to the output of the `A.*B` command.

Name	Value	Class
A	[1 1 1; 1 1 1; 1 1 1]	double
B	[2 2 2; 2 2 2; 2 2 2]	double
ans	[2 2 2; 2 2 2; 2 2 2]	double

Operaciones con vectores y matrices

- + adición o suma
- – sustracción o resta
- * multiplicación
- ' traspuesta
- ^ potenciación
- \ división-izquierda
- / división-derecha
- .* producto elemento a elemento
- ./ y .\ división elemento a elemento
- .^ elevar a una potencia elemento a elemento

Un poco de álgebra



The image shows the MATLAB interface with the following content:

Workspace:

Name	Value	Class
A	[1 2 3; 3 1 2; 2 3 1]	double
ans	2	double
b	[1; 0; 0]	double
x	[-0.2778; 0.0556; 0.3889]	double

Command Window:

```
>> A=[1 2 3; 3 1 2; 2 3 1]
A =
     1     2     3
     3     1     2
     2     3     1

>> b=[1 0 0]';
b =
     1
     0
     0

>> % Ax=b
>> x=inv(A)*b
x =
    -0.2778
     0.0556
     0.3889

>> % Producto escalar
>> sum([1 2 3].*[-1 0 1])

ans =
     2

>>
```

Callouts:

- Llamada a función:** Points to the line `x=inv(A)*b` in the Command Window.
- Comentario:** Points to the line `% Ax=b` in the Command Window.
- Resolución de un sistema de ecuaciones:** Points to the line `x=inv(A)*b` in the Command Window.
- Producto escalar:** Points to the line `sum([1 2 3].*[-1 0 1])` in the Command Window.

Definición de vectores y matrices

The image shows the MATLAB interface with the following components:

- Workspace:** A table listing variables in the workspace.
- Command Window:** A text area showing MATLAB commands and their outputs.
- Callouts:** Four green boxes with arrows pointing to specific parts of the Command Window output.

Name	Value	Class
A	[0 0 0; 0 0 0; 0 0 0]	double
a	[0 0 0 0 0; 0 0 0 0 0]	double

```
>> A=eye(3)

A =

     1     0     0
     0     1     0
     0     0     1

>> A=ones(3)

A =

     1     1     1
     1     1     1
     1     1     1

>> A=zeros(3)

A =

     0     0     0
     0     0     0
     0     0     0

>> a=zeros(2,5)

a =

     0     0     0     0     0
     0     0     0     0     0

>>
```

Callout 1: Matriz identidad (points to the output of `A=eye(3)`)

Callout 2: Matriz cuadrada de 1s (points to the output of `A=ones(3)`)

Callout 3: Matriz cuadrada de 0s (points to the output of `A=zeros(3)`)

Callout 4: Matriz no cuadrada de 0s (points to the output of `a=zeros(2,5)`)

Definición de vectores y matrices

The image shows the MATLAB interface with the Workspace and Command Window panes. The Workspace pane displays two variables: 'A' with value [1 1 1 1] and class 'double', and 'a' with value [10 8 6 4 2] and class 'double'. The Command Window shows the following commands and outputs:

```
>> a=[1:10]
a =
     1     2     3     4     5     6     7     8     9    10

>> a=[1:3:10]
a =
     1     4     7    10

>> a=[1:3:11]
a =
     1     4     7    10

>> a=[10:-2:1]
a =
    10     8     6     4     2

>> A=ones(size(a))
A =
     1     1     1     1     1

>>
```

Annotations in green boxes point to specific parts of the Command Window output:

- Vector de 1 a 10** points to the output of `a=[1:10]`.
- Vector de 1 a 10 cada 3** points to the output of `a=[1:3:10]`.
- Vector de 1 a 11 cada 3** points to the output of `a=[1:3:11]`.
- Vector 10 a 2 cada 2** points to the output of `a=[10:-2:1]`.
- Matriz de 1s con la forma de otra matriz** points to the output of `A=ones(size(a))`.

Definición de vectores y matrices

The image shows a MATLAB interface with two main windows: the **Workspace** and the **Command Window**.

Workspace:

Name	Value	Class
A	[1 0 0; 0 1 0; 0 0 1]	double
B	<6x6 double>	double
ans	[1 1 1 0.17464 2.1...]	double

Command Window:

```
>> A=eye(3)

A =

     1     0     0
     0     1     0
     0     0     1

>> B=[A flipud(A); rand(3) randn(3)]

B =

     1.0000         0         0         0         0         1.0000
         0     1.0000         0         0     1.0000         0
         0         0     1.0000     1.0000         0         0
     0.4447     0.9218     0.4057     0.1746    -0.5883     0.1139
     0.6154     0.7382     0.9355    -0.1867     2.1832     1.0668
     0.7919     0.1763     0.9169     0.7258    -0.1364     0.0593

>> reshape(A,1,9)

ans =

     1     0     0     0     1     0     0     0     1

>> diag(B) '

ans =

     1.0000     1.0000     1.0000     0.1746     2.1832     0.0593

>>
```

Annotations:

- Rand: distrib. Uniforme** (points to `rand(3)`)
- Randn: distrib. Normal** (points to `randn(3)`)
- Formación de una matriz a partir de submatrices** (points to the construction of matrix `B`)
- Reestructuración de una matriz** (points to `reshape(A,1,9)`)
- Una función que opera sobre matrices** (points to `diag(B)'`)

Indexación de matrices

The MATLAB interface shows the following commands and results in the Command Window:

```
>> A=[1 2 3; 4 5 6; 7 8 9]
```

A =

1	2	3
4	5	6
7	8	9

The matrix A is represented as $\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$.

```
>> A(1,1)
```

ans =

1

```
>> A(1,3)
```

ans =

3

```
>> A(3,1)
```

ans =

7

```
>> A(1:2,1:2)
```

ans =

1	2
4	5

The submatrix A(1:2,1:2) is represented as $\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$.

Workspace:

Name	Value	Class
A	[1 2 3; 4 5 6; 7 8 9]	double
ans	[1 2; 4 5]	double

Extracción de un elemento

Extracción de una submatriz

Indexación de matrices

The image shows the MATLAB interface with the following components:

- Workspace:** A table showing variables in the workspace.
- Command Window:** A text area for entering MATLAB commands and displaying their output.

Name	Value	Class
A	[1 2 3; 4 5 6; 7 8 9]	double
ans	[1; 4; 7]	double

Command Window Output:

```
>> A=[1 2 3; 4 5 6; 7 8 9]

A =

     1     2     3
     4     5     6
     7     8     9

>> A(1,:)

ans =

     1     2     3

>> A(:,1)

ans =

     1
     4
     7

>> A(:,1)=ones(3,1)

A =

     1     2     3
     1     5     6
     1     8     9

>>
```

Callouts:

- Extracción de una fila:** Points to the command `A(1,:)` and its output.
- Extracción de una columna:** Points to the command `A(:,1)` and its output.
- Asignación de una columna:** Points to the command `A(:,1)=ones(3,1)` and the resulting matrix.

Indexación de matrices

The image shows a MATLAB interface with a Workspace window and a Command Window. The Workspace window displays two variables: 'a' with value [1 2 3 4 5] and class 'double', and 'idx' with value [4 5] and class 'double'. The Command Window shows a series of commands and their outputs, illustrating different indexing techniques. Four callout boxes provide explanations for specific commands:

- Acceso a una coordenada no existente** (Access to a non-existent coordinate): Points to the command `a(5)=7`, which updates the 5th element of array 'a' to 7.
- Acceso a varios índices al mismo tiempo** (Access to multiple indices at the same time): Points to the command `a(idx)=0`, which sets the elements at indices 4 and 5 to 0.
- Selección de los índices por una condición** (Selection of indices by a condition): Points to the command `idx=find(a>3)`, which finds the indices of elements greater than 3.
- Acceso a varios índices al mismo tiempo** (Access to multiple indices at the same time): Points to the command `a(idx)=[1 3 5]`, which assigns the values 1, 3, and 5 to the elements at the indices stored in 'idx'.

The Command Window output shows the state of array 'a' after each operation:

```
>> a=[1 2 3 4]
a =
     1     2     3     4

>> a(5)=7
a =
     1     2     3     4     7

>> idx=[1 3 5];
>> a(idx)=0
a =
     0     2     0     4     0

>> a(idx)=[1 3 5]
a =
     1     2     3     4     5

>> idx=find(a>3)
idx =
     4     5

>>
```

Indexación de matrices

The image shows a MATLAB interface with a Workspace window and a Command Window. The Workspace window displays two variables: 'A' (5x5 double) and 'ans' (1x5 double). The Command Window shows the following commands and outputs:

```
>> A=magic(5)
A =
    17    24     1     8    15
    23     5     7    14    16
     4     6    13    20    22
    10    12    19    21     3
    11    18    25     2     9

>> sum(A)
ans =
    65    65    65    65    65

>> sum(A')
ans =
    65    65    65    65    65

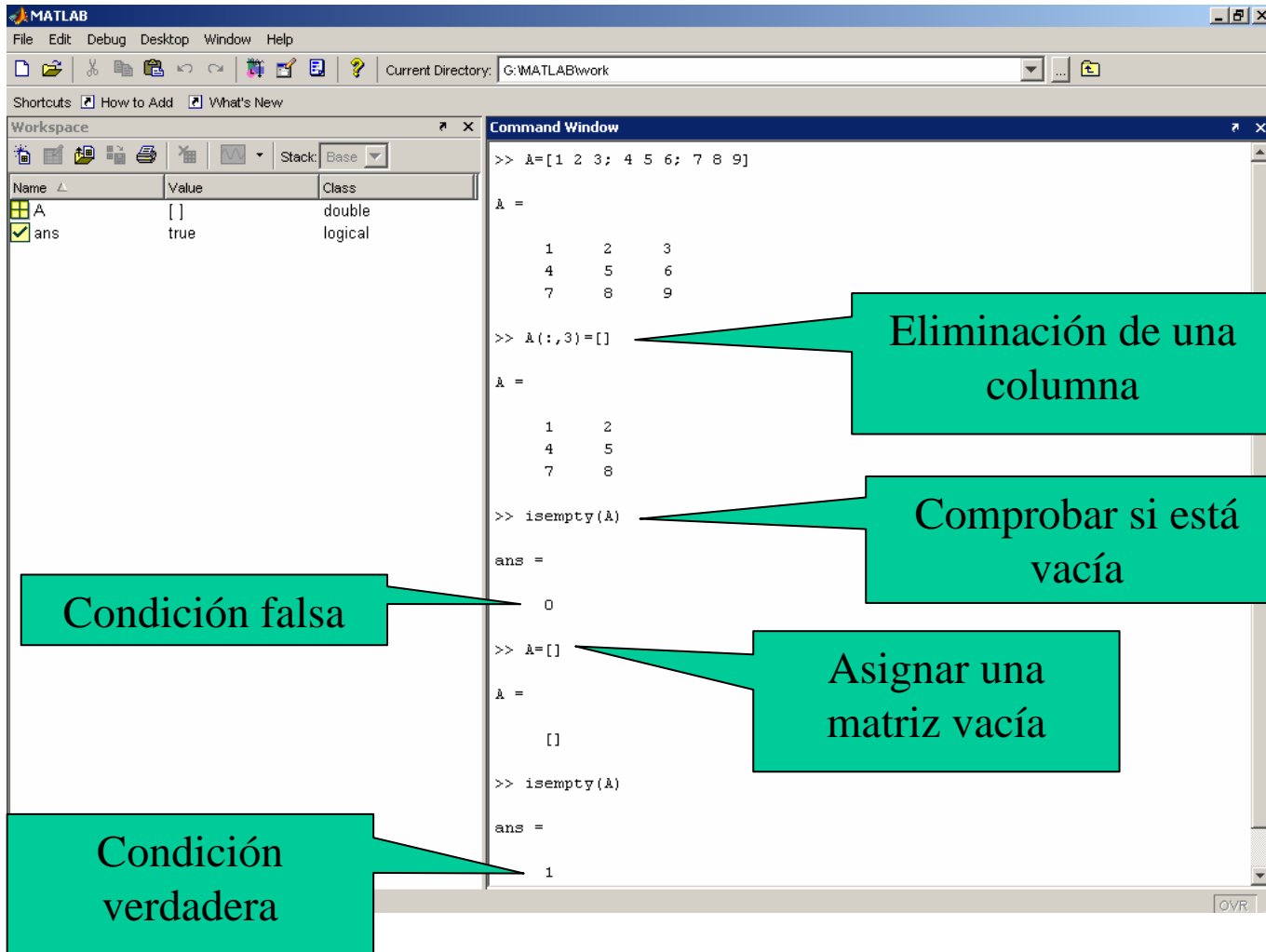
>> A([1 5], [1 3 5])
ans =
    17     1    15
    11    25     9

>> |
```

Callouts from the image:

- Matriz tipo “Sudoku” (points to the magic(5) command)
- Suma por columnas (points to the sum(A) command)
- Suma por filas (points to the sum(A') command)
- Selección de las filas 1 y 5 y columnas 1,3,5 (points to the A([1 5], [1 3 5]) command)

Indexación de matrices



The image shows a MATLAB Command Window with the following code and output:

```
>> A=[1 2 3; 4 5 6; 7 8 9]
A =
     1     2     3
     4     5     6
     7     8     9

>> A(:,3)=[]
A =
     1     2
     4     5
     7     8

>> isempty(A)
ans =
     0

>> A=[]
A =
     []

>> isempty(A)
ans =
     1
```

Annotations in the image:

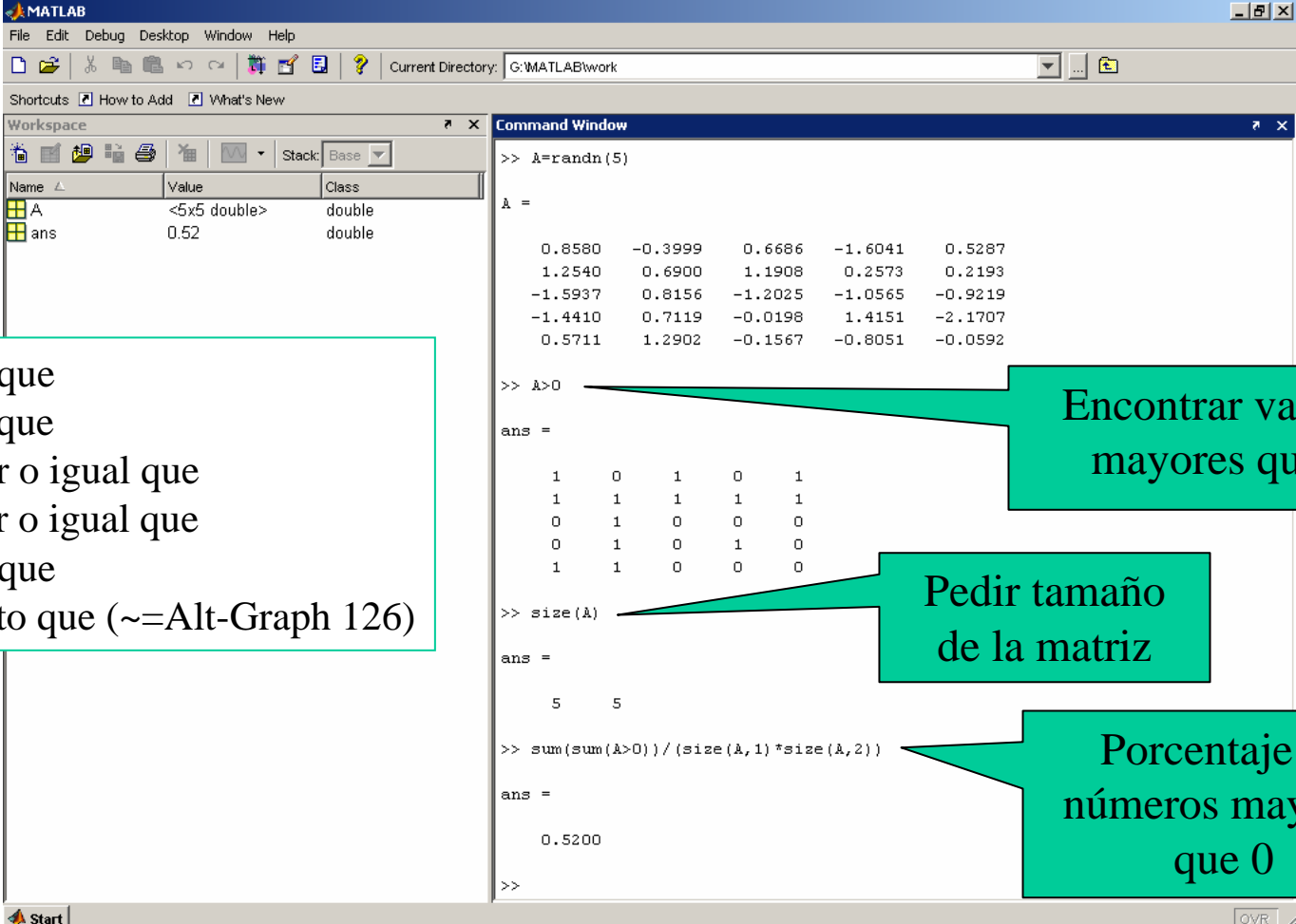
- Eliminación de una columna**: Points to the command `A(:,3)=[]`.
- Comprobar si está vacía**: Points to the command `isempty(A)` when `A` is non-empty.
- Condición falsa**: Points to the output `ans = 0`.
- Asignar una matriz vacía**: Points to the command `A=[]`.
- Condición verdadera**: Points to the output `ans = 1`.

The Workspace window on the left shows the following variables:

Name	Value	Class
A	[]	double
ans	true	logical

Condiciones

< menor que
> mayor que
<= menor o igual que
>= mayor o igual que
== igual que
~= distinto que (~=Alt-Graph 126)



The MATLAB interface shows the following steps in the Command Window:

```
>> A=randn(5)

A =

    0.8580   -0.3999    0.6686   -1.6041    0.5287
    1.2540    0.6900    1.1908    0.2573    0.2193
   -1.5937    0.8156   -1.2025   -1.0565   -0.9219
   -1.4410    0.7119   -0.0198    1.4151   -2.1707
    0.5711    1.2902   -0.1567   -0.8051   -0.0592

>> A>0

ans =

     1     0     1     0     1
     1     1     1     1     1
     0     1     0     0     0
     0     1     0     1     0
     1     1     0     0     0

>> size(A)

ans =

     5     5

>> sum(sum(A>0))/(size(A,1)*size(A,2))

ans =

    0.5200

>>
```

Callout boxes explain the steps:

- Encontrar valores mayores que 0**: Points to the `A>0` command.
- Pedir tamaño de la matriz**: Points to the `size(A)` command.
- Porcentaje de números mayores que 0**: Points to the `sum(sum(A>0))/(size(A,1)*size(A,2))` command.

The Workspace window shows the following variables:

Name	Value	Class
A	<5x5 double>	double
ans	0.52	double

Condiciones

The image shows a MATLAB interface with two main windows: 'Workspace' and 'Command Window'.

Workspace:

Name	Value	Class
A	[-0.234 1.4435 0.7...]	double
ans	<3x3 logical>	logical

Command Window:

```
>> A=randn(3)

A =
   -0.2340    1.4435    0.7990
    0.1184   -0.3510    0.9409
    0.3148    0.6232   -0.9921

>> A>0

ans =
     0     1     1
     1     0     1
     1     1     0

>> A<0

ans =
     1     0     0
     0     1     0
     0     0     1

>> (A<0) | (A>0)

ans =
     1     1     1
     1     1     1
     1     1     1

>>
```

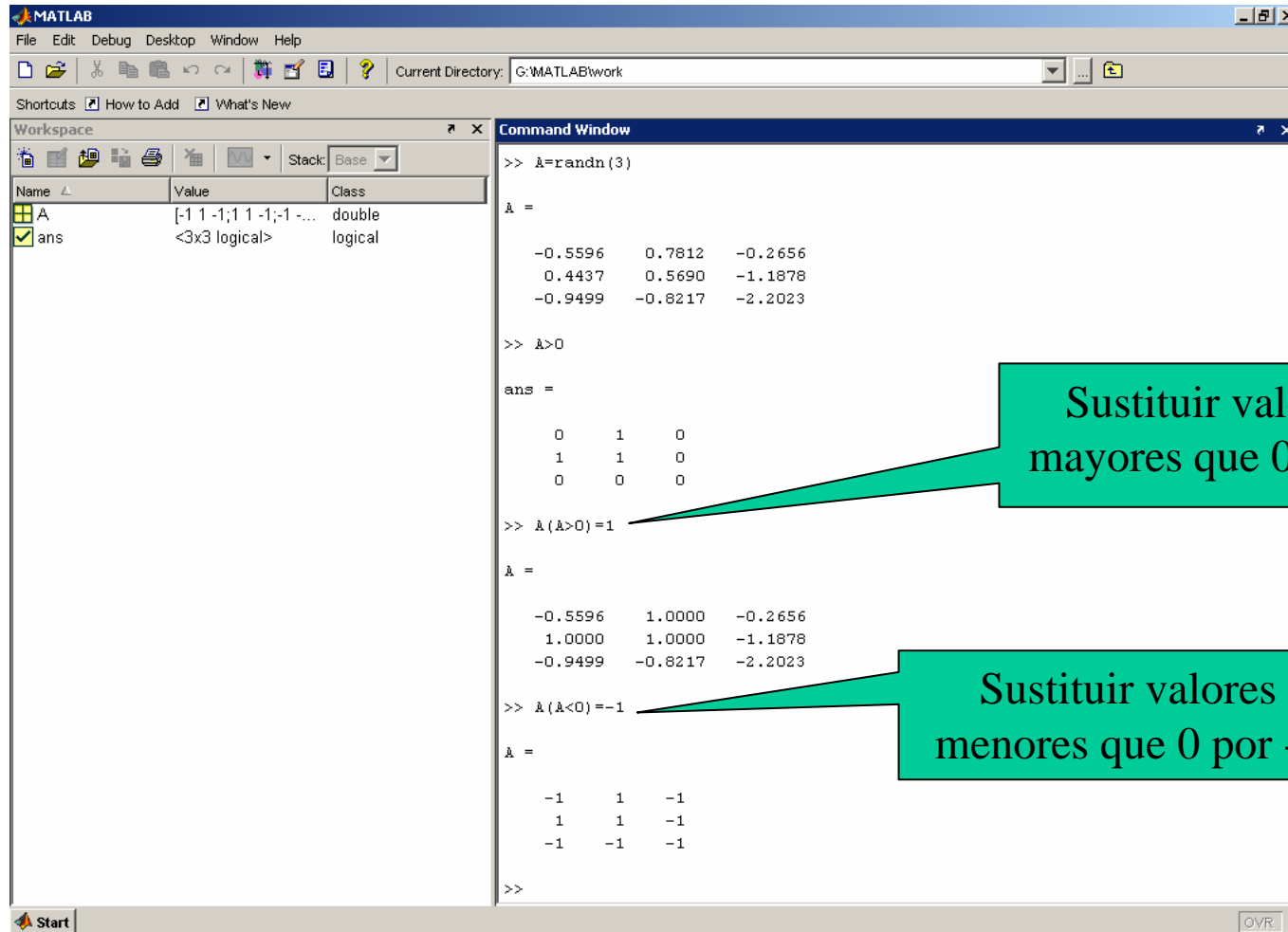
Annotations:

- Diferentes tipos de datos:** Points to the Workspace window, highlighting the different data types (double and logical).
- Encontrar valores mayores que 0:** Points to the command window output for `A>0`.
- Encontrar valores menores que 0:** Points to the command window output for `A<0`.
- Valores menores que 0 o mayores que 0:** Points to the command window output for `(A<0) | (A>0)`.

Logical Operators:

- `&` and
- `|` or
- `~` negación lógica

Condiciones e indexación



The image shows a MATLAB window with the following content:

Workspace:

Name	Value	Class
A	[-1 1 -1; 1 1 -1; -1 -1 -1]	double
ans	<3x3 logical>	logical

Command Window:

```
>> A=randn(3)

A =

-0.5596    0.7812   -0.2656
 0.4437    0.5690   -1.1878
-0.9499   -0.8217   -2.2023

>> A>0

ans =

     0     1     0
     1     1     0
     0     0     0

>> A(A>0)=1

A =

-0.5596    1.0000   -0.2656
 1.0000    1.0000   -1.1878
-0.9499   -0.8217   -2.2023

>> A(A<0)=-1

A =

-1     1    -1
 1     1    -1
-1    -1    -1

>>
```

Two callout boxes provide explanations for the logical operations:

- Sustituir valores mayores que 0 por 1** (Replace values greater than 0 by 1)
- Sustituir valores menores que 0 por -1** (Replace values less than 0 by -1)

Funciones

The image shows a MATLAB interface with a Command Window and a Workspace. The Command Window contains the following code and output:

```
>> x=sin(0)
x =
    0

>> x=sin([0 pi/8 2*pi/8 3*pi/8 4*pi/8 5*pi/8])
x =
    0    0.3827    0.7071    0.9239    1.0000    0.9239

>> x=sin([0 pi/8 ; 2*pi/8 3*pi/8 ; 4*pi/8 5*pi/8])
x =
    0    0.3827
    0.7071    0.9239
    1.0000    0.9239

>> max_por_columnas=max(x)
max_por_columnas =
    1.0000    0.9239

>> [max_por_columnas, indice_max]=max(x);
>> indice_max
indice_max =
     3     2
```

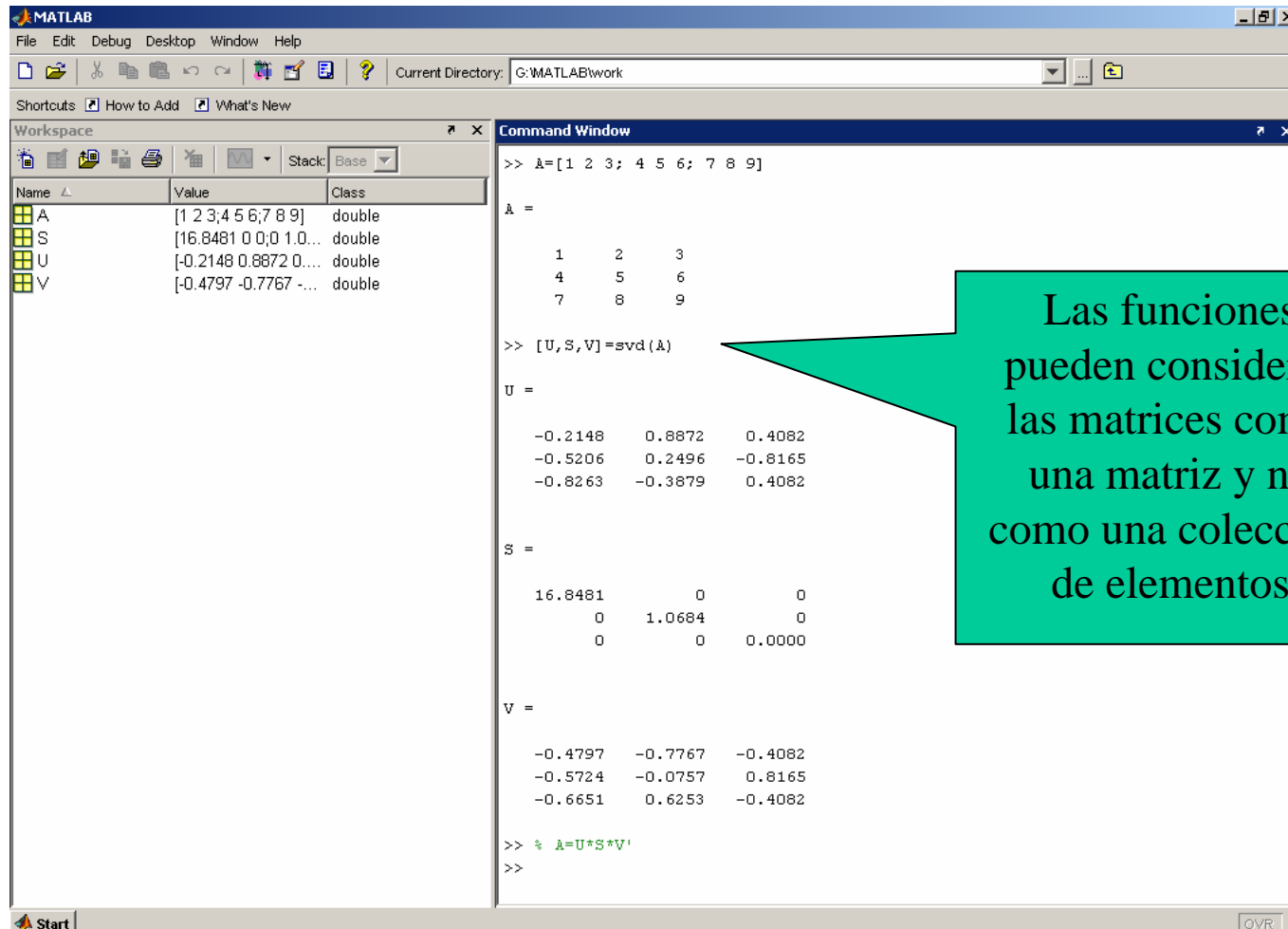
The Workspace shows the following variables:

Name	Value	Class
indice_max	[3 2]	double
max_por_colu...	[1 0.9239]	double
x	[0 0.3827; 0.7071 ...]	double

Callouts explain the function usage:

- Función aplicada a un escalar (points to `x=sin(0)`)
- Función aplicada a un vector (points to `x=sin([0 pi/8 2*pi/8 3*pi/8 4*pi/8 5*pi/8])`)
- Función aplicada a una matriz (points to `x=sin([0 pi/8 ; 2*pi/8 3*pi/8 ; 4*pi/8 5*pi/8])`)
- Función con un argumento de retorno (points to `max_por_columnas=max(x)`)
- Función con dos argumentos de retorno (points to `[max_por_columnas, indice_max]=max(x);`)

Funciones



The image shows the MATLAB environment. The workspace contains four variables: A (a 3x9 double matrix), S (a 3x9 double matrix), U (a 3x9 double matrix), and V (a 3x9 double matrix). The Command Window shows the following commands and outputs:

```
>> A=[1 2 3; 4 5 6; 7 8 9]

A =

     1     2     3
     4     5     6
     7     8     9

>> [U,S,V]=svd(A)

U =

    -0.2148    0.6872    0.4082
    -0.5206    0.2496   -0.8165
    -0.8263   -0.3879    0.4082

S =

    16.8481         0         0
         0     1.0684         0
         0         0     0.0000

V =

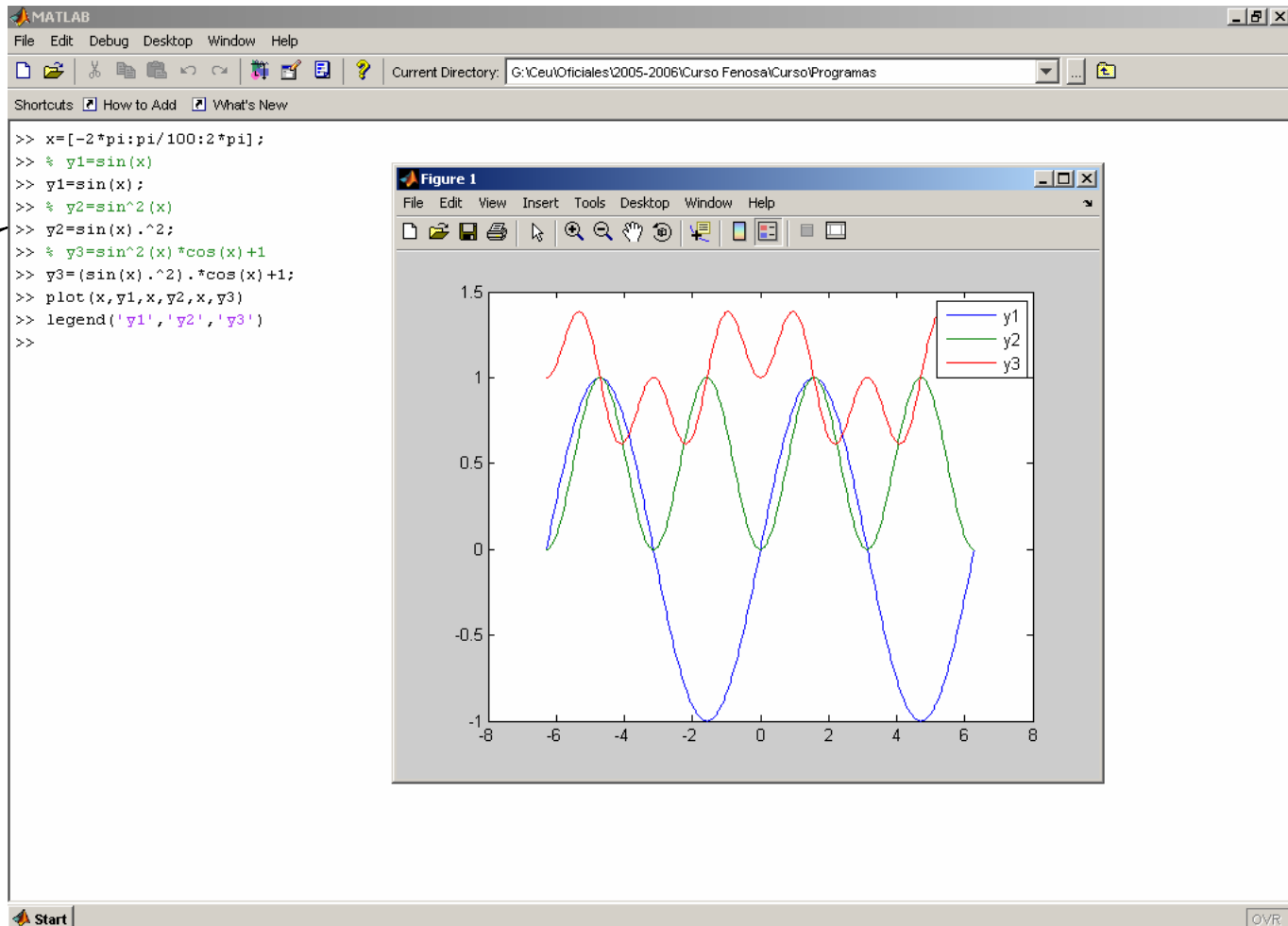
    -0.4797   -0.7767   -0.4082
    -0.5724   -0.0757    0.8165
    -0.6651    0.6253   -0.4082

>> % A=U*S*V'
>>
```

A green callout box points to the SVD calculation, containing the text: "Las funciones pueden considerar las matrices como una matriz y no como una colección de elementos".

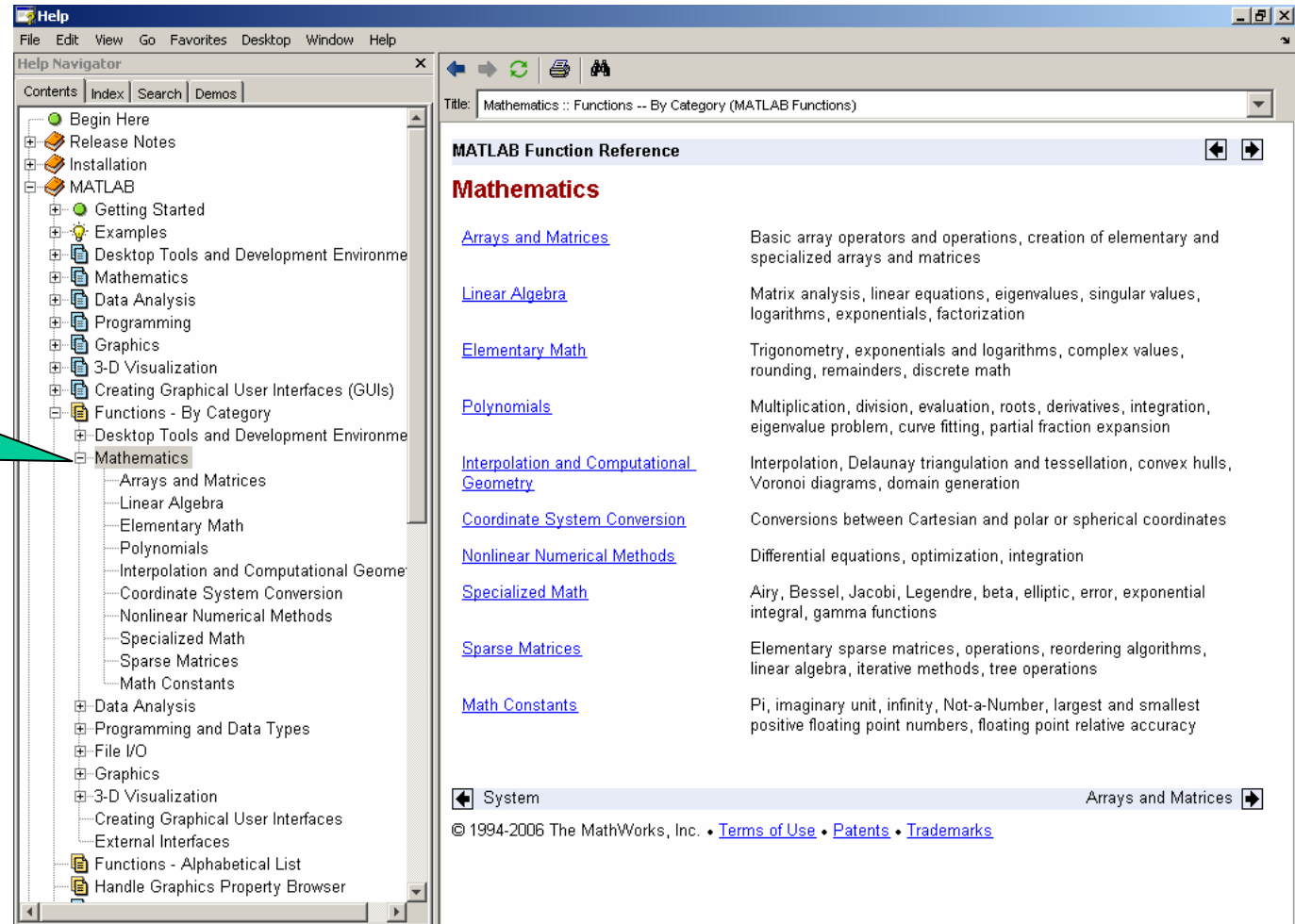
Funciones

Operaciones
elemento a
elemento



Funciones

Explorar las
funciones
matemáticas



Entrada/Salida

The image shows the MATLAB interface with the following components:

- Current Directory:** G:\Ceu\Oficiales\2005-2006\Curso Fenosa\Curso\Programas
- File List:**

All Files	File Type	Size	Last Modified
Evolucion demanda.txt	TXT File	4 KB	13-sep-06
Evolucion poblacion.txt	TXT File	1 KB	14-sep-06
Miprimer_programa.m	M-file	1 KB	13-sep-06
Mi_workspace.mat	MAT-file	1 KB	14-sep-06
Mi_matriz_A.mat	MAT-file	1 KB	14-sep-06
Mi_matriz_A.txt	TXT File	1 KB	14-sep-06
- Workspace:**

Name	Value	Class
A	[1 2 3; 4 5 6; 7 8 9]	double
S	[16.8481 0 0; 0 1.0...	double
U	[-0.2148 0.8872 0....	double
V	[-0.4797 -0.7767	double
- Command Window:**

```
>> A=[1 2 3; 4 5 6; 7 8 9];
>> [U,S,V]=svd(A);
>> whos
Name      Size      Bytes  Class
A         3x3         72  double array
S         3x3         72  double array
U         3x3         72  double array
V         3x3         72  double array

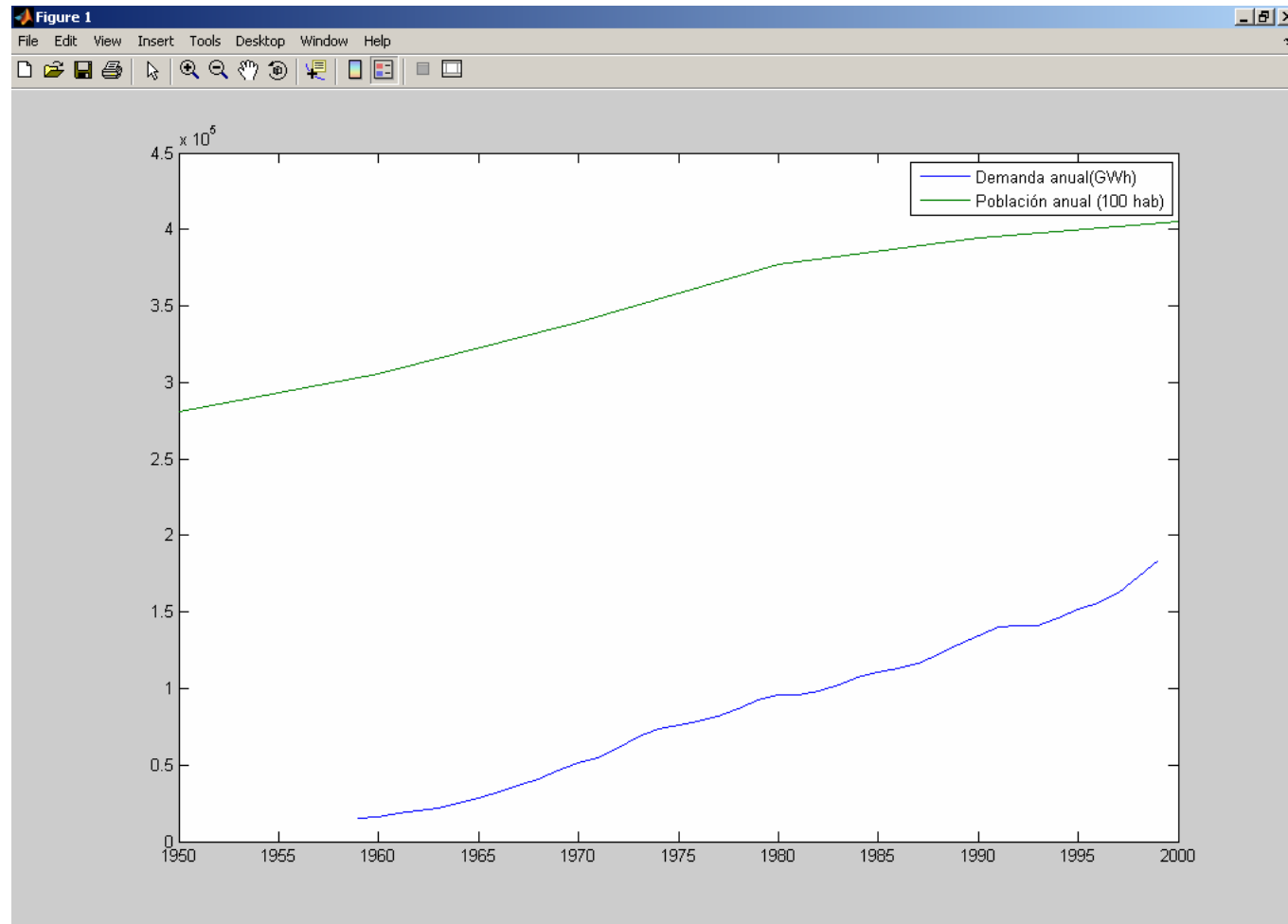
Grand total is 36 elements using 288 bytes

>> save 'Mi_workspace'
>> clear
>> whos
>> load 'Mi_workspace'
>> save 'Mi_matriz_A' A
>> save('Mi_matriz_A.txt','A','-ascii')
>>
```

Callouts:

- Salvar todo el workspace:** Points to the `save 'Mi_workspace'` command.
- Cargar todo el workspace:** Points to the `load 'Mi_workspace'` command.
- Salvar sólo una o varias variables:** Points to the `save 'Mi_matriz_A' A` and `save('Mi_matriz_A.txt','A','-ascii')` commands.
- Salvar una o varias variables en otro formato:** Points to the `save('Mi_matriz_A.txt','A','-ascii')` command.

Ejercicio final



Ejercicio final

1. Cargar los datos de evolución de la demanda eléctrica y la población española desde 1950 a 2000
 1. 'Evolucion demanda.txt' (Fuente: Red Eléctrica de España):
 1. Columna 1: Año
 2. Columna 2-13: Demanda mensual (GWh)
 3. Columna 14: Demanda anual (GWh)
 2. 'Evolucion poblacion.txt' (Fuente: Instituto Nacional de Estadística):
 1. Columna 1: Año
 2. Columna 2: Población (Miles de habitantes)
2. Mostrar en un mismo gráfico ambas tendencias de forma anual
(Sugerencia: puede que haga falta escalar alguna de las series)
3. Poner ambas series en la misma base de tiempos (1959-1999)
(Sugerencia: buscar en la ayuda las funciones de interpolación de MATLAB)



CEU
*Universidad
San Pablo*

CURSO de UTILIZACIÓN PRÁCTICA de MATLAB

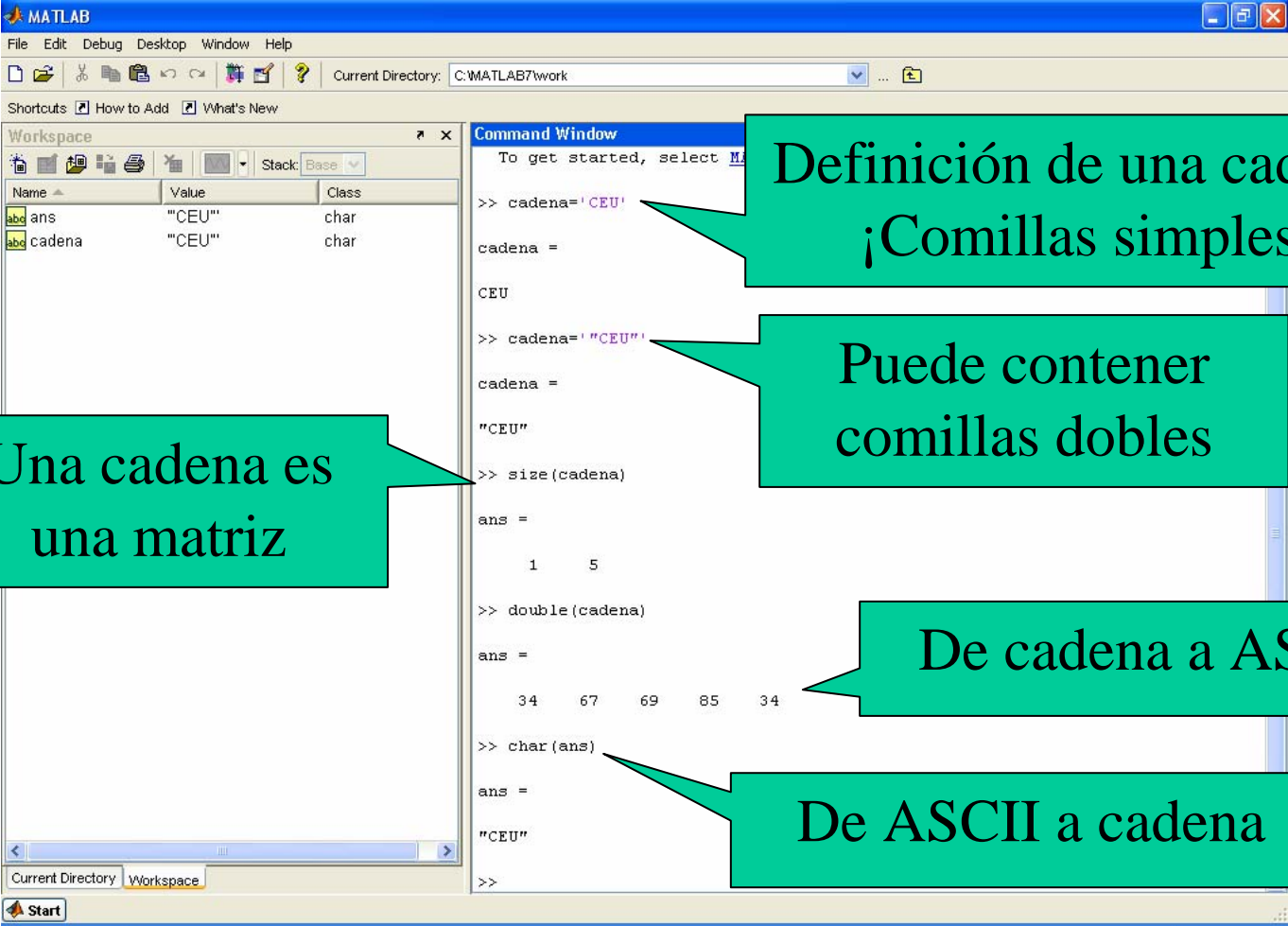
Sesión 2

Carlos Óscar Sánchez Sorzano, Ph.D.
Madrid, July 17th 2006

Cronograma del curso

- Día 1: Operaciones con matrices y vectores. Funciones de librería.
- **Día 2: Otros tipos de datos en MATLAB. Programación en MATLAB.**
- Día 3: Gráficos bidimensionales. Gráficos tridimensionales.
- Día 4: Interfaces de usuario en MATLAB. Generación de programas autónomos
- Día 5: Librerías de interés práctico
- Día 6: Interacción de MATLAB con Office y Visual Basic
- Día 7: Desarrollo de un proyecto

Tipos de datos: Cadenas



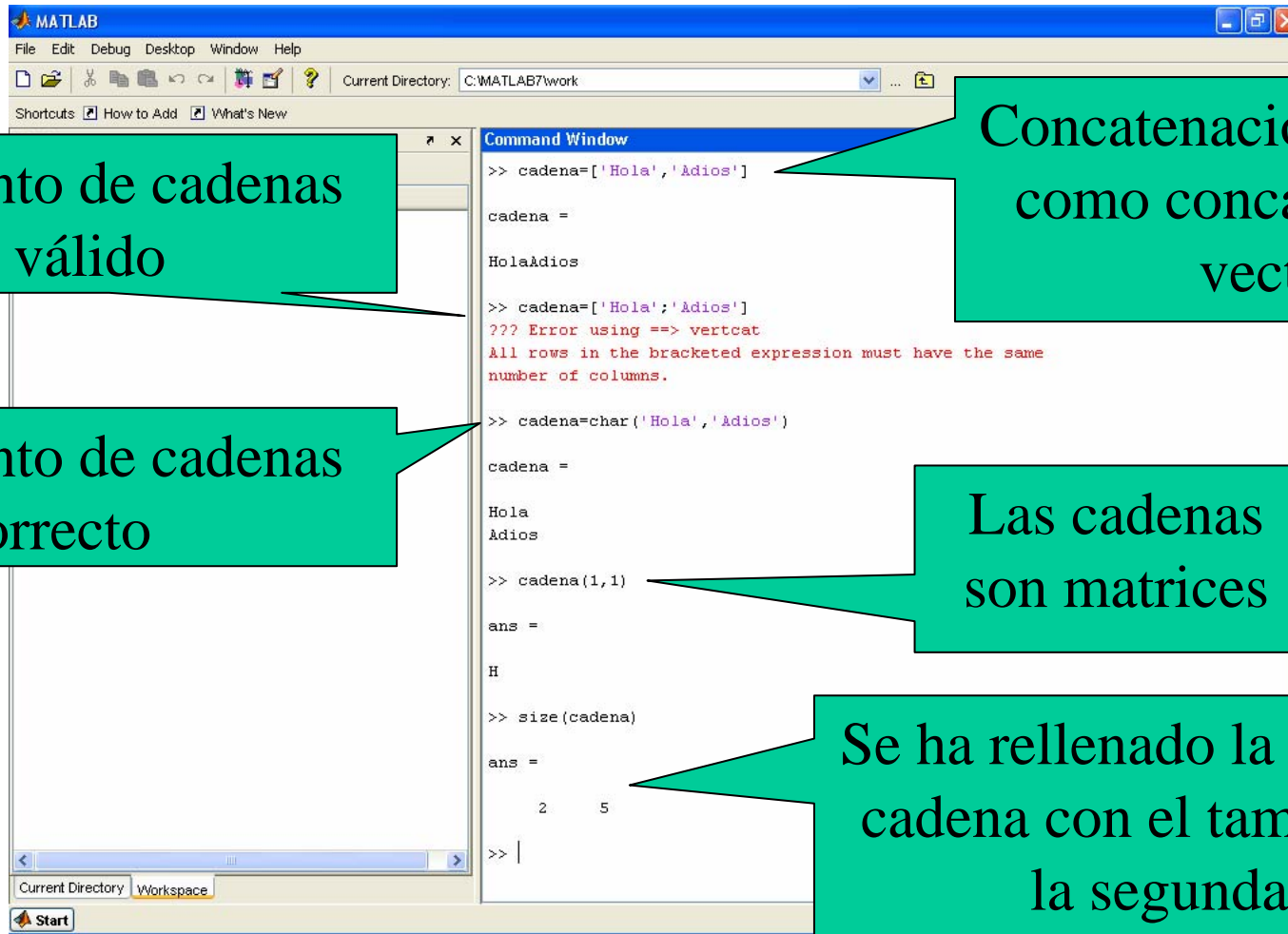
The image shows the MATLAB interface with the Command Window and Workspace. The Command Window contains the following code and output:

```
>> cadena='CEU'
cadena =
CEU
>> cadena="'CEU'"
cadena =
"CEU"
>> size(cadena)
ans =
     1     5
>> double(cadena)
ans =
    34    67    69    85    34
>> char(ans)
ans =
"CEU"
>>
```

Callouts provide additional information:

- Definición de una cadena. ¡Comillas simples!
- Puede contener comillas dobles
- De cadena a ASCII
- De ASCII a cadena
- Una cadena es una matriz

Tipos de datos: Cadenas



```
MATLAB
File Edit Debug Desktop Window Help
Current Directory: C:\MATLAB7\work
Shortcuts How to Add What's New

Command Window

>> cadena=['Hola','Adios']

cadena =

HolaAdios

>> cadena=['Hola';'Adios']
??? Error using ==> vertcat
All rows in the bracketed expression must have the same
number of columns.

>> cadena=char('Hola','Adios')

cadena =

Hola
Adios

>> cadena(1,1)

ans =

H

>> size(cadena)

ans =

     2     5

>> |
```

Apilamiento de cadenas
no válido

Apilamiento de cadenas
correcto

Concatenación de cadenas
como concatenación de
vectores

Las cadenas
son matrices

Se ha rellenado la primera
cadena con el tamaño de
la segunda

Tipos de datos: Cadenas

The image shows a MATLAB Command Window with several lines of code and their outputs. Five green callout boxes with black text point to specific parts of the code:

- Imprimir en una cadena** points to the string `'Voy a imprimir pi '` in the `disp` command.
- Imprimir en consola** points to the `disp` command itself.
- Extraer token** points to the `strtok` function call.
- Reemplazar** points to the `strrep` function call.
- Buscar** points to the `findstr` function call.

```
>> disp(['Voy a imprimir pi ' num2str(pi)])
Voy a imprimir pi 3.1416
>> sprintf('Voy a imprimir pi %f\n',pi)

ans =

Voy a imprimir pi 3.141593

>> [p,r]=strtok(['Voy a imprimir pi ' num2str(pi,10)])

p =

Voy

r =

a imprimir pi 3.141592654

>> r=strrep(r,'r','R')

r =

a impRimiR pi 3.141592654

>> idx=findstr(r,'pi')

idx =

13

>>
```

Workspace variables shown at the bottom left:

Variable	Value	Type
p	'Voy'	char
r	'a impRimiR pi 3....'	char

Tipos de datos: Hipermatrices

The image shows a MATLAB interface with a workspace and a command window. The workspace displays variables A, B, C, and ans. The command window shows the following code and output:

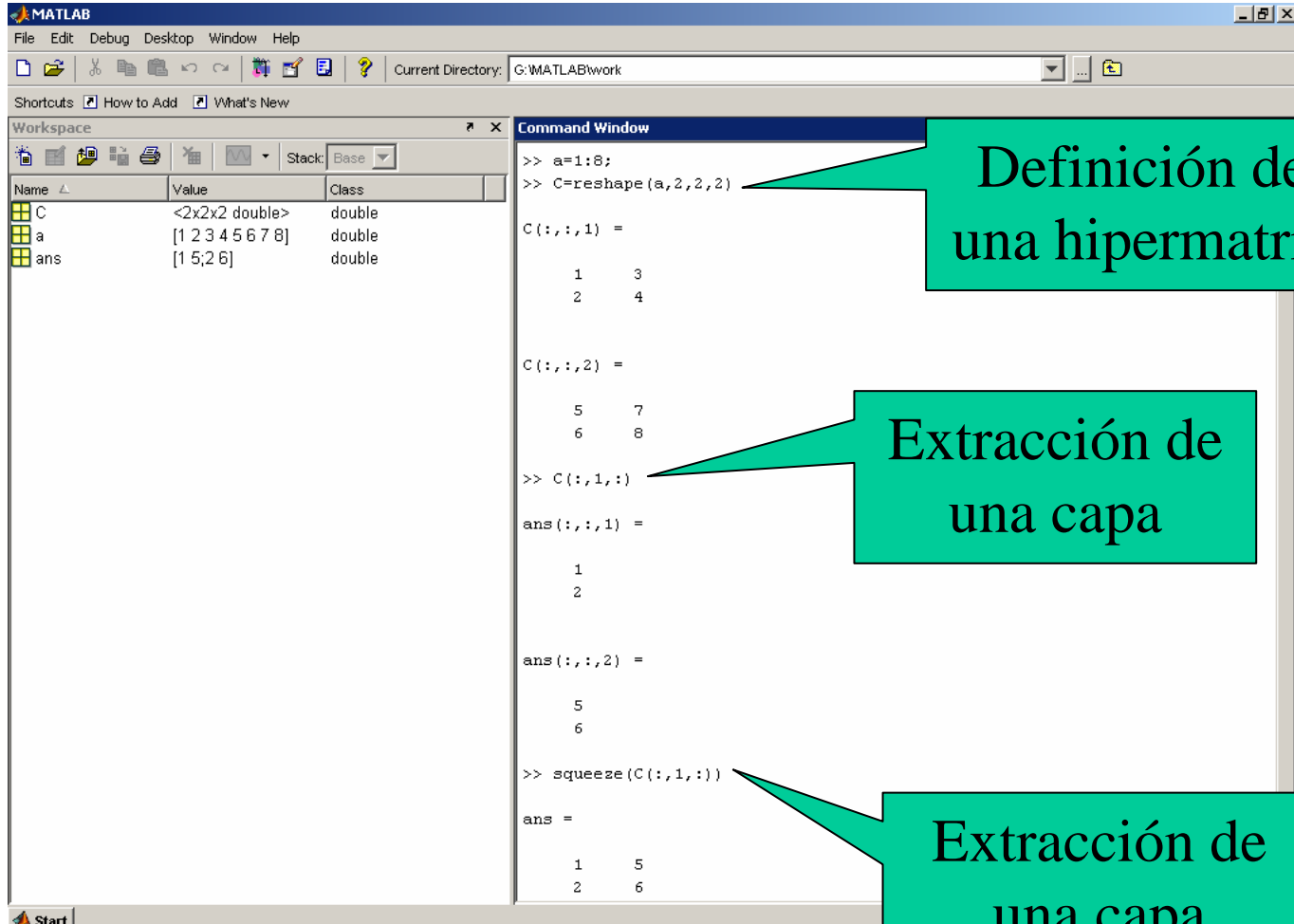
```
>> A=[1 2; 3 4];  
>> B=[5 6; 7 8];  
>> C(:,:,1)=A;  
>> C(:,:,2)=B;  
>> C  
  
C(:,:,1) =  
  
     1     2  
     3     4  
  
C(:,:,2) =  
  
     5     6  
     7     8  
  
>> C(2,2,2)  
  
ans =  
  
     8  
  
>> C=cats(3,A,B)  
  
C(:,:,1) =  
  
     1     2  
     3     4  
  
C(:,:,2) =  
  
     5     6  
     7     8
```

Diagram illustrating the structure of a hypermatrix $A(i,j,k)$ with dimensions i , j , and k .

