

فصل ۸

ترسیم دو بعدی و سه بعدی در متلب

متلب دارای امکانات وسیعی برای نمایش اطلاعات به صورت گرافیکی می باشد. این اطلاعات می تواند مقادیر بدست آمده از یک آزمایش تجربی و یا یک نمودار ریاضی باشد که به شکل های مختلفی می توان آنها را نمایش داد. نوع نمودارها در متلب می تواند خطی ستونی هیستوگرام و یا دایره ای باشد. همچنین ترسیم نمودارهای سه بعدی به صورت رویه و یا برش عرضی نیز ممکن است. ترسیم نمودارها در فضای مختلف (چهار بعدی) نیز به سادگی امکان پذیر است. در اینجا تنها به نمودارهای خطی دو و سه بعدی می پردازیم.

۱.۰.۸ رسم خط و منحنی

۱.۰.۱.۰.۸ رسم خط و منحنی در صفحه D-2

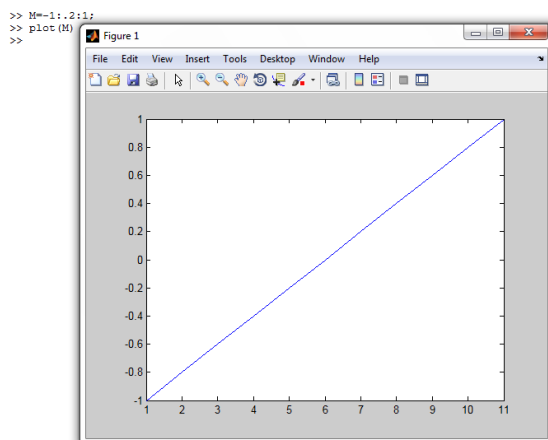
دستور Plot

از این دستور برای رسم خط و منحنی در صفحه (دو بعدی) استفاده می شود. این تابع نیز مانند سایر توابع موجود در متلب بسته به تعداد و نوع ورودی هایش نتایج متفاوتی ارائه می دهد. اگر تابع یک ورودی از نوع ماتریس سطری باشد یک خط پیوسته در یک محور دو بعدی رسم می شود که محور افقی آن نشان دهنده شماره درایه و محور عمودی آن نشان دهنده مقدار هر درایه است. مثلاً به نمودار رسم شده برای آرایه M توجه کنید: نتیجه اجرا دستور در پنجره Figures رسم شده است. با دقت در نمودار ملاحظه می کنید که محور افقی بر اساس تعداد درایه های M شماره گذاری شده و محور عمودی نشان دهنده مقادیر درایه ها از کمترین تا بیشترین مقدار است. هر درایه با خطی به درایه بعد وصل شده است و در نتیجه یک خط پیوسته بوجود آمده است.

- برای مقیاس بندی نمودار پس از رسم آن می توان از دستور grid on استفاده نمود.

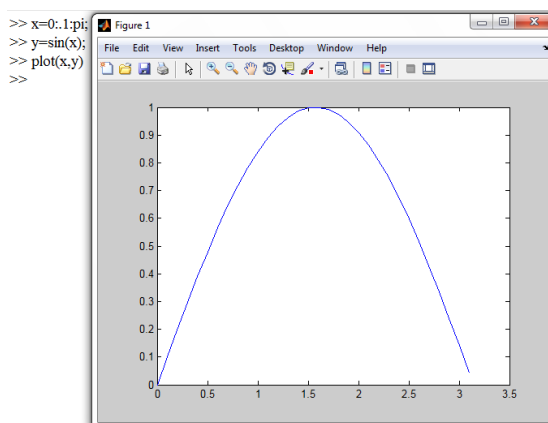
- برای عملیات ضرب توان و تقسیم باید قبل از عملگر از نقطه استفاده شود.

گاهی شما مایلید یک متغیر را به عنوان تابعی از متغیر دیگر رسم کنید. باید ابتدا به متغیر مستقل مقادیری در قالب یک ماتریس سطری اختصاص دهید، سپس تابع آن را تعریف کنید. حال با استفاده از تابع Plot که متغیر مستقل به عنوان پارامتر ورودی اول و تابع به عنوان آرگومان دوم به آن داده شده است، می توانید



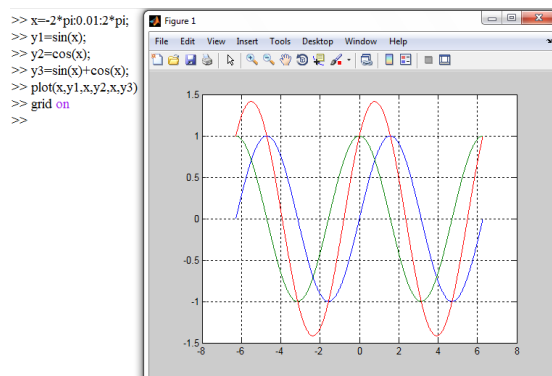
شکل ۱.۸: رسم خط در صفحه

نمودار تابع بر حسب متغیر مستقل را رسم کنید. به عنوان مثال اگر y تابعی از x باشد و بخواهیم y را بر حسب x رسم کنیم از $\text{plot}(x,y)$ استفاده می‌کنیم.



شکل ۲.۸: رسم منحنی در صفحه

در مثال زیر چند نمودار هم‌زمان در شکل رسم شده است. در دستور Plot و دستوراتی که برای رسم نمودار خط و منحنی به کار می‌روند علاوه بر ورودی‌های اصلی تعدادی ورودی که اغلب از نوع متن هستند، نیز می‌تواند باشد که در واقع شکل ظاهری نمودار را تنظیم می‌کند و شکل خط می‌تواند نقطه چین، خط و نقطه، یا دارای برخی علائم مانند $+$ یا برخی اشکال هندسی باشد. همچنین می‌توانید رنگ خط را نیز تعیین کنید. این چند کار هم‌زمان با نوشتن علامت مورد نظر تان برای رسم خط و حرفی که معرف رنگ خط است و نوشتن نمادی که می‌خواهید روی خط ظاهر شود، داخل ”انجام می‌گیرد. مثلاً اگر بنویسید $-r^*$ ، آنگاه خط بریده و قرمز رنگ



شکل ۳.۸: رسم چند منحنی در یک صفحه

خواهد بود و با ستاره‌هایی روی آن علامت‌گذاری می‌شود. علائم روی خط در واقع جایگاه یکایک درایه‌ها را نشان می‌دهند.

جداول زیر حرف مربوط به هر رنگ و علائمی را که می‌توانید استفاده کنید تا خط را رسم و روی آن علامت‌گذاری کنید، بیان می‌کند. با استفاده از دستور `help plot` می‌توانید به صورت مستقیم به این علائم در متلب دسترسی داشته باشید.

