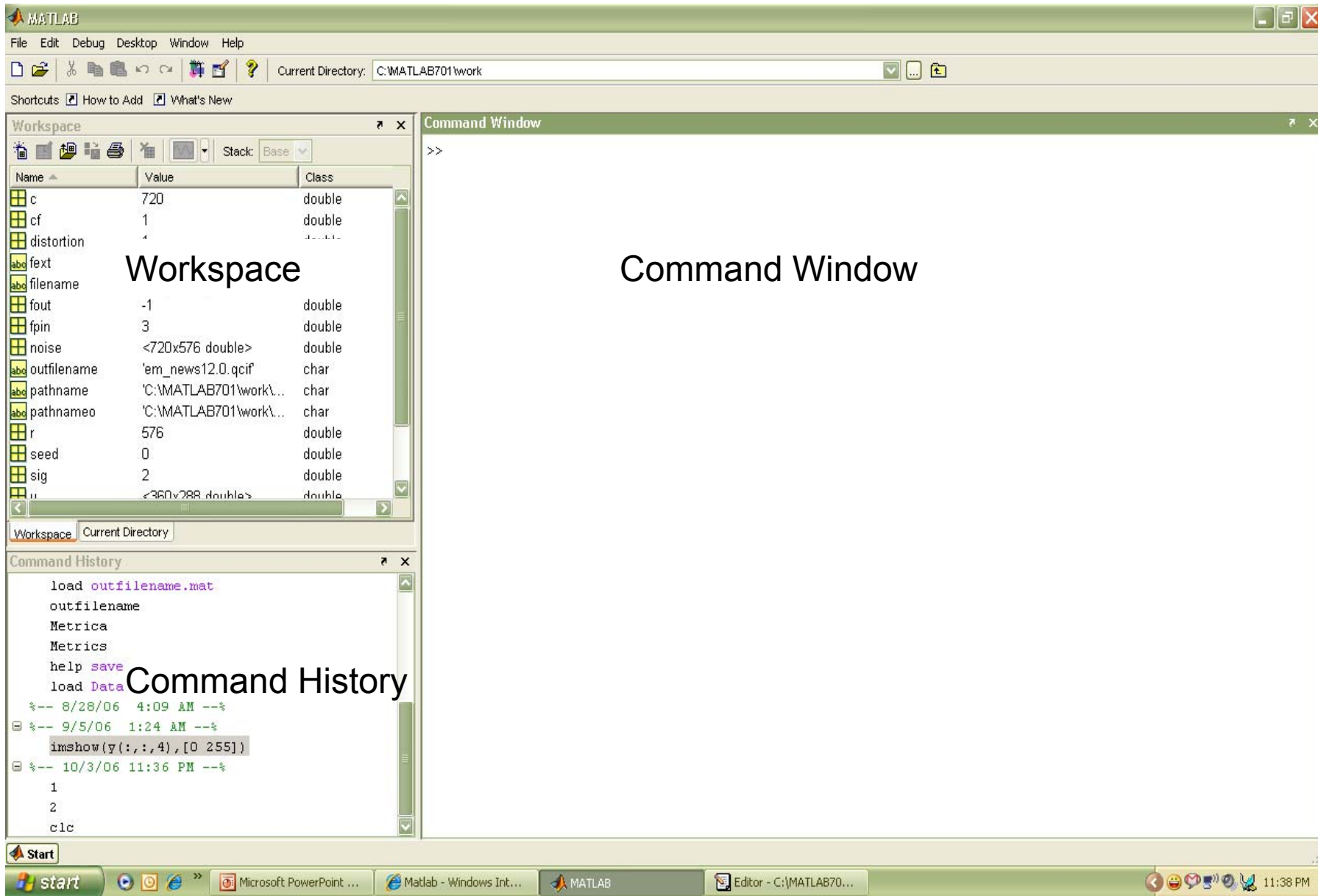


# Matlab Alapok

# Bevezetés

- MatLab : **Matrix Laboratory**
- Numerikus számítások mátrixokkal
  - „*minden mátrix*”
- Miért a Matlab?
  - Felhasználóbarát(GUI)
  - Könnyen használható, gyors interpreter
  - Széles támogatottság (toolbox-ok)
  - Komplex matematikai számítások



# Változók

- Az alap változó mátrix  $\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$
- “a=7;” egy 1x1 mátrix (skalár)
- “a=[1,2,3;4,5,6];” 2x3 mátrix
- “a='array';” 1x5 karakter mátrix
- a(i,j) az a matrix i. sorának és j. oszlopának eleme
- Az “:” operátor
  - >> 1:3                    1 2 3
  - >> 10:-1:5                10 9 8 7 6 5
  - >> a(1:3;j)                a(1,j) a(2,j) a(3,j)