Callouts:

- Definición de una hipermatriz
- Acceso a una hipermatriz
- Definición de una hipermatriz

Tipos de datos: Hipermatrices



The image shows a MATLAB interface with a workspace and a command window. The workspace contains three variables: C (2x2x2 double), a (1x8 double), and ans (1x2 double). The command window shows the following code and output:

```
>> a=1:8;  
>> C=reshape(a,2,2,2)  
  
C(:,:,1) =  
  
     1     3  
     2     4  
  
C(:,:,2) =  
  
     5     7  
     6     8  
  
>> C(:,1,:) =  
  
ans(:,:,1) =  
  
     1  
     2  
  
ans(:,:,2) =  
  
     5  
     6  
  
>> squeeze(C(:,1,:))  
  
ans =  
  
     1     5  
     2     6
```

Annotations in green boxes with arrows pointing to the code:

- Definición de una hipermatriz (points to `C=reshape(a,2,2,2)`)
- Extracción de una capa (points to `C(:,1,:)`)
- Extracción de una capa (points to `squeeze(C(:,1,:))`)

Tipos de datos: Estructuras

The MATLAB interface shows the following steps in the Command Window:

```
>> libro.titulo='Don Quijote'

libro =

    titulo: 'Don Quijote'

>> libro.anno_publicacion=1605

libro =

    titulo: 'Don Quijote'
 anno_publicacion: 1605

>> libro=struct('titulo','Don Quijote','anno_publicacion',1605)

libro =

    titulo: 'Don Quijote'
 anno_publicacion: 1605

>> libro.anno_publicacion

ans =

    1605

>> fieldnames(libro)

ans =

    'titulo'
    'anno_publicacion'
```

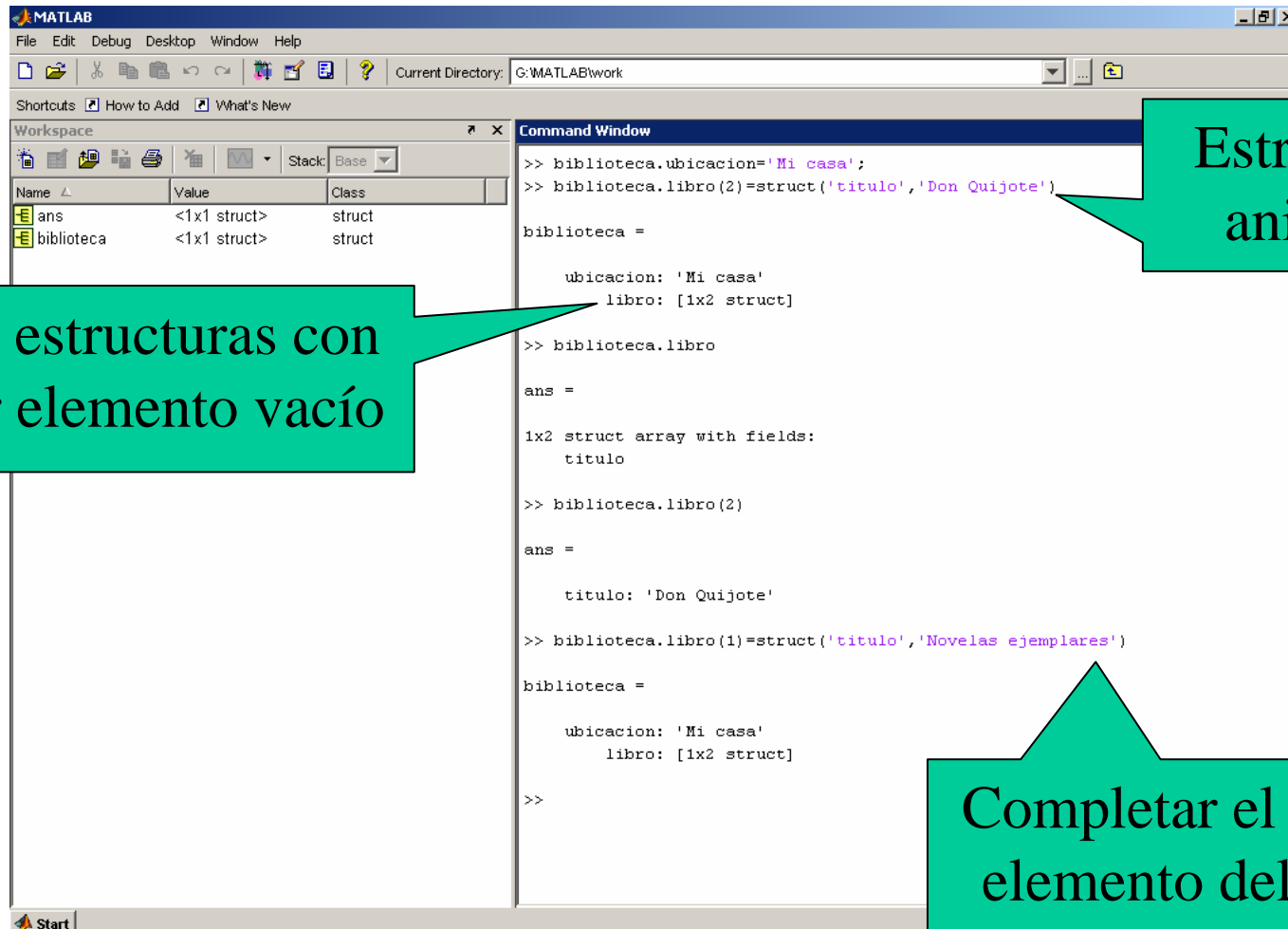
Callouts explaining the steps:

- Declaración de un campo de tipo cadena**: Points to the first line of code (`libro.titulo='Don Quijote'`).
- Declaración de un campo numérico**: Points to the second line of code (`libro.anno_publicacion=1605`).
- Declaración de una estructura**: Points to the third line of code (`libro=struct('titulo','Don Quijote','anno_publicacion',1605)`).
- Acceso a una estructura**: Points to the fourth line of code (`libro.anno_publicacion`).
- Campos de una estructura**: Points to the fifth line of code (`fieldnames(libro)`).

The Workspace window shows the following variables:

Name	Value	Class
ans	<2x1 cell>	cell
libro	<1x1 struct>	struct

Tipos de datos: Estructuras



The image shows a MATLAB interface with three main components: a Workspace window, a Command Window, and three green callout boxes with text annotations.

Workspace Window: Displays a table with columns 'Name', 'Value', and 'Class'. It contains two entries: 'ans' with value '<1x1 struct>' and class 'struct', and 'biblioteca' with value '<1x1 struct>' and class 'struct'.

Command Window: Shows the following MATLAB commands and their outputs:

```
>> biblioteca.ubicacion='Mi casa';  
>> biblioteca.libro(2)=struct('titulo','Don Quijote');  
  
biblioteca =  
  
    ubicacion: 'Mi casa'  
    libro: [1x2 struct]  
  
>> biblioteca.libro  
  
ans =  
  
1x2 struct array with fields:  
    titulo  
  
>> biblioteca.libro(2)  
  
ans =  
  
    titulo: 'Don Quijote'  
  
>> biblioteca.libro(1)=struct('titulo','Novelas ejemplares')  
  
biblioteca =  
  
    ubicacion: 'Mi casa'  
    libro: [1x2 struct]  
  
>>
```

Annotations:

- Array de estructuras con el primer elemento vacío:** Points to the 'ans' variable in the Workspace window.
- Estructuras anidadas:** Points to the 'biblioteca' variable in the Workspace window.
- Completar el primer elemento del array:** Points to the command 'biblioteca.libro(1)=struct('titulo','Novelas ejemplares')' in the Command Window.

Tipos de datos: Arrays de celdas

The image shows the MATLAB environment with the Workspace and Command Window. The Workspace displays a cell array `a` of size `2x2` and a double array `ans` of size `1x2`. The Command Window shows the following commands and outputs:

```
>> a={'Hola' [1 2; 3 4]}
a =
'Hola' [2x2 double]

>> a{1}
ans =
Hola

>> a{2}
ans =
1 2
3 4

>> a{3}=0
a =
'Hola' [2x2 double] [0]

>> a=num2cell([1 2; 3 4])
a =
[1] [2]
[3] [4]

>>
```

Callouts explain the operations:

- Array 1x2**: Points to the initial command `a={'Hola' [1 2; 3 4]}`.
- Acceso al array con { }**: Points to the indexing command `a{1}`.
- Array 2x2**: Points to the command `a=num2cell([1 2; 3 4])`.
- Añadir elemento al array**: Points to the assignment `a{3}=0`.

Tipos de datos: Arrays de celdas y estructuras

The image shows a MATLAB workspace and command window. The workspace contains three variables: 'a' (1x3 cell), 'b' (1x3 cell), and 'st' (1x1 struct). The command window shows the following code and output:

```
>> a={'Hola' [1 2; 3 4] 0}
a =
    'Hola'    [2x2 double]    [0]

>> st=cell2struct(a,{'cadena','matriz','escalar'},2)
st =
    cadena: 'Hola'
    matriz: [2x2 double]
    escalar: 0

>> b=struct2cell(st)
b =
    'Hola'
    [2x2 double]
    [    0]

>> b=b'
b =
    'Hola'    [2x2 double]    [0]

>> disp(b{2})
1    2
3    4

>>
```

Annotations (green boxes with arrows pointing to the code/output):

- Array 1x3**: Points to the first line of code `a={'Hola' [1 2; 3 4] 0}`.
- Conversión a struct**: Points to the second line of code `st=cell2struct(a,{'cadena','matriz','escalar'},2)`.
- Transpuesta a 1x3**: Points to the line `b=b'`.
- De vuelta a celdas: 3x1**: Points to the line `b=struct2cell(st)`.
- La transpuesta no afecta a los contenidos**: Points to the output of `disp(b{2})`.

Programación: Control

```
for i=1:5
    for j=1:5
        A(i,j)=1/(i+j-1);
    end
end
```

```
A=rand(2,3)
for columna=A
    disp(columna)
end
```

```
A=zeros(3,4);
for i=1:size(A,1)
    for j=1:size(A,2)
        A(i,j)=abs(i-j);
    end
end
```

```
for x=1.0:-0.1:0.0
    disp(x)
    if x<0.5 break; end
end
```

```
for i=1:5
    for j=1:5
        if i==j
            A(i,j)=2;
        elseif abs(i-j)==1
            A(i,j)=-1;
        else
            A(i,j)=0;
        end
    end
end
```

```
a=[10 20 30 40];
for i=1:length(a)
    a(i)=a(i)/i;
end
a

a=[10 20 30 40]./[1:length(a)]
```

Programación: Control

```
method = 'Bilinear';

switch lower(method)
    case {'linear','bilinear'}
        disp('Method is linear')
    case 'cubic'
        disp('Method is cubic')
    case 'nearest'
        disp('Method is nearest')
    otherwise
        disp('Unknown method.')
end
```

```
% Precisión de la máquina
eps=1;
while (1+eps)>1
    eps=eps/2;
end
eps=eps*2
```

```
% Contar las líneas de código de magic.m
fid=fopen('magic.m', 'r');
count=0;
while ~feof(fid)
    linea = fgetl(fid);
    if isempty(linea) | strcmp(linea,'% ',1)
        continue
    end
    count=count+1;
end
fclose(fid);
disp(sprintf('%d líneas', count));
```

Programación: Funciones

```
function [mean,stdev]=stat(x)
% Esta función calcula la media y la desviación típica de x
% Uso: [mean,stdev]=stat(x)

% El fichero debe llamarse stat.m

n=length(x);
mean=sum(x)/n;
stdev=sqrt(sum((x-mean).^2/n));
end

help stat
[media,desviacion]=stat([1 2 3 4 5]);
```

```
function [mean,stdev]=stat(x)
n=length(x);
mean=media(x);
stdev=sqrt(sum((x-mean).^2/n));
```

```
function mean=media(x)
mean=sum(x)/length(x);
```

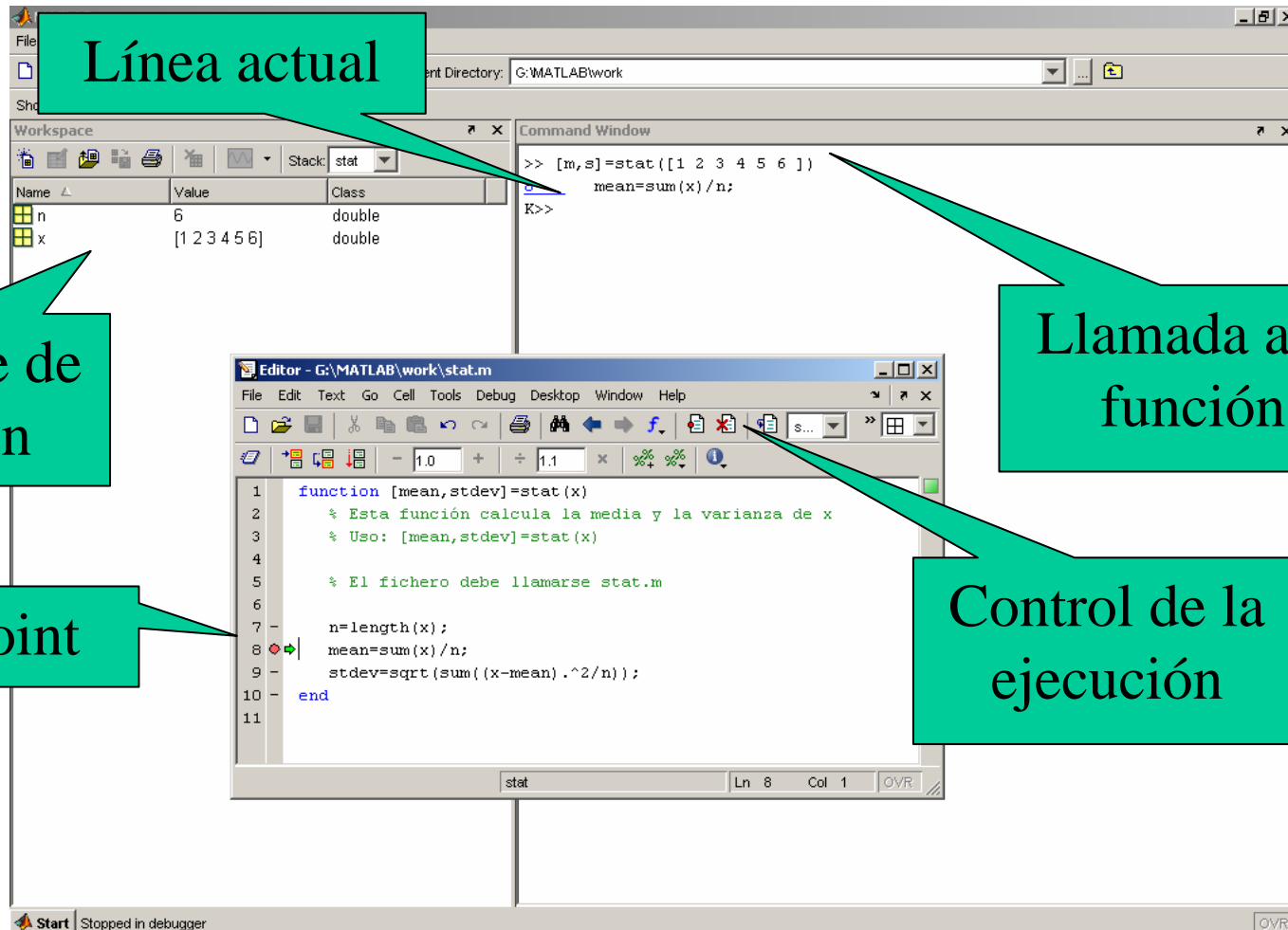
```
function [mean,stdev]=stat(x)
n=length(x);
mean=media();
stdev=sqrt(sum((x-mean).^2/n));

function mean=media()
mean=sum(x)/length(x);
end
end
```

Subfunción

Función
anidada

Entorno de MATLAB: Debugger



Programación: Funciones

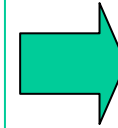
```
function varargout=atan3(varargin)
    if nargin==1
        rad=atan(varargin{1});
    elseif nargin==2
        rad=atan2(varargin{1},varargin{2});
    else
        disp('Error: más de dos argumentos')
        return
    end
    varargout{1}=rad;
    varargout{2}=rad*180/pi;
end
```

```
function [x0, y0] = myplot(x, y, npts, angle, subdiv)
% MYPLOT Plot a function.
% MYPLOT(x, y, npts, angle, subdiv)
%     The first two input arguments are
%     required; the other three have default values.
...
if nargin < 5, subdiv = 20; end
if nargin < 4, angle = 10; end
if nargin < 3, npts = 25; end
...
if nargout == 0
    plot(x, y)
else
    x0 = x;
    y0 = y;
end
```

Programación: funciones vs scripts (batches)

houdini

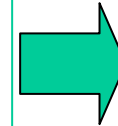
```
% File: houdini.m  
m = magic(4); % Assign 4x4 magic square to m.  
t = m .^ 3;   % Cube each element of m.  
disp(t);     % Display the value of t.
```



Workspace: m,t

houdini(4)

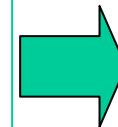
```
% File: houdini.m  
function houdini(sz)  
m = magic(sz); % Assign 4x4 magic square to m.  
t = m .^ 3;   % Cube each element of m.  
disp(t);     % Display the value of t.
```



Workspace:

[m,t]=houdini(4)

```
% File: houdini.m  
function [m,t]=houdini(sz)  
m = magic(sz); % Assign 4x4 magic square to m.  
t = m .^ 3;   % Cube each element of m.  
disp(t);     % Display the value of t.
```



Workspace: m,t

Programación: Control

```
function matrixMultiply(A, B)
try
    X = A * B
catch
    disp '** Error multiplying A * B'
end

A = [1 2 3; 6 7 2; 0 1 5];
B = [9 5 6; 0 4 9];
matrixMultiply(A, B)
```

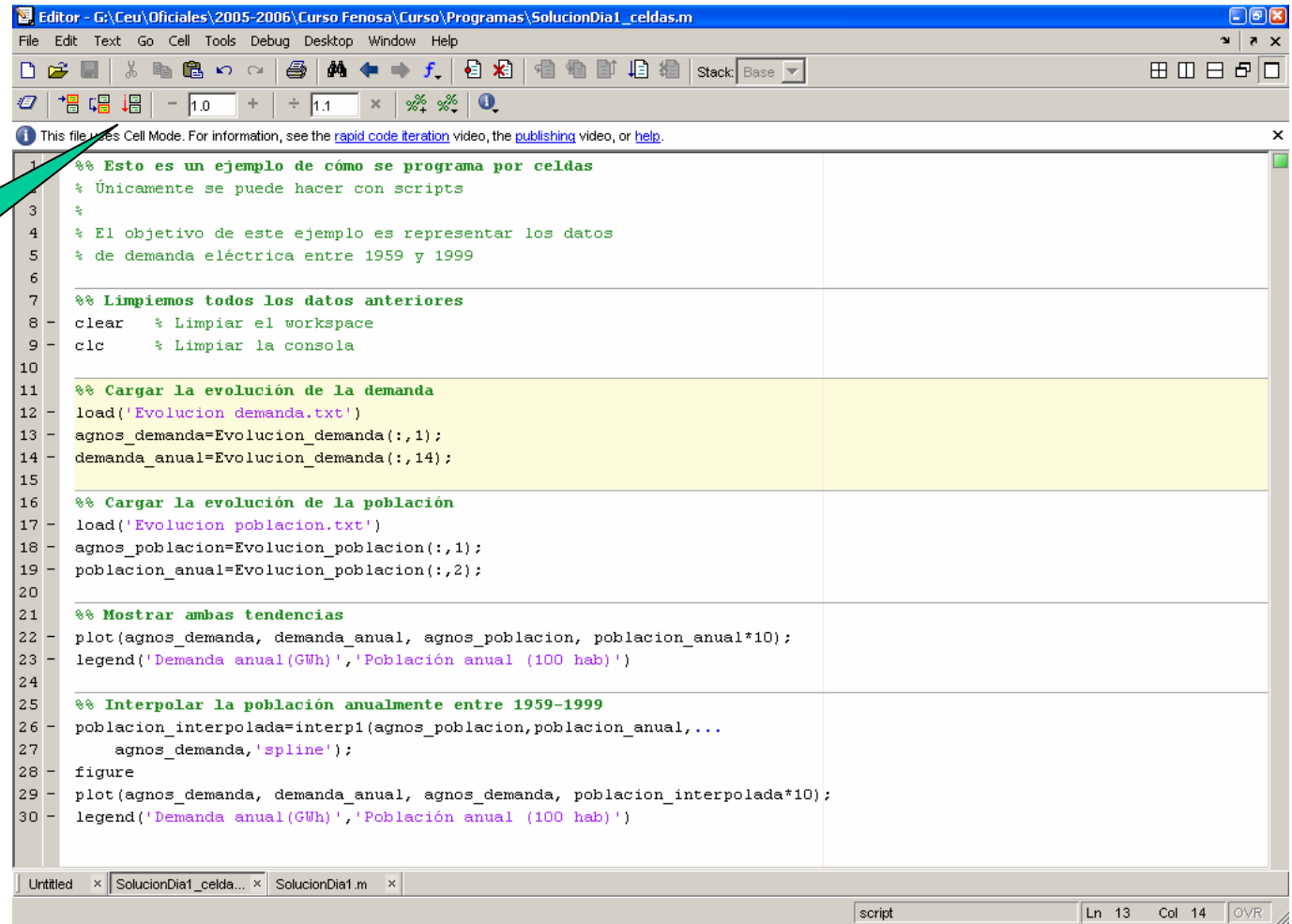
Programación: Entrada/Salida

- Consola/Teclado:
 - input/disp
 - echo, diary
- Cadena
 - sscanf, strread/sprintf
- Fichero:
 - load/save
 - dlmread/dlmwrite
 - tblread/tblwrite
 - xlsread/xlswrite
 - fopen, fread, fgetl, textscan/fwrite, fclose
 - importdata
 - uiimport
- Clipboard:
 - clipboard



Programación: Ejecución por celdas

Ejecución por
celdas



```
Editor - G:\Ceu\Oficiales\2005-2006\Curso Fenosa\Curso\Programas\SolucionDia1_celdas.m
File Edit Text Go Cell Tools Debug Desktop Window Help
[Icons] Stack: Base
This file uses Cell Mode. For information, see the rapid code iteration video, the publishing video, or help.
1 %% Esto es un ejemplo de cómo se programa por celdas
2 % Únicamente se puede hacer con scripts
3 %
4 % El objetivo de este ejemplo es representar los datos
5 % de demanda eléctrica entre 1959 y 1999
6
7 %% Limpiemos todos los datos anteriores
8 clear % Limpiar el workspace
9 clc % Limpiar la consola
10
11 %% Cargar la evolución de la demanda
12 load('Evolucion_demanda.txt')
13 agnos_demanda=Evolucion_demanda(:,1);
14 demanda_anual=Evolucion_demanda(:,14);
15
16 %% Cargar la evolución de la población
17 load('Evolucion_poblacion.txt')
18 agnos_poblacion=Evolucion_poblacion(:,1);
19 poblacion_anual=Evolucion_poblacion(:,2);
20
21 %% Mostrar ambas tendencias
22 plot(agnos_demanda, demanda_anual, agnos_poblacion, poblacion_anual*10);
23 legend('Demanda anual(GWh)', 'Población anual (100 hab)')
24
25 %% Interpolan la población anualmente entre 1959-1999
26 poblacion_interpolada=interp1(agnos_poblacion, poblacion_anual, ...
27 agnos_demanda, 'spline');
28 figure
29 plot(agnos_demanda, demanda_anual, agnos_demanda, poblacion_interpolada*10);
30 legend('Demanda anual(GWh)', 'Población anual (100 hab)')
```

Programación: Documentación

- MiToolBox (Directorio)

- contents.m →
- mifuncion1.m →
- mifuncion2.m
- ...

```
help MiToolBox
```

```
% Esta toolbox sirve para ...  
%  
% Familia de funciones 1  
%     mifuncion1      -  Sirve para ...  
%     mifuncion2      -  Sirve para ...  
...
```

```
help mifuncion1
```

```
function y=mifuncion1(x)  
% Esta función no hace nada  
% Uso: y=mifuncion1(x)  
% Entradas:  
%     x      : vector de entrada para  
%              no hacer nada con él  
% Salidas:  
%     y      : vector de salida  
y=x
```

Programación: Publicación HTML de scripts



Esto es un ejemplo de cómo se programa por celdas

Únicamente se puede hacer con scripts

El objetivo de este ejemplo es representar los datos de demanda eléctrica entre 1959 y 1999

Contents

- ♦ [Limpiemos todos los datos anteriores](#)
- ♦ [Cargar la evolución de la demanda](#)
- ♦ [Cargar la evolución de la población](#)
- ♦ [Mostrar ambas tendencias](#)
- ♦ [Interpolación la población anualmente entre 1959-1999](#)

Limpiemos todos los datos anteriores

```
clear    % Limpiar el workspace
clc      % Limpiar la consola
```

Cargar la evolución de la demanda

```
load('Evolucion_demanda.txt')
agnos_demanda=Evolucion_demanda(:,1);
demanda_anual=Evolucion_demanda(:,14);
```

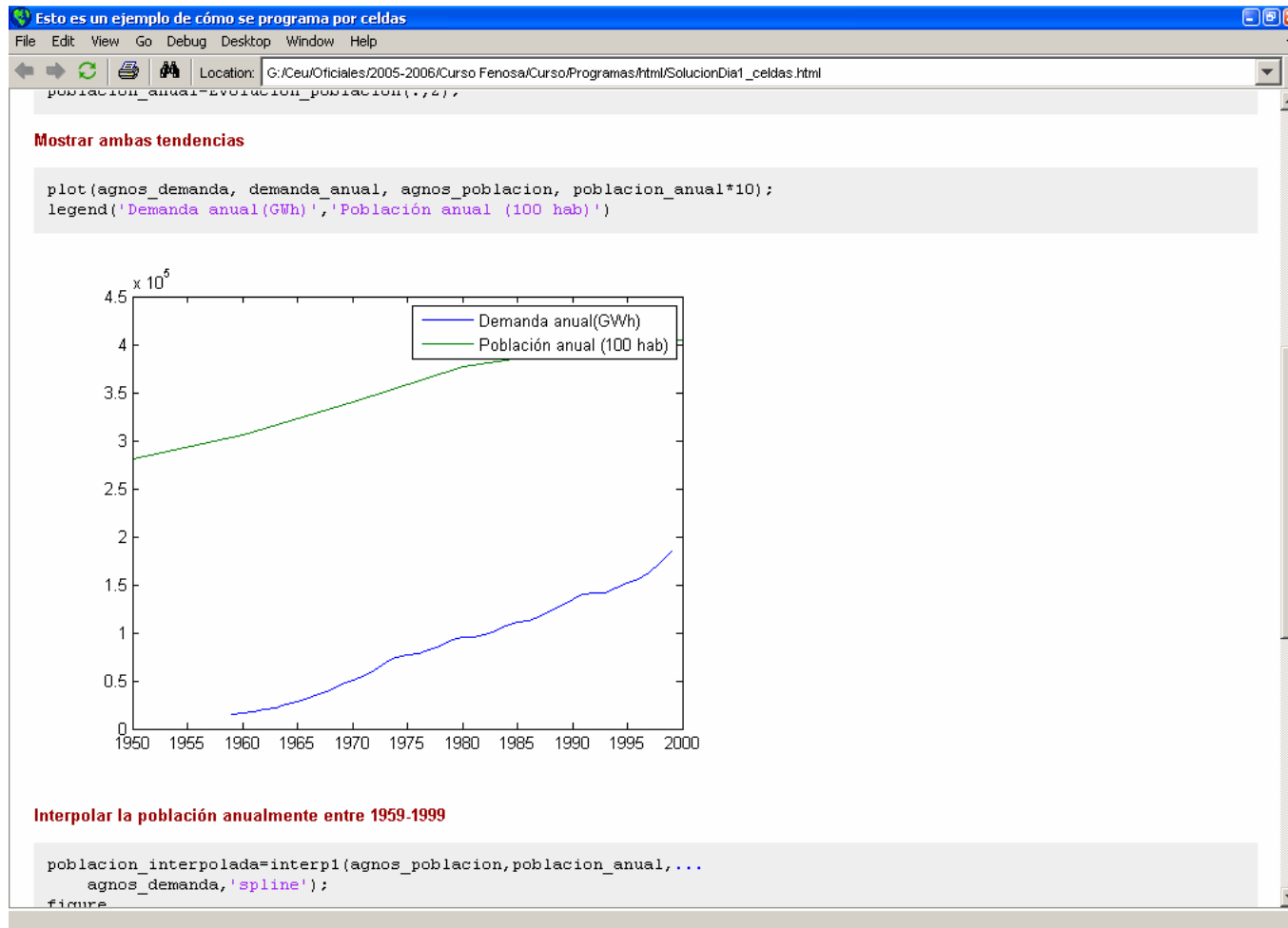
Cargar la evolución de la población

```
load('Evolucion_poblacion.txt')
agnos_poblacion=Evolucion_poblacion(:,1);
poblacion_anual=Evolucion_poblacion(:,2);
```

Mostrar ambas tendencias

```
plot(agnos_demanda, demanda_anual, agnos_poblacion, poblacion_anual*10);
legend('Demanda anual(GWh)','Población anual (100 hab)')
```

Programación: Publicación HTML de scripts



Ejercicio final 1: Entrada/Salida

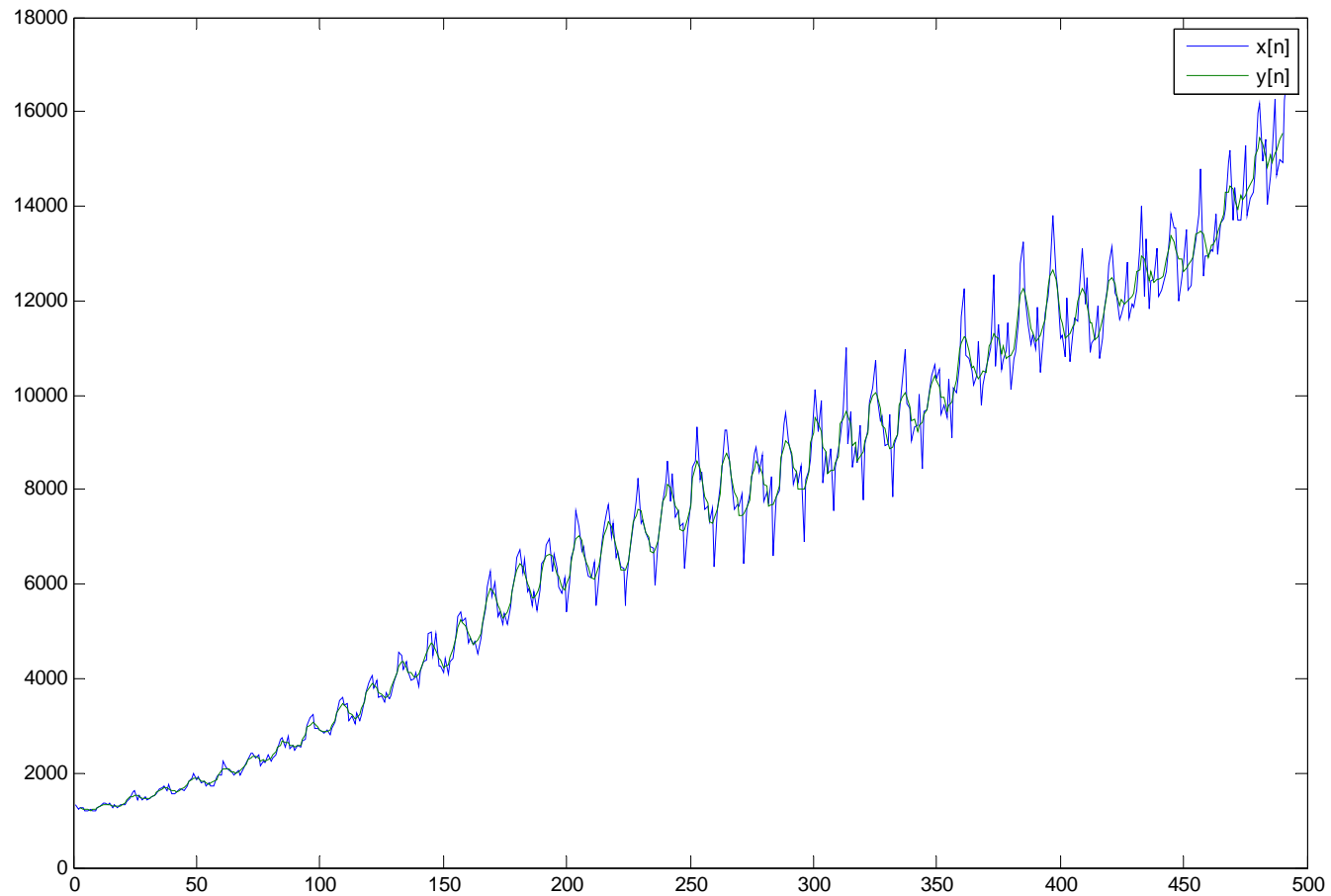
#	Año	Ene	Feb	Mar	Abr	May	Jun	Jul	Ago	Sep	Oct	Nov	Dic	Total
1959		1345	1246	1252	1254	1188	1202	1243	1202	1192	1270	1312	1361	15067
# Década de los 60														
1960		1369	1331	1351	1252	1321	1275	1325	1322	1336	1393	1456	1610	16341
1961		1622	1422	1528	1433	1488	1418	1467	1506	1514	1573	1653	1711	18335
1962		1741	1639	1755	1557	1572	1585	1653	1653	1642	1730	1828	1902	20257
...														

- Ejercicio:
 - Leer el fichero “Evolucion demanda comentada.txt”
separando los años, los datos mensuales y los datos totales.
 - Pedir al usuario el mes y los años que desea representar
 - Representar los datos solicitados

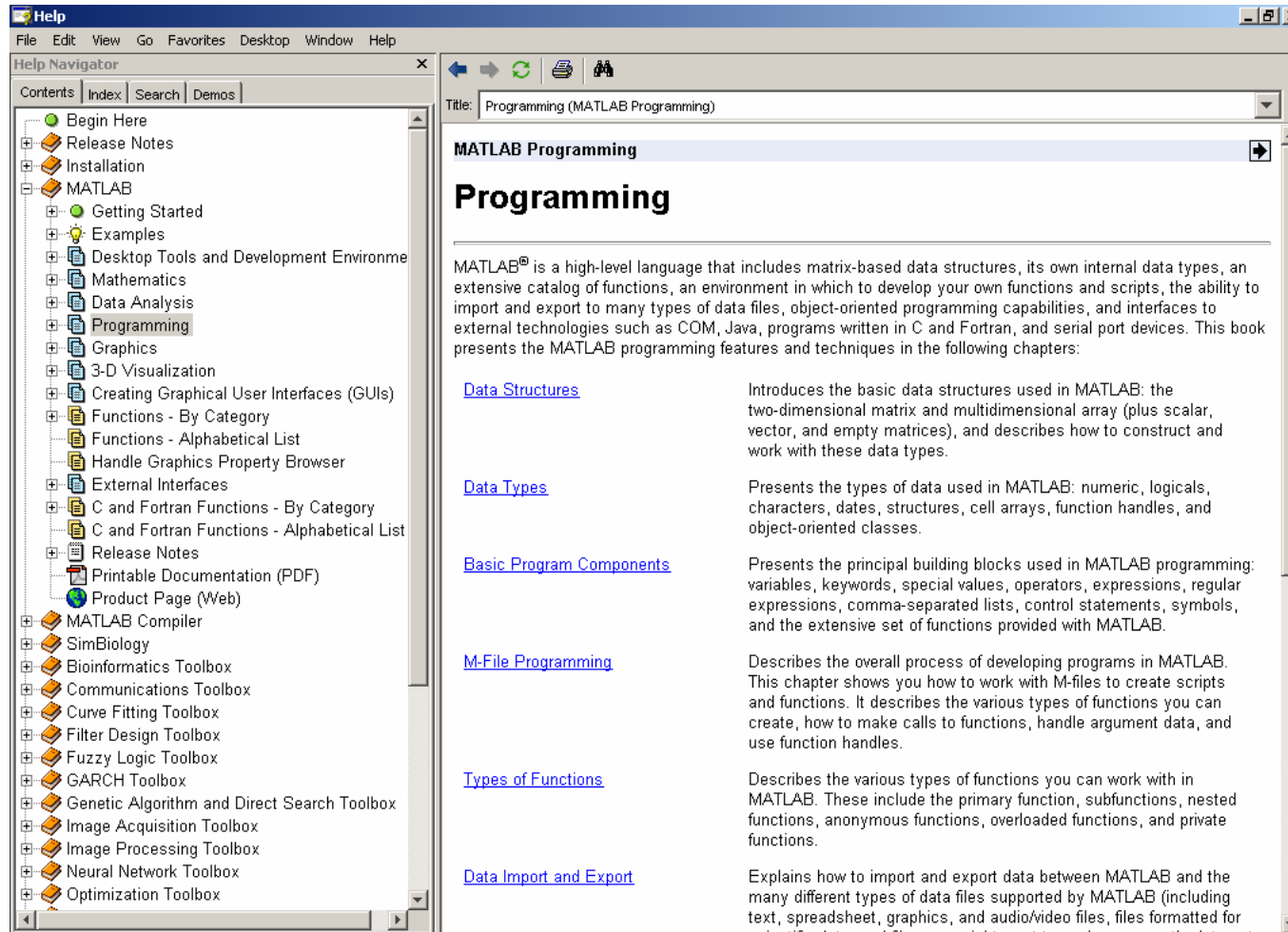
Ejercicio final 2: Cálculo científico

- Ejercicio:
 - Leer el fichero “Evolucion demanda comentada.txt” por medio de una función que recibe el nombre de fichero como parámetro
 - Denominemos $x[n]$ a los datos mensuales
 - $n=1$ para enero de 1959
 - $n=2$ para febrero de 1959
 - ...
 - $n=12$ para diciembre de 1959
 - $n=13$ para enero de 1960
 - etc.
 - Suavizar los datos leídos mediante la fórmula
$$y[n] = \frac{x[n-2] + x[n-1] + x[n] + x[n+1] + x[n+2]}{5}$$
 - Representar la demanda y la demanda suavizada en un mismo gráfico

Ejercicio 2: Cálculo científico



Y si sobra tiempo ...





CEU
*Universidad
San Pablo*

CURSO de UTILIZACIÓN PRÁCTICA de MATLAB

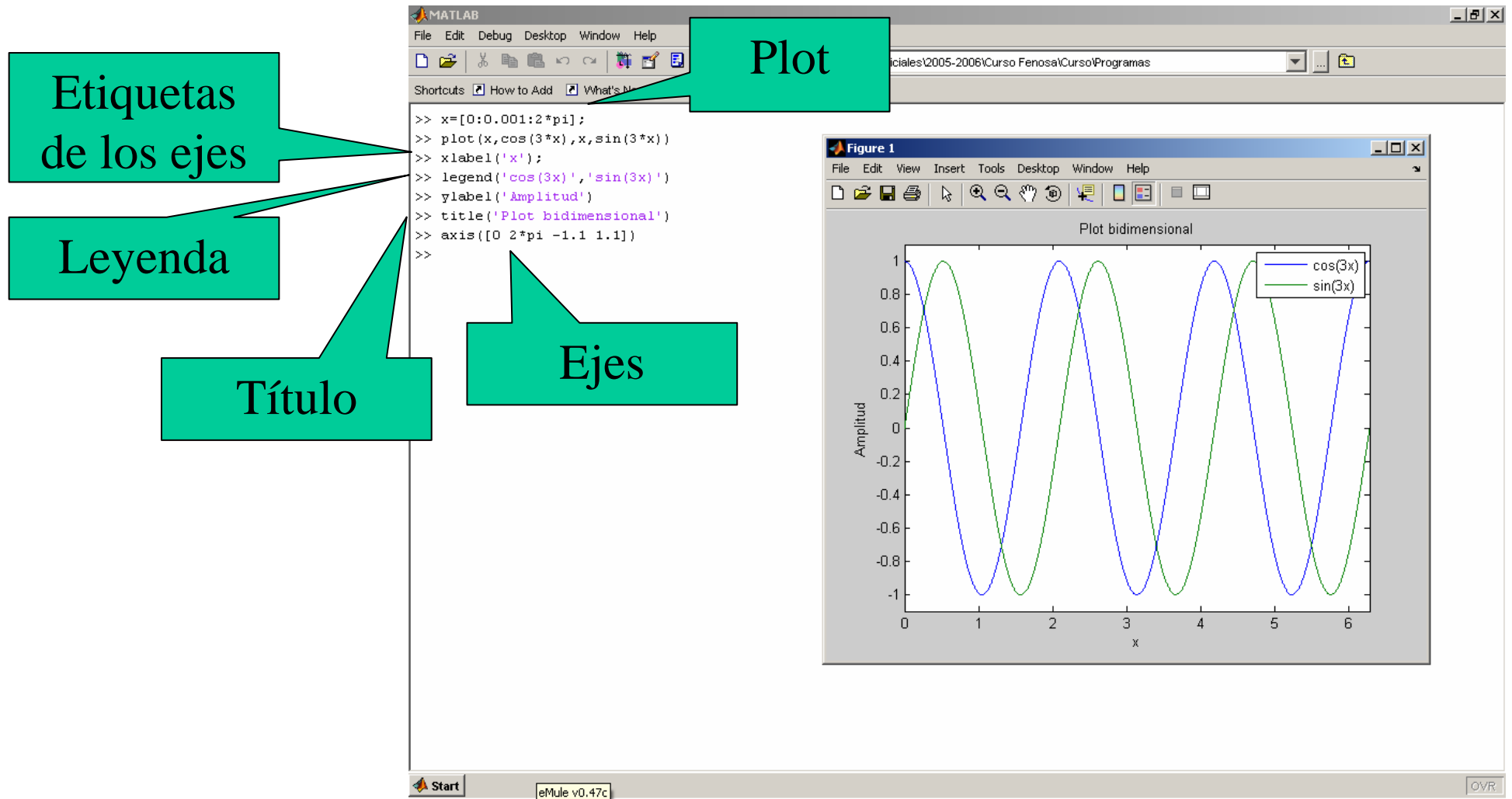
Sesión 3

Carlos Óscar Sánchez Sorzano, Ph.D.
Madrid, July 17th 2006

Cronograma del curso

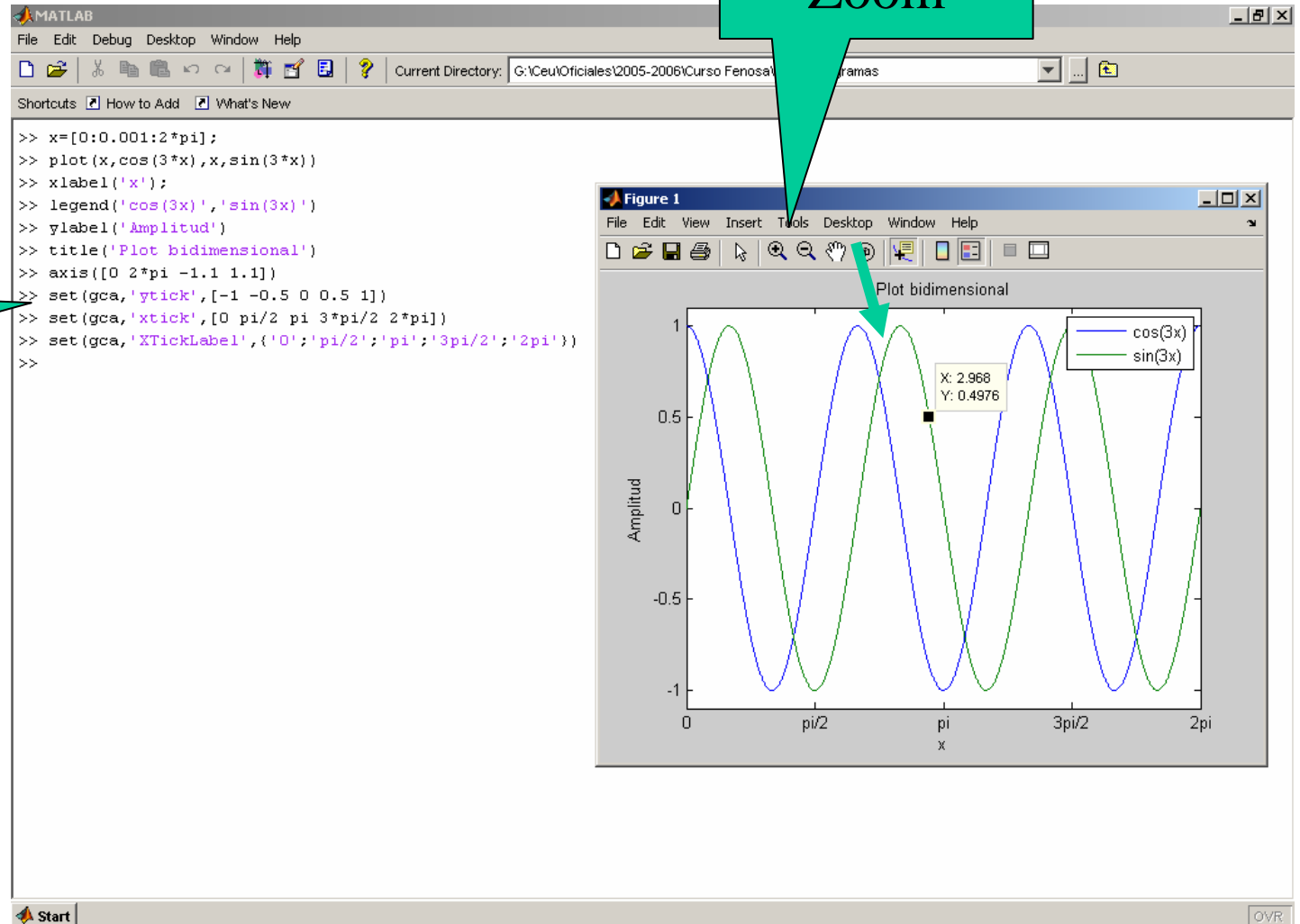
- Día 1: Operaciones con matrices y vectores. Funciones de librería.
- Día 2: Otros tipos de datos en MATLAB. Programación en MATLAB.
- **Día 3: Gráficos bidimensionales. Gráficos tridimensionales.**
- Día 4: Interfaces de usuario en MATLAB. Generación de programas autónomos
- Día 5: Librerías de interés práctico
- Día 6: Interacción de MATLAB con Office y Visual Basic
- Día 7: Desarrollo de un proyecto

Gráficos bidimensionales



Gráficos bidimensionales

Marcas de los ejes



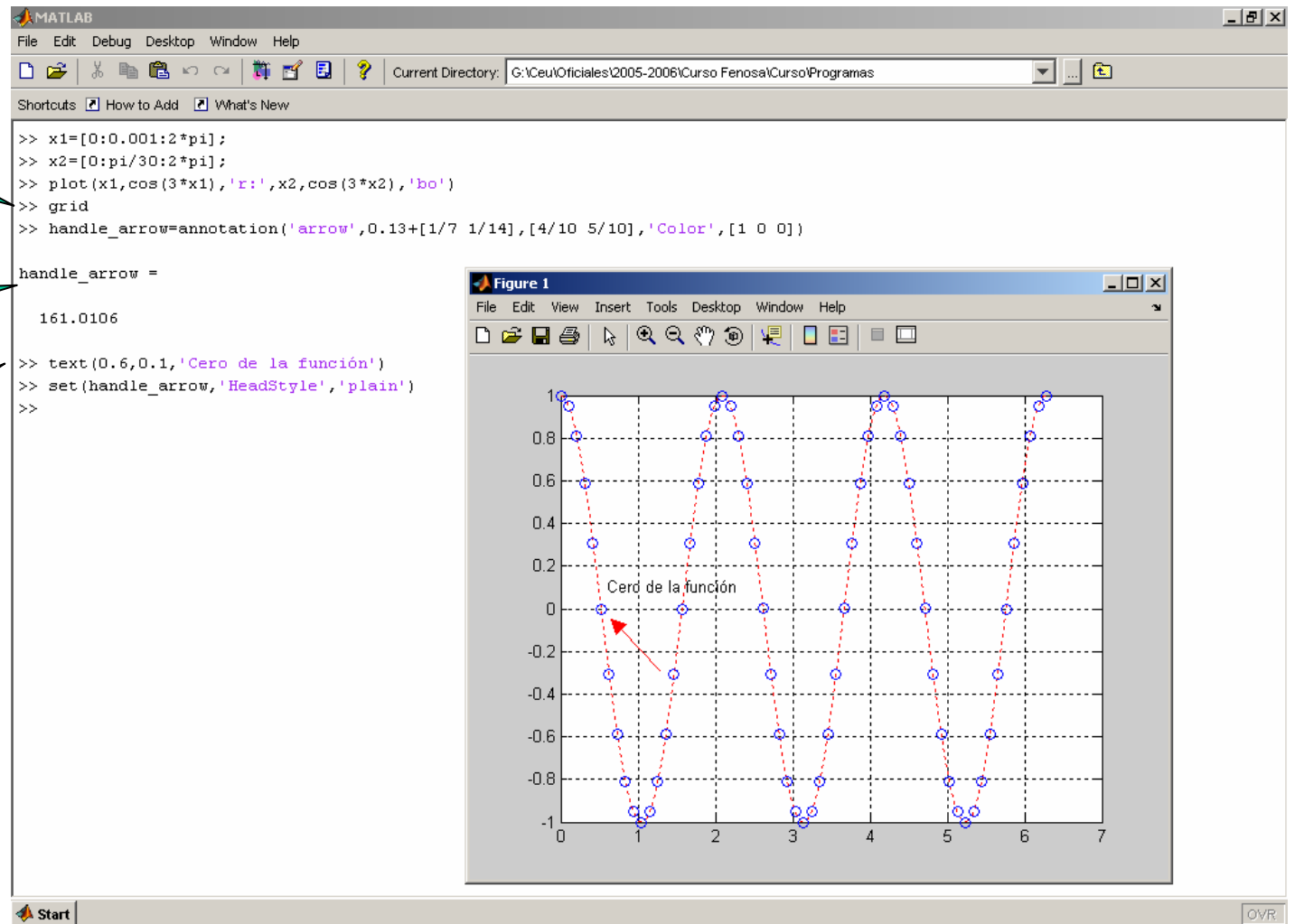
Zoom

Gráficos bidimensionales

Grid

Handle

Anotaciones

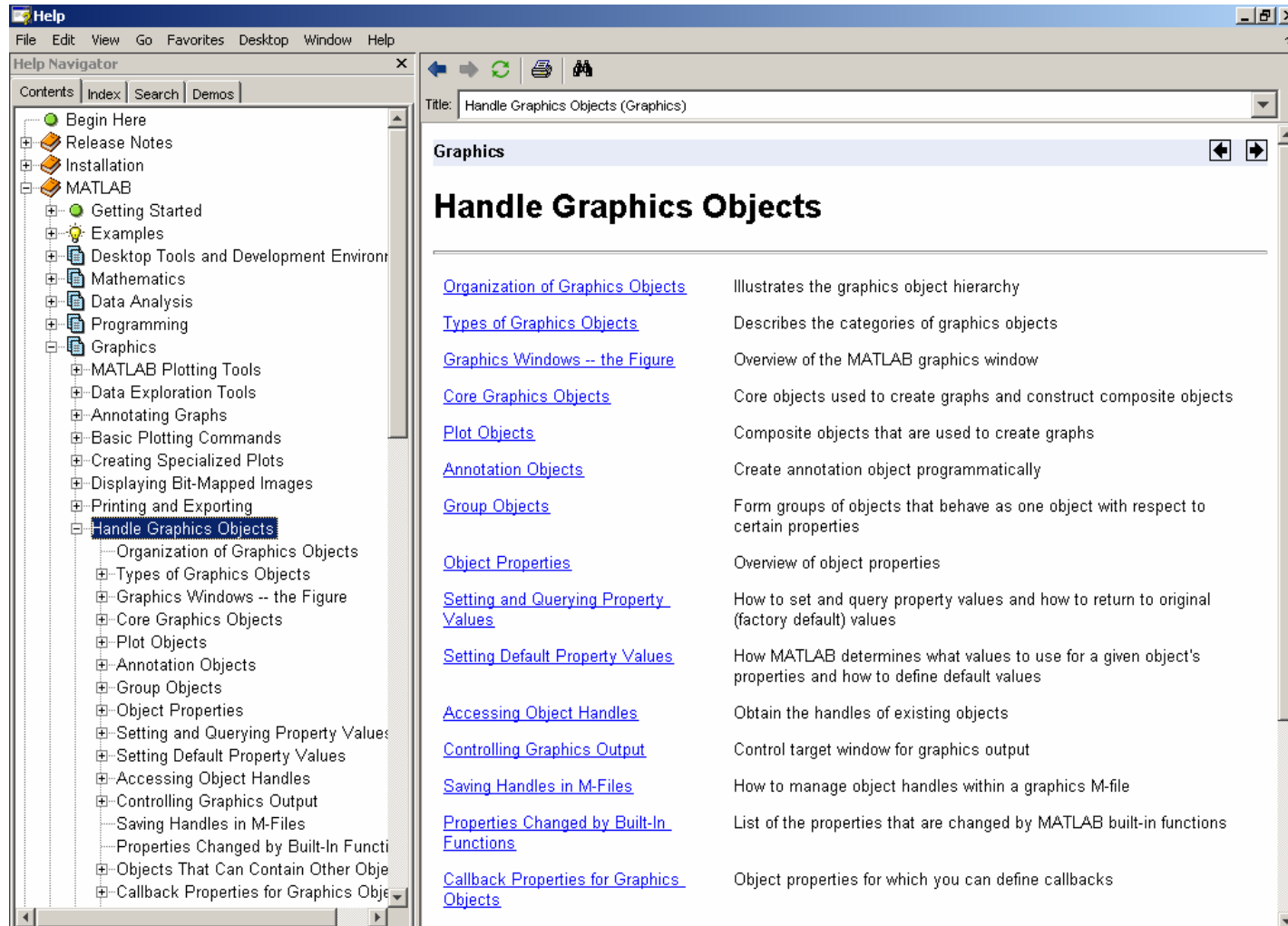


Gráficos bidimensionales

Símbolo	Color	Símbolo	Marcadores (markers)
y	yellow	.	puntos
m	magenta	o	círculos
c	cyan	x	marcas en x
r	red	+	marcas en +
g	green	*	marcas en *
b	blue	s	marcas cuadradas (square)
w	white	d	marcas en diamante (diamond)
k	black	^	triángulo apuntando arriba
		v	triángulo apuntando abajo
Símbolo	Estilo de línea	>	triángulo apuntando a la dcha
-	líneas continuas	<	triángulo apuntando a la izda
:	líneas a puntos	p	estrella de 5 puntas
-.	líneas a barra-punto	h	estrella se seis puntas
--	líneas a trazos		

Tabla 1. Colores, markers y estilos de línea.

