

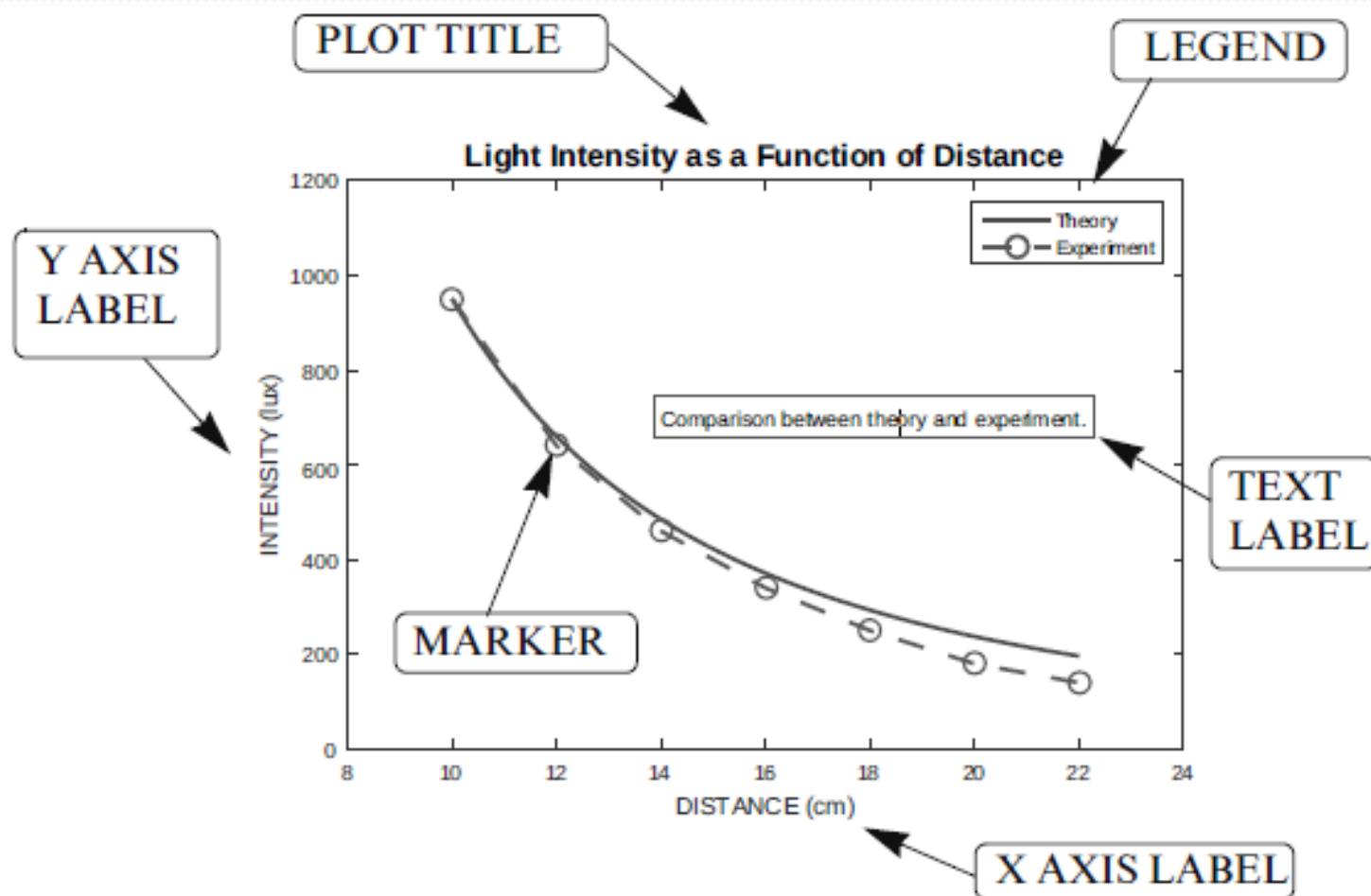
فصل ۵

ترسیمهای دو بعدی

این فصل به ترسیم نمودارهای دو بعدی میپردازد.
گزینه های مختلفی برای این نمودارها وجود دارد:

- محورهای خطی، نیمه لگاریتمی، لگاریتمی
- نوع، رنگ و ضخامت خط
- انواع مختلفی از نشانگرهای نقاط داده
- خطوط شبکه، عناوین، توضیحات و راهنمای نمودار
- ترسیمهای فرعی
- نمودارهای میله ای، پله ای، قطبی

قسمت های مهم یک ترسیم دو بعدی



از دستور `plot` برای ایجاد ترسیمهای ساده دو بعدی استفاده می شود. ساده ترین شکل آن `plot(y)` است

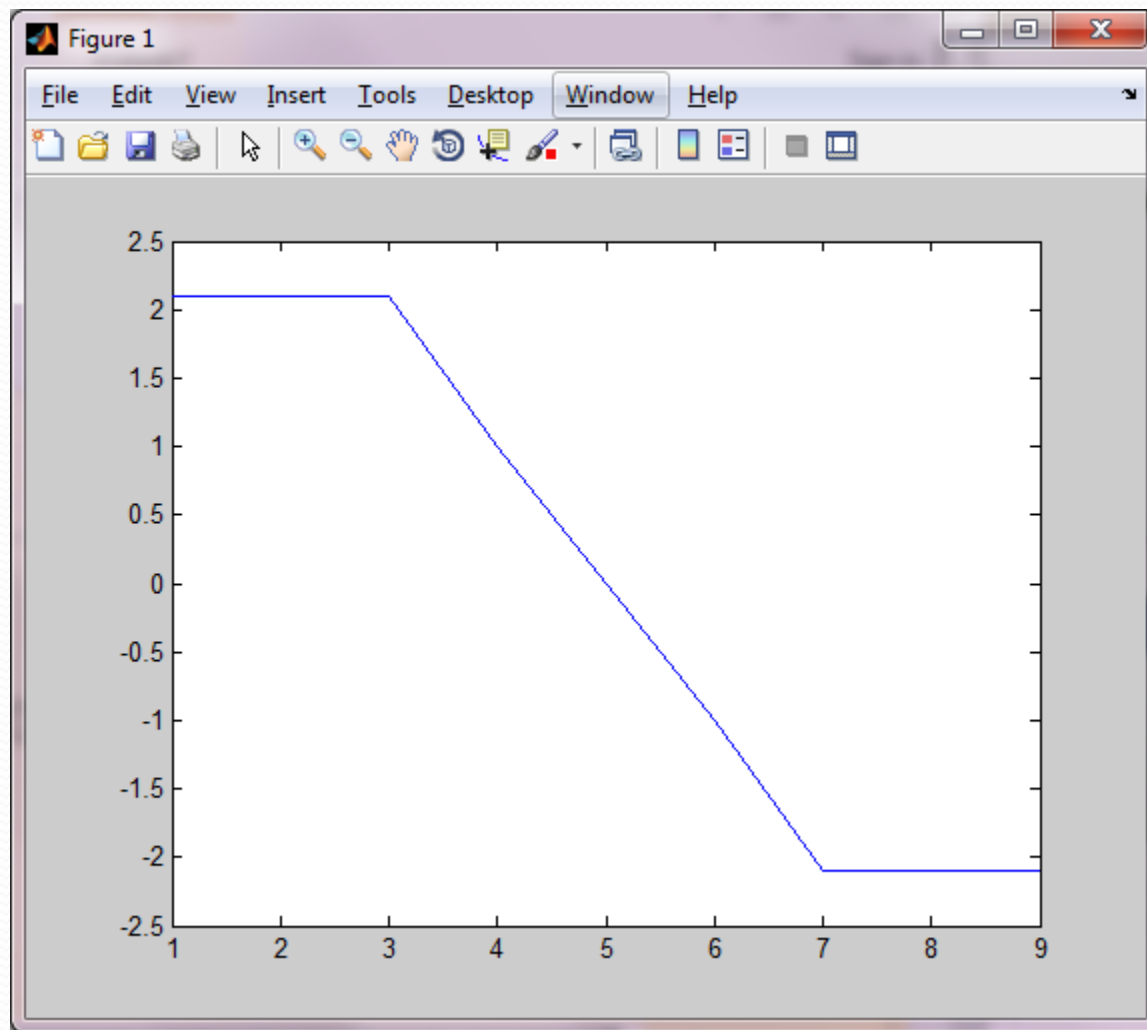
- بردار y را روی محور عمودی، و اعداد ۱ تا N را روی محور افقی رسم میکند. ($N = \text{تعداد نقاط } y$)
- اگر یک پنجره شکل (Figure) باز باشد، ترسیم را در آن انجام میدهد. در غیر اینصورت یک پنجره شکل جدید باز میکند و نمودار را در آن نشان میدهد.

