



INFORMATIKA II

MATLAB 4. deo

Rudarsko-geološki fakultet
Rudarski odsek



Vrste grafikona

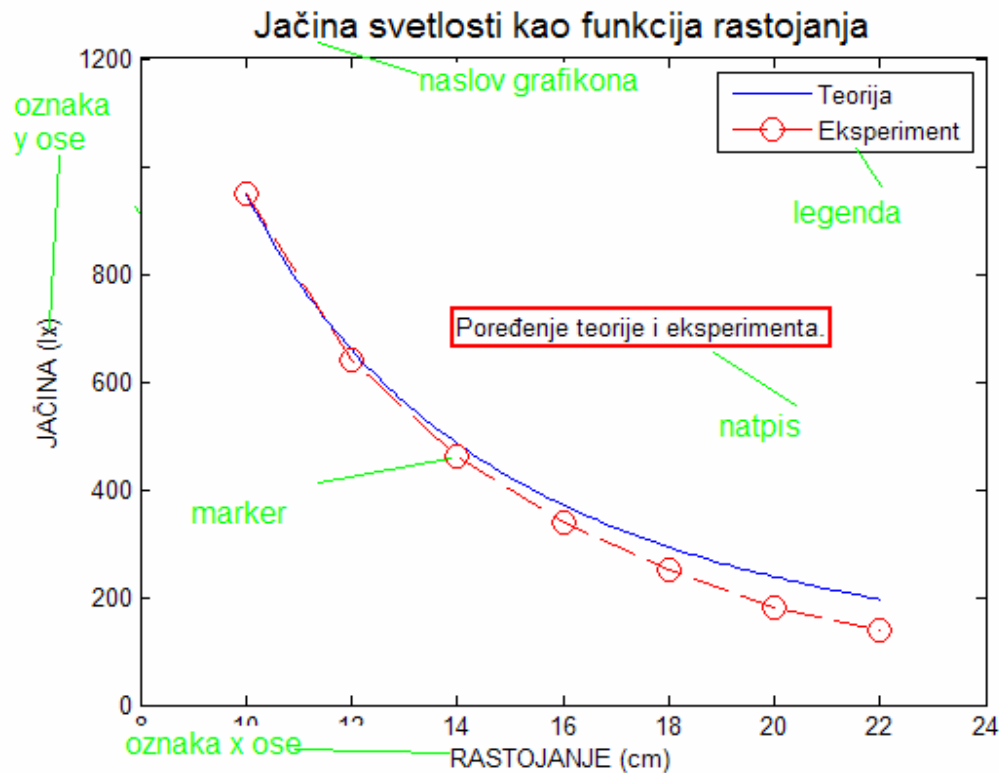
- Grafikoni ili grafici (*plots*) su veoma korisni za predstavljanje podataka u svim oblastima, ali naročito u nauci i tehnici, gde se MATLAB najviše upotrebljava
- MATLAB ima različite komande pomoću kojih se mogu nacrtati različite vrste grafikona:
 - s linearnom podelom na osama
 - s logaritamskom i semilogaritamskom podelom na osama
 - trakastih i stepenastih
 - polarnih
 - trodimenzionalnih konturnih i mrežastih
 - i još mnogo drugih



Formatiranje grafikona

- Izgled grafikona se može podesiti formatiranjem
- Može se izabrati boja, debljina i vrsta linije (puna, isprekidana itd.), mogu se dodati markeri tačkaka i linije mreže, naslovi i komentari
- Na istom grafikonu može biti nacrtano više krivih
- Na istu stranicu može se smestiti više grafikona
- Grafikonu se može dodati i legenda (na primer, ako sadrži više krivih)

Primer jednostavnog grafikona



- Teorijski dobijena kriva iscrtana je punom linijom a eksperimentalno dobijene tačke označene kružićem i spojene crvenom isprekidanom linijom
- Grafikon je formatiran tako da ima naslov, oznake osa, legendu, markere i uokviren natpis



Crtanje dvodimenzionalnih grafikona

- Za crtanje dvodimenzionalnih grafikona služi komanda `plot`
- Najjednostavniji oblik te komande glasi **`plot(x,y)`** pri čemu su argumenti `x` i `y` vektori (jednodimenzionalni nizovi) koji moraju imati isti broj elemenata
- Izvršavanjem komande `plot` nastaje slika u grafičkom prozoru (*Figure*)
- Ako grafički prozor već nije otvoren on se automatski otvara nakon izvršenja komande
- Poput ostalih komandi, i komanda `plot` može biti napisana u komandnom prozoru ili u skript datoteci



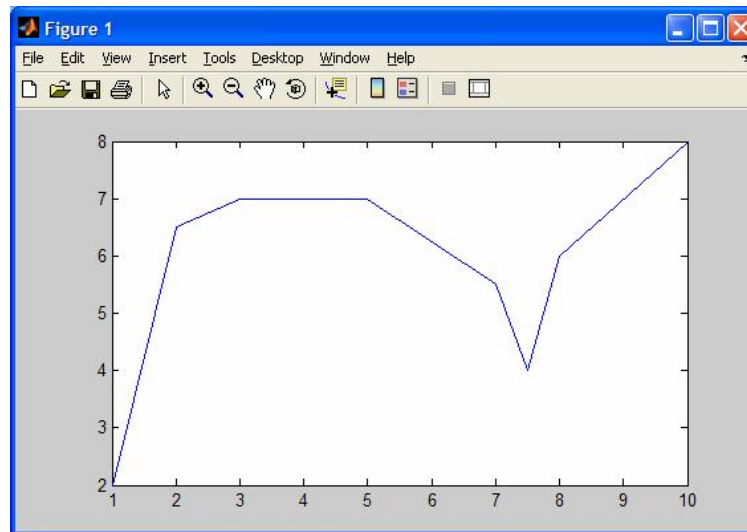
Interpretacija vektora x i y

- Slika sadrži jednu krivu sa x vrednostima na apscisi (horizontalnoj osi) i odgovarajućim y vrednostima na ordinati (vertikalnoj osi)
- Kriva se sastoji od pravolinijskih segmenata koji povezuju tačke čije su koordinate definisane elementima vektora x i y .
- Vektori mogu imati proizvoljna imena
- Za horizontalnu osu upotrebljava se vektor koji je naveden kao prvi argument u zagradi komande *plot*, a za vertikalnu drugi argument
- Kreirana slika ima ose s linearnom podelom i podrazumevan opseg

Primer komande plot

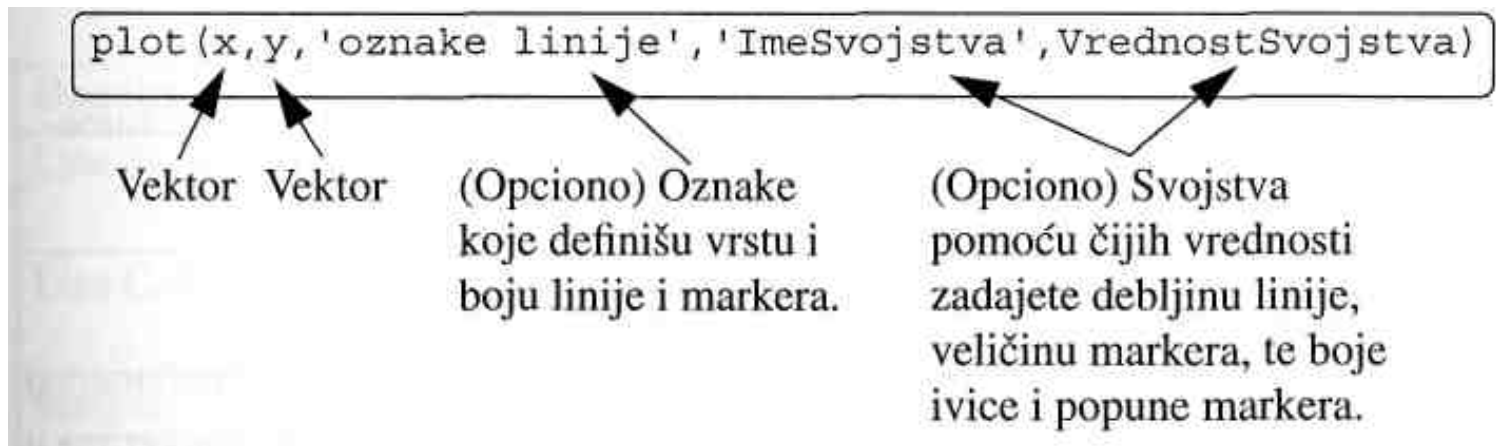
```
Command Window
>>
>> x = [1, 2, 3, 5, 7, 7.5, 8, 10];
>> y = [2, 6.5, 7, 7, 5.5, 4, 6, 8];
>> plot(x,y)
>> |
```

Nakon izvršenja komande *plot*, otvara se grafički prozor i prikazuje iscrtani grafikon, na kome je kriva plava, što je podrazumevana boja linije



Opcioni argumenti komande plot

- Komanda *plot* ima opcione argumente za podešavanje izgleda linije i markera
- Sa opcionim argumentima komanda *plot* ima oblik:





Oznake linije

Oznake su opcione i pomoću njih se definišu vrsta i boja linije, kao i vrsta markera

Vrsta linije	Oznaka
Puna (podrazumevana)	-
Isprekidana	--
Tačkasta	
Crta-tačka	-.

Vrsta markera	Oznaka
znak plus	+
Kružić	o
Zvezdica	*
Tačka	
Kvadrat	s
Romb	d
petokraka zvezda	P
šestokraka zvezda	h

Boja linije	Oznaka
Crvena	r
Zelena	g
Plava	b
Cijan	c
Magenta	m
Žuta	y
Crna	k
Bela	w



Korišćenje oznaka linije

- Oznaka linija se piše kao znakovni niz unutar komande *plot*, pri čemu je redosled oznaka proizvoljan
- Kako su oznake linije opcione to komanda *plot* može da bude i bez oznaka ili da sadrži jednu, dve ili sve tri oznake
- Primeri:
 - `plot(x,y)` Puna plava linija povezuje tačke bez markera (podrazumevano)
 - `plot(x,y,'r')` Tačke povezuje puna crvena linija
 - `plot(x,y,'--y')` Tačke povezuje žuta isprekidana linija
 - `plot(x,y,'*')` Tačke su označene zvezdicama *, (a nisu povezane linijama)
 - `plot(x,y,'g:d')` Zelena tačkasta linija povezuje tačke označene markerima u obliku romba



Ime svojstva i vrednost svojstva

- Svojstva su opciona i služe za zadavanje debljine linije, veličine markera, boja ivica i popune markera
- Ime svojstva se piše kao znakovni niz iza kojeg sledi zarez i vrednost svojstva, sve to unutar komande *plot*
- Četiri svojstva i njihove moguće vrednosti su:

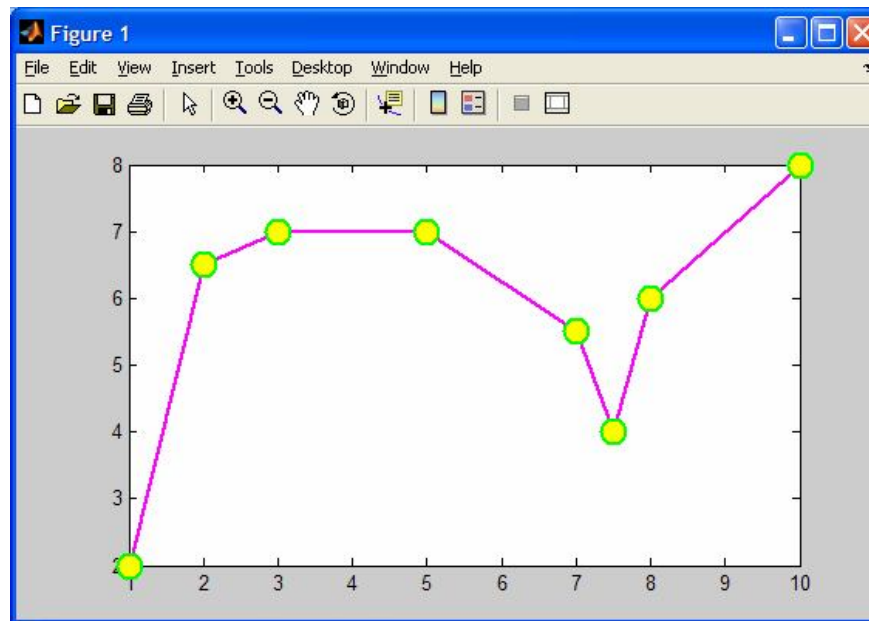
Ime svojstva	Opis	Moguće vrednosti svojstva
LineWidth (ili linewidth)	Zadaje debljinu linije	Broj izražen u tačkama (podrazumevano 0.5)
MarkerSize (ili markersize)	Zadaje veličinu markera	Broj izražen u tačkama
MarkerEdgeColor (ili markeredgecolor)	Zadaje boju markera ili boju ivice za popunjene markere	Oznake boja iz tabele za boje linija napisane kao znakovni niz
MarkerFaceColor (ili markerfacecolor)	Zadaje boju popune za popunjene markere	Oznake boja iz tabele za boje linija napisane kao znakovni niz


Primer

- Komanda:

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

crtá grafikon u kome su tačke povezane punom magenta linijom i označene markerima u obliku kružića, debljina linije je dve tačke, a veličina markera 12 tačaka, pri čemu markeri imaju zelene ivice i žutu popunu





Zadavanje vrste i boje linije, kao i oblika markera kao svojstava

- Pomoću argumenata `ImeSvojstva` i `VrednostSvojstva` mogu se takođe zadati i vrsta i boja linije, kao i oblik markera:

Oznaka	Ime svojstva	Moguće vrednosti svojstva
Line Style	<code>linestyle</code> (ili <code>LineStyle</code>)	Oznaka vrste linije iz navedene tabele, napisana kao znakovni niz
Line Color	<code>color</code> (ili <code>Color</code>)	Oznaka boje linije iz navedene tabele, napisana kao znakovni niz
Marker	<code>marker</code> (ili <code>Marker</code>)	Oznaka oblika markera iz navedene tabele, napisana kao znakovni niz

Primer

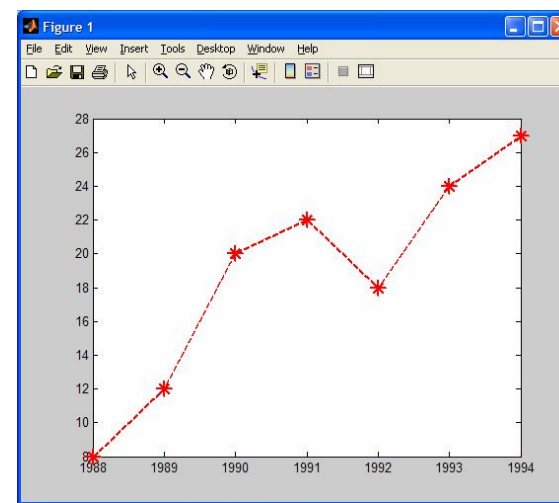
- Tabela u ovom primeru sadrži podatke nekog preduzeća o prodaji za godine od 1988. do 1994
- Za grafikon datih podataka generisati vektore i potom ih upotrebiti u komandi *plot*

GODINA	1988	1989	1990	1991	1992	1993	1994
PRODAJA (u milionima)	8	12	20	22	18	24	27

- Da bi se napravio grafikon od ovih podataka, spisak godina se dodeli jednom vektoru (*god*), a odgovarajući podaci o prodaji drugom vektoru (*pro*)
- Na slici levo je prikazan komandni prozor u kome su komande koje generišu vektore kao i komanda *plot* a na slici desno grafički prozor koji će se otvoriti kada se izvrši zadata komanda *plot*

```
Command Window
>>
>> god=[1988:1:1994];
>> pro=[8 12 20 22 18 24 27];
>> % oznake linija: isprekidana crvena linija i zvezdica marker
>> plot(god,pro,'--r*','linewidth',2,'markersize',12)
>>
```

Command Window Command History





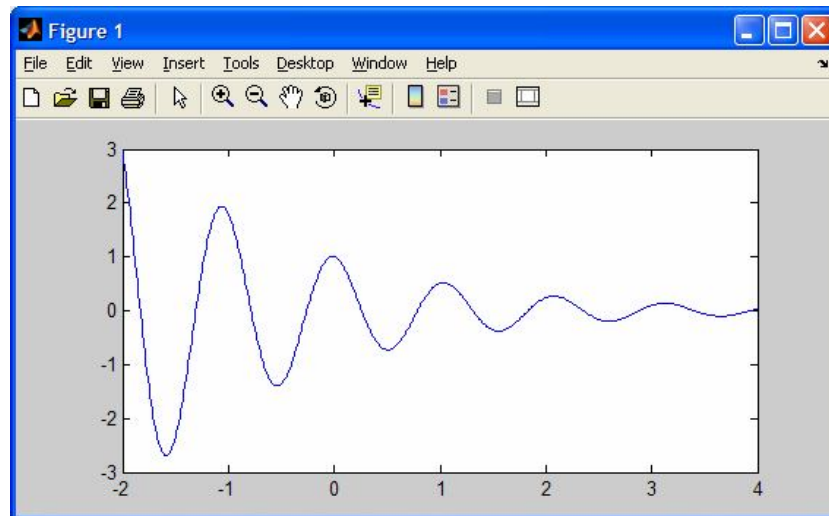
Grafik funkcije

- Veoma često postoji potreba da se nacrtaju grafici nekih funkcija
- U MATLAB-u se funkcije crtaju komandama *plot* i *fplot*
- Da bi se komandom *plot* nacrtao grafikon funkcije $y = f(x)$, prvo se generiše vektor vrednosti x za interval u kome je funkcija definisana, sa određenim korakom
- Zatim se generiše vektor y odgovarajućih vrednosti $f(x)$ izvršavanjem operacija nad pojedinačnim elementima vektora x
- Kada se definišu oba vektora, može se nacrtati grafikon komandom *plot*

Primer

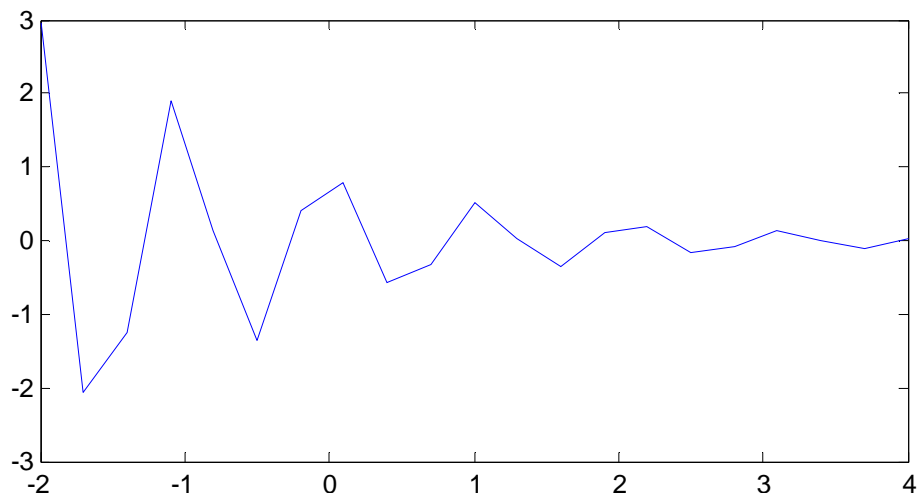
- Komandom *plot* nacrtati grafik funkcije $y=3.5^{-0.5x}\cos(6x)$ u intervalu $-2 \leq x \leq 4$ sa korakom 0.01
- Program za iscrtavanje te funkcije dat je u skript datoteci na slici levo a grafik na slici desno

```
Editor - C:\Program Files\MATLAB\R2006b\work\MLG_Primer1.m
File Edit Text Go Cell Tools Debug Desktop Window Help
1 % Skript datoteka koja crta grafik funkcije
2 x=[-2:0.01:4];
3 y=3.5.^(-0.5*x).*cos(6*x);
4 plot(x,y)
```



Uticaj odabranog koraka na grafik

- Pošto se grafik sastoji od pravolinijskih segmenata koji povezuju tačke, izbor razmaka između elemenata vektora x (korak) utiče na izgled grafika funkcije
- Za funkcije koje se brže menjaju potreban je manji razmak
- U prethodnom primeru je zadati razmak bio 0.01 ali kada se ista funkcija iscrta u istom intervalu domenu uz mnogo veći korak (na primer 0.3) dobija se izobličen grafik funkcije kao na slici
- Grafik na slici nije prikazan u grafičkom prozoru, jer se grafik može kopirati iz grafičkog prozora (u meniju *Edit* odabere se *Copy Figure*) i preneti u druge programe





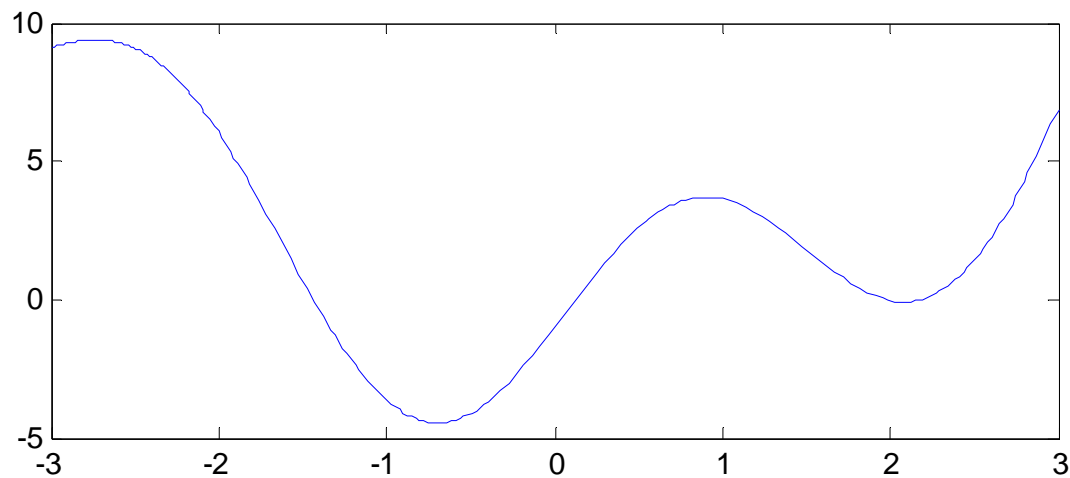
Komanda `fplot`

- Ova komanda iscrtava funkciju oblika $y = f(x)$ u datim granicama x i (opciono) y ose i ima oblik:
`fplot('funkcija',granice,oznake linije)`
gde je 'funkcija' analitički izraz za funkciju upisan neposredno u komandu kao znakovni niz, na primer, ' $8*x^2+5*\cos(x)$ '
- Funkcija za iscrtavanje može sadržati sve MATLAB-ove ugrađene funkcije
- Argument funkcije koju treba iscrtati može biti označen bilo kojim slovom, pa tako na primer, funkcija navedena kao primer može se napisati kao: ' $8*z^2+5*\cos(z)$ ' ili ' $8*t^2+5*\cos(t)$ '
- Funkcija ne sme da sadrži prethodno definisane promenljive, pa tako, na primer, u gornjoj funkciji broj 8 nije moguće dodeliti određenoj promenljivoj i zatim tu promenljivu upotrebiti za definisanje funkcije koju *fplot* treba da iscrtava
- Granice su vektor $[xmin, xmax]$ od 2 elementa koji zadaju domen x (granice iscrtavanja na x osi) ili vektor sa četiri elementa $[xmin,xmax,ymin,ymax]$ koji zadaju i domen x i domen y (granice iscrtavanja na x i y osi)
- Oznake linije su iste kao za komandu *plot*

Primer

- Nacrtati grafik funkcije $y=x^2+4\sin(2x)-1$ u granicama $-3 < x < 3$
- Komanda *fplot* za crtanje ove funkcije je:

```
>> fplot('x^2+4*sin(2*x)-1',[-3 3])
```
- Grafik koji se dobija u grafičkom prozoru prikazan je na slici





Grafikon sa više krivih

- U mnogim situacijama treba nacrtati više krivih na istom grafikonu
- Postoje tri načina za crtanje više krivih na istoj slici: prvi je upotreba komande *plot*, drugi upotreba komandi *hold on*, *hold off*, a treći upotreba komande *line*
- Komandom *plot* se može nacrtati onoliko krivih koliko se parova vektora upiše u komandu, s tim što vektori svakog para moraju biti iste dužine
- Tako, na primer, komanda:
plot(x,y,u,v,t,h)
crta tri krive: y u zavisnosti od x , v u zavisnosti od u , i h u zavisnosti od t , sve na istom grafikonu
- MATLAB automatski crta krive različitim bojama da bi se mogle lakše razaznati
- Posle svakog para vektora može se dodati parametri linije tog grafika za tu krivu
- Na primer, komanda:
plot(x,y,'-b',u,v,'--r',t,h,'g:')
crta $y(x)$ punom plavom linijom, $v(u)$ isprekidanom crvenom linijom, i $h(t)$ tačkastom zelenom linijom

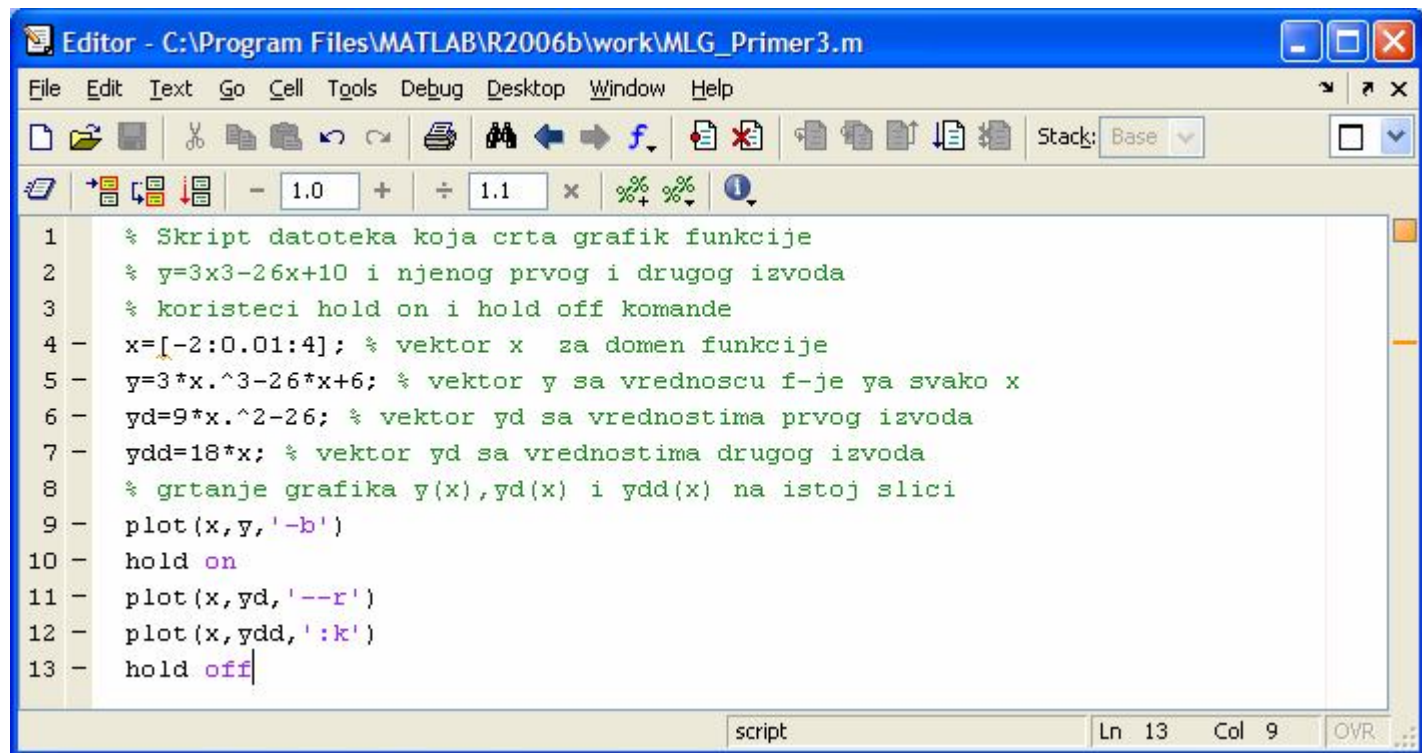


Komande *hold on* i *hold off*

- Da bi se pomoću komandi *hold on* i *hold off* nacrtalo više grafika, prvo treba nacrtati jedan grafik komandom *plot*
- Zatim se upiše komanda *hold on*, čime se ostavlja otvoren grafički prozor s prvim grafikom, zajedno sa svojstvima osa i formatiranjem, ako ih je bilo
- Naredna komanda *plot* nacrtat će novi grafik na istoj (otvorenoj) slici
- Sve dodatne komande *plot* također će nacrtati grafike koji se dodaju istoj slici
- Postupak dodavanja grafika se prekida komandom *hold off*
- Njome se MATLAB vraća u podrazumevani režim rada u kojem komanda *plot* briše prethodni grafik, pri čemu se vraćaju i podrazumevana svojstva osa