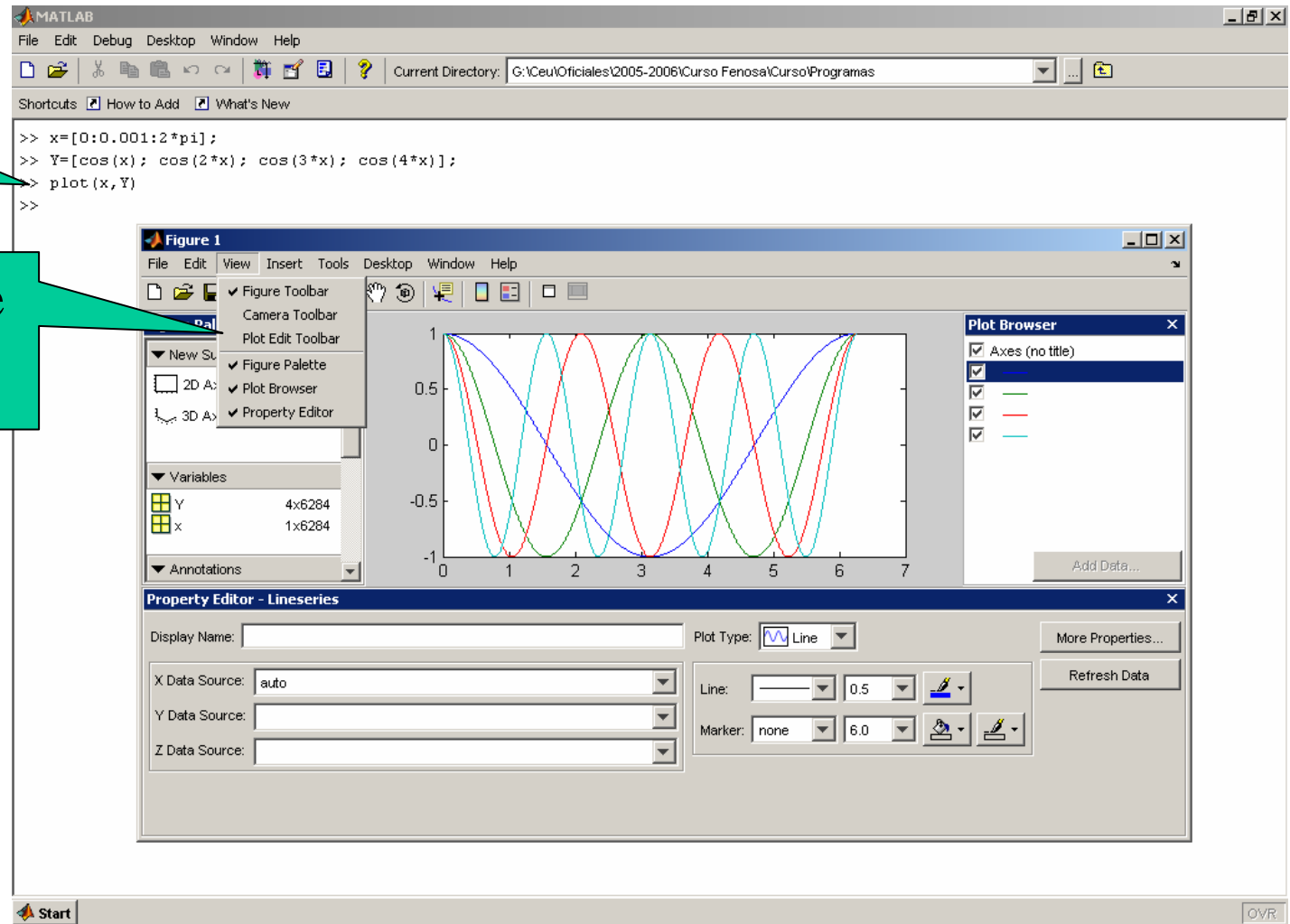
Gráficos bidimensionales



Gráficos bidimensionales

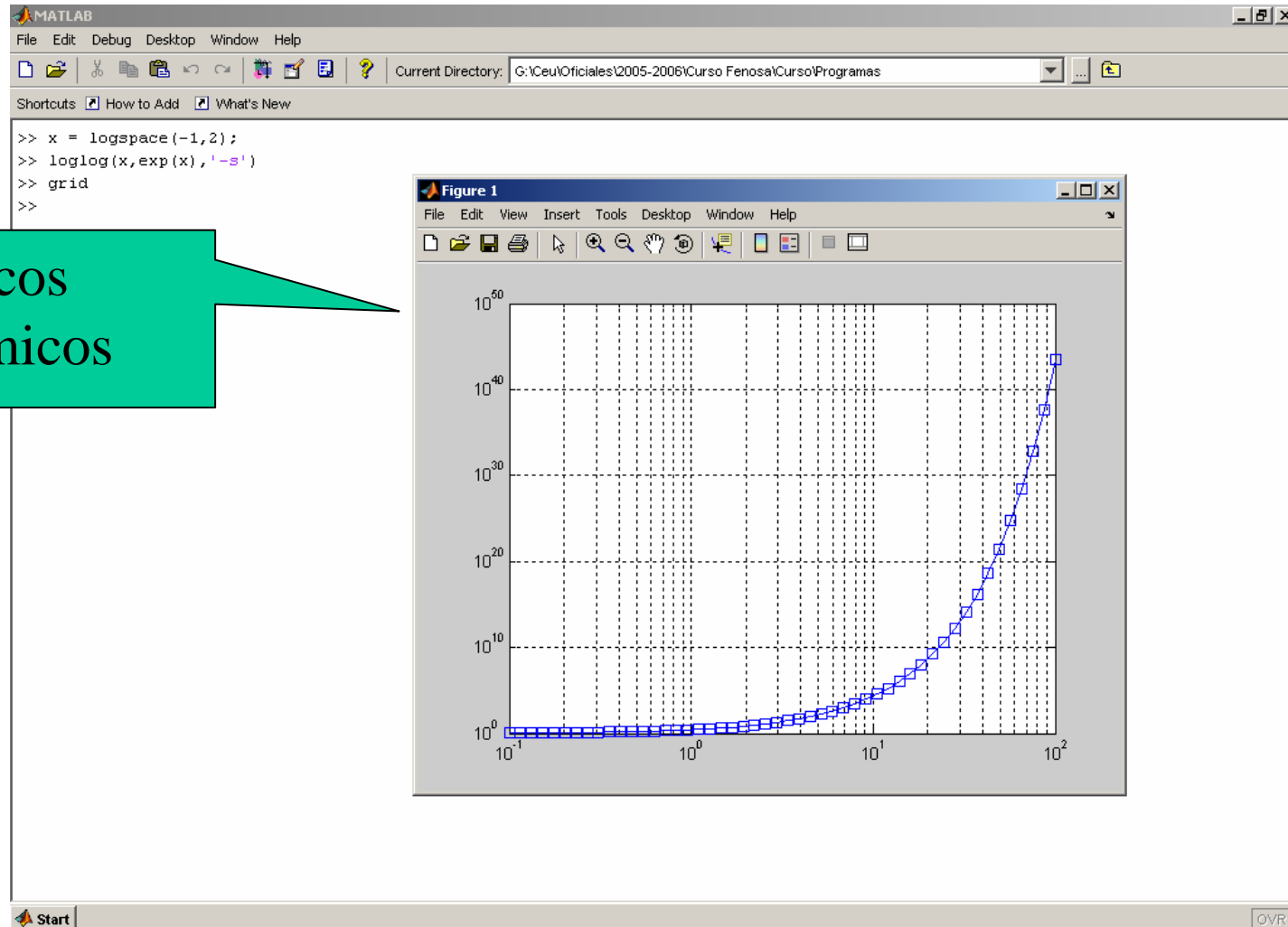
Plot de las filas
de una matriz

Gestión gráfica de
las propiedades

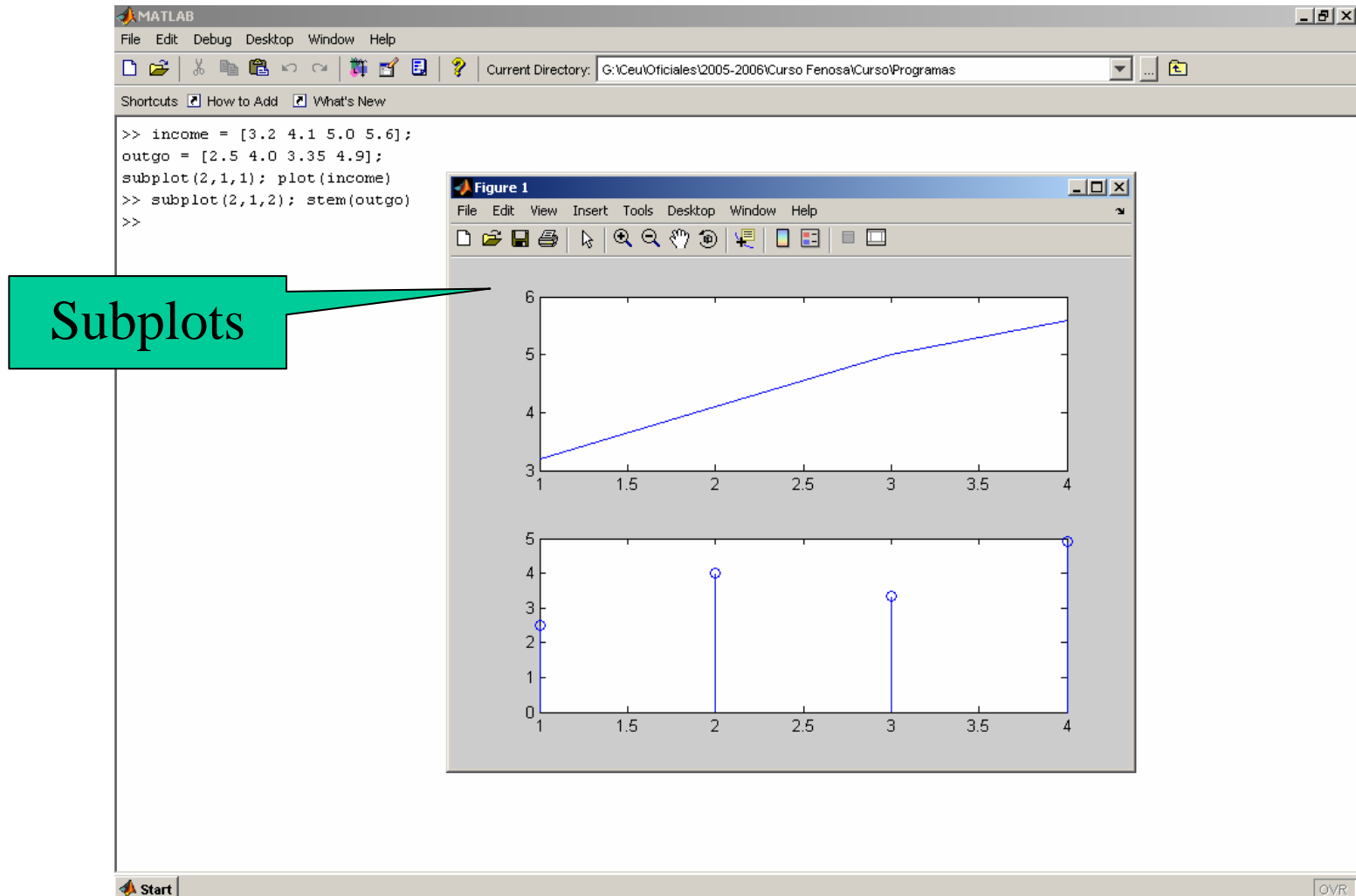


Gráficos bidimensionales

Gráficos
logarítmicos

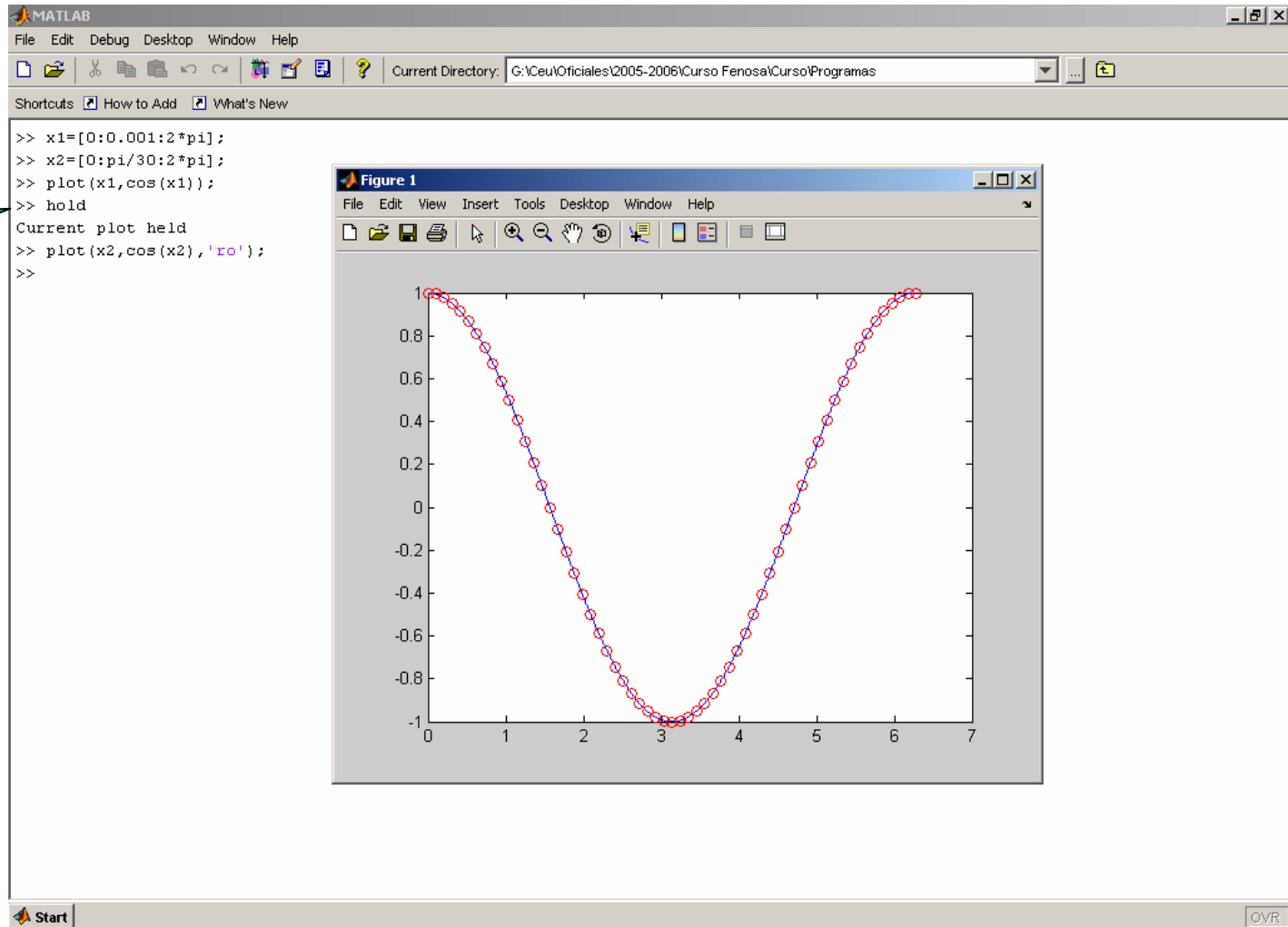


Gráficos bidimensionales

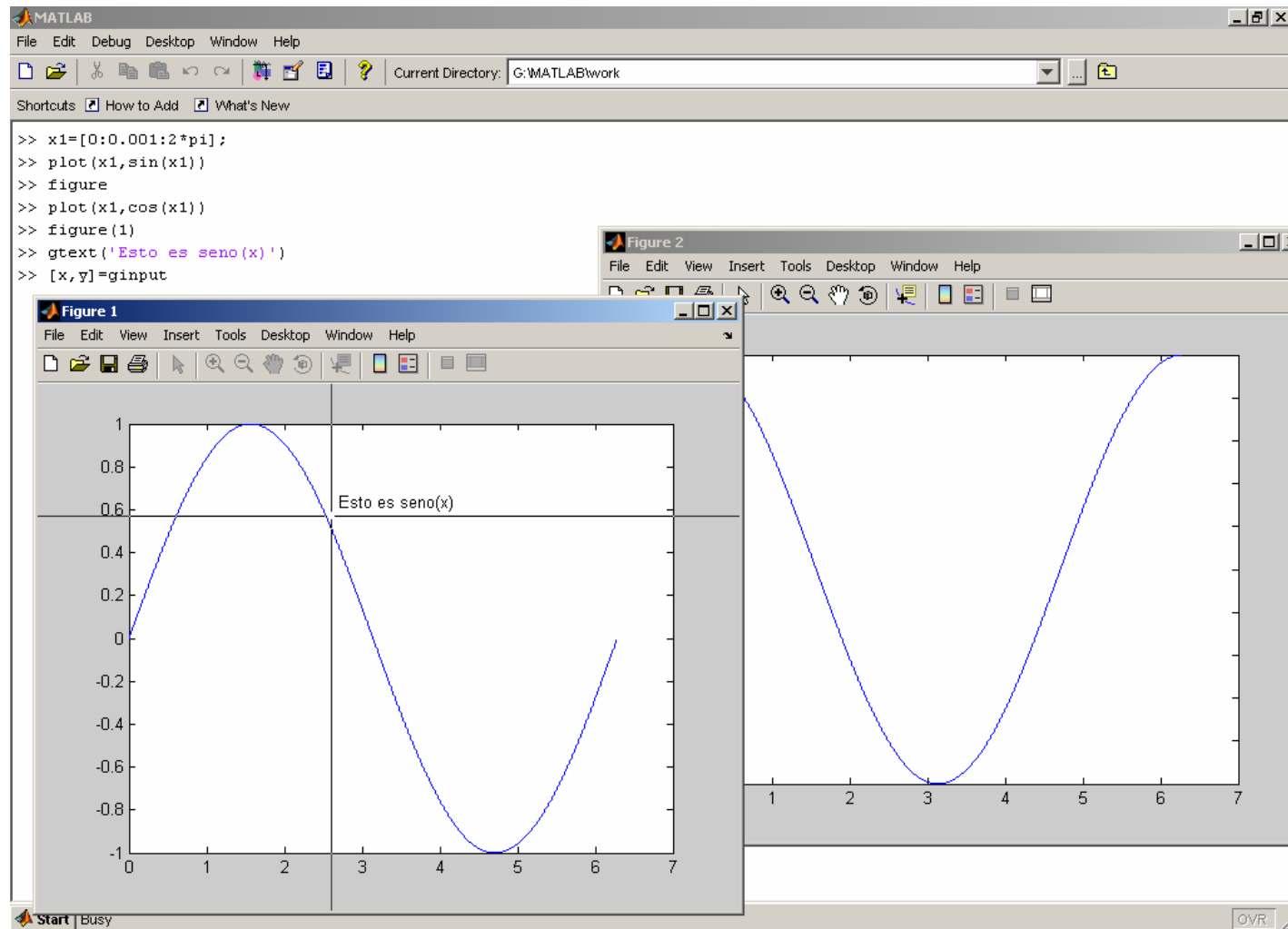


Gráficos bidimensionales

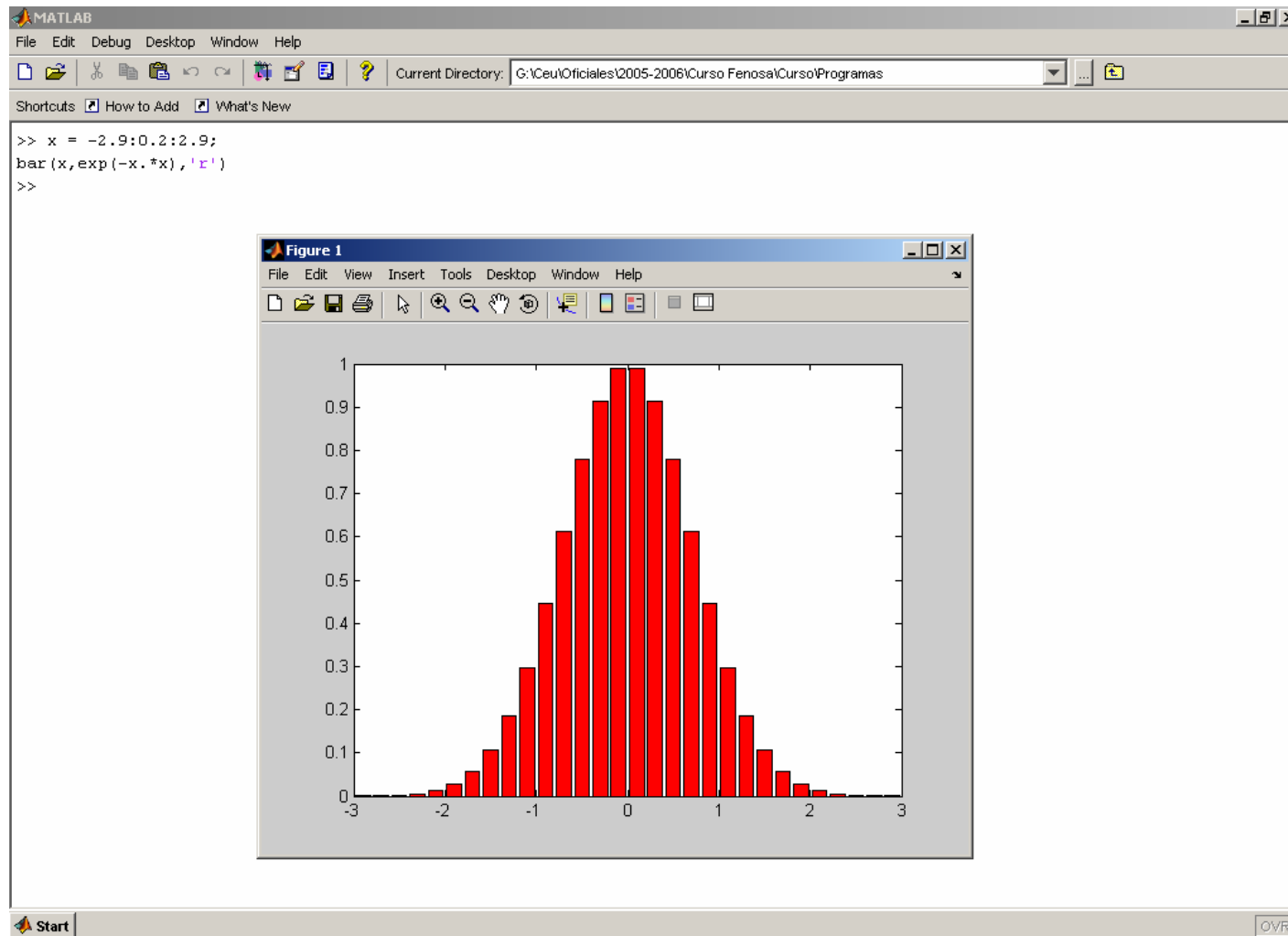
hold



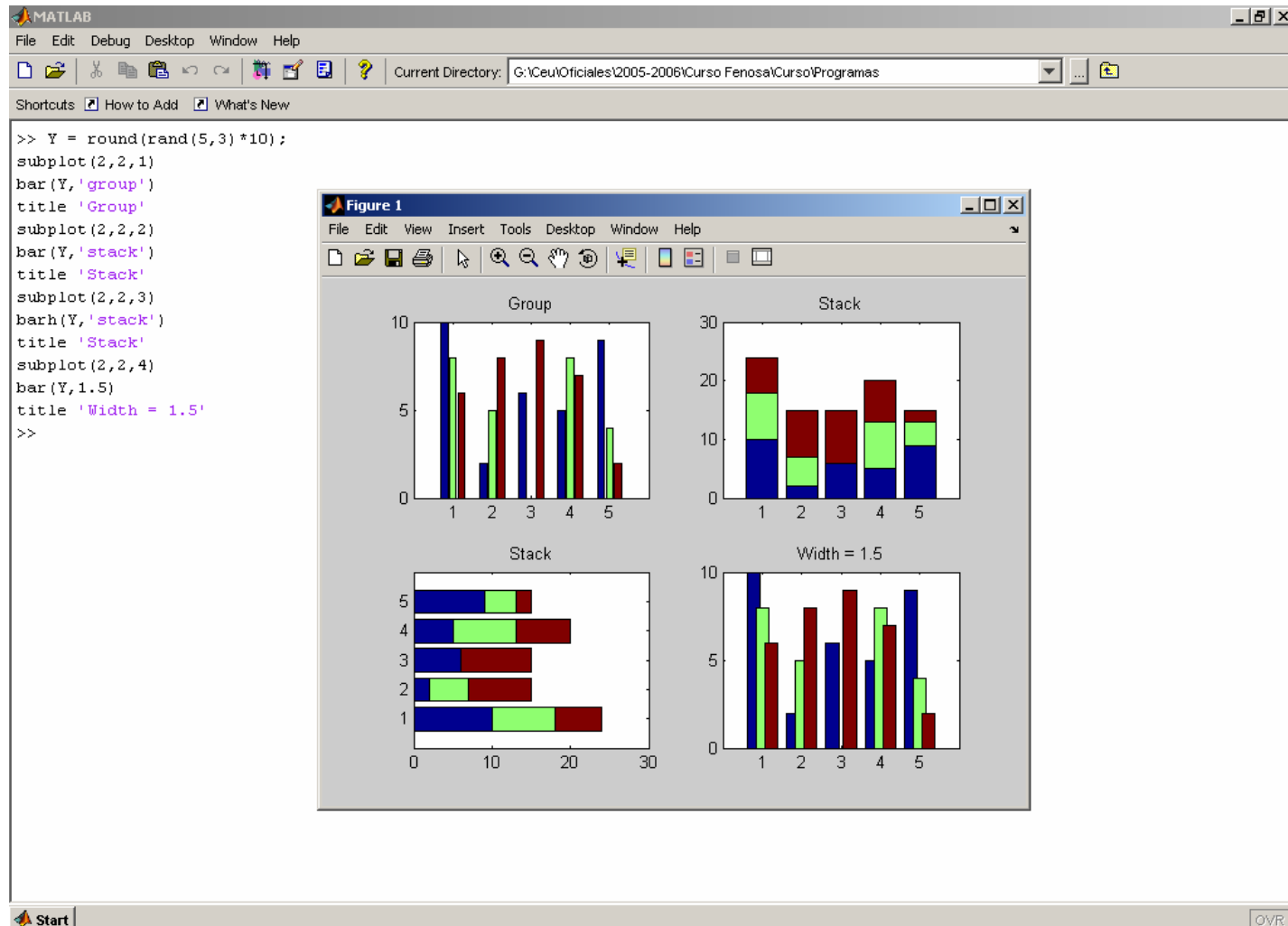
Gráficos bidimensionales



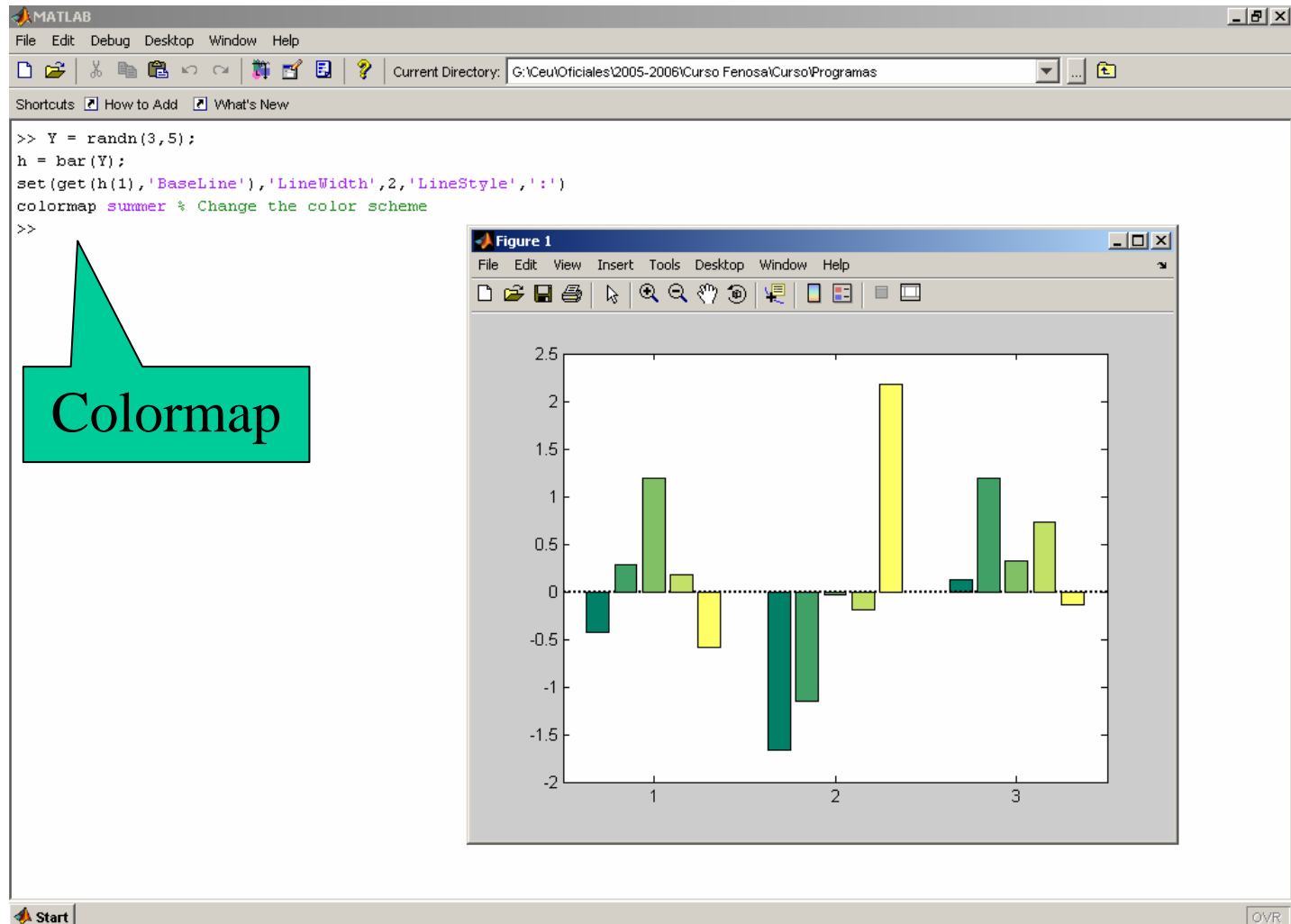
Gráficos bidimensionales



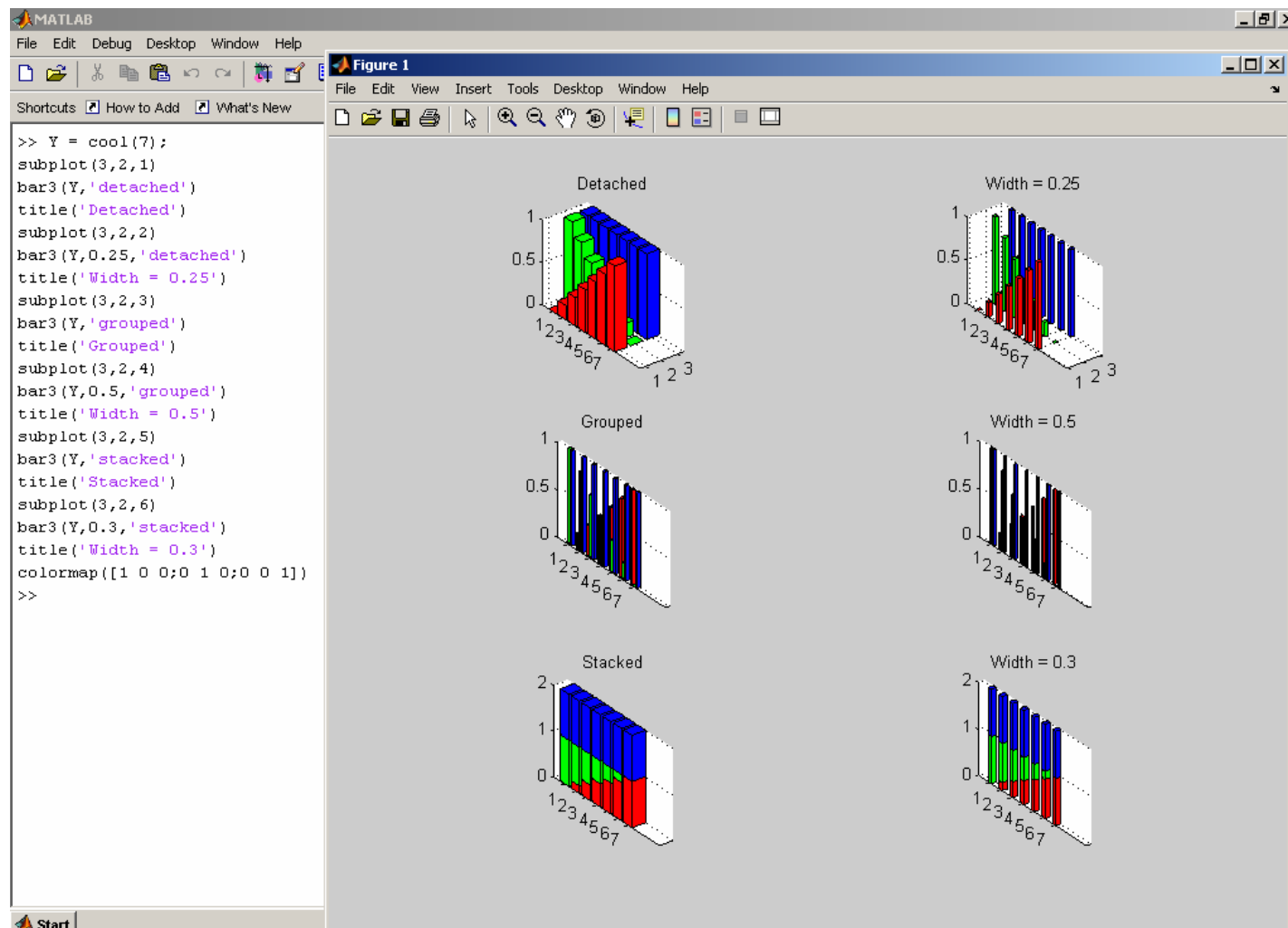
Gráficos bidimensionales



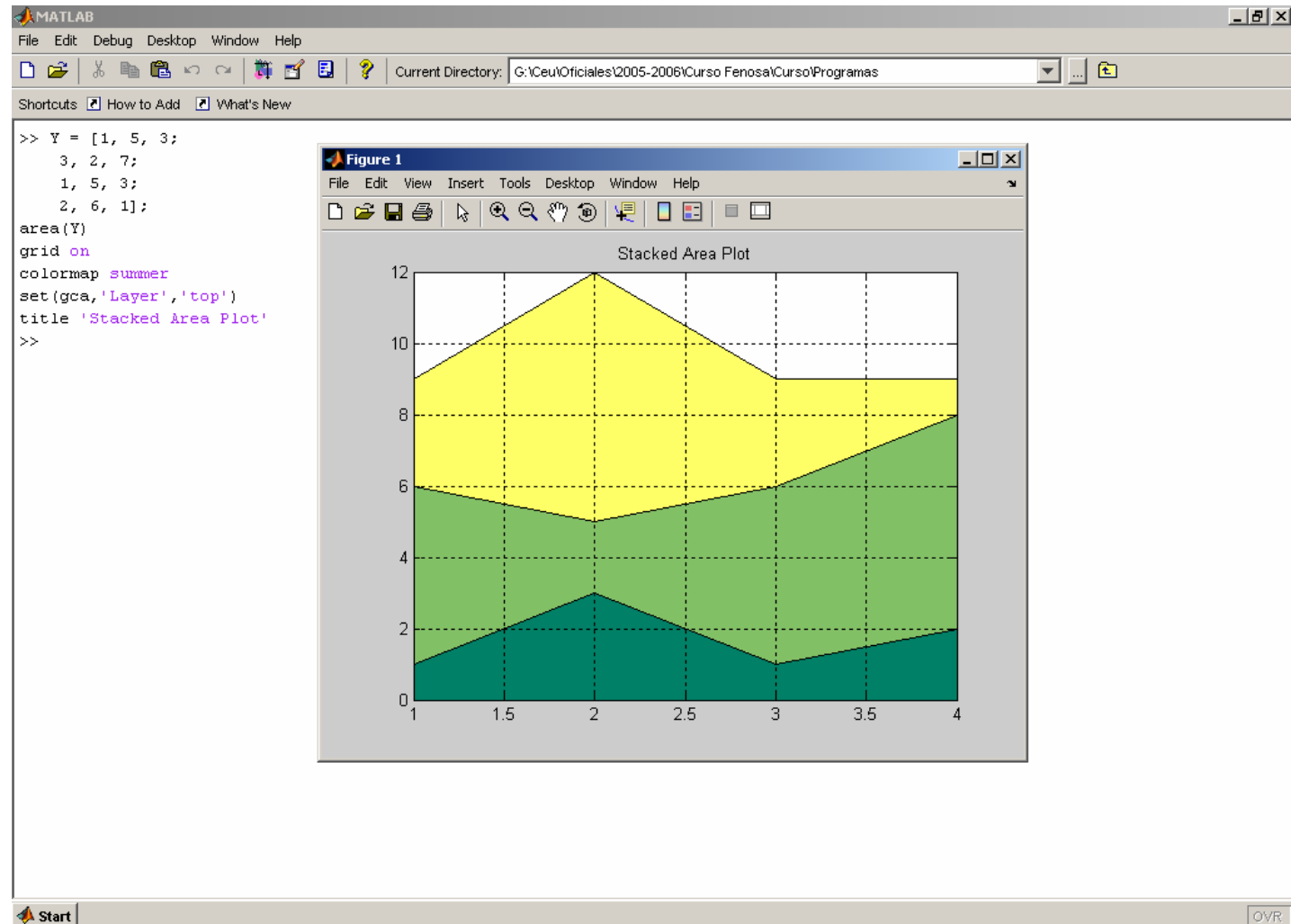
Gráficos bidimensionales



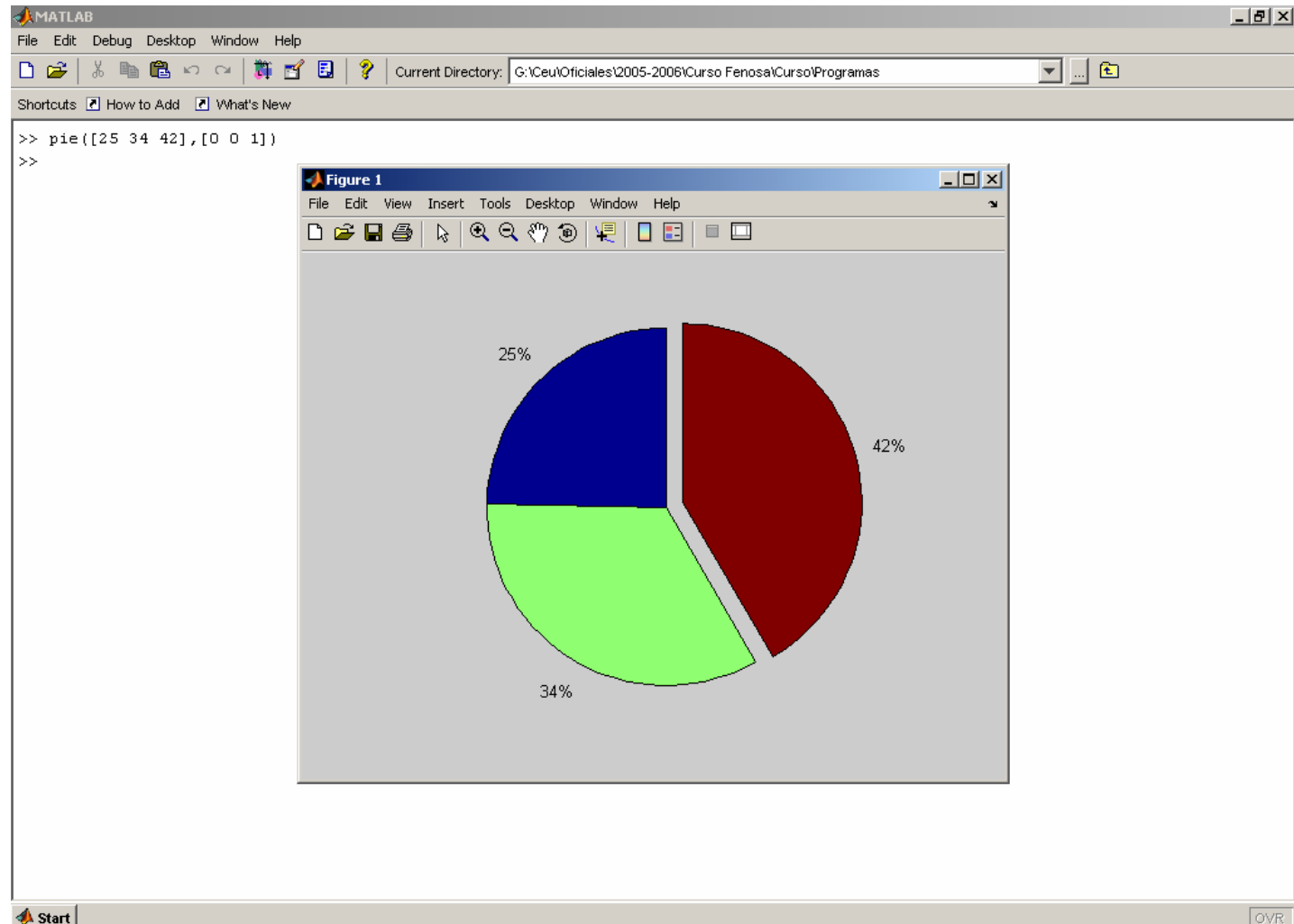
Gráficos bidimensionales



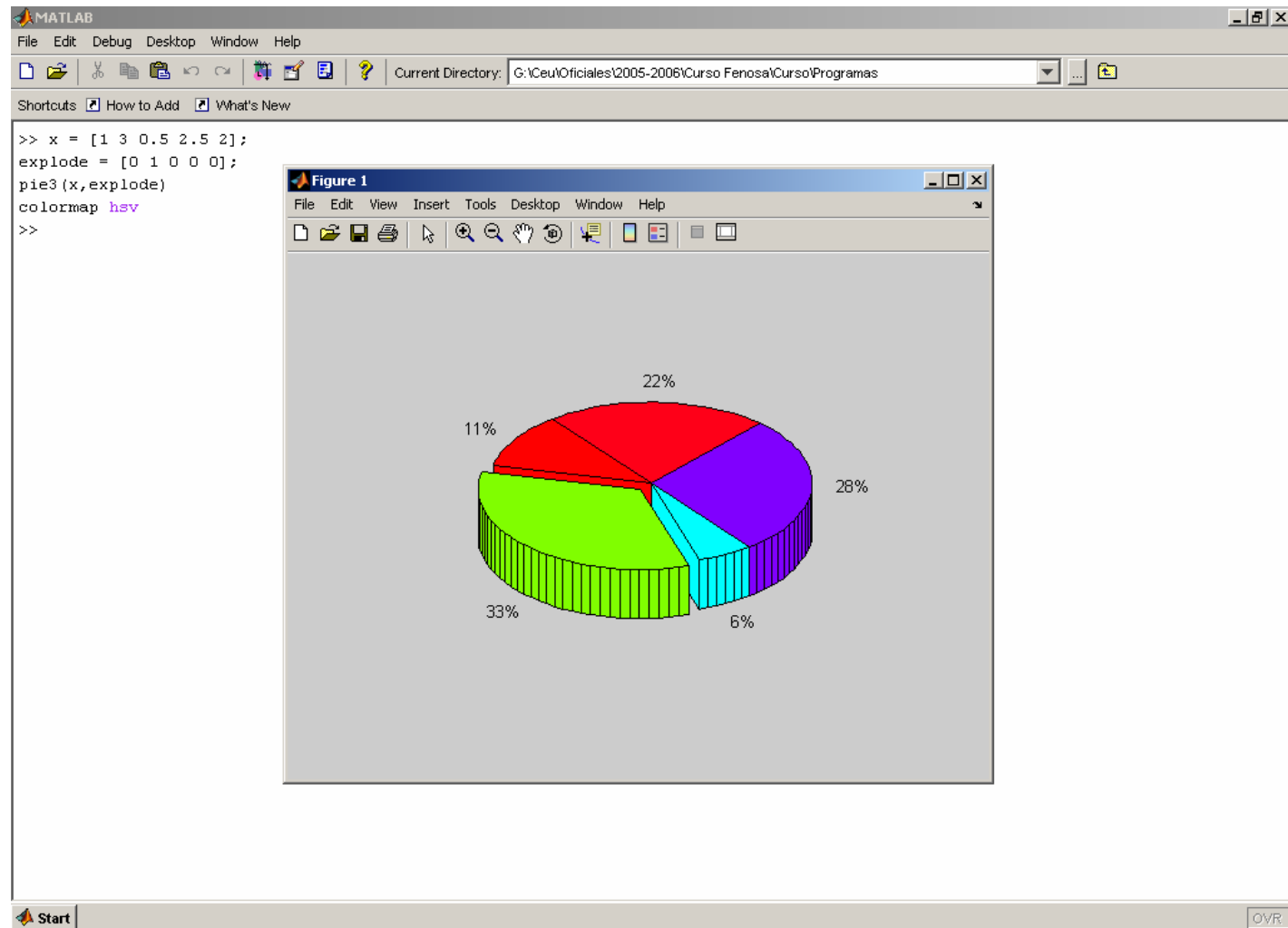
Gráficos bidimensionales



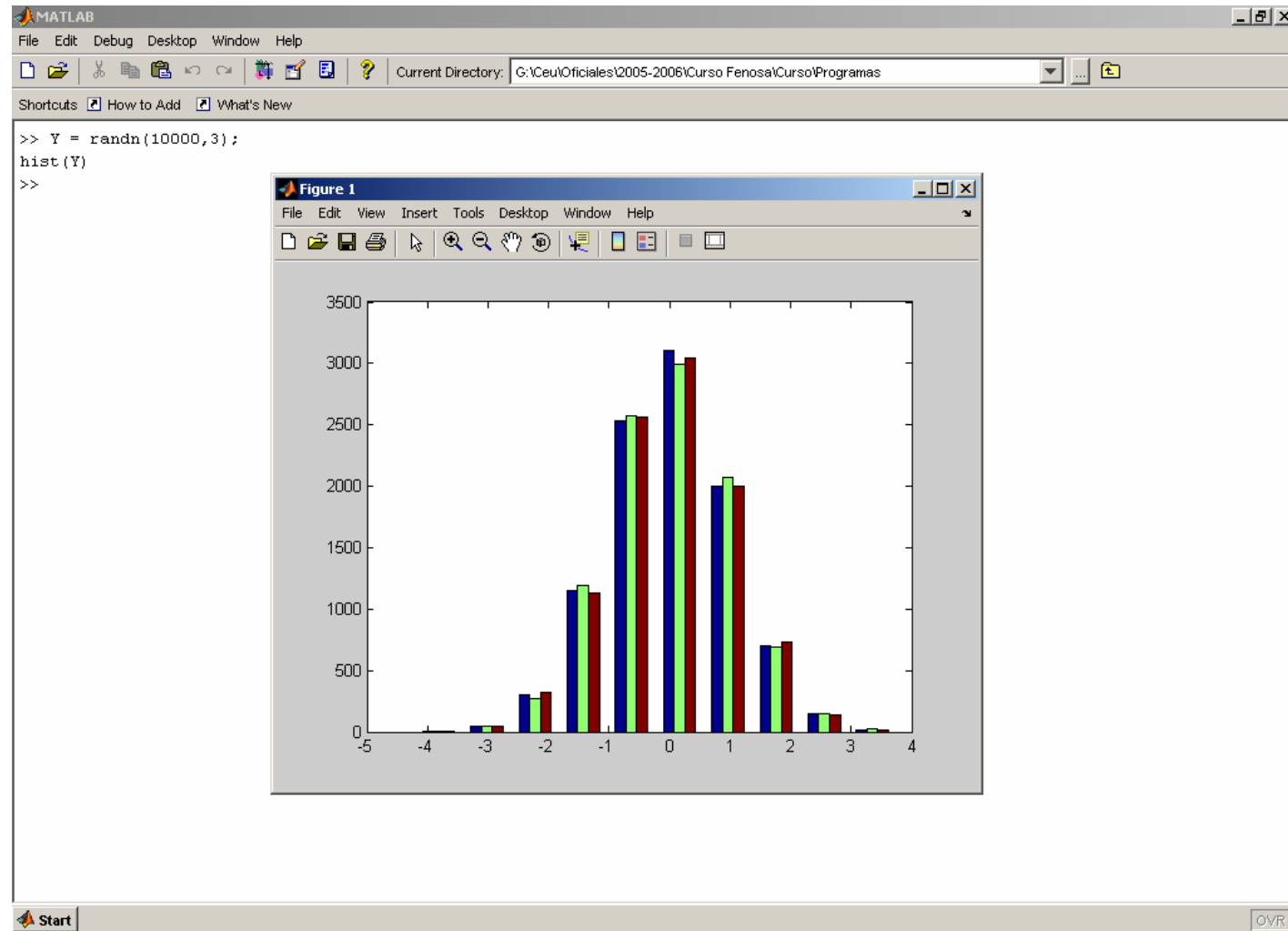
Gráficos bidimensionales



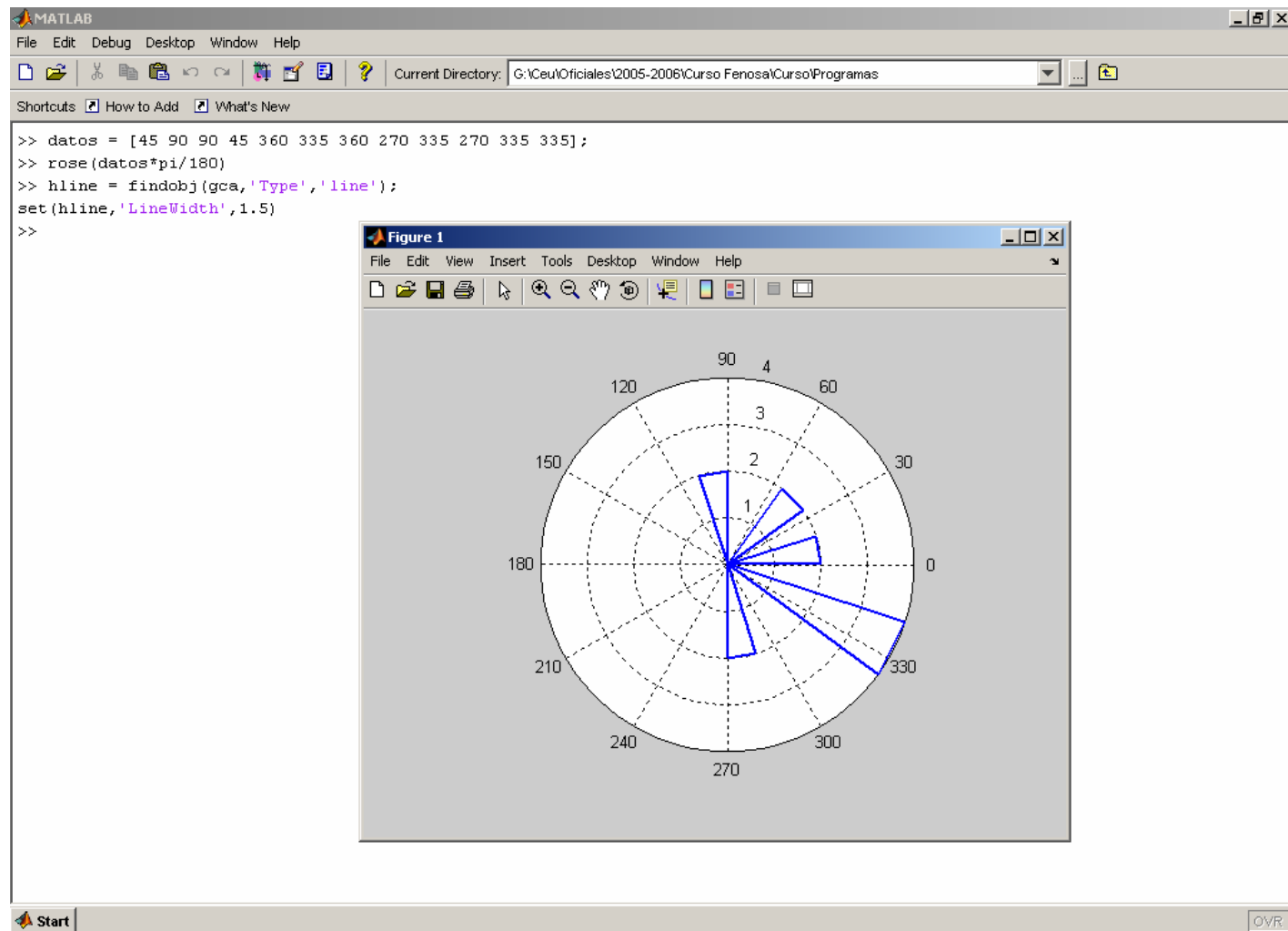
Gráficos bidimensionales



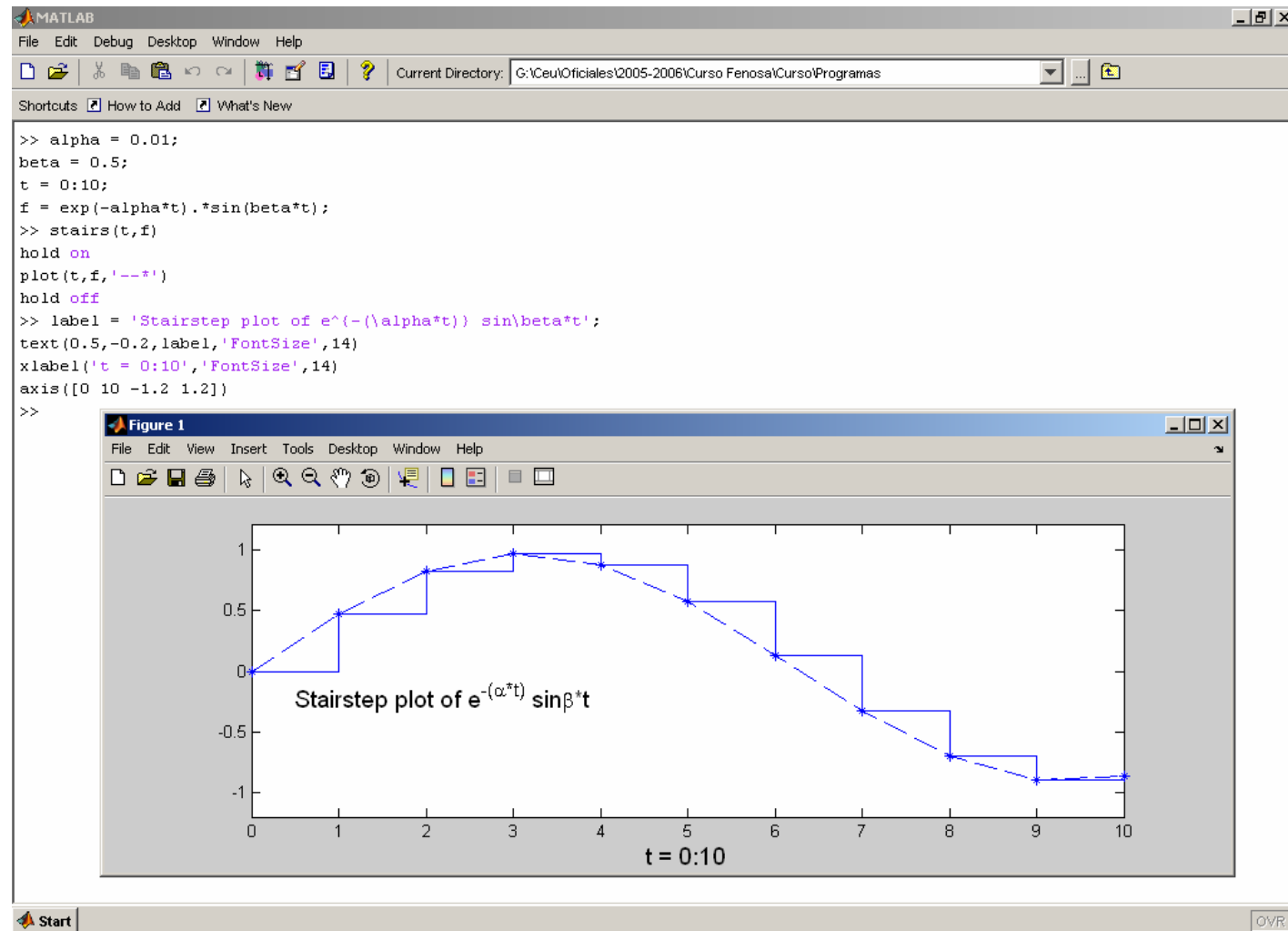
Gráficos bidimensionales



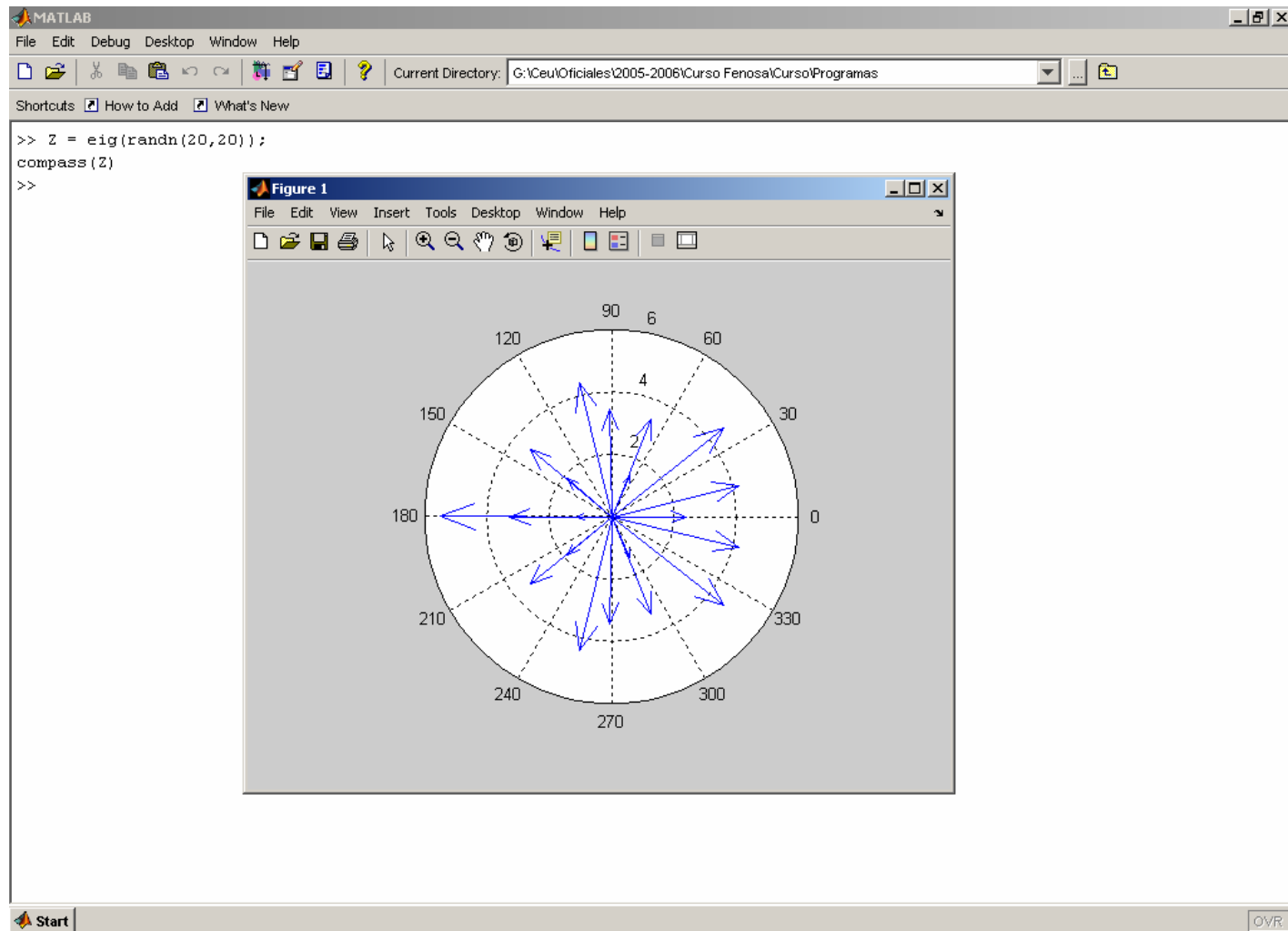
Gráficos bidimensionales



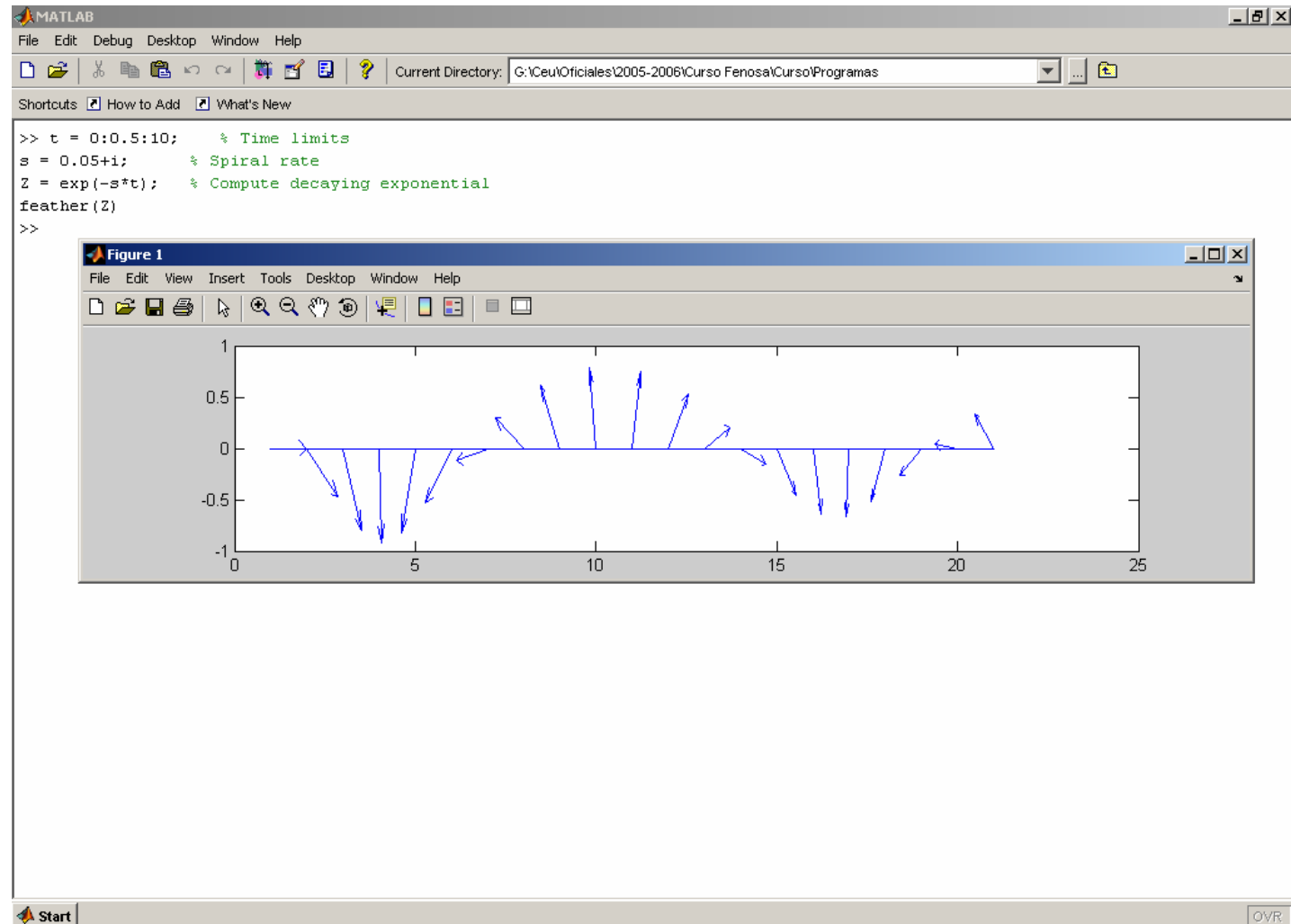
Gráficos bidimensionales



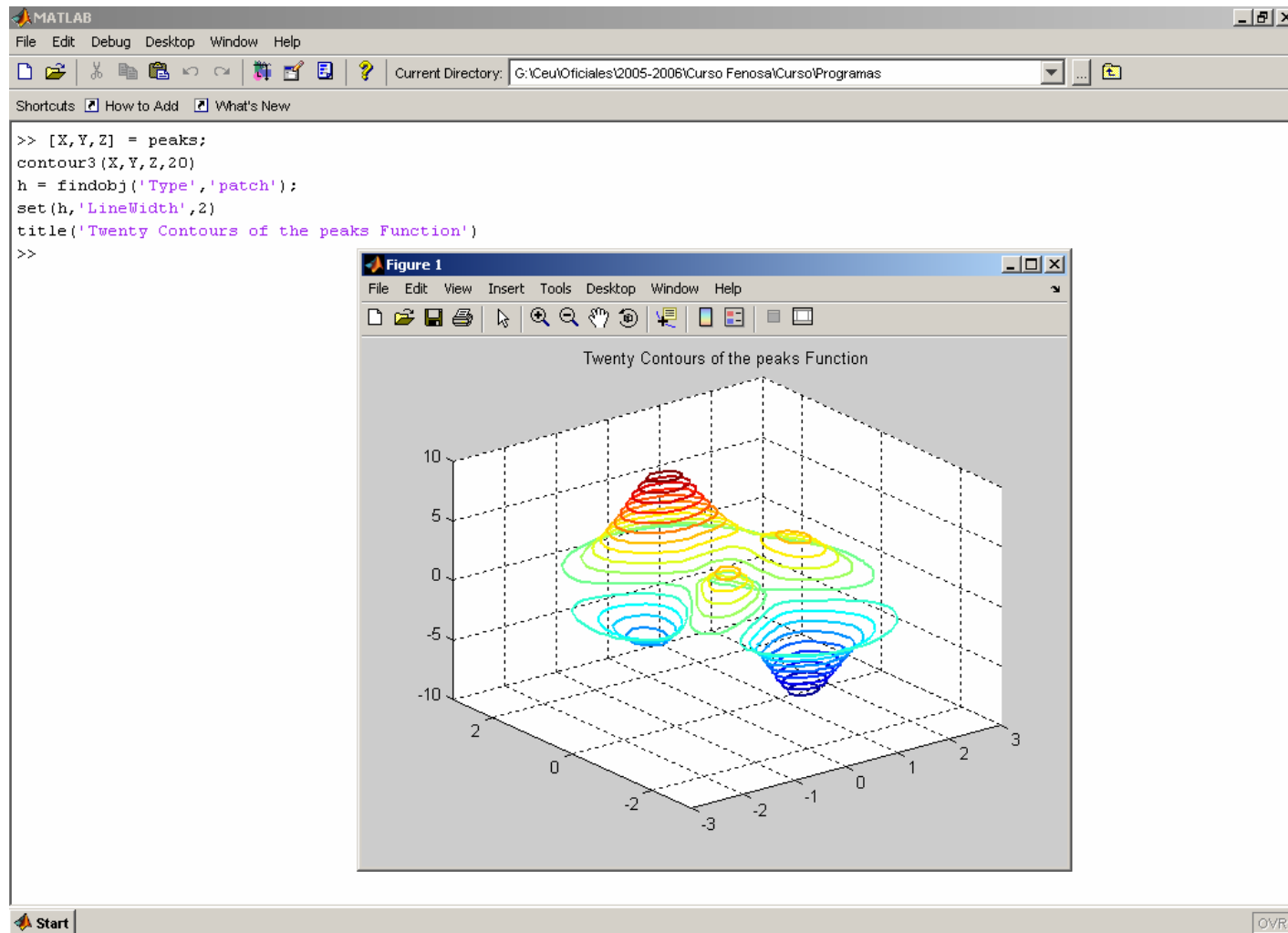
Gráficos bidimensionales



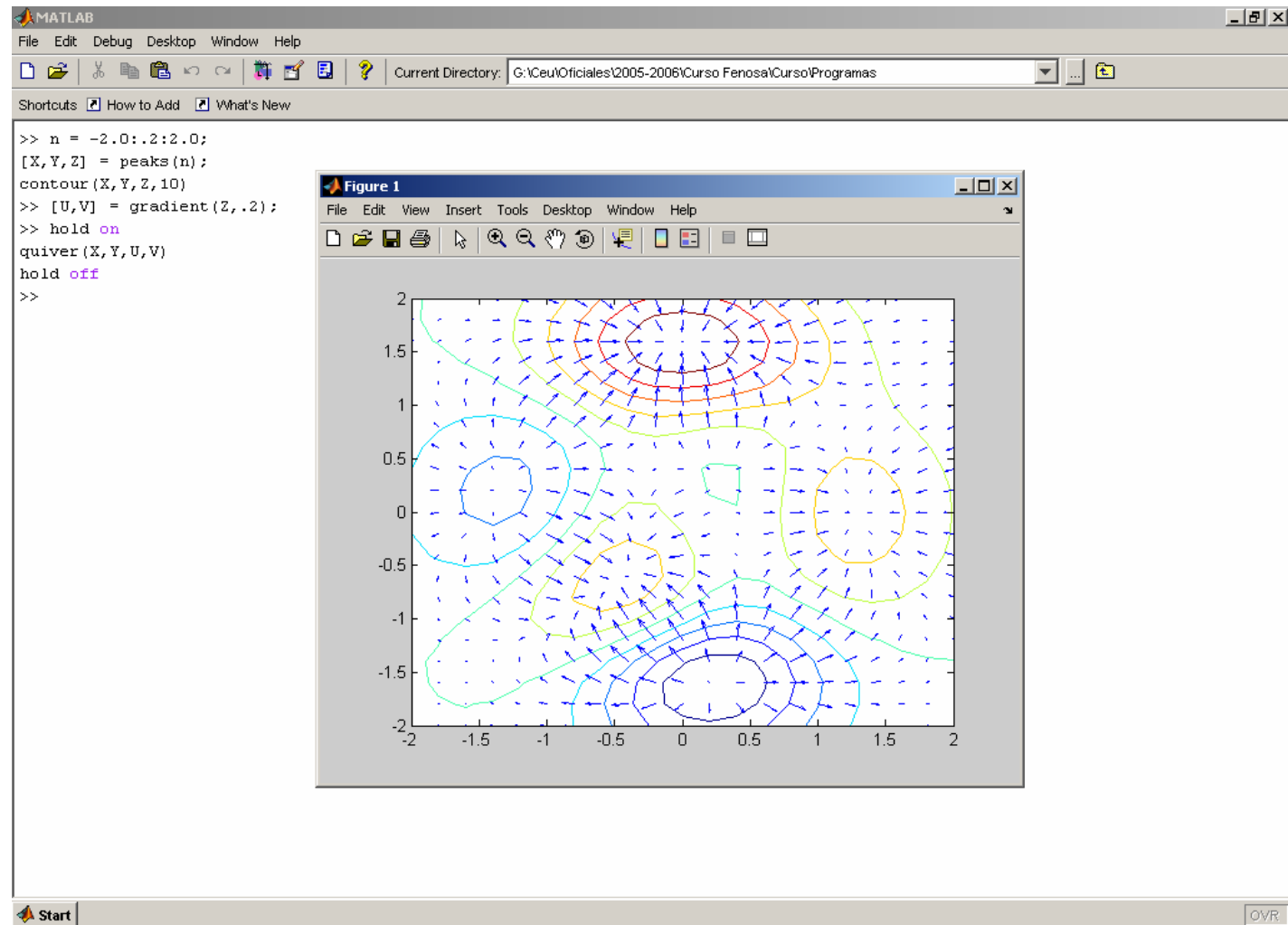
Gráficos bidimensionales



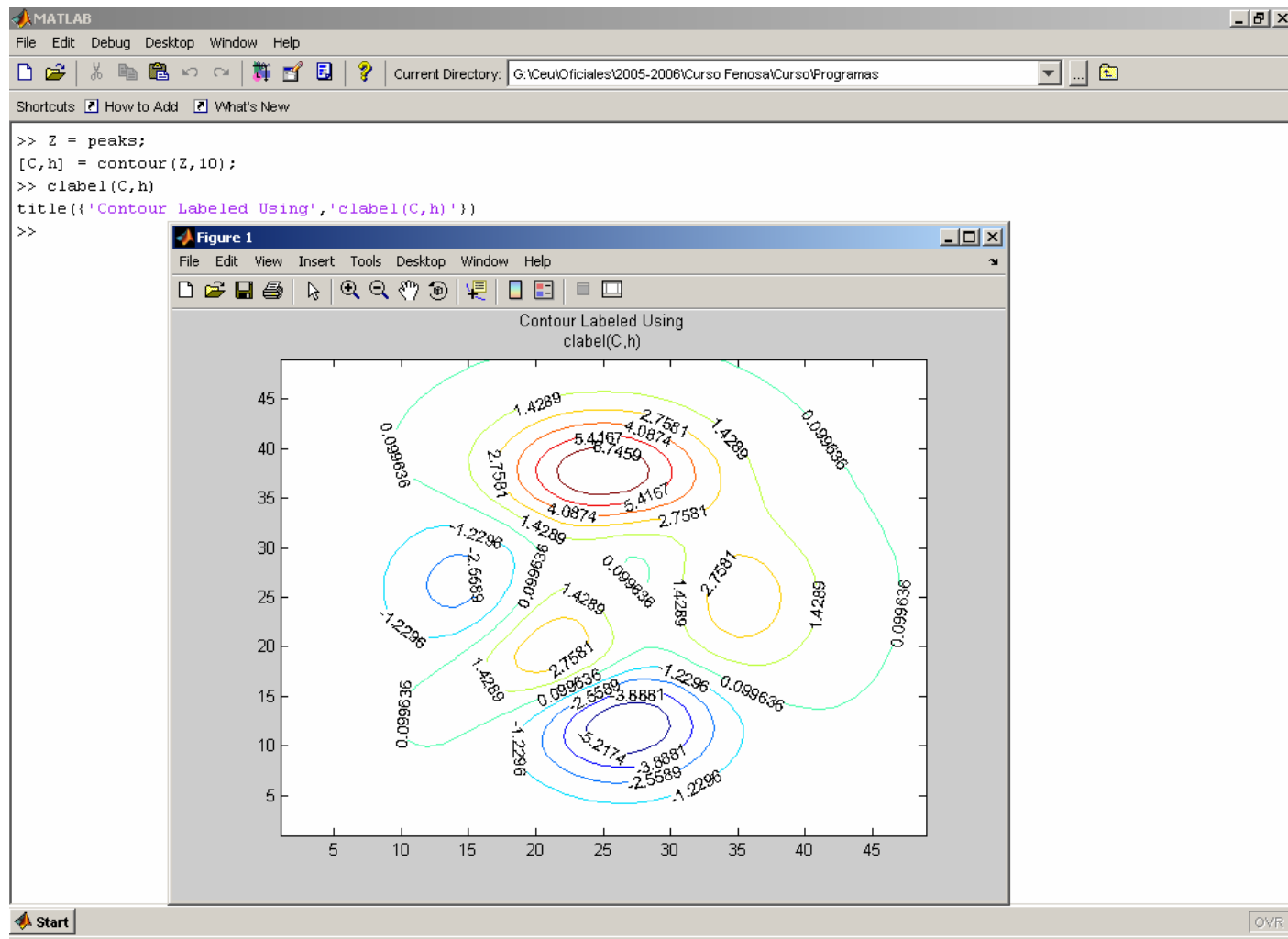
Gráficos tridimensionales



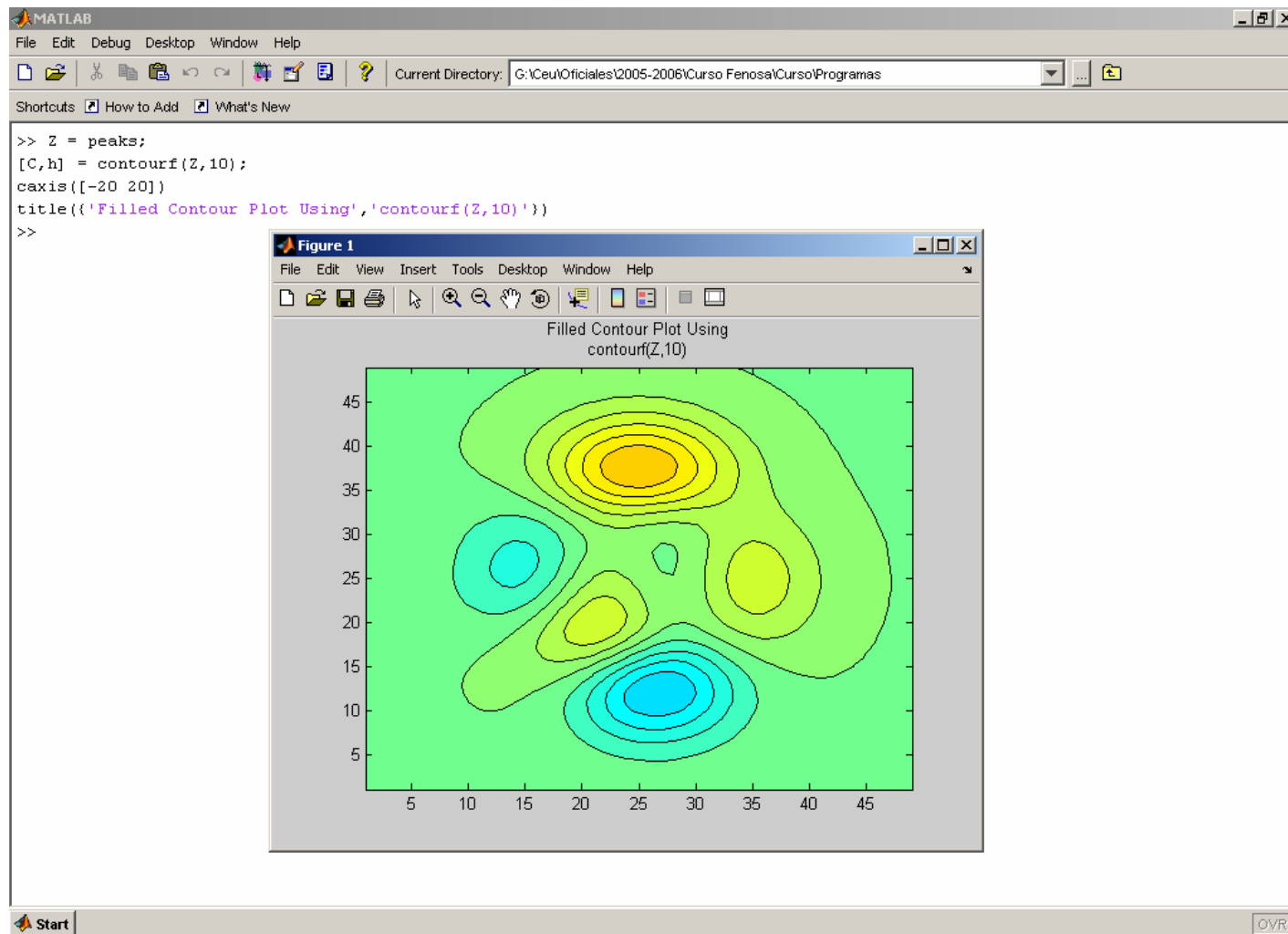
Gráficos bidimensionales



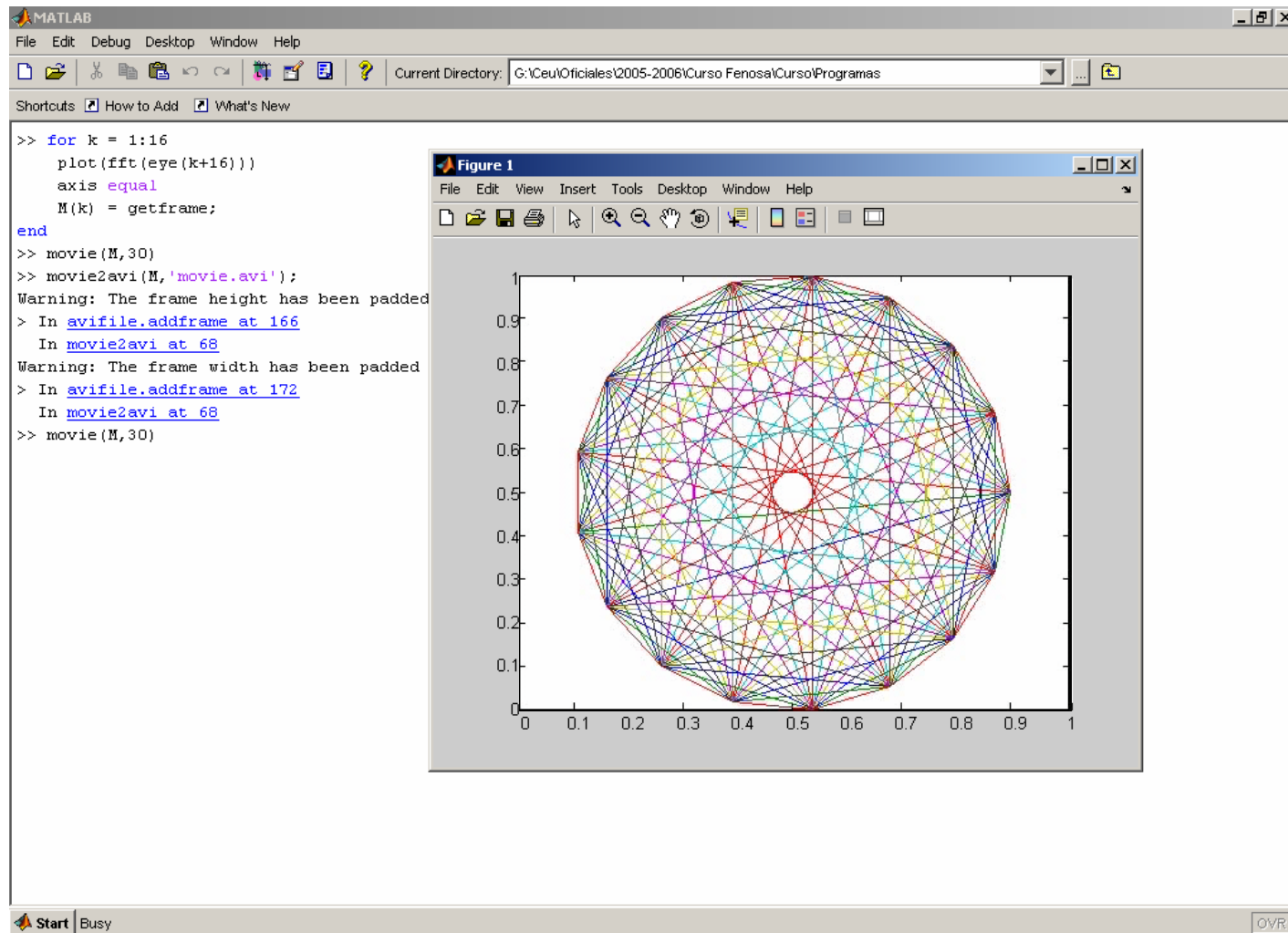
Gráficos bidimensionales



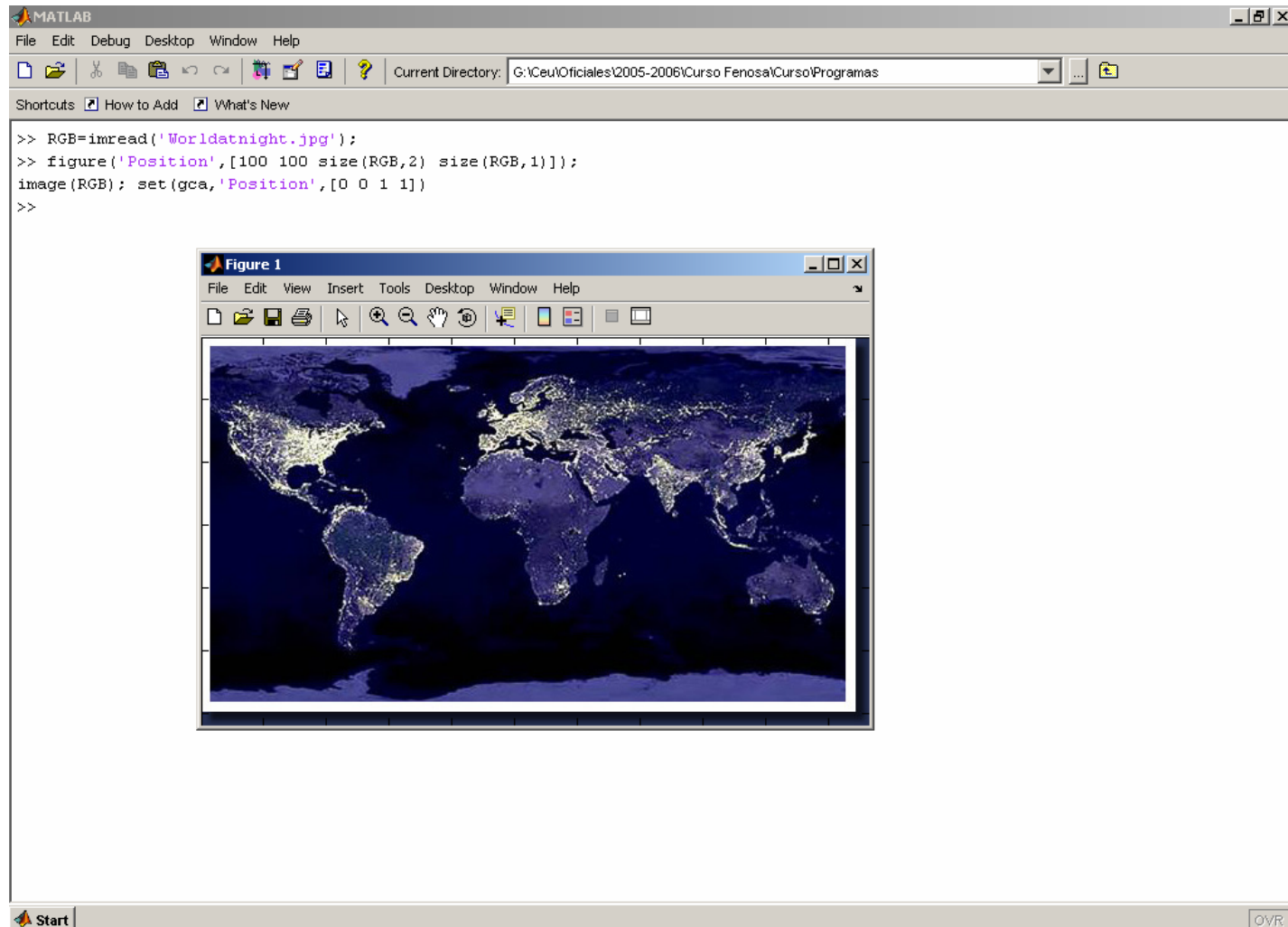
Gráficos bidimensionales



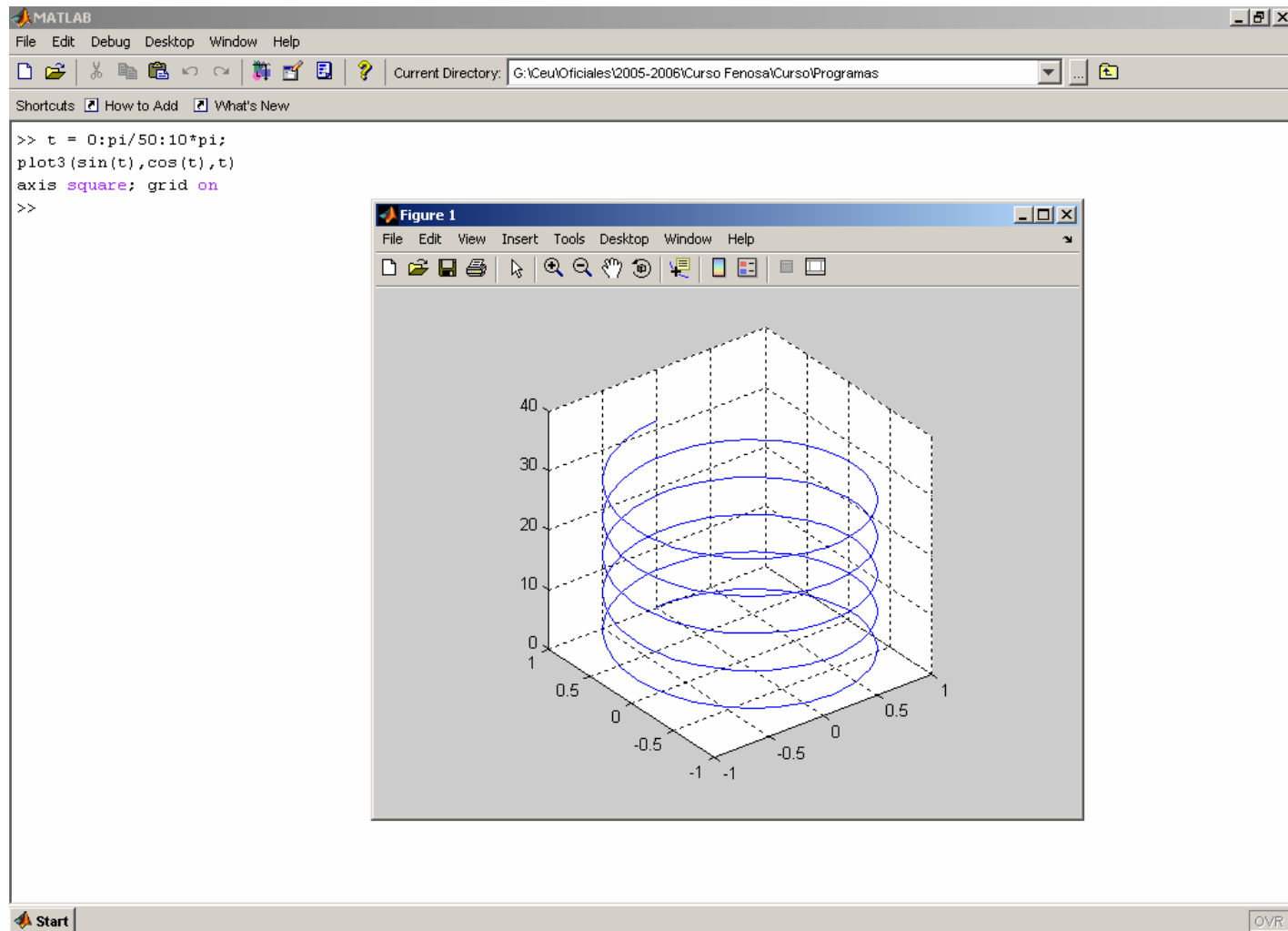
Animaciones



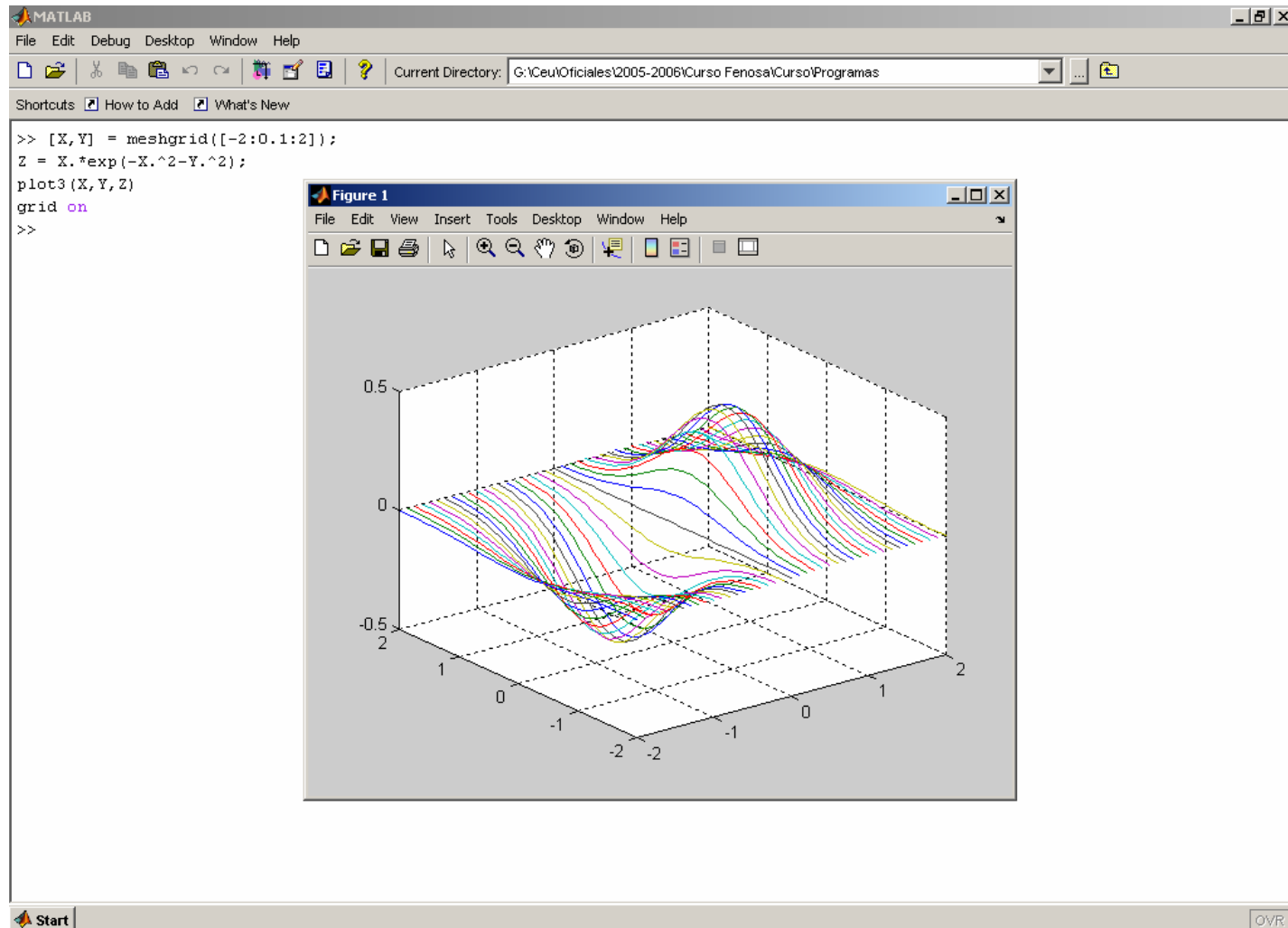
Imágenes



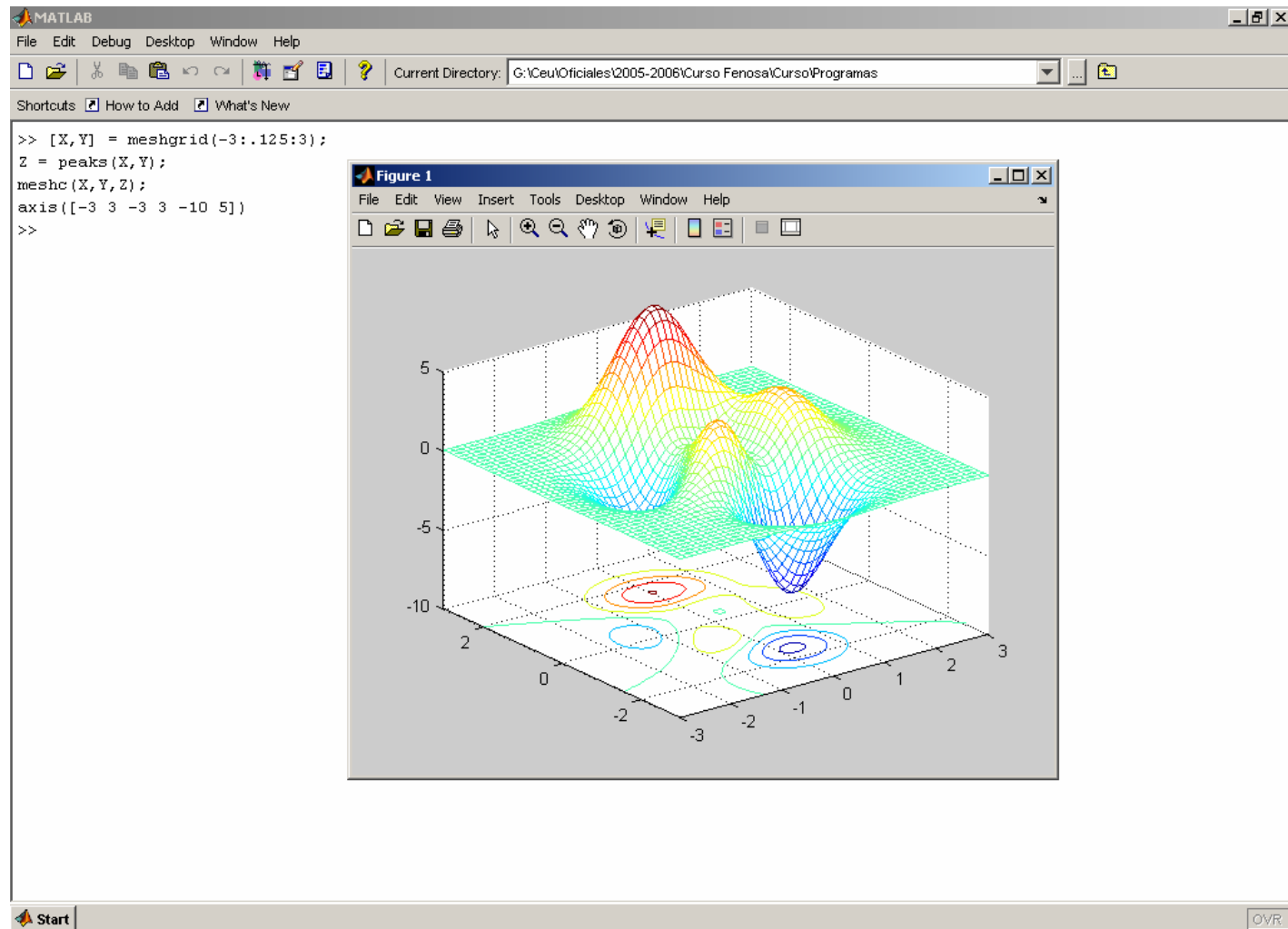
Gráficos tridimensionales



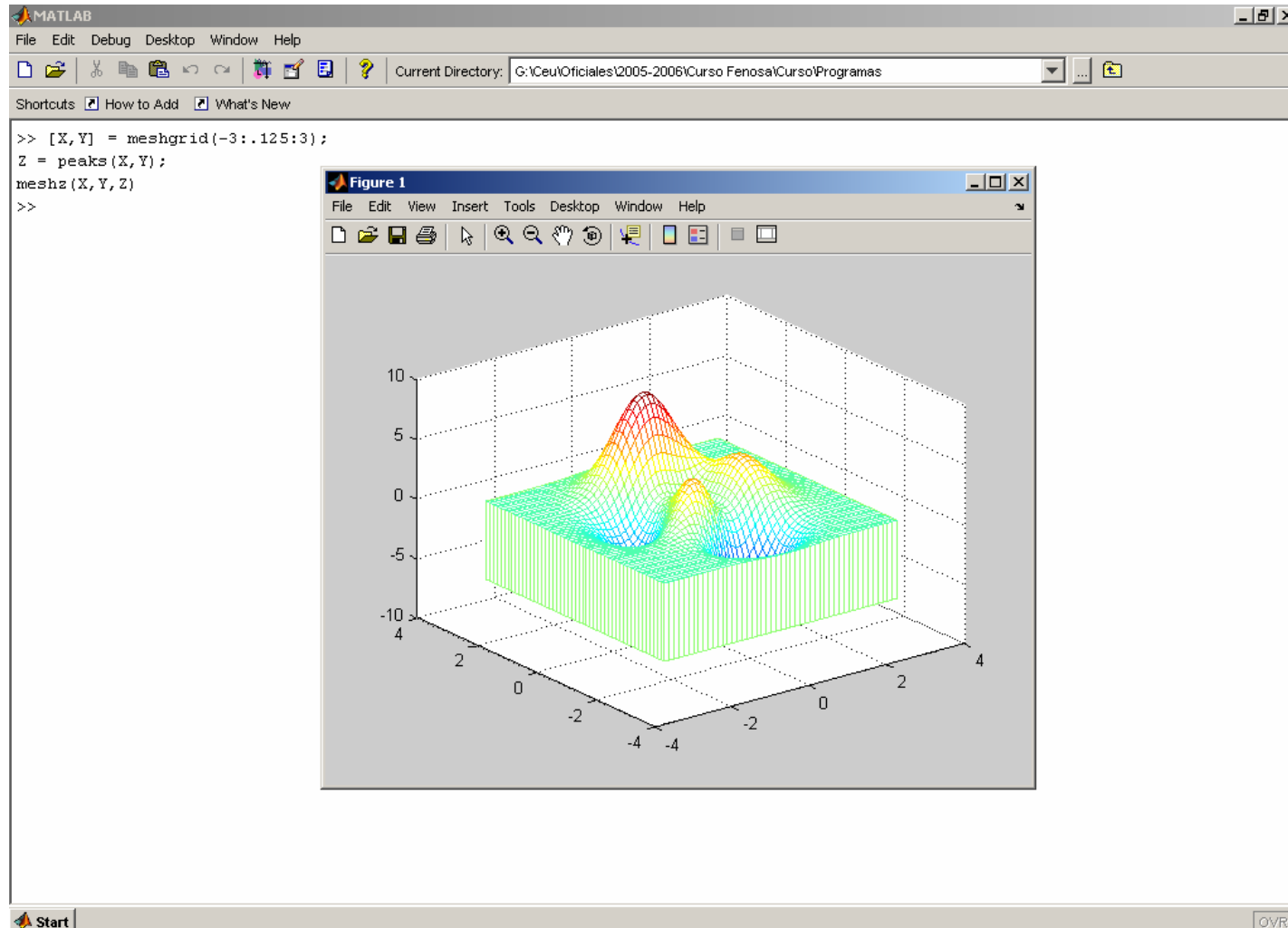
Gráficos tridimensionales



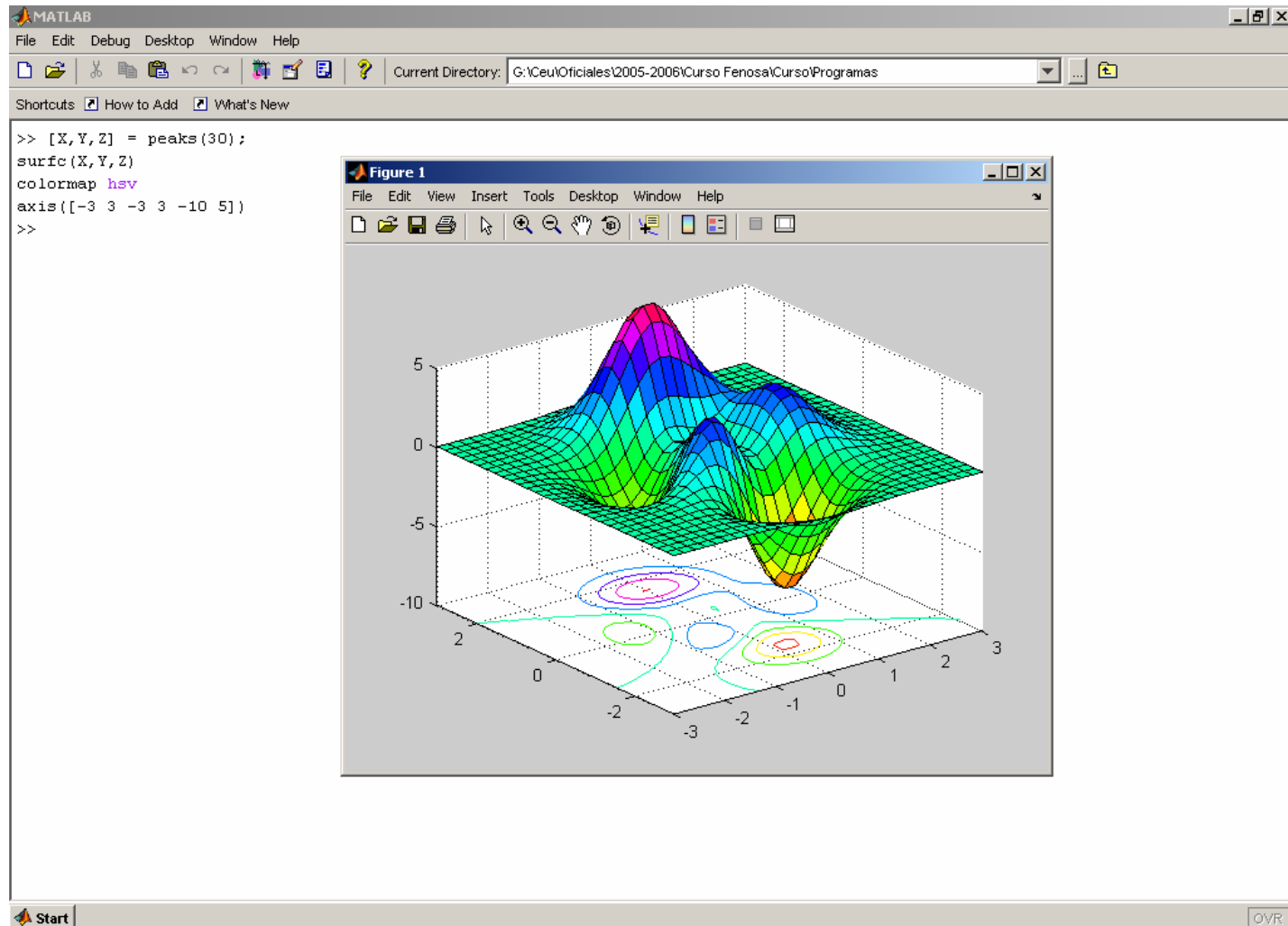
Gráficos tridimensionales



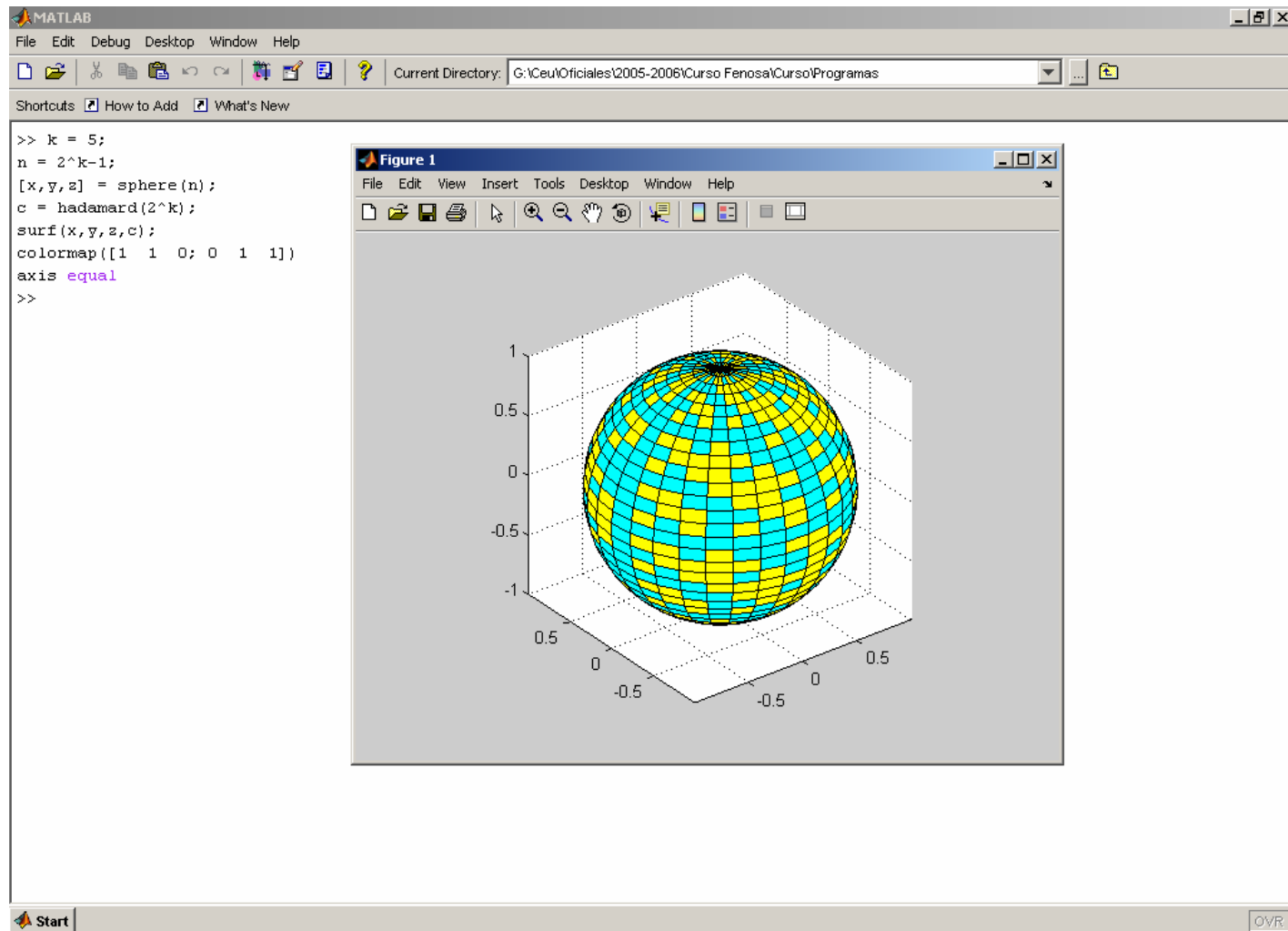
Gráficos tridimensionales



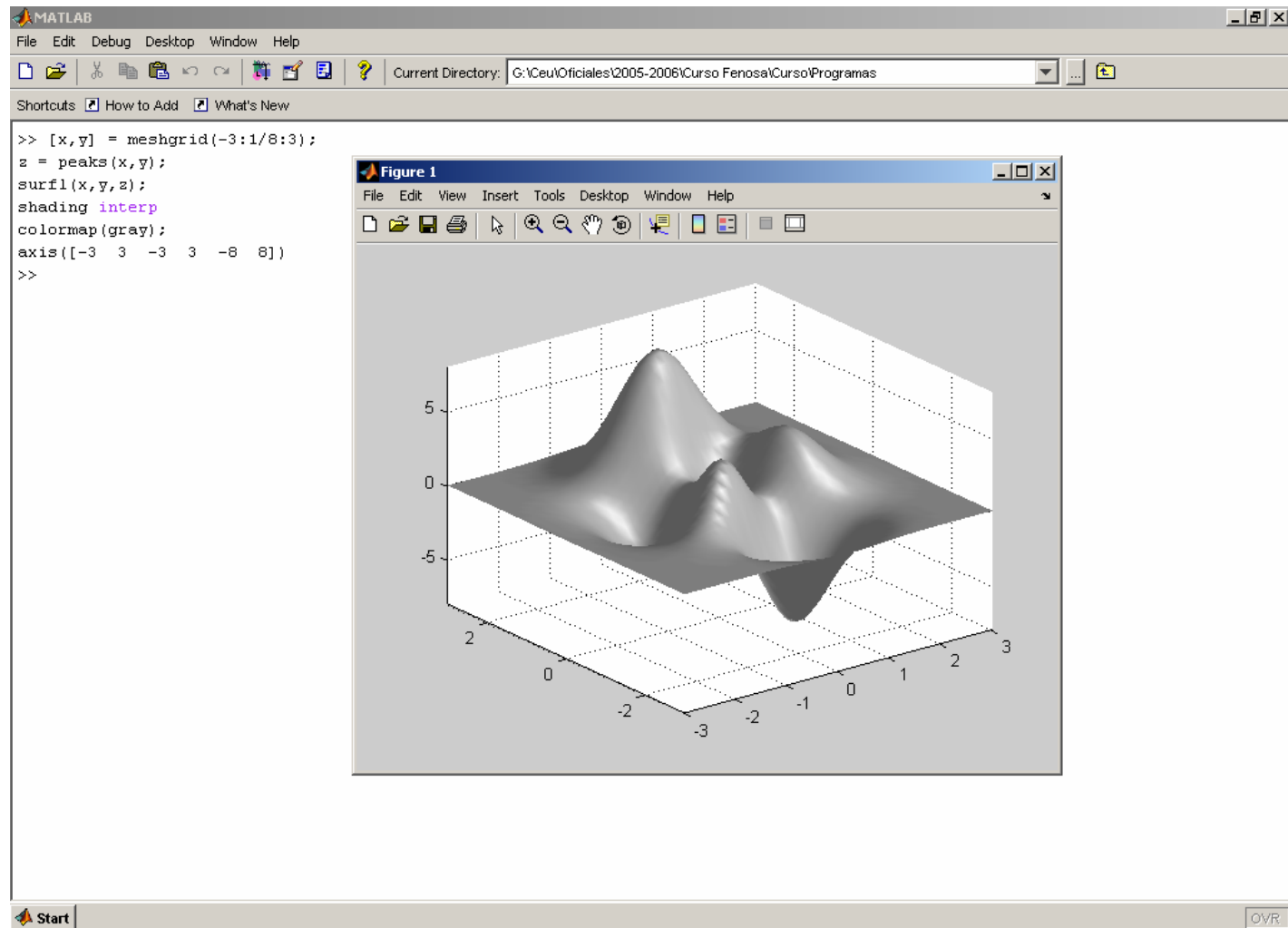
Gráficos tridimensionales



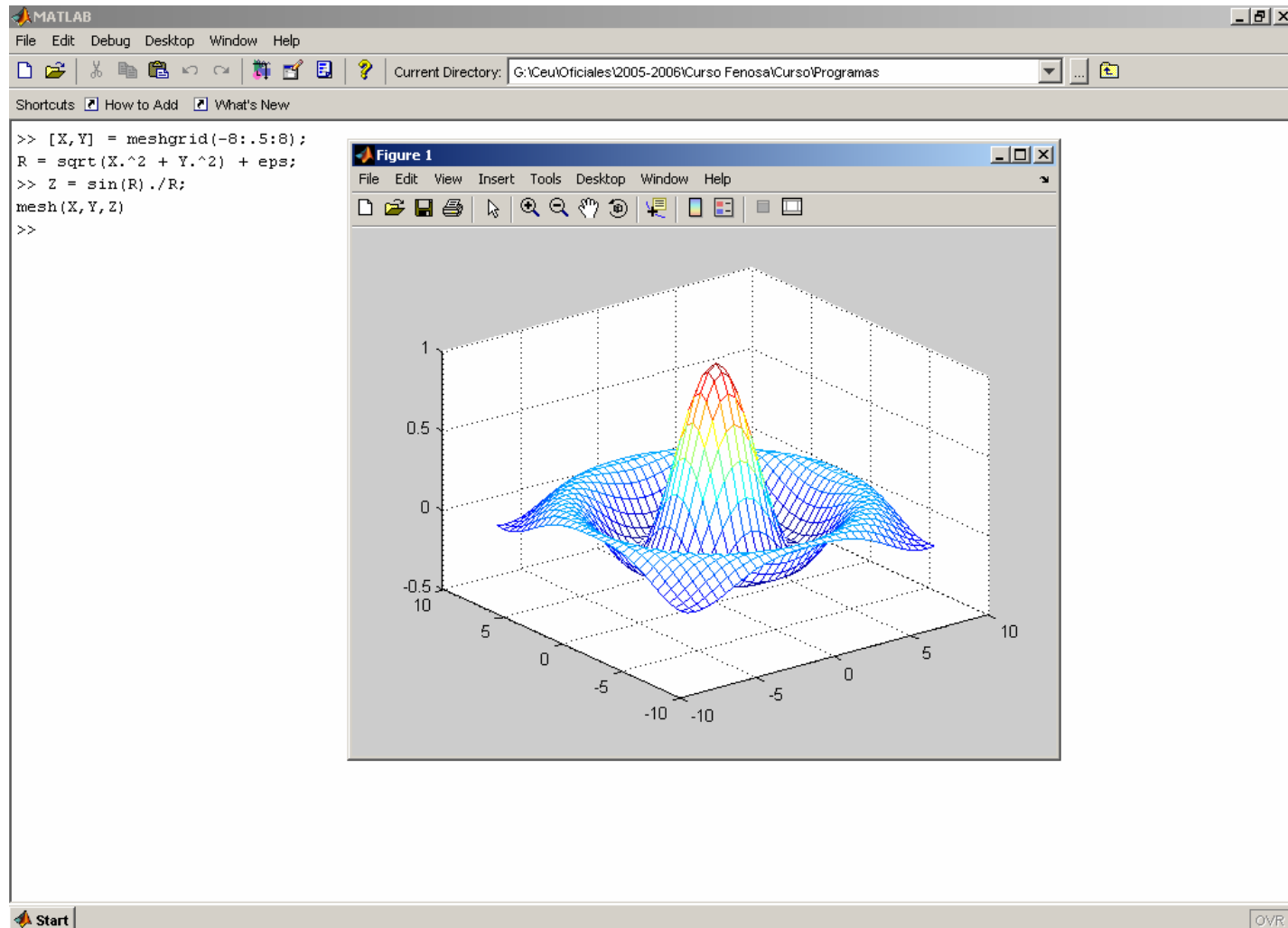
Gráficos tridimensionales



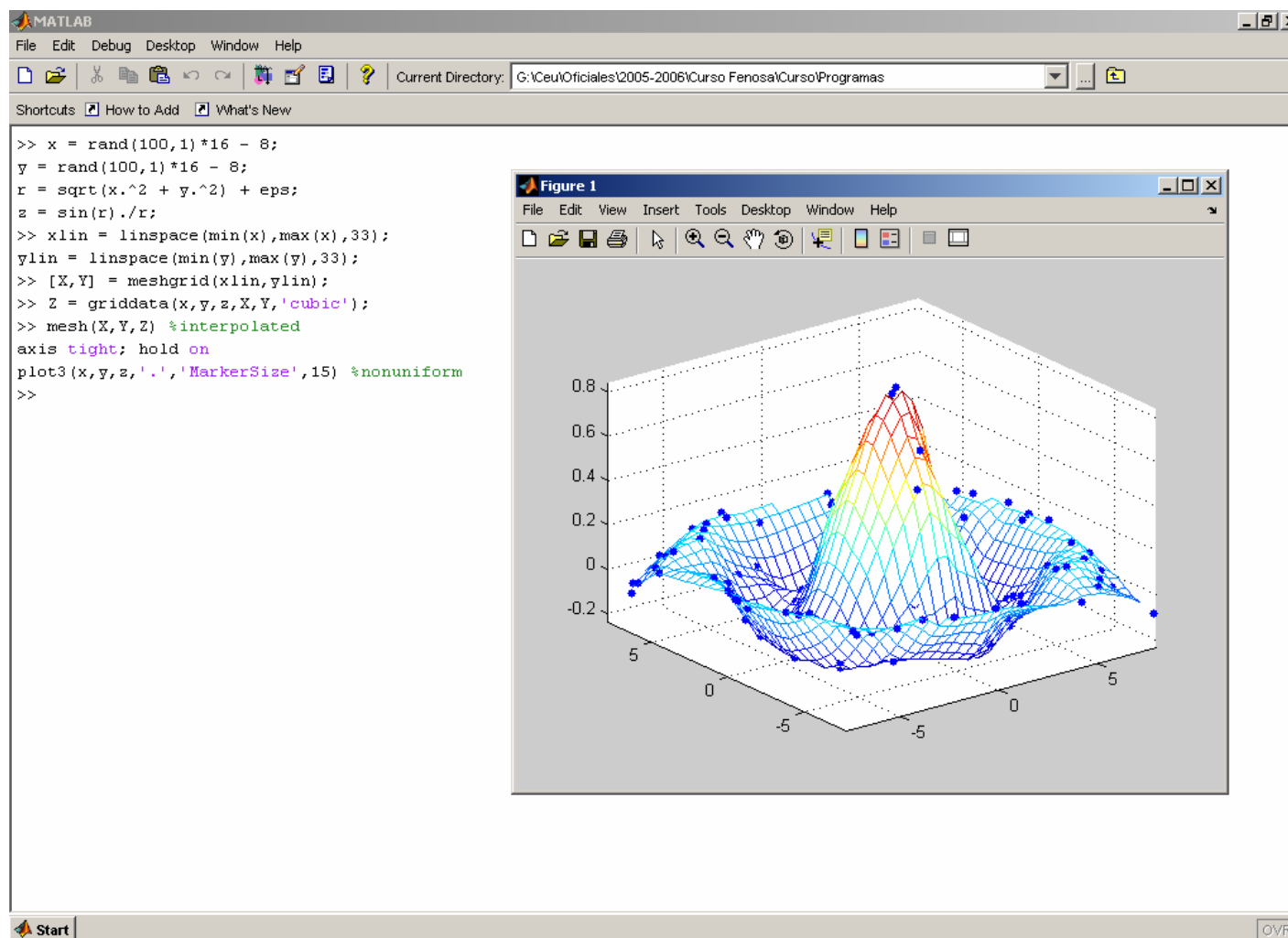
Gráficos tridimensionales



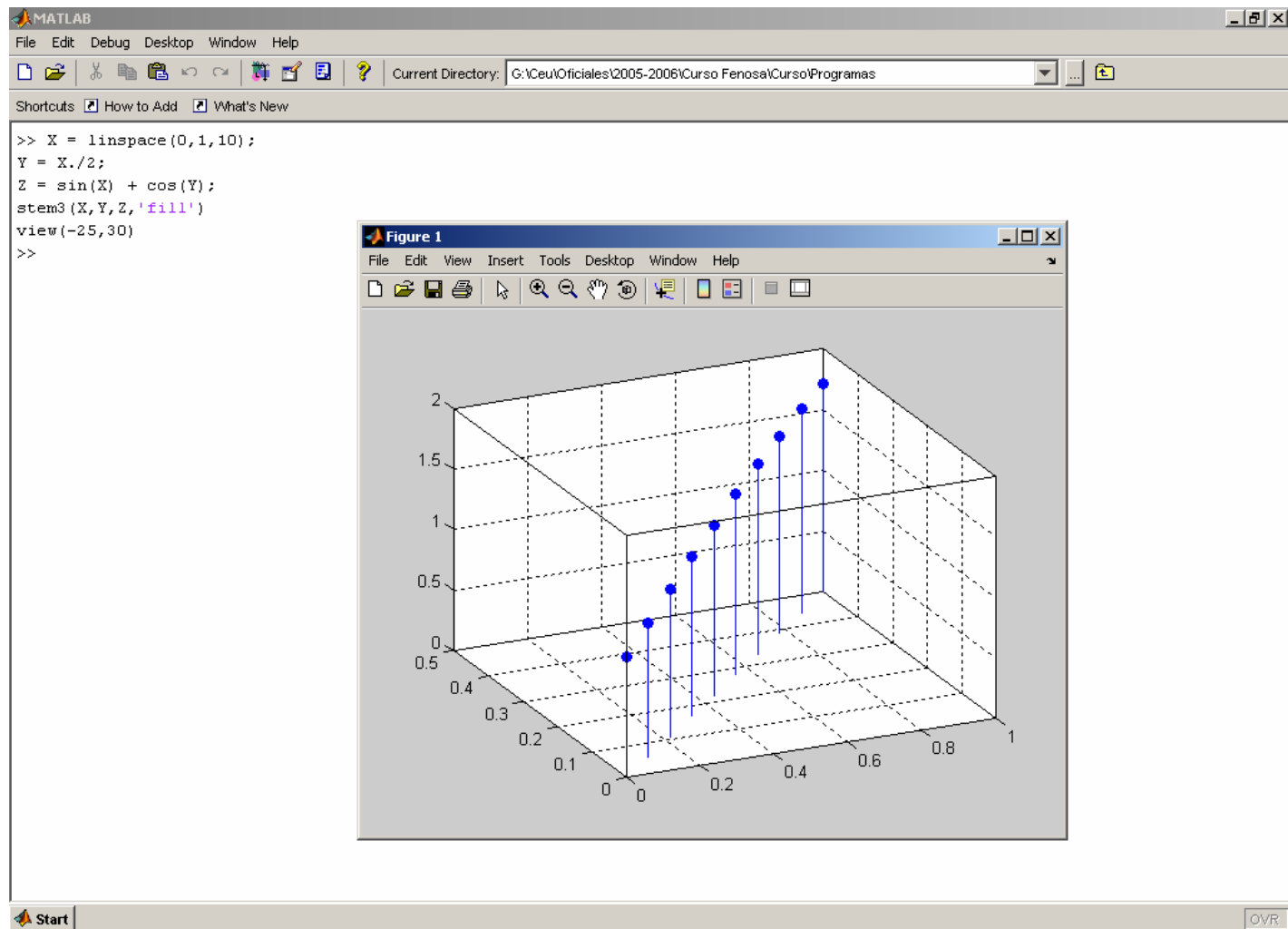
Gráficos tridimensionales



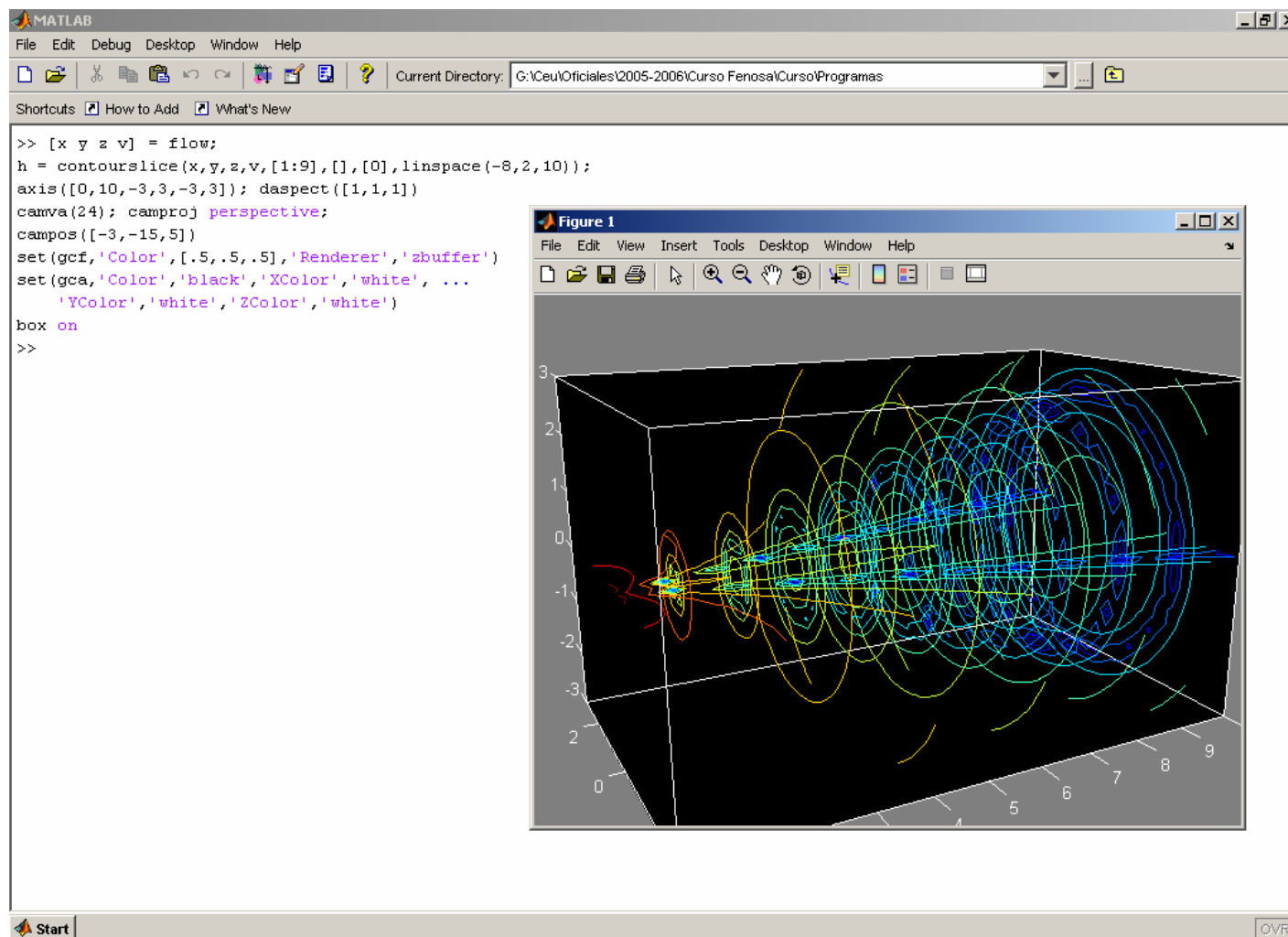
Gráficos tridimensionales



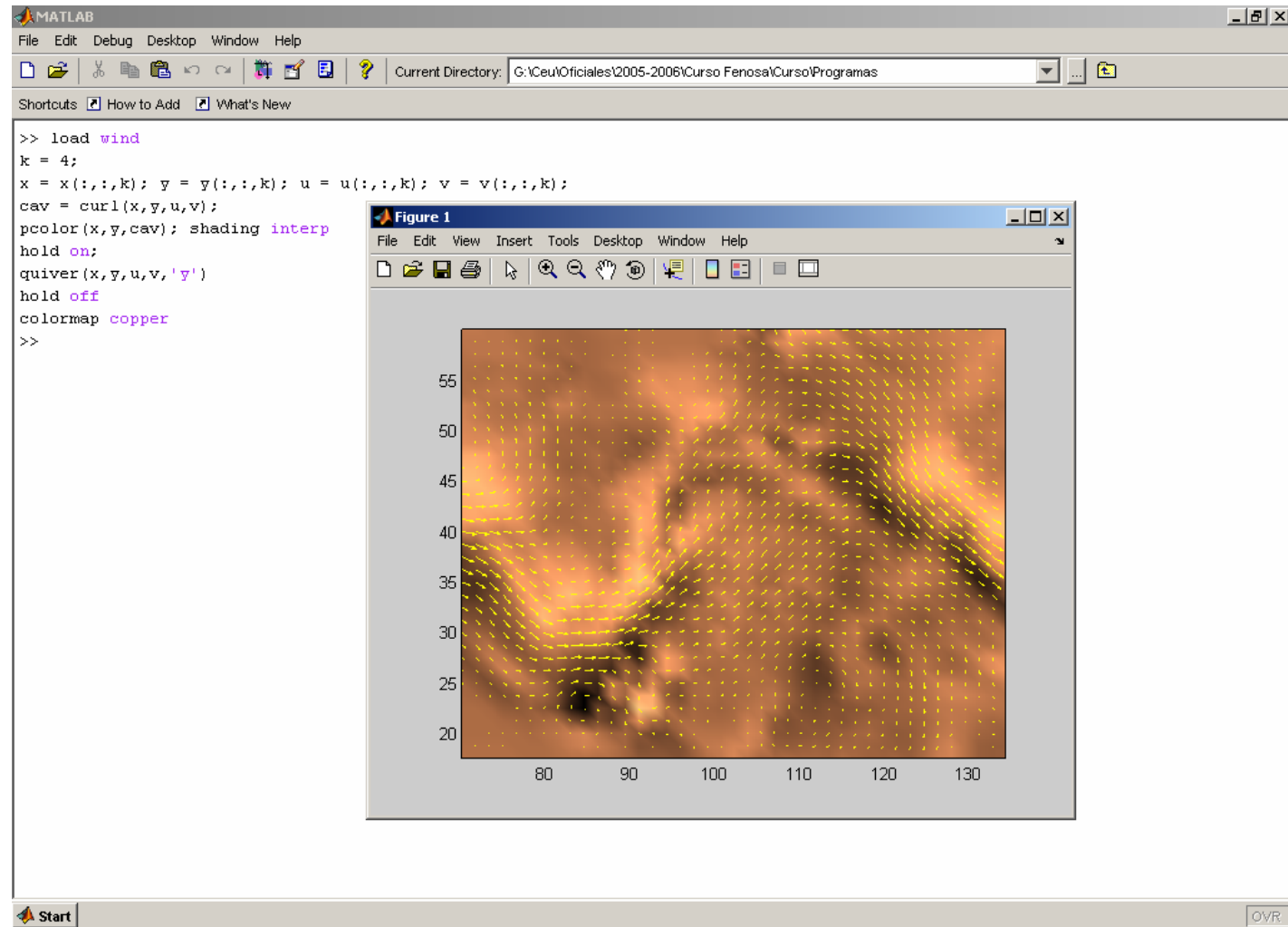
Gráficos tridimensionales



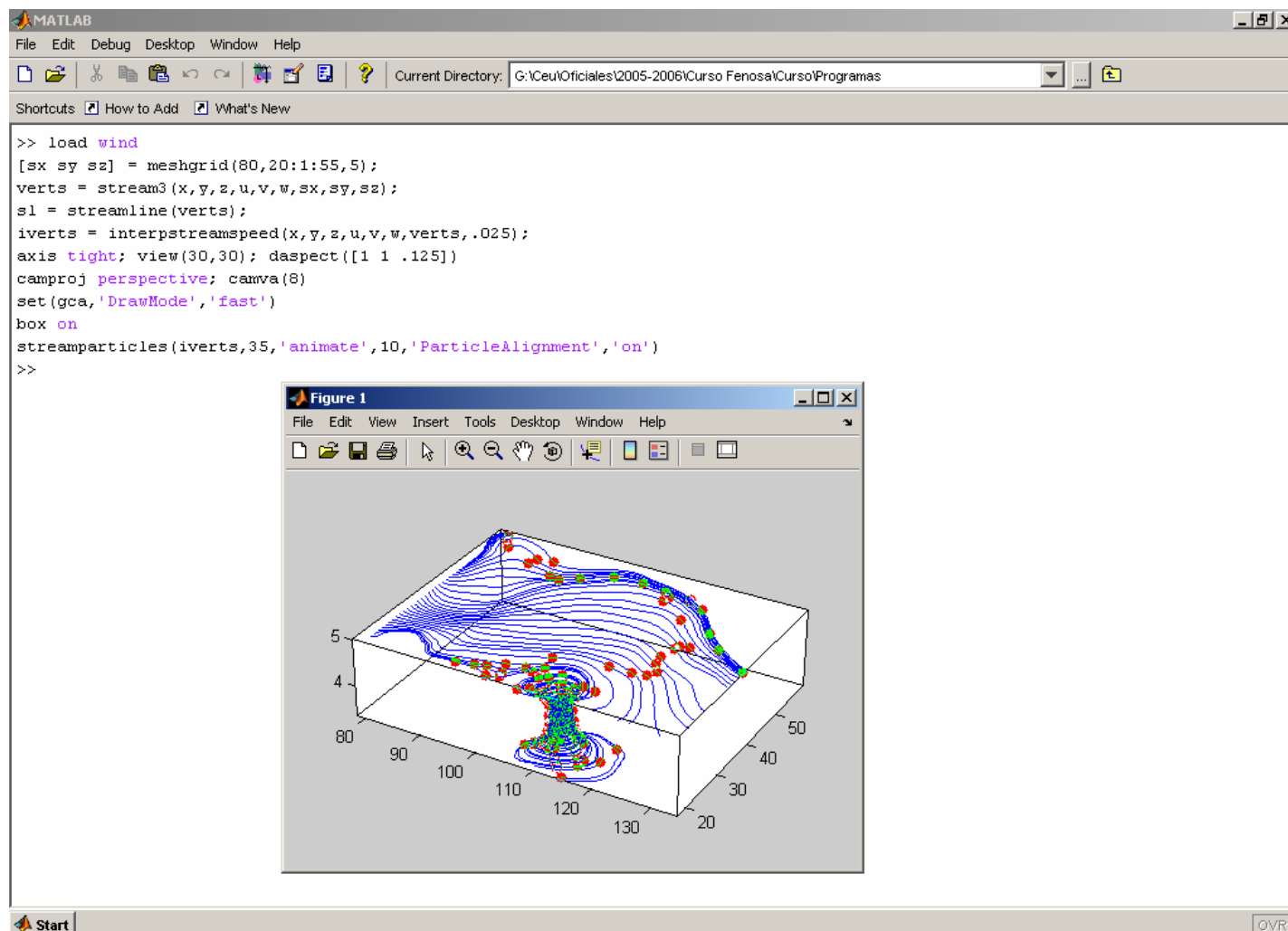
Gráficos tridimensionales



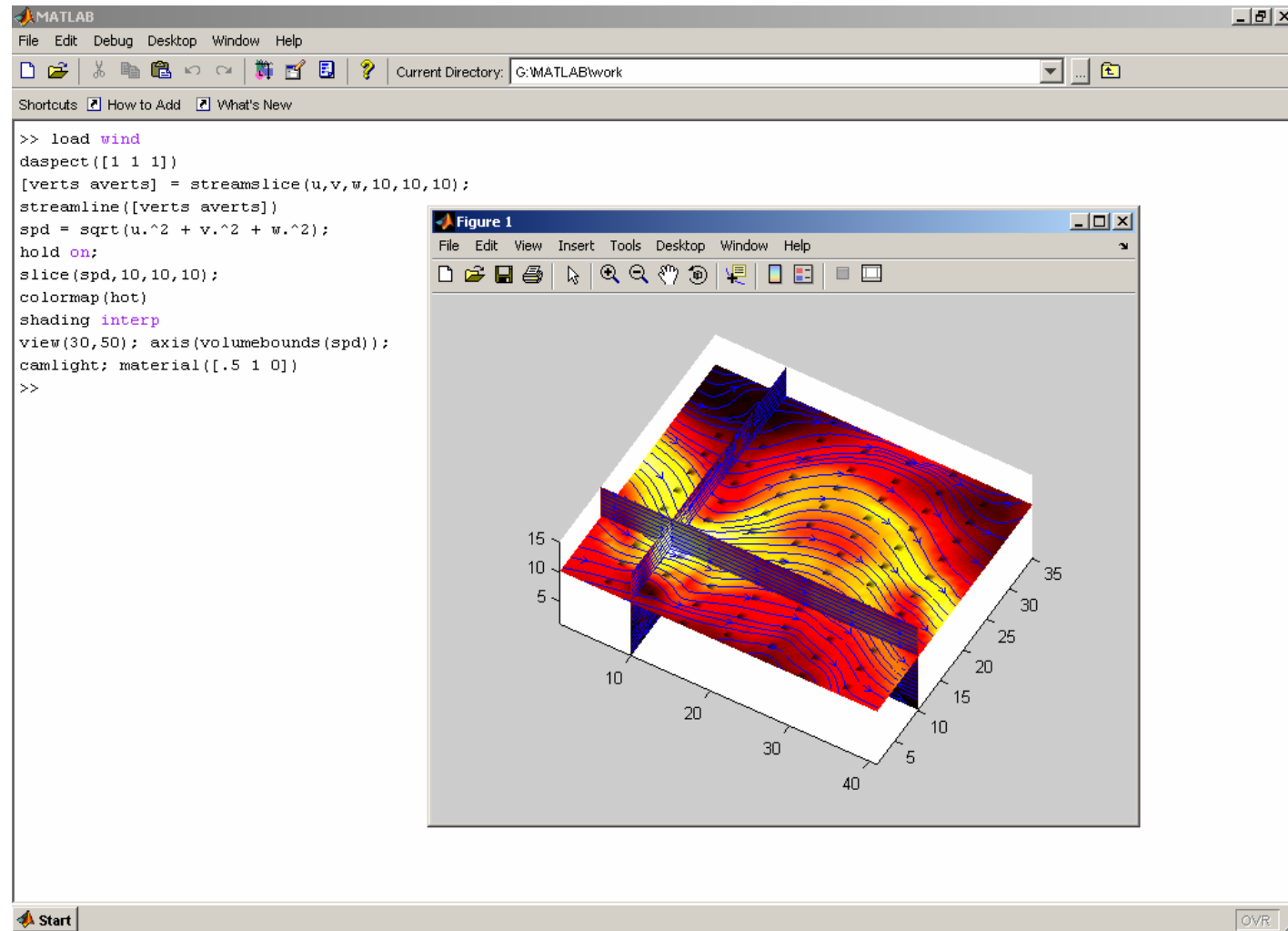
Gráficos tridimensionales



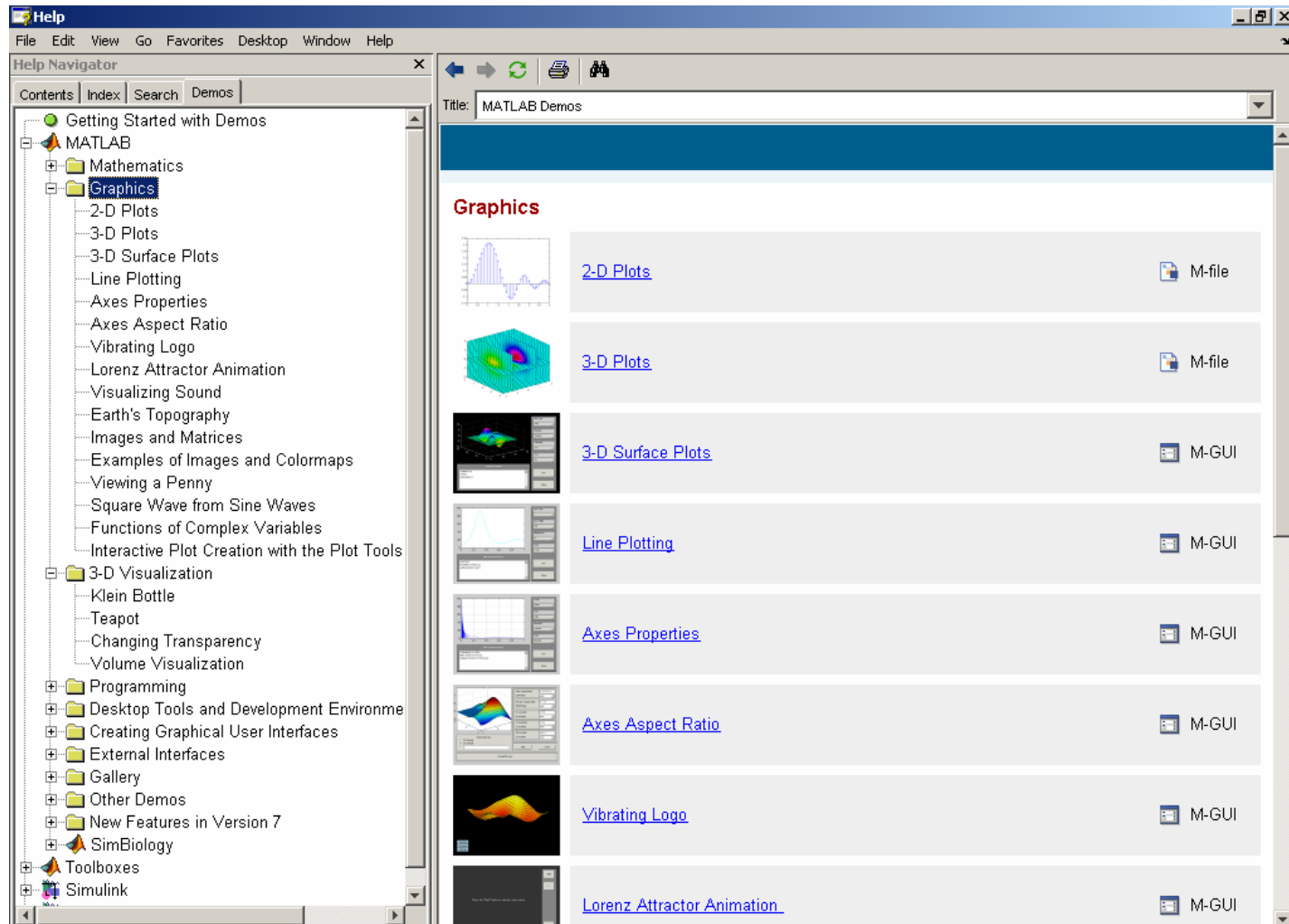
Gráficos tridimensionales



Gráficos tridimensionales

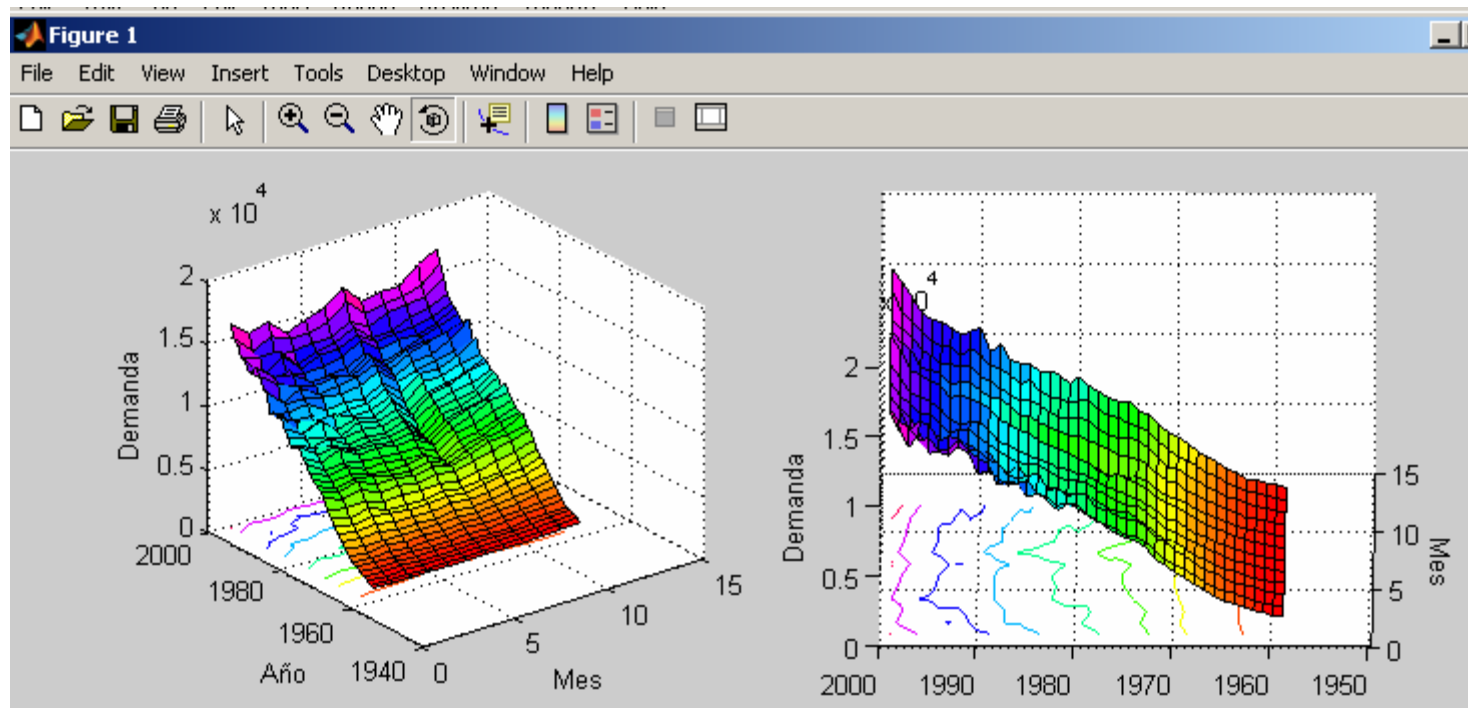


Demos



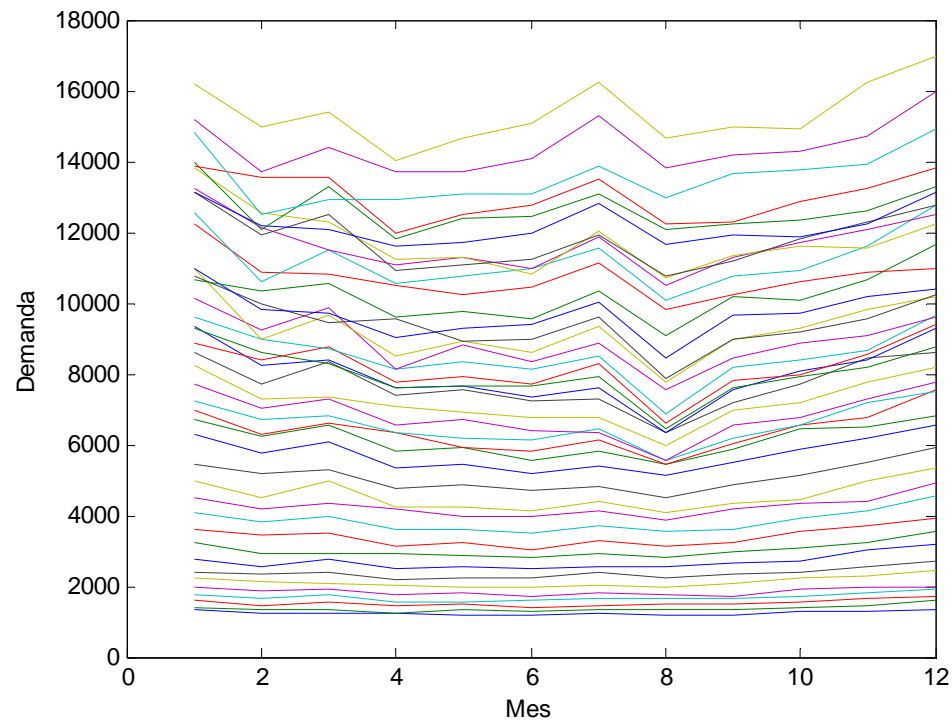