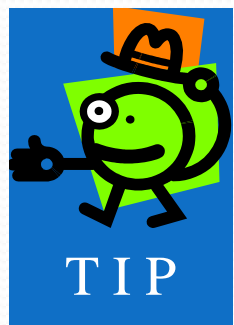
پیش فرض های `plot(y)`

- هر دو محور خطی هستند
- MATLAB محدوده اعداد محورها را به گونه ای مناسب انتخاب میکند
- نقاط با خطوط مستقیم به هم وصل میشوند
- از نشانگر نقاط استفاده نمیشود
- نقاط و خطوط به رنگ آبی هستند

مثال

```
>> y = [ 2.1  2.1  2.1  1  0  -1  -2.1  -2.1  -2.1 ];  
>> plot( y )
```





اگر بعد از اجرای دستور plot، پنجره شکلی نشان داده نشد، آن پنجره در حالت کوچک شده است. روی آیکن آن در نوار وظایف ویندوز کلیک کنید تا نشان داده شود



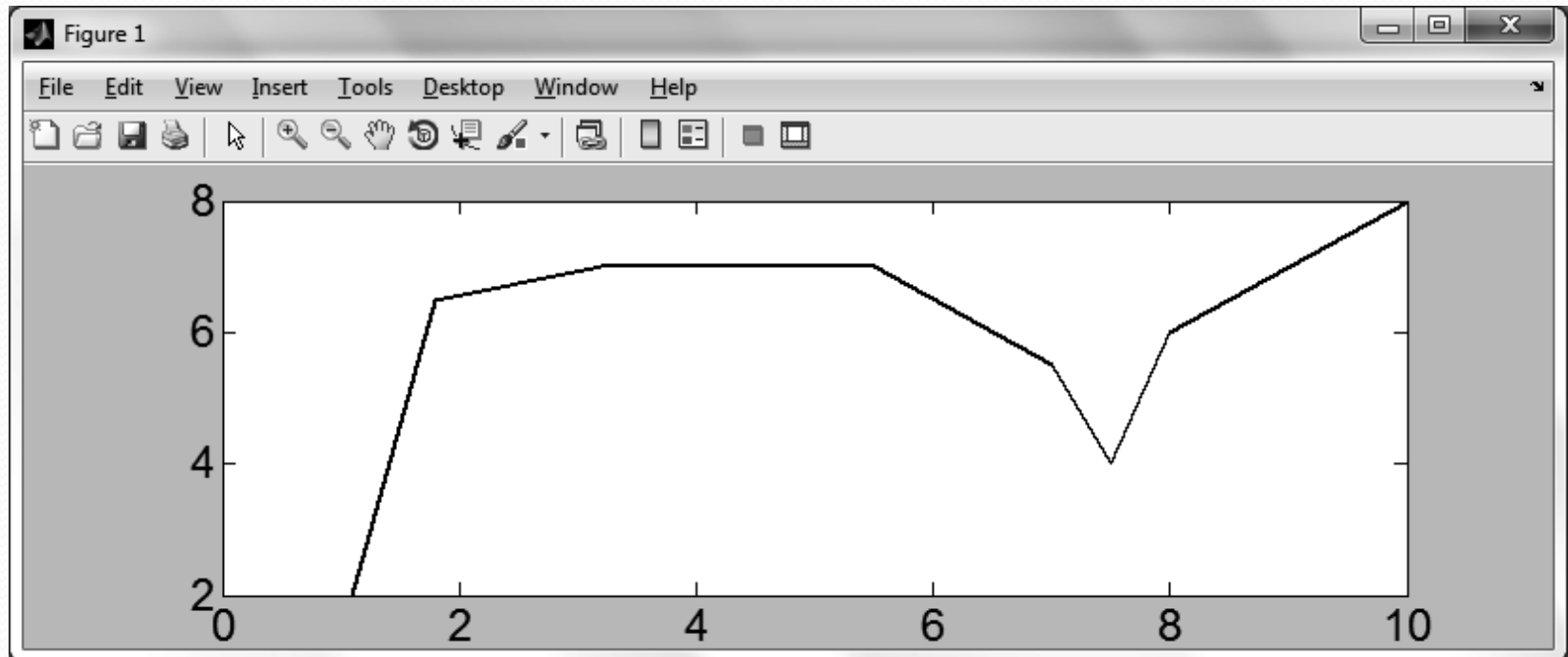
دومین حالت ساده عبارت است از

`plot(x, y)`

- x و y بردارهایی با اندازه یکسان (تعداد درایه های برابر)، ولی به صورت دلخواه سطری یا ستونی هستند
- مقادیر x روی محور افقی و مقادیر y روی محور عمودی نشان داده میشوند
- پیش فرض ها مانند حالت `plot(x)` هستند

مثال

```
>> x=[1.1  1.8  3.2  5.5  7  7.5  8  10];  
>> y=[2  6.5  7  7  5.5  4  6  8];  
>> plot(x,y)
```



برای استفاده از گزینه هایی غیر از پیش فرضها،

```
plot(x,y, 'line specifiers', 'PropertyName', PropertyValue)
```

Vector

Vector

(Optional) Specifiers that define the type and color of the line and markers.

(Optional) Properties with values that can be used to specify the line width, and marker's size and edge, and fill colors.

تعیین کننده های ویژگی خطوط (*Line*)
specifier (نوع و رنگ خطوط و نوع نشانگرها)
 مشخص می کنند

Line Style	Specifier
solid (default)	-
dashed	--

Line Style	Specifier
dotted	:
dash-dot	-.

Line Color	Specifier
red	r
green	g
blue	b
cyan	c

Line Color	Specifier
magenta	m
yellow	y
black	k
white	w

انواع نشانگرهای نقاط

Marker Type	Specifier		Marker Type	Specifier
plus sign	+		square	s
circle	o		diamond	d
asterisk	*		five-pointed star	p
point	.		six-pointed star	h
cross	x		triangle (pointed left)	<
triangle (pointed up)	^		triangle (pointed right)	>
triangle (pointed down)	v			

Notes about using the specifiers:

- The specifiers are typed inside the `plot` command as strings.
- Within the string the specifiers can be typed in any order.
- The specifiers are optional. This means that none, one, two, or all the three can be included in a command.

Some examples:

<code>plot(x,y)</code>	A blue solid line connects the points with no markers (default).
<code>plot(x,y,'r')</code>	A red solid line connects the points.
<code>plot(x,y,'--y')</code>	A yellow dashed line connects the points.
<code>plot(x,y,'*')</code>	The points are marked with <code>*</code> (no line between the points).
<code>plot(x,y,'g:d')</code>	A green dotted line connects the points that are marked with diamond markers.

نام ویژگی و مقدار ویژگی:

- در دستور `plot`، نام ویژگی را بین دو آپوستروف، سپس کاما، و سپس مقدار ویژگی را بنویسید

Property Name	Description	Possible Property Values
<code>LineWidth</code> (or <code>linewidth</code>)	Specifies the width of the line.	A number in units of points (default 0.5).
<code>MarkerSize</code> (or <code>markersize</code>)	Specifies the size of the marker.	A number in units of points.
<code>MarkerEdgeColor</code> (or <code>markeredgecolor</code>)	Specifies the color of the marker, or the color of the edge line for filled markers.	Color specifiers from the table above, typed as a string.
<code>MarkerFaceColor</code> (or <code>markerfacecolor</code>)	Specifies the color of the filling for filled markers.	Color specifiers from the table above, typed as a string.

For example, the command:

```
plot(x,y, '-mo', 'LineWidth', 2, 'markersize', 12,
      'MarkerEdgeColor', 'g', 'markerfacecolor', 'y')
```

creates a plot that connects the points with a magenta solid line and circles as markers at the points. The line width is two points and the size of the circle markers is 12 points. The markers have a green edge line and yellow filling.

یک نکته در مورد تعیین کننده های ویژگی های خطوط:

The three line specifiers, which indicate the style and color of the line, and the type of the marker can also be assigned with a `PropertyName` argument followed by a `PropertyValue` argument. The Property Names for the line specifiers are:

Specifier	Property Name	Possible property values
Line style	<code>linestyle</code> (or <code>LineStyle</code>)	Line style specifier from the table above, typed as a string.
Line color	<code>color</code> (or <code>Color</code>)	Color specifier from the table above, typed as a string.
Marker	<code>marker</code> (or <code>Marker</code>)	Marker specifier from the table above, typed as a string.

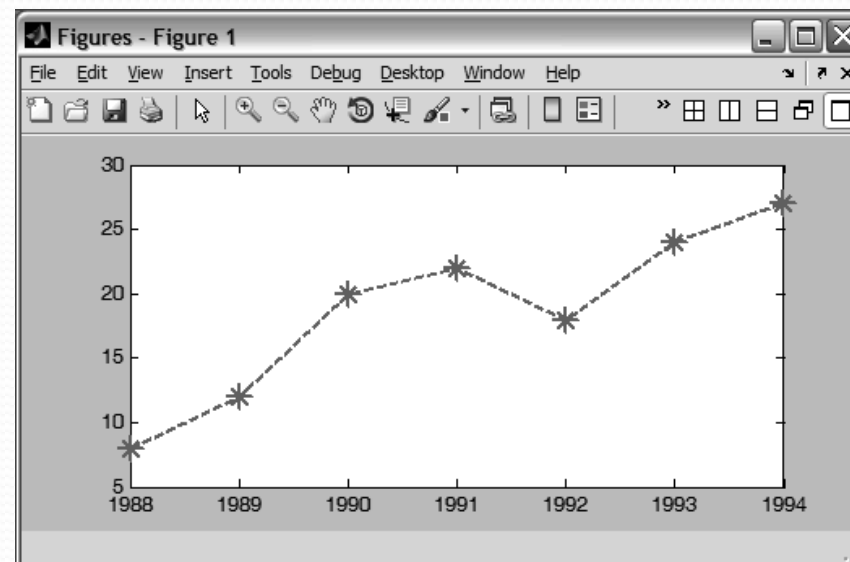
5.1.1 Plot of Given Data

YEAR	1988	1989	1990	1991	1992	1993	1994
SALES (millions)	8	12	20	22	18	24	27

```
>> yr=[1988:1:1994];  
>> sle=[8 12 20 22 18 24 27];  
>> plot(yr,sle,'--r*','linewidth',2,'markersize',12)  
>>
```

Line Specifiers:
dashed red line and
asterisk marker.

Property Name and Property Value:
the line width is 2 points and the
marker size is 12 points.