Primer

- U skript datoteci na slici prikazano je crtanje grafika funkcije i njenih izvoda iz prethodnog primera pomoću komandi *hold on* i *hold off*



```
Editor - C:\Program Files\MATLAB\R2006b\work\MLG_Primer3.m
File Edit Text Go Cell Tools Debug Desktop Window Help
[Icons] Stack: Base
[Icons] - 1.0 + ÷ 1.1 x %> %> ⓘ
1 % Skript datoteka koja crta grafik funkcije
2 % y=3x3-26x+10 i njenog prvog i drugog izvoda
3 % koristeci hold on i hold off komande
4 - x=[-2:0.01:4]; % vektor x za domen funkcije
5 - y=3*x.^3-26*x+6; % vektor y sa vrednoscu f-je za svako x
6 - yd=9*x.^2-26; % vektor yd sa vrednostima prvog izvoda
7 - ydd=18*x; % vektor yd sa vrednostima drugog izvoda
8 % grtanje grafika y(x),yd(x) i ydd(x) na istoj slici
9 - plot(x,y,'-b')
10 - hold on
11 - plot(x,yd,'--r')
12 - plot(x,ydd,':k')
13 - hold off
script Ln 13 Col 9 OVR
```

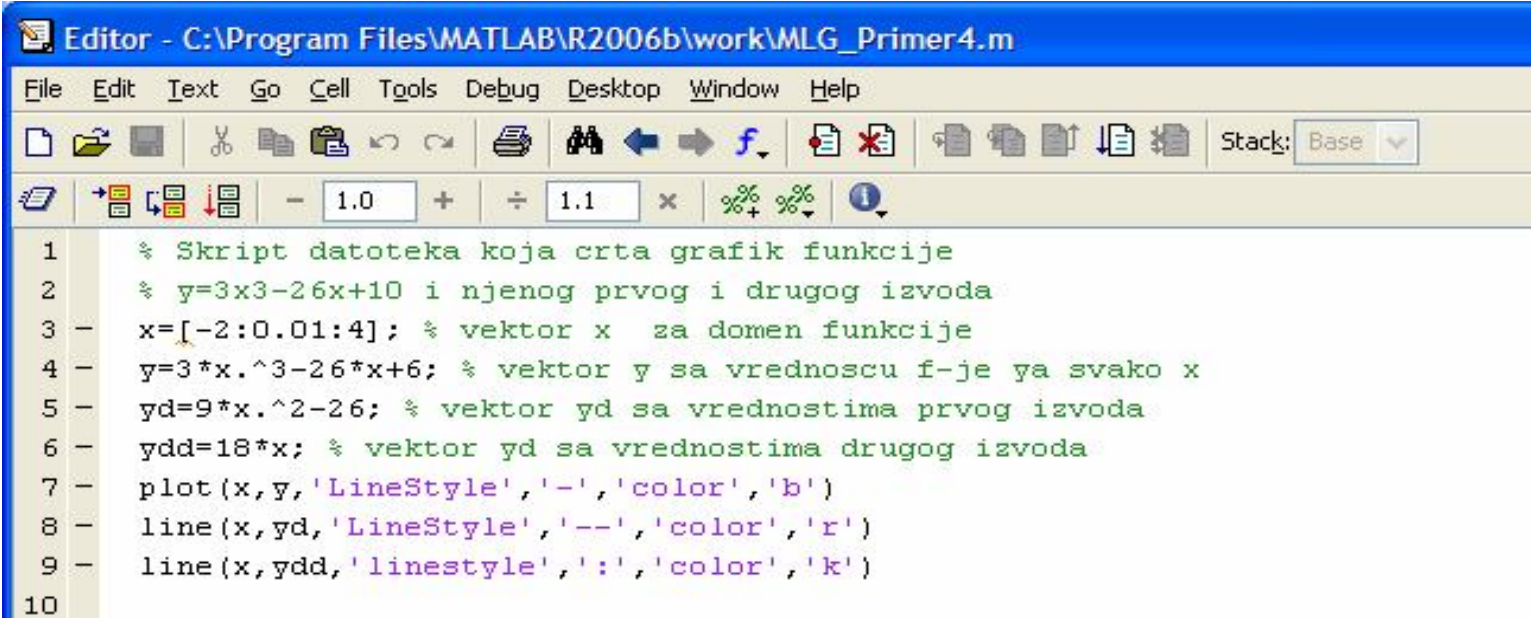


Komanda line

- Ovom komandom se dodaju grafici na postojeći grafikon an njen oblik je:
line(x,y,'ImeSvojstva',VrednostSvojstva)
- Format komande *line* gotovo je jednak formatu komande *plot*, s tim što komanda *line* ne sadrži oznaku linije, već se stil, boja i marker linije mogu zadati pomoću imena svojstva i odgovarajućih vrednosti
- Svojstva pomoću kojih se mogu debljina linije, vrsta, veličina i boje ivice i popune markera su opciona i ukoliko nijedno nije zadato, MATLAB koristi podrazumevana svojstva i vrednosti
- Na primer, komanda:
line(x,y,'linestyle','--','color','r','marker','o')
postojećem grafiku dodaje isprekidanu crvenu liniju s kružnim markerima
- Glavna razlika između komandi *plot* i *line* jeste u tome što *plot* počinje nov grafikon svaki put kada se pokrene, dok *line* dodaje linije postojećem grafikonu
- Da bi se dobio grafikon sa više grafika, uvek se prvo komandom *plot* nacrtava prvi grafik, a zatim mu se komandama *line* dodaju novi grafici - ako se komanda *line* izda pre komande *plot*, pojaviće se poruka o grešci!

Primer

- U skript datoteci na slici prikazano je crtanje grafika funkcije i njenih izvoda iz prethodnog primera pomoću komandi *plot* i *line*



```
Editor - C:\Program Files\MATLAB\R2006b\work\MLG_Primer4.m
File Edit Text Go Cell Tools Debug Desktop Window Help
[Icons] Stack: Base
[Icons] - 1.0 + ÷ 1.1 x % % !
1 % Skript datoteka koja crta grafik funkcije
2 % y=3x^3-26x+10 i njenog prvog i drugog izvoda
3 - x=[-2:0.01:4]; % vektor x za domen funkcije
4 - y=3*x.^3-26*x+6; % vektor y sa vrednoscu f-je ya svako x
5 - yd=9*x.^2-26; % vektor yd sa vrednostima prvog izvoda
6 - ydd=18*x; % vektor yd sa vrednostima drugog izvoda
7 - plot(x,y,'LineStyle','-','color','b')
8 - line(x,yd,'LineStyle','--','color','r')
9 - line(x,ydd,'linestyle',':','color','k')
10
```



Samostalan rad

- Nacrtati dva zasebna grafika funkcije $f(x) = 0.6x^5 - 5x^3 + 9x + 2$; jedan za interval $-4 \leq x \leq 4$, a drugi za $-2.7 \leq x \leq 2.7$
- Nacrtati funkciju $f(x) = \frac{x^2 - x + 1}{x^2 + x + 1}$ za $-10 \leq x \leq 10$
- Komandom `fplot` nacrtati funkciju:
 $f(x) = 0.01x^5 - 0.03x^4 + 0.4x^3 - 2x^2 - 6x + 5$
u domenu $-4 \leq x \leq 6$



Formatiranje grafikona

- Sliku koja sadrži grafikon često treba formatirati da bi dobila odgovarajući izgled i prikazala još neke podatke sem krive
- Grafikonu se mogu dodati oznake osa, naslov grafikona, legenda, linije mreže, opseg namenskih osa i natpisi
- Grafikoni se mogu formatirati MATLAB-ovim komandama koje slede za komandama *plot* i *fplot* ili interaktivno - pomoću editora grafikona u grafičkom prozoru
- Prvi način je pogodan kada je komanda *plot* uključena u skript datoteku
- Ako program u skript datoteci sadrži komande za formatiranje, napraviće formatiran grafikon svaki put kada se izvrši
- Formatiranje pomoću editora grafikona u grafičkom prozoru obavljeno nakon izrade grafikona važi samo za taj grafikon i moraće da se ponovi kada se napravi sledeći



Formatiranje grafikona pomoću komandi `xlabel`, `ylabel`, `title`, `text` i `gtext`

- Komande za formatiranje se unose posle komandi `plot` i `fplot` i ovde će biti navedene neke od najznačajnijih
- Komande `xlabel` i `ylabel` služe za postavljanje natpisa na x i y osu:
`xlabel('tekst kao znakovni niz')` `ylabel('tekst kao znakovni niz')`
- Komandom `title` se grafikonu dodaje naslov:
`title('tekst kao znakovni niz')`
- Pomoću komandi `text` i `gtext` na određeno mesto na grafikonu se može smestiti natpis:
`text(x,y,'tekst kao znakovni niz')` `gtext('tekst kao znakovni niz')`
- Komanda `text` smešta tekst na sliku tako da njegov prvi znak dođe na mesto čije su koordinate x i y (prema osama grafikona), a komanda `gtext` smešta tekst na mesto koje korisnik naknadno odredi, jer se nakon izvršenja te komande otvara grafički prozor u kome korisnik mišem zadaje mesto natpisa



Formatiranje grafikona pomoću komande `legend`

- Komanda `legend` postavlja legendu na grafikon koja prikazuje uzorak linije kojom je iscrtan svaki grafik i pored njega natpis koji zadaje korisnika a njen je oblik:

`legend('znakovni niz1','znakovni niz2',...,pol)`

- Znakovni niz je natpis koji se postavlja pored uzorka linije pri čemu redosled natpisa odgovara redosledu izrade grafika
- Parametar `pol` je opcioni broj kojim se zadaje položaj legende na slici i neke od opcija su:
 - `pol=1` izvan granica osa, na desnu stranu
 - `pol=0` unutar granica osa, na mesto koje najmanje smeta graphicima



Formatiranje teksta u komandama `xlabel`, `vlabel`, `title`, `text` i `legend`

- Tekst znakovnog niza uključenog u navedene komande, koji se prikazuje kada se komande izvrše se može formatirati, odnosno zadati font, veličina, položaj (eksponent, indeks), stil (kurziv, polucrn itd.), boja znakova, boja pozadine i mnogo drugih pojedinosti prikaza
- Tekst se formatira dodavanjem modifikatora u znakovne nizove ili opcionih argumenata `ImeSvojstva` i `VrednostSvojstva` iza znakovnog niza
- Modifikatori su specijalne sekvence znakova umetnute u znakovne nizove

Modifikator	Efekat
<code>\bf</code>	polucrni font
<code>\it</code>	kurziv
<code>\rm</code>	običan font
<code>\fontname{ime fonta}</code>	koristi se zadati font
<code>\fontsize{veličina fonta}</code>	koristi se zadata veličina



Modifikatori

- Modifikator utiče na izgled teksta od mesta na koje je umetnut do kraja znakovnog niza
- Uticaj se može ograničiti na deo znakovnog niza ukoliko se modifikator i tekst na koji on treba da utiče upišu unutar vitičastih zagrada: { }
- Ako se ispred znaka upiše `_` (podvlaka, *underscore*) ili `^` (kapica, *caret*), taj znak će biti prikazan u indeksu, odnosno u eksponentu
- Da bi se više znakova ispisalo u indeksu, odnosno u eksponentu, treba ih upisati unutar vitičastih zagrada, { }, koje slede neposredno iza `_` ili `^`
- Grčko slovo se može uključiti u tekst ukoliko se unutar znakovnog niza iza znaka `\` upiše odgovarajuće ime slova
- Ako se ime slova upiše malim slovima, ispisaće se malo grčko slovo a ukoliko se ime slova započne velikim slovom, ispisaće se veliko grčko slovo

Ime slova u znakovnom nizu	Grčko slovo
<code>\alpha</code>	α
<code>\beta</code>	β
<code>\gamma</code>	γ
<code>\theta</code>	θ
<code>\pi</code>	π
<code>\sigma</code>	σ