# Mátrixok a Matlab-ban

- Mátrix megadása

$$\begin{array}{cc} 3 & 1 \\ 6 & 4 \end{array}$$

```
>> A = [3 1 ; 6 4]
```

```
>> A = [3, 1 ; 6, 4]
```

```
>> B = [3, 5 ; 0, 2]
```

# Mátrixok

- egy vektor  $x = [1 \ 2 \ 5 \ 1]$

$x =$

1    2    5    1

- egy mátrix  $x = [1 \ 2 \ 3; \ 5 \ 1 \ 4; \ 3 \ 2 \ -1]$

$x =$

1        2        3  
5        1        4  
3        2        -1

- transzponálás  $y = x.'$

$y =$

1  
2  
5  
1

# Mátrixok

- $x(i,j)$  elem

$$y=x(2,3)$$

$$y =$$

4

- teljes sor

$$y=x(3,:)$$

$$y =$$

3

2

-1

- teljes oszlop

$$y=x(:,2)$$

$$y =$$

2

1

2

# Alap matematikai operációk

## **Összeadás:**

$$\gg C = A + B$$

## **Kivonás:**

$$\gg D = A - B$$

## **Szorzás:**

$$\gg E = A * B \text{ (Mátrix szorzat)}$$

$$\gg E = A .* B \text{ (Elemenkénti szorzat)}$$

## **Osztás:**

*Bal- és jobboldali osztás*

$$\gg F = A ./ B \text{ (Elemenkénti osztás)}$$

$$\gg F = A / B \text{ (A * inverz B)}$$

$$\gg F = A . \setminus B \text{ (Elemenkénti osztás)}$$

$$\gg F = A \setminus B \text{ (inverz A * B)}$$



# Alap mátrix generálások

## Null mátrix:

>> Z = ZEROS (r, c)

## Mátrix egyesekkel:

>> O = ONES (r, c)

## Egység mátrix:

>> I = EYE (r, c)

r → sorok

c → oszlopok

zeros, ones, eye → Matlab *függvények*

# Komplex számok

- Valós és imaginárius rész
  - **real** % a komplex szám valós része
  - **imag** % a komplex szám imaginárius része
- Magnitúdó és fázis
  - **abs** % a komplex szám magnitúdója
  - **angle** % a komplex szám fázis

# Grafikonok

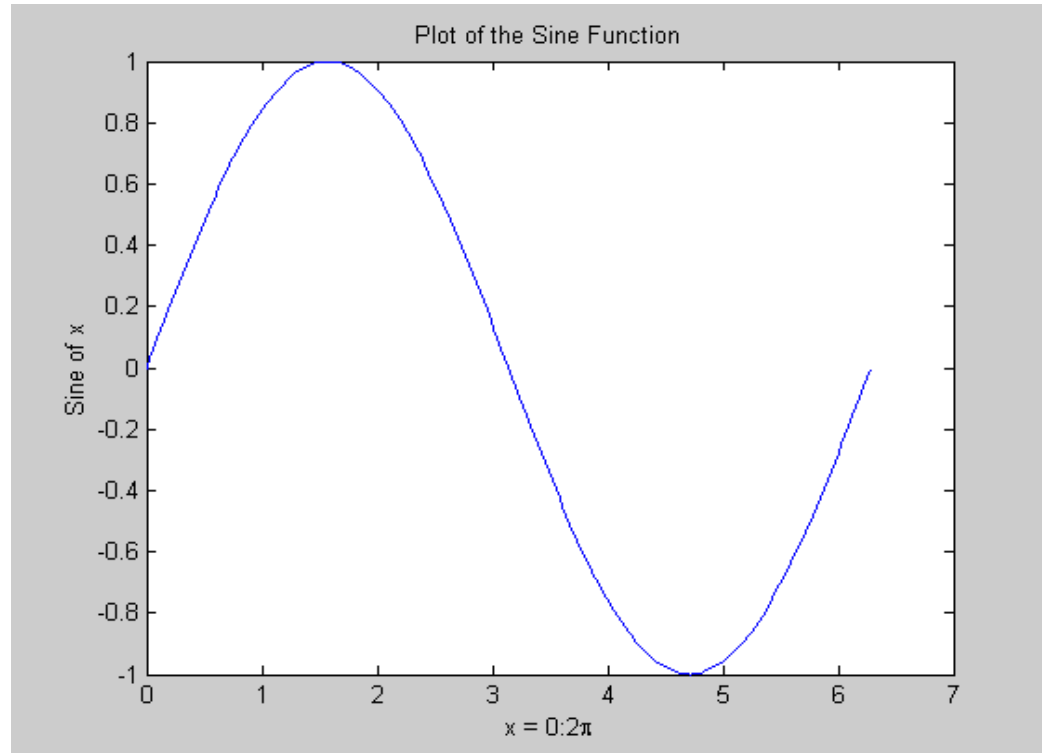
- PLOT(x,y)
  - y kirajzolása x függvényében .
  - XLABEL('label')
  - YLABEL('label')
  - TITLE('title')
- IMAGE(x)
  - kép megjelenítése