یک راه برای ترسیم تابعی از یک متغیر
مستقل:

1. یک بردار حاوی مقادیر متغیر مستقل
ایجاد کنید
2. یک بردار از مقدار تابع به ازای هر درایه
بردار قبلی ایجاد کنید
3. با استفاده از `plot(x, y)` ترسیم را
انجام دهید

5.1.2 Plot of a Function

```
% A script file that creates a plot of
```

```
% the function:  $3.5 \cdot (-0.5^x) \cdot \cos(6x)$ 
```

```
x = [-2:0.01:4];
```

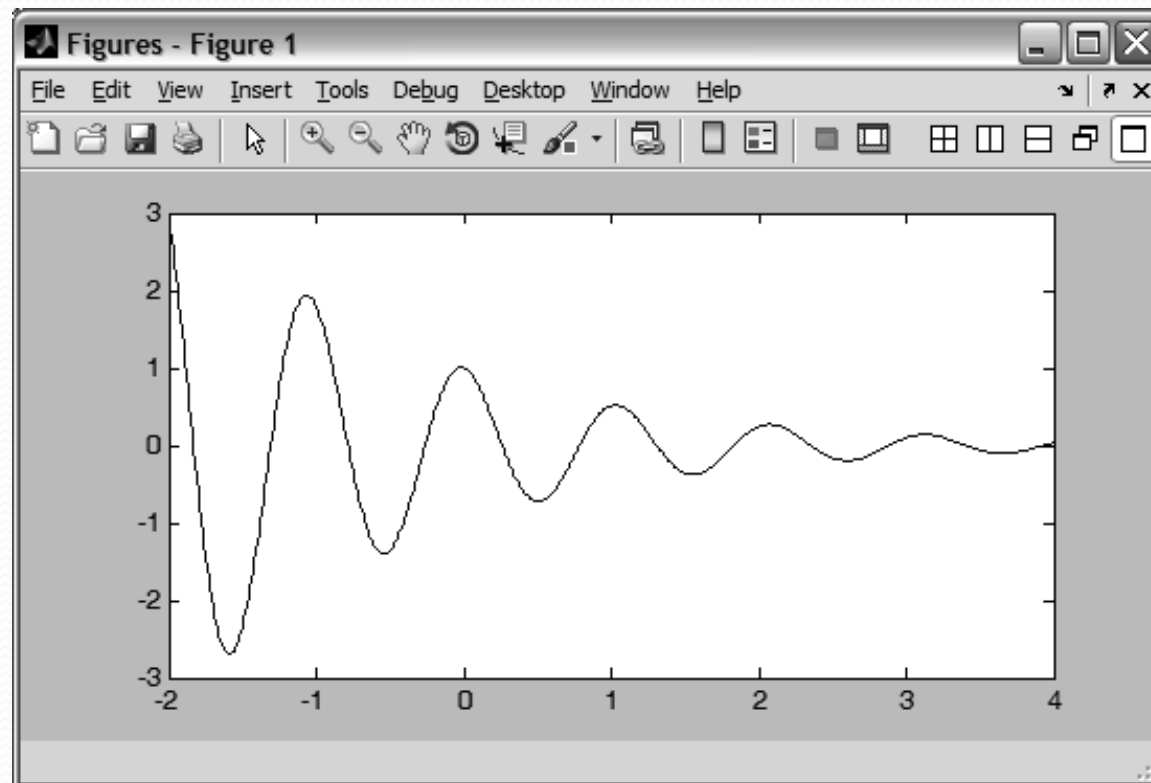
Create vector x with the domain of the function.

```
y = 3.5.^(-0.5*x) .* cos(6*x);
```

Create vector y with the function value at each x.

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

Plot y as a function of x.



برای کپی کردن کل پنجره شکل در برنامه
دیگری مانند Word یا PowerPoint

1. روی پنجره شکل کلیک کنید تا پنجره فعال
شود

2. ALT+PRNTSCRN را بزنید

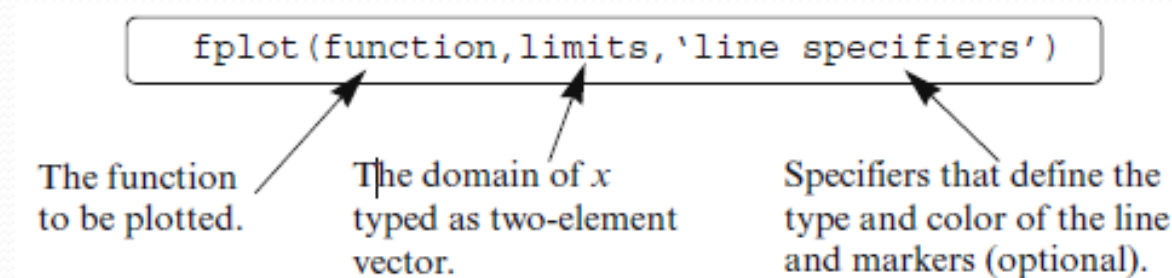
3. در برنامه دیگر Paste کنید

برای کپی کردن تنها ناحیه ترسیم پنجره
شکل در برنامه دیگری مانند Word یا
PowerPoint

1. در پنجره شکل، Edit و سپس
Copy Figure را انتخاب کنید

2. در برنامه دیگر Paste کنید

fplot تابع $y=f(x)$ را در محدوده داده شده ترسیم میکند



- تابع را به صورت ناشناس معرفی کنید (قسمت 7.8 کتاب را ببینید)

• برای تابع $f(x)$ ، صورت ناشناس به این صورت است:
 $f(x) = 8x^2 + 5 \cos x$ ، مثلاً برای $@(x) f(x)$
 صورت ناشناس: $@(x) 8 * x.^2 + 5 * \cos(x)$

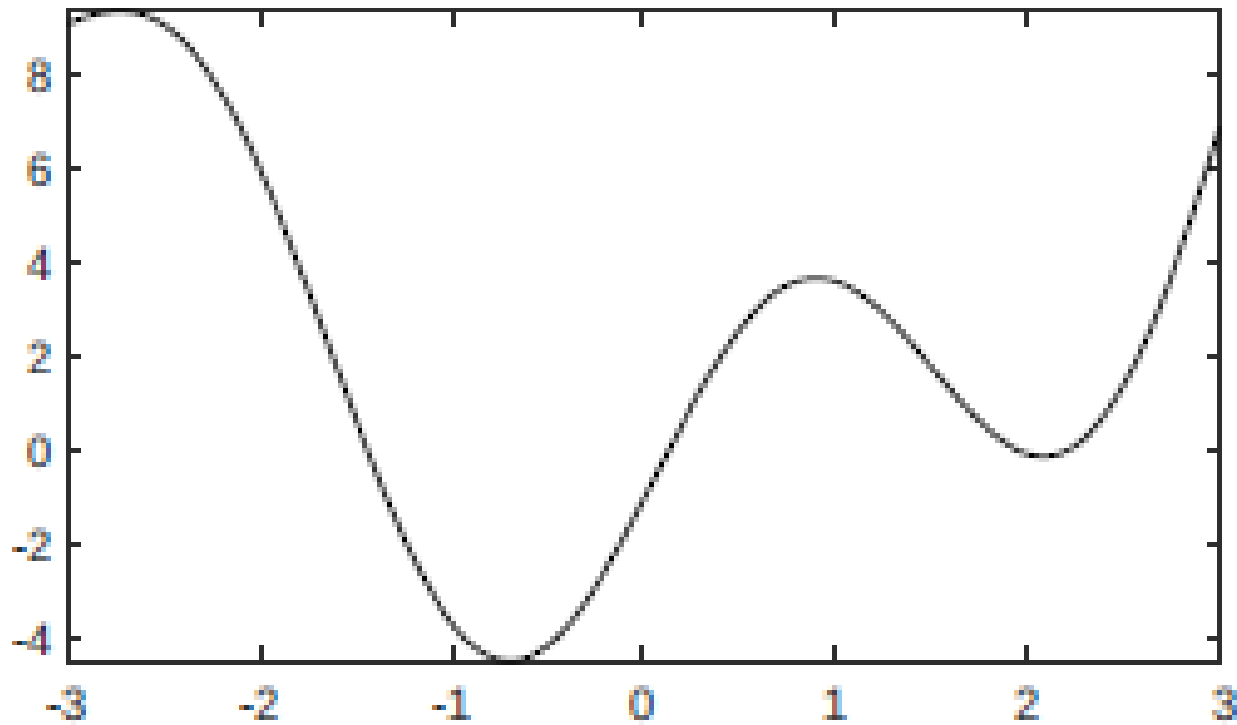
- تابع میتواند یکی از توابع MATLAB یا تابعی که خودتان نوشته اید باشد

`fplot('function',limits,'line specifiers')`

- برای مشخص کردن متغیر مستقل در تابع از هر حرفی میتوان استفاده کرد، مثلاً
 - $8*x^2+5*\cos(x)$ یا $8*t^2+5*\cos(t)$
- `limits` - بردار دو یا چهار درایه ای که محدوده محورهای نمودار را مشخص میکند
 - `[xmin xmax]` -x محدوده محور
 - `[xmin xmax ymin ymax]` - دو محور محدوده هر دو محور
- `line specifiers` - plot مشابه

مثال

```
>> fplot(@(x) x.^2+4*sin(2*x)-1,[-3,3])
```



اغلب می خواهیم نمودار بیش از یک گروه داده را در
یک ترسیم داشته باشیم

MATLAB میتواند این کار را به سه صورت انجام
دهد

دو نمودار اضافه را در یک ترسیم به این صورت رسم کنید (مثال برای سه نمودار)

```
plot(x,y,u,v,t,h)
```

- y را در برابر x ، v را در برابر u و h را در برابر t رسم می کند

- هر جفت از بردارها باید هم اندازه باشند

- اندازه یک جفت میتواند با جفت دیگر نابرابر باشد

- از تعیین کننده های ویژگی خطوط، با نوشتن سه مورد آنها میتوان استفاده کرد. مانند:

```
plot(x,y,'-b', u,v,'--r', 't,h','g:')
```

```
x = [-2:0.01:4];
```

Create vector x with the domain of the function.

```
y = 3*x.^3 - 26*x + 6;
```

Create vector y with the function value at each x.

```
yd = 9*x.^2 - 26;
```

Create vector yd with values of the first derivative.

```
ydd = 18*x;
```

Create vector ydd with values of the second derivative.

```
plot(x,y,'-b',x,yd,'--r',x,ydd,':k')
```

Create three graphs, y vs. x, yd vs. x, and ydd vs. x, in the same figure.

The plot that is created is shown in Figure 5-7.