Dodatne mogućnosti formatiranja teksta

- Tekst prikazan pomoću komandi *xlabel*, *ylabel*, *title* i *text* se može formatirati i dodavanjem opcionih argumenata *imeSvojstva* i *VrednostSvojstva* iza znakovnog niza u kom slučaju bi, na primer, komanda *text* imala oblik:
text(x,y,'tekst kao znakovni niz', ImeSvojstva,VrednostSvojstva)
- Na isti način se dodaju argumenti *imeSvojstva* i *VrednostSvojstva* u ostale tri komande, pri čemu se *imeSvojstva* piše kao znakovni niz, a *VrednostSvojstva* kao broj ukoliko je vrednost svojstva broj, odnosno kao znakovni niz ako je vrednost svojstva reč ili slovo

Ime svojstva	Opis	Moguće vrednosti svojstva
Rotation	Zadaje orijentaciju teksta	Skalar (stepeni); podrazumevano:0
FontAngle	Zadaje kurzivan ili normalan stil znakova	normal, italic; podrazumevano: normal
FontName	Zadaje font za tekst	Ime fonta dostupnog sistemu
FontSize	Zadaje veličinu fonta	Skalar (tačaka); podrazumevano: 10
FontWeight	Zadaje debljinu znakova	light, normal, bold; podrazumevano: normal
Color	Zadaje boju teksta	Oznake za boju
Background-Color	Zadaje boju (pravougaone) pozadine	Oznake za boju
EdgeColor	Zadaje boju ivica pravougaonog okvira oko teksta	Oznake za boju; podrazumevano: ništa
LineWidth	Zadaje širinu ivica pravougaonog okvira oko teksta	Skalar (tačaka); podrazumevano: 0.5



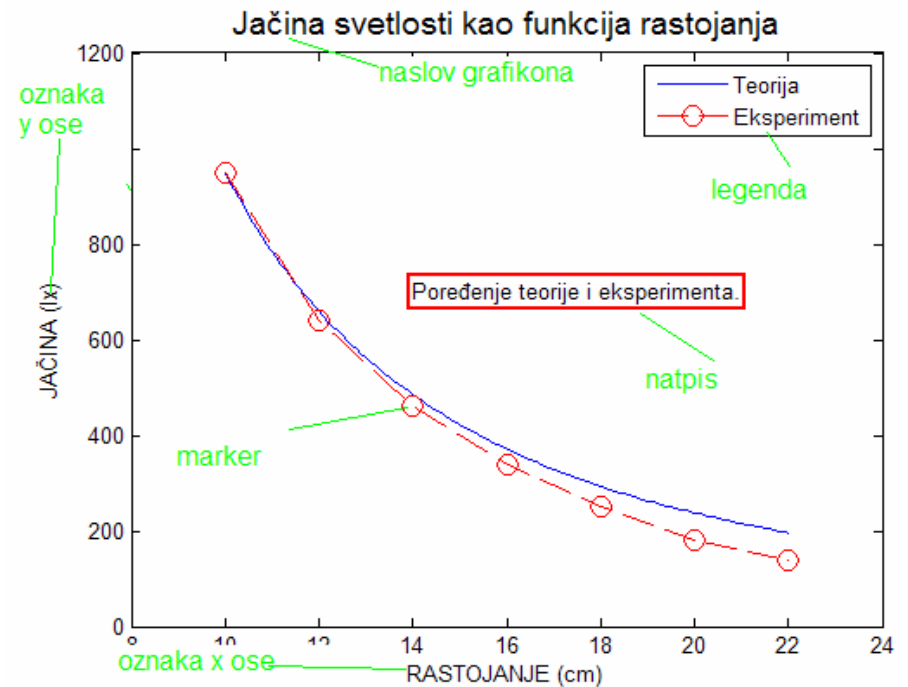
Komande axis i grid

- Kada izvrši komandu *plot(x,y)*, MATLAB crta ose čije su granice određene najmanjom i najvećom vrednošću elemenata od x i y
- Opseg i izgled osa se može promeniti komandom *axis*
- Grafikon često izgleda bolje ako se ose protežu malo van opsega podataka. Neki mogući oblici komande axis su:
 - axis([xmin,xmax,ymin,ymax])** - zadaje granice x i y ose (xmin,xmax,ymin i ymax su brojevi)
 - axis equal** - zadaje istu veličinu obe ose
 - axis square** - zadaje kvadratni oblik područja osa.
 - axis tight** - zadaje granice osa prema opsegu podataka
- Komanda *grid* služi za dodavanje i uklanjanje mreže:
 - grid on** Dodaje linije mreže na grafikon
 - grid off** Uklanja linije mreže s grafikona

Primer

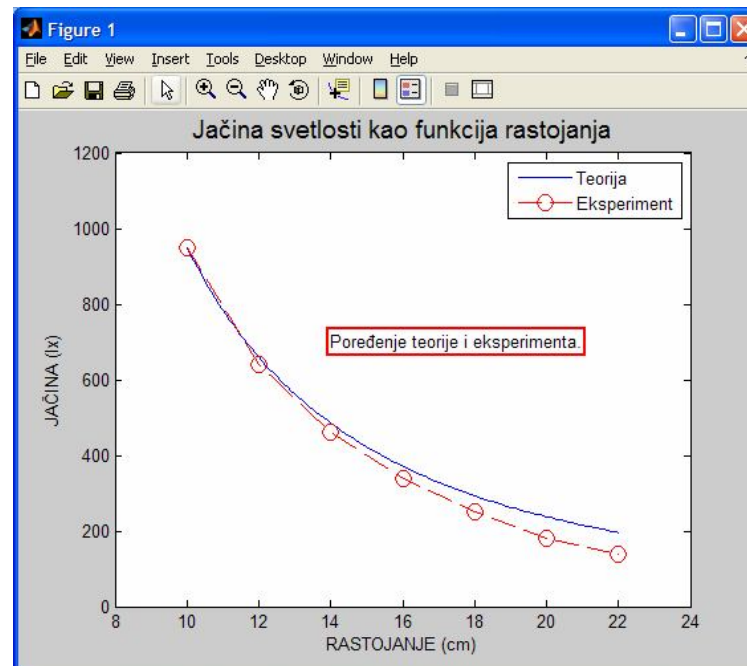
- Komande u skript datoteci za formatiranje grafikona i formatiran grafikon

```
Editor - C:\Program Files\MATLAB\R2006b\work\MLG_Primer5.m
File Edit Text Go Cell Tools Debug Desktop Window Help
1 x = [10:0.1:22];
2 y = 95000./x.^2;
3 xd = [10:2:22];
4 yd = [950 640 460 340 250 180 140];
5 plot(x,y,'-', 'LineWidth', 1.0)
6 xlabel('RASTOJANJE (cm)')
7 ylabel('JAČINA (lx)')
8 title('\fontname(Arial)Jačina svetlosti kao funkcija rastojanja','FontSize',14)
9 axis([8 24 0 1200])
10 text(14,700,'Poređenje teorije i eksperimenta.','EdgeColor','r','LineWidth',2)
11 hold on
12 plot(xd,yd,'ro--', 'linewidth',1.0, 'markersize',10)
13 legend('Teorija','Eksperiment',0)
14 hold off
```



Formatiranje grafikona u editoru grafikona

- Grafikon se može formatirati interaktivno u grafičkom prozoru kada se klikne na grafikon i/ili upotrebe meniji
- Na slici je prikazan grafički prozor i grafikon sa prethodne slike
- Pomoću editora se mogu uneti nove odrednice formata ili izmeniti postojeće





Grafikoni sa logaritamskom podelom osa

- U mnogim naučnim i tehničkim primenama, jedna ili obe ose grafikona imaju logaritamsku (log) podelu
- Logaritamska podela omogućava prikazivanje velikog opsega vrednosti, prepoznavanje karakteristika podataka i mogućih oblika matematičkih odnosa podnesnih za modelovanje podataka
- MATLAB-ove komande za izradu grafikona s logaritamskim osama su:
 - semilogy (x, y)** - crta $y(x)$ s logaritamskom podelom (osnova 10) y ose i linearnom podelom x ose
 - semilogx (x, y)** - crta $y(x)$ s logaritamskom podelom (osnova 10) x ose i linearnom podelom y ose
 - loglog (x, y)** - crta $y(x)$ s logaritamskom podelom (osnova 10) obe ose
- Opciono, komandi se mogu dodati oznake linije, te ime svojstva i vrednost svojstva, kao u komandi *plot*
- Na osi sa logaritamskom podelom nije moguće prikazati broj 0 (pošto funkcija log nije definisana u nuli) niti je moguće prikazati negativne brojeve (pošto funkcija log nije definisana za negativne argumente)



Samostalan rad

- Nacrtati funkciju $f(x) = 1.5x/(x - 4)$ za $-10 \leq x \leq 10$; obratiti pažnju na vertikalnu asimptotu funkcije u $x = 4$
Uputstvo: nacrtati funkciju tako što će se formirati dva vektora za domen x - prvi vektor (x_1) sa elementima od -10 do 3.7 , i drugi (x_2) sa elementima od 4.3 do 10 a zatim za svaki x vektor napraviti po jedan y vektor (y_1 i y_2) sa odgovarajućim vrednostima prema definiciji funkcije; pri crtanju funkcije, nacrtati dve krive na istom grafikonu $y_1(x_1)$ i $y_2(x_2)$
- Nacrtati funkciju
$$f(x) = \frac{x^2 - 5x + 10}{x^2 - 2x - 3}$$
 za $-10 \leq x \leq 10$; obratiti pažnju na to da funkcija ima dve vertikalne asimptote
Uputstvo: nacrtati funkciju tako što će se domen podeliti na tri dela: jedan od -10 do leve asimptote, drugi između asimptota i treći od desne asimptote do 10 ; zadati opseg y ose od -20 do 20
- Nacrtati funkciju $f(x) = 3x \sin(x) - 2x$ i njen izvod na istom grafikonu, za . Funkciju nacrtati punom linijom, a izvod isprekidanom. Dodati legendu i natpise na osama.

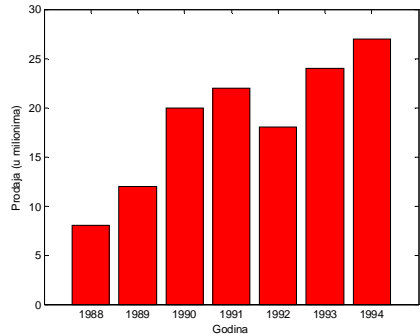
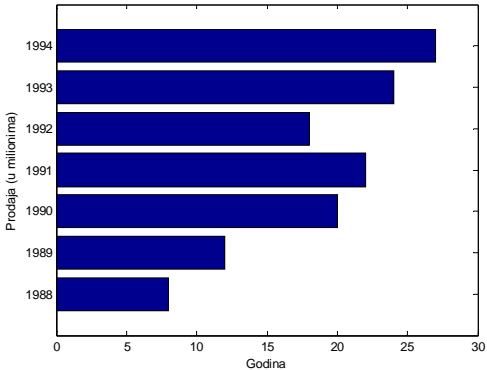


Specijalni grafikoni

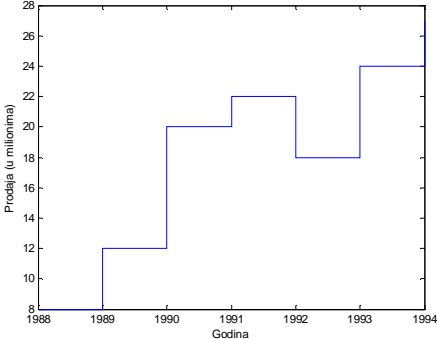
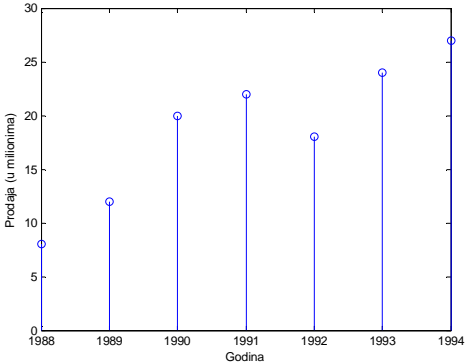
- Svi dosad predstavljeni grafikoni bili su linijski - u njima su tačke u kojima je vrednost funkcije izračunata povezane pravolinijskim segmentima
- Često grafikoni s drugačije povezanim i prikazanim tačkama bolje predstavljaju podatke
- MATLAB ima više opcija za pravljenje različitih grafikona: među njima su trakasti, stepenasti, kružni, grafikon diskretnih podataka i mnogi drugi
- Ovde će biti prikazano nekoliko specijalnih grafikona koji se mogu napraviti pomoću MATLAB-a
- Celokupna lista MATLAB-ovih funkcija za pravljenje grafikona i način njihove upotrebe, može se naći u prozoru sistema za pomoć (prvo odabrati odrednicu „Functions by Category“, zatim „Graphics“ pa „Basic Plots and Graphs“ ili „Specialized Plotting“)

Trakasti grafikoni

- Trakasti grafikon (vertikalni i horizontalni), stepenasti grafikon i grafikon diskretnih podataka, biće ilustrovani na podacima o prodaji iz ranije navedenog primera