## 3D grafikonok:

- MESH
  - 3D háló
- MESHGRID
  - rács kirajzolása
- SURF
  - 3D színes felszín

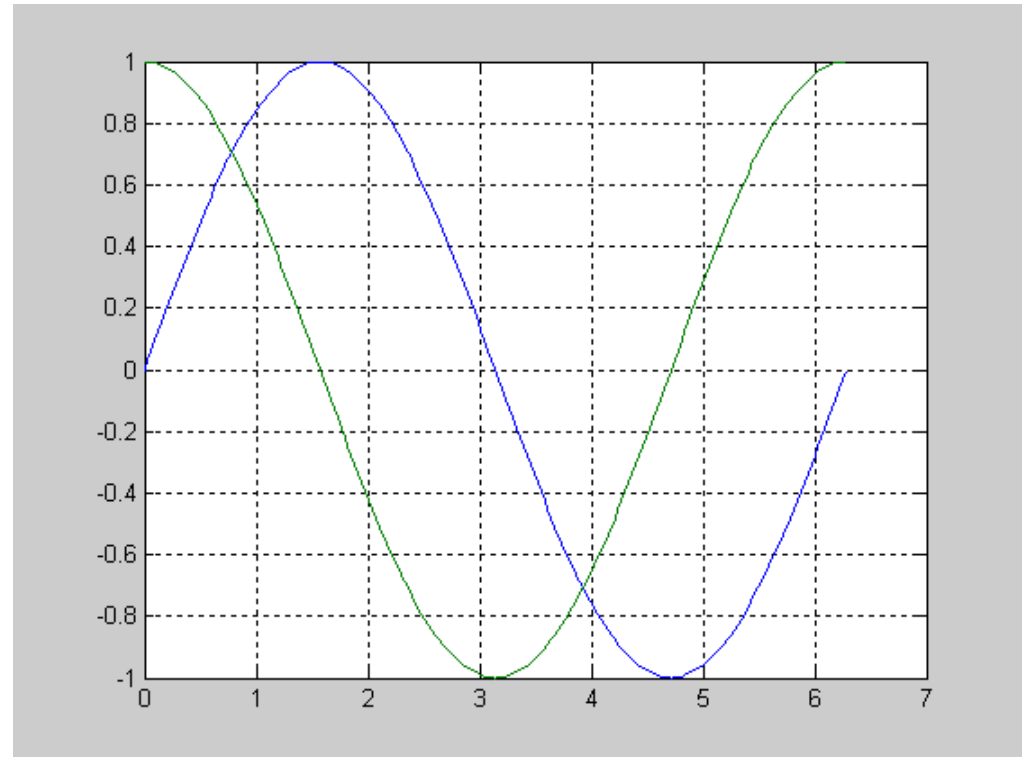
# Grafika példa

```
x = 0:pi/100:2*pi;  
y = sin(x);  
plot(x,y)  
xlabel('x = 0:2\pi')  
ylabel('Sine of x')  
title('Plot of the  
Sine Function')
```



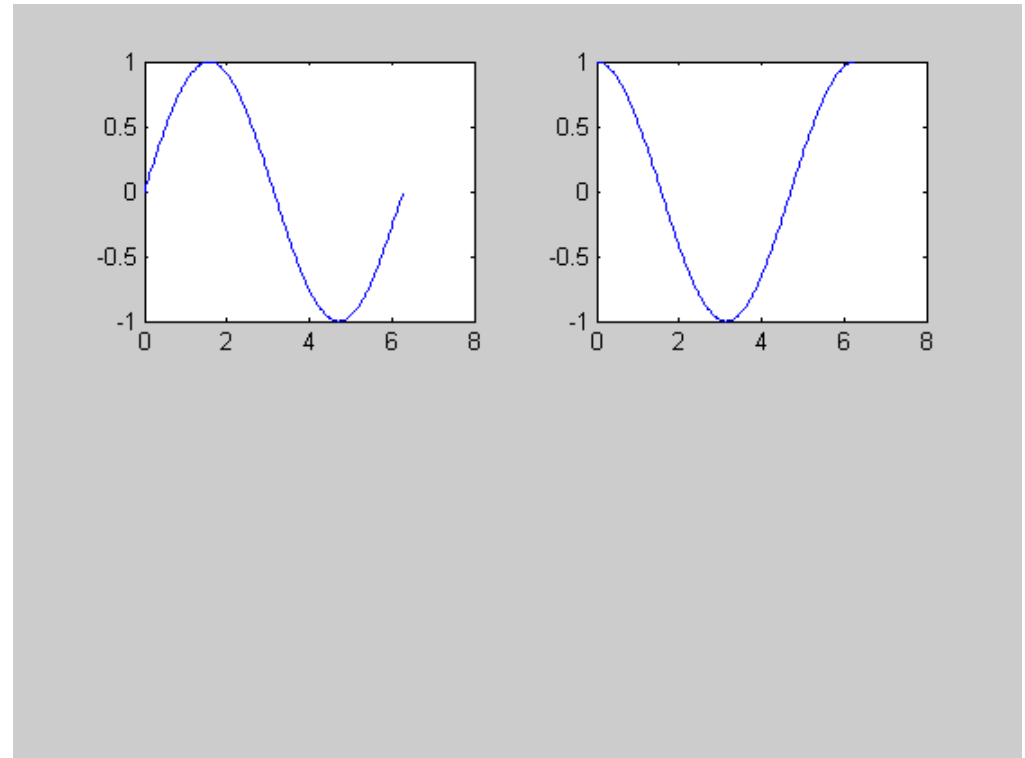
# Több garfikon

```
t = 0:pi/100:2*pi;  
y1=sin(t);  
y2=sin(t+pi/2);  
plot(t,y1,t,y2)  
grid on
```