جدول ۱.۸: انواع خطوط

نوع خط	Line type	علامت
خط پر	<code>solid (default)</code>	-
خط تیره	<code>Dashed</code>	- -
نقطه‌چین	<code>Dotted</code>	:
خط تیره-نقطه‌چین	<code>Dash-Dot</code>	.-

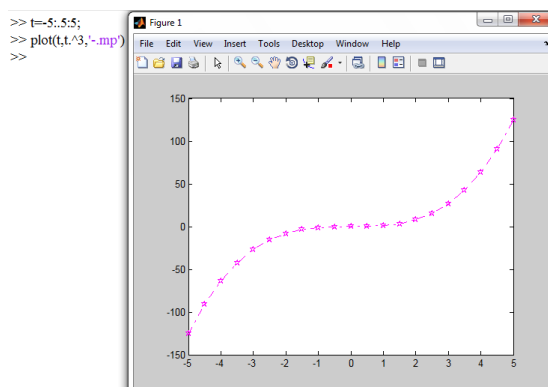
جدول ۲.۸: رنگ‌ها

رنگ	Color	علامت
قرمز	<code>red</code>	r
سبز	<code>green</code>	g
آبی	<code>blue</code>	b
آبی تیره (آبی مایل به سبز)	<code>cyan</code>	c
مایل به قرمز	<code>magenta</code>	m
زرد	<code>yellow</code>	y
سیاه	<code>black</code>	k
سفید	<code>white</code>	w

جدول ۳.۸: انواع علائم برای نشانه‌گذاری جای داده‌ها

نوع علامت	Type Marker	علامت
علامت جمع	plus	+
دایره	circle	o
ستاره	star	*
نقطه	point	.
علامت ایکس	x-mark	X
مربع	square	s or 'square'
لوزی	diamond	d or 'diamond'
مثلث (رو به بالا)	(up) triangle	^
مثلث (رو به پایین)	(down) triangle	V
مثلث (به سمت راست)	(right) triangle	<
مثلث (به سمت چپ)	(left) triangle	>
پنج ضلعی	(pentagram) star five-pointed	p or 'pentagram'
شش ضلعی	(hexagram) star six-pointed	h or 'hexagram'

مثال زیر نمونه‌ای برای استفاده از این جداول را نشان می‌دهد. در هنگام استفاده از علائم دو نکته



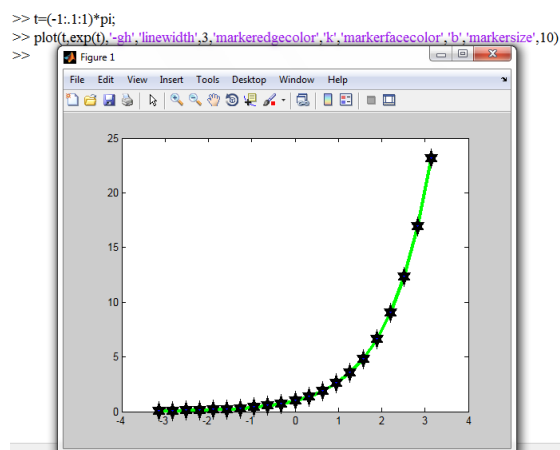
شکل ۴.۸: رسم با استفاده از جداول علائم

حائز اهمیت است:

- ترتیب وارد کردن علامت‌ها برای سه حالت خط، رنگ و نشانه‌گذاری مهم نیست.
- متلب به بزرگ یا کوچک بودن حروف (تنها در این حالت) حساس نیست.

برای سایر اعداد و علائمی که برای ویرایش خط استفاده میشوند ابتدا نام آن‌ها را به صورت متن و سپس عدد یا علامت مربوط را به عنوان ورودی بعدی وارد کنید. مثلاً برای تعیین ضخامت خط ابتدا عبارت 'linewidth' و پس از آن عدد موردنظر را به تابع می‌دهیم. عبارات دیگر، 'markeredgecolor' برای تعیین رنگ دور علائم نشانه‌گذاری، 'markerfacecolor' برای تعیین رنگ داخل علائم نشانه‌گذاری

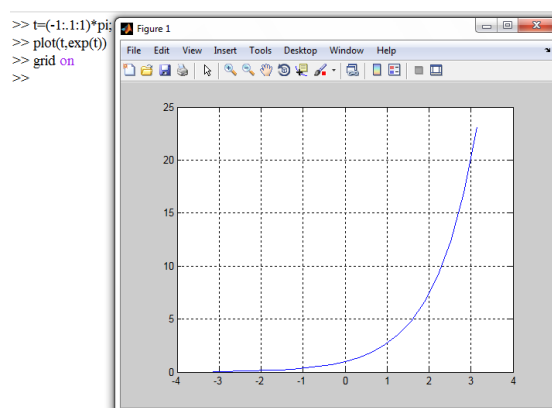
و 'markersize' برای مشخص نمودن اندازه علائم نشانه‌گذاری استفاده می‌شوند. تاثیر این عبارات را در مثال زیر مشاهده می‌کنید.



شکل ۵.۸: ویرایش خط

دستور grid on

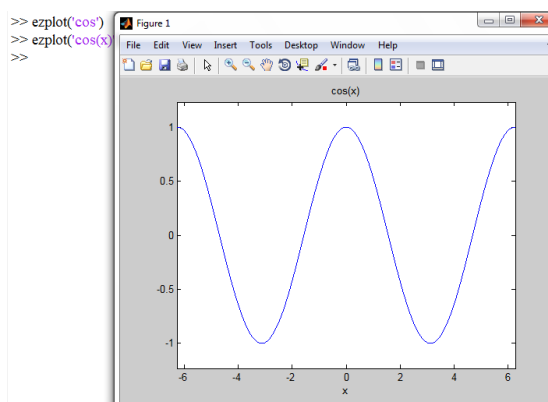
این دستور باعث می‌شود که صفحه رسم نمودار تقسیم‌بندی گردد. البته روشن و خاموش کردن این تقسیم‌بندی‌ها و بسیاری تنظیمات دیگر که در اینجا با استفاده از دستورات اعمال نمودیم، از طریق ابزار تنظیماتی که در پنجره figures قرار داده شده است نیز قابل تنظیم می‌باشد.



شکل ۶.۸: دستور grid on

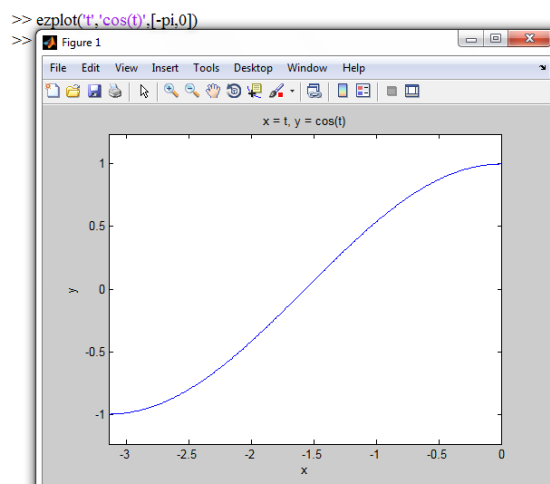
دستور ezplot

این دستور همانند دستور plot عمل می‌کند و به آسانی توابع را می‌توان با آن رسم کرد. متغیر اول و دوم همانند دستور plot متغیر مستقل و وابسته تابع هستند. در صورتی که متغیر دیگری وارد نکنیم به طور خودکار نمودار به ازای بازه $[0, 2\pi]$ برای متغیر مستقل رسم می‌شود و اگر ورودی اول و دوم به هم وابسته نباشند آنگاه برای هر یک بازه پیش فرض $[-2\pi, 2\pi]$ خواهد بود. اما شما می‌توانید یک بازه به عنوان ورودی سوم برای این دستور معرفی کنید. ورودی این دستور می‌تواند یک تابع باشد که به صورت رشته در جلوی این دستور بیان می‌گردد. در مثال زیر مشاهده می‌کنید که حتی اگر تنها نام تابع بیان شود (بدون بیان متغیر x باز هم تابع توسط این دستور رسم می‌گردد).