<p>Vertikalni trakasti grafikon bar(x,y)</p>	 <table border="1"><thead><tr><th>Godina</th><th>Prodaja (u milionima)</th></tr></thead><tbody><tr><td>1988</td><td>8</td></tr><tr><td>1989</td><td>12</td></tr><tr><td>1990</td><td>20</td></tr><tr><td>1991</td><td>22</td></tr><tr><td>1992</td><td>18</td></tr><tr><td>1993</td><td>24</td></tr><tr><td>1994</td><td>27</td></tr></tbody></table>	Godina	Prodaja (u milionima)	1988	8	1989	12	1990	20	1991	22	1992	18	1993	24	1994	27	<pre>god=[1988:1994]; pro=[8 12 20 22 18 24 27]; bar(god,pro,'r') xlabel('Godina') ylabel('Prodaja (u milionima)')</pre>
Godina	Prodaja (u milionima)																	
1988	8																	
1989	12																	
1990	20																	
1991	22																	
1992	18																	
1993	24																	
1994	27																	
<p>Horizontalni trakasti grafikon barh(x,y)</p>	 <table border="1"><thead><tr><th>Godina</th><th>Prodaja (u milionima)</th></tr></thead><tbody><tr><td>1988</td><td>8</td></tr><tr><td>1989</td><td>12</td></tr><tr><td>1990</td><td>20</td></tr><tr><td>1991</td><td>22</td></tr><tr><td>1992</td><td>18</td></tr><tr><td>1993</td><td>24</td></tr><tr><td>1994</td><td>27</td></tr></tbody></table>	Godina	Prodaja (u milionima)	1988	8	1989	12	1990	20	1991	22	1992	18	1993	24	1994	27	<pre>god=[1988:1994]; pro=[8 12 20 22 18 24 27]; barh(god,pro) xlabel('Godina') ylabel('Prodaja (u milionima)')</pre>
Godina	Prodaja (u milionima)																	
1988	8																	
1989	12																	
1990	20																	
1991	22																	
1992	18																	
1993	24																	
1994	27																	

Stepenasti i diskretni grafikoni

<p>Stepenasti grafikon stairs(x,y)</p>		<pre>god=[1988:1994]; pro=[8 12 20 22 18 24 27]; stairs(god,pro) xlabel('Godina') ylabel('Prodaja (u milionima)')</pre>
<p>Grafikon diskretnih podatak stem(x,y)</p>		<pre>god=[1988:1994]; pro=[8 12 20 22 18 24 27]; stem(god,pro) xlabel('Godina') ylabel('Prodaja (u milionima)')</pre>

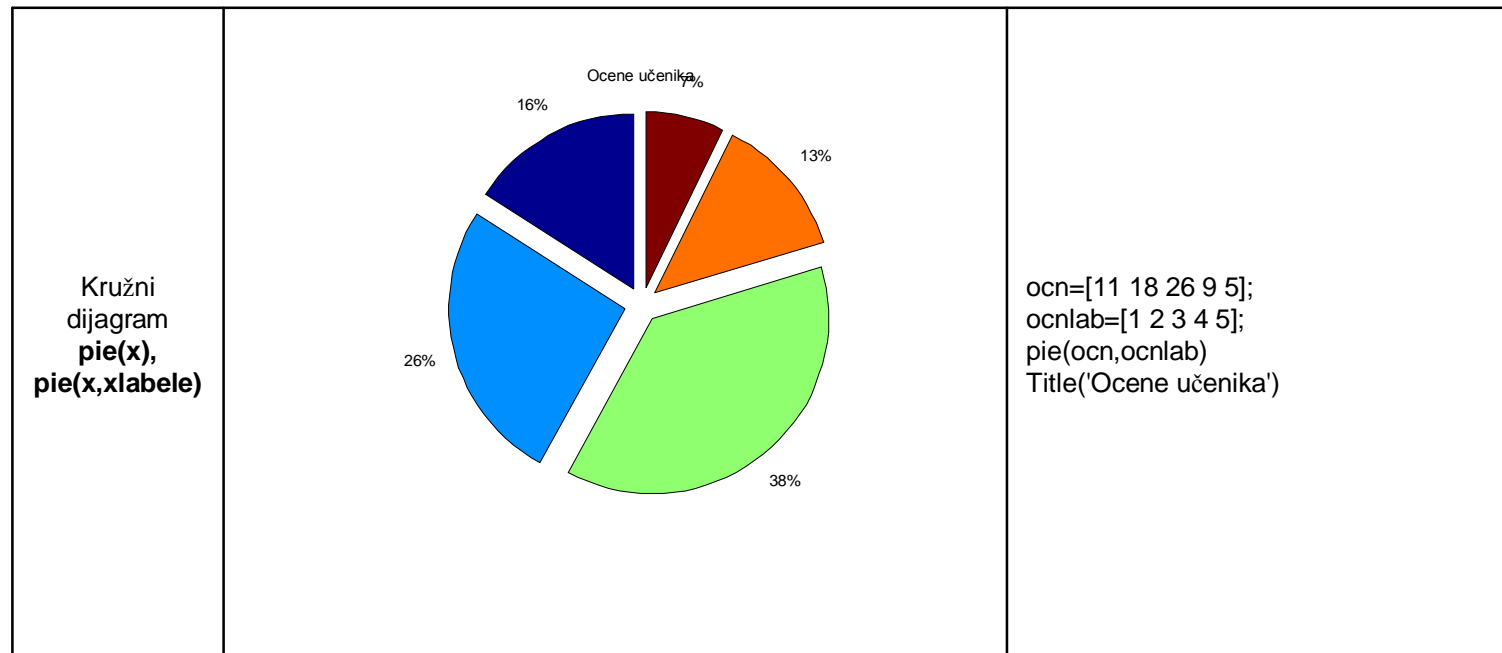


Kružni dijagrami

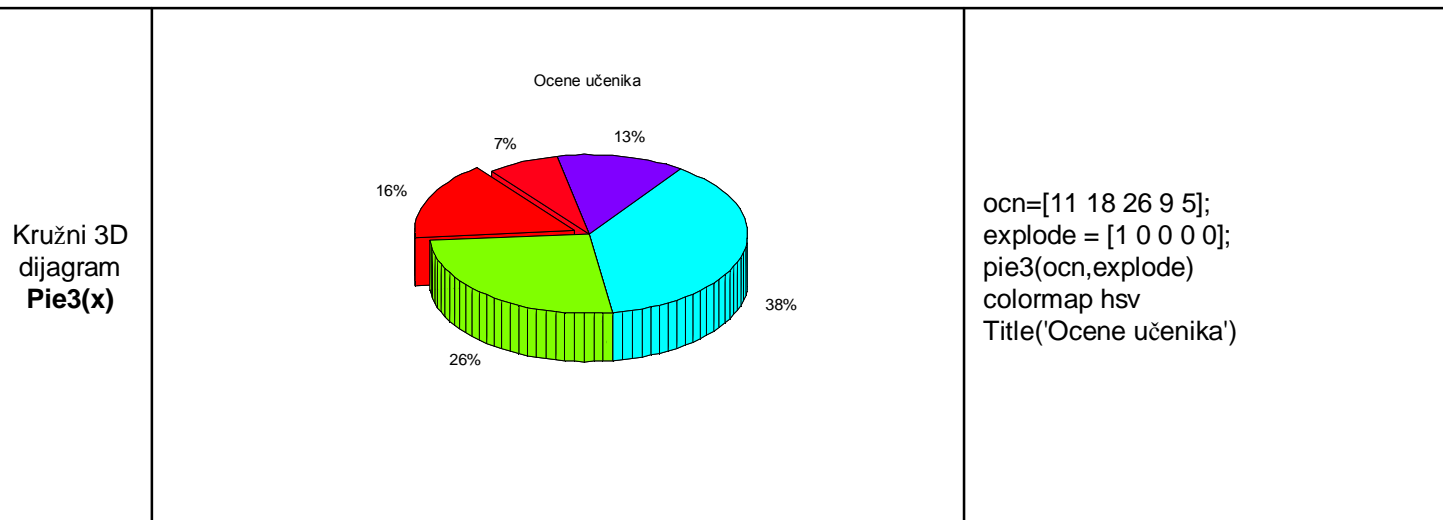
- Kružni dijagrami su podesni za predočavanje relativnih odnosa različitih ali srodnih veličina
- Na primer, u sledećoj tabeli date su ocene dodeljene određenom razredu
- Pomoću njih su napravljene kružni dijagram koji slede

Ocena	1	2	3	4	5
Broj učenika	11	18	26	9	5

Kružni 2D dijagram

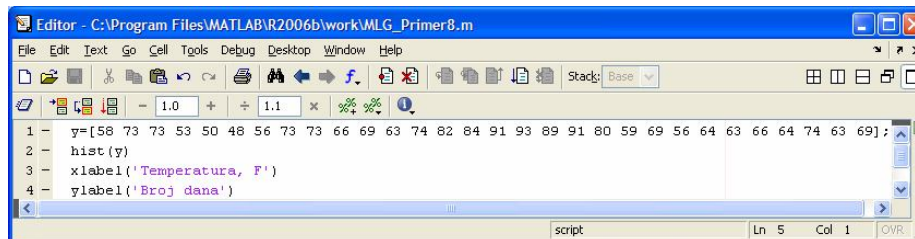


Kružni 3D dijagram



Histogrami

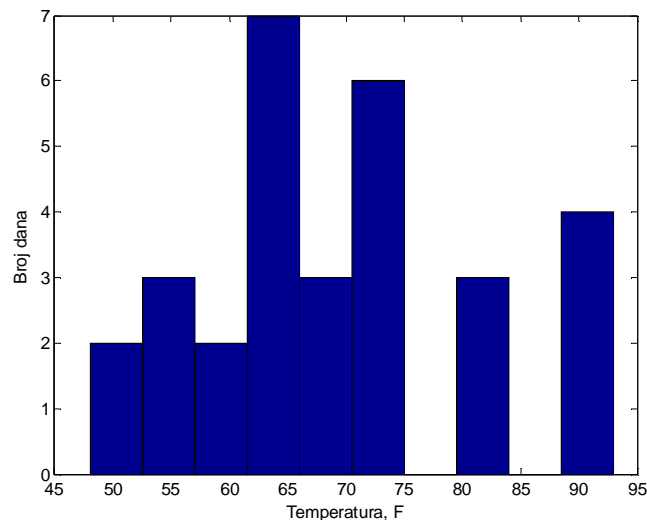
- Histogrami su grafikoni koji pokazuju raspodelu podataka
- Ukupan opseg datog skupa podataka deli se na podopsege, a histogram pokazuje broj tačaka u svakom podopsegu
- Histogram je vertikalni trakasti grafikon u kome su širine traka srazmerne širini odgovarajućeg podopsega, dok su visine traka srazmerne broju tačaka u podopsegu
- U MATLAB-u se histogrami prave komandom *hist*
- Najjednostavniji oblik komande je **hist(y)** – gde je y vektor podataka
- U ovom slučaju MATLAB deli opseg podataka na 10 jednakih podopsega i prikazuje broj tačaka u svakom podopsegu
- Na primer, ako su maksimalne dnevne temperature (u °F):
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, histogram tih podataka se dobija komandama sa slike



```
Editor - C:\Program Files\MATLAB\R2006b\work\WLG_Primer8.m
File Edit Text Go Cell Tools Debug Desktop Window Help
Stacks: Base
1 - y=[58 73 73 53 50 48 56 73 73 66 69 63 74 82 84 91 93 89 91
2 - hist(y)
3 - xlabel('Temperatura, F')
4 - ylabel('Broj dana')
```

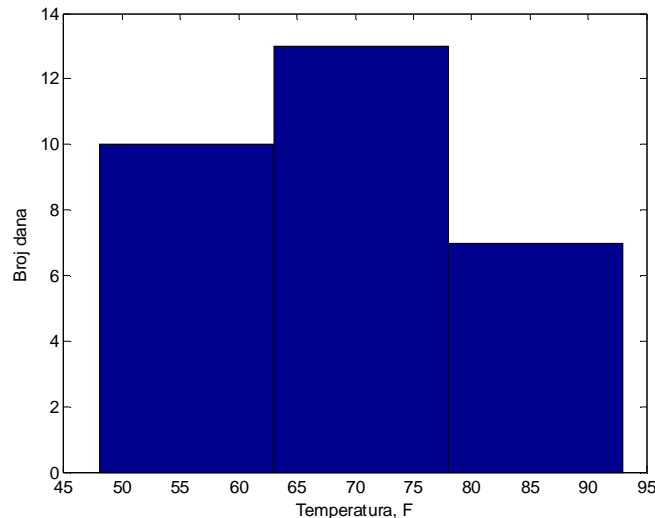
Primer histograma

- Generisani grafikon za skript datoteku sa prethodnog slajda prikazan na slici
- Najmanja vrednost skupa podataka je 48, a najveća 93, što znači da je opseg podataka 45, a širina podopsega 4.5
- Prvi podopseg se prostire od 48 do 52.5 i sadrži dve tačke
- Drugi podopseg se prostire od 52.5 do 57 i sadrži tri tačke itd.
- U dva podopsega (od 75 do 79.5 i od 84 do 88.5) uopšte nema tačaka



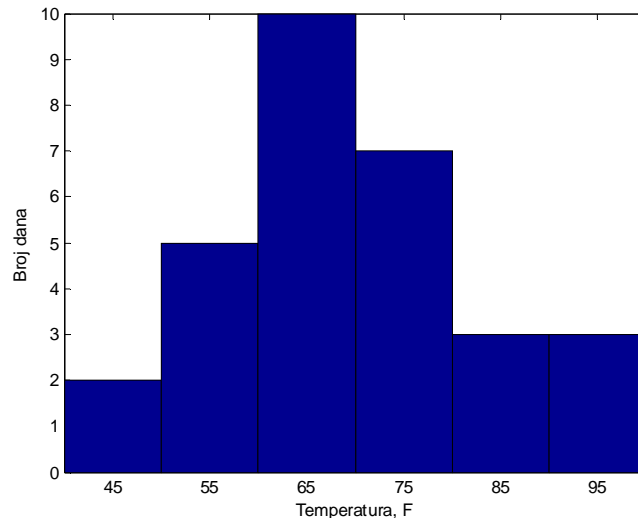
Zadavanje broja podopsega

- Pošto podela opsega podataka na 10 jednakih podopsega ne mora biti podesna sa stanovišta korisnika, broj podopsega može se zadati na dva načina.
- Jedan je da se zada broj podopsega čemu odgovaraju sledeći oblik komande *hist*:
hist(y,broj_podopsega)
gde je parametar 'broj_podopsega' skalar koji definiše broj podopsega
- Da bi se u prethodnom primeru opseg temperatura podelio na 3 podopsega, zadaje se komanda: **hist(y,3)** a rezultat je prikazan na slici