# Több grafikon

```
t = 0:pi/100:2*pi;  
y1=sin(t);  
y2=sin(t+pi/2);  
subplot(2,2,1)  
plot(t,y1)  
subplot(2,2,2)  
plot(t,y2)
```



# Függvények a Matlab-ban

`function` return values = *function\_name* (arguments)

- `function` kulcsszóval kezdődik
- A file első sora!
- Menteni mint “function\_name.m”
- Pontosvessző (;) a kefejezések végén

- Project Function:

```
function showGrayIm (im)
```

Arguments

Kulcsszó

```
% show 8bits/pixel gray scale image  
% Zhou Wang, Sept. 2006
```

Function\_name

```
colormap(gray(256));  
image(im);  
axis('image', 'off');
```

- » **gray(256)** : Generates a 256 x 3 matrix containing *grayscale* color map
- » **colormap(gray(256))** : Transforms / changes the current image color map to new color map generated by gray(256)

# Feltételes és ciklus utasítások

## if, elseif, else

```
if A > B
    s='A greater';
elseif A < B
    s='A less'
elseif A == B
    s='A,B equal'
else
    s='Unexpected situation'
end
```

## switch

```
switch (int(A-B))
    case -1
        s='A less';
    case 0
        s='A,B equal'
    case 1
        s='A greater'
    otherwise
        s='Unexpected situation'
end
```

## for

```
for i=1:5
    sum=sum+i;
end
```



# Feltételes és ciklus utasítások

## Óvakodj a *for*-tól!

Kód 1:

```
count_1 = 0;
for i = 1 : length(A)
    if A(i) == 1
        count_1 = count_1 + 1
    end
end
```

Kód 2:

```
ct_1 = sum(A==1)
```

Az egyes és kettes kód eredménye azonos!

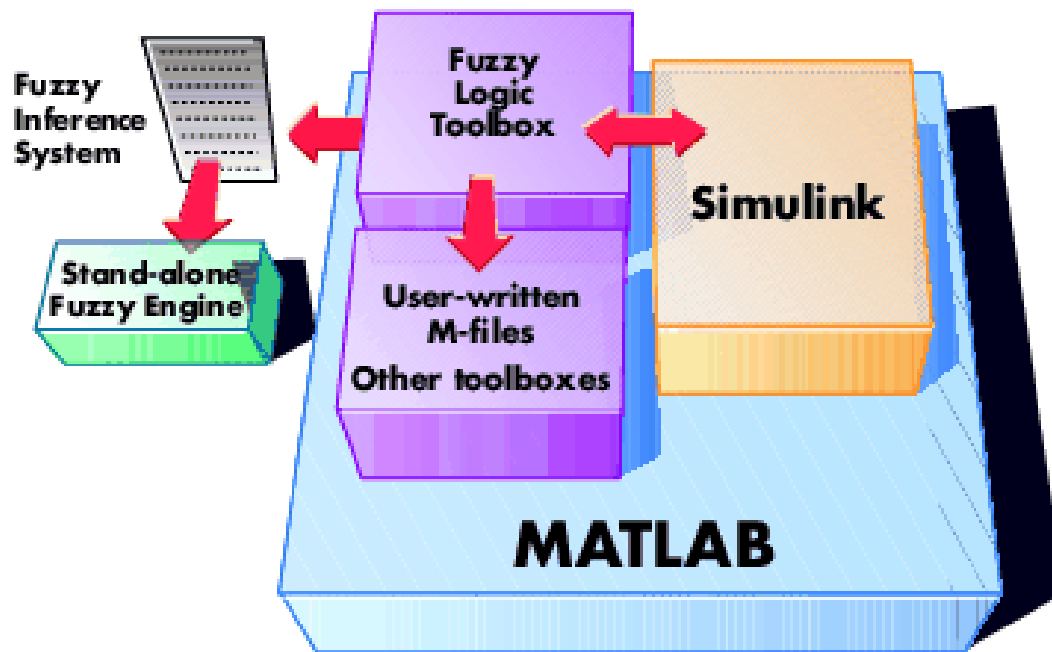
# Input és Output (I/O)