Ejercicio Final 1:

- Representar la demanda frente a los años y los meses



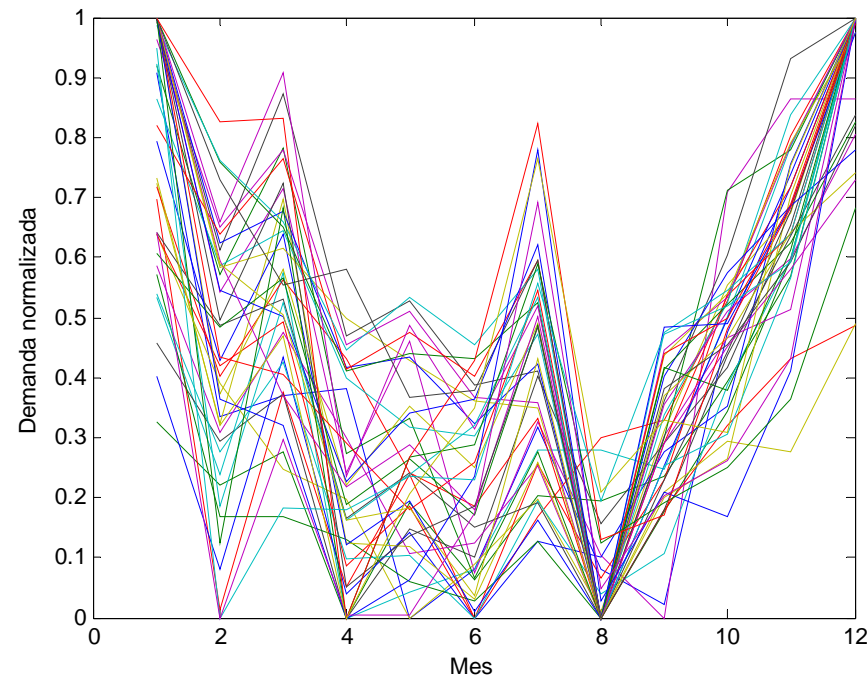
Ejercicio Final 2

- Representar en un mismo gráfico todos los consumos anuales



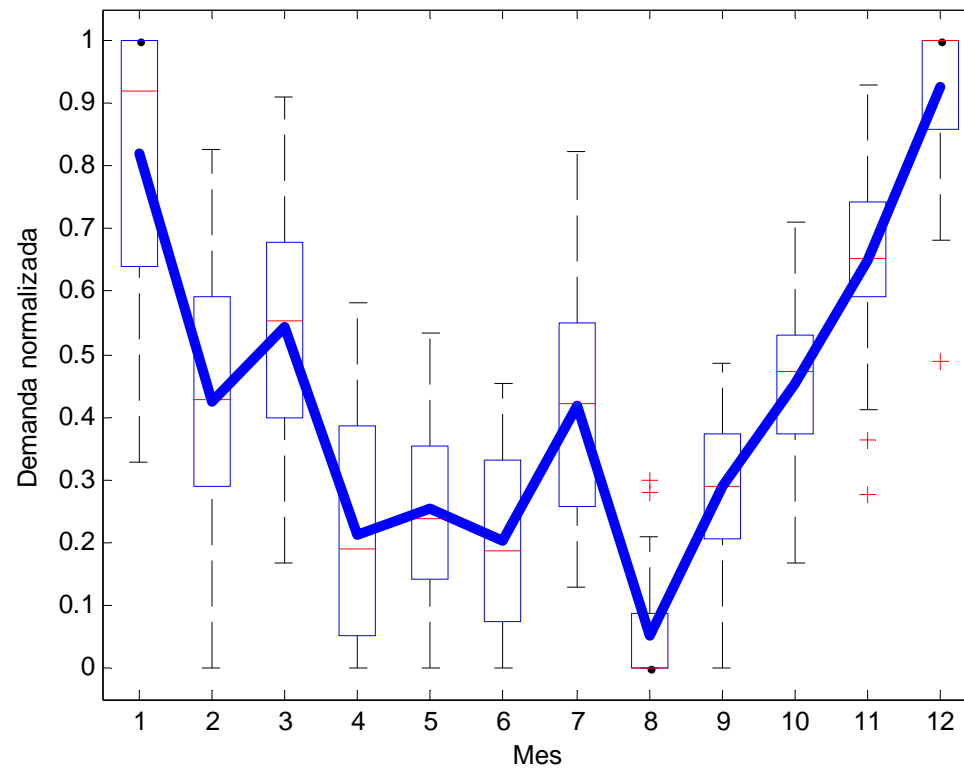
Ejercicio Final 3

- Representar en un mismo gráfico todos los consumos anuales normalizados de forma que todos estén entre 0 y 1



Ejercicio Final 4

- Representar un boxplot de los datos normalizados y superponer la media





CEU
*Universidad
San Pablo*

CURSO de UTILIZACIÓN PRÁCTICA de MATLAB

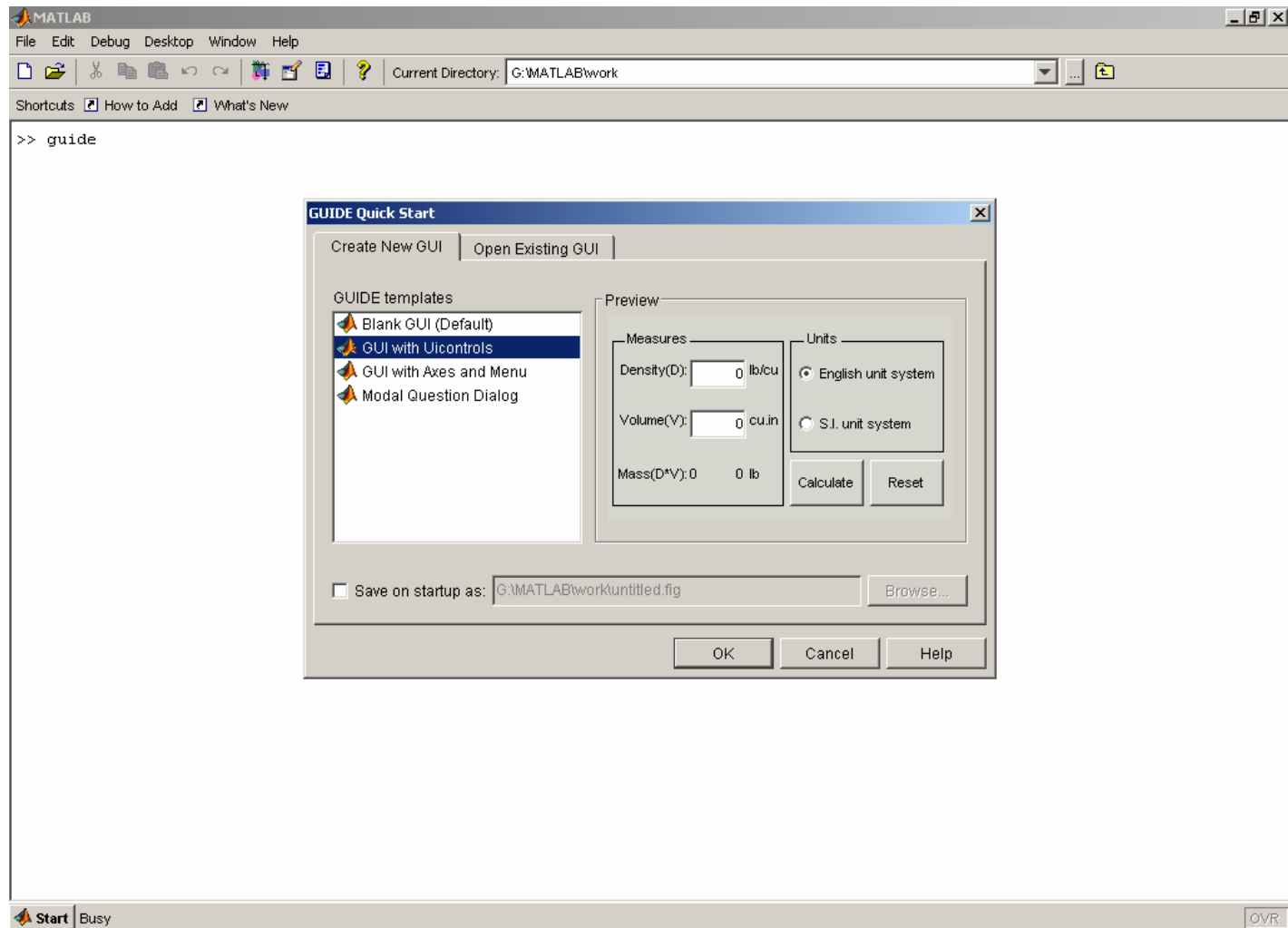
Sesión 4

Carlos Óscar Sánchez Sorzano, Ph.D.
Madrid, July 17th 2006

Cronograma del curso

- Día 1: Operaciones con matrices y vectores. Funciones de librería.
- Día 2: Otros tipos de datos en MATLAB. Programación en MATLAB.
- Día 3: Gráficos bidimensionales. Gráficos tridimensionales.
- **Día 4: Interfaces de usuario en MATLAB. Generación de programas autónomos**
- Día 5: Librerías de interés práctico
- Día 6: Interacción de MATLAB con Office y Visual Basic
- Día 7: Desarrollo de un proyecto

Graphical User Interfaces: GUIs



GUIs

Diseño del
interfaz

Botón de ejecución:
Salvar como
Mi_primer_GUI

Interfaz
ejecutándose

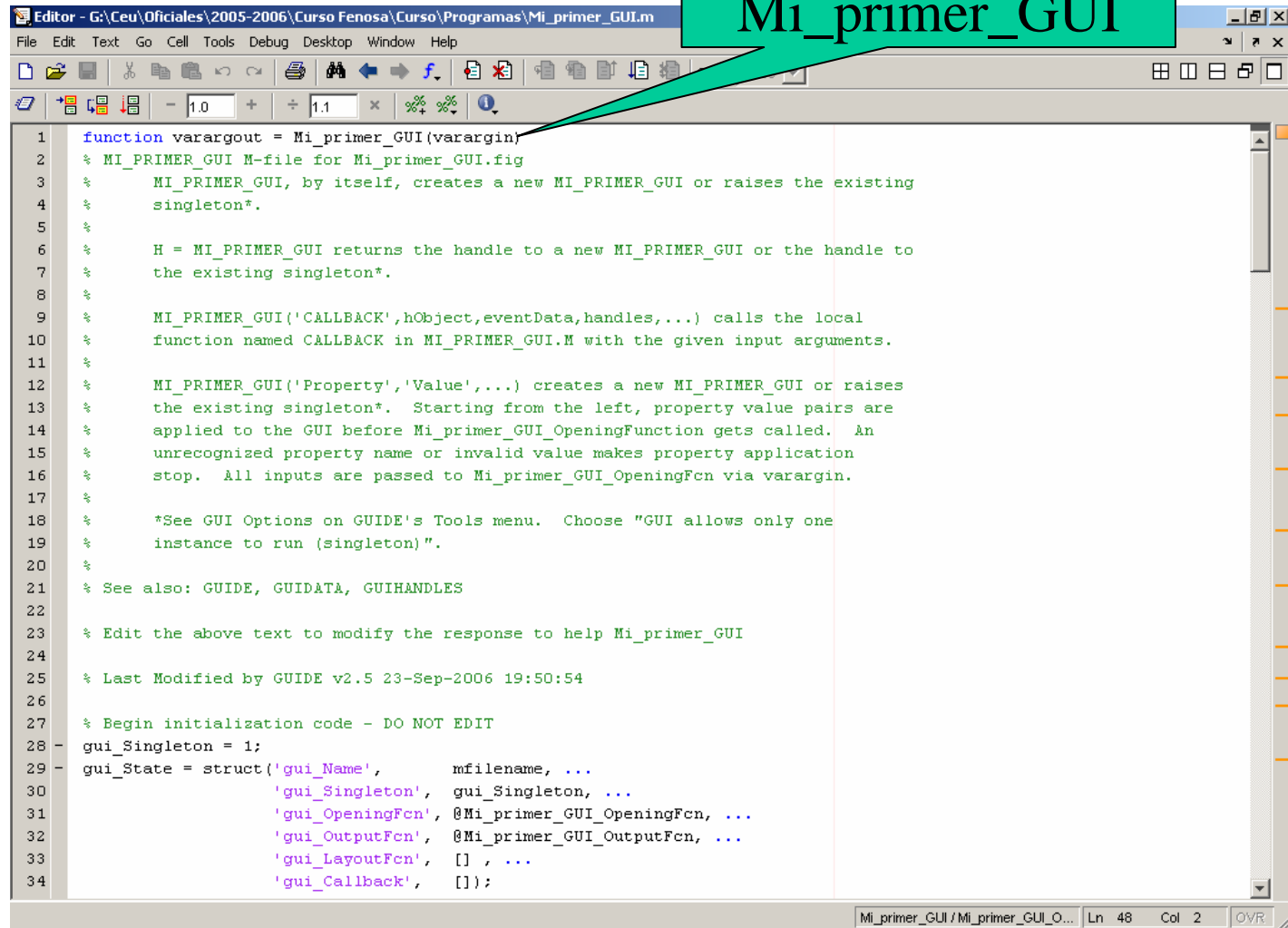
The screenshot displays the MATLAB Editor interface. The main window shows the design of the 'Mi_primer_GUI' interface, which includes input fields for Density (D), Volume (V), and Mass (D*V), along with unit selection buttons (English Unit System, S.I. Unit System) and Calculate/Reset buttons. A yellow callout points to the 'Save as' button, indicating the file should be saved as 'Mi_primer_GUI'. A green callout points to the running GUI window, which shows the same interface with numerical values entered. A blue callout points to the generated code in the editor, which includes initialization and callback functions. The code is as follows:

```
1 function varargout = Mi_primer_GUI(varargin)
2
3 % See also: GUIDE, GUIDATA, GUIHANDLES
4
5 % Edit the above text to modify the response to help Mi_primer_GUI
6
7 % Last Modified by GUIDE v2.5 23-Sep-2006 19:50:54
8
9 % Begin initialization code - DO NOT EDIT
10 gui_Singleton = 1;
11 gui_State = struct('gui_Name',       mfilename, ...
12                  'gui_Singleton',   gui_Singleton, ...
13                  'gui_OpeningFcn', @Mi_primer_GUI_OpeningFcn, ...
14                  'gui_OutputFcn',  @Mi_primer_GUI_OutputFcn, ...
15                  'gui_LayoutFcn',  [], ...
16                  'gui_Callback',    []);
17
18 % If nargin == 0, call the opening function to create the GUI
19 if nargin == 0
20     gui_State.gui_OpeningFcn(gui_State, []);
21 end
22
23 % If nargin > 0, call the output function to return the GUI
24 if nargin > 0