شکل ۷.۸: دستور ezplot با ورودی رشته‌ای

روش دیگری هم برای تعریف تابع وجود دارد. در این روش نیازی به رشته ای تعریف کردن تابع نیست و تنها لازم است که از علامت @ در ابتدا نام تابع استفاده شود یعنی ezplot(@sin). تنها تفاوت با حالت قبل این است که نباید متغیر تابع ذکر شود. در مثال زیر بازه تابع را خودمان انتخاب کردیم.



شکل ۸.۸: دستور ezplot با ورودی رشته‌ای و بازه دلخواه

ورودی‌هایی که جهت تنظیم نمودار در مورد دستور plot وارد می‌کنیم در مورد ezplot قابل استفاده نیست و در صورت استفاده متلب خطا خواهد داد. البته روش‌هایی برای تنظیم نمودار در این حالت نیز وجود دارد که به دلیل طولانی شدن موضوع از پرداختن به آن می‌پرهیزیم.

دستور title

این دستور شکل را نام‌گذاری می‌کند. در داخل آن باید به صورت رشته تعریف گردد (title('string')). هر نامی که بنویسیم در بالای ترسیم نشان داده می‌شود.

دستور legend

اگر چندین نمودار رسم کرده باشیم ممکن است نتوانیم تشخیص دهیم که کدام ترسیم مربوط به کدام نمودار است. بوسیله این دستور می‌توانیم بر حسب رنگ و نوع ترسیم نمودارها را از هم تمییز دهیم. ترتیب نوشتن نام‌ها بدینگونه است که در دستور plot هر کدام از نمودارها ترسیم شده در اینجا هم همانگونه عمل می‌شود.

دستور text

از این دستور برای نوشتن جمله‌ای در مکان خاصی (مختصات خاصی) از شکل استفاده می‌شود. به عنوان مثال دستور test(x,y,'string') یعنی در نقطه (x,y) کلمه string نوشته شود.

دستور clf

با اجرا این دستور یک صفحه figure خالی ایجاد می‌گردد. در هنگام اجرا دستور plot قبل از ترسیم صفحه پاک می‌شود (می‌توان گفت دستور clf اجرا می‌گردد).

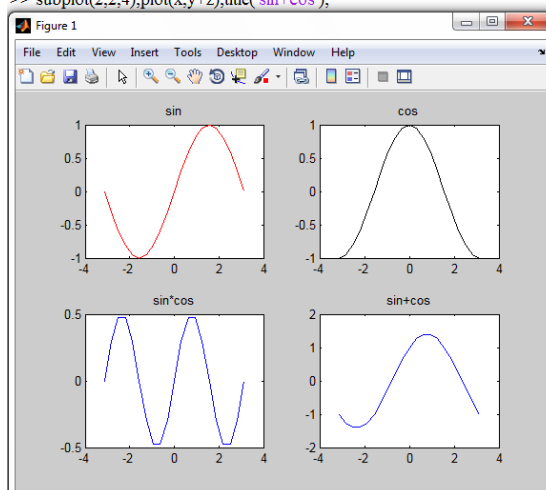
دستور hold

با استفاده از دستور hold on مانع پاک شدن صفحه نمایش می‌شویم تا نمودارهای بعدی بر روی نمودار اولی بیافتند (در این نمودارها می‌توان رنگ‌ها و نوع خط‌ها را متفاوت تعریف کرد). با دستور hold off دوباره با هر بار اجرای plot صفحه نمایش پاک می‌شود و نمودار جدید در آن رسم می‌گردد.

دستور subplot

بوسیله این دستور می‌توان صفحه ترسیم را به چندین قسمت تقسیم نمود. این دستور به صورت sub-plot(m,n,p) استفاده می‌شود. با این دستور صفحه به m سطر و n ستون تقسیم می‌شود و قسمت P-ام را آدرس دهی می‌کند که شماره هر قسمت از ردیف بالا شروع می‌شود.

```
>> x=-pi/10:pi;
>> subplot(2,2,1);plot(x,y,'r');title('sin');
>> subplot(2,2,2);plot(x,z,'k');title('cos');
>> subplot(2,2,3);plot(x,t);title('sin*cos');
>> subplot(2,2,4);plot(x,y+z);title('sin+cos');
```



شکل ۹.۸: دستور subplot

۲.۱.۸ رسم خط و منحنی در فضا (۳-D)**دستور plot3**

این دستور برای رسم نمودار در فضا به کار می‌رود. توجه داشته باشید که نتیجه استفاده از این دستور نیز یک خط است با این تفاوت که این خط دارای بعد سوم بوده و در یک فضا سه بعدی رسم می‌گردد. تفاوت کار با این نمودار این است که سه متغیر به آن داده می‌شود که متغیر اول محور x ، متغیر دوم محور y و متغیر سوم محور z را تعیین می‌کند. توجه داشته باشید که هر سه متغیر باید ماتریس‌های سطری با طول برابر باشند. در اینجا حتماً باید حداقل سه متغیر مربوط به سه محور وارد شوند. پس از سه ورودی اصلی، می‌توانید عبارات مربوط به ویرایش نمودار را نیز وارد کنید. نمودار حاصله سه بعدی بوده و پس از رسم با فعال کردن ابزار Rotate 3D از نوار ابزار بالای پنجره figures و تغییر شکل موس به صورت

یک دایره، می‌توانید کلید سمت چپ موس را در حالی که موس را روی نمودار قرار داده اید، نگه دارید و با حرکت دادن آن به اطراف، نمودارتان را بچرخانید و از جهت‌های دیگر نیز آن را بررسی کنید. اگر در این شرایط روی نمودار کلیک راست کنید، امکانات زیر در اختیار شما قرار می‌گیرد:

- Reset to Original View: با استفاده از این گزینه، شکل به حالت اول خود باز گردانده می‌شود.

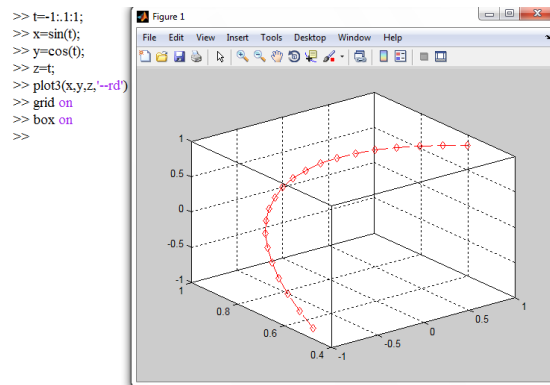
- Go to X-Y View: دیدن شکل تنها در صفحه x-y

- Go to X-Z View: دیدن شکل تنها در صفحه x-z

- Go to Y-Z View: دیدن شکل تنها در صفحه y-z

- Rotate Options: امکاناتی برای تنظیم چگونگی چرخش محور و نمودار در اختیار می‌گذارد.