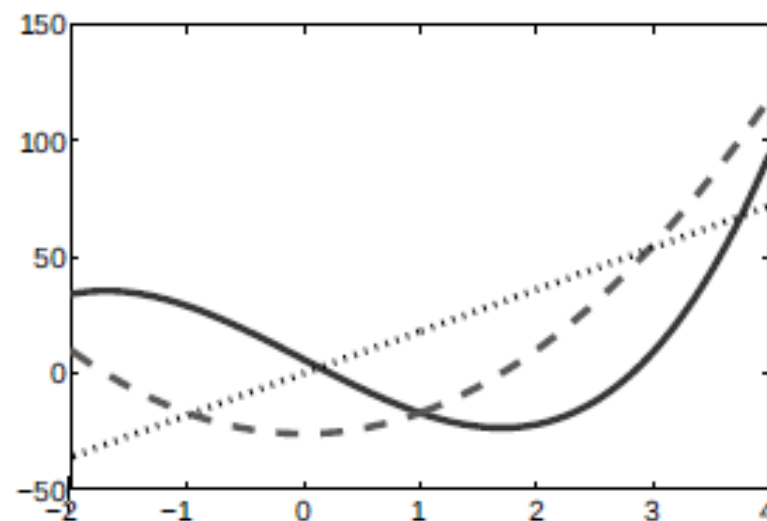


Figure 5-7: A plot of the function $y = 3x^3 - 26x + 10$ and its first and second derivatives.

در حالت کلی، هر بار که دستور `plot` را اجرا کنید ترسیم قبلی را پاک کرده و مورد جدیدی را رسم میکند. برای تغییر این وضعیت:

- اولین نمودار را با `plot` رسم کنید
- دستور `hold on` را اجرا کنید
- `Plot` را برای هر کدام از نمودارهای باقی مانده اجرا کنید
- دستور `hold off` را اجرا کنید

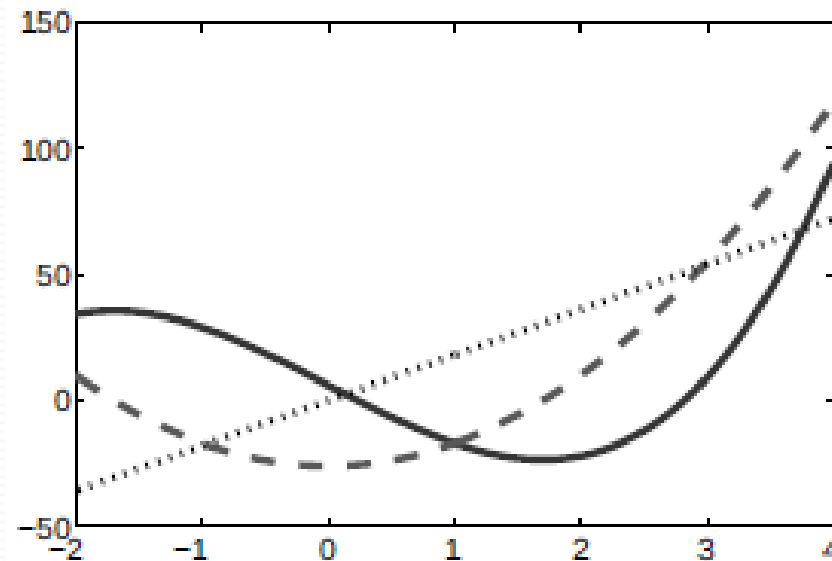
نمودارهایی که پس از `hold on` رسم شوند به ترسیمها اضافه خواهند شد. نمودارهایی که پس از `hold off` رسم شوند ترسیمها را پاک میکنند.

5.3.2 Using the hold on and hold off Commands

```
x=[-2:0.01:4];  
y=3*x.^3-26*x+6;  
yd=9*x.^2-26;  
ydd=18*x;  
plot(x,y, '-b')  
hold on  
plot(x,yd, '--r')  
plot(x,ydd, ':k')  
hold off
```

The first graph is created.

Two more graphs are added to the figure.



دستور line نمودارهای اضافه را به یک ترسیم
موجود اضافه می کند

```
line(x,y,'PropertyName','PropertyValue')
```

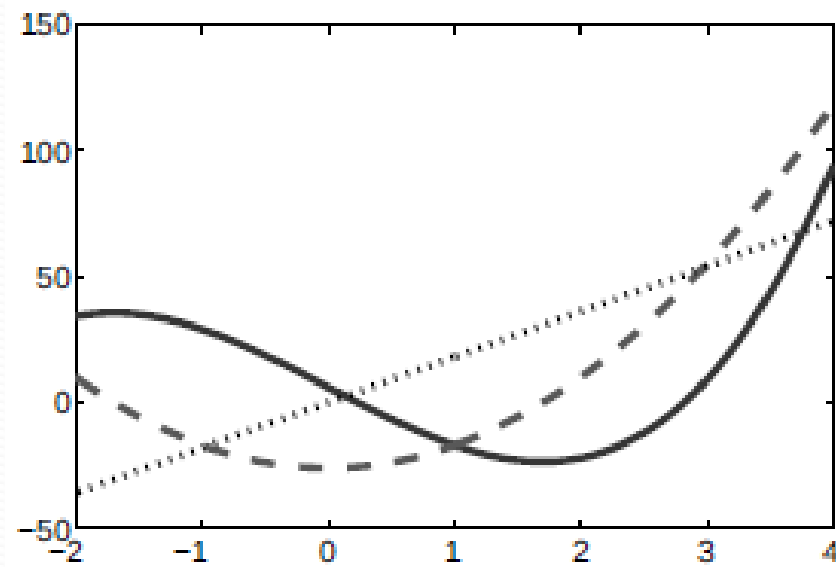
مثال

```
line(x,y,'linestyle','--','color','r','marker','o')
```

نموداری با خطوط به صورت خط چین قرمز و
نشانگرهای دایره ای به نمودار موجود اضافه میکند

5.3.3 Using the line Command

```
x=[-2:0.01:4];  
y=3*x.^3-26*x+6;  
yd=9*x.^2-26;  
ydd=18*x;  
plot(x,y,'LineStyle','-','color','b')  
line(x,yd,'LineStyle','--','color','r')  
line(x,ydd,'linestyle',':','color','k')
```



چگونه میتوان یک ترسیم را با اضافه کردن این موارد
شکیل تر کرد

- برچسب محورها
- عنوان
- راهنما
- متن
- شبکه
- محدود دلفخواه محورها

`plot` یا `fplot` نمودار اصلی را ایجاد میکنند.
پس از اجرای این دستورات، میتوان از این موارد
استفاده کرد

- `xlabel('some text')` اضافه کردن
برچسب زیر محور افقی

- مثال: `xlabel('Time (sec)')`

- `ylabel('some text')` اضافه کردن
برچسب سمت چپ محور عمودی

- مثال: `ylabel('Current (mA)')`

- `title('Some text')` اضافه کردن عنوان بالای نمودار

- مثال: `title('Diode Current')`

- `text(x, y, 'Some text')` اضافه کردن متن به نمودار طوری که اولین کاراکتر در (x, y) باشد
- مثال:

`text(x, y, 'Peak 3.5 sec after first')`

- `gtext('Some text')` پنجره ترسیم باز میشود، کاربر روی نمودار در جایی که میخواهد متن نمایش داده شود کلیک میکند

```
legend('text1','text2',... ,pos)
```

اضافه کردن راهنما

- برای هر نمودار (نقاط داده) خط کوتاهی به همان شکل نمودار نمایش میدهد و متن مشخص شده را اضافه میکند
- اولین متن مربوط به اولین نمودار رسم شده است، دومین متن مربوط به دومین نمودار، و به همین ترتیب
- بیشتر برای نمودارهای چندتایی مناسب است
- مقادیر `pos` در کتاب از MATLAB 7.0 منسوخ شده اند. راهنمای مربوطه را در این خصوص ببینید

قالب بندی متن در دستورات `text xlabel, ylabel, title` و `legend` :

متن نمایش داده شده با دستورات بالا را میتوان قالب بندی کرد

- نوع فونت، اندازه، رنگ نوشته، رنگ پس زمینه، زیر یا بالا نویس بودن و نوع نگارش (bold, italic, etc.) را میتوان تنظیم کرد
- میتوان از حروف یونانی استفاده کرد
- برای قالب بندی میتوان از تنظیم کننده های متن استفاده کرد یا نام و مقدار ویژگیها را در دستور وارد کرد

- تنظیم کننده های متن داخل متن مورد نظر قرار میگیرند و ظاهر متن را تحت تأثیر قرار میدهند
- تمام متن قرار گرفته پس از تنظیم کننده عوض میشود
- برای تنظیم قسمت خاصی از متن از آکلاد باز ({}), تنظیم کننده، متن مورد نظر، آکلاد بسته ({} استفاده کنید

عنوانهای نمونه

```
title('\it What You Should Never See')
```

این را ایجاد میکند

What You Should Never See

```
title('What You Should{\it Never}  
See')
```

این را ایجاد میکند

What You Should Never See

عنوانهای نمونه

```
title('\fontname{Old English Text MT}...  
My Name is Robin Hood')
```

این را ایجاد میکند

My Name is Robin Hood

```
title(...  
'{\fontname{Old English Text MT}Robin  
Hood}...  
was here')
```

این را ایجاد میکند

Robin Hood was here

برخی تنظیم کننده های رایج

- `\bf` – **bold face**
- `\it` – *italic*
- `\rm` – **فونت معمولی**
- `\fontname{fontname}` – **اسم فونت**
- `\fontsize{fontsize}` – **اندازه فونت**

زیر نویس و بالا نویس:

برای تبدیل یک کاراکتر به

- زیرنویس - قبل از آن یک underscore (_) بگذارید
- بالانویس - قبل از آن یک علامت توان (^) بگذارید
- برای چند کاراکتر، مثل حالت قبل ولی کاراکترها را در آکلا بگذارید

• `xlabel('H_2O')` مینویسد H_2O

• `ylabel('e^{-k*\sin(x)}')` مینویسد $e^{-k*\sin(x)}$

مروف یونانی:

برای نوشتن یک حرف یونانی، از یک `backslash` و نام حرف به انگلیسی استفاده کنید

- اگر نام با مروف کوچک نوشته شود حرف یونانی کوچک نوشته خواهد شد
- اگر حرف اول نام بزرگ نوشته شود حرف یونانی بزرگ نوشته خواهد شد

```
ylabel('Standard deviation (\sigma) of resistance in M\Omega')
```