25     gui_State.gui_OutputFcn(gui_State, varargin);
26 end
27
28 % End of file
29
```

Código
generado

GUIs

Función Mi_primer_GUI



```
1 function varargout = Mi_primer_GUI(varargin)
2 % MI_PRIMER_GUI M-file for Mi_primer_GUI.fig
3 % MI_PRIMER_GUI, by itself, creates a new MI_PRIMER_GUI or raises the existing
4 % singleton*.
5 %
6 % H = MI_PRIMER_GUI returns the handle to a new MI_PRIMER_GUI or the handle to
7 % the existing singleton*.
8 %
9 % MI_PRIMER_GUI('CALLBACK',hObject,eventData,handles,...) calls the local
10 % function named CALLBACK in MI_PRIMER_GUI.M with the given input arguments.
11 %
12 % MI_PRIMER_GUI('Property','Value',...) creates a new MI_PRIMER_GUI or raises
13 % the existing singleton*. Starting from the left, property value pairs are
14 % applied to the GUI before Mi_primer_GUI_OpeningFunction gets called. An
15 % unrecognized property name or invalid value makes property application
16 % stop. All inputs are passed to Mi_primer_GUI_OpeningFcn via varargin.
17 %
18 % *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one
19 % instance to run (singleton)".
20 %
21 % See also: GUIDE, GUIDATA, GUIHANDLES
22
23 % Edit the above text to modify the response to help Mi_primer_GUI
24
25 % Last Modified by GUIDE v2.5 23-Sep-2006 19:50:54
26
27 % Begin initialization code - DO NOT EDIT
28 gui_Singleton = 1;
29 gui_State = struct('gui_Name',       mfilename, ...
30                   'gui_Singleton',   gui_Singleton, ...
31                   'gui_OpeningFcn',   @Mi_primer_GUI_OpeningFcn, ...
32                   'gui_OutputFcn',    @Mi_primer_GUI_OutputFcn, ...
33                   'gui_LayerFcn',     [], ...
34                   'gui_Callback',     []);
```

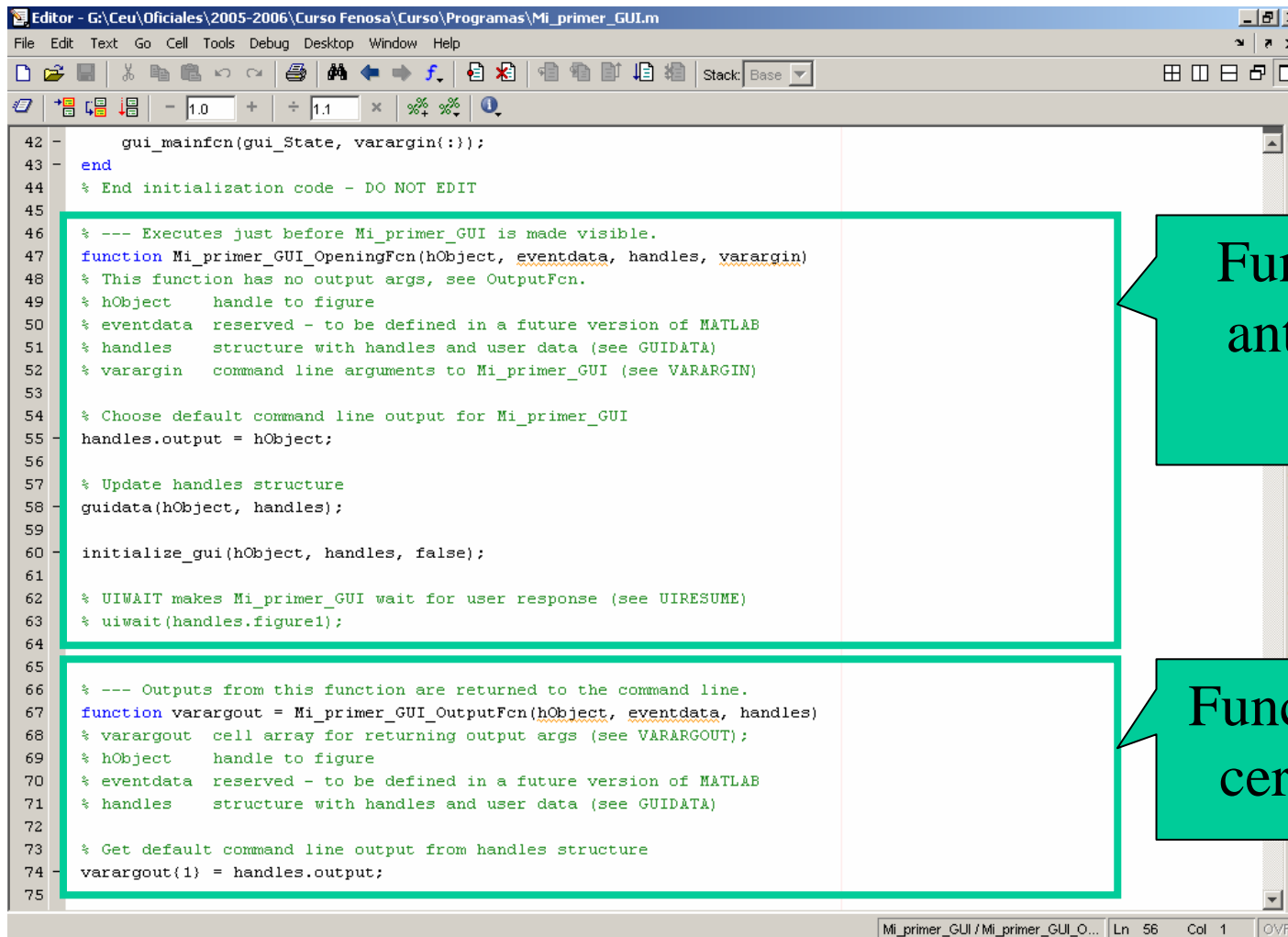
GUIs

```
Editor - G:\Ceu\Oficiais\2005-2006\Curso Fenosa\Curso_Fenosa\Mi_primer_GUI.m
File Edit Text Go Cell Tools Debug Desktop Window Help
[Icons] [Zoom: 1.0] [Divide: 1.1] [Stack: Base]
% MI_PRIMER_GUI('Property','Value',...) creates a new MI_PRIMER_GUI or raises
% the existing singleton*. Starting from the left, property value pairs are
% applied to the GUI before Mi_primer_GUI_OpeningFunction gets called. An
% unrecognized property name or invalid value makes property application
% stop. All inputs are passed to Mi_primer_GUI_OpeningFcn via varargin.
%
% *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one
% instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES
%
% Edit the above text to modify the response to help Mi_primer_GUI
%
% Last Modified by GUIDE v2.5 23-Sep-2006 19:50:54
%
% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn', @Mi_primer_GUI_OpeningFcn, ...
                  'gui_OutputFcn',  @Mi_primer_GUI_OutputFcn, ...
                  'gui_Layerfcn',    [] , ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT
```