برای آنکه از هم‌اندازه بودن طول متغیرهای ماتریس اطمینان حاصل کنیم، می‌توانیم از روش‌های گوناگونی استفاده کنیم. قرار دادن یک متغیر واسطه (مثلاً t)، تعریف یک متغیر و تابع قراردادن بقیه متغیرها در آن و یا استفاده از توابع `length` و `linspace` از آن جمله‌اند.



شکل ۱۰.۸: دستور `plot3`

همان‌طور که می‌بینید نوشتن دستور `grid on` پس از دستور `plot3` موجب مقیاس‌بندی صفحه شده است. دستور `box on` باعث می‌گردد که محور نمودار به صورت یک جعبه شفاف درآید که این حالت نمودار، تجسم شکل سه‌بعدی را راحت‌تر می‌سازد. توجه داشته باشید که دستورات ویرایش نمودار و یا محور آن پس از دستور `plot3` باید نوشته شوند تا روی آن تأثیرگذار باشند.

دستورات `xlabel`، `ylabel` و `zlabel`

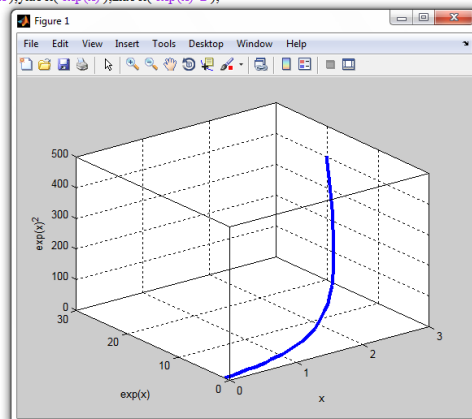
از این دستورات برای نام‌گذاری محورهای نمودار استفاده می‌شود. واضح است که از این دستورات در رسم دوبعدی نیز می‌توان استفاده نمود با این تفاوت که نیازی به تعریف محور z نیست.

فصل ۸. ترسیم دو بعدی و سه بعدی در متلب

```

>> x=0:1:3;
>> y=exp(x);z=y.^2;
>> plot3(x,y,z,'linewidth',3)
>> xlabel('x');ylabel('exp(x)');zlabel('exp(x)^2');
>> box on
>> grid on
>>

```



شکل ۸.۱۱: دستورات xlabel ، ylabel و zlabel

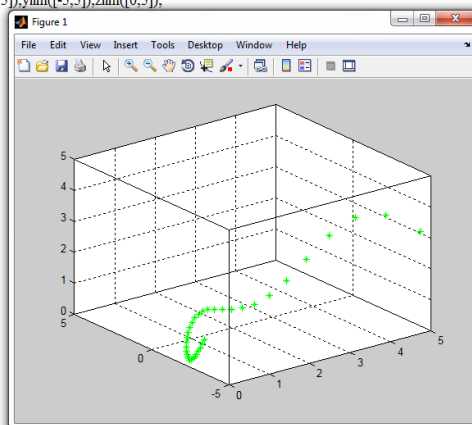
دستورات محدودکننده xlim ، ylim و zlim

این دستورات که ورودی آنها حدود محور مورد نظر را تعیین می‌کند برای نمودار سه بعدی xlim ، ylim و zlim هستند. اما برای یک نمودار دو بعدی xlim و ylim هستند.

```

>> a=-pi/18:pi;
>> b=logspace(-1,1,length(a));
>> c=sin(a)+cos(b);
>> plot3(b,a,c,'g*')
>> xlim([0,5]);ylim([-5,5]);zlim([0,5]);
>> grid on
>> box on
>>

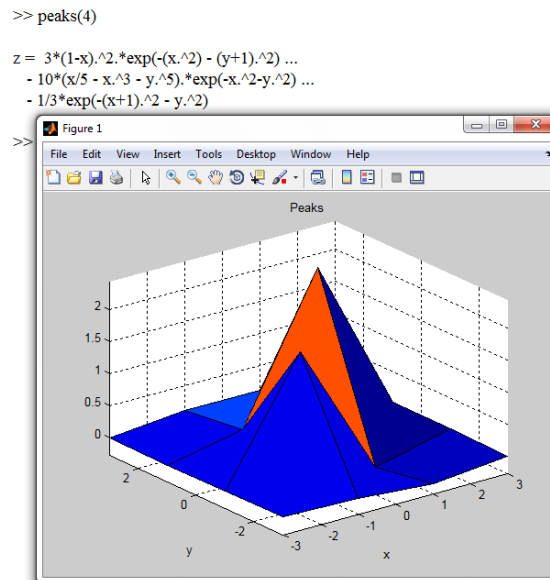
```



شکل ۸.۱۲: دستورات محدود کننده

رسم سطح و لایه برای توابع دو متغیره عددی

رسم لایه و سطح با نمودار یکسان نیست. نمودار یک منحنی و... می باشد (درکل یک خط) ولی سطح اینگونه نمی باشد و یک فضای پیوسته (لایه) است. خودمان می توانیم هر سطحی که بخواهیم بسازیم ولی برای راحتی کار یک تابع سطح در خود متلب قرار داده شده است که با دستور peaks می توان به این سطح دست یافت. در دستور peaks(n)، n دقت ترسیم را نشان می دهد (بازه مربوطه به عدد وارد شده تقسیم می شود و هر چقدر n بیشتر باشد قطعات کوچکتر و همین طور شکستگی نرم تر و رسم دقیق تر خواهد بود). در مورد توابع عددی ابتدا باید دو متغیر مستقل را طوری تغییر دهیم که اولاً هم بعد شوند (تعداد سطر و ستون برابری داشته باشند) ثانیاً همزمان با هر یک از درایه های یکی از آن دو که در تابع قرار می گیرد، تک تک درایه های دیگری نیز در تابع قرار گرفته و نقطه حاصله ثبت شود.



شکل ۱۳.۸: دستور peaks

دستور meshgrid

برای ساختن سطح باید شبکه ای کامل و همگن ساخته شود (منظور از شبکه خطوط عمود برهم است که محل تقاطع مکان نقاط سطح را نشان می دهد). در متلب شبکه را با کمک دستور meshgrid می سازیم. با دستور $[x,y]=\text{meshgrid}(\text{ , })$ می توان دو محور x و y را تقسیم بندی نمود. اگر از دستور $\text{meshgrid}([x,y])$ استفاده شود، هر دو محور به یک اندازه تقسیم بندی می شوند. دستور meshgrid متغیرها را طوری گسترش می دهد که متغیرهای جدید هم بعد و قابل استفاده برای رسم باشند.

```
>> [x,y]=meshgrid(-2:2,-2:2)
```

```
x =
```

```
-2 -1 0 1 2
-2 -1 0 1 2
-2 -1 0 1 2
-2 -1 0 1 2
-2 -1 0 1 2
```

```
y =
```

```
-2 -2 -2 -2 -2
-1 -1 -1 -1 -1
0 0 0 0 0
1 1 1 1 1
2 2 2 2 2
```

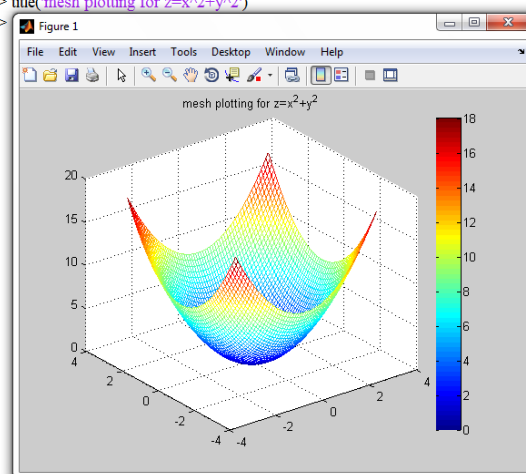
شکل ۸.۱۴: دستور meshgrid

می بینید که این دستور یک ماتریس ۲ در ۲ می سازد و اگر بیشتر دقت کنید می بینید که هر کدام از سطر y و یا ستون x می تواند معرف یک خط باشد (یک سری خط موازی عمودی و افقی).