- **Írás a képernyőre**
  - `disp(['Variable x has value: ' num2str(x)]);`
  - `x` (with no “;” will print `x`)
- **Írás file-ba**
  - Speciális Matlab függvények (pl. `csvwrite('matrix.txt',M)`)
  - File megnyitása: `fid=fopen('file.txt',w);`
  - Írás a file-ba: `fprintf(fid,'%f\n',x);` (c style formatting)
  - File lezárása: `fclose(fid);` (must do this to have it write)
- **Olvasás file-ból**
  - Speciális Matlab függvények (pl. `M = csvread('filename')`)
  - C stílusban: `a = fscanf(fid,'%g %g',[m n]);` visszatérés egy `m x n` mátrix-al.

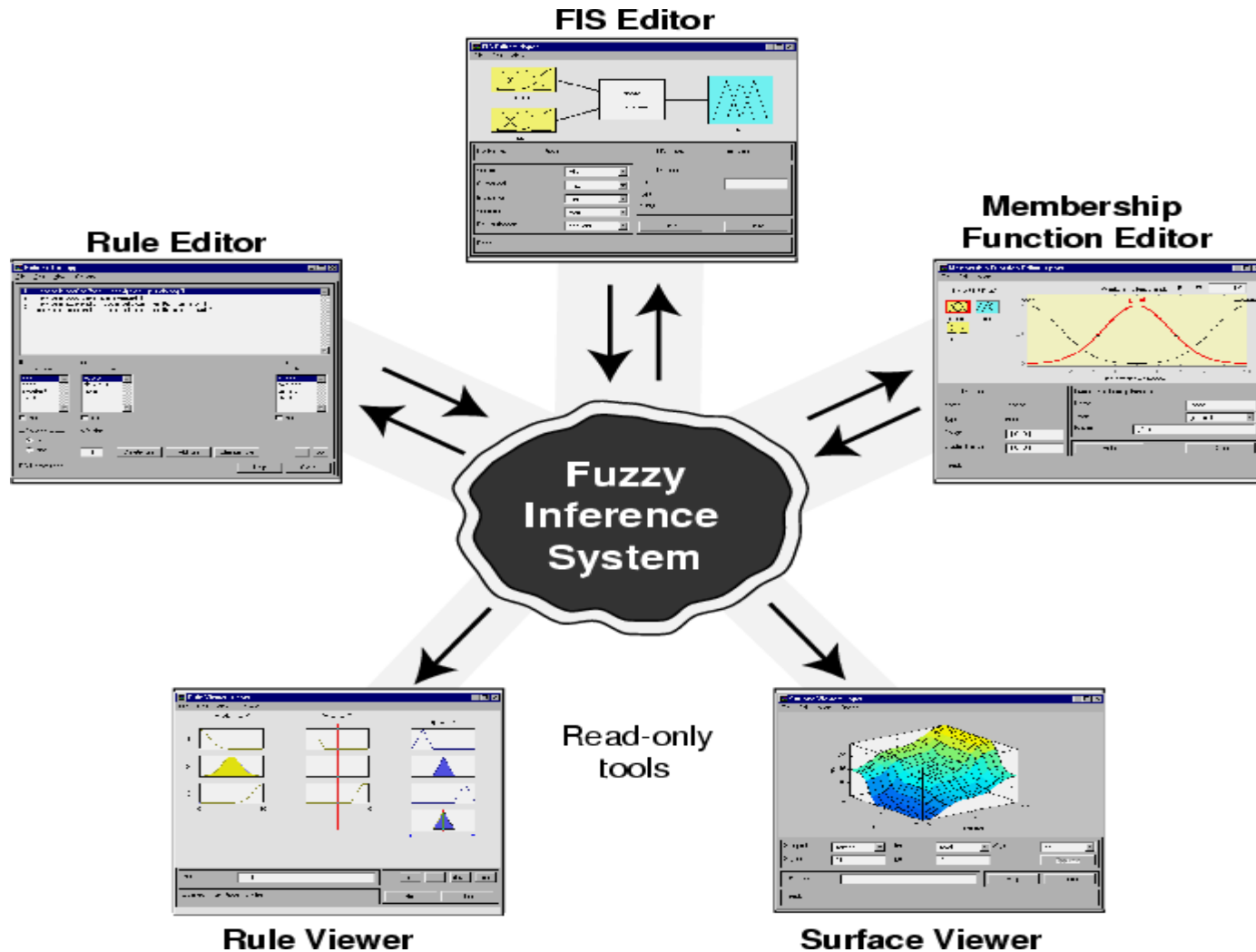
# Help!!