مینویسد

Standard deviation (σ) of resistance in $M\Omega$

برخی حروف یونانی

Characters in the string	Greek Letter
<code>\alpha</code>	α
<code>\beta</code>	β
<code>\gamma</code>	γ
<code>\theta</code>	θ
<code>\pi</code>	π
<code>\sigma</code>	σ

Characters in the string	Greek Letter
<code>\Phi</code>	Φ
<code>\Delta</code>	Δ
<code>\Gamma</code>	Γ
<code>\Lambda</code>	Λ
<code>\Omega</code>	Ω
<code>\Sigma</code>	Σ

در دستورات `xlabel`, `ylabel`, `title`, `text`, همچنین میتوان نحوه نمایش کل متن را با استفاده از نام ویژگی - مقدار ویژگی تنظیم کرد. مثلاً

```
text(x,y,'Some text',PropertyName,PropertyValue)
```

- `PropertyName` یک رشته متنی است
- `PropertyValue` یک عدد یا یک رشته متنی است

مثال

```
text(x,y,'Depth','Rotation',45)
```

مینویسد

Depth

برخی جفت‌های نام ویژگی-مقدار ویژگی

Property name	Description	Possible property values
Rotation	Specifies the orientation of the text.	Scalar (degrees) Default: 0
FontAngle	Specifies italic or normal style characters.	normal, italic Default: normal
FontName	Specifies the font for the text.	Font name that is available in the system.
FontSize	Specifies the size of the font.	Scalar (points) Default: 10
FontWeight	Specifies the weight of the characters.	light, normal, bold Default: normal
Color	Specifies the color of the text.	Color specifiers (see Section 5.1).
Background-Color	Specifies the background color (rectangular area).	Color specifiers (see Section 5.1).
EdgeColor	Specifies the color of the edge of a rectangular box around the text.	Color specifiers (see Section 5.1). Default: none.
LineWidth	Specifies the width of the edge of a rectangular box around the text.	Scalar (points) Default: 0.5

دستور axis:

MATLAB محدود محورها را در دستور plot به نحوی تنظیم میکند که تمام داده ها نمایش داده شوند و محدود اعداد مناسبی باشند. این محدود را میتوان با دستور axis عوض کرد

حالت‌های رایج axis عبارتند از:

`axis([xmin xmax ymin ymax])`

- تعیین محدود هر دو محور

`axis equal`

- تعیین مقیاس یکسان برای هر دو محور

`axis square`

- تعیین ناحیه مربعی برای محورها

`axis tight`

- تعیین محدود محورها بر اساس محدود داده‌ها
(که الزاماً اعداد مناسبی نیستند)

دستور grid:

grid on

- اضافه کردن شبکه به نمودار

grid off

- حذف شبکه از نمودار

این کد

```

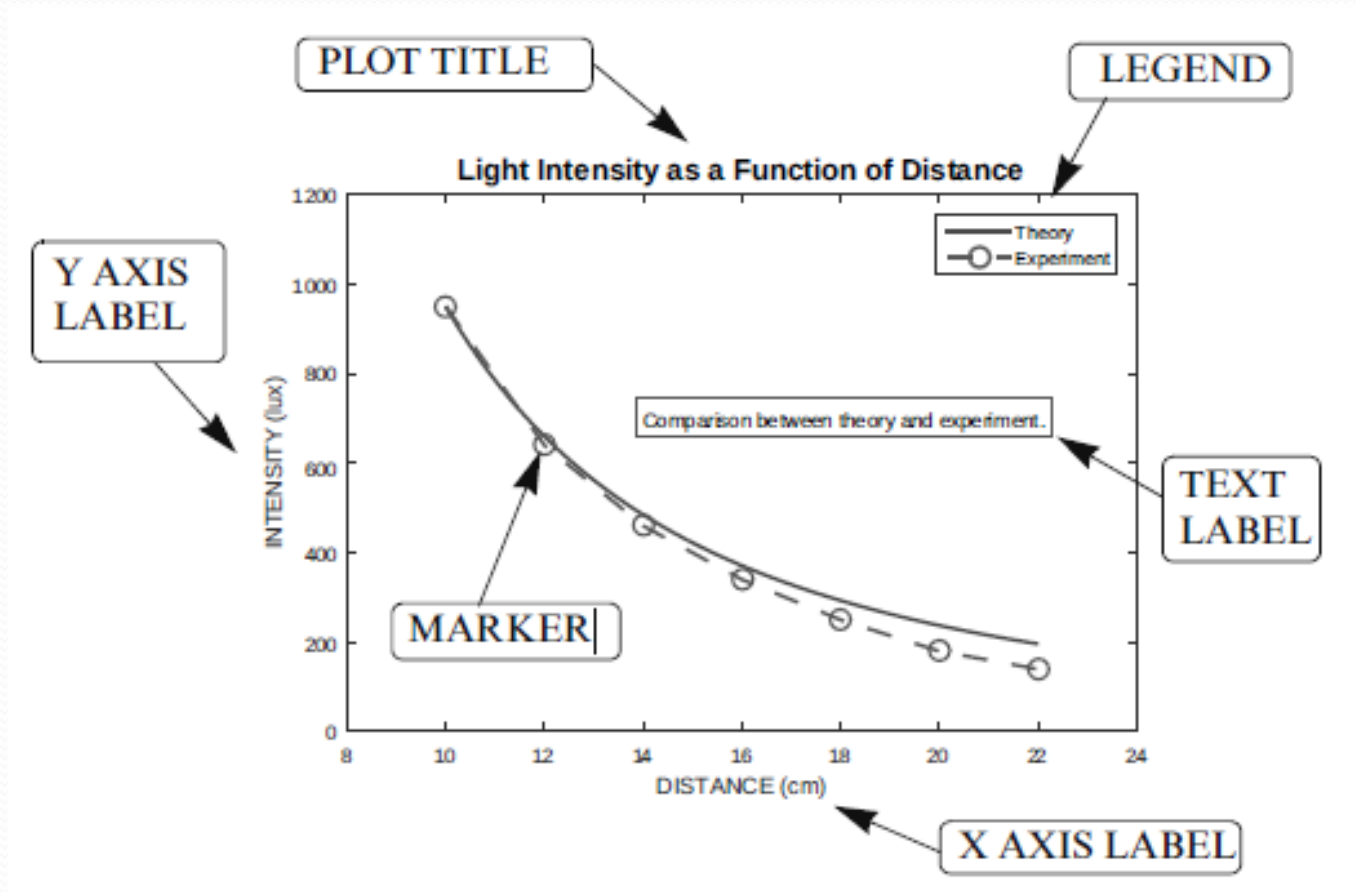
x=[10:0.1:22];
y=95000./x.^2;
xd=[10:2:22];
yd=[950 640 460 340 250 180 140];
plot(x,y,'-','LineWidth',1.0)
xlabel('DISTANCE (cm)')
ylabel('INTENSITY (lux)')
title('\fontname{Arial}Light Intensity as a Function of Distance','FontSize',14)
axis([8 24 0 1200])
text(14,700,'Comparison between theory and experiment.','Edge-
Color','r','LineWidth',2)
hold on
plot(xd,yd,'ro--','linewidth',1.0,'markersize',10)
legend('Theory','Experiment',0)
hold off

```

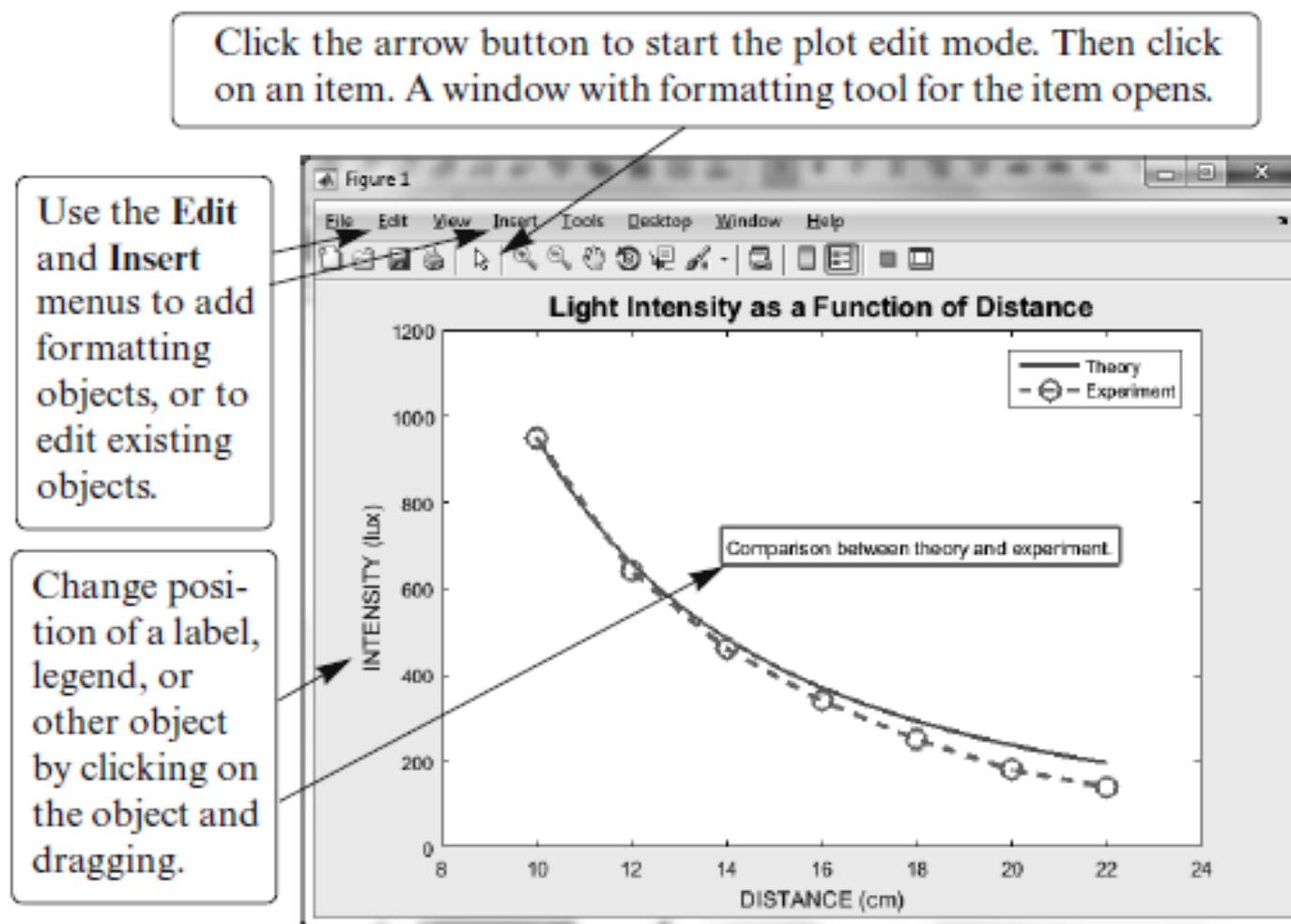
Formatting text inside the title command.