دستور mesh

این دستور شبکه تشکیل شده را ترسیم می کند. سطح نمایش داده شده توسط این دستور صرفاً یک شبکه می باشد. این دستور به صورت $\text{mesh}(x,y,z)$ استفاده می شود.

```
>> [x,y]=meshgrid(-3:1:3);
>> z=x.^2+y.^2;
>> mesh(x,y,z)
>> colorbar
>> title('mesh plotting for z=x^2+y^2')
>>
```



شکل ۸.۱۵: دستور mesh

در مثال بالا در خط چهارم دستور colorbar استفاده شده که این دستور باعث ایجاد یک میله رنگی معرف ارتفاع می باشد و سطر آخر نیز عنوان ترسیم را نشان می دهد.

دستور off hidden

این دستور باعث می‌شود که قسمت‌های عقب و پشت بخش‌های جلوتر پنهان نشده و قابل رویت باشند.

دستور colormap

دستوری است برای تعیین طیف رنگی که نمودار شما با آن رنگ‌آمیزی می‌شود. انواع گوناگون آن در جدول زیر آورده شده است. با قراردادن نام هر حالت در مقابل دستور colormap نمودار با آن طیف رنگی رنگ‌آمیزی می‌شود.

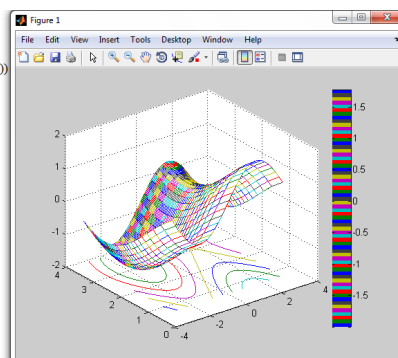


شکل ۱۶.۸: دستور colormap

دستور meshc

خروجی این دستور شبیه به خروجی دستور mesh است. با این تفاوت که روی صفحه زیرین تصویری سایه‌گونه از نمودار افتاده است.

```
>> x=pi:2*pi;
>> y=log(2:50);
>> [a,b]=meshgrid(x,y);
>> meshc(a,b,sin(a)+cos(b))
>> colormap lines
>> colorbar
>>
```

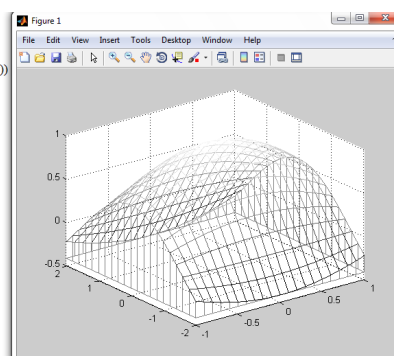


شکل ۱۷.۸: دستور meshc

دستور meshz

این دستور نیز نوعی دیگر از دستور mesh است که نقاط لبه‌ای آن با خطوط عمودی به صفحه زیرین متصل شده‌اند.

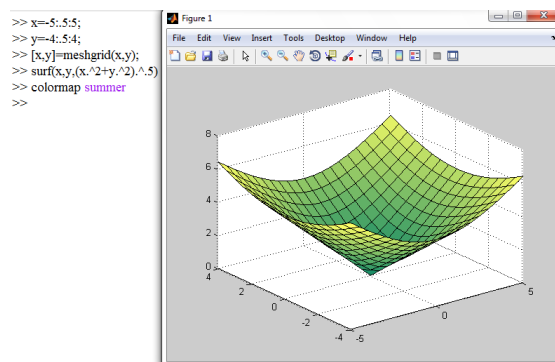
```
>> x=1:-1:1;
>> y=2:-2:2;
>> [x,y]=meshgrid(x,y);
>> meshz(x,y,cos(x).*cos(y))
>> hidden off
>> colormap gray
>>
```



شکل ۱۸.۸: دستور meshz

دستور surf

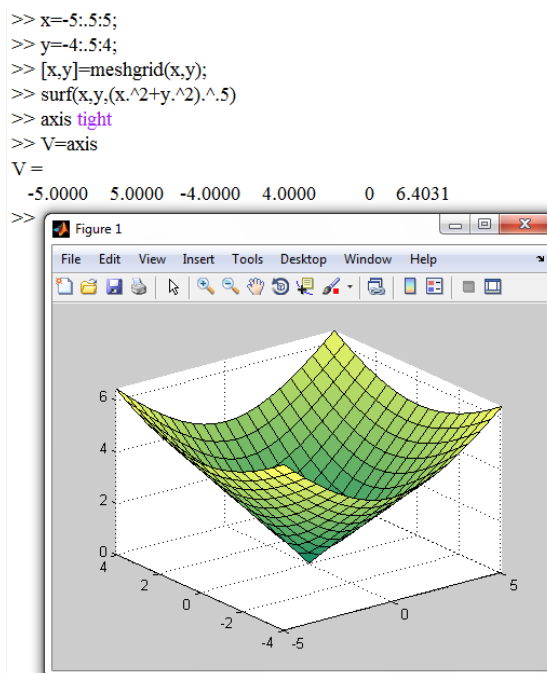
این دستور یک نمودار مشبک با شبکه‌های تو پر تولید می‌کند.



شکل ۱۹.۸: دستور surf

دستور axis

دستوری که باعث ناپدید شدن محورها می‌شود دستور `axis off` می‌باشد. دستور `axis` نیز تنظیمات گوناگونی را می‌پذیرد که همگی در شکل و اندازه محوره‌های نمودار تاثیر می‌گذارند. اگر این دستور به صورت `axis([Xmin Xmax Ymin Ymax Zmin Zmax])` استفاده شود محوره‌های x ، y و z مقیاس‌بندی می‌شوند. در حالت رسم دوبعدی می‌توان تنها محوره‌های x و y را مقیاس‌بندی نمود. با دستور `V=axis` می‌توان مقیاس‌های شکل رسم شده را یافت. اگر شکل دوبعدی باشد بردار سطری V دارای چهار مقدار خواهد بود و اگر شکل سه‌بعدی باشد دارای شش مقدار خواهد بود (هر دو مقدار برای یک بعد می‌باشد). با استفاده از دستور `axis auto` مقیاس‌های محورها به حالت پیش‌فرض خود بر می‌گردند. وقتی که از دستور `subplot` استفاده می‌شود می‌توان با استفاده از دستور `axis manual` تمامی (ها) `subplot` را در یک رنج رسم کرد. دستور `axis tight` باعث می‌شود که در شکل رسم شده دقیقاً در ابعاد اطلاعات تابع رسم شده، رسم شود. می‌توانید با مراجعه به `help` متلب کاربردهای دیگر این دستور را ملاحظه کنید.