Código de
inicialización.
Generado
automáticamente

GUIs

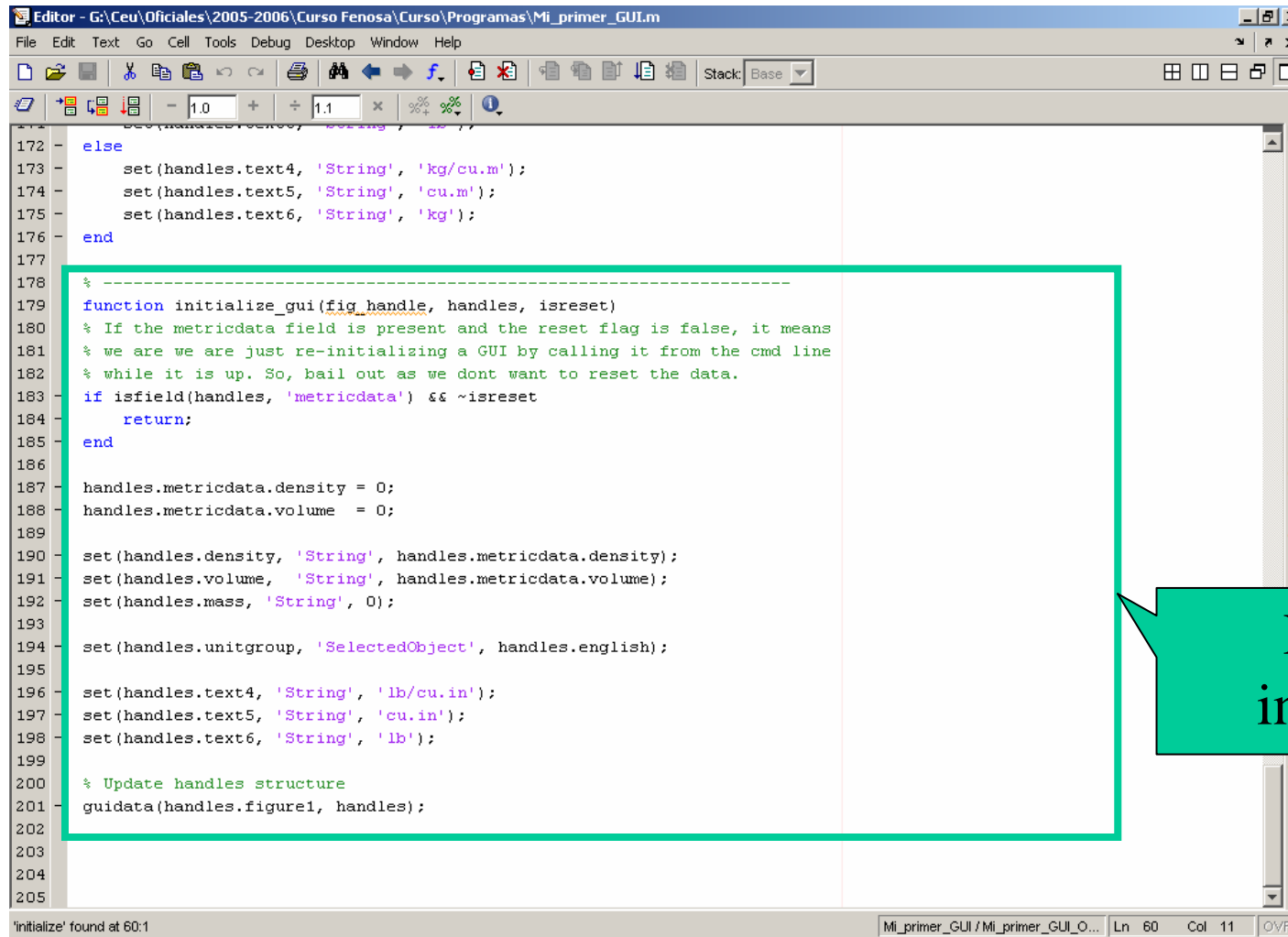


```
42 -     gui_mainfcn(gui_State, varargin{:});
43 - end
44 - % End initialization code - DO NOT EDIT
45 -
46 - % --- Executes just before Mi_primer_GUI is made visible.
47 - function Mi_primer_GUI_OpeningFcn(hObject, eventdata, handles, varargin)
48 - % This function has no output args, see OutputFcn.
49 - % hObject    handle to figure
50 - % eventdata  reserved - to be defined in a future version of MATLAB
51 - % handles    structure with handles and user data (see GUIDATA)
52 - % varargin   command line arguments to Mi_primer_GUI (see VARARGIN)
53 -
54 - % Choose default command line output for Mi_primer_GUI
55 - handles.output = hObject;
56 -
57 - % Update handles structure
58 - guidata(hObject, handles);
59 -
60 - initialize_gui(hObject, handles, false);
61 -
62 - % UIWAIT makes Mi_primer_GUI wait for user response (see UIRESUME)
63 - % uiwait(handles.figure1);
64 -
65 - % --- Outputs from this function are returned to the command line.
66 - function varargout = Mi_primer_GUI_OutputFcn(hObject, eventdata, handles)
67 - % varargout  cell array for returning output args (see VARARGOUT);
68 - % hObject    handle to figure
69 - % eventdata  reserved - to be defined in a future version of MATLAB
70 - % handles    structure with handles and user data (see GUIDATA)
71 -
72 - % Get default command line output from handles structure
73 - varargout{1} = handles.output;
74 -
75 -
```

Función llamada antes de abrir el interfaz

Función llamada al cerrar el interfaz

GUIs



The image shows a MATLAB Editor window with a file named 'Mi_primer_GUI.m'. The code is a function that initializes a GUI. A green box highlights the function definition and its body. A speech bubble points to this box with the text 'Función de inicialización'.

```
172 else
173     set(handles.text4, 'String', 'kg/cu.m');
174     set(handles.text5, 'String', 'cu.m');
175     set(handles.text6, 'String', 'kg');
176 end
177
178 % -----
179 function initialize_gui(fig_handle, handles, isreset)
180 % If the metricdata field is present and the reset flag is false, it means
181 % we are just re-initializing a GUI by calling it from the cmd line
182 % while it is up. So, bail out as we dont want to reset the data.
183 if isfield(handles, 'metricdata') && ~isreset
184     return;
185 end
186
187 handles.metricdata.density = 0;
188 handles.metricdata.volume = 0;
189
190 set(handles.density, 'String', handles.metricdata.density);
191 set(handles.volume, 'String', handles.metricdata.volume);
192 set(handles.mass, 'String', 0);
193
194 set(handles.unitgroup, 'SelectedObject', handles.english);
195
196 set(handles.text4, 'String', 'lb/cu.in');
197 set(handles.text5, 'String', 'cu.in');
198 set(handles.text6, 'String', 'lb');
199
200 % Update handles structure
201 guidata(handles.figure1, handles);
202
203
204
205
```

Speech bubble text: Función de inicialización

GUIs

```
Editor - G:\Ceu\Oficiales\2005-2006\Curso Fenosa\Curso\Programas\Mi_primer_GUI.m
File Edit Text Go Cell Tools Debug Desktop Window Help
[Icons] [1.0] [1.1] [Icons] Stack Base
76
77 % --- Executes during object creation, after setting all properties.
78 function density_CreateFcn(hObject, eventdata, handles)
79 % hObject    handle to density (see GCBO)
80 % eventdata  reserved - to be defined in a future version of MATLAB
81 % handles    empty - handles not created until after all CreateFcns called
82
83 % Hint: edit controls usually have a white background, change
84 %       'usewhitebg' to 0 to use default.  See ISPC and COMPUTER.
85 usewhitebg = 1;
86 if usewhitebg
87     set(hObject,'BackgroundColor','white');
88 else
89     set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
90 end
91
92
93
94 function density_Callback(hObject, eventdata, handles)
95 % hObject    handle to density (see GCBO)
96 % eventdata  reserved - to be defined in a future version of MATLAB
97 % handles    structure with handles and user data (see GUIDATA)
98
99 % Hints: get(hObject,'String') returns contents of density as text
100 %       str2double(get(hObject,'String')) returns contents of density as a double
101 density = str2double(get(hObject, 'String'));
102 if isnan(density)
103     set(hObject, 'String', 0);
104     errordlg('Input must be a number','Error');
105 end
106
107 % Save the new density value
108 handles.metricdata.density = density;
109 guidata(hObject,handles)
```

Función llamada al
crear el objeto
density

Función llamada al
modificar el objeto
density

GUIs

```
Editor - G:\Ceu\Oficiales\2005-2006\Curso Fenosa\Curso\Programas\Mi_primer_GUI.m
File Edit Text Go Cell Tools Debug Desktop Window Help
[Icons] [Stack: Base]
[Icons] [1.0] [1.1] [Icons] [Icons]

111 % --- Executes during object creation, after setting all properties.
112 function volume_CreateFcn(hObject, eventdata, handles)
113 % hObject    handle to volume (see GCBO)
114 % eventdata  reserved - to be defined in a future version of MATLAB
115 % handles    empty - handles not created until after all CreateFcns called
116
117 % Hint: edit controls usually have a white background, change
118 %       'usewhitebg' to 0 to use default.  See ISPC and COMPUTER.
119 usewhitebg = 1;
120 if usewhitebg
121     set(hObject,'BackgroundColor','white');
122 else
123     set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
124 end
125
126
127
128 function volume_Callback(hObject, eventdata, handles)
129 % hObject    handle to volume (see GCBO)
130 % eventdata  reserved - to be defined in a future version of MATLAB
131 % handles    structure with handles and user data (see GUIDATA)
132
133 % Hints: get(hObject,'String') returns contents of volume as text
134 %       str2double(get(hObject,'String')) returns contents of volume as a double
135 volume = str2double(get(hObject, 'String'));
136 if isnan(volume)
137     set(hObject, 'String', 0);
138     errordlg('Input must be a number','Error');
139 end
140
141 % Save the new volume value
142 handles.metricdata.volume = volume;
143 guidata(hObject,handles)
144
```

Función llamada al crear el objeto volume

Función llamada al modificar el objeto volume

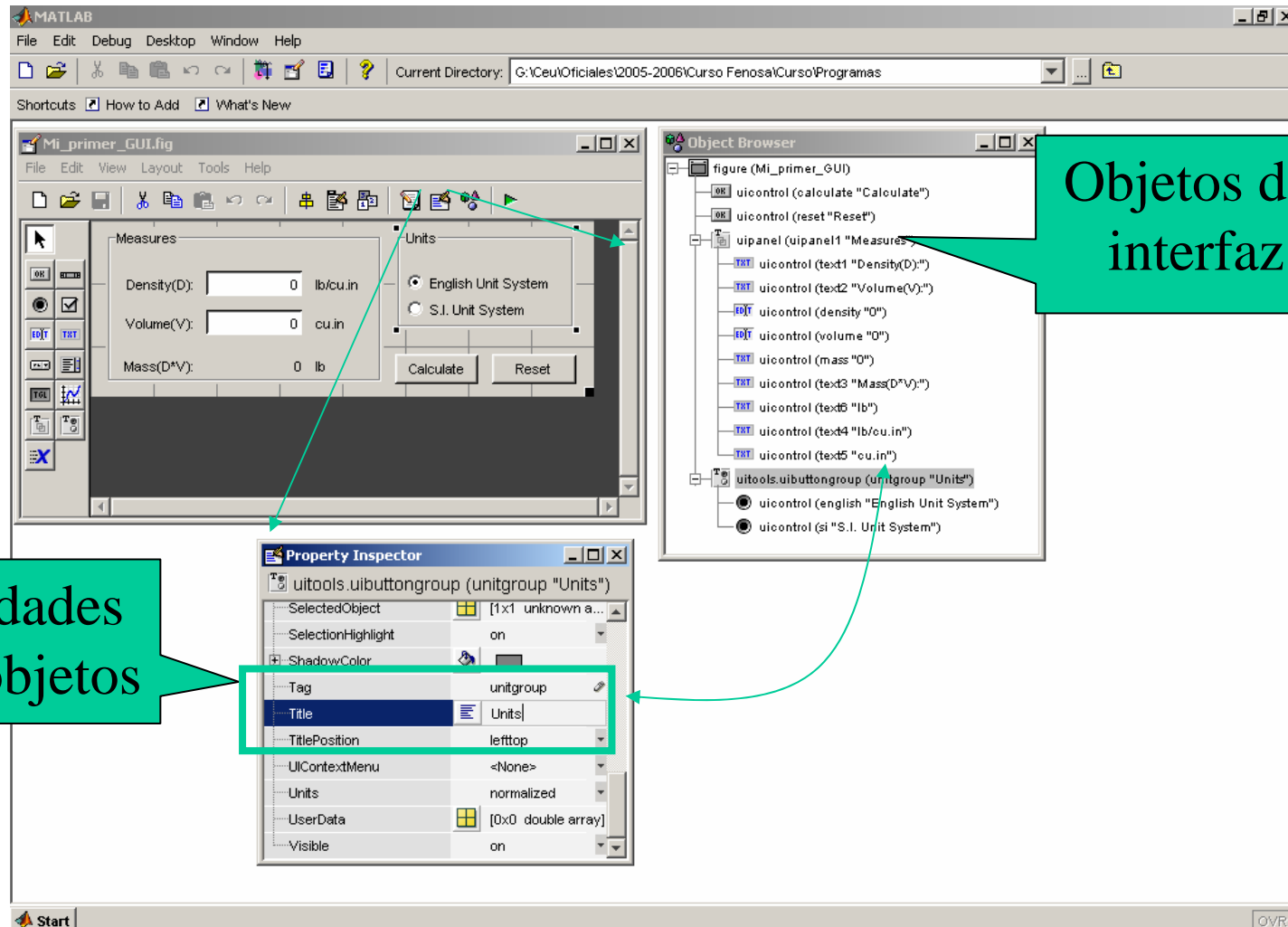
GUIs

```
Editor - G:\Ceu\Oficiales\2005-2006\Curso Fenosa\Curso\Programas\Mi_primer_GUI.m
File Edit Text Go Cell Tools Debug Desktop Window Help
[Icons] [Stack: Base]
[Icons] [1.0] [1.1] [Icons] [Icons]
144
145 % --- Executes on button press in calculate.
146 function calculate_Callback(hObject, eventdata, handles)
147 % hObject    handle to calculate (see GCBO)
148 % eventdata  reserved - to be defined in a future version of MATLAB
149 % handles    structure with handles and user data (see GUIDATA)
150
151 mass = handles.metricdata.density * handles.metricdata.volume;
152 set(handles.mass, 'String', mass);
153
154 % --- Executes on button press in reset.
155 function reset_Callback(hObject, eventdata, handles)
156 % hObject    handle to reset (see GCBO)
157 % eventdata  reserved - to be defined in a future version of MATLAB
158 % handles    structure with handles and user data (see GUIDATA)
159
160 initialize_gui(gcbf, handles, true);
161
162 % -----
163 function unitgroup_SelectionChangeFcn(hObject, eventdata, handles)
164 % hObject    handle to unitgroup (see GCBO)
165 % eventdata  reserved - to be defined in a future version of MATLAB
166 % handles    structure with handles and user data (see GUIDATA)
167
168 if (hObject == handles.english)
169     set(handles.text4, 'String', 'lb/cu.in');
170     set(handles.text5, 'String', 'cu.in');
171     set(handles.text6, 'String', 'lb');
172 else
173     set(handles.text4, 'String', 'kg/cu.m');
174     set(handles.text5, 'String', 'cu.m');
175     set(handles.text6, 'String', 'kg');
176 end
177
178 % -----
```

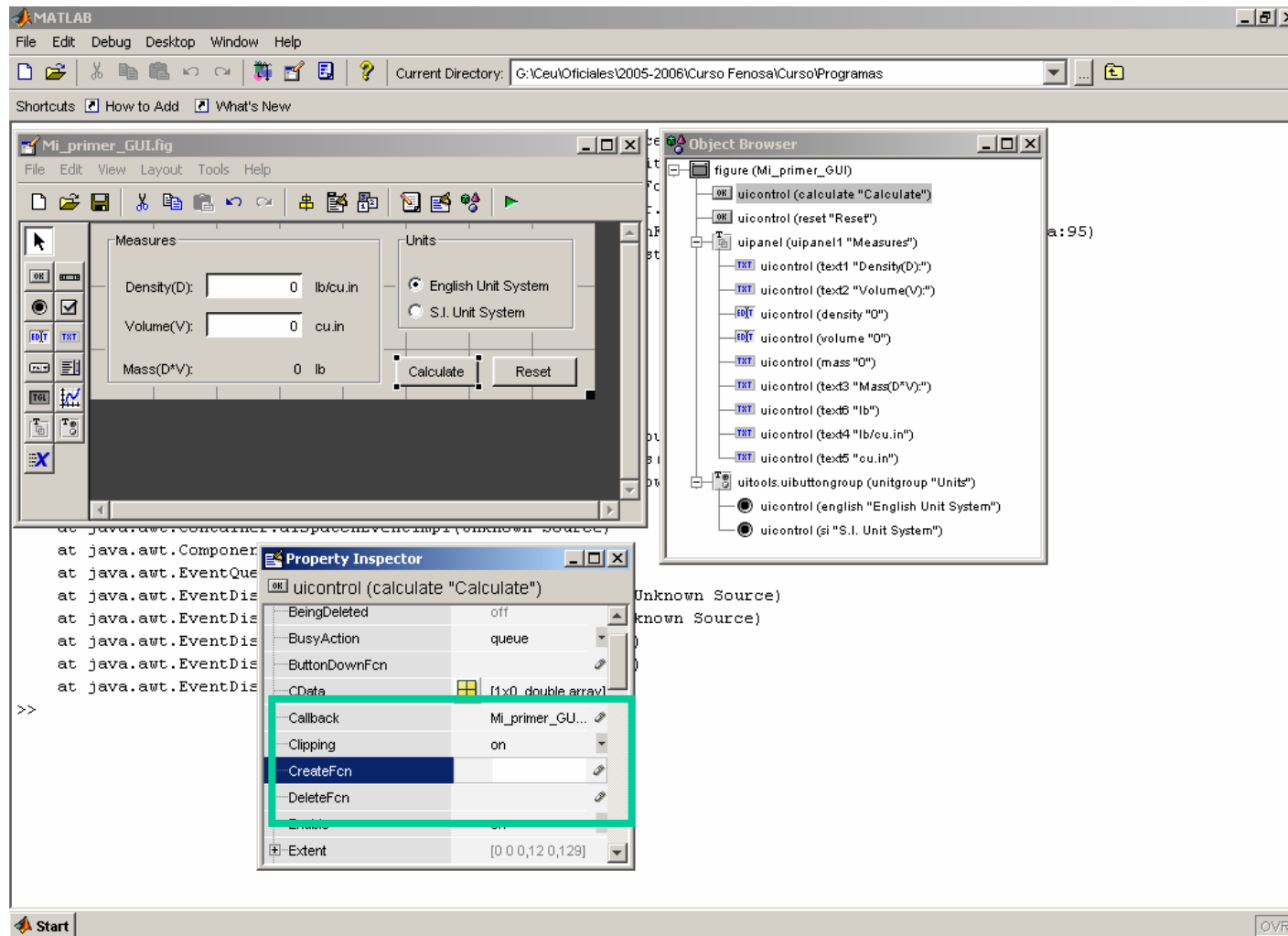
Funciones llamadas
al pulsar “Calculate”
o “Reset”

Función llamada
cuando cambian los
botones de unidades

GUIs

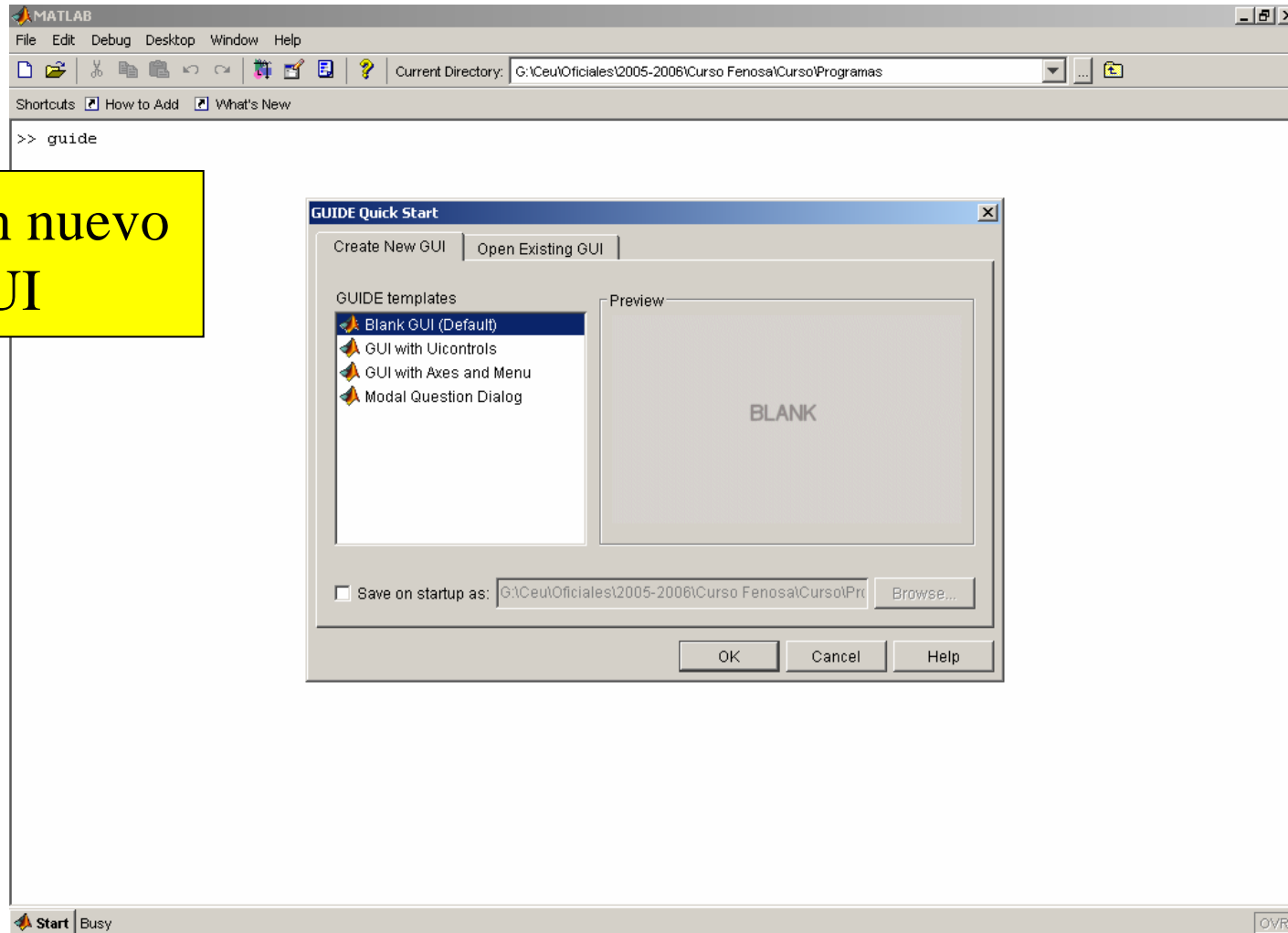


GUIs



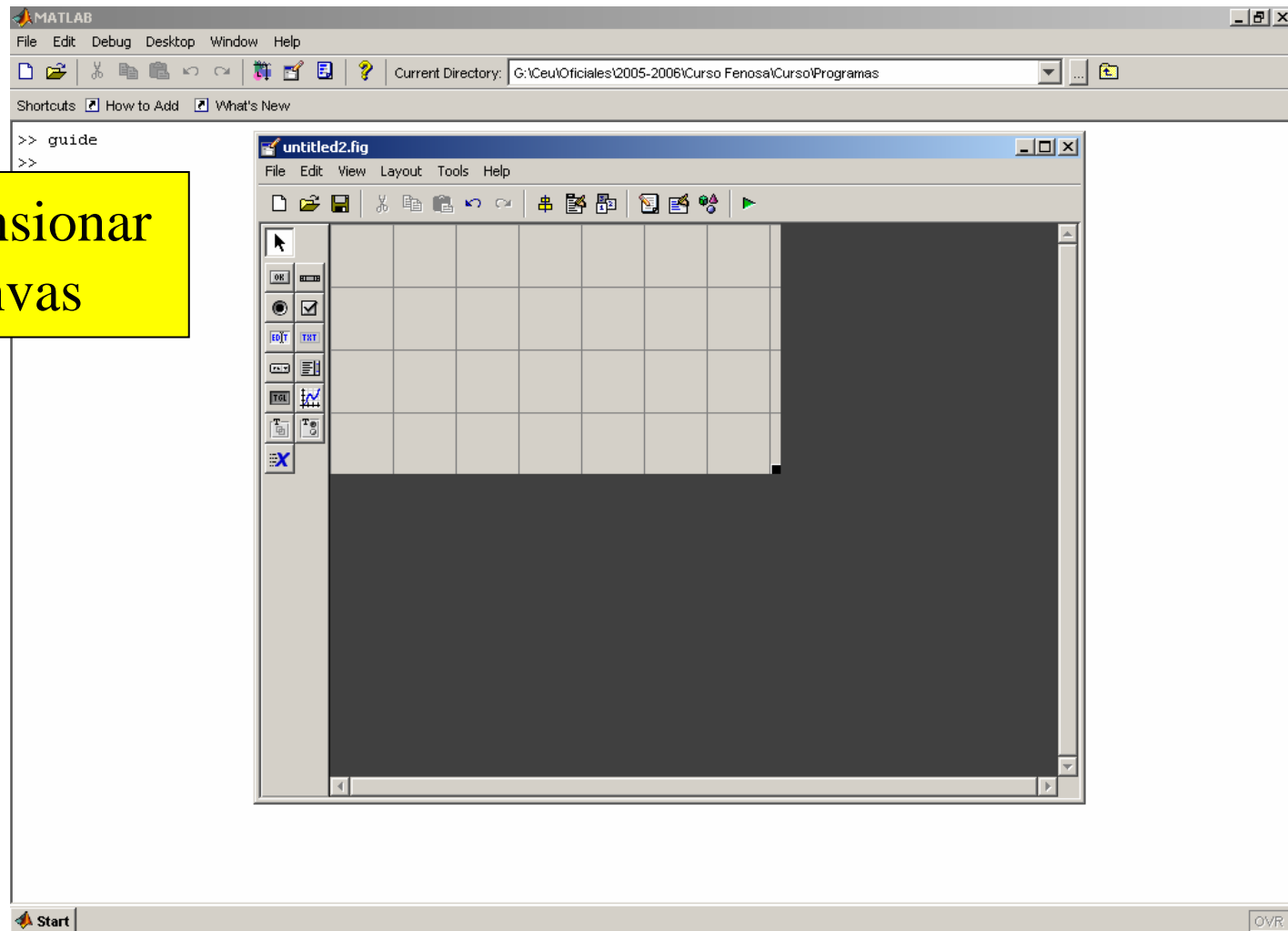
GUIs: Creación

Crear un nuevo
GUI



GUIs: Creación

Redimensionar
el canvas



GUIs: Creación

Salvar

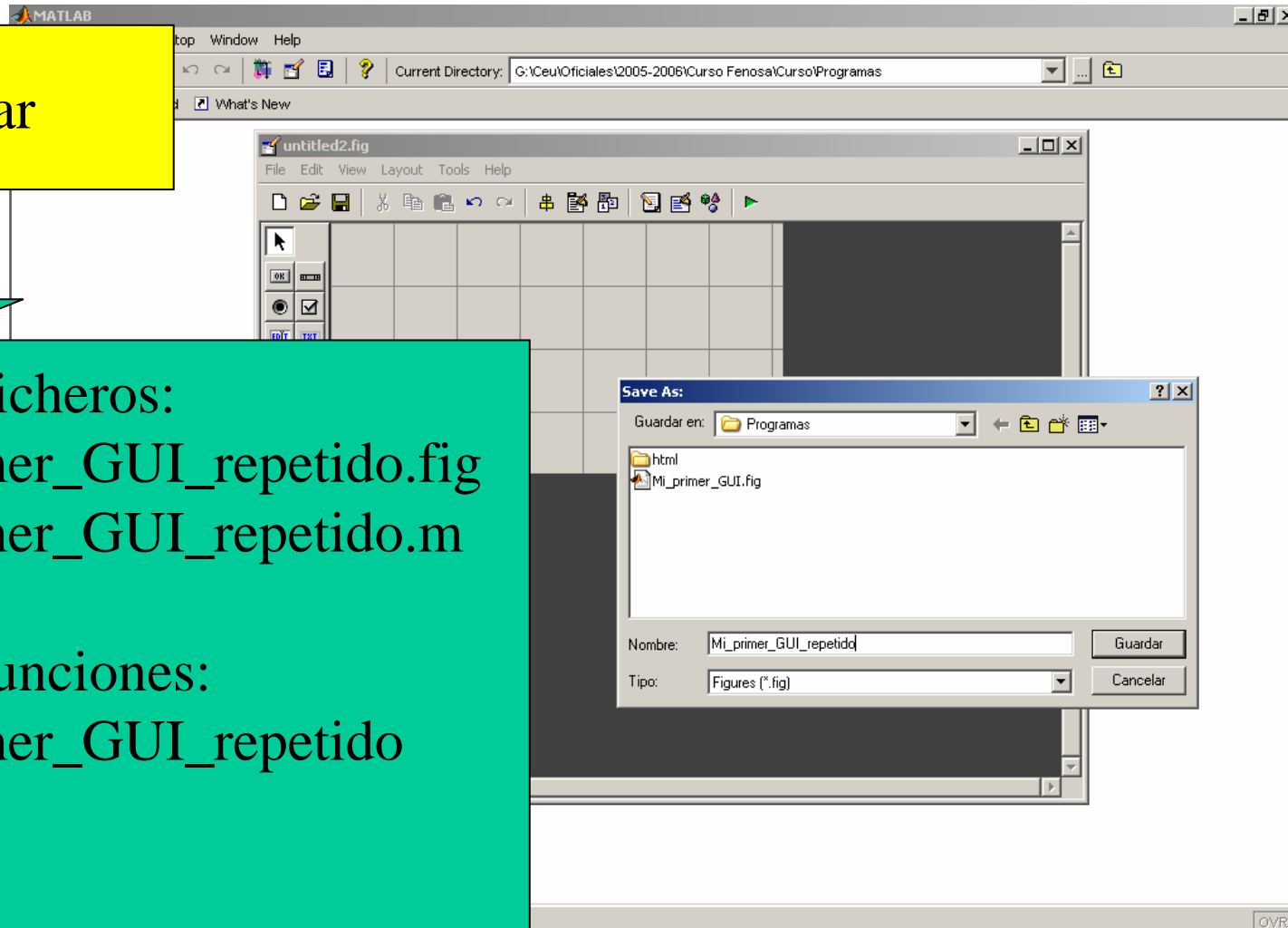


Crea los ficheros:

- Mi_primer_GUI_repetido.fig
- Mi_primer_GUI_repetido.m

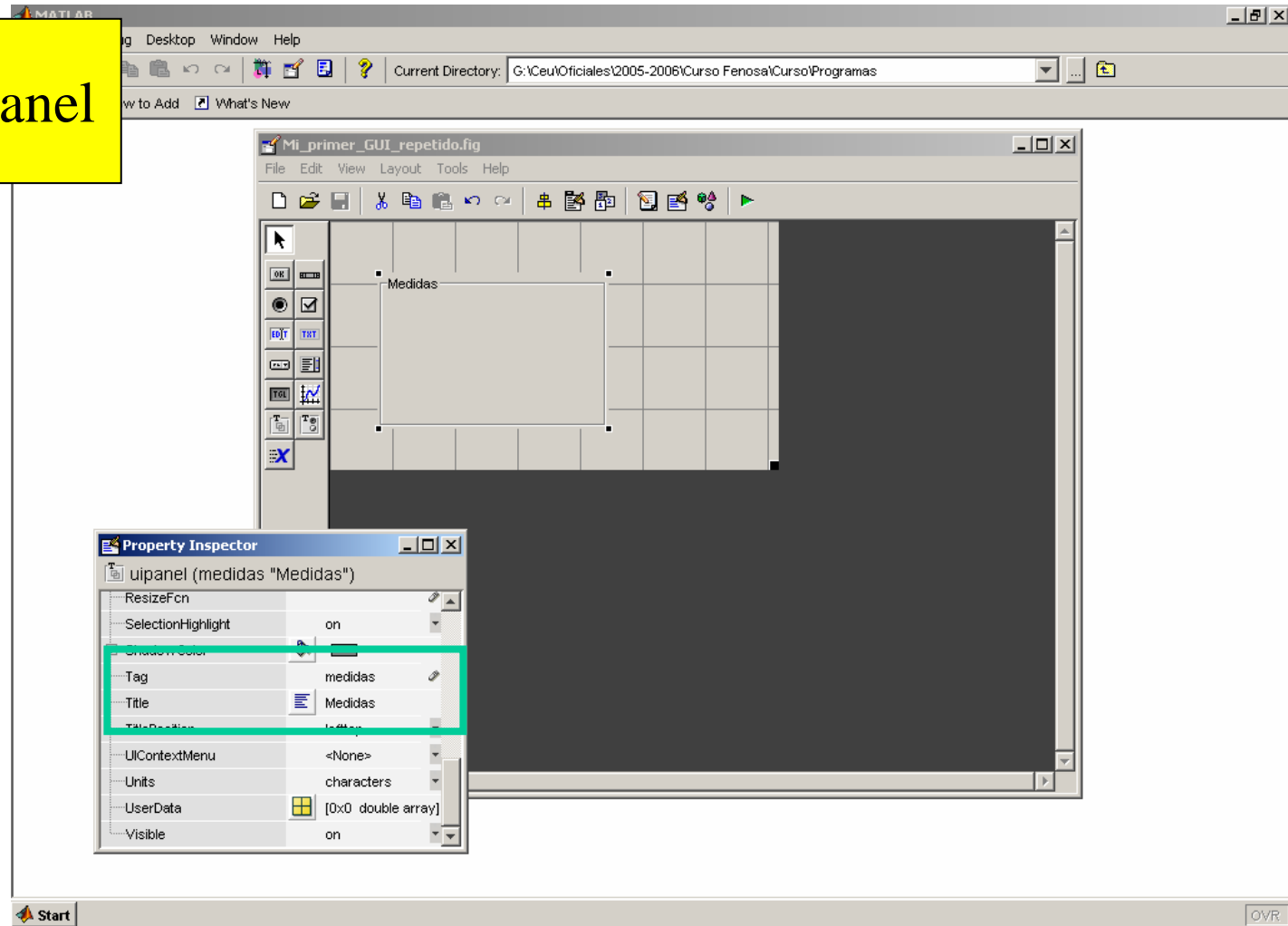
Crea las funciones:

- Mi_primer_GUI_repetido
- Opening
- Closing



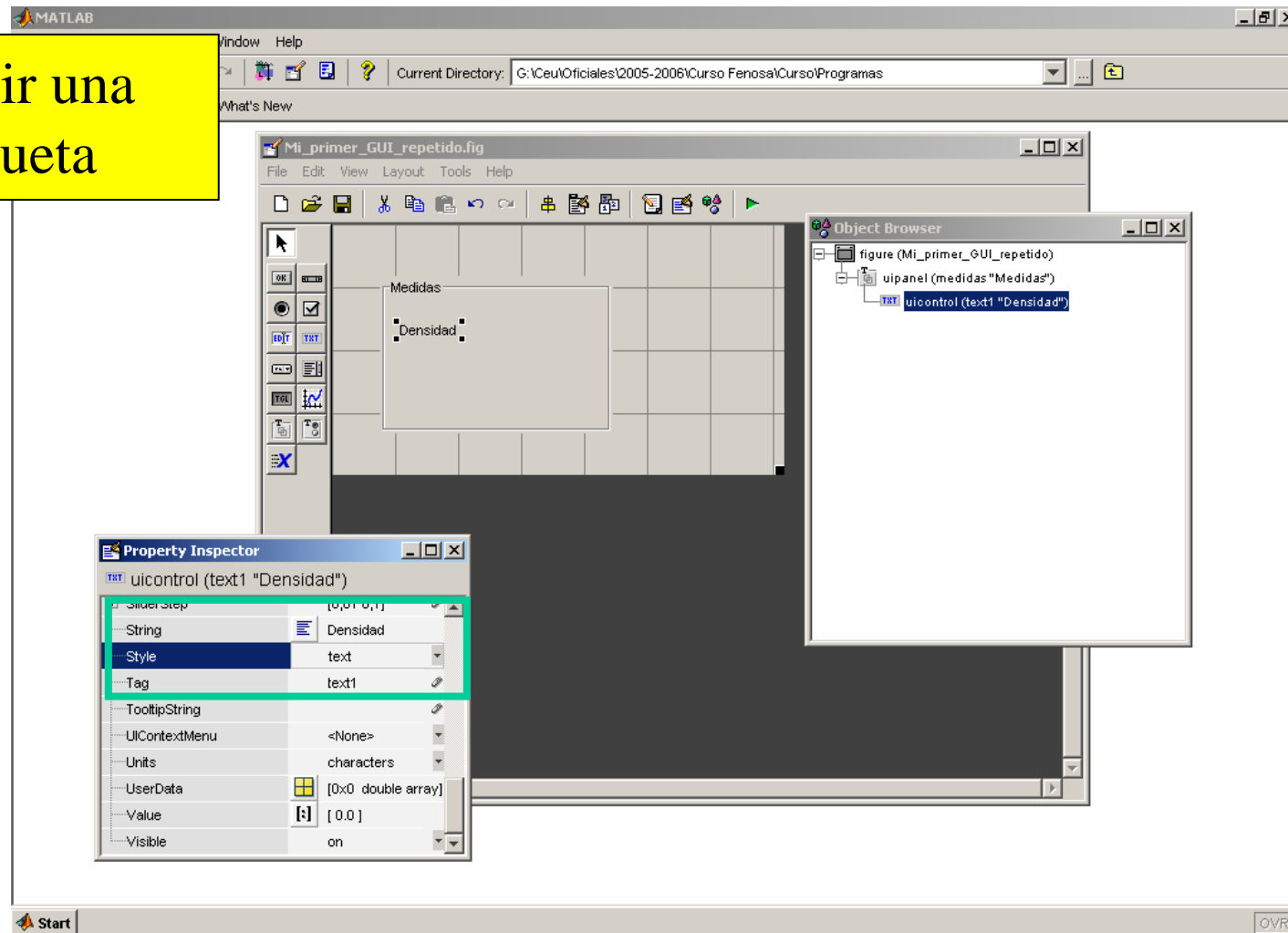
GUIs: Creación

Añadir un panel



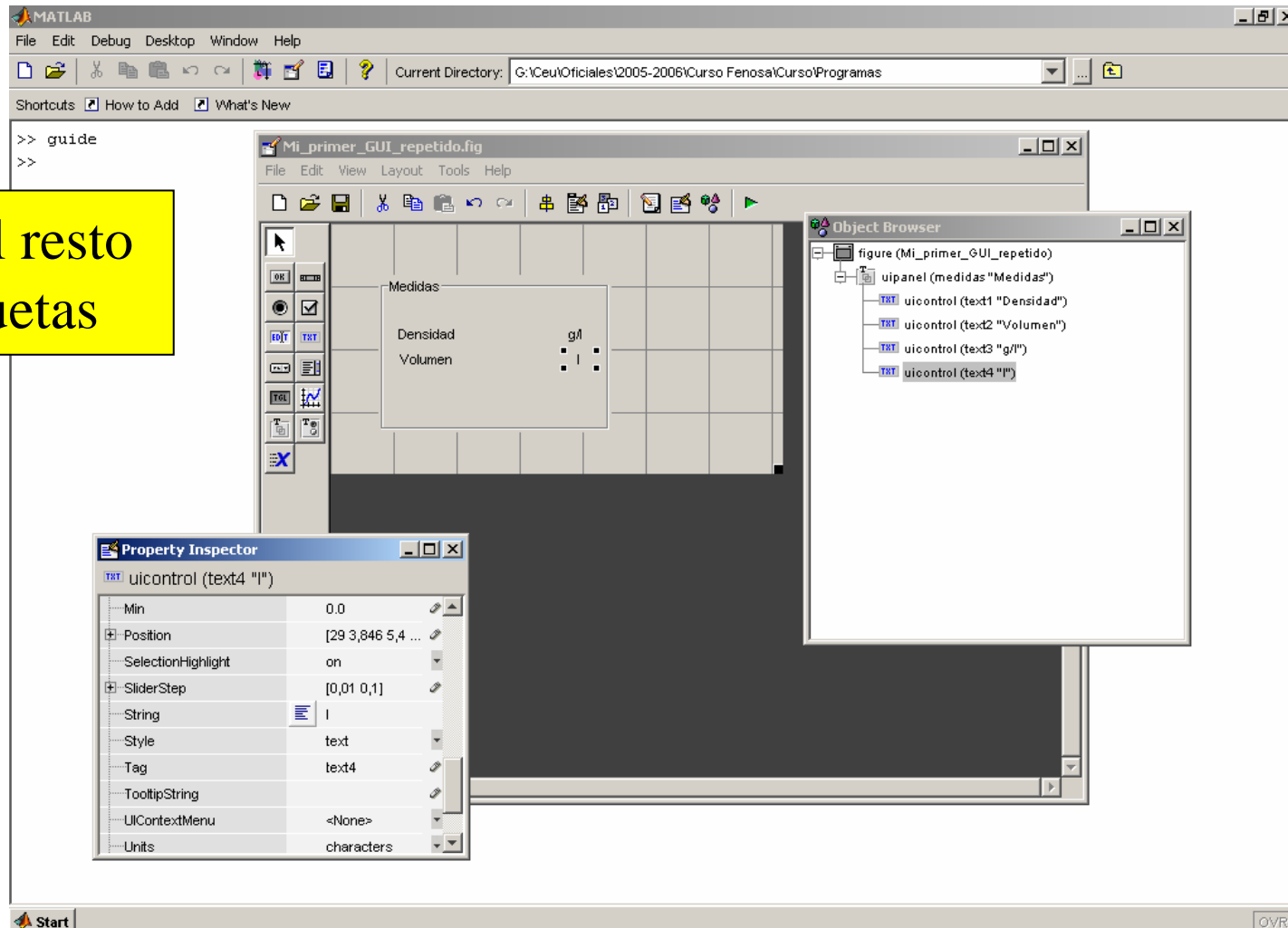
GUIs: Creación

Añadir una
etiqueta



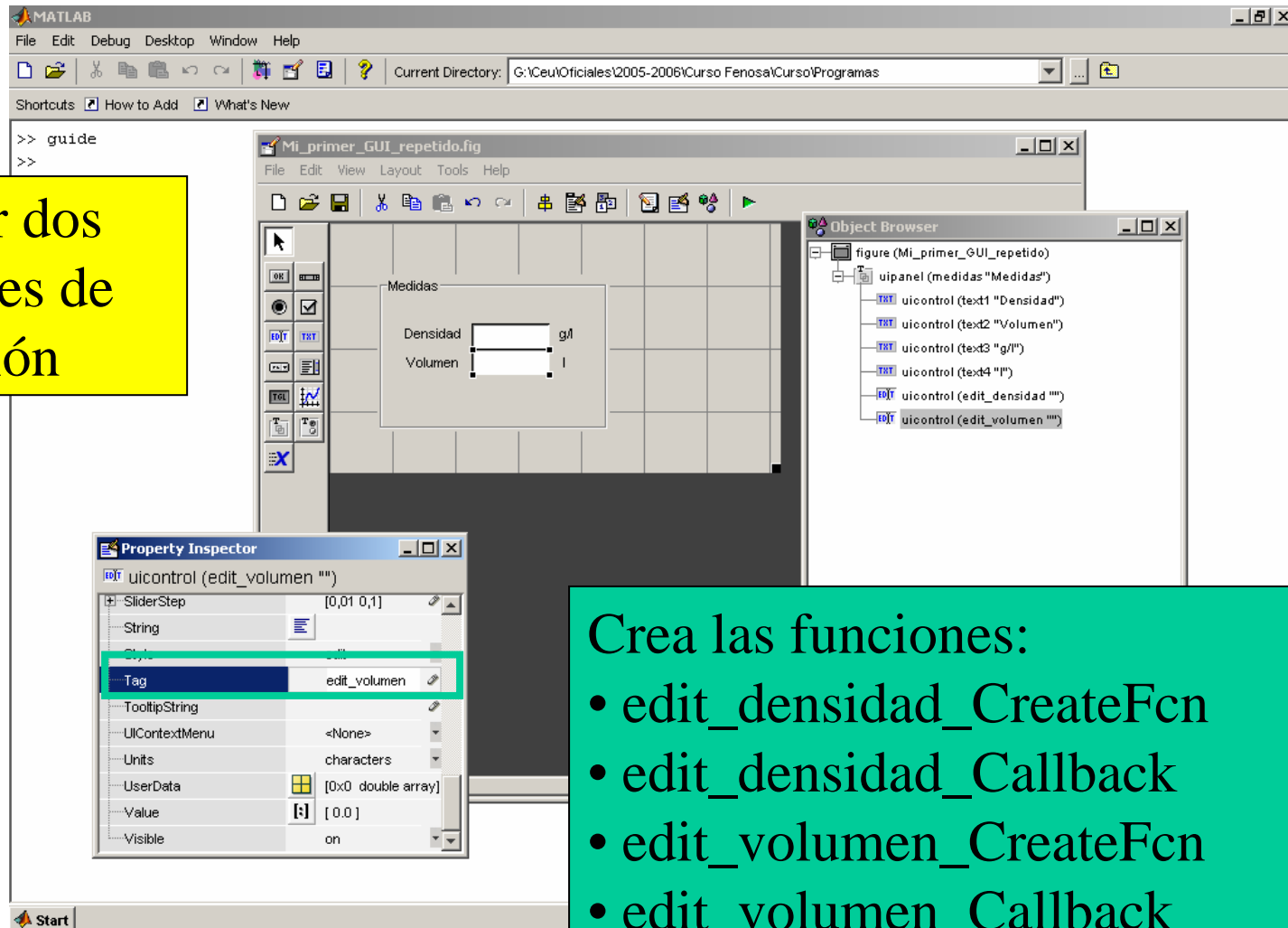
GUIs: Creación

Añadir el resto
de etiquetas



GUIs: Creación

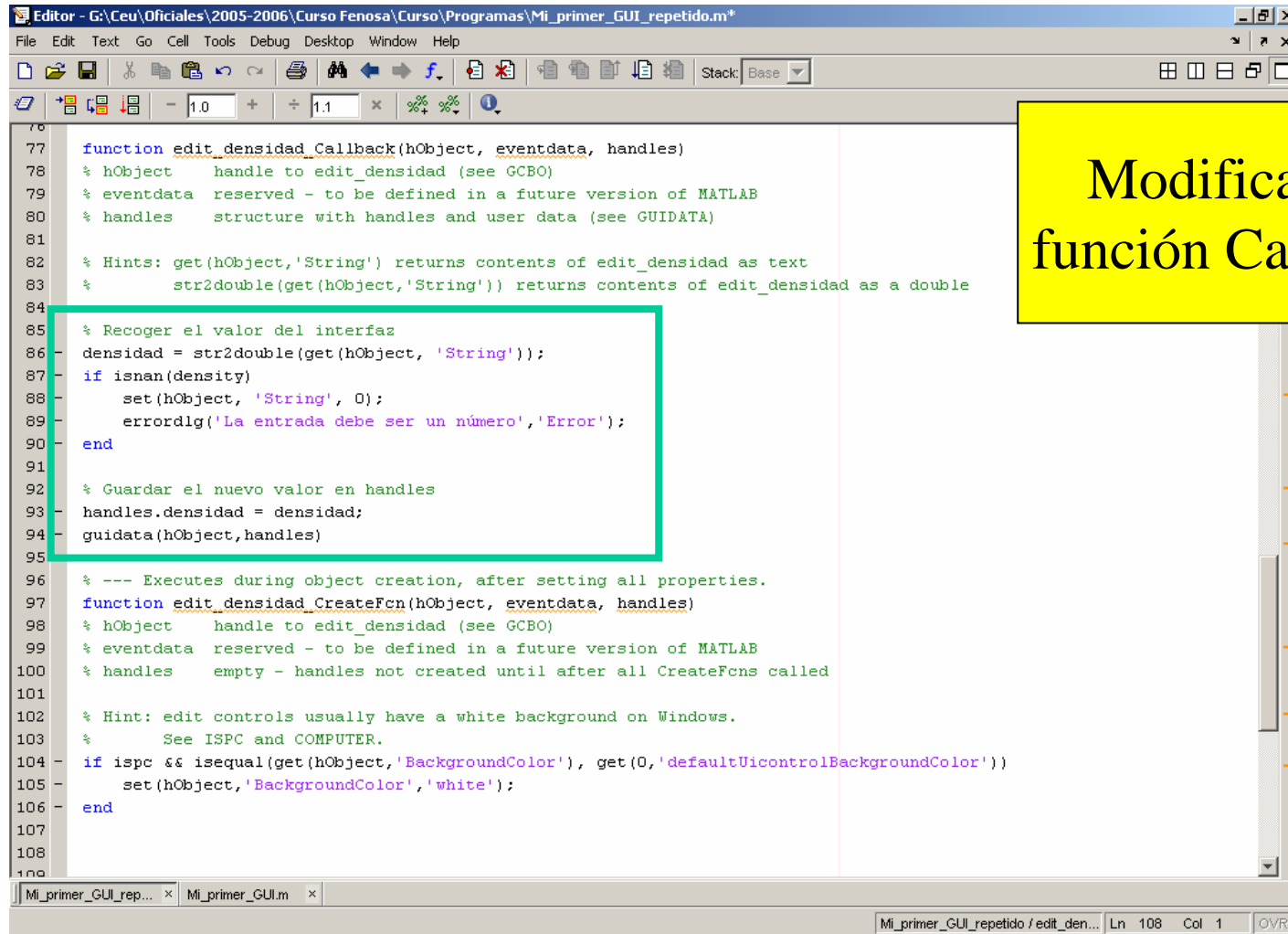
Añadir dos
variables de
edición



Crea las funciones:

- edit_densidad_CreateFcn
- edit_densidad_Callback
- edit_volumen_CreateFcn
- edit_volumen_Callback

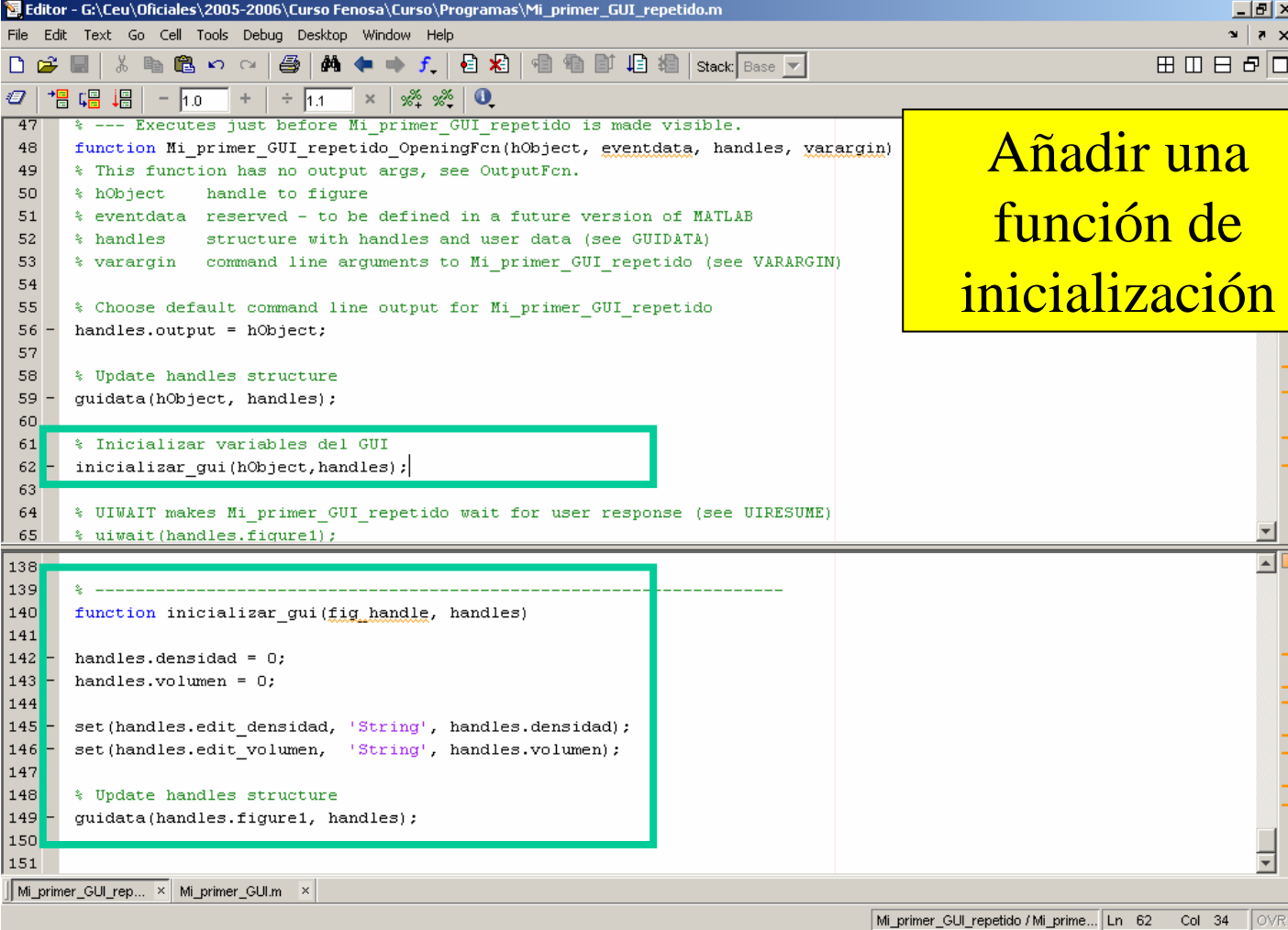
GUIs: Creación



```
76
77 function edit_densidad_Callback(hObject, eventdata, handles)
78 % hObject    handle to edit_densidad (see GCBO)
79 % eventdata  reserved - to be defined in a future version of MATLAB
80 % handles    structure with handles and user data (see GUIDATA)
81
82 % Hints: get(hObject,'String') returns contents of edit_densidad as text
83 %        str2double(get(hObject,'String')) returns contents of edit_densidad as a double
84
85 % Recoger el valor del interfaz
86 densidad = str2double(get(hObject, 'String'));
87 if isnan(densidad)
88     set(hObject, 'String', 0);
89     errordlg('La entrada debe ser un número','Error');
90 end
91
92 % Guardar el nuevo valor en handles
93 handles.densidad = densidad;
94 guidata(hObject,handles)
95
96 % --- Executes during object creation, after setting all properties.
97 function edit_densidad_CreateFcn(hObject, eventdata, handles)
98 % hObject    handle to edit_densidad (see GCBO)
99 % eventdata  reserved - to be defined in a future version of MATLAB
100 % handles    empty - handles not created until after all CreateFcns called
101
102 % Hint: edit controls usually have a white background on Windows.
103 %        See ISPC and COMPUTER.
104 if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
105     set(hObject,'BackgroundColor','white');
106 end
107
108
109
```

Modificar la
función Callback

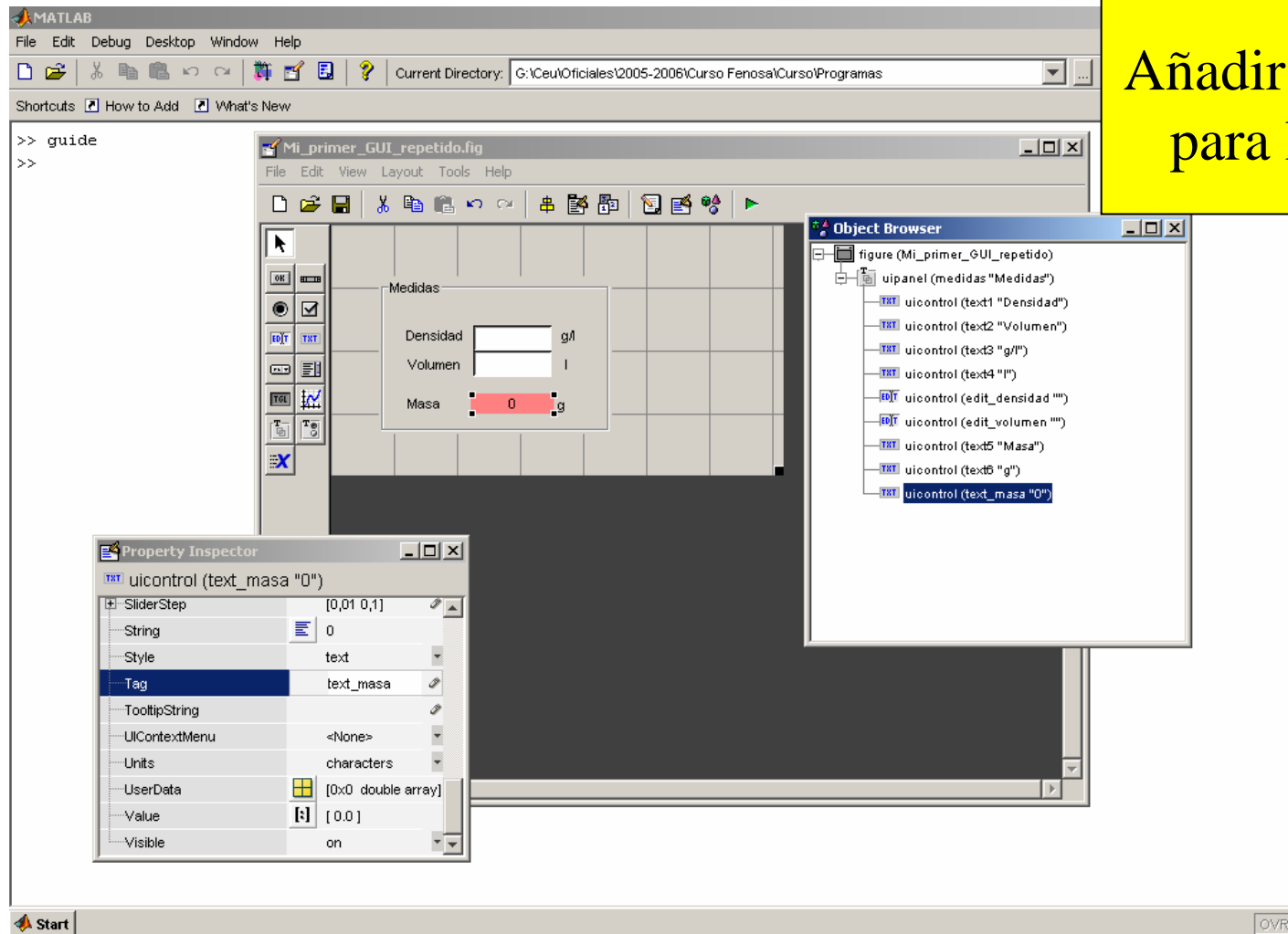
GUIs: Creación



```
Editor - G:\Ceu\Oficiales\2005-2006\Curso Fenosa\Curso\Programas\Mi_primer_GUI_repetido.m
File Edit Text Go Cell Tools Debug Desktop Window Help
[Icons] Stack: Base
47 % --- Executes just before Mi_primer_GUI_repetido is made visible.
48 function Mi_primer_GUI_repetido_OpeningFcn(hObject, eventdata, handles, varargin)
49 % This function has no output args, see OutputFcn.
50 % hObject    handle to figure
51 % eventdata  reserved - to be defined in a future version of MATLAB
52 % handles    structure with handles and user data (see GUIDATA)
53 % varargin   command line arguments to Mi_primer_GUI_repetido (see VARARGIN)
54
55 % Choose default command line output for Mi_primer_GUI_repetido
56 handles.output = hObject;
57
58 % Update handles structure
59 guidata(hObject, handles);
60
61 % Inicializar variables del GUI
62 inicializar_gui(hObject,handles);
63
64 % UIWAIT makes Mi_primer_GUI_repetido wait for user response (see UIRESUME)
65 % uiwait(handles.figure1);
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
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139 % -----
140 function inicializar_gui(fig_handle, handles)
141
142 handles.densidad = 0;
143 handles.volumen = 0;
144
145 set(handles.edit_densidad, 'String', handles.densidad);
146 set(handles.edit_volumen, 'String', handles.volumen);
147
148 % Update handles structure
149 guidata(handles.figure1, handles);
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2630

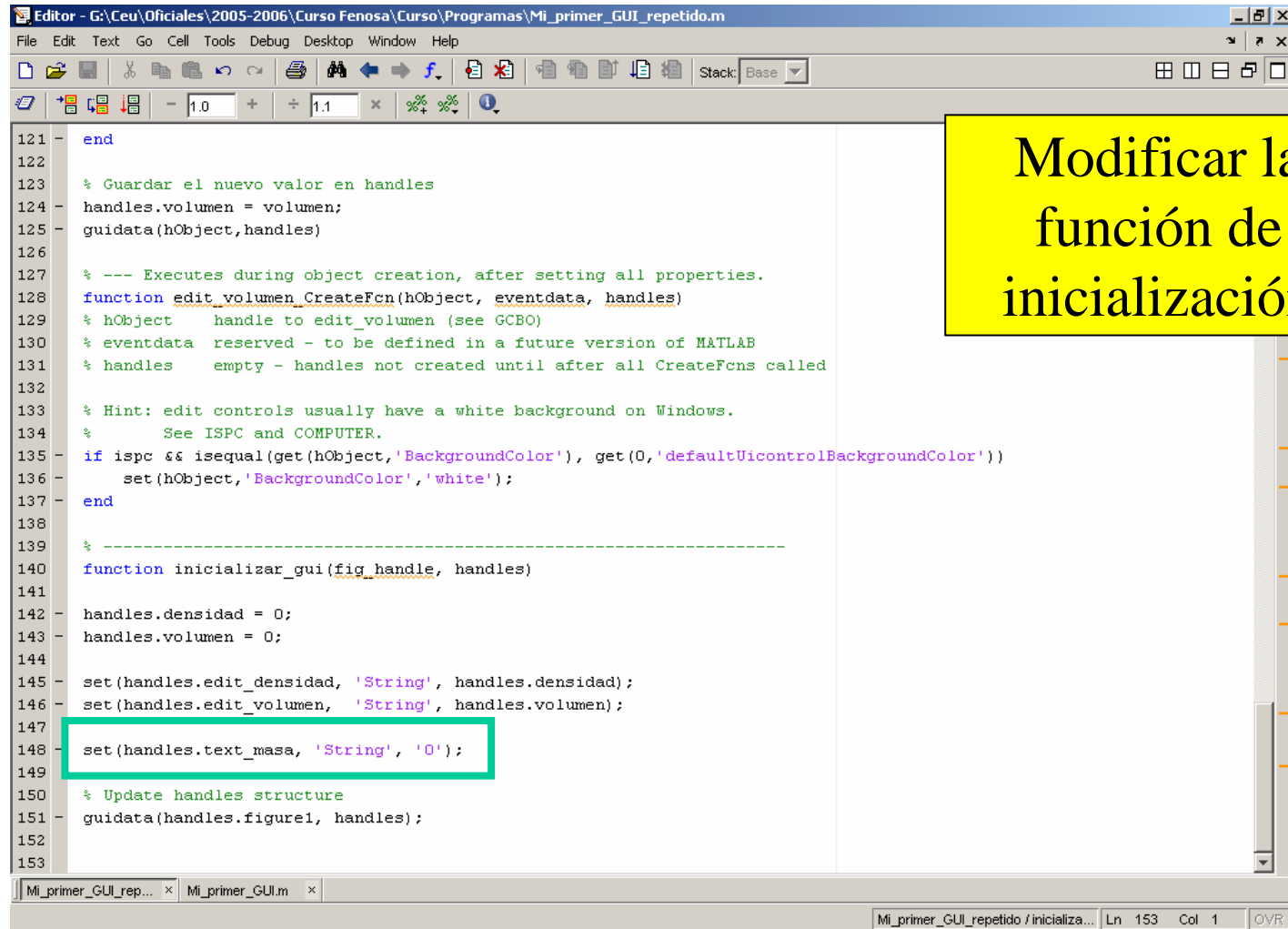
```