Zadavanje centara podopsega

- Drugi način je da se broj i širina podopsega zadaju pomoću vektora x čiji elementi definišu centre podopsega čemu odgovara sledeći oblik komande *hist*:
hist(y,x)
- MATLAB deli opseg na jednake podopsege, a x je vektor kojim se zadaje centralna tačka svakog podopsega (međusobna udaljenost centara ne mora biti jednaka) - granice podopsega su na sredini između njihovih centara
- Slika predstavlja histogram koji prikazuje prethodno navedene temperature raspoređene u 6 podopsega jednake širine od 10 stepeni
- Elementi vektora x za ovaj grafikon bili bi 45, 55, 65, 75, 85 i 95 pa je grafikon dobijen sledećim komandama:
 $x = [45:10:95]$; `hist(y,x)`



Prikazivanje numeričkih podataka

- Komanda *hist* se može upotrebiti sa opcijama koje osim histograma prikazuju i numeričke podatke
- Broj tačaka u svakom podopsegu prikazuju sledeće komande:
`n=hist(y)` **`n=hist(y,broj_podopsega)`** **`n=hist(y,x)`**
gde je *n* izlazni vektor
- Broj elemenata vektora *n* jednak je broju podopsega, a vrednost svakog elementa u *n* jednaka je broju tačaka (učestalosti pojavljivanja) u odgovarajućem podopsegu
- Na primer, histogram sa 10 opsega iz prethodnog primera se može napraviti komandom sa slike tako da vektor *n* pokazuje broj elemenata u svakom podopsegu, tj. da prvi podopseg ima 2 tačke, drugi 3 itd

```
Command Window
>> n=hist(y)

n =

     2     3     2     7     3     6     0     3     0     4

>>
```

Prikazivanje položaja podopsega

- Komandom *hist* se može prikazati i numerički položaj podopsega. To se postiže sledećim oblicima komande:
[n xout]=hist(y) [n xout]=hist(y,broj_podopsega)
gde je xout izlazni vektor u kojem je vrednost svakog elementa jednaka x koordinati centra odgovarajućeg podopsega
- Na primer, za prethodni histogram vektor xout na slici pokazuje da je centar prvog podopsega na 50.25, drugog na 54.75 itd.

```
Command Window
>> [n xout] = hist(y)
n =
     2     3     2     7     3     6     0     3     0     4
xout =
Columns 1 through 9
 50.2500  54.7500  59.2500  63.7500  68.2500  72.7500  77.2500  81.7500  86.2500
Column 10
 90.7500
>> |
```

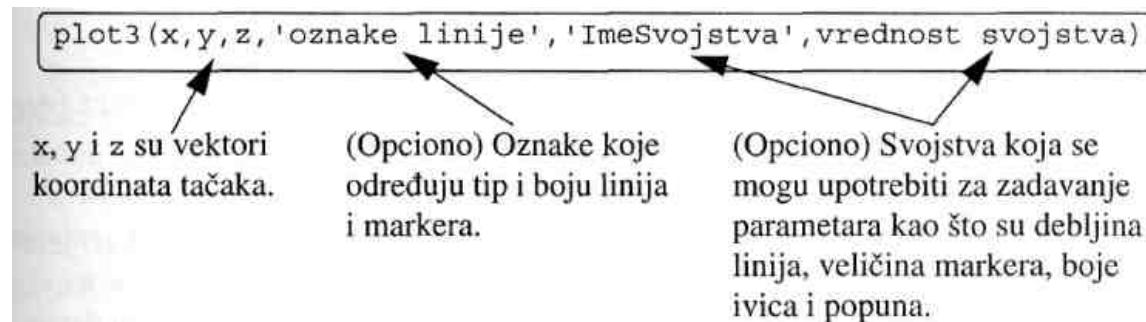


Trodimenzioni grafikoni

- Trodimenzionalni (3D) grafikoni mogu biti veoma korisni kada treba predstaviti podatke u kojima postoji više od dve promenljive
- U MATLAB-u postoji više mogućnosti za prikazivanje trodimenzionalnih grafikona
- To mogu biti linijski, žičani, površinski, mrežasti i mnogi drugi grafikoni
- Grafikoni se lako formatiraju tako da imaju specifičan izgled, a mogu im se dodati i specijalni efekt
- Ovde će biti opisane neke mogućnosti za trodimenzionalno predstavljanje podataka
- Dodatna objašnjenja mogu se naći u prozoru sistema za pomoć, pod odrednicom *Plotting and Data Visualization*

Linijski grafikoni

- Trodimenzionalni linijski grafikon (*lineplot*) jeste linija dobijena povezivanjem tačaka u trodimenzionalnom prostoru
- Osnovni 3-D grafikon crta se pomoću komande `plot3`, veoma slične komandi `plot`, koja ima sledeći oblik:

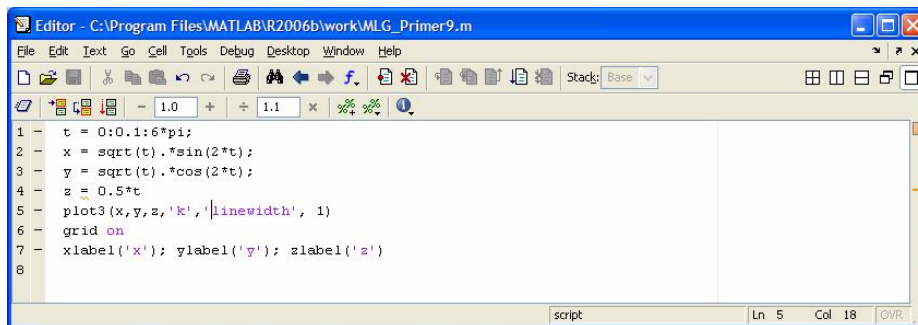


- Sva tri vektora sa koordinatama tačaka koje predstavljaju podatke (x , y i z) moraju imati jednak broj elemenata
- Oznake linija, svojstva i vrednosti svojstava isti su kao i za dvodimenzionalne grafikone

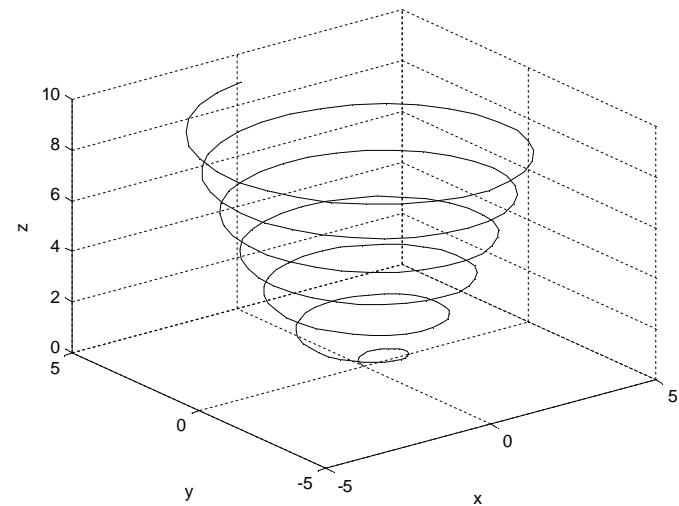
Primer

- Ako su koordinate x , y i z date u funkciji parametra t sledećim formulama: $x = \sqrt{t} \sin(2t)$, $y = \sqrt{t} \cos(2t)$, $z = 0.5t$

grafikon tačaka za $0 \leq t \leq 6$ može se nacrtati pomoću skript datoteke sa slike levo, a kada se ovaj program izvrši, dobiće se grafikon prikazan na slici desno



```
Editor - C:\Program Files\MATLAB\R2006b\work\MLG_Primer9.m
File Edit Text Go Cell Tools Debug Desktop Window Help
+ - 1.0 + 1.1 x
1 - t = 0:0.1:6*pi;
2 - x = sqrt(t).*sin(2*t);
3 - y = sqrt(t).*cos(2*t);
4 - z = 0.5*t
5 - plot3(x,y,z,'k','linewidth', 1)
6 - grid on
7 - xlabel('x'); ylabel('y'); zlabel('z')
8
script Ln 5 Col 18 OVR
```



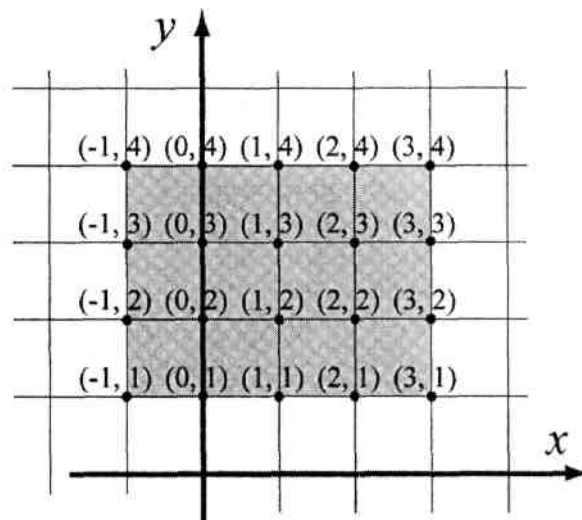


Mrežasti i površinski grafikoni

- Mrežasti grafikoni (*meshplots*) i površinski grafikoni (*surface plots*) su trodimenzionalni grafikoni koji omogućavaju predstavljanje funkcija tipa $z = f(x, y)$, gde su x i y nezavisne promenljive, a z je zavisna promenljiva
- To znači da se u datom domenu vrednost promenljive z može izračunati za svaku kombinaciju x i y
- Mrežasti i površinski grafikoni crtaju se u tri koraka
- Prvi korak je formiranje u ravni x - y rešetke (tj. 3-D koordinatnog sistema) koja pokriva domen (tj. oblast definisanosti) funkcije
- Drugi korak je izračunavanje vrednosti z u svakoj tački rešetke
- Treći korak je crtanje samog grafikona

Formiranje rešetke u ravni x-y:

- Rešetka (*grid*) predstavlja skup tačaka u ravni x-y koji obuhvata oblast definisanosti, tj. domen funkcije
- Gustinu domena (broj tačaka kojim se definiše domen) zadaje korisnik
- Na primer, slika prikazuje rešetku za domen - $1 \leq x \leq 3$ i $1 \leq y \leq 4$, s korakom 1





Formiranje matrica za rešetku

- U prethodno formiranoj rešetki, razmak između tačaka je jedna jedinica
- Tačke rešetke se mogu definisati pomoću dve matrice, X i Y
- Matrica X sadrži x koordinate svih tačaka rešetke, a matrica Y sadrži y koordinate:

$$X = \begin{bmatrix} -1 & 0 & 1 & 2 & 3 \\ -1 & 0 & 1 & 2 & 3 \\ -1 & 0 & 1 & 2 & 3 \\ -1 & 0 & 1 & 2 & 3 \end{bmatrix} \quad i \quad Y = \begin{bmatrix} 4 & 4 & 4 & 4 & 4 \\ 3 & 3 & 3 & 3 & 3 \\ 2 & 2 & 2 & 2 & 2 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$



Funkcija meshgrid

- Matrica X sadrži jednake vrste zato što u svakoj vrsti tačke imaju iste x koordinate, a matrica Y sadrži jednake kolone zato što u svakoj koloni tačke imaju iste y koordinate

- U MATLAB-u postoji ugrađena funkcija *meshgrid* koja omogućava formiranje matrica X i Y i koja ima sledeći oblik:

[X,Y] = meshgrid(x,y)

gde je X matrica x koordinata tačaka rešetke, a x je vektor koji deli domen pomenljive x, dok je Y je matrica y koordinata tačaka rešetke, a y je vektor koji deli domen pomenljive y

- U vektorima x i y, prvi i poslednji element su granice domena
- Gustinu rešetke određuje broj elemenata tih vektora

Primer

- Na primer, matrice X i Y koje odgovaraju rešetki iz prethodnog primer mogu se generisati pomoću komande *meshgrid* sa slike
- Pošto se matrice rešetke formiraju, one se koriste za izračunavanje vrednosti z u svakoj tački rešetke

```
Command Window
>> x=-1:3;
>> y=1:4;
>> [X,Y]=meshgrid(x,y)

X =

    -1     0     1     2     3
    -1     0     1     2     3
    -1     0     1     2     3
    -1     0     1     2     3

Y =

     1     1     1     1     1
     2     2     2     2     2
     3     3     3     3     3
     4     4     4     4     4

>> |
```



Izračunavanje vrednosti z u svakoj tački rešetke

- Vrednost z u svakoj tački rešetke izračunava se element po element, isto kao kada se radi s vektorima
- Kada su nezavisne promenljive x i y matrice (moraju biti jednakih dimenzija), izračunata zavisna promenljiva takode je matrica iste veličine
- Vrednost z na svakoj adresi izračunava se na osnovu odgovarajućih vrednosti x i y
- Nakon definisanja sve tri matrice, one se mogu upotrebiti za crtanje mrežastih ili površinskih grafikona

Primer

- Ako je z zadato formulom: $z = \frac{xy^2}{x^2 + y^2}$

vrednost z u svakoj tački rešetke koju smo prethodno definisali izračunava se kao na slici

```
Command Window
>> Z=X.*Y.^2./(X.^2+Y.^2)

Z =

-0.5000    0    0.5000    0.4000    0.3000
-0.8000    0    0.8000    1.0000    0.9231
-0.9000    0    0.9000    1.3846    1.5000
-0.9412    0    0.9412    1.6000    1.9200

>> |
```



Crtanje mrežastih ili površinskih grafikona

- Mrežasti ili površinski grafikon crta se pomoću komandi *mesh* ili *surf*, koje imaju sledeće oblike:

mesh(X,Y,Z)

surf (X,Y,Z)

gde su X, Y matrice koordinata rešetke, a Z matrica vrednosti zavisne promenljive z u tačkama rešetke