شکل ۸.۲۰: دستور axis

دستور shading

این دستور که سه حالت flat، faceted و interp دارد، برای محو کردن خطوط روی سطح نمودار استفاده می‌شود. کامل‌ترین حالت برای interp رخ می‌دهد که هم خطوط محو می‌شوند و هم تیزی‌های سطح نمودار گرفته شده و سطحی صاف خواهیم داشت.

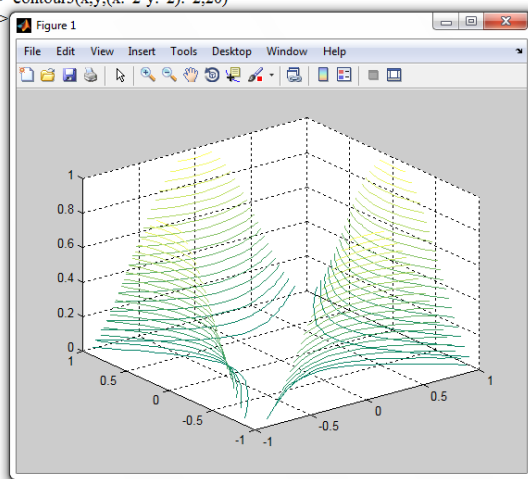
دستور surfc

این دستور نیز مانند دستور meshc تصویری از نمودار بر صفحه زیرین خواهد داشت.

دستور contour3

این دستور یک ورودی به عنوان ورودی چهارم خود می‌پذیرد که به اندازه این عدد نمودار را برش داده و تنها مقاطع بریده شده را نمایش می‌دهد.


```
>> x=-1:1:1;
>> [x,y]=meshgrid(x);
>> contour3(x,y,(x.^2-y.^2).^2,20)
>>
```

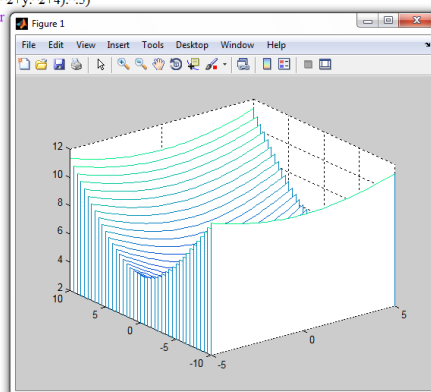


شکل ۱.۸: دستور contour3

دستور waterfall

این دستور نمودار آبشاری از شکل رسم می‌کند.

```
>> x=-10:5:10;
>> y=-5:2:5;
>> [x,y]=meshgrid(y,x);
>> waterfall(x,y,(x.^2+y.^2+4).^5)
>> colormap winter
>>
```

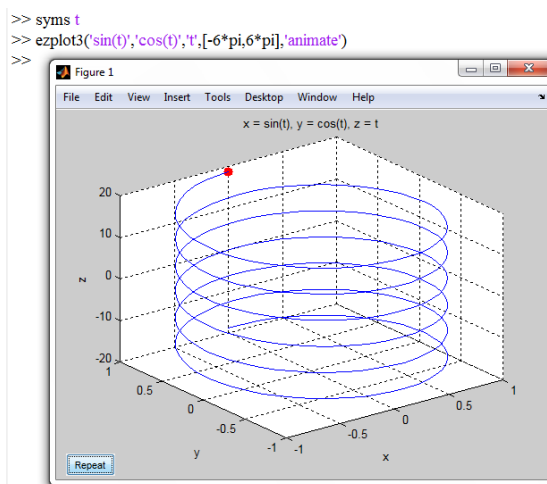


شکل ۲.۸: دستور waterfall

دستور ezplot3

در اینجا شما می‌توانید به ازای متغیرهای symbolic یک منحنی سه‌بعدی رسم کنید. مانند ezplot محدودده پیش‌فرض برای متغیر مستقل $[-2\pi, 2\pi]$ است. به عنوان مثال اگر متغیر هر سه محور به یک

متغیر مانند t وابسته باشند، t در این محدوده می باشد. اگر پس از سه ورودی اصلی، یک بازه قرار دهیم، این بازه حدود t را مشخص می کند. این دستور فقط در این حالت معنا دارد، چرا که `ezplot3` یک نمودار دو متغیره نیست و حتما باید متغیری باشد که هر سه متغیر ورودی به آن وابسته باشند.



شکل ۸.۲۳: دستور `ezplot3`

تذکر: برای تعریف متغیری به عنوان `symbol` از دستور `syms` استفاده می شود. به مثال زیر دقت کنید.

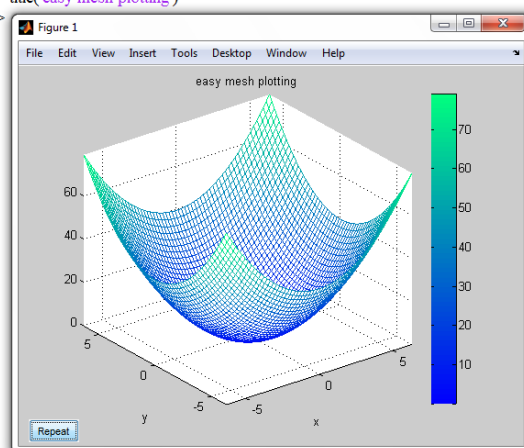
لکه قرمزی که روی محور مشاهده می شود و دکمه تکرار^۱ که در پایین صفحه ظاهر شده است، اثر عبارت `animate` در ورودی تابع می باشد. لکه قرمز پس از رسم نمودار، از کوچکترین نقطه، شروع به حرکت نموده و روی نمودار تا انتهای آن به مسیر خود ادامه می دهد. دکمه تکرار، برای تکرار این حرکت است. این لکه متحرک بخصوص در نمودارهای پیچیده شکل نمودار را قابل تجسم تر می کند. البته دستورات دیگری نیز برای انواع خاصی از نمودار خطها و منحنی ها وجود دارد که به علت آنکه کاربرد آنها به گستردگی موارد ذکر شده نیست، از بیان آنها خودداری می شود.

دستور `ezmesh`

این دستور یک شبکه برای یک تابع سه بعدی تعریف شده می سازد.

^۱ Repeat

```
>> ezmesh('x^2+y^2')
>> colorbar
>> title('easy mesh plotting')
>>
```

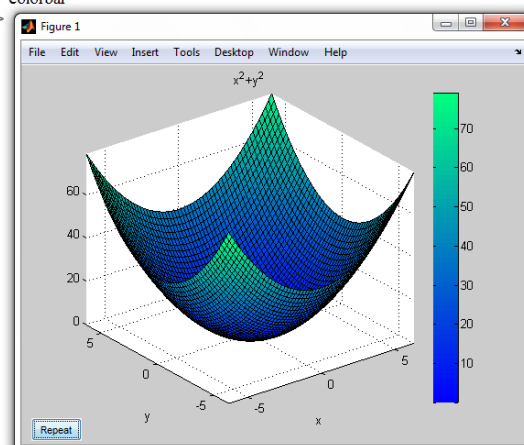


شکل ۲۴.۸: دستور ezmesh

دستور ezsurf

دقیقا مانند دستورات بالا ترسیم سطح را بصورت رنگ و سایه برای سطح را انجام می دهد.

```
>> ezsurf('x^2+y^2')
>> colorbar
>>
```