Formatting text inside the text command.

این نمودار را ترسیم می کند



می توان به صورت تعاملی نمودار را در پنجره ترسیم تنظیم کرد

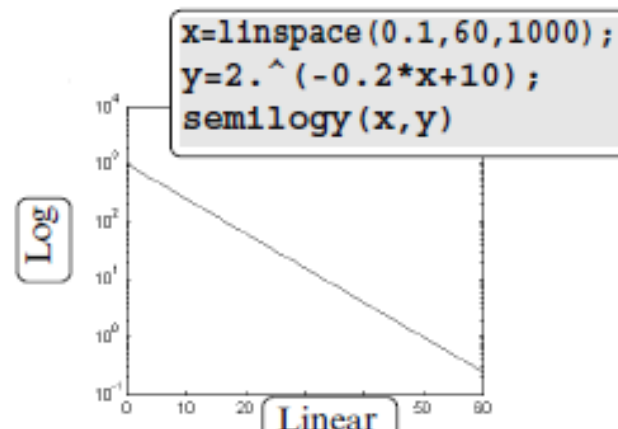
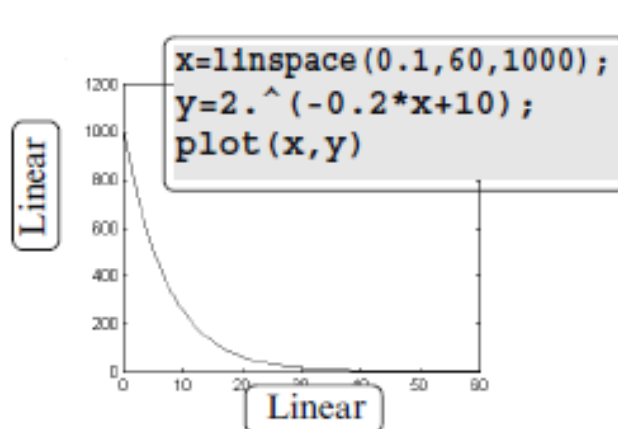


- گاهی وقتها از دستور plot در حالتی که یکی یا هر دوی محورها لگاریتمی هستند استفاده میشود
- برای نمایش داده هایی با مقادیر دارای دامنه گسترده استفاده میشود
 - برای نمایش واضح تر برخی روابط تابعی استفاده میشود
 - مثلاً $y = 10^{(2x+3)}$ در یک ترسیم نیمه لگاریتمی به صورت یک خط راست است

دستورات MATLAB برای ترسیمات لگاریتمی

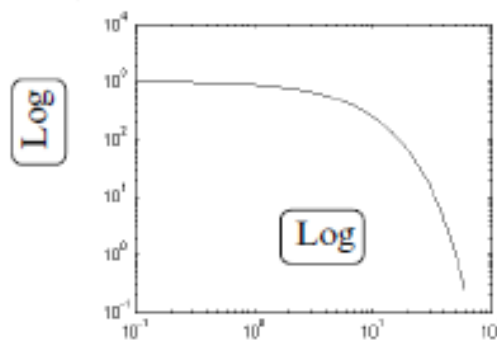
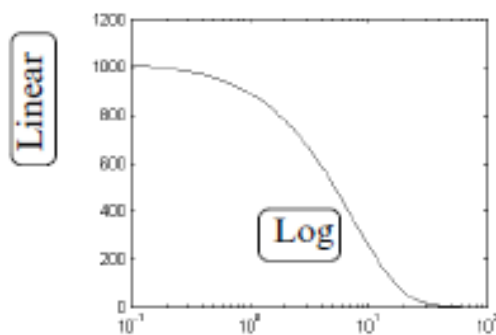
<code>semilogy(x, y)</code>	Plots y versus x with a log (base 10) scale for the y axis and linear scale for the x axis.
<code>semilogx(x, y)</code>	Plots y versus x with a log (base 10) scale for the x axis and linear scale for the y axis.
<code>loglog(x, y)</code>	Plots y versus x with a log (base 10) scale for both axes.

- میتوان از تنظیم کننده های خطوط و نام ویژگی - مقدار ویژگی در `plot` استفاده کرد
- در محورهای لگاریتمی، مطمئن شوید همه داده ها بزرگتر از صفر باشند، و گرنه لگاریتم تعریف نشده است



```
x=linspace(0.1,60,1000);
y=2.^(-0.2*x+10);
semilogx(x,y)
```

```
x=linspace(0.1,60,1000);
y=2.^(-0.2*x+10);
loglog(x,y)
```



در رسم داده های آزمایشگاهی، معمولاً به نوعی معیاری از عدم قطعیت در اندازه گیری نشان داده میشود

- گاهی اوقات از میله های خط استفاده میشود. خطوط عمودی معمولاً کوچک در بالا و پایین نقاط داده. اندازه آنها نشان دهنده عدم قطعیت است
- معیار عدم قطعیت معمولاً خطای استاندارد است، که تقریباً برابر است با انحراف معیار نمونه هایی که برای به دست آوردن یک نقطه داده استفاده شده اند

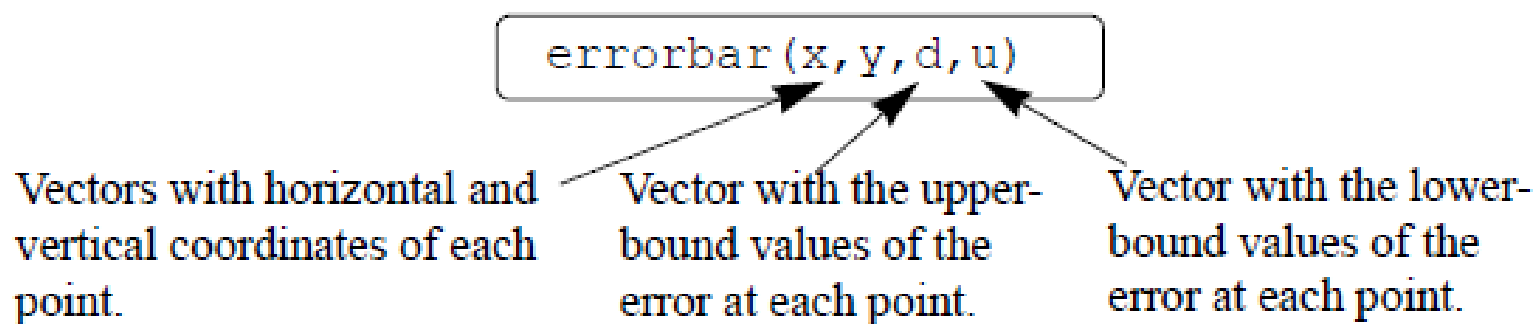
`errorbar(x, y, e)`

- همه بردارها در دستور باید دارای اندازه برابر باشند
- x و y داده های محورهای افقی و عمودی هستند
- e خطای اندازه گیری شده در هر نقطه است
- در هر $y(i)$ ، MATLAB میله خطای عمودی را از $y(i) - e(i)$ تا $y(i) + e(i)$ رسم میکند

`errorbar(x, y, d, u)`

- همه بردارها در دستور باید دارای اندازه برابر باشند
- x و y داده های محورهای افقی و عمودی هستند
- u, d خطای اندازه گیری شده در هر نقطه است
- در هر $y(i)$ ، MATLAB میله خطای عمودی را از $y(i) - d(i)$ تا $y(i) + u(i)$ رسم میکند

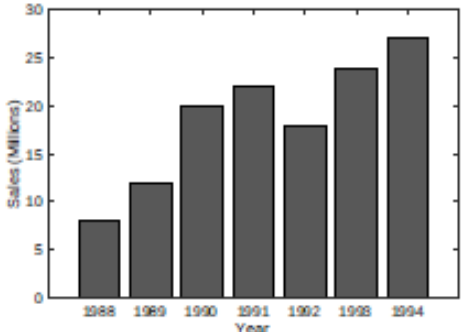
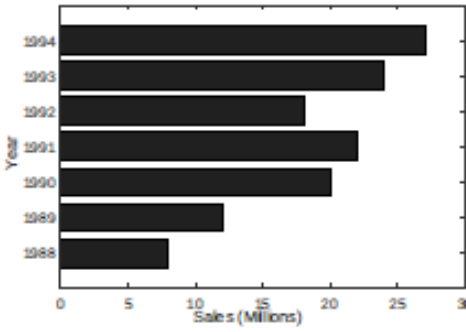
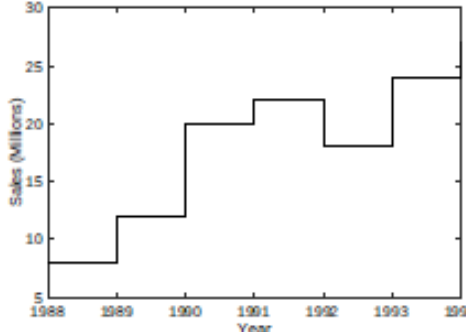
نکته: در شکل زیر جای توضیحات فلشهای دوم و سوم باید عوض شود



MATLAB انواع مختلفی از نمودارها را رسم میکند، مانند میله ای، پله ای، شافه ای، دایره ای و غیره

- برای اطلاعات بیشتر در این خصوص به راهنمای MATLAB، قسمت Graphics، 2-D and 3-D plots مراجعه کنید