- Help Browser használata(.html, .pdf)
  - getstart.pdf, graphg.pdf, using\_ml.pdf
- Parancs sorban
  - `help`
  - `help function`, e.g. `help plot`
- Demok
  - `type demos`
  - `type help demos`

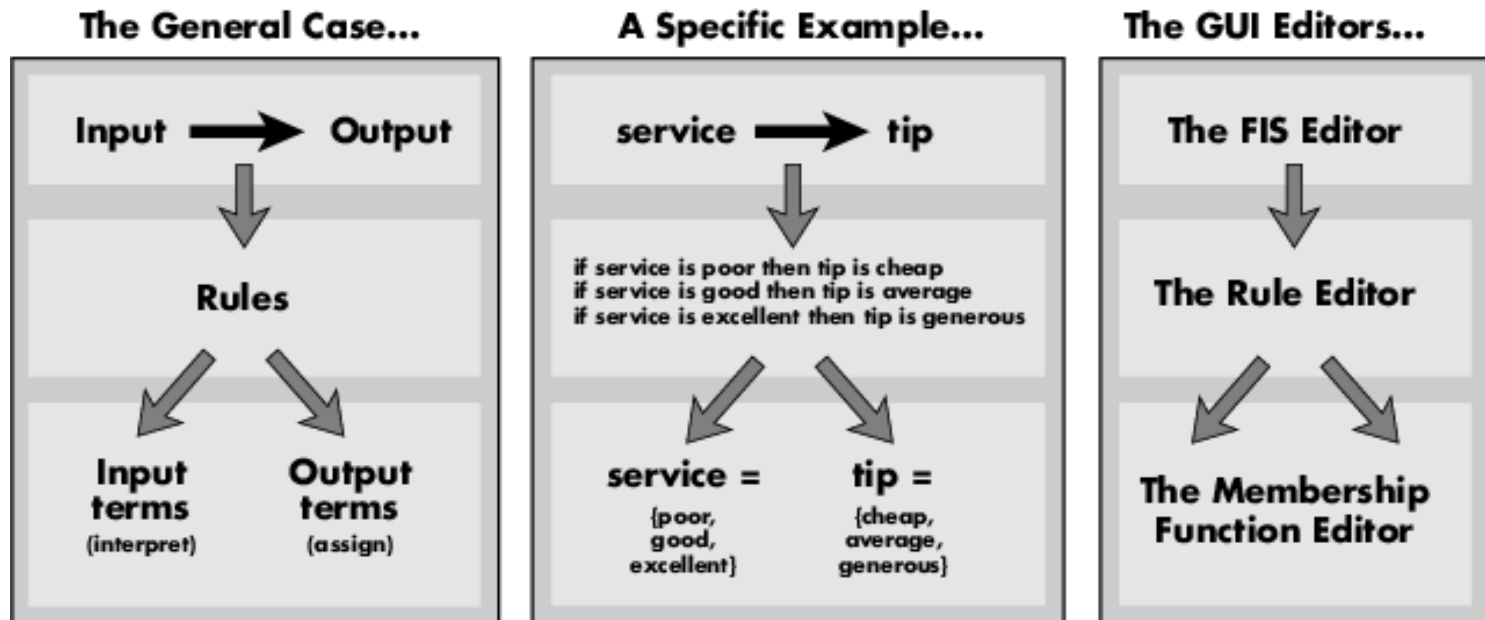
# Matlab Fuzzy Logic Toolbox



# Matlab Fuzzy Logic Toolbox



# Matlab FIS építés koncepciója



# A FIS editor

Double-click on an input variable icon to open the Membership Function Editor.

Double-click on the system diagram to open the Rule Editor.

Double-click on the icon for the output variable icon, to open the Membership Function Editor.

These menu items allow you to save, open, or edit a fuzzy system using any of the five basic GUI tools.

The name of the system is displayed here. It can be changed using one of the Save as.. menu options.

These pop-up menus are used to adjust the fuzzy inference functions, such as the defuzzification method.

This status line describes the most recent operation.

This edit field is used to name and edit the names of the input and output variables.

FIS Editor: tipper

File Edit View

service

tipper  
mamdani

tip

food

FIS Name: tipper F S Type: mamdani

And method: min

Or method: max

Implication: min

Aggregation: max

Defuzzification: centroid

Current Variable

Name: \_\_\_\_\_

Type: \_\_\_\_\_

Range: \_\_\_\_\_

Help Close

System "tipper": 2 inputs, 1 output, and 3 rules.