- Mrežasti grafikon se sastoji od linija koje povezuju tačke
- Na površinskom grafikonu, oblasti između linija mreže ispunjene su bojom

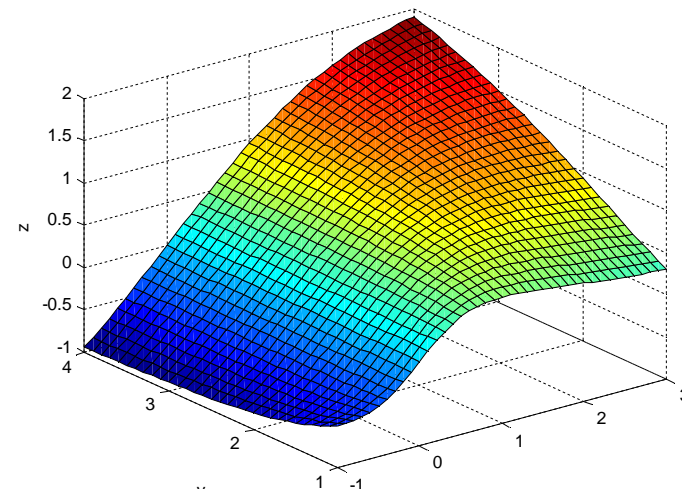
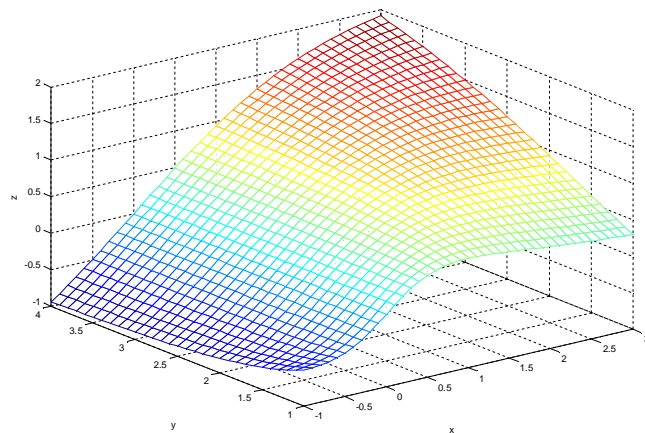
Primer

- Skript datoteka sa slike sadrži program koji formira rešetku, a zatim crta mrežasti (ili površinski) grafikon funkcije

$$z = \frac{xy^2}{x^2 + y^2} \text{ za } -1 \leq x \leq 3 \text{ i } 1 \leq y \leq 4$$

```
Editor - C:\Program Files\MATLAB\R2006b\work\MLG_Primer10.m
File Edit Text Go Cell Tools Debug Desktop Window Help
[Icons] - 1.0 + 1.1 x [Icons] Stack: Base
1 - x=-1:0.1:3;
2 - y= 1:0.1:4;
3 - [X,Y] = meshgrid(x,y);
4 - Z = X.*Y.^2./(X.^2 + Y.^2);
5 - mesh(X,Y,Z) % mrežasti grafikon
6 - % surf(X,Y,Z) % Za površinski grafikon upisati surf (X, Y, Z)
7 - xlabel('x'); ylabel('y'); zlabel('z')
8
```

- U ovom programu vektori x i y imaju relativno mali korak, a manji korak daje gušću rešetku - rezultati ovog programa su prikazani na slici





Dobijanje nepromenljive boje

- Boje na nacrtanim grafikonima menjaju se zavisno od vrednosti z
- Promene boja doprinose utisku da je grafikon trodimenzionalan
- Nepromenljiva boja se može dobiti na dva načina
- Prvi je kada se pomoću editora grafikona (*Plot Editor*) u grafičkom prozoru izabere strelica za uređivanje grafikona, pritisne se grafikon da bi se otvorio prozor za uređivanje svojstava - *Property Editor* - a zatim se izmeni boja pomoću liste *Mesh Properties*
- Drugi način je pomoću komande *colormap(C)* gde je C je vektor od tri elementa koji određuju intenzitete crvene, zelene i plave boje (RGB)
- Svaki element može biti broj između 0 (minimalan intenzitet) i 1 (maksimalan intenzitet)
- Često se koriste sledeće boje:

$C = [1 \ 0 \ 0]$ crvena	$C = [1 \ 1 \ 0]$ žuta	$C = [0 \ 1 \ 0]$ zelena
$C = [1 \ 0 \ 1]$ magenta	$C = [0 \ 0 \ 0]$ crna	$C = [0 \ 0 \ 1]$ plava
$C = [0.5 \ 0.5 \ 0.5]$ siva		

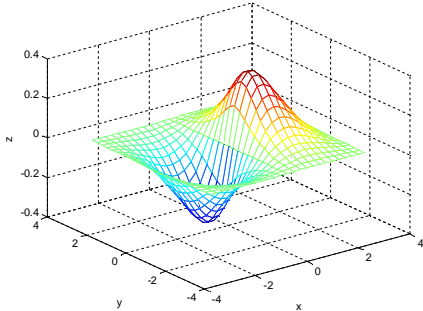
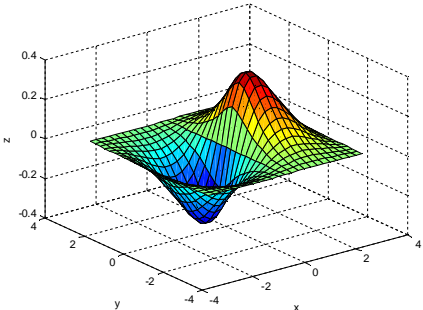


Dodatne komande

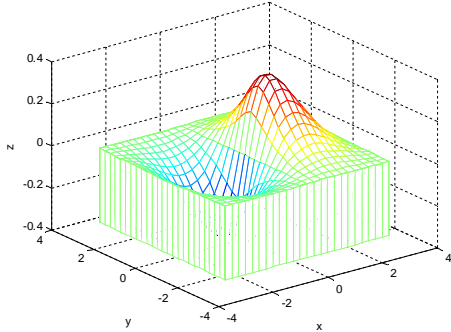
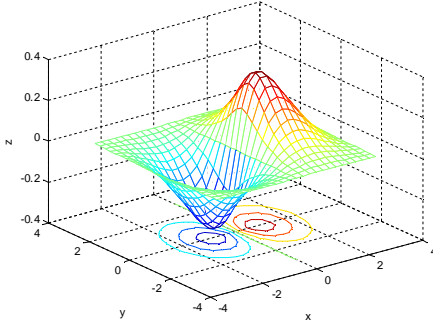
- Kada se da komanda *mesh*, prikazivanje linija rešetke se podrazumeva
- Prikazivanje linija rešetke može se isključiti pomoću komande *grid off*
- Pomoću komande *box* može se nacrtati okvir oko dijagrama
- Komande *mesh* i *surf* mogu se upotrebiti i u obliku *mesh (z)* i *surf (z)* u kom slučaju se vrednosti z crtaju kao funkcija njihovih adresa u matrici - broj vrste je na osi x, a broj kolone na osi y
- Postoji više dodatnih komandi za crtanje grafikona, sličnih komandama *mesh* i *surf*, koje omogućavaju crtanje grafikona drugačijih osobina
- Svi naredni primeri prikazuju grafikone vrednosti funkcije

$$z = 1.8^{-1.5\sqrt{x^2+y^2}} \sin(x)\cos(0.5y) \quad \text{za } -3 \leq x \leq 3 \text{ i } -3 \leq y \leq 3$$

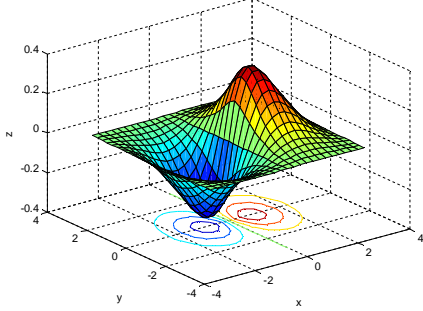
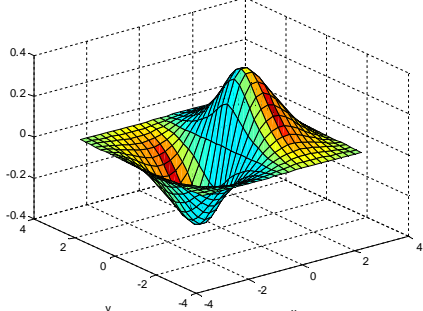


<p>Mrežasti mesh(X,Y,Z)</p>		<pre>x=-3:0.25:3; y= -3:0.25:3; [X,Y] = meshgrid(x,y); Z = 1.8.^(-1.5*sqrt(X.^2+Y.^2)) .*cos(0.5*Y).*sin(X); mesh(X,Y,Z) xlabel('x'); ylabel('y'); zlabel('z')</pre>
<p>Površinski surf(X,Y,Z)</p>		<pre>x=-3:0.25:3; y= -3:0.25:3; [X,Y] = meshgrid(x,y); Z = 1.8.^(-1.5*sqrt(X.^2+Y.^2)) .*cos(0.5*Y).*sin(X); surf(X,Y,Z) xlabel('x'); ylabel('y'); zlabel('z')</pre>

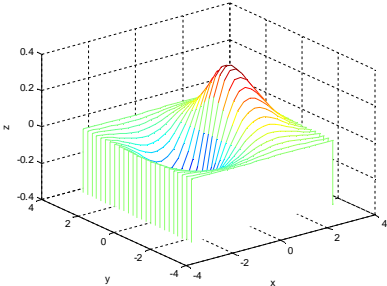
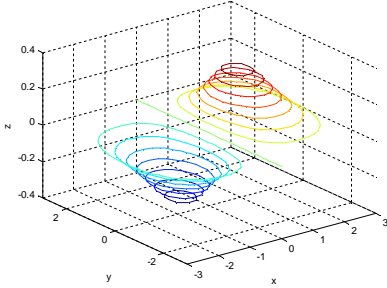
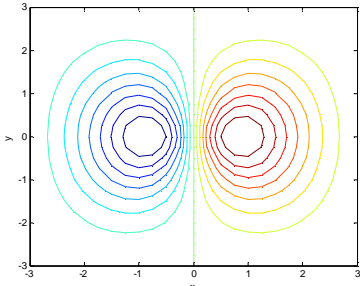


<p>Mrežasti sa zavesom meshz(X,Y,Z)</p>		<pre>x=-3:0.25:3; y= -3:0.25:3; [X,Y] = meshgrid(x,y); Z = 1.8.^(-1.5*sqrt(X.^2+Y.^2)) .*cos(0.5*Y).*sin(X); meshz(X,Y,Z) xlabel('x'); ylabel('y'); zlabel('z')</pre>
<p>Mrežasti sa konturom meshc(X,Y,Z)</p>		<pre>x=-3:0.25:3; y= -3:0.25:3; [X,Y] = meshgrid(x,y); Z = 1.8.^(-1.5*sqrt(X.^2+Y.^2)) .*cos(0.5*Y).*sin(X); meshc(X,Y,Z) xlabel('x'); ylabel('y'); zlabel('z')</pre>



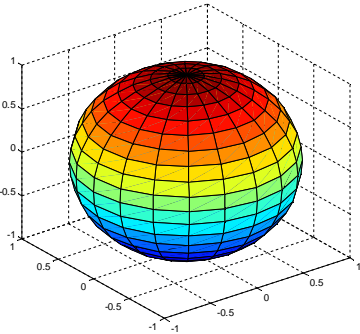
<p>Površinski sa konturom surf(X,Y,Z)</p>		<pre>x=-3:0.25:3; y= -3:0.25:3; [X,Y] = meshgrid(x,y); Z = 1.8.^(-1.5*sqrt(X.^2+Y.^2)) .*cos(0.5*Y).*sin(X); surf(X,Y,Z) xlabel('x'); ylabel('y'); zlabel('z')</pre>
<p>Površinski sa konturom surf(X,Y,Z)</p>		<pre>x=-3:0.25:3; y= -3:0.25:3; [X,Y] = meshgrid(x,y); Z = 1.8.^(-1.5*sqrt(X.^2+Y.^2)) .*cos(0.5*Y).*sin(X); surf(X,Y,Z) xlabel('x'); ylabel('y'); zlabel('z')</pre>



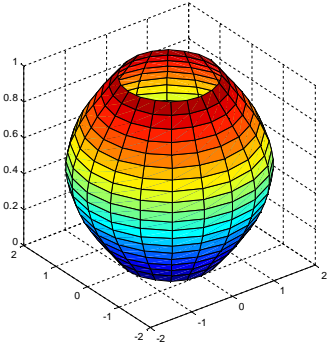
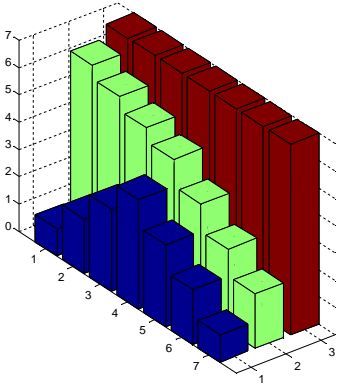
<p>Kaskadni waterfall(X,Y,Z)</p>		<pre>x=-3:0.25:3; y= -3:0.25:3; [X,Y] = meshgrid(x,y); Z = 1.8.^(-1.5*sqrt(X.^2+Y.^2)) .*cos(0.5*Y).*sin(X); waterfall(X,Y,Z) xlabel('x'); ylabel('y'); zlabel('z')</pre>
<p>3D konturni Contour3(X,Y,Z,n)</p>		<pre>x=-3:0.25:3; y= -3:0.25:3; [X,Y] = meshgrid(x,y); Z = 1.8.^(-1.5*sqrt(X.^2+Y.^2)) .*cos(0.5*Y).*sin(X); contour3(X,Y,Z,15) xlabel('x'); ylabel('y'); zlabel('z')</pre>
<p>2D konturni contour(X,Y,Z,n)</p>		<pre>x=-3:0.25:3; y= -3:0.25:3; [X,Y] = meshgrid(x,y); Z = 1.8.^(-1.5*sqrt(X.^2+Y.^2)) .*cos(0.5*Y).*sin(X); meshz(X,Y,Z) xlabel('x'); ylabel('y'); zlabel('z')</pre>