GUIs: Creación



Añadir etiquetas
para la masa

GUIs: Creación



```
121 - end
122
123 % Guardar el nuevo valor en handles
124 handles.volumen = volumen;
125 guidata(hObject,handles)
126
127 % --- Executes during object creation, after setting all properties.
128 function edit_volumen_CreateFcn(hObject, eventdata, handles)
129 % hObject    handle to edit_volumen (see GCBO)
130 % eventdata  reserved - to be defined in a future version of MATLAB
131 % handles    empty - handles not created until after all CreateFcns called
132
133 % Hint: edit controls usually have a white background on Windows.
134 % See ISPC and COMPUTER.
135 if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
136     set(hObject,'BackgroundColor','white');
137 end
138
139 % -----
140 function inicializar_gui(fig_handle, handles)
141
142 handles.densidad = 0;
143 handles.volumen = 0;
144
145 set(handles.edit_densidad, 'String', handles.densidad);
146 set(handles.edit_volumen, 'String', handles.volumen);
147
148 set(handles.text_masa, 'String', '0');
149
150 % Update handles structure
151 guidata(handles.figure1, handles);
152
153
```

Modificar la
función de
inicialización

GUIs: Creación

Editor - G:\Curso\2005-2006\Curso Fenosa\Programas\Mi_primer_GUI_repetido.m

File Edit Text Go Cell Tools Debug Desktop Window Help

Stack: Base

1.0 1.1 x % % ?

```
74 % Get default command line output from handles structure
75 varargout{1} = handles.output;
76
77 function edit_densidad_Callback(hObject, eventdata, handles)
78 % hObject handle to edit_densidad (see GCBO)
79 % eventdata reserved - to be defined in a future version of MATLAB
80 % handles structure with handles and user data (see GUIDATA)
81
82 % Hints: get(hObject,'String') returns contents of edit_densidad as text
83 % str2double(get(hObject,'String')) returns contents of edit_densidad as a double
84
85 % Recoger el valor del interfaz
86 densidad = str2double(get(hObject, 'String'));
87 if isnan(densidad)
88     set(hObject, 'String', 0);
89     errordlg('La entrada debe ser un número','Error');
90 end
91
92 % Guardar el nuevo valor en handles
93 handles.densidad = densidad;
94 guidata(hObject,handles)
95
96 % Actualizar masa
97 update_masa(handles)
98
99 % --- Executes during object creation, after setting all properties.
100 function edit_densidad_CreateFcn(hObject, eventdata, handles)
101     guidata(handles.figure1, handles);
102
103 % Actualizar la masa en la interfaz
104 function update_masa(handles)
105     set(handles.text_masa, 'String', num2str(handles.densidad*handles.volumen))
```

Crear función update_masa

Actualizar la masa en la interfaz

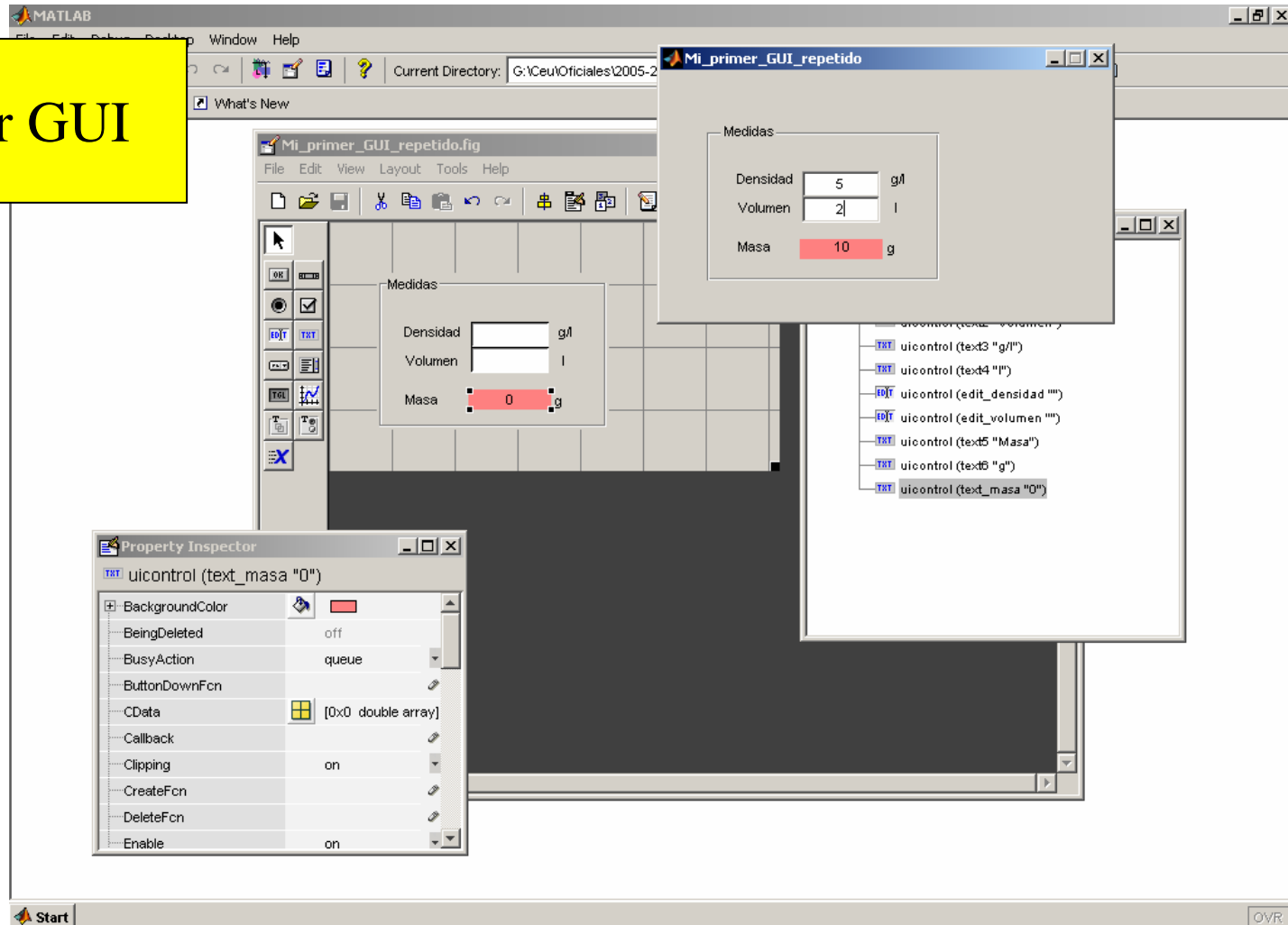
Mi_primer_GUI_rep... x Mi_primer_GUI.m x

Mi_primer_GUI_repetido / inicializa... Ln 159 Col 36

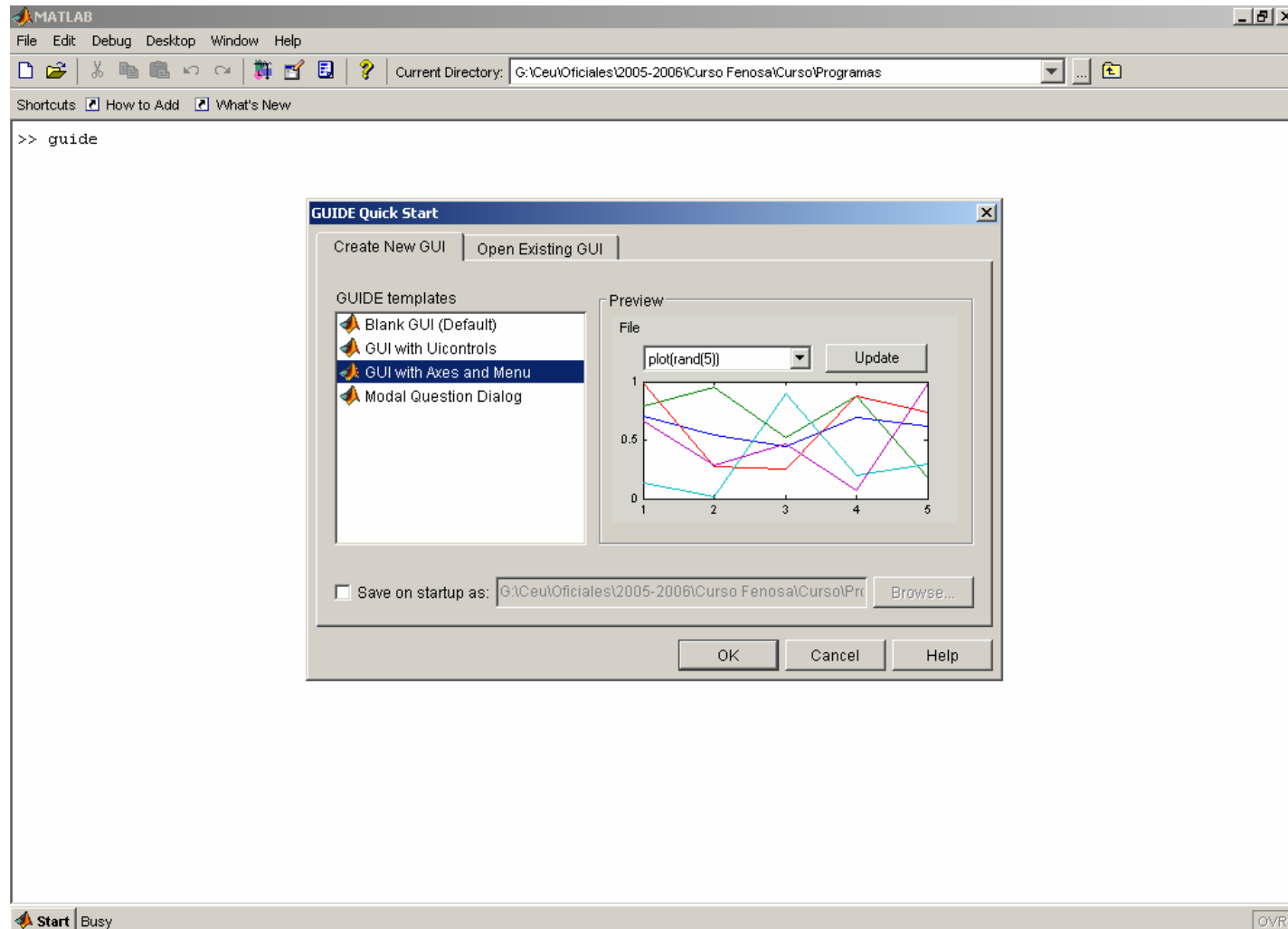
Crear función update_masa

GUIs: Creación

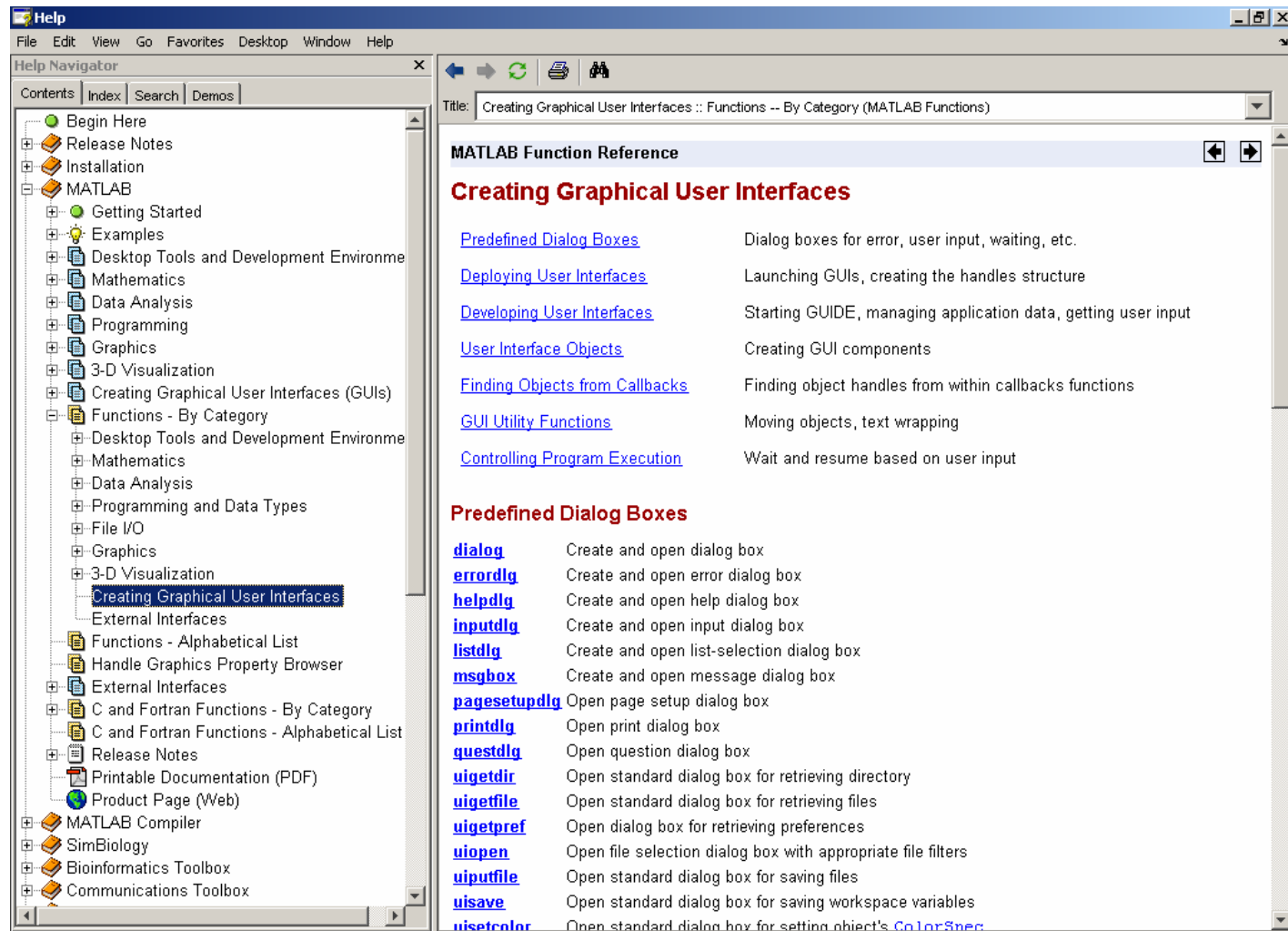
Ejecutar GUI



GUIs



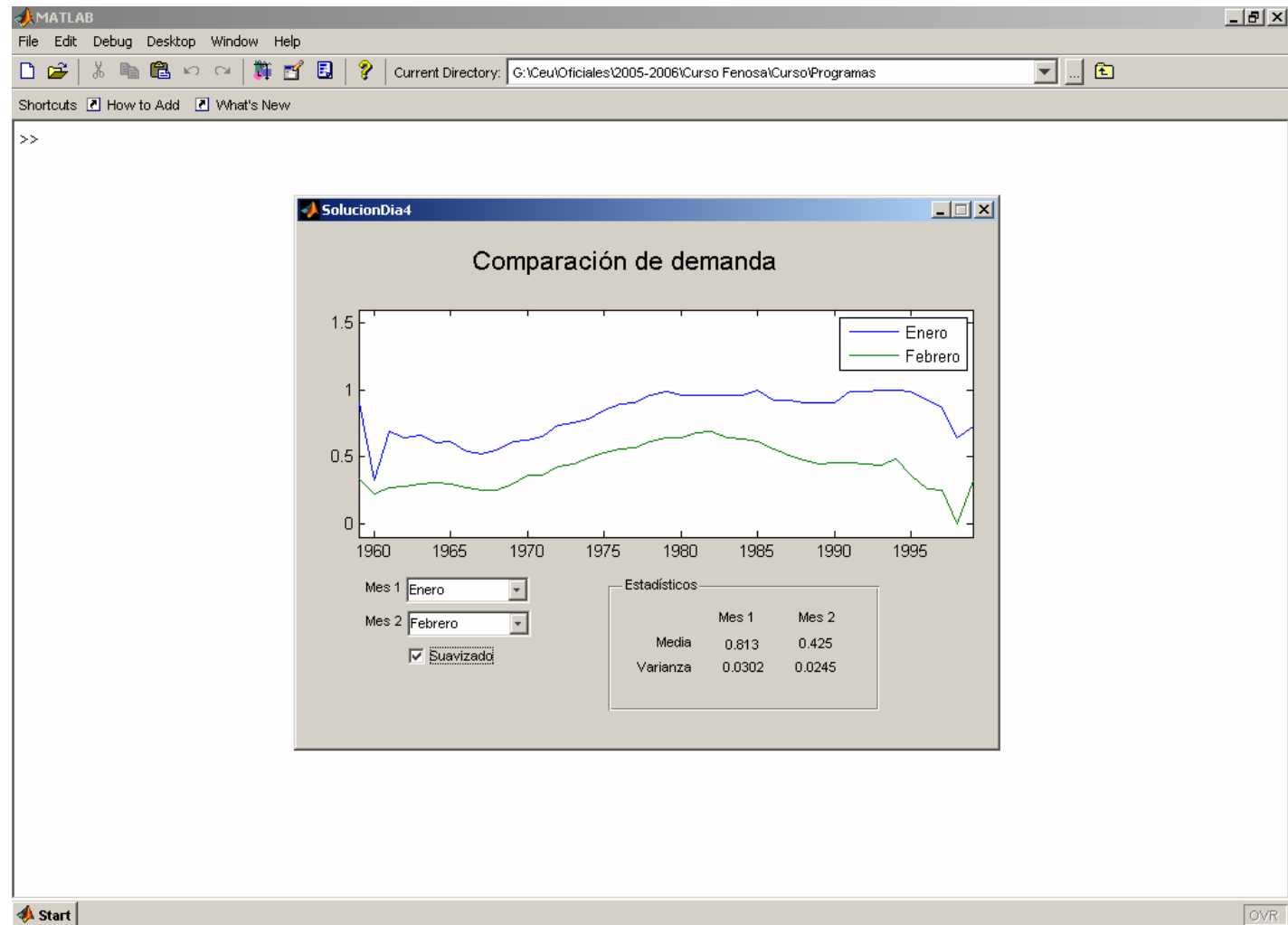
GUIs



Ejercicio final

- Implementar un GUI que permita comparar la demanda normalizada de dos meses.
- La comparación debe poder realizarse opcionalmente con una función de suavizado.
- Mostrar en un panel aparte la media y la varianza de los meses solicitados.

Ejercicio final



Generación de autoejecutables

The screenshot shows the MATLAB environment and a Windows Explorer window. In the MATLAB Command Window, the command `mcc -m SolucionDia4.m` is entered and highlighted with a green box. Below it, the output shows the selection of the Lcc C compiler (Compiler: 1) and the successful registration of DLLs. The Windows Explorer window shows the files generated by the `mcc` command, including `SolucionDia4.exe` (11 KB) and `SolucionDia4_mcr` (Carpetita de arch...), both highlighted with a green box.

MATLAB Command Window:

```
make a CTF shared/dynamically linked library and a1.m
mcc -W cpplib:liba -T link:lib a0 a1

See also compiler/function, mbuild, build

Reference page in Help browser
doc mcc

> mcc -m SolucionDia4.m

Select a compiler:
[1] Lcc C version 2.4.1 in G:\MATLAB\sys\lcc

[OK] None

Compiler: 1

Trying to update options file: C:\Documents and Settings\coss\Application Data\MathWorks\MATLAB\R2006a\compropts.bat
From template: G:\MATLAB\BIN\win32\mbuildopts\lcccomp.bat

Done . . .

--> "G:\MATLAB\bin\win32\mwregsvr" "G:\MATLAB\bin\win32\mwcomutil.dll"

DllRegisterServer in G:\MATLAB\bin\win32\mwcomutil.dll succeeded

--> "G:\MATLAB\bin\win32\mwregsvr" "G:\MATLAB\bin\win32\mwcommngr.dll"

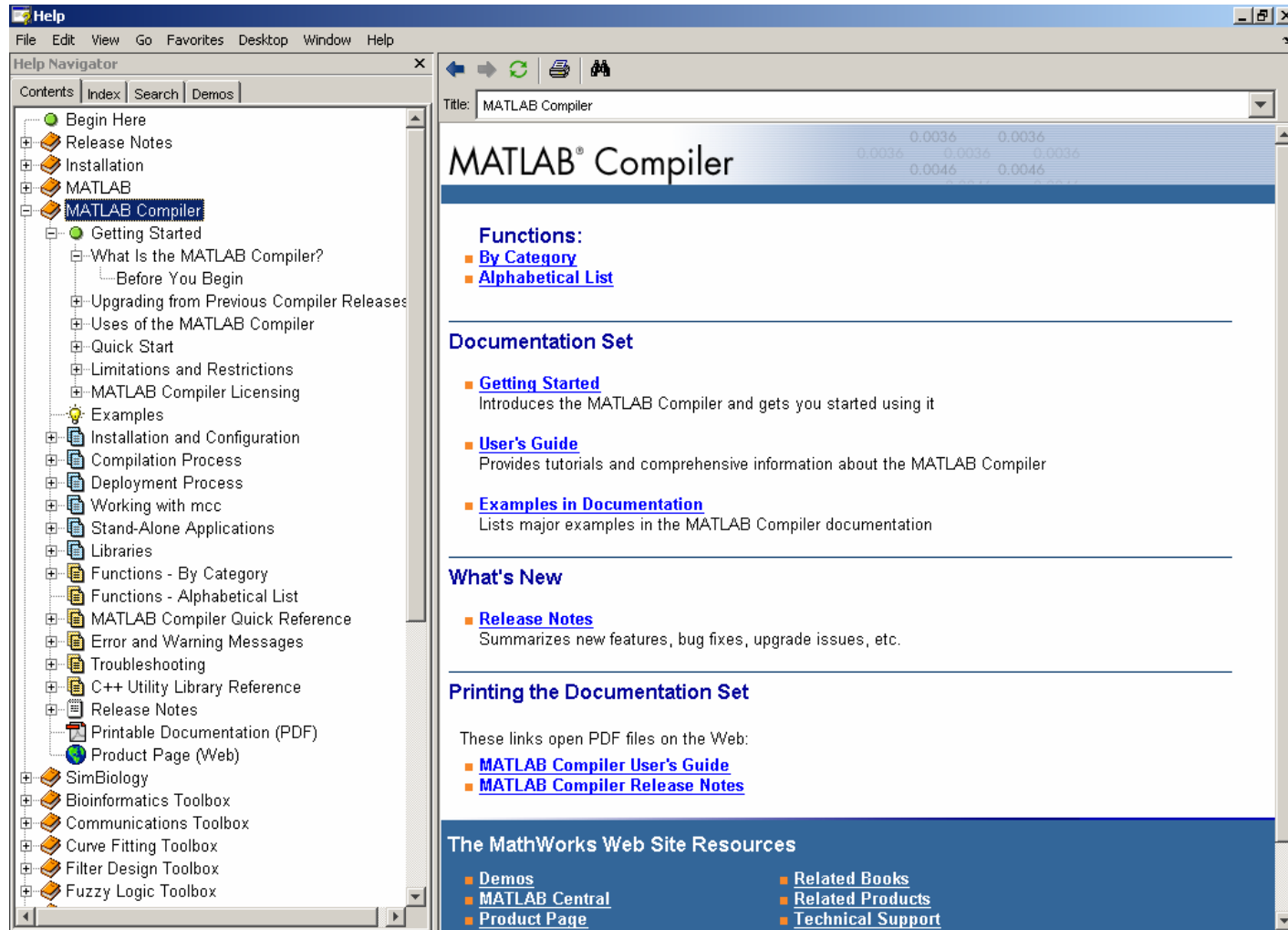
DllRegisterServer in G:\MATLAB\bin\win32\mwcommngr.dll succeeded

>>
```

Windows Explorer - Programas

Nombre	Tamaño	Tipo
SolucionDia2_2.m	1 KB	MATLAB M-file
SolucionDia2_leer_fichero.m	1 KB	MATLAB M-file
SolucionDia3_1.m	1 KB	MATLAB M-file
SolucionDia3_2.m	1 KB	MATLAB M-file
SolucionDia3_3.m	1 KB	MATLAB M-file
SolucionDia3_4.m	1 KB	MATLAB M-file
Worldatnight.jpg	62 KB	Imagen JPEG
SolucionDia4.m	8 KB	MATLAB M-file
SolucionDia4.fig	5 KB	MATLAB figure f
SolucionDia4.mcr	0 KB	Carpetita de arch...
mccExcludedFiles.log	19 KB	Documento de t
SolucionDia4.ctf	56 KB	Archivo CTF
SolucionDia4_main.c	3 KB	C Source File
SolucionDia4_mcc_component...	7 KB	C Source File
SolucionDia4.exe	11 KB	Aplicación
SolucionDia4_mcr		Carpetita de arch...

Generación de autoejecutables





CEU
*Universidad
San Pablo*

CURSO de UTILIZACIÓN PRÁCTICA de MATLAB

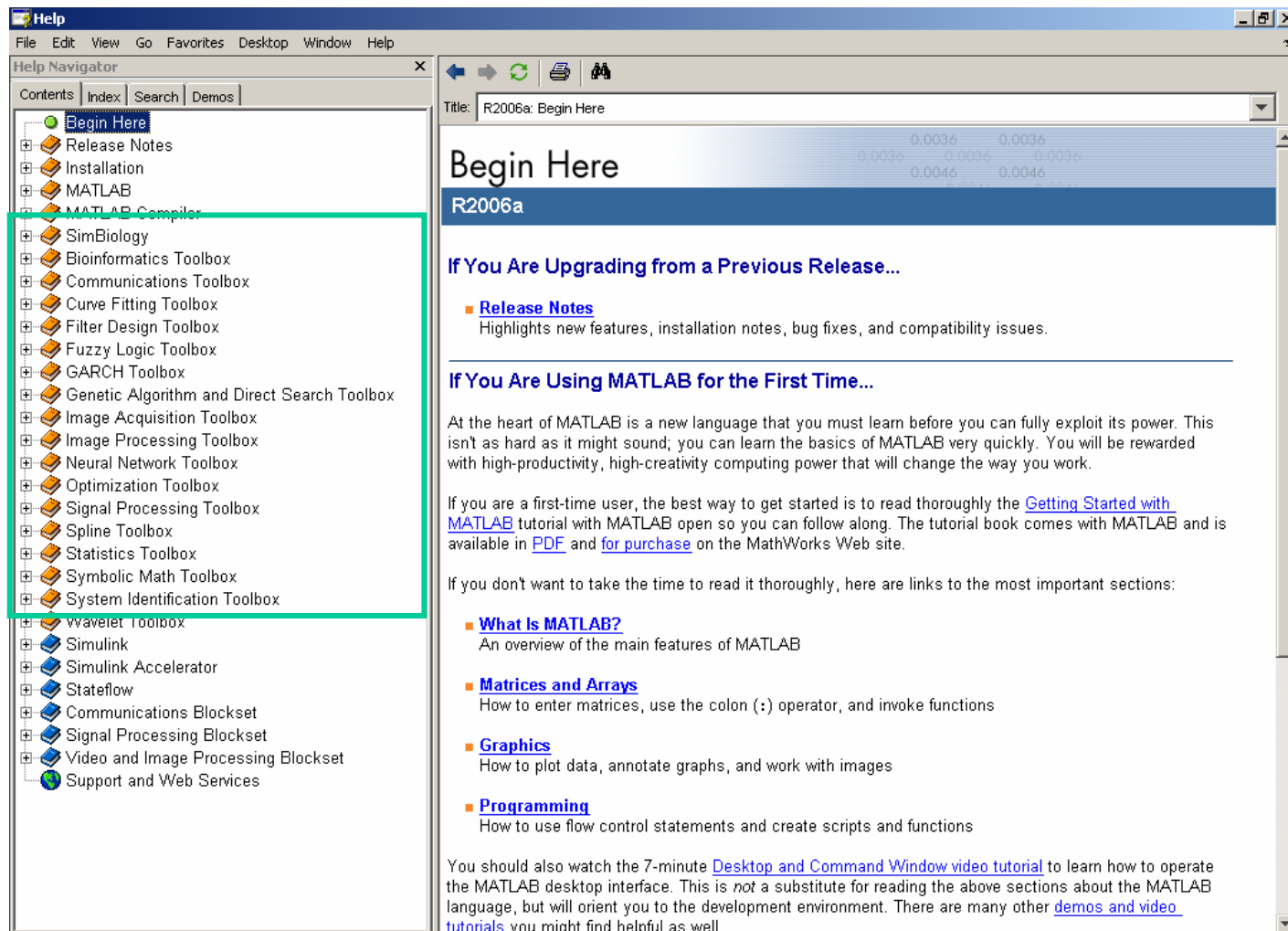
Sesión 5

Carlos Óscar Sánchez Sorzano, Ph.D.
Madrid, July 17th 2006

Cronograma del curso

- Día 1: Operaciones con matrices y vectores. Funciones de librería.
- Día 2: Otros tipos de datos en MATLAB. Programación en MATLAB.
- Día 3: Gráficos bidimensionales. Gráficos tridimensionales.
- Día 4: Interfaces de usuario en MATLAB. Generación de programas autónomos
- **Día 5: Librerías de interés práctico**
- Día 6: Interacción de MATLAB con Office y Visual Basic
- Día 7: Desarrollo de un proyecto

Librerías de interés práctico



The screenshot shows the MATLAB Help Navigator window. The left pane, titled 'Help Navigator', has tabs for 'Contents', 'Index', 'Search', and 'Demos'. The 'Contents' tab is active, showing a tree of topics. The 'Begin Here' topic is highlighted with a green box. The right pane, titled 'R2006a: Begin Here', displays the 'Begin Here' content for R2006a. It includes a table of contents, a section for 'If You Are Upgrading from a Previous Release...' with a link to 'Release Notes', and a section for 'If You Are Using MATLAB for the First Time...' with links to 'What Is MATLAB?', 'Matrices and Arrays', 'Graphics', and 'Programming'.

Begin Here

R2006a

If You Are Upgrading from a Previous Release...

- [Release Notes](#)
Highlights new features, installation notes, bug fixes, and compatibility issues.

If You Are Using MATLAB for the First Time...

At the heart of MATLAB is a new language that you must learn before you can fully exploit its power. This isn't as hard as it might sound; you can learn the basics of MATLAB very quickly. You will be rewarded with high-productivity, high-creativity computing power that will change the way you work.

If you are a first-time user, the best way to get started is to read thoroughly the [Getting Started with MATLAB](#) tutorial with MATLAB open so you can follow along. The tutorial book comes with MATLAB and is available in [PDF](#) and [for purchase](#) on the MathWorks Web site.

If you don't want to take the time to read it thoroughly, here are links to the most important sections:

- [What Is MATLAB?](#)
An overview of the main features of MATLAB
- [Matrices and Arrays](#)
How to enter matrices, use the colon (:) operator, and invoke functions
- [Graphics](#)
How to plot data, annotate graphs, and work with images
- [Programming](#)
How to use flow control statements and create scripts and functions

You should also watch the 7-minute [Desktop and Command Window video tutorial](#) to learn how to operate the MATLAB desktop interface. This is *not* a substitute for reading the above sections about the MATLAB language, but will orient you to the development environment. There are many other [demos and video tutorials](#) you might find helpful as well.

Librerías de interés práctico

The screenshot shows the MathWorks website in a Microsoft Internet Explorer browser window. The address bar displays http://www.mathworks.com/applications/tech_computing/. The page features a navigation menu with links to Home, Select Country, Contact Us, and Store. The main content area is titled "Technical Computing" and describes it as "Mathematical computation, analysis, visualization, and algorithm development". It includes a code snippet:

```
% concatenate the new ele  
c=[c; 1];
```

 and a 3D surface plot. The page also promotes the R2006b release, available in September 2006, and offers a free seminar on "Managing Risk with Extreme Value Theory and Copulas: A MATLAB Financial Case Study". A sidebar on the left lists links for Technical Computing Main Page, Description, Demos and Webinars, User Stories, Training, Recommended Products, Technical Literature, News, and Events. A "Free Technical Kit" section is also visible at the bottom.

Librerías de interés práctico

The screenshot shows a Microsoft Internet Explorer browser window displaying the MathWorks website. The address bar shows the URL <http://www.mathworks.com/applications/controldesign/>. The page features a navigation menu with links for Home, Select Country, Contact Us, Store, and Search. The main content area is titled "Control Design" and includes a section for "Model-Based Design for control systems" with a description of embedded control systems and a list of "Description Topics" such as Model-Based Design, Modeling and Simulation, Control System Software Design, and Embedded Software Testing and Implementation. A "Free Technical Kit" is also advertised. The right sidebar contains links for Contact sales, E-mail this page, and Print this page, along with information about the R2006b release and a Free Seminar on Rapid Prototyping and HIL Testing of Aerospace and Defense Applications Using Simulink.

The MathWorks - Control Design Solutions - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

Atrás Búsqueda Favoritos

Dirección <http://www.mathworks.com/applications/controldesign/> Ir Vínculos

Google Ir Marcadores PageRank 217 bloqueados Corrector ortográfico Traducir Configuración

The MathWorks Accelerating the pace of engineering and science

Home | Select Country | Contact Us | Store Search

Create Account | Log In

Products & Services Industries Academia Support User Community Company

Control Design Main Page

Description

Demos and Webinars

User Stories

Training

Recommended Products

Technical Literature

News

Events

Control Design

Model-Based Design for control systems

Designers of embedded control systems rely on products from The MathWorks to design and create software that is used in aerospace, defense, automotive, industrial equipment, process control, and many other applications. Using MathWorks products can reduce development costs, decrease time-to-market, and improve quality. Learn how you can meet your goals and set new standards on your next embedded control system design project.

Description Topics

- Model-Based Design
- Modeling and Simulation
- Control System Software Design
- Embedded Software Testing and Implementation

Free Technical Kit

News and Events

Contact sales

E-mail this page

Print this page

R2006b

Available Sept. 2006

Learn more

Download now

Free Seminar

Rapid Prototyping and HIL Testing of Aerospace and Defense Applications Using Simulink

Learn more

Upcoming Webinar

Hybrid Dynamic Systems in Automotive Control Design new

Register today

Listo Internet

Librerías de interés práctico

The screenshot shows a Microsoft Internet Explorer browser window displaying the MathWorks Test & Measurement website. The browser's address bar shows the URL http://www.mathworks.com/applications/t_m/. The website header includes the MathWorks logo and navigation links: Home, Select Country, Contact Us, and Store. A search bar is also present. Below the header, a navigation menu lists: Products & Services, Industries, Academia, Support, User Community, and Company. The main content area is titled "Test & Measurement" and features a sub-header "Hardware connectivity and data analysis for test and measurement applications". This section includes an image of electronic test equipment and a description: "The MathWorks Test & Measurement solution provides a complete set of tools for test, data analysis and modeling, and presentation-quality reports, all in a single environment." Below this, there is a "Description Topics" section with links to "Integrating Acquisition and Analysis", "Interfacing with External Devices", "Analyzing and Visualizing Data", "Developing Algorithms and Models", and "Presenting and Reporting Results". A "Free Technical Kit" section is also visible. On the right side of the page, there are links for "Contact sales", "E-mail this page", and "Print this page". Below these, a section for "R2006b" is shown, indicating it is available from September 2006, with links to "Learn more" and "Download now". Further down, a "Free Seminar" section describes a seminar on "Implementing Measurement and Analysis Techniques Using MATLAB" with a "Learn more" link. At the bottom, a "Recorded Webinar" section describes a webinar on "Deploying an Application Containing Data Acquisition, Instrument Control and Data Analysis" with a "Register to view now" link. The left sidebar of the website contains a "Test & Measurement Main Page" section with links to "Description", "Demos and Webinars", "User Stories", "Training", "Recommended Products", "Technical Literature", "News", and "Events". The browser's status bar at the bottom shows "Listo" and "Internet".

Librerías de interés práctico

The screenshot shows a Microsoft Internet Explorer browser window displaying the MathWorks website. The address bar shows the URL http://www.mathworks.com/applications/fin_modeling/. The page features a navigation menu with links to Home, Select Country, Contact Us, Store, Search, Create Account, and Log In. Below this, there are tabs for Products & Services, Industries, Academia, Support, User Community, and Company. The main content area is titled "Financial Modeling and Analysis" and includes a description of MATLAB's capabilities in financial modeling, a list of description topics, and a free technical kit. On the right side, there are links for contact sales, email this page, and print this page, followed by a section for R2006b software available in September 2006, and a featured seminar on managing risk with extreme value theory and copulas.

The MathWorks
Accelerating the pace of engineering and science

Home | Select Country | Contact Us | Store | Search | Create Account | Log In

Products & Services | Industries | Academia | Support | User Community | Company

Financial Modeling and Analysis

Financial modeling, analysis, and application deployment

MATLAB, The MathWorks flagship product, provides a broad range of capabilities to easily access and manipulate large data sets, quickly develop new financial algorithms, and automatically create the necessary components to integrate new models into your existing systems. MATLAB offers you the ability to develop and deploy new models quickly, giving you a competitive advantage.

Description Topics

- [Analyzing and Modeling Financial Data](#)
- [Rapidly Deploying Financial Applications](#)
- [Why use MATLAB](#)
- [Introduction to MATLAB in Finance](#)

Free Technical Kit

News and Events

- Webinar: MATLAB® for C/C++ Programmers

R2006b
Available Sept. 2006
[Learn more](#)
[Download now](#)

Financial Derivatives Toolbox 3.0.1
Model and analyze equity and fixed-income derivatives
[Learn more](#)

Featured Seminar
Managing Risk with Extreme Value Theory and Copulas: A MATLAB Financial Case Study
[Learn more](#)

Free Seminar
Using MATLAB to Develop and Deploy Financial

Librerías de interés práctico

The screenshot shows the MathWorks website in a Microsoft Internet Explorer browser window. The address bar displays the URL http://www.mathworks.com/products/product_listing/index.html. The page features a navigation menu with links to Home, Select Country, Contact Us, Store, and Search. The main content area is titled "Product List" and is organized into two columns: "MATLAB Product Family" and "Simulink Product Family". The MATLAB column lists various toolboxes, including the "Math and Optimization" section, which is highlighted with a green box. The Simulink column lists toolboxes for fixed-point modeling, event-based modeling, and physical modeling. The page also includes a sidebar with links to "Products & Services Main Page", "Product List", "New Products", "Product Overview", "Third-Party Products & Services", and "Purchase Products Online".

The MathWorks
Accelerating the pace of engineering and science

Home | Select Country | Contact Us | Store | Search

Create Account | Log In

Products & Services | Industries | Academia | Support | User Community | Company

Products & Services Main Page
Product List
New Products
Product Overview
Third-Party Products & Services
Purchase Products Online

Product List

By Category | Alphabetically

Show Product Descriptions

MATLAB Product Family

[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](#)

Simulink Product Family

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

Fixed-Point Modeling
[Simulink® Fixed Point](#)

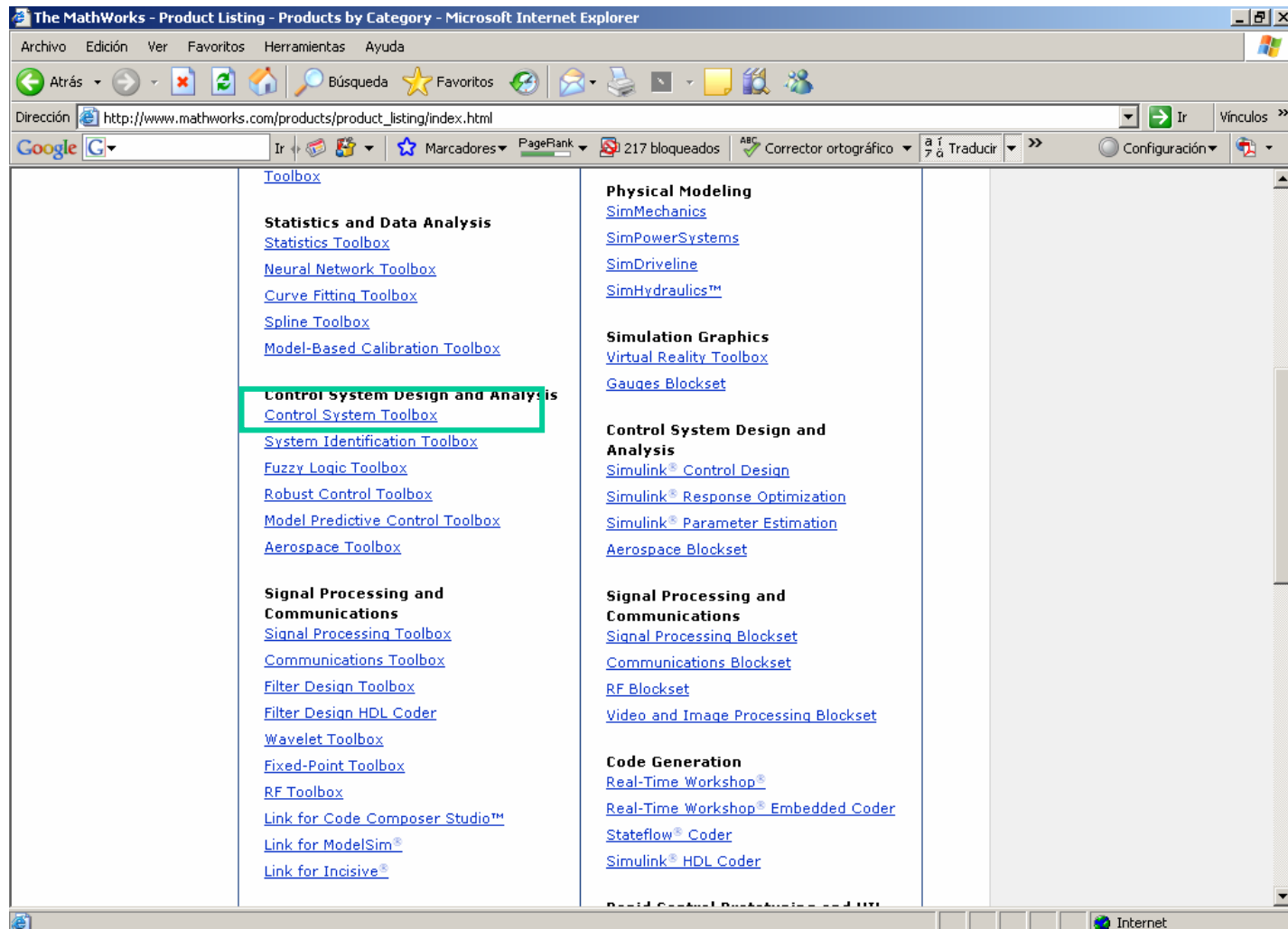
Event-Based Modeling
[Stateflow®](#)
[SimEvents™](#)

Physical Modeling
[SimMechanics](#)
[SimPowerSystems](#)
[SimDriveline](#)
[SimHydraulics™](#)

Contact sales
E-mail this page

Internet

Librerías de interés práctico



Librerías de interés práctico

The screenshot shows a web browser window titled "The MathWorks - Product Listing - Products by Category - Microsoft Internet Explorer". The address bar displays "http://www.mathworks.com/products/product_listing/index.html". The page content is organized into several sections:

- 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](#)
- Verification, Validation, and Testing**
 - [Embedded Target for TI C2000™ DSP](#)
 - [Embedded Target for TI C6000™ DSP](#)
 - [Link for Code Composer Studio™](#)
 - [Link for ModelSim®](#)
 - [Simulink® Verification and Validation](#)
 - [SystemTest](#)
 - [Link for TASKING®](#)
- Third-Party Products**
 - [Third-party products and services](#) that complement MATLAB and Simulink

The footer contains copyright information: "© 1994-2006 The MathWorks, Inc." and links to "Site Help", "Patents", "Trademarks", "Privacy Policy", "Preventing Piracy", and "RSS".

Librerías de interés práctico

The screenshot shows the MathWorks product listing page in a Microsoft Internet Explorer browser window. The address bar displays the URL: http://www.mathworks.com/products/product_listing/index.html. The page content is organized into several columns and sections. The following table summarizes the visible categories and their associated toolboxes, with green boxes highlighting specific items of interest.

Category	Toolboxes
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
Rapid Control Prototyping and HIL SW/HW	xPC Target xPC Target Embedded Option xPC TargetBox® Real-Time Windows Target
Embedded Targets	Embedded Target for Infineon C166® Microcontrollers Embedded Target for Motorola® HC12 Embedded Target for Motorola® MPC555 Embedded Target for TI C2000™ DSP Embedded Target for TI C6000™ DSP
Verification, Validation, and Testing	Link for Code Composer Studio™ Link for ModelSim® Simulink® Verification and Validation SystemTest Link for TASKING®



CEU
*Universidad
San Pablo*

CURSO de UTILIZACIÓN PRÁCTICA de MATLAB

Sesión 6

Carlos Óscar Sánchez Sorzano, Ph.D.
Madrid, July 17th 2006

Cronograma del curso

- Día 1: Operaciones con matrices y vectores. Funciones de librería.
- Día 2: Otros tipos de datos en MATLAB. Programación en MATLAB.
- Día 3: Gráficos bidimensionales. Gráficos tridimensionales.
- Día 4: Interfaces de usuario en MATLAB. Generación de programas autónomos
- Día 5: Librerías de interés práctico
- **Día 6: Interacción de MATLAB con Office y Visual Basic**
- Día 7: Desarrollo de un proyecto

Interacción con Word

The screenshot illustrates the MATLAB Notebook interface and its integration with Microsoft Word. The MATLAB window shows a notebook with a code cell containing the following MATLAB code:

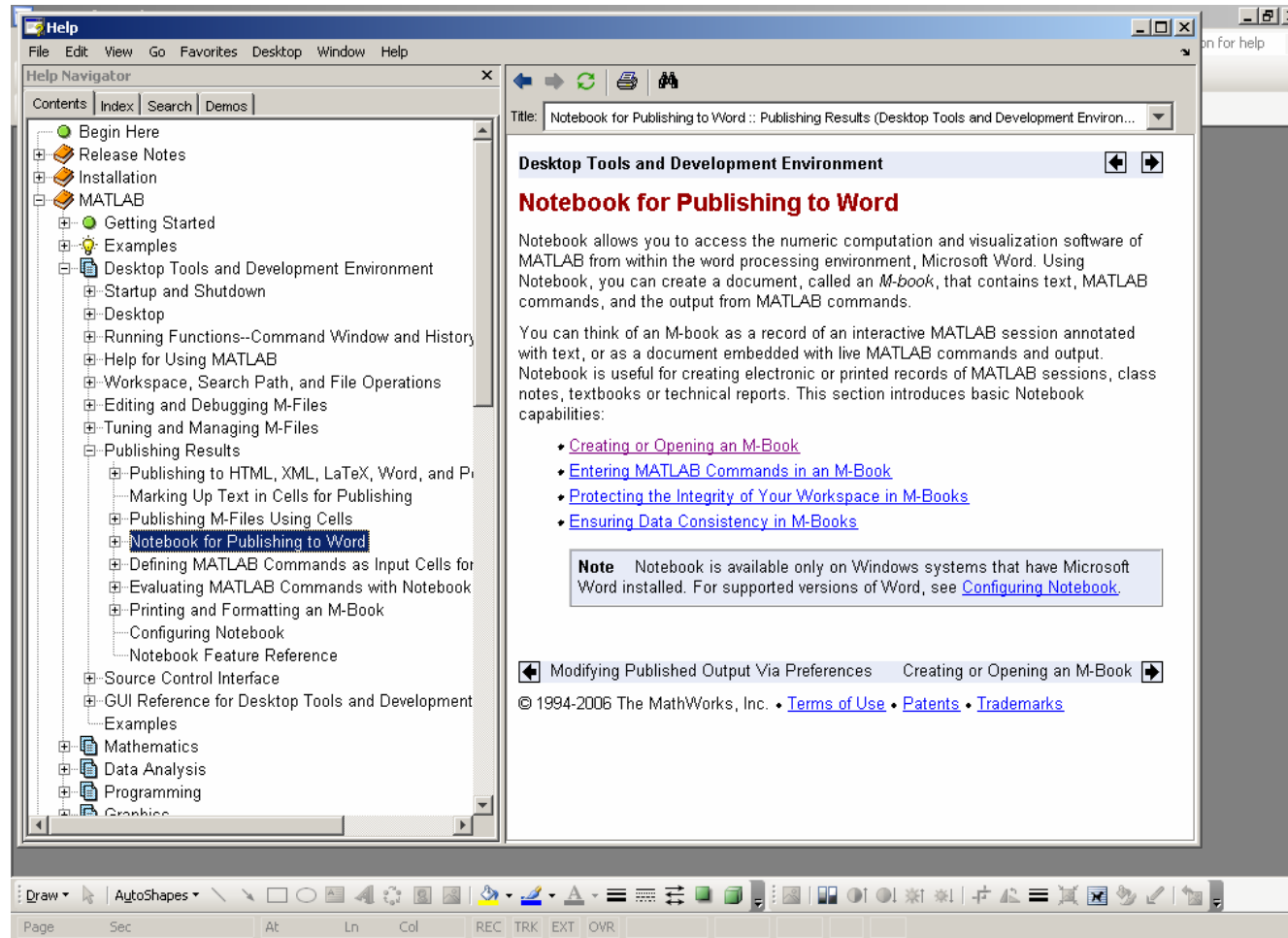
```
t=0:0.01:1;  
plot(t,sin(pi*t))
```

The code has been executed, resulting in a plot of a sine wave. A context menu is open over the plot, listing various actions:

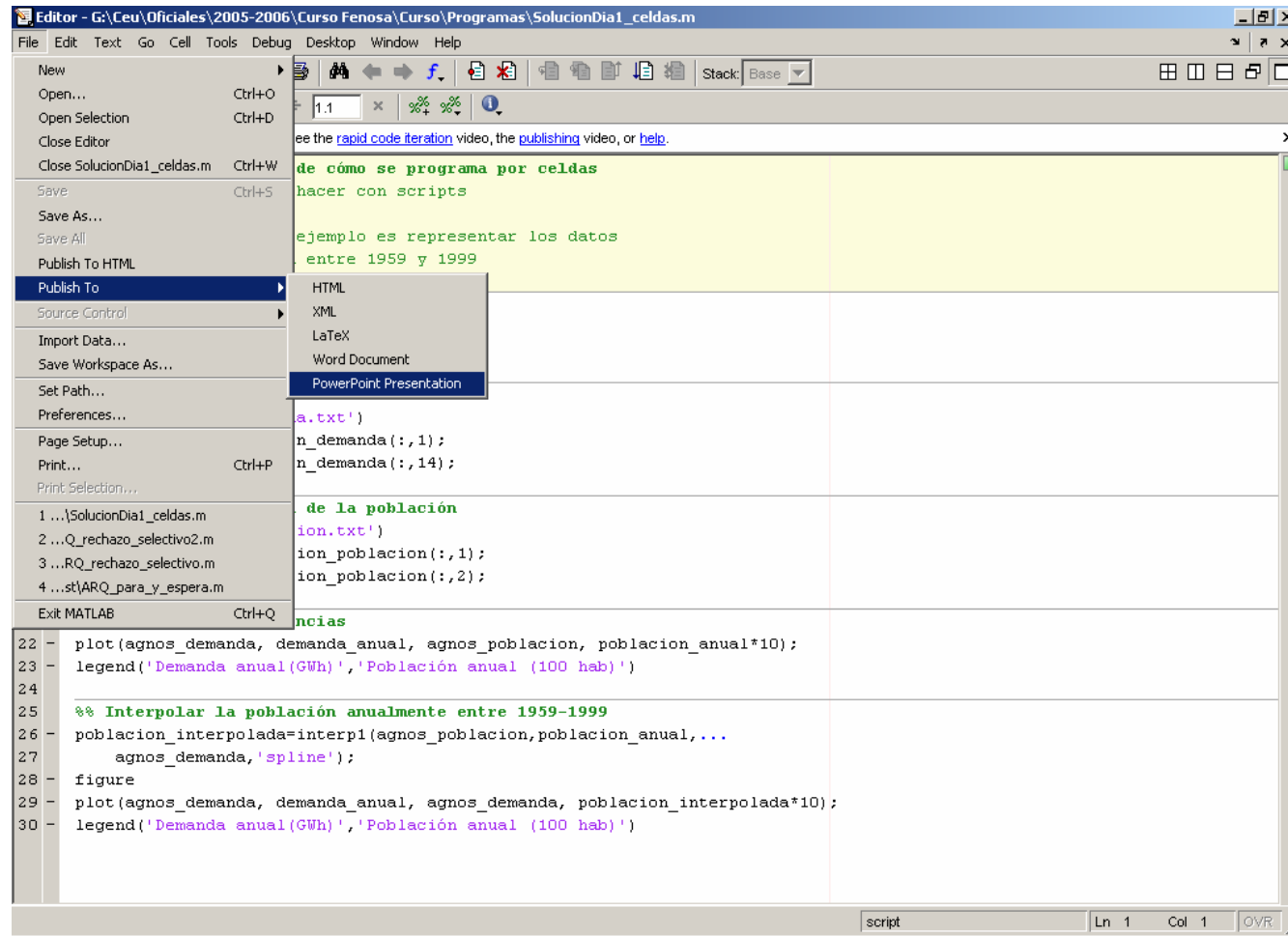
- Define Input Cell
- Define AutoInit Cell
- Define Calc Zone
- Undefine Cells
- Purge Selected Output Cells
- Group Cells
- Ungroup Cells
- Hide Cell Markers
- Toggle Graph Output for Cell
- Evaluate Cell
- Evaluate Calc Zone
- Evaluate M-book
- Evaluate Loop...
- Bring MATLAB to Front
- Notebook Options...

A large green arrow points from the MATLAB notebook area towards the Microsoft Word document, which contains the text "Esto es un ejemplo de interacción con Word". The Word document is titled "Document1 - Microsoft Word" and shows the MATLAB plot embedded within the text.