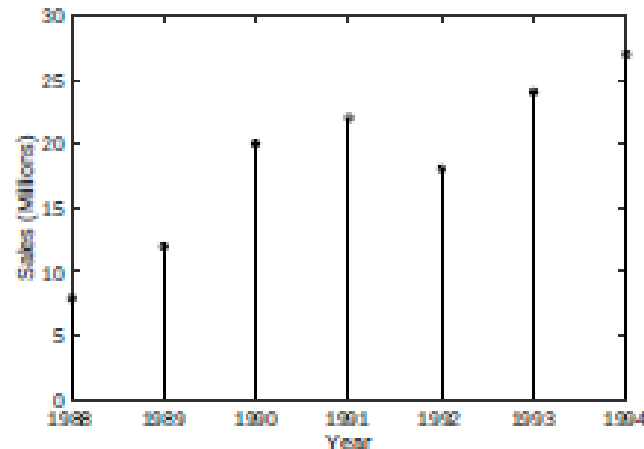
مثال هایی از نمودارهای خاص

<p><u>Vertical Bar Plot</u></p> <p>Function format:</p> <p><code>bar(x,y)</code></p>		<pre>yr=[1988:1994]; sle=[8 12 20 22 18 24 27]; bar(yr,sle,'r') xlabel('Year') ylabel('Sales (Millions)')</pre> <p>The bars are in red.</p>
<p><u>Horizontal Bar Plot</u></p> <p>Function format:</p> <p><code>barh(x,y)</code></p>		<pre>yr=[1988:1994]; sle=[8 12 20 22 18 24 27]; barh(yr,sle) xlabel('Sales (Millions)') ylabel('Year')</pre>
<p><u>Stairs Plot</u></p> <p>Function format:</p> <p><code>stairs(x,y)</code></p>		<pre>yr=[1988:1994]; sle=[8 12 20 22 18 24 27]; stairs(yr,sle) xlabel('Year') ylabel('Sales (Millions)')</pre>

Stem Plot

Function
Format

`stem(x,y)`

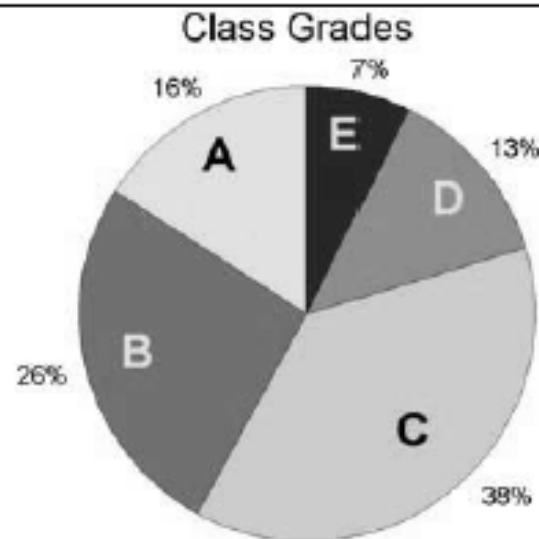


```
yr=[1988:1994];
sle=[8 12 20 22 18 24 27];
stem(yr,sle)
xlabel('Year')
ylabel('Sales (Millions)')
```

Pie Plot

Function
format:

`pie(x)`



```
grd=[11 18 26 9 5];
pie(grd)
title('Class Grades')
```

MATLAB draws the sections in different colors. The letters (grades) were added using the Plot Editor.

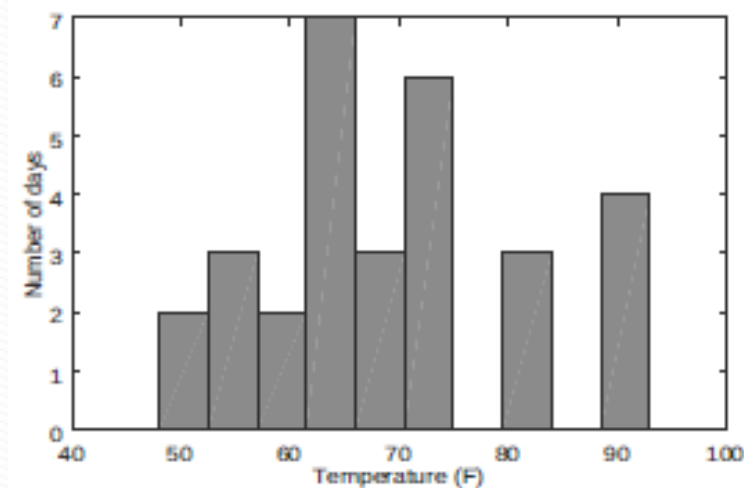
هیستوگرام نموداری از توزیع داده ها است. کل دامنه داده ها به زیر دامنه ها یا دسته های متوالی تقسیم میشود.

در ترسیم هیستوگرام

- هر دسته با یک میله عمودی مشخص میشود
- سمت راست و چپ میله عمودی دامنه داده ها را در دسته نشان می دهد
- ارتفاع میله عمودی تعداد داده ها در دسته را نشان میدهد

دستور `hist` در MATLAB هیستوگرام ایجاد میکند. حالت ساده آن `hist(y)` است.

- y بردار داده ها است
- `hist` داده ها را به ده دسته تقسیم کرده و نمودار را رسم میکند



```
>> y=[58 73 73 53 50 48 56 73 73 66 69 63 74 82 84 91 93 89
91 80 59 69 56 64 63 66 64 74 63 69];
>> hist(y)
```

حالت‌های دیگر hist:

```
hist(y,nbins)
```

- MATLAB دامنه را به nbins دسته مساوی تقسیم میکند

```
hist(y,x)
```

- بردار x مرکز هر دسته را مشخص میکند
- فاصله بین درایه های x میتواند متفاوت باشد
- مرز دسته ها مرکز دسته های مجاور است

در صورت نیاز میتوانید ارتفاع هیستوگرام را داشته باشید

```
n=hist(y)  n=hist(y,nbins)  
n=hist(y,x)
```

- خروجی بردار n است
- اندازه n برابر تعداد دسته ها است
- مقدار هر درایه n برابر تعداد نقاط داده در هر دسته است

در دو حالت میتوانید مرکز دسته ها را نیز داشته باشید

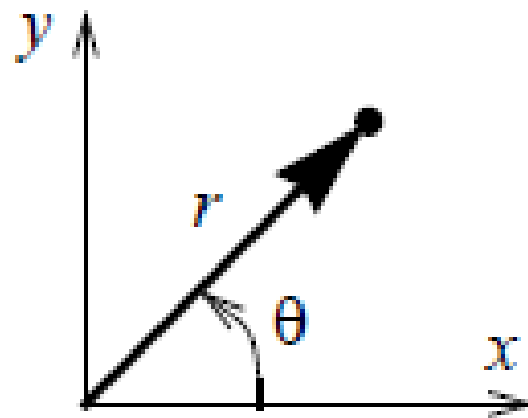
`[n xout]=hist(y)` یا `[n
xout]=hist(y,nbins)`

- `n` مانند قبلی است
- `xout(i)` مرکز دسته `i`ام است

```
>> [n xout]=hist(y)
n =
     2     3     2     7     3     6 |     0     3     0     4
xout =
    50.2500    54.7500    59.2500    63.7500    68.2500    72.7500
    77.2500    81.7500    86.2500    90.7500
```


در مختصات قطبی، نقاط در صفحه با مختصات (r, θ) مشخص میشوند

- r فاصله از مبدأ است
- θ زاویه از محور افقی مثبت است. جهت مثبت آن در جهت خلاف عقربه های ساعت است



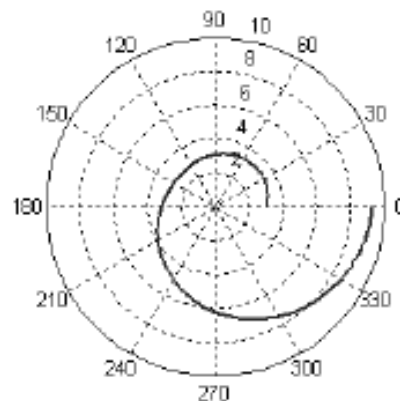
برای رسم نمودارهای قطبی در MATLAB

`polar(theta, radius, 'line specifiers')`

- `theta` به رادیان و به صورتی است که تعریف شد
- `radius` به صورتی است که تعریف شد
- هر دو باید بردار و با اندازه برابر باشند
- تعیین کننده های ویژگی خطوط مانند `plot`

For example, a plot of the function $r = 3\cos^2(0.5\theta) + \theta$ for $0 \leq \theta \leq 2\pi$ is shown below.

```
t=linspace(0,2*pi,200);
r=3*cos(0.5*t).^2+t;
polar(t,r)
```



`subplot(m,n,p)`

پنجره ترسیم را به m ردیف و n ستون
زیرترسیم تقسیم میکند

- زیرترسیم ها از چپ به راست و از بالا به پایین شماره گذاری میشوند، به طوری که بالا سمت چپ شماره ۱ باشد و پایین سمت راست $m*n$ در دستور به این نوع شماره گذاری بر میگردد

(3,2,1)	(3,2,2)
(3,2,3)	(3,2,4)
(3,2,5)	(3,2,6)

شماره زیرترسیمها برای یک
مجموعه $۳*۲$

`subplot(m,n,p)`

- اگر زیر ترسیمها موجود نباشند، دستور آنها را ایجاد کرده و زیر ترسیم p را زیرترسیم جاری میسازد
- اگر زیر ترسیمها موجود باشند، دستور زیرترسیم p را زیرترسیم جاری میسازد
- زمانی که با این دستور زیرترسیم جاری تعیین شود، ترسیمها و قالب بندی های بعدی در این زیرترسیم انجام میشوند

مثال 2-5 را برای کاربرد این دستور ببینید

با اجرای هر دستور ترسیم

1. یک پنجره ترسیم ایجاد میشود (اگر موجود نباشد)