Specijalni trodimenzionalni grafikoni

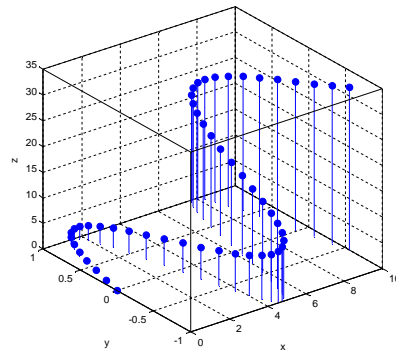
- U MATLAB-u postoje funkcije za izradu raznih vrsta trodimenzionalnih grafikona sa specijalnim efektima
- Lista svih mogućnosti nalazi se u prozoru sistema za pomoć (Help), pod odrednicom *Plotting and Data Visualization*

<p>Crtanje jedinične sfere (sa n površina) sphere, sphere(n)</p>		<pre>[X,Y,Z]=sphere(20); surf(X,Y,Z)</pre>
----------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------	------------------------------------------------



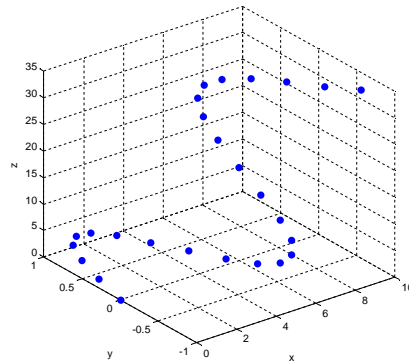
<p>Crtanje cilindra cylinder(r)</p>		<pre>t=linspace(0,pi,20); r=1+sin(t); [X,Y,Z]=cylinder(r); surf(X,Y,Z) axis square</pre>
<p>3D stubičasti Bar3(Y)</p>		<pre>Y=[1 6.5 7; 2 6 7; 3 5.5 7; 4 5 7; 3 4 7; 2 3 7; 1 2 7]; bar3(Y)</pre>

3D grafikon
diskretnih
podataka
stem3(X,Y,Z)



```
t=0:0.2:10;  
x=t;  
y=sin(t);  
z=t.^1.5;  
stem3(x,y,z,'fill')  
grid on  
xlabel('x'); ylabel('y'); zlabel('z')
```

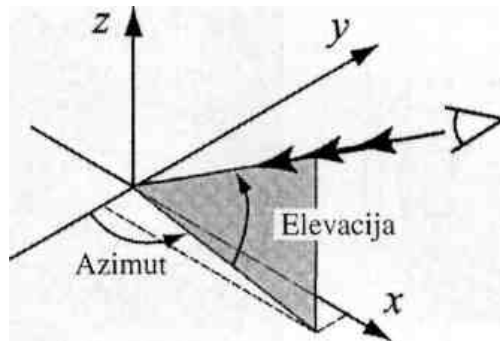
3D tačkasti
Scatter3(X,Y,Z)



```
t=0:0.4:10;  
x=t;  
y=sin(t);  
z=t.^1.5;  
scatter3(x,y,z,'fill')  
grid on  
colormap([0.1 0.1 0.1])  
xlabel('x'); ylabel('y'); zlabel('z')
```

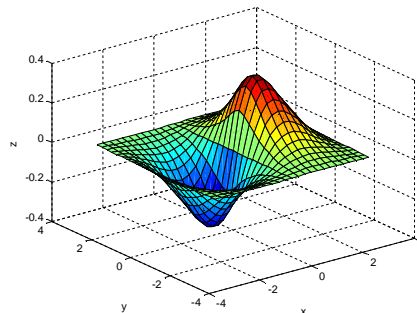
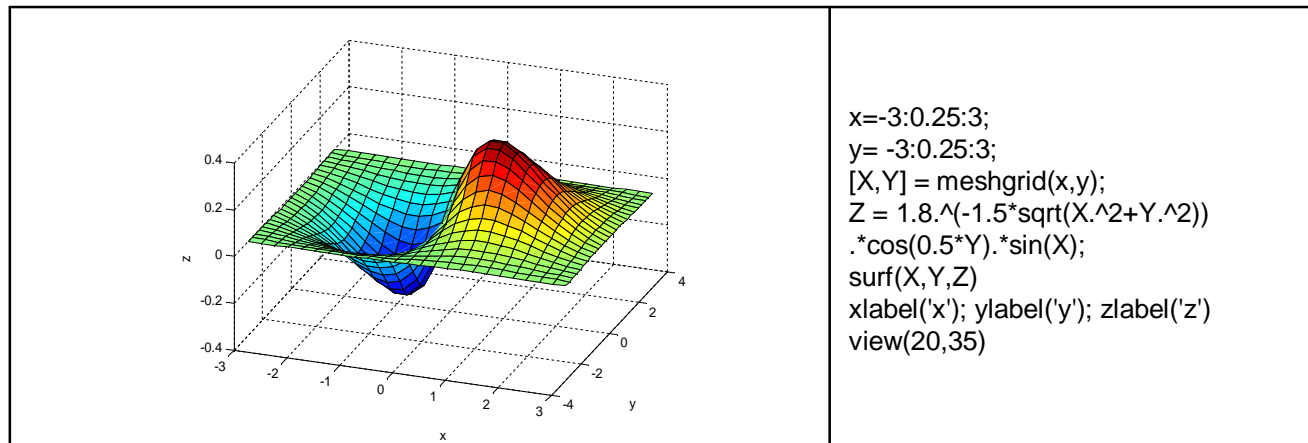
Komanda view

- Komanda *view* određuje pravac posmatranja grafikona
- Pravac posmatranja se zadaje pomoću uglova azimuta i elevacije (slika) ili u obliku koordinata tačke u prostoru iz koje se grafikon posmatra
- Ugao pod kojim se grafikon posmatra zadaje se komandom *view* na sledeći način:
 $view(az, el)$
gde je *az* ugao u odnosu na negativan smer y ose a *el* je ugao elevacije (izražen u stepenima) u odnosu na ravan x-y
- Pozitivna vrednost odgovara uglu otvorenom u pravcu z ose



Primer

- Podrazumevani uglovi posmatranja grafikona jesu $az = -37,5^\circ$ i $el = 30^\circ$.
- Na slici je površinski grafikon iz prethodnog primera (mala slika na dnu) nacrtan ponovo ali su uglovi posmatranja $az = 20^\circ$ i $el = 35^\circ$





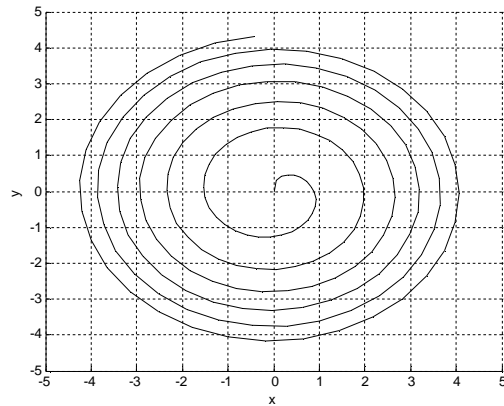
Projekcije na ravni

- Kada se izaberu odgovarajući uglovi azimuta i elevacije, komanda *view* se može upotrebiti za crtanje projekcija 3-D grafikona na različite ravni, prema sledećoj tabeli

Ravan na koiu se projektuje	Ugao <i>az</i>	Ugao <i>el</i>
<i>x-y</i> (pogled odozgo)	0	90
<i>X-z</i> (bočni pogled)	0	0
<i>y-z</i> (bočni pogled)	90	0

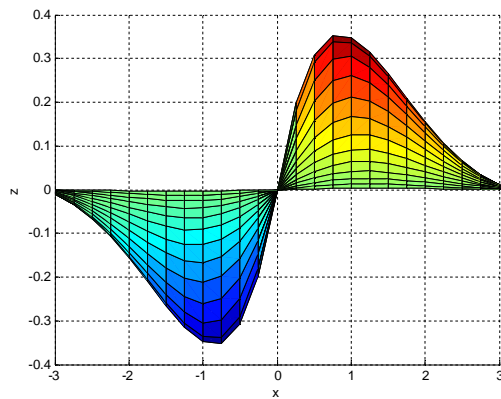
- Pogled odozgo na grafik funkcije

$$x = \sqrt{t} \sin(2t), y = \sqrt{t} \cos(2t), z = 0.5t$$

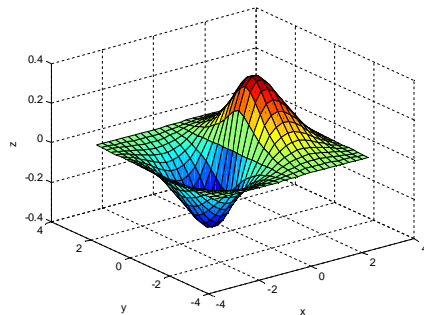


```
t = 0:0.1:6*pi;  
x = sqrt(t).*sin(2*t);  
y = sqrt(t).*cos(2*t);  
z = 0.5*t  
plot3(x,y,z,'k','linewidth', 1)  
view(0,90)  
grid on  
xlabel('x'); ylabel('y'); zlabel('z')
```

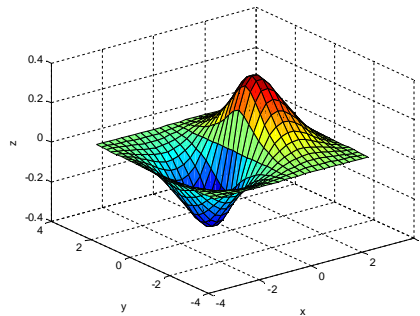
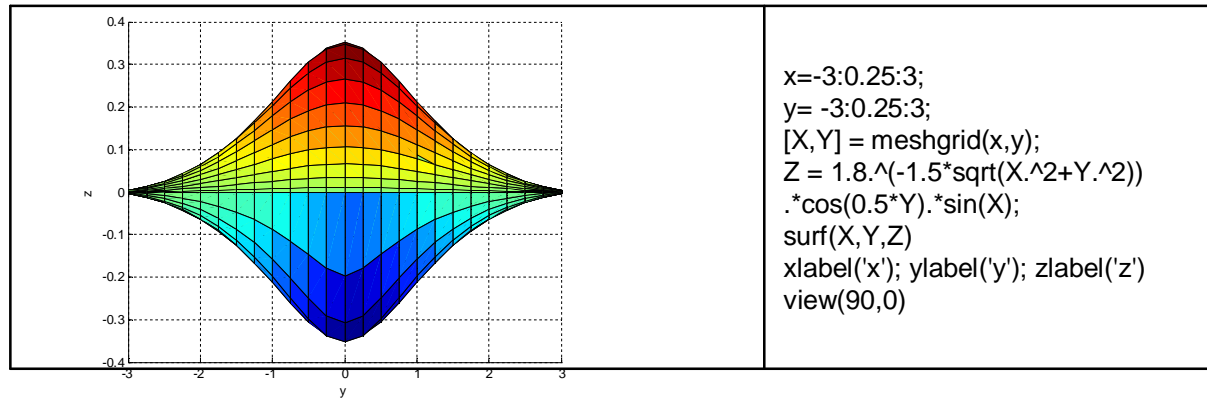
- Bočni izgled funkcije sa slike na dnu



```
x=-3:0.25:3;  
y= -3:0.25:3;  
[X,Y] = meshgrid(x,y);  
Z = 1.8.^(-1.5*sqrt(X.^2+Y.^2))  
.*cos(0.5*Y).*sin(X);  
surf(X,Y,Z)  
xlabel('x'); ylabel('y'); zlabel('z')  
view(0,0)
```



- Projekcija funkcije sa slike na dnu na ravan x-z





Pogled iz tačke

- Pravac posmatranja može se zadati i tako što se zada tačka u prostoru iz koje se posmatra grafikon
- U tom slučaju, komanda view ima oblik $view([x,y,z])$, gde su x , y i z koordinate tačke
- Pravac je onaj koji definišu zadata tačku i početna tačka koordinatnog sistema i ne zavisi od rastojanja tih tačaka
- To znači da je pogled isti iz tačke $[6,6,6]$ kao i iz tačke $[10,10,10]$
- Pogled odozgo može se zadati koordinatama $[0,0,1]$
- Bočni pogled ravni x - z iz pravca negativnog dela y ose može se zadati koordinatama $[0,-1,0]$ itd



Samostalan rad

- Tekući položaj čestice u pokretu dat je u funkciji vremena sledećim jednačinama:
 $x = (2 + 4\cos(t))\cos(t)$
 $y = (2 + 4\cos(t))\sin(t)$
 $z = t^2$
Nacrtati grafikon položaja čestice za $0 \leq t \leq 20$
- Spiralno stepenište se može modelovati pomoću sledećih parametarskih jednačina:

$$x = R \cos\left(2\pi n \frac{t}{h}\right)$$

$$y = R \sin\left(2\pi n \frac{t}{h}\right)$$

$$z = \frac{t}{h}$$

gde je R poluprečnik stepeništa, h je razmak između spratova, a n je broj krugova koje stepenište pravi na svakom spratu.

Zgrada ima dva sprata visine $h = 3$ m.

Nacrtati grafikon za dva primera stepeništa: jedan za $R = 1.5$ m i $n = 3$, a drugi za $R = 4$ m i $n = 2$.

Opcija: neka oba stepeništa budu na istom grafikonu.