# Az MF editor

This is the "Variable Palette" area. Click on a variable here to make it current and edit its membership functions.

These menu items allow you to save, open, or edit a fuzzy system using any of the five basic GUI tools.

This graph field displays all the membership functions of the current variable.

Click on a line to select it and you can change any of its attributes, including name, type and numerical parameters. Drag your mouse to move or change the shape of a selected membership function.

These text fields display the name and type of the current variable.

This edit field lets you set the range of the current variable.

This edit field lets you set the display range of the current plot.

This status line describes the most recent operation.

This edit field lets you change the numerical parameters for the current membership function.

This pop-up menu lets you change the type of the current membership function.

This edit field lets you change the name of the current membership function.



# A szabálybázis szerkesztő

The menu items allow you to save, open, or edit a fuzzy system using any of the five basic GUI tools.

Input or output selection menus.

The rules are entered automatically using the GUI tools.

The screenshot shows the 'Rule Editor: tipper' window. The title bar includes 'File Edit View Options'. The main text area contains three rules:  
1. If (service is poor) or (food is rancid) then (tip is cheap) (1)  
2. If (service is good) then (tip is average) (1)  
3. If (service is excellent) or (food is delicious) then (tip is generous) (1)  
Below the text area, there are three columns of selection menus. The first column is labeled 'If service is' and contains 'poor', 'good', 'excellent', and 'none'. The second column is labeled 'or food is' and contains 'rancid', 'delicious', and 'none'. The third column is labeled 'Then tip is' and contains 'cheap', 'average', 'generous', and 'none'. There are checkboxes for 'not' under each column. Below the menus are radio buttons for 'or' (selected) and 'and', and a 'Weight:' field with the value '1'. At the bottom, there are buttons for 'Delete rule', 'Add rule', and 'Change rule', along with navigation arrows. The status bar at the bottom shows 'FIS Name: tipper', 'Help', and 'Close' buttons.

Link input statements in rules.

This status line describes the most recent operation.

Negate input or output statements in rules.

Create or edit rules with the GUI buttons and choices from the input or output selection menus.

The Help button gives some information about how the Rule Editor works, and the Close button closes the window.

# A szabálybázis néző

The menu items allow you to save, open, or edit a fuzzy system using any of the five basic GUI tools.

Each column of plots (yellow) shows how the input variable is used in the rules. The input values are shown here at the top.

This column of plots (blue) shows how the output variable is used in the rules.

Each row of plots represents one rule (here there are 3). Click on a rule to display it in the status bar.

Slide this line to change your input values, and generate a new output response.

This edit field allows you to set the input explicitly.

This status line describes the most recent operation.

Shift the plots left, right, up, or down with these buttons.

This line provides a defuzzified value.

This plot shows how the output of each rule is combined to make an aggregate output and then defuzzified.

The screenshot shows the 'Rule Viewer: tipper' window with the following components:

- Menu Bar:** File, Edit, View, Options
- Input Labels:** service = 5, food = 5, tip = 15
- Plot Grid:**
  - Column 1 (service): Three yellow plots for rules 1, 2, and 3. Rule 1 shows a decreasing curve, rule 2 shows a bell-shaped curve, and rule 3 shows an increasing curve. A red vertical line is at 5.
  - Column 2 (food): Three yellow plots for rules 1, 2, and 3. Rule 1 shows a decreasing curve, rule 2 is empty, and rule 3 shows an increasing curve. A red vertical line is at 5.
  - Column 3 (tip): Three blue plots for rules 1, 2, and 3. Rule 1 shows a bell-shaped curve, rule 2 shows a bell-shaped curve, and rule 3 shows a bell-shaped curve. A red vertical line is at 15.
- Input Field:** Input: [5 5]
- Plot Points:** Plot points: [1 1]
- Move Buttons:** left, right, down, up
- Status Bar:** Opened system tipper, 3 rules
- Buttons:** Help, Close

# A következtetési felület néző

The menu items allow you to save, open, or edit a fuzzy system using any of the five basic GUI tools.

Use the mouse to rotate the axes.

This plot shows the output surface for any output of the system versus any one or two inputs to the system.

These pop-up menus let you specify the one or two displayed input variables.

These edit fields let you determine how densely to grid the input space.

This edit field lets you set the input explicitly for inputs not specified in the surface plot.