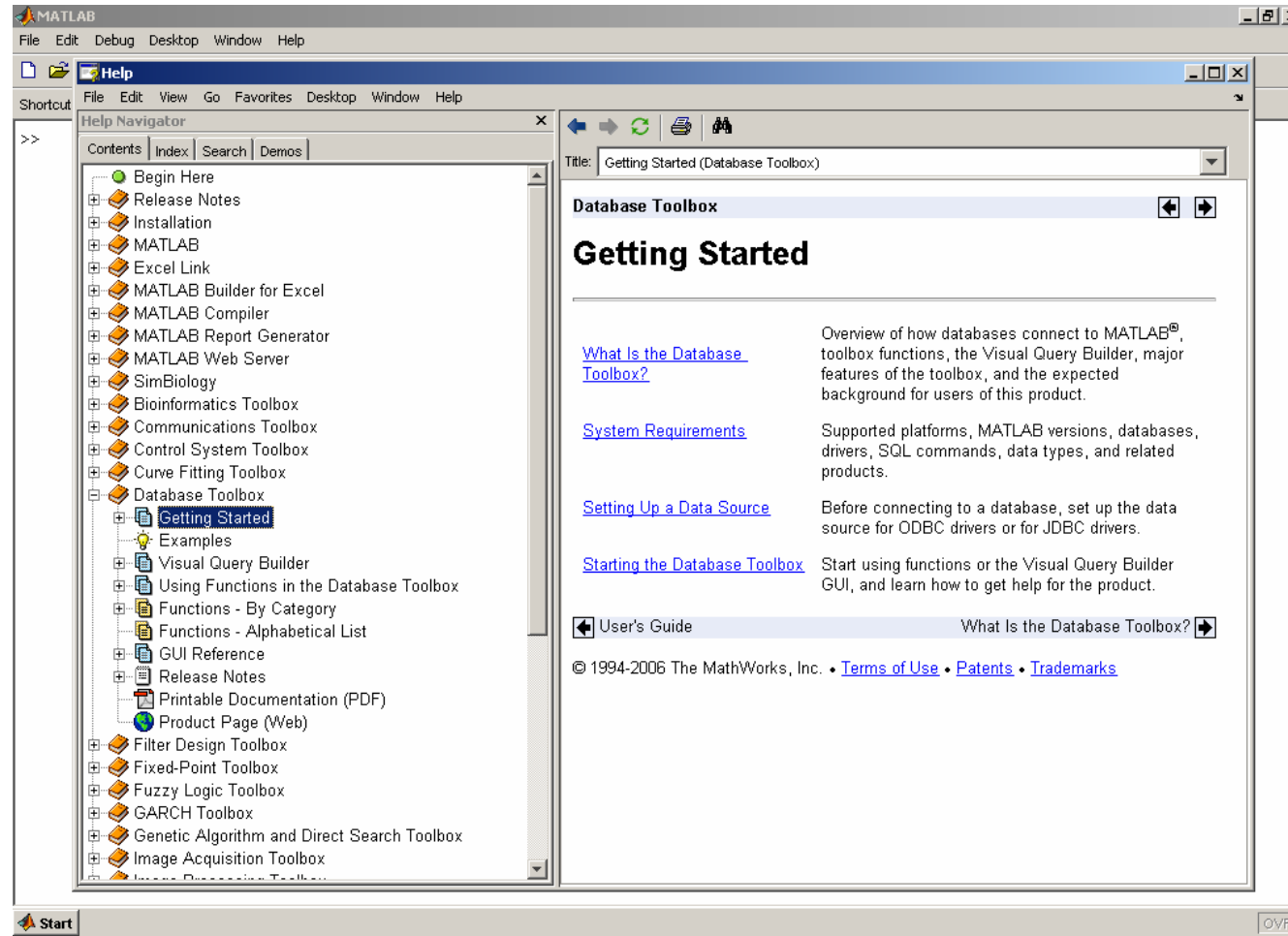
Interacción con Word



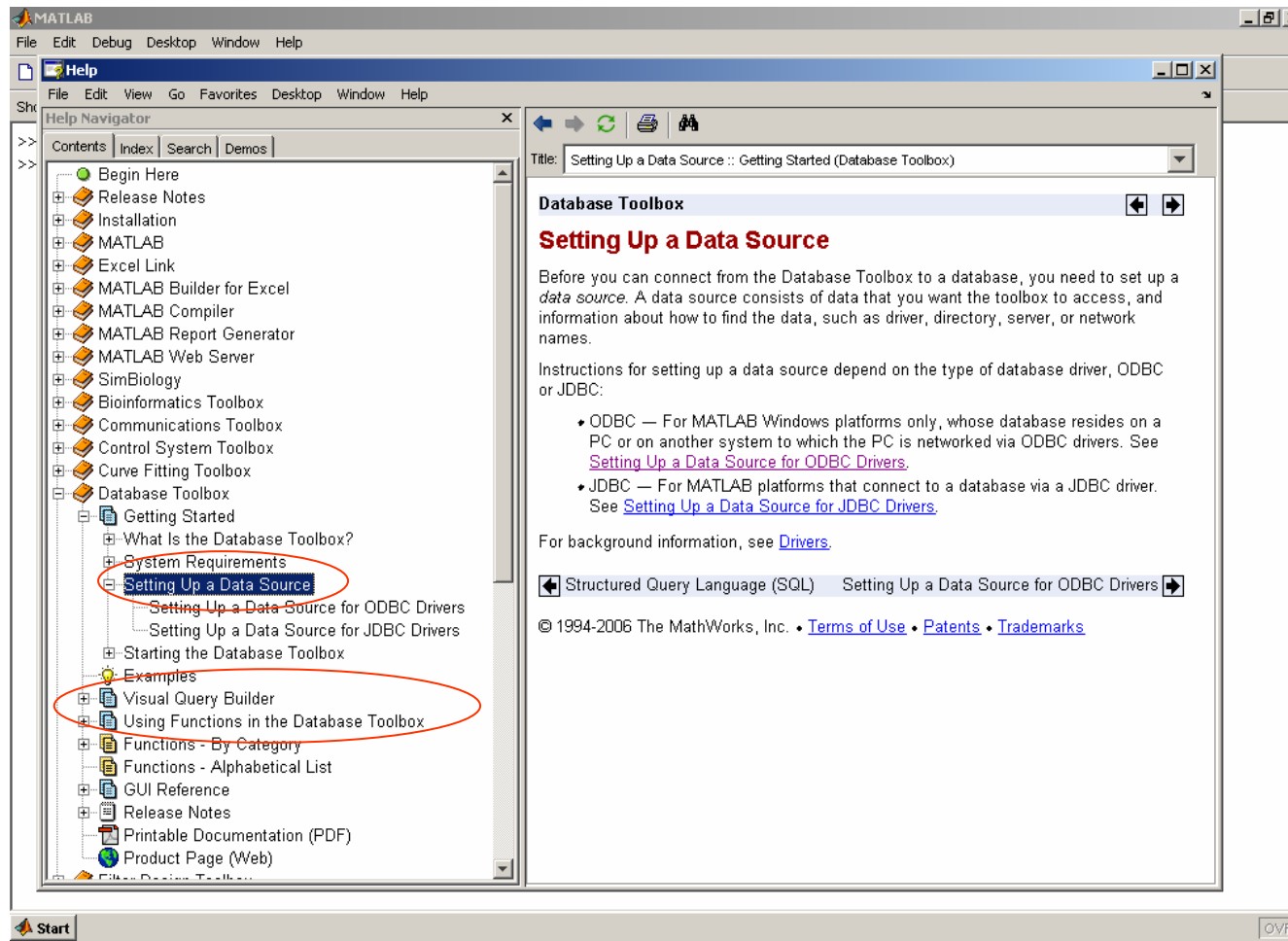
Interacción con Powerpoint



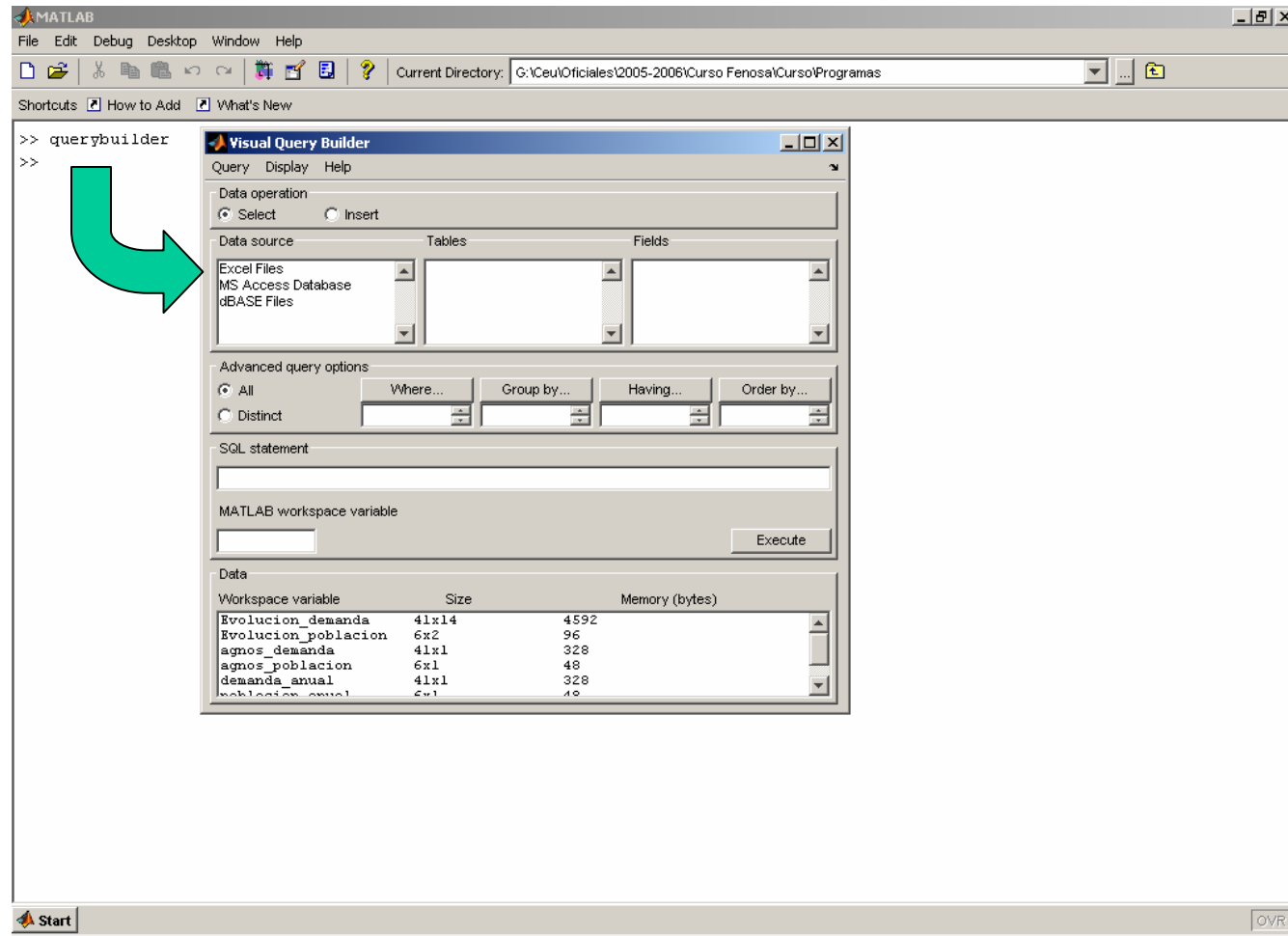
Interacción con Access



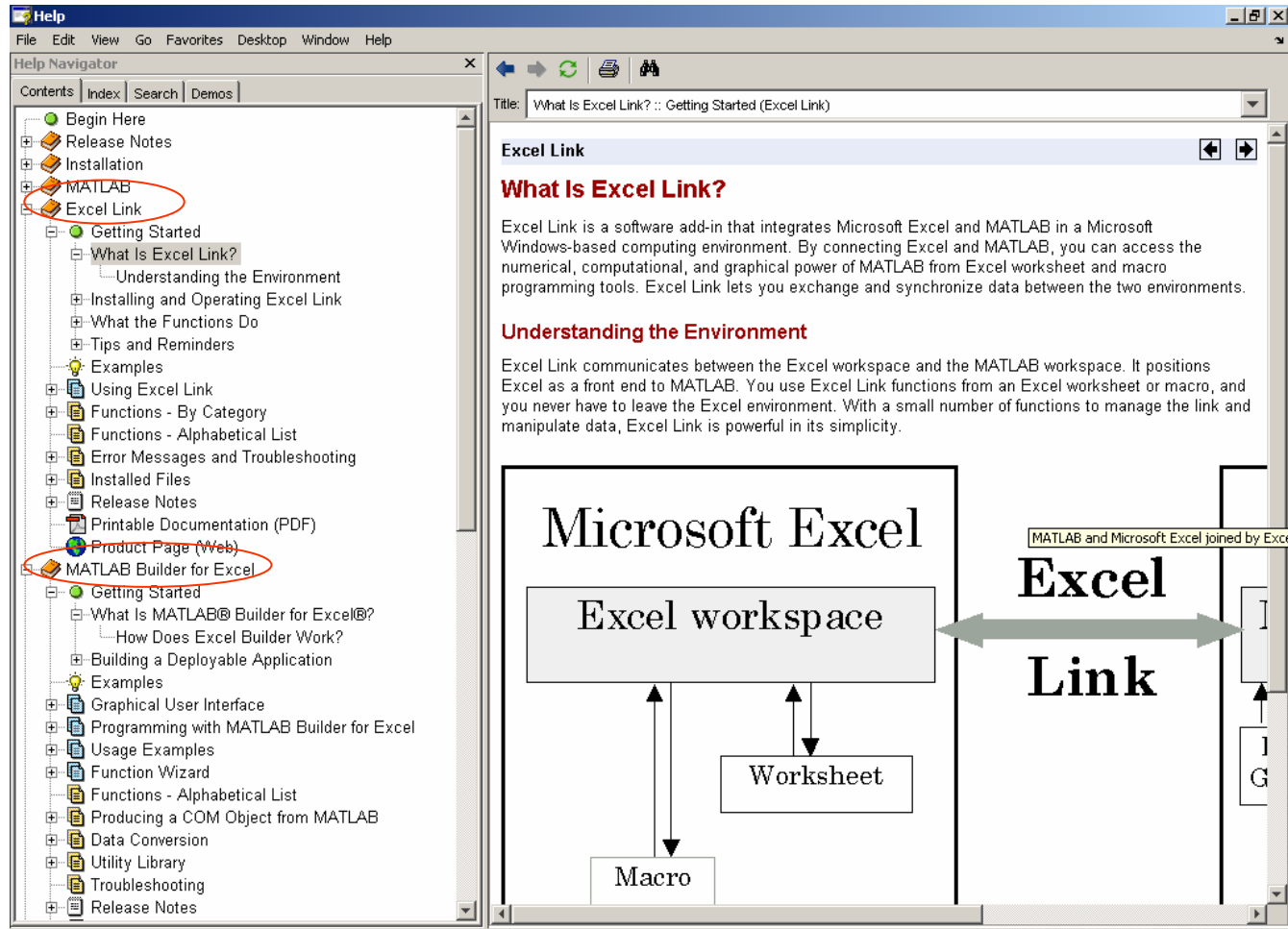
Interacción con Access



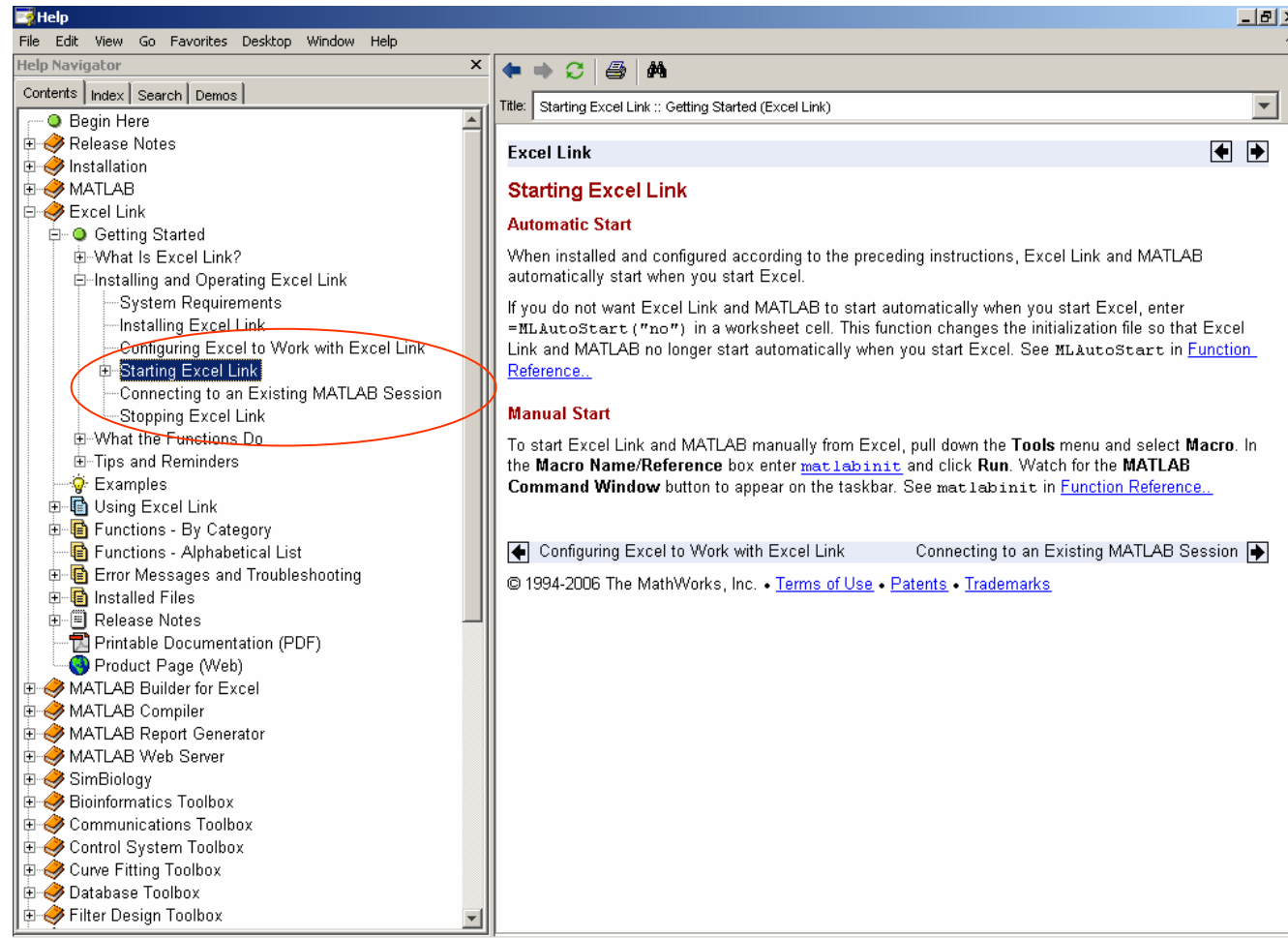
Interacción con Access



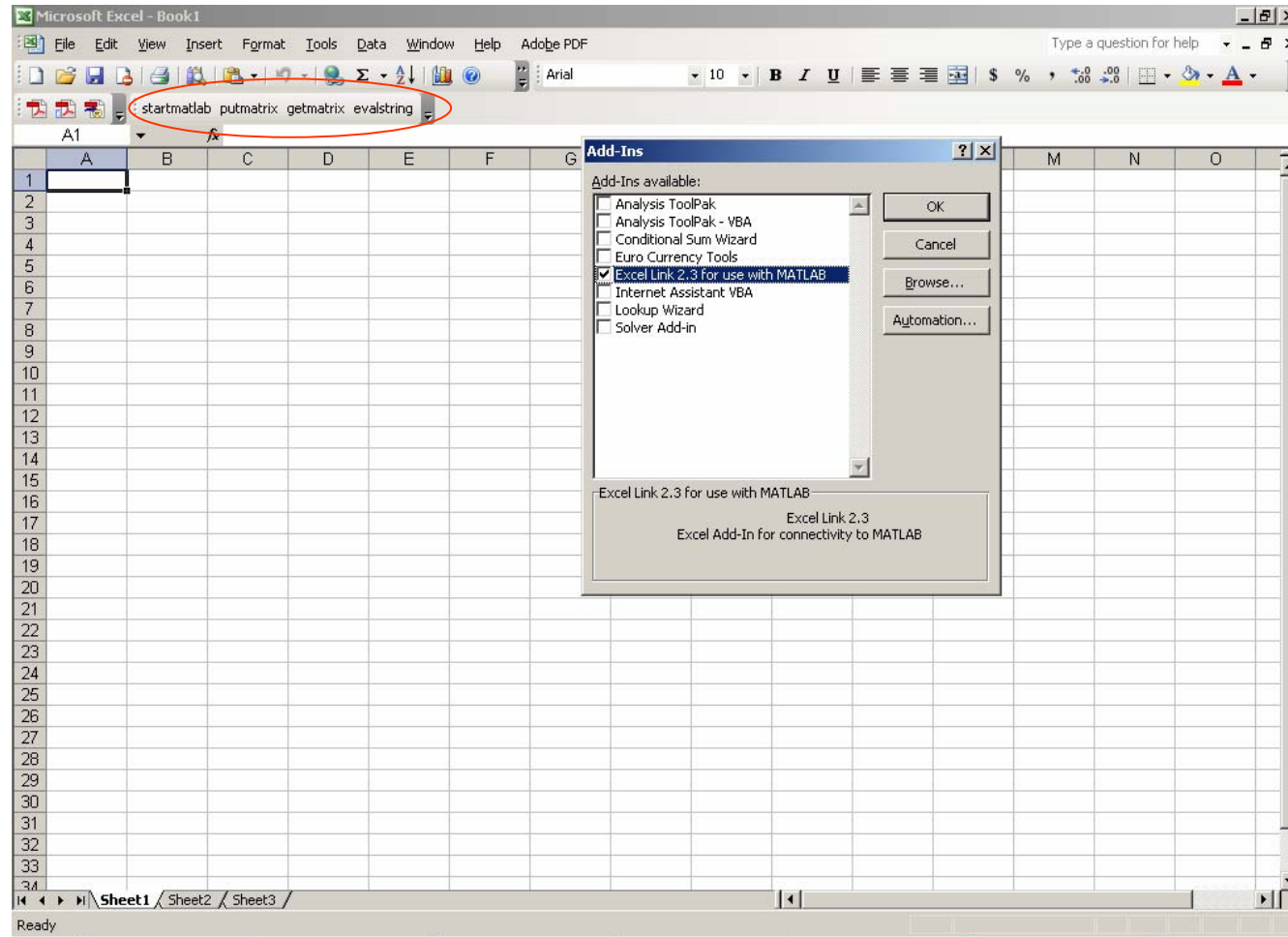
Interacción con Excel: Excel Link



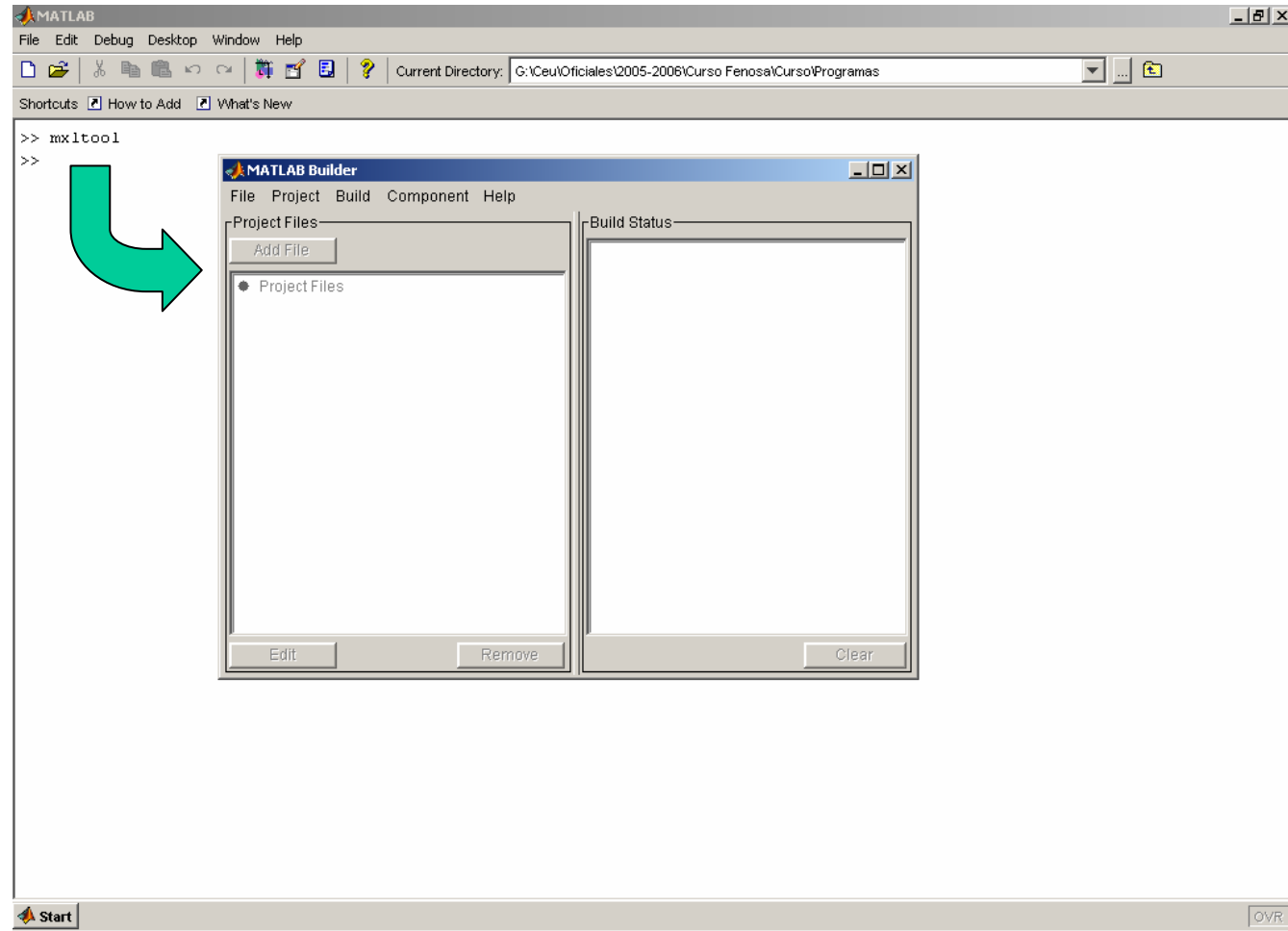
Interacción con Excel: Excel Link



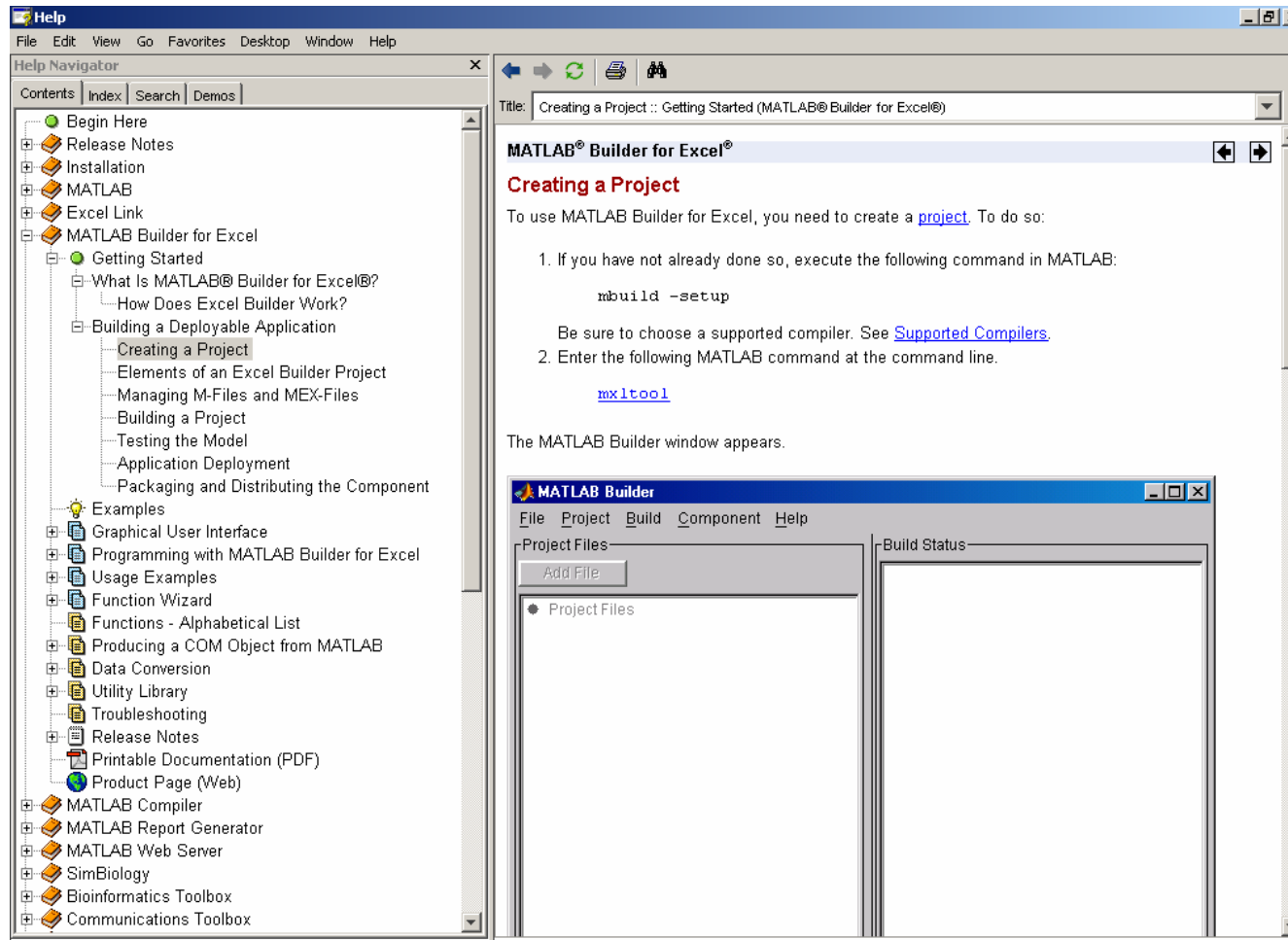
Interacción con Excel: Excel Link



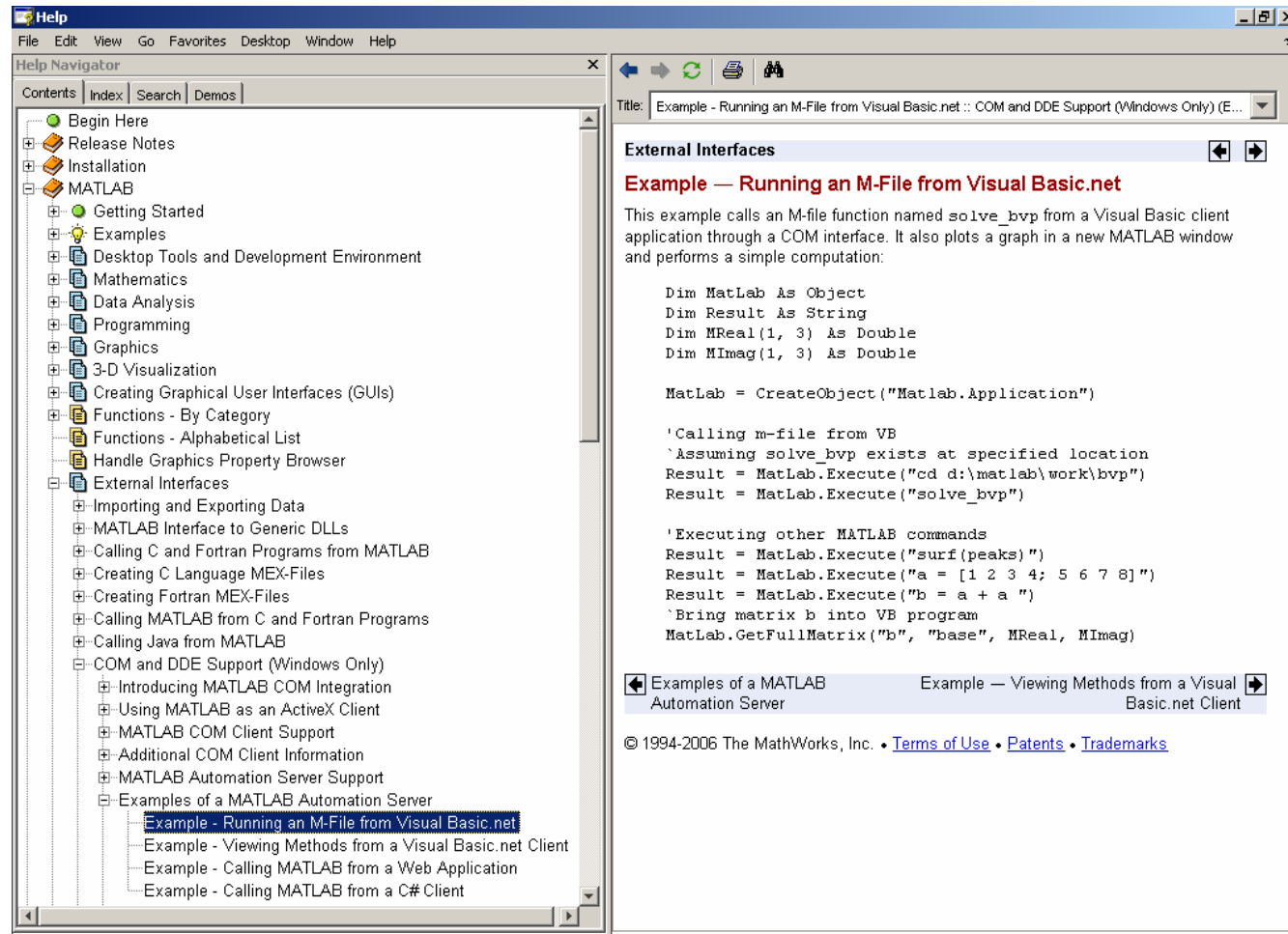
Interacción con Excel: Excel Builder



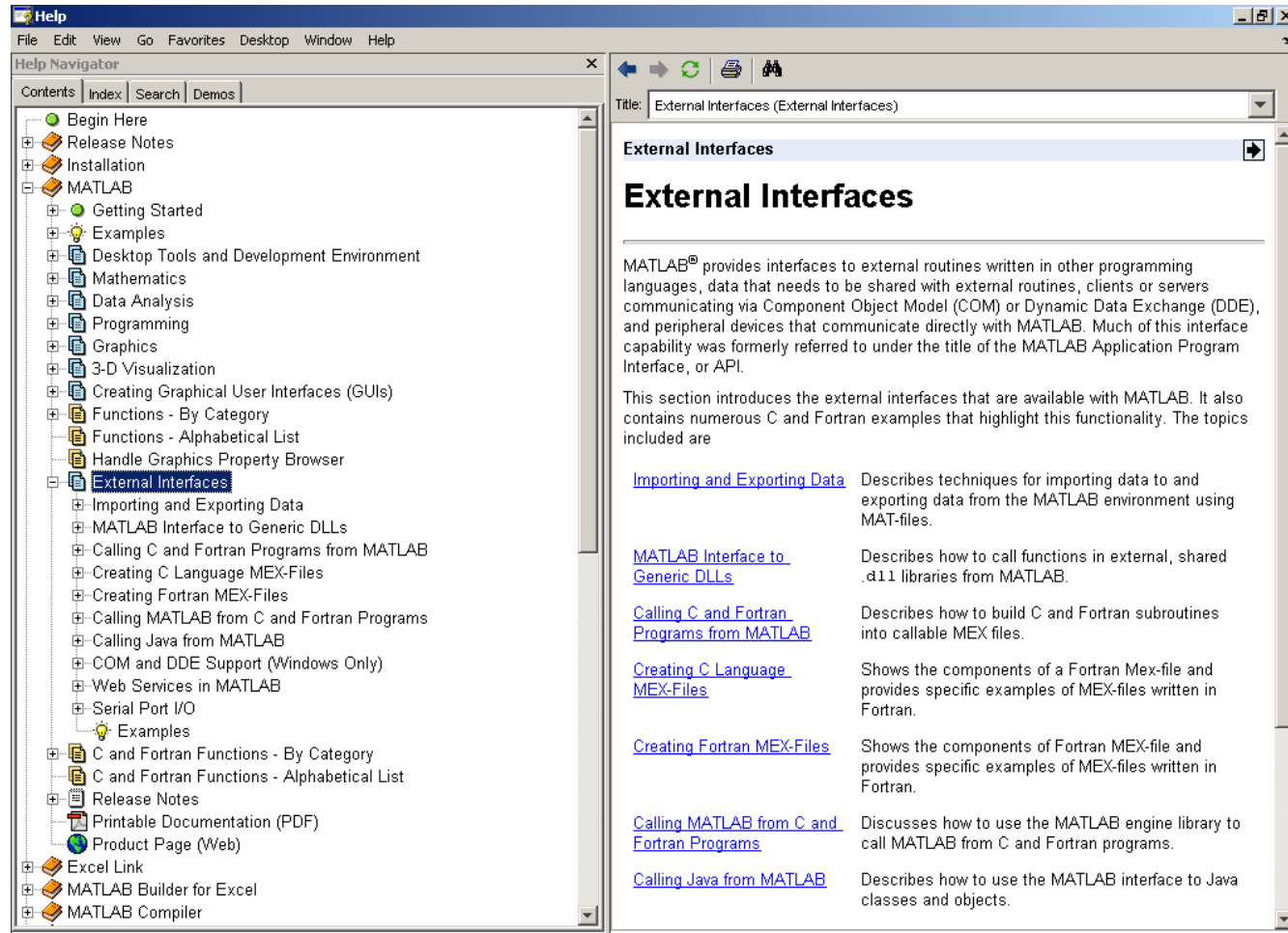
Interacción con Excel: Excel Builder



Interacción con Visual Basic



Ejercicio





CEU
*Universidad
San Pablo*

CURSO de UTILIZACIÓN PRÁCTICA de MATLAB

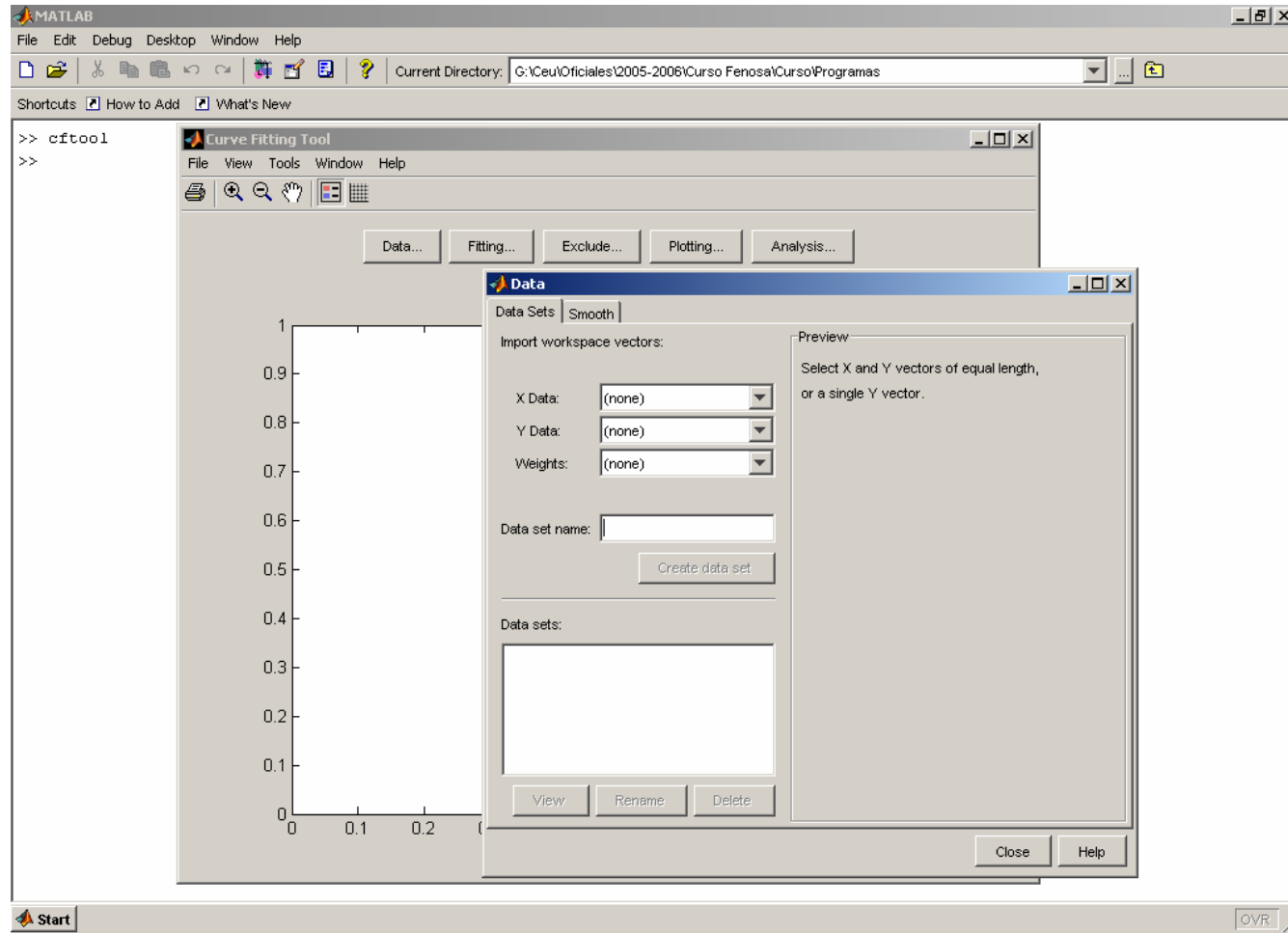
Sesión 7

Carlos Óscar Sánchez Sorzano, Ph.D.
Madrid, July 17th 2006

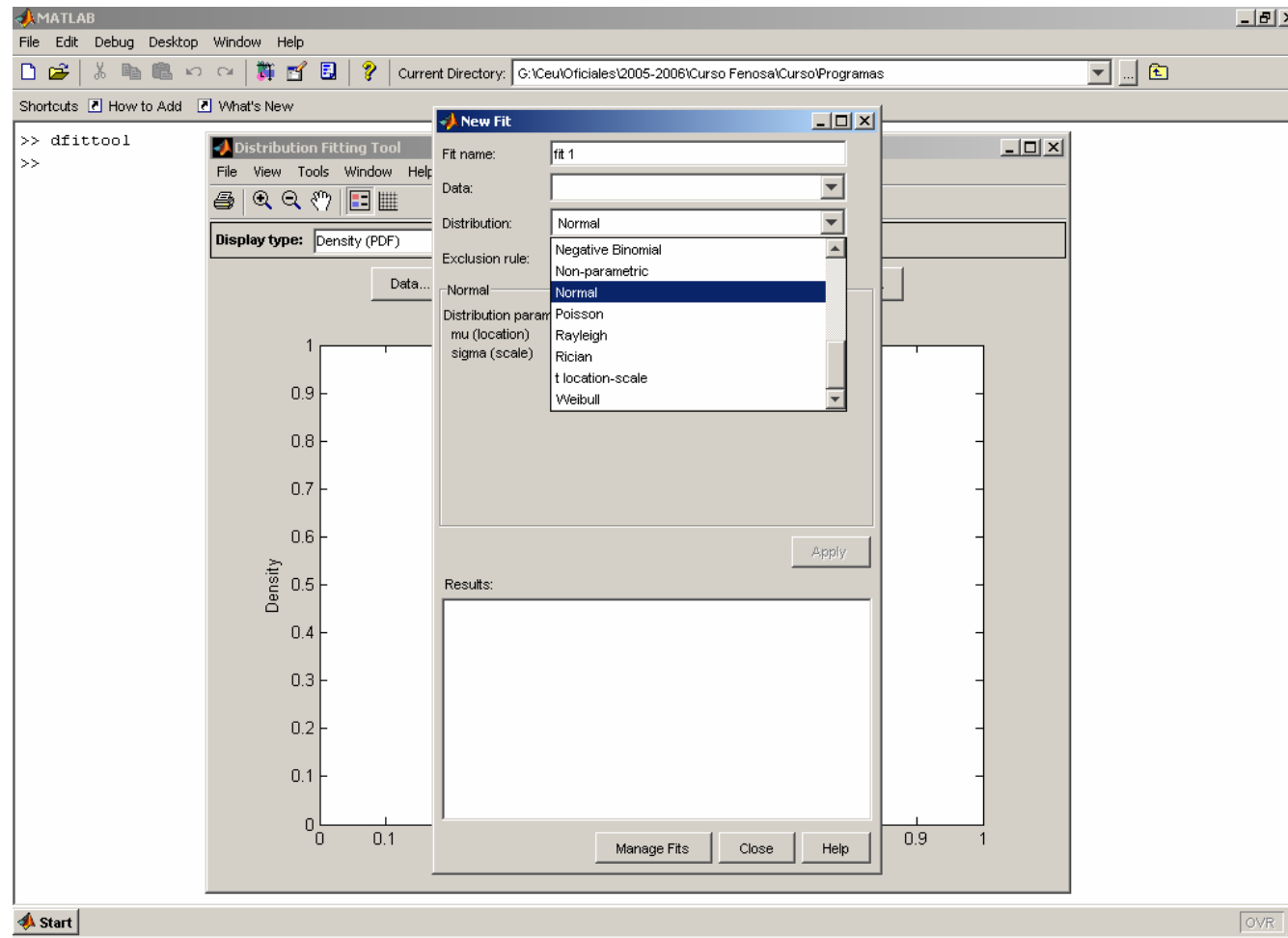
Cronograma del curso

- Día 1: Operaciones con matrices y vectores. Funciones de librería.
- Día 2: Otros tipos de datos en MATLAB. Programación en MATLAB.
- Día 3: Gráficos bidimensionales. Gráficos tridimensionales.
- Día 4: Interfaces de usuario en MATLAB. Generación de programas autónomos
- Día 5: Librerías de interés práctico
- Día 6: Interacción de MATLAB con Office y Visual Basic
- **Día 7: Desarrollo de un proyecto**

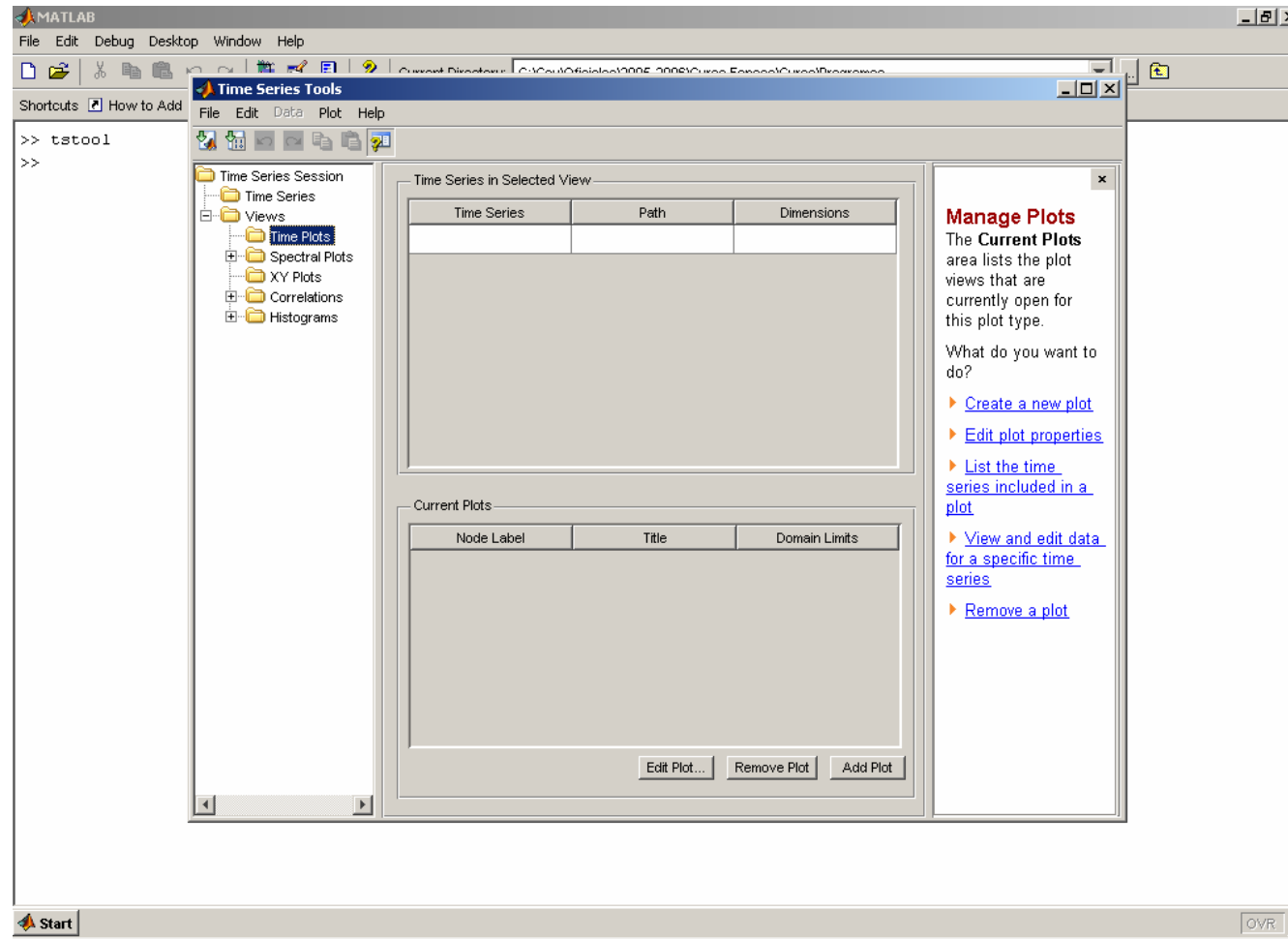
Algo de series temporales: Curve Fitting Tool



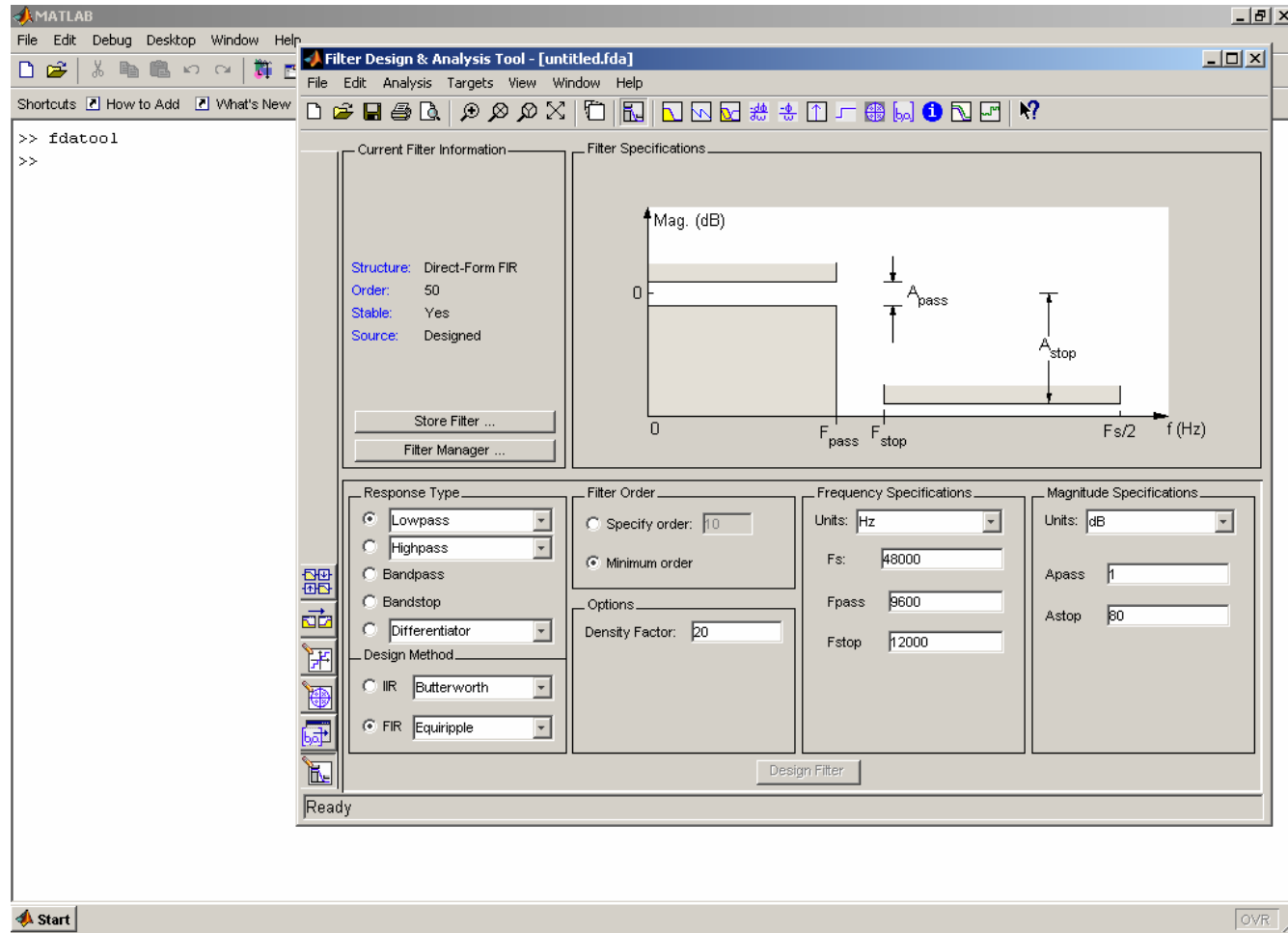
Algo de series temporales: Distribution Fitting Tool



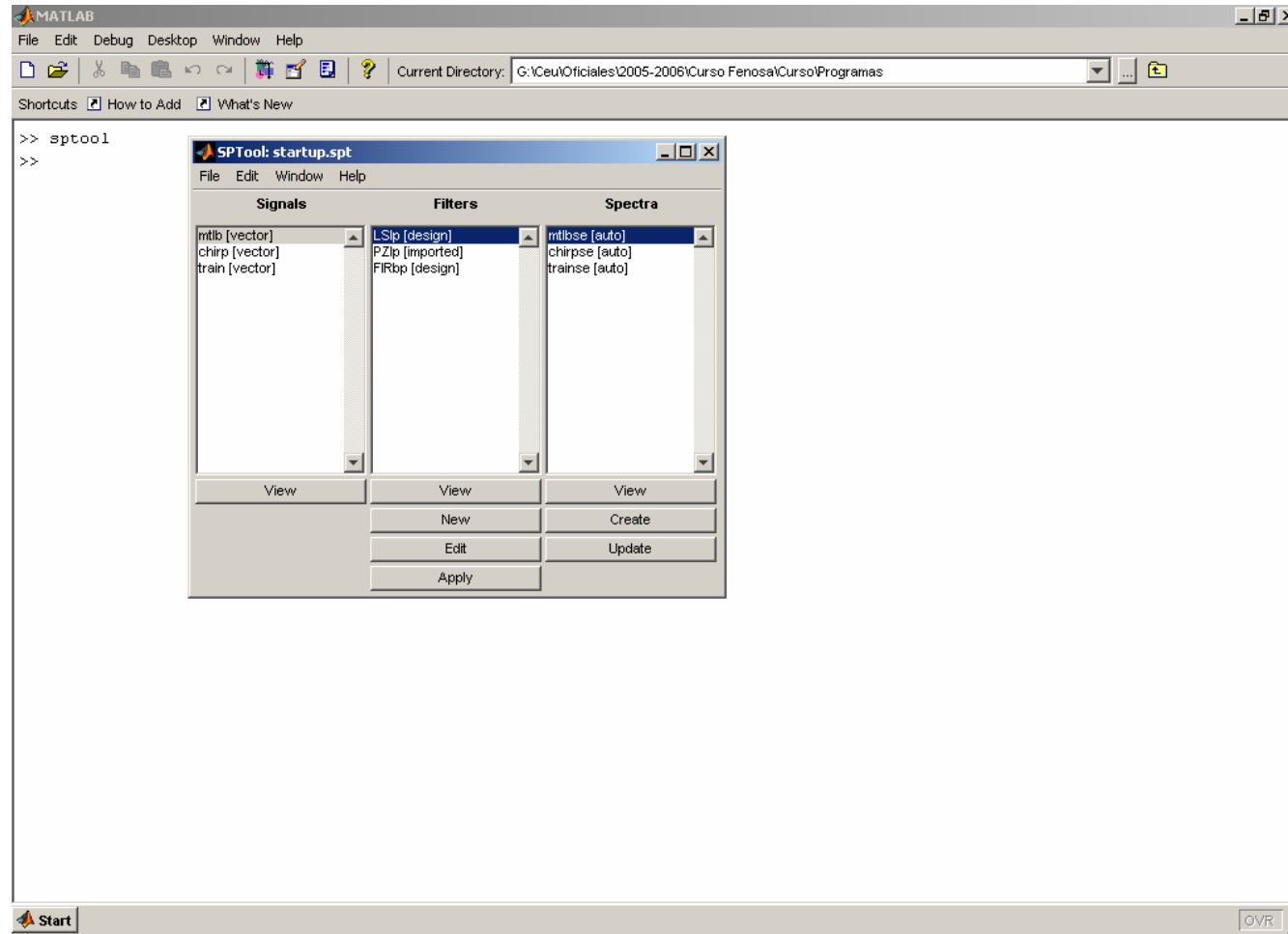
Algo de series temporales: Time Series Tool



Algo de series temporales: Filter Design Tool



Algo de series temporales: Signal Processing Tool



Su nombre aquí