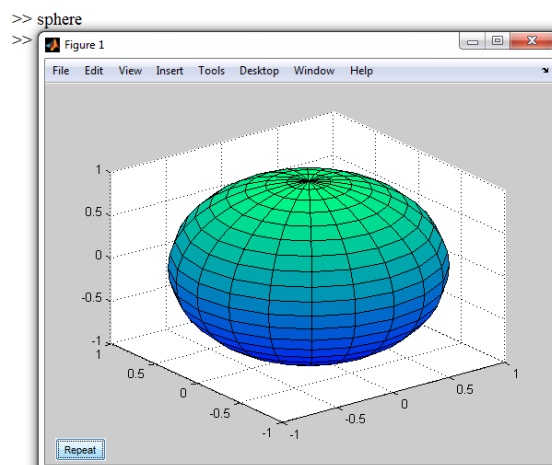


شکل ۲۵.۸: دستور ezsurf

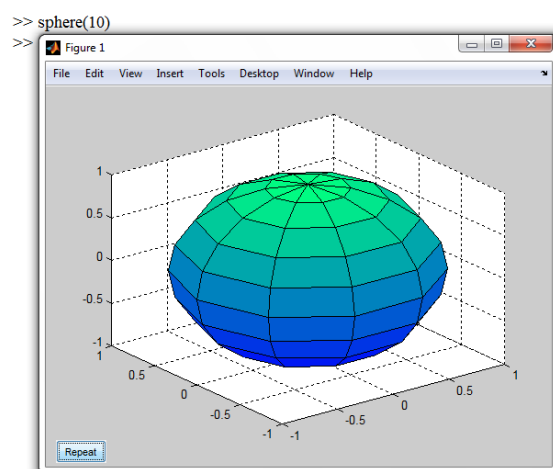
۲۰.۸ نمودارهای خاص

۱۰.۲۰.۸ کره

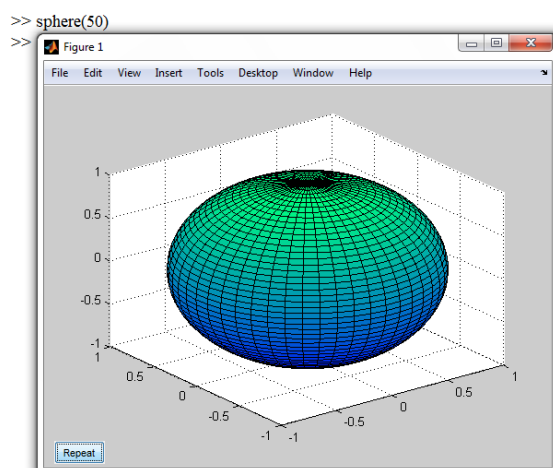
برای رسم کره از دستور sphere استفاده می‌شود. به صورت پیش فرض دستور sphere شکلی متشکل از ۲۰ بند تولید می‌کند. اما شما می‌توانید خودتان تعداد بند دلخواهتان را به دستور داده و شکل را به کره کامل نزدیک‌تر کنید و یا آن را به یک چند وجهی مشخص تبدیل نمایید. همچنین شما می‌توانید دستور را با یک آرایه سه درایه‌ای از متغیرهای موردنظر خود مساوی قرار داده و سپس با استفاده از این متغیرها، کره را با هر کدام از دستوراتی که از سری دستورات رسم رویه می‌شناسید، رسم کنید و یا حتی از این متغیرها برای رسم شکل‌های متفاوت بهره بگیرید. این دستور به تنهایی برای رسم یک کره از نوع رویه‌های طراحی شده است.



شکل ۲۶.۸: دستور sphere

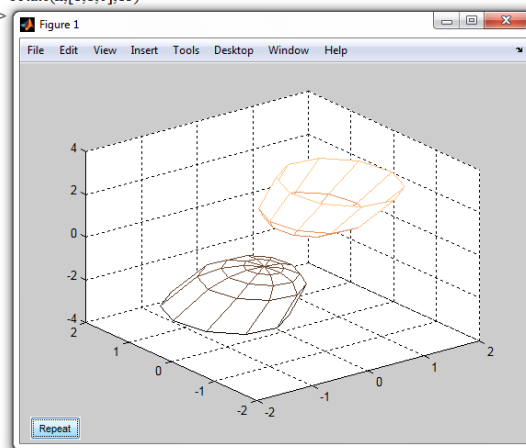


شکل ۲۷.۸: دستور `sphere(۱۰)`



شکل ۲۸.۸: دستور `sphere(۵۰)`

```
>> [x,y,z]=sphere(10);
>> h=mesh(x,y,1./z);
>> colormap copper
>> rotate(h,[1,1,0],15)
>>
```



شکل ۲۹.۸: دستور rotate

بدنیست در اینجا علاوه بر اینکه رسم یک نمودار متفاوت با استفاده از متغیرهای یک کره را می بینید، شما را با دستور چرخش^۲ نیز آشنا کنیم. شما می توانید به نموداری که می خواهید رسم کنید یک نام اختصاص دهید و با دادن این نام به تابع، چرخش نمودارتان را حول خطی در امتداد بردار مورد نظرتان به اندازه درجه دلخواه بچرخانید. در مثال فوق نمودارمان را h نامیدیم و حول امتداد بردار $[1, 1, 0]$ به اندازه ۱۵ درجه منحرف ساختیم.

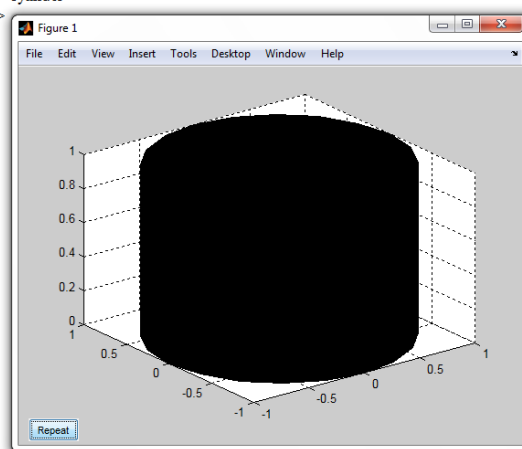
۲.۲.۸ استوانه

در مورد استوانه دستور cylinder استفاده می شود که به تنهایی یک رویه استوانه ای با شعاع یک و ۲۰ وجه و ارتفاع ۱ بوجود می آورد. اگر شما یک ورودی به تابع بدهید با آن به عنوان شعاع استوانه برخورد می کند، که می تواند ثابت یا متغیر باشد. اگر دو ورودی دهید، ورودی اول را شعاع و ورودی دوم را تعداد وجه در نظر می گیرد. همچنین می توانید استوانه را مساوی با یک آرایه سه درایه ای از نام متغیرهای دلخواه قرار داده و با استفاده از مقادیر ذخیره شده در متغیرها نمودارهای گوناگونی رسم کنید.