2. هر ترسیمی در آن پنجره پای میشود

3. ترسیم جدید انجام میشود

برای داشتن ترسیم ها در چند پنجره، دستور
figure به کار می رود

figure

1. پنجره ترسیم جدید ایجاد میکند
 2. عنوان پنجره را Figure n قرار میدهد
 - n به صورتی که پنجره اول ۱ است، دومی ۲، و به همین ترتیب
 3. پنجره جدید را پنجره ترسیم فعال میسازد
 4. پنجره را به روی صفحه نمایش می آورد
- دستورات ترسیم بعدی در پنجره ترسیم فعال اجرا میشوند

مثال

```
>> fplot(@ (x) x.*cos(x), [0,10])
```

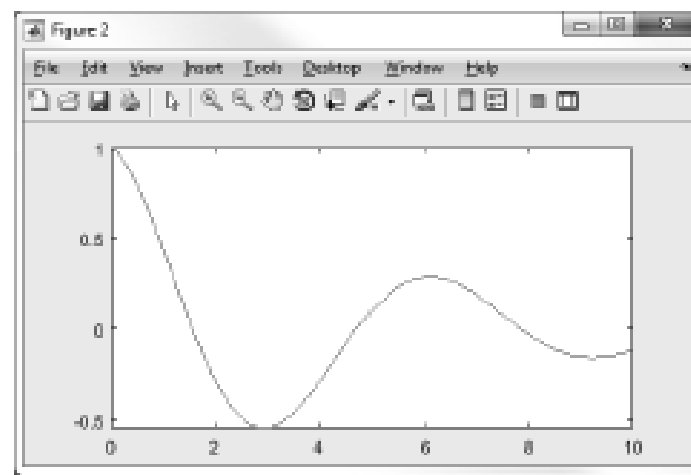
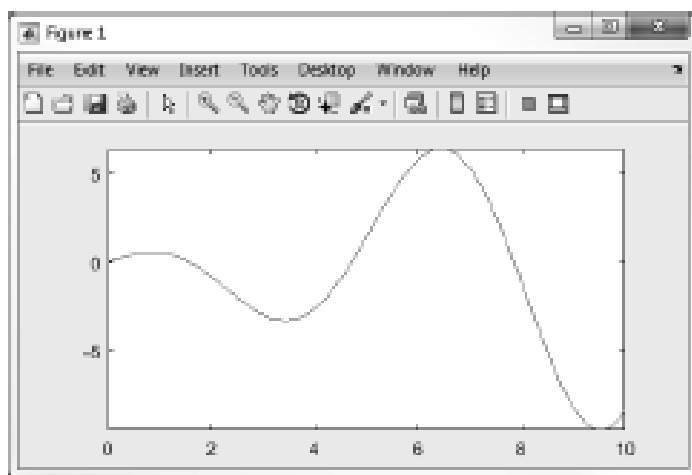
Plot displayed in Figure 1 window.

```
>> figure
```

Figure 2 window opens.

```
>> fplot(@ (x) exp(-0.2*x) .*cos(x), [0,10])
```

Plot displayed in Figure 2 window.



`figure(n)`

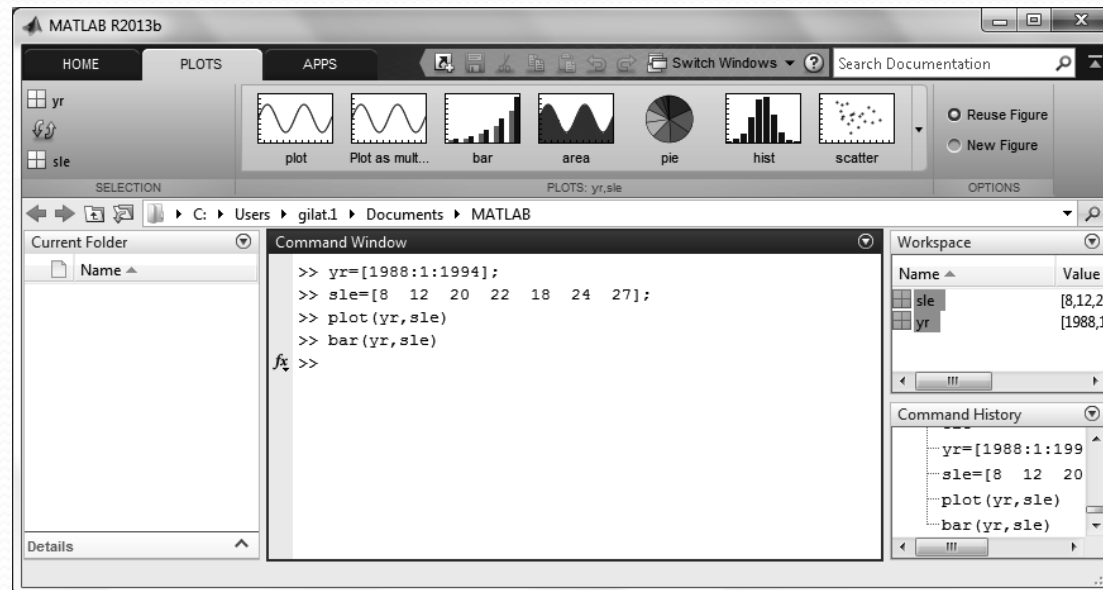
- اگر پنجره ترسیم n موجود باشد آن را پنجره ترسیم فعال میسازد
- اگر پنجره ترسیم n موجود نباشد، آن را ایجاد کرده و پنجره ترسیم فعال میسازد
- در کدها مفید است، در جاهایی که سری داده های ۱ در پنجره ۱ و سری داده های ۲ در پنجره ۲ و به همین ترتیب رسم میشوند

از دستور `close` برای بستن پنجره های
ترسیم استفاده کنید

- `close` پنجره ترسیم فعال را میبندد
- `close(n)` پنجره ترسیم `n` را میبندد
- `close all` همه پنجره های ترسیم را
میبندد

میتوانید ترسیمها را به صورت تعاملی با استفاده از نوار ابزار PLOTS در پنجره فرمان انجام دهید

1. اگر پنجره فضای کاری نمایش داده نمیشود آن را نمایش دهید
2. مطمئن شوید هر متغیری که میخواهید در ترسیم استفاده کنید دارای مقدار است
3. تب PLOT را کلیک کنید

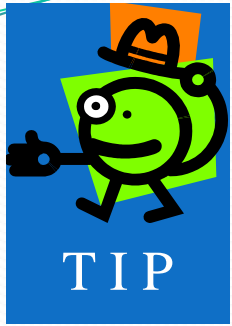


برای ترسیم دو بعدی

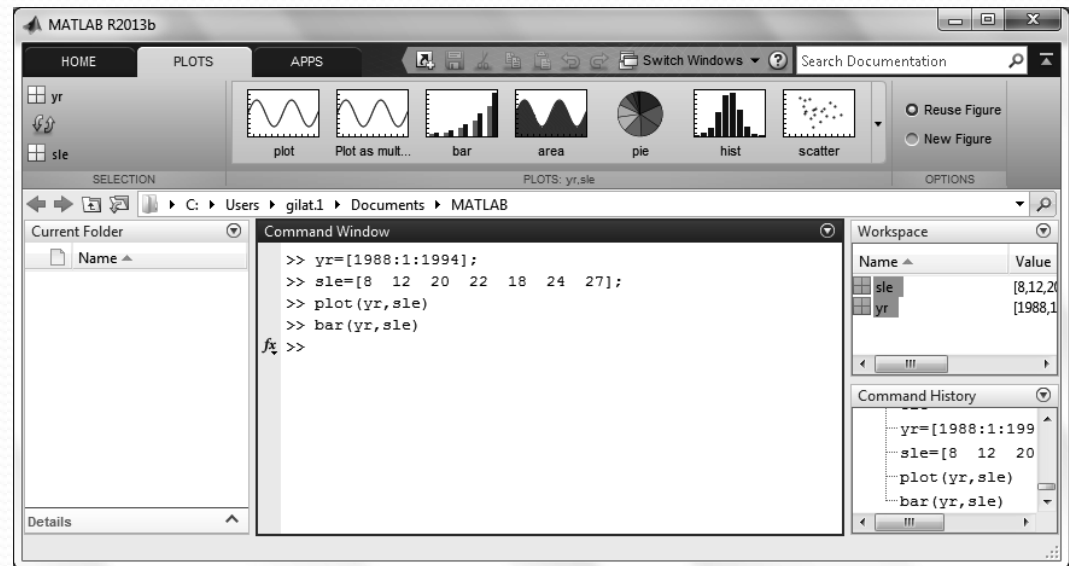
1. در پنجره فضای کاری برداری که میخواهید برای محور افقی استفاده کنید را کلیک کنید
2. با نگه داشتن کلید Ctrl بردار دیگر که مربوط به محور عمودی است را کلیک کنید
 - اگر تنها یک بردار انتخاب کنید شماره درایه ها در محور افقی و مقدار آنها در محور عمودی نمایش داده خواهد شد
3. ترسیم مورد نظر را از نوار ابزار انتخاب کنید

پس از کلیک روی نوع ترسیم مورد نظر، پنجره ترسیم جدیدی حاوی آن ترسیم باز خواهد شد

- برای رسم مجدد داده ها به نوعی دیگر، Reuse Figure را در سمت راست نوار ابزار انتخاب کرده و روی نوع ترسیم جدید کلیک کنید. ترسیم قبلی پاک شده و ترسیم جدید در همان پنجره انجام میشود
- برای ترسیم داده ها به شکلی دیگر، New Figure را در سمت راست نوار ابزار کلیک کرده و نوع جدید ترسیم را انتخاب کنید. ترسیم جدید در پنجره جدیدی که باز خواهد شد انجام میشود



پس از ترسیم، مطلب دستور مربوطه را در پنجره فرمان نمایش میدهد. میتوانید آن را در یک کد کپی کنید تا همان ترسیم را داشته باشید.



شماره تمرین های منتخب

۲۸ •

۳۱ •

۳۲ •

۳۳ •

۳۵ •

۳۶ •

۳۷ •

۴۰ •

۸ •

۱۳ •

۱۷ •

۱۸ •

۲۱ •

۲۲ •

۲۳ •

۲۴ •

۲۵ •