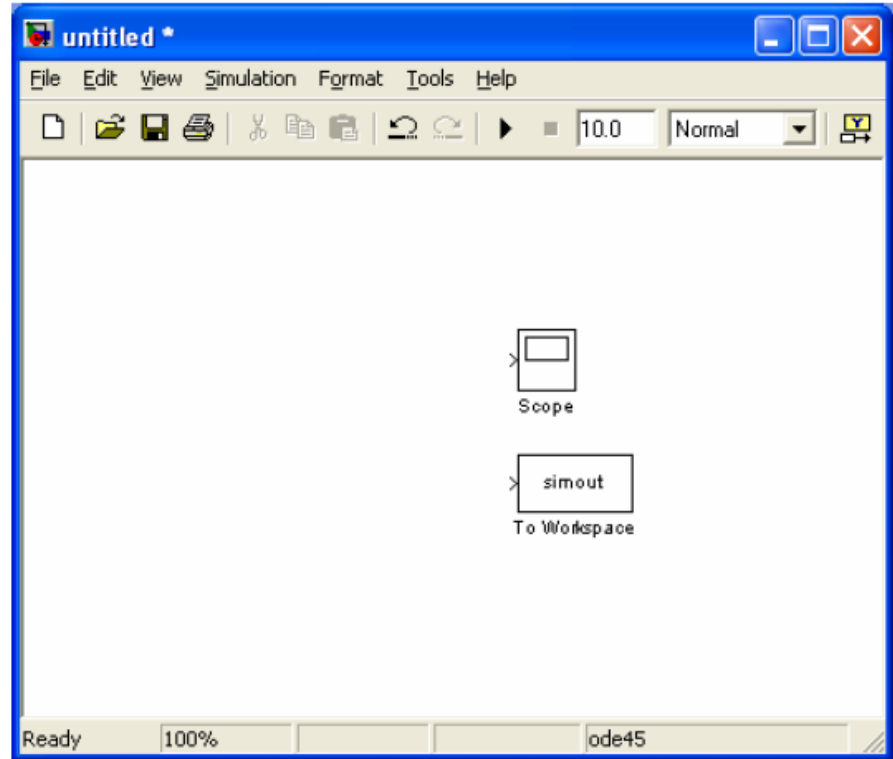
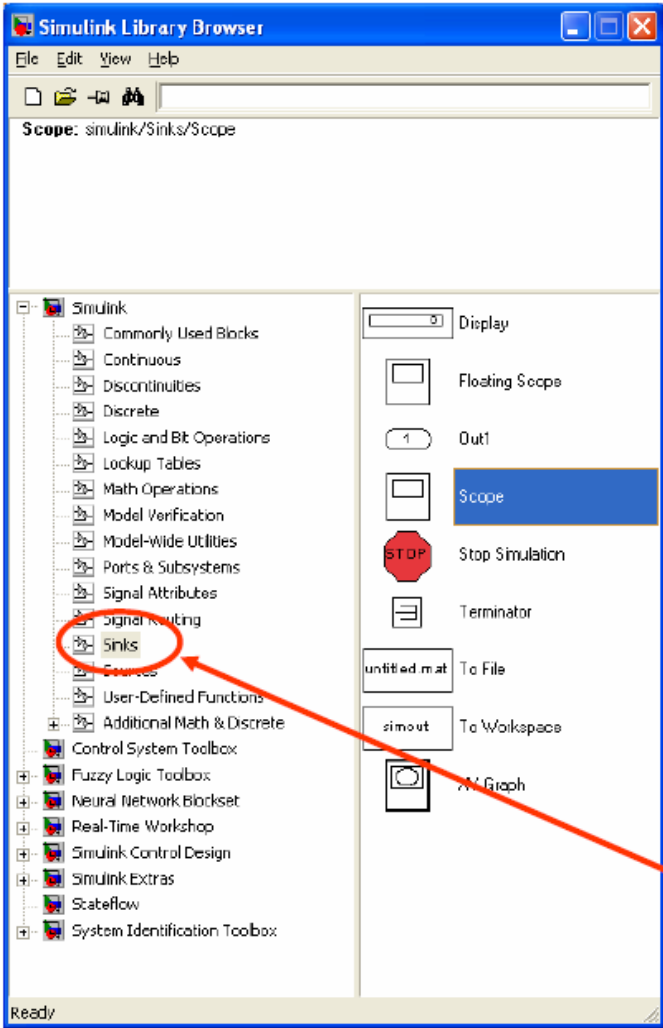
This status line describes the most recent operation.

The Help button gives some information about how the Surface Viewer works, and the Close button closes the window.

Click Evaluate when you're ready to calculate and plot.

This pop-up menu lets you specify the displayed output variable.

The screenshot shows a window titled "Surface Viewer: tipper" with a menu bar (File, Edit, View, Options) and a 3D surface plot. The plot has axes labeled "food", "service", and "tip". Below the plot are control panels for input variables (X, Y, Z), grid size, and evaluation buttons (Evaluate, Help, Close). A status bar at the bottom shows "Ready".



See  
Drag any block you want to use  
into the model.