^۲Rotate

```
>> cylinder
```

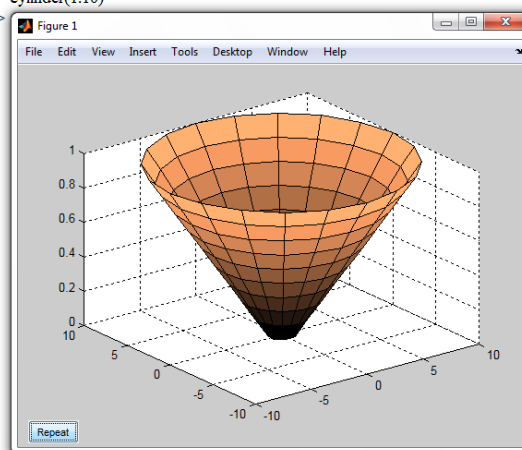
```
>>
```



شکل ۳۰.۸: دستور cylinder

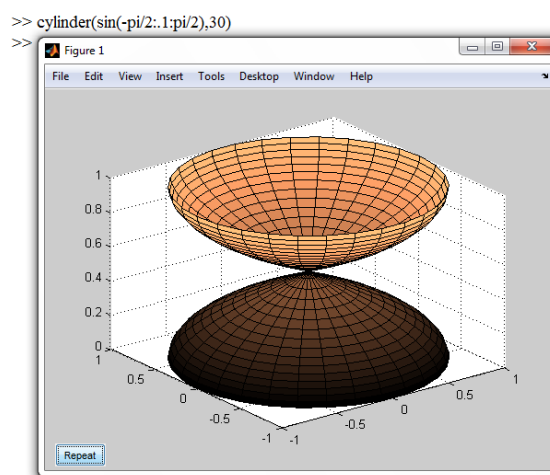
```
>> cylinder(1:10)
```

```
>>
```

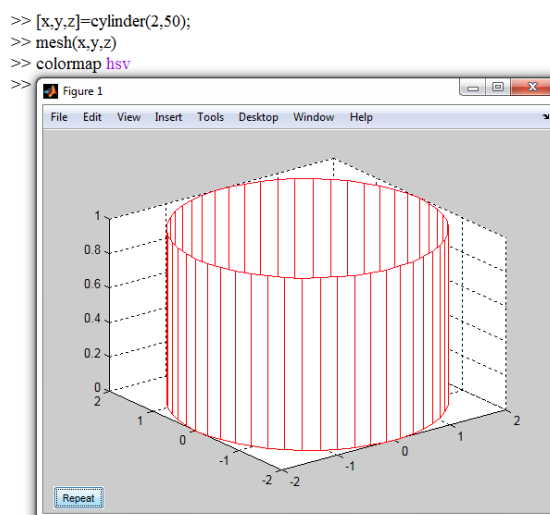


شکل ۳۱.۸: دستور cylinder

فصل ۸. ترسیم دو بعدی و سه بعدی در متلب

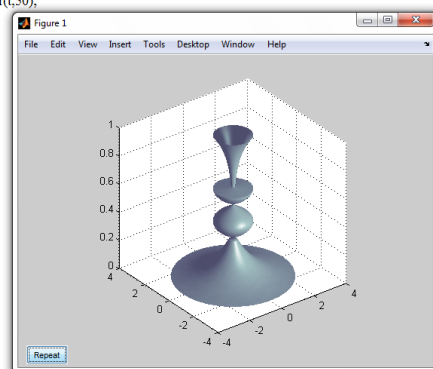


شکل ۳۲.۸: دستور cylinder



شکل ۳۳.۸: دستور cylinder


```
>> t=[tan(2*pi/5:-0.03:0.2) sin(-pi:1*pi/2) logspace(-1,0,50)];
>> [x,y,z]=cylinder(t,50);
>> surf(x,y,z)
>> shading interp
>> colormap bone
>> axis square
>>
```



شکل ۳۴.۸: دستور cylinder

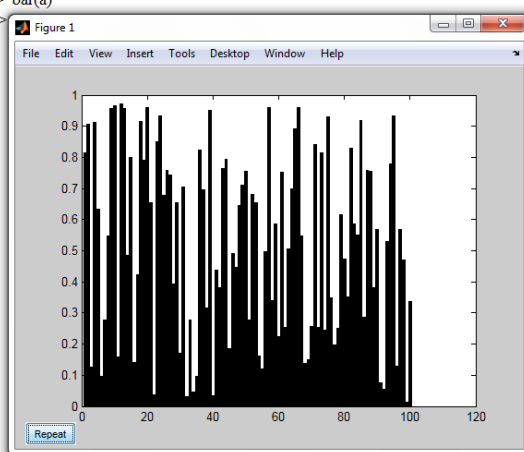
می‌بینید که با دادن شعاع متغیر به استوانه می‌توان اشکال زیبایی آفرید. توجه کنید که خود نرم‌افزار طوری دایره‌های حاصل از مقادیر متغیر اندازه شعاع را در کنار هم قرار می‌دهد که اولین شعاع در ارتفاع صفر و آخرین شعاع در ارتفاع ۱ قرار گیرد و بقیه حلقه‌ها با فواصل یکسان این بین قرار می‌گیرند.

۳.۸ نمودارهای آماری

۱.۳.۸ نمودار میله‌ای

دستور bar نمودار میله‌ای یک مجموعه را رسم می‌کند.

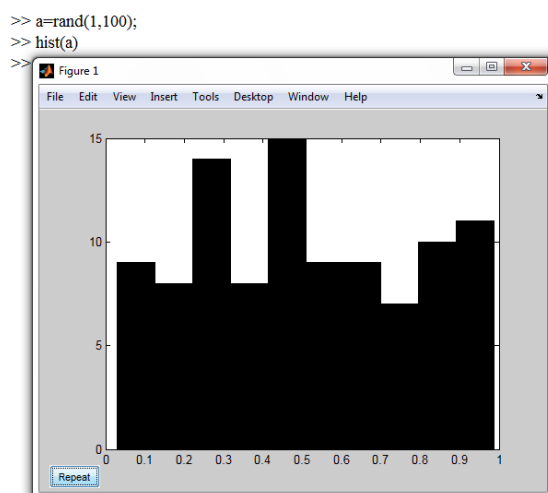
```
>> a=rand(1,100);
>> bar(a)
>>
```



شکل ۳۵.۸: دستور bar

۲.۳.۸ نمودار فراوانی

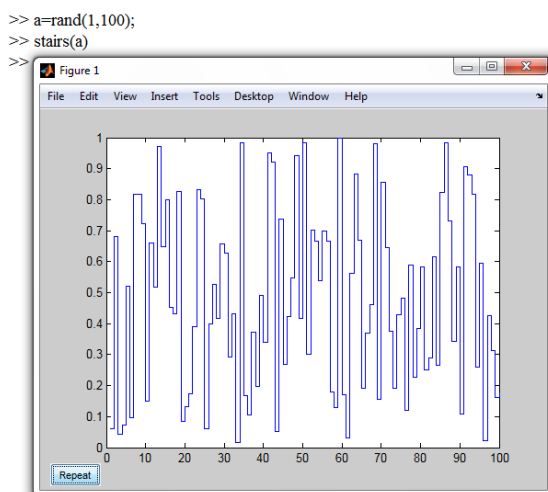
دستور hist نمودار هیستوگرام مربوط به یک مجموعه را رسم می‌کند.



شکل ۳۶.۸: دستور hist

۳.۳.۸ نمودار پله‌ای

دستور stairs نمودار پله‌ای یک مجموعه را رسم می‌کند.



شکل ۳۷.۸: دستور stairs